Hochschule für Technik
und Wirtschaft Berlin

University of Applied Sciences

Project Report

Student Project Business Intelligence – Process Causality Analytics

Field of study: Business Intelligence

Processing period: May 18 2021 - September 17 2021

Fabio Cosimo Andriulo	558214
Timur Burkholz	575306
Maximilian Alexander Reech	575311

Professor:	Prof. Dr. Ingo Claßen
Supervisor:	Benjamin Aunkofer
Partners (Signavio):	Timotheus Kampik, Konstantinos Poulios

Abstract

Process analyses are almost standard for companies today. The question often arises as to how a change actually affects the process. Precisely such questions can usually be answered using business intelligence methods. However, the answers are often complex and not always clear. Even though one often hears about causal relationships, it is not always clear whether it is just a correlation. Machine learning, however, could help where less mathematically savvy individuals fail. Therefore, this project addresses the application of double machine learning, a method for causal analysis, and the consideration of process changes as causes.

Contents

List of Figures

1 Introduction

This document was created as a report about the project. The project is part of the Master's program in Business Computing at HTW Berlin. The project was presented by DATANOMIQ GmbH and accompanied by Signavio GmbH as a cooperation partner.

1.1 Background

As part of business intelligence, process mining is a way for modern companies to obtain information about their processes. It is not uncommon for processes to continue to be adapted and changed over time. The question that always arises is to what extent the changes have influenced the results of the process.

1.2 Goal

In order to specify this question further, it should be found out whether there is a causal relationship between the change in the process and the change in the measurement result. In the best case, the specific changes should be found from all changes, which also influence the measurement result.

1.3 Structure

First, the methods used are presented. This serves to subsequently clarify the reasons for which type of data was used. Afterwards, the implementation of the software is discussed in order to explain its special features, limitations and possibilities.

2 Double Machine Learning

Data-driven causal analysis attempts to measure the effect of a change on an observable outcome. Measuring this effect is possible if the changes are observable. This can be realized through double machine learning. In general, it attempts to estimate the effect of the change on the outcome based on the changes.[1] In order to identify the causal effect of a change, all other influences must be comparable.[2] It is also necessary that the results can arise from the same characteristics, and there should be no combination that can be found only before the change or only after the change.[3]

Machine learning can be used to represent the relationships between changes and their effects in the form of machine models. It should be noted that the representation as a model is always only an approximately correct estimate of the actual relationship.[4] Now we can assume that there are two models. One to calculate the result under condition of the change and one to explain the change. Assuming that a change is calculable, if the calculation of the first model is different under the different conditions (changed/not changed), we can assume that the representation of the change by the second model explains the difference.[5]

[1] see Huber 2020, p. 106.

[2] see ibid., p. 107.

[3] see ibid., p. 109.

[4] see ibid., p. 111.

[5] see ibid., pp. 112 sqq.

3 Data acquisition

3.1 Real Data

To implement such an algorithm, you need data to work with. In the best case, there is real data from a company that has just adapted its processes. The difficulty is that there must be at least two data sets from the same process. One must represent the state before the changes and one the state after the changes.

Since no data could be provided during the course of the project, some data had to be sought. One place to start with is the Business Process Intelligence Challenge, where participants face process mining problems every year. However, even after finding two data sets that had the same origin and the same process, problems arose. First, the datasets were in different languages and second, some of them were poorly described. This is due to the time gap of five years (2012[6], 2017[7]). Therefore, it was decided to simulate the data.

3.2 Simulation

During the project, a simple order-to-cash process was designed in BPMN-format. The order-to-cash process encompasses all steps from when a customer order is placed up until the business is paid (the cash). Those steps include order management and order fulfillment, through to credit management, then invoicing and ultimately payment collection. Since the project does not focus on developing a complex process, a simplified process was used.

A variety of sources were consulted for this purpose and finally the process from the following source was used: Dumas et al. 2018, p. 373.

[6] BPI 2012 https://www.win.tue.nl/bpi/doku.php?id=2012:challenge

[7] BPI 2017 https://www.win.tue.nl/bpi/doku.php?id=2017:challenge

Other processes from Fahland et al. 2020, p. 251 and Dubois 2017, p. 353 were under consideration, but were discarded as the project progressed.

The BPMN process shows the process from ordering a product to delivery. The example company works with two suppliers. The three roles of the ERP system, a warehouse employee and a sales employee are represented by swin lanes. When a customer orders a product, it is checked whether it is already in stock. If it is, the product is ordered from the warehouse and confirmed after the order is received. If the product is not in stock, the raw materials are ordered from the two suppliers and then the product is manufactured. If the product is then available, the product is shipped to the customer's delivery address. At the same time, an invoice is sent to the customer. When the amount is received and the product has been delivered, the order is archived, and the process is finished.

The process is considered the basis for the project. For this purpose, a second similar process will be developed, in which the first process will be modified.

The BPMN process can be found on the next page.

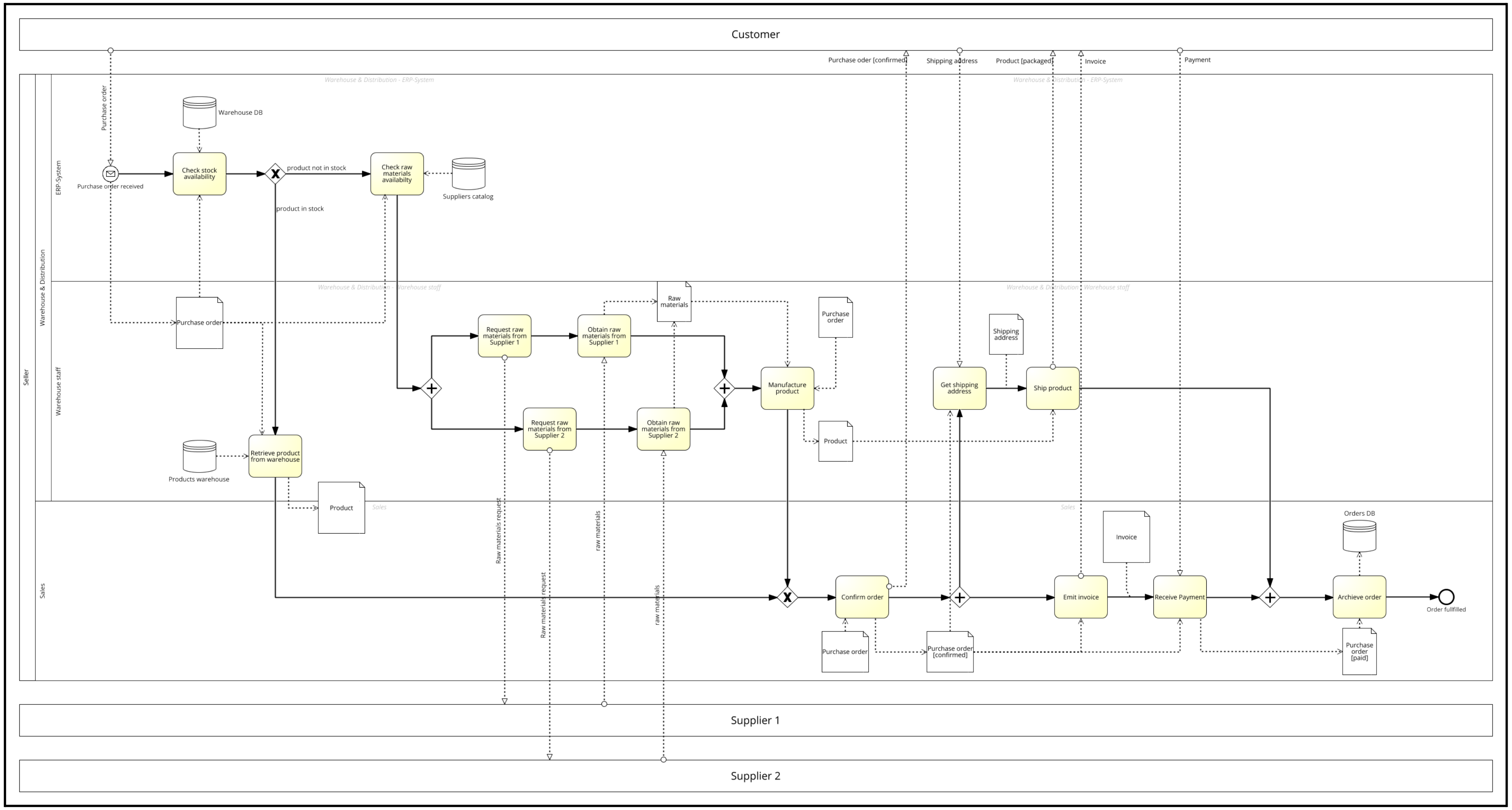

Figure 3.1: BPMN: Process 1 (Unchanged)

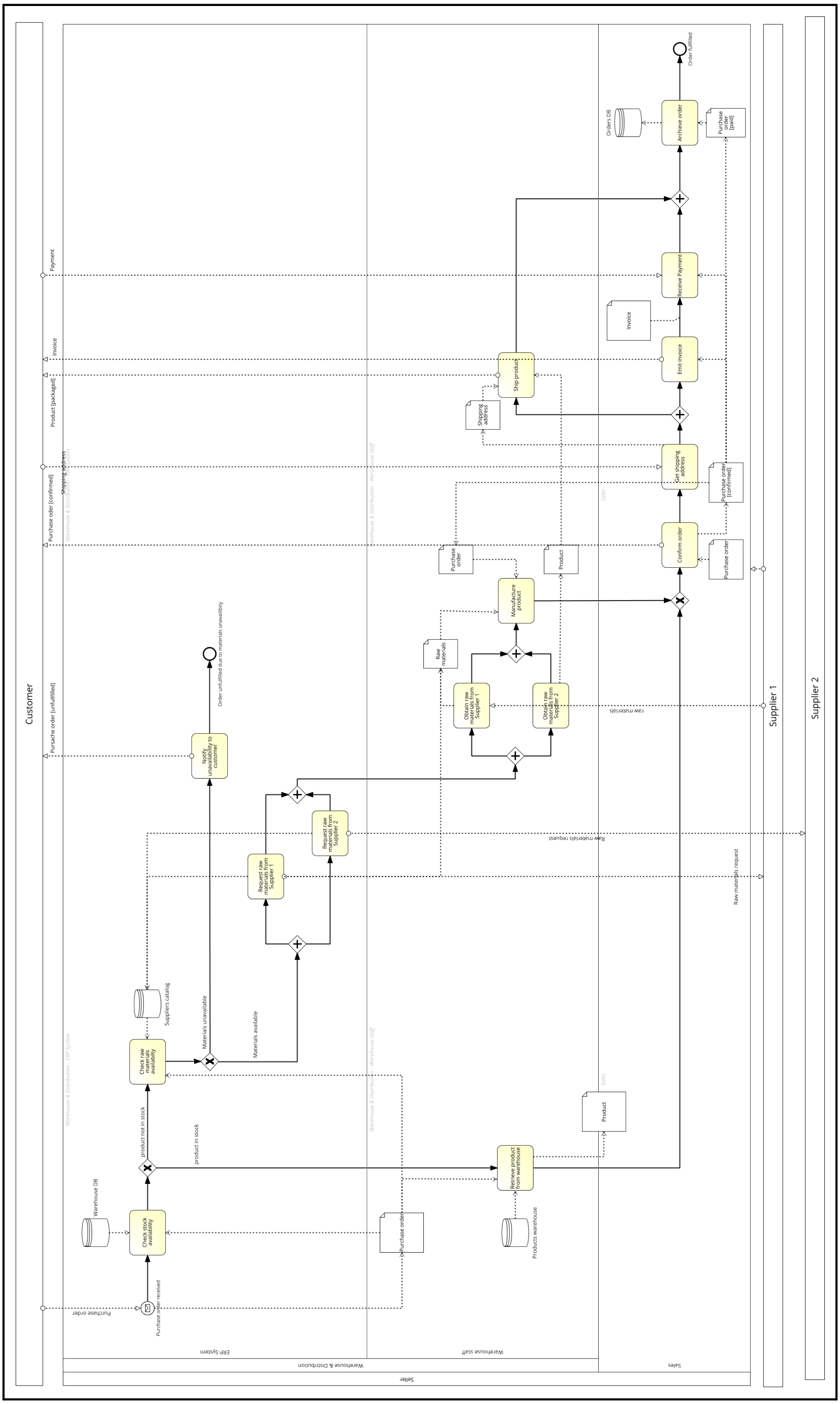

Figure 3.2: BPMN: Process 2 (Changed)

In the previous page (6) you can see the customized process. The markers show the changes. The following adjustments were applied:

1. A new gateway was integrated. Here it is checked whether the raw materials are available at the suppliers. If this is not the case, a new end is reached and the customer is informed that the product is not available.

2. Orders for raw materials have now been automated by the ERP system. A warehouse employee no longer must do this manually. For this purpose, the suppliers' databases were connected and automatic processes were started.

3. The customer's delivery address is now entered directly by the sales staff, who also confirm the order. In addition, this changed the sequence and parallelism of the process steps.

4 Implementation

4.1 Package description

For better understanding, the Python files of the source folder and their functions and methods are discussed below.

4.1.1 misc.py

This module contains methods that could be used in many different places. However, these could not be assigned to the specific modules.

read_bpmn

This method can be used to read BPMN files in XML format. The path (path) to the file must be specified. Additionally a generic path (path) and the name (name) can be specified. The implementation of the pm4py package is used to convert the read file into an image of a BPMN graph.

get_scenario

This method can be used to import scenario files in CSV format. The scenario created in the process can be used to generate measurement values for a KPI in an event log. In this version, only the KPIs time and costs based on time are taken into account. The path to the file must be specified. Additionally, a generic path and the name can be specified. Furthermore, the deviation scale for non-automated activities can be specified. This can be used to simulate the work done manually by people. The basis for this is the normal distribution (random.normal) implemented by the numpy package.

The imported scenario file must have the following formatting.

- Activity: A column representing the names of the activities as a string.
- Execution time: A column representing the average execution times of the activities as time difference hours:minutes:seconds.
- Execution costs: A column representing the initial execution costs of the activities.
- Automated: A column representing the automation of the activities as a boolean value (true/false).
- Resources cost/hour: A column representing the hourly cost of the bound resource of activities.

The method returns a nested Python Dictonary representing a Scenario. On the first level there are two attributes apply_to and functions. The attribute apply_to represents a string that points to another KPI (column in an event log) that is necessary for the calculation. The functions attribute contains another Python dictionary where the names of the activities are assigned to individual so-called lambda functions that reflect the calculation of the KPI. By this structure, a fast application to an evnet log implemented in the pandas DataFrame format is possible.

get_dummy_scenario

This method is very similar to method get_scenario. However, it does not require a file that represents a scenario. However, the names of the activities (activities) are required. Furthermore, the deviation scale for non-automated activities can be specified. Here, too, only the KPIs time and costs are taken into account. The output is therefore identical in form.

get_dummy_ruleset

This method can be used to create a rule set for calculating KPIs. Thereby the method needs the eventlog as pandas DataFrame. Additionally, the name of the column for the case ID and the activity ID can be changed. These are set to pm4py values by default. Furthermore, the number of activities executed in parallel (n_parallels) can be set. By default, the rule set is given the name “cost”, which reflects the KPI cost. However, this can be changed.
The return value is a Python dictionary that matches the name with a Python list of activity names. Thereby sequential activities are coded as Python List and parallel ones as Python Tuple.

min_max_scale

This method can be used to scale numeric columns of a pandas DataFrame (frame) between two values (scale_min, scale_max) by their minima and maxima. By default, the limits zero and one are used.

4.2 Simulation

Since it was necessary to simulate the data, different possibilities to generate an event log from a BPMN were searched. However, it was quickly determined that the options available to us (open source, Signavio) did not provide the desired result. Therefore, it was necessary to include the simulation into our implementation.
Fortunately, there is a package that can be used to read BPMNs in XML format. This kind of BPMN formatting could fortunately be generated by the modeling tools that Signavio provided. With the Python package PM4Py the BPMNs could be read and an event log could be generated.
However, the generated event logs do not contain any measured values. These must therefore be generated by us. This is made possible by an implemented function. Here, additional in-

formation is provided by a CSV file. This represents the activities (Activitiy) with their initial execution costs (Execution costs), their average execution times (Execution time), their hourly costs (Resources cost/hour) and their automation levels (Automated) in binary coding.
As you can already imagine, a distinction is made according to automation. This information is used as a flag to indicate that the execution time should vary. The frame can be defined by the user. The measured values are then generated as follows:

1. Setting the execution time
2. Multiplying the execution time with the hourly cost
3. Adding the execution costs

Efficient execution is ensured by vector-based calculation with individualized functions.

4.3 Data Transformation

Since an event log presents the information unsuitable for double machine learning, it is necessary to transfer the data into a suitable form, so-called case tables. These contrast the individual process executions (rows) with their activities. For each activity (columns), the measured values (execution time, costs) and the number of executions are recorded. For the measured values, it is for example possible to choose whether the sum or the average should be used for multiple executions. However, since this is the aggregation function of Pandas in the background, other summaries are also possible. Missing values can also be filled with a specific value. Since from our point of view, for example, no time was needed for the execution of this step in case of non-execution, we recommend to use the value zero for filling.
Another idea, which unfortunately reached the project late (and was not implemented), the position of the executed activity is to take into account. With reliable information, changes could be better identified.
Now the actual measurement results of the process executions have to be calculated. Unfortunately, for this it was necessary to disassemble the processes again by hand into a suitable

format, since PM4Py (poorly documented) does not seem to allow the output of the activities including the parallelizations. Thereby linear activities are described as Python list and parallel activities as Python tuples.

With the help of this information, the measurement result can now be calculated by adding the measurements over the process. It can now be set which activities should be used for this and whether these should to be calculated differently in the case of parallel processing. For example, the time of the maximum can be taken instead of adding the parallel activities.

The last step is to filter out unnecessary information. These can arise quickly under certain conditions. There are automated activities that are always executed and therefore do not fluctuate in their values or frequency. They do not contain any information.

In addition, duplication occurs quickly because of automated activities that are executed the same number of times in the event of an execution are recorded for each measured value and in their number. This means that all information about the activity can be the same. Accordingly, only one needs to be used. In this case, the different scale may complicate the detection. This, in turn, is easy to filter when the scales are unified.

Filtering out the “duplicates” and deleting the “unnecessary information” now creates a dataset that is clearer for machine learning. However, it must be noted that two are considered at the same time. This means that information about the activities from the dataset of the unchanged process should also be found in that of the changed process. Accordingly, non-existing information about omitted activities must be replaced by values. As already mentioned, it is recommended to replace the values by zero, because the activities were not been performed and therefore their measured values should also be zero.

4.4 Double Machine Learning

Thanks to the processing of the data, it is possible to use double machine learning. To do this, however, it is first necessary to find out which features combination(the information of the activities) best represents the original process. This is achieved by a search algorithm that recursively tests feature combinations. The first step is to determined which goal is to be pursued, i.e. which measurement result (in this case, cost or time) must be considered at any given time. Afterwards, the algorithm first tests out each individual using cross-validation (usually stratified 5-fold convolution). The best result is then combined again with every other feature until there is no improvement.[8]

The best[9] combination of features can now be used to represent the unmodified process. Cross-validation is again used for training. This reduces the dependence on the data and the risk of overfitting.

These models are now used to estimate the measurement results of the modified process as the average of all predictions. The information about the activities of the modified process is used as features. The result now obtained can be compared with the actual measurement. The difference thus reflects the deviation caused by the change of the process.

The last step is very similar to the first one. Now we can use the same method to estimate the difference.[10] Again we try combinations of features as described before. Only this time we get more information when we look not only at the best combination. In fact, we get not only the best result, but all the tried ones.

The most difficult part of this is the analysis of the combinations of features. One must be sure which information of the activities reflect the changes of the process. Moreover, the most obvious features are not always selected. It may be that other features can carry the information just as well, even if the presentation does not suggest it. For example, features that are also used that are next to the actual change on the same path in the process flow. In addition, dependent or nearly identical activities are also interchangeable from the perspective of the algorithm. So the big challenge is to interpret the results correctly.

In this way, however, one can quickly see which information improves the estimate, and thus the explanation of the difference, and which does not.

[8] An attempt is made to use the information from the activities to estimate the measured values of time and cost.

[9] The evaluation is usually described by a negative average squared error. This means that larger values (closer to zero) are better.

[10] An attempt is made to estimate the difference between measurement and prediction based on information about the changed process.

5 Conclusion

The project would certainly have benefited if a practice partner providing had provided its real data. Nevertheless, it is also very interesting to simulate data.

The biggest challenge was probably the preparation of the theoretical input. Precisely because there is little on the one hand, and because it is very mathematical and complex on the other.

In addition, retrospectively, the preparation of the data is absolutely important to make the double machine learning work well.

In the future, it might be interesting to test the algorithms with different machine learning models. This should also be possible without further ado, as long as one sticks to the scikit-learn standard.

Also the consideration of different measurement methods for the processes, as well as evaluation possibilities of the model could be interesting. This could have a strong impact on finding the right feature combinations. One possibility would be to measure the models at interval limits and to determine the accuracy.

Another point worth mentioning is the complexity of the processes. In reality, processes are usually not so clear and simple. Therefore, it can't hurt to test significantly more complicated processes as well.

Finally, it must be said that the results are not always as we expected them to be. Often the results of the models differed in the decimal range. But there were also areas where floating point calculations can become very inaccurate. In conclusion, we did not succeed in finding an adequate reason, even if the deviation occurred in the range $< 10^{-7}$.

6 Appendix

6.1 Double Machine Learning: In this case

In this case, however, we are dealing with process changes. If there is a suitable representation of the process properties in the flow and its results, then double machine learning can be used for this as well. In the following, the idea is explained by means of an example.

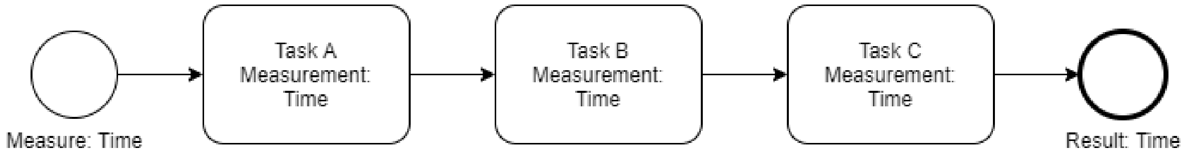

Figure 6.1: Process 1 (Unchanged)

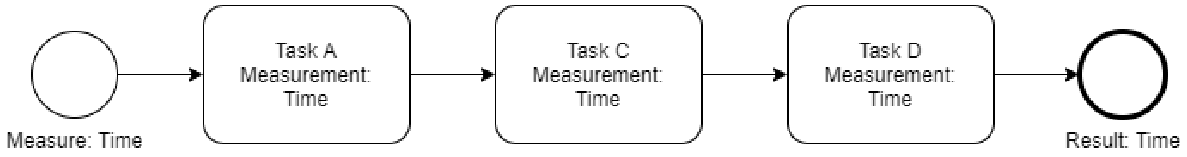

Figure 6.2: Process 2 (Changed)

As can be seen in the figures, Task B has been deleted and Task D has been introduced in the modified process. As a simple example, we now use time as our measurand. The result of the measurement here is the sum of the time spent on the tasks. We can therefore set up the following example functions:[11]

$$p_1 = A + B + C$$

$$p_2 = A + C + D$$

[11] p_1/p_2 representing the results, $A..D$ representing the measurements

We can now assume that a machine learning model (u) can be trained for the unmodified process as the adornment of the estimate p1 and as the characteristics A..C. Then we can try to use the model (u) to make a prediction for our modified process to measure the difference (β) in the outcome.

$$u \approx p_1$$

However, this requires that the model makes a prediction (q) for the outcome of the changed process.

$$u \rightarrow p_2 \approx q$$

Now the difference (β) can be calculated.

$$\beta = p_2 - q$$

A second model (m) shows the effect of the change on the difference.

$$m \approx \beta$$

In this example case, it is clear because the change in results can only be explained by task D.

$$m \equiv D$$

In more complex examples, it is conceivable that the change in the difference may also consist of combination of changed characteristics. In the end, it is only a matter of finding out whether the estimation of the difference was successful. A classical regressive measurement would be the mean squared error. The smaller this is (closer to zero), the more successful the estimation and thus the presentation as causal.

$$m \rightarrow \beta \equiv \beta_m$$

$$causality = mean((\beta - \beta_m)^2)$$

Bibliography

Dubois Eric; Pohl, Klaus (2017). *Advanced Information Systems Engineering*. Springer International Publishing.

Dumas, Marlon et al. (2018). *Fundamentals of Business Process Management*. Springer Berlin Heidelberg.

Fahland, Dirk et al. (2020). *Business Process Management Forum*. Springer International Publishing.

Huber, Martin (2020). *Kausalanalyse mit maschinellem Lernen*. Springer Fachmedien Wiesbaden GmbH.