# A brief description of Jamie's work on $F_L$

R. Debbe BNL

1

The Parton Distribution Functions from a proton are extracted from the MRST2002 parametrization for an array of 49x49 x and $Q^2$ bins.

Cross sections for e+p

$$d^2\sigma/dxdQ^2 = 4\,\pi\,\alpha^2/(Q^4x)\ \ ((1-y)\,F_2 + xy^2\,F_1\,)$$

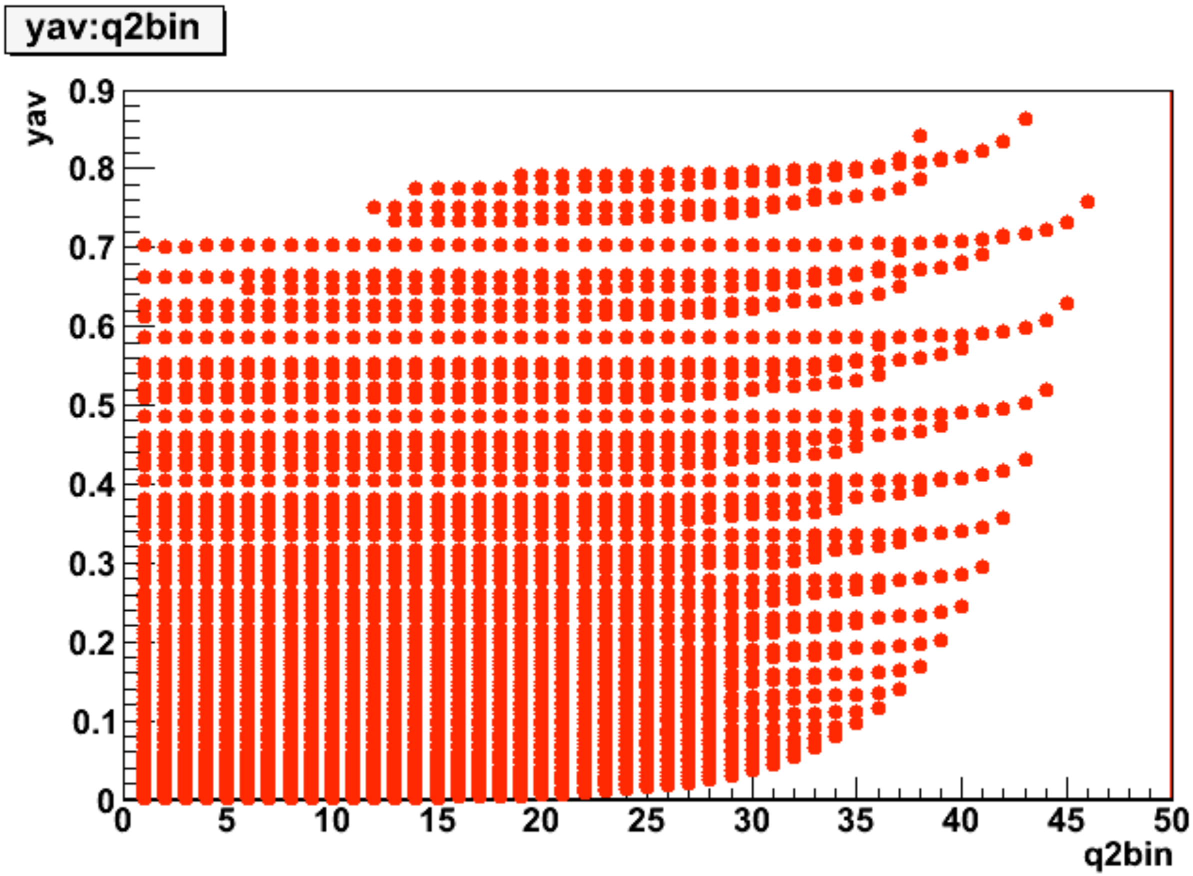
are calculated for 6 (9) energy settings:

Electron: 4 10 20 4 10 20 **4 10 20** GeV

Proton: 50 50 50 100 100 100 **250 250 250** GeV

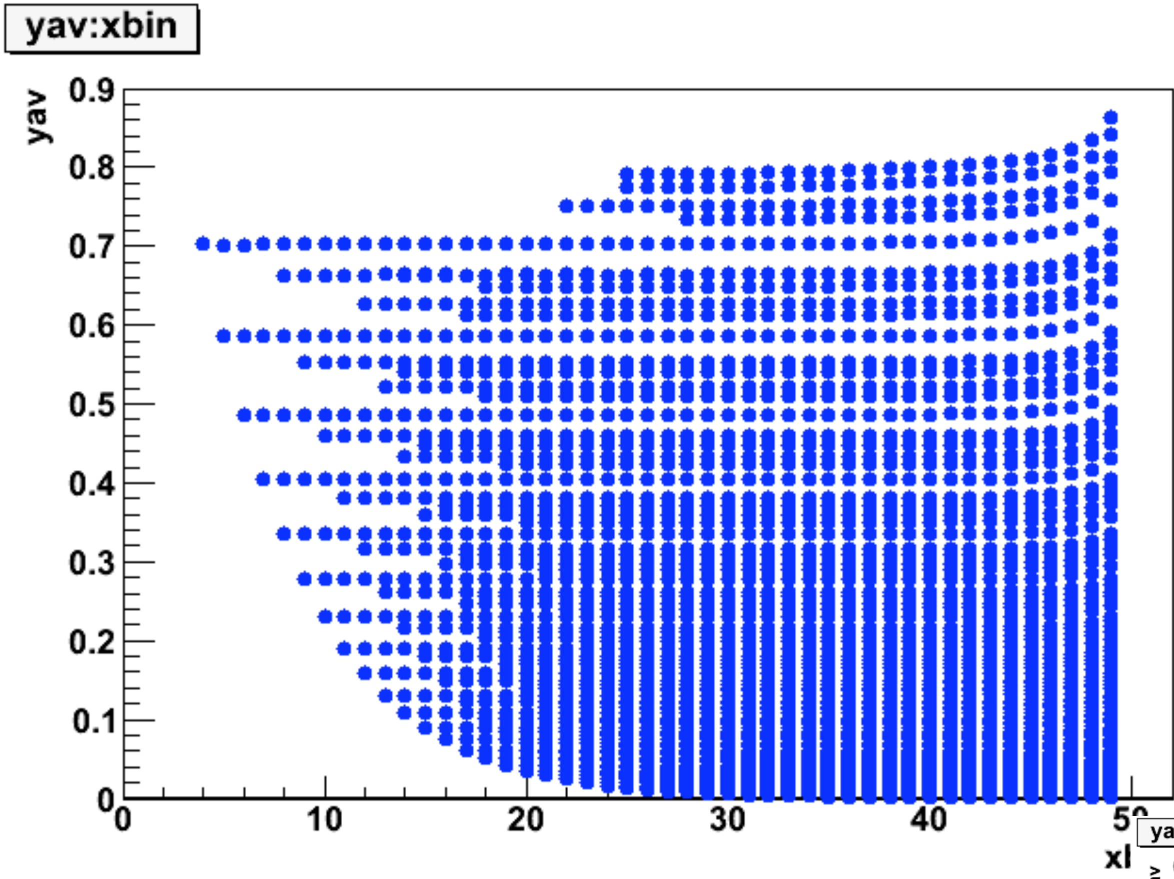
At fixed x and $Q^2$ bins one get at least 3 (4) values of $y=Q^2/xs$ and their corresponding "reduced cross section"

2

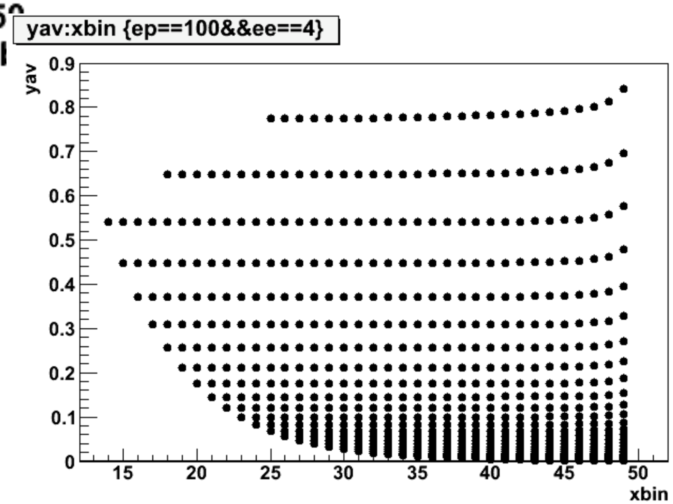


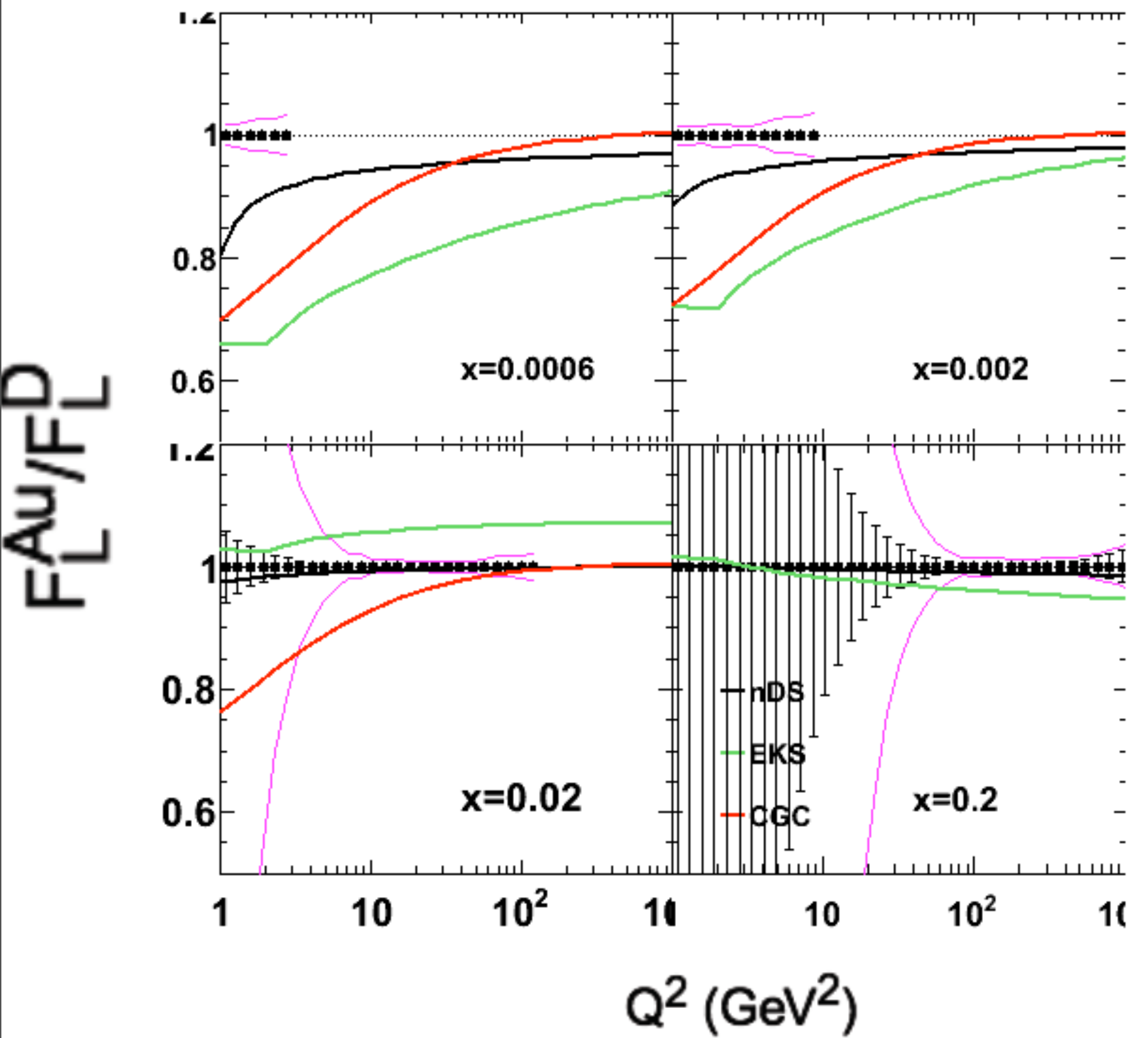

The 49 logarithmic $Q^2$ bins extend from 0. to 3000 $GeV^2$

3




The x bins are also logarithmical

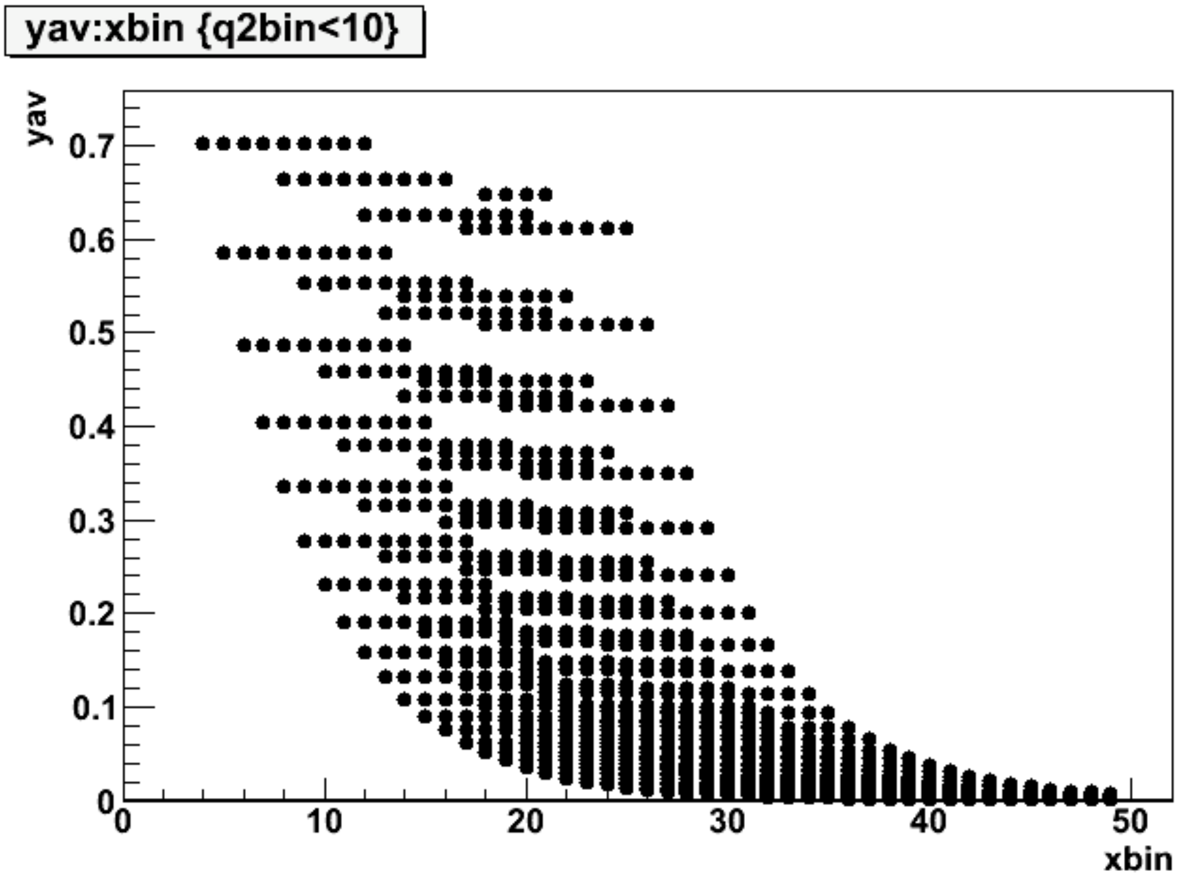





The question I left unanswered in my previous presentation has to do with the **growth of error bars at low values of $Q^2$ specially at high x.**

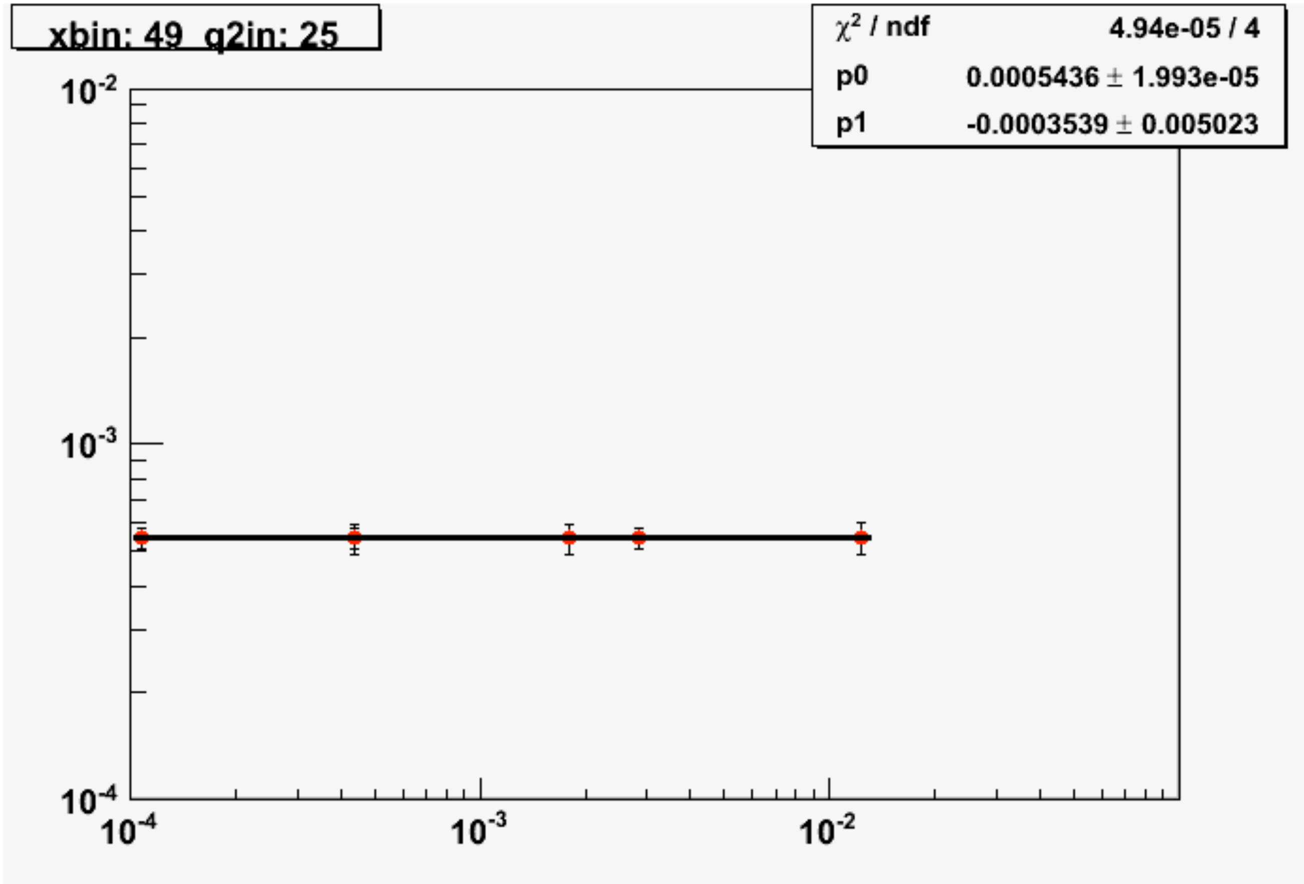
**Magenta** curve shows 1% syst. error added in quadrature to statistical errors of the "reduced" cross section before a second fit is performed.

5




The “phase space” of the e+p collision narrows into regions of small y as one enters x values of x.

6



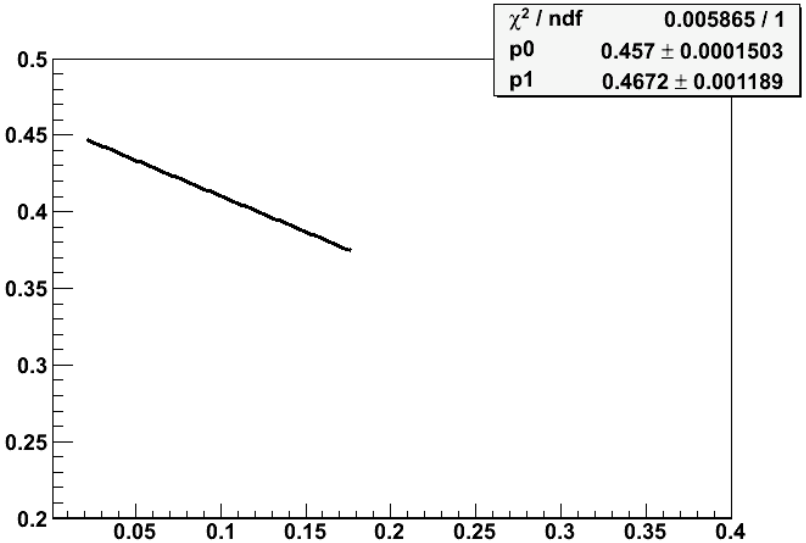

This is one example of a setting where the y values are small, the reduced cross section is basically constant. The fit to a straight line doesn't have enough "lever arm" and allocates big relative error to the slope.

7

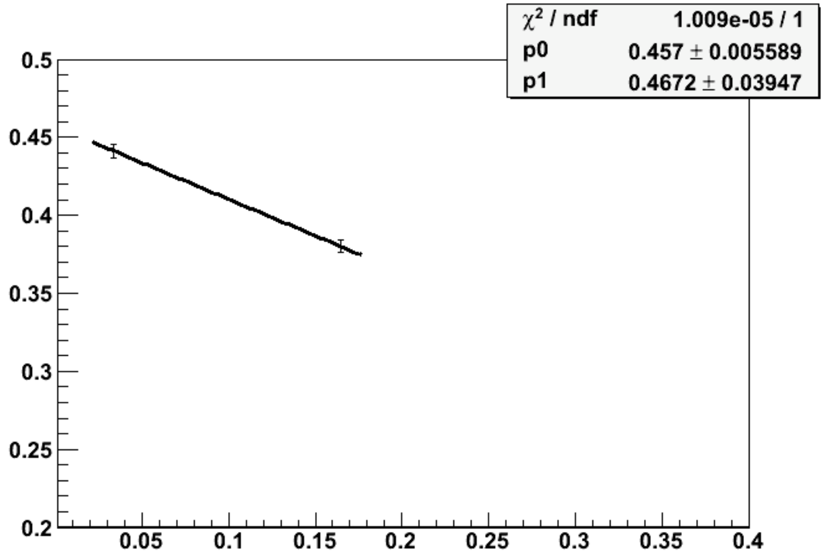
The statistical errors of the calculation are small (now we concentrate on low-x and low $Q^2$). When we add 1% systematic error in quadrature, the **result is a ~1% overall error**.

Another round of fits of “reduced” cross section versus $y^2/Y_+$ produces errors on slope ($F_L$) that have significant magnitude event at the lowest x and $Q^2$.

$\sigma_r$

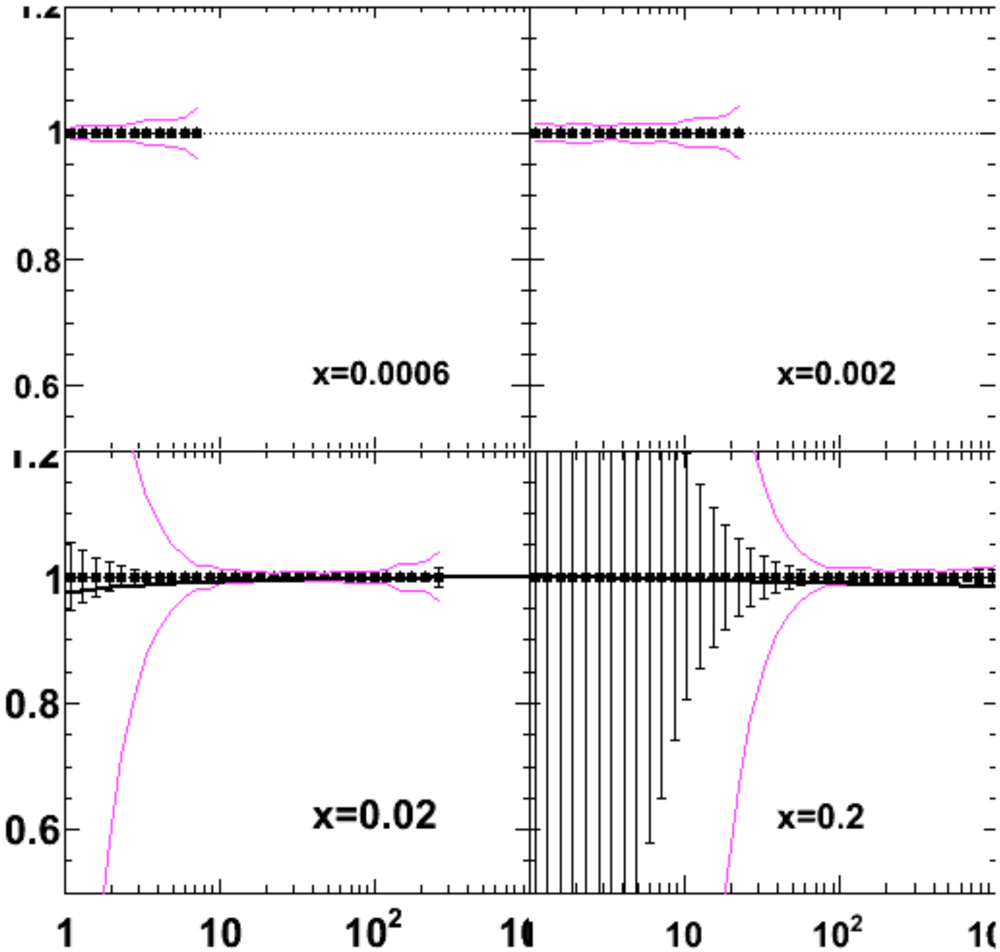



$y^2/Y_+$




$y^2/Y_+$

8




This is a first pass on a file that includes proton energies as high as 250 GeV. I recycled the macro, the ratio doesn't make sense but the figure shows that we get to extend coverage to higher $Q^2$ and it also looks like we extende the region where overall error is ~1-2%

9

At this moment the program is very slow because it calculates averages of 7 quantities in each ($x,Q^2$) “square” using Monte-Carlo integration (VEGAS) with **10000 ! samplings**.

In principle, the integration should converge with a few samplings. I will investigate the effect of doing integrations with smaller number of samplings 10-100.