Preregistration: Estimating the Parameter Recovery of the Hidden Markov Model in gazeHMM

Malte Lüken, Simon Kucharsky, & Ingmar Visser

University of Amsterdam, Department of Psychology

19.05.2020

In this document, we are preregistering a simulation study to estimate the parameter recovery of a hidden Markov model (HMM), which is part of our recently developed algorithm gazeHMM. The purpose of this algorithm is to classify eye movements into distinct events. These events include fixations, saccades, and optionally postsaccadic oscillations (PSOs) and smooth pursuits. The full algorithm will be available on GitHub (www.github.com/maltelueken/gazeHMM) upon publication.

In general, gazeHMM consists of three steps: First, the raw gaze data is preprocessed. Missing samples are identified, the data are filtered, and for each sample, the velocity, acceleration, and difference in angle to the next sample is calculated. Second, an HMM estimates the parameters of response, transition, and initial state models for the specified eye movement events. Moreover, the HMM computes the posterior probability of each sample for belonging to each event. The samples are labeled as the event with the highest posterior probability. Third, a postprocessing routine relabels samples that are violating theoretical and practical constraints. This simulation study will only consider the parameter recovery of the HMM but not the performance of the preprocessing and postprocessing routines in the algorithm.

Model Description

The HMM estimates non-observable states that have generated the data (Visser & Speekenbrink, 2019). In the context of eye-tracking, each state corresponds to an eye movement event. In gazeHMM, the model can have between two and four states. State one always corresponds to fixations, state two always to saccades, state three to PSOs, and state four to smooth pursuits. The HMM is multivariate but conditionally independent on the states. Thus, every state has three independent response models describing velocity, acceleration, and angle. The velocity and acceleration response models for all four states use two-parameter gamma distributions (shape and scale parametrization). The angle response model for the fixation state uses a uniform distribution, while the other three states use a two-parameter von-Mises distribution (mean and concentration parametrization) to describe angle. Both the transition model and the initial state model use a multinomial distribution (with the logit and the identity link function, respectively). Note that no covariates are included in the model, only intercepts for the parameters are estimated.

The HMM is implemented in R (version: 3.6.3; R Core Team, 2020) using the package depmixS4 (Visser & Speekenbrink, 2010). An expectation-maximization algorithm (maximum iterations: 5000, relative tolerance: 1e-08; Dempster et al., 1977; McLachlan & Krishnan, 1997) estimates the parameters given the hidden states, which are estimated through the Viterbi algorithm (Forney Jr, 1973; Viterbi, 1967). The parameters of each response model (except for the uniform distribution) are estimated through maximum likelihood with a spectral projected gradient method (Birgin et al., 2000) and Barzilai-Borwein step lengths (Barzilai & Borwein, 1988) as implemented in the package BB (Varadhan & Gilbert, 2009).

Parameter Recovery

To estimate parameter recovery of the HMM, it will repeatedly generate data with a set of parameters (true parameter values). The same model will then be applied to estimate the parameters from the generated data (estimated parameter values). We will compare the true with the estimated parameter values to assess whether a parameter was recovered by the model. Additionally, we will compare the true states of the HMM with the estimated states to judge how accurately the model recovers the states that generated the data.

Starting values

The HMM will always start with a uniform distribution to estimate the initial state and state transition probabilities. To generate random starting values for the estimation of shape, scale, and concentration parameters, we will use gamma distributions with a shape parameter of $\alpha_{start} = 3$ and $\beta_{start} = \psi_{true}/2$ with ψ_{true} being the true value of the parameter to be estimated. This setup ensures that the starting values are positive, their distributions are moderately skewed, and the modes of their distributions equal the true parameter values. Mean parameters of the von-Mises distribution will always start at their true values.

Design

Parameter Variation

The simulation study will be divided into four parts. In the first part, we will vary the parameters of the HMM. For models with $k \in \{2, 3, 4\}$ states, $q \in \{10, 15, 20\}$ parameters will be varied respectively. For each parameter, the HMM will generate 100 data sets with $N = 2500$ samples and the parameter varied in a specified interval in equidistant steps. This will result in $100 \times (10 + 15 + 20) = 4500$ recoveries. Only one parameter will be varied at once, the other parameters will be set to their default values. All parameters of the HMM will be estimated freely (i.e., there are no fixed parameters in the model). We will not manipulate the initial state probabilities because these are usually irrelevant in the context of eye movement classification. For the transition probabilities, we will only simultaneously vary the probabilities for staying in the same state (diagonals of the tranistion matrix) to reduce the complexity of the simulation. The left over probability mass will be split evenly between the probabilites for switching to a different state (per row of the transition matrix). Moreover, we will not modify the mean parameters of the von-Mises distributions: As location parameters, they do not alter the shape of the distribution and they are necessary features for the HMM to distinguish between different states.

We define approximate ranges for each response variable and will choose true parameter intervals and default values so that they will produce samples that roughly correspond to these ranges. Table 1 shows the assumed ranges for each event and Table 2 shows the intervals and default values for each parameter in the simulation. Parameters will be scaled down by factor 10 (compared to the reported ranges) to improve fitting of the gamma distributions. We will set the intervals for shape parameters of the gamma distributions for all events to [1,5] to examine how skewness influences the recovery (shape values above five approach a symmetric distribution). The scale parameters will be set so that the respective distribution will approximately match the assumed ranges. Since the concentration parameters of the von-Mises distribution are the inverse of standard deviations, they will be varied on the inverse scale.

Sample Size and Noise Variation

In the second part, we will vary the sample size of the generated data and the amount of noise added to it. The model parameters will be set to their default values. For models with $k \in \{2, 3, 4\}$ states and sample sizes of $N \in \{500, 2500, 10000\}$, we will generate 100 data sets ($100 \times 3 \times 3 = 900$ recoveries). These samples sizes roughly correspond to small, medium, and large eye-tracking data sets for a single participant and trial. To simulate noise, we will replace velocity and acceleration values y with draws from a gamma distribution with

$\alpha_{noise} = 3$ and $\beta_{noise} = (y/2)\tau_{noise}$ with $\tau_{noise} \in [1,5]$ varying between data sets. This procedure ensures that velocity and acceleration values remain positive and are drawn from moderately skewed distributions with modes equal to the original values. To angle, we will add white noise from a von-Mises distribution with $\mu_{noise} = 0$ and $\kappa_{noise} \in 1/[0.1, 10]$ varying between data sets. τ_{noise} and κ_{noise} will vary simultaneously in equidistant steps in their intervals.

Variation of Starting Values

In the third part, we will increase the variation in the starting values used for parameter estimation. The model parameters will be set to their default values. For the shape, scale, and concentration parameters, we will simultaneously increase the scale parameters of the starting value gamma distributions: For $k \in \{2, 3, 4\}$ states and $\beta_{start} = (\psi_{true}/2)\tau_{start}$ with $\tau_{start} \in \{1, 2, 3\}$, 100 data sets with $N = 2500$ samples will be generated each ($100 \times 3 \times 3 = 900$ recoveries).

Missing data

In the last part, we will set intervals of the generated data to be missing. The model parameters will be set to their default values. For $k \in \{2, 3, 4\}$ states and $m \in \{1, 3, 5\}$ intervals, 100 data sets with $N = 2500$ samples will be generated ($100 \times 3 \times 3 = 900$ recoveries). The length of the missing data interval $l \in [1, 200]$ samples will vary in equidistant steps between the data sets.

Recovery Analysis

For each parameter, we will calculate the root median square proportion deviation (RMdSPD; analogous to root median square percentage errors, see Hyndman & Koehler, 2006) between the true and estimated parameter values:

$$RMdSPD = \sqrt{Med((\frac{\psi_{true} - \psi_{est}}{\psi_{true}})^2)}.$$

We will treat $RMdSPD < 0.1$ as good, $0.1 \leq RMdSPD < 0.5$ as moderate, and $RMdSPD \geq 0.5$ as bad recovery of a parameter. By taking the median, we will reduce the influence of potential outliers in the estimation and using proportions will enable us to compare RMdSPD values across parameters and data sets.

Additionally, we will apply a bivariate linear regression with the estimated parameter values as the dependent and the true parameter values as the independent variable to each parameter that has been varied on an interval in Part 1. Regression slopes closer to one will indicate that the model better captures parameter change. Regression intercepts different from zero will reflect a bias in parameter estimation.

To assess state recovery, we will compute Cohen's kappa (for all events taken together, not for each event separately) as a measure of agreement between true and estimated states for each generated data set. Higher kappa values will be interpreted as better model accuracy. We will adopt the ranges proposed by Landis & Koch (1977) to interpret kappa values.

Models that can not be fitted will be excluded from the recovery.

Table 1

Approximate ranges of response variables used to generate parameter values

Event	Resp. variable	Range
Fixation	Velocity	0-50
Fixation	Acceleration	0-50
Fixation	Angle	uniform
Saccade	Velocity	50-1000
Saccade	Acceleration	50-500
Saccade	Angle	~0
PSO	Velocity	20-100
PSO	Acceleration	10-90
PSO	Angle	~π
Smooth pursuit	Velocity	20-100
Smooth pursuit	Acceleration	0-30
Smooth pursuit	Angle	~0

Note. Units are °/s (velocity), °/s^2 (acceleration), and radians (angle). ~ indicates that the distribution has a peak at this value. Velocity ranges are based on event velocities reported in Larsson et al. (2013). Since we would also like to test the algorithm on extreme data distributions, we extended the ranges beyond those found in typical eye movement data.

Table 2

HMM parameter values for generating data

State	Parameter	Interval	Default	Description
1-4	ρ_i^*	-	$1/k$	Initial state probability for starting in state i
1-4	$a_{i=j}$	[.01,.99]	0.9	Transition probability for staying in the previous state i
1-4	$a_{i\neq j}$	$(1 - a_{i=j})/(k-1)$	$0.1/(k-1)$	Transition probability for switching to from state i to a different state j
1	α_{vel}	[1,5]	3	Shape parameter of the velocity gamma distribution
1	β_{vel}	[0.1,0.6]	0.35	Scale parameter of the velocity gamma distribution
1	α_{acc}	[1,5]	3	Shape parameter of the acceleration gamma distribution
1	β_{acc}	[0.05,0.25]	0.15	Scale parameter of the acceleration gamma distribution
1	min^*	-	0	Minimum of the uniform distribution

State	Parameter	Interval	Default	Description
1	max^*	-	2π	Maximum of the uniform distribution
2	α_{vel}	[1,5]	3	
2	β_{vel}	[5,15]	10	
2	α_{acc}	[1,5]	3	
2	β_{acc}	[1,5]	3	
2	μ^*	-	0	Mean parameter of the von-Mises distribution
2	κ	$1/[0.1, 10]$	1	Concentration parameter of the von Mises distribution
3	α_{vel}	[1,5]	3	
3	β_{vel}	[0.5,1.5]	1	
3	α_{acc}	[1,5]	3	
3	β_{acc}	[1,5]	3	
3	μ^*	-	π	
3	κ	$1/[0.1, 10]$	1	
4	α_{vel}	[1,5]	3	
4	β_{vel}	[0.5,1.5]	1	
4	α_{acc}	[1,5]	3	
4	β_{acc}	[0.05,0.25]	0.15	
4	μ^*	-	0	
4	κ	$1/[0.1, 10]$	1	

Note. Parameters marked with * will not be varied but always set to their default values. k is the number of states in the model.

References

Barzilai, J., & Borwein, J. M. (1988). Two-point step size gradient methods. *IMA Journal of Numerical Analysis*, *8*, 141–148. https://doi.org/10.1093/imanum/8.1.141

Birgin, E. G., Martinez, J. M., & Raydan, M. (2000). Nonmonotone spectral projected gradient methods on convex sets. *SIAM Journal of Optimization*, *10*, 1196–1211. https://doi.org/10.1137/S1052623497330963

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, *19*(1), 1–38. https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

Forney Jr, G. D. (1973). The viterbi algorithm. *Proceedings of the IEEE*, *61*(3), 268–278. https://doi.org/10.1109/PROC.1973.9030

Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, *22*(4), 679–688. https://doi.org/10.1016/j.ijforecast.2006.03.001

Landis, R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, *33*(1), 159–174.

Larsson, L., Nystrom, M., & Stridh, M. (2013). Detection of saccades and postsaccadic oscillations in the presence of smooth pursuit. *IEEE Transactions on Biomedical Engineering*, *60*(9), 2484–2493. https://doi.org/10.1109/TBME.2013.2258918

McLachlan, G. J., & Krishnan, T. (1997). *The EM algorithm and extensions*. Wiley.

R Core Team. (2020). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.r-project.org/

Varadhan, R., & Gilbert, P. (2009). BB: An R package for solving a large system of nonlinear equations and for optimizing a high-dimensional nonlinear objective function. *Journal of Statistical Software*, *32*(4), 1–26. http://www.jstatsoft.org/

Visser, I., & Speekenbrink, M. (2010). depmixS4: An R package for hidden Markov models. *Journal of Statistical Software*, *36*(7), 1–21. https://doi.org/10.18637/jss.v036.i07

Visser, I., & Speekenbrink, M. (2019). *Mixture and hidden Markov models with R including latent class models*. Springer.

Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, *13*(2), 260–269. https://doi.org/10.1109/TIT.1967.1054010