```
npeph = ConstantArray[0, nevents]; tft0 = ConstantArray[0, nevents];
Do[
  fit = FindFit[Transpose[{ttt, v01[[i]]}],
    0.5157198640558519 + a * 0.3796189845474239 * Sin[x - b], {a, b}, x];
  npeph[[i]] = a /. fit;
  tft0[[i]] = b /. fit;
  , {i, nevents}];
```

FindFit::sszero:
The step size in the search has become less than the tolerance prescribed by the PrecisionGoal option, but the gradient is larger than the tolerance specified by the AccuracyGoal option. There is a possibility that the method has stalled at a point that is not a local minimum. >>

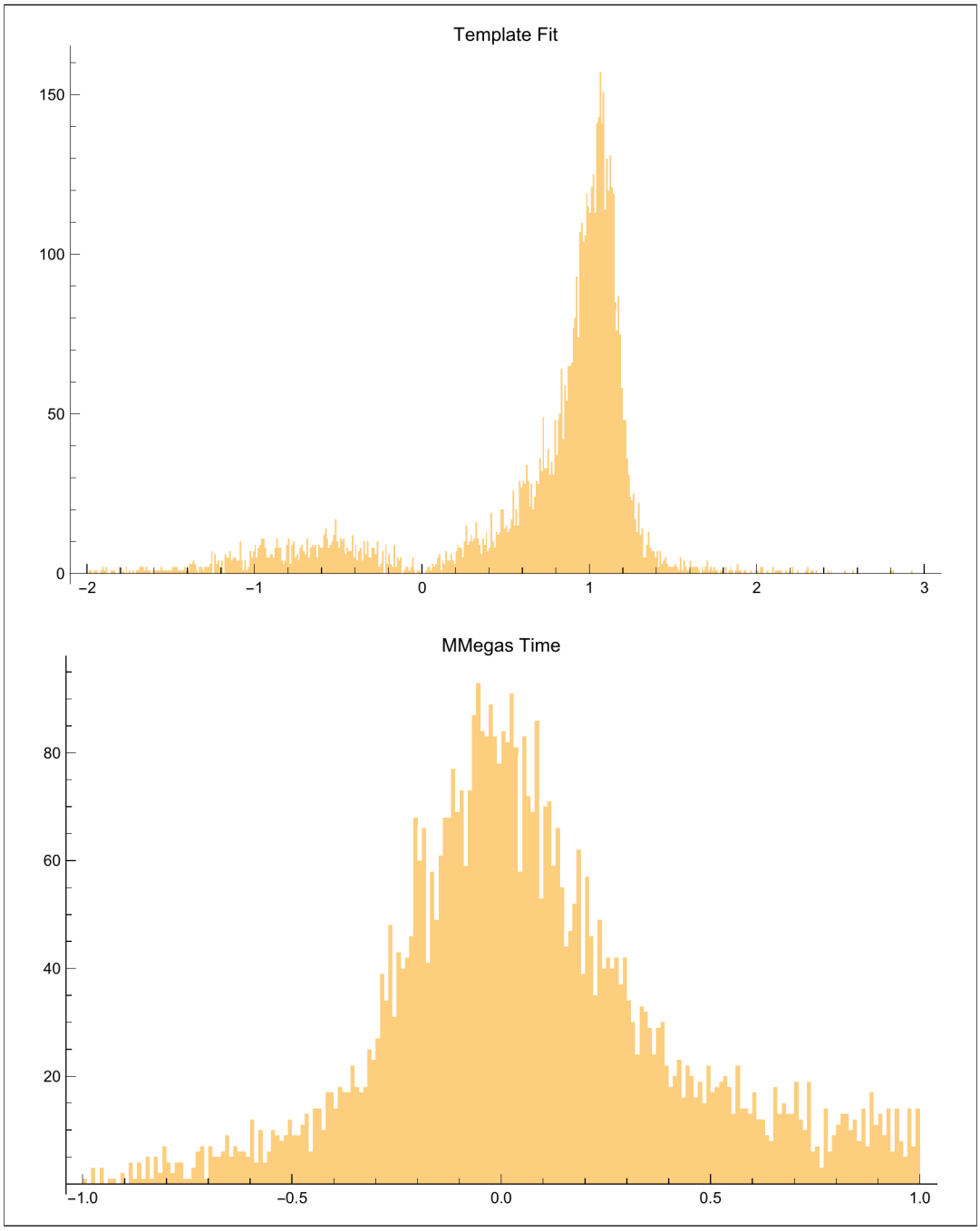
FindFit::sszero:
The step size in the search has become less than the tolerance prescribed by the PrecisionGoal option, but the gradient is larger than the tolerance specified by the AccuracyGoal option. There is a possibility that the method has stalled at a point that is not a local minimum. >>

FindFit::sszero:
The step size in the search has become less than the tolerance prescribed by the PrecisionGoal option, but the gradient is larger than the tolerance specified by the AccuracyGoal option. There is a possibility that the method has stalled at a point that is not a local minimum. >>

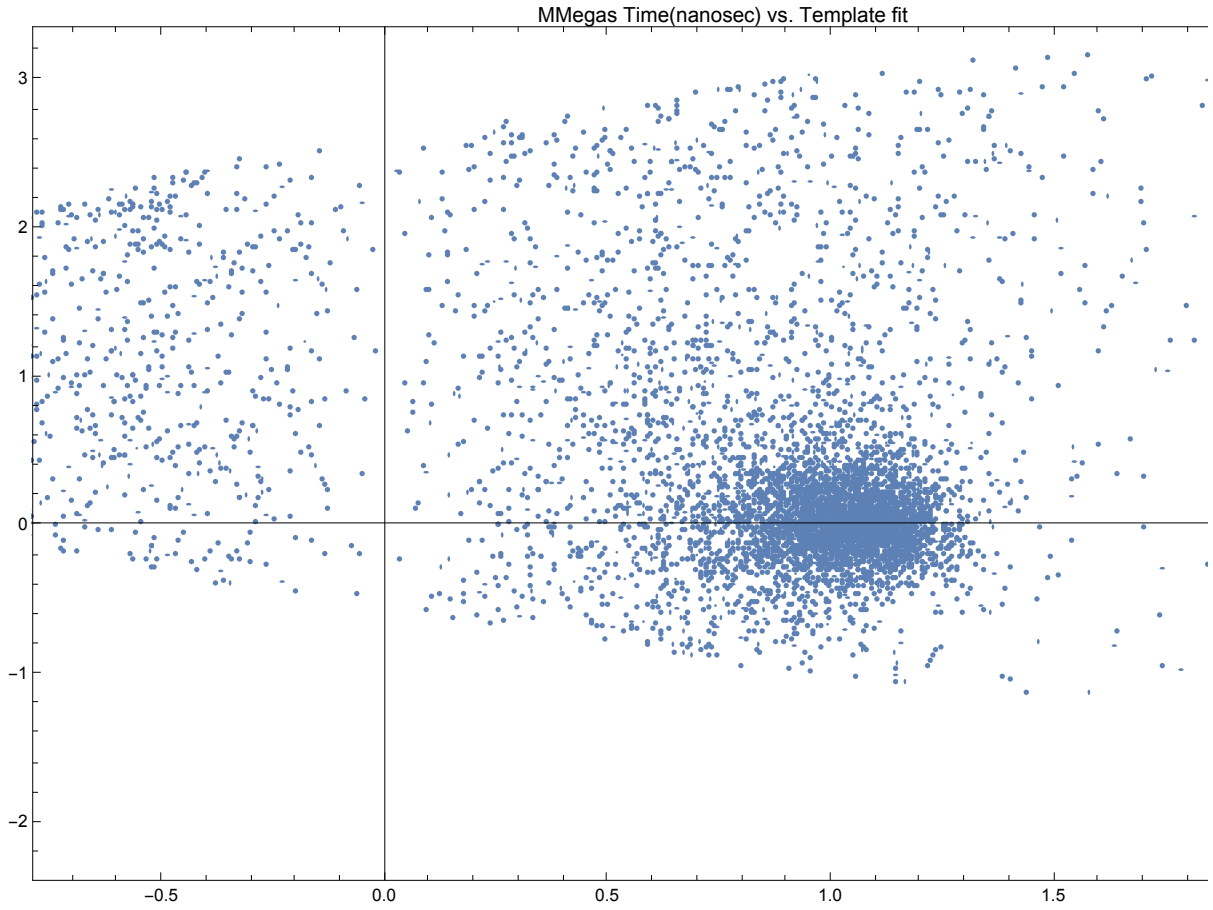
General::stop: Further output of FindFit::sszero will be suppressed during this calculation. >>

It is nice that we get such a narrow peak for the 1's - but surprising that we don' t find a lot of 2' s (ie see Poisson Distribution above). Actually makes a lot of sense. This data set I renormalized to have max time bin at 1. So plot below just tells how well we can resolv (1,2,3) from 0. Will repeat with un-normalized data.

```
GraphicsColumn[{Histogram[npeph, {-2, 3, .01}, PlotLabel → "Template Fit"],
  Histogram[tft0, {-1, 1, .01}, PlotLabel → "MMegas Time"]},
 Frame → True, ImageSize → Large]
```

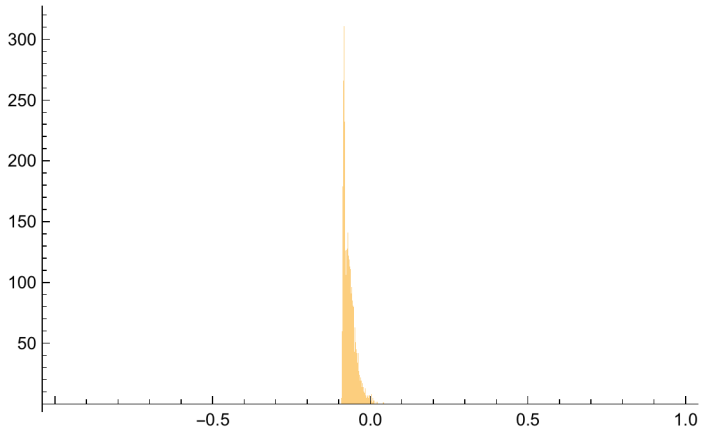


```
ListPlot[Transpose[{npeph, tft0}], Frame → True,
 PlotLabel → "MMegas Time(nanosec) vs. Template fit"]
```




```
npeph1 = ConstantArray[0, nevents];
Do[
  fit = FindFit[Transpose[{ttt, v01[[i]] * peak[[i]]}],
    0.5157198640558519 + a * 0.3796189845474239 * Sin[x], {a}, x];
  npeph1[[i]] = a /. fit;
  , {i, nevents}];
```

```
Histogram[npeph1, {-1, 1, .001}]
```



```
ListPlot[Transpose[{npeph, npeph1}], PlotRange → Full]
```