Standard centrality definitions are based on refmult cuts (Ntracks $|\eta|<0.5$)

“higher moments” analyses make a centrality selection then proceed to calculate the moments of the distributions of the E-by-E multiplicity of specific particles or multiplicity differences.
Note: these moments are not “central moments” so K,S moments are sensitive to the mean values of the multiplicities.

Autocorrelation between the centrality selection and the K,S moments values?

*e.g.* Nihar & Lizhu define “refmult2” $(0.5<|\eta|<1.0)$ for centrality and then calculate the moments in the range $|\eta|<0.5$.

Refmult/BBC/ZDC correlations explored in previous bulkcorr presentation

Problems with ZDC vs BBC and BBC vs refmult cuts:
1. No ZDC at low root-s, BBC full of background
2. Tails in the variable distributions for most central events
3. If refmult involved, still susceptible to autocorrelations
So we need something else
- eta gap, e.g. refmult2 or FTPC... (not lots of data for FTPC – too slow)
OR
- same acceptance as TPC but different detector & particles...

BEMC: not very sensitive to \( h^\pm \), but large signals for gamma’s from \( \pi^0 \)...
\( N_\gamma \approx N_h \)
Wide coverage...
Tower info exists for all root-s values...

\[ \sum E_{\text{bemc}} \]

\[ \sum E_{\text{bemc}(E>0)} \]
Problem though: BEMC tower calibration in BES data is not very good....

done by fill in p+p and AuAu200, otherwise ~once/week.... (J. Stevens)

Tens of GeV shifts in 7.7 & 11.5
~300 GeV shift in 19.6
(no huge jumps in 39 and 62.4)

Not only this, but the sumE pedestal is really wide (tens of GeV)

I can see tower pedestals moving by ~10-20 MeV/ch in different runs
*across the entire BEMC*
At the moment, I am shifting the pedestals for the big shifts and leaving the gains alone...

This is not enough....

to do this right, we really need updated BEMC tower peds/gains....

```c
//------------------------------
Float_t aread::GetBEMCsumE(){
    float sume = emcsume;
    if (kDataSet==19){                 // 7.7 GeV Run-10
        if (runid<=11124063){ sume += -1.43; } else
            if (runid<=11135075){ sume +=  35.36; } else
                { sume +=  -25.98; }
    } else if (kDataSet==20){          // 11.5 GeV Run-10
        if (runid==11156007){ sume += -100.2;  }
        if (runid==11156008){ sume += -100.2;  }
    } else if (kDataSet==23){          // 19.6 GeV Run-11
        if (runid<=12114114){ sume +=  15.90; } else
            if (iday <= 115  ){ sume += -265.40; } else
                if (iday >  115  ){ sume +=   8.95; }
    }
    return sume;
}
```
Pushing on anyway for 62.4 (no big pedestal jumps seen)
Run Hisoshi’s Glauber codes  *(thanks Hiroshi!)*

500k fast glauber events: Npart vs Ncoll
then “scan” in NbdFitMaker to find \((npp,k,x)\) parameters

unlike for refmult, now \(npp\) is just a pure fitting parameter

---

**RefMultTpc**
- \(npp=1.4\)
- \(k=2.0\)
- \(x=0.12\)

**BemcSumE**
- \(npp=2.44\)
- \(k=2.0\)
- \(x=0.12\)

---

This is just the very first try – probably can be improved.....
### Resulting Cuts (by integrating the sim spectrum)

<table>
<thead>
<tr>
<th>Centrality (%)</th>
<th>RefMultTpc</th>
<th>BemcSumE (GeV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>75-80</td>
<td>23</td>
<td>17</td>
</tr>
<tr>
<td>70-75</td>
<td>30</td>
<td>23</td>
</tr>
<tr>
<td>65-70</td>
<td>38</td>
<td>33</td>
</tr>
<tr>
<td>60-65</td>
<td>47</td>
<td>45</td>
</tr>
<tr>
<td>55-60</td>
<td>57</td>
<td>59</td>
</tr>
<tr>
<td>50-55</td>
<td>70</td>
<td>77</td>
</tr>
<tr>
<td>45-50</td>
<td>84</td>
<td>99</td>
</tr>
<tr>
<td>40-45</td>
<td>101</td>
<td>125</td>
</tr>
<tr>
<td>35-40</td>
<td>119</td>
<td>157</td>
</tr>
<tr>
<td>30-35</td>
<td>140</td>
<td>193</td>
</tr>
<tr>
<td>25-30</td>
<td>164</td>
<td>235</td>
</tr>
<tr>
<td>20-25</td>
<td>192</td>
<td>285</td>
</tr>
<tr>
<td>15-20</td>
<td>224</td>
<td>343</td>
</tr>
<tr>
<td>10-15</td>
<td>260</td>
<td>413</td>
</tr>
<tr>
<td>5-10</td>
<td>301</td>
<td>493</td>
</tr>
<tr>
<td>0-5</td>
<td>351</td>
<td>591</td>
</tr>
</tbody>
</table>

*again, this is just the first stab – do not use these cuts yourself!*
Refmullt following BemcSumE centrality selection

Method seems promising, but we really need improved tower calibrations to proceed.