Update: SIMBA in python

Stella Hoffmann

March

Contents

1 What I did

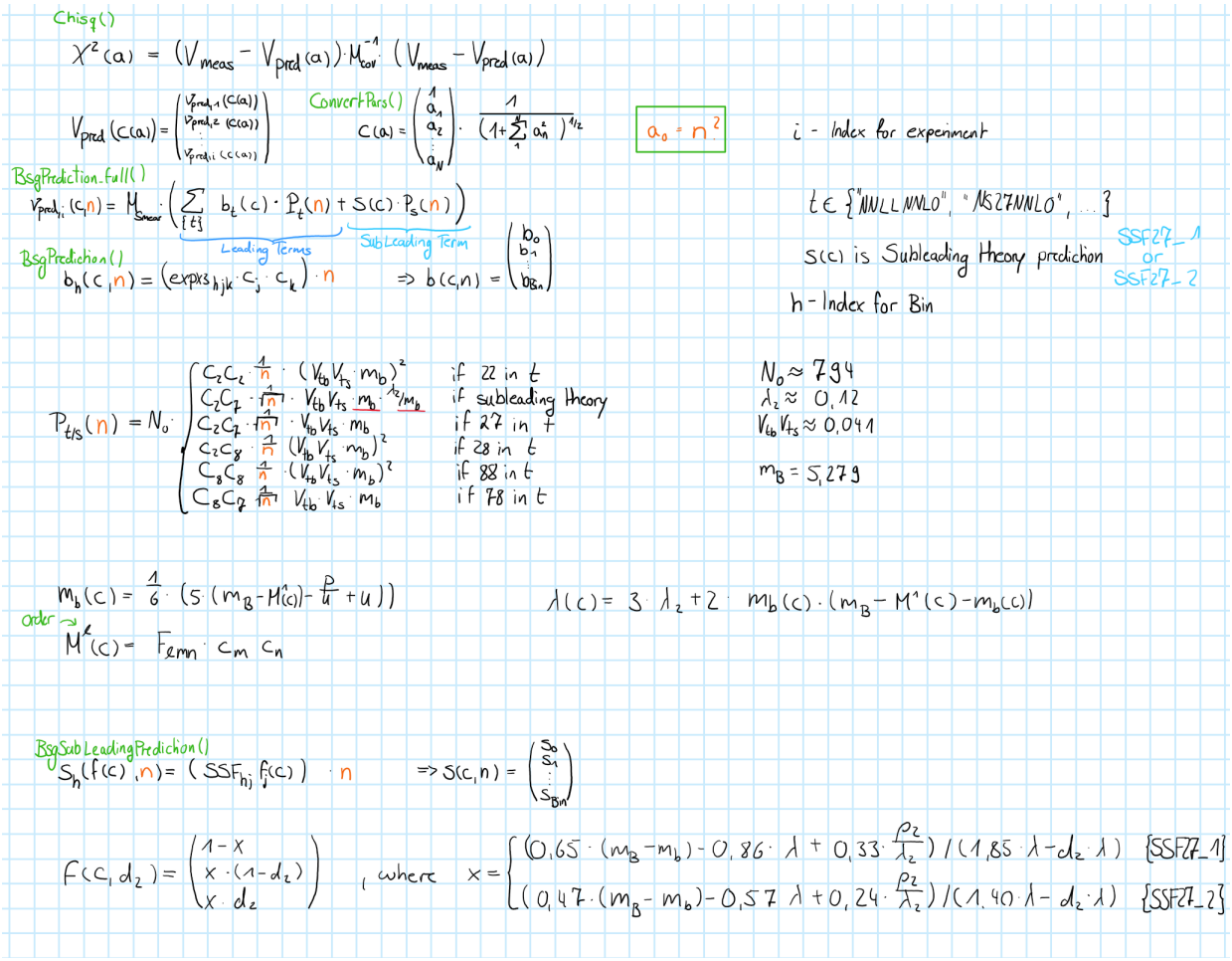

Figure 1: Equations according to my python code

A problem could be the d_2. In my code it's constantly zero. I got this information from the *fit.config* file.

```
##########################################################
#  Subleading theory coefficients
##########################################################

## Default value is
SSF27_1055.d2: 0.
SSF27_2055.d2: 0.
```

Figure 2: Information in *fit.config* file

2 Comparisons

As you can see in the following, something is off with the subleading theory. To compare different fits, I always plotted the Fit with the subleading prediction added on the leading prediction and with just the leading prediction. For the fitting I used 4 to 7 start parameters, which gave different results for the prediction, as it is shown in the columns. Every pair of rows has more amount of measurements included in the fit (*["babar_incl",*

"babar_hadtag", "babar_sem", "belle"]).

2.1 Results for each fit

In the tables and in the following figueres, I circled the best looking fit green. The best looking fit was the one with the lowest χ^2 and the m_b which was closest to the result from the paper, which is listet in the first table in the last row. The best looking fit is without the subleading theory, and looks good for the first two measurements (*"babar_incl", "babar_hadtag"*) but for the third (*"babar_sem"*) it starts to differ a lot from the aimed fit.

Number of included measurements	With or without subleading theory	Number of fitted parameters	χ^2	m_b [GeV]	a_0 (Norm)
2	subleading	4	174477.25	4.6877	0.9204
2	subleading	5	143976.56	4.6900	0.9514
2	subleading	6	145631.54	4.6843	0.9678
2	subleading	7	248645.38	4.5712	0.9708
2	leading	4	230.97	4.6932	0.3826
2	leading	5	323.33	4.8005	0.5210
2	leading	6	262.79	4.8515	0.5774
2	leading	7	292.88	4.1603	0.6252
3	subleading	4	91211.26	4.6821	0.9228
3	subleading	5	67852.28	4.6845	0.9519
3	subleading	6	66673.56	4.6711	0.9680
3	subleading	7	139735.90	4.5526	0.9721
3	leading	4	167.13	3.9984	0.5020
3	leading	5	258.61	4.7932	0.4933
3	leading	6	156.92	3.9980	0.8149
3	leading	7	231.09	4.7644	0.2352
4	subleading	4	45178.57	4.7032	0.9143
4	subleading	5	19.59	3.4633	0.0426
4	subleading	6	23165.42	4.7386	0.9323
4	subleading	7	22746.34	4.5274	0.9208
4	leading	4	12.36	4.5904	0.4373
4	leading	5	11.72	4.4835	0.6715
4	leading	6	10.10	4.7076	0.3762
4	leading	7	14.15	4.2259	0.0892
Result from paper				4.764	

Table 1: Values calculated with the fit-parameters

n_{meas}		n_{pars}	c_0	c_1	c_2	c_3	c_4	c_5	c_6
2	s	4	0.896	-0.394	0.159	0.126			
2	s	5	0.855	-0.438	0.181	0.182	0.102		
2	s	6	0.842	-0.450	0.183	0.203	0.121	0.011	
2	s	7	0.709	-0.101	-0.205	0.625	-0.107	0.184	-0.096
2	l	4	0.890	-0.441	0.009	0.113			
2	l	5	0.025	-0.089	0.758	-0.646	-0.023		
2	l	6	0.032	0.043	0.284	-0.739	0.608	0.020	
2	l	7	0.000	-0.494	0.524	0.457	0.496	0.153	-0.058
3	s	4	0.910	-0.378	0.139	0.095			
3	s	5	0.867	-0.422	0.176	0.169	0.098		
3	s	6	0.852	-0.433	0.174	0.200	0.123	0.030	
3	s	7	0.742	-0.153	-0.116	0.599	-0.097	0.178	-0.115
3	l	4	0.000	0.880	0.473	-0.033			
3	l	5	0.032	-0.090	0.642	-0.760	-0.018		
3	l	6	0.000	0.815	0.568	-0.110	0.019	0.001	
3	l	7	0.000	0.371	-0.061	0.489	-0.775	0.053	0.126
4	s	4	0.857	-0.433	0.214	0.178			
4	s	5	0.022	0.587	0.531	0.494	0.358		
4	s	6	0.242	-0.178	0.812	-0.387	0.062	0.310	
4	s	7	0.132	0.431	-0.225	0.589	0.300	-0.438	0.343
4	l	4	0.231	0.391	-0.891	-0.017			
4	l	5	0.148	-0.163	0.974	0.055	-0.005		
4	l	6	0.034	0.875	-0.445	0.107	-0.096	-0.122	
4	l	7	0.019	0.297	0.117	-0.089	0.062	-0.907	-0.251

Table 2: c_n calculated by a_n which are the fitted parameters

In the following: The **red dots** show my calculated prediction.
The **black dots** show the experimental values, which I extracted from the root files.
The **green line** shows the fit I extracted from the root files.
The **blue line** shows the difference between the green line and the red dots.

2.2 Babar hadronic tag

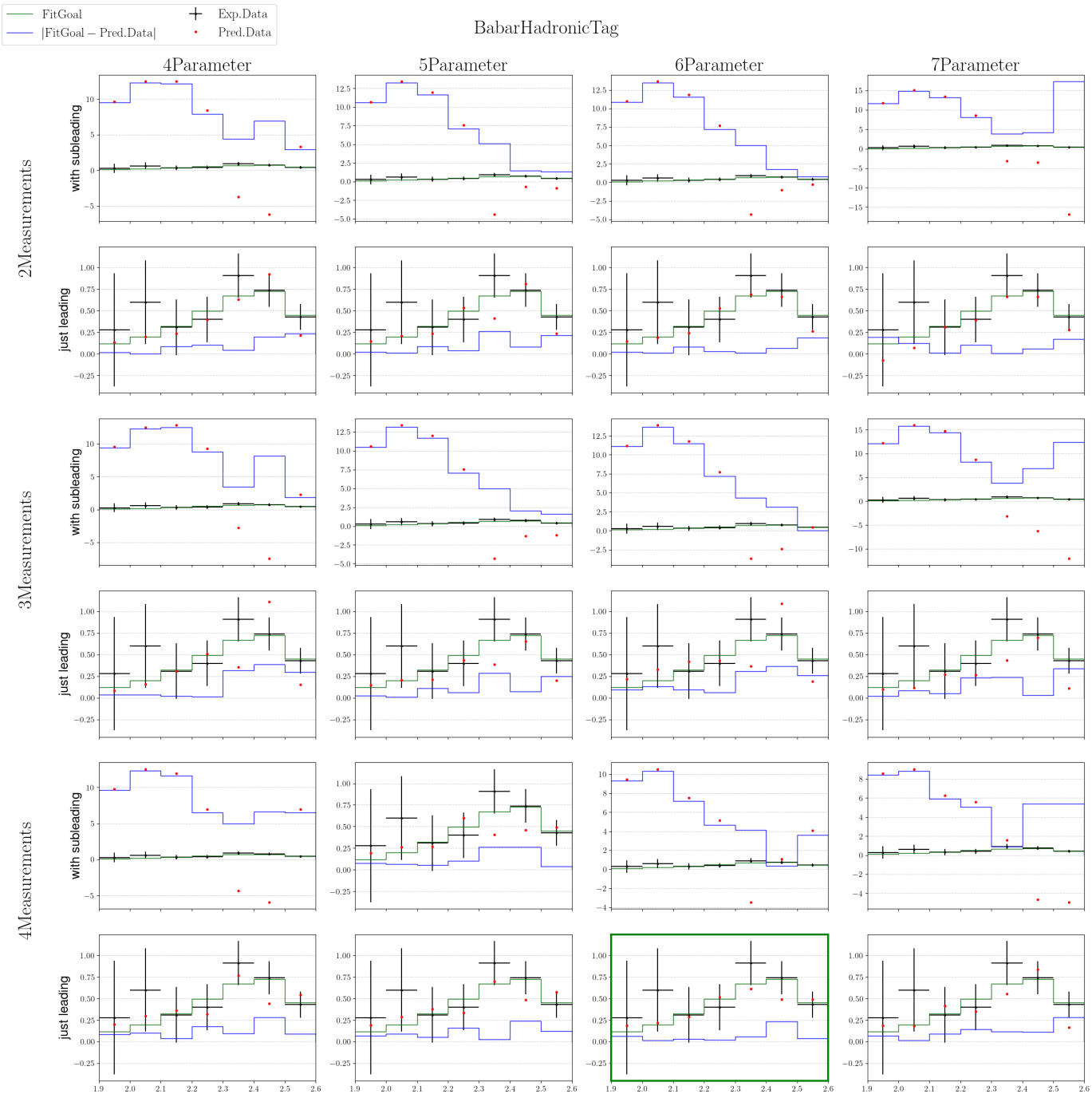

Figure 3: Fit Comparison for 'babar_hadtag'

2.3 Babar inclusive spectra

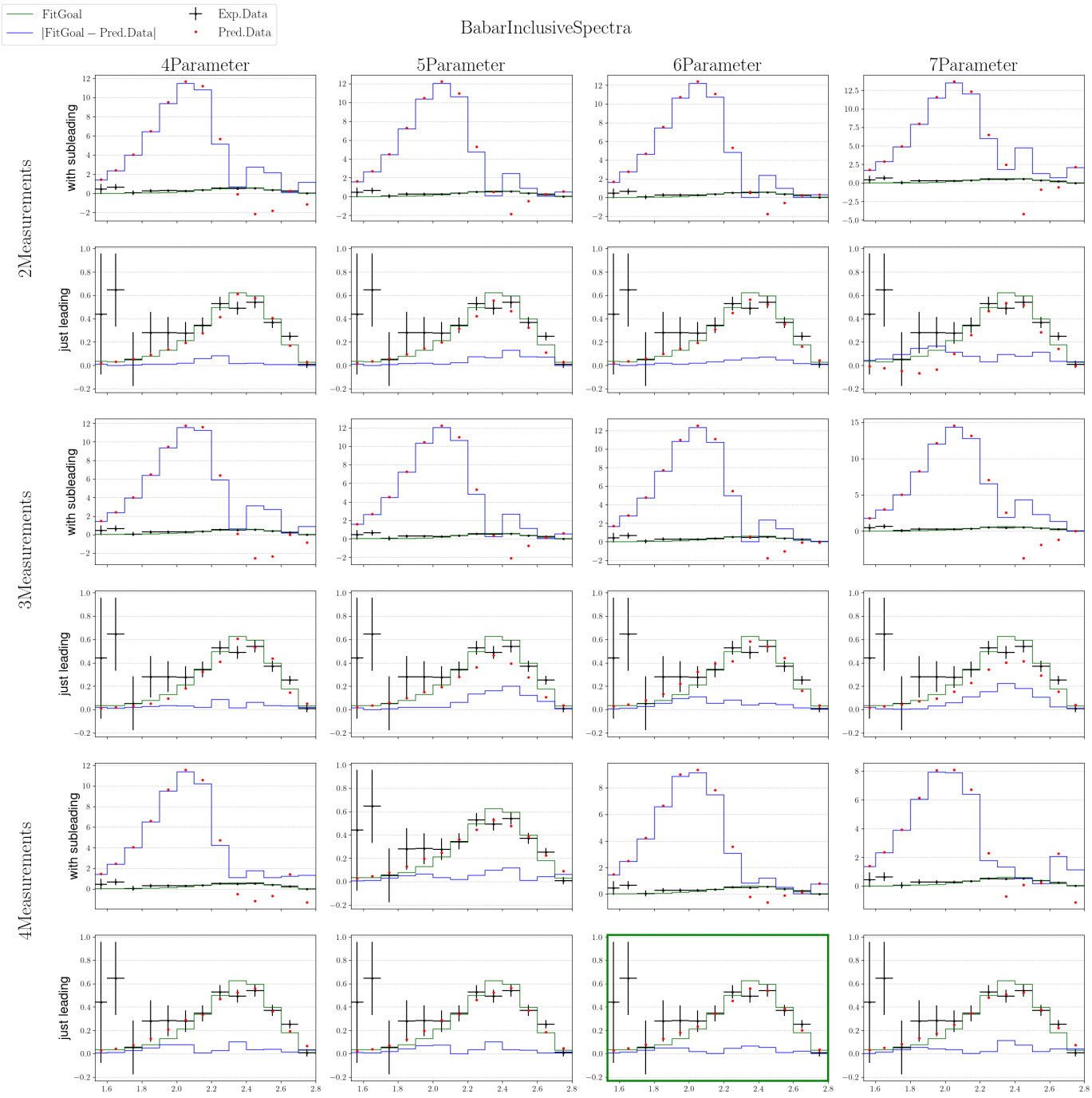

Figure 4: Fit Comparison for 'babar_incl'

2.4 Babar semileptonic

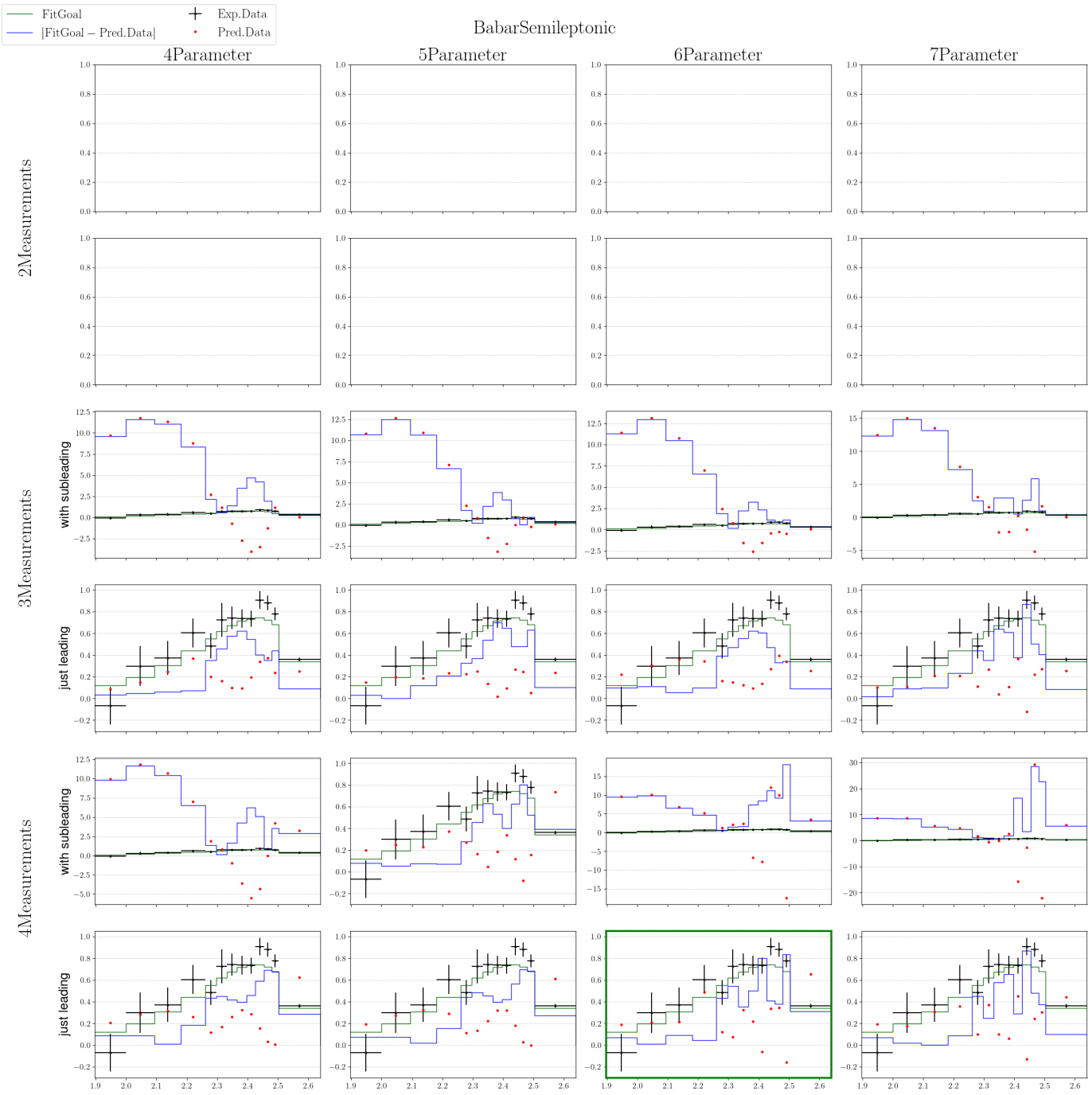

Figure 5: Fit Comparison for 'babar_sem'

2.5 Belle

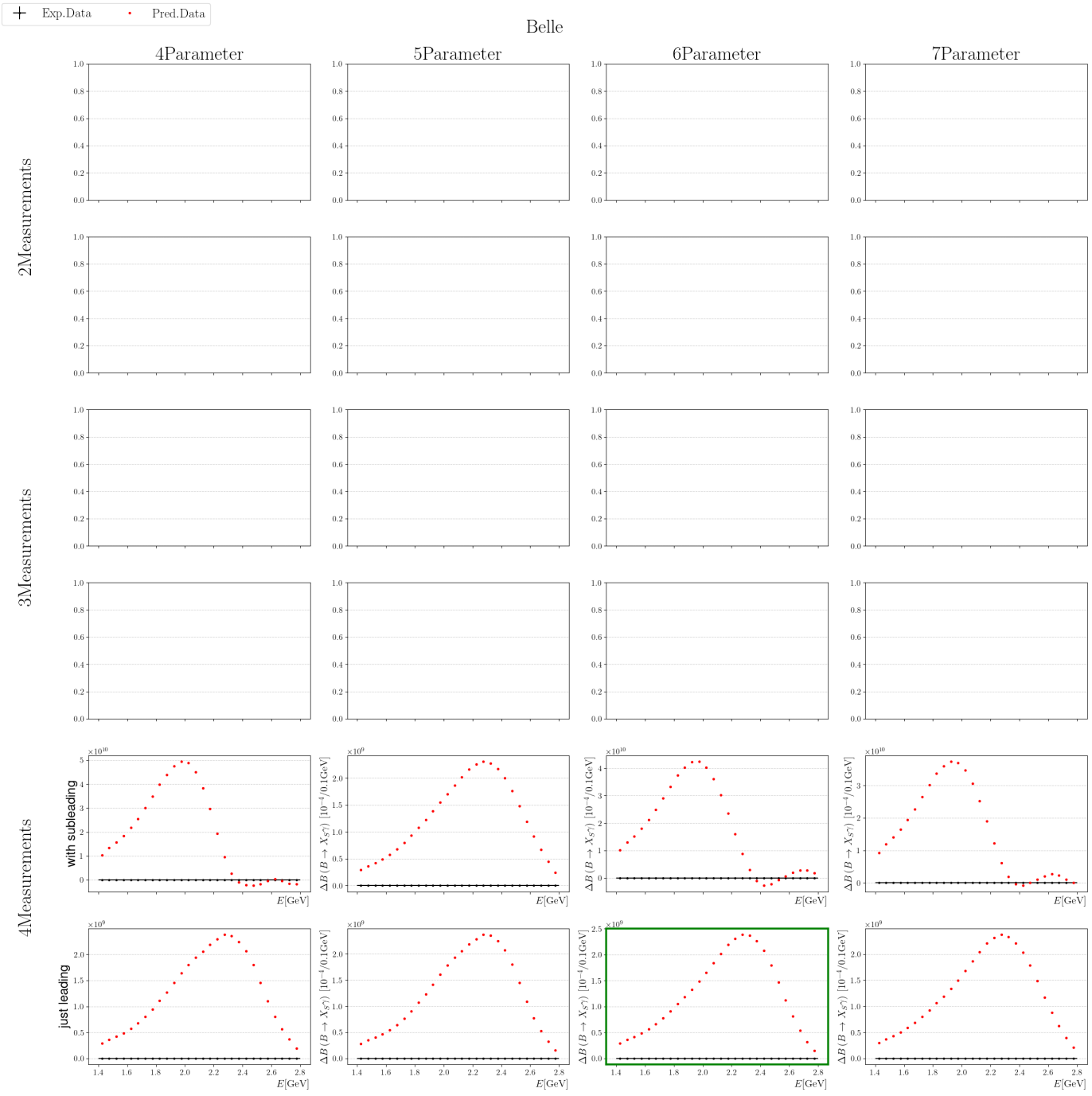

Figure 6: Fit Comparison for 'belle'

2.6 Just Babar semileptonic

The fit for *'babar_sem'* looked off in the previous plots, even though the χ^2 was low. So I fitted only using the data from *'babar_sem'*. As you can see in the following, the fit is not good, while for *'babar_hadtag'* and *'babar_incl'* it works quiet well, there could be a problem with the input I'm giving the fit for *'babar_sem'*.

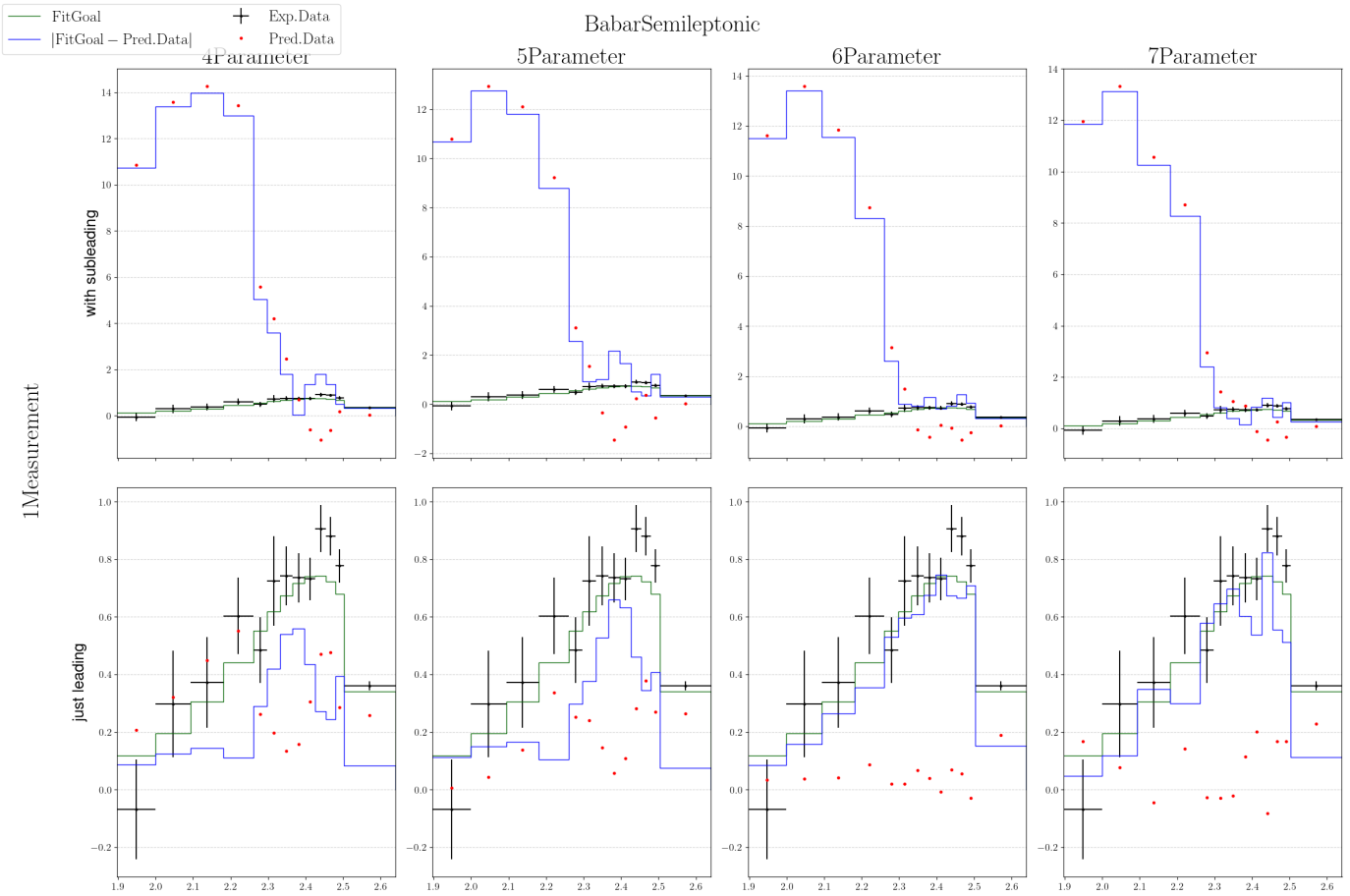

Figure 7: Fit Comparison while using just the data from *'babar_sem'*

Number of included measurements	With or without subleading theory	Number of fitted parameters	χ^2	m_b [GeV]	a_0 (Norm)
1	subleading	4	25004.71	4.6698	0.9373
1	subleading	5	19358.95	4.6747	0.9563
1	subleading	6	17826.66	4.6622	0.9707
1	subleading	7	15242.57	4.6587	0.9786
1	leading	4	116.98	3.9847	0.5392
1	leading	5	150.55	4.0966	0.6603
1	leading	6	237.47	4.0095	0.0095
1	leading	7	168.51	3.5833	0.5549
Result from paper				4.764	

Table 3: Values calculated with the fit-parameters just using *'babar_sem'*

n_{meas}		n_{pars}	c_0	c_1	c_2	c_3	c_4	c_5	c_6
1	s	4	0.931	-0.361	0.053	0.025			
1	s	5	0.905	-0.382	0.130	0.112	0.073		
1	s	6	0.885	-0.401	0.128	0.163	0.101	0.044	
1	s	7	0.874	-0.411	0.108	0.119	0.186	0.063	0.056
1	l	4	0.000	0.870	0.492	-0.029			
1	l	5	0.000	0.856	0.476	-0.202	-0.004		
1	l	6	0.077	0.669	0.224	-0.277	-0.552	-0.339	
1	l	7	0.000	-0.178	-0.780	-0.554	-0.209	-0.060	-0.070

Table 4: c_n calculated by a_n which are the fitted parameters using only data from *'babar_sem'*

3 Open Questions

1. What is d_2 in the subleading theory and what values has it?
2. Something is off with *'babar_sem'*, how can I fix it?