```
(***Need a separate script for cell bodies because cell body ROIs aren'
 t initially chosen based on their activity levels like axon ROIs are***)

(***Input identifying information***)

date = ToString[Evaluate[Input["Input the date of the experiment"]]]

mouse = ToString[Evaluate[Input["Input the mouse identity (e.g. Mouse123)"]]]

sessionNum = Evaluate[Input["Input the session number"]]

(***Import the frame times for the 2P images and calculate the frame rate***)

tpFrameTimes =
  Drop[Drop[(Import[StringJoin["S:/Imaging/Garrett/FMB208_2PRig/", date, "/",
       mouse, "/Session", ToString[sessionNum], "/", date, "_", mouse, "_",
       "Session", ToString[sessionNum], "_2PFrameTimes.txt"], "List"]), 16], -1];

tpFrameRate = Round[Length[tpFrameTimes] / (Last[tpFrameTimes] - First[tpFrameTimes])];

(***For each ROI picked for the session, upload the extracted dF/F0 time series***)

rois = Import[StringJoin["S:/Imaging/Garrett/FMB208_2PRig/", date, "/", mouse,
    "/Session", ToString[sessionNum], "/", "LocomotionData/", date, "_", mouse,
    "_", "Session", ToString[sessionNum], "_locModROIs", ".txt"], "List"];

Table[Evaluate@ToExpression[StringJoin["dFFts", ToString[n]]] =
    ToExpression /@ Import[StringJoin["S:/Imaging/Garrett/FMB208_2PRig/", date,
       "/", mouse, "/Session", ToString[sessionNum], "/dFoverF0TimeSeries/",
       date, "_", mouse, "_Session", ToString[sessionNum], "_",
       "dFoverF0ts_ROI", ToString[n], ".txt"], "List"];, {n, rois}];

(***Import the walk bout start and end times***)

walkBouts =
  ToExpression[Import[StringJoin["S:/Imaging/Garrett/FMB208_2PRig/", date, "/", mouse,
     "/Session", ToString[sessionNum], "/", "LocomotionData/", date, "_", mouse, "_",
     "Session", ToString[sessionNum], "_isolatedWalkBouts.txt"], "List"]];

locOnsetTimes = walkBouts[[All, 1]];

locOffsetTimes = walkBouts[[All, 2]];

(***Get interpolation functions for the dF/F0 time series for each ROI***)

Table[Evaluate@ToExpression[StringJoin["dFFInterpFunc", ToString[n]]] =
    Interpolation[ToExpression[StringJoin["dFFts", ToString[n]]]];, {n, rois}];

(*********************************Z-
 SCORING FROM LOC ONSET************************************)

(***For each ROI, calculate the z-scored dF/F0 for around each onset,
using the 10 to 15 seconds before onset as the baseline***)
```

```
In[•]:= Table[Evaluate@ToExpression[StringJoin["baselinePeriOnsetMeans", ToString[n]]] =
      Table[Mean[Table[(ToExpression[StringJoin["dFFInterpFunc", ToString[n]]])[k],
         {k, locOnsetTimes[[x]] - 15, locOnsetTimes[[x]] - 10, 1 / tpFrameRate}]],
       {x, 1, Length[locOnsetTimes]}];, {n, rois}];

In[•]:= Table[Evaluate@ToExpression[StringJoin["baselinePeriOnsetSDs", ToString[n]]] = Table[
      StandardDeviation[Table[(ToExpression[StringJoin["dFFInterpFunc", ToString[n]]])[
         k], {k, locOnsetTimes[[x]] - 15, locOnsetTimes[[x]] - 10, 1 / tpFrameRate}]],
       {x, 1, Length[locOnsetTimes]}];, {n, rois}];

In[•]:= Table[Evaluate@ToExpression[StringJoin["periOnsetZscoredDFFs", ToString[n]]] =
      Table[Table[(((ToExpression[StringJoin["dFFInterpFunc", ToString[n]]])[i]) -
           (ToExpression[StringJoin["baselinePeriOnsetMeans", ToString[n]]])[[x]]) /
         ((ToExpression[StringJoin["baselinePeriOnsetSDs", ToString[n]]])[[x]]),
        {i, locOnsetTimes[[x]] - 15, locOnsetTimes[[x]] + 6, 1 / tpFrameRate}],
       {x, 1, Length[locOnsetTimes]}];, {n, rois}];

In[•]:= (***For each ROI, make a time series of z-scored dF/F0 and mean z-scored dF/F0,***)

In[•]:= periOnsetTimeVals = Table[n, {n, -15, 6, 1 / tpFrameRate}];

In[•]:= Table[Evaluate@ToExpression[StringJoin["meanZscoredPeriOnsetDFFsTS", ToString[n]]] =
      Mean[Table[Partition[Riffle[periOnsetTimeVals,
           (ToExpression[StringJoin["periOnsetZscoredDFFs", ToString[n]]])[[m]]], 2],
         {m, 1, Length[(ToExpression[StringJoin["periOnsetZscoredDFFs",
              ToString[n]]])]}]];, {n, rois}];

In[•]:= (*********************************Z-
     SCORING FROM LOC OFFSET*********************************)

In[•]:= (***For each ROI, calculate the z-scored dF/F0 for around each offset,
    using the 10 to 15 seconds before offset as the baseline***)

In[•]:= Table[Evaluate@ToExpression[StringJoin["baselinePeriOffsetMeans", ToString[n]]] =
      Table[Mean[Table[(ToExpression[StringJoin["dFFInterpFunc", ToString[n]]])[k],
         {k, locOffsetTimes[[x]] + 10, locOffsetTimes[[x]] + 15, 1 / tpFrameRate}]],
       {x, 1, Length[locOffsetTimes]}];, {n, rois}];

In[•]:= Table[Evaluate@ToExpression[StringJoin["baselinePeriOffsetSDs", ToString[n]]] = Table[
      StandardDeviation[Table[(ToExpression[StringJoin["dFFInterpFunc", ToString[n]]])[
         k], {k, locOffsetTimes[[x]] + 10, locOffsetTimes[[x]] + 15, 1 / tpFrameRate}]],
       {x, 1, Length[locOffsetTimes]}];, {n, rois}];

In[•]:= Table[Evaluate@ToExpression[StringJoin["periOffsetZscoredDFFs", ToString[n]]] =
      Table[Table[(((ToExpression[StringJoin["dFFInterpFunc", ToString[n]]])[i]) -
           (ToExpression[StringJoin["baselinePeriOffsetMeans", ToString[n]]])[[x]]) /
         ((ToExpression[StringJoin["baselinePeriOffsetSDs", ToString[n]]])[[x]]),
        {i, locOffsetTimes[[x]] - 6, locOffsetTimes[[x]] + 15, 1 / tpFrameRate}],
       {x, 1, Length[locOffsetTimes]}];, {n, rois}];

In[•]:= (***For each ROI, make a time series of z-scored dF/F0 and mean z-scored dF/F0,***)
```

```
In[ ]:= periOffsetTimeVals = Table[n, {n, -6, 15, 1 / tpFrameRate}];

In[ ]:= Table[Evaluate@ToExpression[StringJoin["meanZscoredPeriOffsetDFFsTS", ToString[n]]] =
     Mean[Table[Partition[Riffle[periOffsetTimeVals,
          (ToExpression[StringJoin["periOffsetZscoredDFFs", ToString[n]]])[[m]]], 2],
        {m, 1, Length[(ToExpression[StringJoin["periOffsetZscoredDFFs",
             ToString[n]]])]}]];, {n, rois}];

   (***Visualize the z-scored dF/F of each ROI aligned to locomotion onset and offset***)

In[ ]:= Manipulate[{ListLinePlot[
    ToExpression[StringJoin["meanZscoredPeriOnsetDFFsTS", ToString[n]]], PlotRange → All],
   ListLinePlot[ToExpression[StringJoin["meanZscoredPeriOffsetDFFsTS", ToString[n]]],
    PlotRange → All]}, {n, rois}]

In[ ]:= (***Export results***)

In[ ]:= Table[Export[StringJoin["S:/Imaging/Garrett/FMB208_2PRig/", date, "/", mouse, "/Session",
     ToString[sessionNum], "/", "LocomotionData/", date, "_", mouse, "_", "Session",
     ToString[sessionNum], "_PeriOnsetZDFF_PreAndPostBaseline_ROI", ToString[n], ".txt"],
    ToExpression[StringJoin["meanZscoredPeriOnsetDFFsTS", ToString[n]]]], {n, rois}];

In[ ]:= Table[Export[StringJoin["S:/Imaging/Garrett/FMB208_2PRig/", date, "/", mouse, "/Session",
     ToString[sessionNum], "/", "LocomotionData/", date, "_", mouse, "_", "Session",
     ToString[sessionNum], "_PeriOffsetZDFF_PreAndPostBaseline_ROI", ToString[n], ".txt"],
    ToExpression[StringJoin["meanZscoredPeriOffsetDFFsTS", ToString[n]]]], {n, rois}];
```