

Manual PyMTS

1º Passo: Baixar o arquivo “PyMTS_vs2.04.zip” e extrair em uma pasta nomeada “PyMTS”.

Você **não** pode alterar os nomes dos arquivos e pastas dentro da pasta “PyMTS”. Eles serão utilizados por você para programar o experimento.

2º Passo: Adicionar todos os estímulos que serão utilizados no experimento na pasta “stimuli” (isso inclui instruções, estímulos modelo e comparação, consequências etc.).

3º Passo: Adicionar todos os arquivos de blocos de tentativas na pasta “config” (para mais informações sobre como escrever os arquivos ‘csv’, ver sessão “Configurando os blocos de tentativa”).

4º Passo: Abrir o arquivo “configData.json” e escrever em cada uma das linhas. Em cada linha há um primeiro nome entre aspas (ex., “instructions”). Você **não** pode alterar nenhum destes nomes. A descrição do que significa cada um deles e de como preencher cada linha você encontra na sessão “Configurando o experimento”.

5º Passo: Clicar duas vezes no arquivo “PyMTS.exe”, colocar o nome do(a) participante e do(a) experimentador(a) e clicar em “enviar”. O programa deve começar logo em seguida.

6º Passo: O arquivo com os dados da sessão será salvo na pasta “data” com o nome que você colocou para o(a) participante. Nós aconselhamos você a deletar este arquivo desta pasta após a finalização da sessão experimental.

Configurando os blocos de tentativas

Para cada bloco de ensino ou teste você deverá criar um arquivo de ‘csv’. Um arquivo csv pode ser feito utilizando o Excel, e salvando como “CSV UTF-8 (Comma delimited) (*.csv)”. Como exemplo, caso você queira apresentar um bloco de treino das relações AB, um bloco de treino das relações AC e um bloco de teste das relações BC e CB, você deverá adicionar três arquivos ‘csv’ com nomes diferentes (ver arquivo de exemplo).

Na pasta que você recebeu já está disponível um arquivo modelo para ser utilizado. Você deve encontrar dez colunas nomeadas: “sample”, “sample_sound”, “comp”, “correct_comp”, “img_right”, “img_wrong”, “sound_right”, “sound_wrong”, “time_right” e “time_wrong”. Estes nomes **não** podem ser alterados.

As linhas **abaixo** desta primeira linha representam as tentativas (cada linha é uma tentativa), e o número de linhas é a quantidade de tentativas que cada bloco de treino/teste terá (ex., caso seja um bloco de 24 tentativas, o arquivo ‘csv’ deverá ter, ao final, 25 linhas – contando com a primeira linha nomeando as colunas). Cada uma das colunas está explicada abaixo.

sample. Nesta coluna você deverá colocar o nome do arquivo que será apresentado como modelo visual, incluindo o tipo de arquivo. Como exemplo, caso seja um arquivo ‘.jpg’, você deverá incluir essa informação (“A1.jpg” – **sem** as aspas).

sample_sound. Nesta coluna você deverá colocar o nome do arquivo que será apresentado como modelo auditivo, incluindo o tipo de arquivo. Como exemplo, caso seja um arquivo ‘.mp3’, você deverá incluir essa informação (“sA1.mp3” – sem as aspas). Caso nenhum estímulo seja apresentado, coloque a letra “n” (**sem** aspas).

comp. Nesta coluna você deverá colocar todos os estímulos comparação que serão apresentados (utilizando a mesma formatação que o modelo – adicionando o tipo de arquivo no nome), separados por vírgula (ex., “B1.jpg,B2.jpg,B3.jpg” – Atenção para essa parte, você

deve separar os nomes **APENAS** com vírgula, sem adicionar espaço entre a vírgula e o nome do estímulo, e **sem** as aspas).

correct_comp. Nesta coluna você deve colocar o estímulo comparação que será considerado o estímulo correto naquela tentativa (ex., “B1.jpg” – **sem** as aspas).

img_right. Nesta coluna você deve adicionar a imagem que será apresentada caso a pessoa selecione corretamente (a consequência para acerto). Lembre-se que o PyMTS só aceita imagens e que o nome deve incluir o tipo de arquivo (ex., “img_right.png” – sem as aspas). Caso nenhum estímulo seja apresentado, coloque a letra “n” (**sem** aspas).

img_wrong. Nesta coluna você deve adicionar a imagem que será apresentada caso a pessoa selecione incorretamente (a consequência para erro). Lembre-se de incluir o tipo de arquivo (ex., “img_wrong.png” – **sem** as aspas). Caso nenhum estímulo seja apresentado, coloque a letra “n” (**sem** aspas).

sound_right. Nesta coluna você deve adicionar o som que será apresentado caso a pessoa selecione corretamente (a consequência para acerto). Lembre-se de incluir o tipo de arquivo (ex., “sound_right.wav” – **sem** as aspas). O som deve ter a duração que a consequência será apresentada. Caso nenhum estímulo seja apresentado, coloque a letra “n” (**sem** aspas).

sound_wrong. Nesta coluna você deve adicionar o som que será apresentado caso a pessoa selecione incorretamente (a consequência para erro). Lembre-se de incluir o tipo de arquivo (ex., “sound_wrong.wav” – **sem** as aspas). O som deve ter a duração que a consequência será apresentada. Caso nenhum estímulo seja apresentado, coloque a letra “n” (**sem** aspas).

Nas colunas “sample_sound”, “img_right”, “img_wrong”, “sound_right” e “sound_wrong”, caso você **não** queira apresentar algum dos estímulos (ou caso seja um bloco

de teste), coloque a letra “n” (**sem** aspas) no lugar do nome dos estímulos. Você **não** deve excluir as colunas.

time_right. Nesta coluna você deve colocar o tempo que será apresentada a consequência de acerto. Caso nenhuma consequência esteja definida, a tela ficará em preto pelo tempo definido nesta variável.

time_wrong. Nesta coluna você deve colocar o tempo que será apresentada a consequência de erro. Caso nenhuma consequência esteja definida, a tela ficará em preto pelo tempo definido nesta variável.

Você **não** precisa randomizar as tentativas, o programa irá fazer isso automaticamente, atente-se apenas em escrever corretamente todas as linhas. Não há problema em copiar e colar as linhas, você pode fazer uma linha como exemplo e copiar, mudando apenas as informações necessárias.

Configurando o experimento

As configurações gerais do experimento são apresentadas no arquivo “configData.json”. Cada linha se refere a uma característica do experimento e será descrita individualmente abaixo. As linhas já estão preenchidas com modelos que podem ser utilizados para guiar a programação. Não altere a configuração das linhas e siga a formatação que está estabelecida no arquivo modelo.

"screen_color". Aqui você deve colocar a cor que será apresentada no fundo da tela. Para modificar a cor, basta mudar os números dentro das chaves (ex., preto = [0, 0, 0] e branco = [255, 255, 255]). Esta é uma variável que será apresentada em todas as tentativas.

"ITI". Aqui você deve colocar o tempo do intervalo entre tentativas (ITI) em segundos. Caso não queira apresentar intervalo entre as tentativas, basta colocar o número zero. Esta é uma variável que será apresentada em todas as tentativas.

"start_block". Esta é uma variável de controle. Nela você deve colocar o bloco que será iniciado quando abrir o programa. Ela serve para você testar o experimento (ex., caso você queira testar apenas o último bloco para ver se ele está funcionando) ou para os casos nos quais o participante atingiu o número máximo de repetições e você deve decidir se ele finaliza o experimento ou se continua (ex., caso você queira que ele continue, mas que volte a fazer algum bloco de treino, é só colocar aqui qual o bloco que ele deve iniciar). O valor zero representa o primeiro bloco, portando, você deve colocar o número do bloco subtraído de um (ex., se você quiser colocar o terceiro bloco, coloque o número dois).

"blocks". Nesta linha você deverá colocar, dentro das chaves, e separado por vírgulas, os nomes dos arquivos ‘csv’ que estão na pasta “config”, na ordem que eles serão apresentados. Como exemplo, caso você queira apresentar os blocos “treinoAB”, “treinoAC” e “testeEq” nesta ordem, deve escrever: ["treinoAB", "treinoAC", "teste_eq"] (**com** as aspas). É importante se atentar ao número de blocos, pois isso será importante nas linhas seguintes.

"instructions". Nesta linha você deve adicionar as instruções que deseja apresentar antes de cada bloco. Como exemplo, caso você queira apresentar as instruções iniciais antes do primeiro bloco (treinoAB) e antes do bloco de teste (testeEq), você deve escrever: ["inst1.jpg","n", "inst2.jpg"] (**com** as aspas). Aqui, caso você não queira apresentar uma instrução, basta colocar a letra “n” (**com** as aspas). As instruções deverão ser apresentadas em formato de imagem e você deve colocar o nome do arquivo de imagem que será utilizado dentro das chaves. Lembre-se que dentro desta chave deve ter o mesmo número de itens que tem na linha “blocks”. Como exemplo, se a linha “blocks” tem três itens ("treinoAB", "treinoAC", "teste_eq"), a linha de “protocol” deve ter três itens também ("inst1.jpg","n", "inst2.jpg").

"protocol". Nesta linha você deve colocar o tipo de protocolo utilizado em cada um dos blocos. Caso seja um *Delayed Matching to Sample* (DMTS) você deve colocar o valor que será apresentado em **segundos**. Caso seja um *Simultaneous Matching to Sample* (SMTS), você deve escrever “SMTS” (**com** as aspas). Como exemplo, suponhamos que você queira fazer um DMTS de zero segundos no “treinoAB” e no “treinoAC”, e utilizar um SMTS no “testeEq”, você deve escrever: [0, 0, "SMTS"] (SMTS **com** as aspas; números **sem** aspas). Lembre-se de deixar a mesma quantidade de itens que tem na linha “blocks”.

"repetitions". Nesta linha você deve escrever a quantidade de repetições que cada bloco deve ter. Zero significa que ele não terá nenhuma repetição. Como exemplo, caso você queira que os blocos “treinoAB” e “treinoAC” tenham no máximo 3 apresentações (máximo de 2 repetições) e o bloco “testeEq” só uma (máximo de 0 repetições), você deve escrever: [2, 2, 0] (**sem** aspas). Lembre-se de deixar a mesma quantidade de itens que tem na linha “blocks”.

"criteria". Nesta linha você deve escrever o critério para finalizar o bloco com a quantidade de tentativas corretas que ele deve fazer. Como exemplo, suponhamos que no

“treinoAB” e no “treinoAC” (ambos com 24 tentativas), você quer que o participante acerte 100% do bloco, e no “testeEq” (com 24 tentativas também), você quer que o participante acerte no mínimo 91,7% (equivalente a 22 tentativas), você deve escrever: [24, 24, 22] (**sem** aspas). Lembre-se de deixar a mesma quantidade de itens que tem na linha “blocks”.

"pos_sample". Nesta linha você deve colocar as coordenadas para a posição que o estímulo modelo será apresentado. A posição [0, 0] representa o centro da tela, o primeiro item representa o eixo X e o segundo item o eixo Y. Neste exemplo, caso você queira apresentar o estímulo modelo no centro do eixo X mas um pouco acima do centro no eixo Y, você deve colocar algo como: [0, 200] (**sem** aspas). Você pode alterar esses valores e ir testando as posições que preferir.

"pos_comps". Nesta linha você deve colocar as coordenadas para a posição que os estímulos comparação serão apresentados. Para cada estímulo comparação que for apresentado, deve ser adicionada uma posição na tela (seguindo a mesma formatação de colocar os dois números dentro de chaves [X, Y]). Neste exemplo, caso você queira apresentar dois estímulos comparação à esquerda e à direita da parte inferior da tela, você pode escrever: [[-350, -250], [350, -250]] (**sem** aspas). Cada posição deve ser colocada entre chaves, dentro da chave principal, e separada por vírgulas. Você pode alterar esses valores e ir testando as posições que preferir.

"stimulus_size". Nesta linha você deve colocar o tamanho dos estímulos que serão utilizados como modelo e comparação. Nós aconselhamos que você padronize a proporção de todos os estímulos utilizados (ex., 1:1). Neste caso, você também deve colocar entre chaves os valores do eixo X e Y (ex., [250, 250] – **sem** aspas). Você pode alterar esses valores e ir testando o tamanho que preferir.

"consequence_size". Nesta linha você deve colocar o tamanho dos estímulos que serão utilizados como consequência. Nós aconselhamos que você padronize a proporção de

todos os estímulos utilizados como consequência (ex., 4:3). Neste caso, você também deve colocar entre chaves os valores do eixo X e Y (ex., [1800,900] – **sem** aspas). Você pode alterar esses valores e ir testando o tamanho que preferir.

“instructions_size”. Nesta linha você deve colocar o tamanho dos estímulos que serão utilizados como instruções. Nós aconselhamos que você padronize a proporção de todos os estímulos utilizados como instrução (ex., 4:3). Neste caso, você também deve colocar entre chaves os valores do eixo X e Y (ex., [1200,900] – **sem** aspas). Você pode alterar esses valores e ir testando o tamanho que preferir.

“end_text”. Nesta linha você deve colocar o nome do arquivo que será apresentado caso a pessoa termine o experimento ou atinja o número limite de repetições que você definiu na linha “repetitions”. Este arquivo deve estar dentro da pasta “stimuli” (ex., “end_text.png” – **com** aspas).

Os estímulos visuais como dicas e consequências podem ser feitos utilizando programas como o PowerPoint e salvando o arquivo como imagem. Todas as linhas do “configData.json” devem finalizar com uma vírgula (ver arquivo modelo que foi enviado), com exceção da última linha (“end_text”). Você **não** pode alterar o nome das linhas, a indexação (posição que as linhas aparecem e o espaço que elas são apresentadas como parágrafo), nem retirar os símbolos de conjunto “{ }”.

Qualquer dúvida, envie um e-mail para alceuregaco@ufscar.br e acompanhe as atualizações do software na página de GitHub do PyMTS:

<https://github.com/AlceuRegaco/PyMTS>

Dados gerados pelo programa

Participant. O nome ou código do participante.

Experimenter. O nome ou código do pesquisador.

Date. A data da realização do procedimento

Total_Trial. O número total de tentativas do experimento.

Block. O nome do bloco ao qual aquela linha pertence.

Block_Trial. O número total de tentativas do bloco.

Accuracy. Se naquela tentativa o participante acertou (1) ou errou (0).

Total_Correct. O número de respostas corretas acumulada no bloco (esse valor zera quando o participante repete ou atinge o fim do bloco).

Sample. O estímulo modelo visual.

Sample_Sound. O estímulo modelo auditivo.

Comps. Os estímulos comparação respectivamente na posição em que foram apresentados (ex., se no documento “configData.json” você colocou dois espaços para apresentação do estímulo modelo na linha “pos_comp” – [[Posição X], [Posição Y]] – se foi salva a informação ['B1.png', 'B2.png'], quer dizer que o B1 foi apresentado na Posição X e o B2 na Posição Y).

Selected_Comp. O estímulo comparação selecionado pelo participante.

Time_Click_Sample. O tempo até o participante clicar no estímulo modelo visual.

Time_Click_Sample. O tempo até o participante selecionar o estímulo comparação.

Time_Trial. O tempo total da tentativa.