Replication of the Ψ Method: A Bayesian adaptive estimation of psychometric slope and threshold

Hao Zhu (hao.zhu@nyu.edu)
New York University Shanghai, 1555 Century Avenue
Shanghai, China

Abstract

I replicate the Ψ Method, which is a Bayesian adaptive method for estimating both threshold and slope parameters of the psychometric function. The Ψ method updates posterior probabilities in the two-dimensional parameter space of threshold and slope, and makes predictions based on the expected mean threshold and slope values. Each trial, it sets the stimulus intensity that minimizes the expected entropy, which maximizes the expected information gained. This replication was implemented in python and evaluated in computer simulations using the two-alternative forced-choice (2AFC) paradigm. The simulation result suggests a fair replication.

Keywords: Psychometric function; Two-alternative forced-choice; Bayesian adaptive method; Python

Introduction

The Ψ method (Kontsevich & Tyler, 1999) is a sophisticated method that efficiently estimates both the threshold α and slope β parameter of a psychometric function. It's arguably the most efficient of the adaptive methods which target both an estimate of the location and the slope parameter of the psychometric function (Prins et al., 2016). Moreover, It's widely used in psychophysical researches [(Leek, 2001), (Macmillan & Creelman, 2004), (Breakwell, Hammond, Fife-Schaw, & Smith, 2006)]. Therefore, I believe the Ψ method is a good project for replication purposes.

The Ψ method updates a posterior distribution, after each trial responses, across all possible values of both threshold and slope parameters. Then it selects the stimulus intensity for the next trial that will minimize the expected entropy in that posterior distribution. The entropy here is defined as it is in the context of the information theory (Shannon, 1948), which is a measure of uncertainty associated with the values of the threshold and slope parameters. By minimizing the entropy in a posterior distribution, it maximizes the information gained after each trial (Sims & Pelli, 1987), and thus increases the precision of the parameter estimates.

In the following sections, I described the mathematics behind the Ψ method in detail from definition to specific steps of trials. In addition, I showed my simulation result from implementation, which suggested a fair replication.

Method

The logic of Ψ method is designed as follows.

Definition

First, it defines a specific psychometric function (Gumbel in this case) $\Psi_{\alpha,\beta}(x)$ as shown in Equation 1, where the guess rate γ and the lapse rate λ are fixed. It also defines a sample space of the threshold α and slope β, while assuming the actual psychometric function falls within this parameter space.

$$\Psi_{\alpha,\beta}(x) = \gamma + (1-\gamma-\lambda)(1-e^{-10^{\beta(x-\alpha)}}) \tag{1}$$

Second, it defines a prior probability $p_0(\alpha,\beta)$ for the psychometric function, as shown in Equation 2.

$$p_0(\alpha,\beta) = N(\alpha;\alpha_{\text{prior}},\sigma^2_{\text{prior}}) \cdot N(\beta;\beta_{\text{prior}},\sigma^2_{\text{prior}}) \tag{2}$$

Last, it defines the likelihood function as $p(r|\alpha,\beta,x)$, which means the probability of getting responses r given stimulus x with certain α and β. The responses r is binary at either $r = 1$ *success* or $r = 0$ *failure*. The specific likelihood functions for each combination of the parameters r, x, α, β are defined as in Equation 3:

$$\begin{aligned} p(r=1|\alpha,\beta,x) &= \Psi_{\alpha,\beta}(x) \\ p(r=0|\alpha,\beta,x) &= 1-\Psi_{\alpha,\beta}(x) \end{aligned} \tag{3}$$

Trial-trial steps

For i-th trial, it works in the following steps:

1. Calculate the probability of stimulus x produce response r, as in Equation 4.

$$p_i(r|x) = \sum_\alpha \sum_\beta p(r|\alpha,\beta,x) p_i(\alpha,\beta) \tag{4}$$

2. Estimate the posterior probability of psychometric function with parameters α and β given that stimulus x will produce response r, as in Equation 5.

$$p_i(\alpha,\beta|x,r) = \frac{p_i(\alpha,\beta)p(r|\alpha,\beta,x)}{\sum_\alpha \sum_\beta p_i(\alpha,\beta)p(r|\alpha,\beta,x)} \tag{5}$$

3. Estimate the entropy of the posterior probability over the sample space of α and β, given that stimulus x will produce the response r, as in Equation 6.

$$H_i(x,r) = -\sum_\alpha \sum_\beta p_i(\alpha,\beta|x,r) \log p_i(\alpha,\beta|x,r) \tag{6}$$

4. Estimate the expected entropy for each stimulus x, as in Equation 7.

$$\begin{aligned} &E[H_i(x)] = \\ &\quad H_i(x,r=1)p_i(r=1|x) + H_i(x,r=0)p_i(r=0|x) \end{aligned} \tag{7}$$

5. Localize the stimulus x with the minimum expected entropy, as in Equation 8.

$$x_{i+1} = \underset{x}{\operatorname{argmin}} E[H_i(x)] \tag{8}$$

6. Run the trial with the stimulus x_{i+1} to get actual response r_{i+1}.

7. Update the prior probability according to the posterior probability and response r_{i+1}, as in Equation 9.

$$p_{i+1}(\alpha,\beta) = p_i(\alpha,\beta|x_{i+1},r_{i+1}) \tag{9}$$

8. Estimate the parameters α and β based on the updated prior probability distribution, which is the expected value of each parameter, as in Equation 10.

$$\begin{aligned}\alpha_{i+1} &= \sum_{\alpha} \alpha p_{i+1}(\alpha,\beta)\\ \beta_{i+1} &= \sum_{\beta} \beta p_{i+1}(\alpha,\beta)\end{aligned} \tag{10}$$

9. Go back to step 1 and keep looping until i reaches a preset number.

Implementation & Simulation

I implemented and simulated the Ψ method based on the Palamedes toolbox (Prins et al., 2018) in Python at version 3.6.10 (Van Rossum & Drake, 2009) with numpy module at version 1.18.5 (Harris et al., 2020) under Linux system at Ubuntu 16.04.6. The source code and demo are available at my GitHub page: `https://github.com/HaoZ94/PSI-py`

Definition

The Ψ method is simulated for an arbitrary 2AFC task with trial number set to be 50. The parameters for simulation are defined as follows.

First, the sample space for stimulus, threshold and slope are $x \subseteq [-1,1]$ with step of 0.02 (vector with shape 51×1), $\alpha \subseteq [-1,1]$ with step of 0.01 (vector with shape 101×1) and $\beta \subseteq [0.1,5]$ with a step of 0.05 (vector with shape 101×1), respectively. The guess rate γ and lapse rate λ are set to be 0.5 and 0 for the psychometric function $\Psi_{\alpha,\beta}(x)$, respectively. The observer's true psychometric function is assumed to be $\Psi_{\text{observer}}(x)$ (Equation 1) with threshold α_{observer} and slope β_{observer} are set to be 0 and 2, as in Figure 1.

Second, the priors threshold α_{prior} and slope β_{prior} of the prior probability density function $p_0(\alpha,\beta)$ (Equation 2) are set to be 0 and 2, respectively. And the prior standard deviation σ_{prior} is set to be 3. The prior probability density function is a matrix with shape 101×101, as in Figure 2.

Last, the likelihood function $p(r|\alpha,\beta,x)$ (Equation 3) is calculated as look-up table for each combination of r,α,β,x before running any trial to facilitate the computation process. The look-up table for both $r=1$ and $r=0$ is a matrix with shape $51 \times 101 \times 101$.

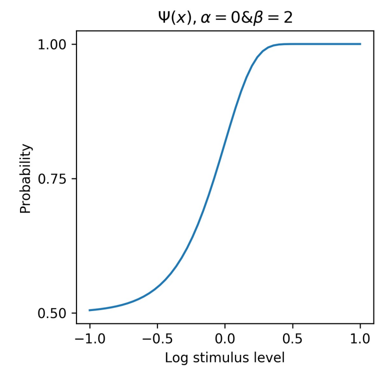

Figure 1: The observer's true psychometric function $\Psi(x)$. The threshold α_{observer} and slope β_{observer} are set to be 0 and 2, respectively.

Trial-trial steps

For the 1$^{\text{st}}$ trial,

1. The probability of stimulus x produce response r, $p_0(r|x)$ (Equation 4) is computed as the sum at the 2$^{\text{nd}}$ and 3$^{\text{rd}}$ dimension of product of prior probability density function $p_0(\alpha,\beta)$ (repeated to shape $51 \times 101 \times 101$ for multiplication) and the likelihood function $p(r|\alpha,\beta,x)$ (look-up table with shape $51 \times 101 \times 101$). For either $r=1$ or $r=0$, $p_0(r|x)$ is a matrix with shape 51×1.

2. The posterior probability $p_0(\alpha,\beta|x,r)$ (Equation 5) is computed as product of $p_0(\alpha,\beta)$ and $p(r|\alpha,\beta,x)$, divided by $p_0(r|x)$ (repeated to shape $51 \times 101 \times 101$ for division). For either $r=1$ or $r=0$, $p_0(\alpha,\beta|x,r)$ is a matrix with shape $51 \times 101 \times 101$.

3. The entropy of the posterior probability $H_0(x,r)$ (Equation 6) is computed as the sum at the 2$^{\text{nd}}$ and 3$^{\text{rd}}$ dimension of the product of $p_0(\alpha,\beta|x,r)$ and its own log ($0\log(0)$ is defined to equal 0, as $\lim_{p\to 0} p\log(p)=0$). For either $r=1$ or $r=0$, $H_0(x,r)$ is a matrix with shape 51×1.

4. The expected entropy $E[H_0(x)]$ for each stimulus x is computed as described in Equation 7, resulted in shape 51×1.

5. Find the stimulus with the minimum expected entropy, which is the initial stimulus for the first trial x_1.

6. A random number n is drawn from a uniform distribution in range from 0 to 1. The n is then compared to $p(r=1|\alpha_{\text{observer}},\beta_{\text{observer}},x_1)$. If $n \leq p(r=1|\alpha_{\text{observer}},\beta_{\text{observer}},x_1)$, the response r_1 is assumed to be 1. If $n_i > p(r=1|\alpha_{\text{observer}},\beta_{\text{observer}},x_1)$, the response r_1 is assumed to be 0.

7. The prior probability $p_1(r|x)$ (matrix with shape 101×101) is updated based on r_1 by selecting the sub-matrix from posterior probability $p_0(\alpha,\beta|x,r)$ (matrix with shape $51 \times 101 \times 101$) with index that matches the stimulus x_1 (Equation 9). For example, if $r_1 = 1$, and index of x_1 in all stimulus x is t, $p_1(r|x)$ is the t-th element of $p_0(\alpha,\beta|x,r=1)$ at the 1$^{\text{st}}$ dimension.

8. The estimate of the threshold α_1 and slope β_1 (Equation

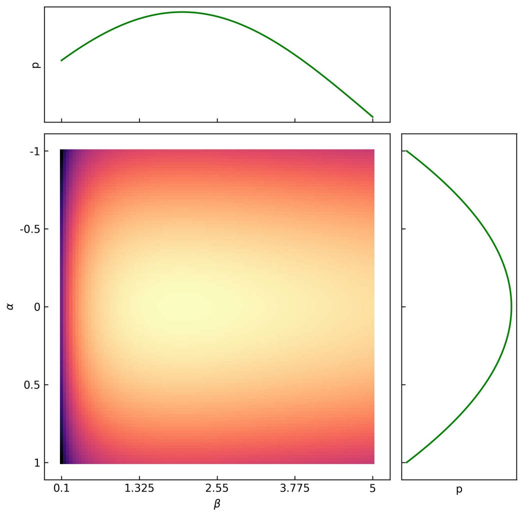

Figure 2: The prior probability distribution $p_0(\alpha, \beta)$. The probability of each combination of threshold α and slope β is indicated by the color of the prior heat-map, where lighter color means higher probability. The probability distribution of α and β are separately shown in green on the top and right of the prior heat-map, respectively.

10) is computed as the total sum of the product of the updated prior $p_1(r|x)$ and the sample space of α and β.

Then repeat the whole process for the rest of trials until the 50-th trial, which is set to be the end.

Results

A example run in which stimulus intensity was guided by the the Ψ method is shown in Figure 3. Correct responses are indicated by the filled symbols, incorrect responses are indicated by open symbols. The black line displays the running estimate of the threshold based on the posterior distribution. The responses were generated by a Gumbel function with $\alpha = 0$, $\beta = 2$, $\gamma = 0.5$, and $\lambda = 0$. The generating function (Ψ_{gen}) is shown on the right in green, the function fitted by the Ψ method (Ψ_{fit}) after completion of the 50 trials is shown on the right in black.

First, it is apparent that, at the start of the run, the Ψ method selects stimulus intensities that are at or near the current threshold estimate. This is, of course, the best placement rule to determine the value of the threshold. However, in order to gain information regarding the slope of the psychometric function, measurements need to be made at multiple points along the psychometric function. Indeed, as the run proceeds, the Ψ method also starts to select stimulus intensities well above and below the running threshold estimate.

The example result from Psychophysics textbook (Prins et al., 2016) is shown in Figure 4. The comparison between the simulated result and textbook example result shows great similarity. It suggests that I made a fair replication.

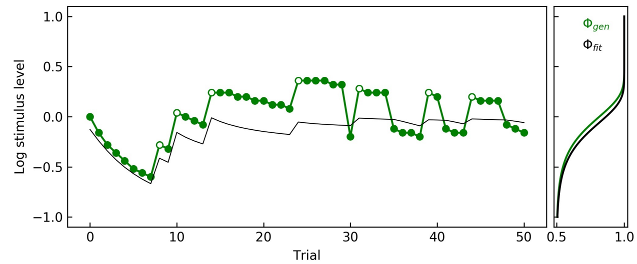

Figure 3: Result of the simulated Ψ method staircase. Correct responses are indicated by the filled symbols, incorrect responses are indicated by open symbols. The black line displays the running estimate of the threshold based on the posterior distribution. The responses were generated by a Gumbel function with $\alpha = 0$, $\beta = 2$, $\gamma = 0.5$, and $\lambda = 0$. The generating function (Ψ_{gen}) is shown on the right in green, the function fitted by the Ψ method (Ψ_{fit}) after completion of the 50 trials is shown on the right in black.

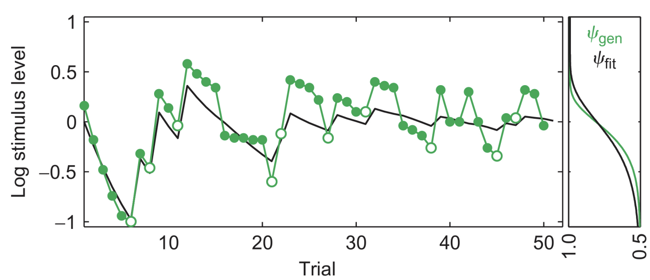

Figure 4: Example of a simulated Ψ method staircase from Psychophysics textbook (Prins et al., 2016). The same plotting conventions are used here as in Figure 3.

Discussion

For every single trial, the Ψ method considers a range of possible stimulus intensities to use on the next trial and for each calculates what the probabilities of a correct response and incorrect response are. It also considers the entropy which would result from both a correct response and an incorrect response. From these, the Ψ method calculates the expected entropy. Then it selects the stimulus intensity that is expected to result in the lowest entropy. That briefly summaries the whole process which the Ψ method does.

In the definition of the psychometric function, I purposely set the guess rate γ and lapse rate λ to fixed number. I did so to reduce the complexity of this replication. These two parameters can be set as free parameters just like the threshold and slope. In fact, the core idea behind the Ψ method is to select stimulus intensities such that the expected information gain regarding the value of parameters is maximized. This principle can be easily extended to the optimization of any parameter. One obvious extension to the Ψ method is to define the posterior distribution across any combination of the four parameters. Prins (2013) had already introduced the Ψ-marginal adaptive method that emphasized on estimating these four parameters. And it's available in the Palamedes toolbox (Prins et al., 2018).

The Ψ method is, in many ways, similar to the QUEST method (Watson & Pelli, 1983). But it's more efficient than the QUEST method, since it estimates not only the threshold parameter but both threshold and slope parameters (the Ψ-marginal method even estimates all four). The slope parameter is important because the slope of a psychometric function ultimately describes the noisiness of the results. Questions related to perceptual learning on the noisiness of the perceptual system may be answered by exploring the slope of the psychometric function.

References

Breakwell, G. M., Hammond, S. E., Fife-Schaw, C. E., & Smith, J. A. (2006). *Research methods in psychology*. Sage Publications, Inc.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... Oliphant, T. E. (2020, September). Array programming with NumPy. *Nature*, *585*(7825), 357–362.

Kontsevich, L. L., & Tyler, C. W. (1999). Bayesian adaptive estimation of psychometric slope and threshold. *Vision Research*, *39*(16), 2729-2737.

Leek, M. R. (2001). Adaptive procedures in psychophysical research. *Perception & psychophysics*, *63*(8), 1279–1292.

Macmillan, N. A., & Creelman, C. D. (2004). *Detection theory: A user's guide*. Psychology press.

Prins, N. (2013). The psi-marginal adaptive method: How to give nuisance parameters the attention they deserve (no more, no less). *Journal of vision*, *13*(7), 3–3.

Prins, N., et al. (2016). *Psychophysics: a practical introduction*. Academic Press.

Prins, N., et al. (2018). Applying the model-comparison approach to test specific research hypotheses in psychophysical research using the palamedes toolbox. *Frontiers in psychology*, *9*, 1250.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, *27*(3), 379–423.

Sims, J. A., & Pelli, D. (1987). The ideal psychometric procedure. In *Investigative ophthalmology and visual science (supplement.*

Van Rossum, G., & Drake, F. L. (2009). *Python 3 reference manual*. Scotts Valley, CA: CreateSpace.

Watson, A. B., & Pelli, D. G. (1983). Quest: A bayesian adaptive psychometric method. *Perception & psychophysics*, *33*(2), 113–120.