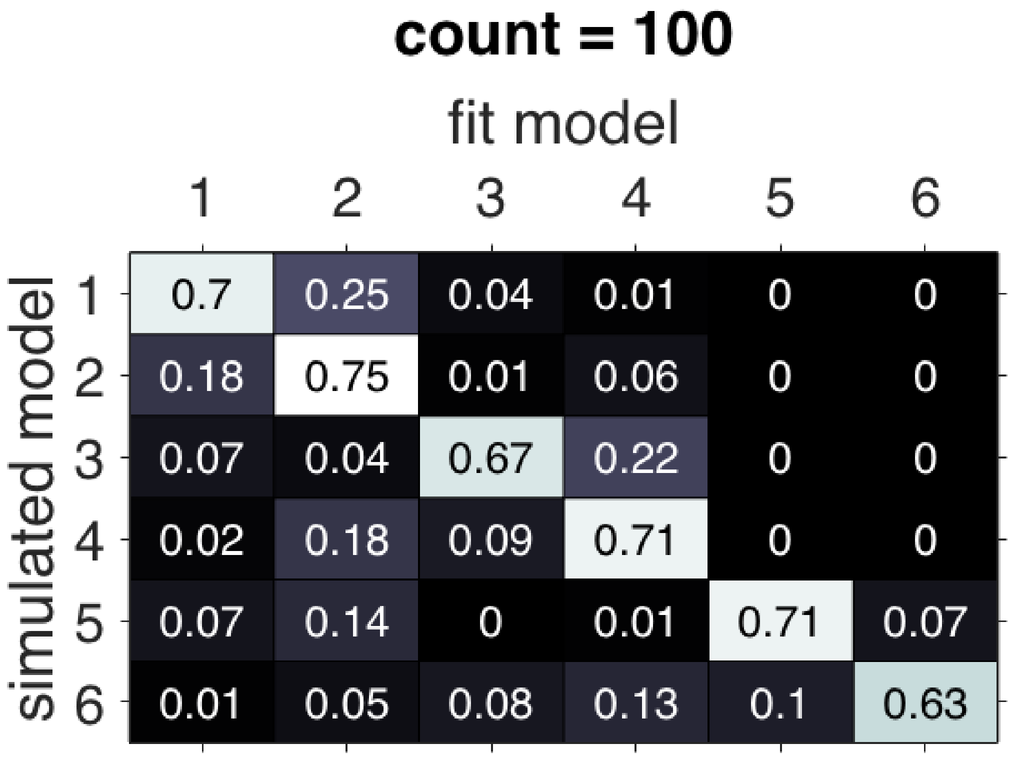
```
% a04_p4_model_recovery

% This script sets ExpInfo.nTrials = 20 and counts = 100 to do model
% recovery and generates a confusion matrix

clear all; close all; clc; rng(1); %parpool(20);

%% initiate confusion matrix

nModel = 6;
CM = zeros(nModel);
counts = 100;

%% experimental parameters

ExpInfo.ifi              = 1000/60;
ExpInfo.SOA              = [-0.5000, -0.3000, -0.2500, -0.2000, -0.1500, -0.1000,...
    -0.0500, 0, 0.0500, 0.1000, 0.1500, 0.2000, 0.2500, 0.3000, 0.5000]*1000; % in ms
ExpInfo.lenS             = length(ExpInfo.SOA);
ExpInfo.nTrials          = 20; % num of trials per SOA

%% ground truth parameters

% used the ground truth parameters from sub7 sess 7 SOA = -300 ms
TruePara                 = {[40, -20, 70, 50, 110, 80, 0.02, 0.04], ...%M1
    [40, -20, 70, 110, 80, 0.02, 0.04], ...%M2
    [40, -20, 70, 50, 110, 0.02, 0.04], ...%M3
    [40, -20, 70, 110, 0.02, 0.04], ...%M4
    [40, 70, 110, 80, 0.02, 0.04], ...%M5
    [40, 70, 110, 0.02, 0.04]};%M6

ParaLabel = {{'\mu_{pre}','\mu_{post}','\sigma_{pre}','\sigma_{post}','criterion_{pre}
    {'\mu_{pre}','\mu_{post}','\sigma','criterion_{pre}','criterion_{post}','\lambda_{
    {'\mu_{pre}','\mu_{post}','\sigma_{pre}','\sigma_{post}','criterion','\lambda_{pre
    {'\mu_{pre}','\mu_{post}','\sigma','criterion','\lambda_{pre}','\lambda_{post}'},.
    {'\mu','\sigma','criterion_{pre}','criterion_{post}','\lambda_{pre}','\lambda_{pos
    {'\mu','\sigma','criterion','\lambda_{pre}','\lambda_{post}'}};

%% functions

% define PMF
P_Afirst                 = @(SOA, mu, sig, c, lambda) lambda/3 + (1-lambda).*normcdf(-
P_Vfirst                 = @(SOA, mu, sig, c, lambda) lambda/3 + (1-lambda).*(1 - norm
P_simultaneous           = @(SOA, mu, sig, c, lambda) ...
    1 - (lambda/3 + (1-lambda).*normcdf(-c, SOA - mu, sig)) ...
    - (lambda/3 + (1-lambda).*(1 - normcdf(c, SOA-mu, sig)));

M1 = {
    @(p) P_Vfirst(ExpInfo.SOA, p(1), p(3), p(5), p(7));...
    @(p) P_simultaneous(ExpInfo.SOA, p(1), p(3), p(5), p(7));...
    @(p) P_Afirst(ExpInfo.SOA, p(1), p(3), p(5), p(7));...
    @(p) P_Vfirst(ExpInfo.SOA, p(2), p(4), p(6), p(8));...
    @(p) P_simultaneous(ExpInfo.SOA, p(2), p(4), p(6), p(8));
    @(p) P_Afirst(ExpInfo.SOA, p(2), p(4), p(6), p(8))};

M2 = {
    @(p) P_Vfirst(ExpInfo.SOA, p(1), p(3), p(4), p(6));...
    @(p) P_simultaneous(ExpInfo.SOA, p(1), p(3), p(4), p(6));...
    @(p) P_Afirst(ExpInfo.SOA, p(1), p(3), p(4), p(6));...
    @(p) P_Vfirst(ExpInfo.SOA, p(2), p(3), p(5), p(7));...
    @(p) P_simultaneous(ExpInfo.SOA, p(2), p(3), p(5), p(7));...
    @(p) P_Afirst(ExpInfo.SOA, p(2), p(3), p(5), p(7))};

M3 = {
    @(p) P_Vfirst(ExpInfo.SOA, p(1), p(3), p(5), p(6));...
    @(p) P_simultaneous(ExpInfo.SOA, p(1), p(3), p(5), p(6));...
    @(p) P_Afirst(ExpInfo.SOA, p(1), p(3), p(5), p(6));...
    @(p) P_Vfirst(ExpInfo.SOA, p(2), p(4), p(5), p(7));...
    @(p) P_simultaneous(ExpInfo.SOA, p(2), p(4), p(5), p(7));...
    @(p) P_Afirst(ExpInfo.SOA, p(2), p(4), p(5), p(7))};

M4 = {
    @(p) P_Vfirst(ExpInfo.SOA, p(1), p(3), p(4), p(5));...
    @(p) P_simultaneous(ExpInfo.SOA, p(1), p(3), p(4), p(5));...
    @(p) P_Afirst(ExpInfo.SOA, p(1), p(3), p(4), p(5));...
    @(p) P_Vfirst(ExpInfo.SOA, p(2), p(3), p(4), p(6));...
    @(p) P_simultaneous(ExpInfo.SOA, p(2), p(3), p(4), p(6));...
    @(p) P_Afirst(ExpInfo.SOA, p(2), p(3), p(4), p(6))};

M5 = {
    @(p) P_Vfirst(ExpInfo.SOA, p(1), p(2), p(3), p(5));...
    @(p) P_simultaneous(ExpInfo.SOA, p(1), p(2), p(3), p(5));...
    @(p) P_Afirst(ExpInfo.SOA, p(1), p(2), p(3), p(5));...
    @(p) P_Vfirst(ExpInfo.SOA, p(1), p(2), p(4), p(6));...
    @(p) P_simultaneous(ExpInfo.SOA, p(1), p(2), p(4), p(6));...
    @(p) P_Afirst(ExpInfo.SOA, p(1), p(2), p(4), p(6))};

M6 = {
    @(p) P_Vfirst(ExpInfo.SOA, p(1), p(2), p(3), p(4));...
    @(p) P_simultaneous(ExpInfo.SOA, p(1), p(2), p(3), p(4));...
    @(p) P_Afirst(ExpInfo.SOA, p(1), p(2), p(3), p(4));...
    @(p) P_Vfirst(ExpInfo.SOA, p(1), p(2), p(3), p(5));...
    @(p) P_simultaneous(ExpInfo.SOA, p(1), p(2), p(3), p(5));...
    @(p) P_Afirst(ExpInfo.SOA, p(1), p(2), p(3), p(5))};

models = {M1; M2; M3; M4; M5; M6};

%% run model fitting for multiple times

% initialization
[AIC, min_NLL, estP] = deal(cell(counts, nModel));

for count = 1:counts
    count

    %% simulate fake datasets for 6 models
    parfor iModel                   = 1:nModel
```

```
        % for the specific model and its corresponding ground-truth
        % parameter sets, generate the probability of reporting: Vfirst,
        % simultaneous, Afirst in pretest; Vfirst, simultaneous, Afirst in
        % posttest
        p_all_cond               = NaN(6, ExpInfo.lenS); % len of conditions x len of S
        Para                     = TruePara{iModel}; % true parameters for this model
        for iCondition           = 1:6
            p_all_cond(iCondition,:) = models{iModel}{iCondition}(Para);
        end

        % simulate data set
        sim_r_org                = cell(1,2);
        for s                    = 1:2 % loop through pre and post session
            p                        = p_all_cond(((s-1)*3+1):s*3, :); % pre and post p
            sim_r_org{s}             = sampleMatrix(p, ExpInfo.nTrials);
        end

        %% fit fake data to 6 models

        % obtain best-fitting parameters and AIC for each model
        [AIC{count, iModel}, min_NLL{count, iModel}, estP{count, iModel}]  = model_rec

        % iBest is the index of model that best fits this specific fake data set
        [M iBEST] = min(AIC{count, iModel});
        BEST = AIC{count, iModel} == M;
        BEST = BEST / sum(BEST);
        CM(iModel,:) = CM(iModel,:) + BEST;

    end
end
```

```
count = 1
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 8).
count = 2
count = 3
count = 4
count = 5
count = 6
count = 7
count = 8
count = 9
count = 10
count = 11
count = 12
count = 13
count = 14
count = 15
count = 16
count = 17
count = 18
count = 19
count = 20
count = 21
count = 22
count = 23
count = 24
count = 25
count = 26
count = 27
count = 28
count = 29
count = 30
count = 31
count = 32
count = 33
count = 34
count = 35
count = 36
count = 37
count = 38
count = 39
count = 40
count = 41
count = 42
count = 43
count = 44
count = 45
count = 46
count = 47
count = 48
count = 49
count = 50
count = 51
count = 52
count = 53
count = 54
count = 55
count = 56
count = 57
count = 58
count = 59
count = 60
count = 61
count = 62
count = 63
count = 64
count = 65
count = 66
count = 67
count = 68
count = 69
count = 70
count = 71
count = 72
count = 73
count = 74
count = 75
count = 76
count = 77
count = 78
count = 79
count = 80
count = 81
count = 82
count = 83
count = 84
count = 85
count = 86
count = 87
count = 88
count = 89
count = 90
count = 91
count = 92
count = 93
count = 94
count = 95
count = 96
count = 97
count = 98
count = 99
count = 100
```

```
%% summary plot of CM

figure(1); clf;
FM = round(100*CM/sum(CM(1,:)))/100;
t = imageTextMatrix(FM);
colormap(bone)
set(t(FM'<0.3), 'color', 'w')
hold on;
[l1, l2] = addFacetLines(CM);
set(t, 'fontsize', 22)
title(['count = ' num2str(count)]);
set(gca, 'xtick', [1:6], 'ytick', [1:6], 'fontsize', 28, ...
    'xaxislocation', 'top', 'tickdir', 'out')
xlabel('fit model')
ylabel('simulated model')
```



	1	2	3	4	5	6
1	0.7	0.25	0.04	0.01	0	0
2	0.18	0.75	0.01	0.06	0	0
3	0.07	0.04	0.67	0.22	0	0
4	0.02	0.18	0.09	0.71	0	0
5	0.07	0.14	0	0.01	0.71	0.07
6	0.01	0.05	0.08	0.13	0.1	0.63