48-well plate Zantiks distance analysis

Table of Contents

1.Select your file

```
zantkisdata=Pamelaphenotypezebra20230908T174503(1:5440, 5:end);
dzantkis=table2array(zantkisdata);
intervals=40
```

```
intervals = 40
```

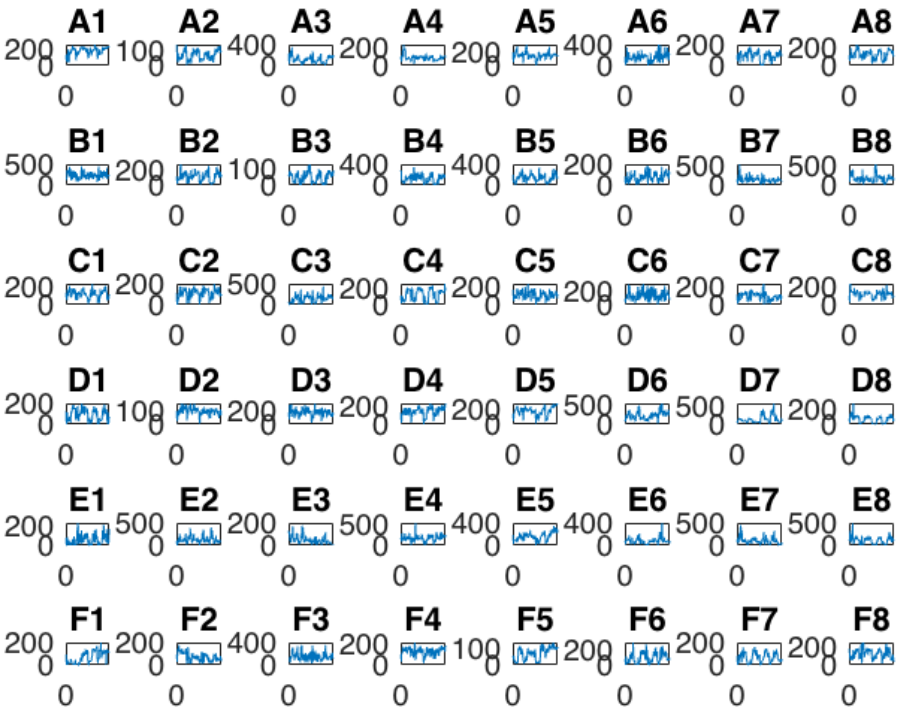
```
seconds = squeeze(sum(reshape(dzantkis',48,intervals,[]),2))';
```

2. 48-well plate for each fish's trajectory

- This part of the code will generate a 48-well plate with each fish's trajectory

```
wellplate=['A1'; 'A2' ;'A3' ;'A4' ;'A5';'A6';'A7';'A8';'B1' ;'B2' ;'B3' ;'B4' ;'B5';'B

for x=1:48
subplot(6,8,x)
plot(seconds(:,x))
title(wellplate(x,:))
end
```



3. Specify each fish's genotype for het by het matings

This part of the code helps you specify each fish's genotype.

- Enter genotypes using numbers 1-48 corresponding to 48-well plate in their corresponding genotype.

```
%This is the part were we tell the code who is whom

% WTfish=seconds(:,[1:12]);
% meanWT=mean(WTfish,2);
% semWT=(std(WTfish')/sqrt(23))';

% %HETfish=seconds(:,[9,10,12,15,16,17,18,20,22,24,25,26,27,28,30,31,32,34,37,38,39,41
% %meanHET=mean(HETfish,2);
% %semHET=(std(HETfish')/sqrt(25))';

% KOfish=seconds(:,[25:39]);
% meanKO=mean(KOfish,2);
% semKO=(std(KOfish')/sqrt(15))';
```

4. Specify genotype for WT vs KO plate setup

- In this part of the code the top half fish are wild type, while the bottom half are knock out.

```
WTfish=seconds(:,[1:24]);
meanWT=mean(WTfish,2);
semWT=(std(WTfish')/sqrt(24))';

KOfish=seconds(:,[25:48]);
meanKO=mean(KOfish,2);
semKO=(std(KOfish')/sqrt(24))';
```

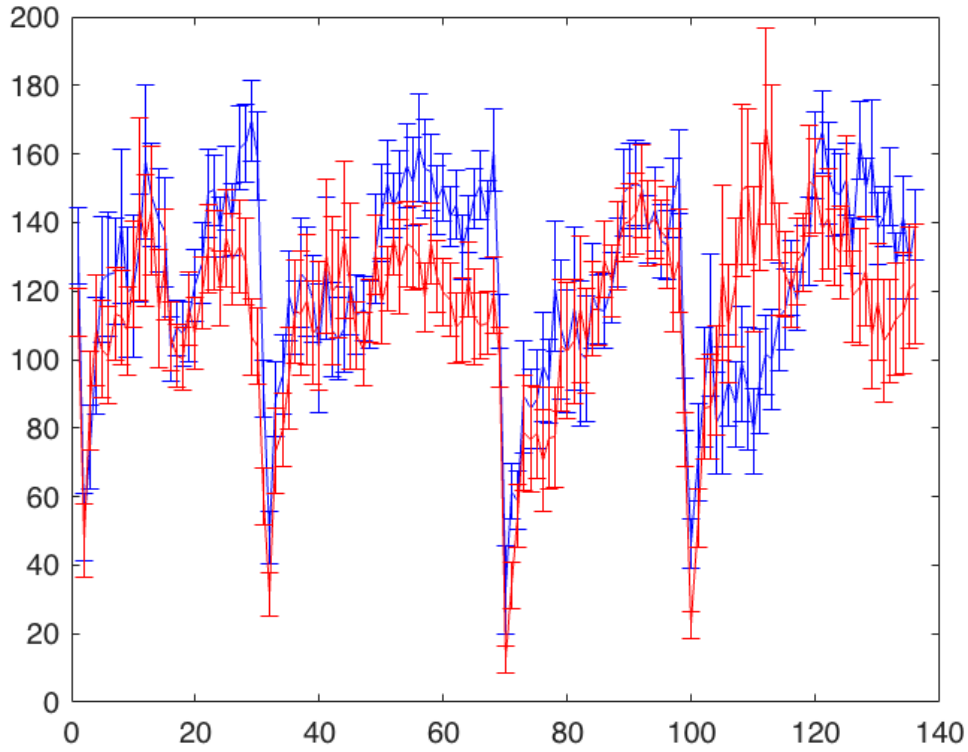
5. Trajectory graph

- This part will graph the trajectory of the fish with respect to time.

```
%code for the graph
figure,errorbar(meanWT,semWT,'blue')

% hold on
% errorbar(meanHET, semHET,['magenta'])

hold on
errorbar(meanKO, semKO,'red')
```



6. Total swim activity

- This part of the code will add up the swim activity of each fish, making a table called "perfishWT" and "perfishKO". There is option for "perfishHET".

```
%All data per fish
perfishWT=table(WTfish);
perfishWT=table2array(perfishWT);
perfishWT=sum(perfishWT)';%one column

perfishKO=table(KOfish);
perfishKO=table2array(perfishKO);
perfishKO=sum(perfishKO)';%one column

%perfishHET=table(HETfish);
%perfishHET=table2array(perfishHET);
%perfishHET=sum(perfishHET)';%one column
```

7. Total swim distance in light and dark phases

- Here the code can separate the total distance swam in dark phases and light phases throughout the protocol.

```
%dark phases of protocol only
% dark_phases_WT=table(WTfish([16:28,46:61,84:99,114:126],:));
% dark_phases_WT=table2array(dark_phases_WT);
% dark_phases_WT=sum(dark_phases_WT)';

% dark_phases_KO=table(KOfish([16:28,46:61,84:99,114:126],:));
% dark_phases_KO=table2array(dark_phases_KO);
% dark_phases_KO=sum(dark_phases_KO)';

%dark_phases_HET=table(HETfish([16:28,46:61,84:99,114:126],:));
%dark_phases_HET=table2array(dark_phases_HET);
%dark_phases_HET=sum(dark_phases_HET)';

%LIGHT phases of protocol only
% light_phases_WT=table(WTfish([1:15,29:45,62:83,100:113],:));
% light_phases_WT=table2array(light_phases_WT);
% light_phases_WT=sum(light_phases_WT)';
%
% light_phases_KO=table(KOfish([1:15,29:45,62:83,100:113],:));
% light_phases_KO=table2array(light_phases_KO);
% light_phases_KO=sum(light_phases_KO)';
```

8. Outlier fish using dark/light behavior

- This part of the code calculates for outlier fish that swim in a different dark/light pattern and removes them from the analysis.

```
%Remove outliers that did not respond to Dark/Light ratio
% ratioWT=dark_phases_WT./light_phases_WT;%this gives a logical
% WT_outliers=isoutlier(ratioWT);
% WT_outliers=~WT_outliers;
```

```
% perfishWT_no_outliers=perfishWT(WT_outliers);
%
% ratioKO=dark_phases_KO./light_phases_KO;%this gives a logical
% KO_outliers=isoutlier(ratioKO);
% KO_outliers=~KO_outliers;
% perfishKO_no_outliers=perfishKO(KO_outliers);

%Remove outliers for total distance
% WT_outliers=isoutlier(perfishWT);
% WT_outliers=~WT_outliers;
% perfishWT_no_outliers_total=perfishWT(WT_outliers);
%
% KO_outliers=isoutlier(perfishKO);
% KO_outliers=~KO_outliers;
% perfishKO_no_outliers_total=perfishKO(KO_outliers);
%
% HET_outliers=isoutlier(perfishHET);
% HET_outliers=~HET_outliers;
% perfishHET_no_outliers_total=perfishHET(HET_outliers);
```