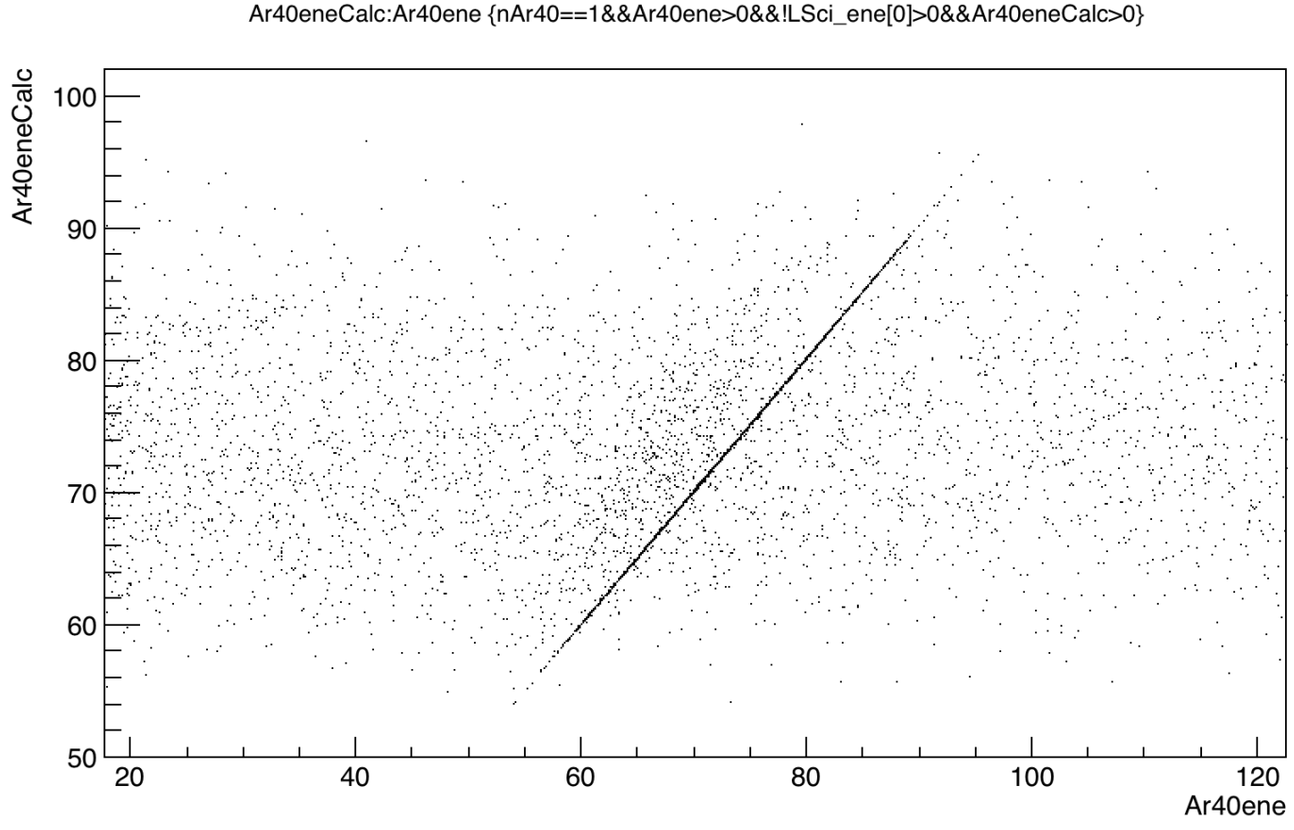
Calculation of Ar40 recoil energy, considering neutron energy loss.

Agreement between simulated and calculated recoil energies was almost total using the previous g4rooter code.

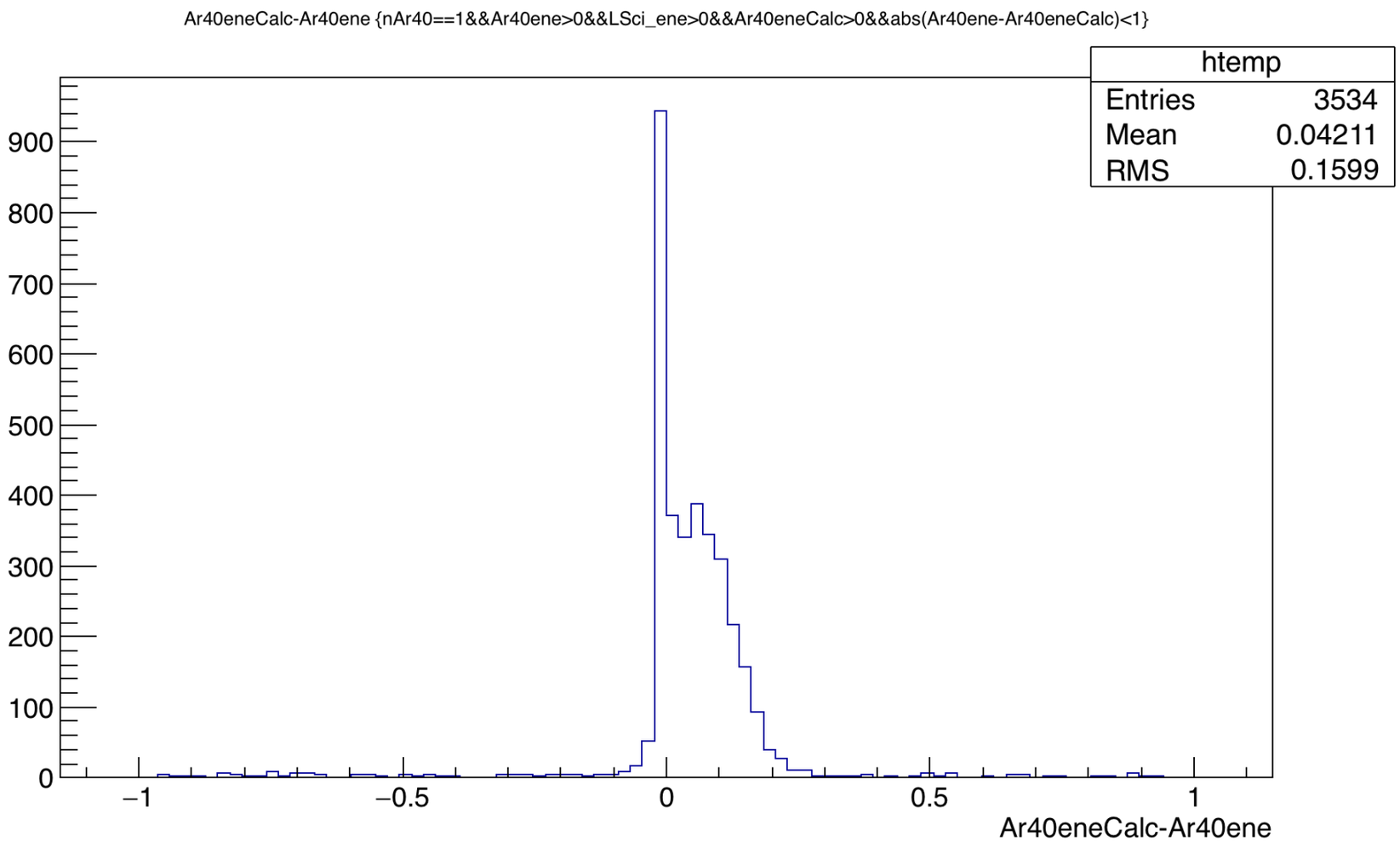
Recoil energy was calculated using:
Ar40eneCalc = 1000*(Ar40_mome * Ar40_mome)/(2*m_Ar);

where the momentum Ar40_mome was calculated by:

n_momentum = sqrt(2*ene0*m_n);
which considers no energy loss from the neutron.



Making a subtraction and pointing to a keV energy range, there is a little discrepancy coming out, visible on the right of the “zero” peak, due to scattering detected by LSci’s 1 to 8, central peak is mostly due to scatterings on “0” LSci.



…so the idea was that maybe taking the neutron energy loss into account we could get rid of this discrepancy…

As a first attempt, I derived a fully relativistic formula for the recoil energy, and then made a first order approximation with a Taylor series, to transform it in a feasible code. It was cumbersome and the approximation was worse than before.

Then I tried another approximation:

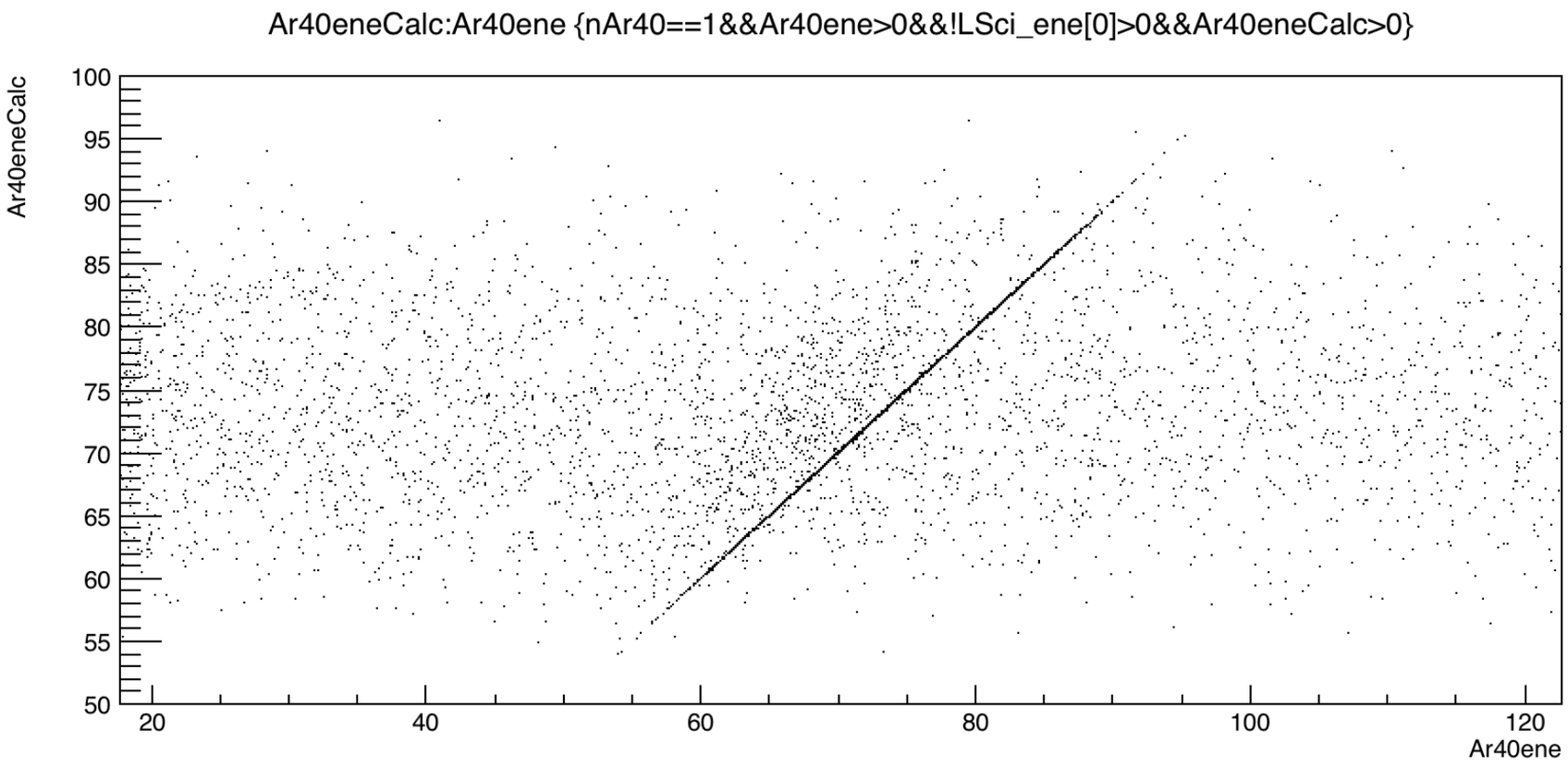
I calculated the momentum of incoming and outcoming neutron with relativistic dispersion, and subtracting the previously calculated recoil energy of Ar40 from the energy of the outcoming neutron, this way:

```
    n_momentum_in = sqrt((ene0+m_n)*(ene0+m_n)-(m_n)*(m_n));//incoming neutron momentum, relativistic calculation
    n_momentum_out = sqrt((ene0+ m_n- Energy_loss_inMeV )*(ene0 + m_n - Energy_loss_inMeV ) -(m_n)*(m_n));
//outcoming neutron momentum, relativistic calculation
```

..then I re-calculated the Ar40 recoil energy, using the second momentum obtained for the neutron, considering the energy loss… :

```
Ar40eneCalc = 1000*(Ar40_mome_final * Ar40_mome_final)/(2*m_Ar);
```

…it looks pretty much the same up to now…



Then, subtracting and comparing with the previous subtraction, we realize that taking into account the energy loss now there is a perfect agreement between simulated and calculated Ar40 recoil energy.