Forecasting of Self-Rated Health Using Hidden Markov Algorithm

J E S P E R L O S O

Master of Science Thesis
Stockholm, Sweden 2014

Forecasting of Self-Rated Health Using Hidden Markov Algorithm

J E S P E R L O S O

Master's Thesis in Mathematical Statistics (30 ECTS credits)
Master Programme in Mathematics (120 credits)
Royal Institute of Technology year 2014
Supervisor at KTH was Timo Koski
Examiner was Timo Koski

TRITA-MAT-E 2014:17
ISRN-KTH/MAT/E--14/17-SE

Royal Institute of Technology
School of Engineering Sciences

KTH SCI
SE-100 44 Stockholm, Sweden

URL: www.kth.se/sci

Abstract

In this thesis a model for predicting a person's monthly average of self-rated health the following month was developed. It was based on statistics from a form constructed by HealthWatch. The model used is a Hidden Markov Algorithm based on Hidden Markov Models where the hidden part is the future value of self-rated health. The emissions were based on five of the eleven questions that make the HealthWatch form. The questions are answered on a scale from zero to one hundred. The model predicts in which of three intervals of SRH the responder most likely will answer on average during the following month. The final model has an accuracy of 80 %.

Contents

1 Introduction

In this thesis Hidden Markov Models, from here on referred to as HMMs or HMM, are used in order to make an algorithm for prediction of a person's one month average of their self-rated health the following month. The model is based on five of the eleven questions that constitute the HealthWatch form. More about HealthWatch can be found in [1]. The HealthWatch form can be accessed by anyone who is willing to create an account at www.healthwatch.se. The responder will answer on a seemingly continuous scale between qualitative measures such as very bad and very good or, very high and very low. The responder cannot see that the scale ranges from zero to one hundred. The numbers between zero and one hundred are the data used to quantify the problem of predicting self-rated health. The questions that are used to create the model are:

How do you feel right now?	SRH (self-rated health)
Do you have control over your life right now?	Control
How efficient are you at work right now?	Efficiency
How is your job satisfaction right now?	Workjoy
How high is your work load right now?	Workload

Table 1: The questions used to construct the model.

The other questions that were not used to construct the model are stated in the table below. The scale is the same as mentioned previously.

How satisfied are you with your social life right now?
How is the job atmosphere right now?
What is your energy level right now?
How stressed do you feel right now?
How is your ability to concentrate right now?
How did you sleep last night?

Table 2: The other questions in the HealthWatch form.

These questions were not used because of impracticalities. The model based on all questions is too complicated. This leads to long computation time which is impractical. The complicity also makes the final prediction model less accurate since the data available is not enough the describe such a complicated model. These problems will be explained later in the report.

This thesis will describe theory about Markov chains in section 2.1, theory about HMMs in section 2.2 and the performance measurements used in section 2.3. After section 2 the approach to the problem including how HMMs are used, how the data was used, the prediction model and the practical implementation of the model is described. This can be found in section 3.1, 3.2, 3.4 and 3.5 respectively. Finally the performance of the model, e.g. the accuracy and

precision, and some conclusions drawn from the results can be found in section 4 and 5.

2 Theoretical background

2.1 Markov Chains Briefly

There exist different kinds of Markov processes. In this section a description of the Markov process used in this report will be described, namely a discrete-time Markov chain. Other types of Markov processes are neglected. More information regarding Markov processes can be found in [2].

A discrete Markov process can move between different states, creating a chain of events. The possible states that can occur all belong to the finite set of N distinct states, S_1, S_2,...,S_N. The Markov process will jump from one state to another, or jump to itself, at regularly spaced discrete time points ($t = 1, 2, ...$) with some probability associated with the state jumped from and the state jumped to. The actual state at time t will from here on be denoted as q_t. An illustration of a Markov process can be seen below in figure 1.

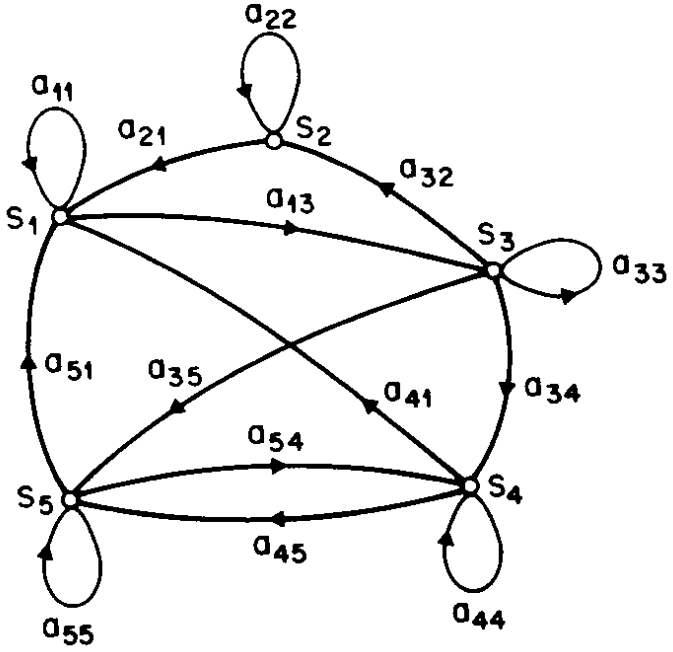

Figure 1: A Markov chain with five states (labeled S_1 to S_5) with selected state transitions [3].

In equation (1) below the probabilistic definition of a Markov chain with a current state and a predecessor state is defined.

$$P(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k, ...) =$$

$$P(q_t = S_j | q_{t-1} = S_i). \tag{1}$$

The Markov process used in this thesis is independent of time, hence the transition probabilities can be defined as

$$a_{ij} = P(q_t = S_j | q_{t-1} = S_i),\ 1 \leq i, j \leq N, \tag{2}$$

where

$$\sum_{j=1}^{N} a_{ij} = 1,\ a_{ij} \geq 0. \tag{3}$$

The transition probabilities are usually displayed in a matrix, the transition matrix, where a_{ij} can be found on row i, column j. The probability of the occurrence state $j = 3$ given the state that occurred last $i = 1$ is a_{13}. This can be found in the transition matrix, illustrated below.

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}.$$

2.2 Hidden Markov Models

A Hidden Markov Model contains a Markov chain whose states themselves cannot be observed, therefore these states are called hidden. However, each hidden state emits something that can be observed. HMMs are commonly used in speech recognition applications. More details on theory and applications with HMMs can be found in [3]. Below follows a definition of Hidden Markov Models. The definition is an interpretation of the definition in [3].

A HMM has the following six characteristics:

1. The number of states in the model, N. The states are denoted as S=$\{S_1, S_2, ..., S_N\}$.
2. The number of distinct observation symbols per state, M. The symbols are denoted as V=$\{v_1, v_2, ..., v_M\}$.
3. The state transition probability distribution $A = \{a_{ij}\}$ where a_{ij} is the same as in equation (2).
4. The observation symbol, i.e. the emission, probability distribution in state j, $B = \{b_j(k)\}$, where
$$b_j(k) = P(O_t = v_k | q_t = S_j),\ 1 \leq j \leq N,\ 1 \leq k \leq M. \tag{4}$$
5. The initial state distribution $\pi = \{\pi_i\}$.

6. O_i:s are conditionally independent given $q_1, ..., q_T$.

With this definition an observed sequence is

$$O = O_1, O_2, ..., O_T, \tag{5}$$

where each O_t is a symbol corresponding to one of the entries in V, see [2]. From here on the model that represents the HMM will be referred to as

$$\lambda = (A, B, \pi). \tag{6}$$

2.2.1 Transition- and emission-matrices

In this thesis A and B, mentioned in section 2.2, the transition- and the emission-matrices are unknown. In order to find good values for A and B the maximum likelihood of the matrices given the known emissions and the transitions is calculated.

First the probability

$$P(O|\lambda), \tag{7}$$

is to be calculated. That is, the probability of the observed sequence of symbols given the model λ. The most straight forward way of doing this is to enumerate the different possible state sequences of length T. First we consider one such sequence:

$$Q = q_1 q_2 ... q_T. \tag{8}$$

Here q_1 is the initial state. The probability of an observational sequence given the state sequence stated above can be calculated as:

$$P(O|Q, \lambda) = \prod_{t=1}^{T} P(O_t|q_t, \lambda). \tag{9}$$

It is assumed statistical independence of observations given the state sequence, hence

$$P(O|Q, \lambda) = b_{q_1}(O_1) \cdot b_{q_2}(O_2) \cdots b_{q_T}(O_T). \tag{10}$$

Using the definitions stated in equation (2) and q_1 as the initial state with probability π, the probability of the state sequence Q given the model λ can be written as

$$P(Q|\lambda) = \pi_{q_1} \cdot a_{q_1 q_2} \cdot a_{q_2 q_3} \cdots a_{q_{T-1} q_T}. \tag{11}$$

Now the joint probability of Q and O is to be calculated. This will give the probability that the symbols are observed and that the states have occured simultaneously. Simply

$$P(O, Q|\lambda) = P(O|Q, \lambda) P(Q, \lambda). \tag{12}$$

In our subsequent, equation (12) is used to construct the model for prediction of self-rated health. Given the data, equation (12) is maximized to find the maximum likelihood values for the transition- and emission matrices A and B. The maximum likelihood was calculated in MATLAB using the function hmmestimate. The function hmmestimate demands that both O and Q are known, which is the case in this report.

The built in function in MATLAB called hmmestimate calculates the maximum likelihood values for the transition- and the emission matrices. It does so by simply counting the number of occurrences for a specific transition and emission. A transition from X to Y occurs n times out of a total of N transitions from state X to some other state, thus the probability of the transition X to Y is $p = \frac{n}{N}$. For the emissions hmmestimate counts how many of an particular emission that is emitted from a particular state. An emission E from the state Y occurs m times out of a total of M different emissions from the state Y. Then the probability that emission E is emitted from state Y is $p = \frac{m}{M}$.

The function has an option of including pseudo-emissions and pseudo-transitions. This option is used in the model in order to deal with the problem of probabilities equal to zero. The option makes it possible for the user to assign a least probability to every instance of the transition- and emission matrices respectively. The implementation and theory behind the pseudo-counts is described further in section 3.2.2.

2.3 Performance of the model

The parameters used to measure the performance of the model constructed for this thesis are accuracy and precision.

Accuracy

Accuracy is a measurement on how often the model predicts the correct value. For future reference; accuracy measures how often the model predicts the true hidden state of a responder, more precisely the monthly average of a responders self-rated health in the following month. It is calculated as in equation (13) below.

$$\text{accuracy} = \frac{\text{number of true predicted states}}{\text{total number of tested objects}}. \tag{13}$$

Precision

The precision of the model is a measurement of how concise the model is. If the model predicts in the same way in two different cases with similar input parameters it has high precision. If the model has high precision it predicts the same outcome for two individuals who are feeling in the same way, based on the HealthWatch questions. In this thesis precision was measured on each of the three possible states. The calculation can be found below in equation (14).

$$\text{precision} = \frac{\text{number of true predicted state Xs}}{\text{number of true predicted state Xs} + \text{false predicted state Xs}}. \tag{14}$$

3 Method and Data

The objective with this thesis is to use Hidden Markov Models (HMMs) to make an algorithm that forecasts how a responder rates its own health on average the following month. Or more precisely; how a responder will answer on the question "How do you feel right now?" on average during the following month. The question comes from the HealthWatch form seen in table 1.

The algorithm was trained by a training set that was randomly generated by drawing with uniform distribution from a larger totally anonymized set. The remaining part of the larger set was then used as a test set. Here the larger set is the usable part of the raw data. The usable part of the data is the part where the condition of a least amount of consecutive measurements is met; this is described in further detail below. When the algorithm's parameters, that resembles the parameters of an HMM, had been estimated it was tested on the test set, which was independent of the algorithm by construction, to see how well it performed.

3.1 Model: The definition of the hidden state

To attain the objective, the problem had to be formulated to fit the description of an HMM. A similar problem was solved in [7] where a stocks value the following day was predicted by modeling it as an HMM. To meet the requirements of an HMM the problem must consist of some hidden state and some observable emissions coming from the hidden state [3]. The hidden state, the state that cannot be observed, was defined as the monthly average of the answer to the question "How do you feel right now?" during the following month. The hidden state was thus the monthly average of the future outcome of that question. The observed variables were chosen as the answer today of the five questions seen in table 1.

To be able to use the mathematics of an HMM in our model the emission cannot be the answer to all five questions in table 1 separately since they have different influence on a responders future SRH. E.g., a low value in one question can have a good effect on SRH while a low value in another question can have bad effect. One answer, on one question would therefore be favorable for the model. This would however make the model very limited. The solution to this problem is explained in section 3.2.

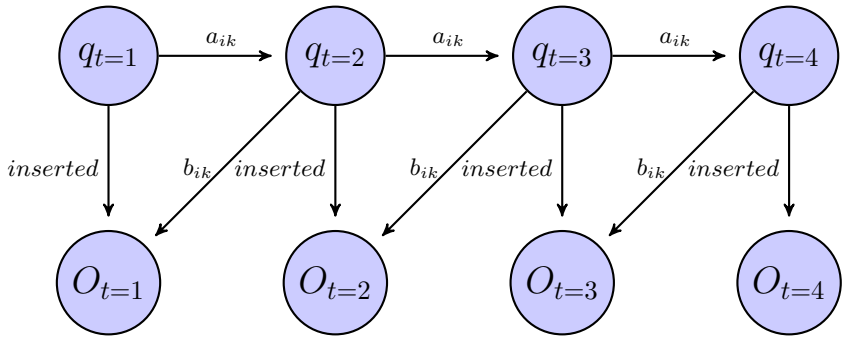

Figure 2: Illustation of a HMM.

q_t	Hidden state at time t which in this report is self-rated health.
O_t	Observable emissions coming from the hidden state at time t which in this report is the answers to the questions in table 1.
a_{ij}	Probability of transitioning from state $q_t = S_i$ to state $q_{t+1} = S_j$.
b_{ik}	Probability that $O_t = v_k$ is emitted from state $q_{t+1} = S_j$.

Table 3: Description of figure 2.

In figure 2 above the hidden states and their respective emission, including their respective time index t, are explained graphically. It illustrates how a hidden state emits something that happened earlier, the answers to the questions in the HealthWatch form. In this way the model can use the mathematics of an HMM and therefore also the theory of the HMM can be applied to the problem that initiated this report [7].

An illustration of how figure 2 looks in terms of the five questions in table 1 can be seen below.

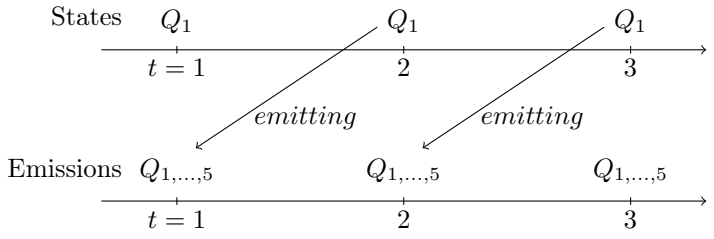

Figure 3: Illustation of an HMM with relation to the five questions.

In figure 3 one can see that Q_1, which corresponds to SRH, appears in both the upper and the lower timeline where Q_1 has the same value in the upper and lower timeline for equal t:s. This is important later in the report since a dependence arises between the state and a part of the emission, the Q_1 part of $Q_{1,\ldots5}$. SRH is modeled as the hidden state in the model described in further

detail below. The idea of the model is to use the five questions at time t, called $Q_{1,...,5}$ in figure 3, and predict what the person's SRH, called Q_1 in figure 3, will be at time $t+1$. In terms of HMM notation the emission O_t is usually emitted from state q_t. This is however not suitable in our case since our state is a future outcome. For this reason the state that emits O_t will be called q_{t+1} from here on.

Both the hidden states, the self-rated health, and the emissions from the hidden states, the answers to the questions in table 1, were known when the model was constructed. Since a hidden state is defined as self-rated health it can be found in the data. In order to get n consecutive measurements the model picks $n+1$ consecutive measurements of SRH, choosing $q_2, ..., q_{n+1}$ as the hidden states for a responder who has emissions at $t = 1, ...n$.

State	*emits*	Emission
q_2	$\rightarrow$	O_1
q_3	$\rightarrow$	O_2
$\vdots$		$\vdots$
q_{n+1}	$\rightarrow$	O_n

Table 4: The HMM nomenclature for this report.

To build the model, monthly averages of the different questions, answered on a scale from zero to one hundred, were used. When trying to predict a responder's average self-rated health during the following month this seems like the obvious choice.

The data had to be filtered before it could be used in MATLAB. There were three criteria that had to be met in order for MATLAB to accept the data and for the HMM to be valid. The criteria were:

1. Consecutive measurements for a responder.
2. Equal length of the responder's series of measurements.
3. Real values in each measurement.

Thus the data was filtered to include responders with a minimum consecutive series of measurements consisting of real values. With this filtering many responders were disregarded, but the problem of missing data was evaded.

There exist other ways of dealing with missing data. Randomly draw from the population with uniform distribution when a measurement is missing, interpolating between two data points, or simply adding the last known value as the unknown value, are commonly used techniques. Since the measurements were considered to be non-linear and very inconsistent none of the above techniques were satisfactory, and therefore not used in the model.

When the data had been filtered and categorized it was imported into MATLAB and inserted into the function hmmestimate where the transition- and

emission matrices A and B were calculated. Another matrix was also calculated with this function, it is referred to as the additional matrix.

3.2 Data

As mentioned earlier an HMM consists of hidden states and emissions emitted from those states. The HealthWatch form has eleven questions which can be answered in a range from zero to one hundred. The answers to the questions will be emissions from a hidden state, namely future self-rated health. The emission has to be one variable that is emitted from the hidden state. For instance it cannot be an array of answers to eleven questions that each influences the state in a different way.

To resolve this problem the answers to the different questions were put together to make one unique answer. For instance if a responder answers $\{X, Y, Z\}$ the unique answer would be $\{XYZ\}$. This will however lead to many different unique answers.

$$\text{number of unique answers} = n^q,$$

$$n = \text{number of answers},$$

$$q = \text{number of questions}.$$

The number of answers in our case is one hundred if it is assumed that only integers can be emitted, which is true in our case. The number of questions is eleven. This means the number of unique answers equals one hundred to the power of eleven; 10^{12}. There are techniques with continuous scale on the emissions from HMMs. To use those techniques a distribution of the emissions has to be chosen. This was not implemented.

The discretization was constructed as intervals of the answers and the number of questions was reduced to five. The five questions were chosen on the grounds that they are important from a health perspective. The number of intervals was limited to three.

$$answer \in \{[0, x], (x, y], (y, 100]\} \tag{15}$$

The intervals' cut-offs, x and y in equation (15), were varied to find a model with high accuracy. To clarify, if any interval had too few observations in the training set the accuracy would be poor for that interval. Two intervals were tested with poor accuracy as a result. A number of intervals greater than three would lead to too many unique answers. Too many unique answers is a problem when the data set is not able to represent all answers. If the data set can not represent all answers the model will have gaps in which future use of the model will not give a correct estimate. Therefore the model that is less complicated could be considered to give better predictions on the cost of less information about that prediction. In practice a better model which is less complicated will give a more certain prediction of a bigger interval. A more complicated model would in the worst case give a prediction with very low accuracy and precision on a tighter interval, which is undesirable.

3.2.1 The impracticality of many unique answers

It is quite easy to understand that a big number of different answers will be impractical since it is desirable to observe most of the emissions in the training set. If an emission is unobserved it will most likely be assigned probability zero of being emitted from a state. This could be inaccurate if the training set does not represent every possible responder.

For example:

Interval	0-30	31-70	71-100
Category	1	2	3

Emission 1: 11111. Translated from $\{10, 15, 11, 21, 30\}$.
Emission 2: 11112. Translated from $\{10, 15, 11, 21, 31\}$.

Emission 1 and 2 are here unique answers and can be seen as one number; 11111 (eleven thousand one hundred and eleven) and 11112 (eleven thousand one hundred and twelve) respectively. When looking at the raw data of the emissions that were translated into the unique answers (far right), one can see that emission 1 and 2 are almost exactly the same. But the model separates them because of the interval assignments. Furthermore, emission 1 was observed in the training set and thus has a probability greater than zero; say $p = 0.20$. Emission 2 was not observed in the training set and thus has probability zero. It is more likely that emission 2's real probability is $p = 0.20$ than $p = 0$ since emission 1 and emission 2 are almost the same. Hence it is desirable to choose the number of questions and intervals as small as possible to reduce the number of unique answers. But reducing the number of unique answers will also render a training set with less information compared to a training set with a greater number of unique answers.

3.2.2 Pseudo-emissions and pseudo-transitions

To deal with the problem of an incomplete training set MATLAB has an option when estimating the transition and emission matrices, namely pseudo-emissions and pseudo-transitions. The idea is to add some probability to every number in the transition and emission matrices so that no probability is set to zero.

The pseudo-emissions:

$$P(O = v_k | q = S_j) = \frac{n_{k,j} + \alpha f_k}{n_j + \alpha}, \tag{16}$$

where $n_{k,j}$ is the number of emitted $O = v_k$ from state $q = S_j$ and n_j is the total number of emissions from state $q = S_j$. The pseudo-counts αf_k are added to make sure that no probability is equal to zero. Some of the f_ks could be set

to zero if some events never occur in reality. α is a constant. The probability should be normalized so that

$$\sum P(O = v_k | q = S_j) = 1, \text{ for } \forall j.$$

When pseudo-emissions and pseudo-transitions are used the emission and transition matrices will not have any entries equal to zero, given that $f_k > 0$ for $\forall k$ and $\alpha > 0$. More about pseudo-emissions and pseudo-transitions can be found in [5].

To find all f_ks a plot of all emissions in the training set was made. This plot shows how likely the emissions are and possibly which emissions that are likely to occur but did not appear in the training set. What is likely to occur without having observed it will be a good guess.

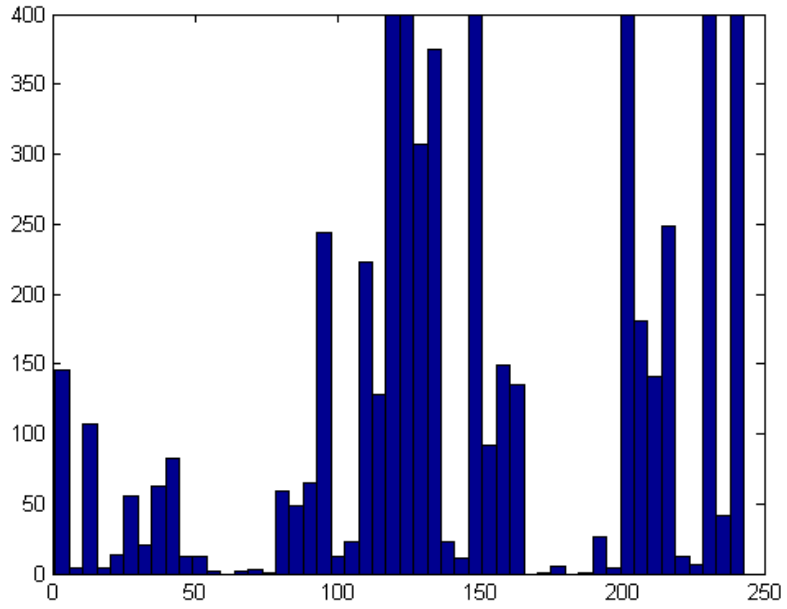

Figure 4: Observed emissions from the training set.

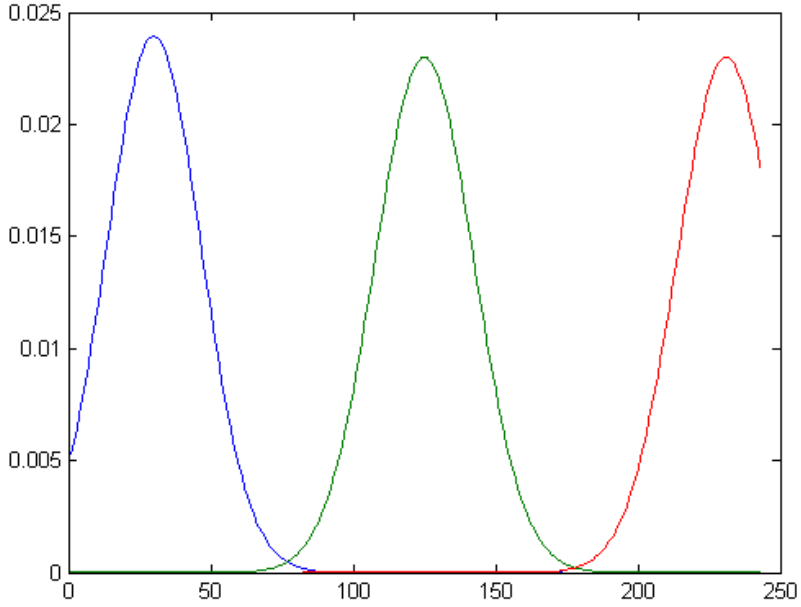

Figure 5: Created distribution of the pseudo-emissions.

In figure 5 the estimated distribution of the emissions can be seen. It was created ad-hoc by using three separate normal distributions with mean and variance chosen to fit figure 4. The idea was to adapt a continuous distribution to the observed emission's distribution. This distribution can now be used to set the f_ks. A one on the x-axis will read a f_1 on the y-axis etc. These f_ks will then be used in equation (16).

The pseudo-transitions:

The pseudo-transitions are calculated by the same concept as the pseudo-emissions.

$$P(q = S_i|q = S_j) = \frac{m_{i,j} + \alpha w_i}{m_j + \alpha}, \tag{17}$$

where $m_{i,j}$ is the number of transitions to state $q = S_i$ from state $q = S_j$ and m_j is the total number of transitions from state $q = S_j$. The probability should be normalized so that

$$\sum P(q = S_i|q = S_j) = 1, \text{ for } \forall j$$

The w_is for the pseudo-transitions were chosen uniformally, i.e. all w_is got the same value. The w_is were chosen in this way to make state one and two more likely, these states were considered to be underrepresented in the data.

In practice the f_ks and the w_is will be inserted into the built in MATLAB function hmmestimate used to estimate the transition- and emission matrices A and B. By inserting the f_ks and the w_is the MATLAB function is able to deal

with the cases of very few or zero observations and as a result the matrices A and B will no longer have any probabilities equal to zero.

3.2.3 Training set

The training set is the data used to create the model. When constructing a statistical model such as the algorithm in this report, it is important to have enough data. If the data set used to train the algorithm is too small the problem of interest might not be described by the model since it does not contain all information needed. By choosing different minimum length of the time series, there are ways to make the training set in different sizes. With the data provided it was possible to make a fairly large training set by reducing the amount of consecutive measurements. This means a training set with a greater number of responders but with fewer measurements per responder. A drawbacks rendering from this approach is seasonal changes, i.e., the model could be biased on if the time series is too short (less than one year). This problem was solved by seasonally adjusting the data described in further detail below. The other alternative was to have longer time series but fewer responders. If the time series is conditioned to contain at least twelve consecutive measurements there is no need for seasonally adjusting the data since all months are included.

3.2.4 Seasonal adjustment of data

The seasonal adjustment was made onto the raw data. First the monthly mean of all responders on all questions respectively was calculated.

$$\mu_{month,question} = \frac{1}{n}\sum_{i=1}^{n} D_{i,month,question},$$

where n=number of observations from one specific month and one specific question.

Next the yearly mean of the answer to the specific questions was calculated.

$$\mu_{question} = \frac{1}{N}\sum_{i=1}^{n}\sum_{month=1}^{12} D_{i,month,question},$$

where N=total number of observations from one specific question.

If there are not any seasonal trends the yearly mean of each question would be the same as the monthly mean of each question regardless of the month upon which the monthly mean was calculated. If this is not the case seasonal trends are present and can be adjusted by subtracting the difference between the monthly mean and the mean on every answer:

$$D^{adjusted}_{i,month,question} = D_{i,month,question} - (\mu_{month,question} - \mu_{question}).$$

More about seasonal adjustment on data can be found in [6].

3.3 Test Set

The test set is supposed to be independent of the model constructed. Thus the model's performance can be tested on the test set. It exist standards on choosing how big the test set should be in relation to the training set. It is however mostly dependent on how big the data set is to start with since a larger test set will render a smaller training set and vice versa. In this thesis a test set of about 150-300 responders was considered enough. The test set was chosen at random by drawing uniformly from the bigger, anonymized usable data set mentioned in section 3.

3.4 Prediction

When the model has been created, i.e. the transition and emission matrices have been estimated by the MATLAB function hmmestimate, it is possible to calculate the most likely hidden state. Remember that the hidden state is the average self-rated health the following month.

$$P(q_{t+1}|Data) = P(q_{t+1}|q_t, O_t) = \frac{P(q_{t+1}, q_t, O_t)}{P(q_t, O_t)} = \frac{P(O_t|q_{t+1})P(q_{t+1}|q_t)P(q_t)}{P(q_t, O_t)} \Rightarrow$$

$$P(q_{t+1}|Data) = \frac{P(O_t|q_{t+1})P(q_{t+1}|q_t)}{P(O_t|q_t)} \tag{18}$$

It was discovered that this approach resembles Partially Hidden Markov Models (PHMMs) [8]. It is however not essential what the model is called, but rather how it performs and that the model is right in a mathematical sense. Therefore the resemblance was not further investigated.

Here $q_{t+1} = S_j$ in equation (18) is the hidden state, i.e. the answer to the question "How do you feel right now?" at $t + 1$. $q_t = S_i$ is the hidden state at t, i.e. $q_t = S_i$ is the answer today on the question "How do you feel right now?". It is assumed that $q_t = S_i$, how a person is feeling today, will give some information on how they will feel in the future. Since this question is answered in the HealthWatch form $q_t = S_i$ is known today and is thus a part of $O_t = \{v_a v_b v_c v_d (q_t = S_i)\}$. Technically we say that $q_t = S_i$ is emitted from $q_{t+1} = S_j$ as an observation, it is one of the answers emitted from the hidden state. O_t is the emission at $t + 1$ and contains the answer to all the questions mentioned in the introduction, including SRH at that time which in fact is $q_t = S_i$. The conditional probabilities in equation (18) are all estimated by inserting data into the MATLAB function hmmestimaste. The matrices estimated are called, as mentioned before; the transition matrix and the emission matrix. There is one more matrix estimated by hmmestimate that we call the additional matrix. The additional matrix describes the dependence between the combined answers on all five questions at time t, including SRH at time t, and SRH at time t.

Conditional probability	Matrix
$P(O_t = v_k \mid q_{t+1} = S_i),\ \forall k, i$	Emission matrix
$P(q_{t+1} = S_i \mid q_t = S_j),\ \forall i, j$	Transition matrix
$P(O_t = v_k \mid q_t = S_j),\ \forall k, j$	Additional matrix

Table 5: Conditional probabilities and their corresponding matrix.

3.5 Construction of the model in practice

To calculate the transition matrix, the emission matrices and the performance of the model MATLAB was used. First the data was configured in Excel to fit the description of an HMM as described in section 3.2. After that the data was imported into MATLAB where the transition and emission matrices were created with the built in function hmmestimate. These are the matrices that consists of the probabilities $P(O_t|q_{t+1})$, $P(q_{t+1}|q_t)$ and $P(O_t|q_t)$. In equation (18) the multiplication in the numerator is equal to a vector with the possible predicted states equal to the index and the probabilities of their occurrences respectively as entries. In fact there are nine different possible combinations. Everyone of those combinations cannot be used since all but three combinations render unrealistic answers such as e.g. an emission from state one and a transition to state two. It is not possible to get an emission from a state that did not occur. In practice a matrix multiplication of $P(O_t|q_{t+1})$ and $P(q_{t+1}|q_t)$ was made. The diagonal elements were then used to form the vector. The maximum of this vector will give the most likely predicted state as the index of this probability. To find the probability, $P(q_{t+1}|Data)$, of this occurrence the before mentioned probability was divided by $P(O_t|q_t)$ where $O_t = v_k$ is known.

4 Results

4.1 Data filtering

The data allowed for twelve consecutive measurements. With fewer consecutive measurements the number of responders increase but the total number of measurements, consecutive measurements times number of responders, does not increase in such a way that it would be beneficial for the algorithm. More consecutive measurements leads to too few responders.

4.2 Performance

Testing the model resulted in an acceptable accuracy of over 83 %, see table 6. The precision was, as expected, high in state three and lower in states one and two, see table 7. The reason for this is most likely the shortage of data in the training set, especially in the case of state one. State three has many observations in the training set and thus high precision.

The model was tested on two different versions of the anonymized data. One version divided the data randomly into a training set consisting of 1080 responders and a test set consisting of 268 responders. The other version divided

the data randomly into a training set consisting of 1180 responders and a test set consisting of 168 responders. The accuracy and precision of the two versions were tested, the results are displayed below in table 6 and 7.

Training/Test set	Accuracy
1080 / 268	83,3 %
1180 / 168	84,5 %

Table 6: The model's accuracy for the different data sets.

Training/Test set	State:	1	2	3
1080 / 268	Precision:	25 %	61.4 %	90.8 %
1180 / 168	Precision:	60.0 %	44.8 %	94.0 %

Table 7: The model's precision for the different states.

Notable in table 7 is that there only were seven observations of state one in the top model and 3 observations of state one in the bottom model.

4.3 Emission intervals

As mentioned in section 3.2 the data had to be modified to fit the description of an HMM. It was also desirable to take several questions into account. This rendered a lot of unique answers that led to a simplification of the problem. The answers to the questions in the HealthWatch form were divided into categories, or intervals, to reduce the number of combinations, remember n^q, categories to the power of number of questions, in section 3.2. The intervals were chosen such that each interval contained a considerable amount of responder since it is desirable to observe every emission. The resulting intervals and their corresponding state are displayed below in table 8.

Category:	1	2	3
SRH:	0-39	40-79	80-100
Control:	0-39	40-79	80-100
Efficiency:	0-39	40-79	80-100
Workjoy:	0-39	40-79	80-100
Workload:	0-39	40-79	80-100

Table 8: The intervals that were chosen to reduce the number of possible emissions.

4.4 State intervals

SRH, the self-rated health, in the future is what this thesis is trying to predict. Future SRH also had to be divided into categories, intervals, in order to make the

problem solvable with the available data. The problem is the same as mentioned above; too many categories will not produce a usable model. The intervals for future SRH were chosen by HealthWatch and the author of this article together. HealthWatch chose the cut-offs such that the intervals would have some meaning in a psychological way while the author made sure that the intervals were not chosen in such a way that the model would be useless. That is, if any interval is chosen to be very small or placed badly, too few observation in that interval would be a fact and the model's prediction capabilities in that interval would be poor. In table 1 below the final intervals for future SRH can be found.

State:	1	2	3
SRH-prediction:	0-39	40-60	61-100

Table 9: The intervals that were chosen to reduce the number of possible states.

4.5 State interval performance

To see how the model performed in the different intervals for future SRH the correctly and incorrectly predicted values were counted. If the model performs bad in a specific interval it will be observable in these numbers. In the table below a correctly predicted state means that the model predicted state S_i and the true hidden state actually was S_i, while the incorrectly predicted states means that the model predicted S_i but the state actually was something else. To get these numbers the correct hidden states has to be available, which they are in our case since all data is available.

3 1 0 1 13 8 1 15 126
2 2 0 5 54 16 1 32 157

Training/Test set	State:	1	2	3
1080 / 268	Correctly predicted:	2	54	157
	Incorrectly predicted:	6	34	16
1180 / 168	Correctly predicted:	3	13	126
	Incorrectly predicted:	2	16	8

Table 10: The model performance on the different states.

These are the data on which both accuracy and precision are based on.

In table 12 below the incorrectly predicted states and their true future states are listed. When dealing with people's health it is in general more serious to do a prognosis that states that a person is well when he or she is not than to state that a person is sick while he or she is not. Therefore the most interesting numbers in table 12 are the ones with a one, or possible a two, as the incorrectly predicted state. The most serious fault would then be to predict state three while the person in reality got to state one the following month. In table 11 the real future state's predictions are stated in percentage. The numbers mean that a real future values of SRH is predicted as state x, y percent of the time.

1080/268	True state:	1	2	3
	1	25.0 %	2.3 %	0 %
Predicted state	2	62.5 %	61.4 %	9.3 %
	3	12.5 %	36.4 %	90.8 %
1180/168		1	2	3
	1	60.0 %	3.5 %	0 %
Predicted state	2	20.0 %	44.8 %	6.0 %
	3	20.0 %	51.7 %	94.0 %

Table 11: Real future states with the percentage distribution on the states that were predicted.

In table 11 above it is important to remember that the lower part consisting of 1180 learning responders and 168 testing responders has fewer test responders than the upper part; 268 responders. With smaller data sets the numbers are less thrust worthy.

It is clear in the table above that not one single person that was in the highest interval of SRH the following month was predicted to be in the lowest interval of SRH. This is however, as stated earlier, not as important as the other way around, namely to predict a high interval of SRH when the responder's real future SRH is in the lowest interval.

The column to the far left shows in what interval a person with real future SRH of one was predicted to have. The upper constellation shows a high percentage at state two and moderately low at state three. The lower constellation has even higher percentage at state three, there are few responders based on these particular figures though.

The middle column shows that a person with real SRH of two has a fairly high percentage at three in both constellations, which is bad, and both have very low at state one.

The column to the right shows that the model performs very good in state three predictions. This is expected since most responders used to train the model have had SRH corresponding to state three. Simply because most people feel well.

Training/Test set	Predicted state	True state	Predicted state	True state
1080 / 268	2	3	3	2
	3	2	3	2
	3	2	3	2
	2	1	2	3
	3	2	3	2
	3	2	2	3
	3	2	3	2
	3	2	3	2
	2	3	2	3
	2	3	3	2
	2	3	2	3
	3	2	2	3
	3	2	3	2
	2	3	3	1
	3	2	2	3
	3	2	3	2
	2	3	3	2
	2	1	3	2
	3	2	3	2
	3	2	2	3
	3	2	2	1
	3	2	1	2
	3	2	2	3
	2	1	3	2
	1	2	2	3
	2	3	3	2
	3	2	3	2
	3	2	2	1
1180 / 168	2	3	3	2
	3	2	3	2
	3	2	3	2
	2	3	2	3
	3	2	3	2
	1	2	2	1
	2	3	3	2
	3	2	2	3
	2	3	3	2
	3	1	3	2
	3	2	2	3
	2	3	3	2
	3	2	3	2

Table 12: The incorrectly predicted states and their true future states.

5 Conclusions

It is possible to predict SRH with an algorithm based on the mathematics of an HMM. The model can however not tell much about a persons future SRH, but it can predict an indication of what the responders SRH will be. The intervals for the predicted SRH are fairly wide and therefore the model points in the direction of which the responder is going in terms of SRH rather then tell that the responder will have a certain SRH one month from now. Due to the accuracy of the model a prediction that shows a low interval of SRH the following month should be taken seriously by the user.

There is a problem with the model's poor precision and accuracy in state one and two. These are the intervals that are most interesting. If a responder is in one of these lower intervals there is a reason to be worried about their health. Therefore the model's poor accuracy and precision are inconvenient.

Further studies should be done to improve the model's performance in interval one and two. To sort the data to contain a higher share of people with lower SRH could be one way of doing this. Then the model would be trained to predict low SRH rather than SRH in general. These, low SRH, subjects could even have the same health patterns. If they have, the model could possibly detect if someone are falling into the same pattern as these low SRH responders that the model is based on.

6 References

[1] D. Hasson and K. Villaume:
An Automated and Systematic Web-Based Intervention for Stress Management and Organizational Health Promotion, 2013.
[2] J. Enger and J. Grandell:
Markovprocesser och köteori, KTH, 2006.
[3] L. R. Rabiner:
A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings of the IEEE, 77, pp. 257–286, 1989.
[4] L. E. Baum:
A statistical estimation procedure for probabilistic functions of Markov processes,
http://en.wikipedia.org/wiki/File:HiddenMarkovModel.svg, 1996.
[5] T. Koski:
Hidden Markov Models, Kluwer 2001.
[6] http://www.abs.gov.au/:
Time Series Analysis: Seasonal Adjustment Methods, 2008.
[7] Hassan, Md Rafiul and Nath, Baikunth:
Stock market forecasting using hidden Markov model: a new approach,
http://modular.math.washington.edu/home/wstein/www/home/simuw/simuw08/refs/hmm/hassan–nath–2005–stock_market_forecasting_using_hidden_markov_model_a_new_approach.pdf,
5:th International Conference on Intelligent Systems Design and Applications, 2005. ISDA'05. Proceedings. 192–196, IEEE, 2005.
[8] Forchhammer, Søren and Rissanen, Jorma:
Coding with Partially Hidden Markov Models,
Data Compression Conference, 1995. DCC '95. Proceedings. 92–101, IEEE, 1995.

TRITA-MAT-E 2014:17
ISRN-KTH/MAT/E—14/17-SE

www.kth.se