Global fits in "Beyond the Standard Model" physics

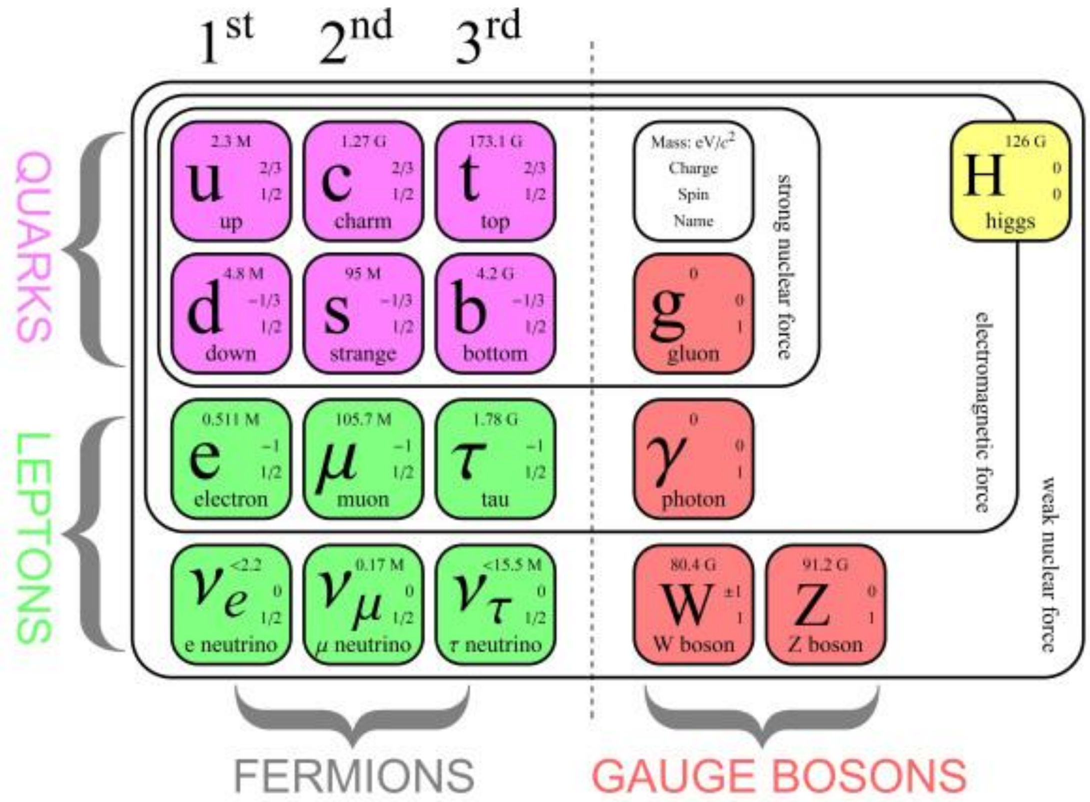
Ben Farmer

Astrophysics group, Imperial College London

Overview

- What is “Beyond the Standard Model” physics? What is a global fit?
- GAMBIT - A software framework for performing global fits
- Some statistical difficulties/fun things
    - An example; “meta-analysis” of LHC searches for SUSY particles
        - Issues computing p-values
    - Bayesian naturalness

The Standard Model of particle physics



The Standard Model of particle physics

$$
\begin{aligned}
\mathrm{L} = &-\frac{1}{4} W_{\mu\nu} W^{\mu\nu} - \frac{1}{4} B_{\mu\nu} B^{\mu\nu} - \frac{1}{4} G_{\mu\nu} G^{\mu\nu} \\
&+ \overline{\psi}_j \gamma^\mu (i\delta_\mu - g\tau_j \cdot W_\mu - g' Y_j B_\mu - g_s \mathrm{T}_j \cdot \mathrm{G}_\mu) \psi_j \\
&+ |\mathrm{D}_\mu \phi|^2 + \mu^2 |\phi|^2 - \lambda |\phi|^4 \\
&- (y_j \overline{\psi}_{jL} \phi \psi_{jR} + y'_j \overline{\psi}_{jL} \phi_c \psi_{jR} + \text{conjugate})
\end{aligned}
$$

The Standard Model of particle physics

$$
\begin{aligned}
L = &-\frac{1}{4} W_{\mu\nu} W^{\mu\nu} - \frac{1}{4} B_{\mu\nu} B^{\mu\nu} - \frac{1}{4} G_{\mu\nu} G^{\mu\nu} \\
&+ \overline{\psi}_j \gamma^\mu \left(i\delta_\mu - g\tau_j \cdot W_\mu - g' Y_j B_\mu - g_s \mathrm{T}_j \cdot \mathrm{G}_\mu \right) \psi_j \\
&+ |\mathrm{D}_\mu \phi|^2 + \mu^2 |\phi|^2 - \lambda |\phi|^4 \\
&- \left(y_j \overline{\psi}_{jL} \phi \psi_{jR} + y'_j \overline{\psi}_{jL} \phi_c \psi_{jR} + \text{conjugate} \right)
\end{aligned}
$$

Parameters:

9 Yukawa couplings (-> quark + lepton masses)
3 CKM angles
1 CKM phase
3 gauge couplings
1 QCD theta angle
2 Higgs parameters (-> mass + VEV)
= 19 free parameters

“Beyond the Standard Model” physics

Goal of the field: look for “cracks” in the Standard Model of particle physics, and eventually discover new fundamental particles and/or forces.

“Beyond the Standard Model” physics

Goal of the field: look for “cracks” in the Standard Model of particle physics, and eventually discover new fundamental particles and/or forces.

Some cracks already?

- What is dark matter?
- What is dark energy?
- What is the right mechanism for neutrino mass generation?
- How do we generate the observed matter/anti-matter asymmetry?

“Beyond the Standard Model” physics

Goal of the field: look for “cracks” in the Standard Model of particle physics, and eventually discover new fundamental particles and/or forces.

Some cracks already?

- What is dark matter?
- What is dark energy?
- What is the right mechanism for neutrino mass generation
- How do we generate the observed matter/anti-matter asymmetry?

...and some mysteries that might be answered by extensions to the Standard Model

- Why three generations? And why the observed spectrum of particle masses?
- Why is the electroweak scale so “low” (vulnerable to arbitrarily large quantum corrections; is there a symmetry to protect from these effects?) (Hierarchy problem -> SUSY?)
- Why is the QCD theta angle ~0? (Strong CP problem -> Axions?)
- Why do the SM gauge couplings seem to *almost* unify at high scales? (Grand Unification?)
- How does gravity fit into the picture? (Quantum gravity)

What is a global fit?

- A statistical analysis of a single “underlying” physical theory, using data from many different kinds of experiments.
- Typically based on the computation of a joint (or “global”) likelihood function.
- Also global parameter optimisation (though really we are just as interested in general exploration)

$$\mathrm{Pr}(\text{All data} \,|\, \text{theory}) = \prod_{i=1}^{N} \mathrm{Pr}(\text{data}_i \,|\, \text{theory})$$

$$f_{\text{joint}}(\vec{x} \,|\, \vec{\theta}, M) = \prod_{i=1}^{N} f_i(x_i \,|\, \vec{\theta}, M)$$

$$L(\vec{x} \,|\, \vec{\theta}, M) = \prod_{i=1}^{N} L_i(x_i \,|\, \vec{\theta}, M)$$

What is a global fit?

- A statistical analysis of a single “underlying” physical theory, using data from many different kinds of experiments.
- Typically based on the computation of a joint (or “global”) likelihood function.
- Also global parameter optimisation (though really we are just as interested in general exploration)

$$\mathrm{Pr}(\mathrm{All\ data}\,|\,\mathrm{theory}) = \prod_{i=1}^{N} \mathrm{Pr}(\mathrm{data}_i\,|\,\mathrm{theory})$$

$$f_{\mathrm{joint}}(\vec{x}\,|\,\vec{\theta}, M) = \prod_{i=1}^{N} f_i(x_i\,|\,\vec{\theta}, M)$$

$$L(\vec{x}\,|\,\vec{\theta}, M) = \prod_{i=1}^{N} L_i(x_i\,|\,\vec{\theta}, M)$$

What is a global fit?

- A statistical analysis of a single “underlying” physical theory, using data from many different kinds of experiments.
- Typically based on the computation of a joint (or “global”) likelihood function.
- Also global parameter optimisation (though really we are just as interested in general exploration)

$$\mathrm{Pr}(\mathrm{All\ data} \,|\, \mathrm{theory}) = \prod_{i=1}^{N} \mathrm{Pr}(\mathrm{data}_i \,|\, \mathrm{theory})$$

What is a global fit?

- A statistical analysis of a single “underlying” physical theory, using data from many different kinds of experiments.
- Typically based on the computation of a joint (or “global”) likelihood function.
- Also global parameter optimisation (though really we are just as interested in general exploration)

$$\mathrm{Pr}(\text{All data} \,|\, \text{theory}) = \prod_{i=1}^{N} \mathrm{Pr}(\text{data}_i \,|\, \text{theory})$$

$$f_{\text{joint}}(\vec{x} \,|\, \vec{\theta}, M) = \prod_{i=1}^{N} f_i(x_i \,|\, \vec{\theta}, M)$$

$$L(\vec{x} \,|\, \vec{\theta}, M) = \prod_{i=1}^{N} L_i(x_i \,|\, \vec{\theta}, M)$$

A typical BSM joint likelihood

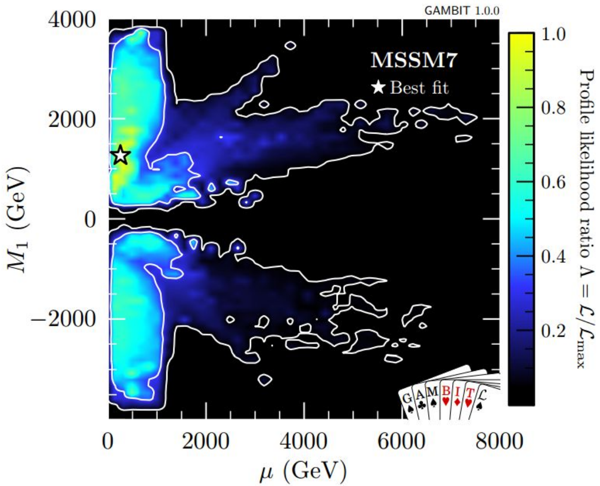
$$L_{\text{joint}} = L_{\text{LHC}} \times L_{\text{DM}} \times L_{\text{astro}} \times \ldots$$

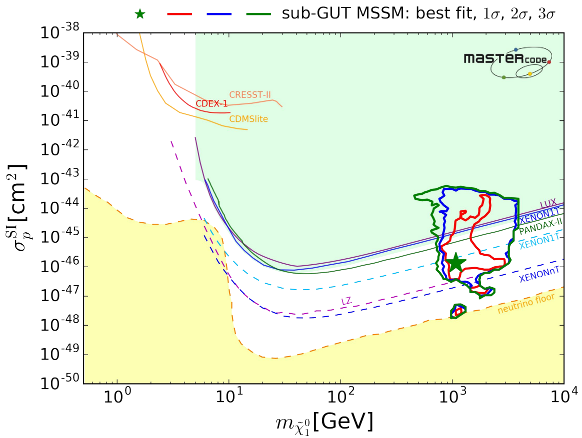
Likelihood term	
LHC sparticle searches	$B \to X_s\gamma$
LHC Higgs	a_μ
LEP Higgs	W mass
ALEPH selectron	Relic density
ALEPH smuon	PICO-2L
ALEPH stau	PICO-60 F
L3 selectron	SIMPLE 2014
L3 smuon	LUX 2015
L3 stau	LUX 2016
L3 neutralino leptonic	PandaX 2016
L3 chargino leptonic	SuperCDMS 2014
OPAL chargino hadronic	XENON100 2012
OPAL chargino semi-leptonic	IceCube 79-string
OPAL chargino leptonic	γ rays (Fermi-LAT dwarfs)
OPAL neutralino hadronic	ρ_0
$B_{(s)} \to \mu^+\mu^-$	σ_s and σ_l
Tree-level B and D decays	$\alpha_s(m_Z)(\overline{MS})$
$B^0 \to K^{*0}\mu^+\mu^-$	Top quark mass

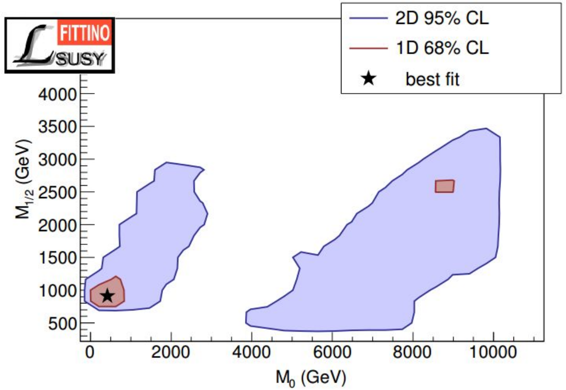
“A global fit of the MSSM with GAMBIT” arxiv: 1705.07917

A typical BSM joint likelihood

$$L_{\text{joint}} = L_{\text{LHC}} \times L_{\text{DM}} \times L_{\text{astro}} \times \ldots$$







GAMBIT arxiv:1705.07917, MasterCode arXiv:1711.00458, Fittino arxiv:1508.05951

If we can compute the global likelihood function, we can obtain either frequentist or Bayesian constraints on the underlying model parameters:

If we can compute the global likelihood function, we can obtain either frequentist or Bayesian constraints on the underlying model parameters:

Frequentist:

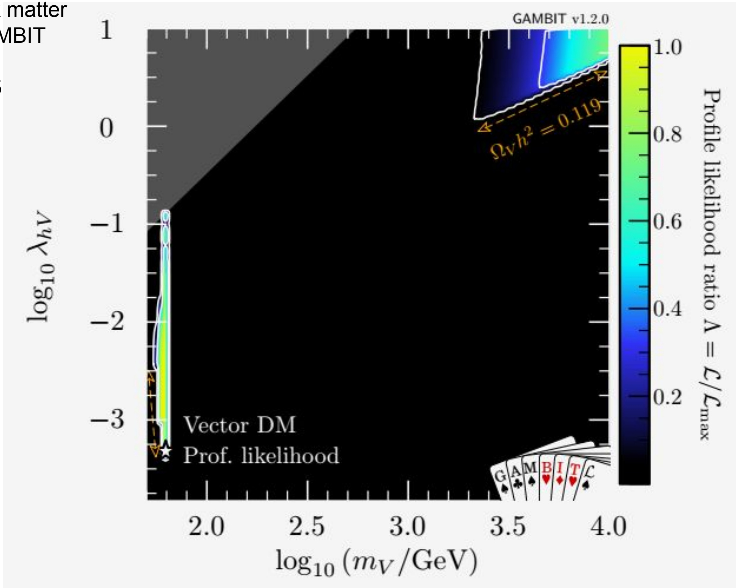
L -> test statistic (likelihood ratio)
-> asymptotic theory (Wilks' theorem)
-> confidence regions

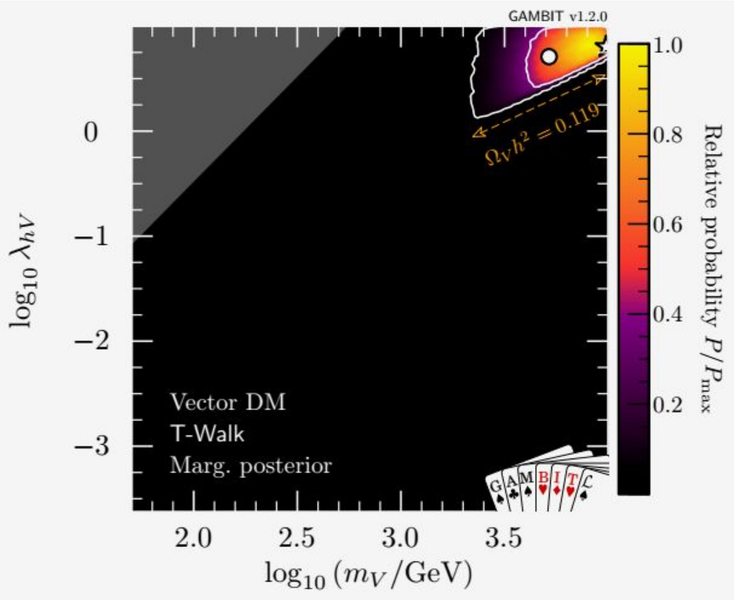
Bayesian:

posterior = L * prior / Z
-> credible regions

Global analyses of Higgs portal singlet dark matter models using GAMBIT

arxiv: 1808.10465





To obtain accurate “likelihood maps”:

Need to

- Evaluate component likelihood functions
- ...at many (typically millions+) of points in each parameter space
- ...in a smart way (sampling algorithms)

This is computationally expensive, and also requires a lot of a “book-keeping”.

Why?

There are *many* possible ways to extend the Standard Model.

Want to systematically explore this model space, and find out which models are already excluded by data! And if they aren't, determine what sort of experiments would be needed to find or exclude them.

This is a complicated task due to the large variety of experiments whose results could be altered by changes to the Standard Model.

GAMBIT

The Global And Modular BSM Inference Tool

An international community with 40+ collaborators (10 experiments, 14 major theory codes)
A new software framework for global fits developed over the past six years
First public code release in May 2017, arXiv:1705.07908 (gambit.hepforge.org)

So far 6 physics studies
arXiv:1705.07917,
arXiv:1705.07935
arXiv:1705.07931,
arXiv:1806.11281
arXiv:1808.10465,
arXiv:1809.02097
+ many more in preparation

Short version:

GAMBIT is a C++ based tool for performing all the book-keeping needed to do a global fit, as well as providing various new calculations for specific new physics predictions and likelihood functions for experiments, and a system for interfacing to existing calculations (software).

- Automated **dependency resolution**
  - track what model parameters are required…
  - ...to compute which physical predictions
  - ...which are required for the likelihood computations
  - (labour saving, + ensure consistency of e.g. SM or DM nuisance parameters across all calculations)
- **Database** of physics calculations and likelihood functions
- Organise **interface** to many existing software packages that compute physics/likelihoods
  - (C++, Fortran, Python, Mathematica)
- Organise interface to **sampling algorithms** (Adaptive Metropolis-Hastings, Nested Sampling, Differential Evolution, ensemble MCMC… continuously adding more)
- **Modular** design (just plug in new calculations as they become available)
- Open source; full code release, full release of all chains for published studies.

Modules:

A module provides GAMBIT with a range of capabilities (the ability to calculate a certain quantity)

DarkBit (arXiv:1705.07920) – dark matter observables
ColliderBit (arXiv:1705.07919) – collider observables (Higgs + SUSY searches from ATLAS, CMS, LEP)
FlavBit (arXiv:1705.07933) – flavour physics (g – 2, $b \to s\gamma$, B decays)
SpecBit (arXiv:1705.07936) – RGE running, masses, mixings, …
DecayBit (arXiv:1705.07936) – decay widths for all relevant particles
PrecisionBit (arXiv:1705.07936) – SM likelihoods, electroweak precision tests
ScannerBit (arXiv:1705.07959) – manages statistics, sampling and optimisation

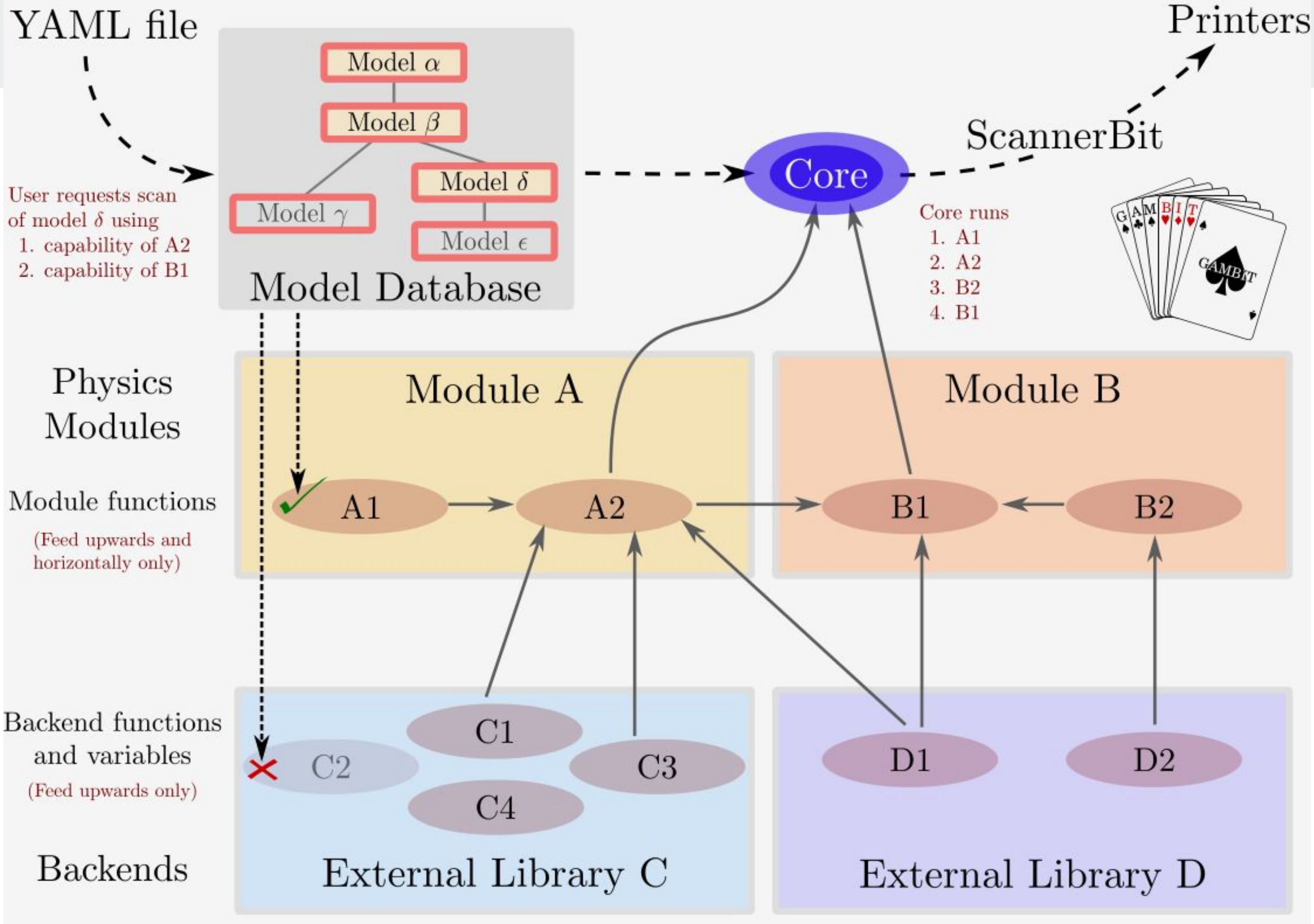
Coming soon:
NeutrinoBit & CosmoBit

(This audience; NuclearBit?)

This audience; NuclearBit?

As a collaboration GAMBIT is focused on BSM physics, but there is no reason the framework could not be used in other fields. “Just” need to plug in the relevant physics calculations!

There is even some important overlap; e.g. Dark Matter direct detection likelihoods are sensitive to uncertainties associated with nuclear structure, which affect predictions for the scattering rates of dark matter particles on different detector materials.



Example data flow during run

Example run: Set parameters/priors

```yaml
Parameters:

  # SM parameters.
  StandardModel_SLHA2: !import include/StandardModel_SLHA2_scan.yaml

  # Nuclear matrix parameters.
  nuclear_params_sigmas_sigmal:
    sigmas:
      range:  [19, 67]
    sigmal:
      range:  [31, 85]
    deltau:
      fixed_value:  0.842
    deltad:
      fixed_value: -0.427
    deltas:
      fixed_value: -0.085
```

```yaml
# SUSY parameters.
CMSSM:
  M0:
    range: [50, 1e4]
    prior_type: log
  M12:
    range: [50, 1e4]
    prior_type: log
  A0:
    range: [-1e4, 1e4]
    flat_start: -1e2
    flat_end: 1e2
    prior_type: double_log_flat_join
  TanBeta:
    range: [3.0, 70]
  SignMu: 1

# Dark matter halo parameters.
Halo_gNFW_rho0:
  rho0:
    range: [.2, .8]
  v0: 235.0
  vesc: 550.0
  vrot: 235.0
  rs: 20.0
  r_sun: 8.5
  alpha: 1
  beta: 3
  gamma: 1
```

Example run: Choose target likelihoods/observables

```yaml
ObsLikes:

    # LHC likelihoods
    - purpose:    LogLike
      capability: LHC_Combined_LogLike

    - purpose:    LogLike
      capability: LHC_Higgs_LogLike

    # LEP likelihoods
    - purpose:    LogLike
      capability: LEP_Higgs_LogLike

    - purpose:    LogLike
      capability: ALEPH_Selectron_LLike

    - purpose:    LogLike
      capability: ALEPH_Smuon_LLike

    - purpose:    LogLike
      capability: ALEPH_Stau_LLike

    - purpose:    LogLike
      capability: L3_Selectron_LLike

    - purpose:    LogLike
      capability: L3_Smuon_LLike

    - purpose:    LogLike
      capability: L3_Stau_LLike
```

```yaml
    # Dark matter likelihoods
    - capability: lnL_oh2
      purpose:    LogLike

    - capability: lnL_FermiLATdwarfs
      purpose:    LogLike

    - capability: XENON100_2012_LogLikelihood
      purpose:    LogLike

    - capability: XENON1T_2018_LogLikelihood
      purpose:    LogLike

    - capability: LUX_2016_LogLikelihood
      purpose:    LogLike

    - capability: PandaX_2016_LogLikelihood
      purpose:    LogLike

    - capability: PandaX_2017_LogLikelihood
      purpose:    LogLike

    - capability: PICO_2L_LogLikelihood
      purpose:    LogLike

    - capability: PICO_60_LogLikelihood
      purpose:    LogLike

    - capability: PICO_60_2017_LogLikelihood
      purpose:    LogLike
```

etc...

Example run: Choose scanning algorithm

```
Scanner:

  use_scanner: de

  scanners:

    multinest:
      plugin: multinest
      like:  LogLike
      nlive: 5000
      tol: 0.1
      updInt: 1

    de:
      plugin: diver
      like: LogLike
      NP: 19200
      convthresh: 1e-5
      verbosity: 1
```

Choose output format:

```
Printer:

  printer: hdf5

  options:
    output_file: "CMSSM.hdf5"
    group: "/CMSSM"

  # printer: ascii
  # options:
  #  buffer_length: 1
  #  delete_file_on_restart: true
  #  output_file: "gambit_output.txt"
```

(Note: can pause and resume runs; pretty useful on HPC facilities with limited walltime)

Example run: Resolve dependency ambiguities

```
Rules:

  # Tell all functions that are able to dump SLHA files to do so.
  #- options:
  #    drop_SLHA_file: true

  # Choose to use DarkSUSY rather than Capt'n General for calculating the capture rate of DM in the Sun
  - capability: capture_rate_Sun
    function: capture_rate_Sun_const_xsec

  # Globally permit maximal mixing between gauge and family eigenstates
  - options:
      gauge_mixing_tolerance: 0.5
      family_mixing_tolerance: 0.5

  # Choose to use gm2calc for g-2
  - capability: muon_gm2
    function: GM2C_SUSY

  # Choose to base the SM prediction for g-2 on e+e- data
  - capability: muon_gm2_SM
    function: gm2_SM_ee

  # Use SuperIso instead of FeynHiggs for b->sgamma
  - capability: bsgamma
    function: SI_bsgamma

  # Use SuperIso instead of FeynHiggs for B_s->mumu
  - capability: Bsmumu_untag
    function: SI_Bsmumu_untag

  # Choose to implement the relic density likelihood as an upper bound, not a detection
  - capability: lnL_oh2
```

Example run: Resolve dependency ambiguities

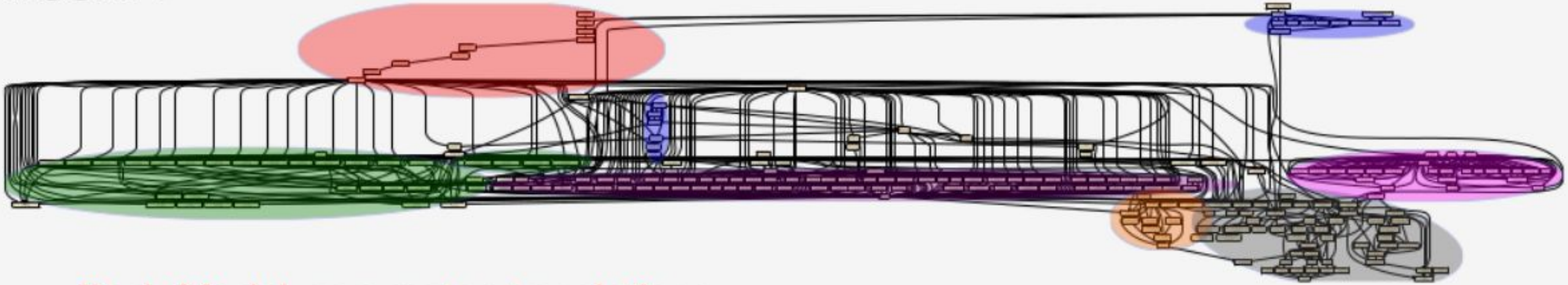
```
Rules:

  # Tell all functions that are able to dump SLHA files to do so.
  #- options:
  #    drop_SLHA_file: true

  # Choose to use DarkSUSY rather than Capt'n General for calculating the capture rate of DM in the Sun
  - capability: capture_rate_Sun
    function: capture_rate_Sun_const_xsec

  # Globally permit maximal mixing between gauge and family eigenstates
  - options:
      gauge_mixing_tolerance: 0.5
      family_mixing_tolerance: 0.5

  # Choose to use gm2calc for g-2
  - capability: muon_gm2
    function: GM2C_SUSY

  # Choose to base the SM prediction for g-2 on e+e- data
  - capability: muon_gm2_SM
    function: gm2_SM_ee

  # Use SuperIso instead of FeynHiggs for b->sgamma
  - capability: bsgamma
    function: SI_bsgamma

  # Use SuperIso instead of FeynHiggs for B_s->mumu
  - capability: Bsmumu_untag
    function: SI_Bsmumu_untag

  # Choose to implement the relic density likelihood as an upper bound, not a detection
  - capability: lnL_oh2
```

Run!

```
gambit -f CMSSM.yaml
```

1. Solves dependency graph

MSSM7:



Red: Model parameter translations

Blue: Precision calculations

Green: LEP rates+likelihoods

Purple: Decays

Orange: LHC observables and likelihoods

Grey: DM direct, indirect and relic density

Pink: Flavour physics

2. Iterate!

Sampler

Likelihood

Make plots, publish paper!

2. Iterate!

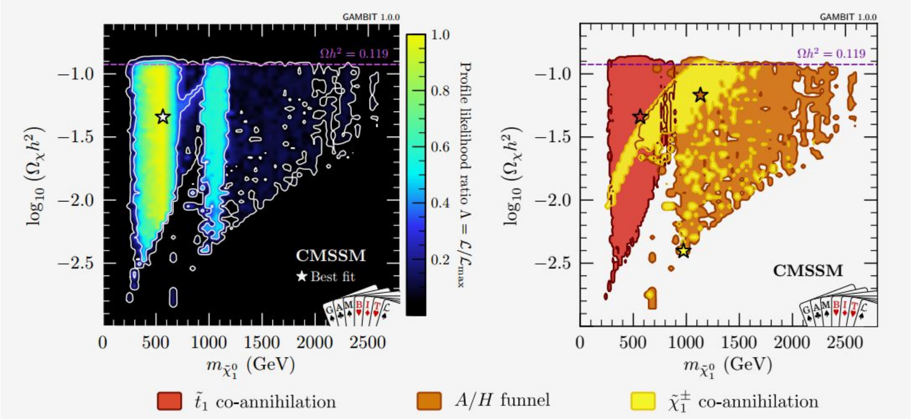
Sampler

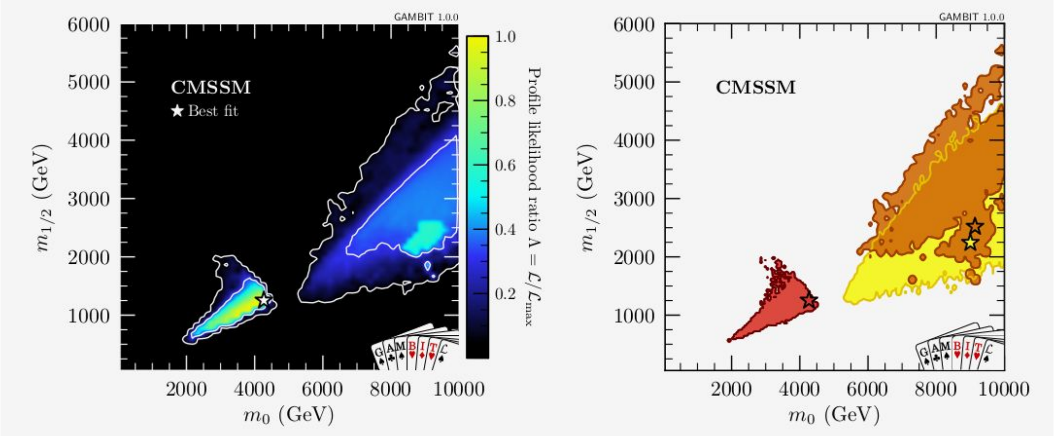
Likelihood

* (We are working on some more advanced “topologies”; e.g. allow more than one L output for fast/slow sampling schemes)

→ Make plots, publish paper!

https://arxiv.org/abs/1705.07935 **“Global fits of GUT-scale SUSY models with GAMBIT”**





A few difficulties

- Typically FAR from well constrained. I.e. nowhere near asymptotic regime.
    - Confidence region coverage typically wrong

A few difficulties

- Typically FAR from well constrained. I.e. nowhere near asymptotic regime.
  - Confidence region coverage typically wrong
- Goodness of fit hard to evaluate (want to detect tensions in the fit; but harder with full likelihoods than simple “chi-squared” type comparisons) -> phenomenon of ever-retreating regions.

A few difficulties

- Typically FAR from well constrained. I.e. nowhere near asymptotic regime.
  - Confidence region coverage typically wrong
- Goodness of fit hard to evaluate (want to detect tensions in the fit; but harder with full likelihoods than simple “chi-squared” type comparisons) -> phenomenon of ever-retreating regions.
- Similarly, hard to evaluate joint statistical significance of “excesses” (example later)

A few difficulties

- Typically FAR from well constrained. I.e. nowhere near asymptotic regime.
  - Confidence region coverage typically wrong
- Goodness of fit hard to evaluate (want to detect tensions in the fit; but harder with full likelihoods than simple “chi-squared” type comparisons) -> phenomenon of ever-retreating regions.
- Similarly, hard to evaluate joint statistical significance of “excesses” (example later)
- Some computations very expensive (particularly LHC, but even DM relic density can sometimes be hard if lots of channels are contributing).
  - GP emulation? We’re looking into it! But our parameter spaces are very “gnarly” due to resonances etc. Typically lots of modes/phases. On-the-fly learning, mode-separation etc? Is this possible?

A few difficulties

- Typically FAR from well constrained. I.e. nowhere near asymptotic regime.
  - Confidence region coverage typically wrong
- **Goodness of fit hard to evaluate (want to detect tensions in the fit; but harder with full likelihoods than simple "chi-squared" type comparisons) -> phenomenon of ever-retreating regions.**
- **Similarly, hard to evaluate joint statistical significance of "excesses" (example later)**
- Some computations very expensive (particularly LHC, but even DM relic density can sometimes be hard if lots of channels are contributing).
  - GP emulation? We're looking into it! But our parameter spaces are very "gnarly" due to resonances etc. Typically lots of modes/phases. On-the-fly learning, mode-separation etc? Is this possible?

LHC likelihoods

Typically we use a “simplified likelihood” to describe each analysis:

$$\mathcal{L}(\boldsymbol{s},\boldsymbol{\gamma}) = \prod_i^{N_{\text{bin}}} \left[\frac{(s_i + b_i + \gamma_i)^{n_i} e^{-(s_i+b_i+\gamma_i)}}{n_i!} \right] \times \frac{1}{\sqrt{\det 2\pi \Sigma}} e^{-\frac{1}{2}\boldsymbol{\gamma}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\gamma}} .$$

(This is best-case; CMS pretty good at providing covariance matrices, but ATLAS typically give us no correlation information at all)

Experiments give us n, b, cov (hopefully).

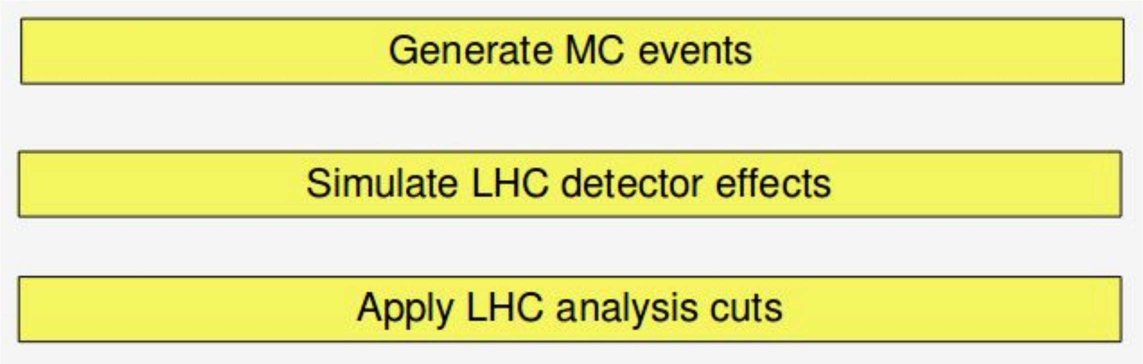
We need to compute s for each point in parameter space.

$$s = \mathcal{L}\sigma A$$

$$\mathcal{L} = 24.3\,\mathrm{fb}^{-1}$$

σ = Production cross section for BSM particle. Computable from theory.

A =



Slow… but we can do it if we throw a lot of CPUs at the problem.

P-values

Likelihood function alone may not be enough to obtain “best-fit” p-values. But we need these if we ever want to discover new physics! (Need to exclude the Standard Model to high statistical significance!)

To compute confidence regions, we use the test statistic

$$q = -2\log\left(\frac{L_{\text{joint}}(\vec{x}|\theta,\hat{\hat{\phi}})}{L_{\text{joint}}(\vec{x}|\hat{\theta},\hat{\phi})}\right)$$

According to asymptotic theory (Wilks’ theorem), q is distributed as chi-squared (with certain DOF) under the null hypothesis (numerator), if certain regularity conditions are met*.

But this can only give us a measure of goodness-of-fit relative to the best fit. IOW this test can never exclude *all* points in a parameter space.

*(and actually these are frequently violated in BSM physics models)

Need a different test statistic.
Depending on the experiment, “s+b” vs “b-only” may be appropriate.
(e.g. Poisson particle counting experiments)

$$q = -2\log\left(\frac{L_{\text{joint}}(\vec{x}\,|\,s(\theta)+b)}{L_{\text{joint}}(\vec{x}\,|\,b)}\right)$$

Asymptotically q is distributed as a normal distribution (arxiv.org/abs/1007.1727) under the b-only hypothesis (which we would like to exclude in order to “discover” new physics)...

...but ONLY for fixed theta! If we scan the parameter space looking for the best new-physics fit, we effectively perform many tests, and select the minimum observed p-value. This is “cheating”
-> **“Problem of multiple comparisons” / “Look-elsewhere effect**”

Difficult to correct for this! Cannot just compute likelihoods, need to do (very expensive!) simulations to determine correct distribution for min(p-value)!

...at least as far as I know! If anyone knows of clever asymptotic results I’d love to hear them!

“Goodness of fit”?

$$q_{\mathrm{GOF}} = -2 \log \frac{\mathcal{L}_{\mathrm{joint}}(\mathbf{s}(\theta), \hat{\eta})}{\mathcal{L}_{\mathrm{joint}}(\hat{\mathbf{s}}, \hat{\hat{\eta}})},$$

Forget our real theory space; go to full space of possible signals
(i.e. free parameter for every signal region/bin)

Good asymptotic properties, no look-elsewhere effect.

But not very powerful! Test is weakened by many degrees of freedom. Our real theory doesn’t make predictions across full space of possible signals. So we are “looking-elsewhere” more than is actually allowed by the theory -> will underestimate significance of any excesses.

Important to be able to do this... makes a big difference!

	Best expected SRs				All SRs; neglect correlations			
Analysis	Local signif. (σ)	SM fit (σ)	EWMSSM fit (σ)	#SRs	Local signif. (σ)	SM fit (σ)	EWMSSM fit (σ)	#SRs
Higgs invisible width	0.9	0.3	0.2	1	0.9	0.3	0.2	1
Z invisible width	0	1.3	1.3	1	0	1.3	1.3	1
ATLAS_4b	0.7	0	0	1	2.1	0	0	2^*
ATLAS_4lep	2.3	2.0	0	1	2.5	1.0	0	4
ATLAS_MultiLep_2lep_0jet	0.9	0.3	0.1	1	1.3	0	0	6
ATLAS_MultiLep_2lep_jet	0	0	0.5	1	0.8	0.5	0.3	3
ATLAS_MultiLep_3lep	1.8	1.6	0.6	1	1.2	0.4	0.3	11
ATLAS_RJ_2lep_2jet	0	0.3	0.5	1	1.5	1.8	1.5	4
ATLAS_RJ_3lep	2.8	2.4	1.0	1	3.5	2.6	0.5	4
CMS_1lep_2b	0.9	0.3	0.3	1	0	0	0	2
CMS_2lep_soft	0.4	0.2	0.2	12	0.4	0.2	0.2	12
CMS_2OSlep	0	0.4	0.6	7	0	0.4	0.6	7
CMS_MultiLep_2SSlep	0.2	0	0	1	0.2	0	0	2
CMS_MultiLep_3lep	0	0	0.5	1	0	0	0	6
Combined	3.5	1.5	0.3	31	4.2	1.3	0	65

Bayesian naturalness

Famous problem in particle physics: Hierarchy problem.

-> Why is the electroweak scale much smaller than the Planck scale?

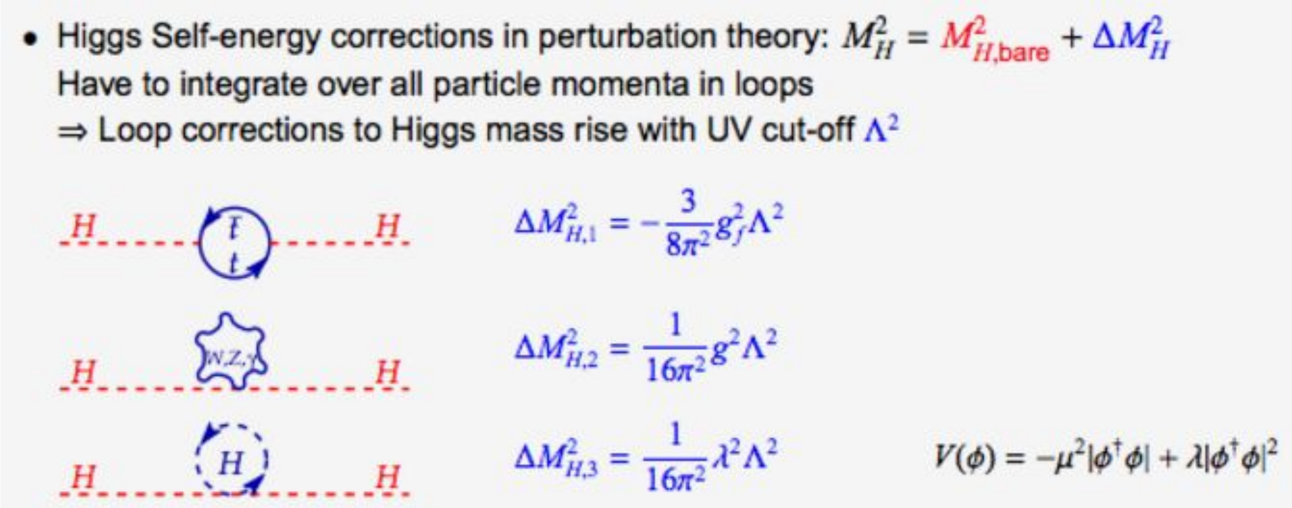


Diagram credit: U. Uwer lecture notes: http://www.physi.uni-heidelberg.de/~uwer/lectures/ParticlePhysics/Vorlesung/Lect-10a.pdf

A popular solution: Supersymmetry

Higgs Self-Energy

$$\Delta M^2_{H,\mathrm{top}} = -\frac{3}{8\pi^2} g_t^2 \Lambda^2 + \ldots$$

Bosons and fermions produce different signs in loops $\Longrightarrow$
Introduce "superpartner" for top = scalar top = "stop" = $\tilde{t}$

$$\Delta M^2_{H,\mathrm{stop}} = +\frac{3}{8\pi^2} g_t^2 \Lambda^2 + \ldots$$

Total correction

$$\Delta M^2_{H,\mathrm{top}} + \Delta M^2_{H,\mathrm{stop}} = -\frac{3}{8\pi^2} g_t^2 (m_{\tilde{t}}^2 - m_t^2) \log \frac{\Lambda^2}{m_{\tilde{t}}^2}$$

▶ "Naturalness" argument: $m_{\tilde{t}}$ should be not much larger than m_t $\qquad m_{\tilde{t}} \sim \mathrm{TeV}$???

Diagram credit: U. Uwer lecture notes: http://www.physi.uni-heidelberg.de/~uwer/lectures/ParticlePhysics/Vorlesung/Lect-10a.pdf

A Bayesian argument?

This “naturalness” idea can be recast into Bayesian language, which lets us include it in Bayesian global fits in a rigorous way,

General idea: Naturalness is about **prior probabilities** and **predictivity**. A theory (or prediction for some observable) is “natural” if the theory predicts the correct observable values with “reasonable” prior probability (or not unreasonably low prior probability).

Toy model

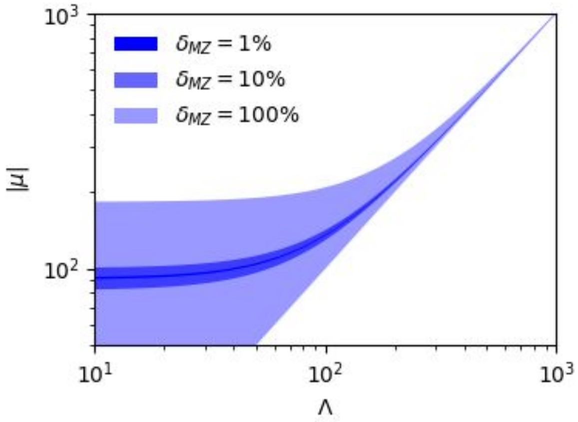
$$m_Z^2 = \mu^2 + \Lambda^2$$

We know mZ ~ 91 GeV. In "how much" of the parameter space can this value be attained?

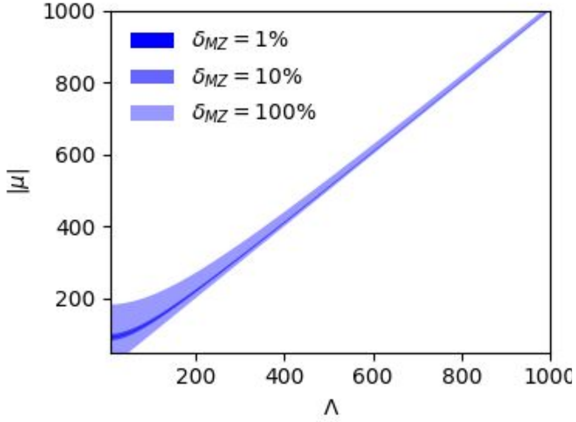
Toy model

$$m_Z^2 = \mu^2 + \Lambda^2$$

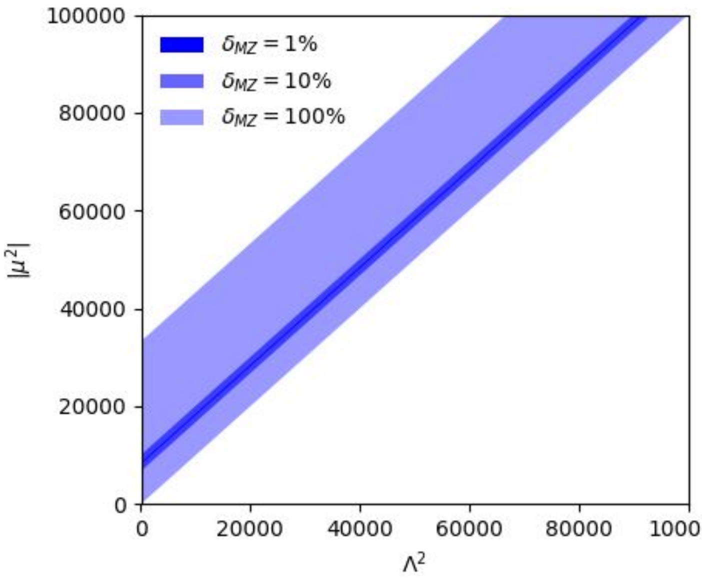
We know mZ ~ 91 GeV. In "how much" of the parameter space can this value be attained?



"log"



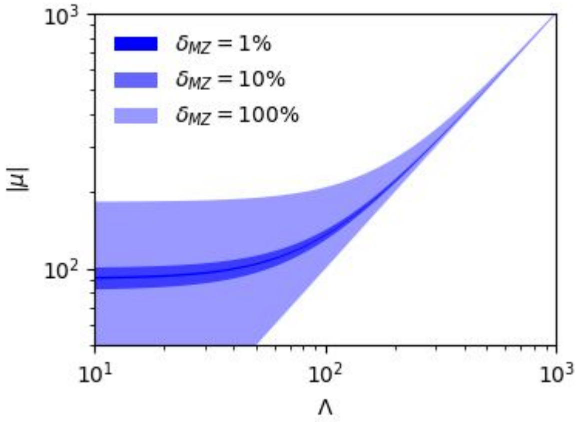
"flat"



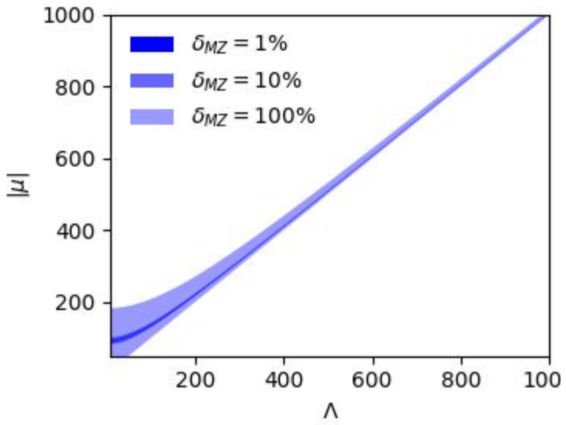
"squared"

$$(\delta m_Z \equiv X\% \times 91\,\mathrm{GeV})$$

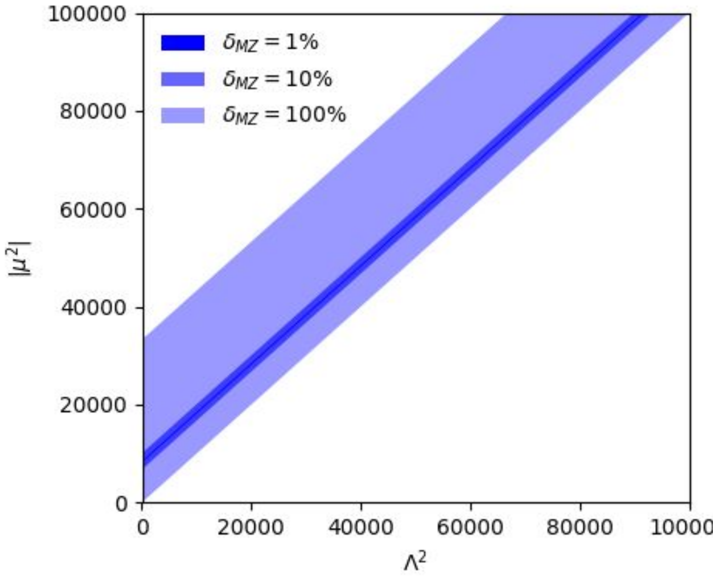
$$m_Z^2 = \mu^2 + \Lambda^2$$



“log”



“flat”

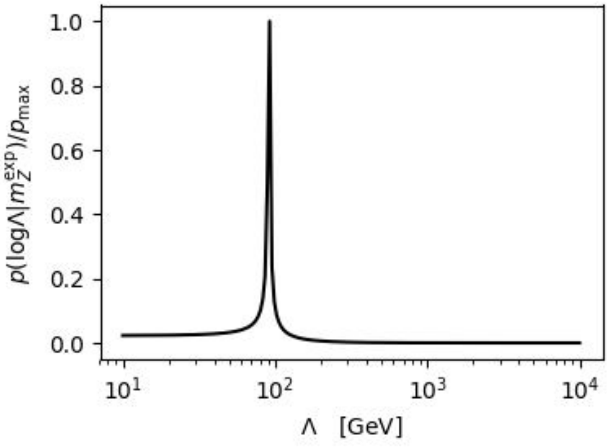


“squared”

Naturalness argument corresponds to having priors similar to either 1 or 2 (log or flat).
Much more prior probability of predicting correct Z mass at low mu and Lambda

Alternatively; posterior favours low mu and Lambda once Z is measured.

Can formalize this for specific models e.g. “effective priors”



$$p(\log\Lambda|m_Z^{\exp}) = \frac{1}{Z}\left|\frac{(m_Z^{\exp})^2}{(m_Z^{\exp})^2 - \Lambda^2}\right|$$

$$\log\mu = \frac{1}{2}\log\left(m_Z^2 - \Lambda^2\right)$$

$$\log\Lambda = \log\Lambda$$

But no time for that! (Some details in supplementary material if interested).

Summary

- Lots of models to attack with lots of experimental data in BSM physics.
- Compute likelihood functions for many experiments under many models -> infer parameters (compare models)
- GAMBIT is an open-source tool to help organise this task!
  - Lots of book-keeping tools, modular, can be applied to general parameter inference tasks if you plug in the required physics.
- Some difficulties computing “look-elsewhere”/”trial” corrections for p-values
- Interesting connections between naturalness arguments/hierarchy problems and Bayesian inference!

Supplementary material

Suppose we want to do some physics with this model.

Suppose it involved exploring the parameter space of the model.

We wouldn't want to waste our time exploring the regions where we don't even get mZ correct, so often we would perform a parameter exchange such that we can directly plug in the correct Z mass! In the toy case we might do:

$$\mu^2 = m_Z^2 - \Lambda^2$$

So now mu is fixed by mZ, and our only free parameter is Lambda.

Let us now be Bayesians. We want to do some inference about our model, so we need a prior for Lambda.

What should we use? Flat? Log?

No!!!

I think it is quite clear that no! We should not do that! If we do this we would completely throw away the effect of the “size” of the viable parameter space in the mu direction!

But more formally, why not?

Well first, the Z measurement should contribute to the likelihood function, which we ignore if we just take the best fit value and then set priors on other free parameters.

But also we fail to carefully consider the effects of the parameter transformation we have done! Priors in one set of parameters will look different under another set of parameters, so we need to make sure we understand what effect our parameter choices are having!

Effect of parameter transformations on priors

$$m_Z^2 = \mu^2 + \Lambda^2$$

$$p(\theta_1, \theta_2) = p(\xi_1, \xi_2) \left| \frac{\partial\{\xi_1, \xi_2\}}{\partial\{\theta_1, \theta_2\}} \right|$$

Effect of parameter transformations on priors

$$m_Z^2 = \mu^2 + \Lambda^2$$

$$p(m_Z^2, \Lambda^2) = p(\mu^2, \Lambda^2) \begin{vmatrix} \frac{\partial \mu^2}{\partial m_Z^2} & \frac{\partial \mu^2}{\partial \Lambda^2} \\ \frac{\partial \Lambda^2}{\partial m_Z^2} & \frac{\partial \Lambda^2}{\partial \Lambda^2} \end{vmatrix}$$
$$= p(\mu^2, \Lambda^2) \begin{vmatrix} 1 & -1 \\ 0 & 1 \end{vmatrix} = p(\mu^2, \Lambda^2)$$

"squared"

$$\mu^2 = m_Z^2 - \Lambda^2$$
$$\Lambda^2 = \Lambda^2$$

$$p(\theta_1, \theta_2) = p(\xi_1, \xi_2) \left| \frac{\partial\{\xi_1, \xi_2\}}{\partial\{\theta_1, \theta_2\}} \right|$$

Effect of parameter transformations on priors

$$m_Z^2 = \mu^2 + \Lambda^2$$

$$p(m_Z^2, \Lambda^2) = p(\mu^2, \Lambda^2) \begin{vmatrix} \frac{\partial \mu^2}{\partial m_Z^2} & \frac{\partial \mu^2}{\partial \Lambda^2} \\ \frac{\partial \Lambda^2}{\partial m_Z^2} & \frac{\partial \Lambda^2}{\partial \Lambda^2} \end{vmatrix}$$
$$= p(\mu^2, \Lambda^2) \begin{vmatrix} 1 & -1 \\ 0 & 1 \end{vmatrix} = p(\mu^2, \Lambda^2)$$

"squared"

$$\mu^2 = m_Z^2 - \Lambda^2$$
$$\Lambda^2 = \Lambda^2$$

$$p(\theta_1, \theta_2) = p(\xi_1, \xi_2) \left| \frac{\partial\{\xi_1, \xi_2\}}{\partial\{\theta_1, \theta_2\}} \right|$$

$$p(m_Z, \Lambda) = p(\mu, \Lambda) \begin{vmatrix} \frac{\partial \mu}{\partial m_Z} & \frac{\partial \mu}{\partial \Lambda} \\ \frac{\partial \Lambda}{\partial m_Z} & \frac{\partial \Lambda}{\partial \Lambda} \end{vmatrix}$$
$$= p(\mu, \Lambda) \begin{vmatrix} \frac{m_Z}{\mu} & \frac{-\Lambda}{\mu} \\ 0 & 1 \end{vmatrix} = p(\mu, \Lambda) \cdot \left| \frac{m_Z}{\mu} \right|$$

"flat"

$$|\mu| = \sqrt{m_Z^2 - \Lambda^2}$$
$$\Lambda = \Lambda$$

Effect of parameter transformations on priors

$$m_Z^2 = \mu^2 + \Lambda^2$$

$$p(m_Z^2, \Lambda^2) = p(\mu^2, \Lambda^2) \begin{vmatrix} \frac{\partial \mu^2}{\partial m_Z^2} & \frac{\partial \mu^2}{\partial \Lambda^2} \\ \frac{\partial \Lambda^2}{\partial m_Z^2} & \frac{\partial \Lambda^2}{\partial \Lambda^2} \end{vmatrix}$$

$$= p(\mu^2, \Lambda^2) \begin{vmatrix} 1 & -1 \\ 0 & 1 \end{vmatrix} = p(\mu^2, \Lambda^2)$$

"squared"

$$\mu^2 = m_Z^2 - \Lambda^2$$
$$\Lambda^2 = \Lambda^2$$

$$p(\theta_1, \theta_2) = p(\xi_1, \xi_2) \left| \frac{\partial \{\xi_1, \xi_2\}}{\partial \{\theta_1, \theta_2\}} \right|$$

$$p(m_Z, \Lambda) = p(\mu, \Lambda) \begin{vmatrix} \frac{\partial \mu}{\partial m_Z} & \frac{\partial \mu}{\partial \Lambda} \\ \frac{\partial \Lambda}{\partial m_Z} & \frac{\partial \Lambda}{\partial \Lambda} \end{vmatrix}$$

$$= p(\mu, \Lambda) \begin{vmatrix} \frac{m_Z}{\mu} & \frac{-\Lambda}{\mu} \\ 0 & 1 \end{vmatrix} = p(\mu, \Lambda) \cdot \left| \frac{m_Z}{\mu} \right|$$

"flat"

$$|\mu| = \sqrt{m_Z^2 - \Lambda^2}$$
$$\Lambda = \Lambda$$

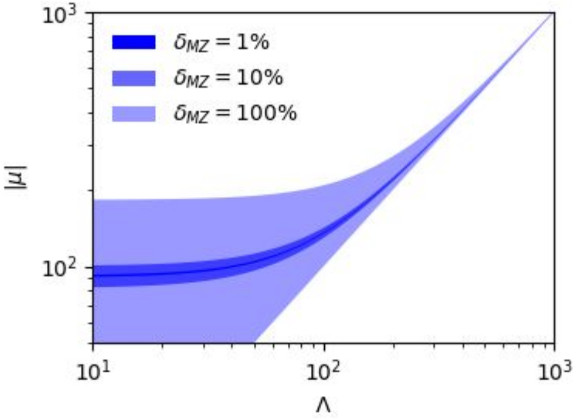
$$p(\log m_Z, \log \Lambda) = p(\log \mu, \log \Lambda) \begin{vmatrix} \frac{\partial \log \mu}{\partial \log m_Z} & \frac{\partial \log \mu}{\partial \log \Lambda} \\ \frac{\partial \log \Lambda}{\partial \log m_Z} & \frac{\partial \log \Lambda}{\partial \log \Lambda} \end{vmatrix}$$

$$= p(\log \mu, \log \Lambda) \begin{vmatrix} \frac{m_Z^2}{\mu^2} & \frac{-\Lambda^2}{\mu^2} \\ 0 & 1 \end{vmatrix} = p(\log \mu, \log \Lambda) \cdot \left| \frac{m_Z^2}{\mu^2} \right|$$

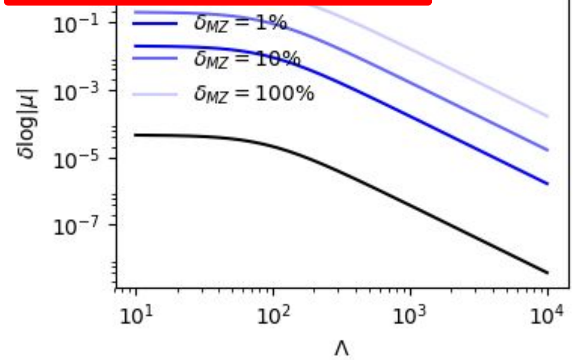
"log"

$$\log \mu = \frac{1}{2} \log \left(m_Z^2 - \Lambda^2 \right)$$
$$\log \Lambda = \log \Lambda$$

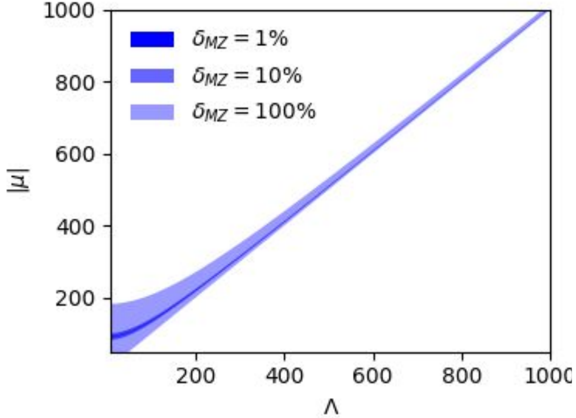
“log”



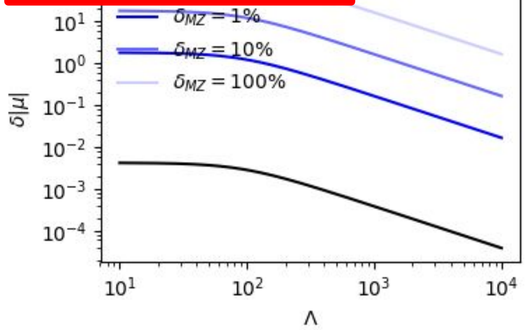
$$\left|\frac{m_Z^2}{\mu^2}\right| = \left|\frac{m_Z^2}{m_Z^2 - \Lambda^2}\right|$$



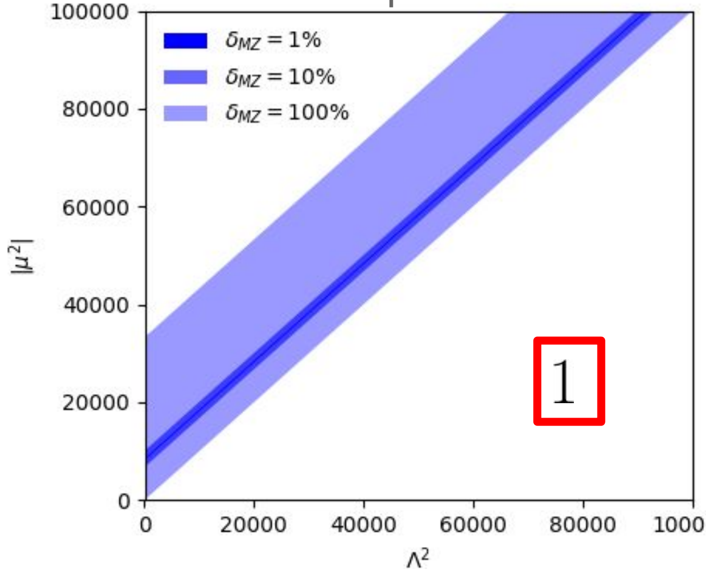
“flat”



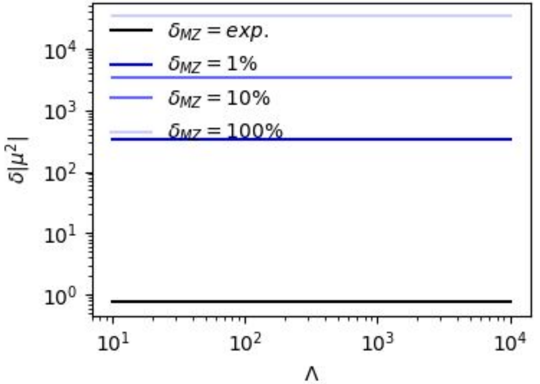
$$\left|\frac{m_Z}{\mu}\right| = \left|\frac{m_Z}{\sqrt{m_Z^2 - \Lambda^2}}\right|$$



“squared”



1



“Effective” priors

So we have seen clear volume effects that we don’t want to ignore.

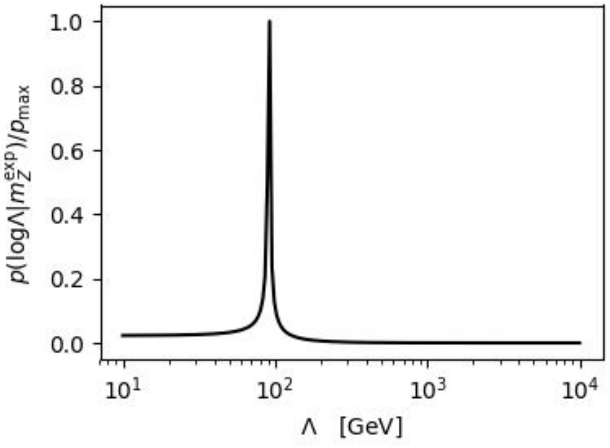
But suppose that we *really* want to do our coordinate transformation plus parameter fixing trick to reduce the dimensionality of the parameter space we need to consider.

We can construct a “new” prior (in fact, a posterior!) that properly encodes this information if we just follow Bayes’ theorem, and then integrate out the parameter we want to remove.

“Effective” priors

$$
\begin{aligned}
p(\log \Lambda | m_Z^{\exp}) &= \int_\Omega p(\log m_Z, \log \Lambda | m_Z^{\exp}) \, \mathrm{d} \log m_Z \\
&= \frac{1}{Z} \int_\Omega p(m_Z^{\exp} | \log m_Z, \log \Lambda) \cdot p(\log m_Z, \log \Lambda) \, \mathrm{d} \log m_Z \\
&= \frac{1}{Z} \int_\Omega \delta(\log m_Z^{\exp} - \log m_Z) \cdot p(\log \mu, \log \Lambda) \cdot \left| \frac{m_Z^2}{\mu^2} \right| \, \mathrm{d} \log m_Z \\
&= \frac{1}{Z} \int_\Omega \delta(\log m_Z^{\exp} - \log m_Z) \cdot \left| \frac{m_Z^2}{m_Z^2 - \Lambda^2} \right| \, \mathrm{d} \log m_Z \\
&= \frac{1}{Z} \left| \frac{\left(m_Z^{\exp}\right)^2}{\left(m_Z^{\exp}\right)^2 - \Lambda^2} \right|
\end{aligned}
$$

“Effective” priors



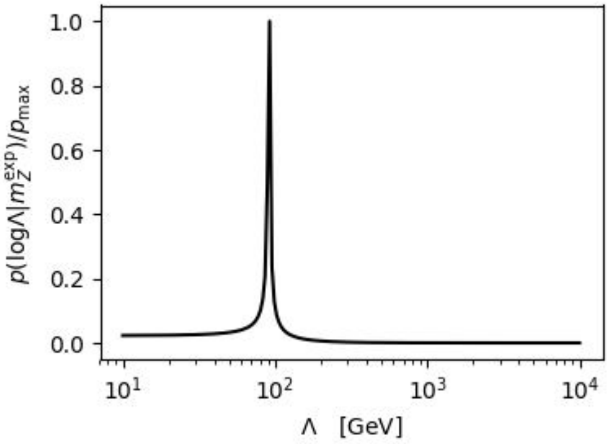
$$p(\log\Lambda|m_Z^{\rm exp}) = \frac{1}{Z}\left|\frac{\left(m_Z^{\rm exp}\right)^2}{\left(m_Z^{\rm exp}\right)^2 - \Lambda^2}\right|$$

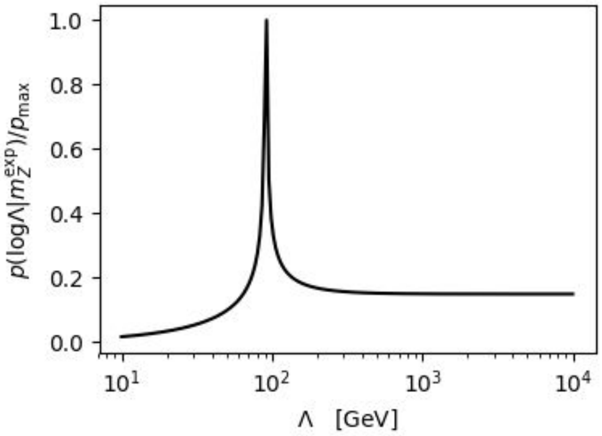
“log”

$$\log\mu = \frac{1}{2}\log\left(m_Z^2 - \Lambda^2\right)$$

$$\log\Lambda = \log\Lambda$$

“Effective” priors





$$p(\log\Lambda|m_Z^{\rm exp}) = \frac{1}{Z}\left|\frac{\left(m_Z^{\rm exp}\right)^2}{\left(m_Z^{\rm exp}\right)^2 - \Lambda^2}\right|$$

$$p(\log\Lambda|m_Z^{\rm exp}) = \frac{1}{Z}\left|\frac{\left(m_Z^{\rm exp}\right)^2}{\sqrt{\left(m_Z^{\rm exp}\right)^2 - \Lambda^2}}\right|\cdot\Lambda$$

“log”

$$\log\mu = \frac{1}{2}\log\left(m_Z^2 - \Lambda^2\right)$$

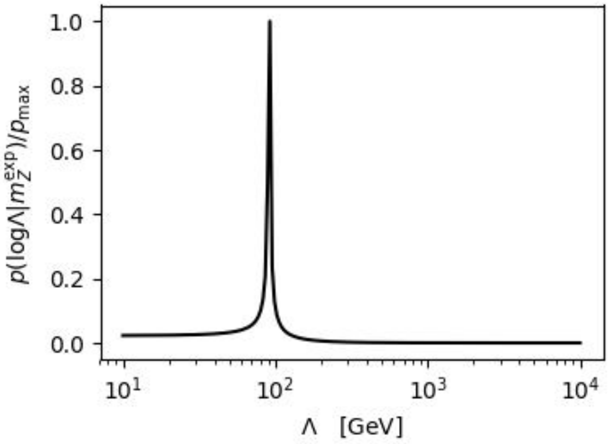
$$\log\Lambda = \log\Lambda$$

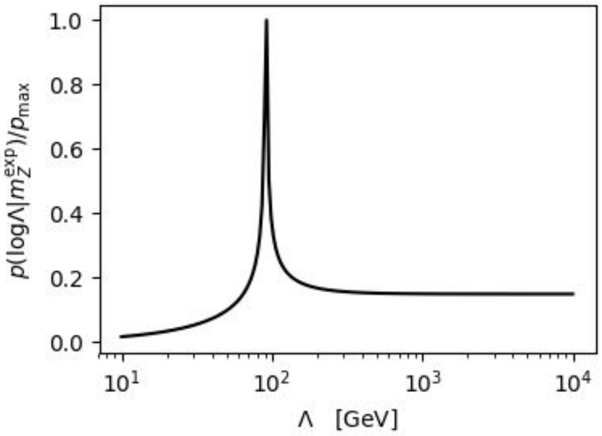
“flat”

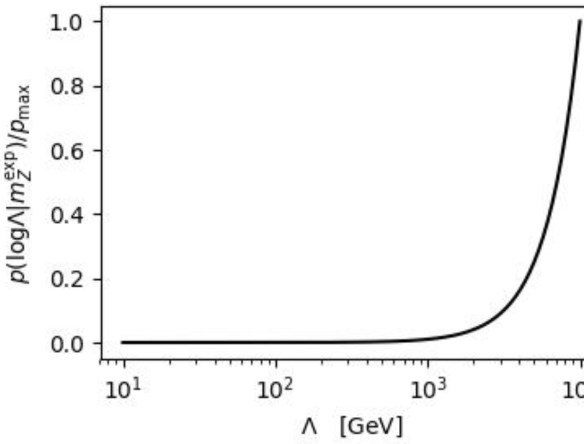
$$|\mu| = \sqrt{m_Z^2 - \Lambda^2}$$

$$\Lambda = \Lambda$$

“Effective” priors







$$p(\log\Lambda|m_Z^{\rm exp}) = \frac{1}{Z}\left|\frac{(m_Z^{\rm exp})^2}{(m_Z^{\rm exp})^2 - \Lambda^2}\right|$$

“log”

$$\log\mu = \frac{1}{2}\log\left(m_Z^2 - \Lambda^2\right)$$

$$\log\Lambda = \log\Lambda$$

$$p(\log\Lambda|m_Z^{\rm exp}) = \frac{1}{Z}\left|\frac{(m_Z^{\rm exp})^2}{\sqrt{(m_Z^{\rm exp})^2 - \Lambda^2}}\right|\cdot\Lambda$$

“flat”

$$|\mu| = \sqrt{m_Z^2 - \Lambda^2}$$

$$\Lambda = \Lambda$$

$$p(\log\Lambda|m_Z^{\rm exp}) = \frac{1}{Z}\cdot\Lambda^2$$

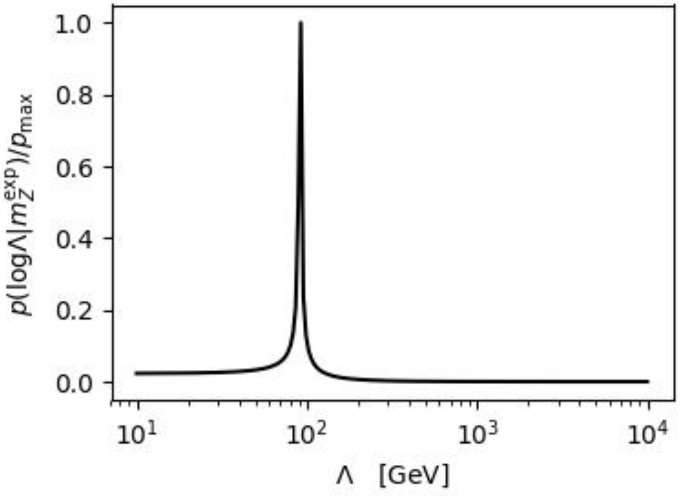
“squared”

$$\mu^2 = m_Z^2 - \Lambda^2$$

$$\Lambda^2 = \Lambda^2$$

This is the essence of the Hierarchy Problem.

Given the observed electroweak scale, we expect the Standard Model to be modified by physics at a scale not much above the electroweak scale, if we were *a priori* ignorant of what scale this physics should exist at.



Equivalently, if there are quadratic quantum corrections to the electroweak scale coming from new physics at a higher scale (expect at least Planck), then it is very unlikely to observe the weak scale at ~100 GeV.

With effective priors like these we can “automatically” capture this theoretical intuition.

(Toy model doesn’t make sense for scales below EW; not a good effective field theory here)

Relationship to "conventional" fine-tuning measures

$$\Delta_{BG} \equiv max_i\,[c_i] \quad \text{where} \quad c_i = \left|\frac{\partial \ln m_Z^2}{\partial \ln a_i}\right| = \left|\frac{a_i}{m_Z^2}\frac{\partial m_Z^2}{\partial a_i}\right|$$

Our toy model, when transforming from log mu to log mz, has Jacobian:

$$J = \begin{vmatrix} \frac{\partial \log \mu}{\partial \log m_Z} & \frac{\partial \log \mu}{\partial \log \Lambda} \\ \frac{\partial \log \Lambda}{\partial \log m_Z} & \frac{\partial \log \Lambda}{\partial \log \Lambda} \end{vmatrix} = \left|\frac{\partial \log \mu}{\partial \log m_Z}\right|$$

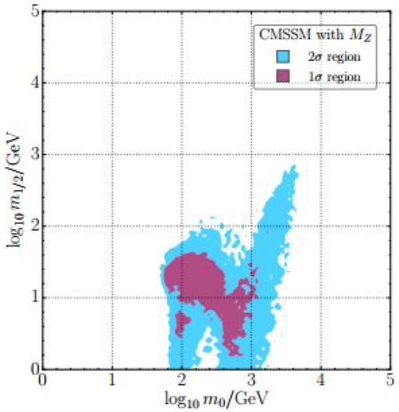
So the inverse of this (the Jacobian of the inverse transformation) is something that rather closely resembles Delta_BG! For the sake of comparison we define:

$$\Delta_J = J^{-1}$$

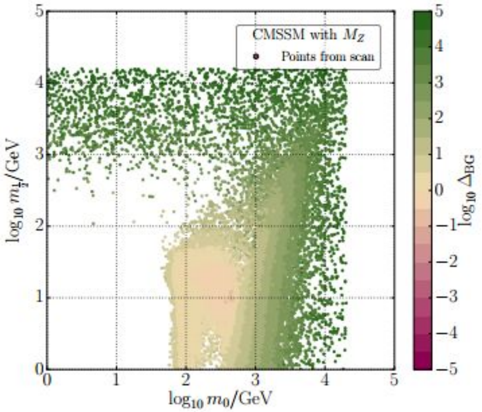
In one dimension it is identical to Delta_BG, but for more complicated transformations it captures much more information about correlations between parameter directions.

Scans constrained only by mZ

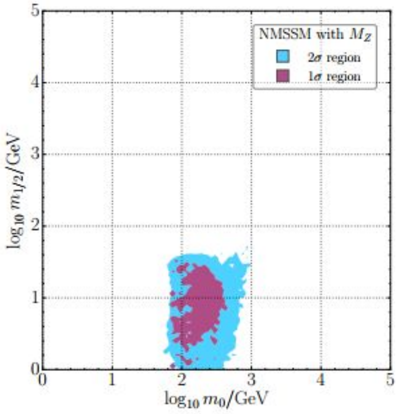
arxiv.org/abs/1709.07895



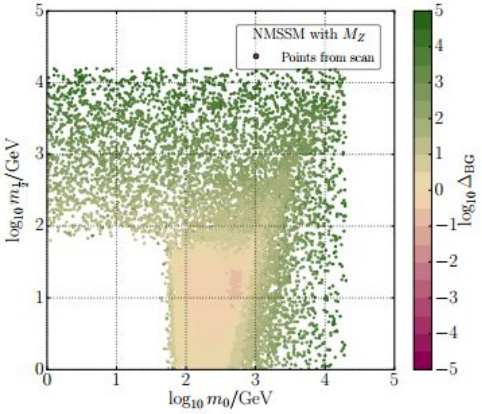
(a) CMSSM credible regions



(b) CMSSM Δ_{BG}



(c) NMSSM credible regions



(d) NMSSM Δ_{BG}

“Look-elsewhere effect”

Related concepts: “trial correction”, “p-hacking”, “data-dredging”, “cherry-picking”

Basic idea:

If you do many different sorts of p-value calculation, and then pick the one with the lowest p-value after looking at the data, you have just screwed up the frequentist properties of your test procedure and your p-value isn’t the right number anymore.

“Look-elsewhere effect”

Coin example: In each trial, take the lowest p-value out of the “H/T” test and the “N_runs” test.

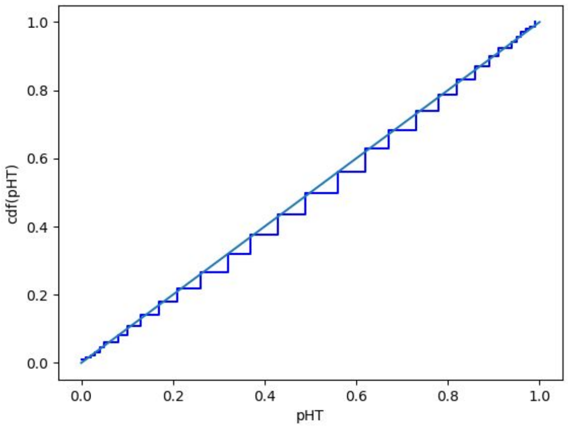
I.e $$p = \min(p_{H/T}, p_{N_{\rm runs}})$$

"Look-elsewhere effect"

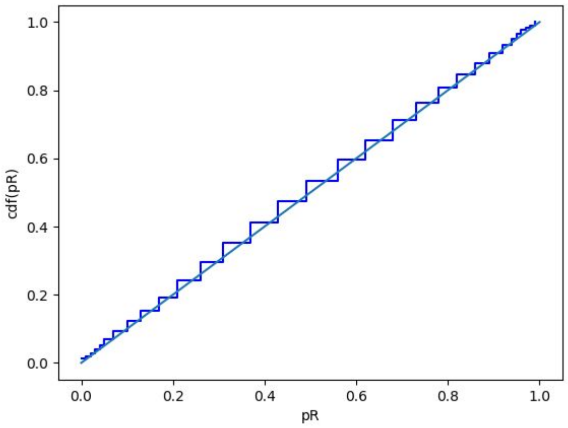
Coin example: In each trial, take the lowest p-value out of the "H/T" test and the "N_runs" test.

I.e $$p = \min(p_{H/T}, p_{N_{\mathrm{runs}}})$$

CDF H/T p-value



CDF runs p-value



CDF min p-value