PROCEEDINGS OF SCIENCE

# smelli – the SMEFT Likelihood

**Peter Stangl**[a,*]

[a] *Albert Einstein Center for Fundamental Physics, Institute for Theoretical Physics, University of Bern, Sidlerstrasse 5, CH-3012 Bern, Switzerland*

*E-mail:* stangl@itp.unibe.ch

I present the Python package `smelli` that implements a global likelihood function in the space of dimension-six Wilson coefficients in the Standard Model Effective Field Theory (SMEFT). The likelihood includes contributions from a large number of flavor and other precision observables, currently 399 in total.

*Tools for High Energy Physics and Cosmology - TOOLS2020*
*2-6 November, 2020*
*Institut de Physique des 2 Infinis (IP2I), Lyon, France*

*Speaker





PoS(TOOLS2020)035

## Contents





## 1. Introduction

The Standard Model (SM) of particle physics is an extremely successful model. However, there are several experimental as well as theoretical indications for new physics (NP) beyond the SM. Whether a given NP scenario describes the experimental data better than the SM can be conveniently quantified by the ratio of the NP likelihood $L_{\text{NP}}$ and the SM likelihood $L_{\text{SM}}$ or, equivalently, by the difference of the log-likelihoods

$$\Delta \log L = \log L_{\text{NP}} - \log L_{\text{SM}} . \tag{1}$$

These likelihood functions are constructed from a set of measured observables and take into account uncertainties and correlations from both the measurements and the theoretical predictions.

A set of observables for which certain NP scenarios can describe the experimental data considerably better than the SM have been found e.g. in $B$ meson decays. These so-called $B$ anomalies correspond to deviations from the SM predictions in measurements of neutral current $b \to s\ell\ell$ and charged current $b \to c\ell\nu$ transitions. In particular, deviations have been found in

*(i)* angular observables of $B \to K^*\mu^+\mu^-$ [1–5],

*(ii)* branching ratios of $B \to K\mu^+\mu^-$, $B \to K^*\mu^+\mu^-$, and $B_s \to \phi\mu^+\mu^-$ [6–8],

*(iii)* the lepton flavor universality (LFU) observables $R_{K^{(*)}}$ [9–12], which are $\mu/e$ ratios of $B \to K^{(*)}\ell^+\ell^-$ branching ratios,

*(iv)* the branching ratio of $B_s \to \mu^+\mu^-$ [13–17],

*(v)* the LFU observables $R_{D^{(*)}}$ [18–25], which are $\tau/e$ and $\tau/\mu$ ratios of $B \to D^{(*)}\ell\nu$ branching ratios.

While *(i)* and *(ii)* could be afflicted by underestimated hadronic uncertainties, the observables in *(iii)*, *(iv)*, and *(v)* are theoretically clean probes of NP [26–28]. Considering the above $B$-decay observables and parameterizing NP in $b \to s\ell\ell$ and $b \to c\ell\nu$ transitions in terms of Wilson coefficients in the Weak Effective Theory (WET), simple one- and two-parameter scenarios show a sizable $\Delta \log L \sim 20$ (cf. e.g. [29–34]).

These intriguing hints for NP have led to extensive model building. In the process, important insights have been gained:

- The fact that NP above the electroweak (EW) scale has to respect SM gauge invariance leads to important correlations between low-energy observables. For example, explanations of $R_{D^{(*)}}$ in terms of left-handed contributions to $b \to c\tau\nu$ imply also contributions to $b \to s\nu\nu$, which are constrained by $B \to K^{(*)}\nu\bar{\nu}$ [35].
- One-loop contributions can have very important effects. This has been observed in models explaining $R_{D^{(*)}}$ and $R_{K^{(*)}}$ using mostly 3rd generation couplings. They actually modify $\tau$ and $Z$ decays at one loop, which leads to strong constraints [36]. Another example is provided by models explaining $R_{D^{(*)}}$ using a contribution to semi-tauonic operators, which generate an effect in $b \to s\ell\ell$ at one loop [37, 38].

Essentially every model that explains some of the $B$ anomalies predicts deviations from the SM also in other observables. In many cases, this leads to strong constraints or exclusion of a model. So phenomenological analyses that consider only a small set of observables or neglect one-loop contributions are in many cases not sufficient to show that a given model agrees with experimental data better than the SM. In order to show this, it is in general necessary to

- compute *all relevant observables* $\vec{O}(\vec{\xi})$ (flavor observables, EW precision observables (EWPO), etc.) in terms of the Lagrangian parameters $\vec{\xi}$ of a NP model,
- take into account loop effects when computing the observables,
- compare the theory predictions to experimental data by constructing the NP likelihood $L_{\text{NP}}$.

Performing these steps again and again for each single model one wants to analyze is a tedious task. Fortunately, analyses of NP models can be tremendously simplified by making use of the SM effective field theory (SMEFT) in an intermediate step.

## 2. The SMEFT Likelihood

Assuming that the scale of NP $\Lambda_{\text{NP}}$ is considerably larger than the EW scale and EW symmetry breaking is realized linearly, the NP effects in a given observable can be expressed in terms of the Wilson coefficients $C_i$ of the SMEFT, which are defined by the SMEFT Lagrangian [39, 40]

$$\mathcal{L}_{\text{SMEFT}} = \mathcal{L}_{\text{SM}} + \sum_{n>4} \sum_i \frac{C_i}{\Lambda_{\text{NP}}^{n-4}} O_i \,, \tag{2}$$

where $O_i$ are local SM gauge invariant operators constructed from the SM fields and $n$ is their canonical dimension.

The SMEFT is a powerful tool since it can connect the model building at the high scale $\Lambda_{\text{NP}}$ to the phenomenology at lower scales without the need to compute hundreds of observables in each model. A phenomenological analysis can be split into

- a model-dependent part that consists of matching the NP model to the SMEFT at the scale $\Lambda_{\text{NP}}$,
- the model-independent phenomenology, which corresponds to
  - running down the Wilson coefficients $\vec{C}$ from $\Lambda_{\text{NP}}$ to the low scale at which the observables are computed,
  - predicting all the relevant observables $\vec{O}(\vec{C})$ in terms of the Wilson coefficients $\vec{C}$,
  - constructing the NP likelihood $L_{\text{NP}}(\vec{O}(\vec{C}))$ that compares the predictions to experimental measurements,
  - computing $\Delta \log L$ using eq. (1) in order to compare the NP model to the SM.

While it might be preferable to perform the model-dependent matching at one-loop, a large number of important one-loop effects is actually already included by the model-independent renormalization group (RG) running and mixing in the SMEFT.

Using the above procedure, a SMEFT likelihood function $L_{\text{NP}}(\vec{C})$ can tremendously simplify analyses of NP models. Many likelihood functions in the SMEFT have been considered in the literature (see e.g. [41–55]). However, most of them are constructed from observables in one or few specific sectors, like EWPO, Higgs physics, top physics, $B$ physics, or lepton flavor violating observables. But as discussed above, NP models generically predict new effects in several observables of various sectors. Furthermore, SMEFT operators belonging to different sectors mix under renormalization. Consequently, to test a NP model, the sectors should not be considered separately. It is in fact necessary to construct the *global* SMEFT likelihood, taking into account as many observables from as many sectors as possible.

## 3. The `smelli` Python package

In [56], we have started constructing a global SMEFT likelihood that is provided by the Python package `smelli` (SMEFT likelihood). It is based on

- the Python package `flavio` [57] that can compute hundreds of flavor and other precision observables in and beyond the SM, while properly accounting for theory uncertainties,
- the Wilson coefficient exchange format (WCxf) [58] that is used to represent and exchange large sets of Wilson coefficients in various EFTs and bases,
- the Python package `wilson` [59] that performs the RG evolution in the SMEFT and the WET as well as the matching between them.

`smelli` is built upon these tools and implements a SMEFT likelihood function constructed from currently 399 observables. In particular, it includes

- flavor-changing neutral current $B$ decays,
- lepton flavor universality tests in charged- and neutral-current $B$ and $K$ decays,
- meson-antimeson mixing in the $K$, $B$, and $D$ systems,
- charged lepton flavor violating $B$, tau, and muon decays,
- the anomalous magnetic moments of the electron, muon, and tau,
- $Z$ and $W$ pole EWPO,
- nuclear and neutron beta decays,
- Higgs signal strengths.

Given any combination of SMEFT or WET Wilson coefficients, `smelli` computes the $\Delta \log L$ for each of the above sectors and then sums all of them to obtain the global $\Delta \log L$.

The *full global* likelihood is work in progress and the development is open to everyone. The open-source code of `smelli` is available at https://github.com/smelli/smelli.

### 3.1 Installation

The requirements for `smelli` are a working installation of Python version 3.5 or above and the Python package manager `pip`. If both are present, `smelli` can be installed from the command line by entering

```
python3 -m pip install smelli --user
```

This will download `smelli` and all its dependencies from the Python package archive (PyPI) and install it in the user's home directory without requiring root privileges (due to the option `--user`).

### 3.2 Using smelli

Like any Python package, `smelli` can be used

- as a library imported from other scripts,
- directly in the command line interpreter,
- in an interactive session, e.g. in a Jupyter notebook.

How to use `smelli` is demonstrated in the following with examples from an interactive Jupyter notebook. This notebook is available at https://github.com/peterstangl/smelli-talk. For further information on the features of `smelli`, see [56] and the API documentation at https://smelli.github.io.

#### 3.2.1 Instantiating the likelihood

The main functionality of `smelli` is provided by the `GlobalLikelihood` class. It is imported by

In:
```
from smelli import GlobalLikelihood
```

If the `GlobalLikelihood` class is instantiated without any argument,

In:
```
gl = GlobalLikelihood()
```

the likelihood is defined in the space of SMEFT Wilson coefficients in the Warsaw basis (for details on the specifications of the supported EFTs and bases, see the WCxf website at https://wcxf.github.io/bases.html). The EFT and basis of a given GlobalLikelihood instance can be accessed via its eft and basis attributes.

In:
```
gl.eft, gl.basis
```

Out: ('SMEFT', 'Warsaw')

In order to create a likelihood function of Wilson coefficients in the WET, one can provide the eft and basis arguments on instantiation of a GlobalLikelihood instance.

In:
```
gl_wet = GlobalLikelihood(eft='WET', basis='flavio')
gl_wet.eft, gl_wet.basis
```

Out: ('WET', 'flavio')

#### 3.2.2 Fixing a point in Wilson coefficient space: 3 equivalent ways

The point in the Wilson coefficient space at which the likelihood should be computed is defined using the parameter_point method. This method returns an instance of the GlobalLikelihoodPoint class that can be used to compute $\Delta \log L$. The values of the Wilson coefficients can be provided in three equivalent ways:

- A dictionary of Wilson coefficients as well as the scale in GeV at which they are defined can be passed directly as arguments.

In:
```
pp = gl.parameter_point({'lq3_2223': 1e-9}, scale=1000)
```

- An instance of the Wilson class from the wilson package can be passed as a single argument.

In:
```
from wilson import Wilson
w = Wilson({'lq3_2223': 1e-9}, scale=1000,
           eft='SMEFT', basis='Warsaw')
pp = gl.parameter_point(w)
```

- A WCxf file, e.g. a file in YAML format named my_wcxf.yaml and containing

```
eft: SMEFT
basis: Warsaw
scale: 1000
values:
  lq3_2223:
    Re: 1e-9
```

can be read in by providing the path to the file as argument.

In:
```
pp = gl.parameter_point('my_wcxf.yaml')
```

#### 3.2.3 Computing the likelihood

After the Wilson coefficients have been fixed and an instance of `GlobalLikelihoodPoint` has been created, it can be used to compute $\Delta \log L$. In `smelli`, the global $\Delta \log L$ is given in terms of the sum of several individual $\Delta \log L$ that are constructed from subsets of observables. To access all these individual $\Delta \log L$, the method `log_likelihood_dict` can be used. It returns a dictionary containing the names of the individual likelihoods and the corresponding $\Delta \log L$ values. Using the above defined parameter point, one gets

```
In: pp.log_likelihood_dict()
```

```
Out: {'fast_likelihood_quarks.yaml': 18.063309775625527,
      'fast_likelihood_leptons.yaml': -7.954151298861234e-05,
      'likelihood_ewpt.yaml': 0.0019331634397694586,
      'likelihood_eeww.yaml': -0.0001731988511934901,
      'likelihood_lept.yaml': 3.7762380644679183e-07,
      'likelihood_rd_rds.yaml': 0.27864506193111893,
      'likelihood_lfu_fccc.yaml': 0.0005027179997831865,
      'likelihood_lfu_fcnc.yaml': 3.0607966063245655,
      'likelihood_bcpv.yaml': 0.013775072147421241,
      'likelihood_bqnunu.yaml': -0.119578242544371,
      'likelihood_lfv.yaml': 0.0,
      'likelihood_zlfv.yaml': 0.0,
      'likelihood_higgs.yaml': 2.176258307784451e-05,
      'global': 21.299153554766516}
```

While the global $\Delta \log L$ is provided by `log_likelihood_dict`, its value can also be directly returned using the `log_likelihood_global` method.

```
In: pp.log_likelihood_global()
```

```
Out: 21.299153554766516
```

Apart from $\Delta \log L$, it is also possible to compute the total $\chi^2_{\rm NP}$, defined by

$$\chi^2_{\rm NP} = -2\,\log L_{\rm NP}\,, \tag{3}$$

where $L_{\rm NP}$ is normalized such that it is 1 if the central values of the theory predictions are equal to the central values of the measurements for all observables. A dictionary containing the individual values of the total $\chi^2_{\rm NP}$ is returned by the `chi2_dict` method.

```
In: pp.chi2_dict()
```

```
Out: {'fast_likelihood_quarks.yaml': 160.14558316478963,
      'fast_likelihood_leptons.yaml': 23.57908813232271,
      'likelihood_ewpt.yaml': 35.3618189920579,
      'likelihood_eeww.yaml': 61.19130715429686,
      'likelihood_lept.yaml': 1.4486600571844703,
```

```
 'likelihood_rd_rds.yaml': 34.10567278343568,
 'likelihood_lfu_fccc.yaml': 49.155325606131306,
 'likelihood_lfu_fcnc.yaml': 24.16370720780219,
 'likelihood_bcpv.yaml': 5.140098429647292,
 'likelihood_bqnunu.yaml': 21.417983245315177,
 'likelihood_lfv.yaml': 8.998264557313096,
 'likelihood_zlfv.yaml': -0.0,
 'likelihood_higgs.yaml': 55.781752694208386,
 'global': 480.4892620245047}
```

These values are particularly useful for computing p-values from the total $\chi^2_{\mathrm{NP}}$ and the number of observations. The latter are returned by the `number_observations_dict` method of the `GlobalLikelihood` instance (which can be conveniently accessed using the `likelihood` attribute of the `GlobalLikelihoodPoint` instance).

```
In: pp.likelihood.number_observations_dict()
```

```
Out: {'fast_likelihood_quarks.yaml': 144,
 'fast_likelihood_leptons.yaml': 7,
 'likelihood_ewpt.yaml': 30,
 'likelihood_eeww.yaml': 48,
 'likelihood_lept.yaml': 2,
 'likelihood_rd_rds.yaml': 11,
 'likelihood_lfu_fccc.yaml': 63,
 'likelihood_lfu_fcnc.yaml': 21,
 'likelihood_bcpv.yaml': 6,
 'likelihood_bqnunu.yaml': 22,
 'likelihood_lfv.yaml': 41,
 'likelihood_zlfv.yaml': 7,
 'likelihood_higgs.yaml': 67,
 'global': 469}
```

Note that here an “observation” is defined as an individual measurement of an observable. Thus, the number of observations is always greater than or equal to the number of observables.

#### 3.2.4 Table of observables

`smelli` provides information on individual observables. In particular, the theoretical and experimental central values and uncertainties as well as the pull compared to the SM or the experimental data can be obtained. All this information is contained in an “observable table” that is returned in the form of a Pandas [60, 61] `DataFrame` object by the method `obstable`.

```
In: df = pp.obstable()
```

In a Jupyter notebook, a Pandas `DataFrame` is shown as a table.

In:
```
df
```

Out:

| | experiment | exp. unc. | theory | th. unc. | pull exp. | pull SM |
|---|---|---|---|---|---|---|
| **a_mu** | 0.00116592 | 6.31304e-10 | 0.00116592 | 4.25176e-10 | 3.49239 | -4.46085e-05 |
| **Rtaul(B->D*lnu)** | 0.296146 | 0.015608 | 0.244875 | 0 | 3.30606 | -0.389707 |
| **(<dR/dtheta>(ee->WW), 198.38, 0.8, 1.0)** | 6.535 | 0.236 | 7.236 | 0 | 2.97036 | 0.0112166 |
| **BR(W->taunu)** | 0.1138 | 0.0021 | 0.108417 | 0 | 2.56345 | -0.00503662 |
| **epsp/eps** | 0.00166382 | 0.000227703 | -3.12549e-05 | 0.000637111 | 2.50537 | 0.0147821 |
| **...** | ... | ... | ... | ... | ... | ... |
| **BR(tau->phie)** | 0 | 1.88467e-08 | 0 | 0 | 0 | 0 |
| **BR(tau->phimu)** | 0 | 5.10684e-08 | 0 | 0 | 0 | 0 |
| **BR(Z->emu)** | 0 | 2.33094e-07 | 0 | 0 | 0 | 0 |
| **BR(Z->etau)** | 0 | 2.59807e-06 | 0 | 0 | 0 | 0 |
| **BR(Z->mutau)** | 0 | 2.69574e-06 | 0 | 0 | 0 | 0 |

399 rows × 6 columns

The Pandas `DataFrame` is a convenient object for tabulated data and provides many useful features. E.g. one can sort the rows by the values of a given column,

In:
```
df.sort_values('pull SM', ascending=True)[:5]
```

Out:

| | experiment | exp. unc. | theory | th. unc. | pull exp. | pull SM |
|---|---|---|---|---|---|---|
| **(<dBR/dq2>(Bs->phimumu), 1.0, 6.0)** | 2.55342e-08 | 3.72621e-09 | 4.04247e-08 | 6.44267e-09 | 2.0007 | -3.24157 |
| **(<Rmue>(B0->K*ll), 1.1, 6.0)** | 0.681356 | 0.123108 | 0.746295 | 0 | 0.623038 | -2.4685 |
| **BR(Bs->mumu)** | 2.73001e-09 | 3.80964e-10 | 2.73442e-09 | 1.47033e-10 | 0.0108006 | -2.29374 |
| **(<dBR/dq2>(Bs->phimumu), 15.0, 19.0)** | 4.05106e-08 | 5.09449e-09 | 4.08896e-08 | 4.5361e-09 | 0.0555647 | -2.21418 |
| **(<dBR/dq2>(B0->K*mumu), 15.0, 19.0)** | 4.35409e-08 | 3.61869e-09 | 4.35383e-08 | 6.16124e-09 | 0.000370693 | -2.20919 |

or select a specific row by its name.

In:
```
df.loc[['Rtaul(B->D*lnu)']]
```

Out:

| | experiment | exp. unc. | theory | th. unc. | pull exp. | pull SM |
|---|---|---|---|---|---|---|
| **Rtaul(B->D*lnu)** | 0.296146 | 0.015608 | 0.244875 | 0 | 3.30606 | -0.389707 |

### 3.2.5 Plots

Given a likelihood function, one common task is to plot this function in a 2D plane. In order to simplify this, `smelli` provides a method to compute the plot data for all individual likelihoods. For demonstration, it is convenient to define a `GlobalLikelihood` instance for which the likelihood can be computed much faster than in the default case. This can be achieved by considering only a subset of observables, e.g. only EWPO and the Higgs signal strengths.

In:
```
gl_ewpt_higgs = GlobalLikelihood(include_likelihoods=[
    'likelihood_ewpt.yaml',
    'likelihood_higgs.yaml',
])
```

The next step is to define a function of the two plot parameters that returns a dictionary of Wilson coefficients. This function defines what is actually plotted. It can be a trivial function that takes two Wilson coefficients as arguments and just returns them, but it can also be a complicated function of two NP model parameters that returns a large set of Wilson coefficients depending on these two parameters. As an example, we will reproduce figure 2 of [62] and plot the likelihood in the space of the $S$ and $T$ parameters. They are proportional to the SMEFT Wilson coefficients $C_{\phi WB}$ and $C_{\phi D}$, and their relations are given by

$$C_{\phi WB} = \frac{g_L\, g_Y}{16\,\pi\, v^2}\, S\,, \qquad C_{\phi D} = -\frac{g_L^2\, g_Y^2}{2\,\pi\,(g_L^2 + g_Y^2)\, v^2}\, T\,. \tag{4}$$

Consequently, plugging in the SM parameters, the function that takes $S$ and $T$ as arguments and returns a dictionary of Wilson coefficients can be defined as follows.

In:
```
def wc_fct(S, T):
    return {
        'phiWB': S * 7.643950529889027e-08,
        'phiD': -T * 2.5793722852276787e-07,
    }
```

This function can now be used as the first argument of the `plot_data_2d` method of the `GlobalLikelihood` instance. The second argument is the scale at which the Wilson coefficients are defined, followed by the minimum and maximum values for the x- and y-axis. In the function call below, also two optional arguments are given: the number of `steps` in each direction (`steps` = 10 results in plot data computed on a $10 \times 10$ grid), and the number of CPU threads to be used for the computation.

In:
```
plot_data = gl_ewpt_higgs.plot_data_2d(
    wc_fct,
    91.1876,
    -0.2, 0.2, -0.1, 0.3,
    steps=10,
    threads=8,
)
```

The plot_data_2d method returns a dictionary with the names of the individual likelihoods as keys and values that are again dictionaries. The keys in these latter dictionaries are x, y, and z and the values are arrays. Here, x and y correspond to the coordinates in the 2D plane and z to the values of $\Delta\chi^2 = -2\,\Delta\log L$ at these coordinates. The dictionaries with keys x, y, and z are constructed in such a way that they can be directly fed to the contour plotting function of the flavio package. The relevant submodules for plotting have to be imported from flavio and matplotlib [63] (on which the flavio plotting functions are based on).

In:
```
import flavio.plots as fpl
import matplotlib.pyplot as plt
```

In order to plot $\Delta\chi^2$ contours corresponding to a given pull in units of $\sigma$, the contour levels can be defined using the flavio function delta_chi2, which takes the number of $\sigma$ and the number of degrees of freedom as arguments.

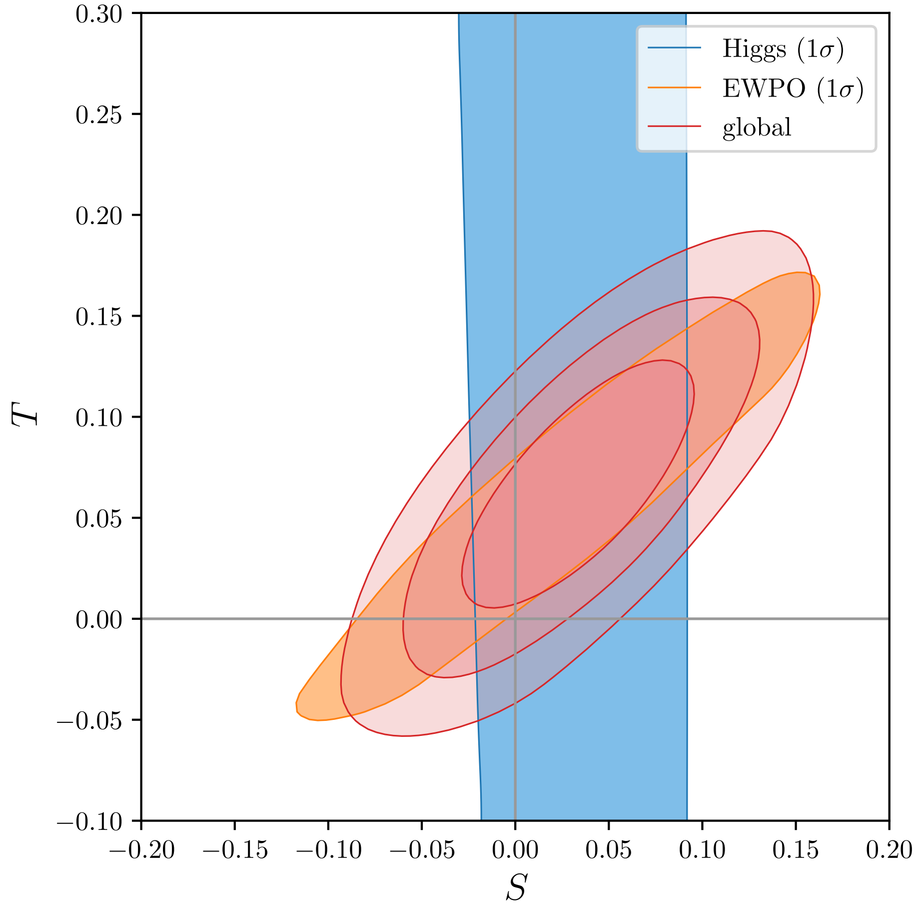
In:
```
levels_1sig = [fpl.delta_chi2(1, dof=2)]
levels_123sig = [fpl.delta_chi2(n_sigma, dof=2) for n_sigma in (1,2,3)]
```

The data can now be plotted. The function fpl.contour is called three times, once for each of the three different likelihoods: Higgs physics, EWPO, and their combination. Furthermore, horizontal and vertical axes as well as labels are added. A value larger than one for the argument interpolation_factor of fpl.contour makes the contours appear smooth. However, if the plot data has been computed on a small grid, interpolation_factor can obscure the fact that the data might be insufficient for a reasonable plot. In fact, for more reasonable plots, the number of steps should be increased to at least 20 (but this of course also increases the computing time). From the data computed above, the plot is then generated by the following code.

In:
```
plt.figure(figsize=(5,5))
fpl.contour(**plot_data['likelihood_higgs.yaml'], levels=levels_1sig,
            label=r"Higgs ($1\sigma$)", interpolation_factor=9,
            color='C0')
fpl.contour(**plot_data['likelihood_ewpt.yaml'], levels=levels_1sig,
            label=r"EWPO ($1\sigma$)", interpolation_factor=9,
            color='C1')
fpl.contour(**plot_data['global'], levels=levels_123sig,
            label=r"global", interpolation_factor=9,
            color='C3')
plt.axhline(c='0.6', linewidth=1)
plt.axvline(c='0.6', linewidth=1)
plt.xlabel(r'$S$')
plt.ylabel(r'$T$')
plt.legend()
plt.show()
```



PoS(TOOLS2020)035






## 4. Conclusions

Models that explain experimental deviations from the SM in certain observables generically predict also effects in other observables. This is e.g. the case for most models that explain the $B$ anomalies. Consequently, to test such models, one has to consider a *global* likelihood constructed from as many observables as possible.

This article shows how to use the python package `smelli`, which implements a global SMEFT likelihood function. It can be used to either test models, or to interpret data model-independently in the WET and the SMEFT. To date, 399 flavor and other precision observables are included in the likelihood.

The full *global* likelihood is work in progress. Since `smelli` is completely open source, you are welcome to join us on https://github.com/smelli/smelli and to participate in the effort to make `smelli` truly global.

### Acknowledgments

I thank Jason Aebischer, Jacky Kumar, and David M. Straub for the collaboration `smelli` is based on and Matthew Kirk for contributing to `smelli` on GitHub.

## References


[1] LHCb collaboration, R. Aaij et al., *Measurement of CP-Averaged Observables in the $B^0 \to K^{*0}\mu^+\mu^-$ Decay*, Phys. Rev. Lett. **125** (2020) 011802, [2003.04831].

[2] CMS collaboration, V. Khachatryan et al., *Angular analysis of the decay $B^0 \to K^{*0}\mu^+\mu^-$ from pp collisions at $\sqrt{s} = 8$ TeV*, Phys. Lett. **B753** (2016) 424–448, [1507.08126].

[3] ATLAS collaboration, *Angular analysis of $B_d^0 \to K^*\mu^+\mu^-$ decays in $pp$ collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector*, 2017.

[4] CMS collaboration, A. M. Sirunyan et al., *Measurement of angular parameters from the decay $\mathrm{B}^0 \to \mathrm{K}^{*0}\mu^+\mu^-$ in proton-proton collisions at $\sqrt{s} = 8$ TeV*, Phys. Lett. **B781** (2018) 517–541, [1710.02846].

[5] LHCb collaboration, D. Gerick, *10th workshop on "Implications of LHCb measurements and future prospects", 28-30 October, 2020*, https://indico.cern.ch/event/857473/contributions/4060371/attachments/2133689/3593528/Implications2020_LHCbRareDecays_DavidGerick.pdf.

[6] LHCb collaboration, R. Aaij et al., *Differential branching fractions and isospin asymmetries of $B \to K^{(*)}\mu^+\mu^-$ decays*, JHEP **06** (2014) 133, [1403.8044].

[7] LHCb collaboration, R. Aaij et al., *Angular analysis and differential branching fraction of the decay $B_s^0 \to \phi\mu^+\mu^-$*, JHEP **09** (2015) 179, [1506.08777].

[8] LHCb collaboration, R. Aaij et al., *Measurements of the S-wave fraction in $B^0 \to K^+\pi^-\mu^+\mu^-$ decays and the $B^0 \to K^*(892)^0\mu^+\mu^-$ differential branching fraction*, JHEP **11** (2016) 047, [1606.04731].

[9] LHCb collaboration, R. Aaij et al., *Test of lepton universality with $B^0 \to K^{*0}\ell^+\ell^-$ decays*, JHEP **08** (2017) 055, [1705.05802].

[10] LHCb collaboration, R. Aaij et al., *Search for lepton-universality violation in $B^+ \to K^+\ell^+\ell^-$ decays*, Phys. Rev. Lett. **122** (2019) 191801, [1903.09252].

[11] Belle collaboration, A. Abdesselam et al., *Test of lepton flavor universality in $B \to K^*\ell^+\ell^-$ decays at Belle*, 1904.02440.

[12] Belle collaboration, A. Abdesselam et al., *Test of lepton flavor universality in $B \to K\ell^+\ell^-$ decays*, 1908.01848.

[13] CMS collaboration, S. Chatrchyan et al., *Measurement of the $B_s^0 \to \mu^+\mu^-$ Branching Fraction and Search for $B^0 \to \mu^+\mu^-$ with the CMS Experiment*, Phys. Rev. Lett. **111** (2013) 101804, [1307.5025].

[14] CMS, LHCb collaboration, V. Khachatryan et al., *Observation of the rare $B_s^0 \to \mu^+\mu^-$ decay from the combined analysis of CMS and LHCb data*, Nature **522** (2015) 68–72, [1411.4413].

[15] LHCb collaboration, R. Aaij et al., *Measurement of the $B_s^0 \to \mu^+\mu^-$ branching fraction and effective lifetime and search for $B^0 \to \mu^+\mu^-$ decays*, Phys. Rev. Lett. **118** (2017) 191801, [1703.05747].

[16] ATLAS collaboration, M. Aaboud et al., *Study of the rare decays of $B_s^0$ and $B^0$ mesons into muon pairs using data collected during 2015 and 2016 with the ATLAS detector*, JHEP **04** (2019) 098, [1812.03017].

[17] LHCb collaboration, *Combination of the ATLAS, CMS and LHCb results on the $B^0_{(s)} \to \mu^+\mu^-$ decays*, 2020.

[18] BaBar collaboration, J. P. Lees et al., *Evidence for an excess of $\bar{B} \to D^{(*)}\tau^-\bar{\nu}_\tau$ decays*, Phys. Rev. Lett. **109** (2012) 101802, [1205.5442].

[19] BaBar collaboration, J. P. Lees et al., *Measurement of an Excess of $\bar{B} \to D^{(*)}\tau^-\bar{\nu}_\tau$ Decays and Implications for Charged Higgs Bosons*, Phys. Rev. **D88** (2013) 072012, [1303.0571].

[20] Belle collaboration, M. Huschle et al., *Measurement of the branching ratio of $\bar{B} \to D^{(*)}\tau^-\bar{\nu}_\tau$ relative to $\bar{B} \to D^{(*)}\ell^-\bar{\nu}_\ell$ decays with hadronic tagging at Belle*, Phys. Rev. **D92** (2015) 072014, [1507.03233].

[21] Belle collaboration, Y. Sato et al., *Measurement of the branching ratio of $\bar{B}^0 \to D^{*+}\tau^-\bar{\nu}_\tau$ relative to $\bar{B}^0 \to D^{*+}\ell^-\bar{\nu}_\ell$ decays with a semileptonic tagging method*, Phys. Rev. **D94** (2016) 072007, [1607.07923].

[22] Belle collaboration, S. Hirose et al., *Measurement of the $\tau$ lepton polarization and $R(D^*)$ in the decay $\bar{B} \to D^*\tau^-\bar{\nu}_\tau$*, Phys. Rev. Lett. **118** (2017) 211801, [1612.00529].

[23] LHCb collaboration, R. Aaij et al., *Measurement of the ratio of branching fractions $\mathcal{B}(\bar{B}^0 \to D^{*+}\tau^-\bar{\nu}_\tau)/\mathcal{B}(\bar{B}^0 \to D^{*+}\mu^-\bar{\nu}_\mu)$*, Phys. Rev. Lett. **115** (2015) 111803, [1506.08614].

[24] LHCb collaboration, R. Aaij et al., *Measurement of the ratio of the $B^0 \to D^{*-}\tau^+\nu_\tau$ and $B^0 \to D^{*-}\mu^+\nu_\mu$ branching fractions using three-prong $\tau$-lepton decays*, Phys. Rev. Lett. **120** (2018) 171802, [1708.08856].

[25] Belle collaboration, A. Abdesselam et al., *Measurement of $\mathcal{R}(D)$ and $\mathcal{R}(D^*)$ with a semileptonic tagging method*, 1904.08794.

[26] M. Bordone, G. Isidori and A. Pattori, *On the Standard Model predictions for $R_K$ and $R_{K^*}$*, Eur. Phys. J. **C76** (2016) 440, [1605.07633].

[27] M. Beneke, C. Bobeth and R. Szafron, *Enhanced electromagnetic correction to the rare B-meson decay $B_{s,d} \to \mu^+\mu^-$*, Phys. Rev. Lett. **120** (2018) 011801, [1708.09152].

[28] M. Bordone, M. Jung and D. van Dyk, *Theory determination of $\bar{B} \to D^{(*)}\ell^-\bar{\nu}$ form factors at $O(1/m_c^2)$*, Eur. Phys. J. **C80** (2020) 74, [1908.09398].

[29] J. Aebischer, W. Altmannshofer, D. Guadagnoli, M. Reboud, P. Stangl and D. M. Straub, *B-decay discrepancies after Moriond 2019*, *Eur. Phys. J.* **C80** (2020) 252, [1903.10434].

[30] M. Algueró, B. Capdevila, A. Crivellin, S. Descotes-Genon, P. Masjuan, J. Matias et al., *Emerging patterns of New Physics with and without Lepton Flavour Universal contributions*, *Eur. Phys. J.* **C79** (2019) 714, [1903.09578].

[31] A. Datta, J. Kumar and D. London, *The B anomalies and new physics in* $b \to s e^+ e^-$, *Phys. Lett.* **B797** (2019) 134858, [1903.10086].

[32] K. Kowalska, D. Kumar and E. M. Sessolo, *Implications for new physics in* $b \to s\mu\mu$ *transitions after recent measurements by Belle and LHCb*, *Eur. Phys. J.* **C79** (2019) 840, [1903.10932].

[33] T. Hurth, F. Mahmoudi and S. Neshatpour, *Implications of the new LHCb angular analysis of* $B \to K^* \mu^+ \mu^-$ *: Hadronic effects or new physics?*, *Phys. Rev.* **D102** (2020) 055001, [2006.04213].

[34] M. Ciuchini, M. Fedele, E. Franco, A. Paul, L. Silvestrini and M. Valli, *Lessons from the* $B^{0,+} \to K^{*0,+} \mu^+ \mu^-$ *angular analyses*, 2011.01212.

[35] A. J. Buras, J. Girrbach-Noe, C. Niehoff and D. M. Straub, $B \to K^{(*)} \nu\bar{\nu}$ *decays in the Standard Model and beyond*, *JHEP* **02** (2015) 184, [1409.4557].

[36] F. Feruglio, P. Paradisi and A. Pattori, *On the Importance of Electroweak Corrections for B Anomalies*, *JHEP* **09** (2017) 061, [1705.00929].

[37] C. Bobeth and U. Haisch, *New Physics in* $\Gamma_{12}^s$*:* $(\bar{s}b)(\bar{\tau}\tau)$ *Operators*, *Acta Phys. Polon.* **B44** (2013) 127–176, [1109.1826].

[38] A. Crivellin, C. Greub, D. Müller and F. Saturnino, *Importance of Loop Effects in Explaining the Accumulated Evidence for New Physics in B Decays with a Vector Leptoquark*, *Phys. Rev. Lett.* **122** (2019) 011805, [1807.02068].

[39] W. Buchmuller and D. Wyler, *Effective Lagrangian Analysis of New Interactions and Flavor Conservation*, *Nucl. Phys.* **B268** (1986) 621–653.

[40] B. Grzadkowski, M. Iskrzynski, M. Misiak and J. Rosiek, *Dimension-Six Terms in the Standard Model Lagrangian*, *JHEP* **10** (2010) 085, [1008.4884].

[41] A. Efrati, A. Falkowski and Y. Soreq, *Electroweak constraints on flavorful effective theories*, *JHEP* **07** (2015) 018, [1503.07872].

[42] A. Falkowski, M. González-Alonso and K. Mimouni, *Compilation of low-energy constraints on 4-fermion operators in the SMEFT*, *JHEP* **08** (2017) 123, [1706.03783].

[43] S. Alioli, V. Cirigliano, W. Dekens, J. de Vries and E. Mereghetti, *Right-handed charged currents in the era of the Large Hadron Collider*, *JHEP* **05** (2017) 086, [1703.04751].

[44] M. González-Alonso, O. Naviliat-Cuncic and N. Severijns, *New physics searches in nuclear and neutron $\beta$ decay*, Prog. Part. Nucl. Phys. **104** (2019) 165–223, [1803.08732].

[45] A. Falkowski, M. Gonzalez-Alonso, A. Greljo and D. Marzocca, *Global constraints on anomalous triple gauge couplings in effective field theory approach*, Phys. Rev. Lett. **116** (2016) 011801, [1508.00581].

[46] C. Bobeth and U. Haisch, *Anomalous triple gauge couplings from B-meson and kaon observables*, JHEP **09** (2015) 018, [1503.04829].

[47] A. Falkowski and K. Mimouni, *Model independent constraints on four-lepton operators*, JHEP **02** (2016) 086, [1511.07434].

[48] A. Biekoetter, T. Corbett and T. Plehn, *The Gauge-Higgs Legacy of the LHC Run II*, SciPost Phys. **6** (2019) 064, [1812.07587].

[49] N. P. Hartland, F. Maltoni, E. R. Nocera, J. Rojo, E. Slade, E. Vryonidou et al., *A Monte Carlo global analysis of the Standard Model Effective Field Theory: the top quark sector*, JHEP **04** (2019) 100, [1901.05965].

[50] I. Brivio, S. Bruggisser, F. Maltoni, R. Moutafis, T. Plehn, E. Vryonidou et al., *O new physics, where art thou? A global search in the top sector*, JHEP **02** (2020) 131, [1910.03606].

[51] S. Bißmann, J. Erdmann, C. Grunwald, G. Hiller and K. Kröninger, *Constraining top-quark couplings combining top-quark and **B** decay observables*, Eur. Phys. J. **C80** (2020) 136, [1909.13632].

[52] S. van Beek, E. R. Nocera, J. Rojo and E. Slade, *Constraining the SMEFT with Bayesian reweighting*, SciPost Phys. **7** (2019) 070, [1906.05296].

[53] G. Durieux, A. Irles, V. Miralles, A. Peñuelas, R. Pöschl, M. Perelló et al., *The electro-weak couplings of the top and bottom quarks – global fit and future prospects*, 1907.10619.

[54] A. Falkowski, M. González-Alonso and O. Naviliat-Cuncic, *Comprehensive analysis of beta decays within and beyond the Standard Model*, 2010.13797.

[55] J. Ellis, M. Madigan, K. Mimasu, V. Sanz and T. You, *Top, Higgs, Diboson and Electroweak Fit to the Standard Model Effective Field Theory*, 2012.02779.

[56] J. Aebischer, J. Kumar, P. Stangl and D. M. Straub, *A Global Likelihood for Precision Constraints and Flavour Anomalies*, Eur. Phys. J. **C79** (2019) 509, [1810.07698].

[57] D. M. Straub, *flavio: a Python package for flavour and precision phenomenology in the Standard Model and beyond*, 1810.08132.

[58] J. Aebischer et al., *WCxf: an exchange format for Wilson coefficients beyond the Standard Model*, Comput. Phys. Commun. **232** (2018) 71–83, [1712.05298].

[59] J. Aebischer, J. Kumar and D. M. Straub, *Wilson: a Python package for the running and matching of Wilson coefficients above and below the electroweak scale*, *Eur. Phys. J.* **C78** (2018) 1026, [1804.05033].

[60] The pandas development team collaboration, *pandas-dev/pandas: Pandas*, Feb., 2020. 10.5281/zenodo.3509134.

[61] Wes McKinney, *Data Structures for Statistical Computing in Python*, in *Proceedings of the 9th Python in Science Conference* (Stéfan van der Walt and Jarrod Millman, eds.), pp. 56 – 61, 2010, DOI.

[62] A. Falkowski and D. Straub, *Flavourful SMEFT likelihood for Higgs and electroweak data*, *JHEP* **04** (2020) 066, [1911.07866].

[63] J. D. Hunter, *Matplotlib: A 2d graphics environment*, *Computing in Science & Engineering* **9** (2007) 90–95.