Manual EST toolbox

Simon Geirnaert, Tom Francart, Alexander Bertrand

June 6, 2019

Contents

The EST toolbox provides an implementation of the expected switching time (EST) metric to evaluate the performance of neural decoders for auditory attention detection in the context of neuro-steered hearing prostheses as published in [1, 3]. The EST metric is an interpretable, single-number metric that combines both accuracy and decision time. It allows easy comparison between neural decoders based on a relevant criterion, independent of the evaluated window lengths.

1 Important information

The toolbox can be found online at Github. By downloading and installing this software and the associated files on your computing system, you agree to use the software under the terms and conditions as specified in the License agreement (which can be found in the directory or online). By distributing the EST code freely online, we hope that it can be used by other researchers as a new standard performance metric in the auditory attention detection community, possibly with extensions to other brain-computer interface applications. If you wish to use this toolbox, please cite [1, 3, 2] in any related or resulting publications.
Note that this code (as it is a first version) is still prone to errors. If you find any bugs, please report them to simon.geirnaert@esat.kuleuven.be. Furthermore, we welcome any suggestion or improvement for the toolbox.

2 Getting started

2.1 Installation

Download the EST toolbox, unzip it to any directory and add the directory to the MATLAB path. This can be done by browsing to the chosen directory and running

```
addpath(pwd); % Add the current directory to the MATLAB path
savepath;      % Save the path
```

2.2 Contents.m

To show a complete listing of all the functions in the EST toolbox, once it is installed to the directory `est-toolbox`, run:

```
doc est-toolbox
```

The command displays the file `Contents.m`, which shows grouped functions together with a concise description:

```
% EST TOOLBOX
% Version 1.0, 06-03-2019
%
% MAIN FUNCTION
%   computeEST            - Compute the expected switching time based
%                            on (tau,p)-points.
%
% OPTIMIZATION MARKOV CHAIN MODEL
%   optimizeMarkovChain  - Compute the optimal AAD Markov chain.
%
% BASIC FEATURES MARKOV CHAIN MODEL
%   ssDistribution       - Compute the steady-state distribution of
%                           the AAD Markov chain.
%   stateToAmpl          - Convert a state index to a relative
%                           amplification level.
%   lbCfdInt             - Compute the lower bound of the
%                           P0-confidence interval of the AAD Markov
%                           chain.
%   evState              - Compute the expected value of the AAD
%                           Markov chain.
%
% FUNDAMENTAL METRICS
%   meanHittingTime      - Compute the mean hitting time from state i
%                           to k.
%   transitTime          - Compute the expected transit time to state
%                           k.
%
% CONSTRUCTION PERFORMANCE CURVE
%   interpolatePerfCurve - Interpolate the performance curve through
%                           evaluated performance points.
```

2.3 Quick start guide

2.3.1 Computation of the EST performance metric

The EST is computed in four steps:

1. Construction of the $p(\tau)$-performance curve by interpolating through the evaluated (on real EEG and audio data) (τ_i, p_i)-points (decision time,accuracy).
2. Optimization of the Markov chain in the number of states N for each sampled τ on the $p(\tau)$-performance curve.
3. Computation of the transit time $T(p(\tau), \tau, \hat{N}_\tau)$ per sampled τ and corresponding optimal number of states $\hat{N}_\tau$.
4. The EST is equal to the minimal transit time over all evaluated transit times: $\text{EST} = \min_\tau T(p(\tau), \tau, \hat{N}_\tau)$.

These steps are implemented in the *main*-function `computeEST.m`. Given the evaluated (τ_i, p_i)-performance points `(tau,p)`, the EST can be computed with:

```
est = computeEST(tau,p);
```

The default hyperparameter values are $P_0 = 0.8$ (confidence level), $c = 0.65$ (lower bound confidence interval), $N_{\text{min}} = 5$ (minimal number of states) and $K = 1000$ (number of samples evaluated on the performance curve). These hyperparameters can be adapted via extra arguments in the EST-function.

2.3.2 Designing an optimal Markov chain model for a neuro-steered hearing prosthesis

In Section II.D of [3], a methodology is proposed to design an optimal Markov chain model for an adaptive gain control system in a neuro-steered hearing prosthesis. For a fixed accuracy p and hyperparameters P_0, c and N_{min}, the optimal number of states can be found with:

```
Nopt = optimizeMarkovChain(p,Nmin,P0,c);
```

The optimal model for a certain neural decoder (represented by evaluated (τ_i, p_i)-points) can be identified by extra outputs of the EST-function:

```
[est,Nopt,tauOpt,pOpt] = computeEST(tau,p,'Nmin',Nmin,'P0',P0,'c',c);
```

3 Function documentation

The MATLAB files in the EST toolbox contain a documentation consisting of:

- a short description
- a more extensive description with in- and output arguments
- a detailed description of the input arguments

- contact information.

In addition, a short demo can be found in `estDemo.m` to show an example of how to compute the EST metric with the toolbox.
[more to be completed]

References

[1] Simon Geirnaert, Tom Francart, and Alexander Bertrand. A New Metric to Evaluate Auditory Attention Detection Performance Based on a Markov Chain. *Internal Report*, March 2019.

[2] Simon Geirnaert, Tom Francart, and Alexander Bertrand. EST toolbox, March 2019. Available online.

[3] Simon Geirnaert, Tom Francart, and Alexander Bertrand. Expected Switching Time: a Interpretable Performance Metric to Evaluate Neural Decoders for Auditory Attention Detection. *Internal Report*, March 2019.