Lilith-2 tutorial

Sabine Kraml

in collaboration with

Tran Quang Loc, Dao Thi Nhung, and Le Duc Ninh

arXiv:1908.03952 — SciPost Phys. 7, 052 (2019) — GitHub

LPSC GRENOBLE | MODANE

ICISE QUY NHON VIETNAM

Lilith : Light Likelihood fit for the Higgs

- Lilith is a **light and easy-to-use Python tool** for constraining new physics from signal strength measurements of the 125 GeV Higgs boson.
- The **Higgs likelihood** is evaluated from the user input, given in XML format **in terms of reduced couplings or signal strengths**.
- Based on experimental results from ATLAS and CMS stored in an **easily extensible XML database**.

Features of Lilith-2 :

- Use of variable Gaussian and generalized Poisson likelihoods for a **better treatment of asymmetric uncertainties**.
- Use of **N-dim correlation matrices** for ordinary and variable Gaussian likelihoods.
- Database 19.09 contains the published ATLAS and CMS Run 2 results for 36/fb as of Sep 2019.
- Update for full Run-2 luminosity results in progress.

https://github.com/sabinekraml/Lilith-2

In this tutorial, we explain what's inside and how to use the code

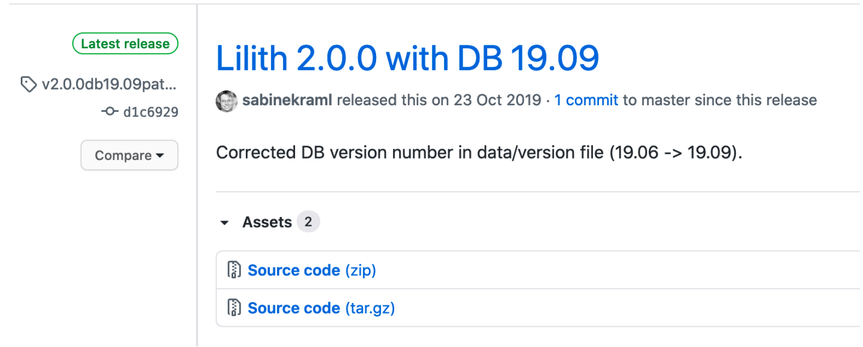
Download & installation

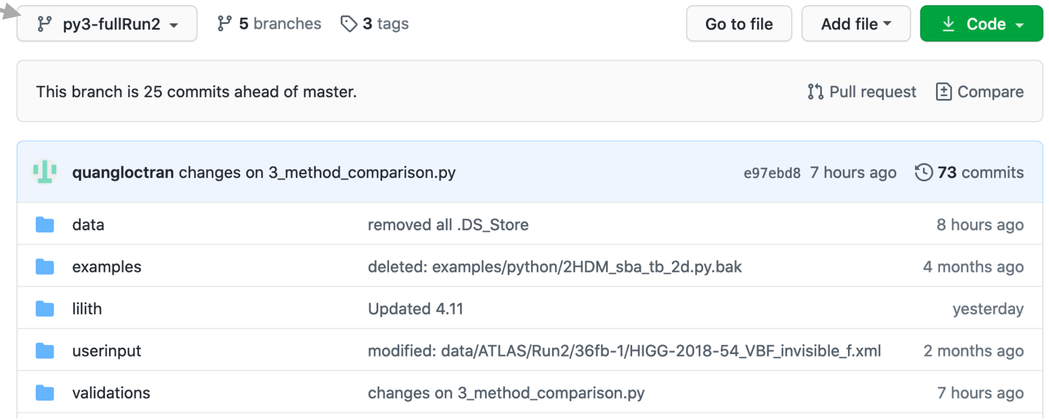
- v2.0 release (Python 2):

  https://github.com/sabinekraml/Lilith-2/releases

- Python 3 version (with preliminary DB updates): py3-fullRun2 branch *beware, work in progress*

  https://github.com/sabinekraml/Lilith-2/tree/py3-fullRun2





Requirements: **SciPy and NumPy**; the example codes doing a likelihood profile analysis require **iminuit**.

Try it out

```
> ./run_lilith.py -h
```

```
Usage of Lilith:
----------------
./run_lilith model_input_xml [experimental_input_list]
Example: ./run_lilith userinput/example_mu.xml [data/latest.list]

[options]
 -h or --help : dump this help
 -v or --verbose : print useful information on the screen
 -t or --timer : print execution time of various tasks
 -s or --silent : nothing is printed on the screen
 -m or --mu= : output signal strengths in XML format
 -c or --couplings= : return missing couplings instead of the likelihood
 -r file or --results=file : prints the output in a file
```

Try it out

```
> ./run_lilith.py userinput/example_mu.xml data/latestRun2.list
```

```
                        <><><><><><><><><><><><><><><><><><><>
                            Lilith version 2.0
                        database version 20.11dev
                        <><><><><><><><><><><><><><><><><><><>

    . User input: userinput/example_mu.xml

    . Experimental input: data/latestRun2.list
                          (21 files, Ndof = 53)

    -2log(likelihood) = 48.398237
```

Try it out

```
> ./run_lilith.py userinput/example_couplings.xml data/latestRun2.list
```

```
                          <><><><><><><><><><><><><><><>
                              Lilith version 2.0
                          database version 20.11dev
                          <><><><><><><><><><><><><><><>

        . User input: userinput/example_couplings.xml

        . Experimental input: data/latestRun2.list
                              (21 files, Ndof = 53)

        -2log(likelihood) = 68.265759
```

See commented examples in **`userinput`** folder

User input files

Signal strength input

```
<?xml version="1.0"?>

<lilithinput>
  <!-- signal strengths in theory space, like mu(gg -> H -> ZZ) -->
  <signalstrengths part="h">
    <mass>125</mass>

    <mu prod="ggH" decay="gammagamma">1.0</mu>
    <mu prod="ggH" decay="VV">1.0</mu>
    <mu prod="ggH" decay="bb">1.0</mu>
    <mu prod="ggH" decay="tautau">1.0</mu>
    <mu prod="ggH" decay="mumu">1.0</mu>

    <mu prod="VBF" decay="gammagamma">1.0</mu>
    <mu prod="VBF" decay="VV">1.0</mu>
    <mu prod="VBF" decay="bb">1.0</mu>
    <mu prod="VBF" decay="tautau">1.0</mu>
    <mu prod="VBF" decay="mumu">1.0</mu>

    [……]

    <redxsBR prod="ZH" decay="invisible">0.0</redxsBR>
    <redxsBR prod="VBF" decay="invisible">0.0</redxsBR>
  </signalstrengths>
</lilithinput>
```

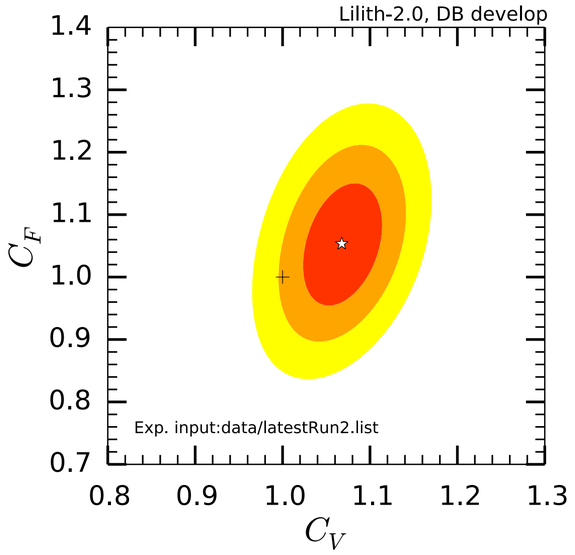
Reduced couplings input

```
<?xml version="1.0"?>

<lilithinput>
  <!-- reduced couplings to the Higgs boson as input —>
  <reducedcouplings>
    <mass>125</mass>

    <C to="tt">1.0</C> <!-- top quarks -->
    <C to="cc">2.0</C> <!-- charm quarks -->
    <C to="bb">1.0</C> <!-- bottom quarks -->
    <C to="tautau">1.0</C> <!-- tau leptons -->
    <C to="mumu">1.0</C> <!-- muon leptons -->

    <C to="ZZ">1.3</C>

    <C to="gammagamma">1.0</C>
    <C to="Zgamma">1.0</C>

   <precision>BEST-QCD</precision>

   <extraBR>
      <BR to="invisible">0.0</BR>
      <BR to="undetected">0.0</BR>
    </extraBR>
  </reducedcouplings>
</lilithinput>
```

Try it out

```
> python ./examples/python/CVCF_2d.py
```

```
***** reading parameters *****
***** scan initialization *****
***** running scan *****
***** scan finalized *****
minimum at CV, CF, -2logL_min =  1.06767676768 1.05353535354 43.147006893
***** plotting *****
results are stored in /Users/kraml/....
***** done *****
```



Now let's walk through the database
and the code examples …

Detailed, manual-style explanations:

Bernon, Dumont, arXiv:1502.04138

Kraml, Loc, Nhung, Ninh, arXiv:1908.03952