Blind tests with D. Higinbotham & new idea

Xuefei Yan

Nov 7 2017

Key points

- Finished blind tests with D. Higinbotham
- Key points after/inspired-by discussion with D.H.
  - GE fitting studies are (in general) in 2 categories: for extracting Rp or for as-much-as-possible/full description of GE(Q^2) spectrum
  - No consensus for: should one include high Q^2 data (how high?) if one just wants Rp, but not full GE(Q^2) functional form?
  - Noise and binning of data will affect fitting as well (could make high order polynomial fit, etc. problematic: getting noise, not real R)
  - These can be studied: D.H.'s and mine may compensate each other
- A new fitting lib is partially finished, and (hopefully) can run easily for testing these & model dependence of R extraction from PRad data

Table of fit results

Fit type	# of parameters	χ^2	χ^2 per dof	R (fm)
Dipole with free norm	2	25.04	0.61	0.8800±0.0055
Monopole with free norm	2	24.13	0.59	0.8899±0.0056
Gaussian with free norm	2	29.41	0.72	0.8696±0.0053
Polynomial up to Q^4 with free norm	3	24.46	0.60	0.8840±0.0082
Polynomial ratio with free norm	3	24.05	0.59	0.8877±0.0095
Inverse polynomial with free norm	4	24.13	0.59	0.8899±0.0056

Note: inverse polynomial is an extension of monopole

- Monopole: $G_E = p_0 \left(1 + \frac{Q^2}{p_1}\right)^{-1}$
- Inverse polynomial: $G_E = p_0(1 + p_1 \times Q^2 + p_2 \times Q^4 + p_3 \times Q^6)^{-1}$
- Adding p2 and p3 changes the fitting result beyond 4^{th} digit (not seen in presented precision)

- Q^2 range (41 points): 3.1×10^{-4} – 0.07 GeV^2
- D.H. used binning & size of uncertainty from PRad proposal
- Plots are attached at the end

Summary

- Fitting same data, different functional forms can give different result
  - As one do not know what the REAL charge distribution is, one needs to do multiple try-outs
- D.H. also shared his fitting programs (fitting and quality checking), which gave ~same results
  - Can use these for cross check
- A new fitting lib for PRad was made, so all the fits above (and more) can be run by one click (with < 50 lines in main C++ code)
  - Features are being built to enable useful tests (next page)

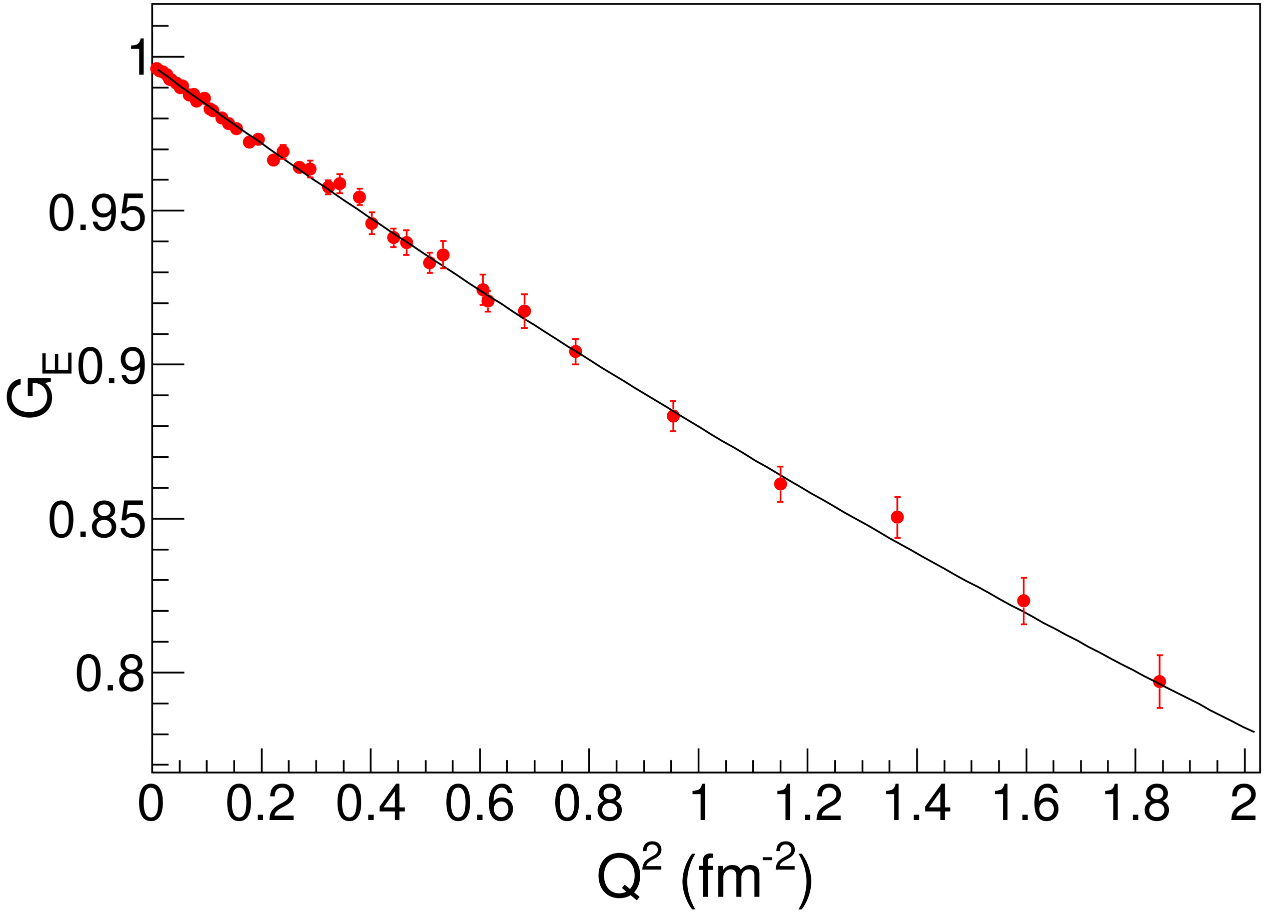
New lib (C++ class) for model-dependence tests of R extraction

- New idea
  - One should test with N model inputs, with M types of noise (Gaussian, non-Gaussian) ~size of total PRad uncertainties, and fit with K functional forms, to see whether (and how much) Rp from PRad data has model dependence
- Generator (i)
  - Generate GE value at Q^2 with certain models
  - It can also read from txt files (data, or other people's table)
- Noise adder (j)
  - Add noise to generated GE table
- Fitter (k)
  - Fit GE vs. Q^2 table with certain functional forms
- Then one can loop through G(i), N(j) and F(k), and compare input R with R from fit
- Able to add more types of G, N and F

Plots attached

- Fits of D.H. fake data 1
- Fits of D.H. fake data 2

Full dipole with free-normalization functional form (2 parameter) fit: 41 points

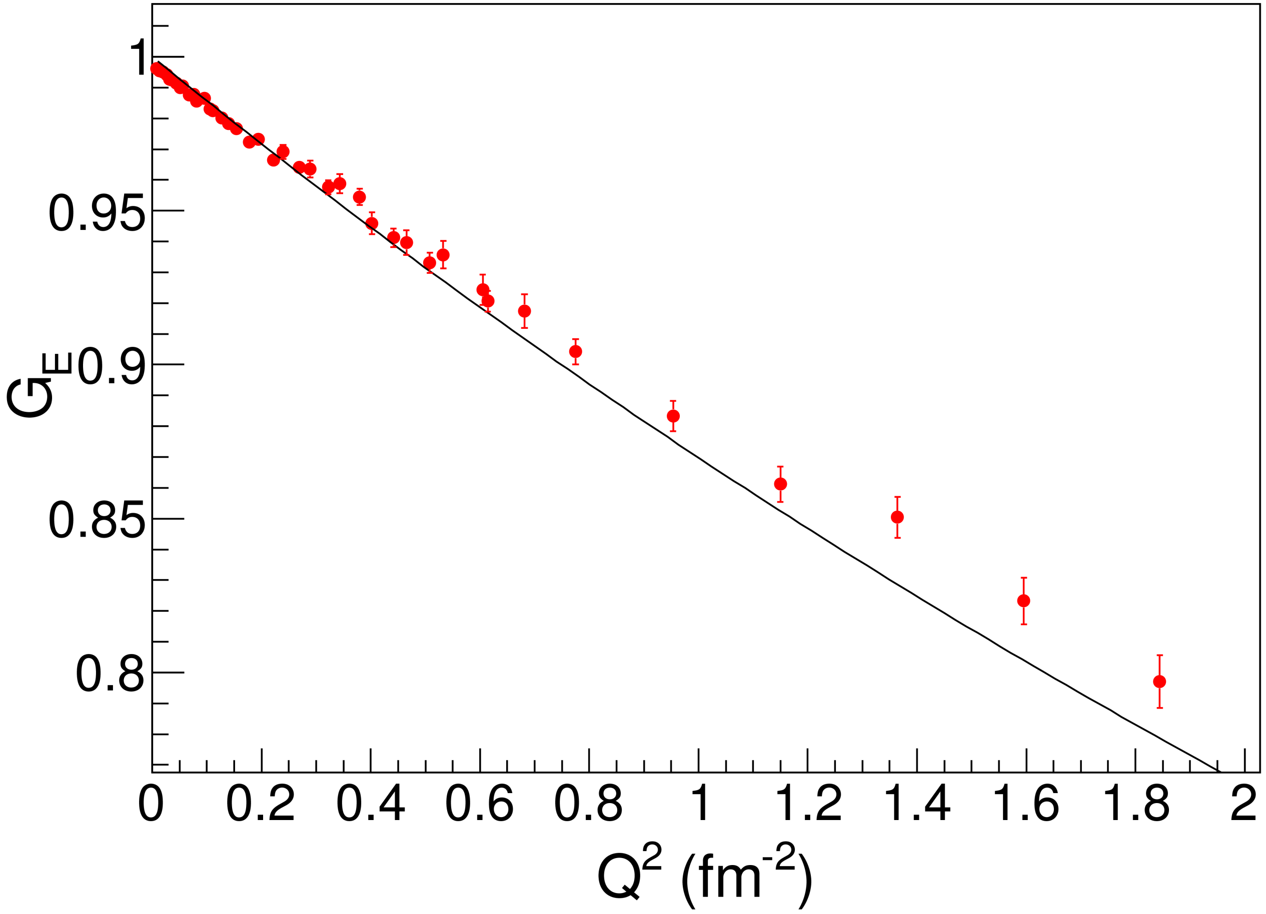


- χ^2=25.04
- χ^2 per dof= 0.61
- R=0.8800$\pm$0.0055 fm

- Q^2 range: $3.1 \times 10^{-4} - 0.07$ GeV^2
- D.H. used binning & size of uncertainty from PRad proposal

Functional form: $G_E = p_0\left(1+\frac{Q^2}{p_1}\right)^{-2}, R = \left(-6\frac{d(G_E/p_0)}{dQ^2}|_{Q^2=0}\right)^{1/2}$

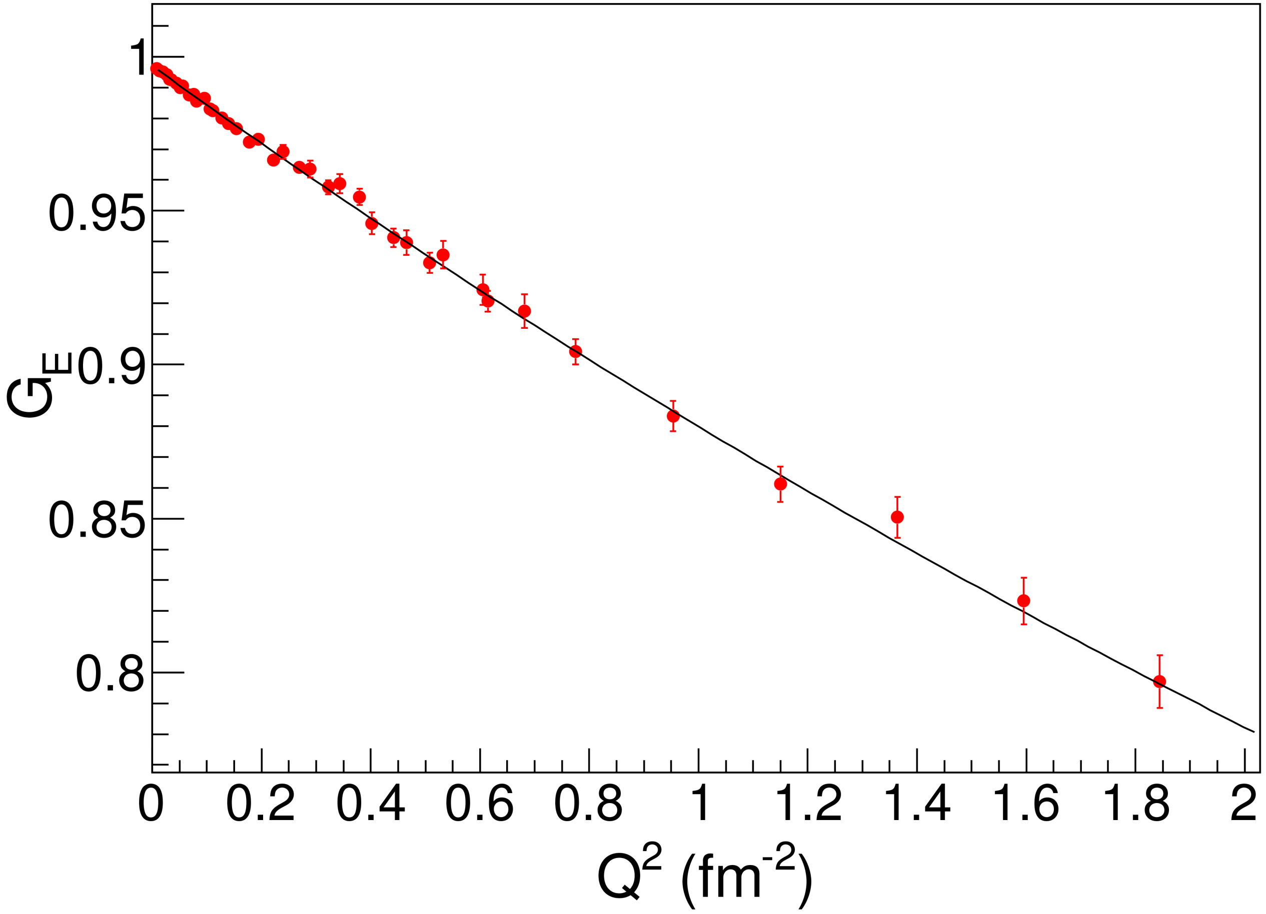
Full dipole functional form (1 parameter) fit: 41 points [BAD FIT]



- χ^2=438.29
- χ^2 per dof= 10.69
- R=0.9311±0.0046 fm

Functional form: $G_E = \left(1 + \frac{Q^2}{p1}\right)^{-2}$, $R = \left(-6\frac{dG_E}{dQ^2}|_{Q^2=0}\right)^{1/2}$

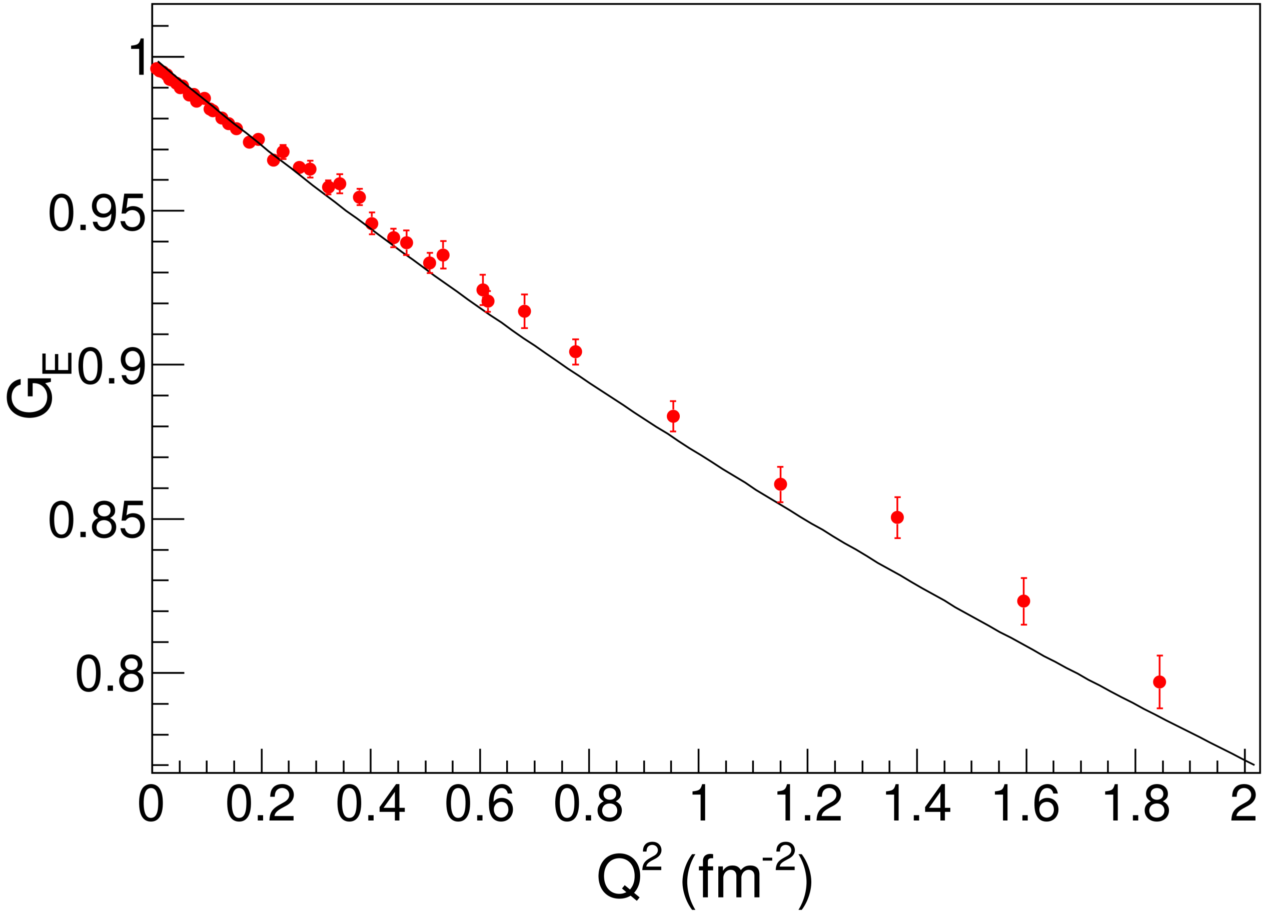
Full monopole with free-normalization functional form (2 parameter) fit: 41 points



- χ^2=24.13
- χ^2 per dof= 0.59
- R=0.8899±0.0056 fm

Functional form: $G_E = p_0\left(1+\frac{Q^2}{p_1}\right)^{-1}, R = \left(-6\frac{d(G_E/p_0)}{dQ^2}|_{Q^2=0}\right)^{1/2}$

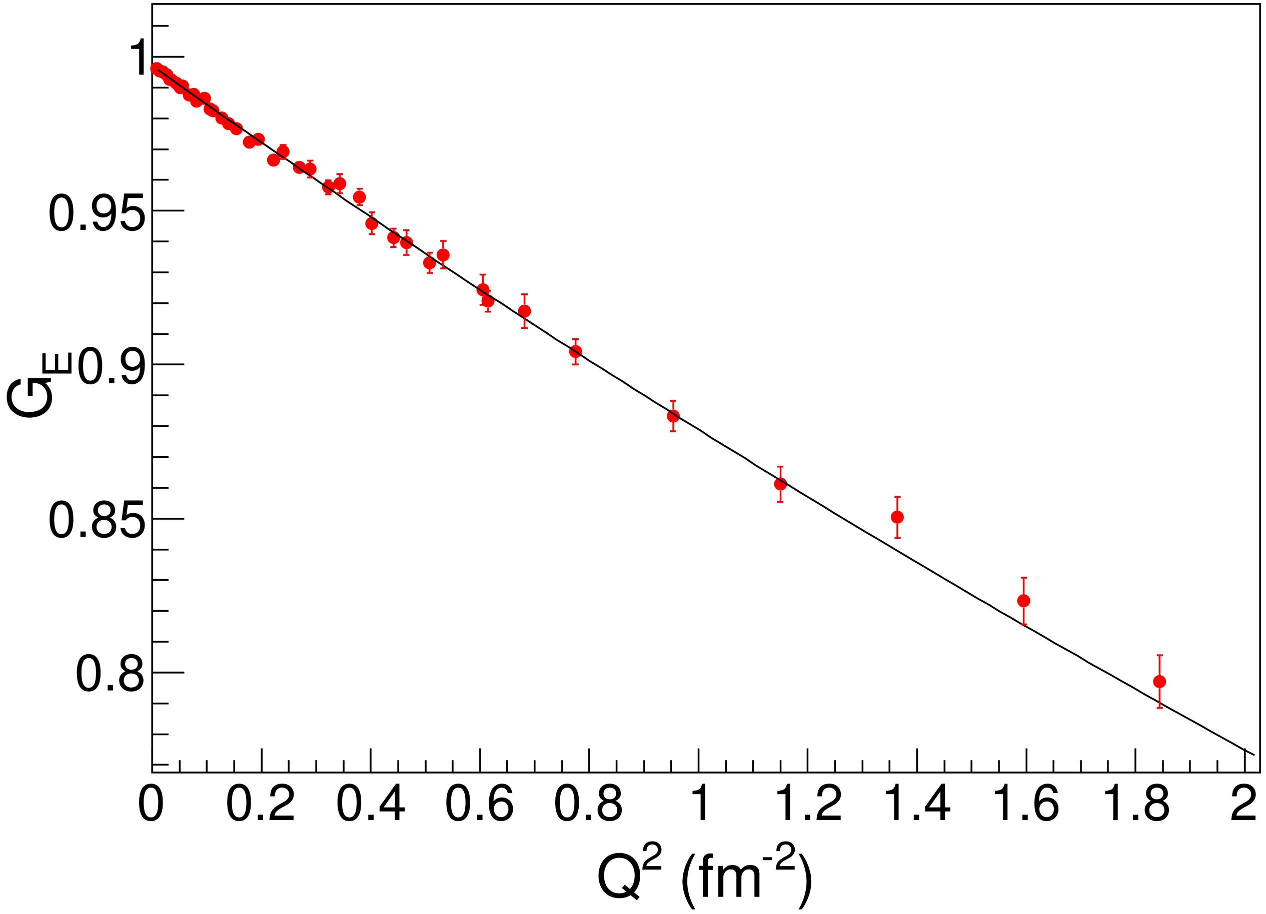
Full monopole functional form (1 parameter) fit: 41 points [BAD FIT]



- χ^2=407.87
- χ^2 per dof= 9.94
- R=0.9420±0.0048 fm

Functional form: $G_E = \left(1 + \frac{Q^2}{p1}\right)^{-1}$, $R = \left(-6\frac{dG_E}{dQ^2}|_{Q^2=0}\right)^{1/2}$

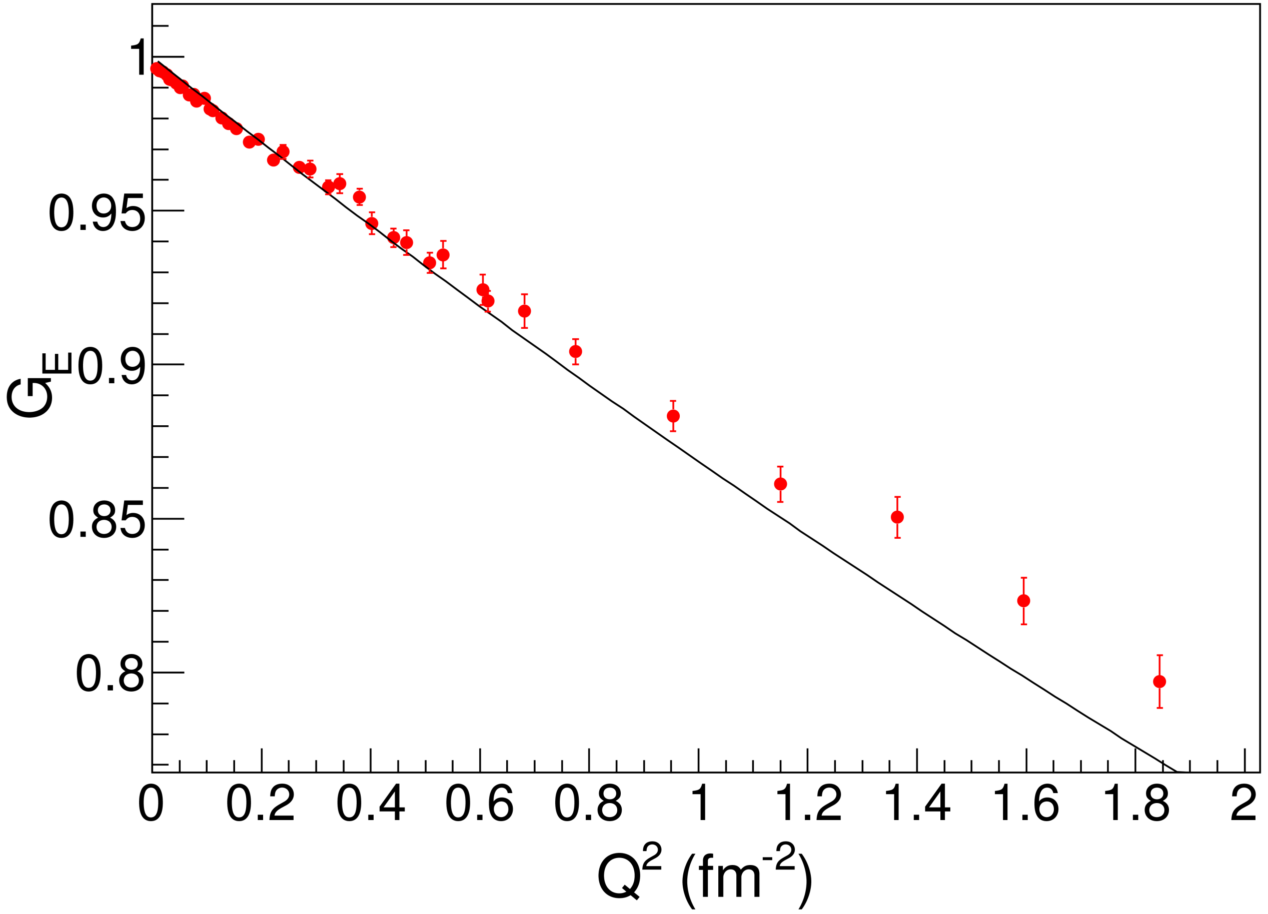
Full Gaussian with free-normalization functional form (2 parameter) fit: 41 points



- χ^2=29.41
- χ^2 per dof= 0.72
- R=0.8696±0.0053 fm

Functional form: $G_E = p_0 Exp(-\frac{Q^2}{p_1}), R = \left(-6\frac{d(G_E/p_0)}{dQ^2}|_{Q^2=0}\right)^{1/2}$

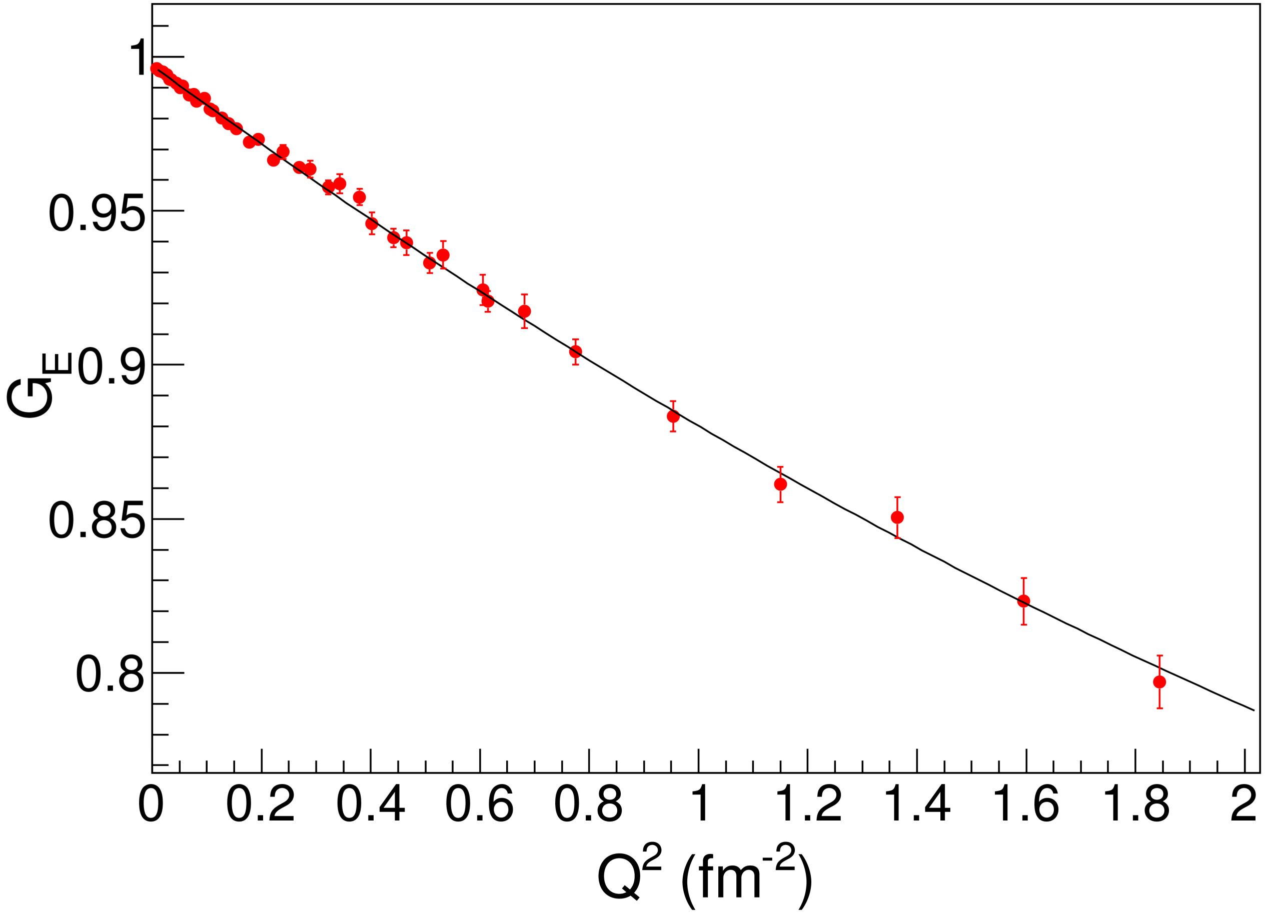
Full Gaussian functional form (1 parameter) fit: 41 points [BAD FIT]



- χ^2=475.64
- χ^2 per dof= 11.6
- R=0.9196$\pm$0.0045 fm

Functional form: $G_E = Exp(-\frac{Q^2}{p1})$, $R = \left(-6\frac{dG_E}{dQ^2}|_{Q^2=0}\right)^{1/2}$

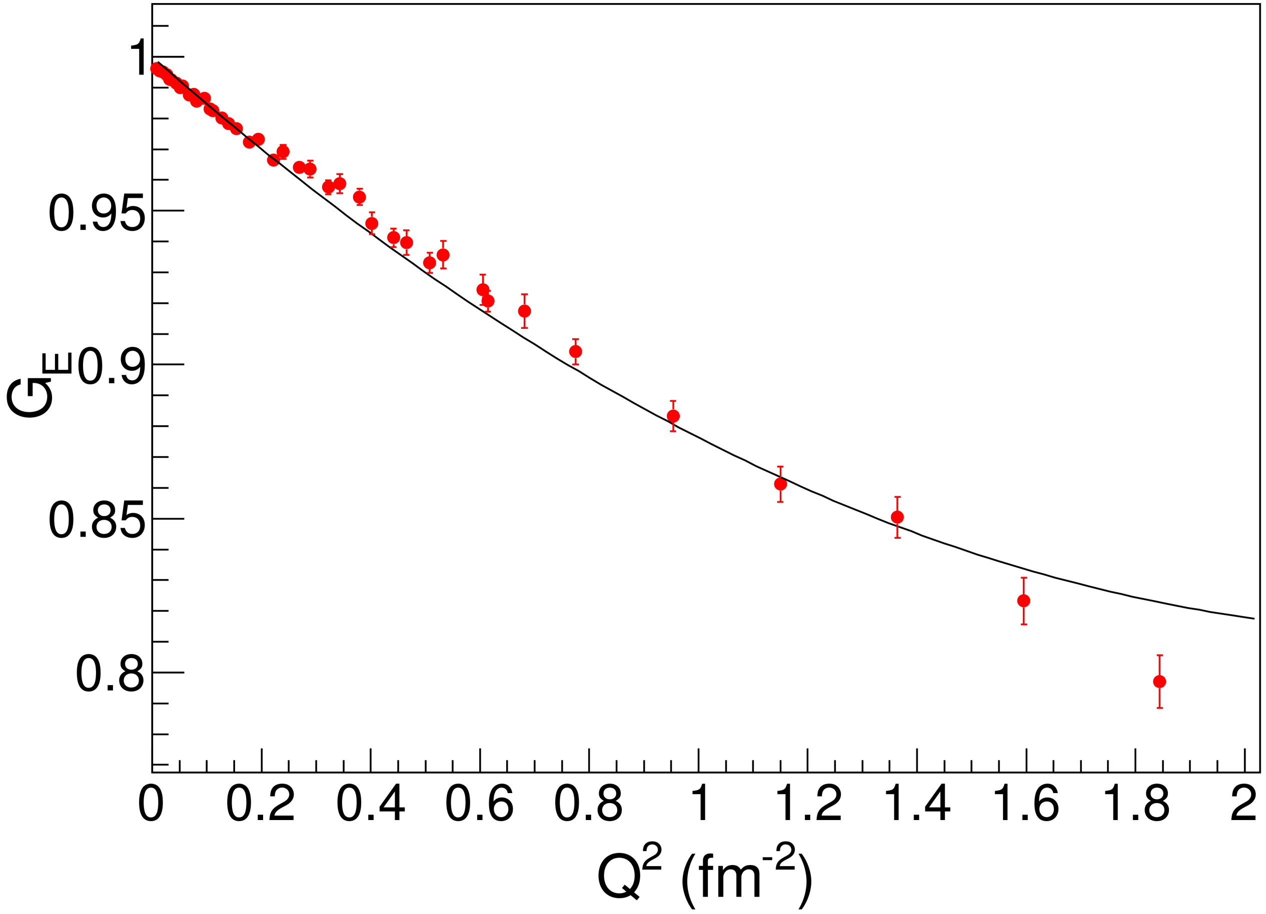
Multiple parameter polynomial fit (with free normalization parameter): 41 points



- χ^2=24.46
- χ^2 per dof= 0.60
- R=0.8840±0.0082 fm

Functional form: $G_E = p_0[1 + \sum_{n\geq 1} \frac{(-1)^n}{(2n+1)!} \langle r^{2n} \rangle Q^{2n}]$, up to Q^4

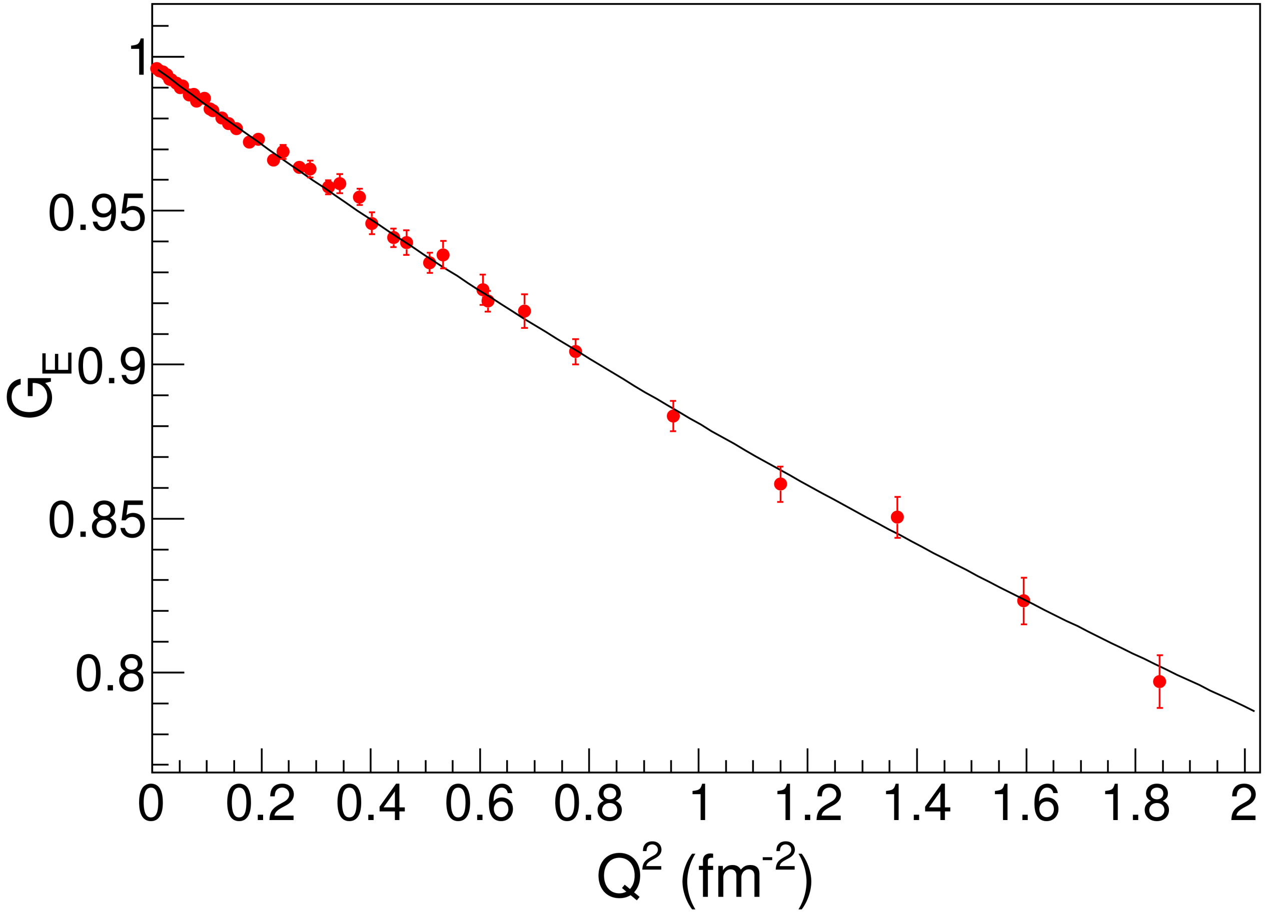
Multiple parameter polynomial fit: 41 points [BAD FIT]



- χ^2=369.17
- χ^2 per dof= 9.00
- R=0.9686±0.0063 fm

Functional form: $G_E = 1 + \sum_{n\geq1} \frac{(-1)^n}{(2n+1)!} \langle r^{2n} \rangle Q^{2n}$, up to Q^4

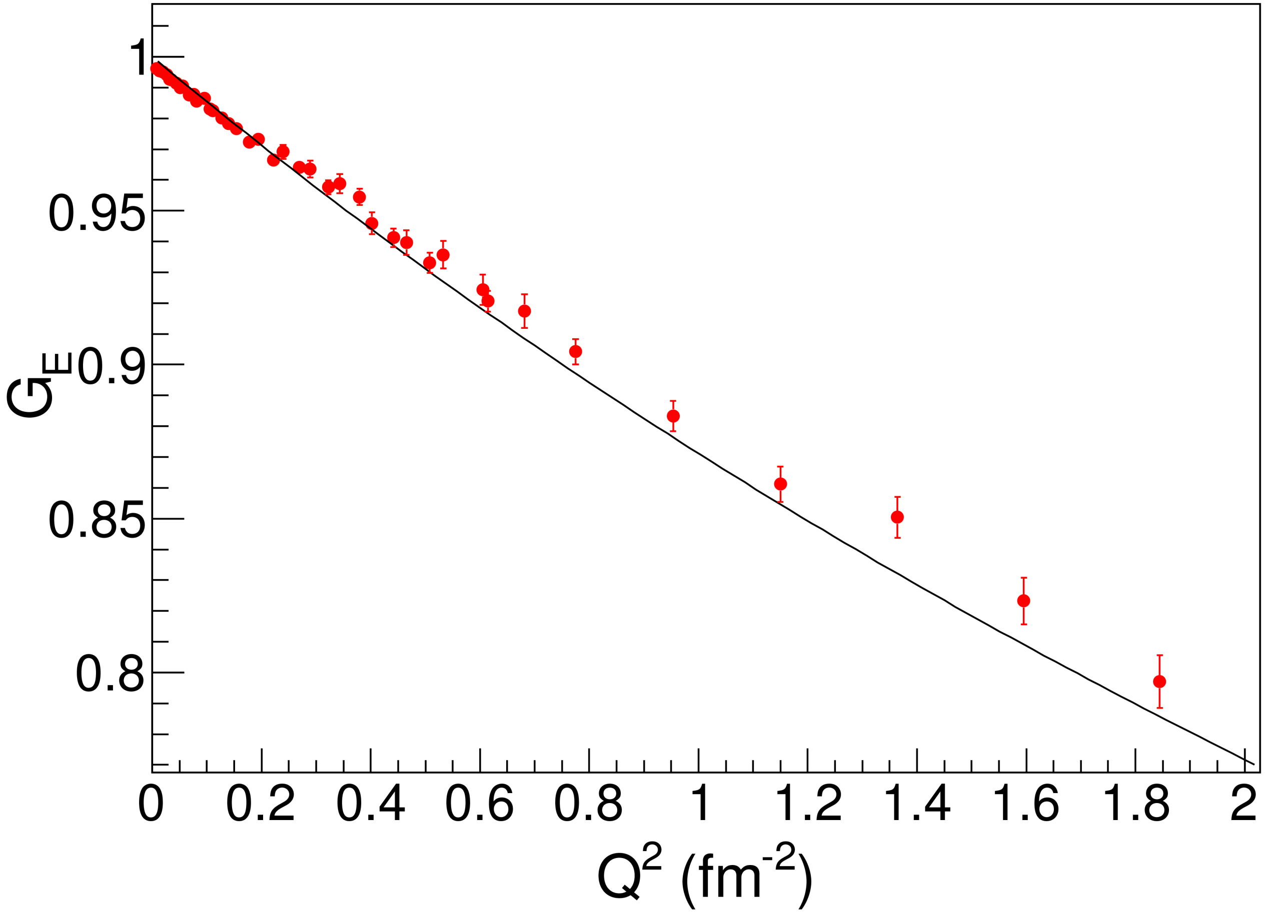
Inverse polynomial with free normalization (4 parameter) fit: 41 points



- χ^2=24.13
- χ^2 per dof= 0.59
- R=0.8899$\pm$0.0056 fm

Functional form: $G_E = p_0(1 + p_1 \times Q^2 + p_2 \times Q^4 + p_3 \times Q^6)^{-1}$, $R = \left(-6\frac{d(G_E/p_0)}{dQ^2}|_{Q^2=0}\right)^{1/2}$

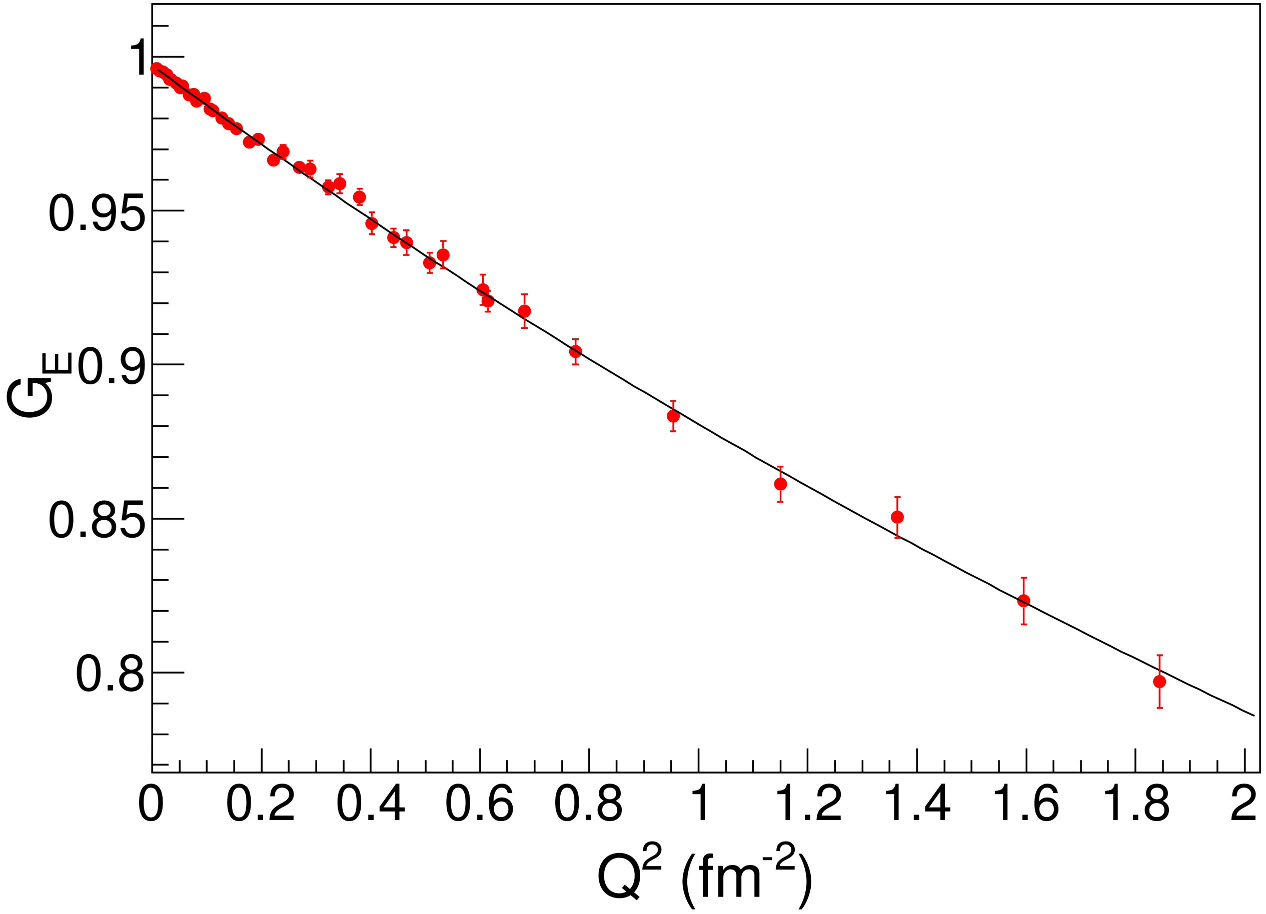
Inverse polynomial (3 parameter) fit: 41 points [BAD FIT]



- χ^2= 407.87
- χ^2 per dof= 9.94
- R=0.9420±0.0048 fm

Functional form: $G_E = (1 + p_1 \times Q^2 + p_2 \times Q^4 + p_3 \times Q^6)^{-1}$, $R = \left(-6 \frac{dG_E}{dQ^2}|_{Q^2=0}\right)^{1/2}$

Polynomial ratio (3 parameter) fit: 41 points

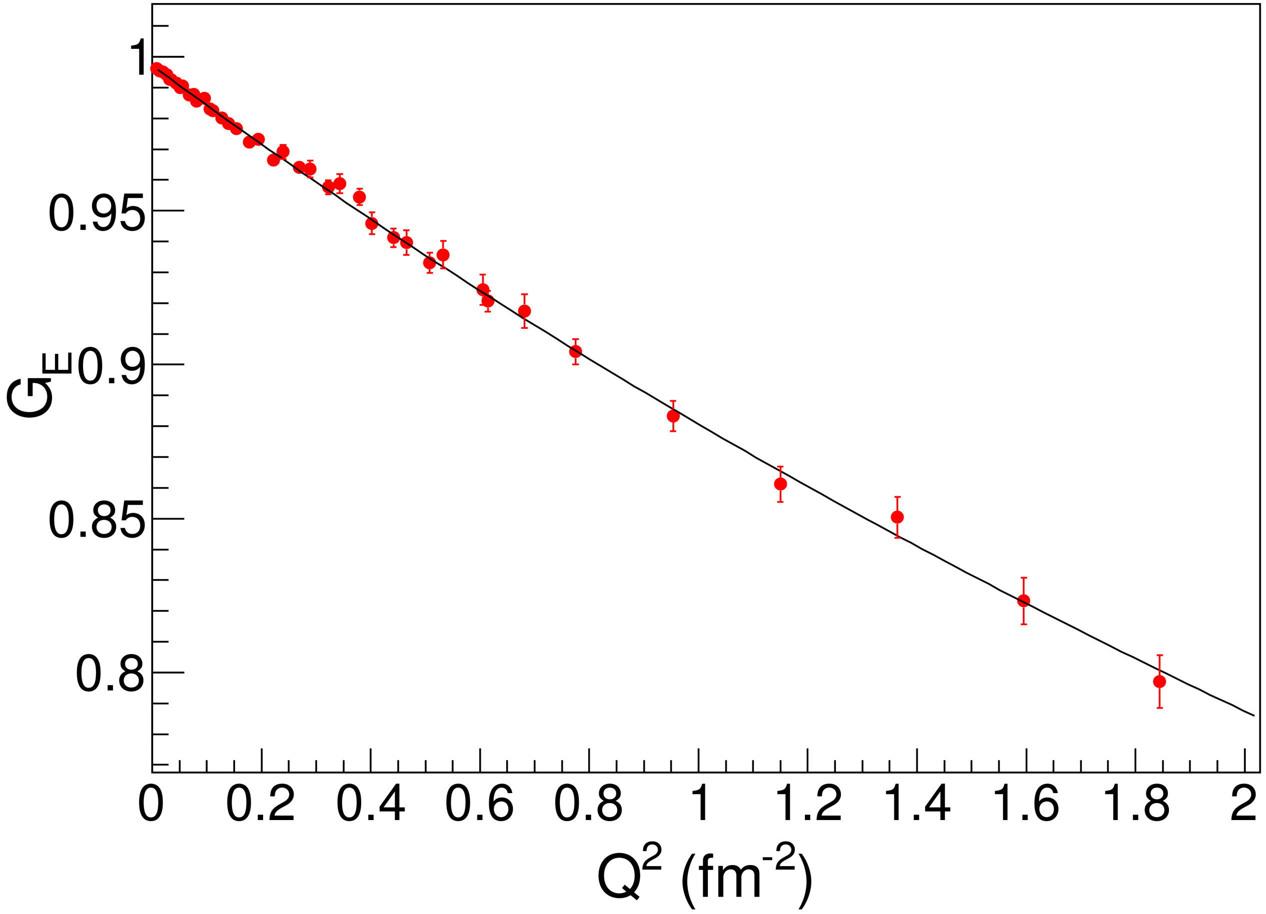


- χ^2=24.05
- χ^2 per dof= 0.59
- R=0.8877$\pm$0.0095 fm

- *D.H.'s favorite functional form*

Functional form: $G_E = p_0(1 - p_1^2 Q^2/6 + p_2 Q^2)/(1 + p_2 Q^2)$, $R = \left(-6 \frac{dG_E}{dQ^2}|_{Q^2=0}\right)^{1/2} = p_1$

Polynomial ratio (3 parameter) fit: 41 points (BAD estimation of fitting error)

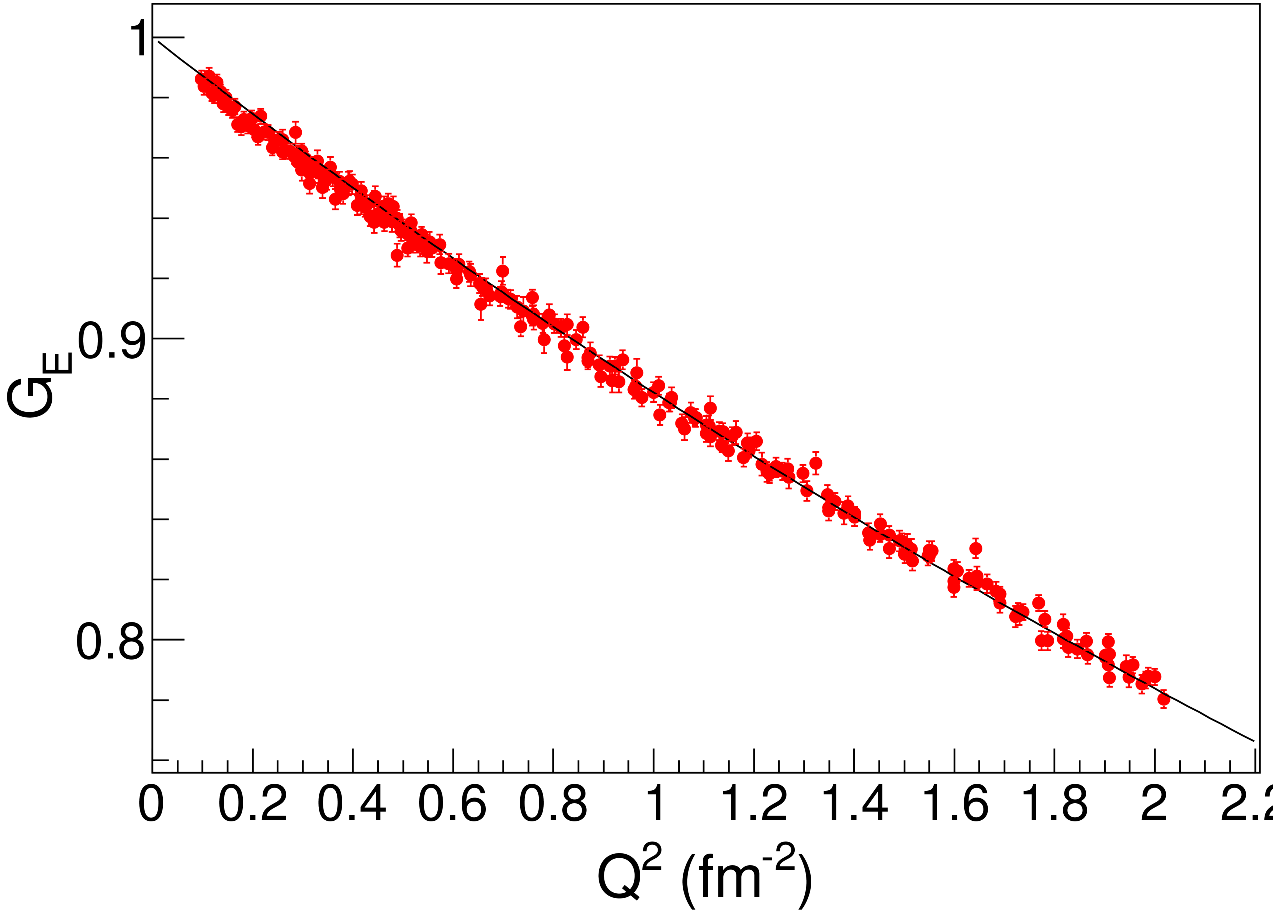


- χ^2=24.05
- χ^2 per dof= 0.59
- R=0.8877±0.1781 fm

- Simple (conservative) error estimation:
  - $R_1 = \sqrt{-6(\frac{p_1 - pE_1}{p_0 + pE_0} - p_2 - pE_2)}$
  - $R_2 = \sqrt{-6(\frac{p_1 + pE_1}{p_0 - pE_0} - p_2 + pE_2)}$
  - $RE = |R_1 - R_2|/2$
  - Capital "E" means error

Functional form: $G_E = (p_0 + p_1 Q^2)/(1 + p_2 Q^2)$, $R = \left(-6\frac{dG_E}{dQ^2}|_{Q^2=0}\right)^{1/2} = \sqrt{-6(\frac{p_1}{p_0} - p_2)}$

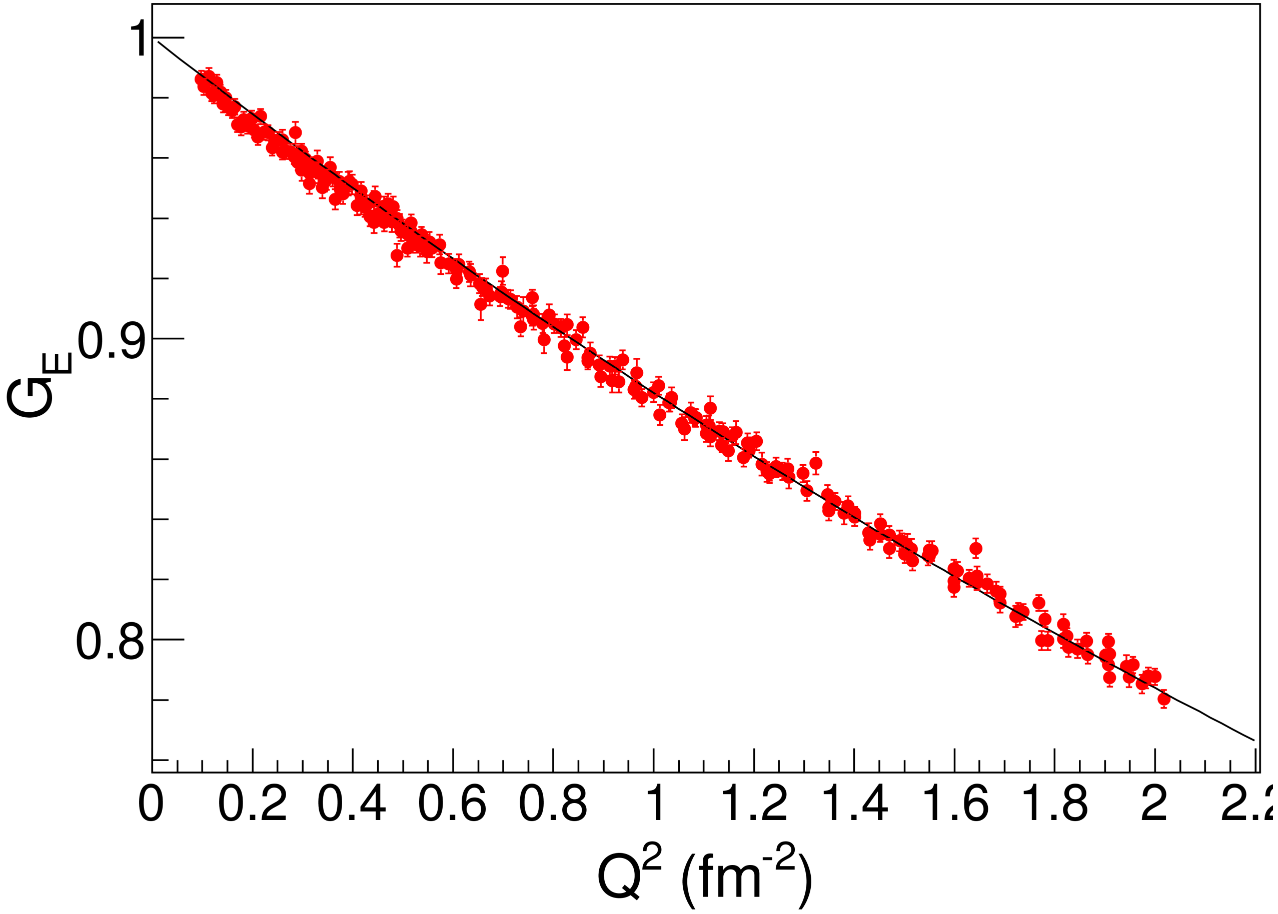
Full dipole functional form (1 parameter) fit: 245 points ($3.8 \times 10^{-3} - 0.08$ GeV^2)



- χ^2=317
- χ^2 per dof= 1.29
- R=0.8816$\pm$0.0008 fm

Functional form: $G_E = \left(1 + \frac{Q^2}{p1}\right)^{-2}$, $R = \left(-6 \frac{dG_E}{dQ^2}|_{Q^2=0}\right)^{1/2}$

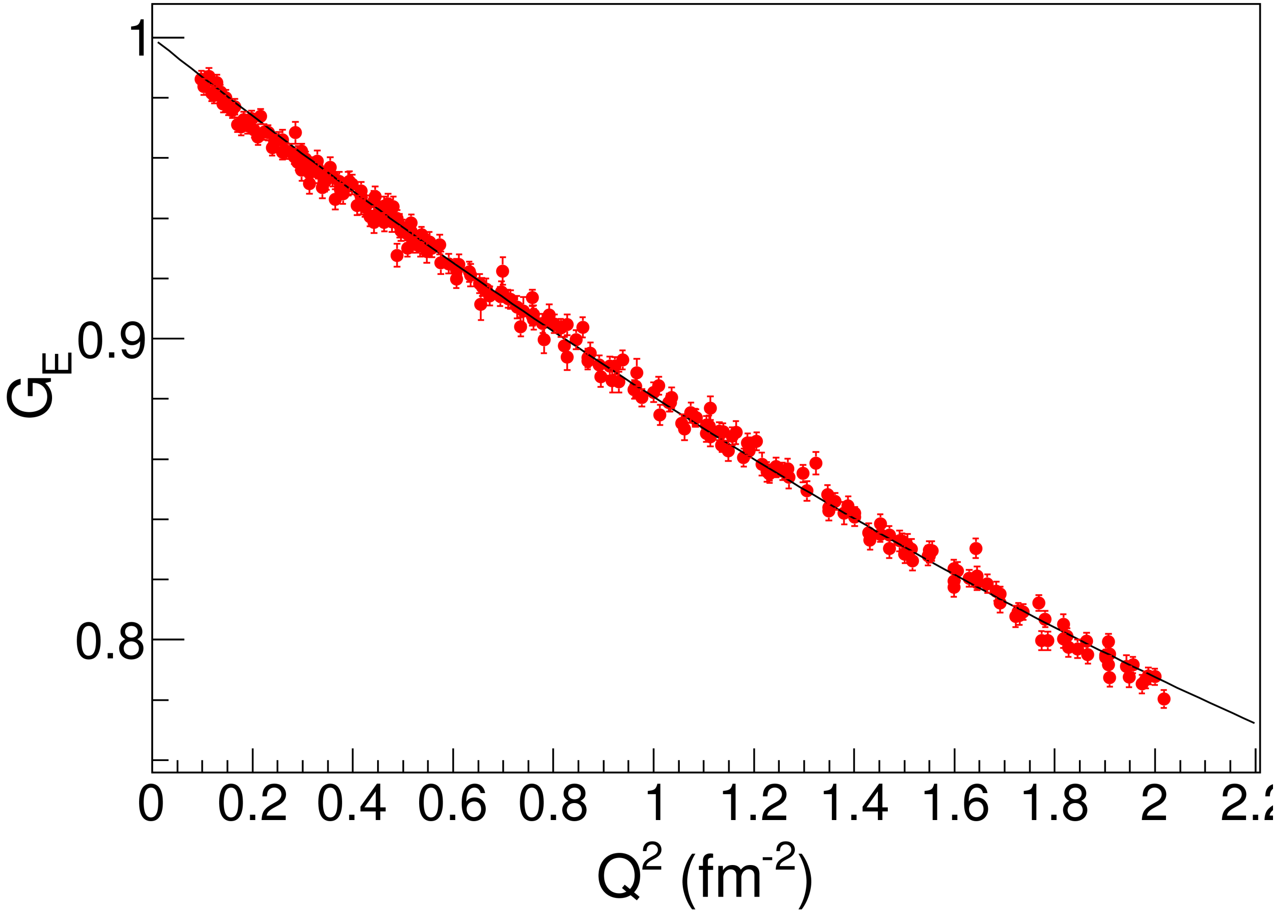
Dipole polynomial expansion fit: 245 points



- Up to Q^4
  - χ^2=280
  - χ^2 per dof= 1.14
  - R=0.8919$\pm$0.0008 fm
- Up to Q^6
  - χ^2=331
  - χ^2 per dof= 1.35
  - R=0.8802$\pm$0.0008 fm
- Up to Q^8
  - χ^2=315
  - χ^2 per dof= 1.29
  - R=0.8818$\pm$0.0008 fm
- Plots visually look ~same

Functional form: $G_E = 1 + \sum_{n\geq 1} \frac{(-1)^n}{(2n+1)!} \langle r^{2n} \rangle Q^{2n}$, $\langle r^{2n} \rangle = \frac{(n+1)(2n+1)}{6} \langle r^2 \rangle \langle r^{2n-2} \rangle$

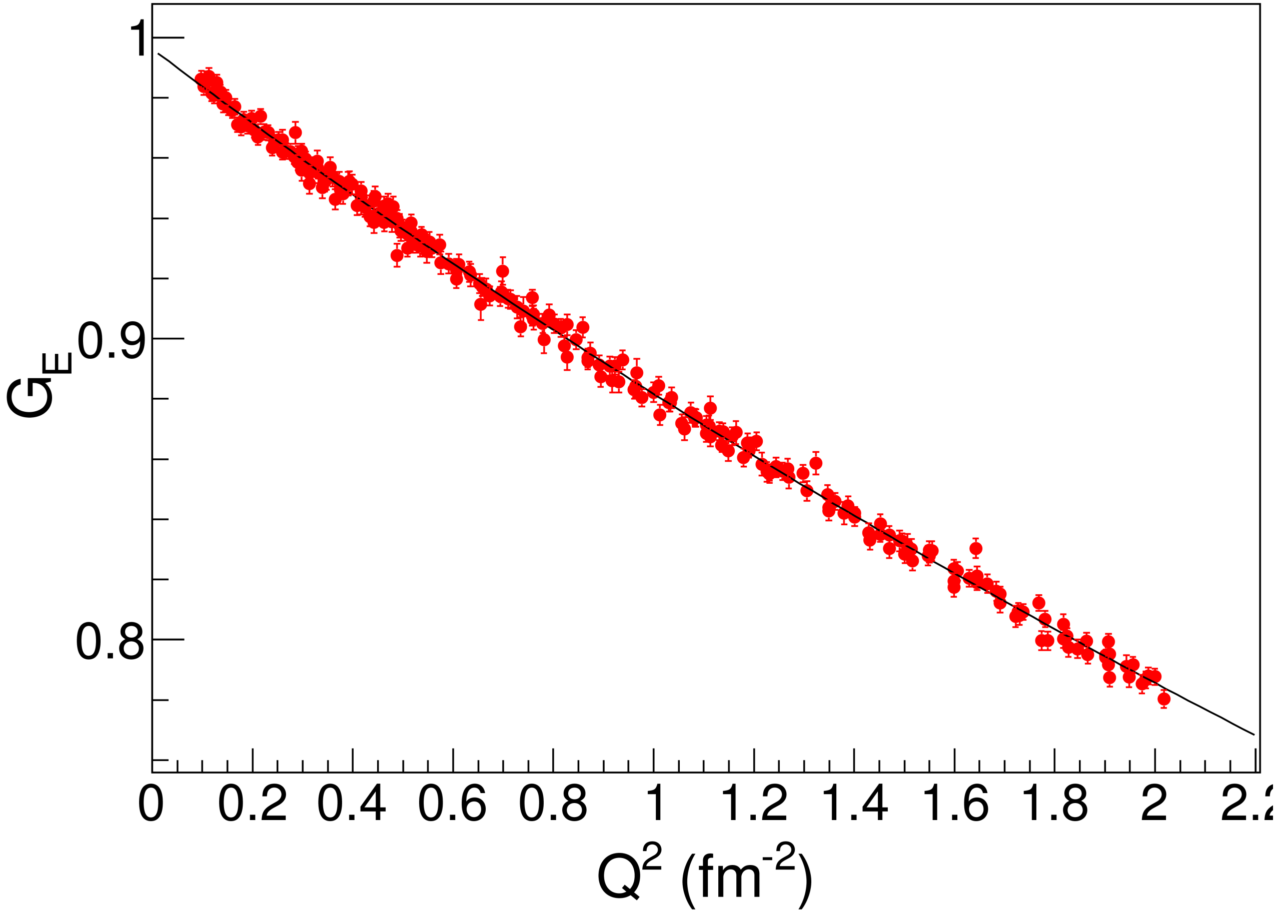
Two parameter polynomial fit: 245 points



- χ^2=279
- χ^2 per dof= 1.14
- R=0.8919$\pm$0.0022 fm

Functional form: $G_E = 1 + \sum_{n\geq 1} \frac{(-1)^n}{(2n+1)!} \langle r^{2n} \rangle Q^{2n}$, up to Q^4

Polynomial ratio (3 parameter) fit: 245 points



- χ^2=224.80
- χ^2 per dof= 0.92
- R=0.8702±0.0059 fm

Functional form: $G_E = p_0(1 - p_1^2 Q^2/6 + p_2 Q^2)/(1 + p_2 Q^2)$, $R = \left(-6 \frac{dG_E}{dQ^2}|_{Q^2=0}\right)^{1/2} = p_1$