

# MemToolbox tutorial

(This tutorial uses MTB X.X, and was last updated on 6/6/2012.)

The MemToolbox is ...

**Example 1:** Mixture modeling in just two lines of code.

The simplest way to use the MemToolbox is through `MemFit`, a function that houses much of the toolbox's functionality under one roof. A full analysis of your data using `MemFit` is just two steps away. First, you'll need to specify your data as a vector of errors, one for each trial, in units of degrees on the color wheel.

```
>> errors = [-89, 29, -2, 6, -16, 65, 43, -12, 10, 0, 178, -42, 52, 1, -2];
```

Next, call `MemFit` on the error vector.

```
>> MemFit(errors);
```

The toolbox will then run through its analysis of your data, showing you a histogram of the errors, the name and parameterization of the model it is fitting to the data, and the maximum a posteriori and credible intervals of the model parameters inferred from the data. Then it will ask whether you'd like to see the fit.

```
Error histogram:  -180 _____.'_'_____ +180
                  Model:  Standard mixture model with bias
                  Parameters:  mu, g, sd
```

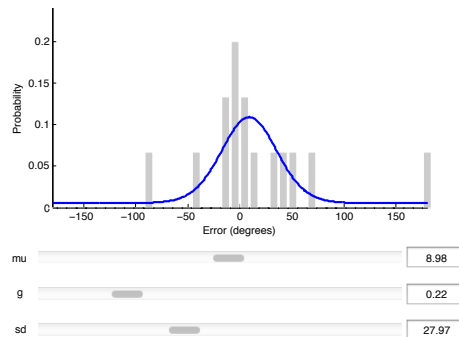
```
Just a moment while MTB fits a model to your data...
```

```
...finished. Now let's view the results:
```

parameter	MAP estimate	lower CI	upper CI
mu	7.490	-18.367	29.573
g	0.219	0.037	0.666
sd	27.827	8.963	68.719

```
Would you like to see the fit? (y/n): y
```

If you respond in the affirmative (by entering the letter `y` and hitting the return key), you'll see an interactive visualization that lets you manipulate the model parameters and see their impact on the predicted distribution of data.



Hooray. In just two lines of code, you've performed mixture modeling using the MemToolbox.

### Example 2: Choosing a different model.

In the previous example, the toolbox picked a model for you. Choosing a different model is easy. The toolbox comes preloaded with a number of them, and you can see a full listing by typing `help MemModels` at the command line. One of the models, `StandardMixtureModel`, is identical to the model used in the previous example, except that the bias parameter  $\mu$  is constrained to be zero: the model is unbiased. The first step is to pick a model:

```
>> model = StandardMixtureModel();
```

The second step is to specify your data:

```
>> errors = [-89, 29, -2, 6, -16, 65, 43, -12, 10, 0, 178, -42, 52, 1, -2];
```

Finally, call `MemFit`:

```
>> MemFit(errors, model);
```

The toolbox will now walk through the same analysis as before, but using the model that you chose.

### Example 3: Specifying your data as a structure with auxiliary variables.

So far, the two models we have used operate solely over the errors made by the participant on the task. But some models require auxiliary data. For example, the swap model advocated in Bays & Husain (2009) requires not only the errors but also the values of the non-target items.

### Example 4: Using the built-in data sets.

The toolbox comes loaded with a number of data sets. These can be accessed through the function `MemData`. For example:

```
>> data = MemData(1)

data =

    errors: [3120x1 double]
 subjectID: 's1'
         n: [3120x1 double]
```

[Some note about where to find a description of each data set.]

### Example 5: Peeking under the hood at MCMC.

Description of MCMC. How to read the various plots (converging chains, tradeoffs between parameters, samples from the posterior distribution, and posterior predictive checks).

### Example 6: Using the parallel toolbox to speed up the analysis.

(Just show how to open up the pool.)

### Example 7: Model comparison.

A good example for this would be to start with `StandardMixtureModel` vs. `StandardMixtureModelWithBias`, and point out how the posterior predictive check shows you exactly where the unbiased one fails. Then compare `StandardMixtureModelWithBias` to Student's  $t$  model, and again point out how the posterior predictive check shows you where the non-peaked one fails.

**Example 8:** Creating your own model.