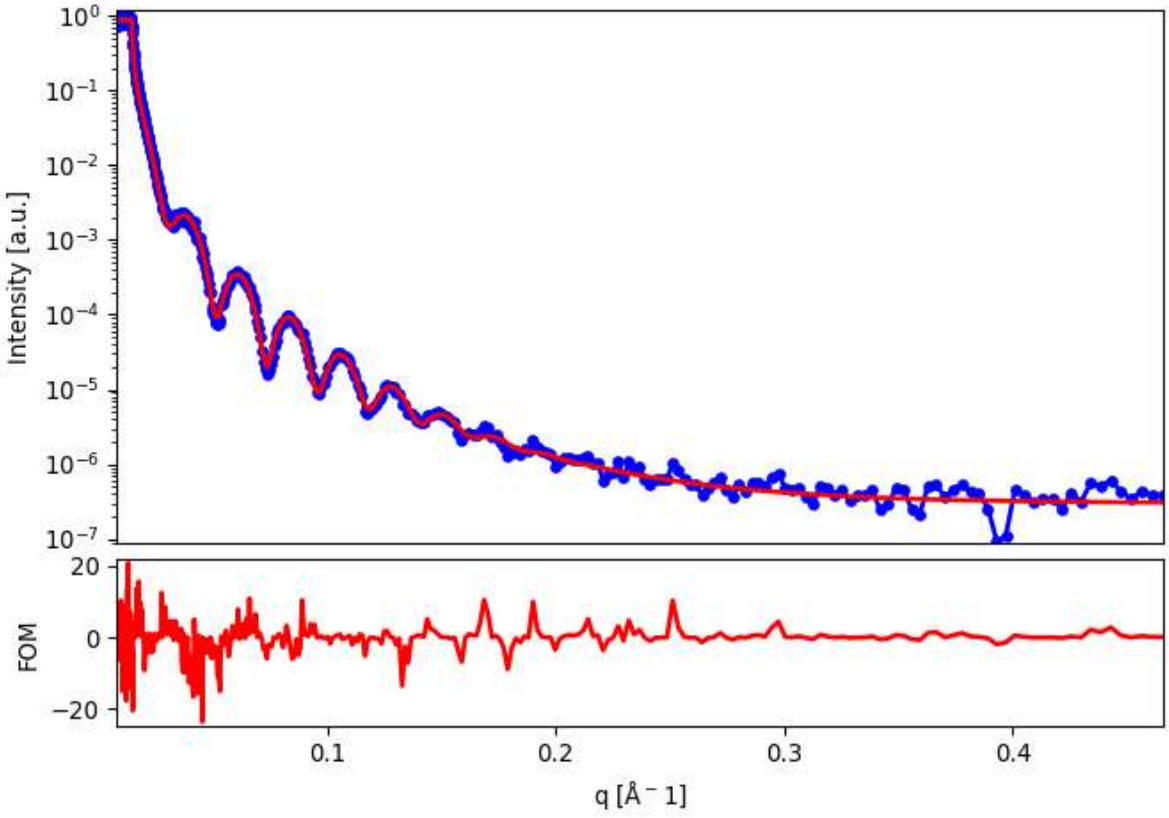
GenX fit results for ORSO example dataset

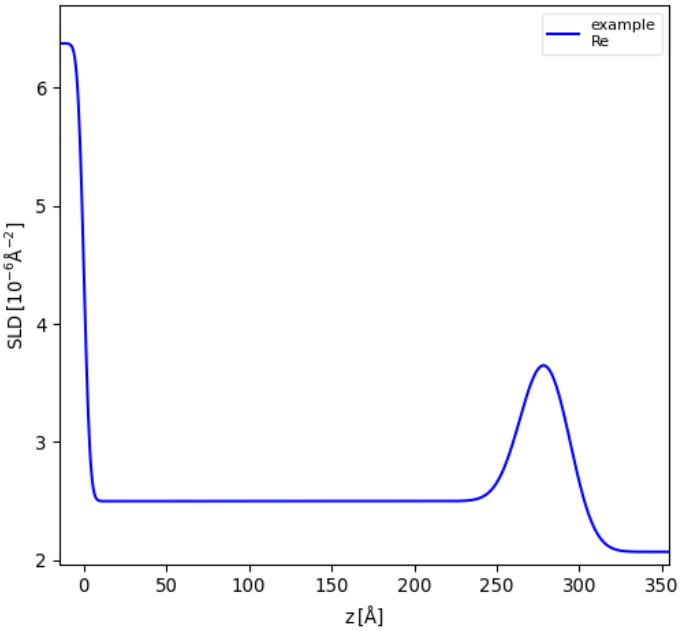
Background set to 3e-7
Intensity set to 0.88

Fitted parameters:

1. SiO2 thickness
2. SiO2 roughness = polymer roughness
3. polymer thickness
4. polymer SLD
5. polymer roughness
6. D2O roughness

Results:

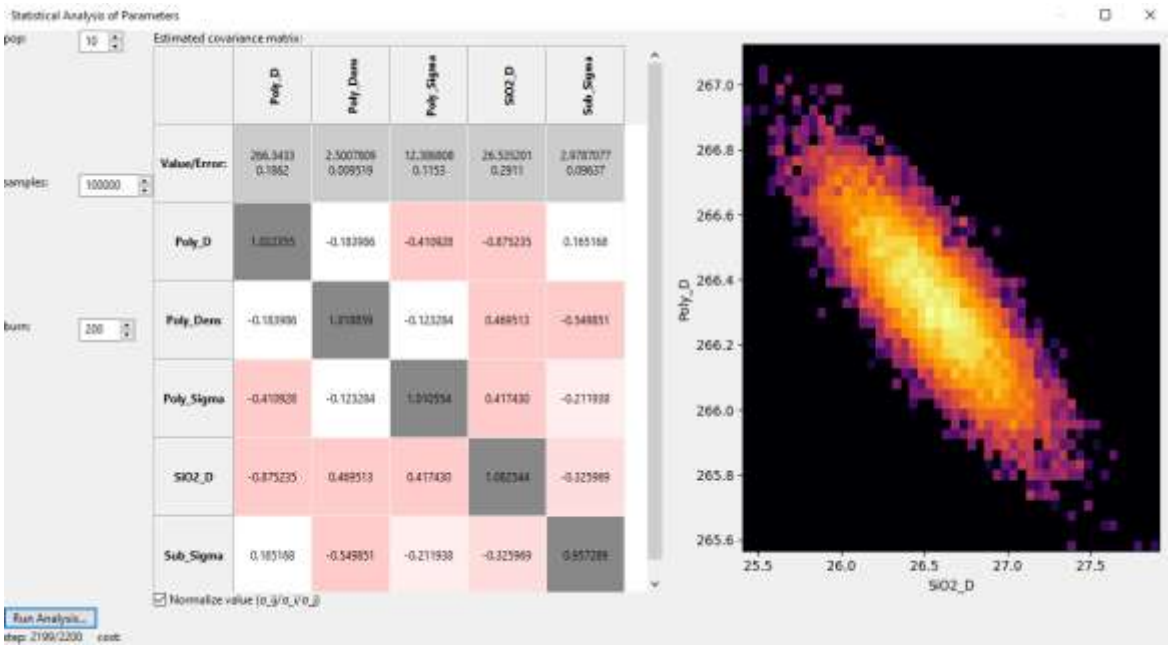




Results GenX fit and error estimation (Chi² with errorbars):

Fit Parameter	Value	Fitted	min	max	error estimation (low, high)
SiO2.setD	26.53520112442988	True	15.0	40.0	(-1.563e+00, 1.966e+00)
Poly.setD	266.34330268934417	True	195.0	325.0	(-1.050e+00, 1.041e+00)
	0.0	False	0.0	0.0	-
Poly.setDens	2.5007808705955425	True	2.025	3.375	(-5.387e-02, 5.525e-02)
	0.0	False	0.0	0.0	-
Poly.setSigma	12.386807928605094	True	2.0	15.0	(-6.696e-01, 7.393e-01)
Sub.setSigma	2.97870765001404	True	2.0	15.0	(-6.807e-01, 6.369e-01)

Statistical Analysis BUMPS MCMC:



Normalized Covariance Matrix (sigma_ij/sigma_i/sigma_j):

	Polymer Thickness	Polymer SLD	Polymer Roughness	SiO2 Thickness	D2O Roughness
Value Error	266.3433 0.1862	2.5007809 0.009519	12.386808 0.1153	26.535201 0.2911	2.9787077 0.09637
Polymer Thickness	1.022355	-0.183986	-0.410928	-0.875235	0.165168
Polymer SLD	-0.183986	1.018859	-0.123284	0.469513	-0.549851
Polymer Roughness	-0.410928	-0.123284	1.010554	0.417430	-0.211938
SiO2 Thickness	-0.875235	0.469513	0.417430	1.082544	-0.325969
D2O Roughness	0.165168	-0.549851	-0.211938	-0.325969	0.957289

Comparison of Results GenX/bumps:

	Polymer Thickness	Polymer SLD	Polymer Roughness	SiO2 Thickness	D2O Roughness
GenX	266.3433 (-1.050e+00, 1.041e+00)	2.5007809 (-5.387e-02, 5.525e-02)	12.386808 (-6.696e-01, 7.393e-01)	26.535201 (-1.563e+00, 1.966e+00)	2.9787077 (-6.807e-01, 6.369e-01)
MCMC	266.3433 0.1862	2.5007809 0.009519	12.386808 0.1153	26.535201 0.2911	2.9787077 0.09637

Code for GenX Script:

```
from numpy import *
import models.spec_nx as model
from models.utils import UserVars, fp, fw, bc, bw

# BEGIN Instrument DO NOT CHANGE
from models.utils import create_fp, create_fw
inst = model.Instrument(probe='neutron', wavelength=1.54, coords='q', I0=0.88,
res=0.001, restype='full conv and varying res.', respoints=11, resintrange=3,
beamw=0.01, footype='no corr', samplelen=10.0, incangle=0.0, pol='uu',
Ibkg=3e-07, tthoff=0.0,)
inst_fp = create_fp(inst.wavelength); inst_fw = create_fw(inst.wavelength)

fp.set_wavelength(inst.wavelength); fw.set_wavelength(inst.wavelength)
# END Instrument

# BEGIN Sample DO NOT CHANGE
Amb = model.Layer(sigma=0.0, dens=2.329*0.602214/28.0855, d=0.0, f=fp.Si*1,
b=bc.Si*1, xs_ai=0.0, magn=0.0, magn_ang=0.0)
SiO2 = model.Layer(sigma=3.0, dens=3.0/113.005, d=20, f=fp.Si*1+fp.O*2,
b=bc.Si*1+bc.O*2, xs_ai=0.0, magn=0.0, magn_ang=0.0)
Poly = model.Layer(sigma=3, dens=2.7, d=260, f=(1e-20+1e-20j), b=0.1,
xs_ai=0.0, magn=0.0, magn_ang=0.0)
Sub = model.Layer(sigma=0.0, dens=1.107*0.602214/20.0276, d=0.0,
f=fp.H*2+fp.O*1, b=bc.i2H*2+bc.O*1, xs_ai=0.0, magn=0.0, magn_ang=0.0)

ML = model.Stack(Layers=[Poly , SiO2], Repetitions = 1)

sample = model.Sample(Stacks = [ML], Ambient = Amb, Substrate = Sub)
# END Sample

# BEGIN Parameters DO NOT CHANGE
cp = UserVars()
# END Parameters

SLD = []
def Sim(data):
    I = []
    SLD[:] = []
    # BEGIN Dataset 0 DO NOT CHANGE
    SiO2.setSigma(Poly.sigma)
```

```
inst.setRes(data[0].res)
d = data[0]
I.append(sample.SimSpecular(d.x, inst))
if _sim: SLD.append(sample.SimSLD(None, None, inst))
# END Dataset 0
return I
```