


# Improving hospital layout planning through clinical pathway mining

Ines Verena Arnolds<sup>1</sup> · Daniel Gartner<sup>2</sup> 

Published online: 9 April 2017

© The Author(s) 2017. This article is an open access publication

**Abstract** Clinical pathways (CPs) are standardized, typically evidence-based health care processes. They define the set and sequence of procedures such as diagnostics, surgical and therapy activities applied to patients. This study examines the value of data-driven CP mining for strategic healthcare management. When assigning specialties to locations within hospitals—for new hospital buildings or reconstruction works—the future CPs should be known to effectively minimize distances traveled by patients. The challenge is to dovetail the prediction of uncertain CPs with hospital layout planning. We approach this problem in three stages: In the first stage, we extend a machine learning algorithm based on probabilistic finite state automata (PFSA) to learn significant CPs from data captured in hospital information systems. In that stage, each significant CP is associated with a transition probability. A unique feature of our approach is that we can generalize the data and include those CPs which have not been observed in the data but which are likely to be followed by future patients according to the pathway probabilities obtained from the PFSA. At the same time, rare and non-significant CPs are filtered out. In the second stage, we present a mathematical model that allows us to perform hospital layout planning decisions based on the CPs, their probabilities and expert knowledge. In the third stage, we evaluate our approach based on different performance measures. Our case study results based on real-world hospital data reveal that using our CP mining approach, distances traveled by patients can be reduced substantially as compared to using a baseline method. In a second case study, when using our approach for reconstructing a hospital and incorporating expert knowledge into the planning, existing layouts can be improved.

---

✉ Daniel Gartner  
gartnerd@cardiff.ac.uk

Ines Verena Arnolds  
ines.arnolds@kit.edu

<sup>1</sup> Institute of Operations Research, Discrete Optimization and Logistics, Karlsruhe Institute of Technology, Karlsruhe, Germany

<sup>2</sup> School of Mathematics, Cardiff University, Cardiff, UK

**Keywords** Machine learning · Sequential pattern mining · Clinical pathways · Hospital layout planning · Healthcare operations management

## 1 Introduction

Planning the layout of a new hospital or reconfiguring existing ones is a complex task and the development of quantitative planning approaches has gained attention since the late 1970s (Elshafei 1977). Confusing layouts can add to patients' anxiety (Landro 2014) and uncertainty in patient flows challenges strategic decision making in healthcare (Blumenthal 2009). Also, new treatment methods, length of stay reduction and shifting from inpatient to outpatient care can lead to variation and uncertainty in hospital-wide patient flows. Therefore, learning significant clinical pathways (CPs) from data and dovetailing them with strategic hospital decision making in the context of hospital layout planning is the focus of this study.

We approach the problem in three stages: In the first stage, we choose an algorithm to learn significant CPs from large transactional data. In that stage, we address the problem to determine CP probabilities including those which have not been observed in the data but are likely to occur in the future. In the second stage, we present a mathematical model that allows us to perform hospital layout planning decisions based on the CPs and their probabilities as learned in the first stage. In the third stage, we evaluate our approach based on a real-world setting using different performance measures.

The remainder of this paper is structured as follows. In the next section, we review literature on CP mining and the use of CPs for hospital operations management. In this section, we also differentiate our work from other hospital layout planning approaches and highlight similarities and differences with the estimation of rare event probabilities. In Sect. 3, we provide a description of the sequential pattern mining approach employed in our study. In this section, we also present a mathematical model for hospital layout planning and define performance metrics that will be evaluated in our study. In Sect. 4, a brief computational study is provided in order to demonstrate the effectiveness of our approach based on hypothetical data. In Sect. 5, we give a presentation of our results using real data. We finally provide a conclusion and outline streams for further research.

## 2 Related work

We break down related work into the following four streams: In the first stream, we delimit our work from evidence-based use of CPs since our work follows paradigms from data-driven CP and process mining. In the second stream, we review the use of CPs driven by healthcare operations management. We then highlight similarities and differences to related work on hospital layout planning problems and delimit our work from the estimation of rare event probabilities and sequential pattern mining approaches. We finally provide a summary of similarities and differences with existing work.

### 2.1 Medical-, data-driven clinical pathway and process mining

CPs support a consistent application of evidence-based medicine for the best patient outcomes. Often this has the effect of placing an emphasis on the reduction of unwarranted variation in clinical practice (Wennberg et al. 1977). Similarly, van de Klundert et al. (2010) define CPs as standardized, typically evidence-based health care processes. Instead of build-

ing our research on the paradigm of CPs as a sub-discipline of evidence-based medicine, we follow a data-driven approach to infer CPs from data which has been studied by [Zhang et al. \(2015a\)](#), among others. The novelty of our approach is, however, that expert opinions can be incorporated into the learning process as Sect. 3 will reveal and that we bridge the gap between CP mining and operational decision making.

Mining healthcare processes has been the focus of previous literature and a review of approaches is provided by [Rojas et al. \(2016\)](#). [Mans et al. \(2015\)](#) conclude that data mining techniques cannot be used for process discovery, conformance checking, and other forms of process analysis. Some research exists that analyzes process variation in healthcare based on similarity measures between CPs (e.g. [Huang et al. 2013](#); [Combi et al. 2009](#)). This is different to our work since we will allow for variations in CPs while filtering out non-significant ones. Adherence in CPs is investigated by [van de Klundert et al. \(2010\)](#). [Zhang et al. \(2015b\)](#) and [Zhang et al. \(2014\)](#) apply a hierarchical clustering approach to determine the most likely CP. The authors study the patients' health conditions and treatment approaches. [Iwata et al. \(2013\)](#) use a clustering and temporal analysis approach in order to identify missing information in existing CPs.

## 2.2 Healthcare operations-driven use of clinical pathways

Using CPs as an input of their model, [Cardoen and Demeulemeester \(2008\)](#) provide a strategic instrument for evaluating future changes to the hospital setting. Their simulation can be used to evaluate extreme or unrealistic conditions which can provide insights in the system. On an operational decision level, CPs are used in a variety of patient scheduling applications ([Gartner and Padman 2017](#)). For example, [Gartner and Kolisch \(2014\)](#) use CPs to schedule elective patients hospital-wide on scarce resources. Their work uses elective patients' CPs as an input parameter in their models and assume that once an elective patient contacts the hospital, the pathways are fixed. However, our work bridges strategic decisions with sequential pattern mining for both, elective and non-elective patients.

## 2.3 Hospital layout planning

In general, layout planning aims at arranging organizational units inside a building such that the available area is used optimally and total distances are minimized. Most layout planning applications arise in industrial environments. When optimizing manufacturing facilities, the most common objective is to minimize traffic areas and traveled distances for produced goods. Thus, reliable information about movements of goods during the production process is needed. Hospital layout planning is typically located on a strategic decision level. Problems are reviewed, among others, in [Arnolds and Nickel \(2015\)](#) where [Elshafei \(1977\)](#) is most relevant for our work because of their travel distance minimization objective and model formulation as a quadratic assignment problem. However, we not only consider the planning of new hospitals but also the reconstruction and improvement of existing ones. We allow for an existing hospital to fix facilities at their location by fixing variables. Furthermore, we add constraints which bound the maximum travel distances between specialties.

## 2.4 Estimation of rare event probabilities and sequential pattern mining

Estimating the probability of rare CPs has similarities to estimating rare event probabilities. [Bachoc et al. \(2015\)](#) approached the latter problem by adapting the Hastings–Metropolis algorithm on Markov chains. [Guyader et al. \(2011\)](#) estimate the tail probability given quantiles

or the other way around to predict quantiles using a tail probability. Similarly to [Guyader et al. \(2011\)](#), they extend the Hastings–Metropolis algorithm. Both approaches are different to our study because, based on the similarity of patient flows, we merge states and rare events that occurred in our data can be filtered out. Also, rare events which have not been observed yet but might occur in the future can be assigned a probability.

Sequential pattern mining discovers frequent sub-sequences as patterns in a sequence database and a taxonomy of different algorithms is provided by [Mabroukeh and Ezeife \(2010\)](#). [Carrasco and Oncina \(1994\)](#) described an algorithm which is based on generating a prefix tree and then state-merging is carried out using a similarity measure. [Jacquemont et al. \(2009\)](#) extended this algorithm. Similarly, [Herbst and Karagiannis \(1998\)](#) use a Bayesian model merging approach for the induction of Hidden Markov Models. The difference between ([Herbst and Karagiannis 1998](#); [Carrasco and Oncina 1994](#); [Jacquemont et al. 2009](#)) lies in particular in the state merging process where [Jacquemont et al. \(2009\)](#) employ a statistical view of it which we follow in our work. Moreover, we incorporate blacklisting into the merging which is not addressed in any of the discussed works.

In conclusion, the approaches proposed in this paper can be categorized into and differentiated from the literature as follows: First, we select and implement a machine learning approach in which clinical activities and their relations between each other are learned from data. Second, we provide a mathematical model that incorporates this information. Finally, we present evaluation criteria based on cross-validation in order to evaluate the machine learning approach in combination with layout planning. The sequential pattern mining can cope with forbidden state merges so that e.g. expert knowledge can be taken into account by introducing a blacklist. Thus, we are able to incorporate a decision maker's opinion into the layout planning procedure which is especially important when reconfiguring a layout as our study will reveal in Sect. 5.

### 3 Methods

In order to detect significant CPs, we evaluate a sequential pattern mining algorithm devised by [Carrasco and Oncina \(1994\)](#) which is extended by [Jacquemont et al. \(2009\)](#). The rationale to select this algorithm from the literature on sequential pattern mining is because the learning approach has only a weak representational bias ([van der Aalst 2011](#)) on the process model ([Weber 2014](#)). One explanation is that we focus on the hospital-wide specialty flow and therefore, we can neglect parallel bookings on different specialties. The algorithm learns a probabilistic deterministic finite state automaton (PDFA) which is, under certain conditions, equivalent to learning a hidden Markov model (see [Dupont et al. 2005](#)). For standard textbooks covering automata theory we refer to [Hopcroft et al. \(2007\)](#).

#### 3.1 Learning a probabilistic deterministic finite state automaton (PDFA)

The algorithm first learns a probabilistic prefix-tree acceptor and afterwards, states are merged recursively by using a similarity measure.

*Probabilistic Prefix Tree Acceptor (PPTA)* In a PPTA which can be drawn as a graph, states are represented by a circle. In each circle, we write the index of the state and, after a colon, the probability to be final. States which have a probability greater than zero to be final are double-circled. The initial state is labeled by a “start” arc pointing it. Each transition between the predecessor and the successor state is represented by an arc. The label on each transition

consists of the transition symbol and the corresponding transition probability in parentheses. The transition symbol is, in our healthcare application, the clinical procedure which leads from one state to another. The proportion of sequences coming from the previous state to the following state is the probability of this transition. Using training sequences e.g. from Table 1a in Sect. 4, a PPTA can be constructed which is shown in Fig. 1 in Sect. 4.

*Probabilistic Deterministic Finite State Automaton (PDFA)* PDFAs are a generalization of PPTAs and we can formally describe them as follows: Let  $Q$  denote a finite set of states with state  $q_i \in Q \forall i \in \mathbb{N}_{\geq 0}$ . Let  $q_0 \in Q$  denote the initial state. Let  $\Sigma$  be an alphabet in which letters are denoted by  $z \in \Sigma$  and  $z = \#$  denotes the termination letter of a sequence. In our study, a letter corresponds to a ‘clinical activity’. The term is used as a synonym with ‘clinical procedure’. A clinical activity is performed on a facility (such as an operating room, imaging device or ward). Let  $q(q_i, z) \rightarrow q_j$  be an injective transition function leading to state  $q_j$  from state  $q_i$  with letter  $z$ . Specifically,  $q_i$  is equal to  $q_j$  when we have a loop on state  $q_i$  and letter  $z$ . Let  $\pi(q_i, z) \in [0, 1]$  be a probability function on the transitions and let  $\pi_F(q_i) \in [0, 1]$  be a function that assigns to each state a probability to be final. Then,  $A := (Q, \Sigma, q(q_i, z), q_0, \pi(q_i, z), \pi_F(q_i))$  is a tuple that defines our PDFA.

### 3.2 State merging

In order to avoid an overfitting phenomenon, the algorithm to build a PDFA based on a PPTA merges states. This means that states are chosen in a lexicographical order and if they are sufficiently similar, according to a compatibility function, they are merged. This function recursively tests if the frequencies of each letter outgoing from the two considered states are not statistically different. Based on Hoeffding’s bound (see [Hoeffding 1963](#)), this test decides that two states  $q_1$  and  $q_2$  can be merged if the condition described by Eq. (1) holds true. Here,  $\alpha^{\text{aut}}$  represents a generalization parameter while  $n(q_1)$  and  $n(q_2)$  are the number of sequences entering in  $q_1$  and  $q_2$ , respectively.

$$|\pi(q_1, z) - \pi(q_2, z)| < \sqrt{\frac{1}{2} \ln \frac{2}{\alpha^{\text{aut}}} \cdot \left( \frac{1}{\sqrt{n(q_1)}} + \frac{1}{\sqrt{n(q_2)}} \right)} \quad (1)$$

*Blacklisting-enhanced state merging* In order to avoid sequences to be generated that are from a medical point of view irrelevant, we check each time when we merge two states whether the incoming letter to and the outgoing letter from that new state are reasonable. To avoid that two letters forbiddingly follow each other, we introduce a set  $\mathcal{B}$  (blacklist) of forbidden tuples of letters, denoted by  $(z_i, z_j) \in \mathcal{B}$ .

### 3.3 Improving hospital layout planning through clinical pathway mining

In addition to material transportation costs that arise in industrial applications the most significant characteristic in a service environment of a hospital are the distances traveled by patients: Long travel distances do neither support the healing process nor patient satisfaction. To minimize total distances, accurate movement probabilities have to be determined which we will evaluate by combining the PDFA with a layout planning problem which will be introduced next.

#### 3.3.1 Problem description

We extend the well-known quadratic assignment problem by the possibility to fix specialties to locations as well as taking into account maximum distances between specialties. The formal

description of our hospital layout problem reads as follows: Let  $\mathcal{S}$  denote the set of specialties and let  $\mathcal{L}$  denote possible locations in a hospital in which specialties can be located. Let  $f_{i,k}$  be the transition frequency between specialty  $i \in \mathcal{S}$  and specialty  $k \in \mathcal{S}$ . Here, a specialty does not only refer to specialty departments such as the radiology department but also to facilities such as the operating theater. Furthermore, let  $d_{j,l}$  denote the distance between location  $j \in \mathcal{L}$  and location  $l \in \mathcal{L}$ . Let  $\mathcal{C}$  denote a set of specialty tuples and let  $\bar{D}_{i,k}$  be the maximum distance allowed between two specialties  $(i, k) \in \mathcal{C}$ , for example, between the surgery room and the ICU. Let  $\mathcal{W}$  denote a whitelist which is a set of tuples  $(i, j) \in \mathcal{W}$  representing specialty  $i \in \mathcal{S}$  and location  $j \in \mathcal{L}$ . Especially, when specialties such as the emergency department must be located in a defined area as for example near the entrance or when a hospital evaluates reorganization this is an important feature as we will learn in our experimental study.

### 3.3.2 Model formulation

Using the binary variables

$$x_{i,j} = \begin{cases} 1, & \text{if specialty } i \in \mathcal{S} \text{ is assigned to location } j \in \mathcal{L} \\ 0, & \text{otherwise} \end{cases}$$

we model the hospital layout problem as follows:

$$\text{minimize } \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{L}} \sum_{k \in \mathcal{S}} \sum_{l \in \mathcal{L}} f_{i,k} \cdot d_{j,l} \cdot x_{i,j} \cdot x_{k,l} \tag{2}$$

subject to

$$\sum_{j \in \mathcal{L}} x_{i,j} = 1 \quad \forall i \in \mathcal{S} \tag{3}$$

$$\sum_{i \in \mathcal{S}} x_{i,j} = 1 \quad \forall j \in \mathcal{L} \tag{4}$$

$$x_{i,j} = 1 \quad \forall (i, j) \in \mathcal{W} \tag{5}$$

$$\sum_{j \in \mathcal{L}} \sum_{l \in \mathcal{L}} d_{j,l} \cdot x_{i,j} \cdot x_{k,l} \leq \bar{D}_{i,k} \quad \forall (i, k) \in \mathcal{C} \tag{6}$$

$$x_{i,j} \in \{0, 1\} \quad \forall i \in \mathcal{S}, j \in \mathcal{L} \tag{7}$$

Objective function (2) minimizes the traveling distances and Constraints (3) ensure that each specialty is assigned to exactly one location. Constraints (4) ensure that each location contains exactly one specialty. Constraints (5) ensure that a specialty is fixed at its desired location while Constraints (6) ensure that maximum distances between two specialties are not exceeded. Variable definitions and the domains are provided by (7). The presented model is non-linear but can be linearized by introducing additional continuous variables (Xia and Yuan 2006).

### 3.4 Evaluation methods and metrics

The clinical pathways and the layouts can be evaluated using different methods and metrics which are introduced in the following.

### 3.4.1 Mean absolute deviation (MAD) between clinical pathway probabilities

A trivial baseline approach is to split the transactional data into a training and testing set. For each pathway, the relative frequency is computed. The absolute differences in probabilities between the pathway from the training set and the testing set are then calculated and averaged. Appendix refapp:corssval provides implementation details of this method in Java.

Another approach to calculate the MAD between pathway probabilities is to calculate the pathway probabilities by using the PDFA obtained from the learning set. The absolute differences in probabilities between the PDFA approach and the testing set are then calculated and averaged.

### 3.4.2 Significant clinical pathways

Given a maximum length  $l$ , we can enumerate pathways having length  $1, 2, \dots, l$ . For example, using the alphabet  $\Sigma = \{A, B, C\}$  and maximum length  $l = 5$ , we can enumerate pathways from  $A, AA, AAA, \dots, CCCCCB, CCCCC$ . However, only a fraction of these pathways are actually significant. To check whether a pathway is significant, we introduce the percentile  $\alpha^{\text{sig}}$ , the  $z$ -value from the standard normal distribution  $z_\alpha$  and the sample size  $N$  in the training sample. Given CP  $w$ , its probability  $p(w)$ , the sample size from the training set  $N$  and the  $z$ -value from the standard normal distribution  $z_\alpha$ , we obtain a threshold  $k$  using the following equation (see Jacquemont et al. 2009):

$$k = z_\alpha \cdot \sqrt{\frac{p(w) \cdot (1 - p(w))}{N}}. \quad (8)$$

If the probability  $p(w)$  of pathway  $w$  is larger than its threshold  $k$ , the pathway can be considered significant.

### 3.4.3 Error of the layout planning problem (ELPP)

In order to demonstrate the effectiveness of the automaton approach for layout planning, we can incorporate significant clinical pathways into the layout planning problem as formulated in models (2)–(7). Based on the assignment of specialties to locations determined by the mathematical program, we calculate the walking distances using the test data. We finally compare them with the distances obtained using perfect information. In doing so, we assume that both the training and test data are known. We denote this measure as ELPP.

### 3.4.4 Cross-validation

Rather than using one training and one test set of CPs, we can carry out cross-validation experiments (Bishop 2006; Witten and Frank 2011). Suppose, we have a set of folds  $\mathcal{F}$  with consecutive integer numbers running from  $1, 2, \dots, |\mathcal{F}|$ , for example,  $|\mathcal{F}| = 10$  in the case of 10-fold cross-validation. In this case, the set of folds comes up to  $\mathcal{F} := \{1, 2, \dots, 10\}$  where each fold  $f \in \mathcal{F}$  contains disjoint subsets of the observed transactional data. We index unique CPs by indices  $p \in \mathcal{P}$ . Let  $\pi_{p,f}^{\text{train}}$  be the probability of each unique pathway  $p \in \mathcal{P}$  for the training fold  $f \in \mathcal{F}$  and let  $\pi_{p,f}^{\text{test}}$  be the probability of pathway  $p \in \mathcal{P}$  for the test fold  $f \in \mathcal{F}$ . Then, using Algorithm 1 we determine the MAD across the  $|\mathcal{F}|$  folds.

Similarly, we determine the ELPP across folds  $\mathcal{F}$  using Algorithm 2.

**Algorithm 1:** Pseudocode for calculating the MAD in the cross-validation experiments based on folds  $\mathcal{F}$  and unique pathways  $\mathcal{P}$

```

1 for  $f \in \mathcal{F}$  do
2   for  $p \in \mathcal{P}$  do
3     Determine  $\pi_{p,f}^{\text{train}}$  and  $\pi_{p,f}^{\text{test}}$ 
4  $MAD = \frac{\sum_{f \in \mathcal{F}} \sum_{p \in \mathcal{P}} |\pi_{p,f}^{\text{train}} - \pi_{p,f}^{\text{test}}|}{|\mathcal{F}| \cdot |\mathcal{P}|}$ 
    
```

**Algorithm 2:** Pseudocode for calculating the ELPP in the cross-validation experiments based on folds  $\mathcal{F}$  and unique pathways  $\mathcal{P}$

```

1 for  $f \in \mathcal{F}$  do
2   for  $p \in \mathcal{P}$  do
3     Determine  $\pi_{p,f}^{\text{train}}$  and  $\pi_{p,f}^{\text{test}}$ 
4   Calculate layout  $\mathcal{X}_f$  and  $ELPP_f$ 
5  $ELPP = \frac{\sum_{f \in \mathcal{F}} ELPP_f}{|\mathcal{F}|}$ 
    
```

Once we have learned the probability of each significant pathway  $p \in \mathcal{P}$ , we split them into a set of tuples  $(i, j) \in \mathcal{E}$  which we denote as edges where  $i$  and  $j$  represent letters i.e. clinical activities in the pathways or facilities to be visited. Based on the layout  $\mathcal{X}_f$  learned in fold  $f \in \mathcal{F}$ , we can now compute the walking distances based on the distribution of pathways in the test set of pathways.

Based on the MAD and ELPP measures, we can determine confidence intervals. In our experimental study, we use the paired corrected  $t$ -test (Nadeau and Bengio 2001) implemented in the Java-based WEKA machine learning library from Witten and Frank (2011).

### 4 Hospital-wide layout planning under uncertain clinical pathways: an example

The following example illustrates the approach for learning significant CPs. We will demonstrate the effectiveness of the automaton approach by using two different values for the generalization parameter  $\alpha^{\text{aut}}$ . Furthermore, we show how we feed the result into the layout planning problem.

Assume, we have an alphabet  $\Sigma = \{A, B, C\}$  which represents the clinical procedures “radiotherapy procedure”, “diagnostic procedure”, and “surgical procedure” encoded by the letters  $A, B$  and  $C$ , respectively. Table 1a shows a sample set of 10 training sequences which will be used to learn the trivial and the automaton approach. The 10 testing sequences shown in Table 1b will later be used to evaluate the approaches.

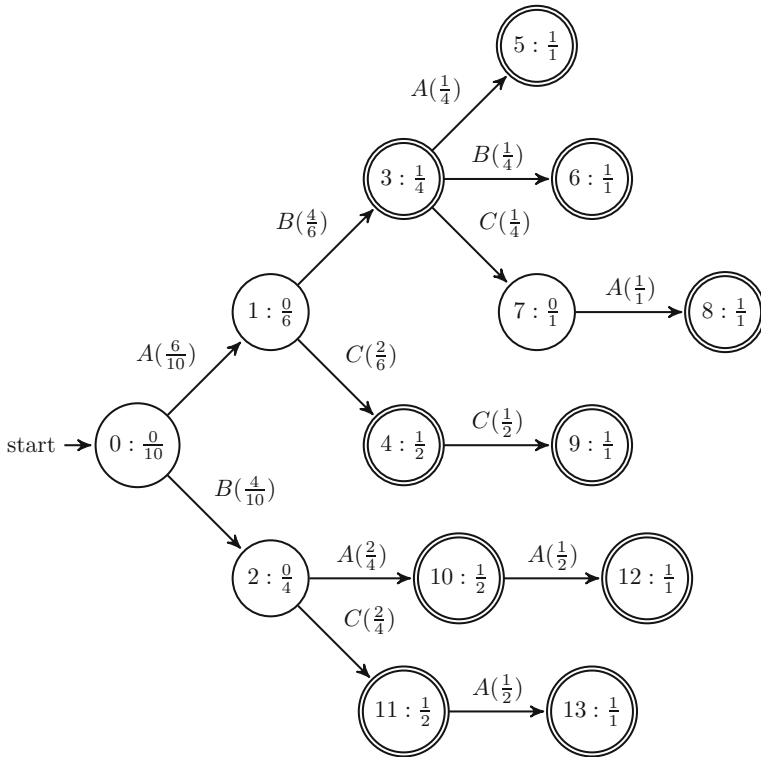
#### 4.1 Learning a probabilistic deterministic finite state automaton (PDFA)

To construct the PDFA, we first have to build a PPTA as explained in Sect. 3.1. In our example dataset shown in Table 1, we observe 10 sequences, of which 6 and 4 sequences start with letter  $A$  and  $B$ , respectively. Hence, we branch after the root state 0 to states 1 and



**Table 1** Sample of 20 sequences broken down by a training (a) and a testing (b) subsample

(a)				
<i>AB</i>	<i>ABA</i>	<i>ABB</i>	<i>ABCA</i>	<i>AC</i>
<i>ACC</i>	<i>BA</i>	<i>BAA</i>	<i>BC</i>	<i>BCA</i>
(b)				
<i>BCA</i>	<i>BCA</i>	<i>ABCA</i>	<i>ABCA</i>	<i>AAC</i>
<i>BAAC</i>	<i>CBAA</i>	<i>CB</i>	<i>CBA</i>	<i>BCAA</i>



**Fig. 1** PPTA corresponding to the sequences of Table 1(a)

2. Branching after these states is similar. For example, of the 4 sequences which start with a *B*, 2 sequences have an *A* as second letter leading to state 10 with probability  $\pi(2, A) = \frac{2}{4}$ . Then, sequence *BA* terminates at state 10 with a 50% chance. The final PPTA is shown in Fig. 1.

Now, if we evaluate the probability of, for example, pathway *ABB*, we start at state 0 and reach state 1 by the first letter *A*. The state is reached by 6 of 10 sequences and as a consequence the transition probability is  $\pi(0, A) = \frac{6}{10}$ . Using the second letter *B* as transition symbol, we reach state 3 with transition probability  $\pi(1, B) = \frac{4}{6}$ . Now, the third letter *B* of the sample pathway leads to state 6 with transition probability  $\pi(3, B) = \frac{1}{4}$ . Since our pathway has no more letters, we terminate at this state by probability  $\pi_F = 1$ . We can now compute the probability of the pathway *ABB* by the product of the transition

probabilities times the acceptance probability of the state reached after the last transition.

$$\pi^{ABB} = \frac{6}{10} \cdot \frac{4}{6} \cdot \frac{1}{4} \cdot \frac{1}{1} = \frac{24}{240} = \frac{1}{10}.$$

### 4.2 State merging with $\alpha^{\text{aut}} = 0.2$

We first test whether states 0 and 1 can be merged. Plugging the  $\alpha^{\text{aut}}$  parameter and the frequencies into the Hoeffding’s bound (see Inequality (1)) yields the following expression:

$$\sqrt{\frac{1}{2} \ln \left( \frac{2}{\alpha^{\text{aut}}} \right)} \cdot \left( \frac{1}{\sqrt{n(0)}} + \frac{1}{\sqrt{n(2)}} \right) = \sqrt{\frac{1}{2} \ln \left( \frac{2}{0.2} \right)} \cdot \left( \frac{1}{\sqrt{10}} + \frac{1}{\sqrt{6}} \right) = 0.777.$$

For each letter  $z$  outgoing from states 0 and 1, we now calculate the absolute difference of probabilities  $|\pi(0, z) - \pi(1, z)|$ . For  $z = A$ , the difference comes up to  $|\pi(0, A) - \pi(1, A)| = \left| \frac{6}{10} - 0 \right| =$

$$0.600 < 0.777, \text{ for } z = B \text{ it is } |\pi(0, B) - \pi(1, B)| = \left| \frac{4}{10} - \frac{4}{6} \right| = 0.267 < 0.777,$$

for  $z = C$  it is  $|\pi(0, C) - \pi(1, C)| = \left| 0 - \frac{2}{6} \right| = 0.333 < 0.777$  and for  $z = \#$  it is

$$|\pi(0, \#) - \pi(1, \#)| = |0 - 0| = 0 < 0.777.$$

Accordingly, states 0 and 1 can be merged which is shown in Fig. 2a. One can observe, this automaton is non-deterministic because the letter  $B$  leaves the initial state twice. As a consequence, we have to check whether states 2 and 3 can be merged. Thus, we calculate the Hoeffding’s bound as follows:

$$\sqrt{\frac{1}{2} \ln \left( \frac{2}{\alpha^{\text{aut}}} \right)} \cdot \left( \frac{1}{\sqrt{n(2)}} + \frac{1}{\sqrt{n(3)}} \right) = \sqrt{\frac{1}{2} \ln \left( \frac{2}{0.2} \right)} \cdot \left( \frac{1}{\sqrt{4}} + \frac{1}{\sqrt{4}} \right) = 1.07.$$

This bound is greater than 1 and since the differences of probabilities (left-hand side of Inequality (1)) can never be greater than 1, we can merge these two states. Now, merging states 2 and 3 again yields to a non-deterministic automaton because the newly merged state now has an  $A$  transition to states 5 and 10 as well as a  $C$  transition to states 7 and 11. Again, we check whether we can merge those states and after further tests and merging procedures to make the automaton deterministic, we obtain the PDFA shown in Fig. 2b. If in any of those succeeding tests the Hoeffding’s bound wasn’t fulfilled, then the original two states 0 and 1 could not be merged.

Now, the probabilities for the new states and transitions have to be calculated. The transition probabilities after the merge of states 0 and 1 are:

$$\begin{aligned} \pi(0, A) &= \frac{n(0,A) + n(1,A)}{\sum_{z' \in \Sigma \cup \{\#\}} n(0,z') + \sum_{z' \in \Sigma \cup \{\#\}} n(1,z')} = \frac{6+0}{10+6} = \frac{6}{16}, \\ \pi(0, B) &= \frac{n(0,B) + n(1,B)}{\sum_{z' \in \Sigma \cup \{\#\}} n(0,z') + \sum_{z' \in \Sigma \cup \{\#\}} n(1,z')} = \frac{4+4}{10+6} = \frac{8}{16} \text{ and} \\ \pi(0, C) &= \frac{n(0,C) + n(1,C)}{\sum_{z' \in \Sigma \cup \{\#\}} n(0,z') + \sum_{z' \in \Sigma \cup \{\#\}} n(1,z')} = \frac{0+2}{10+6} = \frac{2}{16}. \end{aligned}$$

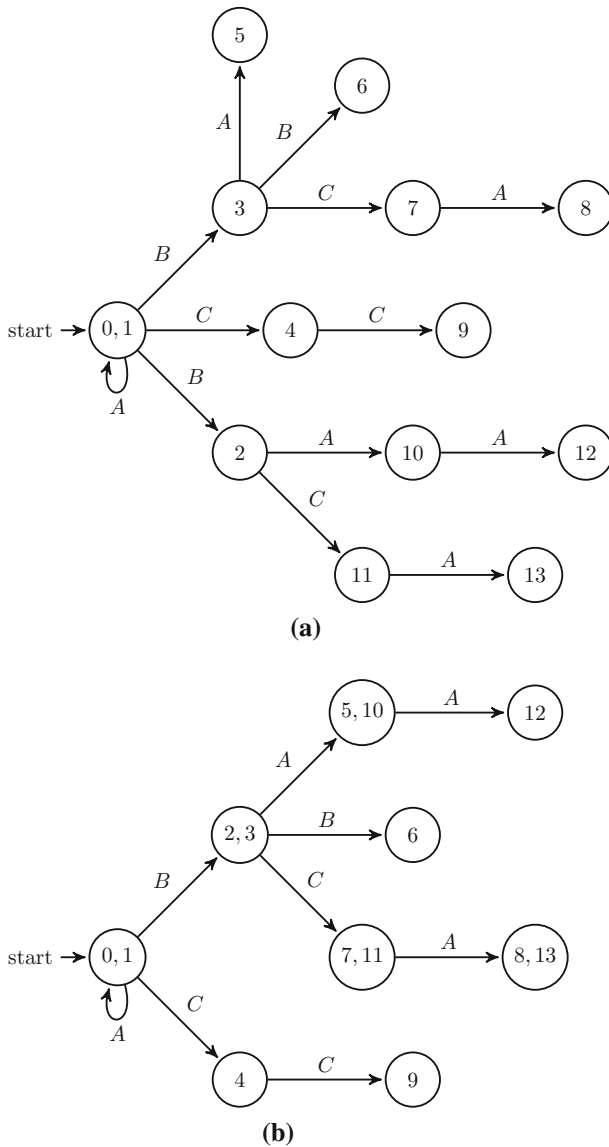
The new probability of state 0 to be final is:

$$\pi_F(0) = \pi(q(0, \#)) = \frac{n(0,\#) + n(1,\#)}{\sum_{z' \in \Sigma \cup \{\#\}} n(0,z') + \sum_{z' \in \Sigma \cup \{\#\}} n(1,z')} = \frac{0+0}{10+6} = 0.$$

For simplicity, we recalculate only those transition probabilities which have changed as follows:  $\pi(2, A) = \frac{2+1}{4+4} = \frac{3}{8}, \pi(2, B) = \frac{0+1}{4+4} = \frac{1}{8}, \pi(2, C) = \frac{2+1}{4+4} = \frac{3}{8}, \pi(2, \#) = \frac{0+1}{4+4} = \frac{1}{8}, \pi(10, A) = \frac{0+1}{1+2} = \frac{1}{3}, \pi(10, \#) = \frac{1+1}{1+2} = \frac{2}{3}, \pi(11, A) = \frac{1+1}{1+2} = \frac{2}{3}, \pi(11, \#) = \frac{0+1}{1+2} = \frac{1}{3}$  and  $\pi(13, \#) = \frac{1+1}{1+1} = \frac{2}{2}$ . The result is the PDFA as shown in Fig. 3a.

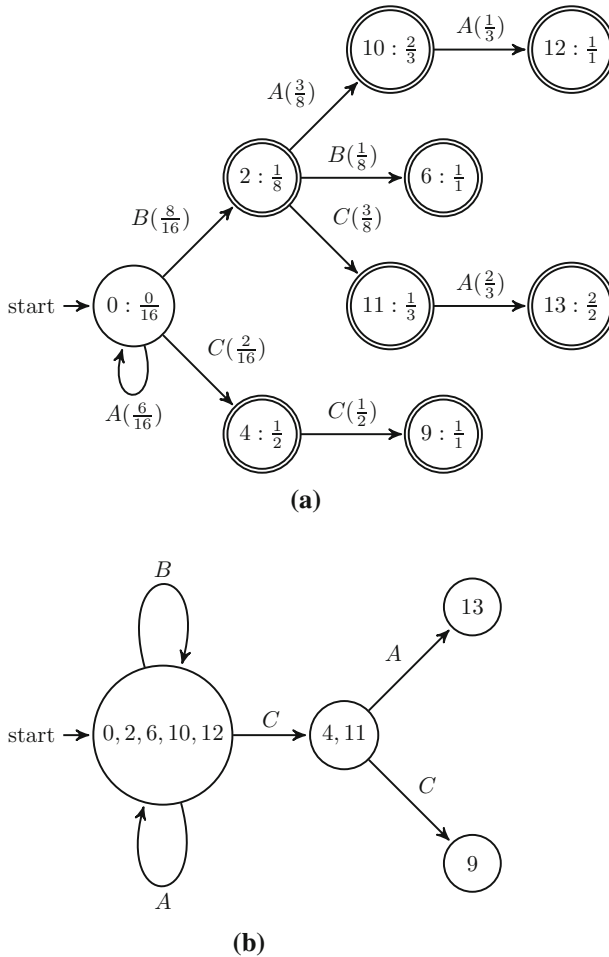
We now check whether we can merge states 0 and 2. Accordingly, the Hoeffding’s bound is calculated as follows:  $\sqrt{\frac{1}{2} \ln \left( \frac{2}{0.2} \right)} \cdot \left( \frac{1}{\sqrt{16}} + \frac{1}{\sqrt{8}} \right) = 0.916$ . Again, for each letter we

$$\text{calculate the difference of frequencies which are for } z = A : \left| \pi(q(0, A)) - \pi(q(2, A)) \right| =$$



**Fig. 2** First (a) and second (b) part of the merge of states 0 and 1 without updating numbers and probabilities

$\left| \frac{6}{16} - \frac{3}{8} \right| = 0 < 0.916$ , for  $z = B : \left| \pi(q(0, B)) - \pi(q(2, B)) \right| = \left| \frac{8}{16} - \frac{1}{8} \right| = 0.375 < 0.916$ , for  $z = C : \left| \pi(q(0, C)) - \pi(q(2, C)) \right| = \left| \frac{2}{16} - \frac{3}{8} \right| = 0.250 < 0.916$  and for  $z = \# : \left| \pi(0, \#) - \pi(2, \#) \right| = \left| 0 - \frac{1}{8} \right| = 0.125 < 0.916$ . Now having this precondition that states 0 and 2 can be merged, we check whether nodes 0 and 10, 0 and 12 and 0 and 6 are compatible and can be merged, too. Moreover, we check whether or not states 4 and 11 can



**Fig. 3** PDFA with updated probabilities after first merge (a) and without updated probabilities after merging states 0 and 2 (b)

be merged. Observe that merging states 0 and 2 yields into a non-deterministic automaton because then, the letter  $A$  goes from the new state 0 to state 10 while the letter  $A$  also remains at state 0. Therefore, we not only have to pull state 2 but also states 6, 10 and 12 into the newly merged state 0 which is shown in Fig. 2b. For simplicity we skip the explanation of the remaining merging steps, the result is the final PDFA as shown in Fig. 4a. The state merging with  $\alpha = 0.9$  is shown in Appendix 1 and the final PDFA is shown in Fig. 4b.

### 4.3 Improving layout planning through clinical pathway mining

Now, assume we have the following distance matrix between the locations  $d_{i,j} = ((0, 50, 200), (50, 0, 50), (200, 50, 0))$ . We solve the hospital layout planning problem with the probability distributions from the trivial and the automaton approach using the training sample of pathways. The results are shown in Sect. 4.4.3.

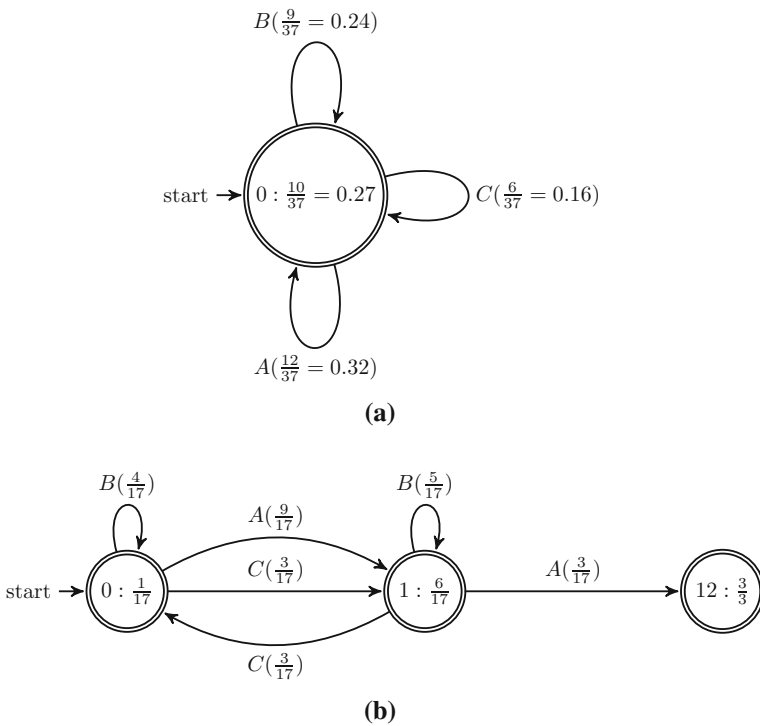


Fig. 4 Final PDFAs with  $\alpha^{aut} = 0.2$  (a) and  $\alpha^{aut} = 0.9$  (b)

### 4.4 Evaluation results

Comparing the two PDFAs that were generated above underlines that  $\alpha^{aut}$  has a substantial influence on the merging process and thus on the generalization of the original data. Using  $\alpha^{aut} = 0.2$ , a very generalized PDFa with only node 0 is left while using  $\alpha^{aut} = 0.9$  leads to a less generalized automaton with 3 from the original set of 14 states. Another observation is that all sequences from Table 1(a) are represented by both PDFAs. In addition, the automaton for  $\alpha^{aut} = 0.2$  can represent the sequence  $CBA A$  which is not included in the learning data, see Table 1(a). However, that sequence is not represented by the automaton’s language for  $\alpha^{aut} = 0.9$ , see Fig. 4b. The reason for this difference in the generalization of the original data is that  $\alpha^{aut}$  appears in the denominator of the Hoeffding’s bound calculation. The bigger the value of  $\alpha^{aut}$  the lower the threshold and, as a consequence, the lower the generalization. On the contrary, a small value of  $\alpha^{aut}$  leads to a more general automaton.

#### 4.4.1 Mean absolute deviation (MAD) between clinical pathway probabilities

From the CPs of Table 1, we observe in total 16 unique pathways. The probability distributions of these pathways using the trivial and the automaton approach as well as the MADs are shown in Table 2.

The figures reveal that the trivial approach fails to estimate probabilities of pathways which are not in the training set. Using the automaton approach, however, the pathway  $CB$  which is not in the training set receives a probability of 0.011 and 0.018 for  $\alpha^{aut} = 0.2$  and  $\alpha^{aut} = 0.9$ , respectively. The pathway  $CBA A$  which is failed to be discovered by the automaton approach

**Table 2** Pathway probability distribution for the three approaches for the training and test data

i	AB	ABA	ABB	ABCA	AC	ACC	BA	BAA
<i>(a) Pathways 1 to 8</i>								
$\pi_i^{trivial}$	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
$\pi_{i,0.2}^{PDFA}$	0.021*	0.007	0.005	0.001	0.014	0.002	0.021*	0.007
$\pi_{i,0.9}^{PDFA}$	0.055*	0.027*	0.016	0.005	0.005	0.006	0.044*	0.022*
$\pi_i^{test}$	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.0
BC	BCA	AAC	BAAC	CBAA	CB	CBA	BCAA	MAD
<i>(b) Pathways 9 to 16 and mean absolute deviations of probabilities</i>								
0.1	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.1
0.011	0.003	0.005	0.001	0.001	0.011	0.003	0.001	<b>0.066</b>
0.015	0.007	0.000	0.000	0.000	0.018	0.009	0.000	0.072
0.0	0.2	0.1	0.1	0.1	0.1	0.1	0.1	

The automaton approach with the best performance figure is in bold, significant CPs ( $\alpha^{sig} = 0.33$ ) are marked with an asterisk(\*)

with  $\alpha^{aut} = 0.9$ , receives a probability of 0.001 for  $\alpha^{aut} = 0.2$ , similar to the ones of *BAAC* and *AAC*.

#### 4.4.2 Significant clinical pathways

To compute significant pathways, we decide to use a large  $\alpha^{sig} = 0.33$  because using a small one would give us no significant CP at all. We get  $z_\alpha = 0.440$  while we observed  $N = 10$  pathways from the training data (see Table 1(a)). For example in the case of  $w = ABA$  and the automaton with  $\alpha^{aut} = 0.9$ ,  $p(ABA) = 0.027$ , the threshold comes up to  $k = 0.440 \cdot \sqrt{\frac{0.027 \cdot (1 - 0.027)}{10}} = 0.0226$ . Since this is smaller than the probability of the pathway (which was 0.027), the pathway is significant. Significant CPs are flagged by an asterisk in Table 2.

#### 4.4.3 Error of the layout planning problem (ELPP)

We compute the walking distances in the optimal solution which means that we determine the hospital layout using the testing sample of pathways. The ELPP is then the difference between the walking distance from the perfect information problem and the one obtained with the trivial or automaton approach. The layouts and the ELPP are shown in Table 3.

### 5 Experimental investigation

In the following, we provide an experimental investigation of the presented methods. We first give a description how we generated the sequences (CPs) followed by an overview of the hospital and its distances between different locations. Afterwards, our evaluation metrics are introduced followed by a presentation of the results.

**Table 3** Layouts obtained with the trivial, the automata and the perfect information approaches

Approach	Location			ELPP
	1	2	3	
<i>(a)</i>				
Trivial	B	C	A	450
PDFA $\alpha^{aut} = 0.9$	B	C	A	450
PDFA $\alpha^{aut} = 0.2$	A	B	C	<b>300</b>
Perfect information	C	A	B	0

The best performance figure is in bold

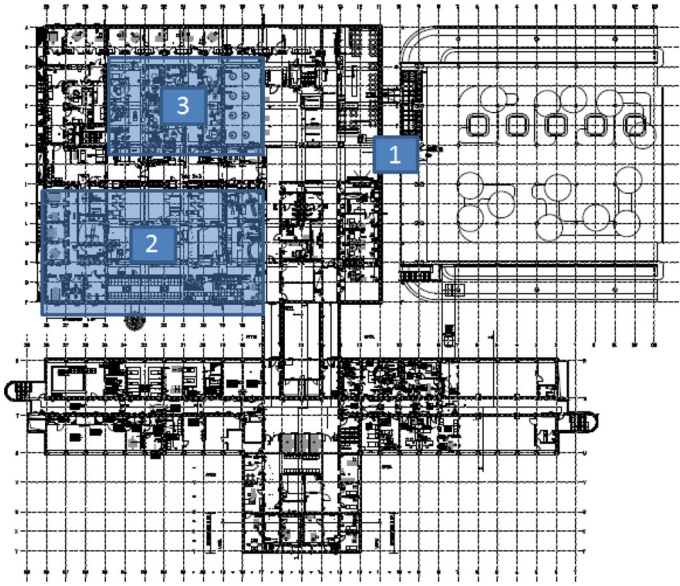
**Table 4** Overview of the alphabet that corresponds to the different specialties, functional units as well as entrance and exit

<i>A</i>	Internal medicine
<i>B</i>	Surgery department
<i>C</i>	Urology department
<i>D</i>	Gynecology department
<i>E</i>	ENT department
<i>F</i>	Orthopedics department
<i>G</i>	Intensive-care
<i>H</i>	Ophthalmology department
<i>I</i>	Radiology department
<i>J</i>	Operating theater
<i>N</i>	Functional diagnostics
<i>X</i>	Entrance and exit

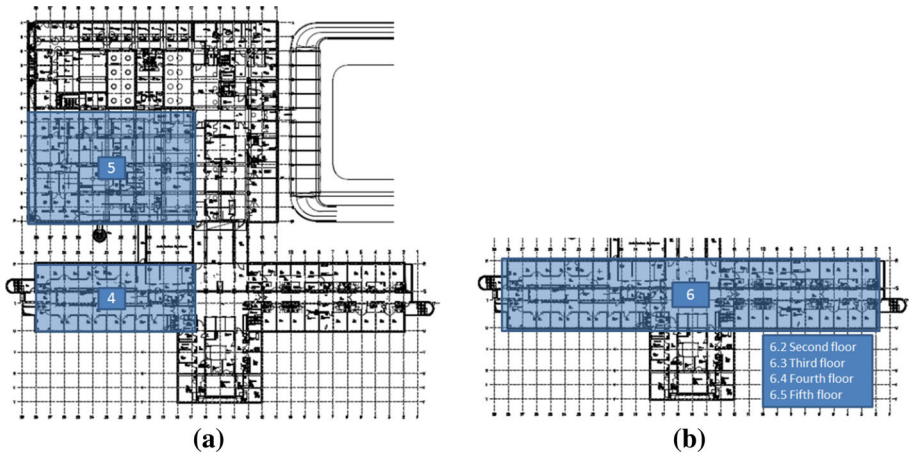
### 5.1 Data and sequence generation

We tested the sequential pattern mining and layout planning approach experimentally on data from a 350-bed sized hospital in Germany. Similarities between the U.S. healthcare system and other developed-world countries are that the data was collected for the billing of diagnosis-related groups (DRGs). As a consequence, we expect a similar data quality in other DRG systems such as U.S. and developed-world countries that employ DRG systems. We extracted 15,858 CPs from the hospital information system which corresponds to the same number of patients observed in the year 2011.

We generated sequences using a Java routine which accesses a MySQL database that contains three tables: A master table with DRG and demographic patient information, a table containing timestamps and clinical procedures coded by the International Classification of Procedures in Medicine (ICPM) and a table which contains timestamps when the patient was admitted and discharged from each specialty. We joined the three tables by patient IDs, sorted the result by patient ID and timestamp and relabeled the ICPM code or specialty code with a unique letter. We finally concatenated the letters of each patient to get the sequences. Table 4 provides an overview of our alphabet. For example, the sequence *XBNBJGBX* represents a patient who enters the hospital and is admitted to the surgical ward. Afterwards, he receives a cardiovascular check in the functional diagnostics unit before he gets back to the surgical ward. Next, he receives a surgery and is admitted to the ICU. Then, he recovers at the surgical ward and leaves the hospital at its exit.



**Fig. 5** Entrance and Exit (1), Radiology (2) and Functional Diagnostics (3)



**Fig. 6** Current layout of the collaborating hospital **a** ICU (4) and operating theaters (5). **b** Ward layouts of the second to fifth floor

## 5.2 Layout of the collaborating hospital

Figures 5 and 6 provide a ground plot of the collaborating hospital's current layout.

The entrance, exit, functional and radiology diagnostic units shown in Fig. 5 are located on the lower level where the offices of the administrative staff are shown on the lower part of the picture. The ICU and the operating rooms are located on the first floor of the building and shown in Fig. 6a. Wards are located on the second to fifth floor and are shown in Fig. 6b.



### 5.3 Distance matrix generation

To run our experiments, we set up transfer time matrix (9). The order of the columns/rows represent the order of the letters, see Table 4. For example, the first column/row represents the current position of the internal medicine department while the second column/row represents the current position of the surgery department and so on. The fifth floor has three positions in which the urology, ear, nose, throat (ENT) and ophthalmology department are located. As a consequence, distances are zero between these locations, see column/row 3.

The rationale behind using transfer times instead of distances is because elevators exist in the hospital which are typically used when patients are transported. The data reflects an average working day. Each entry represents the time required in seconds from location  $j$  to location  $l$ .

$$d_{j,l} = \begin{pmatrix} 0 & 18.79 & 23.93 & 26.51 & 23.93 & 18.79 & 18.79 & 23.93 & 72.48 & 67.34 & 86.28 & 109.28 \\ 18.79 & 0 & 18.79 & 23.93 & 18.79 & 0 & 23.93 & 18.79 & 75.06 & 72.48 & 88.86 & 111.86 \\ 23.93 & 18.79 & 0 & 18.79 & 0 & 18.79 & 26.51 & 0 & 78.08 & 75.06 & 91.88 & 114.88 \\ 26.51 & 23.93 & 18.79 & 0 & 18.79 & 23.93 & 29.53 & 18.79 & 83.24 & 78.08 & 97.04 & 120.04 \\ 23.93 & 18.79 & 0 & 18.79 & 0 & 18.79 & 26.51 & 0 & 78.08 & 75.06 & 91.88 & 114.88 \\ 18.79 & 0 & 18.79 & 23.93 & 18.79 & 0 & 23.93 & 18.79 & 75.06 & 72.48 & 88.86 & 111.86 \\ 18.79 & 23.93 & 26.51 & 29.53 & 26.51 & 23.93 & 0 & 26.51 & 67.34 & 48.55 & 81.14 & 104.14 \\ 23.93 & 18.79 & 0 & 18.79 & 0 & 18.79 & 26.51 & 0 & 78.08 & 75.06 & 91.88 & 114.88 \\ 72.48 & 75.06 & 78.08 & 83.24 & 78.08 & 75.06 & 67.34 & 78.08 & 0 & 115.89 & 13.8 & 36.8 \\ 67.43 & 72.48 & 75.06 & 78.08 & 75.06 & 72.48 & 48.55 & 75.06 & 115.89 & 0 & 129.69 & 152.69 \\ 86.28 & 88.86 & 91.88 & 97.04 & 91.88 & 88.86 & 81.14 & 91.88 & 13.8 & 129.69 & 0 & 50.6 \\ 109.28 & 111.86 & 114.88 & 120.04 & 114.88 & 111.86 & 104.14 & 114.88 & 36.8 & 152.69 & 50.6 & 0 \end{pmatrix} \tag{9}$$

### 5.4 Evaluation metrics

The sequential pattern mining approach is assessed using two evaluation metrics: Mean absolute deviation (MAD) and the error based on the layout planning problem (ELPP). The MAD is assessed by calculating the mean absolute difference between the probability distribution of significant CPs using the automaton approach and the actual probability distribution. We also assess a trivial baseline approach which uses the prior probability distribution of CPs.

### 5.5 Results

All computations were performed on a 2.4 GHz PC (Intel Core i7 4700MQ) with 32 GB RAM running a Windows 7 operating system. The mathematical model was coded in Java in an ILOG Concert environment. The solver used was ILOG CPLEX 12.6 (64 bit). We implemented the sequential pattern mining approach in Java, too.

We now compare the performance of the approaches broken down by MAD and ELPP and provide a comparison of the layout of our collaborating hospital with the layout that minimizes the ELPP based on the optimal parameter combination found by varying  $\alpha^{sig}$  and  $\alpha^{aut}$ . Finally, we show the results of our discussion of the solution with the hospital.

#### 5.5.1 Cross-validation results

Table 5 shows our 2-fold cross-validation experiments in which we varied the generalization parameter  $\alpha^{aut}$ .

**Table 5** Overview of the results of the cross-validation with different  $\alpha$ 

	$\alpha^{\text{aut}}$	MAD	ELPP
	0.001	0.083*	147.222*
	0.005	0.083*	147.222*
	0.01	0.083*	147.222*
	0.05	0.083*	147.222*
	0.1	0.083*	147.222*
	0.2	0.083*	147.222*
	0.3	0.083*	147.222*
	0.4	0.083*	147.222*
	0.5	0.088*	235.722
	0.6	0.088*	235.722
	0.7	0.091*	203.833
	0.8	0.097*	183.722*
	0.9	0.086*	148.722*
	1.0	0.111*	229.833
	Trivial approach	0.120	200.970

Significant improvements as compared to the trivial approach are highlighted with an \* (at 5% confidence level)

The table reveals that for the cross-validation experiments the lowest MAD is 0.083 and the lowest ELPP is 147.222. The figures also show that the MAD increases with increasing  $\alpha^{\text{aut}}$  which leads to the hypothesis that an over-generalized automaton overestimates some pathway probabilities.

### 5.5.2 Results of the MAD and ELPP metrics

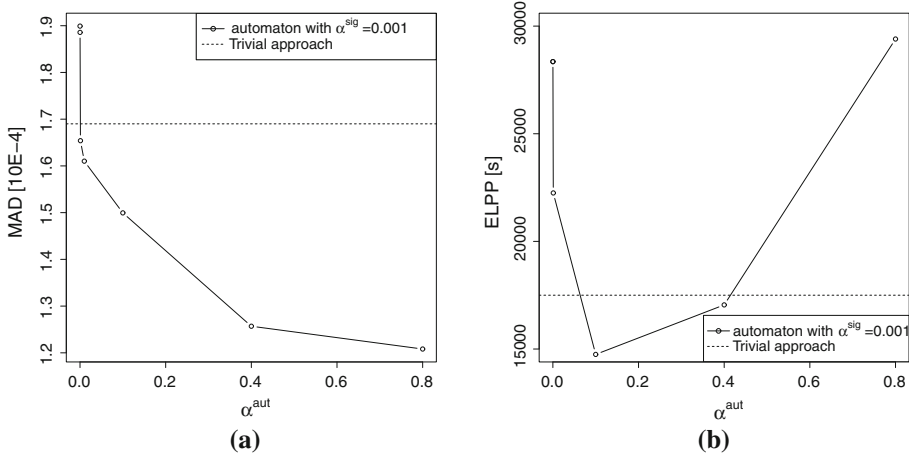
Figure 7a, b show the MAD and ELPP results, respectively. We varied  $\alpha^{\text{aut}}$  while we fixed  $\alpha^{\text{sig}} = 0.001$ . In addition, we restricted the maximum length of each CP to 5.

The results using MAD as metric show that using a small  $\alpha^{\text{aut}}$  value outperforms the trivial approach. More precisely, at a level of  $\alpha^{\text{aut}} = 0.001$  the MAD becomes approximately  $1.65\text{E}-4$  which is lower than the MAD of the trivial approach which is  $1.69\text{E}-4$ . Another observation is that the slope remains negative but it becomes more and more flat until  $\alpha^{\text{aut}} = 0.8$  is reached.

Similarly, the ELPP results show a descent of the error for small  $\alpha^{\text{aut}}$ . A more detailed analysis reveals that at  $\alpha^{\text{aut}} = 0.1$  the trivial approach which has an ELPP=17,500 is outperformed. However, with  $\alpha^{\text{aut}} > 0.4$ , the ELPP of the automaton approach again becomes worse than the trivial approach.

### 5.5.3 Evaluation of the hospital layouts

In order to see how the specialties would optimally be located we will use the whole sample data and evaluate the trivial as well as the automaton approach to calculate the probabilities and solve the layout planning problem. The sample data is now perfect information, since it represents the actual CPs observed at the collaborating hospital. Furthermore, we fix the entrance and exit at their original location. Table 6 shows the current hospital layout as well as the layouts obtained by the trivial and the automaton approach under perfect information.



**Fig. 7** MAD (a) and ELPP (b) for the trivial and the automaton approaches

**Table 6** Comparison of the allocation results for the collaborating hospital

Department	Location		
	Original	Trivial	Automaton
A-Internal medicine	6.2	2	2
B-Surgery	6.3	6.3	6.3
C-Urology	6.4	6.4	6.4
D-Gynecology	6.5	3	3
E-ENT department	6.4	6.5	6.5
F-Orthopedics	6.3	6.4	6.2
G-Intensive-care	4	6.4	6.4
H-Ophthalmology	6.4	5	5
I-Radiology	2	6.2	6.4
J-Operating theater	5	6.3	6.3
N-Functional diagnostics	3	4	4
X-Entrance and exit	1	1	1

5.5.4 Fixing hospital specialties based on recommendations from the hospital

Once again, the entrance and exit have been fixed on their current location. The only specialties that remain on their original position according to both approaches are the surgery department and urology. Both approaches locate the operating theater next to the surgery department which is reasonable. In most cases the trivial and the automaton approach give the same recommendations. Only the radiology department and orthopedic department are interchanged.

Although both approaches provide similar results we have to reconsider the allocation with regard to practicability when changing the current layout. Departments as the operating theater and radiology department can hardly be moved to another location as there are lots of special machines that cannot easily be moved to another location. That is why we additionally fixed the intensive-care, radiology department, operating theater and functional diagnostics. The resulting allocations for the different approaches can be seen in Table 7.

**Table 7** Optimal allocation of specialties at the collaborating hospital with some fixed assignments

Department	Location		
	Original	Trivial	Automaton
A-Internal medicine	6.2	6.3	6.2
B-Surgery	6.3	6.2	6.3
C-Urology	6.4	6.4	6.4
D-Gynecology	6.5	6.3	6.3
E-ENT department	6.4	6.4	6.4
F-Orthopedics	6.3	6.4	6.4
G-Intensive-care	4	4	4
H-Ophthalmology	6.4	6.5	6.5
I-Radiology	2	2	2
J-Operating theater	5	5	5
N-Functional diagnostics	3	3	3
X-Entrance and exit	1	1	1

Having fixed some specialties, both approaches deliver the same result except for the internal medicine and the surgery department which are interchanged. They stay on their current position with the automaton approach. The urology department and ENT specialty are recommended to stay at floor 4. Departments that should be located in upper floors than before are the ophthalmology department from floor 4 to 5, the orthopedics department from floor 3 to 4 and internal-medicine, following the trivial approach, from floor 2 to 3. It is recommended that the gynecology department moves down from floor 5 to 3 and the surgery department regarding to the trivial approach from floor 3 to 2. A possible explanation for this setup may be that the gynecology department and surgery department should be located closer to the functional areas in the ground and first floor.

## 5.6 Discussion and generalizability of the results

### 5.6.1 Limitation of using transfer times

When setting up the transfer time matrix, we argue that patients use elevators to get from one specialty to another. This is true for patients that have to be transported, for example, from a ward to the operating theater. However, some patients may simply use the stairs for getting from one floor to another. Also, waiting times for an elevator may vary considerably during a working day. For example, there may be high traffic during breakfast, lunch and dinner times when food has to be transported to the specialties and back. Furthermore, times are depending on walking speeds which might be very different for different patient types or their transportation mode (walking, wheelchair and bed).

### 5.6.2 Generalizability of the results

The approaches presented in this paper enhance the current state of the art in literature on the strategic decision level in healthcare operations management. It links the work of [Cardoen and Demeulemeester \(2008\)](#) on the strategic decision level with clinical medical work on CP mining. From an operations management point of view, the sequential pattern mining approach presented could be used in a patient scheduling problem which is located on an operational decision level, see [Gartner and Kolisch \(2014\)](#).

### 5.6.3 Calibration of $\alpha^{aut}$

As could be seen in the example and the computational results, the automaton-based approach is sensitive to the  $\alpha^{aut}$  parameter which controls the state merging process. A parameter optimization of  $\alpha^{aut}$  should be carried out before applying the approach in practice. Otherwise, it can happen that a trivial approach outperforms the automaton-based pathway mining approach.

### 5.6.4 Applicability for existing and new hospitals

Our approach can be used for both applications: reorganization of existing and building of new hospitals. The difference is that for reorganizing a hospital, specialties would be fixed to locations (see Constraints (5)). This might also be necessary when planning a new hospital where for example the emergency room should be near to the entrance area on the ground floor. Fixing any specialty in advance leaves enough room for any other improvement (total travel distances or transfer times) that might be achieved by changing the location of the remaining specialties. However, the improvement potential may be reduced if some facilities are fixed to their locations rather than reorganizing or building the hospital without fixing variables.

## 6 Conclusion

In this paper, we have dovetailed clinical pathway (CP) mining with hospital-wide layout planning: First, we have selected and extended a machine learning approach to learn CPs from data. Then, we have presented a mathematical model for hospital layout planning which takes into account clinical pathways. It features not only the planning of new hospitals but also the reconfiguration of existing ones by partially fixing specialties to locations. We evaluated the approach in a cross-validation setting and have shown results based on different evaluation measures and level of detail. Depending on its generalization parameter, the chosen automaton approach outperformed a baseline approach significantly.

Future work will focus on parameter optimization. For example, a full factorial test design will be run to determine a (near optimal) generalization parameter  $\alpha^{aut}$ , paired with  $\alpha^{sig}$  which can filter out non-significant CPs. Alternatively, a heuristic search approach may be beneficial to determine both parameters.

Further extensions will be to evaluate patient types and transportation modes and trading off walking distances and transportation costs. Finally, we will test the applicability of our approach towards operational decisions such as hospital-wide patient scheduling.

**Acknowledgements** The authors sincerely thank the editor and the two anonymous referees for their careful review and excellent suggestions for improvement of this paper.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## Appendix 1: Abbreviations, sets, indices and decision variables

See Table 8.

**Table 8** Abbreviations, sets, parameters, indices and decision variables

Abbreviations	
CP	Clinical pathway
ELPP	Error of the layout planning problem
ENT	Ear nose and throat department
ICU	Intensive Care Unit
MAD	Mean absolute deviation
OECD	Organisation for Economic Co-operation and Development
PDFA	Probabilistic deterministic finite state automaton
PFSA	Probabilistic finite state automaton
PPTA	Probabilistic prefix tree acceptor
Sets and indices	
$\mathcal{B}$	Set of blacklisted tuples $(z_i, z_j) \in \mathcal{B}$ which refer to letters $z_i$ and $z_j$
$\mathcal{C}$	Set of tuples $(i, j) \in \mathcal{C}$ which refer to specialties $i, j \in \mathcal{S}$
$\mathcal{F}$	Set of folds in the cross-validation experiments
$\mathcal{L}$	Set of locations
$\mathcal{P}$	Set of unique CPs
$\mathcal{X}_f$	Layout solution for fold $f \in \mathcal{F}$
$\alpha^{\text{aut}}$	Generalization parameter
$\alpha^{\text{sig}}$	Significance level to for CPs
$(z_i, z_j) \in \mathcal{B}$	Set of blacklisted tuples containing letters $z_i$ and $z_j$
$\pi(q_i, z)$	Probability function on the transition of $q_i$ given $z \in \Sigma$
$\pi_F(q_i)$	Probability for state $q_i$ to be final
$f_{i,k}$	Transition frequency between specialty $i \in \mathcal{S}$ and specialty $k \in \mathcal{S}$
$d_{j,l}$	Distance between location $j \in \mathcal{L}$ and location $l \in \mathcal{L}$
$\overline{D}_{i,k}$	Maximum distance allowed between two specialties $(i, k) \in \mathcal{C}$
$(i, j) \in \mathcal{C}$	Set of tuples containing specialties $i, j \in \mathcal{S}$
$q_0 \in \mathcal{Q}$	Initial state
$q(q_i, z)$	Transition function leading to a state given state $q_i$ and letter $z \in \Sigma$
$\mathcal{Q}$	Finite set of states
$\Sigma$	Set of letters (alphabet)
$\mathcal{S}$	Set of specialties
$\mathcal{W}$	Whitelist of tuples $(i, j) \in \mathcal{W}$ which refer to specialty $i \in \mathcal{S}$ which is fixed on location $j \in \mathcal{L}$
Parameters and indices	
$\alpha^{\text{aut}}$	Generalization parameter
$\alpha^{\text{sig}}$	Significance level to for CPs
$d_{j,l}$	Distance between location $j \in \mathcal{L}$ and location $l \in \mathcal{L}$
$\overline{D}_{i,k}$	Maximum distance allowed between two specialties $(i, k) \in \mathcal{C}$
$f_{i,k}$	Transition frequency between specialty $i \in \mathcal{S}$ and specialty $k \in \mathcal{S}$
$\pi(q_i, z)$	Probability function on the transition of $q_i$ given $z \in \Sigma$
$\pi_F(q_i)$	Probability for state $q_i$ to be final

**Table 8** continued

$q_0 \in \mathcal{Q}$	Initial state
$q(q_i, z)$	Transition function leading to a state given state $q_i$ and letter $z \in \Sigma$
$(i, j) \in \mathcal{W}$	Whitelist of tuples where specialty $i \in \mathcal{S}$ must be fixed on location $j \in \mathcal{L}$
$\Sigma$	Alphabet
Decision variables	
$x_{i,j}$	1, if specialty $i \in \mathcal{S}$ is assigned to location $l \in \mathcal{L}$ , 0 otherwise

## Appendix 2: State merging with $\alpha^{\text{aut}} = 0.9$

In order to demonstrate the influence of a different  $\alpha^{\text{aut}}$  on the merging process, we now employ  $\alpha^{\text{aut}} = 0.9$  as parameter to calculate whether the PPTA's states from Fig. 1 can be merged. Using this  $\alpha^{\text{aut}}$ , we already see at the beginning that the states 0 and 1 cannot be merged because plugging  $\alpha^{\text{aut}} = 0.9$  and the frequencies into Eq. (1) yields  $\sqrt{\frac{1}{2} \ln\left(\frac{2}{\alpha^{\text{aut}}}\right)} \cdot \left(\frac{1}{\sqrt{n(0)}} + \frac{1}{\sqrt{n(1)}}\right) = \sqrt{\frac{1}{2} \ln\left(\frac{2}{0.9}\right)} \cdot \left(\frac{1}{\sqrt{10}} + \frac{1}{\sqrt{6}}\right) = 0.458$ . For  $z = A$  :  $|\pi(q(0, A)) - \pi(q(1, A))| = \left|\frac{6}{10} - 0\right| = 0.600 > 0.458$ . Consequently, states 0 and 1 cannot be merged with  $\alpha^{\text{aut}} = 0.9$ . However, states 0 and 2 can be merged. Again, we skip the explanation of the remaining merging steps and show the final PDFA in Fig. 4b.

## Appendix 3: Cross-validation using the WEKA Java API

The following code gives an overview of how we computed the MAD using cross-validation and the the WEKA Java API (Witten and Frank 2011). Let 'data' be an object of the WEKA class Instances.java and let nFolds be the number of crossvalidation folds, for example 10 in the case of 10-fold cross-validation.

```
double calcMADPathways() {
for(int n=0;n<nFolds;n++)
{
Instances train = data.trainCV(nFolds, n);
Instances test = data.testCV(nFolds, n);

for(int inst=0;inst<train.numInstances();inst++)
probDistributionTrain[n][((int) train.instance(inst).value(0))
+= (double) 1/train.numInstances();

for(int inst=0;inst<test.numInstances();inst++)
probDistributionTest[n][((int) test.instance(inst).value(0))
+= (double) 1/test.numInstances();

for(int pathway = 0;pathway<data.instance(0).attribute(0).numValues();pathway++)
error[n] += Math.abs(probDistributionTrain[n][pathway]
- probDistributionTest[n][pathway]);

MADperFold[n] = error[n]/data.numInstances();
SumMADs += MADperFold[n];
}
}
```

```

MAD =  $\sum \text{MADs} / n\text{Folds}$ ;

returnError = MAD;

return returnError;
}

```

## References

- Arnolds, I., & Nickel, S. (2015). Applications of Location Analysis, Springer, chap Layout Planning Problems in Health Care.
- Bachoc, F., Bachouch, A., & Lenôtre, L. (2015). Hastings–Metropolis algorithm on Markov chains for small-probability estimation. *ESAIM: Proceedings and Surveys*, 48, 276–307.
- Bishop, C. (2006). *Pattern recognition and machine learning*. New York: Springer.
- Blumenthal, D. (2009). Stimulating the adoption of health information technology. *New England Journal of Medicine*, 360(15), 1477–1479.
- Cardoen, B., & Demeulemeester, E. (2008). Capacity of clinical pathways: A strategic multi-level evaluation tool. *Journal of Medical Systems*, 32(6), 443–452.
- Carrasco, R. C., & Oncina, J. (1994). Learning stochastic regular grammars by means of a state merging method. In *Proceedings of the International Conference on Grammatical Inference* (pp. 139–152).
- Combi, C., Gozzi, M., Oliboni, B., & Juarez, J. (2009). Temporal similarity measures for querying clinical workflows. *Artificial Intelligence in Medicine*, 46(1), 37–54.
- Dupont, P., Denis, F., & Esposito, Y. (2005). Links between probabilistic automata and hidden Markov models: Probability distributions, learning models and induction algorithms. *Pattern Recognition*, 38(9), 1349–1371.
- Elshafei, A. N. (1977). Hospital layout as a quadratic assignment problem. *Operational Research Quarterly*, 28(1), 167–179.
- Gartner, D., & Padman, R. (2017). Handbook of research on healthcare administration and management, IGI global, chap mathematical programming and heuristics for patient scheduling in hospitals: a survey, (chap. 38), pp. 627–645.
- Gartner, D., & Kolisch, R. (2014). Scheduling the hospital-wide flow of elective patients. *European Journal of Operational Research*, 223(3), 689–699.
- Guyader, A., Hengartner, N., & Matzner-Løber, E. (2011). Simulation and estimation of extreme quantiles and extreme probabilities. *Applied Mathematics and Optimization*, 64(2), 171–196.
- Herbst, J., & Karagiannis, D. (1998). Integrating machine learning and workflow management to support acquisition and adaptation of workflow models. In *Database and expert systems applications, 1998. Proceedings. Ninth International Workshop on*, IEEE, pp. (745–752).
- Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301), 13–30.
- Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2007). *Introduction to automata theory, languages and computation* (3rd ed.). Reading: Addison Wesley.
- Huang, Z., Lu, X., & Duan, H. (2013). Similarity measuring between patient traces for clinical pathway analysis. In N. Peek, R. Marin Morales, & M. Peleg (Eds.), *Artificial intelligence in medicine* (pp. 268–272)., Lecture notes in computer science Berlin: Springer.
- Iwata, H., Hirano, S., & Tsumoto, S. (2013). Mining clinical pathway based on clustering and feature selection. In K. Imamura, S. Usui, T. Shirao, T. Kasamatsu, L. Schwabe, & N. Zhong (Eds.), *Brain and health informatics* (pp. 237–245)., lecture notes in computer science Berlin: Springer International Publishing.
- Jacquemont, S., Jacquenet, F., & Sebban, M. (2009). Mining probabilistic automata: A statistical view of sequential pattern mining. *Machine Learning*, 75(1), 91–127.
- Landro, L. (2014). A Cure for hospital design—strategies to keep patients and their visitors from getting lost. The Wall Street Journal.
- Mabroukeh, N. R., & Ezeife, C. I. (2010). A taxonomy of sequential pattern mining algorithms. *ACM Computing Surveys (CSUR)*, 43(1), 3.
- Mans, R. S., van der Aalst, W., & Vanwersch, R. J. (2015). *Process mining in healthcare: Evaluating and exploiting operational healthcare processes*. Berlin: Springer.
- Nadeau, C., & Bengio, Y. (2001). Inference for the generalization error. *Machine Learning*.



- Rojas, E., Munoz-Gama, J., Sepúlveda, M., & Capurro, D. (2016). Process mining in healthcare: A literature review. *Journal of Biomedical Informatics*, 61(6), 224–236.
- van de Klundert, J., Gorissen, P., & Zeemering, S. (2010). Measuring clinical pathway adherence. *Journal of Biomedical Informatics*, 43(6), 861–872.
- van der Aalst, W. M. (2011). On the representational bias in process mining. In: *2011 20th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, IEEE, (pp. 2–7).
- Weber, P. (2014). A framework for the analysis and comparison of process mining algorithms. Ph.D. thesis, University of Birmingham.
- Wennberg, J., Blowers, L., Parker, R., & Gittelsohn, A. (1977). Changes in tonsillectomy rates associated with feedback and review. *Pediatrics*, 59(6), 821–826.
- Witten, I., & Frank, E. (2011). *Data mining: Practical machine learning tools and techniques* (3rd ed.). San Francisco: Morgan Kaufmann.
- Xia, Y., & Yuan, Y. X. (2006). A new linearization method for quadratic assignment problems. *Optimization Methods and Software*, 21(5), 805–818.
- Zhang, Y., Padman, R., & Wasserman, L. (2014). On learning and visualizing practice-based clinical pathways for chronic kidney disease. In *Proceedings of AMIA Annual Symposium*.
- Zhang, Y., Padman, R., & Patel, N. (2015a). Paving the COWpath: Learning and visualizing clinical pathways from electronic health record data. *Journal of Biomedical Informatics*, 58(12), 186–197.
- Zhang, Y., Padman, R., Wasserman, L., Patel, N., Teredesai, P., & Xie, Q. (2015b). On clinical pathway discovery from electronic health record data. *IEEE Intelligent Systems*, 1, 70–75.