Journal of Biomedical Informatics 48 (2014) 94–105

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Learning Bayesian networks for clinical time series analysis

Maarten van der Heijden [a,b,*], Marina Velikova [a], Peter J.F. Lucas [a,c]

[a] Institute for Computing and Information Sciences, Radboud University Nijmegen, The Netherlands
[b] Department of Primary and Community Care, Radboud University Nijmegen Medical Centre, The Netherlands
[c] Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands

ARTICLE INFO

Article history:
Received 28 March 2013
Accepted 10 December 2013
Available online 18 December 2013

Keywords:
Chronic disease management
Bayesian networks
Machine learning
Temporal modelling
Clinical time series
Chronic obstructive pulmonary disease

ABSTRACT

Introduction: Autonomous chronic disease management requires models that are able to interpret time series data from patients. However, construction of such models by means of machine learning requires the availability of costly health-care data, often resulting in small samples. We analysed data from chronic obstructive pulmonary disease (COPD) patients with the goal of constructing a model to predict the occurrence of exacerbation events, i.e., episodes of decreased pulmonary health status.
Methods: Data from 10 COPD patients, gathered with our home monitoring system, were used for temporal Bayesian network learning, combined with bootstrapping methods for data analysis of small data samples. For comparison a temporal variant of augmented naive Bayes models and a temporal nodes Bayesian network (TNBN) were constructed. The performances of the methods were first tested with synthetic data. Subsequently, different COPD models were compared to each other using an external validation data set.
Results: The model learning methods are capable of finding good predictive models for our COPD data. Model averaging over models based on bootstrap replications is able to find a good balance between true and false positive rates on predicting COPD exacerbation events. Temporal naive Bayes offers an alternative that trades some performance for a reduction in computation time and easier interpretation.

1. Introduction

Clinical data often takes the form of time series, which, when interpreting all the variables concerned in their mutual context, offer a description of the progression of a disease over time. While insight into the evolution of a disease is an important aspect of the management of any disease, whether acute or chronic, for patients with a chronic disease the evolution is of even more importance, as often the disease will not disappear. In this context it is in particular important to be able to detect when the disease becomes worse, i.e., to detect and possibly prevent *exacerbations*. For any chronic disease it is therefore of clinical interest to study the interaction between different variables, ranging from signs and symptoms to environmental factors, in terms of both static and temporal relationships. If we can capture this knowledge in a model, predictions regarding health status made by means of such models can be used to assist in chronic disease management, for example by advising on therapy adjustment. Furthermore, disease models are important for epidemiological purposes, for example for survival analysis, as well as for cost-effectiveness analysis and policy planning.

A complication that arises when analysing clinical time series is that it is often hard to obtain sufficient data, for example because the event of interest is relatively low frequency or because taking measurements is costly, time consuming or inconvenient for the patient. In addition, the reality of gathering clinical data is that observations are made at irregular time intervals and the data will contain missing values. Patients will sometimes forget to provide data, or omit some evidence for unknown reasons. Also measuring devices may sometimes fail or readings may not be recorded. These observations pose a challenging research question, which we seek to answer in this paper, namely, whether we can learn useful predictive models from clinical data with the combined characteristics of missing values and limited availability.

The research methods we propose draw their inspiration from various existing methods, which have proven to be successful in machine learning applications. Yet the combination of these methods has not been applied to clinical time series analysis. Temporal variants of Bayesian networks are our main tools to reason about causal and temporal processes in a probabilistic manner. In particular we use dynamic Bayesian networks (DBNs), where 'dynamic' should be interpreted as modelling the temporal dynamics of the process. They provide interpretable and versatile models to

* Corresponding author at: Institute for Computing and Information Sciences, Radboud University Nijmegen, The Netherlands.
E-mail addresses: m.vanderheijden@cs.ru.nl (M. van der Heijden), peterl@cs.ru.nl (P.J.F. Lucas).

1532-0464/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jbi.2013.12.007

describe time series data and can be used to classify states and make predictions about future states. To learn network structures in the presence of missing observations we make use of the structural expectation–maximisation (EM) algorithm [1], which iteratively completes the data and performs a search for the best network structure to explain the data. Here we employ a variant of structural EM, with a different approach to filling in values for the missing data. Finally, to tackle the problem of data sparsity, we consider bootstrap methods originally developed in statistics [2]. In the context of Bayesian networks these methods have been applied to the analysis of micro-array data and we extend them here to the learning of temporal models from clinical time series.

To provide a concrete clinical context for this research, we focus on chronic obstructive pulmonary disease (COPD) as an application area. This disease has many characteristics that are typical for any chronic disease, although symptoms and signs will be mostly different from those other diseases. COPD is a major chronic disease in terms of morbidity and mortality; it affects the respiratory system, decreasing lung capacity and obstructing airways, thus interfering with normal breathing. An important aspect of COPD which is particularly relevant in the present context is the progressive nature of the disease. Specifically episodes of acute deterioration have a profound impact on patient well-being and on health-care costs [3]. These exacerbations are mainly caused by airway infections resulting in symptom worsening [4]. Important to note is also that patients with frequent exacerbations usually have faster disease progression, which makes exacerbation prevention a particularly relevant goal. Additionally, a faster treatment response to exacerbations appears to lead to better recovery [5].

The main contributions of the paper are as follows:

- We formulate an algorithm to learn temporal probabilistic models from limited clinical time series with missing values. The main novelty of the algorithm lies in combining learning of dynamic Bayesian networks from clinical data using structural EM with block bootstrapping for small data samples.
- We propose a variant of our learning algorithm based on naive Bayes networks, which has the attractive properties of reduced computational complexity, thus easy construction, while offering good prediction performance.
- The proposed learning algorithms are used to build predictive models of COPD patient's health status, focussing on day to day progress of signs and symptoms that can rapidly change during exacerbation events. These predictive models are novel in the context of COPD as they can handle both the dynamic nature and uncertainty inherent in the disease progression. As such, these models can be embedded within clinical or home-based applications for chronic disease management.
- We evaluate the learning procedure on COPD synthetic and patient data and show that it is effective in terms of structure discovery of interesting variable relationships, interpretability and prediction performance of the models learned.
- The results from this research demonstrate important clinical implications not only for the prediction of COPD exacerbations but also for the clinical relevance of the methods proposed for chronic disease management applications in general.

2. Related research

2.1. Clinical time series analysis

Survival analysis is a popular clinical application of time series analysis and here the technique of Cox-regression is normally used for model construction [6]. However, Cox-regression has important limitations; in particular, it is not suitable to model independence assumptions. Our work is more closely related to the research described in [7], which argues for using DBNs for prognostic models in medicine. Variants of DBNs have also been used to model temporal dynamics of organ failure in patients in intensive care units (ICU) [8], although there no structure learning was used. Further, a Bayesian network has been developed on the basis of electronic health record data to predict the onset of COPD in asthma patients; however, temporal information was not explicitly taken into account [9]. In the specific context of long term disease management for COPD, related research has focussed on facilitating remote communication [10,11] and automatic data interpretation is still uncommon. In [12] a telehealth system is described that has been applied to COPD and contains a decision support component. However, currently the decision support is limited to rule based detection of abnormal values and trend detection. Automatic interpretation of monitoring data using machine learning while taking time and uncertainty into account, therefore, appears to be a useful contribution to the area of chronic disease management.

2.2. Modelling and machine learning techniques

Early work on using Bayesian networks for prediction includes dynamic network models [13], with, for example, a clinical application to predicting sleep apnea [14]. In our work, we used and extended techniques for learning Bayesian networks from data with missing values [1] and learning from small samples using bootstrapping [15]. These methods are used extensively in bioinformatics [16], but application to the domain of clinical time series analysis constitutes a new and interesting challenge. The bootstrap methods used for small data samples are related to what is known as *bagging* in the machine learning literature [17], but are usually applied to learning decision trees instead of Bayesian networks. Our augmented temporal naive Bayes model is an extension of the TAN classifier from [18] to a prediction model that takes time dependencies into account.

In [19] a method is proposed to learn DBNs with changing dependency structures. The models include hidden variables that influence the structure of the model depending on the value of a particular variable that controls the structure change. This method is of interest when sufficient data is available to learn changing dependencies. Finally, in [20] an approach is described to use steady state information in addition to time series data to learn DBNs. Steady state data from the limiting distribution of the Markov chain describing the process is used as an additional source of information.

A different approach to modelling temporal processes is used in temporal nodes Bayesian networks (TNBNs), which model events instead of dynamics [21]. TNBNs are similar to Bayesian networks, but temporal nodes take time intervals as values. The intervals represent the time since a parent event. We made a TNBN model for our COPD domain to compare to the dynamic models, which will be discussed in Sections 5.3 and 6.3.

3. The Aerial project: mobile COPD management

The methods we developed and used in this paper were needed as part of a research project, called Aerial, aimed at the detection of worsening in patients with COPD, i.e., exacerbations. Here we briefly describe the system and the design of the study to sketch the practical clinical context of the work.

3.1. A system that supports self-management

In order to facilitate self-management of COPD by patients we developed a system with the capability to gather patient-specific

information, to interpret the gathered data automatically and to offer feedback and advice to the patient. The general architecture of the Aerial system is shown in Fig. 1. The system consists of a smartphone as the main component taking care of communication and computation. Questionnaire data is collected from the patient on the smartphone, which also communicates wirelessly with the sensors used to obtain objective information on the patient's health status. A web-based system allows scheduling tasks and collecting patient data centrally. The web-centre receives the data from the smartphone and provides data access for health-care workers. Patient data are interpreted in the smartphone by means of a disease-specific probabilistic model, different variants of which are studied in this paper. A more comprehensive description of the system can be found elsewhere [22].

3.2. Patient population and data gathering

The aim of the study was to investigate the prerequisites to facilitate self-management of COPD-patients in a home-care setting. We tested the Aerial system at home to see whether this would result in usable data with the goal of gathering a data set from which to learn models for COPD-exacerbation prediction. Ten participants were recruited from hospitals and general practices in the Netherlands, 7 male and 3 female, between 53 and 76 years of age (mean (sd): 65.6 (6.8)). All participants gave written informed consent. Inclusion criteria were GOLD II or III (disease severity classification on a scale from I to IV) [23] and sufficient cognitive capability to operate the system. There were two possible inclusion paths: stable patients, although having had exacerbations in the past, and patients that reported to a physician because of an exacerbation. The first path provides information on stable patients, and possibly on exacerbation onset; the second on exacerbation recovery and possibly relapse.

Patients were monitored daily for a duration of approximately 4 weeks. Each day they answered a set of binary questions about their symptoms and performed spirometry and pulse-oximetry measurements. All questions were formulated such that a 'yes' answer indicates a symptom worse than the baseline condition for that patient. The set of recorded variables consists of the symptoms dyspnea (D), sputum volume (SV), sputum purulence (SP), cough (C), wheeze (W), temperature (T), malaise (M) and activity (A); plus the measurements SpO_2 (SO) (blood oxygen saturation) and FEV_1 (F) (forced expiratory volume in 1 s). Exacerbations (E) were defined as at least two consecutive days of an increase in at least two major respiratory symptoms (dyspnea, sputum volume or sputum purulence) or one major and at least one minor respiratory symptom (see e.g. [24]). A total of 189 data records were collected, which when regularised by adding missing values for days that were not recorded resulted in a data set of 250 records. The reason for explicitly representing missing values is that the learning algorithms assume regularly spaced data. Missing data was partly a consequence of technical issues and partly due to patients omitting data for unknown reasons, resulting in 30% missing data. Of all the collected data records 60 were collected during an exacerbation.

4. Probabilistic models for temporal data analysis

As the research aim was to develop models that were able to interpret temporal clinical data, in particular revealing the progression of COPD including its stochastic variation, especially *probabilistic models* were considered to be attractive for that purpose. Probabilistic models can express temporal trends and handle missing values when data of a specific patient must be interpreted. The successful use of dynamic Bayesian networks in the context of micro-array analysis [16], where sparsity of data is also a problem, made us wonder whether similar methods might also be successfully applied to clinical problems.

We start with a brief summary of basic methods used in the research, leading up to a method to learn models from sparse clinical time series data.

4.1. Dynamic Bayesian networks

A *Bayesian network* [25,26], is a probabilistic graphical model represented as a pair $\mathrm{BN} = (G, P)$. Here, $G = (V, A)$ is a directed acyclic graph consisting of *vertices* V, corresponding one-to-one to random variables of interest, and $A \subseteq V \times V$ are *arcs*, representing probabilistic dependencies between variables. Furthermore, P is a joint probability distribution defined by a family of conditional probability distributions of the form $P(V \mid \mathrm{pa}(V))$, that is, the probability that V takes on a specific value given the values of its parent variables, $\mathrm{pa}(V)$. The network represents the joint distribution over the random variables, which can be factored according to the dependencies represented in the graph, resulting in:

$$P(V_1, V_2, \ldots, V_n) = \prod_{i=1}^{n} P(V_i \mid \mathrm{pa}(V_i)),$$

where $V_i \in V$ is the representation of a random variable in the graph G. Any probability of interest can be computed from this joint probability distribution.

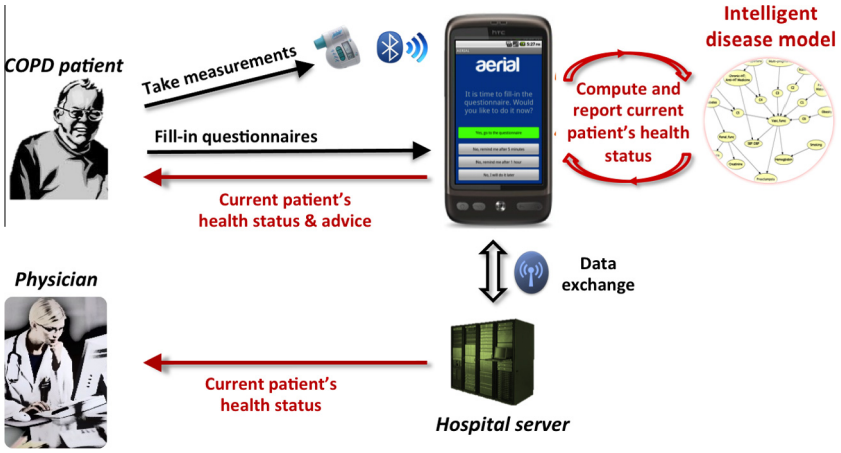

Fig. 1. Schematic of the system setup.

A *dynamic Bayesian network* [27,28], DBN for short, is an extension of a Bayesian network to a distribution over a sequence of random variables. It is particularly well suited to represent a Markov process

$$X_1 \to X_2 \to \cdots \to X_t \to \cdots$$

where X_t represents a random variable at a particular moment in time t. The joint distribution can be decomposed using the chain rule, writing $X_{1:T}$ for $X_1, \ldots, X_t, \ldots, X_T$:

$$P(X_{1:T}) = P(X_T \mid X_{1:T-1})P(X_{1:T-1}).$$

In general a DBN is a factorisation of a probability distribution, like its atemporal counterpart. In addition, when X_t is composite it can be represented as a BN. This BN is called a *time slice* and relations between time slices can be modelled by introducing arcs in the graph between random variables in different time slices. A DBN factorisation can then be written as:

$$P(X_{1:T}) = \prod_t \prod_i P(X_{t,i} \mid \mathrm{pa}(X_{t,i}))$$

where i indexes variables within a time slice and $\mathrm{pa}(X)$ denotes the parents of X in the graph. A hidden Markov model, a popular stochastic model used for pattern recognition, is a special case of a DBN.

A common assumption is that there is only a limited time frame that influences the current state of the process, as opposed to the complete history, which simplifies model learning. When assuming an nth-order Markov process we obtain:

$$P(X_{1:T}) = P(X_T \mid X_{T-n:T-1})P(X_{T-n:T-1}),$$

recursively. In the context of clinical data analysis, this assumption makes sense for two reasons: first, as time passes physiological and disease processes will change and older information will be less informative about the current state of the patient; second, it will often not be possible to obtain reliable information about the past. For smaller n more temporal independence is introduced and when only sparse data is available, it is common to make the most restrictive version of the Markov assumption, first-order, such that the future state of the process only depends on the present:

$$P(X_{t+1} \mid X_{1:t}) = P(X_{t+1} \mid X_t).$$

Hence, all parents of a variable X will be in the same time slice or in the previous time slice. From a medical point of view this means that the current health status provides the most information about the future. Given that clinical data is often sparse, an important practical consequence of this assumption is that it simplifies the model, and hence reduces the amount of data we need. When we now also assume that the process is *stationary*, that is $P(X_{t,i} \mid \mathrm{pa}(X_{t,i})) = P(X_{t',i} \mid \mathrm{pa}(X_{t',i}))$ for all t, t', we obtain a *two-slice DBN* consisting of an initial network BN_0 and a transition network $\mathrm{BN}_{\to}$. A process through time can now be modelled by a sequence of repetitions of transition networks. When modelling a chronic disease over a long period, it may be that stationarity is not a reasonable assumption. For COPD one might argue that it is also useful to consider separate models for the disease stages (GOLD I-IV [23]). However the COPD data we study here is limited to GOLD II and III by the inclusion criteria, and the amount of available data is too limited to make a distinction. Yet, in general, it appears useful to consider recent techniques to learn non-stationary DBNs [29].

4.2. Model learning

Given a data set we can use machine learning techniques to learn a model from the data. Two main tasks are usually distinguished: (i) finding a network structure, and (ii) finding the parameters that best describe the data given a network structure. See [30,31] for fairly comprehensive overviews of probabilistic learning.

4.2.1. Parameter learning

Parameter learning entails finding the optimal parameters $\hat{\theta}$ for a given network structure G, that explain the data D:

$$\hat{\theta} = \arg\max_\theta P(D; \theta, G).$$

This is the maximum likelihood estimate, parametrised by θ (the semicolon indicates that it is a parameter, not a conditional). The maximum likelihood estimate is straightforward to compute, but may suffer from overfitting; especially for sparse data, which is the case that we are considering here. A Bayesian approach makes explicit the uncertainty in the parameters by taking θ to be a random variable, leading to:

$$P(\theta \mid D, G) = P(D \mid \theta, G)P(\theta \mid G)/P(D \mid G),$$

from Bayes' theorem, where we compute a distribution over parameters, with some prior distribution $P(\theta \mid G)$. Assuming independent and identically distributed data, global and local parameter independence, the likelihood can be decomposed in local likelihood terms according to the graph:

$$P(D \mid \theta, G) = \prod_{d \in D} \prod_{i=1}^{n} P(X_i = k \mid \mathrm{pa}(X_i) = j, \theta).$$

Since we are using discrete variables in our COPD model, we use multinomial distributions for the local likelihood terms $P(X_i = k \mid \mathrm{pa}(X_i) = j, \theta)$. The conjugate prior distribution of the multinomial is a Dirichlet distribution, where conjugacy implies that the posterior is also a Dirichlet distribution. We can interpret the hyperparameters α of the Dirichlet distribution as pseudo-counts which leads to a parameter estimate

$$\theta_{ijk} = P(X_i = k \mid \mathrm{pa}(X_i) = j) = \frac{\mid D[k,j] \mid + \alpha[k,j]}{\mid D[j] \mid + \alpha[j]},$$

where $D[k,j]$ is an index expression selecting the data cases where $X_i = k$ and $\mathrm{pa}(X_i) = j$ and $D[j] = \sum_k D[k,j]$. In principle it is possible to specify different prior pseudo-counts for different cases, leading to analogous index expressions for α, in practice α is often constant.

Parameter estimation using likelihood decomposition properties works well for complete data, which is unrealistic for clinical models. The expectation–maximisation (EM) algorithm can be used to learn parameters when we have missing values [32].

Let $D = \langle O, H \rangle$ be a data set consisting of observed values O and hidden values H; O_i, H_i indicate a single data point. The EM algorithm iteratively adjusts the parameters in two steps:

$$\text{E-step}: \ \text{Complete } D \text{ according to } P(H_i \mid O_i, \theta^s)$$
$$\text{M-step}: \ \theta^{s+1} = \arg\max_\theta P(D \mid \theta)$$

where s is the step counter. The E-step completes the data by filling in values for the hidden variables given the current parameters (usually implemented by computing expected sufficient statistics); and the M-step computes the maximum likelihood parameters based on the completed data. This algorithm provably improves the likelihood in each step until some maximum is reached, although this could be a local maximum [32]. EM gives us no guarantee of finding a global maximum; however, its results are normally sufficiently good.

4.2.2. Structure learning

Hand-crafting a network structure is difficult and time consuming and hence it is useful to also try to learn the network structure from data. Even for a limited number of variables as we have in our case, the search space is of such a size that exhaustive search is

infeasible. Alternative methods include constraint based and score based techniques. See [33] for a comparison on a large set of hospital management data, where constraint based methods appear to perform somewhat better. However, an important disadvantage of constraint based methods, that rely on independence tests, is that early errors propagate and can have large effects. Additionally, independence tests tend to be less reliable especially on small samples, for example because distribution assumptions are not met. In [34] arguments for and against constraint based methods are examined and a hybrid approach is proposed, which is, however, not directly applicable to time series. We opt here for a standard score-based search procedure that tests local changes to the graph, greedily maximising the score $\log P(G \mid D)$, which by Bayes' rule can be written as

$$\log P(G \mid D) \propto \log P(D \mid G) + \log P(G).$$

Taking a Bayesian approach the probability $P(D \mid G)$ can be written as the integral

$$P(D \mid G) = \int_{\theta} P(D \mid \theta, G) f(\theta \mid G)\, \mathrm{d}\theta,$$

with $f(\theta \mid G)$ a probability density function, marginalising out the parameters. For complete data the likelihood again decomposes according to the graph and assuming conjugate priors there is a closed form solution for the marginal likelihood $P(D \mid G)$ (see for example [30]). A common choice for the structure prior $P(G)$ results in the BDe-score [35], which uses a prior such that we have score equivalence. That is, networks that encode the same independence information will have the same score.

Unfortunately, the task becomes increasingly difficult with missing data. Finding optimal parameters when we have missing values depends on the structure, while structure search clearly depends on the data, including the missing values. There is no particularly efficient solution to this problem, but we can use the same general principle as we did to learn parameters from partially observed data. This idea is called structural expectation maximisation [1]. In essence the structural EM algorithm alternates between completing the data based on the current structure and parameters and finding a model that has a better score. The same caveat as mentioned for the EM algorithm applies here as well, structural EM does not guarantee finding the global maximum likelihood solution. When the number of missing values increases the search space becomes larger and we are less likely to find a globally optimal model.

4.3. Small data samples

With the model learning techniques described above we can try to find models that reasonably fit our data, even when some of the data is missing. However, as we are learning statistical models there is a strong dependence on the amount of available data. The more data we have the better we can learn relations. Unfortunately, gathering data is often a difficult, time consuming process. For the COPD self-management application the demanding logistics of studying a new system in a home-care setting, made data collection hard. Furthermore, sometimes the model space is so large that obtaining sufficient data to learn models directly may be infeasible. Previous research in determining gene interaction patterns from micro-array data [15,16], is another example of research that stumbled across the mentioned problems. The clinical data we study here is, however, of a different nature than micro-array data, as signs and symptoms will clearly produce other temporal dynamics. Although the number of variables was fairly limited in our case, we had a combination of missing values within records and a limited number of records in total. Hence we needed to combine the EM procedure with some technique to deal with the small data sample.

4.3.1. Bootstrapping

The data sparsity in modelling gene data led to using bootstrapping [2] as a way to estimate the uncertainty of relations in Bayesian networks [15]. The idea is as follows: bootstrapping can be used as a nonparametric estimate of a statistic of interest, for which we can take the presence of arcs in the Bayesian network graph. The intuition behind bootstrap replications of a data set is that if we assume a generative model, the actual data we see are just a particular realisation which may be unrepresentative of the underlying process. Bootstrapping a number, say m, of new realisations by resampling (with replacement) $\mid D \mid$ samples from the original data and learning models from those data sets, we can estimate whether an arc a should be present in our model:

$$P(a) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{I}(a, G_i),$$

where $\mathbb{I}(a, G_i)$ is an indicator function that is 1 when a is present in the graph learned from bootstrap realisation i.

Since we are analysing temporal data, the situation is somewhat more difficult. Simply resampling from the original data will discard all the ordering information, which is clearly undesirable. Instead we have to apply a kind of bootstrapping that preserves the correlation between different time points. In the statistics literature a number of methods have been developed to do so, and we opted to use fixed-length block bootstrapping. As the name suggests this entails resampling blocks of data points, which preserves the time relations within the block. The *moving block bootstrap* splits the data in $\mid D \mid - l + 1$ blocks, where l is the block length. Block i contains the data points from i to $i + l - 1$. A bootstrap time series is the concatenation of $\mid D \mid / l$ sampled blocks. See e.g. [36] for a theoretical comparison of block bootstrap methods.

From the bootstrap results we can construct a model that includes the features (arcs) that have a high probability. Another possibility is to use model averaging, where we compute the prediction probability on validation data D_{val} as the average over the predictions of the bootstrap models

$$P(D_{\text{val}}) = \frac{1}{m} \sum_{i} P(D_{\text{val}} \mid G_i). \tag{1}$$

However, each of the networks G_i has some score attached to it, so a better approximation to the real probability can be obtained from the weighted average

$$P(D_{\text{val}}) = \frac{1}{m} \sum_{i} P(D_{\text{val}} \mid G_i) P(G_i \mid D_i), \tag{2}$$

where D_i indicates the ith bootstrapped data set.

4.4. A method to learn temporal models from small data samples with missing values

With all the machinery described above, we can now start putting together the method to learn models from time series. The main idea is a variant of structural EM applied to bootstrap resampled data. The new features of this procedure are that it combines sparse data techniques with structure learning on small samples of time series data, which is typical for clinical data.

Algorithm 1.

1 Construct m data sets $D_{1:m}$ from the original data D using block bootstrapping;
2 **foreach** $D_i \quad i \in \{1, \ldots, m\}$ **do**
3 **for** $j = 0, 1, 2, \ldots$ *until convergence or reaching a time limit* **do**
4 Learn parameters θ_j for the DBN structure G_j with EM;
5 Complete the data D_i by sampling from the posterior $P(H \mid O, \theta_j)$;
6 Search for a DBN structure G_{j+1} on the completed data that improves $P(G_j \mid D_i)$;
7 **end**
8 **end**
9 Compute the model average predictions

This algorithm generalises the bootstrap method for sparse data [15], to temporal models. In order to learn the structure of a temporal model, we need a generalisation of the structural EM algorithm [38]. In each iteration we learn the structure of a two-slice network consisting of BN_0 and $BN_{\rightarrow}$. There are, however, some differences with the algorithm in [38]. In particular, we take a different approach to completing the data. On line 5 of our algorithm, we sample values from the posterior distribution $P(H \mid O, \theta_j)$ instead of computing fractional sufficient statistics. That is we sample values for the missing data H from the posterior given the observed values O and the current parameters θ_j. This allows us to decouple the parameter learning and the structure search. The result is a complete data set without (fractional) expected values filled in for the missing values, which lets us use structure search methods developed for complete data. Additionally, sampling from the posterior $P(H \mid O, \theta_j)$ has the interesting property that it results in automatic small data perturbations. Because structural EM is not guaranteed to find a global maximum, the data perturbations help in starting the search from slightly different departure points. The posterior sampling provides a way to let the search reach different parts of the search space. In [39] some of the differences between hard-assignment, EM (soft-assignment) and posterior assignment are studied in the context of clustering, which is closely related to filling-in missing values.

4.4.1. Augmented temporal naive Bayes

Although a full model is useful to gain insight into the domain, for our application it is important that the performance on predicting exacerbations is good. Therefore, it appears useful to emphasise this goal during model construction. To do so we start from the concept of a naive Bayes classifier, with *exacerbation*, E, as class variable. The rationale is that naive Bayes is a good baseline classifier that can be extended with the information obtained from structure learning. The idea of modelling dependencies between naive Bayes feature variables has been used to construct tree augmented naive Bayes (TAN) classifiers [18]. We here propose an augmented temporal naive Bayes classifier. The presence of arcs from our class variable *exacerbation* to each of the other variables in the same time slice is enforced, ensuring the naive Bayes structure. Structure search can then identify dependencies between the other variables, also through time. We retain the inner loop of Algorithm 1 to find the dependencies that best explain the data, but instead of bootstrapping we restrict the model search space by predefining part of the network structure. The acyclicity of the graph implies that $X_t \rightarrow E_t$ is not present, whereas we need to explicitly blacklist arcs of the type $X_t \rightarrow E_{t+1}$ for all X. The structure search will identify temporal dependencies but one could argue that since we are interested in the temporal behaviour of the class variable that we should also enforce the presence of the arc $E_t \rightarrow E_{t+1}$. It turns out that in this specific case of a COPD model, when the arc is not enforced, it is found by the search procedure.

5. Experimental setup

In the previous section we described a general method to learn DBNs from small sample time series data. Here we describe the experimental evaluation of the methods for the clinical problem of predicting exacerbations of COPD. The experiments serve to evaluate, first, whether the models that result from the learning procedure are accurate; second, to explore what models can be learned from the COPD data. For the first goal we use synthetic data from a synthetic model, described below. The second goal can be further refined, as we are interested in how the learned models compare to the model constructed in cooperation with a pulmonologist; and we want to find out what the performance of these models is on predicting exacerbation events.

5.1. Data

5.1.1. Synthetic data

To test our model learning procedure we first study the results with synthetic data from a known model. The model contains the same variables as the Aerial data (Section 3) to stay as close as possible to the real data context. The arcs were not chosen to be clinically correct; however, to make the results transferable to the real data the model resembles a clinical model. Analogous to the other models both atemporal and temporal arcs are present. The structure of the model is shown in Fig. 3a. The parameters were assigned by hand, without attempting to capture the real relations. From this model we generated four data sets of the same length as the Aerial data, one without missing values and the others with 10%, 20% and 30% missing values respectively. The latter is similar to the percentage of missing values in the Aerial data. We then applied our model learning procedure to each of these data sets in

order to be able to compare the learned models to the known source model.

5.1.2. Aerial data

The Aerial data consists of time series from 10 patients, gathered during the pilot study of our disease management system. The characteristics of this data set have been described in Section 3 as part of sketching the background for this work. The data set is used as the input for the learning procedure, to learn models that explain the COPD-monitoring time series.

5.1.3. Validation data

An independent data set [42] was used for validation. It provides time series of COPD-exacerbation related variables, used solely for external validation, so no model parameters were learned from this set. As the data is used for testing our model retrospectively, the set of variables is incomplete and contains 8 out of the 11 variables in the Aerial data. The data consists of time series from 13 patients of the London COPD cohort who had an exacerbation, with a total of 2849 data entries, of which 406 were during an exacerbation. The data contains values for the variables dyspnea, sputum volume and purulence, wheeze, cough, temperature, and oxygen saturation (SpO_2). We consider two variants of this data set, the complete validation set, denoted by D_{val}, and a deduplicated version D_{dedup}. The latter consists of the same data, but with consecutive identical entries removed, resulting in 605 entries, of which 128 during an exacerbation. The idea behind removing duplicates is that we are interested in predicting relevant state changes instead of finding models that are often correct by predicting that nothing changes. By removing repeating sequences we ensure that we make predictions with data that frequently changes state. Note that we only remove exact duplicates, so a change in the observations without a change in exacerbation label still constitutes a change.

5.2. Evaluation metrics

To evaluate the models and their performance we will use standard metrics from classification. Classification evaluation is often based on measuring true positives (TP), cases that are correctly classified as positive; false positives (FP), incorrectly classified as positive; and analogously for true/false negatives (TN; FN). The true positive rate (TPR) is then defined as $TP/(TP + FN)$ and the false positive rate (FPR) as $FP/(FP + TN)$. Plotting a curve of TPR-FPR for different cut-off points results in an ROC-curve, which is often summarised in a single number by computing the area under the curve or AUC.

We distinguish two situations: (i) the performance of network structure discovery and (ii) the prediction performance of different models. For structure discovery, a true positive is an arc present in both the source and learned network graph and a false positive is an arc not present in the original network. True and false negatives are defined analogously. For prediction, the usual interpretations in terms of data records is used.

To evaluate the prediction results we also compute the Brier score [43], which is defined as

$$BS = \frac{1}{|D|}\sum_{i=1}^{|D|}(p_i - l_i)^2,$$

where p is the predicted probability and l the correct label. A Brier score of zero indicates perfect prediction.

5.3. Benchmark models

5.3.1. Static expert model

The model constructed in cooperation with pulmonologists from the Radboud University Medical Centre, described in [22], is used as a baseline model. This model is static, in the sense that it does not explicitly model time effects. To make temporal predictions we simply interpret the exacerbation probability as the prediction probability at a future time point.

5.3.2. Dynamic expert model

Additionally we constructed a dynamic version of the expert model by adding identity arcs to each variable. The graph is shown in Fig. 2. For the additional parameters we choose values heuristically, assuming that remaining in the same state is more likely than switching state. It should be noted that although we refer to this model as dynamic expert model, the pulmonologist was only involved in the construction of the static model and the dynamic version should therefore be seen only as a naive extension for comparison purposes.

5.3.3. Temporal nodes Bayesian network

As a further comparison we also constructed a temporal nodes Bayesian network [21]. This model is based on the static expert model structure described above, where the exacerbation variable is replaced by a temporal node that takes as values whether an exacerbation occurs in the intervals *today* or *tomorrow* or does not occur. The intervals are relative to the observations, i.e., the *exacerbation* temporal node models the probability of an exacerbation on the same day and the day after the measurements have been taken. The other variables are instantaneous nodes, as they model the health state at the time of measurement. Parameters for this model were learned from the Aerial data. Performance was tested using a *one against all* scheme. For example the prediction performance for *tomorrow* was tested with *tomorrow* as the positive class and the other two values as the negative class.

5.4. Implementation

The structure was learned using the BDe-score [35], with a Dirichlet prior with $\alpha = 1$ to learn the parameters. A greedy structure search procedure with random restarts was used. For the bootstrapping we used the implementation from the R-package *boot* [37], with a fixed block length of 5. The EM parameter learning was implemented using Smile [40], and the DBN structure search on the completed data was implemented in Banjo [41]. In practice, convergence of the complete procedure can be very slow due to the

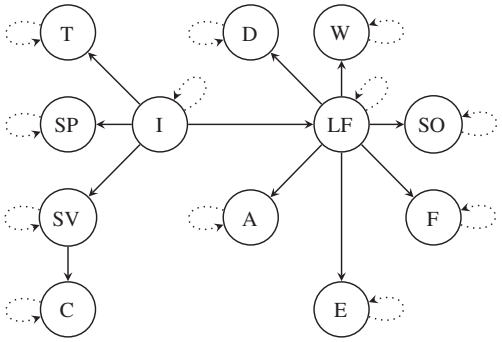

Fig. 2. Expert model. The variables are A = activity; C = cough; D = dyspnea; E = exacerbation; F = FEV_1; I = infection; LF = lung function; SO = SpO_2; SP = sputum purulence; SV = sputum volume; T = temperature; W = wheeze. Temporal arcs are dotted.

size of the search space. Within each iteration we use the generalised EM result that it is sufficient to update the model when it has a higher score, without having to maximise the score. Per iteration we set a time limit on the structure search only accepting the new structure when a higher score was found.

6. Results

6.1. Synthetic data

To get a feel for the performance of the methods we used, let us first look at the results of synthetic data from a known model (Fig. 3a).

6.1.1. Structural comparison

For the structural comparison we compare the arcs in the graph. The direction of arcs not involved in a v-structure can be reversed without influencing the conditional independencies in the graph and can therefore be represented by a line. However, the temporal arcs cannot be inverted by definition, as the future cannot influence the past. The representation where the direction of reversible arcs is dropped is called an essential graph. Table 1 shows the true positives and false positives/negatives of the learned model arcs. The true negatives are less interesting because the learning procedure is forced to only consider graphs with bounded indegree. For the complete data we do not actually need the structural EM method, but for completeness we still report the results.

Some interesting observations can be made. Looking at the true positive rate (TPR), we see that as the percentage of missing values increases the TPR decreases, as expected. The false negatives caused by failing to detect that there is both a direct and a temporal arc $D \rightarrow A$ and $D \rightarrow M$ is not a very serious issue, because whether these should be distinguished as separate arcs is quite sensitive to the parameters. Furthermore, all models missed the arc $F \rightarrow SO$, but looking at the parameters we see that the distribution is close to uniform which makes it almost impossible to detect the dependence from limited data. These kind of errors could also occur in the real data.

Table 1
Structural equivalence scores for models from synthetic data.

Data set	TP	FP	FN	TPR
Complete	11	4	3	$11/14 \approx 0.79$
Missing 10%	11	2	3	$11/14 \approx 0.79$
Missing 20%	8	3	6	$8/14 \approx 0.57$
Missing 30%	5	4	9	$5/14 \approx 0.36$

We also computed bootstrap models for the data with 30% missing values. In Fig. 3b the resulting aggregated model is shown. For each bootstrap replication we computed the best network using the procedure described above and then we averaged over the models to compute the probability of finding the arc. Due to the computational cost of the search procedure, bootstrapping was restricted to 9 resampled data sets. The arcs shown in the graph were found in a majority of the bootstrap models, i.e., the probability of the arc is $P(a) > 0.5$. Because the scores of the best networks are fairly close together, it makes little difference whether we use Eq. (1) or Eq. (2) to compute the arc scores. Observe that there are only a limited number of arcs with a high score, which is to be expected if there are multiple models that explain the data reasonably well. The arcs that are found are correct, except for $E \rightarrow SO$, which means 5 arcs are true positives and 1 is a false positive. The TPR is $5/14 \approx 0.36$ which is the same as the result without bootstrapping for data with 30% missing values. However, the number of false positives is lower for the bootstrapped model.

6.1.2. Prediction performance

We now turn to the goal of density estimation, for which prediction performance can be used as a measure. In the context of

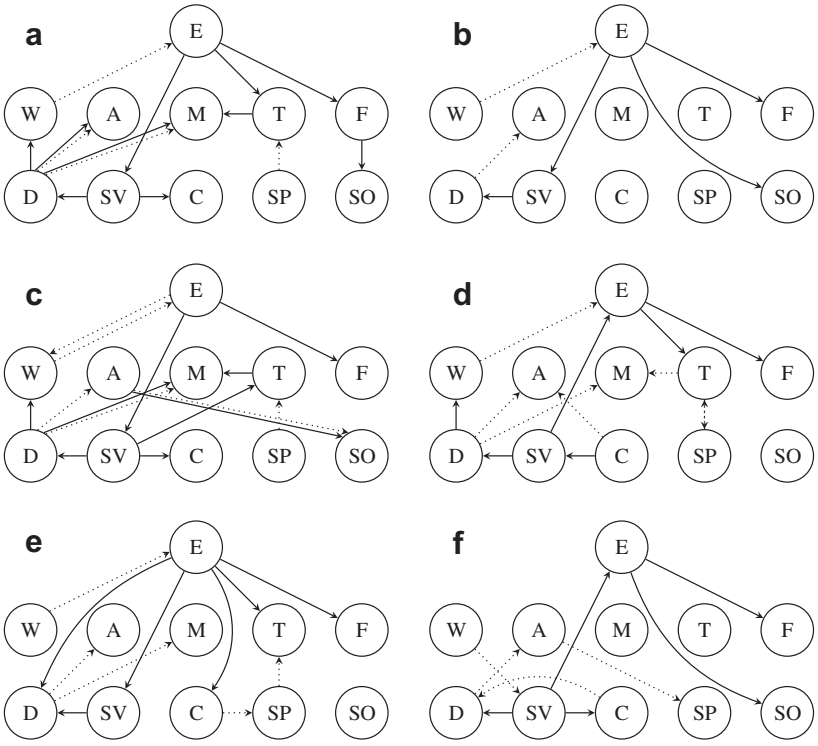

Fig. 3. (a) model that generated the data; (b) majority vote of arcs according to the bootstrap models; (c) model learned from complete data; (d) Model from data with 10% missing; (e) 20% missing; (f) 30% missing. The variables are A = activity; C = cough; D = dyspnea; E = exacerbation; F = FEV_1; M = malaise; SO = SpO_2; SP = sputum purulence; SV = sputum volume; T = temperature; W = wheeze. Temporal arcs are dotted.

COPD management, we are primarily interested in the probability of exacerbation. We compute the probability $P(E_{t+1} \mid O_t)$, which is the prediction probability of an exacerbation at time $t+1$ given the observations at time t (recall that the observations were made daily, so a single step is a day). A new data set with 30% missing values was generated from the original synthetic model as test data. An ROC-analysis shows that all models perform almost identically, with an area under the curve AUC = 0.63. This performance may seem low, but an ROC-analysis with the true model that generated the data also results in an AUC of 0.63. This is a consequence of the fact that the generating model contains limited temporal dependencies. Computing the auto- and cross-correlations for the generated time series also shows that correlations are small. Therefore we cannot expect to make very good predictions on these data. Inspection of these correlations in the real data shows the presence of more temporal information, so prediction performance analysis will be of more interest there.

The analysis of synthetic data indicates that even from a small sample of time series data we can reasonably reconstruct the network structure. As expected the performance is sensitive to the presence of missing values, but it appears that some amount of missing values is tolerable. These results provide sufficient confidence to analyse the COPD-monitoring data.

6.2. Aerial data models

The Aerial project aims to provide disease management for COPD patients. In order to achieve this goal we have performed a pilot study as described in Section 3. We now turn to the analysis of the gathered data to see whether our learning procedure is capable of finding relevant patterns.

6.2.1. Model learning and bootstrapping result

The result after 100 iterations of structural EM is shown on the left in Fig. 4. We will refer to the model learned from the Aerial data as the *Aerial Model*. It is not as easy as with the synthetic data to ascertain whether this structure is correct. We can, however, compare it to the structures found by the bootstrap replications. The same procedure as used on the synthetic data leads to a majority graph of the bootstrap replications of the Aerial data, shown on the right in Fig. 4. Comparing the two graphs we see that the majority bootstrap model is sparser than the Aerial model; the arc $E \to C$ is not present in the Aerial model and the relations between the variables T, M and E differ in arc direction and temporal signature. All other arcs in the majority bootstrap model are also present in the Aerial model. When we test the prediction performance of our models in Section 6.3, we will call the model average over the bootstrap models the *bootstrap model*.

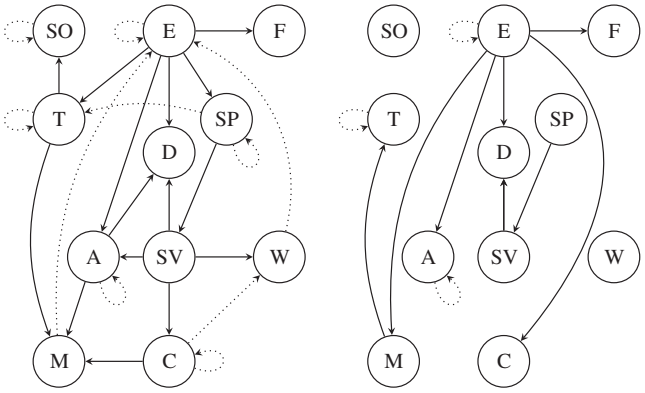

Fig. 4. Aerial model: Structure of the best learned network on the Aerial data (left). Majority graph learned from the bootstrap replications of the Aerial data (right). The variables are A = activity; C = cough; D = dyspnea; E = exacerbation; F = FEV_1; M = malaise; SO = SpO_2; SP = sputum purulence; SV = sputum volume; T = temperature; W = wheeze. Temporal arcs are dotted.

To get some insight in which variables are good predictors according to the model, it is informative to approach the prediction problem from the other side and look at the probability of predictors given the event, $P(X_{t-\delta} \mid E_t)$. In Table 2 the probabilities of some observations for $\delta \in \{0, 1, 2\}$ are shown. We can see that *dyspnea* is a good direct predictor, as its probability increases leading up to an exacerbation. The variable *activity* denotes a decrease in daily activities and is similarly predictive. *Sputum volume* and *cough* give less information about exacerbations over time.

6.2.2. Structural comparison to expert and naive Bayes models

It is informative to compare the result of the structure search to the expert model introduced in Section 5.3. Although this expert model is static, we can at least see whether the found dependencies are similar irrespective of whether arcs are temporal. The comparison is hindered by the presence of hidden variables in the expert model which are not present in the learned models – learning models with hidden variables is a difficult problem, see e.g. [44] for a possible approach. As a consequence we expect to find dependencies between symptoms that are mediated by a hidden variable in the expert model.

Although we cannot really interpret the model in Fig. 4 causally, it helps to keep the clinical meaning of the variables in mind when comparing the model with the expert model. The dependence between sputum volume and cough is found in both the expert and learned model. The learning procedure finds a dependence between exacerbation (E) and FEV_1 (F) which is an indicator of lung function, which is in line with the expert model. Fever is a strong indicator of infection, which is often the cause of an exacerbation, explaining the dependency $E \to T$ in the Aerial model, whereas the expert models the dependence of temperature on the hidden variable infection by the arc $I \to T$. In the expert model activity is influenced by lung function, which is captured by the dependence between activity, exacerbation and sputum volume in the Aerial model. So although there are clear differences between the models, the Aerial model appears to identify dependencies that can be explained. It should be noted, however, that identifiability of relations between subjective symptoms can be a problem with limited data, so there might exist other models that are also reasonable and perform similarly. The advantage of Bayesian networks in this context is that they at least provide a simple way to inspect the relations and for example check with a clinician whether the found relations are clinically defensible.

If we compare the structure of the Aerial model to the temporal naive Bayes structure (Fig. 5), we see that most of the dependencies between the features remain intact. Note that the structure of this model is restricted by both the naive Bayes arcs from exacerbation to all the other variables and by the complexity limit in our search procedure which bounds the number of parents to three. For the augmented naive Bayes model five temporal arcs are found, self loops for E, A, SP and T and the arc $SP \to T$, all of which are also present in the Aerial model. The atemporal arc $T \to SV$ is not present in the Aerial model and the path

Table 2
Probabilities of predictors given evidence of an exacerbation.

X	$P(X_{t-3} \mid E_t)$	$P(X_{t-2} \mid E_t)$	$P(X_{t-1} \mid E_t)$	$P(X_t \mid E_t)$
Dyspnea	0.55	0.60	0.67	0.79
Sputum volume	0.43	0.46	0.49	0.48
Cough	0.50	0.53	0.53	0.50
Activity	0.75	0.78	0.84	0.91

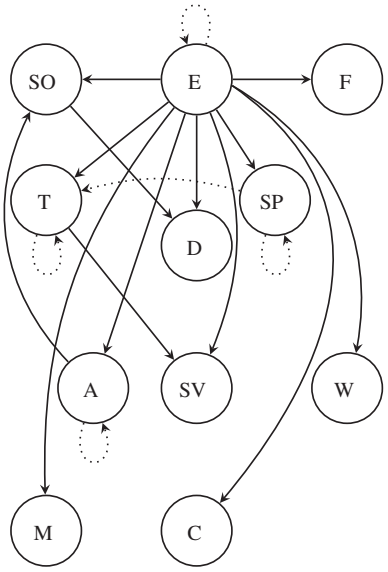

Fig. 5. Temporal augmented naive Bayes structure. Temporal arcs are dotted.

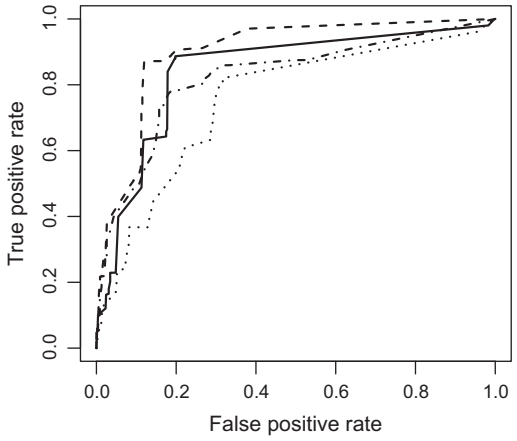

Fig. 6. ROC-curve on the validation data set D_{val} for the Aerial model predictions (solid); for the bootstrap averaged model (dash); and on the deduplicated validation data D_{dedup} for the Aerial model (dot) and bootstrap averaged model (dash-dot).

$A \rightarrow SO \rightarrow D$ is represented by the arc $A \rightarrow D$ in the Aerial model. Based on the data we have it is hard to determine directly which of these models is correct. Therefore we now turn to the evaluation of the models in terms of exacerbation prediction performance.

6.3. Prediction performance and validation

Ultimately, we are interested in prediction performance. We can use our bootstrap replications as an alternative cross-validation, measuring performance of the Aerial model on the bootstrap data. To do so we concatenated all the bootstrapped data and computed the TPR, FPR and AUC for the combined data. The results with the Aerial model are modest, with TPR = 0.76, FPR = 0.40 and AUC = 0.66. Although ideally our model should be tested on prospective data – a project to gather validation data is in planning – we can gain some insight in the generalisability of our results by testing the performance on a different data set.

We performed an ROC-analysis on the validation data set D_{val}, that has not been used for model learning. The ROC-curves for the Aerial and bootstrap models are shown in Fig. 6. The Aerial model reaches an AUC of 0.84 with one day ahead predictions on the validation data, which is an encouraging result. The bootstrap model (model average over the bootstraps), outperforms the Aerial model, obtaining an AUC of 0.90. In practice one has to decide on a cut-off point somewhere on the ROC-curve which gives a reasonable trade off between true and false positive rates. Our system can adapt the kind of feedback that is given to the patient based on the probability of exacerbation, incorporating different cut-off points for different kinds of advice. We should be careful, however, in interpreting the prediction results as consecutive time points are often the same, positively skewing the results by predicting that nothing changes.

To test the influence of repetitions in the test data we also checked the performance of the models on D_{dedup}, the deduplicated version of the data. This way we can see how the model reacts to state changes, which is ultimately what we are trying to detect. In Table 3, we summarise the results for different models: the static and dynamic expert model, the augmented temporal naive Bayes model and the Aerial and bootstrap models on the regular validation data and the deduplicated data. For all models we compute the performance measures from the prediction probabilities $P(E_{t+\delta} \mid O_t)$ for $\delta \in \{1, 2\}$.

As expected performance drops for all models on D_{dedup}, with the most notable decrease in the expert models. As AUC alone is not sufficiently informative, we also look at true and false positive rates, computed with the point on the ROC-curve closest to $(0, 1)$ as cut-off point. The true and false positive rates show that although the baseline static model appears to perform on par with the temporal models, this is only true on average over cut-off points. At the theoretically optimal cut-off we see that detecting changes is difficult with a static model, as one would expect. The Aerial model and the temporal naive Bayes perform similarly in this condition, but are outperformed by the bootstrapped model in terms of false positive rate without a significant change in true positive rate. The Brier scores show a similar pattern, although differences in scores are small.

The performance for two day ahead predictions is also shown in Table 3. A decrease in performance is expected as health status can change quickly over time. The same caveat with respect to predicting 'no change' applies, so again the performance on the deduplicated data is of more interest. In terms of AUC the performance drop seems relatively small, however, false positive rates are get-

Table 3
Summary of AUC, True Positive Rate (TPR), False Positive Rate (FPR) and Brier Score (BS) for different models on the validation data set D_{val} and the deduplicated data D_{dedup}.

Data	Model																			
	Expert static				Expert dynamic				Temporal ANB				Aerial model				Bootstrap model			
	AUC	TPR	FPR	BS	AUC	TPR	FPR	BS	AUC	TPR	FPR	BS	AUC	TPR	FPR	BS	AUC	TPR	FPR	BS
$t+1$																				
D_{val}	0.86	0.89	0.25	0.11	0.85	0.89	0.25	0.19	0.89	0.90	0.21	0.09	0.84	0.89	0.20	0.10	0.90	0.87	0.12	0.09
D_{dedup}	0.76	0.59	0.14	0.14	0.76	0.60	0.15	0.20	0.77	0.80	0.31	0.15	0.75	0.82	0.32	0.15	0.82	0.78	0.19	0.13
$t+2$																				
D_{val}	0.84	0.86	0.26	0.11	0.83	0.86	0.26	0.19	0.87	0.87	0.22	0.10	0.86	0.82	0.15	0.10	0.86	0.84	0.13	0.11
D_{dedup}	0.72	0.72	0.39	0.15	0.72	0.54	0.16	0.20	0.72	0.78	0.36	0.16	0.73	0.70	0.31	0.15	0.76	0.69	0.21	0.15

ting less acceptable for a useful application. The static expert model shows some remarkable behaviour by increasing its TPR for two day ahead predictions, but this is mostly a consequence of finding the optimal cut-off, resulting in a higher TPR but also very high FPR. The naive Bayes model performs best in terms of TPR, but the bootstrap model remains the best performer in terms of FPR. Hence it appears that different approaches are all capable of making predictions with some accuracy, but that model averaging over bootstrap models performs somewhat better overall. This shows that bootstrapping is a useful technique to learn clinical models from small data samples of patient monitoring data.

6.3.1. TNBN prediction performance

We tested the performance of the TNBN model on D_{dedup}. For same day predictions the AUC, TPR and FPR are 0.80, 0.75 and 0.24 respectively. While for predicting whether *tomorrow* an exacerbation will occur the same measures are 0.66, 0.67 and 0.43. Comparing these latter values to the results in the $t+1$ row of Table 3, we see the TNBN performs worse than the other models. A possible explanation for this result is that because the TNBN does not model the dynamics the only data that is available to estimate the parameters for (*exacerbation*, *tomorrow*) are the actual data points a day before an exacerbation. The DBNs in contrast try to model the temporal interactions between variables during both stable and exacerbation episodes. TNBNs are in principle suitable for these kind of modelling problems; however, when limited data is available it seems performance suffers.

6.3.2. Bootstrap block length

Finally, we checked that prediction performance is not overly sensitive to the choice for the bootstrap block length parameter. Three block length values were tested: 3, 5 and 7. The prediction performance results are similar for all values, but a block length of 7 performs somewhat worse, which is likely because it is too long relative to the total time series length. In general the block length should be chosen to capture the time scale of the effects of interest but short enough compared to the length of the series to attain sampling variation. A block length of 5 therefore seems a good choice for our data.

7. Discussion

Chronic disease management using automatic data interpretation requires analysing time series to make predictions. We studied time series data from a pilot study with chronic obstructive pulmonary disease patients, with the purpose of developing a predictive model for COPD exacerbations. In this section we discuss the impact of our findings, limitations and future work.

There has been quite some work on telehealth and monitoring for chronic diseases in general and COPD in particular [10], but automatic data interpretation has not played an important role up to now. However, the methods based on developments in artificial intelligence offer powerful tools for automatic interpretation. Bayesian networks are well suited to deal with data analysis in a clinical context, because the models are interpretable – one can ask a clinician whether the relations found with structure learning make clinical sense. Equally important for use in a home-care setting is that it remains possible to make predictions when data values are missing. Since missing values are virtually impossible to prevent completely, this is an important requirement. Dynamic Bayesian networks generalise the capabilities of Bayesian networks to time series analysis. An issue that relates to model interpretability is that the best performance is obtained by a model average, which is more difficult to understand. Cooperation with a clinician seems advisable, to make sure that differences in models describe possible domain features. Alternatively, augmented naive Bayes may be a better choice when a fully interpretable model is desired. In the augmented naive Bayes model only the interpretation of the dependencies has to be verified to be clinically valid and at least in our data the performance difference with the model average was fairly small.

Structure learning of Bayesian networks has been studied extensively [30], but the present challenge was combining structure learning techniques for DBNs with sparse data methods. Structural EM is in itself already a computationally expensive operation and bootstrapping data sets makes it even more daunting, which is clearly a limitation when using these methods for very large models. The computation time for our COPD data, however, although long in absolute terms (about a day per data set on current hardware), is negligible compared to the time investment of data acquisition. Therefore it may still be worth considering even for larger models. The current result also indicates that if model averaging is too expensive, augmented temporal naive Bayes provides an alternative that is easier to obtain because the model space is restricted, but offers good performance relative to the computational complexity. Although model learning is relatively expensive, computing predictions is easy due to the small size of the models (in fact in our application model predictions are computed on a smartphone). The model can be updated offline when sufficient data have been collected to warrant recomputation. In practice the computational cost is therefore not a limiting factor.

For our COPD application our results are promising as we are able to learn models that appear quite capable of predicting exacerbation occurrences. Summarising the trade-off between the different models we can conclude the following. Of the models that were analysed the temporal augmented naive Bayes approach is particularly interesting as it gives good predictive results but is simpler than a full structure learned model. Yet, when insight in the underlying clinical process is the main interest, structure learning will be preferable.

There are limitations to the extend to which we can rely on the results of the validation. The test data has not been gathered specifically for this purpose and does not contain all the variables in our model, also the number of missing values may not be typical for the data that needs to be analysed in the final system. Furthermore, the decision on what is an acceptable trade-off between true and false positives is beyond the technical scope and depends on clinical views, cost considerations and regulatory requirements. As the project's initial focus was on analysing the monitoring data, background information on the patient, including current treatment, and more general environmental factors (e.g. time of year, weather conditions) have not been taken into account at present, but are part of future work. Whether our results generalise will be tested in a follow-up study that is currently being planned, in which a more substantive group of patients will be monitored, extensive baseline information will be taken into account and exacerbation events will be verified by a pulmonologist as gold standard. A confirmation of the present prediction results would then open up the way for a practical implementation of automatic data interpretation for COPD disease management.

References

[1] Friedman N, The Bayesian structural EM algorithm. In: UAI '98: proceedings of the 14th conference on uncertainty in artificial intelligence; 1998.
[2] Efron B. Bootstrap methods: another look at the jackknife. Ann Statist 1979;7(1):1–26.
[3] Viegi G, Pistelli F, Sherrill D, Maio S, Baldacci S, Carrozzi L, et al. epidemiology and natural history of COPD. Eur Respir J 2007;30(5):993–1013.
[4] Wedzicha J, Seemungal T. COPD exacerbations: defining their cause and prevention. Lancet 2007;370(9589):786–96.

[5] Wilkinson T, Donaldson G, Hurst J, Seemungal T, Wedzicha J. Early therapy improves outcomes of exacerbations of chronic obstructive pulmonary disease. Am J Respir Crit Care Med 2004;169(12):1298–303.
[6] Selvin S. Survival analysis for epidemiologic and medical research. Cambridge University Press; 2008.
[7] van Gerven M, Taal B, Lucas P. Dynamic Bayesian networks as prognostic models for clinical patient management. J. Biomed. Inf 2008;41(4):515–29.
[8] Peelen L, de Keizer N, de Jonge E, Bosman R, Abu-Hanna A, Peek N. Using hierarchical dynamic Bayesian networks to investigate dynamics of organ failure in patients in the Intensive Care Unit. J Biomed Inf 2010;43(2):273–86.
[9] Himes B, Dai Y, Kohane I, Weiss S, Ramoni M. Prediction of chronic obstructive pulmonary disease (COPD) in asthma patients using electronic medical records. J Am Med Infor Assoc 2009;16(3):371–9.
[10] McLean S, Nurmatov U, Liu J, Pagliari C, Car J, Sheikh A. Telehealthcare for chronic obstructive pulmonary disease. Cochrane Database Syst Rev 2011;7.
[11] Polisena J, Tran K, Cimon K, Hutton B, McGill S, Palmer K, et al. Home telehealth for chronic obstrcutive pulmonary disease: a systematic review and meta-analysis. J Telemed Telecare 2010;16(3):120–7.
[12] Basilakis J, Lovell N, Redmond S, Celler B. Design of a decision-support architecture for management of remotely monitored patients. IEEE Trans Inf Technol Biomed 2010;14(5):1216–26.
[13] Dagum P, Galper A, Horvitz E. Dynamic network models for forecasting. In: UAI '92: proceedings of the 8th conference on uncertainty in artificial intelligence; 1992. p. 41–8.
[14] Dagum P, Galper A. Forecasting sleep apnea with dynamic models. In: UAI '93: proceedings of the 9th conference on uncertainty in artificial intelligence; 1993. p. 64–71.
[15] Friedman N, Goldszmidt M, Wyner A. Data analysis with Bayesian networks: a bootstrap approach. In: UAI '99: proceedings of the 15th conference on uncertainty in artificial intelligence; 1999.
[16] Kim S, Imoto S, Miyano S. Inferring gene networks from time series microarray data using dynamic Bayesian networks. Briefings Bioinformatics 2003;4(3):228–35.
[17] Breiman L. Bagging predictors. Mach Learn 1996;24(2):123–40.
[18] Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Mach Learn 1997;29(2-3):131–63.
[19] Tucker A, Liu X. Learning dynamic Bayesian networks from multivariate time series with changing dependencies. In: IDA '03: proceedings of the 5th international symposium on intelligent data analysis, LNCS, vol. 2810. 2003. p. 100–10.
[20] Lähdesmäki H, Shmulevich I. Learning the structure of dynamic Bayesian networks from time series and steady state measurements. Mach Learn 2008;71(2-3):185–217.
[21] Arroyo-Figueroa G, Sucar L. A temporal Bayesian network for diagnosis and prediction. In: UAI '99: proceedings of the 15th conference on uncertainty in artificial intelligence; 1999.
[22] van der Heijden M, Lucas P, Lijnse B, Heijdra Y, Schermer T. An autonomous mobile system for the management of COPD. J Biomed Inf 2013;46(3):458–69.
[23] GOLD. Global initiative for obstructive lung disease. <www.goldcopd.com>.
[24] Bischoff E, Boer L, Molema J, Akkermans R, van Weel C, Vercoulen J, et al. Validity of an automated telephonic system to assess COPD exacerbation rates. Eur Respir J 2012;39(5):1090–6.
[25] Cooper G. A diagnostic method that uses causal knowledge and linear programming in the application of Bayes' formula. Comput. Methods Programs Biomed 1986;22(2):223–37.
[26] Pearl J. Probabilistic reasoning in intelligent systems: networks of plausible inference. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1988.
[27] Dean T, Kanazawa K. A model for reasoning about persistence and causation. Comput Intell 1989;5(3):142–50.
[28] Murphy KP. Dynamic Bayesian networks: representation, inference and learning, Ph.D. thesis. University of California, Berkeley; 2002.
[29] Robinson J, Hartemink A. Learning non-stationary dynamic Bayesian networks. J Mach Learn Res 2010;11:3647–80.
[30] Koller D, Friedman N. Probabilistic graphical models: principles and techniques. The MIT Press; 2009.
[31] Murphy K. Machine learning: a probabilistic perspective. Cambridge, MA: MIT Press; 2012.
[32] Dempster A, Laird N, Rubin D. Maximum likelihood from incomplete data via the EM algorithm. J Roy Stat Soc Ser B (Methodological) 1977;39(1):1–38.
[33] Acid S, de Campos L, Fernández-Luna J, Rodríguez S, Rodríguez J, Salcedo J. A comparison of learning algorithms for Bayesian networks: a case study based on data from an emergency medical service. Artif Intell Med 2004;30(3):215–32.
[34] Dash D, Druzdzel M. A hybrid anytime algorithm for the construction of causal models from sparse data. In: UAI '99: proceedings of the 15th conference on uncertainty in artificial intelligence; 1999.
[35] Heckerman D, Geiger D, Chickering D. Learning Bayesian networks: the combination of knowledge and statistical data. Mach Learn 1995;20:197–243.
[36] Lahiri S. Theoretical comparisons of block bootstrap methods. Ann Stat 1999;27(1):386–404.
[37] Canty A, Ripley B. boot: bootstrap R (S-Plus) functions; 2012. <http://cran.r-project.org/web/packages/boot/>.
[38] Friedman N, Murphy K, Russell S. Learning the structure of dynamic probabilistic networks. In: UAI '98: proceedings of the 14th conference on uncertainty in artificial intelligence; 1998.
[39] Kearns M, Mansour Y, Ng A, An information-theoretic analysis of hard and soft assignment methods for clustering. In: UAI '97: proceedings of the 13th annual conference on uncertainty in artificial intelligence; 1997.
[40] Decision Systems Laboratory. University of Pittsburgh, SMILE: Structural Modeling, Inference, and Learning Engine. <http://genie.sis.pitt.edu/>.
[41] Hartemink A. Banjo: Bayesian Network Inference with Java Objects, Duke University; 2010. <http://www.cs.duke.edu/~amink/software/banjo/>.
[42] Hurst J, Donaldson G, Quint J, Goldring J, Patel A, Wedzicha J. Domiciliary pulse-oximetry at exacerbation of chronic obstructive pulmonary disease: prospective pilot study. BMC Pulm Med 2010;10(1):52.
[43] Brier G. Verification of forecasts expressed in terms of probability. Mon Weather Rev 1950;78(1):1–3.
[44] Elidan G, Friedman N. Learning the dimensionality of hidden variables. In: UAI '01: proceedings of the 17th conference on uncertainty in artificial intelligence; 2001.