Mining and Modeling the Phenomenology of Situational Awareness

Monte Hancock[1] and Michael Higley[2]

[1] Celestech Inc., United States
monte.hancock@celestech.com
[2] Engineering Development,
SDI Industries, Inc.,
United States
mhigley@sdiindustries.com

Abstract. One expert has said "Most simply put, situational awareness (SA) is knowing what is going on around you." [1] "Knowing" is not just having a checklist of facts. Functionally, SA is about possessing information of sufficient scope and accuracy to support decision making that facilitates effective action. Augmented Cognition research shows that presenting too much data, even of high quality, can be as harmful to effective cognition as presenting little or no data [2]. Research has shows that in 35% of aviation errors in SA, all needed information was present, but not attended to by decision makers. [3] This work presents a formal but simple mathematical SA Model, and describes the application of data mining and modeling to SA errors resulting from inattention to the most salient facts. The model is applied to two data sets to demonstrate a general approach to automating the detection and diagnosis of SA errors.

Keywords: situational awareness, decision support, coincidental correctness, context error.

1 Organization of This Paper

After presenting in section 2 some historical background and information on current favored techniques in SA, section 3 goes through an “informal” SA case study from which four general SA principles are inferred. These motivate aspects of the formal treatment in section 5. Section 4 defines the notion of a specific type of SA loss (here referred to as a context error) as a violation of the tenets of a formal domain model called an ontology. This is foundational to the new presented in the following sections. Section 5 is the heart of the new work being presented. It gives a detailed method (with a particular algorithmic implementation, one of many possible) for using data mining and modeling to detect and characterize context errors. Section 6 summarizes our conclusions, and section 7 contains citations for references.

D.D. Schmorrow and C.M. Fidopiastis (Eds.): AC 2014, LNAI 8534, pp. 125–135, 2014.

2 Background

Functionally, SA is about possessing information that is of sufficient scope and accuracy to support decision making that facilitates effective action. A naive approach to acquiring and maintaining SA might attempt to present every conceivably useful datum to the decision maker. But research in augmented cognition and human factors has shown that presenting too much data, even if it is of high quality, can be as harmful to effective cognition as presenting little or no data [2]. Research has also shown that in 35% of aviation errors in SA, all the needed information was present, but not attended to by the decision maker.[3] Even when no data are available, a decision maker's resort to prior probabilities, well-crafted "best-practice", and time-tested operational protocols often produces acceptable outcomes, or at least favors actions unlikely to make things dramatically worse.

The first references to the term Situational Awareness (SA) are found in documents generated by the U.S Air Force shortly after the Korean War, and relate to tactical assessment of fighter pilot behavior in an aerial dogfight. Combat pilots need to know not only where the enemy plane is, but where it will be a few seconds hence. This involves gathering information, analyzing it, and making projections based on that analysis. This was described by Air Force Col. John Boyd as the "observe- orient-decide-act loop" or OODA loop, also called the Boyd cycle. To win a dogfight, he said, the pilot must "get inside" the opponent's loop, that is, to continually infer the opponent's assessment of the situation; losing one's own situational awareness was called being "out of the loop".[5]

SA finds natural application in complex decision problems, such as those requiring the fusion of many variables to establish an unambiguous foundation for decision making. For example, the power transmission and distribution industry has begun to apply SA to system state monitoring. This application was driven by the realization that "system operators could not assess the extent of the disturbance or what corrective action should be taken due to the volume and format in which data were displayed, making real-time evaluation of the situation more difficult."[6]

Probably the most well-established application of SA methodologies is for optimal (or at least, stable) control of multi-component "systems of systems", such as modern air traffic control, satellite communications, international shipping, automated manufacturing, large- scale military operations, and the like. Applications for optimal control are also finding their way into the analysis, protection, and control of computer networks, such as automated network intrusion detection. Less obvious, perhaps, are SA applications having an ergonomic emphasis, such as the formulation of best practice for medicine, law, and other professional specialties. In particular, the shift of clinicians to "evidence-based medicine" makes maintenance of situational awareness fundamental to modern clinical protocols.

A widely-used theoretical framework for situational awareness that marks the frontier of recent SA research is the Template Model. A template corresponds to a particular sub-process or mode in the problem domain (e.g., piloting an aircraft on final approach); it consists of a collection of assumptions about state variables in that mode, methods allowable for that mode, ad hoc inferencing rules specific to the mode, etc. It is a temporary, disposable ontology (section 4.1) suited to support decision-making in a focused subset of the domain. Templates are made executable by placing them within a state-machine harness that allows simulation and experimentation. In the Template Model, the preeminent task of the decision maker is the selection and instantiation of an appropriate template. This selection is informed by available information and domain knowledge, and is a realization of the decision maker's mental model of the domain. It is clear that in the Template Model, a decision-makers' selection/use of a template that conflicts with objective reality is a profound error. More generally, any shortfall in either knowledge or information that causes the decision-maker's mental model of the situation to deviate from objective reality could lead to incorrect decisions. We refer to incorrect decisions that arise from loss of situational awareness as context errors (CE) (section 4).

3 Informal Sa Model: System Engineering Case Study

The case study in this section recounts the SA "lessons learned" during a system development project. These are reduced to a collection of colloquial heuristics, general and useful in their own right, that correspond to phenomena observed in the formal results presented in section 5. The goal of this informal case study is the characterization of patterns that could have alerted system developers to loss of situational awareness, had they been noticed and understood.

3.1 It Worked the First Time

During system development, solution approaches had that worked initially later failed inexplicably. The work began with a prototype that was implemented in an environment of near perfect conditions… conditions assumed to be typical of all deployment sites. When the prototype design was finalized and implemented at three new sites, it proved to be unstable, and could meet specifications. Retrospective analysis determined that the early success had not really been understood; the factors underlying the early good results were not present in the other operational environments. The fundamental SA problem here was the attribution of good results to an inadequate design, rather than to an overly-forgiving test environment.

3.2 Silver Bullet

With under-performing prototypes now installed at several sites, an engineering plan for addressing stability issues was needed. Option one was to enumerate the specific problems, and address them separately. Option two was to try a system wide ad hoc add-on that might fix all the problems at once. The temptation to try the "one-size-fits-all" miracle was irresistible. In this case, developers faced the "Silver Bullet" challenge: given the choice between solving a complex problem with high probability in several steps, or hoping for the easy (but unlikely) one-shot win, human nature has a natural tendency to choose the latter. The loss of situational awareness here was in allowing wishful thinking to overrule scientific thinking.

3.3 Low Hanging Fruit

The Silver Bullet didn't address all the problems, but it did make the systems more stable. The remaining issues had to be triaged for sequential solution. Which should be addressed first? Human nature tends to follow the Pareto Effect: the task that has the highest fruitfulness to effort ratio will be worked first. In this case, the use of the Silver Bullet made the problems it didn't solve even harder. The loss of situational awareness here is failing to consider all the effects arising from a selected course of action.

3.4 Ignoring Red Flags

As the development passed the halfway point in cost and schedule, the question arose whether pulling the Silver Bullet out and going with piecemeal approach more likely to succeed should be considered. Discarding a partially successful course already embarked upon in favor of one that is (in hindsight) clearly preferable often makes engineering sense. But it is hard to believe that it makes cost and schedule sense, or psychological sense. Even hard-bitten engineers can become so committed an approach that "going down with the ship" is preferable to changing course.

In this case, there were early indications that a core system component was unlikely to be durable enough for the deployment environment. However the option to replace it was not considered; the decision made was to find work-arounds, an arguably unnecessary compromise solution. If this red flag had been heeded earlier, the work-arounds would have been unnecessary. The loss of situational awareness here was intentional inattention to important observations.

4 Context Errors

The context of a problem domain is the (often dynamic) totality of all its hidden and observable state variables. In this sense, the context can be thought of as the empirical "reality" which the decision maker must comprehend to be effective.

Colloquially stated, a context error has been committed when the decision maker thinks things are one way, while they are actually another. Context Errors are particularly difficult to address, because decision makers can become psychologically invested in a template choice that was made early in the reasoning process, when available information was incomplete. Research has shown that even experienced experts will sometimes develop elaborate rationalizations to discount recent information that conflicts with the mental model created from earlier information… a kind of "anti-recency effect". [4]

In this paper, context errors are detected by using a regression procedure to reconstruct each attribute of a situation vector from the vector's other attributes; this will not be possible if the vector of attributes holds internal inconsistencies under the ontology. This has the advantage of allowing historical data to establish the definition of "nominality", as well as thresholds for normalcy, and associations that reveal the causes of inconsistency.

4.1 Ontologies

An ontology is a framework for specifying and interrelating the components of knowledge within a domain. It is assumed here that a reasoner's mental model of a situation is created by selecting and associating elements across the components of an ontology, usually based upon sensory input. Our ontologies consist of seven components:

1. a lexicon of terms
2. a representational scheme for using terms to express facts, concepts, and rules
3. a set of state variables, $S = \{s_k\}$
4. a set of constraints (i.e., equations of state)
5. long-term memory (information base of static facts and concepts)
6. short-term memory (recent history of facts, state-variable values)
7. Templates (scripts, checklists of customary/best practice, rule sets, etc.)

The following is an example illustrating the content of each component for a very tiny ontology. An ad hoc phrase structure grammar, lists, and functional notation are used to express facts and concepts; predicate logic is used to express rules:

1. Eiffel Tower: instance(thing)
   Eiffel Tower The Movie!: instance(thing)
   political unit: instance(cultural abstraction)
   city: instance(political unit)
   Paris: instance(city)
   Bijou: instance(business)
2. [attribute1(entity)=v1, …,attributeN(entity)=vn];…
3. [location; ground speed; …]
4. ground speed(Paris)=0; ground speed(Bijou)=0;…
5. location(Eiffel Tower)=Paris
   location (Eiffel Tower The Movie!)=Bijou
6. time=1400 GMT
7. [location(x)=y AND ground speed(y)=0] → (go to x → go to y)

Other formalisms are possible; for example, this ontology could be modeled as a set of graphs and a State Machine. Yet another approach is presented in chapter 11 of [7].

A context error can occur when the reasoner's mental model of the situation is predicated upon an incorrect association. For example, suppose a reasoner wants to see "Eiffel Tower The Movie!". Conflating the movie with the structure in component 1 will result in the moviegoer booking a flight to Paris. A context error is committed.

5 Data Mining and Modeling for SA

The core of the new work is presented in the following sections.

1. **A data mining method for detecting SA errors as <u>contextual anomalies</u>:** deviations from nominality arising from violations of domain context.
2. **A data mining method for formulating a preliminary <u>SA Hypothesis</u>:** a clique of indicators suggesting the nature of a suspected "context error".

Data mining is the principled detection, characterization, and exploitation of actionable patterns in data. In practice, it amounts to the application of the scientific method to data, and so is essentially a modeling activity. [7]

Data mining is used in this work to detect and characterize latent information accompanying, or antecedent to, loss of SA in a problem domain.

The Colloquial SA Model serves as a guide, identifying specific types of SA loss. Can these be detected by automated means? If detectable, to what extent can they then be characterized?

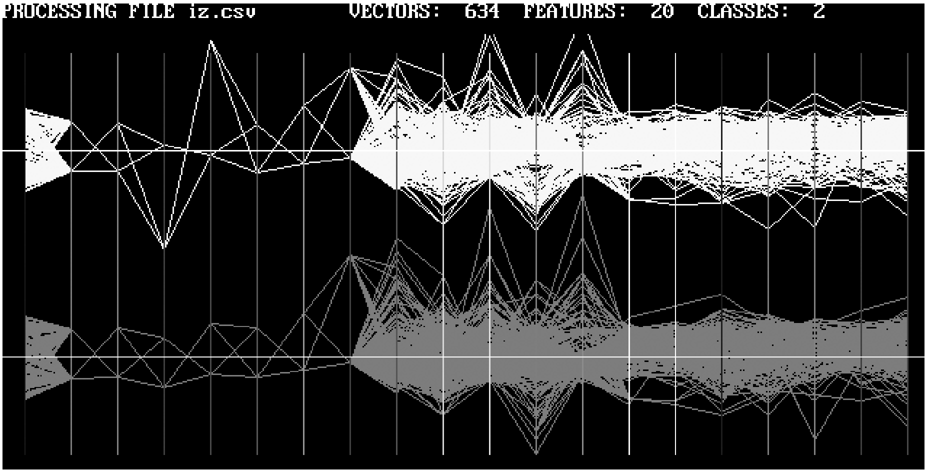
5.1 Informal SA Heuristics Revisited

- It Worked the First Time (coincidental correctness)
  In computer science, this type of error is referred to as coincidental correctness: getting the right answer for the wrong reason. Human decision makers tend to stick with what works, particularly when under pressure. When coincidental correctness confirms their confidence in a flawed mental model, they can become committed to it, even to the point of discounting hard contradictory evidence.
- Silver Bullet
  The most direct path to a solution might not be the best path. In the case study, engineers tried to solve problems caused by earlier lapses in judgment by going for the "quick fix". But this saddled them with an embedded obstacle over which they stumbled for the rest of the development.
- Low Hanging Fruit (confusers)
  In the case study, giving in to the temptation to gain easy benefit by perusing a sub-optimal strategy made subsequent work on the project more complex. Sometimes the side-effects of even good ideas are not themselves very good at all. In the formal SA work described below, it is shown that having more information is not always better if that information of limited relevance.
- Ignoring Red Flags (misplaced focus)
  In Ergonomics, "Tunnel Vision" is a well-known condition where reasoners focus on a small number of trusted and well-understood factors to the exclusion of all others. Emerging problems that could have been detected and addressed are missed because its indicators were not among those considered by hyper-focused decision makers.

The results of our experiment testing these principles are in section 5.3.2.

5.2 Description of the Data Set

The data used in the empirical modeling experiments for this work consist of 634 rows (instances), each having 20 attributes. Attributes 1 – 7 were nominal demographic data, and attributes 8 – 20 were numeric. There were two ground-=truth classes in the data: class 1, and class 2. The data are nearly balanced by class. The data were z-scored by columns as part of the conditioning process. The Inselberg Plot shows each attribute on its own axis; class 1 is the bottom trace set, and class 2 is the top trace set:



5.3 Emprirical Mining and Modeling Process and Results

The data set were subjected to two types of analysis: mining to determine the relative salience of attributes, and modeling to detect and characterize indicators of SA loss.

Principled analysis of both types proceeds by deliberate, ordered stages, each formulated to feed into the next so that the latent information bound across the data set is made more accessible. The procedures used for this work are now described.

Data Staging. Once data have been collected, they must be organized, or staged, for attribute extraction. This includes two activities: Randomization, and Partitioning.

Randomization refers to any process that eliminates associations that are side effects of the data collection and storage process. For example, if data are in a flat file (e.g., a spreadsheet) that has been sorted on some primary key (such as collection date, location, alphabetically, etc.), records will be ordered in a manner that could produce biased sets if they are sampled in a naïve way.

Partitioning separates the data into two sets:

Analysis Set. This set is used for model development. Any classifiers, detectors, decision support applications, etc., will be developed using this data. It is the “training data.”

Validation Set. This set is used to determine the performance of any models created.

Both of these sets must be representative samples of the problem space within which the data mining results will be applied. This generally means that they are numerically balanced in the way they represent the problem, and are not skewed by sampling bias. For example, they will have approximately the same proportion of each ground truth class as the population.

Experiment to Determine the Relative Salience of Attributes. To use the data for SA experiments, the ground truth class was regarded as the process goal, and the attributes held in the feature vectors were regarded as the state variables.

The first experiment performed was to determine which attributes were the most informative for determining the ground truth class. Knowing which attributes are most informative, the principles from the informal case study can be assessed.

There are many ways to assess a subset of features for information content. A notional description of a Monte Carlo approach is now described. The information assessment begins by reading in the data to be analyzed, and computing the mean and standard deviation for each feature for each of the ground truth classes. That is, the mean and standard deviation are computed for each column for all the rows that are in ground truth class 1, giving the center and variability of the class 1 data; then, for class 2 data, and so on.

To determine which columns contain information useful for classification of the data into its ground truth classes, we tested all 2^n-1 non-empty subsets of the available columns while selecting subsets randomly, and keep track of which subset gives the best results for a weighted nearest neighbor classifier (described later). The process operates as follows:

Table 1.

Algorithm Phase A	
Step 1	Read in the whole data file
Step 2	Segment into calibration, training, and validation files (row order randomized)
Step 3	Compute centers and standard deviations for each class in the calibration segment

Table 2.

Algorithm Phase B	
Step 1	Select a subset of the columns to test (a *clique*)
Step 2	Use the centers and standard deviations computed in Phase A for the clique to assign each data point in the training segment to a class (weighted nearest neighbor classifier)
Step 3	Compute performance statistics for this clique (e.g., % correct) on the training segment

Repeat Phase B for all 2n-1 non-empty attribute cliques. The attributes in the best clique (highest accuracy score on the test set) are the ones that, as a group, have the most useful information for classification of those tested.

A weighted nearest-neighbor classifier is based upon well-known statistical principles. It was chosen for this application for several reasons, but the most important is that no retraining is required when a new feature clique is to be evaluated; features not selected are ignored in the calculation. This makes it possible to run a large number of clique tests very quickly. Runs for this work had up to 223-1 ~ 8.3 million attribute cliques.

Here the results of this data mining study of attribute salience for three of our informal heuristics:

Coincidental Correctness: We found that there are collections of attributes that work well for certain portions of the data space, but are sub-optimal when applied across the entire problem space.

Confusers: We found that even when the best clique of attributes was used, the inclusion of other attributes in the clique could dramatically degrade the performance on goal class recognition.

Misplaced Focus: Several features were regarded a priori (based upon conventional wisdom) to be particularly salient, but were found to have much less salience than combinations of seemingly weaker attributes. Maintaining a tight focus grounded in prior beliefs would result in frequent loss of situational awareness in this domain.

5.4 Anomaly Detection

The goal of our SA work is look for a method of identifying nascent indicators of SA loss, addressing such issues as Tunnel Vision, Coincidental Correctness, etc.

It is hypothesized that if the decision-maker's mental model of the situation does not match objective reality, at some point inconsistencies will begin to emerge in the components of that model. These will be seen as state vectors assuming values which, while they might individually be valid, should be occur simultaneously if the system is in a nominal state.

Because of the complexity of modern systems, a decision maker under stress might not be able to fuse subtle, multi-factor indications of SA loss. For example, the National Transportation Safety Board has concluded that stress/complexity-induced loss of SA has been a factor in many aircraft accidents.

We use intra-vector regression for anomaly detection in the vectors of attributes. If a pattern consisting of several parts is not unusual, then it should be possible to hide some of its parts, and use pattern matching to infer these hidden parts from those that are not hidden. In a certain sense, parts that can be inferred in this way conform to what is expected, and are not novel. However, when some part of a pattern cannot be inferred from the others, it must in some way be unusual in the context of the whole pattern.

For example, suppose a medical record says a patient's gender is male, and their diagnosis is gestational diabetes (i.e., the patient is a pregnant man). These feature values cannot both be correct, so an anomaly has been detected.

This suggests a method for using pattern matching to detect anomalies. For each part of a pattern, a learning engine is created to infer that part from the others. Items are run through the engine to determine whether all of their parts make sense in context. Items that contain many parts that cannot be inferred by the engine are deemed novel. Using scores computed during processing by the learning engine, items are given anomaly scores.

5.5 An Anomaly Detection Algorithm for Numeric Data

One simple inter-vector imputation method is to replace missing values with their population means, a O(n) process. This naïve approach is simple, but ignores context within the record. For numeric data, a more sophisticated method is the nearest neighbor normalization technique. This can be applied efficiently even to large data sets with many dimensions (in a brute force approach this is a $O(n^2)$ process). The following is an explanation of the nearest neighbor normalization method used in this work.

Anomaly Detection Procedure

$V_j = (f_{j,1}, f_{j,2}, f_{j,3}, \ldots, f_{j,k-1}, f_{j,k}, f_{j,k+1}, \ldots, f_{j,N}, g_j)$ – Situation being tested

${}_k\tilde{V}_j = (f_{j,1}, f_{j,2}, f_{j,3}, \ldots, f_{j,k-1}, \mathbf{0}, f_{j,k+1}, \ldots, f_{j,N}, g_j)$ – k^{th} feature projected out

$$\left.\begin{array}{l} {}_k\tilde{V}_1 = (f_{1,1}, f_{1,2}, f_{1,3}, \ldots, f_{1,k-1}, 0, f_{1,k+1}, \ldots, f_{1,N}, g_1) \\ {}_k\tilde{V}_2 = (f_{2,1}, f_{2,2}, f_{2,3}, \ldots, f_{2,k-1}, 0, f_{2,k+1}, \ldots, f_{2,N}, g_2) \\ {}_k\tilde{V}_3 = (f_{3,1}, f_{3,2}, f_{3,3}, \ldots, f_{3,k-1}, 0, f_{3,k+1}, \ldots, f_{3,N}, g_3) \\ \qquad \vdots \qquad\qquad\qquad \vdots \\ {}_k\tilde{V}_L = (f_{L,1}, f_{L,2}, f_{L,3}, \ldots, f_{L,k-1}, 0, f_{L,k+1}, \ldots, f_{L,N}, g_L) \\ \qquad \vdots \qquad\qquad\qquad \vdots \\ {}_k\tilde{V}_M = (f_{M,1}, f_{M,2}, f_{M,3}, \ldots, f_{M,k-1}, 0, f_{M,k+1}, \ldots, f_{M,N}, g_M) \end{array}\right\}$$

The Reference Data are a catalog of nominal situations expressed as features vectors in a state space.

— Reference data set having k^{th} feature projected out

A two-step Imputation Procedure: **(There are many ways this imputation can be performed. This one was selected here for simplicity of presentation.)**

"Scale in the corresponding feature from the nearest in-class reference vector"

1. Find ${}_k\tilde{V}_L$ = Reference vector having $g_L = g_j$ for which $\| {}_k\tilde{V}_L - {}_k\tilde{V}_j \|$ is smallest
2. Then the imputed value for $f_{j,k}$ is $\| {}_k\tilde{V}_j \,/\, {}_k\tilde{V}_L \| f_{L,k}$

Repeat this procedure for each feature to obtain S_j, the vector synthesized from the features of V_j by the two-step Imputation Procedure.

If $\| V_j - S_j \| > \theta_{g_j}$, the decision threshhold for class g_j, then V_j is anomalous.

Feature-level threshholds are then applied to the imputation errors:

If $\left| f_{j,k} - \| {}_k\tilde{V}_j \,/\, {}_k\tilde{V}_L \| f_{L,k} \right| > \varphi_{g_j}$, include feature j in the specific anomaly hypothesis:
"These aspects of the situation are out of nominal tolerances by these amounts."

Threshhold values can be set manually, or by reference to their sampling distributions.

The nearest neighbor normalization technique can be applied to nominal data, but in that application the available symbol in the matching vector is usually copied directly over without further processing.

6 Conclusions

Data Mining methods for feature evaluation can be used to construct SA algorithms to formalize informal SA principles.

Data Mining methods for anomaly detection can be used to develop automatic tools for detecting loss of situational awareness, and producing hypothetical forensic characterizations that are supported by objective, numerical methods.

Both of these approaches are calibrated during modeling using historical data sets, and do not require that a human expert establish rules, templates, or operational parameters a priori. The definition of "nominality" is inferred from historical data.

References

1. Endsley, M.: Situation Awareness Analysis and Measurement, ch. 1. Taylor & Francis e-Library. Lawrence Erlbaum Associates, Inc. (2009)
2. Hancock, M., Wood, K., et al.: Helping the User Fail : Ergonomic Modalities and Distraction. In: Kaber, D., Boy, G. (eds.) Advances in Cognitive Ergonomics, ch. 16. CRC Press (2011) ISBN 978-1-4398-3491-6
3. Jones, D.G., Endsley, M.R.: Sources of Situation Awareness Errors in Aviation. Aviation, Space and Environmental Medicine 67(6), 507–512 (1996)
4. Adelman, L., Bresnick, T., Black, P.K., Marvin, F.F., Sak, S.G.: Research with Patriot Air-Defense Officers: Examining Information Order Effects. Human Factors 38(2), 250–261 (1996)
5. Eric, S., Toner, M.D.: Creating Situational Awareness: A Systems Approach. N.p.: n.p. Institute of Medicine Forum on Medical and Public Health Preparedness for Catastrophic Events. Web (June 10, 2009), http://www.iom.edu/~/media/Files/Activity%20Files/PublicHealth/MedPrep/Jun-10-11-2009-Commissioned%20Papers/Jun-10-11-2009-Commissioned-Paper-Creating-Situational-Awareness-A-Systems-Approach.pdf (February 6, 2014)
6. Connors, E.S., Endsley, M.R., Jones, L.: Situation Awareness in the Power Transmission and Distribution Industry. In: Rep. SA Technologies. Web (2007), http://www.nerc.com/pa/rrm/hp/2012%20Human%20Performance%20Conference/SA%20for%20Power%20T%20D-HFES.pdf (February 6, 2014)
7. Hancock, M.F.: Practical Data Mining. CRC Press (2012) ISBN 978-1-4398-6836-2