

Using the email function

Typing help email in the command window will bring up:

```
>> help email  
-----help for email.m-----
```

email:

This function emails recipient(s) with included attachment(s).

The syntax is

```
email(emails,attachments,ARG1,ARG2)
```

"emails" is either a email address or a cell array of multiple email addresses.

"attachments" is either a single path name to a file or a cell array of multiple path names.

ARG1 and ARG2 contain the subject line and the message text. Both are optional, and can be in any order.

However:

subject **MUST** be a single string that will be used in the subject line of the email message **MUST** be a cell array. Note that even though it is a cell array, it can contain only 1 element.

If no argument is given for subject, it defaults to "Gallistel Lab." If no argument is given for message, it defaults to "Here are the attachments for: [date]."

see also sendmail.m

In example 1, we are only sending 3 of the 15 figures created, so we define those separately. Then we may want to include the data files in the email, so the code shows how to take the last 12 loaded sessions (from the structure) and include those. It also shows how to use multiple email addresses. Note that if you do not want to use `TSloadsessions` command but still send the data files, you will have to set each data path separately, like we do for the figures here.

In example 2, we send all of the figures created, and we don't care about sending the data files. Here we only use 1 email address, but show how to use the message and subject arguments

Example 1:

```
if sendFigures
    % sendFigures is a true or false flag for whether or not to send
    % the figures
    Figure13 = 'C:\..some path in here...\Figure13.fig'
    Figure14 = 'C:\..some path in here...\Figure14.fig'
    Figure15 = 'C:\..some path in here...\Figure15.fig'
    % here we define the figures that we will send. 'path x' is the full
    % path to that file.

    attachments = {Figure13, Figure14, Figure15};
    % here we put the Figures into a cell array called 'attachments'

if sendData
    pathName = 'X:\Documents and Settings\Gallistellab\Desktop\Aaron
Switch\Data\PhaseTHREE'
    % sendData is a flag for sending the data files, path name is the
    % directory the files are in.
    Files = Experiment.Info.FilesLoaded(end-11:end);
    %To get the names of the last 12 loaded sessions
    % command gets the last 12 sessions that TSloadsessions has
    %loaded. Note, if you only want the last 6, the code would be
    % (end-5:end)
    for i = 1:12 % for each file
        Files{i} = [pathName Files{i}]; % add paths to filenames
        %for each data file, add the pathName to the front of the filename to
        % get an entire file path
    End
    attachments = [attachments Files]
    % update attachments to include the data files.
end

    emails =
    {'freeston@eden.rutgers.edu', 'fbalci@ruccs.rutgers.edu', 'jackgib@eden.r
utgers.edu', 'chasetopher@gmail.com'};
    % sets emails to a cell array with the email addresses in it.
    email(emails, attachments);
    % sends the email.
    % Note here that since I did not specify a message or subject
    % they will default to 'Here are the attachments for: [date]'
    % and 'Gallistel Lab', respectively.
End
```

Example 2.

```
%% Email Figures Only
sub = Experiment.Subject(end).Id;
% get the subject ID
EmailFigsTemp = '';
CurrentDir = 'C:\..\some path here\Figures';
% declare EmailFigsTemp, which will be a variable that will
% hold the strings until we put them into a cell array
% CurrentDir is the path where the figures are.

for f=1:5;
    saveas(figure(f),['Figure' num2str(f) '_Subject' num2str(sub)...
'_Training.fig']);
    EmailFigsTemp = strvcat(EmailFigsTemp,[CurrentDir '\Figure' num2str(f)
... '_Subject' num2str(sub) '_Training.fig']);
end
%for the first 5 figures generated, save them and include them
%in EmailFigs Temp

attachments = cellstr(EmailFigsTemp); clear EmailFigsTemp;
% turn all the path names for the figures into a cell array

emails = 'freeston@eden.rutgers.edu';
%set email address. Note that in this example it is only a string, not
%a cell array of strings

email(emails,attachments,{'This is the body'},'This is the subject');
% emails the figures. Note that in this example I include a
% message and a subject. I could have swapped the last two
% arguments:
%email(emails,attachments,'This is the subject',{'This is the body'});
%but in either case, the body needs to be a cell array, and the
%subject needs to be a single string.
```