Focusing Business Improvements Using Process Mining Based Influence Analysis

Teemu Lehto[1,2(✉)], Markku Hinkka[1,2], and Jaakko Hollmén[2]

[1] QPR Software Plc, Helsinki, Finland
teemu.s.lehto@gmail.com
[2] Department of Computer Science, School of Science,
Aalto University, Espoo, Finland

Abstract. Business processes are traditionally regarded as generalized abstractions describing the activities and common behaviour of a large group of process instances. However, the recent developments in process mining and data analysis show that individual process instances may behave very different from each other. In this paper we present a generic methodology called influence analysis for finding business improvement areas related to business processes. Influence analysis is based on process mining, root cause analysis and classification rule mining. We present three generic target levels for business improvements and define corresponding probability-based interestingness measures. We then define measures for reporting the contribution results to business people and show how these measures can be used to focus improvements. Real-life case study is also included to show the methodology in action.

Keywords: Process analysis · Process improvement · Process mining · Classification rule mining · Root cause analysis · Data mining · Influence analysis · Contribution

1 Introduction

Many organizations have major problems in their business operations. These problems include too long lead times, delayed customer deliveries, bad product quality, operational inefficiencies causing high operational costs, failure to comply with regulations and bottlenecks in sales processes limiting growth. Problem is that with current methodologies it is difficult, expensive and time-consuming to identify the causes for these business problems. One reason for difficulty is that causality itself is a difficult concept in dynamic business systems [10]. In addition the theory of constraints highlights the importance of finding the most relevant constraints which limit any system in achieving more of its goals [5].

Inability to identify root causes for business problems means that business improvements are not targeted to right issues. This further leads to (1) increased costs when the inefficient operations are not improved and resources are spent on improving things providing only small benefits, (2) decreased sales when the

M. La Rosa et al. (Eds.): BPM Forum 2016, LNBIP 260, pp. 177–192, 2016.
DOI: 10.1007/978-3-319-45468-9_11

Table 1. Benefit vs. effort matrix

		Potential benefit	
		Small	Large
Effort: Resources and time needed to implement change	Small	Good if small improvements are enough	BEST CASE: small investment and large benefits
	Large	WORST CASE: large investment and small benefits	Good if large improvements are needed

constraints for making more sales are not removed and (3) continuing regulatory problems when issues keep on repeating. If we identify wrong reasons, then our development efforts are inefficient.

Business improvements can be achieved by developing a better process design and deploying that design to all businesses (business process re-engineering). Alternatively improvements can be achieved by discovering the current problematic areas where the actual operations deviates from the intended design (fixing operative issues like giving training for individual employees). All identified improvement ideas should be prioritized based on the benefit potential and implementation effort needed as shown in Table 1.

Traditionally the improvement areas are identified based on the discussions with people participating in the execution of these processes. In this paper we present an influence analysis methodology that provides a data-driven approach to finding these areas. Influence analysis contains two main ideas: a technique for identifying as many as possible dimensions for categorizing the process instances and a technique for ranking the areas based on business process improvement potential and effort. In practice we can easily identify about 1.000 dimensions each having an average 100 distinct categories for a dataset of 1 million cases. Then the task is to rank the 100.000 individual categories so that the worst and best performing categories are identified and shown to business people so that they can make decisions for focusing the development efforts.

For ranking the individual categories we adapt an idea that it is easier to conduct business improvements when they are limited to certain subset of cases rather than the whole set of cases. This means that the effort is proportional to the amount of cases while the benefit is proportional to the amount of problematic cases. This means that we should focus the improvements to those subsets that have the highest density of problematic cases. On the other hand it is easy to find segments that only have one case and that is a problematic case, so that the density of problematic cases is 100 %. So we need to take into account also the absolute size of the potential benefit which means that we want to find those segments that have the highest density and largest absolute size.

Influence analysis methodology presented in this paper includes the following steps: 1. Identify the relevant business process and define the case, 2. collect event and case attribute information, 3. create new categorization dimensions, 4. form a binary classification of cases such that each case is either problematic or successful, 5. select a corresponding interestingness measure based on the desired level of business process improvement effect, 6. find the best categorization rules, and 7. present the results to business people.

The rest of this paper is organized as follows: Sect. 2 introduces relevant background in process mining and data analysis. Section 3 presents our influence analysis methodology for focusing business improvements. We have also included some actual project experiences and advice to the corresponding steps. Section 4 presents experiments of using our analysis with sample data and Sect. 5 shows a real-life example. Summary and conclusions are presented in Sect. 6.

2 Related Work

The idea of root cause analysis is well known and studied. It includes steps like problem understanding, problem cause brainstorming, data collection, analysis, cause identification, cause elimination and solution implementation [1]. Process mining based contribution analysis methodology presented in this paper supports all these steps and makes root cause analysis itself much more efficient.

Over the past 20 years organizations have been building data warehouses and business intelligence systems to store operational data created during business operations [7]. In 2012 the amount of available data had grown so much that the term Big Data was introduced to highlight new possibilities of data analysis [9]. There are many data mining and statistical analysis techniques that can be used to turn this data into knowledge [11,13]. There has also been more work in detection of differences between groups [19] and finding contrast sets [2].

Recent studies in the field of process mining have highlighted the usage of process mining for business process analysis [16]. Decision tree learning has been used to explain why certain activity path is chosen within the process [14] discovering decisions made during the process flow. Causal nets have been further studied as a tool and notation for process discovery [17]. Our work is partly based on enriching and transforming process-based logs for the purpose of root cause analysis [15]. We also adapt ideas from the framework for correlating business process characteristics [3]. So far these process mining techniques have been focusing on discovering processes, making findings and creating predictions based on the models. In this paper we extend the current process mining framework with easy-to-use presentation metrics which allow the business users to identify root causes for business problems interactively. Our method can also be regarded as an example of abductive reasoning that starts from an observation and tries to find a hypothesis that accounts for the observation [8].

Probability-based interestingness measures are functions of a 2×2 contingency table. Table 2 shows the generic representation of a contingency table for a rule $A \rightarrow B$, where $n(AB)$ denotes the amount of cases satisfying both A and

Table 2. 2×2 Contingency table for rule $A \rightarrow B$

	B	$\bar{B}$	
A	$n(AB)$	$n(A\bar{B})$	$n(A)$
$\bar{A}$	$n(\bar{A}B)$	$n(\bar{A}\bar{B})$	$n(\bar{A})$
	$n(B)$	$n(\bar{B})$	N

Table 3. Contingency table for rule $product = hats \rightarrow durationdays \geq 20$

	B	$\bar{B}$	
A	1	3	4
$\bar{A}$	2	4	6
	3	7	10

B, and N denotes total amount of cases. An example contingency table for a rule $product = hats \rightarrow durationdays \geq 20$ in a database that contains a total of 10 cases such that 3 cases take long time, 4 cases belong to category *hats*, and one case meets both conditions i.e. the product delivered is *hats* and it took a long time is shown in Table 3.

Probability-based objective measures have been introduced by Piatetsky-Shapiro [11] and well studied by many researchers. Geng shows a summary of 37 different measures all having a clear theoretical background and characteristics [4]. However a typical business person is not familiar with the measures and has difficulties in understanding the business meaning for each measure. In this paper we will present three probability-based objective measures that are derived from a business process improvement levels. Business people can decide the level of improvement they are planning to achieve and select a measure based on that level.

3 Influence Analysis Methodology

3.1 Identify the Relevant Business Process and Define the Case

First task is to identify a high level problem in the operations. If there is no problems, then the potential for business improvements is zero and our method gives no results. After identifying the high level problem we continue by identifying the business process whose instances will be classified as successful or problematic based on whether they experienced the problem or not. If all cases are problematic then again our approach gives no results since improving every area in the organization would be similarly beneficial.

3.2 Collect Event and Case Attribute Information

Scope of our analysis depends on the amount of data available for the analysis in event and case logs. Since our goal is to create new insight for business people we encourage to use all possible event and case attribute data that is available, even though that typically introduces a lot of noise and data that is not relevant regarding the analysed problems. Generation of suitable log files with extended attributes is well studied area [3]. There also exists methods for enriching and aggregating event logs to case logs [15]. Here are some key steps for constructing event and case logs:

- Starting point is to identify the relational database table whose rows correspond to cases C.
- Identify for each case c_i in C, a set of objects O_i such that every object o_{ij} in O_i is linked to c_i directly. Then add recursively all objects linked to o_{ij} as long as the objects seem to be relevant concerning the analysis objectives. Note that since all tables in relational databases are typically somehow linked to each other this may lead to thousands of linked objects for each case.
- Form event log for c_i by including one event for every timestamp attribute of the case c_i and any linked object o_{ij}.
- Form case log for c_i by aggregating all attribute values of c_i and every object o_{ij} in O_i, thus creating potentially thousands of case attributes for each case. Suitable aggregation functions include *count*, *sum*, *max*, *min*, *average*, *median*, *concatenate*, *first* and *last*.
- Further augment every case c_i by adding external events that have occurred during the lifetime of the case. Example of external events include *machinebreak-started*, *machinebreak-completed*, *weekend*, *strike*, *queuetoolong* and *badweather*.

3.3 Create New Categorization Dimensions

The purpose of this step is to create new categorization dimensions for the cases. All these dimensions will then be used when finding the best improvement focus areas, so the more dimensions we have the larger the coverage of our analysis will be. Table 4 shows examples of dimensions that can be created for every event log based on the log itself.

3.4 Form a Binary Classification of Cases Such that Each Case Is Either Problematic or Successful

Purpose of this step is to express any discovery related to a business process as an attribute value for each process instance. This binary classification attribute specifies whether the case is problematic or successful. In practice a wide range of process mining methods can be used to make process discoveries [16].

Table 5 shows some example business problems that have been discovered using process mining methods and the corresponding illustrative functions for creating binary classification.

3.5 Select a Corresponding Interestingness Measure Based on the Desired Level of Business Process Improvement Effect

In this step we select an interestingness measure that will be used for finding the best business improvement areas. We propose the following requirements for the interestingness measure:

1. *Easy-to-understand by business people.* Business people are supposed to make actual decisions based on the analysis results so they must understand the

Table 4. Illustrative category dimensions

New category dimensions	Business rationale for including the dimension
Amount of Events per case, amount of Unique Events per case	Cases with very large or small amount of events often behave differently than the others
Start and end timestamp of the whole case	Exact calendar date, month or week is used to detect process changes over the time. Day of the week and Month of the year are useful for discovering periodic and seasonal behaviour
Start and end time of an individual event type	Same rationale as the case level attribute above, this will create at least one new dimension for each event type
One new dimension for specifying the amount of event occurrences separately for each event type	Often the fact that a particular event is executed several times for a case is a root cause for business problems

Table 5. Illustrative binary classifications for discovered business problems

Business problem discovered using process mining methods	Illustrative function for creating the binary classification
Some cases are not completed within the agreed service level agreement	$c.totalduration() > ServiceLevelAgreement$
Cases should not include multiple AddressChanged activities	$c.activitycount('AddressChange') > 2$
Suspiciously many cases have started in March 2015	$c.startmonth() =' 2015 - 03'$
First AddressChanged event should not be recorded by John	$c.getActivity('AddressChanged').first().recordedBy() =' John'$
Size of produced product have bigger than agreed variance	$c.product().size() - mean(product.size()) > \sigma$

results. It is thus important to minimize magic in our analysis and give as simple to understand business meaning for the results as possible.

2. *Big Benefits.* Selected interestingness measure should identify areas that include as many problematic cases as possible. This requirement corresponds to the benefit dimension of Table 1.
3. *Small Effort.* Implementing the change should require as small effort as possible. This requirement corresponds to the effort dimension of Table 1.

Regarding the first requirement *easy-to-understand by business people* we have identified three corresponding target levels for operational business improvements that business people are familiar with:

1. *ideal.* Improvement project will be ideal, all problems will be removed and after the project every future case will be completed without any problems.
2. *other average.* Focus area can be improved so that it reaches the current average performance of other areas. After the improvement project the share of problematic cases in the focus area will be equal to the average share of problematic cases in the other business areas before the improvements.
3. *as-is average.* Focus area can be improved so that it reaches the current average performance of all areas. After the improvement project the share of problematic cases in the focus area will be equal to the average share of problematic cases in the whole business before the improvements.

Regarding the second requirement *Big benefits* we calculate the overall density of problematic cases after the improvement. Table 6 shows these overall density measures calculated for the three identified change types when A is the set of cases selected as a target for business process improvement, B is the set of problematic cases before improvement and B' is the set of problematic cases after improvement.

Table 6. Change types

Change type	To-Be density of problematic cases for the selected segment A after the change $P(B'\|A)$	Overall to-be density of problematic cases after the change $P(B') = P(B'\|A)P(A) + P(B\|\bar{A})P(\bar{A})$	Change in overall density of problematic cases $P(B') - P(B)$
ideal	Zero density $= 0$	$0P(A) + P(B\|\bar{A})P(\bar{A}) = P(B) - P(AB)$	$-P(AB)$
other average	Average of current cases excluding this segment $= P(B\|\bar{A})$	$P(B\|\bar{A})P(A) + P(B\|\bar{A})P(\bar{A}) = P(B\|\bar{A})$	$P(B\|\bar{A}) - P(B)$
*as-is*average	Average of current cases including this segment $= P(B)$	$P(B)P(A) + P(B\|\bar{A})P(\bar{A}) = P(A)P(B) + P(B) - P(AB)$	$P(A)P(B) - P(AB)$

Regarding the third requirement *Small Effort* we say that the effort needed to improve a segment is relational to the size of the segment $P(A)$, i.e. the bigger the segment is the bigger the effort needed to make improvement.

Table 7 summarizes the identified change types according to the three requirements. Change type *ideal* sorts the results by the amount of problematic cases thus maximizing benefits. Since it does not take into account the size of the segment at all it performs poorly against the small effort requirement. Change type *other average* performs well regarding the benefits but it fails to make a difference between different sized segments including all problematic cases. It is also a bit difficult for business people to understand since the benefit potential of each segment is related to the average performance of all other segments, which needs

to be realized separately for each segment. Change type *as-is average* performs well regarding the benefits, is easy enough to understand for business people and takes into account the cost needed to implement the change.

Table 7. Change types by requirements

Change type	Easy to understand	Big benefits	Small effort to achieve
ideal	+ + +	+ + +	-
other average	+	+ +	+ +
as-is average	+ +	+ +	+ + +

Based on Table 7 we propose to use the change type *as-is average* as the target level for operational business improvements. We thus select the corresponding interestingness measure from Table 6 as $P(AB)-P(A)P(B)$, which is also known as $Leverage(A \rightarrow B)$. Business meaning of this measure is that if the segment specified covered by the antecedent of a rule is improved so that it reaches average performance, then the change in the total density of problematic cases is reduced by $P(AB) - P(A)P(B)$. For the communication purposes we define the following measures.

Definition 1. *Let B be a set of problematic cases and A be a set of cases that will be improved in order to reach an as-is-average density of problematic cases. Then the $Contribution(A \rightarrow B)$ is $n(AB)-\frac{n(A)n(B)}{N}$, where $n(AB)$ is amount of problematic cases in segment A before improvement, $n(A)$ is amount of cases in segment A, $n(B)$ is original amount of problematic cases and N is total amount of cases. This measure tells how many cases will be improved when business improvement is focused on segment A.*

Definition 2. *Let B be a set of problematic cases and A be a set of cases that will be improved in order to reach an as-is-average density of problematic cases. Then the $Contribution\%(A \rightarrow B)$ is $Contribution(A \rightarrow B)/n(B)$, where $n(B)$ is amount of problematic cases before business improvement. This measure tells how big share of the total business problem is improved when business improvement is focused on segment A.*

Definition 3. *Let B be a set of problematic cases and At be a case attribute for which all problematic segments will be improved in order to reach an as-is-average density of problematic cases. Then the $AttributeContribution\%(At \rightarrow B)$ is $\frac{1}{2}\sum_{A_i \in AttributeValues(At)} Abs(Contribution\%(A_i \rightarrow B))$, where $AttributeValues(At)$ is the set of all the sets of cases such that each individual set of cases contains all the cases having one specific attribute value for At. $AttributeValues(At)$ has thus one set of cases for every separate value for At. This measure tells how potential the attribute is as a target for business process improvement, the higher the value is the better the potential. The division by 2 is used to ensure that $AttributeContribution\%$ is always between 0 and 100 %.*

Attribute contribution is used to quickly identify those case attributes that contribute most to the finding. If there are large differences in the distribution of problematic cases for the different values of At, then the attribute contribution for At is high. If attribute contribution is low for attribute At, then we know that At does not include relevant causes for the problematic cases.

3.6 Find the Best Categorization Rules and Attributes

Run a rule learning algorithm using the information defined in previous steps. Analysis is performed by identifying a set of rules $A \rightarrow B$ where B is the binary classification value. Analysis shows how much the overall density of problematic cases changes when a selected business change is targeted to the segment covered by the antecedent A of the rule.

According to the requirement *easy-to-understand by business people* we have received good results by limiting the antecedent A to contain only one conditional attribute (=dimension/column) and one category value for the column. The fact that simple rules perform very well on most business datasets has also been presented by Holte [6]. It is also possible to construct antecedents based on multiple conditional attributes and using data mining algorithms to find combinations that have high contribution. However, if antecedents contain multiple attributes then benchmarking all combinations results in a very long report.

3.7 Present the Results to Business People

A full influence analysis report shows all discovered rules sorted by the selected interestingness measure. Top of the list contains the problematic cases (=best improvement areas) and bottom of the list contains the best practice examples.

Curse of dimensionality is typically a big problem when finding causes from several thousand or more features. Our methodology solves this during the presentation step by only showing a fixed amount of top and bottom rules. For example an analysis may contain 1.000 dimensions with a total of 100 million distinct single dimension antecedents. Our suggestion is to only show for example the top 100 and bottom 100 antecedents. In this way the interesting dimensions are likely to have at least some values in the top or bottom ranges and user can continue checking that attribute in more detail.

Another possibility is to show the report first only for the dimensions. In the previous example where we have 1.000 dimensions we first show them ordered by the *AttributeContribution*% and user the selects one attribute for more details.

Influence analysis report for one attribute show the antecedents for one case attribute at a time. This view is specifically easy to understand for business people since the problematic and best practice areas are clearly shown in this benchmark report.

4 Example Analysis

In this chapter we will present an example analysis conducted according to the methodology steps described in Sect. 3.

Table 8. Case data

Case	Product
1	Hats
2	Hats
3	Jeans
4	Shirts
5	Hats
6	Shirts
7	Shirts
8	Jeans
9	Shirts
10	Hats

Table 9. Event log data

Case	Event log
1	{order(20150101), orderchange(20150107), production(20150115, Ger), delivery(20150119)}
2	{order(20150101), production(20150107, Ger), delivery(20150110)}
3	{order(20150101), orderchange(20150108), production(20150115, Swe), delivery(20150121)}
4	{order(20150101), production(20150112, Fin), delivery(20150113)}
5	{order(20150101), orderchange(20150110), production(20150120, Fin), delivery(20150127), delivery(20150206)}
6	{order(20150101), production(20150108, Ger), delivery(20150113)}
7	{order(20150101), production(20150106, Ger), delivery(20150112)}
8	{order(20150101), production(20150108, Fin), delivery(20150114), delivery(20150122)}
9	{order(20150101), production(20150112, Ger), delivery(20150117)}
10	{order(20150101), production(20150111, Ger), delivery(20150118)}

1. Let us analyse an order to delivery process where each case is an order.
2. Table 8 contains case attribute information containing the product and region for each case. Table 9 contains an event log for each case specifying the activity name and date of the activity occurrence in format *yyyymmdd*. Event *production* also includes the name of the country where production was conducted as event attribute.
3. Table 10 shows new categorization attributes that have been calculated based on the previous data. *Duration days* is based on total case duration. *#del* is the amount of events of type *delivery* occurring in the case. *Region* is the production country taken from the event *production*. *weekday* is the day of the week when the *production* event was conducted. *#order changes* is the amount of events of type *order change* occurring in the case. *trace* is the full event type sequence for the whole case.
4. Problematic cases are identified with a binary classification B such that $B = true$ if $durationdays \geq 20$ else $false$. With this classification the cases 3, 5 and 8 have $B = true$ so the original density of problematic cases is $P(B) = 3/10 = 0.3$

Table 10. Example derived case data

Case	Dur. days	#del	Region	Weekday	#order changes	Trace
1	18	1	Ger	Fri	1	order-orderchange-production-delivery
2	9	1	Ger	Thu	0	order-production-delivery
3	20	1	Swe	Fri	1	order-orderchange-production-delivery
4	12	1	Fin	Tue	0	order-production-delivery
5	36	2	Fin	Wed	1	order-orderchange-production-delivery-delivery
6	12	1	Ger	Fri	0	order-production-delivery
7	11	1	Ger	Wed	0	order-production-delivery
8	21	2	Fin	Fri	0	order-production-delivery-delivery
9	16	1	Ger	Tue	0	order-production-delivery
10	17	1	Ger	Mon	0	order-production-delivery

Table 11. Contribution values for all rules $A \rightarrow B$ where B is $durationdays \geq 20$

Antecedent	n(A)	n(AB)	ideal		average		as-is avg	
			$\Delta_1 n$	$\Delta P(B_1)$	$\Delta_2 n$	$\Delta P(B_2)$	$\Delta_3 n$	$\Delta P(B_3)$
$\#deliveries = 2$	2	2	-2	-0.2	-1.75	-0.18	-1.4	-0.14
$product = jeans$	2	2	-2	-0.2	-1.75	-0.18	-1.4	-0.14
$customer = female$	6	3	-3	-0.3	-3	-0.3	-1.2	-0.12
$\#orderchanges = 1$	3	2	-2	-0.2	-1.57	-0.16	-1.1	-0.11
$Region = Finland$	3	2	-2	-0.2	-1.57	-0.16	-1.1	-0.11
$ProductionWeekday = Fri$	4	2	-2	-0.2	-1.33	-0.13	-0.8	-0.08
$Region = Sweden$	1	1	-1	-0.1	-0.78	-0.08	-0.7	-0.07
$trace = order - orderchange - production - delivery - delivery$	1	1	-1	-0.1	-0.78	-0.08	-0.7	-0.07
$trace = order - production - delivery - delivery$	1	1	-1	-0.1	-0.78	-0.08	-0.7	-0.07
$ProductionWeekday = Wed$	2	1	-1	-0.1	-0.5	-0.05	-0.4	-0.04
$trace = order - orderchange - production - delivery$	2	1	-1	-0.1	-0.5	-0.05	-0.4	-0.04
$product = hats$	4	1	-1	-0.1	0.33	0.03	0.2	0.02
$ProductionWeekday = Mon$	1	0	0	0	0.33	0.03	0.3	0.03
$ProductionWeekday = Thu$	1	0	0	0	0.33	0.03	0.3	0.03
$ProductionWeekday = Tue$	2	0	0	0	0.75	0.08	0.6	0.06
$\#orderchanges = 0$	7	1	-1	-0.1	3.67	0.37	1.1	0.11
$customer = male$	4	0	0	0	2	0.2	1.2	0.12
$product = shirts$	4	0	0	0	2	0.2	1.2	0.12
$\#deliveries = 1$	8	1	-1	-0.1	7	0.7	1.4	0.14
$Region = Germany$	6	0	0	0	4.5	0.45	1.8	0.18
$trace = order - production - delivery$	6	0	0	0	4.5	0.45	1.8	0.18

5. Table 11 shows the influence analysis results for each of the presented three change types: *as-is average*, *other average* and *ideal*. Results are sorted by the change type *as-is average* effects. According to these results the business improvement efforts should focus in segments $\#deliveries = 2$ and $product = jeans$, since in both of these segments the amount of problematic cases will drop by 1.4 as shown in column $\Delta_3 n$.

5 Case Study: Rabobank Group ICT

We evaluated the influence analysis with a publicly available data from Rabobank Group ICT used in BPI Challenge 2014 [18]. The data contained 46.616 cases and a total of 466.737 events. After a process mining analysis we discovered that the average duration for cases is 5 days and median duration is 18 h. We decided to consider all cases that took more than one week to complete as problematic resulting in a total of 7.400 (15.9 %) problematic cases. Table 12 shows that the biggest contributor for this finding is $Impact = 5$. There is a total of 16.741 cases with $Impact = 5$, out of which 3.535 (21.1 %) are problematic. As a *contribution*% this corresponds to 11.9 % of the total amount of problematic cases. For process performance point of view this is intuitive since it is probably acceptable to have low (5 = lowest on scale 1..5) impact cases taking a long time compared to higher impact cases. Table 12 also shows that 28.5 % of cases having ServiceComp WBS (CBy) equal to WBS000091 are completed in more than one week, which makes WBS000091 a candidate for business process improvements. If WBS000091 would reach the average level of performance, then there would 4.2 % less problematic cases.

Table 13 shows antecedents that have the biggest negative contribution. These can be regarded as the reasons why cases are completed within one week more often than average. If $\#Reassignments$ is zero, then only 5.9 % of cases will take more than one week. If these cases would take as long time as average

Table 12. Top positive contributors

Antecedent	n(A)	n(AB)	$P(B\|A)$	*Contribution*	*Contribution%*
Impact = 5	16741	3535	21.1 %	877	11.9 %
Urgency = 5	16779	3538	21.1 %	874	11.8 %
Priority = 5	16486	3473	21.1 %	856	11.6 %
# Related Interactions = 2	2736	1108	40.5 %	674	9.1 %
#Update From Customer = 1	1692	793	46.9 %	524	7.1 %
Closure Code = Other	16470	3137	19.0 %	522	7.1 %
# Reassignments = 2	5378	1340	24.9 %	486	6.6 %
# Reassignments = 3	2191	814	37.2 %	466	6.3 %
# Reassignments = 4	1606	701	43.6 %	446	6.0 %
Category = request for information	8846	1810	20.5 %	406	5.5 %
CI Type (CBy) = computer	3404	865	25.4 %	325	4.4 %
ServiceComp WBS (CBy) = WBS000091	2453	700	28.5 %	311	4.2 %
CI Type (CBy) = application	29456	4979	16.9 %	303	4.1 %

Table 13. Top negative contributors

Antecedent	n(A)	n(AB)	$P(B\|A)$	*Contribution*	*Contribution%*
# Reassignments = 0	27468	1628	5.9 %	−2732	−36.9 %
# Related Interactions = 1	43058	5907	13.7 %	−928	−12.5 %
Reopen Time = (blank)	44332	6285	14.2 %	−752	−10.2 %
ServiceComp WBS (CBy) = WBS000073	13173	1401	10.6 %	−690	−9.3 %
Service Component WBS (aff) = WBS000073	13342	1437	10.8 %	−681	−9.2 %
Impact = 3	6591	602	9.1 %	−444	−6.0 %
Priority = 3	6703	620	9.2 %	−444	−6.0 %
Urgency = 3	6536	607	9.3 %	−431	−5.8 %
CI Type (CBy) = subapplication	7711	800	10.4 %	−424	−5.7 %
Closure Code = User error	3554	152	4.3 %	−412	−5.6 %
Category = incident	37748	5582	14.8 %	−410	−5.5 %
CI Type (aff) = subapplication	7782	841	10.8 %	−394	−5.3 %
CI Name (aff) = SUB000456	3050	138	4.5 %	−346	−4.7 %

Table 14. Benchmark of distinct values of ServiceComp WBS (CBy)

ServiceComp WBS (CBy)	Contribution
WBS000091	4.2 %
WBS000072	2.8 %
WBS000088	2.4 %
WBS000162	2.2 %
WBS000263	1.4 %
WBS000296	1.4 %
WBS000271	1.1 %
WBS000092	0.9 %
...	
WBS000128	−0.6 %
WBS000094	−0.6 %
WBS000307	−0.7 %
WBS000152	−0.7 %
WBS000016	−0.8 %
WBS000228	−1.0 %
WBS000095	−1.7 %
#N/B	−1.7 %
WBS000073	−9.3 %

Table 15. Analysis on case attribute level

Case attribute	Attribute contribution
Handle Time (Hours)	54 %
KM number	38 %
#Reassignments	37 %
CI Name (CBy)	35 %
CI Name (aff)	34 %
Service Component WBS (aff)	27 %
Related Interaction	26 %
ServiceComp WBS (CBy)	24 %
Closure Code	15 %
#RelatedInteractions	13 %
Impact	12 %
Urgency	12 %
Priority	12 %
CI Type (CBy)	10 %
CI Subtype (CBy)	10 %
CI Subtype (aff)	8 %
CI Type (aff)	6 %
Category	6 %
#RelatedIncidents	1 %
Related Change	1 %
#RelatedChanges	0 %
Status	0 %
Alert Status	0 %

cases, then there would be 36.9 % more problematic cases. Another observation from Table 13 is that only 10.6 % of cases having ServiceComp WBS (CBy) equal to WBS000073 are completed late, which makes WBS000073 a positive benchmark.

ServiceComp WBS (CBy) was identified both as having a high positive and negative contribution. For business people it is often beneficial to show the contribution of all distinct values for this case attribute in one list order by contribution as shown in Table 14.

In Table 15 we see all case attributes listed by their attribute contribution. Obviously *HandleTime* in hours correlates strongly with case duration. Case attributes *CINames* and *ServiceComponents* have a strong correlation with cases taking a long time, which can be seen from their high attribute contribution. We also see that *#RelatedChanges* and *AlertStatus* have a very small effect to cases taking more than one week.

In this chapter we used influence analysis with real case data. We were able to identify causes for cases lasting more than one week. We also observed a benchmarking report for a particular case attribute $ServiceCompWBS(CBy)$ that seems to contribute a lot to the finding. All the results have been shown in easy-to-understand lists ordered by the contribution metric. If these results would have been shown to the business people it is likely that they would have combined this information with their tacit knowledge and discovered even more underlying cause-effect relationships.

6 Summary

In this paper we have presented a methodology that makes operational development more effective. Our methodology is suitable for every business process that has large enough volume of cases. Using our influence analysis method a workshop group consisting of business people identifies problems and focuses business improvement resources for eliminating these problems.

We have first shown how to collect the required data and how to process the data by creating new dimensions and binary classification metric. We then present an interestingness measure that is easy to understand by business people and helps in selecting the focus area for business process improvement such that it maximizes improvement benefits and minimizes implementation costs. We propose using the change type *as-is average* with interestingness measure $P(AB) - P(A)P(B)$. We then defined three measures *Contribution*, *Contribution%* and *AttributeContribution%* to be used in influence analysis report. Finally we have applied our analysis to a real-life data.

We have used the influence analysis in more than 100 customer projects during the past 5 years. In practice the problem areas and best practice areas have been accurately discover by influence analysis. Influence analysis is implemented to a commercial product [12] showing both the change type *ideal* and change type *as-is average* results. Interactive usage in workshop meetings has proven to be very valuable and it motivates business people in same room to share their tacit knowledge to deepen the influence analysis findings. Typical scenario is that participants first try to guess the most influencing factors and when they then see the results their own hypotheses are strengthened or weakened. This process further facilitates participants' thinking and collaboration with each other. Based on the discussion the organization then selects the focus areas for business process improvements and starts monitoring the performance on monthly intervals using the same contribution measures.

This method applies to finding root causes for problems that occur very rarely as well as to maximizing objectives like delivery accuracy that should reach about

99 % performance. Also the method can be used to evaluate potential risks in any given segment by checking those areas that have low density of problematic cases in the as-is situation. Influence analysis also has an important application in deciding whether the organization should improve the whole process design or improve certain problem areas. If the contribution values for all rules are relatively low, then there is no clear problem that should be fixed. Thus if no focus area is found and business still needs to be improved, there is a need to improve the whole process design.

Acknowledgements. We thank QPR Software Plc for the practical experiences from a wide variety of customer cases and for funding our research.

References

1. Andersen, B., Fagerhaug, T.: Root Cause Analysis: Simplified Tools and Techniques. ASQ Quality Press, Milwaukee (2006)
2. Bay, S., Pazzani, M.: Detecting group differences: mining contrast sets. Data Min. Knowl. Disc. **5**(3), 213–246 (2001)
3. de Leoni, M., van der Aalst, W.M.P., Dees, M.: A general framework for correlating business process characteristics. In: Sadiq, S., Soffer, P., Völzer, H. (eds.) BPM 2014. LNCS, vol. 8659, pp. 250–266. Springer, Heidelberg (2014)
4. Geng, L., Hamilton, H.J.: Interestingness measures for data mining: a survey. ACM Comput. Surv. (CSUR) **38**(3), 9 (2006)
5. Goldratt, E.M.: Theory of Constraints. North River, Croton-on-Hudson (1990)
6. Holte, R.C.: Very simple classification rules perform well on most commonly used datasets. Mach. Learn. **11**(1), 63–90 (1993)
7. Inmon, W.H.: Building the Data Warehouse. Wiley, New York (2005)
8. Kakas, A.C., Kowalski, R.A., Toni, F.: Abductive logic programming. J. Logic Comput. **2**(6), 719–770 (1992)
9. Mayer-Schnberger, V., Cukier, K.: Big Data: A Revolution that Will Transform How We Live, Work, and Think. Houghton Mifflin Harcourt, Boston (2013)
10. Pearl, J.: Causality: Models, Reasoning and Inference, vol. 29. MIT Press, Cambridge (2000)
11. Piatetsky-Shapiro, G.: Discovery, analysis and presentation of strong rules. In: Knowledge Discovery in Databases, pp. 229–248 (1991)
12. QPR Software Plc.: QPR Software to Offer Business Process optimization with Automated Business Process Discovery Software QPR Process Analyzer, Press release 15 Feb 2011
13. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986)
14. Rozinat, A., van der Aalst, W.M.P.: Decision Mining in ProM. Springer, Heidelberg (2006)
15. Suriadi, S., Ouyang, C., van der Aalst, W.M.P., ter Hofstede, A.H.M.: Root cause analysis with enriched process logs. In: La Rosa, M., Soffer, P. (eds.) BPM Workshops 2012. LNBIP, vol. 132, pp. 174–186. Springer, Heidelberg (2013)
16. van der Aalst, W.M.P., et al.: Process mining manifesto. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 169–194. Springer, Heidelberg (2012)

17. van der Aalst, W.M.P., Adriansyah, A., van Dongen, B.: Causal nets: a modeling language tailored towards process discovery. In: Katoen, J.-P., König, B. (eds.) CONCUR 2011. LNCS, vol. 6901, pp. 28–42. Springer, Heidelberg (2011)
18. Van Dongen, B.F.: BPI Challenge 2014. Rabobank Nederland. Dataset (2014). http://dx.doi.org/10.4121/uuid:c3e5d162-0cfd-4bb0-bd82-af5268819c35
19. Webb, G.I., Butler, S., Newlands, D.: On detecting differences between groups. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2003 (2003)