LEARNING TO GENERATE INDIVIDUAL SEQUENCE DATA FROM POPULATION STATISTICS USING DYNAMIC BAYESIAN NETWORKS

By

MOHAMMED AZMAT QURESHI

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

Of the Requirements

For the Degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

THE UNIVERSITY OF TEXAS AT ARLINGTON

May 2018

For JAWAN

ACKNOWLEDGEMENTS

I would like to thank my advisor, Professor Manfred Huber, for supporting me over the years, and for giving me so much freedom to explore and discover different areas of probabilistic approach and analysis. I appreciate his efforts in taking time to advise me whenever being approached and share interesting facts and stories for just about anything. I would also like to thank my committee members Dr. Jean Gao and David Levine for taking time to serve on my committee and for their valuable inputs and comments.

I would like to thank Arun, Bassam, Bhushan, Budhyant, Kunal, Mahesh, Sandeep, Somnath, Sourabh, Thomas, Vamsi and all my friends in LEARN Lab who have been part of my journey and helped me make a family away from home. I would like to thank Ankur, Apurv, Adi, Milind, Vipul and Veenu, my friends from my hometown for always reminding me that happiness is still in being silly.

I would like to thank my family for teaching me to laugh even in most miserable situations, My Grandparents for their unconditional love, Mohammed Jawed, for being a loving father and making me believe in myself, Nasreen Bano, my mom who always and will always pamper me slightly more than my sister, Dr Afshan, my little sister who has always been on my side except when she is fighting with me. Finally, I would like to thank my wife Wajiha in putting up with all my shenanigans, for listening to my never-ending stories on loop, to course correct me whenever I went astray, for reminding me things which I forget and especially the things that I should not.

I love and respect you all.

Abstract

LEARNING TO GENERATE INDIVIDUAL SEQUENCE DATA FROM POPULATION STATISTICS USING DYNAMIC BAYESIAN NETWORKS

MOHAMMED AZMAT QURESHI, MS

The University of Texas at Arlington, 2018

Supervising Professor: Manfred Huber

Data collection rose exponentially with the dawn of the 21st Century, However the most important data to humans, individual health data, is difficult to get approved for public research, as medical history is very sensitive to be distributed. The only available public data which can be retrieved from institutions like the Centre for Disease Control (CDC), World Health Organization (WHO), National Health Interview Survey (NHIS), etc. largely only contain population statistics for different attributes of a person.

What we propose here is a generative model which would learn to create data sequences for a population, each sequence mimicking an individual person's behavior, such that the set of generated sequences represents this entire population by matching the available population statistics using Dynamic Bayesian Networks. The data would contain a population in which each person will have their exercise, injury and illness data over time. Various factors are interlinked within and between time slices, e.g. the amount of exercise a person does at time *t,* depends on factors [Age, Health-Status, Person Type] at time *t,* and exercise done at time *t-1.* Although the data generated by the learning model is not real, it should be closer to it, supporting algorithm development and initial testing.

Table of Contents

1 INTRODUCTION

1.1 Motivation

As of 2016 total health care expenditure in United States Alone is $3.3 trillion or $10 grand per person. [12] Although enough data is collected by the hospitals and insurance firms, it is never distributed. Even anonymized data doesn't often see the light of day in the world of public research as the data itself is very sensitive.

1.1.1 Health and Fitness

Health is defined as a state of complete physical, mental and social well-being and not merely the absence of disease or infirmity.

Over the years many advancements have been made in health care and with technology we have revolutionized our approach towards health and its care. With the dawn of information age, data collection has grown exponentially, with data being collected for literally every possible preference of a person, their music preference, movie preference, food preference etc. However, there is often no emphasis given to collecting sequential data, and if at all any heed is paid towards it, the data is often not made public due to privacy concerns. Thereby we end up in a situation where we virtually have no publicly available sequential data in the health care domain.

1.1.2 Sequential Data

Sequential Data is important in many aspects of science and engineering. This sequential data can be a time series, which can be analyzed either online where data arrives in real discrete intervals of time, or offline where all the data is gathered beforehand.

To do any type of research on sequential health and fitness data, we need those data points, so that different types of research can be done. Our aim here is to develop a generative learning model out of population statistics that are made available either by Center for Disease Control, World Health Organization, etc. The model should be able to generate individual person's health trajectories, given his personal attributes. The data generated for each person, although not real, should give a sense of realism for each individual and for the entire population as it matches expert knowledge and the population statistics.

1.2 Approach

A person's age, their Injuries and diseases causes changes to their regular physical activity or normal routines of life, while few illnesses like heart attack; if survived could cause a dramatic change in physical fitness and getting rid of bad habits such as smoking. Smoking internally could potentially make a person twice as likely to get cancer than otherwise. While these dependencies are not generally contained in available healthcare statistics, they are available in qualitative form from experts and other health data sources. On the other hand, population statistics available from diverse health organizations provide quantitative information regarding prevalence of conditions. To consider both information sources and simulate these events to obtain individual data sequences for subsequent algorithm development and testing, we use a Dynamic Bayesian Network as the core of our model, with structure largely derived from qualitative expert knowledge and parameters such as transition from one-time step to another learned to conform with the quantitative population statistics. To achieve this, we use forward and backward inference in our model to generate a sample data for us to learn. Parameters are learned using a hill climbing approach.

2 BACKGROUND

2.1 Health

2.1.1 History

Health has been key objective of humans, thus since the very beginning huge importance has been given to health and wellbeing. Since the time of ancient Greece medicine has been there, and proper study of human health is evolving since that time. Over the course of the last century science and technology has grown exponentially and humans as a species have learnt a lot about maintaining a healthy lifestyle and staying fit.

With the advent of the industrial revolution, human started mass producing everything including food, drinks, clothes etc. With the rise in the number of machines humans use we have recently seen a reduction in the physical effort of the average individual. This, in turn, has given rise to obesity in large numbers. Over the last few generations obesity has been increasing, leading to significant health side-effects that have burdened health care systems. This development, in turn, has given rise lately to increased awareness towards healthy lifestyles and more health self-management.

2.1.2 Health Maintenance

Healthy lifestyle can be achieved and maintained by bringing discipline in one’s life, in a few basic areas:

- Diet
- Exercise
- Sleep

2.1.3 Role of Science

Science has always been part of health, from finding cures for diseases to making modern machines detect tumors and cancers within our body. Science has always stressed research to increase our quality of life, life expectancy, the recovery rate of a person from the state of sickness to the state of health and well-being.

2.2 Fitness

Fitness is one aspect of health that relates to maintaining a healthy body by performing physical activities. With the increase in obesity in humans it has become imperative that a person maintains some amount of physical activity in their daily routine.

A person can do exercises such as walking, jogging, running, cycling, aerobics, swimming etc. With busy modern life many people turn to gymnasiums to do their daily exercises and use machines to effectively work on specific muscles.

2.2.1 Exercise

As mentioned above exercise has become an essential part of modern life. Doing exercise can reduce obesity in people. A person doing exercise feels great, stays away from comfort food, and receives positive feedback about himself or herself, and continuing the same path accumulates his rewards.

Every exercise has an exercise rate which determines exertion and with it how fast a person is burning calories. The higher the rate of exercise, the faster the calories are burned by a person.

3 TECHNICAL BACKGROUND

3.1 Bayesian Networks

3.1.1 Background

A Bayesian network is a directed acyclic graphical model, which represents a probabilistic relation between random variables, i.e. every node in a Bayesian network can be thought of as a random variable and every edge as a probabilistic dependency among the corresponding random variables. Each node is associated with a probability function which takes as input a set of values for that node's parents and provides a probability or probability distribution as an output.

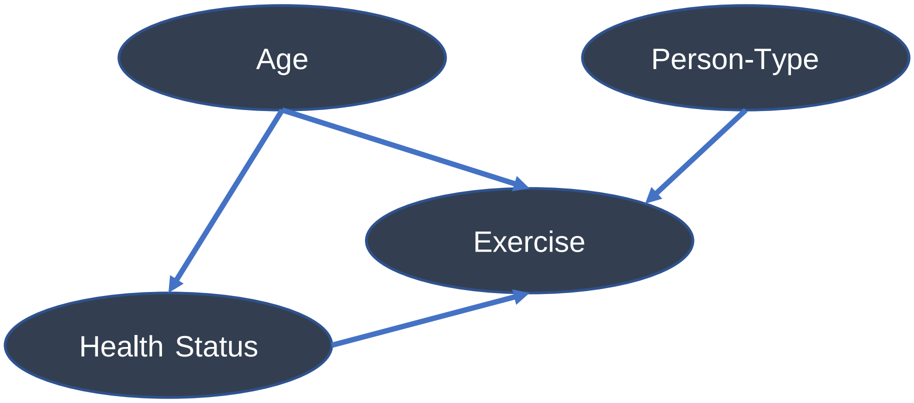


Figure 1. Simple Bayesian Network for Health Data.

The above figure is a simple Bayesian Network, where Health Status depends on a person's Age and Exercise depends on Age, Health-Status and Person Type.

The joint probability function for the network in the above figure would be

$$P(E, A, PT, PHS) = P(E| A, PT, PHS) * P(PHS| A) * P(PT) * P(A)$$

where the name of the above variables is abbreviated to E = Exercise, A = Age, PT = Person-Type and PHS = Health Status.

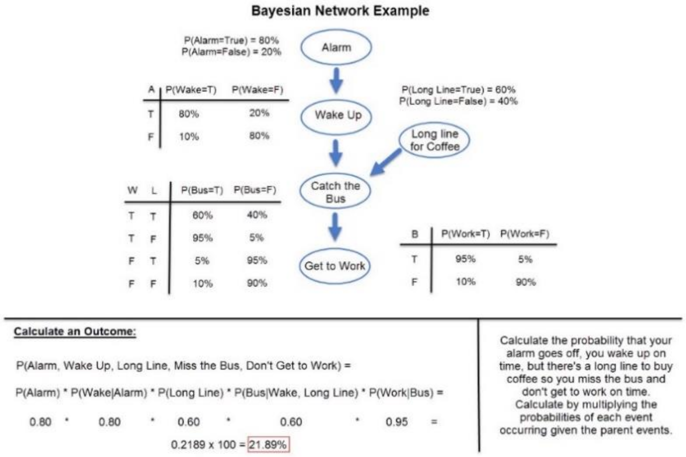
This representation as a graphical model is intuitive and provides an effective representation and computational structure for the joint probability distribution of a given set of random variables. [4]

The structure of a Bayesian network is defined by nodes (vertices) and directed edges. The nodes represent random variables and are usually visualized as circles/ellipses that bear the name of the corresponding random variable. The edges represent dependencies between the variable they connect. The tail of the arrow represents the parent or serves as an evidence variable; the head represents the variable which is conditionally dependent on its parents.

The acyclic nature of the Bayesian network ensures that we do not have cyclic dependencies, i.e. that a parent cannot be a descendant of its descendants. [4]

The main aim of this representation of conditional dependencies is to permit the reduction of the size of the joint distribution table which generally grows exponentially with the number of variable. Using knowledge of conditional dependencies, this can be reduced to a complexity that gross exponentially only in the number of parents, thus providing a more efficient and effective way to compute posterior probabilities as the complexity of the inference is reduced.

Consider the following example [5] shown in Figure 2; it represents a model of a person going to work on time, given certain events that can occur with some given probabilities.



P(Alarm=True) = 80%
P(Alarm=False) = 20%

A	P(Wake=T)	P(Wake=F)
T	80%	20%
F	10%	80%

P(Long Line=True) = 60%
P(Long Line=False) = 40%

W	L	P(Bus=T)	P(Bus=F)
T	T	60%	40%
T	F	95%	5%
F	T	5%	95%
F	F	10%	90%

B	P(Work=T)	P(Work=F)
T	95%	5%
F	10%	90%

Calculate an Outcome:

P(Alarm, Wake Up, Long Line, Miss the Bus, Don't Get to Work) =

P(Alarm) * P(Wake|Alarm) * P(Long Line) * P(Bus|Wake, Long Line) * P(Work|Bus) =

0.80 * 0.80 * 0.60 * 0.60 * 0.95 =

0.2189 x 100 = 21.89%

Calculate the probability that your alarm goes off, you wake up on time, but there's a long line to buy coffee so you miss the bus and don't get to work on time. Calculate by multiplying the probabilities of each event occurring given the parent events.

Figure 2. Simple Bayesian Network (from [5]).

The events that can occur are that a person hears an alarm, that a person wakes up on time given he has heard the alarm (denoted by Wake up). That a person catches the bus on time (denoted by Catch the bus) given there was a long line for coffee (denoted by Long Line for coffee) and he woke up on time. Get to work represents the event of him/her getting to work on time.

The joint probability for an event that the person hears his alarm, does not get up in time, is lucky and does not find a long line at the coffee shop, misses the bus and does not get to the work on time is represented by

P (Alarm, ~Wakeup, ~Longline Coffee, ~Catch the Bus, ~Get to Work) =

P(Alarm) * P (~ Wakeup | Alarm) * P(~longline) * P (~Catch the Bus | ~wakeup, ~longline) * P (~Get to work | ~ Cath the bus) = 0.80*0.20*0.4*0.9*0.90 = **0.05184**

3.1.2 Inference in Bayesian Networks

“Inference is the task of computing the probability of each state of a node in a Bayesian Network when other variables are known”. For us to perform inference we must do belief propagation, i.e. update the beliefs in each variable given observations of some variables. [1]

Consider the following

P (Alarm | Get to Work) = $\frac{P(Alarm, Get\ to\ Work)}{P(Get\ to\ Work)}$

where,

$$P(Alarm, Get\ to\ Work)$$
$$= \sum P(Alarm, WakeUp, LongLine, Catch\ Bus, Get\ to\ Work)$$

$$summed\ over\ all\ possible\ values\ for\ Wakeup, Longline, Catch\ Bus$$

Since inference in Bayesian networks can be NP hard, approximate inferences can be done using Markov Chain Monte Carlo which uses different sampling algorithms such as Gibbs sampling, Metropolis Hastings algorithm, rejection sampling, importance sampling, or the forward-backward propagation algorithm. [4]

3.1.3 Learning Bayesian Networks

Learning Bayesian Network parameters is slightly easier that learning the structure (i.e. the conditional dependencies between nodes) of the network itself. For a known Bayesian network structure with partial observability, expectation maximization (or gradient ascent) and MCMC are good tools for learning the problem.

3.2 Hidden Markov Model

3.2.1 Markov Chain

A Markov chain is an observed Markov model which contains the information about states and state transition probabilities. It models the probabilistic evolution of a system over time by allowing propagation and inference of the state distribution over time from observation sequences. The transition probabilities are represented by a transition matrix. A Markov chain works on the principle of the Markov Assumption, which states that the probability to go to the next state depends only on the current state and not on any state or transition we have made so far.

3.2.2 Hidden Markov Model (HMM)

Hidden Markov Models (HMMs) are a formal representation for Markov chains, aimed at permitting inference about the state of the underlying process from a sequence of observations. To do this, HMM models include the (hidden – i.e. unobservable) system state, the observations, as well as the probabilistic relations between observations and states and between states in consecutive time steps, assuming the Markov assumption holds. Inference can then be used to determine the probability distribution over the possible states given the available observation sequence as well as to determine the best model parameters given a larger set of observed data sequences.

Figure 3 shows a general representation of an HMM. Here States X_1…. X_t are hidden states of the model and Y_1…. Y_t are observations. Along with the above state and observation sets a hidden Markov model also contains Initial probabilities of the states, transition probabilities and overserved probabilities.

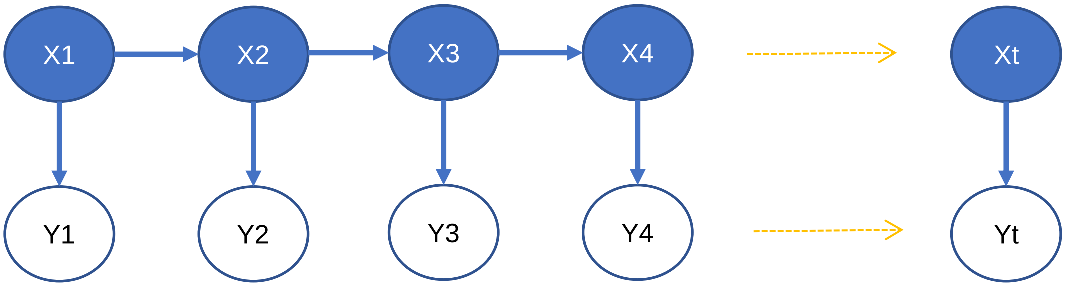


Figure 3. Hidden Markov Model

Likelihood in a Hidden Markov Model can be computed by

$$P(Y) = P(Y|X) * P(Y)$$

$$P(Y) = \sum_{X} P(Y, X) = \sum_{X} P(Y|X) * P(X)$$

The forward algorithm used to compute likelihood is given as:

```
function FORWARD (observations of Len T, state-graph of Len N)
    returns forward-prob create a probability matrix forward [N+2, T]
for each state s from 1 to N do
    forward[s,1] ←a0, s ∗ bs(o1)
for each time step t from 2 to T do
    for each state s from 1 to N do
            $forward [s, t] \leftarrow \sum forward [s', t - 1] * a_{s',s} * b_s(o_t)$
forward [qF, T] ← ∑ forward [s, T] ∗ as, qF
return forward [qF, T] [6]
```

3.3 Dynamic Bayesian Network

3.3.1 Introduction

Most events in our life can’t be based on a point of time, but rather described through multiple states of observations that collectively lead to one final event. [1] Hidden Markov Models are a commonly used representation for such temporal processes. However, inference in HMMs can become rapidly intractable as the number of states needed to represent the system increases.

Since temporal analysis is an important part of AI and reasoning, it is important to build models that reduce the complexity of the inference process to allow for real-world problems to be represented. In the case of non-temporal systems, Bayesian networks, as indicated before, allow the reduction of the complexity of the inference process by utilizing known dependency relations between the random variables describing the system state. However, they do not account for time and provide no features representing temporal dependencies. To address this, the concepts in Bayesian networks and HMMs can be combined, leading to Dynamic Bayesian Networks which are a way to represent factored HMMS, where the state and transitions are structured around dependency relations and thus inference can be accelerated.

According to [1,7] a dynamic Bayesian network should represent a system that evolves over time. The model should be able to update the user as time proceeds and predicts further behavior of the system. A dynamic model can be classified as a temporal model where each time slice of that temporal model is a state of a system and moving across different time slices corresponds to the change in the state across time.

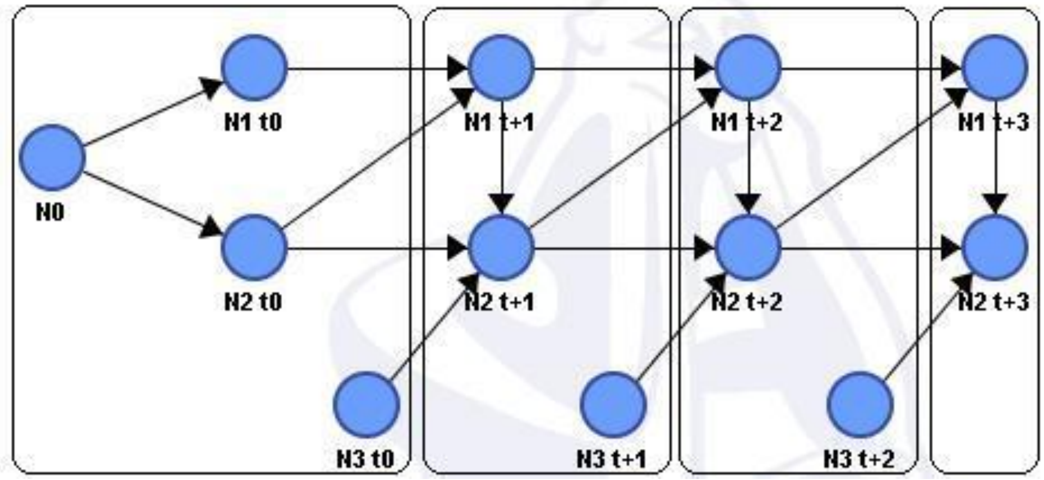


Figure 4. Dynamic Bayesian Network [8]

The above figure shows a basic Dynamic Bayesian network where the entire network is divided into discretized time slices. Each node is represented by its name and the time step they are in. The nodes can have intra time step dependencies or inter time step dependency or both.

Parents of a node are all the nodes in the same time step on which this node directly depends, and all the nodes in the previous time steps which have a direct relation with that node. In Figure 4, for example, the parents of $N_{2, t+1}$ are $N_{1, t+1}$, $N_{2, t0}$, $N_{3, t0}$.

Figure 5 below shows a Dynamic Bayesian Network in a single time step.

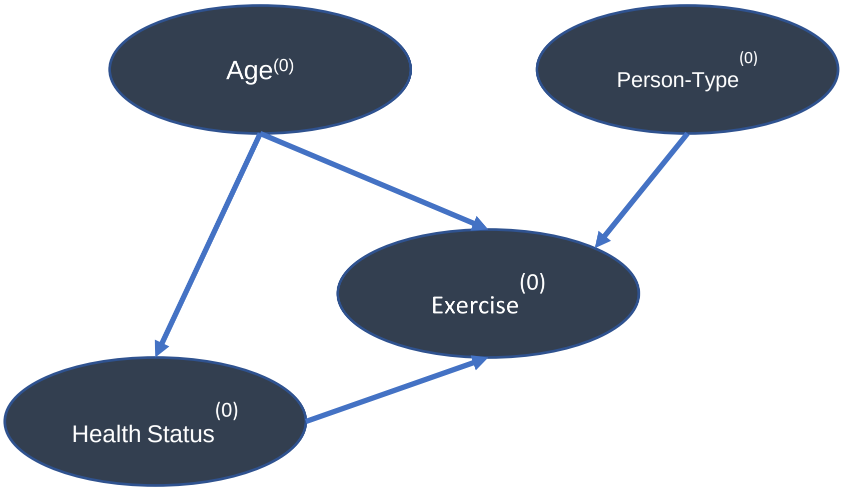


Figure 5. Single Time Slice of a Dynamic Bayesian Network

This figure is very similar to the Bayesian network in Figure 1, the only difference it has that it now contains a time step tag, which in the case above is zero. Using the above Bayesian Network, we can construct an arbitrary long dynamic Bayesian network. A Dynamic Bayesian Network for the single step network shown in Figure 5 is shown in Figure 6.

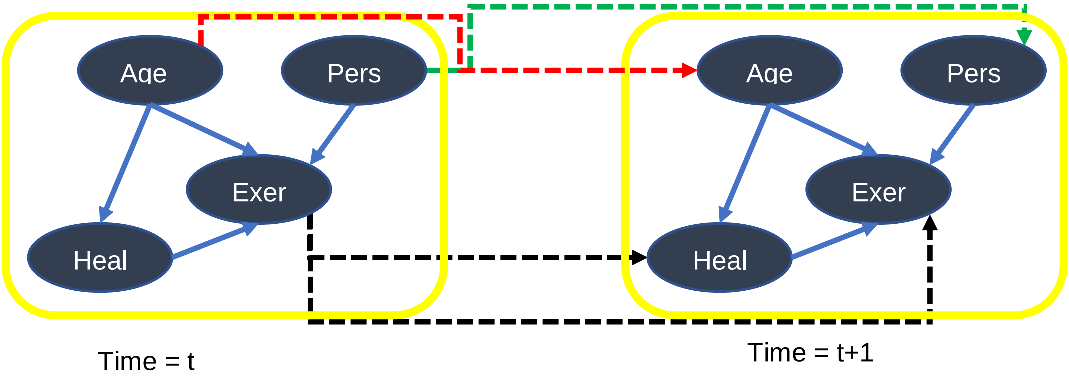


Figure 6. Dynamic Bayesian Network with State Dependencies as in Figure 5.

Figure 6 is like Figure 4; here we can see that Exercise at time t+1 depends on the Age, Person-Type and Health Status at time t+1, and Exercise at time t.

An unrolled Dynamic Bayesian Network model can be divided into three major components: i) its Transition Model, ii) its conditional distribution, and iii) its dependency model.

Given random variables $X_1 \ldots . X_n$, and observations $Y_1 \ldots . Y_n$, at time t and random variables $X_1' \ldots . X_n'$ and observations $Y_1' \ldots . Y_n'$, at time t+1, then ([1])

$$P(X', Y') = P(X'|X) * P(Y'|X') * P(X)$$

Here the conditional distribution is given by,

$$P(X'|X) = \prod_{i=1}^{n} P(X_i'|Pax_i')$$

and the dependency model for both state t and state t+1 is,

$$P(Y'|X') = \prod_{i=0}^{n} P(yi|x_i \ )$$

Using these equations our DBN can be used to model and perform inference for arbitrarily long sequence of states.

3.3.2 Inference

Only 1-time step of the observation nodes and none of the hidden state variables can be observed at a time in a DBN. That means that to determine the hidden variables at a different time step or to predict possible observations at a later point in time, we must infer all the unknowns in the network. To do inference in a Dynamic Bayesian network we need to calculate P (X_0^{T-1} | Y_0^{T-1}) where Y_0^{T-1} represents a set of observations made until the last time step while the hidden variables are represented by X_0^{T-1} = {x_1, …, x_{T-1}}. This can be achieved using forward-backward propagation. [1,2]

3.3.2.1 Forwards Propagation

Let $\alpha_t(x_t)$ be a forward probability distribution that describes the joint probability observation, collected until time t and hence can be represented as

$$\alpha_t(x_t) = P(Y_0^t, x_t)$$

With initial condition

$$\alpha_0(x_0) = P(x_0)$$

we get

$$\alpha_{t+1}(x_{t+1}) = P(y_{t+1} \mid x_{t+1}) \sum_{x_t} P(x_{t+1} \mid x_t) \alpha_t(x_t)$$

As described in [1], one of the interesting results of forward propagation is the term for likelihood of the observation data sequence Y_0^{T-1}. From the definition of the forward factor $\alpha_t(x_t)$ in the equation above we can say that

$$P(Y_0^{T-1}) = \frac{\alpha_{T-1}(x_{T-1})}{\sum_{x_t} \alpha_t(x_t)}$$

It can be observed that the probability of the observation sequence is proportional to the forward factor of the last hidden state, the equation above can be useful in determining how well a DBN can perform for a given sequence in an MLE framework. [1]

3.3.2.2 Backward Propagation

In the backwards approach we have the conditional probability of the observations from time t+1 until time T-1 conditioned on the values of the state t, and its distribution is given by $\beta_t(x_t)$. [1]

$$\beta_t(x_t) = P(Y_{t+1}^{T-1} | \, x_t)$$

With its final value as

$$\beta_T(x_{T-1}) = 1$$

we get

$$\beta_{t-1}(x_{t-1}) = \sum_{x_t} P(x_t \mid x_{t-1})\beta_t(x_t)P(y_t \mid x_t)$$

The smoothing can be given by

$$\gamma_t(x_t) = \; P(x_t \mid Y_0^{T-1}) = \frac{\alpha_t(x_t)\beta_t(x_t)}{\sum_{x_t} \alpha_t(x_t)\, \beta_t(x_t)}$$

Where γ is the smoothing operator. [1]

4 DATA GENERATION

4.1 Base Statistics

4.1.1 Overview

As mentioned earlier, temporal data for health and fitness of a person is either virtually inexistent or that type of data is very difficult to obtain publicly. However, we do get statistical data for diseases and injuries from agencies such as the World Health Organization, the Centre of Disease Control, and other reliable sources of their fields. These data sets tend to contain population statistics (rather than individual data) and include a range of aspects of health, including injury, diseases, and traits. Similarly, basic exercise and fitness data is available that can be correlated.

4.1.2 Injury

Injury can happen in many ways, some due to negligence some due to work hazards, some just due to bad luck. According to the World Health Organization (WHO), about 5.8 million people die each year because of injuries. This accounts for 10% of the world's deaths, 32% higher than the number of fatalities that result from malaria, tuberculosis, and HIV/AIDS combined. [22]

Injuries also effect the rate of activeness of a person, which directly effects the amount of exercise a person is doing. Also, about 121 out of every 10,000 injured people get injured while doing exercise or recreational sports, thus further complicating and emphasizing the link between injury and exercise.

4.1.3 Diseases

Disease often happens when the immunity and self-repair capabilities of the human body are stressed or overwhelmed, which can be made more likely by many factors but one which stands out is age. Although during the initial growth years immunity of the body increases with the increase, it starts to drop with the advent of old age.

A disease can be as small as general fever, a mild flu, simple cold, mild cough to as grave and life changing as cancer, or heart attack, and potentially lead to death.

4.1.3.1 Heart Attacks

About 720,000 people in the U.S. suffer heart attacks each year. Of these, 515,000 are a first heart attack and 205,000 happen in people who have already had a heart attack. In 2011, about 326,200 people experienced out-of-hospital cardiac arrests in the United States. Of those treated by emergency medical services, 10.6 percent survived. Of the 19,300 bystander-witnessed out-of-hospital cardiac arrests in the same year, 31.4 percent survived. [9]

4.1.3.2 Cancer

Cancer is the second-leading cause of death among Americans. Almost one of every four deaths in the United States is due to cancer. The 2017 *United States Cancer Statistics* report indicates in 2014 (the most recent year of incidence* data available), 1,596,486 Americans received a new diagnosis of invasive cancer† and 591,686 Americans died from this disease. [9]

4.1.3.3 Influenza

According to WHO Influenza is spread by influenza virus and can vary in symptoms from mild to severe. Its symptoms include running nose, fever, sore throat, cough headache etc. There are 3 types of influenza virus, Type – A, B and C. Nearly 12% of the people catch flu in a season, with a mortality rate of 1 in 6000.

4.1.4 Traits

4.1.4.1 Obesity

A person whose weight is higher than what is considered as a normal weight adjusted for height is described as overweight or having obesity. [9]

According to the National Health and Nutrition Examination Survey (NHANES)

- More than 1 in 3 adults were overweight.
- More than 2 in 3 adults were overweight or have obesity.
- More than 1 in 3 adults were considered to have obesity.
- About 1 in 13 adults were considered to have extreme obesity.
- About 1 in 6 children and adolescents ages 2 to 19 were considered to have obesity.

Factors that may contribute to weight gain among adults and youth include genes, eating habits, physical inactivity, TV, computer, phone, and other screen time, sleep habits, medical conditions or medications, and where and how people live, including their access to healthy foods and safe places to be active. [9,11]

4.1.4.2 Smoking

According to the CDC and US Dept of Health and Human Services Cigarette smoking is the leading cause of preventable disease and death in the United States, accounting for more than 480,000 deaths every year, or about 1 in 5 deaths.

Cigarette smoking is found to be higher in the younger population, the distribution by age is as follows, [9]

- About 13 of every 100 adults aged 18–24 years (13.1%)
- Nearly 18 of every 100 adults aged 25–44 years (17.6%)
- 18 of every 100 adults aged 45–64 years (18.0%)
- Nearly 9 of every 100 adults aged 65 years and older (8.8%)

4.2 Structure and States

To translate health factors and population statistics into a Dynamic Bayesian Network that can subsequently be used to derive individual observation and state sequences that match the available data, a state and observation representation as well as a temporal transition structure must be designed that matches the available knowledge and that can be adapted using learning algorithms to produce a sample distribution that recreates the population statistics as closely as possible.

4.2.1 Introduction

Our aim is to generate the data for a population of people who have different attributes that are interlinked according to our Dynamic Bayesian Network. Figure 7 below Illustrates our DBN for a single time slice.

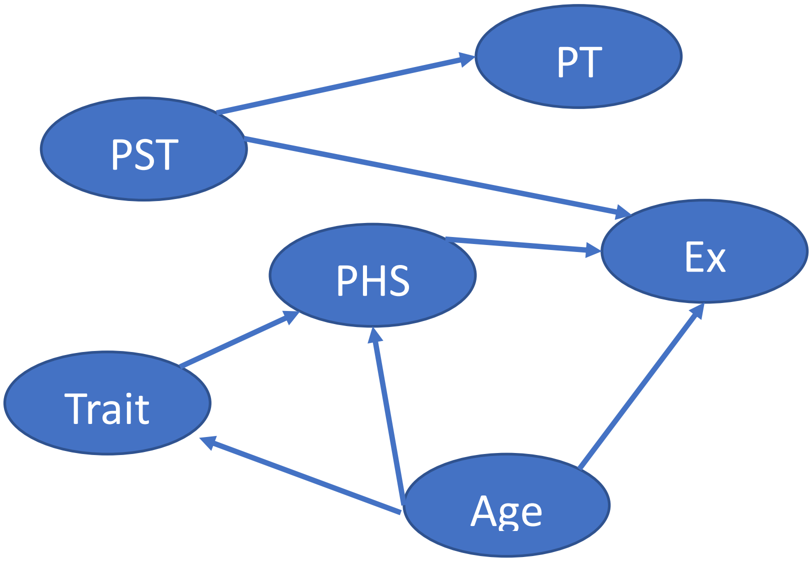


Figure 7. Time Slice of the State attributes in Our Health DBN

The different random variables mentioned Figure 7 are:

- Person-Type (PT)
- Person Sub Type (PST)
- Person Health Status (PHS)
- Exercise (Ex)
- Age
- Traits

4.2.1.1 *Person Type*

The random variable Person Type can take on 3 values, namely Active, Sedentary and Obese and states the type of the person at a given time. The possible temporal transitions between these values can be seen by the following figure

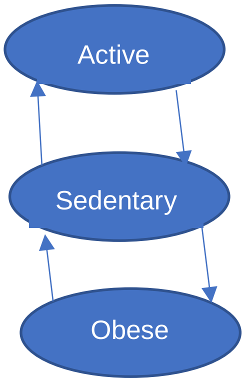


Figure 8. Person Type Values and Transitions between them.

Clearly it can be seen that the person cannot jump from Active to Obese and vice versa directly, since an active person cannot suddenly become sedentary. Therefore, we have another random variable, Person Sub Type, so that the population is not just defined by 3 values, but rather a set of many values.

4.2.1.2 *Person Sub Type*

As the name suggests it is a sub category of person type and has values between 0-99, this random variable directly influences exercise in the current time step and gets influenced by the exercise done in the previous time step. It directly influences Person Type as Person type will transition to a higher level or lower level based on its value.

An example for this is can be observed as a Person whose current Person Type is Sedentary and Person Subtype is 97. If, for example, while exercising for that time step the person observes a change of +5 in its Person Subtype value, then instead of having a 102 as the final value, we would roll the value to 02 and Change the person type to Active.

An important thing to note is that if the <Person-Type, Person Subtype> has values <Active, 97> and then, they get a +5 increase in subtype, in that case we restrict a roll over and value stays at maximum of 99, i.e. <Active, 99>; the same goes for a person who is <Obese, 3> and we observe a -5 decrease In the person subtype; it will not go below <Obese, 0>. This use of a type and numeric sub type allows for a fine grained and more realistic evolution and simulation of an individual, considering more accurately the actual effects of exercise and treatment.

4.2.1.3 Person Health Status

Person Health Status not only provides the current health state of a person, which can be one of the followings:

- Active
- Normal
- Recovering
- Injured
- Ill
- Deceased

Along with the health status it also provides key observations for us, such as Injuries, accidents, diseases. The transitions of different values of this variable are,

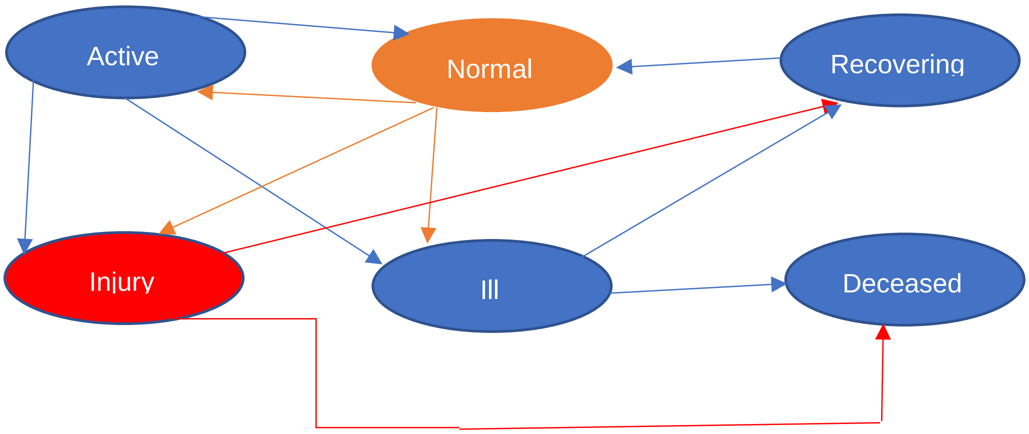


Figure 9. Person Health Status Internal Transitions.

It can be observed from Figure 9 that Injury and Illness can happen when the person is in an Active state or a Normal state. It is assumed in our case that the person cannot get injured or sick again when the person is in a Recovering state. The person can be Deceased only through Illness or Injury. The simulation for the Deceased person stops at the time step when the value of person health status becomes Deceased.

4.2.1.4 Exercise

Exercise just represents a relative amount of exercise based on the pervious time step's exercise, and the current time step's Age, Person Health Status and Person Subtype, i.e. the random variable just contains the probability of doing more exercise, or less exercise, based on the previous state's value.

4.2.1.5 Age

Age is a linearly increasing value of a person and puts them in their respective categories based on how much they have aged. This variable allows to model the effects of age on the other factors, such as exercise and disease. For example, a 50-year simulation of a teenager should put him in senior citizens category, and then the person should be susceptible to the probabilities pertaining to that category.

4.2.1.6 Traits

Traits are special features which can boost criteria for certain events for certain individuals. For example, a person is twice being like to get cancer if the person smokes than a person who does not.

4.2.2 Structure

4.2.2.1 Unrolled Dynamic Bayesian Network

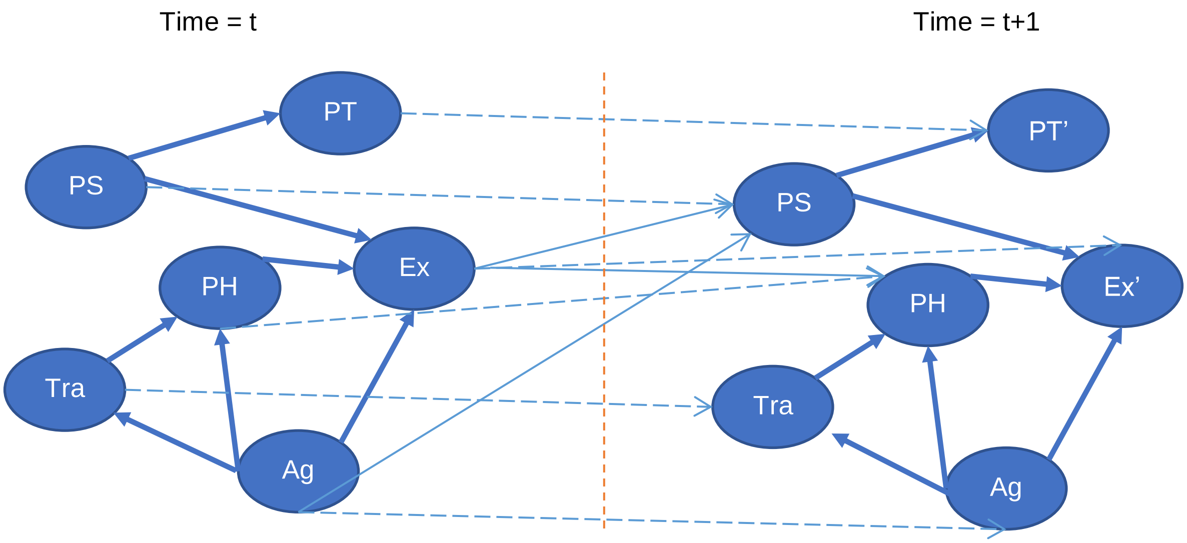


Figure 10: An Unrolled DBN with Variable in States at Time t and t+1

As discussed previously each random variable at a given time t can be inferred using its parents in the current state and in the state at time t-1. An example derived from the structure in Figure 10 is as follows:

$$P\ (PHS') = P(PHS' \mid Age', Traits', PHS)$$

4.2.3 Data Structures

To implement the DBN illustrated above, the data structures for the network must be fleshed in. In this work, this is divided into personal, fitness, illness, and calendar data.

4.2.3.1 Person Data

A person's data contains the information about each person in our population data:

Person Data Structure	
Attribute	**Comments**
Id	Unique Identification of a person
Type	Corresponds to the observed value of person type
Subtype	Contains information of the current value of person sub type variable.
DOB	Date of Birth of a person, to be increased by 1/52 for each time step (since our time step is 1 week).
Health Status	Maintains the current health state a person is, it can take values as mentioned in Section 4.2.1.3
DOD	Holds the date when Person Health Status became Diseased, remains Null otherwise.
Traits	Special features of a person are reflected in the attribute, a person can have zero or multiple traits.
Average	Contains a list of averages such as average sleep time, average active time, average inactive heart rate, average active heart rate, these averages help in inferring future values of the person's trajectory.
Fitness Data	Contains a list of exercises for a person.
Illness Data	Contains a list of Illness and Injuries of person over time.

Table 1: Attributes of the Person Data Structure

4.2.3.2 Fitness Data

Each fitness data point corresponds to an activity done by a person.

Fitness Data Structure	
Attribute	**Comments**
Id	Unique Identifier for an exercise done by a person in a week.
Person Id	Belongs to the person who has performed this exercise.
Exercise Type	Contains information about the exercise itself that was performed.
Exercise Rate	Is a parameter that gives us a multiplier which is used to modify person subtype.
Heart Rate Active	Maintains an active heart rate while the exercise is performed

Table 2: Attributes of the Fitness Data Structure

4.2.3.3 Illness and Injury Data

Each entry denotes any injury or illness incurred by a person.

Illness Data Structure	
Attribute	**Comments**
Id	Unique Identification for an injury or illness that happened to a person
Person Id	Belongs to the person who has performed this illness or injury.
Illness Type	Contains information about the Illness or injury itself.
Recovery Rate	Recovery Rate determines how fast the person can recover and move from Person.HealthStatus.Recovery to Person.HealthStatus.Normal as can be seen in Figure 9.
Recovery Time	Defines the time needed for a person to recover from the ailment.

Table 3: Attributes of the Illness Data Structure

4.2.3.4 Calendar Data

This data accounts represents a summary for all the activity that is done in the population and maintains a sparsity vector that avoids people getting stuck in cycles to maintain overall population statistics.

Calendar Data Structure	
Attribute	**Comments**
Id	Unique Identification for calendar entry, equivalent to week number,
Fitness Id	Contains fitness id for that person
Illness Id	Contains illness id for that person
Illness this week	Number of people this week getting injured or sick for each type of Illness.

Table 4: Attributes of the Calendar Data Structure

4.3 Inference

Inference on our DBN is performed using the forward and backward algorithm. For a population over increasing intervals of time, 1 year, then 2 years, then 3, and so on, this is done in terms of weeks, so we do 52 iterations of forward pass, then we do 52 iterations of backward pass. The choice of 1 week as the step size was chosen as a compromise between the level of detail provided for an individual and the computational complexity of doing inference to generate extended duration individual trajectories while maintaining overall population statistics.

4.4 Initial Sampling

To generate “individuals” that match the intended population, a population is created based on the data structures discussed above, with random initial values for time step t = 0, except for the variable age, which is defined by the random DOB given to a person. The random initial values for the other attributes are drawn according to the appropriate population statistics for the specific age.

4.5 Forward Algorithm

The belief of each variable at each time step is calculated according to the values of their observed variables in the previous state and the conditional probability distribution table of that variable. The belief state can be calculated at each time step, but doing this does not, in a strict sense, produce the most likely state *sequence*, but rather the most likely state at each time step, given the previous history. (Toman, 2014) [13] This distinction is important mainly when analyzing a given data sequence in the context of observed values (e.g. the population statistics).

4.6 Importance Sampling

Importance sampling uses samples generated from a distribution to estimate properties of another distribution. It can be used in simulations done using Monte Carlo methods to reduce variance. Where preference is given to certain values of the input random variable over others with the goal that these preferred values are generated more often than others, thereby reducing the variance. The tricky part in importance sampling is to have a biased distribution that prefers key regions of the input variables. [24]

Designing a good importance sampling distribution becomes exponentially difficult with the increase in the complexity of the model. However, the idea remains the same. So, for a random variable X with p(x) as its pdf, then to calculate

$$\mu f = Ep[w(X)f(X)]$$

With

$$\mu f = \int f(x)p(x)dx$$

and for a sampling probability density q(x), where q(x) > 0 if f(x)p(x) ≠ 0, we have

$$\mu f = E_q[w(X)f(X)]$$

Here w(x) is the ratio of p(x) and q(x) and E_q[] is the expectation w.r.t q(x). So, a sample of independent draws of $x_1 \ldots x_n$ from q(x) can be estimated by [25]

$$\mu f' = \frac{1}{m}\sum_{i=1}^{m} w(x(i))f(x(i))$$

4.7 Learning

Given our population statistics, internal variable constraints, and the defined structure of Dynamic Bayesian network, we learn our parameters by adjusting the CPD's such that after sufficient training, the data generated by our network, follows the guidelines of our population statistics.

Here we learn conditional probability distribution tables for each of the nodes in our Dynamic Bayesian network at time t and at time t+1. A *conditional probability distribution* Table (CPD) is a concise version of a joint probability distribution table. The CPD of a variable X_1 at time t, contains all the possible values the variable can take given all its observations of the variables that it conditionally depends on

CPDs can be visualized as a 2d array where rows represent cardinality of the variable, and columns represent the product of cardinality of all the observed variables. Let us take the following example

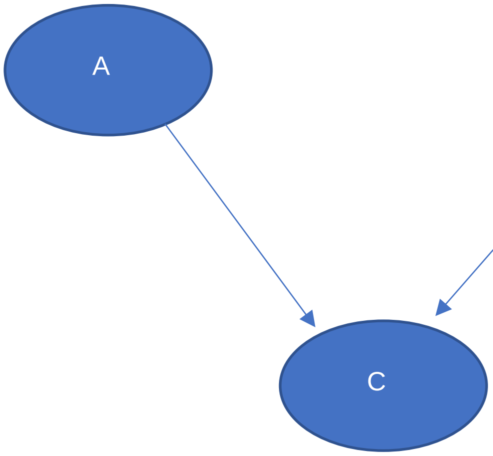


Figure 11: A simple Bayesian Network.

Given that variable A can have 3 values, B can have 4 values, and C can have 2 values, the CPD for variable C will form a matrix with dimensions as (2,12). That is a matrix with 2 rows and 12 columns. A sample representation of CPD can be seen in the table below

	A_1	A_1	A_1	A_1	A_2	A_2	A_2	A_2	A_3	A_3	A_3	A_3
	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4	B_1	B_2	B_3	B_4
C_1	**0.1**	**0.3**	**0.3**	**0.2**	**0.4**	**0.5**	**0.1**	**0.3**	**0.3**	**0.2**	**0.4**	**0.5**
C_2	**0.9**	**0.7**	**0.7**	**0.8**	**0.6**	**0.5**	**0.9**	**0.7**	**0.7**	**0.8**	**0.6**	**0.5**

Table 5: A Sample Conditional Probability Table

4.7.1 Hill Climbing

Hill Climbing is used for optimization problems that are computationally hard to solve and is best suited for situations where the state description itself has all the information required to find a solution. Hill Climbing does not need to maintain a search tree as it looks only at the current state to determine the next state. An evaluation function is used to iteratively improve the current state. It considers all possible states and measures them using the evaluation function of the state at that point. [23]

A major roadblock in hill climbing is local maxima of the functions which indicates to the algorithm that there are no more states nearby that are as good as that local maximum. To avoid getting stuck in a local maximum we can implement methods like simulated annealing and random restart hill climbing.

The most generic form of the hill climbing algorithm involves evaluating all neighbors of a given point and greedily selecting the point which results in the highest value of the function we are maximizing. The choice of neighbors is defined by the temperature factor, which is used to "jitter" the variables with some controlled variance. Hill climbing algorithms are typically used in scenarios where no inherent gradient is observable in the system. In such cases, only a few noisy or sparse, indirect evaluation functions are available which can be used to derive the step direction of each variable by searching locally in a greedy fashion.

The temperature value defines the amount of random noise to be added to a given variable in a system. There is an apparent trade-off induced by this variable. In situations where the search space is stuck in a plateau or possible a small local optimum, a high temperature value is more likely to get unstuck. However, lowering the temperature allows the system to fine tune the results.

In this work, hill climbing is used to optimize our constraint functions in terms of our defined probabilities. Thus, it finds probabilities that maximize the separate predefined, constrained statistics evaluation functions.

The following brief pseudo code describes the basic operation of the hill climbing scheme:

Hill Climbing Pseudo code:

```
current Node = start Node;
temperature=tau
loop do
   neighbor list = find neighbors (current Node, temperature);
   next Node=argmax (neighbor list)
   if EVAL (next Node) <= EVAL (current Node)
      return current Node;
   current Node = next Node;
```

4.8 Results

The following example is a miniaturized version of our model, shown here as a proof of concept. The DBN of the example looks like the figure below.

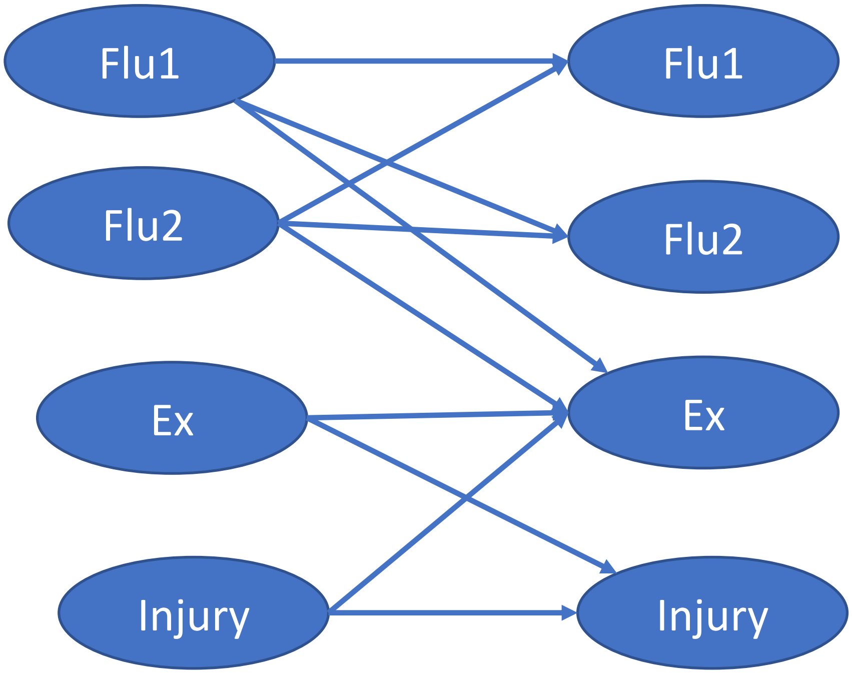


Figure 12: An example Dynamic Bayesian Network

Here, each random variable is a Boolean variable, each having a cardinality of two. To perform hill climbing to learn parameters, we need to define the performance metric to ascend on. In this case, we can at each time step during training consider the similarity of the properties of a set of generated individual trajectories from the given population statistics. The metric to optimize the sum of all the similarities throughout the duration of the generated trajectories. Considering this akin to a Reinforcement Learning problem, we could consider this similarity at each step the reward, and the change in transitions as the actions (in this case corresponding to virtual action that maintain or change the subsequent values, such as performing exercise, taking medication, etc.). For each step during generation, we perform the "actions" according to the current probabilities and then obtain a "reward" corresponding to the appropriateness of the generated state values in the context of the available population statistics.

In this example we have five additional variables that have fixed transition structures but influence state transitions, these variables represent additional constraints on the behavior of the system:

- *Flu2_History*: It imposes a constraint on Flu2, such that P(Flu1 **|** Flu2_History) in the same year is zero, i.e. if a person had flu2 sometime earlier in a year he will not have flu1.
- *Flu1_History*: Maintains the fact that a person got sick with flu1 in the current year.
- *Week*: This is a discretized slice of time in our DBN, and it helps maintain seasons (Spring, Summer, Fall, Winter).
- *Average_Week_Activity_Level*: It helps to maintain the amount of time person choose to do exercise in a given week.
- *Injury_history*: It allows the network to know that the person suffered from the injury in previous week, which would affect his choice in choosing this week's observations.

Our observation variables as shown in Figure 12 are Flu1, Flu2, Injury and Exercise. For these variables, several statistical relations are used for training that relate to their efficacy and frequency as indicated below:

4.8.1 Flu1

It is a type of sickness that, although milder than Flu2, causes havoc during the outdoorsy days of sunshine during summer. A person upon getting this flu, sees a drop in their exercise routine for the next week.

4.8.2 Flu2

This sickness is a bit more severe than Flu1, i.e. its statistics indicate longer duration and more sever reduction in exercise. It also is such that a recipient of this sickness is temporarily immune to Flu1.

4.8.3 Injury

Injury in this example corresponds to exercise related hazard, hence on observing that a person suffered an injury, a drop in their exercise can also be observed. Injury is saved for the state at time t into the Injury_History state variable and is used to make decisions for next state.

4.8.4 Exercise

Exercise is an observation variable which gives us the information about how many times a person chooses this action. To differentiate between an active person from an obese person, a probabilistic policy is implemented. The person is more likely to do more exercise given he did more exercise in last time step (for active person type), and less exercise if he did less last time (for obese person type). The exercise takes a dive if a person is unwell, reduced exercise also reduces chances of getting injured due to too much exercise.

4.8.5 Graphs

The following graphs show sample learning curves for model training, illustrating that the hill climbing approach can successfully increase the performance of the model a represented by an increase in the similarity (reward) of the generated transition sequences with the underlying quantitative statistics. In these cases, the target of the training is a population of highly active population.

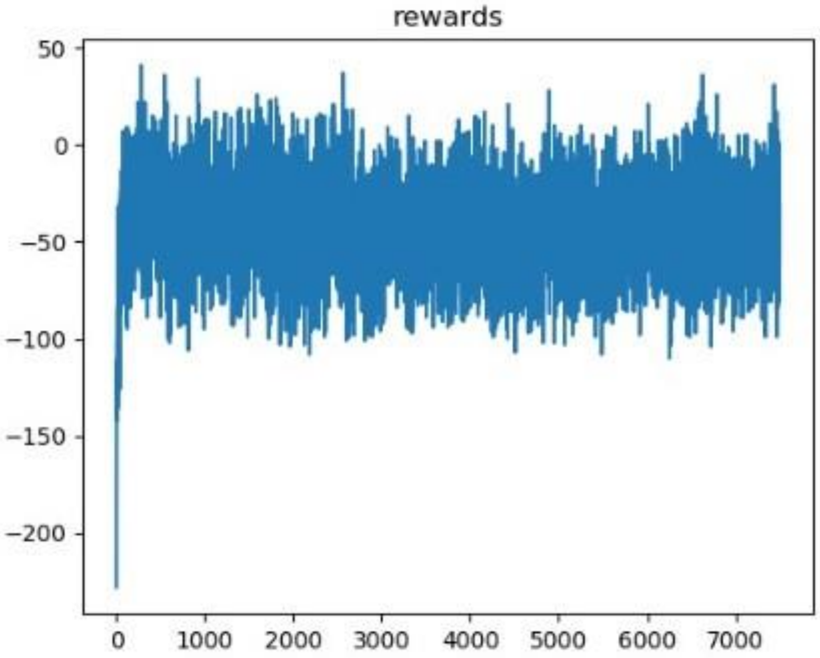


Figure 13: Learning Curves for a moderately active person

Once model parameters are learned, these can be used to generate individual data sequences which correspond loosely to samples of individuals in the population. The following figures show three examples of generated persons in terms of important variable attributes (Flu 1, Flu 2, Activity, and Injury). Examples shown here loosely correspond to samples picked from trained networks to correspond to types of individuals. In each of the examples, the X axis represents a year with three events per week for a person, while the Y axes indicate the state of the corresponding variable at that time.

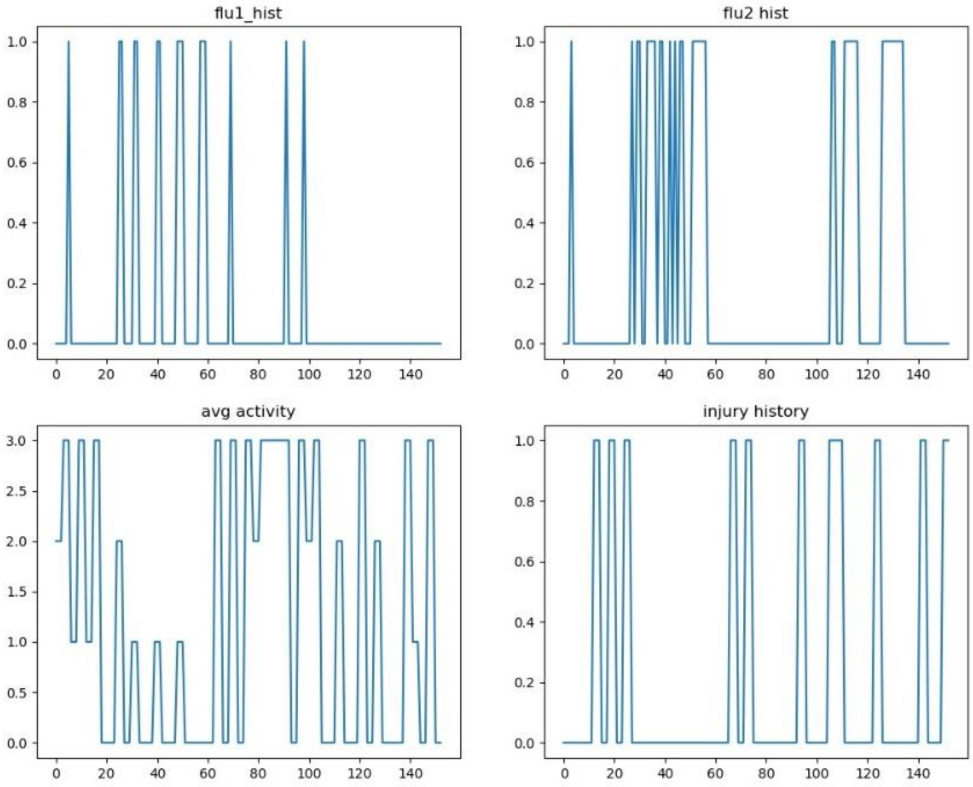


Figure 14. Observations of a moderately active person

Here the X axis represents a year with three events per week for a person the graph of flu1_hist, displays the amount of time person got flu, and it can be easily observed that the person got flu’s during summer a lot more than any other time. Also, it can be observed that when he got flu2, his exercise levels dropped considerably, considering that the person was not well enough to either exercise or do any sort of extensive, physical activity.

The following graph shows how different types of sickness, both flu1 and flu2 affects a person's exercise habits over the course of a year. It can be seen, that the person, reduced their activity level once they are not well.

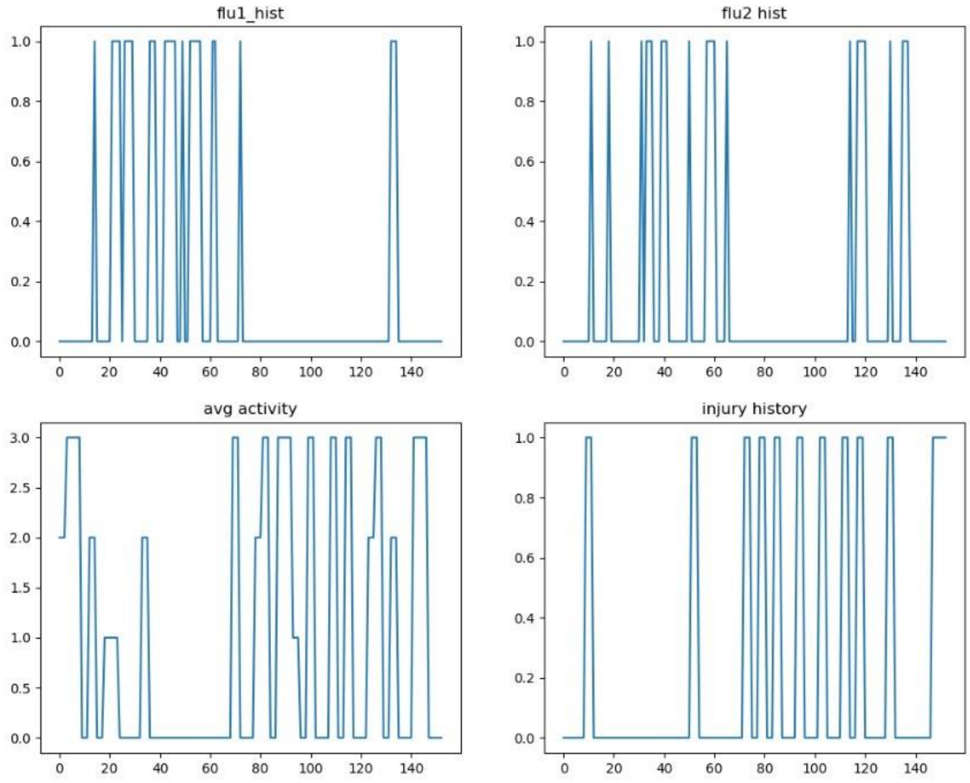


Figure 15: Observations for person who was not well for quite some time.

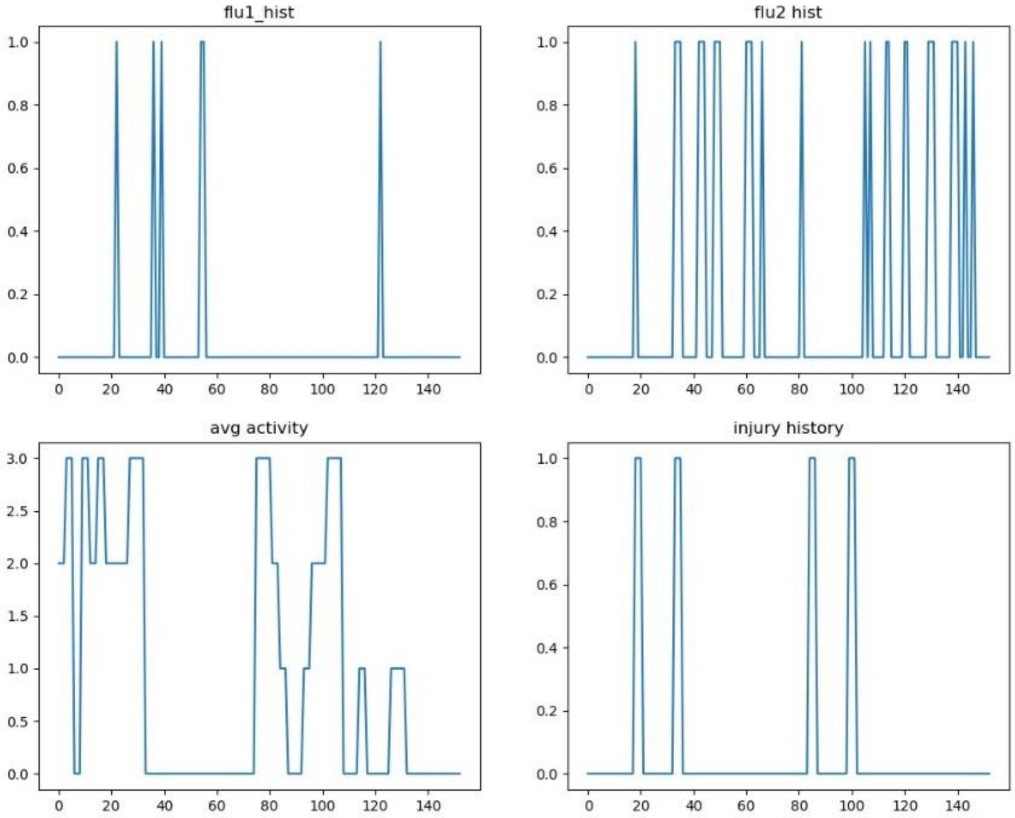


Figure 16: Observations of a sedentary person

Here the X axis represents a year with three events per week for a person

The above graph depicts the effects of flu2 on flu1 and the effects of flu2 and flu1 on exercise, for a person who does seldom exercise and as a result they are less susceptible to any injury. Also, we can see that a person on getting sick with flu2, reduces their chance of getting flu1 along with It can clearly depict that the injury occurs on doing extensive high intensity exercise.

The below graph displays the learning curve for a person, where it can be observed that the jumps get smaller and smaller over time, as we start to learn more and more about the parameters.

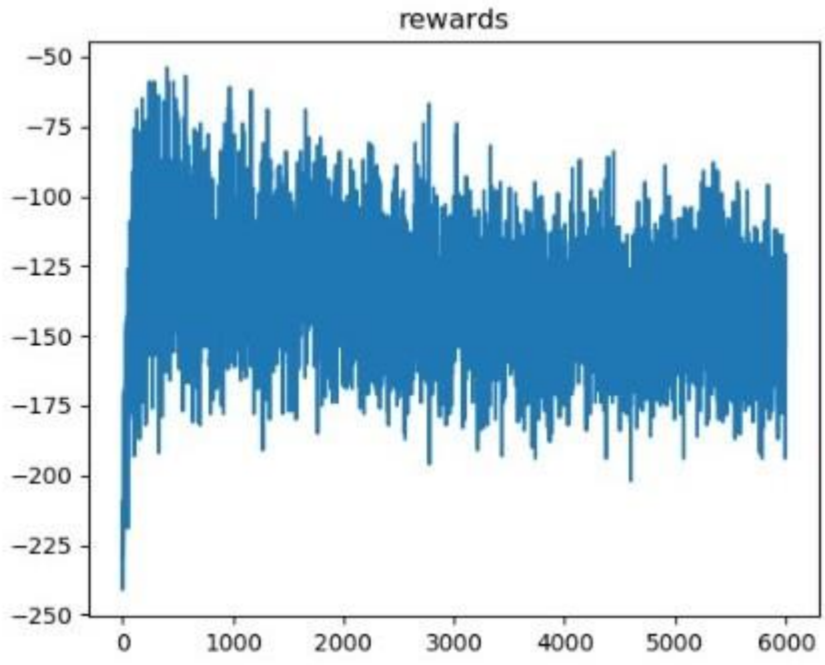


Figure 17: Shows the learning curve for the data generated for a sedentary person.

5 FUTURE WORK AND CONCLUSION

5.1 Conclusion

A dynamic Bayesian network which learns the parameters to generate a temporal data has been implemented in this thesis. The model is still in its infant stage and can be scaled and learned more efficiently going forward.

The main motivation behind this was to have a system which can generated reasonably realistic individual health data trajectories with which different types of research can be done without relying on insurance firms and hospitals which are reluctant in sharing their data or which have data which is in a format too sensitive to be made public. To be realistic, the proposed approach combines general health expert knowledge regarding the relations between different health and lifestyle attributes (encoded into qualitative structural attributes of the DBN), with available population statistics which are translated through the learning framework into appropriate generative parameters (represented as quantitative parameters inside the model). The goal here is to ensure that the individual data sequences generated using the DBN as closely as possible recreate the known population statistics. Using this generated data, it should then be possible to perform initial tests and evaluations of machine learning based analysis and intervention algorithms as to their ability to operate on individual health sequence data. While the resulting learned parameters might not be completely correct for real human data, the basic characteristics of the algorithms should be reflected on the simulated data due to its incorporation of expert knowledge and its conformity with basic real-world statistics.

Below is some immediate future work that can extend the applicability of this approach.

5.2 Improving the Network

The network needs to have an even better learning algorithm which takes less computation and learns faster.

5.3 Features and Traits

More features to the network can be added, so that the Dynamic Bayesian Network can grow and can generate more parameters than before.

5.4 Collaborative Filtering on Generated Data

A collaborative filtering approach can be implemented on the data that is generated by our DBN to identify similar people on various trends we get from the data itself.

6 References

1. Dynamic Bayesian Networks: A State of Art, V. Mihajlovic, M. PetKovic
2. Learning Dynamic Bayesian Networks, Zoubin Ghahramani
3. Dynamic Bayesian Networks: Representation, Inference and Learning, K. Murphy
4. Ben-Gal I., Bayesian Networks, in Ruggeri F., Faltin F. & Kenett R., Encyclopedia of Statistics in Quality & Reliability, Wiley & Sons (2007)
5. https://iknowfirst.com/stock-market-algorithm-what-if-analysis-response-modeling
6. Speech and Language Processing. Daniel Jurafsky & James H. Martin.
7. Exploring Dynamic Bayesian Belief Networks for Intelligent Fault Management Systems R. Sterritt, A.H. Marshall, C.M. Shapcott, S.I. McClean
8. http://library.bayesia.com/display/BlabC/Dynamic+Bayesian+Networks
9. Center for Disease Control
10. https://gis.cdc.gov/grasp/fluview/flu_by_age_virus.html
11. National Heart, Lung, and Blood Institute, National Institutes of Health. Clinical guidelines on the identification, evaluation, and treatment of overweight and obesity in adults: the evidence report. www.nhlbi.nih.gov/health-pro/guidelines/archive/clinical-guidelines-obesity-adults-evidence-report . Published September 1998. Accessed July 25, 2017.
12. National Health Expenditures 2016 https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/NationalHealthExpendData/downloads/highlights.pdf
13. Toman, A. Stat Papers (2014) 55: 233. https://doi.org/10.1007/s00362-013-0525-y
14. Improving the synthetic data generation process in spatial microsimulation models Dianna M Smith, Graham P Clarke, Kirk Harland

15. Markov Decision Evolutionary Games with Time Average Expected Fitness Criterion. Eitan Altman

16. Collaborative Filtering for Implicit Feedback Datasets Yifan Hu AT&T Labs – Research Florham Park, NJ 07932 Yehuda Koren∗ Yahoo! Research Haifa 31905, Israel Chris Volinsky AT&T Labs – Research Florham Park, NJ 07932

17. Collaborative Filtering Recommender Systems - By Michael D. Ekstrand, John T. Riedl and Joseph A. Konstan

18. Content-Boosted Collaborative Filtering for Improved Recommendations - Prem Melville and Raymond J. Mooney and Ramadass Nagarajan

19. Amazon.com recommendation - Item-to- item collaborative filtering - by Linden, G; Smith, B; York, J

20. Collaborative Filtering Recommender Systems - by Mehrbakhsh Nilashi; Karamollah Bagherifard; Othman Ibrahim;

21. Sequence-Learning Algorithm Based on Backward Chaining, Sanjay S. Joshi, Benoit Guilhabert

22. http://www.who.int/violence_injury_prevention/key_facts/VIP_key_facts.pdf

23. Russell, S. J., & Norvig, P. (2004). *Artificial Intelligence: A Modern Approach*. Upper Saddle River, NJ: Prentice Hall.

24. Rubinstein, R. Y., & Kroese, D. P. (2011). Simulation and the Monte Carlo method (Vol. 707). John Wiley & Sons

25. Importance Sampling: A Review Surya T Tokdar and Robert E Kass http://www2.stat.duke.edu/~st118/Publication/impsamp.pdf

26. Capp´e, O., Guillin, A., Marin, J. M. and Robert, C. P. (2004). Population Monte Carlo. Journal of Computational and Graphical Statistics (13) 907-929

Biographical Information

Mohammed Azmat Qureshi completed his bachelor's degree Computer Science and Engineering from Visvesvaraya Technological University in Belgaum, Karnataka, India in 2011. He worked as a software developer in Dell from 2011 – 2014. He started his master's Degree in Computer Science at University of Texas at Arlington in 2014 specializing in Artificial Intelligence. His interests include Statistics, Machine Learning, Data Mining, Deep Learning, Mobile Computing and Software Engineering.