EGR115

Using `putcsv()`

From the help file included with the `getcsv()` and `putcsv()` .p files:

```
[status, msg] = putcsv(filename, A)

 Parameters:
    filename: name of file including extension, as a string

    A: scalar, numeric array, string, or cell array
       "A" must be only a scalar; string; or 2D cell array that contains
        only numeric and string values.  It may not contain any embedded
        arrays or cell arrays.

 Functionality:
    Writes cell or numeric array "A" to file "filename" as comma-separated values,
    overwriting the file if it exists.


 Return values:
    status: value of Boolean true if write was successful, false otherwise
    msg: Error string to assist debugging
```

Summary:

- `putcsv()` requires a file name character vector string as argument and an array, character vector, or scalar to write to that file. The file name must include the `.csv` extension.

- The file to be written should be in the same folder as the program, and must be available for writing (meaning it cannot be open in Excel, LibreOffice, or other locking programs).

- `putcsv()` returns two values:
  The first is a Boolean value: `true` if the write was successful, `false` otherwise.
  The second is a string message to assist in debugging. The string will be empty if the writing was successful.

- Always test for successful writing before continuing your program: check the return values! For example:

```
filename = 'MyFile.csv';
successful_write = putcsv(filename, CA);

if successful_write

  % code when write was successful...

else

  fprintf('Error writing to %s!', filename);

end
```

- When debugging, the second return value can be useful:

```
filename = 'MyFile.csv';
[successful_write, msg] = putcsv(filename, CA);

if successful_write

  % code when write was successful...

else

  fprintf('Error writing to %s!', filename);
  fprintf('Status message: %s', msg);

end
```