technische universität dortmund

IT in Produktion und Logistik

# Knowledge Discovery in Supply Chain Transaction Data by Applying Data Farming

Master Thesis to obtain the Academic Degree, Master of Science in Logistics


Submitted by : Wenzheng Su

Matriculation.-Nr : 164293



Issued on : 09. NOV. 2015

Submitted on : 17. MAI. 2016



Mentors & Examined by : Univ.-Prof. Dr.-Ing. Markus Rabe

Dipl.-Inf. Anne Antonia Scheidler

TO MY PARENTS

献给我的父亲母亲

# Contents

# List of Figures

# List of Tables

# List of Formulae

# List of Abbreviations

| | |
|---|---|
| **1TS** | 1st Tier Supplier |
| **1TSs** | 1st Tier Suppliers |
| **2TSs** | 2nd Tier Suppliers |
| **3PLs** | 3rd Party Logistics Providers |
| **3TSs** | 3rd Tier Suppliers |
| **ANOVA** | ANalysis Of VAriance |
| **B2B** | Business to Business |
| **DES** | Discrete Event Simulation |
| **EDI** | Electronic Data Interchange |
| **EM Clustering** | Expectation Maximization Clustering |
| **IT** | Information Technology |
| **JIS** | Just in Sequence |
| **KDD** | Knowledge Discovery in Databases |
| **KPIs** | Key Performance Indicators |
| **SC** | Supply Chain |
| **SCs** | Supply Chains |
| **SCM** | Supply Chain Management |
| **SCMSs** | Supply Chain Management Systems |
| **SCOR** | Supply Chain Operation Reference |
| **VM** | Vehicle Manufacturer |
| **VMI** | Vendor Managed Inventory |
| **V&V** | Verification and Validation |

# 1 Introduction

With the growing popularity of the Internet and E-Commerce, all the parties of Supply Chain (SC) are on the solid cooperation in information sharing, in order to achieve higher customer satisfaction and lower cost. However, integrating the data and processes among the partners might cause the information quality issues, which will influence the operational process performance significantly. (Huang and Hu 2004). On the other hand, in the global dynamic SC every member is forced to keep up with rapid changes in demand, for which the information quality is required so that the SC processes disturbance can be avoided and inventory costs can be saved (Wu and Olson 2008). Therefore, many approaches contributed to decision support and discovery on the transaction data in Supply Chain Management Systems (SCMSs) (Huang and Hu 2004).

However, Düsing (2010) points out that there are still challenges with database issues e.g. sample size, existing data format and data quality in details. Furthermore, verification and validation (V&V) accompanying processes does not belong to knowledge discovery in database model. For bridging this technical gap, Rabe and Scheidler (2015) propose data farming. Data farming is a methodology and capability that makes use of high performance computing to run models many times in order to generate data as sufficiently for the statistical investigations (Horne and Meyer 2005; Nato Report 2014). With respects to the theories and current technical state aforementioned, application and development of data farming in the SCM could be one of the solutions for generating transaction data with the required quality and sufficient sample size for knowledge discovery in global changing SC landscapes.

This thesis aims to develop a conceptual approach to knowledge discovery in supply chain transaction data by applying data farming. For demonstrating this, three tasks need to be accomplished. Firstly, the existing SC simulation model for order delivery processes between the 1st Tier Suppliers (1TSs) and a Vehicle Manufacturer (VM), strictly called as Original Equipment Manufacturer (OEM), will be expended. In this simulation model the new additional and existing parametric variations are going to be attributed to the individual processes action rules in order to increase the complexity and accuracy of data attributes as precisely as the real SC landscapes observations imply. Secondly, the generated data will be transformed in an adequate format for data analysis algorithms. At the last step, the transformed data are going to be analyzed with different clustering algorithms by using RapidMiner. To test and rank their results, the appropriate statistical methods will be utilized and help drawn a conclusion if order processes behave in a regular or disturbed manner.

This thesis follows a logical path through the major areas in SCM, knowledge discovery and data farming. Constructing this thesis begins with a theoretical overview of SCM, especially with the focus on the current trends and challenges in the automotive industry. This triggers the discussion topic of “match the demand” which derives from the material requirement forecasting and builds a foundation of supplier delivery call-off plans. These will be implemented in terms of the SC strategy, tactical and operational stages in order to include the possible processes disturbance causes, which could be reflected and recorded in SCMSs, so the specific application of the information technology in SCM needs to be envisaged. Then, it comes to a general approach to the performance measurement in order to indicate whether the processes are economically

successful or dominate the relatively competitive advantages in the sector. Subsequently, the knowledge discovery in databases (KDD) background of disciplines and various data mining techniques applying in SCM as well as the results evaluation methods are summed up for data analysis. For the data analysis under the framework of this thesis, the clustering algorithms, including the clustering algorithms art, characters as well as the statistical tests will be presented.

The next topic to discuss is how to generate the SC transaction data by using discrete event simulation. The key to answering this question lies in a general methodology of SC transaction data and data farming, which is essentially carried out by discrete event simulation. Thus, it is necessary to present an overview of the simulation procedure and tools for implementing simulation model and output data analysis.

Afterwards, it will implement the tasks of this thesis with regards to the aforementioned theoretical backgrounds and technical state. The simulation model which depicts the 1TS delivery processes will be expanded firstly. After a certain simulation runs, the processes transaction data can be generated. Following, it will present how to analyze these farmed data by using clustering algorithms in order to classify which order delivery processes are regular and which are disturbed.

At last, it will overview the implementation procedure of this work, discuss the conclusion and recommend the potential research works on this thesis.

# 2 Knowledge Discovery in Supply Chains

In this chapter the theoretical approach to knowledge discovery in SCs will be presented. **Section 2.1** will characterize the SCM in the automotive industry in terms of the supplier structure, delivery strategy, IT structure and key performance indicators. **Section 2.2** will describe the conceptual framework of knowledge discovery and one of the data mining methods, clustering, will be discussed in **section 2.3**.

## 2.1 Supply Chain Management in the Automotive Industry

This section concerns the theoretical background and current trends of SCM, especially in terms of 1TS order delivery processes. It will discuss the possible causes that could lead to processes disturbance. In order to obtain an overview of the visual SC landscapes, a general introduction about the IT application in SCs will be provided. Process performance measurement expresses abstract SCs objectives in competitive advantages and indicates the development potentials, so Key Performance Indicators (KPIs) will be introduced.

### 2.1.1 Theorical Background and Current Trends

SCM and logistics are often expressed in a synonymous way. However, Lysons and Farrington (2012) state that the application of logistics is essential to the efficient management of the SC. Furthermore, Christopher (2011) points out that there are some synonymous terms of SCM from the different viewpoints. *Demand chain management* reflects the fact that the chain should be driven by the market. *Supply chain network* implies the expanding SC structure with the multiple suppliers and customers. *Value chain* or *value-added chain* emphasizes the product or service differentiation with respect to the competitive advantages. In German industry, especially in the automotive industry, SCM is replaced by *value networks*, which explains the complexity of SCs and intensive market competition (Schulz et al. 2013). SCs encompass a number of key flows: physical flows of materials, flows of information, and flows of resources for supporting SC operation processes like finance, human resource and manufacturing equipments (Mangan et al. 2008). For matching the idea of this thesis, the following definition of SCM is adopted and described in **Figure 2.1.1** in terms of the current state in the automotive industry.

*"Supply chain management is the management across a network of upstream and downstream organizations of material, information and resource flows that lead to the creation of value in the form of products and/or services*." (Mangan et al. 2008).

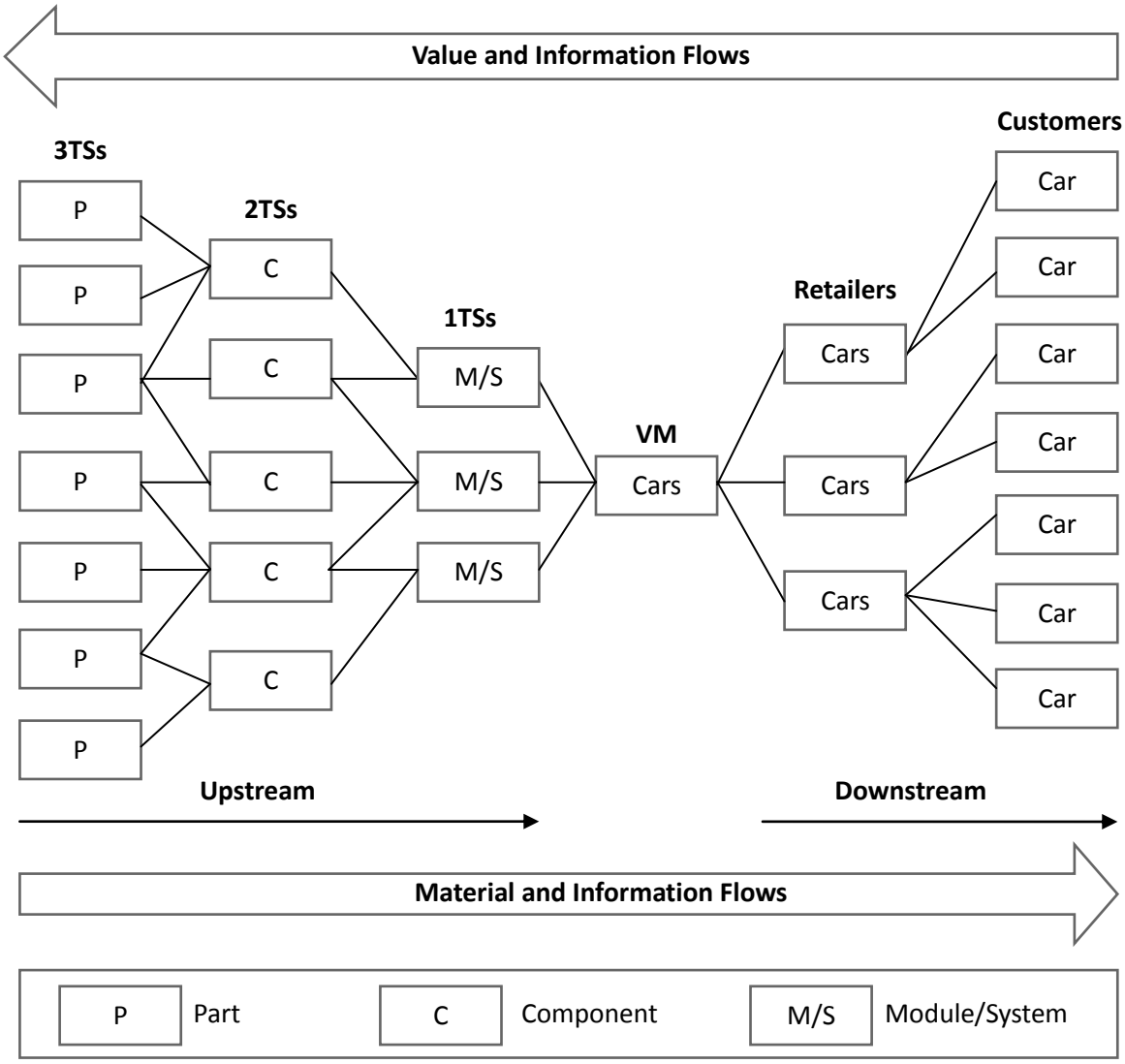
Träger el al. (2013) summarize five essential aims of SCM in the automotive industry for obtaining the competitive advantages:

1. Increasing customer's satisfaction with performance value or benefits, i.e. ecological product profile, individual confabulated car, valued-added services;
2. Reducing the cost of transport and inventory by operating the logistics processes effectively;

3. Reducing the order cycle time by quick response ( sale of market);
4. Delivery quality in terms of avoiding the retouring processes costs and lose customers;
5. Pursuing the high flexibility for keeping the management systems adaptable to the changed demands.

In addition to these points, Grunewald (2015) states that achieving the ecological aim is the precondition of achieving the other aims with respects to environmental protection. It is to observe that each of the aims has a correlative impact on the others (Christopher 2011) and impossible to avoid the conflict among them (Grunewald 2015).

**Figure 2.1.1. Supply Chain Network in the Automotive Industry**




Based on: Grunewald (2015), p. 11; Lysons and Farrington (2012), p. 101; Mangan et al. (2008)

From the viewpoint of material flow, a supply chain network as shown in **Figure 2.1.1**, the horizontal structure represents the outbound logistics processes among the downstream and upstream SC parties which are classified into tier levels, so that the influence of each tier can be indicated with the resource collaboration and processes optimization. The vertical structure reflects the independent inbound logistics processes inside the plants or warehouses of each SC partner, ex-

cept for the customer (Lysons and Farrington 2012). The 3rd Party Logistics Providers (3PLs) and the other outsourcing firms can be involved directly or indirectly. The 3LPs provide a range of logistics services, e.g. transportation and warehousing, vendor managed Inventory (VMI), haulage contractor and consignment warehouse (Lysons and Farrington 2012; Göpfert and Braun 2013). The SCM tasks model, most taken in German publications, encompasses three management fields: supply chain strategy, supply chain planning and supply chain operation. *Supply chain strategy* concerns network design in terms of configuration of products and production processes, therefore, supply chain strategy has been replaced by *supply chain configuration* in German automotive industry (Träger el al. 2013). The management objects are long-term periodical including the sale and distribution planning, site selection as well as defining the cooperative relationship with suppliers. *Supply chain planning* focuses the middle-term, also understood as tactical objects. The main issue is to set up the master plan which describes the demand forecasting, material requirement planning, production program planning and production planning. The primary mission of *supply chain execution* is to trace and track the status of procurement, manufacture and distribution. It refers to the short-term operational activities such as order fulfillment which involves process controlling of production, manufacturing, material inventory as well as transportation. This field means a determined point to gain competitive advantages with regard to the flexibility level for adopting the changed demand. Günther el al. (2005) add *supply chain event management* as a monitoring all of the operational activity performance, particularly the quality of service level, including analysis of logistics costs and inventory.

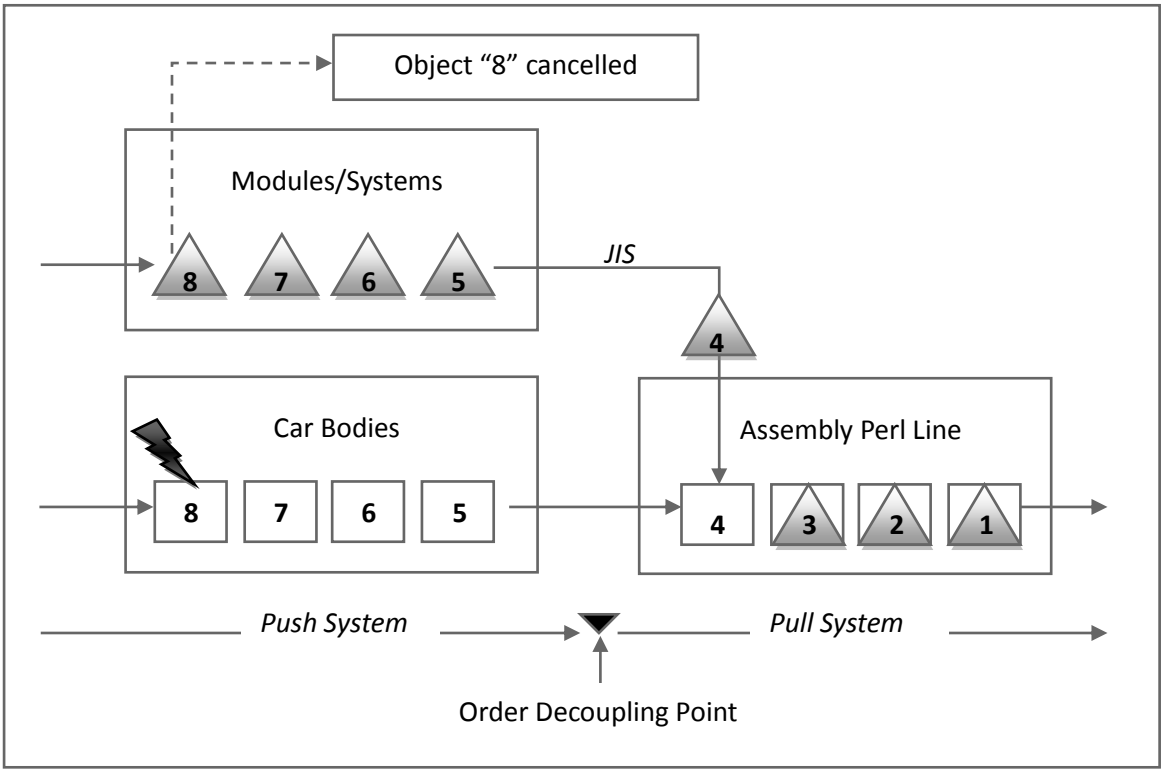
Because of the high-leveled individual product configuration and costs pressure, the SCs upstream structure has been becoming *lean*. However, it leads to high complexity of product configuration and increasing the *order decoupling points* (Alicke 2005; Lysons and Farrington 2012). The disadvantages of this new form is that the entire SCs are not possible fully or well-integrated from the multidimensional perspectives of the overall corporate strategy such as Know-How protection, partnership trust. Firstly, the delivery quality issues maybe occur, because the SCs details about the product specifications are blur to the 2TSs. Secondly, the delivery maybe delay or be not possible at all, if the 2TSs and the 3PLs are not integrated in the VM's SCMSs or they don't have the high manufacture flexibility to response and fulfill the changed orders in the very short time. Thirdly, the 2TSs have heterogeneous SCM software application or the proprietary access to the SCMSs partly. This could cause *bullwhip effect*, because of the lack of information transparency about the material requirement on the side of VMs. (Göpfert and Braun 2013; Schweppe 2008; Gehr and Hellingrath 2007)

### 2.1.2 1st Tier Suppliers Delivery Processes

The partnership between 1TSs and a VM is actually regarded more than customer-supplier-relationship, rather a long-term contractual alliance that is consolidated over time. 1TSs arranges the manufacture processes as an independent assembly shop for configuring the systems or modules (**Figure 2.1.1**) which are delivered to the assembly shop at VM *just in sequence* (JIS) in terms of *assembly peal line* (Klug 2013; Heinecke et al 2012). The fundamental approach of SCM is to match the demand so that the VM can deliver the car to the customer in time and avoid *bullwhip effect*, in which the small demand fluctuations caused by information flowing upstream and downstream in the SC become high variability swings at production stage. 1TSs receive the hardest impact of the bullwhip effect (Lysons and Farrington 2012; Klug 2013). In practice, 1TSs of a

VM are distinguished between internal 1TSs and external 1TSs. Manufacture shops of the internal 1TSs are the press shop, body shop and print shop, and other aggregate shops which produce the modules with the competitive corn technology as motors (Grunewald 2015). The internal 1TSs normally locate on the VM's plant the same as the assembly shop, with the sorter buffers between them (Grunewald 2015; Klug 2013). The external 1TSs deliver the systems or modules. In this thesis, the 1TSs represent the external 1TSs. Because the customer drives the demand, the assembly mechanism between 1TS and VM follows *push-pull strategy* (Klug 2013). A *push strategy* is when products are manufactured in anticipation of demand and production is based on long-term forecasts, namely built to forecast or made to stock, and associated with high inventory levels, high manufacturing mass and the transportation costs, due to the quick response to demand changes. A *pull strategy* is when products are manufactured to specific orders rather than forecasts, also called *built to order* or *assembly to order.* Thus, demand is certain and inventory is low or non-existent. A pull strategy needs that the exact information about customer demand is quickly transmitted to the various SC participants, so that the bullwhip effect is avoided. (Lysons and Farrington 2012; Klug 2013)

**Figure 2.1.2. Assembly Process with Push-Pull Strategy**



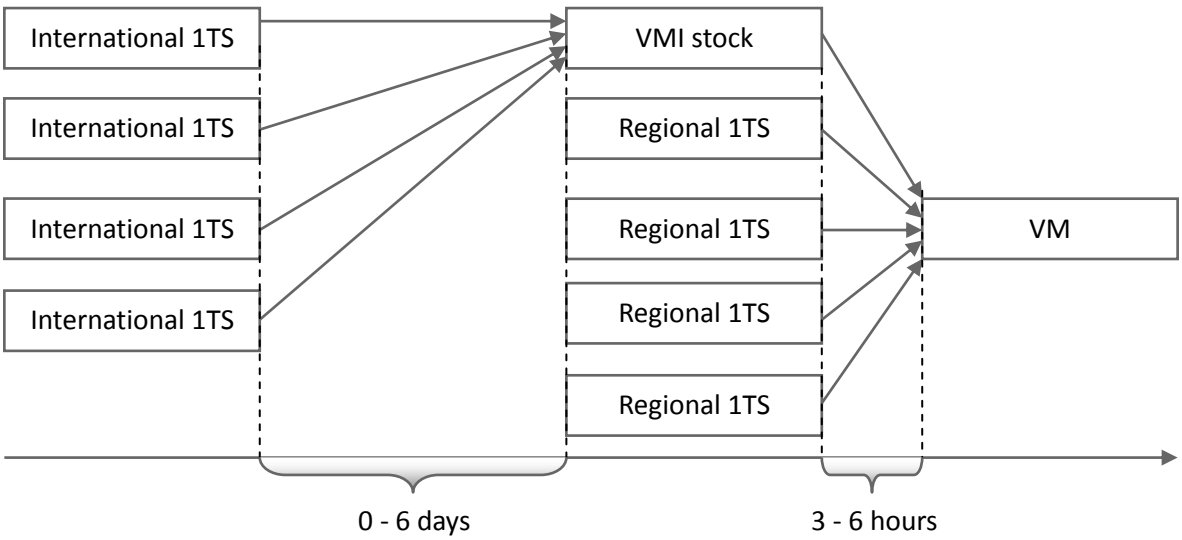

Based on Klug (2013) p. 94; Lysons and Farrington (2012) p.330

Referring to VDA ( Verband der Automobilindustrie e.V. ), delivery call-off is built upon three stages. At the first stage, call-off refers to the production capacity and material requirement planning for a period of 6 to 12 months in preview, in the special case for 18 months. This call-off is a rolling forecast and scheduled weekly in partial deliveries. At the second stage, call-off in details is released based on the ordered modules or systems for a period of maximal 15 days in preview. This is a rolling forecast, too and actualized daily. At the third stage, the production-synchronous call-off refers to several partial deliveries with the small quantity daily, which normally takes between 3 and 6 hours following JIS principle (Klug 2013). Because this thesis focuses on the delivery pro-

cesses between the 1TSs and a VM, it is necessary to discuss delivery strategy of the 1TSs in the German automotive industry at first.

According to the study of Schweppe (2008), the majority of the German VMs have the international 1TSs from Asia, South America, South Africa or other EU countries. However, because of the good delivery performance and quality of the purchased objects, the regional 1TSs still remain the leading position. As illustrated in **Figure 2.1.3**, the international 1st Tier Supplier (1TS) transports their modules or systems to the stock which is located near to the VM's plant and operated by the VMI logistic provider (Göpfert and Braun 2016; Schweppe 2008). This transport process can be carried out by ship, train or track, and takes up to 6 days (Arndt 2014), according to the demographical conditions and traffic infrastructures respectively. By using the VMI stock, the international 1TSs can keep the certain stock level as agreed in the purchase contract and deliver the models or systems to the VM's assembly lines JIS. In contrary, the regional 1TSs deliver the models or systems directly from their own inventory stocks to the VM's assembly lines JIS (Vahrenkamp and Kotzab 2012).

**Figure 2.1.3. 1st Tier Suppliers Delivery Strategy in the Automotive Industry**

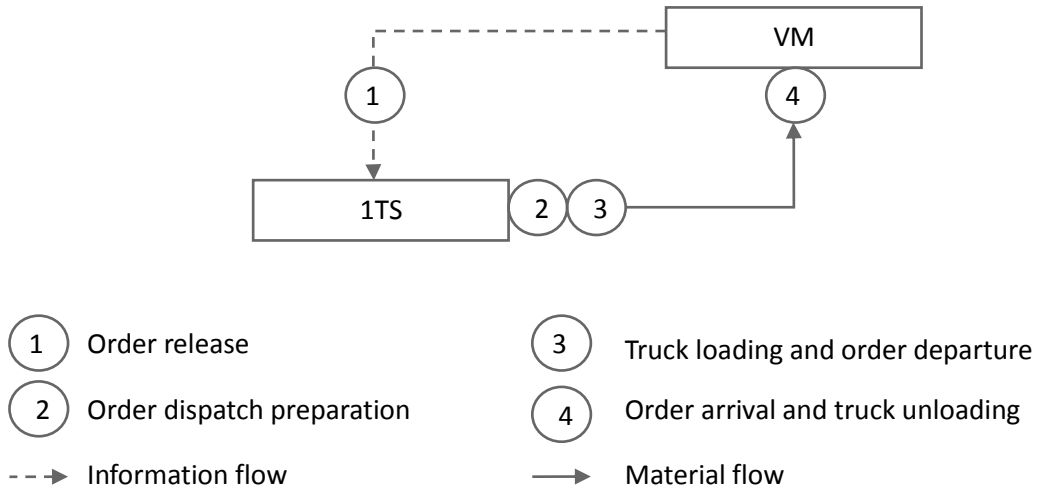



In practice, there are two material flows during the 1TSs delivery processes: module/system flow and container flow, which are regarded as independent management issues in terms of push-strategy with *KANBAN* (Göbl and Froschmayer 2011; Wildemann 2007) and full truck loading (Gehr and Hellingrath 2007). In this thesis, the container flow will not be taken into consideration, but could be handled in future works. **Figure 2.1.4** illustrates a typical 1TS delivery process.

1. VM sends the actual information about the materials requirement to 1TS daily. This information due to the aforementioned call-off details should be shared in real-time.
2. 1TS will check out his own inventory status. Generally, the 1TS is responsible to keep the safety stock for avoiding the bottlenecks effect on the VM assembly line. If there is no sufficient quantity in stock to deliver on the date as VM scheduled, 1TS will release the delivery call-off to the 2TSs by EDI, Email or fax. In the meantime the 1TS arrange the transportation in the delivery schedule with the 3PL.

3. Ordered objects are checked out after the delivery note (quality, quantity, and destination) and loaded in the truck. Before the loaded truck moves off, 1TS will inform VM when the truck should arrive on the delivery destination.
4. The order delivery process is closed, when the truck is unloaded and the VM confirms the delivery. The received objects are transferred directly to the assembly line without quality control.

**Figure 2.1.4. 1st Tier Supplier Delivery Process**




Based on: Klug (2013) and Alicke (2005) p. 174

The material requirement forecasting is just an assumption (Lysons and Farrington 2012), therefore, the logistics processes could behave in a stochastic way and result in the short-term delivery change. The typical events which trigger rescheduling delivery can be that customers change their orders on different weekdays (Wilke 2012) and the VMs arrange their production plan according to the certain shift calendar ( Kropik 2009). Because of the bottlenecked or delayed delivery on the 1TSs side, a VM has to reschedule their assembly plan (Alicke 2005; Wilke 2012). This causes further delivery change, as shown in **Figure 2.1.2**, if the car body of the fixed assembly order "8" is not to be available at the assembly line JIS, the ordered module or system of the fixed assembly order "8" will be not used and cancelled. In another word, one object less will be delivered than the previously scheduled quantity. This situation can be also treated as delivery disturbance which relatives to the extern 1TSs, though it may be caused by the intern 1TSs.

However, rescheduling delivery drives the occurrence of a disturbance that endangers the synchronized delivery of all JIS modules and systems to the assembly line (Spille 2009; Heinecke et al. 2012). Spille (2009) discusses the four delivery risk factors. The purchased objects are not delivered on the scheduled date, with the scheduled quantity or the right quality as the technical specifications required, or at the calculated price. Heinecke et al. (2012) light out that in practice it refers to difficulties in aligning the material flow of the right modules or systems with the correct machines, because of the high instability with frequently missing or wrong material coupled with a continuously moving assembly line. This can be taken as further risk factor of a delivery "*right place*" (Waters 2011). Schleppe (2008) points out the quality cannot be assessed or tested when the JIS delivery arrives, but that possibly will be found out several weeks later. Secondly, only in

the case of transport and transit processes, the delivered items could be damaged and going to be assessed as quantity reduction, but not as a quality problem (Spille 2009). About price issue, that is handled rather than at the SC strategy and execution stages, so that it doesn't come to 1TSs delivery processes which belongs to the SC operation tasks. On the other hand, because of the special contractual agreements and signal sourcing strategy, the VMs have enormous dependency on the 1TSs and the price fluctuation in a short-term vision could lead to the loss of profitability from the VM's side, not or barely influence on the procurement processes (Schleppe 2008; Spille 2009). Due to the explanation of **Figure 2.1.4**, there are no issues about the price and quality involved in the 1TSs delivery processes at the operational stage, and for now, no literatures light out that the price fluctuation is taken as a direct cause of delivery processes disturbance in the JIS case.

Until now, the factors which can disturb the JIS delivery processes have been discussed in terms of delivery time, quantity, quality and price. Especially, this thesis puts the focus on delivery time and studies on which JIS delivery process is regular, which disturbed, in the case of rescheduling JIS deliveries. With the simulation results of the different rescheduling strategies for the 100 simulation time days, Heinecke et al. (2012) show that: The rescheduled deliveries can be delayed up to 20 hours later than JIS required; The respective *lead time* of 1TSs are 18, 12, and 6 days and on-time delivery reliability are 86%, 91% and 97% respectively.

### 2.1.3 Information Technology in Supply Chains

In this section, the IT application in SCs is to be discussed. Information Technology is identified as one of the four *SCM enablers* (the other three are: organizational infrastructure, strategic alliances and human resource management) for ensuring SCM success (Lysons and Farrington 2012). The *Supply Chain Management Systems* (SCMSs) can be regarded as the systems where a set of the IT tools are subdivided for supporting enterprise in implementing SCM concept from the hierarchical or cooperative SCs perspective, but without a special focus (Schulze 2009). Schulze (2009) classifies IT application in SCM into four catalogues. Supply chain planning rephrases the analysis of resource requirement management and optimization processes by collaboration. Supply chain execution concerns and monitors the status of logistics operation processes. Supply chain integration embeds the planning and operation systems. On behalf of this foundation, the coordination of the entire SCs business processes across enterprises can be implemented as well as every independent sub processes. Supply chain interface enables communication to the market environment and establishes a framework of the E-Business development to create a potential for further system integration with a new SC partner. **Table 2.1.1** shows the IT application in SCM.

Achieving visibility throughout the SCs by systems integration is important in the search of competitive advantage (Bennett and Klug 2011). A wide range of software products for systems integration are proposed by the standard and individual application on the differentiated price levels. However, the software purchasing needs a range of complex calculation and evaluation methods and it can be managed as a project (Harnisch 2015; Schulze 2009). Based on the current publications, following it is a summary of the perspectives of solving the SCMSs integration problems.

Trautmann (2014) proposes the *changeable multi- intelligent agent in real-time* represents an intelligent interface for implementing the collaboration and coordination tasks as well as making

decisions such as a command to the other agents, carrying out by the system program. This information is being shared in real-time so that it drives the high level processes mechanism and automatic optimization. This technique based on the RFID theory by using software-agent is able to integrate with EAI and the entire system are also as embedded system, i.e. the project of Fraunhofer IML – smart container.

**Table 2.1.1. IT Application in Supply Chain Management**

| Supply Chain Management | IT Application |
|---|---|
| Supply Chain Planning | ▪ Production Planning and Control Systems<br>▪ Advanced Planning Systems (APS)<br>▪ Collaborative Supply Chain Management (CSCM) Systems |
| Supply Chain Execution | ▪ Production Planning and Control Systems<br>▪ Warehouse Management Systems (WMS)<br>▪ Transportation Management Systems (TMS)<br>▪ Supply Chain Event Management (SCEM) Systems |
| Supply Chain Integration | ▪ Enterprise Application Integration (EAI) Systems |
| Supply Chain Interface | ▪ Electronic Business Tools (Electronic Market Places) |

Based on Schulze (2009)

Görgülü and Pickl (2013) state that a new kind of *business intelligence*, namely a combination of the computational, evolutionary algorithms, system dynamics, data farming and modern heuristics via modern soft computing approaches, needs to be approached for the complex SC networks. Simulation combining heuristics method, also defined as *simheuristics* (Juan et al. 2015), enables that integrated databases can be generated for decision support (Juan et al. 2015; Ickerott 2007).

### 2.1.4 Key Performance Indicators

This section will present the approaches to measure process performance which implies competitive advantages of an enterprise. One precondition of obtaining competitive advantages is to optimize the processes continuously. For achieving this, the parameters of the process goals, also regarded as *target values* are defined and transformed into KPIs in order to compare with the processes *actual values*. KPIs can be categorized into qualitative set as good, bad, quick, slow and quantitative set, for example the value in percentage (Alicke 2005). On the other hand, setting up a controlling system of an enterprise with KPIs depends on the enterprise management strategy and goals (Christopfer 2011; Alicke 2005). Generally, there are two KPI concepts in terms of processes optimization: Supply Chain Operation Reference Metric (SCOR Metric) and Balanced Scorecard (Alicke 2005).

The first approach is SCOR Metric which is the standard instrument of SCOR Model to measure the performance of SC operations with the overall orientation on the cross-enterprise processes of plan, source, make, deliver and return. These five perspectives are interpreted into KPIs portfolio with the focus on the indicators of order fulfillment process in order to develop the processes

performance and achieve the *perfect order*. A perfect order means that an order is delivered with the right quantity and quality, in time as guaranteed, with the right documents, right functional specifications as a customer expected and right payment of a customer in time. Thus, a process of a perfect order should perform without disturbances.

The second approach is *Balanced Scorecard in SCM* which is constructed based on the enterprise strategy with the viewpoints of finance, customers, processes and development potentials. Therefore, a controlling instrument of an enterprise can be established and monitor the KPIs systematically. Monitoring KPIs is implemented by calculating the deviation of the target values and *actual values*. If this deviation is beyond the defined value interval where a process performs as expected, a signal of process disturbance will be alerted. To define the target values needs to *implement benchmarking* at first. This derives a wide range of *decision*-marking issues which can be solved by applying *data envelopment analysis* (Alicke 2005). However, KPIs data collection cannot be easy, because it often associates with data quality issues. For example, the processes data could be implemented in inconsistent data formats or not in a multidimensional data model. Furthermore, missing value could be resulted by human mistakes (Alicke 2005). For solving these problems, knowledge discovery and data mining techniques are widely applied for data analysis (Cios et al. 2007).

## 2.2 Knowledge Discovery

This section provides an introductory of data, information and knowledge as well as the relationships among them. This creates a basic understanding of the concept of Knowledge Discovery in Database which will be discussed in detail. Furthermore, a concrete description of data mining methods and a summary of their applications in SCs will be presented.

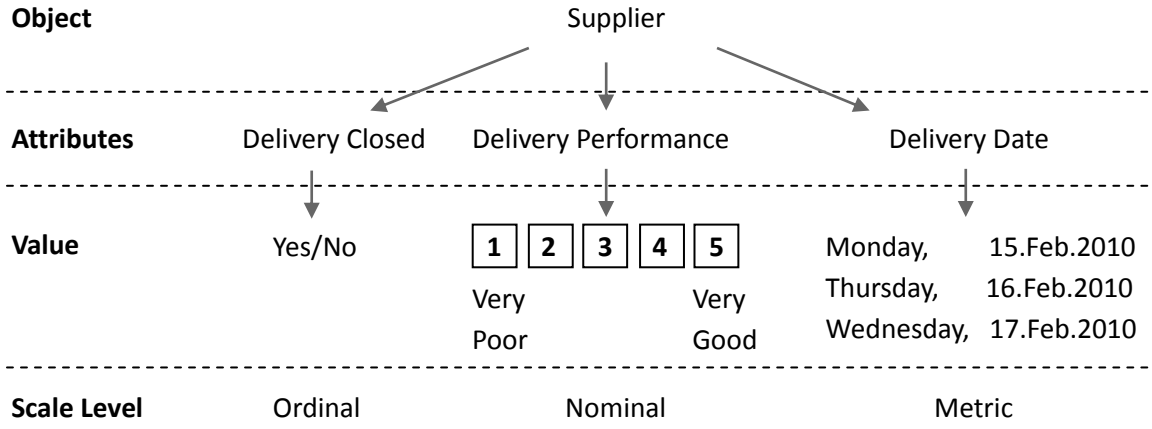
### 2.2.1 Data, Information and Knowledge

This section focuses on the theoretical background of data, information and knowledge. As shown in **Figure 2.2.1**, a *value* of an *attribute* is a single unit of information at the most elementary level. The *objects* described by attributes are combined to create data sets which in turn are staved as *flat files* and in other formats using databases and data warehouses (Cios et al. 2007). *Data* are defined as a collection of symbols and characters with their corresponding syntax (Cleve and Lämmel 2014). Data are differed into unstructured, semi-structured and structured data. *Unstructured data* are such as graphics or text. *Semi-structured* data refer to the combination of unstructured data and structured data, e.g. websites dominate a structure, but they are described with texts, which they place in the categories of unstructured data. The *structured data* are mostly comprehended as relational database tables or data in the similar data format (Cleve and Lämmel 2014).

As illustrated in **Figure 2.2.1** which takes the example of a supplier in SCs, data can be categorized into three types for data analysis. *Nominal data* are qualitative concepts such as delivery status if the delivery process is closed (yes, no), and can be transformed into numbers as delivery closed (yes=1, no=2). Nominal data can be compared with each other. *Ordinal data* can also be

transformed into numbers, but rather for ranking or ordering the labels as delivery performance (very bad = 1, bad =2, normal=3, good=4, very good=5). Ordinal data can be compared with each other. *Metric data* are such date which presents information that ordinal data describe. Data can also be classified in discrete and continuous data. *Discrete data* are the sort of these which only a finite amount of values can be adopted, while *continuous data* are the numerical values which each optional numerical value only can be adopted within the definition range. Continuous data can be converted into discrete data, e.g. by constituting interval (Cleve and Lämmel 2014; Cleff 2011). Precisely speaking, this category (**Figure 2.2.1**) can also be treated as the basis for scaling and coding the values of attributes in order to calculate the data similarity.

**Figure 2.2.1. Value, Attribute, Object and Scale Level**



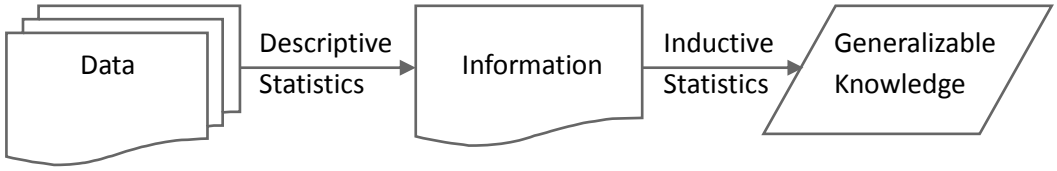

Based on: Cleff (2011), p. 20

Cios et al. (2007) summarize the five typical data quantity and quality issues. High dimensionality implies the massive amount of data, referring to the number of objects, attributes and values. Imprecise data, fuzzy or tough data sets can be used to process imprecise information. Incomplete data are lack of the significant attributes or amount of objects. Redundant data refer to the replication of the identical objectives, attributes, or the irrelative data that don't affect the information quality. Missing values could be resulted by manual mistakes or data integration. Noise in the data is defined as a value that is a random error or variance in a measured attribute. Cios et al. 2007 (p.37-44) and Cleve and Lämmel 2014 (p.195-205) contribute more details about data quantity and quality issues. Data consist of facts and then become *information*, when they are considered in certain context and have a meaning. Information is treated as an interpretation of the dedicated data. When information is utilized in the connection of ability, information turns into *knowledge* (Cleve and Lämmel 2014). Turban et al. (2011) define "Knowledge is understanding, awareness, or familiarity acquired through education or experience; anything that has been learned, perceived, discovered, inferred, or understood; the ability to use information."

As presented in **Figure 2.2.2**, to extract the knowlegde from the data and information needs descriptive and inductive statistics. *Descriptive statistics* consists of a collection of methods, with which information can be extracted by description data of the population, for instance illustration of graphics, tables and calculation of descriptive parameters. *Inductive statistics* aims to draw a conclusion about the population from a sample group. Application of descriptive statistics pursues collecting data in various formats, processing them and transforming them into information. After

this information is analyzed and evaluated by the inductive statistics methods, the general knowledge is generated (Cleff 2011).

**Figure 2.2.2. From Data to Information and Knowledge**




Source: Cleff (2011), p. 5

Knowledge can be represented in the way of rules, graphs and networks. Rules are conditional statements of the form (IF condition THEN conclusion), where the condition and conclusion are descriptors of the pieces of knowledge about the domain, while the rules themselves express the relationship between these descriptors. Trees form the relationship among attributes in a vertical hierarchy, commonly used as decision tree emphasizing on the collection of rules. For each rule, the tree starts from its root and moving down to one of the terminal nodes which represent concepts or attributes with their corresponding values. Networks are illustrated as generalized graphs in the sense that at each node of the graph some local processing capabilities are encountered. The network not only represents the knowledge, but also contains the underlying processing illustrated at the local level. (Cios et al. 2007)

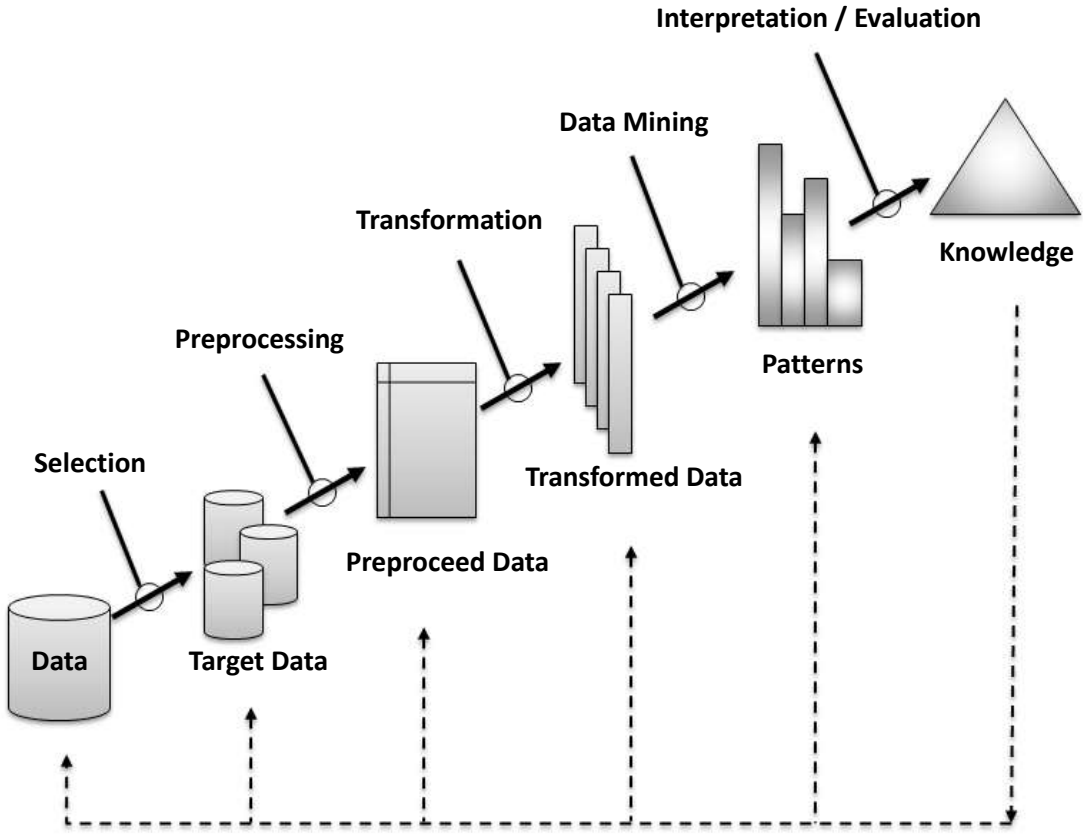
A model can be defined as a description of causal relationships between input and out variables. In some cases, the theory is also regarded as a model, but in practice, the model is adopted for representing theory in certain facts of case. An artificial model is combined with different theoretical consideration to approach the concept of reality in abstraction and simplification, attempting to depict the real problem in a model (Cleff 2011). The *models* are always imperfect, therefore there are always model errors associated with them. Model error is calculated as the difference between the observed value and the expected value, and they can be identified if there is an absolute or squared error between them. When a model is generated from data, in this case it is called “fit the model to the data” and the generated model is regarded as a prediction. To select the best one for obtaining meaningful and durable conclusion, the prediction candidates need to be validated for their goodness of fit, namely fit error. The goodness of prediction is treated as the generalization error. Goodness of prediction refers to the concepts of over fitting the data, or under fitting the data. *Over fitting* relates to an unnecessary increase in model complexity. In contrast, *under fitting* describes a situation that the model is too simple to fit the data well. Therefore, the model needs to be evaluated before they are selected (Cios et al. 2007).

### 2.2.2 Knowledge Discovery in Databases

As presented in previous section, the knowledge is extracted from the processed data. This is the theoretical fundament of the model of *Knowledge Discovery in Databases* (KDD), also called KDD is as *Knowledge Discovery Process* (KDP), which historically was coined at the first KDD shop (Piatetsky-Shapiro 1991) in 1989 to emphasize that knowledge is the end product of a data-driven discovery. Fayyad et al. (1996) propose the first basic structure of the model and define KDD as “the nontrivial process of identifying valid, novel, potentially useful, and ultimately

understandable patterns in data". At an abstract level, the KDD field is concerned with the development of methods for making sense of data (Fayyad et al. 1996). As illustrated in **Figure 2.2.3**, KDD consists of nine steps outlined as follows:

**Figure 2.2.3. Steps that Compose the KDD Process**




Source: Fayyad et al. (1996)

1. The first step begins with developing an understanding of the application domain. This step includes learning the relevant prior knowledge and the goals of the end user of the discovered knowledge and identifying the goal of the KDD process from the management perspectives.
2. The concern of the second step is to create a target data set. This step contains selecting a data set, focusing on a subset of variables or data samples, on which discovery is to be performed,
3. The third step works on data cleansing and preprocessing. Basic operations of this step include removing noise, collecting the necessary information, deciding on strategies for handling missing data fields, accounting for information that time sequence records and identified changes.
4. The fourth step deals with data reduction and projection. The content of this step is to find useful attributes to represent the data depending on the project goal. With dimensionality reduction or transformation methods, the irrelative attributes can be delimited in order to ensure a precise result with an effective effort.
5. The fifth step starts with data mining process, searching for patterns of interest in a particular representational form, including classification rules or trees, regression and clustering.

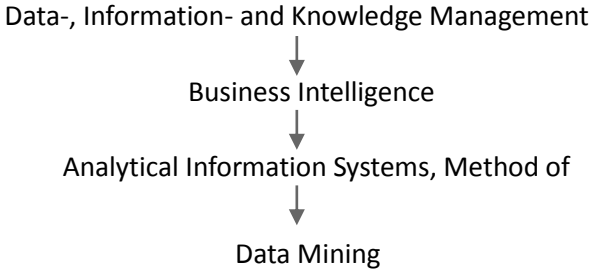
6. The sixth step dues to choosing the data mining algorithms with the overall criteria of the KDD process, searching for patterns in the data and deciding which models and parameters may be appropriate.
7. The seventh step aims to generate the patterns in a particular representational form or a set of such representation such as classification rules or trees, regression and clustering.
8. The eighth step focuses on interpreting mined patterns, possibly returning to any of steps through 7 for further iteration. This step can also involve visualization of the data based on the extracted models.
9. The ninth step is acting on the discovered knowledge and consists of incorporating the knowledge into another system for further action, or simply documenting it and reporting it to interested parties. This process also includes checking for and resolving potential conflicts with previously believed or extracted knowledge.

Cios et al.(2007) state that the future of knowledge discovery model lies in achieving overall integration of the entire process through the use of popular industrial standards, such as eXtensible Markup Language and Predictive Model Markup Language

### 2.2.3 Data Mining

As discussed in previous section, the data mining methods are used repeatedly in the KDD process. This section presents an overview of the data mining. With respects to the business informatics lexicon (**Figure 2.2.4**), Cleve and Lämmel (2014) state *data mining* should be regarded as an approach of analytical information system, which is subordinated in *business intelligence*, arranged in business intelligence. Business intelligence is a conceptual framework for decision support, combined with architecture, databases, data warehouse analytical tools and applications (Turban et al. 2011). From the viewpoint of the informatics science, business intelligence has the cross-references to information and knowledge management, databases, data warehouse, *artificial intelligence* as well as data mining. In the literal sense of informatics science, business intelligence is adopted as an essential application for direct support for decision-making, including Online Analytical Processing, Management Information Systems and Executive Information Systems. Artificial intelligence is the subfield of computer science concerned with symbolic reasoning and problem solving. Furthermore, artificial intelligence is a scientific domain of knowledge processing and applied as a technique for presenting the results of analysis (Cleve and Lämmel 2014).

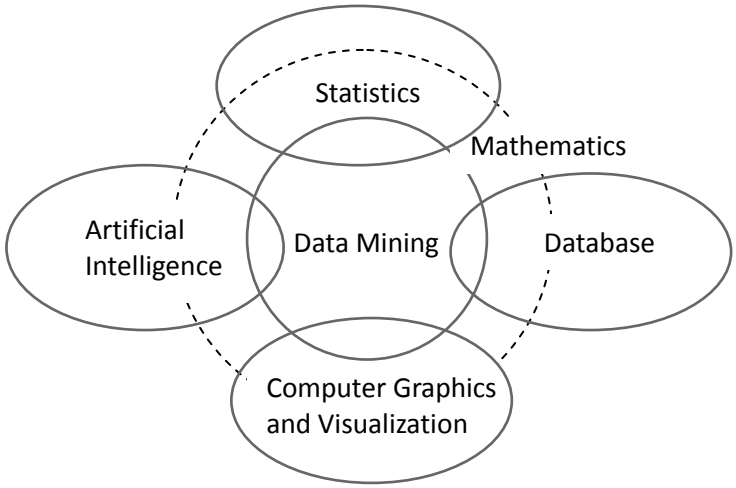
**Figure 2.2.4. Data Mining and Business Intelligence in the Lexicon Hierarchy of Informatics**




Souce: Cleve and Lämmel (2014), p. 3

Data mining is an interdisciplinary technique (**Figure2.2.5**). All of the final analytical methods are based on the *mathematics*. Especially, *statistics* is important on behalf of data analysis at the step of data preparation and sets up a fundament of a couple of data mining methods. Furthermore, statistics is a test tool to identify the best knowledge pattern. *Data warehouse*, where data mining extracts the knowledge pattern, orients on time stamps fulfilled with data from heterogeneous information systems and possibly in inconsistency data format. *Expert system* is a better knowledge system which endures to simulate the performance of the human experiments in independent application fields. As a knowledge storage, expert system enables knowledge presentation that data mining extracts. *Machine learning* is a computer learning capacity that programs can generate and present the knowledge from the input data. *Visualization and computer graphics* are regarded as a technique of data mining not only to provide a visual knowledge presentation, but also to highlight the data relationships that might be not sensed from human viewpoint. (Cleve and Lämmel 2014)

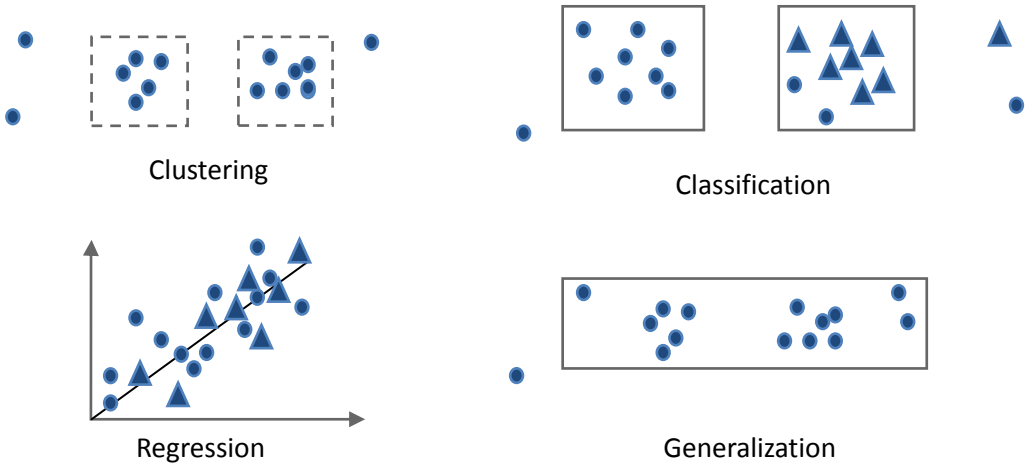
**Figure 2.2.5. Interdisciplinary of Data Mining**




Souce: Cleve and Lämmel (2014), p. 12

Data mining is the application of efficient algorithms, which can discover the expected or believed pattern. For now, there is no unified agreement with the data mining task fields according to the current publications. As shown in **Figure 2.2.6**, data mining tasks can be categorized into five fields: clustering, classification, regression, association rules and generalization. The goal of *clustering* is to partition the database in groups of objects so that the objects of a cluster can present in a similar way and the objects of different clusters can present in a dissimilar way. The *outliers* are the objects that belong to the small grouped cluster. *Classification* is learning a function that maps an attribute value into one of several predefined classes. *Regression* is a statistical method for estimating the relationships among variables and is widely applied for prediction and forecasting where its use has substantial overlap with the field of machine learning. One of the important application of the *association rules* is to describe presenting and strong relationships within the transaction processes, e.g. “WHEN A AND B THEN C”. *Generalization* aims to express an amount of data compactly as well as possible. Under this amount of data, the values of attributes are generalized and the number of data sets is reduced in order to optimize the classification results (Ester and Sander 2000). The high-level primary goals of data mining in practice tend to be prediction and description (Fayyad et al. 1996).

**Figure 2.2.6. Tasks of Data Mining**




Based on: Ester and Sander (2000) p. 5; Fayyad et al. (1996)

Data sets are always in a dynamic state by adding, replacing, removing or deleting objects or features. That leads to different incremental and decremental data mining. Incremental data mining merges that new knowledge is generated from new data and the existing knowledge. Decremental data mining refers to generating new knowledge from a new data set which is mixed up with existing data set and new data (Cios et al. 2007).

**Table 2.2.1. Unsupervised Learning and Supervised Learning**

| | |
|---|---|
| **Unsupervised Learning** | Clustering: K-Means Algorithm, Expectation Maximization Clustering |
| | Association Rules: Generalized Sequential Patterns |
| **Supervised Learning** | Statistical Methods: Bayesian Methods, Regression |
| | Decision Tree, Rule Algorithms |
| | Artificial Neural Networks |

Based on Cios et al. (2007)

### 2.2.4 Knowledge Discovery Methods in Supply Chains

The typical industrial application of knowledge discovery techniques is the CRISP – DM model (CRoss-Industry Standard Process for Data Mining), which was first established in the late 1990s. It is a leading industrial model, which is characterized by an easy-to-understand vocabulary and good documentation, mainly due to knowledge discovery experience in practical, industrial and real-world (Cios et al. 2007). Cleve and Lämmel (2014) point out the application of CRISP – DM should be implemented in the framework of a project management. With respects to the current publications and research, **Table 2.2.2** provides a short overview of data mining applications in supply chains.

| | Business Management | Data Mining Methods |
|---|---|---|
| **Supply Chain SCOR Processes** | | |
| Source | Supplier Relationship Management (SRM) | Decision Trees: Virtually any suppliers problems that can be reduced to, e.g. for each decision, a set of possible outcomes, together with an assessment of the likelihood of each outcome occurring. Source: Lysons and Farrington (2012), p. 599 |
| Plan | Advanced Planning and Optimazation | Regression : Forecasting and estimating customer demand for a new product. Source: Fayyad et al. (1996) |
| Make | Manufactruing Integration and Intelligence | Association Rule: Identifying the cause roots of product fauiler, optimizing the manufacturing capacity and enabling the condition-based maintenance. Source: Turban et al. (2011), p. 205 |
| Deliver | Transportation Management Systems ( TMS), Warehouse management | Genetic Algorithm: Evoluating the improvecd hypothese of operating VMI in an uncertain demand environment. Source: Borade and Sweeney (2015) |
| Return | Reverse Logistics Management | Clustering Algorithms: With k-Mean algorithm to categorizing the returned commodities in order to improve the manufacturing processes quality. Source: Mohammadi et al. (2014) |
| **Supply Chain Related Processes** | | |
| Engineering and Design | Product Lifecycle Management (PLM) | Multi Agent Data Mining System: Supporting production planning decisions based on the analysis of historical demand for products and on the information about transitions between phases in product life cycles. Source: Parshutin (2010) |
| Sales/ Marketing/Service | Customer Relationship (CRM) ; Field services ; Spare Parts Managment | Clustering Algoriths: Assigning customers in different segments based on their demographics and purchase behaviours. Source: Turban et al. (2011), p. 200 |

## 2.3 Clustering Algorithms

In this section a comprehensive introduction of clustering algorithms is given. Subsequently, an overview of the characters and applications of the different clustering algorithms are discussed. At last, the methods to evaluate and compare the results of clustering algorithms are described in detail.

### 2.3.1 Theoretical Background of Similarity Measures

The goal of clustering algorithms is to compare the strings which encompass objects, attributes and values, in order to group similar objects together. This is based on the precondition that the similarity among the data sets can be quantified and measured by *distance functions*:

$$\text{dist}(x, y) \tag{F.2.3-1}$$

which distance between two strings x and y is measured, so that the similarity of two data sets can be defined as:

$$\text{simil}(x, y) \tag{F.2.3-2}$$

in dependency of their distance measurement. Lengthen the distance between the two data sets it is, lessen the similarity they have.

$$\text{simil}(x, y) = f(\text{dist}(x, y)) \tag{F.2.3-3}$$

Cleve and Lämmel (2014) introduce the typical distance functions as similarity measures as follows:

Hamming-Distance $\text{dist}_H(x,y) = \text{count}_i(x_i \neq y_i)$ (F.2.3-4)

Euclidean Distance $\text{dist}_E(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$ (F.2.3-5)

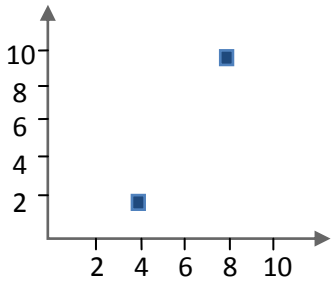
Manhattan-Distance $\text{dist}_{Man}(x, y) = \sum_i |x_i - y_i|$ (F.2.3-6)

Maximum-Distance $\text{dist}_{Max}(x, y) = \max_i (|x_i - y_i|)$ (F.2.3-7)

Weighted Euclidean Distance $\text{dist}_{EW}(x, y) = \sqrt{\sum_{i=1}^{n} w_i (x_i - y_i)^2}$ (F.2.3-8)

*Hamming-distance* function is only applied for counting the positions where are the differences between the data sets and adopts the nominal, ordinal and metric data. Generally, *euclidean distance* function can measure the distance between the two site points in the space as well as the two points in the mathematical dimensions. *Euclidean distance* function can only be utilized for the metric data measurement. *Manhattan-distance* function calculates the sum of every step of the two different routes between two objects in the two dimensions. *Maximum-distance* function measures specifically the longest distance in a dimension. Weighted Euclidean Distance function is developed on the Euclidean distance and only used for the numerical attributes for weighting the attributes which could influence more on the distance (Cleve and Lämmel 2014). **Figure 2.3.1** shows the results by calculating different distance functions.

**Figure 2.3.1. Examples of the Distance Functions**




dist $_H$ = 2

dist $_E \approx$ 8,9

dist $_{Man}$ = 12

dist $_{Max}$= 8

Source: Cleve and Lämmel (2014), p. 40

Clustering algorithms calculate the distance between the data sets only when the attributes are metric, and the data sets have to normalize in the interval (0, 1) first, if the maximum value and minimum value are available. This can be implemented by the transformation function as follows:

$$x_{new} = \frac{x - \min(x_i)}{\max(x_i) - \min(x_i)} \qquad (F.\ 2.3\text{-}9)$$

Furthermore, Siegel (1988) and RapidMiner tutorial (2015) provide the proportion method to normalize the attribute values as proportion of the total sum of the respective attribute i.e. each attribute value is divided by the total sum of that attribute values in the interval (0,1).

### 2.3.2 Art of Clustering Algorithms

This section will cover one technique of unsupervised learning, clustering algorithms, which are utilized for generating the association rules. As in **section 2.3.1** discussed, clustering is a suitable, unsupervised algorithm of discovering structure on its own by exploring similarities. between data. Therefore, clustering is also used for attributes classification. According to Cleve and Lämmel (2014), clustering algorithms can be divided into four categories: partitioning clustering, hierarchical clustering, Density Based Spatial Clustering Of Application with Noise (DBSCAN) and clustering with self-organizing map. *Partition based clustering* begins with the arbitrary k-clusters which can be represented by the medoids or centroids. Based on the first step, the optimization can be formalized so that the further steps go with reordering these initial clusters by calculating the new medoids or centroids until each object is assigned to a certain cluster. Partition based clustering follows tow principles. Firstly, every cluster is built from one object at least. Secondly, every object belongs to only one cluster. K-means and k-medoids are typical algorithms of partitioning clustering. The essence of *hierarchical clustering* lies in merging the clusters with minimal distance so that a dendrogram structure of clusters is created. *DBSCAN* builds a cluster relayed on the formation of clustering on the basis of the density of data points. DBSCAN especially deals with density-based spatial clustering of applications with noise. *Clustering with self-organizing map* is a method combined with clustering algorithms and neural net work. This self-organizing map is trained by the input data in a two-dimension generally, in order to group the similar output musters in a cluster. (Cleve and Lämmel 2014)

Especially, the focus of this thesis lies in the partitioning clustering which includes k-means algorithm, k-medoids algorithm and expectation maximization clustering. Following the concrete descriptions are given.

### i. K-Means Algorithm

*K-means algorithm* predefines the number of the cluster and follows the principle that each object is assigned to precisely one of a set of clusters. An object which contains *n* attribute value can be understood as a point in an *n*-dimensional coordination system. The value of k is generally an integer that implies how many clusters are defined to be grouped. If the objects in one cluster, it means that they have the most similarity which is based on the distance between them by using, e.g. Euclidean distance function. K-means is an algorithm that follows the iterative optimization processes. First of all, an initial center of a cluster is selected optionally and represents a centroid of a cluster. This process is going to be repeated in order to improve the quality of the cluster grouping. In other words, optimization of the distance between the reordered cluster structure, until all the objects find the own clusters which they belong to (**Figure 2.3.2**). The optimal ordering clusters means that the sum of all the distances between the objects and their respective centers of cluster should be minimal and can be measured by the compactness "cost " of a cluster $C_i$ :

$$\text{Cost}(c_i) = \sum_{x \in c_i} \text{dist}(x_i, x) \qquad \text{(F. 2.3-10)}$$

The sum of all cluster costs in total is obtained by:

$$\text{Cost} = \sum_{i=1}^{k} \text{Cost}(c_i) \qquad \text{(F. 2.3-11)}$$

This situation can be expressed as follows:

**PROCEDURE** K-Means modification

Create random k initial cluster $C_i$

//all objects as random are assigned to a cluster

**REPEAT**

Reorder: = false

Fix the centroide $x_1, x_2, \ldots, x_k$ of the cluster

**IF** x exists, which is nearer to an another cluster center $x_j$ than to its actual center $C_j$

**THEN** Assign x is assigned to the cluster $C_i$

Reorder: = true

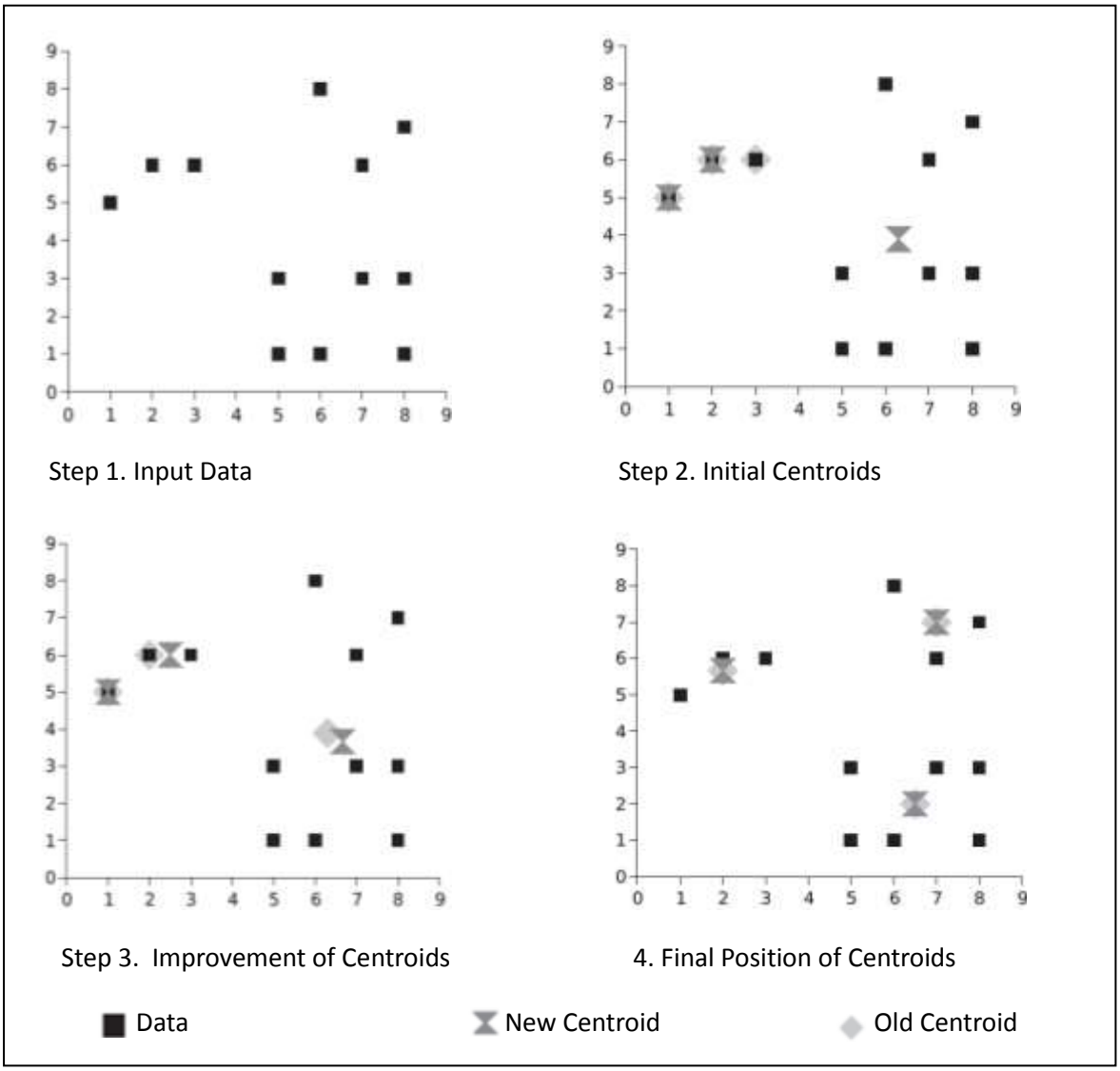
**ENDIF**

**UNTIL NOT** Reorder

**END** K-Means modification

The number of the reordering clusters depends on the data sample size and user's predefinition in terms of the respective target. An advantage of k-means algorithm is able to group the stable clusters by proceeding iterative optimization. On the other hand, k-means is relative easy to implement by calculating the distance and reordering the new centroids. Disadvantage is that the result quality of k-means is influenced by the quality of the initial cluster partitions. Furthermore, k-means is sensitive to outliers, because of reordering centroids by using the distance function. If the input data are nominal or ordinal attributes, they have to be transformed in numerical value. K-means algorithm can also result the convex cluster which is founded in other clusters. (Cleve and Lämmel 2014)

**Figure 2.3.2. Optimization Processes of the K-Means Algorithm**




Source: Cleve and Lämmel (2014), p.142-143

### ii. K-Medoids Algorithm

The centroid is the arithmetic mean, namely "average", of all the points in the cluster. This arithmetic mean is defined by the average attributes values that belong to all the objects in a cluster. Sometimes, this centroid could be one of the objects in the cluster. To k-medoids algorithm, instead of the centroid, cluster is represented by its *medoid*. Firstly, k objects are selected as representative points of clusters. By distance function, each object is assigned to the cluster to whose medoid it has the shortest distance. As long as the reordering the clusters proceeds, the medoid of the clusters are going to be recalculated until the cluster quality calculated by cost function (**F. 2.3-11**) does not improve further. The reordering processes of the *k-medoids* algorithm proceeds as follows:

**PROCEDURE** K-Medoid

Select k objects $m_1 \ldots m_k$ as cluster representatives

Assign all objects to a cluster $m_i$ according to the results of the distance calculation

**REPEAT**

Reorder: = false

**IF** x and $m_i$ exist, and the exchange of their respective roles as normal objects and medoid can improve the current cluster quality

**THEN** exchange their respective roles by calculating distance

Reorder: = true

**ENDIF**

**UNTIL NOT** Reorder

**END** K-Medoids

In contrary to k-means algorithm, k-medoids is not sensitive to outliers, because the distance calculation proceeds with the existing objects which themselves are members of the clusters. For this reason k-medoids is regarded as the robust clustering algorithm. (Cleve and Lämmel 2014; Cios et al. 2007)

#### iii. Expectation Maximization Clustering

Being different to k-means and k-medoids algorithms, the expectation maximization (EM) clustering groups the clusters based on the frequency how often an object is assigned to a certain cluster. In order to find out this frequency, all the objects, which need to be assigned to more clusters, approximate the cluster by Gauß's distribution. EM clustering begins with k as a random Gauß's distribution optionally and calculates the frequency. $W_{i,}$ represents account of objects which are assigned to a cluster $C_i$:

(F. 2.3-12)

$$P(C_i \mid x) = W_i \frac{P(C_i \mid x)}{P(x)}$$

Subsequently, the function:

$$E = \sum_x \log (P(x)) \qquad \text{(F. 2.3-13)}$$

will determine whether the given data with the maximal frequency are from the calculated Gauß' distribution or not. In other words, E should be maximal by the iterative improvement steps. Furthermore, Cleve and Lämmel (2014) provide a concrete description in detail.

### 2.3.3 Methods of Evaluation

Evaluating the results of clustering algorithms is not easy, because clustering is an unsupervised algorithm. That means the there is no existing target model to compare with. Therefore the evaluation of clustering results whether the clusters are well grouped or not, can be carried out in two ways. One way is to compare the results of different clustering algorithms with the identical data sample size. The other way is to measure the cluster quality in the German verb, and especially appropriates to k-means and k-medoids algorithms in terms with the distance calculation. Cleve and Lämmel (2014) provide two approaches to measure the cluster quality as follows:

Approach 1 measures the compactness quality $G_1$ of a cluster by calculating the sum of the deviations between the objects $x \in C_i$ in a common cluster $C_i$ and the their quadrates $m_i$. k is the predefined account of the clusters. Smaller the value of $G_1$ is, better the quality the cluster performs.

$$G_i = \frac{1}{\sum_{i=1}^{k} \sum_{x \in c_i} dist(x, m_i)^2} \quad \text{(F. 2.3-14)}$$

Approach 2 Calculates the sum of the distance quadrates between the individual quadrates mj and mi . in the contrary to the approach 1, bigger the value of G2 is, better the quality the cluster behaves.

$$G_i = \sum_{1 \le i \le j \le k} dist(x, m_i)^2 \quad \text{(F. 2.3-15)}$$

### 2.3.4 Methods of Statistical Test and Ranking Results

This section outlines the statistical test methods that evaluate the data modeling results. Evaluating a data model begins with calculating its error (Cios et al. 2007). If there are more than two data models to compare with each other, the ANalysis Of VAriance (ANOVA) method is applied (Siegel 1988).

One-Way ANOVA is a statistical technique which takes the variability of data sets with the complex attributes dimensions. As being measured by the variance, and partitioned into meaningful component parts, it can test whether or not the means of several groups are all equal, therefore generalizes t-test to more than two groups. If it is to multiple two-sample *t*-tests, that would result in an increased chance of committing a type I error. For this reason, ANOVA is thought of as extending the two samples *t*-test to more than two samples and useful in comparing more than two means. 'False positive' or type I error is defined as the probability that a decision to reject the null hypothesis will be made, when it is in fact true and should not have been rejected. (RapidMiner 6, 2015; Siegel 1988)

Siegel (1988) provides an outline of the procedures that will be involved in doing an *F*-Test for differences among several groups. Firstly, the necessary assumption should be checked against the model, then two kinds of averaged sums of squares, namely the between groups and the within groups average square, will be computed, in order to see whether there is a difference among the data models. Before this calculation step, a baseline measure of underlying variability within the models is necessary. The *F*-test is then based on the ratio of these average squares, and this ratio is compared to a critical value from a respective table to see whether the test is significant or not. Finally, if the test is significant, then the detailed differences among the data models may be analyzed further in the next state of testing. For this procedure, there are three assumptions to be satisfied with respects to a fair approximation:

Assumption 1: Each group of data is normally distributed. This assumption is used for the data transformation

Assumption 2: The variability is the same from one group to another. This assumption goes for the analysis of variance.

Assumption 3: The data were obtained by random sampling from the population. In particular, all data observations are independent of each other. This is a fixed part of the experimental design.

*F*-test measures how different the group averages are from one another with respects to the general overall amount of randomness in the situation and the function is described as follows:

$$F = \frac{\text{Average Square Between Groups}}{\text{Average Square Within Groups}} \qquad \text{(F. 2.3-16)}$$

The conclusion due to the significance level is drawn based on the critical value in a corresponding critical value table or calculated by a statistic tool automatically (Cleff 2008; Siegel 1988). After finding the critical value, it will:

1. "Decide that the differences are statistically significant if the *F*-value is larger than the critical value". In this case, the null hypothesis is rejected and come to conclusion that there are indeed some differences among the groups. On the other hand, it will
2. "Decide that the differences are not statistically significant if the *F*-value is smaller than the critical value". In this case, the null hypothesis is accepted and concludes that there is no strong evidence in favor of the groups being different.

The second step of ANOVA test is to compare more than two data groups by computing the confidence interval, but be sure to use the standard errors and degrees of freedom. If 0 is not within the confidence interval, then these two data groups are declared to be significantly different. If 0 is within the interval, then there is no significant difference between the groups. Siegel (1988) provides an introduction about this method and procedure steps in a comprehensive way.

On the other hand, this thesis also concerns how to measure the cluster models performance and rank their results. To solve this, a significant orientation must be mentioned that one intention of clustering algorithms is to fulfill the classification task. Therefore, the methods for evaluating the classification quality can be chosen. Cleve and Lämmel (2014) provide an overview of methods for evaluating the classification quality. This thesis will adopt "Error Quotient" in the German verbal described as follows:

$$\text{Error Quotient} = \frac{\text{Sample Size of the Wrong Classificaiton}}{\text{Total Sample Size of all Classification}} \qquad \text{(F. 2.3-17)}$$

For presenting the result in a better impressive way, this function is transformed in "Success Quotient" in the German verbal described as following:

$$\text{Success Quotient} = 1 - \text{Error Quotient} \qquad \text{(F. 2.3-18)}$$

# 3 Supply Chain Transaction Data and Data Farming

This chapter will discuss how to generate supply chain transaction data by using data farming. Firstly, it will show the theoretical background of supply chain transaction data and data farming. Subsequently, a concrete description of discrete event simulation and V&V techniques will be provided. Finally, the software tools for implementing a data farming case study and data analysis will be introduced.

## 3.1 Supply Chain Transaction Data

Before starting an argument about transaction data, this section discusses the term "transaction" at first. According to Gray and Reuter (1993), if the database is treated as an abstraction that represents the real state, then *transaction* is understood as transformation of the real state that is mirrored by the execution of a program. A transaction is characterized as follows:

- Transaction request and reply: The request or input message that started the operation
- Transaction: All effects of the execution of the operation
- Transaction program: The programs that execute the operation

**Table 3.1.1. Classification of Transaction Data**

| Classified in | Data | Transaction Data |
|---|---|---|
| **Execution in Electronic Data Processing** | ▪ Master Data | |
| | ▪ Inventory Data | |
| | ▪ Change Data | |
| | ▪ Moving Data | √ |
| **Digital Data** | ▪ Numerical Data | √ |
| | ▪ Alphabets Data | √ |
| | ▪ Alpha Numerical Data | √ |
| **Electronic Data Processing** | ▪ Input Data | |
| | ▪ Output Data | √ |

Source: Arndt (2014)

Referring to the B2B activities, transaction data are resulted by the virtual and physical state transformation of material and immaterial exchange in the electronic data processing (Arndt 2014). During the electronic data processing, the input data are to be transformed into output data by a program manipulation (Arndt 2014; Lewis et al. 2002). Based on Gray and Reuter (1993) and Staud (2005) from the viewpoint of data modeling, *transaction data* can be defined as output data which is the transformed state of the raw data in unstructured or semi-structured form by setting up a logical relationship between the existing data in databases, and this processes of creating data relationship can be regarded as transaction processes by programs. For this thesis, the classification of the transaction data in **Table 3.1.1** is adopted.

**Table 3.1.2. Example for Procurement Data**

| 1. Stage | 2. Stage | 3. Stage | 4. Stage | Example |
|---|---|---|---|---|
| **Procurement** | **Master Data** | Supplier Data | Delivery Address | Delivery Location |
| | | | Product | Product Code |
| | | SCs Data | Supplier | Supplier ID |
| | | | Transport | Shipment |
| | **Moving Data** | Order Data | Time Stamps | Scheduled Received Date |
| | | | | Rescheduled Received Date |
| | | | | Actual Received Date |
| | | | Status | Delivery Closed<br>Delivery not Closed |
| | | | Quantity | Scheduled Quantity |
| | | | | Rescheduled Quantity |
| | | | | Actual Received Quantity |

Based on: Arndt (2014)

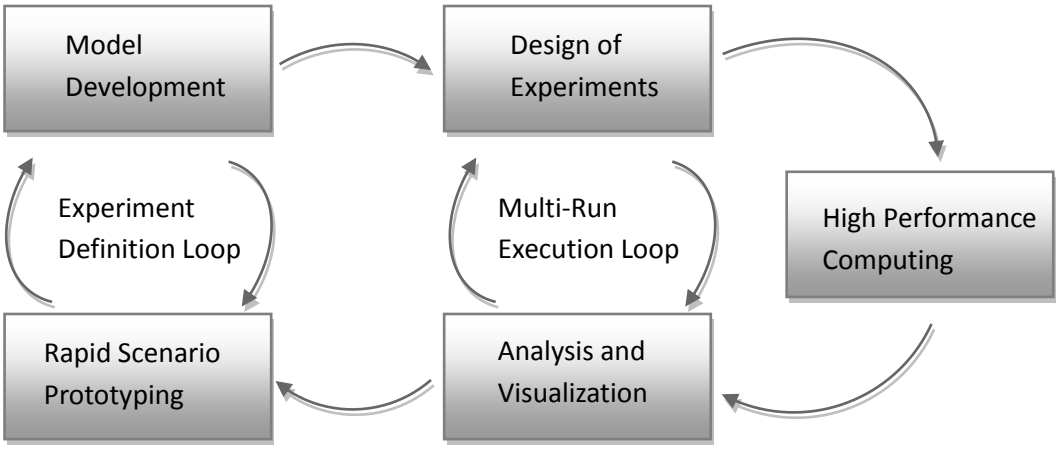
Master data and inventory data orient on the state of data and do not intend to change over a long period. Master data focus on the data cross-enterprises, such as supplier data and customer data as shown in **Table 3.1.2**. Inventory data contain the inbounds logistics data about quantity, e.g. items quantity of a stock, and items value according to XYZ-ABC metric. Change data and moving data reflect the process status of order fulfillment. The change data trigger the inventory data to transform in another state, such as booking a number of items from inventory account for production processes. The moving data describe the dynamical transformation states and correlatively affect the change of the inventory data state so that the enterprise performance can be indicated and assessed (Arndt 2014; Lysons and Farrington 2012). **Table 3.1.2** provides a moving data example of the procurement processes. From the aforementioned characters of the transaction data, the data due to time stamps and quantity are meaningful for a SCs simulation model.

## 3.2 Data Farming

Data farming is firstly introduced by Brandstein and Horne (1998), and has been developed in order to support decision-makers in answering questions that are not addressed by traditional modeling and simulation processes (Horne and Meyer 2004). International data farming workshops (IDFW) take place twice a year under the direction of the Naval Postgraduate School (NPS) Monterey, California in order to exchange knowledge in the field of data farming, covering topics such as model development or experimental designs. In 2010 the NATO Research and Technology Organization (RTO) has started the Modeling and Simulation Group MSG-088 to evaluate and further develop the data farming methodology to be used for decision support

within the NATO. This task group deals with the six realms of data farming (**Figure 3.2.1)**, each of which is represented in a corresponding subgroup of the MSG-088 (Nato Report 2014).

**Figure 3.2.1. Data Farming "Loop of Loops "**




Source: Nato Report (2014)

With a relative economical effort, data farming is based on the idea that by using agent-based simulation it is able to run these simulation models as long as fast enough, in order to generate data as sufficiently as the statistical investigations require. In these simulation models, parametric variation is attributed to the behavior rules of the individual agents. This provides quantitative analysis of complex questions, obtaining robust results, the comparing of results, and "What-if?" analyses (Nato Report 2014). Thus, Data farming is regarded as a decision support method that can lead to insights into the potential consequences of different hypotheses for decision-making (Hofmann 2013). Data farming follows an iterative process with a set of embedded loops that incorporate the five realms, following order: rapid scenario prototyping, model development, design of experiments, high performance computing, and analysis and visualization (Nato Report 2014).

The goal of the first realm *rapid scenario prototyping* is to implement all relevant aspects of a scenario into a suitable simulation model in the context of a question-based analysis. Therefore, the major product of *rapid scenario prototyping* in combination with model development is a tested and documented *base case* scenario as output of the "scenario building loop". The parameters and values have strong influence on the model behavior, especially important for scenario implementation. Thus, it might be necessary to do limited data farming experiments to find meaningful ranges of parameters settings in the sense of a model calibration.

The second realm is *model development* is to simulate the required scenario on the required *level of detail* with the given set of input parameters as well as *measures of effectiveness* for gaining the robust results and comparing them. Measures of effectiveness is the mean of the simulation results and calculated by the statistical methods after each signal simulation run, because the generated data can be of different nature and running one simulation only provides one singular result. An important topic is *reusability of models* and makes a simulation model interoperable with other models and data easily farmable. Interoperability can be reached out by exposing and documenting the input and output variables of a model. Furthermore, any random number generators in models should have their seed values exposed as input variables, so that simulations can be

repeated. In addition, data validation should be properly documented and provided. This realm combines with the rapid scenario prototyping to make up the “experiment definition loop”.

The third realm is *design of experiments*. An experiment stands for a test or a series of tests where the analyst makes intentional changes to input variables of a system so that one can observe and identify the reasons for changes in the output responses. Design of experiments dates back to Ronald S. Fisher’s pioneering work in the 1920s related to agriculture. *Design of experiments* is an information gathering technique that involves making experiments by defining factors with variable levels and enables an efficient exploration of experimental parameter spaces. In *design of experiments* terminology, a *factor*, also called *variable*, refers to an input or a parameter in simulation. *Design of experiments* deals with planning and conduct of experiments so that the output data can be analyzed to reach valid and draw the conclusions. *Confidence intervals* indicate the precision of the experimental technique, used for measurement. When the same technique is used repeatedly, the resulting intervals may be expected to contain the true value with a frequency at least equal to the confidence level 1-α (Clarke et al. 2010). This is used to define the replication of simulation runs (Eley 2012). The calculation of confidence intervals as follows (Robinson 2004):

$$CI = \overline{X} \pm t_{n-1.\ \alpha/2}\ \frac{s}{\sqrt{n}} \qquad \text{(F. 3.2-1)}$$

with $\overline{X}$ = Average of output parameters for the replications

s = Standard deviation of output parameters for the replications

n = Number of the replications

$t_{n-1.\ \alpha/2}$ = Quartile of *t*-distribution with n-1 degree of freedom and a significant α/2

The standard deviation is calculated as follows:

$$S = \sqrt{\frac{\sum_{i=1}^{n} (X_i - \overline{X})^2}{n-1}} \qquad \text{(F. 3.2-2)}$$

with $X_i$ = Replications i

The fourth realm *high performance computing* is the executable side of data farming, and copes with the techniques to efficiently perform thousands of simulation runs on high performance computer clusters thus providing reasonable runtimes even for encompassing experiments. The main purpose of *high performance computing* in the context of data farming is to provide the means to execute a data farming experiment. Other purposes are for analysis and visualization of the output and for generating scenarios used in future data farming experiments. There are six elements involved in a data farming experiment. The first element is a “data farmable” model which actually is the executable model. The second element encompasses a set of model inputs, generically called the “base case”. The third element is a specification of experiment which is the set of variables as the set of model inputs. The fourth element includes a set of *high performance computing* resources, both software and hardware. The fifth element is data farming software. The sixth element contains a set of model outputs.

The fifth realm is *analysis and visualization* that involves techniques and tools for data processing of large datasets resulting from the data farming experiment. The concluding statistical analyses examine the simulation output data upon outliers or unexpected developments as described

throughout the Nato report (2014). The analysis and visualization of results enables answering the what-if questions for the support of decision-making.

As aforementioned, this classical *data farming "Loop of Loops"* is primarily designed for the military application, therefore all mentioned elements or data farming study steps can be taken or selected optionally related to the conditions and specifications in a domain. But the basic theory of data farming, as Brady et al. (2013) formulate, can be adopted for the other domain applications in a general understandable way:

1. *Plant a seed*: Creating a model that interprets the domain information. All uncertainties or missing parts of the domain information are modeled in order to look into the space of possibilities.
2. *Grow the data*: Executing simulation model. Output data are generated by selecting at random from the possibilities which are input as collected and estimated parameters.
3. *Harvest*: Gathering the farmed data and analysis these farmed data via data mining.
4. Improve the model: Using the result of the third step *to improve* the model and restart the first step.

In order to create a logical bridge for describing how the theory of the original data farming concept for military application is adopted in the other domain application, **Table 3.2.1** is set up based on the both of Nato Report (2014) and Brady et al. (2013).

Nato Report (2014) provides an overview of data farming applications as follows:

- Sensitivity studies – By using data farming the input parameters of more complex research areas can be examined to seek the statistical variability of the model.
- Validation and Verification – Data farming enables a fully test of a model's reaction to various input parameters.
- Model development – Data farming can enhance and accelerate the model development by running simulation models over larger parameters rapidly.
- Scenario and outliers analysis – Allowing the model to be executed over a much larger number of input parameters and number of random variations, data farming enables a more complete view of the possible outcomes of different scenarios and to identify which combinations of input parameters or random variations result in outliers.
- Heuristic search and discovery – Data farming encompasses the ability to apply iterative methodologies for model analysis such genetic algorithms and other sophisticated optimization and search methodologies.
- Generation of massive test data sets – Data farming can be used in conjunction with simulation systems to generate massive data sets to test learning algorithms and other data mining tools. This is particularly valuable where actual data may not be available.

**Table 3.2.1. Data Farming Elements**

<table>
<tr><th>Brady et al.(2013)</th><th colspan="3">Nato Report (2014)</th></tr>
<tr><td></td><td colspan="3">Start: “What-if” questions</td></tr>
<tr><td rowspan="13">1. Plant Data Seed<br>4. Model Development</td><td rowspan="9">Experiment Definition Loop</td><td colspan="2">1. Rapid Scenario Prototyping</td></tr>
<tr><td>1.1</td><td>Implementation of all relevant aspects of a scenario into a suitable simulation model in the context of a question-based analysis</td></tr>
<tr><td>1.2</td><td>Combination with model development which is a tested and documented base case scenario as output of the “scenario building loop”</td></tr>
<tr><td>1.3</td><td>Limitation data farming experiments to find meaningful ranges of parameters</td></tr>
<tr><td colspan="2">2. Model Development</td></tr>
<tr><td>2.1</td><td>Setting the level of detail of the simulation scenario</td></tr>
<tr><td>2.2</td><td>Definition of the Seed Value</td></tr>
<tr><td>2.3</td><td>Calculating measures of effectiveness</td></tr>
<tr><td>2.4</td><td>Reusability of models</td></tr>
<tr><td rowspan="12">Multi-run Execution Loop</td><td colspan="2">3. Design of Experiments</td></tr>
<tr><td>3.1</td><td>A series of tests to compare the input and output parameters</td></tr>
<tr><td>3.2</td><td>Replication of simulation runs</td></tr>
<tr><td rowspan="7">2. Grow the Data</td><td colspan="2">4. High Performance Computing</td></tr>
<tr><td>4.1</td><td>Data farmable model</td></tr>
<tr><td>4.2</td><td>A set of model inputs</td></tr>
<tr><td>4.3</td><td>A specification of experiment</td></tr>
<tr><td>4.4</td><td>A set of high performance computing resources</td></tr>
<tr><td>4.5</td><td>Data farming software</td></tr>
<tr><td>4.6</td><td>A set of model outputs</td></tr>
<tr><td rowspan="4">3. Harvest<br>via Data Mining</td><td colspan="2">5. Analysis and Visualization</td></tr>
<tr><td>5.1</td><td>Data processing</td></tr>
<tr><td>5.2</td><td>Statistical analyses</td></tr>
<tr><td>5.3</td><td>Results for the support of decision-making through answering the what-if questions</td></tr>
</table>

## 3.3 Discrete Event Simulation

As mentioned in **section 3.2**, data farming is based on the computer simulation. Therefore in this section the discussion topic comes to the discrete event simulation combined with the *procedure model of simulation study with V&V*. Separately, the V&V techniques and applications are outlined in detail.

### 3.3.1 Theoretical Background of Simulation

This section provides an elementary introduction of the computer simulation. Simulation is the most common descriptive modeling method and applied to many areas of decision making (Turban et al. 2011). The definition of simulation is very tied with the terms of system and model. *Model* is an abstract depiction of a system. A *system* is derived from Greek word σύστημα and means a collection of elements and their characters, which are connected each other by an interrelation. Normally the singular systems are too complex to be comprehended or controlled. Thus, a reduction of essential elements and system interrelation is on behalf of abstraction during modeling processes. This is an original idea of the term of *simulation*, which is defined as an imitation of a system, and its model is called *simulation model*. If this simulation model requires computer calculation, it is called *computer simulation* which Hedtstück (2013) terms as *digital simulation*, for which the model is described in a mathematical form and implemented in a set of computer programs, also termed as *simulation tools* (Eley 2012).

Described in detail systematically, Hedtstück (2013) defines *simulation* from the aspect of application as a technique for setting up a model of a real or imaginary system and being studied with experimental intention in order to obtain new knowledge about the system and derive the handling instructions. A *system* is an amount of *objects*, which could be correlative to each other and described by *attributes*. The *state* of a system is defined by system's objects and their correlations as well as every single attribute value. A system is normally embedded in the *system surroundings*, which could be understood as a summary of objects outside the system. An object can be classified in four types: An object is *temporary*, when it appears only occasionally, otherwise it should be *permanent*; An object is *stationary* or called as a station, when it always stays on the same site; An object is a *movable object*, when its site changes; But if its sit changes as general as well, then an object is called as a *dynamical object*. System can be typed into dynamic system and continuous system. A *dynamical system* has a change of states of a system. If this change is constant, dynamical system is *constant*. If the change of state occurs at the discrete point of the time, it is called as a *discrete system*. In the case that there is one point of time at least, when the state of a system is dependent on random factors which are not predictable, and then this system is defined as a *stochastic system*, otherwise it is defined as a *determinate system*. The state of a *discrete dynamical system* is resulted by the correlations of a range of objects. If a change of the state of a dynamical system happens at every point of time with positive real number, it reflects the facts that this system has a *continuous time parameter*. But if at discrete point of time, so it is called *discrete time parameter*. (Hedtstück 2013)

The change of a system state is called as *event*, thereby the quantity of objects in a system, an attribute value of an object at least or a correlation between objects is changed. An *event* is an incident, which takes no real time but is on the time level in a system. An event shows up at a

point of time which is referred to as *time stamp* of an event. An activity is a process between an event and its next following event. The time this process takes is termed as *real time* of an activity. An activity does not change a system state. A *determinate activity* has predefined time duration. If the end of an activity depends on the random variable, so it is called as *stochastic activity*. In general, an event could be from a complex nature, when it changes the state of different objects. An event could be resulted by several parts of the other events, which affect every single event and take place at the same time. Under this correlation an event is named as *conditional event*, when its entry time is as same as the entry time of the other events and depends on them. Otherwise, it is called as an *unconditional event*. A *process* refers to as a dynamical system, which follows a *processing logic*. This processing logic defines the quantity of the possible courses of the process instances. The processing logic decides, which state transitions probably are in the process, and is determined by object's attributes, connection between the stationary objects as well as special processing regulation. A process is termed as a stochastic process, when the random variables play a role in the change of a state. (Hedtstück 2013)

A simulation run or an *experiment* is expressed as an imitation of a system's behaviors with a model over a defined time interval, in which the model is carried out precisely only once when simulation running. This time interval, when the system is analyzed, is called as simulation period. However, simulation time depicts that the time has passed just as same as in the real system. Simulation time is distinguished to compute time, which means the required time for the system analysis over a defined simulation period. If the random numbers parameters are needed in the simulation model, it is necessary to run the simulation repeatedly. This interprets that the simulation model behaves always in a different way and outputs different results every time when the simulation model restarts running. If it refers to a determinate model, then it needs only one simulation run (Eley 2012). *Discrete event system simulation* contributes replaying all the events in the discrete processes, thereby each of event routines, which are dependent on the event type, is carried out by for each of events. An *event routine* is a program code which is a part of simulation software for calculating new states, planning new future events and implementing statistical evaluation. Under some conditions there are no new events to schedule, or the scheduled evens must be discarded and the new events have to be arranged. Event routines are only set up for the unconditional events and contain all the calculations that are necessary for the conditional events. The stochastic system is attributed by the random events whose point of entry time or characteristics is not predicable. Random events cannot be completely expressed by mathematic, and that's why stochastic is utilized. Possible random characteristics are summed up as *random variables*, which can be characterized by *probability distribution*. The most used statistical distribution is the normal distribution (Banks 2014). Normal distribution describes the symmetrical phenomenon and approximates the sum of independent random variables (Eley 2012). Computer is a determinate machine, where the random events are not left. That means the same data output the same results in the same program. Computer program is regarded as a *random generator* that can create random variables in sufficient quality in order to enable to analyze the stochastic system by simulation (Hedtstück 2013). Every modern program language provides a predefined random generator as a library function. The generating algorithm of all random generators is based on the recursive calculation:

$$X_i = F(X_i - 1) \qquad (\text{F. } 3.3\text{-}1)$$

The next random variable $X_i$ can be calculated with start value $X_i$ from the previous random variable $X_{i-1}$ by a certain function F with the same start values the same sequence of numbers are

generated. Only the finite different numbers can be presented on a computer, so an already calculated number shows up again after a finite step, because from this number the sequence of numbers repeats themselves periodically. The sequence of numbers computer generates are termed as *pseudo random variables* in the determinate way. There are two preconditions for generating random variables with computer. The random variables must be independent and capable of sufficing every probability distribution. Hedtstück (2013) provides an overview of methods for generating random variables.

### 3.3.2 Procedure Model for Simulation Study with V&V

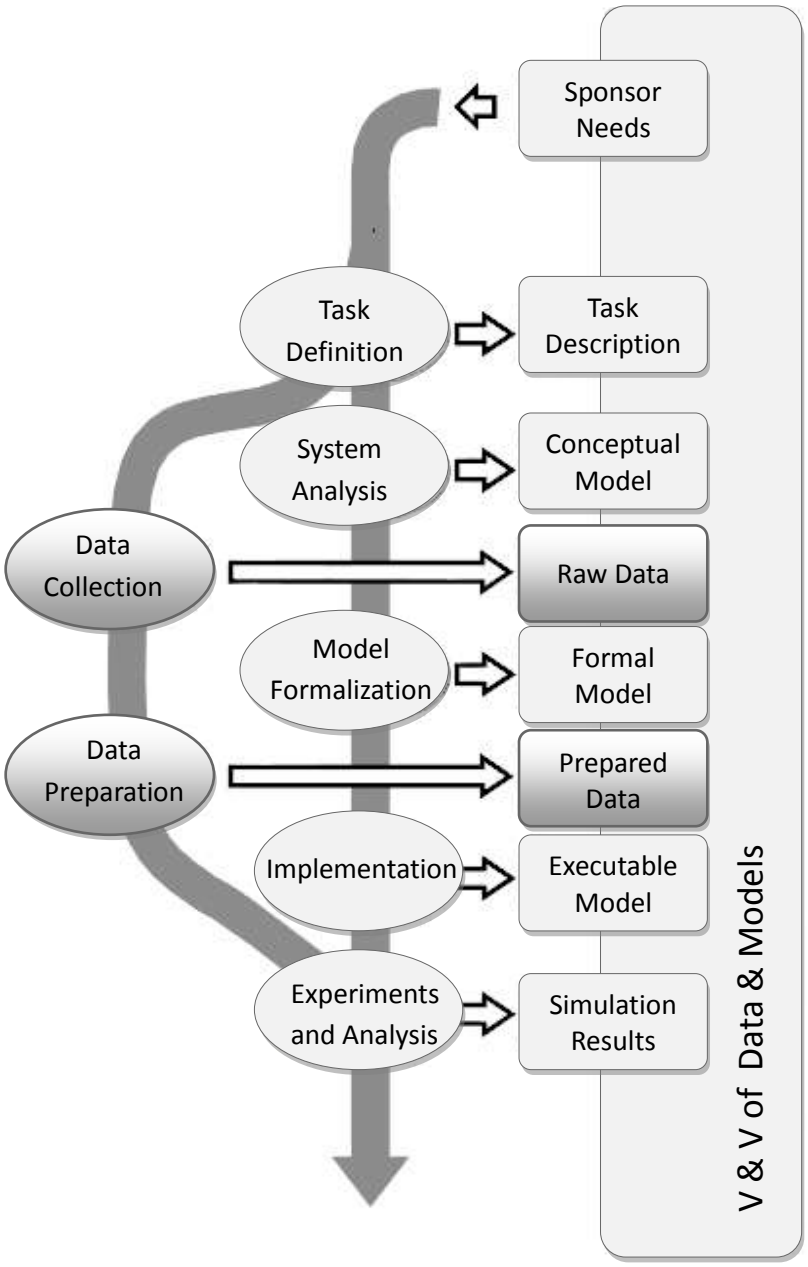
Following it will introduce a *procedure model for simulation with V&V* (**Figure 3.3.1**), for enhancing successful results step for step, the V&V accompanying the simulation processes can provide a robust simulation result (Rabe et al. 2009). On the other hand, simulation cannot accomplish the optimization task without using statistic, mathematics or computer science which also get V&V involved at last (Rabe et al. 2009; Eley 2012). Procedure model for simulation with V&V which is proposed based on a guideline of the German engineers' association VDI (VDI 2009). The principle of this procedure model is that every phase results must be validated and documented before starting with the next procedure phase. On the other hand, specific V&V activities are indispensable within each single phase of the modeling process. The phase results can be simulation models and documents or in a combination form of the both separately. The procedure model starts with "sponsor needs" which is rather than as the offer handling activities for purchasing a simulation project in the business sense. Therefore, this step can be set up individually by the different potential project owner. Consequently there are no results of the simulation study to summarize. "Sponsor needs" documents the initial requirements from the project owner. The contents of the documents deal with the basic disagreement between the project partners and enable the latter supplementary by all the relative project responsibilities.

The first phase begins with tasks definition which should supplement the information that is not issued in the sponsor need as well as the specifications sheets. The results of this phase are task description and should describe the concrete and appropriate tasks and the parts of the project contents which already have been checked out if the issued tasks are possible to be carried out or how to be carried out. Especially, the crucial business issues about budget and time line are settled under the condition of the disagreement between the project owner and project contractor.

The second phase comes to *system analysis* which results the conceptual model. In this phase it will define the model mechanism and the complexity level about system delimitation which need to be weighted and estimated if it will form the expected model. The documents of this phase contents the aim, input, output, elements and their relationship.

Between the second and third phase, there is a flow of *data collection* which deals with the raw data and can be done as the modeling phases proceeding. These raw data are gathered based on the results of the second phase, task description, from the different real systems in the variety format and need to process in order to satisfy the requirements for the next phase. But, before the formal model comes out, the raw data have to be further processed as completely as the prepared data need for the executable model.

**Figure 3.3.1. Procedure Model for Simulation Study with V&V**




Source: Rabe et al. (2008)

The third phase deal with *model formalization* that is developed based on the conceptual model and start with the simulation. In this phase, there are only the technical issues about the simulation implementation without business discussion any more. On the other hand, the formal model should be described independently without using the simulation tools. This brings a conflict partly, because already in the second phase the some details about formal model have got involved or the considerations about the simulation tool are already taken. But due to V&V, it is necessary to do so in order to examine the formal model.

Before the fourth phase begins, another data flow comes again and is defined as *data preparation*, namely the transformed data for the coming phase of executable model. As aforementioned, the prepared data are processed based on the raw data and actually the input data for the executable

simulation model. According to the expected simulation results, these data can be transformed in a proper format respectively as well as statistical distribution.

The fourth phase is *implementation* which handles with the executable model, namely, run a visual simulation model. The preconditions of this phase are the validated input data, hardware and software for running a simulation model.

The fifth phase is *experiments and analysis* and gains the simulation results as following steps:

- Setting up the experiment plans and hypotheses about the systems which is defined to analysis
- Implementation, documentation and explanation of the experiments.
- Analysis of the experiment results and causality among the input parameters.
- Draw a conclusion from the hypothesis test for a real system

### 3.3.3 Verificaiton and Validation

In the VDI guideline, verification is defined as the formal proof of a simulation model's correctness. Applying this in software development, verification is to test if the created simulation program of the connectional model is corrected reproducible and to provide a proof of the consistency between the program implementation and its specification. The general understanding of verification is "Are we creating the X right?" Rabe et al. (2008) define "Verification is to test, if a model was transformed from a descriptive art in another descriptive art." The verified model should be accurate. In practice, the verification is so often only handled as a program codes test. According to VDI guideline, validation is to test if the model is sufficiently acceptable to represent the original system. Another common expression of validation is "Are we creating the right X?" Rabe et al. (2008) define "Validation is the continuous test if the model reproduces the behavior of the depicted system sufficiently and precisely" and provide an overview of the V&V techniques across a simulation study as a project. Here only introduction to the selective V&V techniques which will be used for the implementing the conceptual approach in **chapter 4**. is given. The reasons are that firstly the contents of this work are limited, secondly this thesis will implement a case study, not a project, for which the most V&V techniques deals with the discussion among the project partners.

The first is *animation* which is carried out by running a 2D or 3D visual model in a computer for checking out the validity of the model behavior. This technique is effective to find out the failure of program structure and logical relationship between the model elements and their parameters and supervision whether the model proceeds in plausible time instance as well as the real system. Disadvantage of this technique is that the model behavior beyond the simulation time cannot be supervised. Therefore the test results can only be accepted in a defined simulation time. The second is *desk checking*. This technique is a subjective test carried out by the model builder who checks his work through again in terms of the completeness, correctness, consistency and clarity. Disadvantage of this technique is not easy to find out the self-made failure. The third is *structured through* which relates to the phase results and their logical relationship especially when the *executable model* will be carried out. This technique can also be used in the modeling phase of *system analy*sis and *model formulation*. The last on is the statistical test (Rabe et al. 2008). Eley

(2012) recommends the *chi-squared test* method for validating the simulation output data in the logistics field. The test method itself belongs to the group of the test distributions and should draw a conclusion, how suitable the observed frequency distribution of a nominal variable matches an expected frequency distribution. A *sample* is from a population with unknown distribution function F(x) and an expected distribution function is defined as $F_0(x)$. *Chi-quadrate test is* to test if the alternative hypothesis should be rejected or disproved. Translating this theory in validating the simulation output data is comprehended in this way: *Null hypothesis* $H_0$ is that the frequency distribution of existing validated simulation $F_0(x)$ is equal with the frequency distribution of the validated output data to be validated output data F(x), expressed in the way of $(H_0)$: $F_0(x) = F(x)$. Otherwise, it is alternative hypothesis $(H_0)$: $F_0(x) \neq F(x)$. Rejecting or disproving a hypothesis is decided by significance level $\alpha$. If $H_0$ is not to be rejected, the simulation output data can be disproved for the further generating data or accepted as real data in sufficient quality for data analysis (Siegel 1988). One way in which a measure of goodness of fit statistic is *Pearson's chi-squared test* and can be constructed as follows:

$$x^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \qquad \text{(F. 3.3-2)}$$

is distributed under $H_0$ asymptotically (for $n \to \infty$) $x^2$ with v freedom grads

n = number of bins of the sample size k,
$O_i$ = an observed frequency (i.e. count) for bin I, and
$E_i$ = an expected (theoretical) frequency for bin i, asserted by the null hypothesis.

As long as the test result: $x^2 > x^2_{v;\,1-\alpha}$ with $v = k - 1$, $H_0$ is not to be rejected. Under condition of $x^2 > x^2_{v;\,1-\alpha}$ with $v = k - 1$, the expected frequency is calculated by:

$$E_i = (F(Y_u) - F(Y_l))\, N \qquad \text{(F. 3.3-3)}$$

where:

F = the cumulative distribution function for the distribution being tested,
$Y_u$ = the upper limit for class I,
$Y_l$ = the lower limit for class i, and
N = the sample size.

## 3.4 Tools for Simulation and Data Analysis

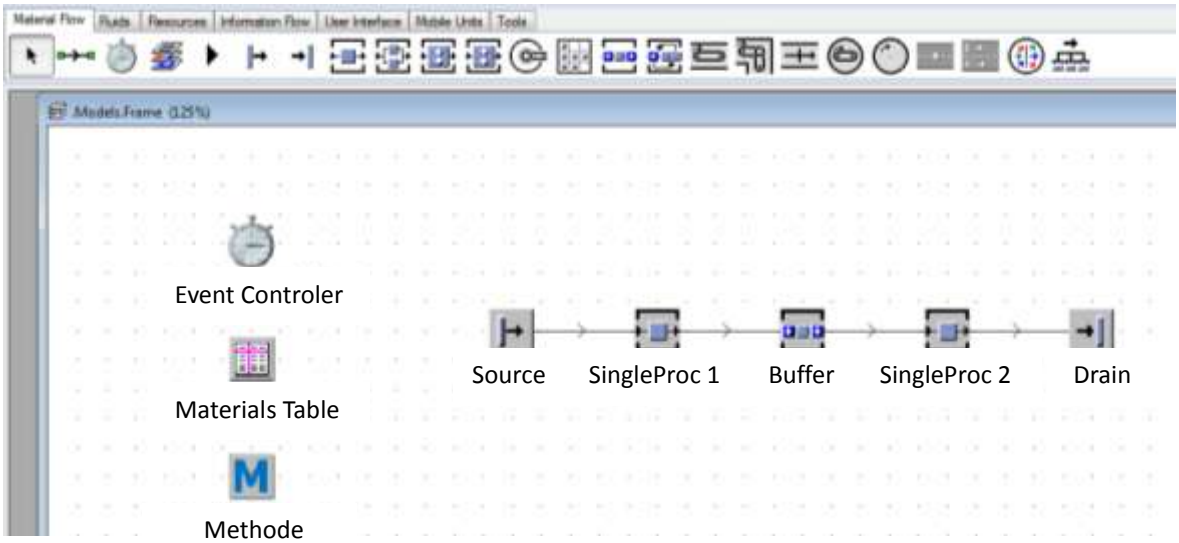
To implement the tasks of this thesis, there are two software tools to be applied. Tecnomatix Plant Simulation for expanding the existing simulation model. For analyzing the simulation output data, the RapidMiner will be used. Both of the software tools are the most applied in the academic studies and domain specifications with a wide range of the functionalities. However, only the part of the tool elements, which this thesis needs are to be presented.

**Tecnomatix Plant Simulation**

Plant simulation is a material flow simulation software developed by Siemens for modeling and simulating production systems and processes, the flow of materials and logistic operations. Tecnomatix Plant Simulation enables optimization of the material flow, resource utilization and

logistics for all levels of plant planning from global production facilities, through local plants, to specific lines. The application allows comparing complex production alternatives, including the immanent process logic, by means of computer simulations. Plant Simulation is used in a wide range of industries, especially in the Automotive Industry Workgroup Material Flow Simulation. Plant Simulation is a discrete, event-controlled simulation program and only inspects those points in time, at which events take place within the simulation model. The basic elements for creating a simulation model as follows.

**Figure 3.4.1. Work Panel of Tecnomatix Plant Simulation**




- Entity: An object, that changes its location during the simulation run since it entries, is a *movable* element in the German verb. Entity symbolizes the e.g. work piece and transport container and *temporary* element which can be created by source and destroyed by drain.
- Resource: Objects are unmovable during the simulation run. Therefore they are permanent elements and depict e.g. machine and *SingleProc*. The work status of resource can be classified into available or unavailable.
- Queue: An object deals with a kind of resource for storing the movable elements, when the resource is not available. Queue represents like buffer and follows FIFO and LIFO principles.
- Attribute: Attributes are the behavior characters of the entities and resources, and their values can be checked and chosen. For example, if a queue is not available, the value of the quer can be select as *true*.
- Method: Methods are the control program by using SimTalk program languages to release the commands how the relative objects should behave under some certain conditions during the simulation runs. With methods the attributes value can be changed and manipulated, as well as new attributes can be created.
- Table: A table can be treated as storage where the input data and output data are collected. The value of an attribute can be read and written automatically by programming a method.
- Variables: The information and data can be stored in the variables and used during the simulation run. There is a wide range of the utilizations of variables can be manipulated by programming the methods in order to control the simulation processes.

- Random numbers variant: Random numbers variant is in the event controller and creates random processes by the statistical distribution. Especially it is to notice that the different random numbers variants create different random processes.
- Event controller: Event controller coordinates and synchronizes the events, which take place during the simulating run. Event controller also enables to start, stop and reset the simulation processing as well as define the simulation duration and random numbers variant.

The *Source* produces the parts that the stations, symbolized by the *SingleProc*, are going to process. The *Drain* removes the parts, symbolized by the *Entities*, from the production line after the SingleProc has processed them. The Source can represent the receiving department, while the Drain can represent the shipping department. A *chart* can also be inserted that visualizes the results of our simulation run in different ways. *Connector* in the Toolbox is to activate connect mode, when it is located over an object.

**RapidMiner**

RapidMiner is a statistical software application developed by the firm of RapidMiner and provides an integrated environment for machine learning, data mining, text mining, predictive analytics and business analytics. It is used widely for business and commercial applications, rapid prototyping, and application development. Especially, RapidMiner performs well and popular in implementing all steps of the data mining process including data preparation, results visualization, validation and optimization. For accomplishing one of the tasks of this thesis, the relative tool elements and their functionalities are selected to introduce. First of all, **Figure 3.4.2** shows the basic elements of the work panel where a data modeling process can be created. In addition, **Table 3.4.1** provides the descriptions of the introduced elements.

**Figure 3.4.2. RapidMiner Elements**

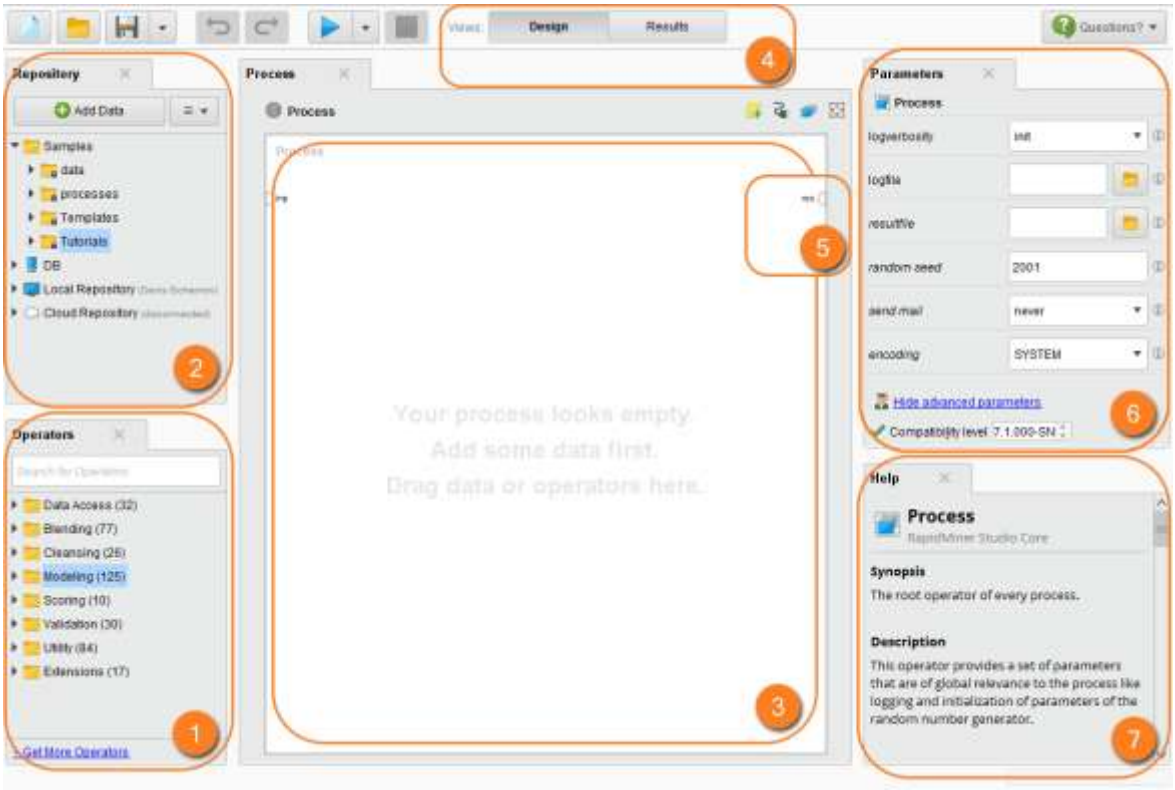
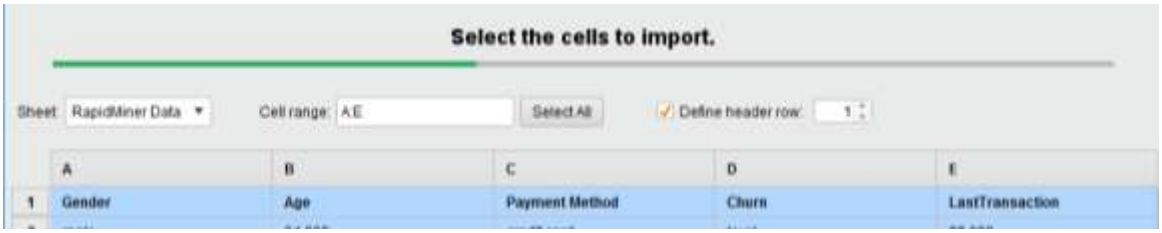


**Table 3.4.1. Description of RapidMiner Elements**

| Number | Name | Description |
|---|---|---|
| 1 | Operators | Building blocks used to create data modeling processes. |
| 2 | Repository | Data storage within RapidMiner Studio for data modeling processes. |
| 3 | Process panel (Main process) | Working area for creating data modeling processes. |
| 4 | Views | Work area for accessing specific functionality |
| 5 | Ports | Interfaces for input and output among the connected operators and processes. |
| 6 | Parameters | Settings that modify process operator's functionalities. |
| 7 | Help | Descriptions of the functionalities of the selected operator. |
| 1, 2, 3, 6, 7 | Panels | Tools available to a view. |

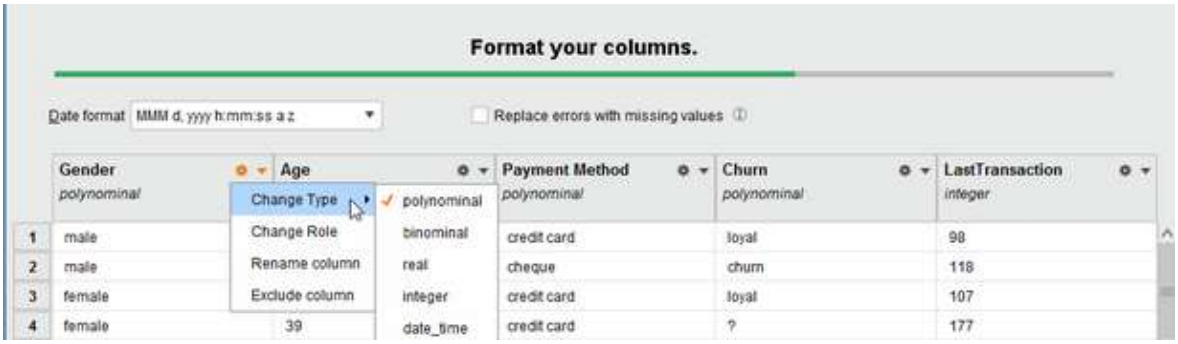
1. Importing data: Using the import wizard is able to import the data set into the repository. Then a range of cells for import which are necessary for the data mining are selected.

**Figure 3.4.3. Importing Data in RapidMiner - 1**



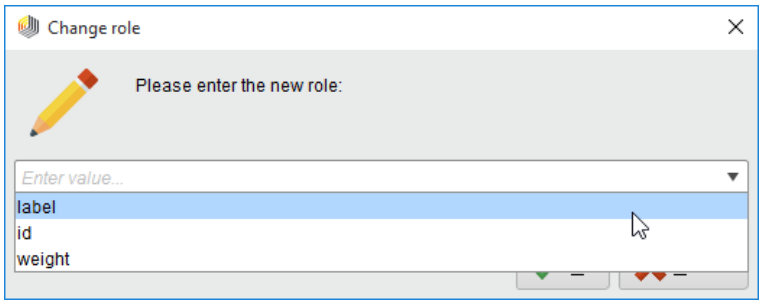

The words, in a dialog window below the column names, define the data types for each attribute. The data type specifies the values for an attribute as polynomial, numeric, integer, etc.

**Figure 3.4.4. Importing Data in RapidMiner - 2**



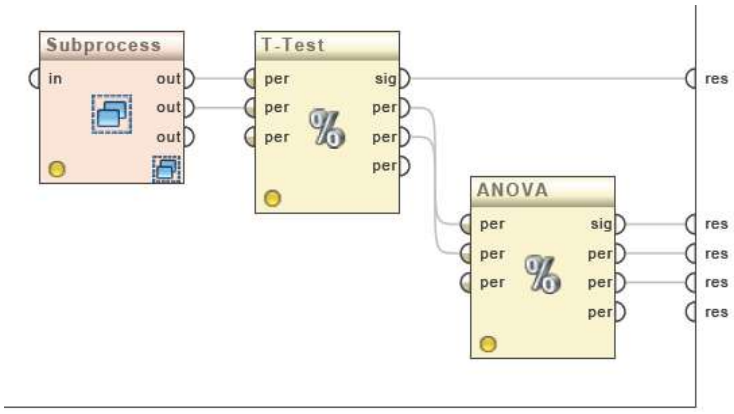
The values in the row are to identify the attribute for modeling. Here is a term *churn* in RapidMiner for labeling the target attribute, from which the rules or attributes behavior should be predicted or learnt. Therefore, it needs to set that column's role to *label*. RapidMiner will use the other attributes for learning to predict a value for each missing label, i.e., to classify the example. There can be only one label per data set.

**Figure 3.4.5. Importing Data in RapidMiner - 3**




2. Data Visualization: At this step the segments of the results view are available for data visualization by using results tabs, data filter, data screen, statistics screen and charts screen. Each screen provides results in a particular format for the relevant object.
3. Creating Model: This process is encompassed by the following steps.
   - Retrieve the data: Drag the imported data set onto the *process panel*. RapidMiner uses the *retrieve* operator to incorporate the data.
   - Filter out examples with missing labels, if this situation occurs.
   - Add a data mining algorithms operator such as a decision tree and clustering that can be found by search dialog.
   - Save the process, if it is convinced that will gain the target result model.
4. Applying a Model: If the data mining algorithms are supervised like decision tree, the model created at the step 3 should have been resulted by training data and this step is necessary except for the clustering algorithms and association rules analysis.
5. Evaluation: At this step, it also divides into the evaluation of supervised algorithms and unsupervised algorithms. For the supervised algorithms, the validation operation such as x-validation, cross-validation and split validation operator are available. For the unsupervised algorithms clustering, the operators such as cluster distance performance and cluster density performance can be applied. RapidMiner also provide the statistical test operator like ANOVA to enable the comparison of more than two models.

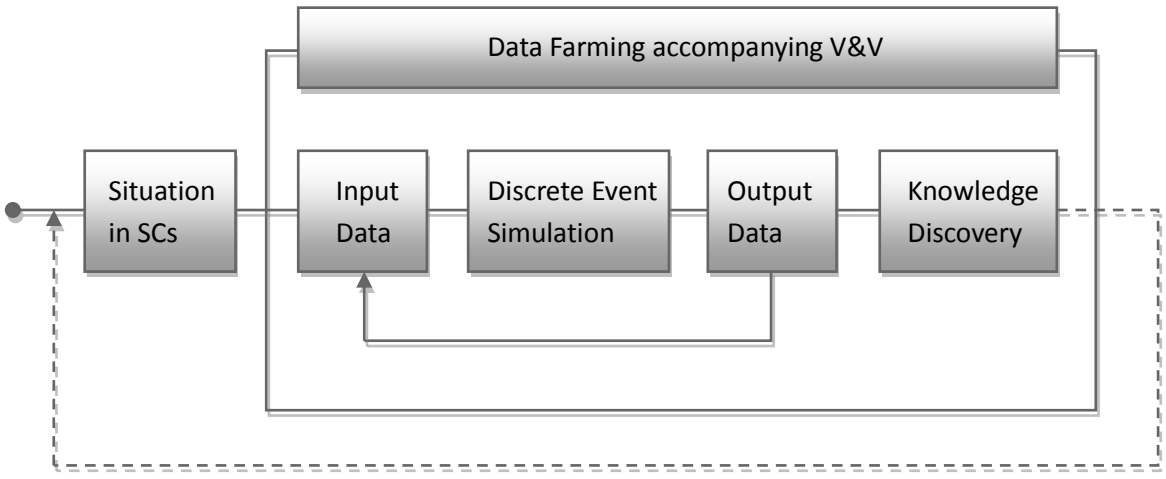
**Figure 3.4.6. Evaluation with RapidMiner**



# 4 Conceptual Approach to Knowledge Discovery in Supply Chain Transaction Data by Applying Data Farming

This chapter will develop a conceptual approach to knowledge discovery in supply chain transaction data by applying data farming step by step based on the afore introduced theories, relating to the current situation in the field of the automotive SCs (**section 2.1**). As noticed, there are some similar and common steps between the *data farming "Loop of Loops "* (**section 3.2**) and *Procedure model for simulation study with V&V* (**section 3.3**), i.g. the similar part: *Data farming* starts with the "What-if question", while *Procedure model for simulation study with V&V* with the "Spousor needs", and the common part: Both of the two procedure models end with the results analysis by using the statistical test methods of *knowledge discovery* (**section 2.2**) and drawing a conclustion for solving the domain question. For implementing this conceptual approach, it needs make a compromise and decide a clear guideline to proceed the development steps. Therefore, as this thesis will adopt the *procedure model for simulation study with V&V* as the framework and selective elements of data farming (**Table 3.2.1)** as the support application. These JIS delivery processes rules can be discovered in the SC transaction data by using data mining methods (**section 2.2; section 3.1**). Based on the all relative aspects of the situation analysis, the simulation input data are collected, while the conceptual model being established (**Figure 4.1**). After each simulation run, the output data have to be validated comparing with the input data by the proper statistical tests. If the output data are accepted by the statistical tests, they can be used further as the input data for the next simulation run, after a certain simulation runs calculted by the confidence interval methods (**section 3.2**). For drawing a conclusion, the results analysis is implementated by the algorithms of *knowledge discovery*. The entire procedure of data farming accompanies V&V according to the procedure model for simulation study with V&V. The dotted line means the application of the discovered knowledge in SCs, so that it builds a knowledge generation cycle which can be used for further research.

**Figure 4.1. Knowledge Discovery in Supply Chains by Applying Data Farming accompanying V&V**



## 4.1 Expansion of the Existing Simulation Model

As afore discussed, this approach follows the *procedure model for simulation study with V&V* as the guide for implementing a case study. Instead of the part “Spousour needs” which is used for the project purchasing, it will start with “What-if question”. Subsequently, the phases of the anaysis and expansion of the exsiting simulation model are presented in detail. Finally, the V&V on the output data is discussed.

### 4.1.1 Analysis of the Existing Simulation Model

This section is the first task of this thesis and also handled as the first part of the second modeling phase *system analysis*. The result of this section should be a component of the *conceptual model*. Because of the limitation of the contents capacity, the analysis of the existing simulation model only puts the focus on the model scenario and input parameters.

**“What-if question” in the automotive SCs**

According to **chapter 1** and **section 2.1**, one of the main reasons that today’s supply chains are not able to derive real benefits from the knowledge effectively is the insufficient information quality during the EDI processing in real time which are resulted from the inhomogeneous SC IT landscapes and operational logistics processes. On the other hand, the SCs are processing in a dynamical changing way: the final customers change their orders in a short term, or the assembly lines at VMs reschedule their assembly plans. This can lead to the disturbance of the JIS delivery. If the JIS delivery processes are disturbed, the corresponded *assembly orders* will fall out at the VM and even cause the chain reaction that the delivery to the final customer is delayed. Thus the “What-if question” in this case should be “with what kinds of performance will a JIS delivery process behave, if this delivery is rescheduled” Segmenting this SC question into details, creates an orientation on the *task definition*.

**i. Task definition**

The *task definition* is the first phase of the procedure of the simulation study. In this phase, the “What-if question” will be formulated correctly in details. The first task is to analyze the existing simulation model which can be treated as *base senario*. The relative documents about input parameters, tests and reuasability of the model are especially import for the model expansion, and to define the input parameter relating to the “What-if question”. The second task is expanding the existing simulation model which starts with the relevant aspects of the JIS delivery disturbance as introduced in the **section 2.1.2**. Based on the time expanse and results accuracy, it has to set the level of detail of the simulation scenario which should be carried out in the phase of the *system analysis*. This needs to define the input data attributes as seed values of the simulation model. Another important issue is that the expanded model should be reusable to generate the data further. Furthermore, the proper statistical methods are chosen for testing the output data and calculating the simulation runs in terms with *design of experiments*. The third task is the executable model which only can be accomplished with some preconditions, as listed in **Table 6**. The result of the task definition is the *task description* which will be explained in the coming sections.

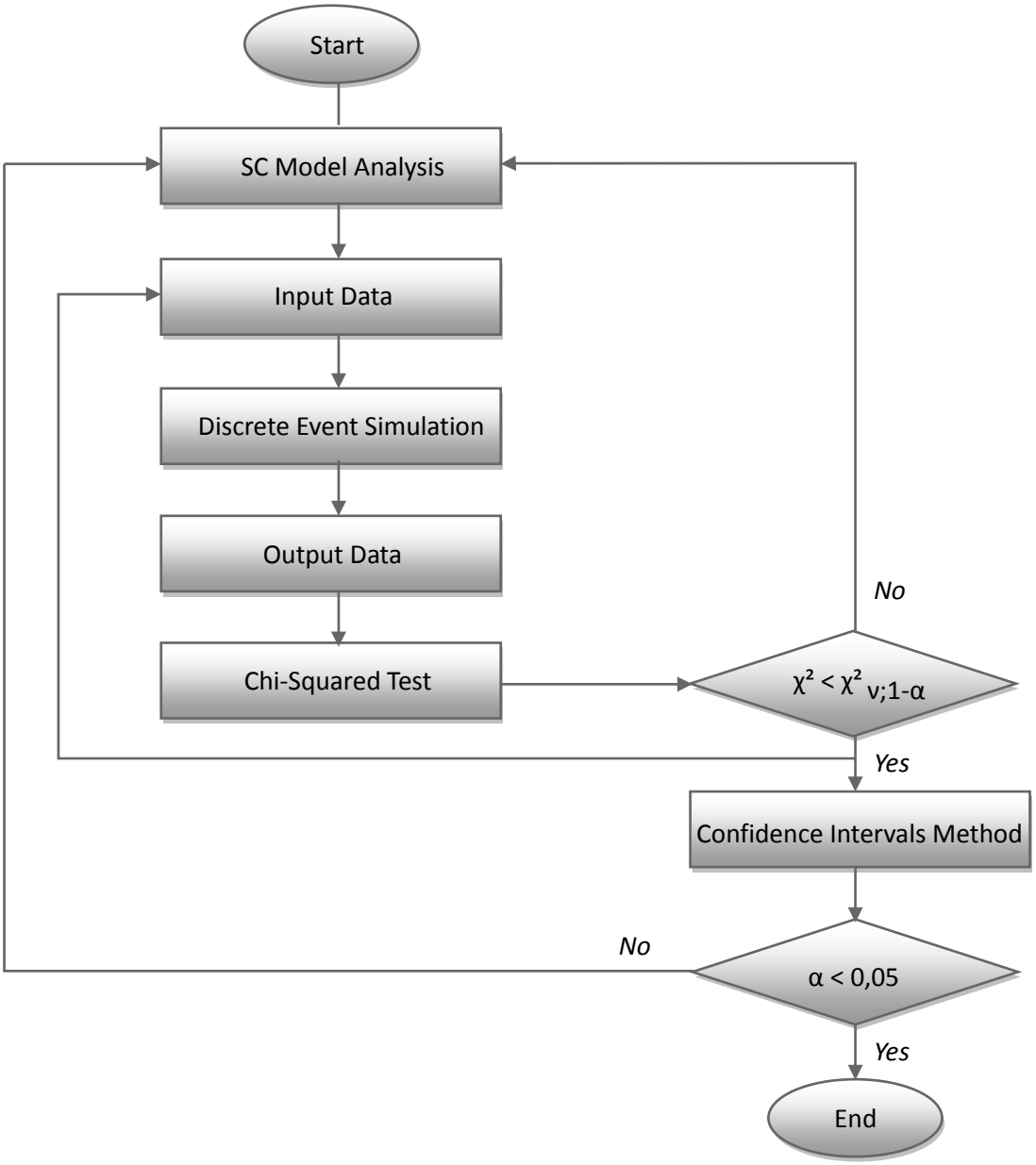
**Table 4.1.1. Task Definition**

| | Task Definition | Data Farming Elements (Table 5) | Task Description |
|---|---|---|---|
| 1. Plant Data | 1. Analysis the Existing Simulation Model | 1.1 Combination with model development which is a tested and documented base case scenario as output | Analysis of the existing models and documents |
| | 2. Expanding the Existing Simulation Model | 1.2 Implementation of all relevant aspects of a scenario into a suitable simulation model in the context of a question-based analysis | According to the section 2.1.2 and analysis results of the existing models and documents |
| | | 2.1 Setting the level of detail of the simulation scenario | According to the analysis results |
| | | 2.2 Definition of the seed value | According to the analysis results |
| | | 3.1 A series of tests to compare the input and output parameters | Qui-squared test |
| | | 3.2 Replication of simulation runs | Confidence intervals method |
| 2. Grow the Data | 3. Executable Model | 4.1 Data farmable model | By running the model |
| | | 4.2 A set of model inputs | According to the analysis results |
| | | 4.4 Computer software and hardware | ASUS F201E |
| | | 4.5 Data farming software | Tecnomatix Plant Simulation Version 12. |
| | | 4.6 A set of model outputs | After each simulation run |
| 3. Harvest | 4. V&V on Output Data | 3.1 A series of tests to compare the input and output parameters | Calculating by qui-squared test |
| | | 3.2 Replication of simulation runs | Calculating by confidence intervals method |

The **Figure 4.1.1** provides a procedure program of the expansion of the existing simulation model. The program begins with the SC model analysis which results in the setting up of the input parameters. After each simulation run, the output data are validated by the chi-squared test. If the sum of the $x^2$ value bigger than the 5% critical value, the output data will be accepted as valid data and can be input further as *data seed* for the next simulation run. If not, then it has to go back to

the SC experiment and adjust the input parameters. Data farming needs to fix the certain simulation runs when the output are sufficent to satisfy the requirements of the data quality. By using the confidence intervals method, the cumulative calculation of the mean value and their standard deviation are implemented after each V&V on the simulation output. If the cumulative significance level bigger than 5%, data farming will continue. If not, data farming will begin with the final *harvest* which means the phase of output data transformation and analysis with knowledge discovery algorithms.

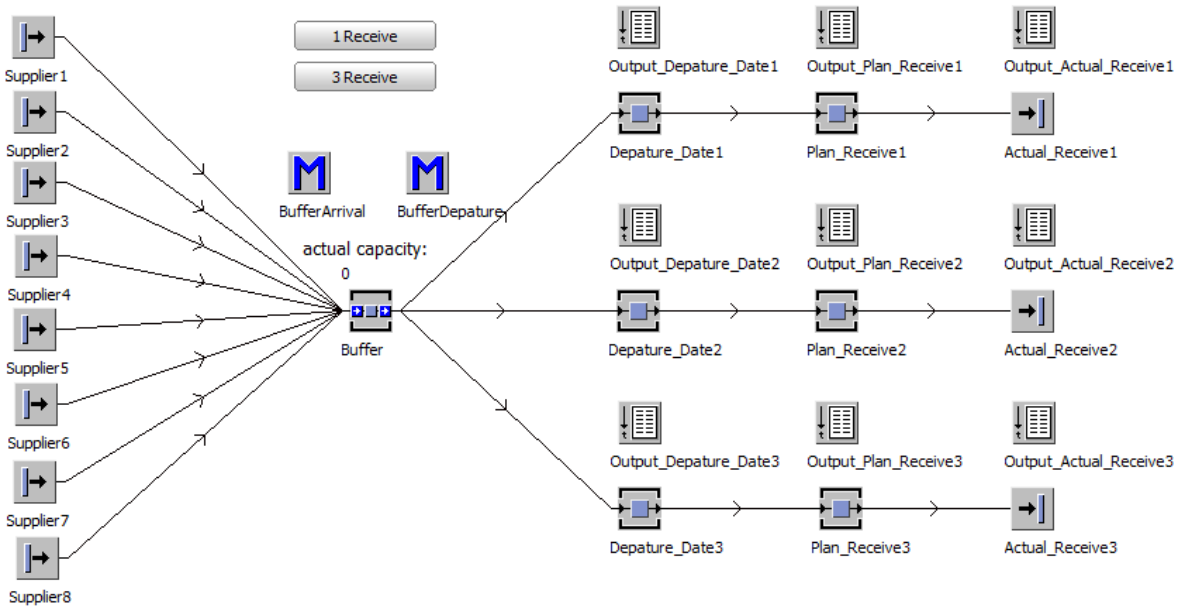
**Figure 4.1.1. Program of the Expension of the Existing Simulation Model**




Arndt (2014) and Baydar (2015) created the simulation model (**Figure 4.1.2**) about the delivery processes between the 1TSs and a VM in the automotive industry. The motivation of setting up this simulation model is that the insufficient data quality during the information sharing among the SC partners is not able to support for making the right decisions in SCs. If the material demand information from the VMs are not shared with their 1TSs in real time, that can cause the bottle

necked situation and delayed delivery to the final customers. On the other hand, the SC transactions data with the good quality can help to make an accurate forecast on the further material demand (**section 3.1**). Therefore, the information quality impacts on the performances of the SCs heavily. For solving this problem, it can use the simulation technique to generate the transaction data with the advantage of the economical efforts.

**Figure 4.1.2. The Existing Simulation Model**




Arndt (2014) sets up the original model for generating the SC transaction data which are characterized as the time stamp. The input parameters are real data from the B2B networks in the automotive industry and collected as well as prepared based on the study goal and task definition. The input data are transformed in the statistical distribution as a data generator in an executable model. The input data are the *delivery distribution* in the form of normal distribution, *delivery duration* and *delivery deviation* in the form of discrete empirical distribution. This original simulation model is created by the software tool Tecnomatix Plant Simulation Version 11 by using the elements such source, single processes and drains which are described in details in **section 3.4**. In this model, there is one source, named as *Delivery_Departure*, three single processes, named as *Departure_Date*, *Plan_Arrival* and three drains, named as *is Actual_Arrival*. The orders are created at the *Delivery_Departure* by the parameter of the delivery normal distribution so that *Departure_Date* can be created at the *event controller* which sets up the simulation time. The input parameter table of the "Delivery_Duration" which is described by the empirical discrete distribution is inserted at each *Plan_Arrival*, and then the data of the *Plan_Arrival* can be generated and written in the each corresponding *TimeSequence "Output_ Plan_Arrival"*. By inserting the parameter table "Delivery_Deviation" in the format of empirical discrete distribution at each drain *Actual_ Arrival*, the data of *Actual_Arrival* can be generated and also written in the each corresponding *TimeSequence "Output_Actual_Arrival"*. The simulation time is set to 938 days and the entire simulation model creates 121 deliveries. The maximal *delivery duration* is 8 days and the maximal *derivation delivery arrival* is up to 6 days. Therefore the maximal *delivery lead time* is calculated as 14 days. From the statistical simulation results, there is a fluctuation

period which reflects the accuracy of the simulation results. This relates to the input parameters and is acceptable, if the results are validated and not rejected by the hypothesis test.

**Table 4.1.2. Information of the Existing Simulation Model**

| | **Arndt (2014)** | **Baydar (2015)** |
|---|---|---|
| **Depicted Systems** | ▪ Delivery processes between 1TS and a VM in the automotive sector | ▪ Delivery processes between 1TS and a VM in the automotive sector |
| **Input Data** | ▪ Delivery normal distribution<br>▪ Delivery-duration<br>▪ Tolerance-delivery-arrival-date | ▪ Quantities-per-delivery-dEmp<br>▪ Quantities-per-delivery |
| **Output Data** | ▪ Delivery departure date<br>▪ Scheduled-delivery-arrival-date<br>▪ Is-delivery-arrival-date | ▪ Arrival-quantity-distribution |
| **Visualization with Siemens Plant Simulation** | ▪ 2D<br>▪ Vision 11.<br>▪ Student license | ▪ 2D<br>▪ Vision 12.<br>▪ Student lichens |
| **Level of Detail** | ▪ Delivery date<br>▪ Delivery departure at supplier<br>▪ Delivery arrival at VM | ▪ Delivery quantity<br>▪ Delivery departure at supplier<br>▪ Buffer capacities control<br>▪ Delivery arrival at VM |
| **Interface** | ▪ MS Excel for input and output | ▪ MS Excel for input and output |

Baydar (2015) expands the model based on the Arndt (2014) with the delivery quantity parameters and adds 8 suppliers which share a common buffer together. This idea is based on the *consignment stock* design which is popular used in the automotive sector in order to reduce stock cost and keep the safety stock level. Every supplier has the similar *delivery normal distribution* to the Arndt (2014)'s model for creating the deliveries. For creating the delivery quantities the three information tables are established. The table "Quantities_Per_Delivery" contains the input data about the empirical discrete distribution at the *Supplier_X*, containing the name of the delivered items and their corresponding delivery *frequency*. The table "Quantities_Per_Delivery" contains name of the items and their corresponding delivery quantities. By programming control methods *order assignment* and *arrival at VM*, the attributes in these two tables are operated when the simulation runs, and collected in the table "Arrival_Quantity_Distribution" which contains the arrived items name and quantities automatically when the executable model stops. **Table 4.1.2** summarizes the information about the existing simulation model which gives an orientation on the system analysis and setting up the level of the model detail for expanding the model.

This section can also be treated as a part of *system analysis*. Therefore, V&V technique for this section is to take *structured walked through* and the information in **Table 4.1.2** will be input in the next procedure phase and help set up the conceptual model (**section 3.3.3**). The description of this section can be handled as the document.

### 4.1.2 Expansion of the Exisiting Simulation Model

According to the task description (**Table 4.1.1**) this section lays the focus on expanding the simulation model due to the "What-if question" in SCs. Because of the limitation of the contents this section implements the phases of conceptual model, raw data, formal model and prepared data in a summarized way. Therefore, some of the arguments are not processed in detail, but more attentions are put on the information of the formal model which affects on the executable model directly and heavily. The reason for this handling way is that the focus of this thesis is to generate the transaction data by using DES and then analyze these data so as to classify the JIS deliveries in regular and disturbed processes.

Expanding and designing the simulation model begins with the scenario of the material flow following the JIS principle. As discussed, the items that the 1TSs delivery are *built to forecast* however, because of the special characters of the automotive sector, the 1TSs normally are obligated to keep a safety stock level in terms of the contracts. According to the current situation about the geographical distribution (**Figure 2.1.3**) of the 1TSs in the German automotive sector, the expanded model is built with four international 1TSs, namely supplier 1, supplier 2, supplier 3 and supplier 4, from e.g. Asia or South America and the East European countries. Therefore, they need to store their items at the common VMI stock in a sufficient quantity as forecasted and assigned for the *fixed assembly orders*. The regional 1TSs which locate not far away more than 50 km to the assembly plant of a VM, can deliver their items directly from their own stocks.

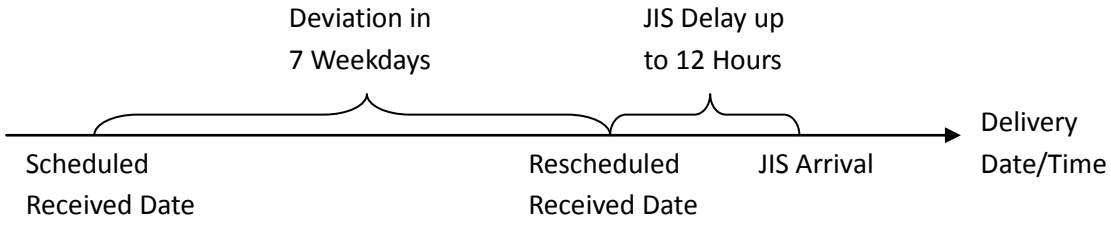
**Table 4.1.3**. **New Attributes of the Expanded Model**

| **1. Stage** | **2. Stage** | **3. Stage** | **4. Stage** | **Attributes** |
|---|---|---|---|---|
| **Procurement** | **Master Data** | Supplier Data | Delivery Address | Assembly Line |
| | **Moving Data** | Order Data | Time Stamps | Scheduled Received Date |
| | | | | Rescheduled Received Date |
| | | | | JIS Arrival Date |

The attributes which relate to the delivery rescheduling in a short-term are the delivery date and delivery quantity. In order to measure the performance of the JIS delivery process, it will take *on time delivery reliability* (**section 2.1.4**). However, this thesis decides to concentrate on the rescheduled received date. Firstly, there is the correlation between the delivery date and delivery quantity and it needs a mathematical description. For example, if the JIS items A from the supplier 1 for the *fixed assembly order* delayed or failed, the other correlated items B, C and D from the other suppliers for the same *fixed assembly order* from the suppliers have to be canceled. This argument proposes another question: why the other two important logistic processes

performance indicators such as *quality* and *price* will be not taken into account. Firstly, the quality are assessed or tested when the JIS delivery arrives, but that possibly will be found out while assembly lines are processing or several weeks. Secondly, in the logistics processes the delivered items can be damaged in the case of transport and transit operation, but the damaged items are going to be assessed as quantity reduction, but not as a quality problem. About price issues, they are rather than addressed at the SC strategy and execution stages, but do not come to operational delivery processes which are taken account into the SC operation tasks. On the other hand, because of the single sourcing strategy and unique technological specification, the automotive VMs have enormous dependency on the 1TSs and the price fluctuation in a short-term vision could reduce profitability from the VM's side, but not or barely influence on the procurement processes. For now, no research publications outline that the price fluctuation is taken as a direct cause of delivery processes disturbance in the JIS case to a VM. The further attributes that could have influence on the delivery disturbance could be the days in a week like Monday, Thursday as so on, because of the rush hours during the JIS transportation. It also can be the assembly lines at the VM, because each assembly line has individual assignments of assembly orders in terms of the certain configuration parameters (**section 2.1.2**). Therefore, the new attributes of the expanded model are listed in the **Table 4.1.3**.

**Figure 4.1.3. Rescheduled JIS Receive**




Because this thesis aims to discover the relationship between the rescheduled received data and JIS delivery performance, but there are no real data of the rescheduled deliveries available. As discussed in the **section 3.2**, in the case of absence of the real data, it needs to ask the options from the experts of the corresponding field. For this thesis, the input parameters are collected from the research publications as mentioned in the **section 2.1.2** respectively. Therefore, in this thesis, there is no result of *raw data*. The scheduled received date in the preview period could be rescheduled within the *delivery lead time*, e.g. in the interval 6-18 days as concluded from the simulation study and the expanded model will take the interval 0-7 days, and make an assumption that almost 76% deliveries would be rescheduled. The parameters of JIS delays in the interval 0-12 hours, and the on-time delivery reliability between 86% and 97% respectively (**section 2.1.2**). **Figure 4.1.3** illustrates the *JIS receive* in format of date time under the condition of rescheduling deliveries. In order to reflect the JIS reality in the automotive sector and set up the appreciate simulation parameter, the expanded model should reflect the delivery performance in the on-time delivery reliability about 86% at least (**Table 4.1.4**). For creating JIS deliveries, the parameter is set to supplier normal distribution (**Table 4.1.5**), because normal distribution performs sufficiently and is popular applied in the discrete event simulation (**section 3.3.1**). According to the geographical distribution and the different supplier integration grad, the foreign 1TSs - supplier 1, supplier 2, supplier 3 and supplier 4 – deliver their items in a relative larger time instance than the rest

regional 1TSs. Therefore, their values of the mean, standard deviation and minimum as well as maximum are bigger than the corresponding values of the other four regional 1TSs.

**Table 4.1.4. Input Parameters**

| Deviation Re/Scheduled | | |
|---|---|---|
| In Days | Frequency | % |
| 0 | 130 | 76% |
| 1 | 2 | 1% |
| 2 | 4 | 2% |
| 3 | 9 | 5% |
| 4 | 5 | 3% |
| 5 | 8 | 5% |
| 6 | 6 | 4% |
| 7 | 6 | 4% |
| **∑** | **170** | **100%** |

| JIS Delay | | |
|---|---|---|
| In Hours | Frequency | % |
| 0.0000 | 155 | 88% |
| 2:00:00.0000 | 2 | 1% |
| 4:00:00.0000 | 2 | 1% |
| 6:00:00.0000 | 3 | 2% |
| 8:00:00.0000 | 3 | 2% |
| 10:00:00.0000 | 5 | 3% |
| 12:00:00.0000 | 5 | 3% |
| **∑** | **170** | **100%** |

**Table 4.1.5. Delivery Normal Distribution**

| **Supplier** | **Delivery Normal Distribution** | | | |
|---|---|---|---|---|
| | μ | σ | Minimum | Maximum |
| Supplier 1 | 4,19 | 6,17 | 0 | 30 |
| Supplier 2 | 4,12 | 6,12 | 0 | 30 |
| Supplier 3 | 4,08 | 6,08 | 0 | 29 |
| Supplier 4 | 4,00 | 6,00 | 0 | 29 |
| Supplier 5 | 3,20 | 5,20 | 0 | 28 |
| Supplier 6 | 3,10 | 5,15 | 0 | 28 |
| Supplier 7 | 3,08 | 5,10 | 0 | 27 |
| Supplier 8 | 3,04 | 5,06 | 0 | 27 |

As explained at the beginning of this section, the focus of this section lies in the informaiton of formal model as summarized in **Table 4.1.6** based on the aspects afore discussed.The simulation model should reflect the delivery processes between the 1TSs and a VM**.** As discussed in **section 2.1.2**, container flows are operated in the reality as an independent material flows, the unavailable containers JIS can cause enormous processes disturbances with regards to KANBAN systems. These processes are not meant to getting involved in this work, but can be implemented in the further work. Meanwhile, the adoptability of the stock capacity at VMI stock and regional 1TSs are not adopted in this thesis, but can be studied further in terms of rescheduling delivery quantity. Furthermore, the transport time will not be taken into account, because it connects with the JIS receive date. It means, if a foreign supplier confirms a rescheduled delivery date, the time

of shipping and transport to the VMI stock should be calculated out and this calculated time should not be more than the delivery lead time allows. Secondly, when it refers to the time of JIS transport, it takes about 3-6 hours and this can be reflected by JIS arrival date time. It means, if the JIS transport processes are disturbed, delivery will delay.

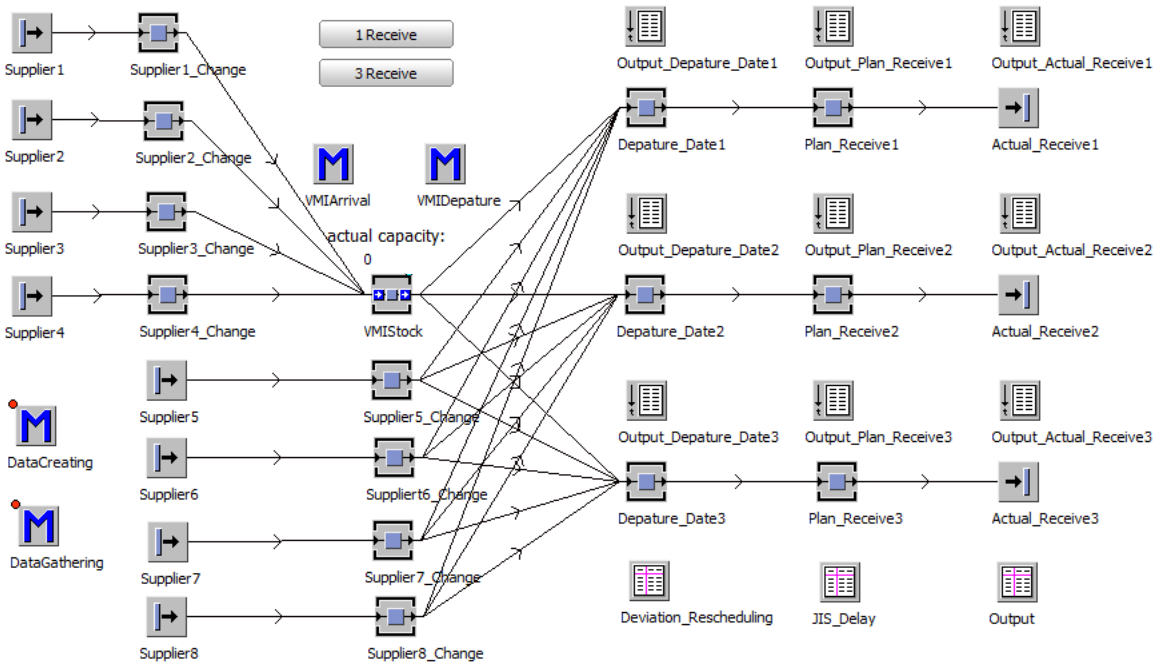
**Table 4.1.6. Information of Formal Model**

| **Task and System Description** | |
|---|---|
| Depicted System | ▪ Delivery departures from the 1TSs<br>▪ Delivery arrives at the VM |
| Conditions | ▪ Without consideration of the container flows<br>▪ Without consideration of suppliers' and VMI stock capacity<br>▪ Without consideration of quality and price attributes |
| Input Data | ▪ Normal distribution function of the deliveries<br>▪ Empirical distribution of the derivation between the scheduled received date and rescheduled received date<br>▪ Empirical distribution of the derivation between the rescheduled received date time and JIS arrival date time |
| Output Data | ▪ Assembly line<br>▪ Scheduled received date time<br>▪ Rescheduled received date time<br>▪ JIS arrival date time |
| Visualization | ▪ 2D |
| **Model Input Data** | |
| Input Data | ▪ Normal distribution function of the JIS deliveries (**Table 12**)<br>▪ Empirical distribution of the derivation between the scheduled received date and rescheduled received date (**Table 11**)<br>▪ Empirical distribution of the derivation between the rescheduled received date time and JIS arrival date time (**Table 11**) |
| **Modeling System Structure** | |
| Level of Detail | ▪ 8 Suppliers<br>▪ 3 Assembly Lines<br>▪ 1 VMI Stock |
| Interface | ▪ MS Excel |
| V&V Output | ▪ Qui-Squared Test |
| Replication | ▪ Confidence Interval Method with $\alpha$=5% |

### 4.1.3 Execuable Model

This section describes modeling phase *implementation* and the result is an executable Model. This part also performs as a step of "Multi-run Execution Loop" of data farming (**Table 5**). Based on the existing model, the four elements *source* represent the international 1TSs, renamed as supplier 1, 2, 3 and 4, and share a common buffer symbolized as *VMI Stock* which is used for the JIS delivery. The other four elements *source,* renamed as supplier 5, 6, 7 and 8 respectively, stand for the regional 1TSs and there is no buffer between them and the VM, because they deliver items directly from their own stocks. In order to identify and distinguish the results of each simulation run, the value of the *random numbers variants* has to be set to one bigger than the last simulation run as the every simulation run new starts. This is going to repeat as long as the significance level calculated by the confidence intervals function is bigger than 5% (**section 3.2**).

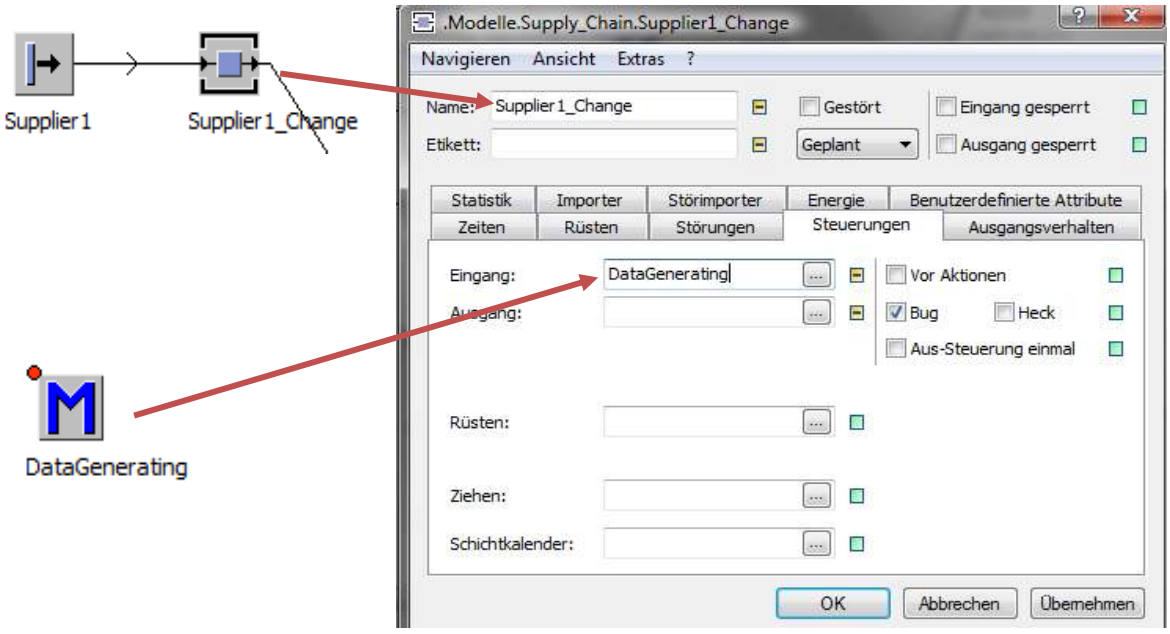
**Figure 4.1.4. Simulation Model of Rescheduled Delivery Processes**




For each of the supplier a source station named as "SupplierX_Change" representing "Supplier1_Change", "Supplier2_Change" and so on, is added in order to set up a control program "Scheduled_Receive" for creating the scheduled date. Actually, the "Scheduled_Receive" means that the scheduled order which should be received by the assembly line from the viewpoint of a VM. In order to explain in a simple way, the following term "Supplier X" represents the "Supplier 1", "Supplier 2" and so on. The simulation time is defined as 100 days at the *event controller*, because of the implementation expense that is resulted by the longer simulation run time and more generated data (**section 3.2**). The goal of this thesis is to farm the transaction data for clustering algorithms, therefore simulation time is estimated for 100 days for this thesis. *Event controller* coordinates and synchronizes the events which are the "Scheduled_Date" as one of the output attribute and also enables to start, stop and reset the excusable model (**section 3.4**). As the simulation start, the "Scheduled_Receive" are created at each source "Supplier X" by the delivery normal distribution (**Table 4.1.5)** within the simulation time for 100 days. This normal distribution can be manual written in the field of the "Instance" at each "Supplier X" in the data format "time", interpreted in the way "dd:hh:mm:ss.xxx" with terms of the value interval of the attribute "JIS_Delay" which is between 0-12 hours. After "Scheduled_Receive" being created, it will be

processed further by “SupplierX_Change” at once with a certain dwelling time which is decided by the discrete empirical distribution “Derivation_Re/Scheduled” (**Table 4.1.4**). This “Derivation_Re/Scheduled ” is loaded in a information table where the input parameters can be filled. By programming a command in the method “DataGenerating”(**A.1. Code**), it begins to generate the data “Rescheduled_Receive”. This method is inserted in the field of “entry” in the “control” function of the each “SupplierX_Change”. If no bugger occurs, then the connection between these input parameters and the executable model for data generating is working. This situation can also be treated as V&V step by the *animation* (**section 3.3.3**). By the method “DataGathering”(**A2. Code**), the generated data “Rescheduled_Receive” will be written in the “output” table in the format of “datetime”. This output value will be given further to the next step of generating the “JIS_Delay”, while the random values of “Derivation_Re/Scheduled” being created, the other random values “JIS_Delay” are being created in the same way, but by calculating the discrete empirical distribution loaded in the other table of input data “JIS_Delay” in the format of “time”, due to the value interval between 0 and 12 hours. The corresponding output data of the input “JIS_Delay ”are the “JIS_Arrival” and will be written in the table ” Output” table in the format “datetime” by the method “DataGathering”. The last target attribute of the output data is “Assembly Line” and will be collected in the format “integer” instead of “string” by programming a command in the method “DataGathering”, because it will be used for clustering algorithms with all the data in numerical format (**section 2.3**).

**Figure 4.1.5. Method for Generating Data**




V&V technique for this section is *animation,* because it relates to the run the visual model, if no bugger happens, it means the executable model performances in valid manner. But the performance of this executable model can only be supervised within the simulation time for 100 days. If bugger occurs, a dialog window of console will come out for debugger explanation. Following this information, it has to review the logical relationship between the simulation entities to adjust the input parameters according to the early phase results (**section 3.3.3**). The description of this section can be handled as a part of the document. After every simulation run, the output data will be updated in the MS Excel file named “Output”, evaluated and documented. If the

results are not rejected by the test, then the output data are valid to input in the nest executable model, but with a different random number which is reset at the *event controller*, in order to distinguish the results of different simulation run.

### 4.1.4 Verification and Validation

This section refers to the experiments of this simulation study which is carried out by calculating the simulation replication. For achieving this, chi-squared test and confidence intervals method are applied. Furthermore, every step of V&V on the output data is documented in detail in the file "Output".

After every simulation run, the output data are exported in the excel table and validated by chi-squared test, in order to provide the firmed evidence, whether the designed model is the expected model. That means the designed model should generate the similar output parameters to the original input parameters but with different start values. Because the input data are the empirical distribution, the output data need to be transformed in as a set of population percentage first, in order to determinate if the output data are significantly fair to the input data (**section 3.3.3**). Translating in statistical terms, the output data should be called as "observed data" or "alternative hypothesis values" and the "expected data" are treated as "hypothesized values". The *null* hypothesis, $H_{0,}$ is that the actual population percentages are exactly equal to the hypothesized values. The *alternative* hypothesis, $H_1$, is that the actual population percentages are different from the hypothesized values. The "expected" result means the *null* hypothesis. Translating in this simulation case:

$H_0$: The simulation output data are not rejected and **not** significant **unfair** to be input further in the next simulation run.

$H_1$: The simulation output data are rejected and significant **unfair** to be input further in the next simulation run.

**Table 4.1.7. Chi-Squared Test on Output Data**

| Output Data | Comparison with Critical Value α=5% | Test Result |
|---|---|---|
| Deviation Re/Scheduled<br>Degrees of Freedom: 7 | Sum of $\left[\frac{(\text{Observed-Expected})^2}{\text{Expected}}\right] < 14.07$ | Not rejected |
| | Sum of $\left[\frac{(\text{Observed-Expected})^2}{\text{Expected}}\right] > 14.07$ | Rejected |
| JIS Delay<br>Degrees of Freedom: 6 | Sum of $\left[\frac{(\text{Observed-Expected})^2}{\text{Expected}}\right] < 12.59$ | Not rejected |
| | Sum of $\left[\frac{(\text{Observed-Expected})^2}{\text{Expected}}\right] > 12.59$ | Rejected |

The chi-squared test proceeds as follows:

1. Compute the *expected number* of the "Deviation Re/Scheduled" and "JIS Delay in Hours" by multiplying the population proportion of the input data by the total generated sample size, in other words, the sum of the generated orders.
2. For each category, subtract the expected number from the observed number, namely the output data, then square the result. This is a measure of the discrepancy between the output data and the hypothesized population percentages.
3. For each category, divide the result of step 2 by the expected number. This has the effect of adjusting for the fact that when larger numbers are expected, larger deviations also generally occur.
4. Sum of the values from step 3 is to obtain the chi-squared statistic. The larger this number is, the more different the output data are from the hypothesized population proportions.
5. Find the degrees of freedom of "Deviation Re/Scheduled", which is 7, and "JIS Delay", which is 6.
6. Compare the computed chi-squared statistic from step 4 to the critical value 5% in the chi-squared table. In this simulation model case, critical value 5% for degrees of freedom 7 is 14,07 and critical value 5% for degrees of freedom 6 is 12,59.

If the output are not rejected by the chi-squared test, it will go to examine the replication of the simulation runs by the confidence interval method which proceeds a cumulative calculations of the mean values and their deviations of the output data one simulation run after the another until the significance level α of the both of the "Deviation Re/Scheduled" and "JIS Delay" are smaller than 5%.

**Table 4.1.8. Result of Chi-Squared Test**

| Replication | Deviation Re/Scheduled | | | JIS Delay | | |
|---|---|---|---|---|---|---|
| | $X^2$ | $X^2_{v;1-\alpha}$ | $H_0$ | $X^2$ | $X^2_{v;1-\alpha}$ | $H_0$ |
| **1** | 14,07 | 6,63 | Not rejected | 12,59 | 8,28 | Not rejected |
| **2** | 14,07 | 5,23 | Not rejected | 12,59 | 6,23 | Not rejected |
| **3** | 14,07 | 3,80 | Not rejected | 12,59 | 10,91 | Not rejected |
| **4** | 14,07 | 10,31 | Not rejected | 12,59 | 8,91 | Not rejected |
| **5** | 14,07 | 7,81 | Not rejected | 12,59 | 8,52 | Not rejected |
| **6** | 14,07 | 30,27 | Rejected | 12,59 | 4,73 | Not rejected |

In an ideal way, a successful data farming study in this thesis should pass the qui-squared test after each simulation run, until the significance level α is smaller than 5%. As interpreted in **Table 4.1.8**, the result of the "Deviation Re/Scheduled" after the sixth simulation run does not pass the qui-squared text. Therefore, it has to check all of the documents through again and summarize the possible causes as follows:

1. Inaccurate estimation of the input data.
   For implementing this thesis, the real data are not available to collect, but have to be estimated and formulated in statistical distribution based on several publication research results and state of art in **chapter 2** and **chapter 3**. On the other hand, it lacks of the SC expertise's experience which can help to adjust the estimated input data. These lead to the unexpected simulation results.

2. Insufficient simulation time.
   The simulation time is set to only 100 days at *event controller*, so that the simulation results can only be evaluated and drawn a conclusion within 100 simulation days (**section 3.2**). An assumption exists that if the simulation time were set to longer than 100 days with the same input data, would the output data pass the qui-squared test until the significance level of the confidence intervals reach to 5%? Which correlation between the simulation time and results, can be researched further based on this thesis.

3. Shortcoming of the task definition for the simulation study.
   The phase *task definition* guides the entire simulation study procedure and plays a significant role of the success of a simulation study (**section 3.3.3**). If the shortcoming in this phase exists, it should be identified by V&V techniques, before the result of this phase is documented and adopted by the further procedure phases. Under the framework of this thesis, the V&V technique for this phase is only possible to be carried out by *desk checking* and *structured walkthrough* without discussion with the second or third participate. This art of the absolute subjective test can lead to that the shortcoming cannot be discovered or adjusted by oneself who implements the task definitions (**section 3.3.4**).

The V&V on the output data is significant for a simulation study, because the knowledge is supposed to be extracted from this phase and it creates the fundamental ground for data analysis. On the other hand, this section also addresses the issue of model fit as mentioned in **section 2.2.1**. To proceed the task implementation of this thesis, one set of the validated simulation output data will be applied for the further data analysis.

## 4.2 Output Data Transformation

This section concerns the output data transformation which should be assigned as "Data Processing" in the table of "Data Farming Elements" (**Table 4.2.1**). This step is symbolized as the start point of "Harvest", because the output data were already processed by V&V tests and can be analyzed for extracting knoweldge. The original data farming concept applied in military domain to preceeds a "Multi-un Execution Loop" in order to improve the accuracy level of data analysis results and adjust the parameters of data modeling in terms of the enormous variations. However, this loop is only processed in **section 4.1.3** and section **4.1.4**, because the objective of this thesis is to analyze the simulation output data with clustering algorithms. As introduced in **section 2.3.2**, data mining can be processed in decremental way and generate new knowledge from a new data set which is mixed up with existing data set and new data in order to suite a data analysis study about the dynamical systems.

**Table 4.2.1. Data Farming Elements – Data Processing**

| **Brady et al.(2013)** | **Nato Report (2014)** | | |
|---|---|---|---|
| **3. Harvest** | **Multi-run Execution Loop** | **5. Analysis and Visualization** | |
| | | 5.1 | Data Processing |

As noticed, data transformation is the third step of the *KDD process* (**Figure 2.2.3**). However, there is no need to process the steps of *data collection* and *data preprocessing*, because the works on reduction of data dimension, handling with the outlier and noise has been already accomplished during the simulation study. After the sufficient simulation runs as the result the confidence interval test confirmed, the output data are loaded in the excel file for preparing the clustering analysis (**Figure 4.2.1**). For this thesis output data transformation is processed in two steps. The first step is to be carried out with MS Excel sheet for transform the attributes in an adequate data format. The second step is to rescale all the attributes values in the interval (0, 1) by using RapidMiner "Normalization" operator as illustrated in **Figure 4.2.4**.

**Figure 4.2.1. Example for Output Data Format**

| Assembly Line | Scheduled Date | Rescheduled Date | JIS Date Time | Deviation Re/Scheduled | JIS Delay |
|---|---|---|---|---|---|
| 1 | 13.02.2010 05:29 | 19.02.2010 05:29 | 19.02.2010 05:29 | 6 | 0:00 |
| 1 | 19.02.2010 05:29 | 19.02.2010 05:29 | 19.02.2010 05:29 | 0 | 0:00 |
| 1 | 19.02.2010 05:29 | 24.02.2010 05:29 | 24.02.2010 05:29 | 5 | 0:00 |
| 1 | 24.02.2010 05:29 | 25.02.2010 05:29 | 25.02.2010 05:29 | 1 | 0:00 |
| 1 | 25.02.2010 05:29 | 25.02.2010 05:29 | 25.02.2010 05:29 | 0 | 0:00 |
| 1 | 25.02.2010 05:29 | 25.02.2010 05:29 | 25.02.2010 05:29 | 0 | 0:00 |
| 1 | 25.02.2010 05:29 | 25.02.2010 05:29 | 25.02.2010 11:29 | 0 | 6:00 |

As discussed, clustering algorithms can only accept metric value in order to measure their similarities. Data transformation aims to replace the output data values with metric or numeric values so that they become easier to be interpreted and calculated by the clustering algorithms (**section 2.3**). Already in simulation phase, the value of the "Assembly Line" is set to integer. The problem about date time is not simple to solve, but how to transform them in metric form. The

essential meaning of date time should be kept after transformation, and at the same time the transformation should be convenient for similarity measures. Each of generated data samples by simulation has a data type in “datetime” or “time”. The values of this data type need to be scaled in proper instance unite in which the other attributes values can also be transformed. Therefore it has to consider an alternative solution. First of all, all the data in the form of data time have a common information part, namely the weekdays as, Monday, Thursday and so on. Secondly, for production program planning the weekdays are also attributed in a plan pattern. This transformation step can be accomplished by using RapidMiner application “Date to Numerical”, but in this case, it is more effective to transform directly in MS Excel sheet by setting the data type of “Monday, 15 February 2010”, then by using a search function SVERWEIS the column of “JIS Delay” is transformed according to the matrix where the relationship between the days of a week from Monday to Sunday and the numbers from 1 to 7 is built, so a new column of “JIT Weekdays” is created as shown in **Figure 4.2.2**.

**Figure 4.2.2. Example for Transformed Output Data Format**

| Assembly Line | Scheduled Date | Rescheduled Date | JIS Date Time | JIS Weekdays | Deviation Re/Scheduled | JIS Delay |
|---|---|---|---|---|---|---|
| 1 | 13.02.2010 05:29 | 19.02.2010 05:29 | Freitag, 19. Februar 2010 | 5 | 6 | 0,000 |
| 1 | 19.02.2010 05:29 | 19.02.2010 05:29 | Freitag, 19. Februar 2010 | 5 | 0 | 0,000 |
| 1 | 19.02.2010 05:29 | 24.02.2010 05:29 | Mittwoch, 24. Februar 2010 | 3 | 5 | 0,000 |
| 1 | 24.02.2010 05:29 | 25.02.2010 05:29 | Donnerstag, 25. Februar 2010 | 4 | 1 | 0,000 |
| 1 | 25.02.2010 05:29 | 25.02.2010 05:29 | Donnerstag, 25. Februar 2010 | 4 | 0 | 0,000 |
| 1 | 25.02.2010 05:29 | 25.02.2010 05:29 | Donnerstag, 25. Februar 2010 | 4 | 0 | 0,000 |
| 1 | 25.02.2010 05:29 | 25.02.2010 05:29 | Donnerstag, 25. Februar 2010 | 4 | 0 | 0,250 |

**Figure 4.2.3** presents the data after the first transformation step and these data will be imported in the RapidMiner. The processes of the data import in RapidMiner follows the procedure as introduced in **section 3.4**, and the data are saved in the *local respository* named as “Cluster JIS Delay”( **Figure 4.2.4**).

**Figure 4.2.3. Example for Import Data in MS Excel View**

| Assembly Line | JIS Weekdays | Deviation Re/Scheduled | JIS Delay |
|---|---|---|---|
| 1 | 5 | 6 | 0,000 |
| 1 | 5 | 0 | 0,000 |
| 1 | 3 | 5 | 0,000 |
| 1 | 4 | 1 | 0,000 |
| 1 | 4 | 0 | 0,000 |
| 1 | 4 | 0 | 0,000 |
| 1 | 4 | 0 | 0,250 |

The second step of data transformation proceeds in data modeling process by using the operator “Normalize” (**Figure 4.2.5**) so that it enables cording the attribute values in the interval (0,1). The operator “Normalize” provides the normalization methods “range_transformation” and “proportion_transformation”. When the “range_transformation” is selected, two parameters (min, max) appear in the parameter view. Range transformation normalizes all attribute values in the specified range (min, max). Min and max are specified using min and max parameters respectively as mentioned in **section 2.3.2**. Because this operator doesn’t provide a view of normalized data, **A.5** describes normalization in details according to the transformation formula **F. 2.3-9**.

**Figure 4.2.4. Example for Import Data in RapidMiner View**

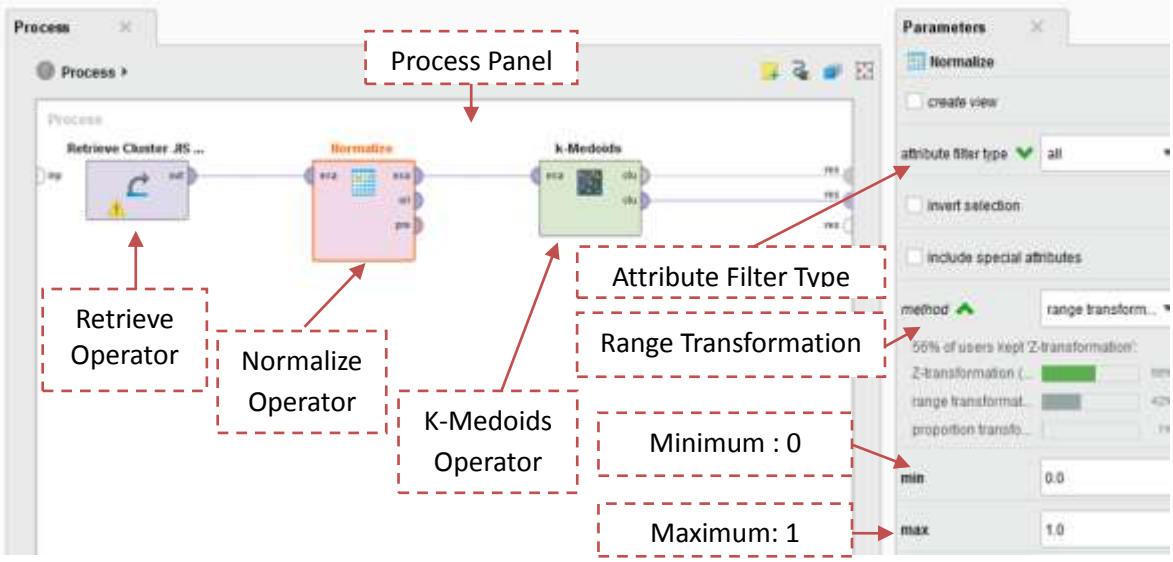
ExampleSet (//Local Repository/Cluster JIS Delay)

ExampleSet (156 examples, 0 special attributes, 4 regular attributes)

| Row No. | Assambly line | JIS week days | Deviation Re/scheduled | JIS delay |
|---|---|---|---|---|
| 1 | 1 | 5 | 6 | 0 |
| 2 | 1 | 5 | 0 | 0 |
| 3 | 1 | 3 | 5 | 0 |
| 4 | 1 | 4 | 1 | 0 |
| 5 | 1 | 4 | 0 | 0 |
| 6 | 1 | 4 | 0 | 0 |
| 7 | 1 | 4 | 0 | 0.250 |

There are several common functions of data transformation between RapidMiner and MS Excel. The user can decide which is convenient or effective according to the data sample size, data format requirements as well as user skills level and experiences.

**Figure 4.2.5. Data Normalization with RapidMiner**



## 4.3 Analysis of Output Data with Clustering Algorithms

This section constructs the last realm of data farming and the last phase of the procedure model of simulation study by using knowledge discovery techniques (**Table 4.3.1**).

**Table 4.3.1. Data Farming Elements – Statistical Analyses and Knowledge Discovery**

| **Brady et al.(2013)** | **Nato Report (2014)** | |
|---|---|---|
| **3. Harvest** | **Multi-run Execution Loop** | **5. Analysis and Visualization** |
| | | 5.2 Statistical Analyses |
| | | 5.3 Results for the support of decision-making through answering the what-if questions |

As introduced in **section 2.2.3**, knowledge discovery is an interdisciplinary technique combined with computer graphic and visualization for data modeling, mathematics for evaluating data analysis results and statistics for interpretation and validation of data analysis results. For data modeling, three types of partitioning methods: k-means, k-medoids and expectation maximization clustering are used, because it is permissible to try different algorithms on the same data (**section 2.3.2**). The classification results of different cluster algorithms will be displayed by using software tool RapidMiner and documented in detail.

### 4.3.1 Modeling

Before starting with data modeling, it needs to recall that one of the tasks of this thesis is to classify the generated transaction data of JIS deliveries as regular or disturbed processes. This impies that output data should be grouped in two clusters. One cluster should represent “JIS regular” and another should represent “JIS disturbed”. Out of the clustering algorithms which are introcued in **section 2.3.2**, the partitioning clustering methods come as the first to consider, because these methods are applied when k as the number of clusters is given. After transformed output data are imported in RapidMinder, data modeling are implemented by k-means algorithm, k-medoids algorithm and expectation maximization clustering separately.

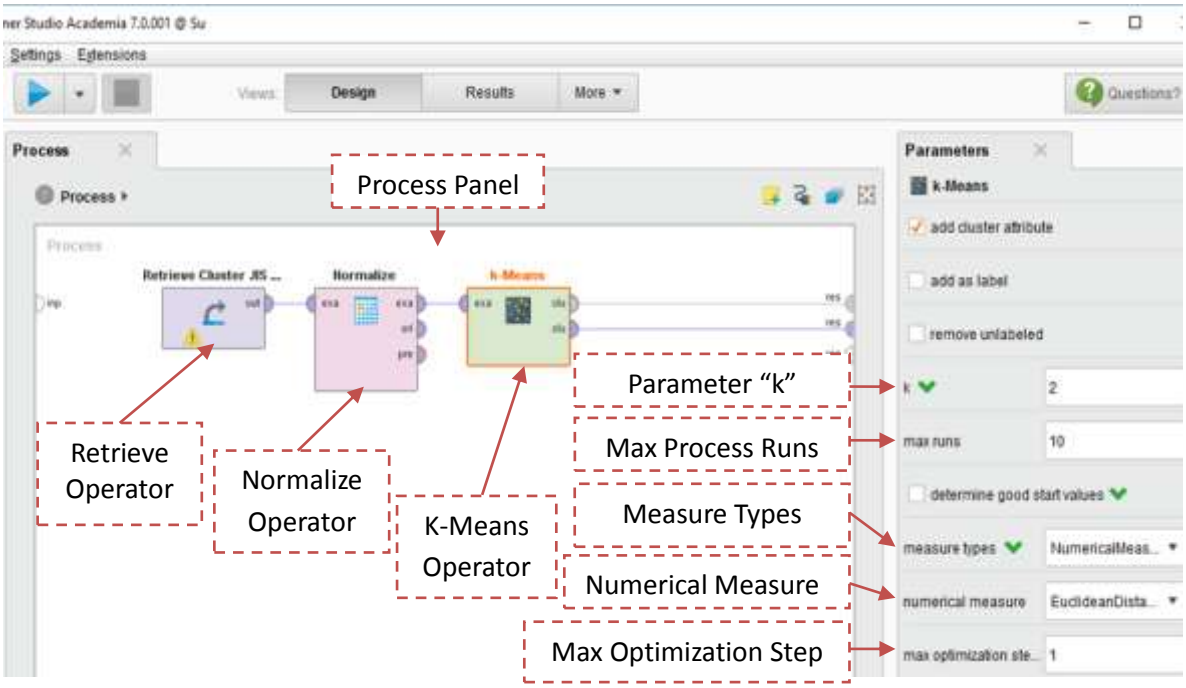
#### i. K-Means Algorithm

The first cluster algorithm for data modeling begins with k-means algorithm and modeling of output data with RapidMinder proceeds as introduced in **section 3.4**. Firstly, the data file named as “Cluster JIS Delay” is dragged from the repository panel onto the process panel so that the RapidMiner could use the retrieve operator to incorporate the data. Secondly, it needs a “normalize” operator to rescale attribute values to fit in a specific range (0, 1) in order to create the same scale for a fair comparison between them. Thirdly, the k-means operator is dragged onto the process panel and the parameters are selected. “k” is set to “2”, because it aims to obtain two clusters. The parameter of “max runs” specifies the maximal runs of k-means with random initialization. This is not significant to this thesis, so it is set to “10” optionally. The parameter “measure types” is to select “Numerical Measures”, because all of values are already transformed

in numerical type by “normalize” operator. The parameter “numerical measure” is set to “Euclidean Distance”, because this distance function is used for n dimensional data sample and suits to data samples of “Cluster JIS Delay” which contain four attributes. The parameter “max optimization step” is set to “1” in order to interpret the iterative optimization procedure of grouping clusters. Before letting RapidMiner run the modeling process, it needs to connect the operators to the corresponding input and output ports which provide graphical data displays. **Figure 4.3.1** illustrates the entire process of creating a modeling as well as setting parameters.

As described in **section 2.3.2**, k-means clustering begins with selecting two imaginary objects for creating a centroid of a cluster. Applied in this thesis for creating two clusters four imaginary objects are selected to create two centroids for cluster 0 and cluster 1 respectively. It needs to mention that the centroid itself is a fictive object and not a member of a cluster. By calculating the distance between each centroid and all of the other objects except for the objects which are selected to create centroids respectively, the objects are assigned to a cluster to whose centroid their distance are shorter than to the centroid of the other cluster. This procedure repeats when each time the centroids move and an object changes its membership of a cluster. Both of the centroids of the old and new clusters are recalculated, until the centroids do not move again at the new optimization step.

**Figure 4.3.1. Creating a K-Means Modeling Process**




As summarized in **Table 4.3.2**, the first optimization step results 64 objects for cluster 0 and 92 objects for cluster 1. The Centroid of cluster 0 is (0,133; 0,599; 0,415; 0,089) and the Centroid of cluster 1 is (0,788; 0,438; 0,149; 0,092). It certainly needs to precede the optimization further, but before doing it, the results should be documented in MS Excel sheet “Optimization1” of the file “k-Means” for the latter evaluation.

To start with the second optimization step only needs to set the parameter “max optimization step” to “2” and let modeling process run again. This time the old centroids which were created in the first optimization step are replaced by the new centroids and a couple of objects in cluster 1

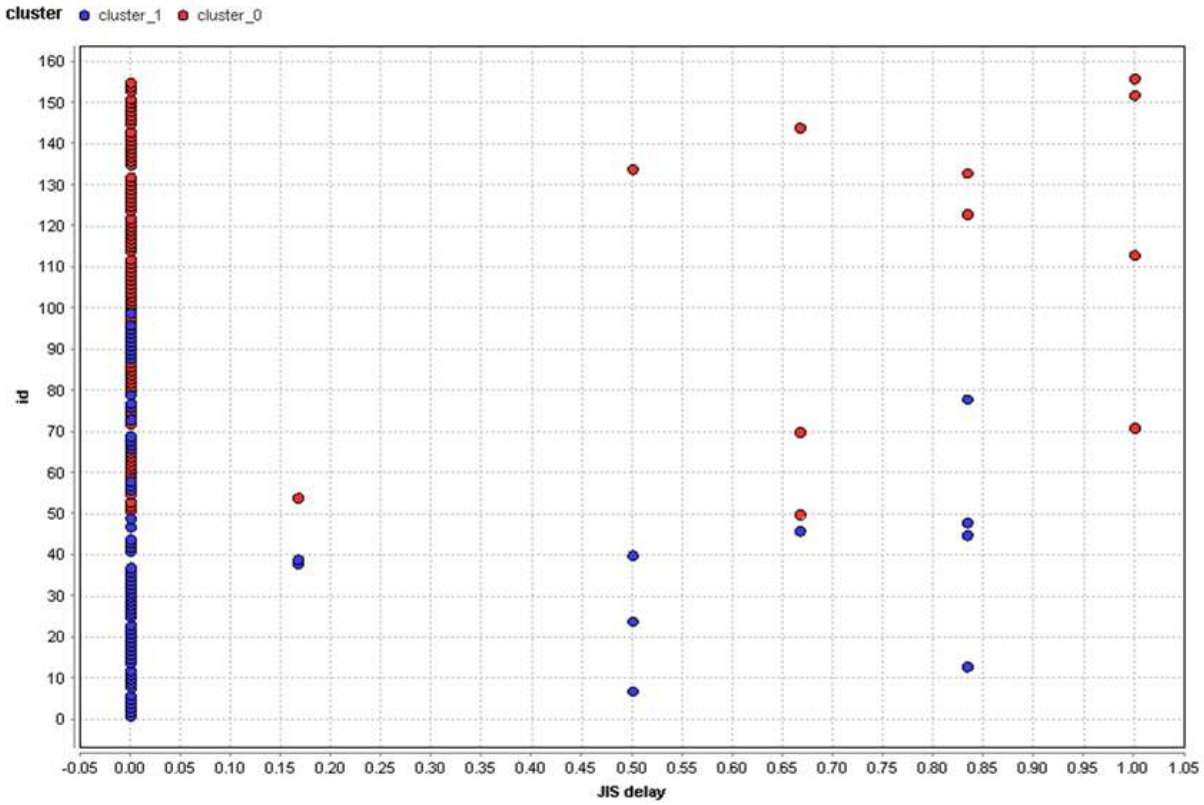
turn to cluster 0, because their distance to the new centroid of cluster 0 is shorter than to the new centroid of cluster 1. Therefore, the new centroid of cluster 0 is (0,827; 0,348; 0,208; 0,103) and the new centroid of cluster 1 is (0,108; 0,673; 0,312; 0,078). This optimization process is analog to run repeatedly, until the new centroid is equal to the old centroid. As shown in **Table 4.3.2**, it is to notice that clusters of the fourth optimization step and clusters of the third optimization step have the same centroids. That means the centroids do not change again at the fourth optimization step and should be taken as the final result. 85 objects are assigned to cluster 0 and 71 objects are assigned to cluster 1. The Centroid of cluster 0 is (0,812; 0,351; 0,208; 0,098) and the Centroid of cluster 1 is (0,169; 0,688; 0,318; 0,082). As discussed in **section 2.3.2**, another necessary precondition for drawing a conclusion that the optimization process should stop, is that the distance value calculated by cluster compactness cost function should be minimal. This topic will be discussed further on the topic of results evaluation.

**Table 4.3.2. Centroid Data of K-Means Algorithm**

| **Optimization Step** | **Sample Size** | **Cluster** | **K-Means Centroid** | | | |
|---|---|---|---|---|---|---|
| | | | **Assembly Line** | **JIS Weekdays** | **Deviation Re/Scheduled** | **JIS Delay** |
| 1 | 64 | Cluster 0 | 0,133 | 0,599 | 0,415 | 0,089 |
| | 92 | Cluster 1 | 0,788 | 0,438 | 0,149 | 0,092 |
| 2 | 81 | Cluster 0 | 0,827 | 0,348 | 0,208 | 0,103 |
| | 95 | Cluster 1 | 0,187 | 0,673 | 0,312 | 0,078 |
| 3 | 85 | Cluster 0 | **0,812** | **0,351** | **0,208** | **0,098** |
| | 71 | Cluster 1 | **0,169** | **0,688** | **0,318** | **0,082** |
| 4 | 85 | Cluster 0 | **0,812** | **0,351** | **0,208** | **0,098** |
| | 71 | Cluster 1 | **0,169** | **0,688** | **0,318** | **0,082** |

**Figure 4.3.2** visualizes the cluster 0 in blue and the cluster 1 in red. K-means operator also provides the result sample list as shown in **A.7** to overview which data sample belongs to which cluster. It is to notice that cluster 0 and cluster 1 are not grouped with the clear boundary. The reason is that the centroids of k-means are the fictive objects which distort the form of cluster dramatically, especially with the attribute values “JIS Delay”. The most of “JIS Delay” values are “0”, because the simulation output data reflect the fact that the most JIS deliveries are regular in the automotive industry. The rest of “JIS Delay” values which are not equal to “0” are the generated *outliers* by data farming. These outliers are not mistakes, but a group of data this thesis tries to identify and observe what kind of impact their behaviors have on the JIS delivery processes. This situation expresses that the k-means algorithm is sensitive to the outliers.

**Figure 4.3.2. Visualization of K-Means Algorithm Result of the 3rd Optimization Step**




### ii. K-Medoids Algorithm

Analog to k-means algorithm, k-medoids algorithm groups objects in a cluster based on the distance measurement. Being different to k-means, k-medoids calculates the distance using a medoids, not a centroid. A medoid is one member of a cluster and its distance to other members of the cluste should be the shortest. As introduced in **section 2.3.2**, the optimization process proceeds following a principle of reordering medoids. If an existing object can perform better quality in cluster grouping than the existing medoids, this new candidate will replace the existing medoids as new medoids. The quality in cluster grouping is evaluated by the cost function (**F. 2.3-11**) and the optimal value of the cluster quality should be minimal.

The modeling process is similar with the k-means modeling, but instead of the operator “k-Means” the operator” k-Medoids” is dragged onto the process panel (**Figure 4.2.5**). The parameters are set as same as the “k-Medoids”, because both of the algorithms aim to obtain two clusters and follow the similar optimization principle. As summarized in **Table 4.3.3**, the optimization process has been through only three steps. At the first optimization step, 68 objects are grouped in cluster 0 and 88 objects are assigned in cluster 1. The cluster quality of this step performs with value “132,0628371”. At the second optimization step, a couple of objects change their membership of the clusters. 17 objects are assigned in cluster 0 and 139 objects get together for cluster 1. With the quality value “121,8417540”, the cluster quality of the second step is obviously better than the first step. Following the reordering principle, the new candidate object replaces the old centroid. The centroid of cluster 0 doesn’t change, but centroid of cluster 1 turned to (1; 0,667; 0; 0). After the third optimization step, the cluster quality performs as same as at the second step, but the memberships of the clusters have changed. Cluster 0 is reconstructed by 142 objects and cluster 1 is reformed by 14 objects. However, this thesis decides to adopt the result of the third step in

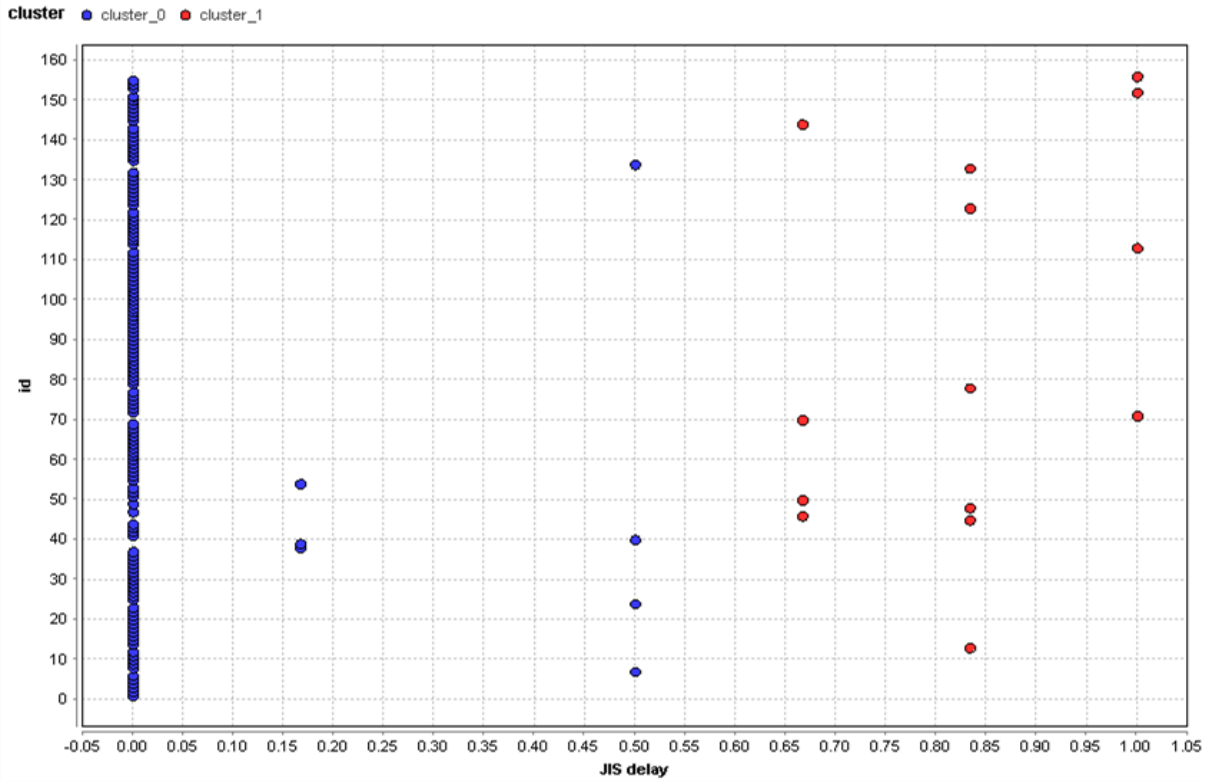
terms of the cluster membership. Firstly, it relates to the latter comparison with the results of the other cluster algorithms in order to test their differences and rank their performances. Cluster 0 is defined as of a majority of objects which represent “JIS regular” and cluster 1 is set as a minority of objects which interpret “JIS disturbed”. Unexpectedly, the cluster grouping of the second step shows in the opposite way and brings troubles in latter procedure of statistical test and ranking performances. Secondly, the cluster grouping of the third step matches the requirements of latter procedure with the identical quality as the second step performs.

**Table 4.3.3. Cost Values of K-Medoids Algorithm**

| **Optimization Step 1** | | **Centroid** | | | **Cost= ∑ dist(Centroid, x)** |
|---|---|---|---|---|---|
| **Cluster** | **Sample Size** | **Attribute** | **Cluster 0** | **Cluster 1** | |
| Cluster 0 | 68 | Assembly Line | 1 | 0,5 | 132,0628371 |
| | | JIS Weekdays | 0,667 | 0,333 | |
| Cluster 1 | 88 | Deviation Re/Scheduled | 0 | 0,149 | |
| | | JIS Delay | 1 | 0 | |
| **Optimization Step 2** | | **Centroid** | | | **Cost= ∑ dist(Centroid, x)** |
| **Cluster** | **Sample Size** | **Attribute** | **Cluster 0** | **Cluster 1** | |
| Cluster 0 | 17 | Assembly Line | 1 | 1 | 121,8417540 |
| | | JIS Weekdays | 0,667 | 0,667 | |
| Cluster 1 | 139 | Deviation Re/Scheduled | 0 | 0 | |
| | | JIS Delay | 1 | 0 | |
| **Optimization Step 3** | | **Centroid** | | | **Cost= ∑ dist(Centroid, x)** |
| **Cluster** | **Sample Size** | **Attribute** | **Cluster 0** | **Cluster 1** | |
| Cluster 0 | 142 | Assembly Line | 1 | 1 | 121,8417540 |
| | | JIS Weekdays | 0,667 | 0,667 | |
| Cluster 1 | 14 | Deviation Re/Scheduled | 0 | 0 | |
| | | JIS Delay | 0 | 1 | |

**Figure.4.3.3** visualizes the final result that cluster 0 is in blue and cluster 1 is in red. The clusters of k-Medoids have a clear boundary between them and are not distorted as k-means clusters interpret, because grouping clusters is based on the medoid. Being different to the fictive centroid, medoids is an object assigned to a certain cluster. For this reason, the result of k-medoids algorithm performs in a robust way against outliers. The complete results of all optimization steps are documented in MS Excel “k-Medoids” in detail.

**Figure 4.3.3. Visualization of K-Medoids Algorithm Result of the 3rd Optimization Step**




### iii. Expectation Maximization Clustering

The EM Clustering is one of the partitioning clustering algorithms and the number of clusters k can be predefined as k-means and k-medoids algorithms (**section 2.3.2**). The procedure of EM is similar to the k-means algorithm, but extending this basic approach by computing probabilities of cluster memberships based on one or more probability distributions. The goal of this clustering algorithm is to maximize the overall probability how often an object is assigned to a certain cluster.

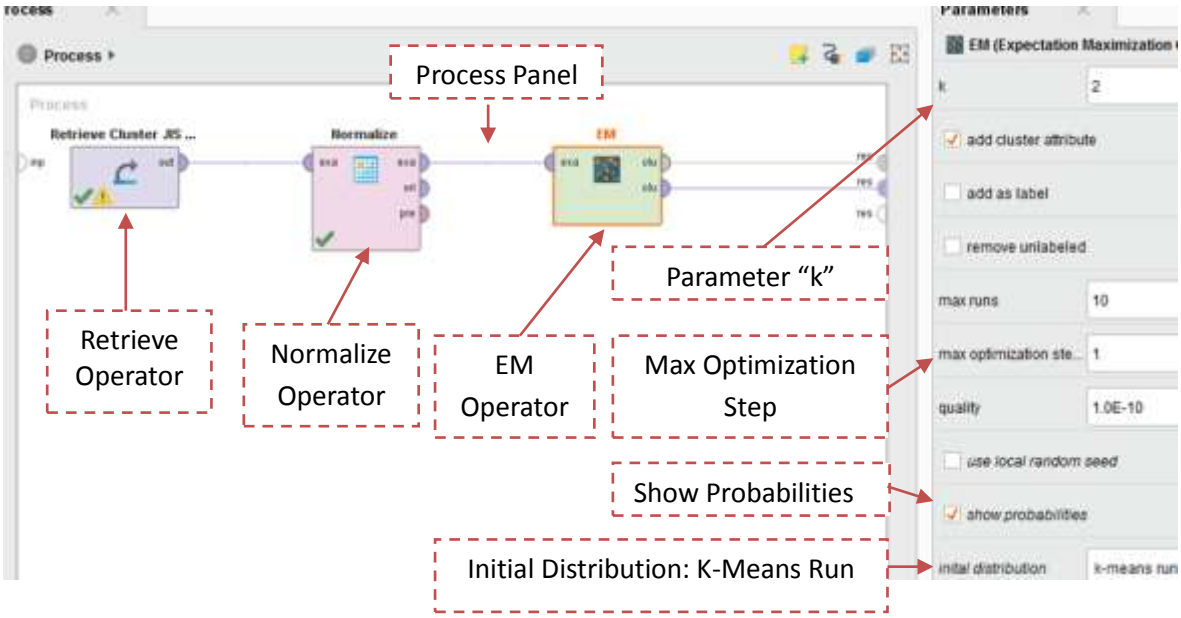
**Figure 4.3.4. Example for EM Result**

| Optimization step | ID | Cluster | Probability Cluster 0 | Probability Cluster 1 | Assembly line | JIS week days | Deviation Re/scheduled | JIS delay | Probability P(x) | Log (P(X)) |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | cluster_0 | 0,932636389 | 0,067363611 | 0 | 0,66666667 | 0,857142857 | 0 | 71,07781778 | 1,85173409 |
| 1 | 2 | cluster_0 | 0,849830954 | 0,150169046 | 0 | 0,66666667 | 0 | 0 | 72,40270473 | 1,85975479 |
| 1 | 3 | cluster_0 | 0,824386954 | 0,175613046 | 0 | 0,33333333 | 0,714285714 | 0 | 72,80980874 | 1,86218989 |
| 1 | 4 | cluster_0 | 0,804740977 | 0,195259023 | 0 | 0,5 | 0,142857143 | 0 | 73,12414437 | 1,8640608 |
| 1 | 5 | cluster_0 | 0,795443552 | 0,204556448 | 0 | 0,5 | 0 | 0 | 73,27290317 | 1,8649434 |

In order to present EM modeling process in a comprehensive way, a simple sample is explained at first. It starts with an imaginary object which creates an initial probability P (cluster 0) and P (cluster 1) for every object in the sample site. As illustrated in **Figure 4.3.4**, an object with ID "1" is described as (0; 0,667; 0,857; 0) and assigned to cluster 0, because its probability of being assigned to cluster 0 is 0,9326 and higher than to cluster 1 with the probability 0,0573. Every object obtains a value of P(X) which is the sum of "P (cluster 0) x sample size of cluster 0 + P (cluster 1) x sample size of cluster 1". At the first optimization step, the sample size of cluster 0 is 70 and the sample size of cluster 1 is 86. So the P(X) of the object with ID number 1 is "0,9326 x 70 +0,0573 x 86 = 71,0778". Subsequently, the value of the P(X) is manipulated further by the function Log (P(X)). Equally, every object obtains a value of Log(P(X)). Summing up these Log (P(X))

values of all of objects is resulted in the value of E, symbolized for “Expectation”. The function of this calculation is described as E =∑ Log (P(X)). Every optimization step obtains a E value which is expected to be the maximum. When the E value reaches the maximum, then it can draw a conclusion that optimization process can stop.

**Figure 4.3.5. Creating a EM Modeling Process**



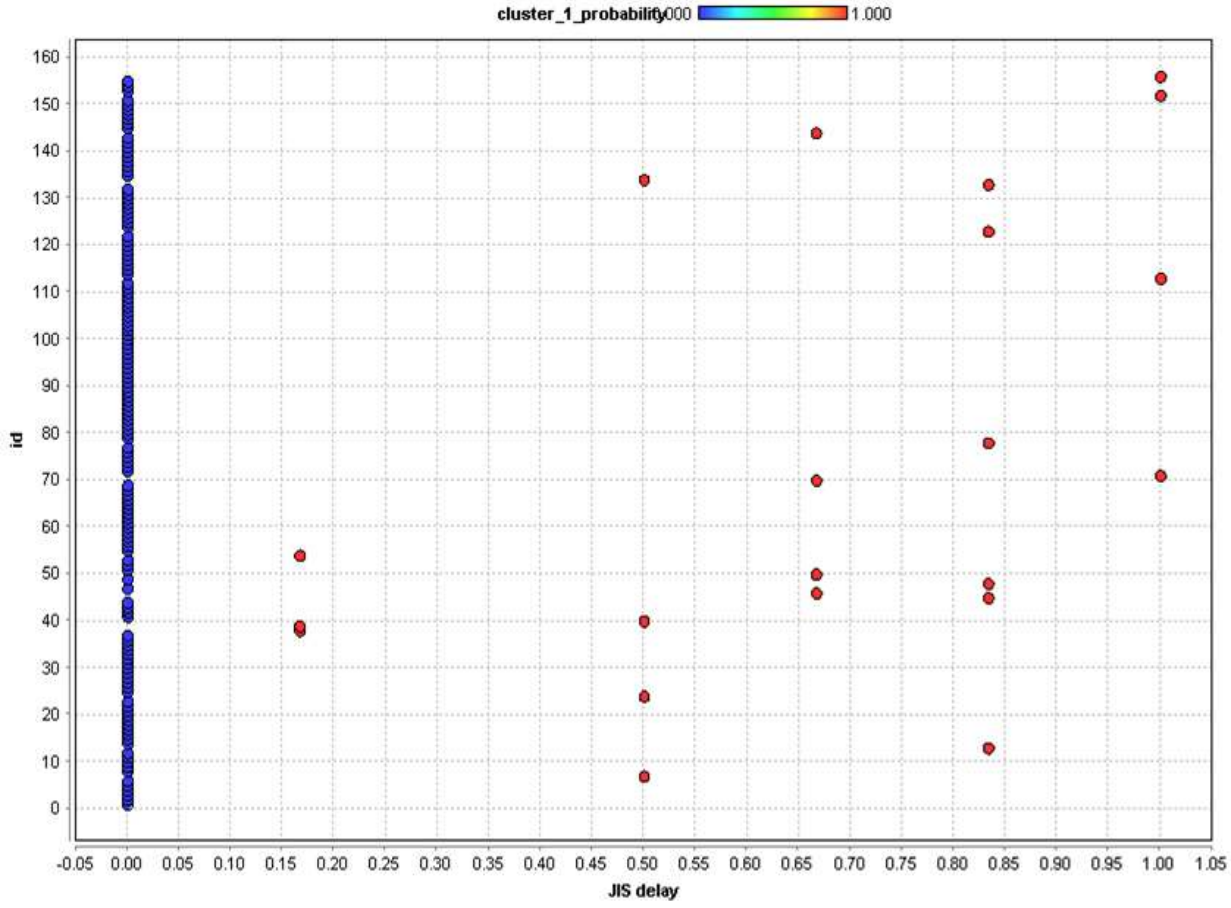

Calculating P(X) is a very complicated procedure, but RapidMiner provides a convenient function that user can obtain the P(X) values after every modeling process run. Following, it will introduce the EM modeling step by step. As illustrated in **Figure 4.3.5**, the process of EM modeling with RapidMiner is similar to the k-means and k-medoids, but by dragging the “EM” operator onto the process panel. Because it aims to group two clusters, the parameter “k” is set to “2”. The parameter “max optimization step” is defined as “1” according to the iterative process. Parameter “quality” specifies the quality that must be fulfilled before the algorithm stops and is set to “1.0E-10” which is sufficient for this case. Selecting the “show probabilities” will provide a list of probability values after the process run. “Initial distribution” is set to “k-means run” so that the clustering process will indicates the initial distribution of the centroids as k-means algorithm. After setting parameters, it begins to run the process and the result of the first step is shown in: 70 objects are assigned to the cluster 0; 86 objects are assigned to the cluster 1. Exporting the list of the probabilities of grouping clusters from RapidMiner in the MS Excel sheet enables the calculation of E value as illustrated in **Figure 4.3.4**. **Tabel 4.3.4** summarizes the E values of all optimization steps. The first step obtains E=295,76. Certainly it needs to run the process again in order to compare the results of the different steps. For the further optimization steps, it only needs to set the parameter “max optimization steps” to the corresponding number. It is clearly to notice that the results of the first, second and third step are slightly different between each other, but from the fourth step the E performs better with value 311,05929. The E value of the fifth step is 315,3611 which is better than the result of the fourth step. From the sixth step, there is no object at all assigned to cluster 0. This situation leads to a decision that the iterative optimization process should stop. The reasons are listed as follows: Firstly, because the intention of EM clustering for this thesis is to create two clusters, a cluster without an object is meaningless.

Secondly, the E value of the fifth step already provided the maximal E value. Therefore, it can draw a conclusion that the result of the fifth step is taken as the final result of EM clustering. **Figure. 4.3.6** visualizes the final result that cluster 0 is in blue and cluster 1 is in red. The results of the complete EM modeling are documented in the file "EM Clustering".

**Tabel 4.3.4. E Values of EM Clustering**

| Optimization Step | Sample Size | Cluster | Probability | E = $\sum$ Log (P(X)) |
|---|---|---|---|---|
| 1 | 70 | Cluster 0 | 69,68102476 | 295,7636229 |
| | 86 | Cluster 1 | 86,31897524 | |
| 2 | 68 | Cluster 0 | 71,49754288 | 295,6143512 |
| | 88 | Cluster 1 | 84,50245712 | |
| 3 | 70 | Cluster 0 | 79,97668269 | 295,1105926 |
| | 86 | Cluster 1 | 76,02331731 | |
| 4 | 21 | Cluster 0 | 124,1190876 | 311,0592902 |
| | 135 | Cluster 1 | 31,88091237 | |
| **5** | **21** | **Cluster 0** | **134,9984658** | **315,3611013** |
| | **135** | **Cluster 1** | **21,00153425** | |
| 6 | 0 | Cluster 0 | n.a | n.a |
| | 156 | Cluster 1 | n.a | |

**Figure. 4.3.6. Visualization of EM Clustering Result of the 3rd Optimization Step**



### 4.3.2 Evaluation

After the documentation of the data modeling results, the next step is to evaluate the modeling results. It is not easy to examine the results of clustering algorithms, because they are unsupervised methods and the expected model or pattern is totally unknown (section **2.2.3**). Another difficulty is that different clustering algorithms needs different evaluation methods because of their individual theoretical backgrounds.

Generally, k-means and k-medoids have most common theoretical backgrounds, namely grouping objects in a cluster based on distance measures. Therefore, they can be evaluated by cluster compactness cost function (**F. 2.3-10**) and cluster quality function I (**F. 2.3-14**). Cluster quality function II (**F. 2.3-15**) is not needed here, because k is predefined as "2" and this function is only meaningful to apply in the case that there are more than two clusters to group. Furthermore, RapidMiner provides the cluster centroid data after each modeling process run so that the distance between the cluster centroid and their objects can be measured. There are several evaluation operators as "similary to data", "cluster distance performance", but it cannot present the results in the way as mentioned in **section 2.3.3**. In order to follow the structural logic of this thesis, it decides to carry out this evaluation in MS Excel sheet where the results of each optimization step are saved and can be implemented following the introduced mathematical functions. As discussed in **section 2.3.2**, data modeling process accompanies results evaluation which is made by comparing the cluster quality of the old and new clusters, because cluster quality decides whether the iterative process of cluster grouping should proceed further or not.

#### i. Calculation of the Cluster Cost

The first evaluation method is cost function based on the distance measurement. For this case, the Euclidean Distance Function is taken and the proceeds as follows:

**Figure 4.3.7. Example for K-Means Result**

| Optimization step | ID | Cluster | Assembly Line | Centroid | (Centroid-x)² | JIS Weekdays | Centroid | (Centroid-x)² | Deviation Re/Scheduled | Centroid | (Centroid-x)² | JIS Delay | Centroid | (Centroid-x)² | dist(Centroid, x) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 80 | cluster_0 | 0,5 | 0,133 | 0,134689 | 0 | 0,599 | 0,358801 | 0,571 | 0,415 | 0,02434 | 0 | 0,089 | 0,007921 | 0,725084133 |
| 1 | 88 | cluster_0 | 0,5 | 0,133 | 0,134689 | 0,833 | 0,599 | 0,054756 | 0,571 | 0,415 | 0,02434 | 0 | 0,089 | 0,007921 | 0,470852418 |
| 1 | 96 | cluster_0 | 0,5 | 0,133 | 0,134689 | 0,5 | 0,599 | 0,009801 | 0,714 | 0,415 | 0,0894 | 0 | 0,089 | 0,007921 | 0,491743836 |
| 1 | 99 | cluster_0 | 0,5 | 0,133 | 0,134689 | 0,5 | 0,599 | 0,009801 | 1 | 0,415 | 0,34223 | 0 | 0,089 | 0,007921 | 0,703303633 |
| 1 | 100 | cluster_0 | 0,5 | 0,133 | 0,134689 | 0,167 | 0,599 | 0,186624 | 0,714 | 0,415 | 0,0894 | 0 | 0,089 | 0,007921 | 0,647020092 |
| 1 | 49 | cluster_1 | 0,5 | 0,788 | 0,082944 | 0,833 | 0,788 | 0,002025 | 0 | 0,149 | 0,0222 | 0 | 0,092 | 0,008464 | 0,340049996 |
| 1 | 52 | cluster_1 | 0,5 | 0,788 | 0,082944 | 0 | 0,788 | 0,620944 | 0 | 0,149 | 0,0222 | 0 | 0,092 | 0,008464 | 0,857060675 |

As shown in **Table 4.3.5**, k-means performs better cost value than k-medoids. That means the member of the clusters which are grouped by k-means algorithm has the shorter distance to each other than the member of the clusters which are grouped by k-medoids algorithm. However, this conclusion is only drawn in this case. Most of all, it depends on the data sample size, number of cluster k and amount of attributes.

**Table 4.3.5. Evaluation- Cost Values**

| Optimization Step | K-Means: Cost= ∑ dist(Centroid, x) | K-Medoids: Cost= ∑ dist(Centroid, x) |
|---|---|---|
| 1 | 119,639586 | 132,0628371 |
| 2 | 85,8763646 | 121,8417540 |
| 3 | 85,5548433 | 121,8417540 |

### ii. Calculation of the Cluster Quality

The second evaluation method is quality measurement function for measuring the quality of a cluster, but distance measurement is the precondition. Therefore, there is a common calculating step between the distance measures and quality measures and it begins with the step where the result of "∑ (Centroid-x)²". All of these values of objects should be summed up together, and then divided by 1. The result is "cluster quality" assigned in "G= 1/∑ (Centroid-x)²". Bigger this value is, better the cluster quality is. As listed in **A.1**, the second optimization step of k-means results a slightly better cluster quality than the first step. The cluster quality of the third step is almost equal to the second step, but the cluster centroid moves further on the third step, so it can draw a conclusion that the cluster quality of the second optimization step performs better.

**Table 4.3.6. Evaluation- Quality Values**

| Optimization Step | K-Means: G= 1/∑ (Centroid-x)² | K-Medoids: G= 1/∑ (Centroid-x)² |
|---|---|---|
| 1 | 0,0097401 | 0,007594377 |
| 2 | 0,0190847 | 0,008658509 |
| 3 | 0,0191921 | 0,008658509 |

Because quality evaluation is based on the distance measurement, the quality value of k-means is better than the quality value of k-medoids (**Table 4.3.6**). This result can be predicted when the cost value of k-means is better than the cost value of k-medoids.

### iii. EM Clustering

For EM clustering is a little different that the results are evaluated by the E value which is already carried out during the optimization process. Another reason that EM cannot be evaluated by cost function is that the EM operator in RapidMiner does not present the cluster centroid data after the modeling process run, but model data in the form of covariance matrix and the calculation result of this matrix is "0". Therefore it cannot evaluate the EM model by distance or quality measures functions as k-means and k-medoids.

### 4.3.3 Statistical Test and Ranking Results

For data modeling the three clustering algorithms, k-means, k-medoids and EM Clustering have been used, because they can observe the data insight more preciously to model the same data samples with the different algorithms so that the data analysis results can provide the enhanced conclusion. To achieve this, it needs to choose a statistical test method which should suite the test three data analysis results. Therefore, ANOVA is chosen to be used. Just as the theoretical interpretation of hypothesis tests showed in **section 2.3.4**, there are two types of errors. Type I error occurs when there are no differences between the results of the three clustering algorithms, because of randomness it could be mistakenly decided that there were differences. This is usually the more serious error and the one controlled at 5%. Thus, when there are no true population differences among these three clustering results, the difference only 5% of the time will wrongly be found. When there are true differences among the population results, the type II error might be made and lead to the wrong decision that there are no differences. This error is not easy to control, because it depends on how different are the means of the clustering models actually from one another. Subsequently, the *F*-test is required to find the significance first which is resulted by dividing the *average square between the clustering models* by the *average square within the clustering models*. The complete implementation steps of calculation are documented in the MS Excel file "ANOVA" in detail and are not going through here, but focusing on the analysis of the test results. As presented in **Figure.4.3.8**, the *F*-values of cluster attribute "Deviation Re/Scheduled" are smaller than 1 so that the *F-value table 5%* shows "not statistically significant". This conclusion implies that there is no significant relation between the generated transaction data "Deviation Re/Scheduled" and "JIS Delay". The reason could be that the relation of "Deviation Re/Scheduled" and "JIS Delay" were not formulated in a correct mathematical or statistical way. Another reason could be that the simulation time was too short to generate the data in a sufficient size.

**Table 4.2.7. *F* -Test Result**

| ***F* -Value = Average Square between Clustering Models / Average Square with Cluster Models** | | | ***F* -Critical Value 5% ( Degrees of Freedom : 3-1=2)** | |
|---|---|---|---|---|
| **Attribute** | **Cluster 0** | **Cluster 1** | **Cluster 0** | **Cluster 1** |
| **Assembly Line** | 20,664 26477 | 16,68587326 | 3,49 | 3,68 |
| **JIS Weekdays** | 7,471370105 | 5,605127566 | 5,79 | 5,79 |
| **Deviation Re/Scheduled** | 0,66303472 | 0,436728457 | Not statistically significant | Not statistically significant |
| **JIS Delay** | 13,01622786 | 41,20154526 | 3,68 | 3,07 |

Next it comes to the cluster models comparison to see how different the clusters models are. As introduced in **section 2.3.4**, ANOVA provides a statistical test of whether or not the means of several data groups are all equal, and therefore generalizes *t*-test to more than two data groups by comparing two, three, or more means, because doing multiple two-sample t-tests would result in an increased chance of committing a type I error. As **A.10** provided, ANOVA has been though all

the attributes values test among the three cluster models and no single interval contains 0, therefore the conclusions can be drawn as follows:

- K-means cluster model and k-medoids cluster model are significantly different from each other.
- K-means cluster model and EM cluster model are significantly different from each other.
- K-medoids cluster model and EM cluster model are significantly different from each other.

Finally, it will see which cluster model performs the best in terms of the classification quality. The objective of clustering algorithms is to classify which cluster represents “JIS regular”, which represents “JIS disturbed”. Firstly, it needs to make a declaration of the values of “JIS Delay”. “0” stands for “JIS regular”, because the delivery is arrived in time. The rest values interpret the “JIS disturbed”. Then, it will calculate for each cluster model how many values of “JIS Delay” not being equal to “0” are assigned to cluster 0. The result is taken as “wrong classification”. If a “JIS Delay” value with “0” is assigned to cluster 1, then it will be taken as “wrong classification”. Calculating the success quotient of each cluster model by **F. 2.3-18**, the results are ranked as listed in **Table. 4.3.8**: k-means cluster model performs with 0,538 success quotient, k-medoids cluster model with 0,955, and EM cluster model shows the best result with 1, namely 0 error.

**Table 4.3.8. Ranking Cluster Models**

| **K-Means Cluster Model** | | |
|---|---|---|
| **Optimization Step: 3** | **Wrong Classification** | **Success Quotient** |
| Cluster 0: JIS regular | 11 | 0,538461538 |
| Cluster 1: JIS disturbed | 61 | |
| Total Sample Size | 156 | |
| **K-Medoids Cluster Model** | | |
| **Optimization Step: 3** | **Wrong Classification** | **Success Quotient** |
| Cluster 0: JIS regular | 7 | 0,955128205 |
| Cluster 1: JIS disturbed | 0 | |
| Total Sample Size | 156 | |
| **EM Cluster Model** | | |
| **Optimization Step: 5** | **Wrong Classification** | **Success Quotient** |
| Cluster 0: JIS regular | 0 | 1 |
| Cluster 1: JIS disturbed | 0 | |
| Total Sample Size | 156 | |

From the rank of the cluster models, k-medoids and EM clustering perform with the robust results. Therefore, from the viewpoint of statistics, k-medoids and EM clustering can be treated as the robust method with an ability to withstand outliers. K-means can make it difficult to detect these

outliers, so it can be treated as a nonrobust method. K-means algorithm is sensible to outliers, because the initial centroids are not the real data samples and it can lead to the distorted cluster. Much more the cluster distorted are, smaller the success quotient of the right classification the cluster shows. K-medoids algorithm results robust clusters, because the selected centroids are the representative objects of the cluster and the outliers cannot lead to the distorted clusters. **Figure 4.3.3** illustrates the clusters are grouped with the closed objects without no objects which belong to the other cluster.

The EM cluster model presents the statistical observation in the objects assignment in a cluster based on the probabilities and presents the best clustering results in classification than the other tow in this case. The reason can also be concluded from the viewpoint of statistics, because EM clustering is based on the statistical theory of probability. Probability helps understand the behavior of random systems. An object is assigned in a cluster can be regarded as an event and the probability of this event is an indication of how likely it is that this event will happen and gives a measure of the event's predictability. This is an interesting argument that EM clustering explores a data model in a known random system, whereas this explored data model is unknown, because EM clustering is an unsupervised clustering algorithm.

However, it does not mean the EM clustering performs the best in general application from all perspectives. The results depend on the focus of the evaluation, e.g. cost performance, quality performance, and success quotient of the right classification as well as a range of the examination concepts of clustering algorithms. The focus of this thesis lies on the classification, so in this case EM clustering can be regarded as the best method of these used clustering algorithms. Furthermore, the numbers of the attributes, data sample size and the predefined k number can also effect on the clustering results. One approach of thesis is to use both a robust and a nonrobust cluster algorithms, using the robust algorithm as a check on the validity of the regular classification by giving up some sensitivity to outliers, yet reporting the nonrobust cluster algorithm but more sensitive answer as the final value provided there is general agreement between the these three cluster algorithms. Therefore, k-means cluster model performs better cost and quality value than the k-medoids cluster model (**Table 4.3.5; Table 4.3.6**).

# 5 Prototypical Illustration

From **chapter 2** to **chapter 4**, it has worked through with the subject "Conceptual Approach to Knowledge Discovery in Supply Chain Transaction Data by Applying Data Farming". This chapter will present the prototypical application of the proposed approach in the case of the 1TSs delivery processes in the automotive industry, following the established conceptual approach (**Figure 4.1**) which is based on the theoretical state as introduced in **chapter 3**. First, the current 1TS delivery situation will be formulated as introduced in **chapter 2** and triggers the idea for applying data farming in the SCs field with the orientation on the collection of transaction data in terms of the simulation task description (**Table 4.1.1**). Second, the summarized information of the conceptual model (**Figure 2.1.3)** will be provided and should be interpreted in the executable model with Tecnomatix Plant Simulation for generating transaction data. After each simulation run, the output data are validated by the chi-squared test (**Table 4.1.7**) and documented in the extra file. How many simulation runs should be executed sufficiently is decided by the confidence intervals test (**A.4**). If the significance level $\alpha$ which calculated by the confidence intervals test is smaller than 5%, the output data will be transformed in the required data format for the clustering algorithms (**Chapter 4.2**). Afterwards, as discussed in **section.4.3.1** the different clustering algorithms will be used for modeling the same output data in order to compare the data modeling results and select the best data model (**section 4.3.2**). Finally, by using ANOVA test (**section. 4.3.3**), it enables to draw a conclusion which clustering algorithm performs the best results for classifying the clusters in regular and disturbed JIS delivery processes.

## 5.1 1st Tier Supplier Delivery in the Automotive Industry

The prototypal case refers to the push strategy from the viewpoint of the 1TSs who have to keep a safety stock level before the VM releases the JIS call-off signal. As discussed in **chapter 2.1**, currently the strategic goals that the German VMs pursue are to shorten customer order delivery times of the individual car configuration, to keep promised delivery dates with high reliability and to allow customers to change their wishes about the car configuration in a short term. In order to achieve these goals, the SC collaboration with their 1TSs has to be solid and effective in terms of the JIS delivery which should be operated just in several hours. On the other hand, because of the cost pressure most German VMs have expanded their 1TSs network in the lower cost countries predominantly. This results a series of barriers in the way of the SC goals. Firstly, the culture issues lead to the inhomogeneous configuration processes and delivered items with insufficient quality. Secondly, the demand information from the side of the VMs can be delayed to share with the 1TSs, because of the heterogeneous SCMSs (**section 2.1.3**). Thirdly, the long transport distance implies the risk in transport disturbance. In contrary, the German local 1TSs perform with a relative high-leveled reliability. Therefore, in the prototypical simulation model the VM has JIS deliveries from four foreign 1TSs as and from four regional 1TSs as illustrated in **Figure 2.1.3**.

The international 1TSs have to deliver the parts at the VMI stock firstly, and the other four regional 1TSs can deliver their items from their own stocks directly. However, the JIS disturbances happen occasionally and that could be caused by different reasons in terms of the complicated SCs. For example, that could be resulted by the rescheduled delivery from the side of the final customer, or

from the side of the VM in terms of the changed assembly plan, or by disturbed transport. These JIS delivery disturbances may cause that the processes of assembly orders at VM are disturbed. Therefore, to obtain an observation of JIS delivery processes rules can help the decision-maker inspect the JIS processes performance and adjust the management parameters in order to avoid the JIS disturbances. For example, for which assembly line, or on which weekdays the deliveries the disturbance the most probably happen, so that a SC manager can adjust the assembly line capacity or reschedule the JIS delivery date based on the defined KPIs (**section 2.1.4**). To achieve this, it needs to obtain the relative data analysis and expert's opinions due to the processes disturbances so as to extract the knowledge for supporting decision-making. SC transaction data can record the information of the delivery processes disturbances. However, because of different data quality issues as manual mistakes while inputting data in SCMSs, or the technical disturbances as well as the different data formats from different data sources (**section 2.1.3**). That is why it will apply data farming for generating the expected transaction data in required quality for generalizing the performance rules of the SC operational processes.

## 5.2 Generating Supply Chain Transaction Data by Applying Data Farming

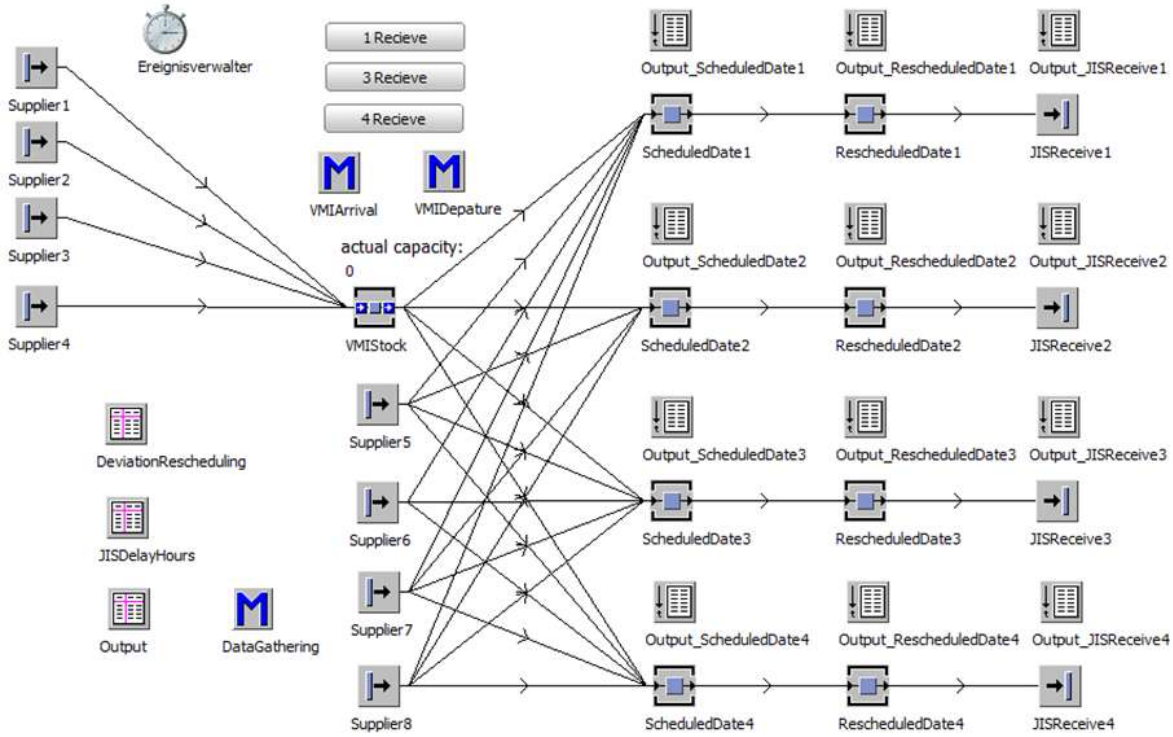
After formulating the current state of JIS delivery processes, it begins with SC transaction data farming following the "Procedure Model for Simulation Study with V&V"(**Chapter 4**). The first phase is the task definition based on the **Table 4.1.1**. The task content and selecting the application elements of "Data Farming Loop of Loops"(**Figure 3.2.1**) depend on the individual specification of SC processes. Accompanying V&V techniques, the results of this phase can be examined by *structured walkthrough* or *desk checking* and the validated results are documented to help orientate on the further procedure phases. The second phase *system analysis* follows the **section 4.1.1** with focus on the analysis of the depicted system. The following questions should be answered, according to the state of the JIS delivery disturbance:

- Which SC organization unites should be involved? Are the VMI logistic providers, the transport service provider or the intern car body suppliers also taken into account in the JIS delivery process?
- How many organizational unites should be described in the depicted system?
- Which operative actions should be processed? Should the dispatching process actions at VMI or the local 1TSs also be observed?
- Is the container flow also to be depicted in terms of KABAN system?
- Which transaction data should be treated as the key input data, namely the data seed, so that the generated data can provide the sufficient quality and size for seeking the process disturbance cases?
- How long the simulation time should be defined so that the out data can reflect the real system and help observe the transaction data insight?

After answering these questions the conceptual model should be established with the summarized information in details. These information contains the level of the simulation model details and should be examined by V&V techniques, recommended as *cause-effect graph* and *structured walkthrough*. The results of this phase should be documented as well for supporting in create a formal model in the next phase. At the same time the required input data should be

collected which leads to the following questions: Where are the sources of the data seed? Are the data seed in the SCMs available in sufficient quality? If not, the alternative information source from the SC expertise can be gathered by an interview or a brain storming meeting. In general, the SC experts should be involved entirely before the phase of the executable model (**section 2.1.2**). The selected attributes of the input data should be mapped and set in a required data format.

**Figure 5.1. Prototypical Executable Model**




Subsequently, it comes to the third phase *formal model* (**Table 4.1.6**) which is described in a formal way. Thereby, the input data should be given in the real data format or in the statistical distribution form in the executable model (**section**.**4.1.2**). To implement the fourth phase *executable model*, it follows the procedure as explained in section **section**.**4.1.3** and based on the given framework of the old simulation model. However, according to the programming experience and skills level as well as the functionalities of the data farming software, it could be carried out in alternative ways. **Figure 5.1** presents a simple alternative executable model scenario for generating the "Reschedulded_Date" and "JIS_Receive_Date". Unlike to the excusable model as illustrated in **Figure 4.1.4**, the element *SingleProc* can be represented for creating the output data, being connected with its own corresponding *TimeSequence* where the output data are written in details when each simulation run ends up. In order to gather all of the output data written in the *TimeSequence* in a convenient way, a method named "DataGathering" is programmed, then the automatic connection between each *TimeSequence* and the table "output" can be created.

The V&V technique for evaluating the result of this phase is processed by qui-squared test after each simulation run. This is also the precondition to fix the number of certain simulation run by using confidence intervals method. The results which should have passed the qui-squared test firstly are calculated in the cumulative way in order to indicate at which simulation run the significance level α is smaller than 5% (**Figure 4.1.1**).

## 5.3 Output Data Transformation

As presented in **section 4.2**, the simulation output data are in the different data format as “integer”, “datetime” and “time”. To analyze the output data with clustering algorithms, the data transformation is processed in two steps. Firstly, all of the output data need to be transformed in an equate data format, namely numerical and metric value. This step can be operated in MS Excel sheet or by the appropriate transformation operator of RapidMiner. Selecting the data transformation tool depends on the data size, functionality performance and user’s skills level. The second step is operated in the data modeling process by adding an operator “Normalization” so that all attribute values are normalized in the specified range (min, max) and transformed into interval (0,1) as interpreted in **A.5**. There are several common functions for data transformation between the RapidMiner and MS Excel, the user can decide which is convenient or effective according to the data sample size and data analysis requirements.

## 5.4 Classification of Delivery Processes by using Clustering Algorithms

Data modeling is carried out by different cluster algorithms in order to observe the best analysis results. As presented in **section 4.3**, first of all it needs to choose the cluster algorithms according to the analysis intension. If the number of clusters k is predefined, the partitioning clustering algorithms will be selected. Data modeling is presented by graphical display with the analysis software tool where the corresponding modeling parameters are set and follows the individual optimization principle of clustering algorithms as introduced in **section 4.3.1**, until the model performs with the best result. For k-means and k-medoids algorithms, the cost function and the quality function are used in terms of the distance measurement. For EM clustering which focuses on the probability of an object assignment of the clusters, it uses the expectation and maximization functions. Every modeling step is recorded in the format of documents, e.g. **A.6** and **A.8**, and models in electronic format.

The task of the modeling process is to find the best cluster which interprets the meaning of the output data most insightfully. **Section 4.3.2** has already presented that after each optimization modeling process, an overview of the modeling results can be visualized by RapidMiner as well as the cluster model summary in detail. Model evaluation actually has got involved during the modeling processes. For k-means and k-medoids models, the smaller value calculated by the cost function expresses the better cluster quality and is taken as the optimal result. For EM clustering, the maximal value of the expectation function is regarded as the best cluster quality.

The last task is to compare the different cluster models and rank their performances, by carrying out a complicated procedure of a statistical test. ANOVA comes to analyze more than two data models and fits to the situation of this case. As discussed in **section 4.3.3**, it begins with *F*-test for drawing the first conclusion whether the corresponding attributes value are significance for the cluster grouping. Then the comparison between cluster models is implemented by using ANOVA and the conclusion is drawn in two ways “The cluster models are significant different from each other” or “The cluster models are not significant different from each other”. To find out which cluster model provides the best classification of “JIS regular” and “JIS disturbed”, it can be achieved by using “error quotient” or “success quotient” function (**Figure. 4.3.9**).

# 6 Conclusion and Further Work

The present work has demonstrated a conceptual approach to knowledge discovery in supply chain transaction data by applying data farming. It began with the theoretical background and current state in SCM in German automotive industry, knowledge discovery techniques and data farming. For shooting the problem field of the quality issues of supply chain transaction data, a new approach was developed by combining the knowledge discovery and data farming accompanying V&V processes. By implementing a SC simulation study about JIS delivery processes in the automotive industry, it presented that supply chain transaction data were generated by using the plant simulation and the output data of each simulation run were examined by the chi-squared test. To fix the replication of simulation run, the confidence intervals method was used. In order to rescale the output data in the suitable format that clustering algorithms require, data transformation was processed. Subsequently, three partitioning clustering algorithms: k-means, k-medoids and EM clustering were used for analysis of the same output data for classification of the regular and the disturbed JIS deliver processes.

The first milestone of this thesis was to generate the transaction data in the expanded simulation model, instead of obtaining the expected output data, the causes of failure were summarized. The second milestone was data transformation was accomplished by using MS Excel and RapidMiner operator. The last milestone was achieved with the conclusion that it is possible to classify the JIS transaction data in a “JIS regular” cluster and a ”JIS disturbed ” cluster. With the test results this thesis shows the application of this approach is possible. However, because of the high-leveled complexity and correlative factors in the SCs, the high accuracy of the simulation input data is a challenge. The result of the output data analysis with clustering algorithms shows that the supply chain transaction data can be classified in the expected groups.

The first challenge of this thesis was absence of the sample size of real data and it required a collection of the research studies for making an estimation to set up the simulation input data. This reflects the conflict between the time effort and result accuracy. The second challenge was the combination of the data farming concept with the procedure of a simulation study model, because there are several common procedure steps. This needed to make a decision which application method of each concept should be chosen to establish a new conceptual approach for the application in SCs. The third challenge was to choose the appropriate mathematical functions and statistical methods to evaluate and compare the different cluster models in terms of the unsupervised character and different modeling principles of the clustering algorithms. Finally, the carefully and finely dealing with an amount of the documentations is significant to the results analysis and improvement. Basically, the present work has achieved the aim of setting up a conceptual approach to knowledge discovery in supply chain transaction data by applying data farming.

For the further research studies in the field of SC simulation, several interesting perspectives based on the present work are recommended as follows:

1. The potential of expansion of the SC simulation model:
   In terms of the JIS delivery in the automotive SCs, container flow is regarded as an independent management field and plays a significant role for processes performance.

Therefore, the SC simulation model can be expanded with the container flow for analysis of the relation between the absence of the containers and JIS delivery performance. On the other hand, the stocks of VMI provider and regional 1TSs are taken as the KANBAN buffers. For this reason, the relation between the stock capacity and JIS delivery performance can be studied further with orientation on the delivered items. Furthermore, based on the current SC model, the 2nd tier suppliers and transport service providers can also be adopted to increase the model complexity.

2. The potential of design of experiment study:
   First, the simulation experiment can only examine the output data which are generated within the simulation time. Longer the simulation time is, more data are generated. It seems that the relation between the simulation time and the level of the output data accuracy can be worked on further. Second, based on the current SC simulation model it can be studied further how many attributes should be added for increasing the complexity which should be optimal in terms of model fit. Third, the input data of the simulation model in this thesis were set in normal distribution and discrete empirical distribution, but they can also be set in the other statistical distributions which simulation tool provides. An argument on the results of different statistical distributions is supposed to be a topic of the further study.

3. The potential of clustering study:
   First, based on the current work the simulation output data can be classified further by the hieratical clustering algorithms such as agglomerative and divisive clustering algorithms. This perspective is to propose a study on the connection among the clusters which classify the SC processes performances. Second, the output data can also be analyzed with the method of cluster-self organized map which is combined with clustering and artificial neural networks. The results comparison of cluster-self organized map and the clustering algorithms which this thesis used can be worked on further. Third, to classify the processes performance, the methods of decision tree and neural net works can be used. A further work on the comparison of the classification SC processes performance among decision tree, artificial neural networks and clustering is to recommend.

In the time of economic globalization and Internet of Thing, SCs have been being facilitated in a rapidly changing environment and the heterogeneous SCMSs of the word-wide SC participants trend to be distributed widely further. Behind these the existing and potential risk factors of SC disturbances are hidden. However, these risk factors can be indicated and identified by using data farming so as to extract the performance rules of the operational processes for supporting decision-making in a comparative economical effort.

# 7 Literature and Reference


Alicke, K.: Planung und Betrieb von Logistiknetzwerken- Unternehmensübergreifendes Supply Chain Management. Berlin Heidelberg : Springer-Verlag, 2005

Arndt, V.: Ereignisdiskrete Simulation einer Supply Chain zur Generierung von Transaktionsdaten. Dortmund, Technische Universität Dortmund, Fachgebiet IT in Produktion und Logistik, Masterarbeit, 2014, http://www.itpl.mb.tu-dortmund.de/cms/de/forschung/Abschlussarbeiten/MA_2014_Arndt.pdfl

Bennett, D.; Klug, F.: Logistics Supplier Integration in the Automotive Industry. In International Journal of Operations & Production Management, Vol. 32 Iss 11, 2012, pp. 1281 – 1305

Boppert, J.; Walch, D.: Adaptives Wissensmanagement – Abschöpfung und gezielte Nutzung von Mitarbeiter Know-How. In: Neue Wege in der Automobillogistik Die Vision der Supra-Adaptivität, Günthner, W.-A. (ed). Berlin Heidelberg: Springer, 2007, p. 407-408

Borade, A.B.; Sweeney, E.: Decision support system for vendor managed inventory supply chain: a case study. International Journal of Production Research, 53:16, 2015, p. 4789-4818

Brady, A.; Keung, J.; Hihn, J.; Williams, S.; El-Rawas, O.; Green, P.; Boehm, B.: Learning Project Management Decisions: A Case Study with Case-Based Reasoning versus Data Farming IEEE Transactions on Software Engneering, Vol. 39, No. 12, 2013

Chandra, C.; Tumanyan, A.: Organization and Problem Ontology for Supply Chain Information Support System. In: Data and Knowledge Engineering (61) Heft 2. Elsevier B.V , 2007, p. 263-280

Chen, M.-C.; Huang, C.-L.; Chen, K.-Y.; Wu, H.-P.: Aggregation of orders in distribution centers using data mining. Elsevier: Expert Systems with Applications 28, 2005, p. 453–460

Cleve, J.; Lämmel, U.: Data Mining. München: Oldenbourg Verlag, 2014

Christopfer, M.: Logistics and Supply Chain Management. Harlow [u.a.], England: Financial Times Prentice Hall, 2011

Desouza, K.C: Managing Knowledge with Artificial Intelligence - An Introduction with Guidelines for Nonspecialists. Westport, Irland: Quorum Books, 2002, p. 1-11

Düsing, R.: Knowledge Discovery in Databases. In: Chamoni, P.; Cluchowski, P. (eds.): Analytische Informationssysteme. Business Intelligence Technologien und Anwendung, 4. Aufl. Berlin [u.a.]: Springer, p. 282-295

Fayyad, U.M.; Patetsky-Shapiro, G.; Smyth, P.: From Data Mining to Knowledge Discovery in Databases. American Association for Artificail Intelligence 17, 1996, p. 37 – 54

Gehr, F.; Hellingrath, B.: Logistik in der Automobilindustrie : Innovatives Supply Chain Management für wettbewerbsfähige Zulieferstrukturen. Berlin, Heidelberg: Springer-Verlag, 2007

Gleißner, H.; Femerling, Ch.: Logistik. Wiesbaden: Gabler, 2008

Göbl, M.; Froschmayer, A.: Logistik als Erfolgspotenzial: The Power of Logistics. Wiesbaden: Gabler Verlag, 2011

Göpfert, I.; Braun, D.: Stand und Zukunft des Supply Chain Managements in der Automobilindustrie – Ergebnisse einer empirischen Studie. In: Automobillogistik – Stand und Zukunfstrends, Göpfert, I.; Braun, D.; Schulz, M. (eds). Wiesbaden: Springer, 2013, p. 27-36

Göpfert, I.; Grünert, M.: Logistiknetze der Zukunft : das neue Hersteller-Zulieferer Verhältnis in der Automobilindustrie. In: Görfert, Ingrid (eds): Logistik der Zukunft. Wiesbaden: Gabler, 2009, p. 143

Göpfert, I.; Braun, D.: Wirkungen von Supply-Chain-Management-Maßnahmen bei Automobilzulieferern und-herstellern. Ergebnisse einer empirischen Studie. In Logistik der Zukunft - Logistics for the Future, Göpert, I. (eds) : Wiesbaden: Springer Fachmedien, 2016, p. 219-230

Görgülü, Z.-K. and Pickl, S.: Adaptive Business Intelligence: The Integration of Data Mining and Sytemes Engineering into an Advanced Decision Support as an Integral Part of the Business Strategy. In: Business Intelligence and Performance Management : Theory, Systems and Industrial Applications, Rausch, P.; Ayesh, A.; Sheta, A.-F (eds). London: [u.a.]: Springer, 2013, p. 43-56

Gray, J.; Reuter, A: Transaction Processing: Concepts and Techniques. San Mateo, Calif: Morgan Kaufmann, 1993

Grunewald, M.: Ein Ansatz zur Integration von Bestandsmanagement und Tourenplanung. Wiesbaden: Springer Fachmedien, 2015, p. 1-22

Günther, H.-O. ; Mattfeld, D.-C, ; Suhl, L.: Supply Chain Management und Logistik : Optimierung, Simulation, Decision Support. Heidelberg: Physica-Verlag, 2005

Günther, H.-O; meyr, H. (eds): Supply Chain Planning – Quantitative Decision Support and Adbanced Planning Solutions. Berlin Heidelberg: Springer, 2009

Harnisch, St.: Einkauf und Einsatz von Unternehmenssoftware: Empirische Untersuchungen zum anwenderseitigen Software-Lebenszyklus . Wiesbaden: Springer, 2015

Hensen, Ch.: Nachhaltige Effizienzsteigerung durch den Einsatz eines wertorientierten Process Performance-Managements: Am Beispiel der Automatisierung des OTC-Prozesses. Hamburg: Diplomica Verlag, 2008

Hofmann, M.: Simulation-based Exploratory Data Generation and Analysis (Data Farming): A Critical Reflection on Its Validity and Methodology. In: The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology, 10. 2013, Heft 4, p. 381 - 393

Horne, G.E.; Meyer, T. E.: Data Farming: Discovery Surprise. Proceedings of the 2004 Winter Simulation Conference, 2004

Horne, G.E.; Meyer, T. E.: Data Farming: Discovering Surprise. Proceedings of the 2005 Winter Simulation Conference. IEEE Press 2005, p. 1082 – 1087

Huang, S.-M.; Hu, T.-C.: Knowledge Discovery for Supply Chain Management Systems: A Schema Composition Approach. In: Issues in Information System, 5.2004, P. 523 - 529

Ickerott, I.: Agentenbasierte Simulation für das Supply Chain Management. In: Informationsmanagement und Unternehmensführung. Schriften des IMU, Universität Osnabrück 2Lohmar [u.a.]: Eul, 2007

Juan, An.-A.; Faulin, J.; Grasmanc, Sc.-E.; Rabe, M.; Figueira, G.: A Review of Simheuristics: Extending Metaheuristics to Deal with Stochastic Combinatorial Optimization problems. Elsevier: Operations Research Perspectives 2, 2015, p. 62–72

Kleijnen, J.P.C.; Sanchez, S.M; Lucas, T.W.; Cioppa, T.M.: State-of-the-Art Review: A User’s Guide to the Brave New World of Designing Simulation Experiments. Informs Journal on Computing. Vol. 17, No. 3, Summer 2005, pp. 263-289

Klug, F.: Logistikmanagement in der Automobilindustrie - Grundlagen der Logistik im Automobilbau. Berlin Heidelberg: Springer, 2010

Klug, F.: Optimaler Push/Pull-Mix bei der Produktionsplanung und -steuerung mit stabiler Auftragsfolge. In Automobillogistik – Stand und Zukunfstrends, Göpfert, I.; Braun, D.; Schulz, M. (2013) (eds). Wiesbaden: Springer, 2013, p. 84-105

Kropik, M : Produktionsleitsysteme in der Automobilfertigung. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2009, p. 93-94

Lau, H.C.W ; Zhao,Y. ; Chuang, N.S.H ; Ho, G.T.S. : Optimization of Supply Chain Design Based on Knowledge Discovery of Distribution Network. International Journal of Production Economics, 122. 2009, Heft 1, p. 176 - 187

Lewis, P.M.; Bernstein, A.; Kifer, M.: Databases and transaction processing: An Application Oriented Approach. Boston [u.a.]: Addison-Wesley, 2002, p. 15-28

Hedtstück, U.: Simulation Diskreter Prozesse. Berlin Heidelberg: Springer, 2013

Li, L.: Supply Chain Management: Concepts, Techniques and Practices: Enhancing Value Through Collaboration. Singapore [u.a.]: World Scientific, 2007

Lysons, K.; Farrington, Br.: Purchasing and Supply Chain Management. Harlow, England: Pearson Education Limited, 2012

Mangan, J; Lalwan, C; Butcher, T: Global Logistics and Supply Chain Management. Chichester: Wiley, 2008, p. 10-11

Mohammadi, L.; Fazlollahtabar, H.; Mahdavil, I.; Tajdin, A.: Data Mining on Return Items in a Reverse Supply Chain. Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management Bali, Indonesia, January 7 – 9, 2014, p. 1467-1472

Meyr, H.; Stadtler, H.; Kilger, Ch.: Supply Chain Management and Advanced Planning: Concepts, Models, Software, and Case Studies. Berlin, Heidelberg: Springer, 2015

Mortensen, O.; Lemonie, O.-W.: Integration between VM and TPL Providers. International Journal of Operations & Production Management. Vol. 28 No. 4, 2008, pp. 331-359

Nato Report (2014): STO Technical Report - Data Farming in Support of Nato. NATO Science and Technology Organization: https://www.cso.nato.int/pubs/rdp.asp?RDP=STO-TR-MSG-088

Prokhorov, D.: Computational Intelligence in Automotive Applications**.** Berlin Heidelberg: Springer, 2008

Parshutin, S.: Managing Product Life Cycle with Multi-Agent Data Mining System . In: Advances in Data Mining. Applications and Theoretical, Hutchison, D.; Kittler, J.; Kanade, T. (eds) . Berlin, Heidelberg: Springer, 2010, p. 308-322

Rabe, M.; Scheidler, A.-A.: An Approach for Increasing the Level of Accuracy in Supply Chain Simulation by using Patterns on Input Data. Proceedings of the 2014 Winter Simulation Conference, 2014, p. 1897-1906

Rabe, M.; Scheidler, A.-A.: Farming for Mining -Decision Support by Simulation in Supply Chain Management. In Rabe, M.; Clausen, U. (eds.) Simulation in Production and Logistics 2015. Stuttgart: Fraunhofer IRB Berlag, 2015

Rabe, M.; Spieckermann, S.; Wenzel, S.: Verifikation und Validierung für die Simulation in Produktion und Logistik: Vorgehensmodelle und Techniken. Berlin Heidelberg: Springer, 2008

Robinson, St.: Simulation: The Practice of Model Development and Use. Chichester, Eng, Hobocken, N.J: Wileey, 2004, p. 154

Schulz, M.; Göpfert, I.; Wellbrock, W.: Trends in der Automobillogistik. In: Automobillogistik – Stand und Zukunfstrends, Göpfert, I.; Braun, D.; Schulz, M. (2013) (eds). Wiesbaden: Springer, 2013, p. 1-21

Schulze, U.: Informationstechnologieeinsatz im Supply Chain Management. Wiesbaden: Gabler, 2009, p. 73-90

Schweppe, F.: Supply Chains in der Globalisierung - Birgt weltweite Beschaffung "trojanische Pferde" für die Automobilindustrie?. Stuttgart: Fraunhofer IRB Verlag, 2008

Setiawan, F. A., Wibowo, W. C.; Ginting N. Br: Handling Uncertainty in Ontology Construction Based on Bayesian Approaches: A Comparative Study. In: Intelligence in the Era of Big Data, Intan, R.; Chi, Ch.-H.; Palit, H. N. ;Santoso, L. W. (eds). Berlin Heidelberg: Springe 2015, p. 235

Siegel, A.F.: Statistics and Data Analysis – An Introduction. New York [u.a.]: Wiley, 1988

Spille, J.: Typspezifisches Risikomanagement für die Beschaffung von Produktionsmaterialien in der Automobilzulieferindustrie. Aachen: Shaker Verlag, 2009, p.105-p.107

Staud, J. L.: Datenmodellierung und Datenbankentwurf: Ein Vergleich aktueller Methoden. Berlin Heidelberg: Springer 2005

Trautmann, A.: Multiagentensysteme im Internet der Dinge –Konzepte und Realisierung. In: Internet der Dinge. Bullinger, H.-J. and Hompel, M.(eds). Berlin Heidelberg: Springer, 2007, p. 281-294

Träger, D.; Wellbrock, W.; Kanowski, K.-D.: Tier-n Management – Innovatives Supply Chain Management bei der Daimler AG. In: Automobillogistik – Stand und Zukunfstrends, Göpfert, I.; Braun, D.; Schulz, M. (2013) (eds). Wiesbaden: Springer, 2013, p. 40-56

Turban, E.; Sharda, R.; Delen, D.: Decision support and business intelligence systems. Boston [u.a.]: Pearson, 2011

Wang, J.: Information Technologies, Methods and Techniques of Supply Chain Management. US: Business Science Reference [u.a.] , 2012, p. 1-10

Wetzstein, B.; Ma, Z.; Leymann, F.: Modeling Key Performance Indicators. In: Business Information Systems, Fensel and Abramowicz (eds). Berlin, Heidelberg: Springer, 2008, p. 231-232

Wiendahl, H.-P.; Betriebsorganisation für Ingenieure. München Wien: Carl Hanser, 2008, p. 285-286

Wildemann, H.: Partnerschaftliche Prozessintegration am Fallbeispiel Just in Sequence. In: Neue Wege in der Automobillogistik: Die Vision der Supra-Adaptivität, Günthner, W.-A. (ed). Berlin Heidelberg: Springer, 2007, p.104-120

Wu, D.; Olson, D.L.: Supply Chain Risk, Simulation, and Vendor Selection. In: Int. J. Production Economics (114), 2008, p. 646– 655

Vahrenkamp, R.; Kotzab, H.: Logistik: Management und Strategien. München: Oldenbourg verlag, 2012, p.218-224

Waters, D.: Supply Chain Risk Management: Vulnerability and Resilience in Logistics. London [u.a.]: Kogan Page, 2011, p.43

Wilke, J.: Supply Chain Koordination durch Lieferverträge mit rollierender Mengenflexibilität: Eine Simulationsstudie am Beispiel von Lieferketten der deutschen Automobilindustrie. Wiesbaden: Springer, 2012, p. 59-70

# 8 Appendix

### A.1. Code: Data Generating

```
is
        RandomDeviation_Rescheduling: time;
        RandomJIS_Delay:time;
        y: integer;
do
        @.createAttr("Scheduled_Receive","datetime");
        @.Scheduled_Receive := Ereignisverwalter.AbsZeit;
        if  y:= 0 then y:=y+1;
        else  y:=y;
        RandomDeviation_Recsheduling:=z_dEmp(y, Deviation_Rescheduling);
        RandomJIT_Delay:=z_dEmp(y, JIS_Delay);
        @.createAttr("Deviation_Rescheduling","time");
        @.Deviation_Rescheduling:= Deviation_Rescheduling;
        @.createAttr("Rescheduled_Receive","datetime");
        @.Rescheduled_Receive:= @.Scheduled_Receive + @.Deviation_Rescheduling;
        @.createAttr("JIS_Delay","time");
        @.JIS_Delay:= RandomJIS_Delay;
        @.createAttr("JIS_Arrival","datetime");
        @.JIS_Arrival:= @.Rescheduled_Receive + @.JIS_Delay;
        end;
end;
```

### A.2. Code: Data Gathering

```
is
        x: integer;
do
        x := Output.YDim + 1;
        Output ["Scheduled_Receive",datetime] := @.Scheduled_Receive;
        Output ["Rescheduled_Receive", datetime] := @.Rescheduled_Receive;
        Output ["JIS_Arrival",datetime] := @.JIS_Arrival;
        @. createAttr(" ("Assembly_Line","Integer");
        if @.Location = Actual_Receive1 then
                @.Assembly_Line := "1";
        elseif @.Location =  Actual_Receive2 then
                @.Assembly_Line := "2";
        else
                @.Assembly_Line := "3";
        end;
        Output ["y", integer] := @.y;
end;
```

## A.3. Example for Chi-Squared Test on Output Data

| Output_Deviation Re/Schedulded | | | Input_Deviation Re/Schedulded | | | $\chi^2$ - Chi-Squared Test | | | |
|---|---|---|---|---|---|---|---|---|---|
| in Days | Observed Number = O | Frequency in % | in Days | Frequency | Frequency in % | Expected Number = E | O-E | $(O-E)^2$ | $\chi^2 = (O-E)^2/E$ |
| 0 | 144 | 79% | 0 | 130 | 76% | 139,1764706 | 4,823529412 | 23,26643599 | 0,167172194 |
| 1 | 2 | 1% | 1 | 2 | 1% | 2,141176471 | -0,141176471 | 0,019930796 | 0,009308339 |
| 2 | 8 | 4% | 2 | 4 | 2% | 4,282352941 | 3,717647059 | 13,82089965 | 3,227407886 |
| 3 | 7 | 4% | 3 | 9 | 5% | 9,635294118 | -2,635294118 | 6,944775087 | 0,720764203 |
| 4 | 2 | 1% | 4 | 5 | 3% | 5,352941176 | -3,352941176 | 11,24221453 | 2,100193924 |
| 5 | 7 | 4% | 5 | 8 | 5% | 8,564705882 | -1,564705882 | 2,448304498 | 0,285859729 |
| 6 | 8 | 4% | 6 | 6 | 4% | 6,423529412 | 1,576470588 | 2,485259516 | 0,386899375 |
| 7 | 4 | 2% | 7 | 6 | 4% | 6,423529412 | -2,423529412 | 5,87349481 | 0,914371903 |
| ∑ | 182 | 100% | ∑ | 170 | 100% | 182 | | | |
| | | | | | | Chi-Squared Statistic = $\sum (O-E)^2/E$ | | | 7,811977552 |
| | | | | | | Degrees of Freeedom | | | 7 |
| | | | | | | Critical Value α= 5%: | | | 14,07 |
| | | | | | | Ho: | | Not rejected | |

| Output_JIS Delay | | | Input_JIS Delay | | | $\chi^2$ - Chi-Squared Test | | | |
|---|---|---|---|---|---|---|---|---|---|
| in Hours | Observed Number = O | Frequency in % | in Hours | Frequency | Frequency in % | Expected Number = E | O-E | $(O-E)^2$ | $\chi^2 = (O-E)^2/E$ |
| 0.0000 | 164 | 90% | 0.0000 | 150 | 88% | 160,5882353 | 3,411764706 | 11,64013841 | 0,072484378 |
| 2:00:00.0000 | 0 | 0% | 2:00:00.0000 | 2 | 1% | 2,141176471 | -2,141176471 | 4,584636678 | 2,141176471 |
| 4:00:00.0000 | 2 | 1% | 4:00:00.0000 | 2 | 1% | 2,141176471 | -0,141176471 | 0,019930796 | 0,009308339 |
| 6:00:00.0000 | 2 | 1% | 6:00:00.0000 | 3 | 2% | 3,211764706 | -1,211764706 | 1,468373702 | 0,457185951 |
| 8:00:00.0000 | 7 | 4% | 8:00:00.0000 | 3 | 2% | 3,211764706 | 3,788235294 | 14,35072664 | 4,468174962 |
| 10:00:00.0000 | 3 | 2% | 10:00:00.0000 | 5 | 3% | 5,352941176 | -2,352941176 | 5,53633218 | 1,034259858 |
| 12:00:00.0000 | 4 | 2% | 12:00:00.0000 | 5 | 3% | 5,352941176 | -1,352941176 | 1,830449827 | 0,341952165 |
| ∑ | 182 | 100% | ∑ | 170 | 100% | 182 | | | |
| | | | | | | Chi-Squared Statistic = $\sum (O-E)^2/E$ | | | 8,524542125 |
| | | | | | | Degrees of Freedom | | | 6 |
| | | | | | | Critical Value α= 5%:: | | | 12,59 |
| | | | | | | Ho: | | Not rejected | |

| Deviation Re/Schedulded | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| SR Replication | Mean Value μ | Cumulative Mean Value | Standard Deviation σ | Two-Side 5% Critical Values | | Lower Quartile | Upper Quartile | Confidence α |
| | | | | Degrees of Freedom | Critival Value | | | |
| 1 | 0,77869286 | 0,77869286 | n.a | n.a | n.a | n.a | n.a | n.a |
| 2 | 0,86003881 | 0,81936583 | 0,02876014 | 1 | 12,706 | 0,56097041 | 1,07776126 | 31,54% |
| 3 | 0,85536871 | 0,83136679 | 0,02760694 | 2 | 4,302 | 0,76279776 | 0,89993582 | 8,25% |
| 4 | 0,8760633 | 0,84254092 | 0,02785585 | 3 | 3,182 | 0,79822227 | 0,88685957 | 5,26% |
| 5 | 0,87974232 | 0,8499812 | 0,02804689 | 4 | 2,776 | 0,81516197 | 0,88480043 | 4,10% |
| 6 | | | | 5 | 2,228 | | | |

| JIS Delay | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| SR Replication | Mean value μ | Cumulative mean value | Standard deviation σ | two-side 5% critical values | | Lower quartile | Upper quartile | Confidence α |
| | | | | Degrees of freedom | Critival value | | | |
| 1 | 0,02486911 | 0,02486911 | n.a | n.a | n.a | n.a | n.a | n.a |
| 2 | 0,03606965 | 0,03046938 | 0,00395999 | 1 | 12,706 | -0,00510914 | 0,0660479 | 116,77% |
| 3 | 0,03506601 | 0,03200159 | 0,00375462 | 2 | 4,302 | 0,02267601 | 0,04132716 | 29,14% |
| 4 | 0,0349045 | 0,03272732 | 0,00355855 | 3 | 3,182 | 0,02706566 | 0,03838897 | 17,30% |
| 5 | 0,03525641 | 0,03323314 | 0,00340093 | 4 | 2,776 | 0,02901099 | 0,03745528 | 12,70% |
| 6 | | | | 5 | 2,228 | | | |

**A.5. Data Normalization in the Interval (0,1)**

| Attribute | Original value | Min | Max | Code |
|---|---|---|---|---|
| **Assembly Line (integer)** | 1 | 1 | 3 | 0,000 |
| | 2 | 1 | 3 | 0,500 |
| | 3 | 1 | 3 | 1,000 |
| **JIS Weekdays (integer)** | 1 | 1 | 7 | 0,000 |
| | 2 | 1 | 7 | 0,167 |
| | 3 | 1 | 7 | 0,333 |
| | 4 | 1 | 7 | 0,500 |
| | 5 | 1 | 7 | 0,667 |
| | 6 | 1 | 7 | 0,833 |
| | 7 | 1 | 7 | 1,000 |
| **Deviation Re/Scheduled (integer)** | 0 | 0 | 6 | 0,000 |
| | 1 | 0 | 6 | 0,167 |
| | 2 | 0 | 6 | 0,333 |
| | 3 | 0 | 6 | 0,500 |
| | 4 | 0 | 6 | 0,667 |
| | 5 | 0 | 6 | 0,833 |
| | 6 | 0 | 6 | 1,000 |
| **JIS Delay (real)** | 0,000 | 0 | 0,5 | 0,000 |
| | 0,083 | 0 | 0,5 | 0,167 |
| | 0,167 | 0 | 0,5 | 0,333 |
| | 0,250 | 0 | 0,5 | 0,500 |
| | 0,333 | 0 | 0,5 | 0,667 |
| | 0,417 | 0 | 0,5 | 0,833 |
| | 0,500 | 0 | 0,5 | 1,000 |

## A.6. K-Means Model Summary

| Type | Missing | Statistics | | |
|---|---|---|---|---|
| | | Min | Max | Average |
| Real | 0 | 0 | 1 | 0,519 |
| Real | 0 | 0 | 1 | 0,504 |
| Real | 0 | 0 | 1 | 0,258 |
| Real | 0 | 0 | 1 | 0,091 |

| Optimization Step: 1 | | | | | | |
|---|---|---|---|---|---|---|
| K-Means Cluster Model | | Centroid | | | Distance Measures | Quality Measures |
| | | Attribute | Cluster 0 | Cluster 1 | Cost= ∑ dist(Centroid, x) | G= 1/∑ (Centroid-x)² |
| Cluster 0: | 64 | Assembly Line | 0,133 | 0,788 | 119,639586 | 0,0097401 |
| Cluster 1: | 92 | JIS Weekdays | 0,599 | 0,438 | | |
| Total Number: | 156 | Deviation Re/Scheduled | 0,415 | 0,149 | | |
| | | JIS Delay | 0,089 | 0,092 | | |

| Optimization Step: 2 | | | | | | |
|---|---|---|---|---|---|---|
| K-Means Cluster Model | | Centroid | | | Distance Measures | Quality Measures |
| | | Attribute | Cluster 0 | Cluster 1 | Cost= ∑ dist(Centroid, x) | G= 1/∑ (Centroid-x)² |
| Cluster 0: | 81 | Assembly Line | 0,827 | 0,187 | 85,87636464 | 0,019084769 |
| Cluster 1: | 75 | JIS Weekdays | 0,348 | 0,673 | | |
| Total Number: | 156 | Deviation Re/Scheduled | 0,208 | 0,312 | | |
| | | JIS Delay | 0,103 | 0,078 | | |

| Optimization Step: 3 | | | | | | |
|---|---|---|---|---|---|---|
| K-Means Cluster Model | | Centroid | | | Distance Measures | Quality Measures |
| | | Attribute | Cluster 0 | Cluster 1 | Cost= ∑ dist(Centroid, x) | G= 1/∑ (Centroid-x)² |
| Cluster 0: | 85 | Assembly Line | 0,812 | 0,169 | 85,55484333 | 0,019192141 |
| Cluster 1: | 71 | JIS Weekdays | 0,351 | 0,688 | | |
| Total Number: | 156 | Deviation Re/Scheduled | 0,208 | 0,318 | | |
| | | JIS Delay | 0,098 | 0,082 | | |

| Optimization Step: 4 | | | | | | |
|---|---|---|---|---|---|---|
| K-Means Cluster Model | | Centroid | | | Distance Measures | Quality Measures |
| | | Attribute | Cluster 0 | Cluster 1 | Cost= ∑ dist(Centroid, x) | G= 1/∑ (Centroid-x)² |
| Cluster 0: | 85 | Assembly Line | 0,812 | 0,169 | 85,55484333 | 0,019192141 |
| Cluster 1: | 71 | JIS Weekdays | 0,351 | 0,688 | | |
| Total Number: | 156 | Deviation Re/Scheduled | 0,208 | 0,318 | | |
| | | JIS Delay | 0,098 | 0,082 | | |

## A.7. K-Means Model Sample

| Optimization Step | ID | Cluster | Assembly Line | Centroid | (Centroid-x)² | JIS Weekdays | Centroid | (Centroid-x)² | Deviation Re/Scheduled | Centroid | (Centroid-x)² | JIS Delay | Centroid | (Centroid-x)² |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 50 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,333 | 0,351 | 0,000324 | 0,571 | 0,208 | 0,131769 | 0,667 | 0,098 | 0,323761 |
| 3 | 51 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0,714 | 0,208 | 0,256036 | 0 | 0,098 | 0,009604 |
| 3 | 52 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 53 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 54 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0,167 | 0,098 | 0,004761 |
| 3 | 55 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 59 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,333 | 0,351 | 0,000324 | 0,714 | 0,208 | 0,256036 | 0 | 0,098 | 0,009604 |
| 3 | 60 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,333 | 0,351 | 0,000324 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 61 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,333 | 0,351 | 0,000324 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 62 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,333 | 0,351 | 0,000324 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 63 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,333 | 0,351 | 0,000324 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 64 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,333 | 0,351 | 0,000324 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 65 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0,714 | 0,208 | 0,256036 | 0 | 0,098 | 0,009604 |
| 3 | 70 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0,667 | 0,098 | 0,323761 |
| 3 | 71 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 1 | 0,098 | 0,813604 |
| 3 | 72 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,167 | 0,351 | 0,033856 | 0,143 | 0,208 | 0,004225 | 0 | 0,098 | 0,009604 |
| 3 | 74 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0,143 | 0,208 | 0,004225 | 0 | 0,098 | 0,009604 |
| 3 | 75 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 80 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0,571 | 0,208 | 0,131769 | 0 | 0,098 | 0,009604 |
| 3 | 81 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 82 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 83 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0 | 0,351 | 0,123201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 84 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,167 | 0,351 | 0,033856 | 0,143 | 0,208 | 0,004225 | 0 | 0,098 | 0,009604 |
| 3 | 85 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,167 | 0,351 | 0,033856 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 86 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,167 | 0,351 | 0,033856 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 87 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,167 | 0,351 | 0,033856 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 97 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,5 | 0,351 | 0,022201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 98 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,5 | 0,351 | 0,022201 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 100 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,167 | 0,351 | 0,033856 | 0,714 | 0,208 | 0,256036 | 0 | 0,098 | 0,009604 |
| 3 | 101 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,167 | 0,351 | 0,033856 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 102 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,167 | 0,351 | 0,033856 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |
| 3 | 103 | cluster_0 | 0,5 | 0,812 | 0,097344 | 0,333 | 0,351 | 0,000324 | 0,143 | 0,208 | 0,004225 | 0 | 0,098 | 0,009604 |
| 3 | 104 | cluster_0 | 1 | 0,812 | 0,035344 | 0,833 | 0,351 | 0,232324 | 0 | 0,208 | 0,043264 | 0 | 0,098 | 0,009604 |

## A.8. K-Medoids Model Summary

| Type | Missing | Statistic | | |
|---|---|---|---|---|
| | | Min | Max | Average |
| Real | 0 | 0 | 1 | 0,519 |
| Real | 0 | 0 | 1 | 0,504 |
| Real | 0 | 0 | 1 | 0,258 |
| Real | 0 | 0 | 1 | 0,091 |

| Optimization Step: 1 | | | | | | |
|---|---|---|---|---|---|---|
| K-Medoids Cluster Model | | Centroid | | | Cost= ∑ dist(Centroid, x) | G= 1/∑ (Centroid-x)² |
| | | Attribute | Cluster 0 | Cluster 1 | | |
| Cluster 0: | 68 | Assembly Line | 1 | 0,5 | 132,062837173 | 0,007594377 |
| Cluster 1: | 88 | JIS Weekdays | 0,667 | 0,333 | | |
| Total Number: | 156 | Deviation Re/Scheduled | 0 | 0,149 | | |
| | | JIS Delay | 1 | 0 | | |

| Optimization Step: 2 | | | | | | |
|---|---|---|---|---|---|---|
| K-Medoids Cluster Model | | Centroid | | | Cost= ∑ dist(Centroid, x) | G= 1/∑ (Centroid-x)² |
| | | Attribute | Cluster 0 | Cluster 1 | | |
| Cluster 0: | 17 | Assembly Line | 1 | 1 | 121,841754010 | 0,008658509 |
| Cluster 1: | 139 | JIS Weekdays | 0,667 | 0,667 | | |
| Total Number: | 156 | Deviation Re/Scheduled | 0 | 0 | | |
| | | JIS Delay | 1 | 0 | | |

| Optimization Step: 3 | | | | | | |
|---|---|---|---|---|---|---|
| K-Medoids Cluster Model | | Centroid | | | Cost= ∑ dist(Centroid, x) | G= 1/∑ (Centroid-x)² |
| | | Attribute | Cluster 0 | Cluster 1 | | |
| Cluster 0: | 142 | Assembly Line | 1 | 1 | 121,841754010 | 0,008658509 |
| Cluster 1: | 14 | JIS Weekdays | 0,667 | 0,667 | | |
| Total Number: | 156 | Deviation Re/Scheduled | 0 | 0 | | |
| | | JIS Delay | 0 | 1 | | |

| Optimization Step: 4 | | | | | | |
|---|---|---|---|---|---|---|
| K-Medoids Cluster Model | | Centroid | | | Cost= ∑ dist(Centroid, x) | G= 1/∑ (Centroid-x)² |
| | | Attribute | Cluster 0 | Cluster 1 | | |
| Cluster 0: | 142 | Assembly Line | 1 | 1 | 121,841754010 | 0,008658509 |
| Cluster 1: | 14 | JIS Weekdays | 0,667 | 0,667 | | |
| Total Number: | 156 | Deviation Re/Scheduled | 0 | 0 | | |
| | | JIS Delay | 0 | 1 | | |

## A.9. EM Clustering Model Summary

| Optimization Step: 1 | | | | | | | |
|---|---|---|---|---|---|---|---|
| EM Cluster Model | | Centroid | | | Probability Cluster 0 | Probability Cluster 1 | E =∑ Log (P(X) ) |
| | | Attribute | Cluster 0 | Cluster 1 | | | |
| Cluster 0: | 70 | Assembly Line | n.a | n.a | 69,68102476 | 86,31897524 | 295,7636229 |
| Cluster 1: | 86 | JIS Weekdays | n.a | n.a | | | |
| Total number: | 156 | Deviation Re/Scheduled | n.a | n.a | | | |
| | | JIS Delay | n.a | n.a | | | |
| **Optimization Step: 2** | | | | | | | |
| EM Cluster Model | | Centroid | | | Probability Cluster 0 | Probability Cluster 1 | E =∑ Log (P(X) ) |
| | | Attribute | Cluster 0 | Cluster 1 | | | |
| Cluster 0: | 68 | Assembly Line | n.a | n.a | 71,49754288 | 84,50245712 | 295,6143512 |
| Cluster 1: | 88 | JIS Weekdays | n.a | n.a | | | |
| Total number: | 156 | Deviation Re/Scheduled | n.a | n.a | | | |
| | | JIS Delay | n.a | n.a | | | |
| **Optimization Step: 3** | | | | | | | |
| EM Cluster Model | | Centroid | | | Probability Cluster 0 | Probability Cluster 1 | E =∑ Log (P(X) ) |
| | | Attribute | Cluster 0 | Cluster 1 | | | |
| Cluster 0: | 76 | Assembly Line | n.a | n.a | 79,97668269 | 76,02331731 | 295,1105926 |
| Cluster 1: | 80 | JIS Weekdays | n.a | n.a | | | |
| Total number: | 156 | Deviation Re/Scheduled | n.a | n.a | | | |
| | | JIS Delay | n.a | n.a | | | |
| **Optimization Step: 4** | | | | | | | |
| EM Cluster Model | | Centroid | | | Probability Cluster 0 | Probability Cluster 1 | E =∑ Log (P(X) ) |
| | | Attribute | Cluster 0 | Cluster 1 | | | |
| Cluster 0: | 21 | Assembly Line | n.a | n.a | 124,1190876 | 31,88091237 | 311,0592902 |
| Cluster 1: | 135 | JIS Weekdays | n.a | n.a | | | |
| Total number: | 156 | Deviation Re/Scheduled | n.a | n.a | | | |
| | | JIS Delay | n.a | n.a | | | |
| **Optimization Step: 5** | | | | | | | |
| EM Cluster Model | | Centroid | | | Probability Cluster 0 | Probability Cluster 1 | E =∑ Log (P(X) ) |
| | | Attribute | Cluster 0 | Cluster 1 | | | |
| Cluster 0: | 21 | Assembly Line | n.a | n.a | 134,9984658 | 21,00153425 | 315,3611013 |
| Cluster 1: | 135 | JIS Weekdays | n.a | n.a | | | |
| Total number: | 156 | Deviation Re/Scheduled | n.a | n.a | | | |
| | | JIS Delay | n.a | n.a | | | |
| **Optimization Step: 6** | | | | | | | |
| EM Cluster Model | | Centroid | | | Probability Cluster 0 | Probability Cluster 1 | E =∑ Log (P(X) ) |
| | | Attribute | Cluster 0 | Cluster 1 | | | |
| Cluster 0: | 0 | Assembly Line | n.a | n.a | n.a | n.a | n.a |
| Cluster 1: | 156 | JIS Weekdays | n.a | n.a | | | |
| Total number: | 156 | Deviation Re/Scheduled | n.a | n.a | | | |
| | | JIS Delay | n.a | n.a | | | |

| K-Means Cluster Model | | Differece in Average comparing with K-Medoids | | Standard Error | | Difference in Average ± t x Standard Error | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Cluster 0 | Cluster 1 | Cluster 0 | Cluster 1 | Cluster 0 - | Cluster 0 + | Conclusion | Cluster 1 - | Cluster 1 + | Conclusion |
| Cluster 0: | 85 | 0,30120062 | -0,438128772 | 0,050604017 | 0,090197236 | 0,30120062 | 0,30120062 | significantly different | -0,438128772 | -0,438128772 | significantly different |
| Cluster 1: | 71 | -0,16073318 | 0,259103622 | 0,045407717 | 0,096521398 | -0,16073318 | -0,16073318 | significantly different | 0,259103622 | 0,259103622 | significantly different |
| Total Number : | 156 | -0,04711656 | 0,032144869 | 0,040446402 | 0,109884189 | -0,04711656 | -0,04711656 | significantly different | 0,032144869 | 0,032144869 | significantly different |
| | | 0,07929432 | -0,751130784 | 0,016060501 | 0,458770286 | 0,07929432 | 0,07929432 | significantly different | -0,751130784 | -0,751130784 | significantly different |

| K-Medoids Cluster Model | | Differece in Average comparing with EM | | Standard Error | | Difference in Average ± t x Standard Error | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Cluster 0 | Cluster 1 | Cluster 0 | Cluster 1 | Cluster 0 - | Cluster 0 + | Conclusion | Cluster 1 - | Cluster 1 + | Conclusion |
| Cluster 0: | 142 | -0,01536262 | 0,130952381 | 0,04435622 | 0,106423446 | -0,01536262 | -0,01536262 | significantly different | 0,130952381 | 0,130952381 | significantly different |
| Cluster 1: | 14 | 0,004289773 | -0,055484123 | 0,039801478 | 0,113885306 | 0,004289773 | 0,004289773 | significantly different | -0,055484123 | -0,055484123 | significantly different |
| Total Number : | 156 | -0,006940902 | 0,054421769 | 0,042939405 | 0,129652021 | -0,0069409 | -0,0069409 | significantly different | 0,054421769 | 0,054421769 | significantly different |
| | | 0,01761268 | 0,158682539 | 0,017050426 | 0,121966573 | 0,01761268 | 0,01761268 | significantly different | 0,158682539 | 0,158682539 | significantly different |

| EM Cluster Model | | Differece in Average comparing with K-Means | | Standard Error | | Difference in Average ± t x Standard Error | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Cluster 0 | Cluster 1 | Cluster 0 | Cluster 1 | Cluster 0 - | Cluster 0 + | Conclusion | Cluster 1 - | Cluster 1 + | Conclusion |
| Cluster 0: | 135 | 0,285838 | -0,307176391 | 0,051092917 | 0,042708267 | 0,285838 | 0,285838 | significantly different | -0,307176391 | -0,307176391 | significantly different |
| Cluster 1: | 21 | -0,156443407 | 0,203619499 | 0,045846413 | 0,045702749 | -0,15644341 | -0,15644341 | significantly different | 0,203619499 | 0,203619499 | significantly different |
| Total Number : | 156 | -0,054057462 | 0,086566638 | 0,049460919 | 0,052030012 | -0,05405746 | -0,05405746 | significantly different | 0,086566638 | 0,086566638 | significantly different |
| | | 0,096907 | -0,592448245 | 0,019639996 | 0,048945803 | 0,096907 | 0,096907 | significantly different | -0,592448245 | -0,592448245 | significantly different |

# Eidesstattliche Versicherung

| Su, Wenzheng | 164293 |
|---|---|
| Name, Vorname | Matr.-Nr. |

Ich versichere hiermit an Eides statt, dass ich die vorliegende ~~Bachelorarbeit~~/Masterarbeit* mit dem Titel:

**Knowledge Discovery in Supply Chain Transaction Data by Applying Data Farming**

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

| Dortmund, 17.Mai.2016 | |
|---|---|
| Ort, Datum | Unterschrift |

*Nichtzutreffendes bitte streichen

**Belehrung:**

Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50.000,00 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden. (§ 63 Abs. 5 Hochschulgesetz - HG - )

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird gfls. elektronische Vergleichswerkzeuge (wie z.B. die Software „turnitin“) zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.

Die obenstehende Belehrung habe ich zur Kenntnis genommen:

| Dortmund, 17.Mai.2016 | |
|---|---|
| Ort, Datum | Unterschrift |