Table of Contents

```
% Modeling and Programming Methods to Streamline Biomaterial
 Development
% Daphne Sze
% November 2022

% Matlab script for Importing absorbance 260 data and Prism Fits
```

Import Excel file of Raw abs260 data

```
% Stores each data from each sheet in column of cell array
%   - Column 1: ID (defines what data is extracted in the row)
% row 1: sheet names
% row 2: Full data
% row 3: searches for 'pbs' case sensitive
% row 4: searches for 'NaDeo' case INsensitive
% row 5: searches for 'DNase' case INsensitive
% row 6: searches for 'DNase2' case INsensitive

% gets sheet names from file
                                  % change file name here %
sheets = ['ID',cellstr(sheetnames('102921_RawData.xlsx'))'];

%Import Data into cell array
for k=2:(numel(sheets))
  %Import full data table
  sheets{2,1} = 'RawTable';
  sheets{2,k} =
 readtable('102921_RawData.xlsx','sheet',k-1,'PreserveVariableNames',
 true);
  %Import PBS wash readings
  Ind = find(contains(table2cell((sheets{2,k}(:,1))),'pbs'));
  sheets{3,1} = 'pbs';
  if ~isempty(Ind) && Ind(1) > 1
    sheets{3,k} = sheets{2,k}([Ind(1)-1;Ind],:);
  elseif ~isempty(Ind)
    sheets{3,k} = sheets{2,k}(Ind,:);
  else
    sheets{4,k} = [];
  end
  %Import NaDeo readings
  Ind = find(contains(table2cell((sheets{2,k}
(:,1))),'NaDeo_','IgnoreCase',true));
  sheets{4,1} = 'NaDeo';
  if ~isempty(Ind) && Ind(1) > 1
      sheets{4,k} = sheets{2,k}([Ind(1)-1;Ind],:);
  elseif ~isempty(Ind)
     sheets{4,k} = sheets{2,k}(Ind,:);
  else
     sheets{4,k} = [];
  end
    %Import DNase readings
  Ind = find(contains(table2cell((sheets{2,k}
(:,1))),'DNase_','IgnoreCase',true));
  sheets{5,1} = 'DNase';
   if ~isempty(Ind) && Ind(1) > 1
       sheets{5,k} = sheets{2,k}([Ind(1)-1;Ind],:);
   elseif ~isempty(Ind)
       sheets{5,k} = sheets{2,k}(Ind,:);
   else
       sheets{5,k} = [];
   end
  Ind = find(contains(table2cell((sheets{2,k}
(:,1))),'DNase2_','IgnoreCase',true));
  sheets{6,1} = 'DNase2';
  if ~isempty(Ind) && Ind(1) > 1
      sheets{6,k} = sheets{2,k}([Ind(1)-1;Ind],:);
  elseif ~isempty(Ind)
      sheets{6,k} = sheets{2,k}(Ind,:);
  else
      sheets{6,k} = [];
  end
end

clearvars -except sheets
```

Normalize data and output to excel file

```
% Sets start time and absorbance to 0
% Excel file format :Each sheet should contain one set of data at A1


Normdat = sheets;
filename = 'NormData.xlsx';
for c=2:(size(Normdat,2))
    for r =3:(size(Normdat,1))
        if ~isempty(Normdat{r,c})
            Normdat{r,c}.Properties.VariableNames(2) = {'Time (min)'};
            %Find value of start time
            time = Normdat{r,c}{1,2};
            %find mean abs260 value at time 0
            norm = nanmean(Normdat{r,c}{1,3:end});
            %Remove norm value from all Reps
            Normdat{r,c}{:,3:end} = (Normdat{r,c}{:,3:end}) - norm;
            %Set time to start at 0
            Normdat{r,c}{:,2} = (Normdat{r,c}{:,2}) - time;
            EmpDat = convertvars(Normdat{r,c},@isnumeric,@NanEmp);
            %Save as excel and csv file
 writetable(EmpDat,filename,'Sheet',strcat(Normdat{1,c},'_',Normdat{r,1}));
 writetable(EmpDat(:,2:end),strcat('NormCSV/',Normdat{1,c},'_',Normdat{r,1},'.csv'
        end
    end
end

clearvars -except sheets Normdat
```

Import Fit from Prism

```
% Normalized data was fit to model in prism
% Fit values are imported along with relevant experimental variables

%Load Prism results file
FitVal = readcell('NormCSV/Results042322.txt');
%Initialize tables
Fitdat = Normdat;
FitTab = [{'Fig';'Reagent';'YM';'Y0';'k';'R squared'}];

%Define Mass, native DNA content and final DNA content used in decell
m= [1000;1000;500;1000;500;1000;1000;1000];
nDNA= [11.1;7.1;11.1;11.1;7.1;7.1;11.1;11.1];
fDNA= [2.3;0.8;3.2;1.2;0.1;0.1;2.5;2.5];
Mass =
 table(Fitdat(1,2:end)',repmat({'Mass'},length(m),1),m,'VariableNames',
{'Fig' 'Param' 'Val'});
NatDNA =
 table(Fitdat(1,2:end)',repmat({'NatDNA'},length(nDNA),1),nDNA,'VariableNames',
{'Fig' 'Param' 'Val'});
FinDNA =
 table(Fitdat(1,2:end)',repmat({'FinDNA'},length(fDNA),1),fDNA,'VariableNames',
{'Fig' 'Param' 'Val'});


%Insert empty rows
row = 3;
for r = 0:(size(Fitdat,1)-row) %For NaDeo and DNase
    n=row+(r*2);
    %add row to contain parameter data
    Fitdat(n+1:end+1,:) = Fitdat(n:end,:);
    Fitdat(n+1,:) = {[]};
    Fitdat{n+1,1} = strcat(Fitdat{n,1},'_Param');
    for c=2:(size(Normdat,2)) %For all sheets
        %find sheetname from Results.txt
         sheetname = find(strcmp(FitVal(:,1),strcat({'Nonlin fit of
 '},Fitdat{1,c},'_',Fitdat{n,1})));
         if ~isempty(sheetname) %Has data
             %Initailize table
             Fitdat{n+1,c} = cell2table(FitVal(sheetname+2:sheetname
+5,1:2),'VariableNames',{'Param' 'Val'});
```

```
             %Get Prism Fit data
             FitTab = [FitTab,
[Fitdat{1,c};Fitdat{n,1};FitVal(sheetname+2:sheetname+5,2)]];
             %Get Experimental Data
             Dur = cell2table({'Dur', Fitdat{n,c}
{end,2}},'VariableNames',{'Param' 'Val'});
             T0 = cell2table({'T0', sheets{row+r,c}
{1,2}},'VariableNames',{'Param' 'Val'});
             Tend = cell2table({'Tend', sheets{row+r,c}
{end,2}},'VariableNames',{'Param' 'Val'});
             %Add to table
             Fitdat{n+1,c} = [Fitdat{n
+1,c};Mass(c-1,2:3);NatDNA(c-1,2:3);FinDNA(c-1,2:3);T0;Tend;Dur]; %Mass
 unused

             %get wash times if applicable
             if row+r >3 && ~isempty(sheets{row+r-1,c})&&
 ~isempty(sheets{row+r,c})
              Twash = cell2table({'twash', sheets{row+r,c}{1,2} -
 sheets{row+r-1,c}{end,2}},'VariableNames',{'Param' 'Val'});
              Fitdat{n+1,c} = [Fitdat{n+1,c};Twash];
             end
             %set NaN values
             if iscell(Fitdat{n+1,c}.Val)
                 Fitdat{n+1,c}.Val(cellfun(@ischar,Fitdat{n+1,c}.Val))
 = {nan};
                 Fitdat{n+1,c}.Val = cell2mat(Fitdat{n+1,c}.Val);
             end

         end
    end
end
```

Published with MATLAB® R2019b