CMS Public Statistical Models

G. Ortona for the CMS Collaboration

Dall-e impression of a statistical model in particle physics

Why do we want to publish Statistical models

"*The statistical models used to derive the results of experimental analyses are of incredible scientific value and are essential information for analysis preservation and reuse*"
[SciPost Phys. 12, 037 (2022), arXiv:2109.04981](https://arxiv.org/abs/2109.04981)

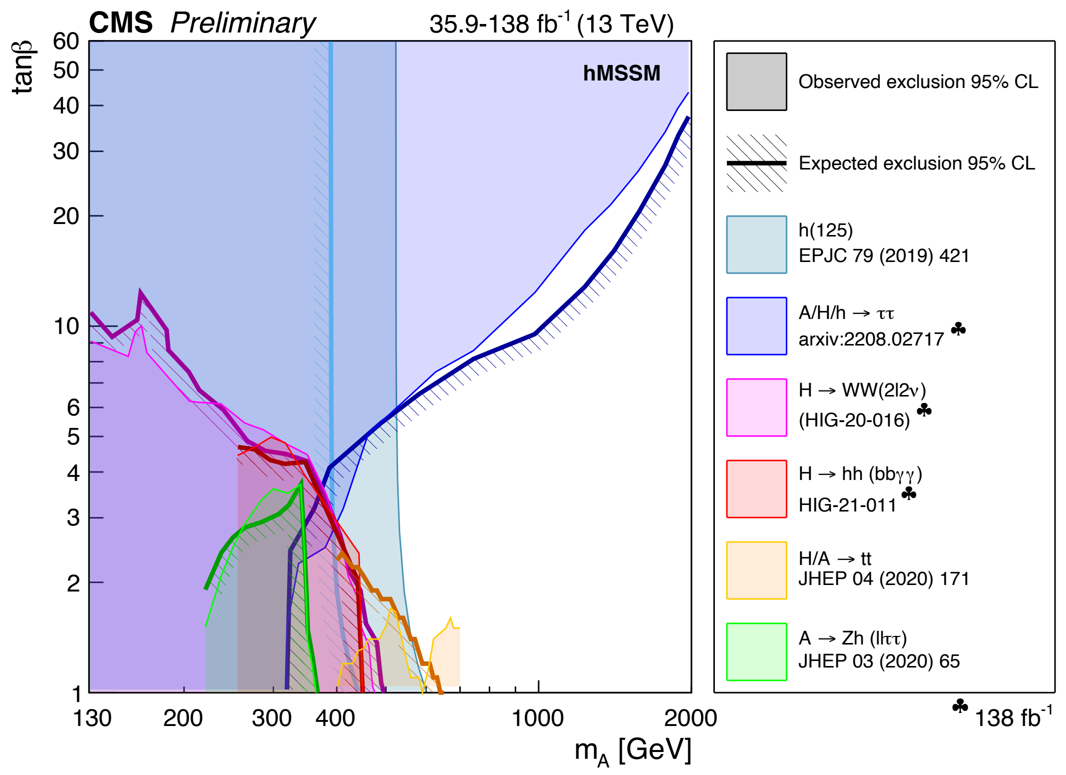
Statistical models provide an excellent resource for the community. The discussion about publishing them started in the early 2000s. Picked momentum in the ATLAS and CMS collaborations with LHC-Run2 results, also thanks to data-sharing options such as HEPData.

Publishing them will help maximize the scientific impact of the analysis, and facilitate

- Preservation and documentation: the mathematical construction of the analysis in full detail.
- Reinterpretation and reuse (within and outside the collaborations)
- Combination of multiple analyses
- Combination of analyses across experiments
- Education on statistical procedures
- Tool development: Statistical software updates can use real world examples to test and debug their recent developments.

The interpretation spectrum

Exclusion contours in full models

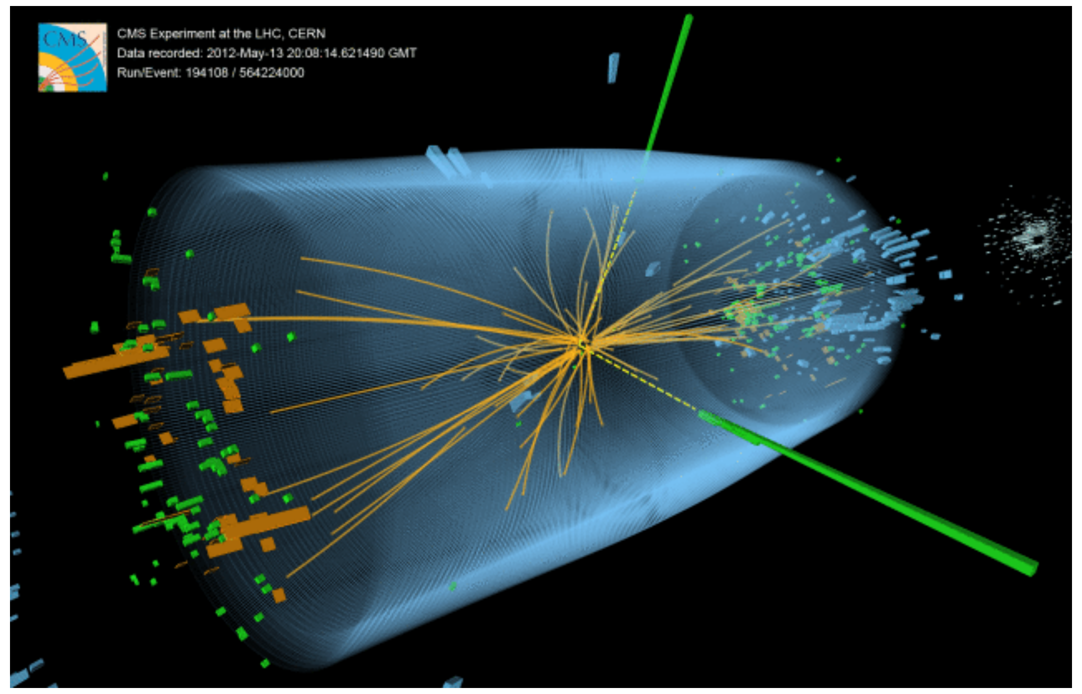


POIs intervals

likelihood function

Unfolded/corrected data

Raw data



- Highest level of interpretation included
- Easy to communicate
- Immediately relevant for specific models

- The likelihood function contain all the information needed to process an experimental results
- Does not need (too much) internal knowledge
- Can be used to recast results (assuming we provide the necessary tools)

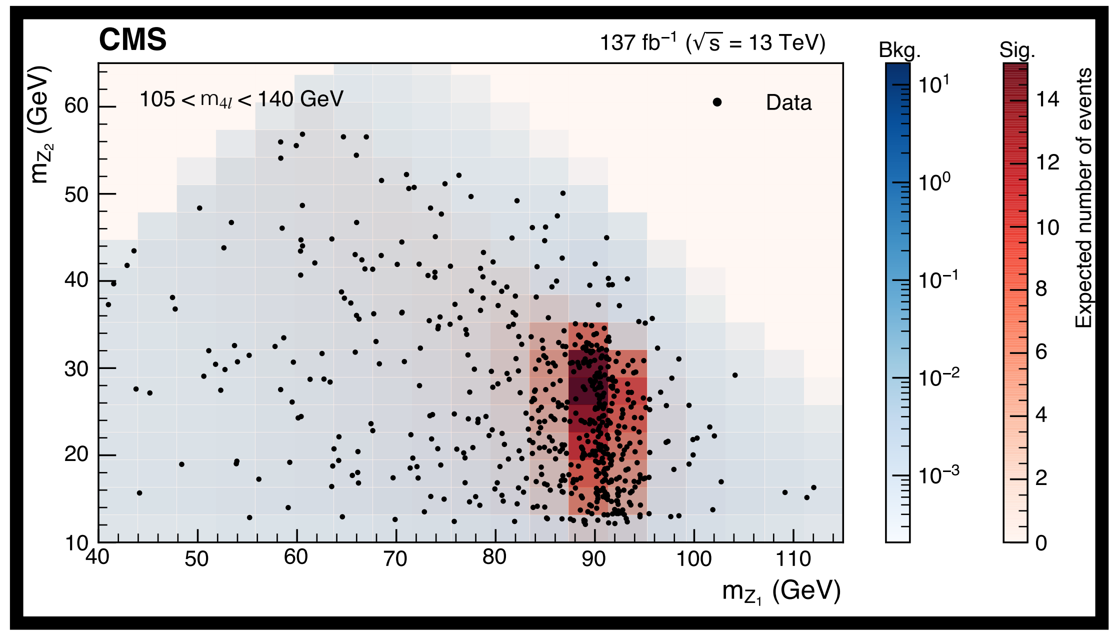
- Minimal interpretation
- Requires detailed knowledge to be used safely

~~Stolen from~~ Courtesy of N. Wardle from an idea of P. Owen @ Reinterp2021

A CMS likelihood

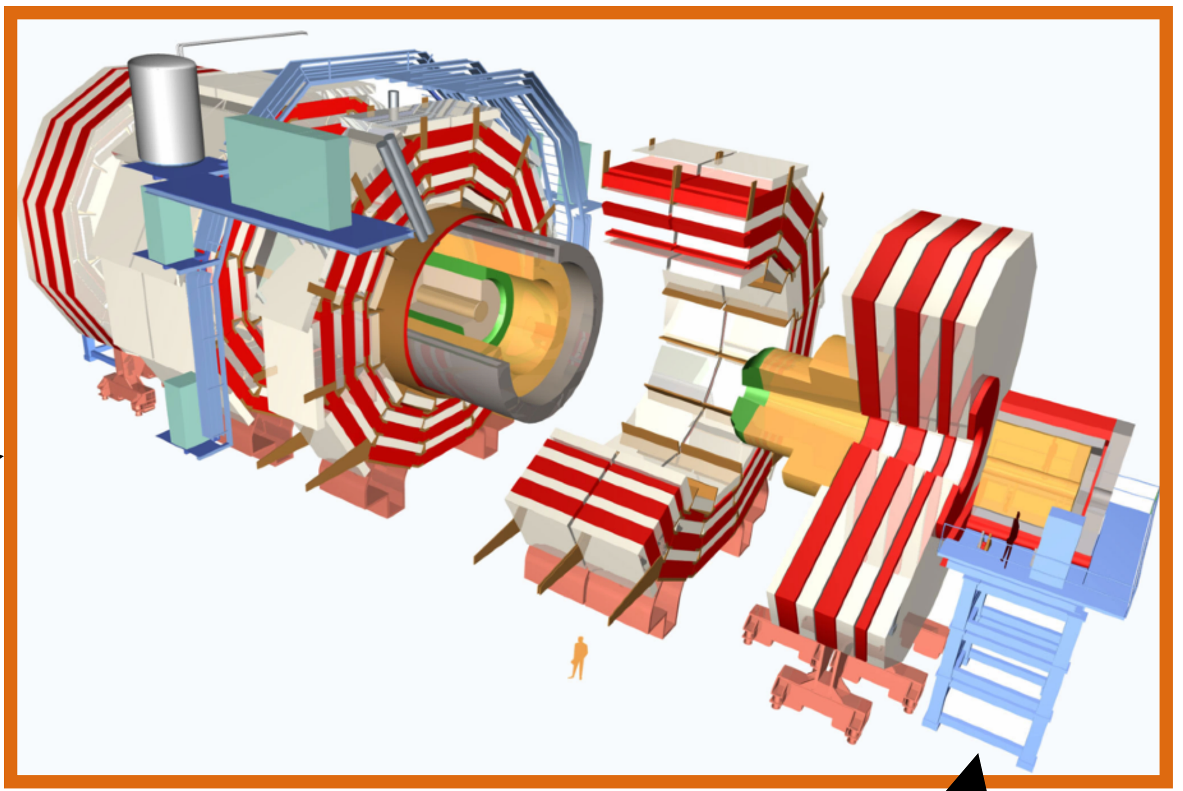
The experimental likelihood function is the most complete picture we have of all the parameters entering a measurement and their relations.

- Takes into account the data, the POIs, global and auxiliary observables, nuisance parameters…
- Highly factorized



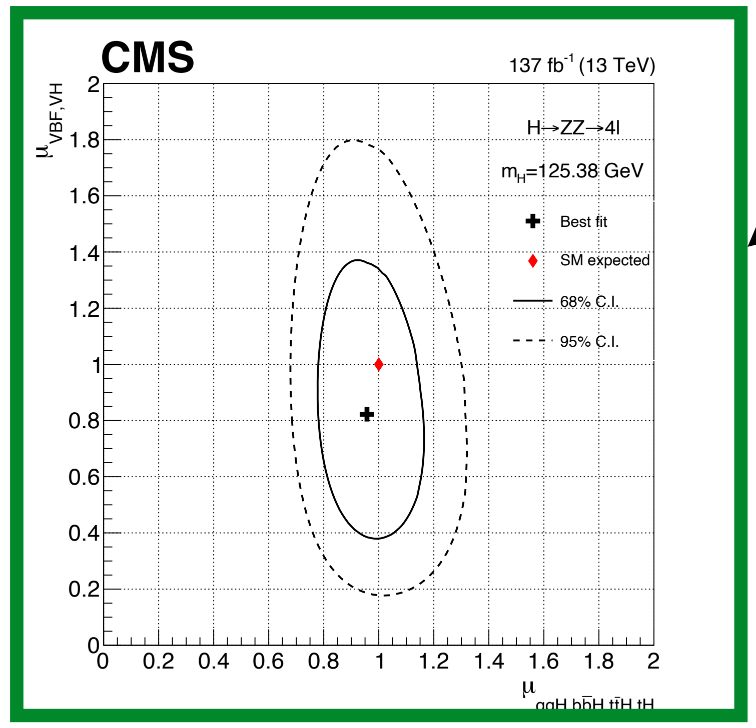
data in each channel

nuisance parameters parametrise the experimental/theoretical systematic uncertainties

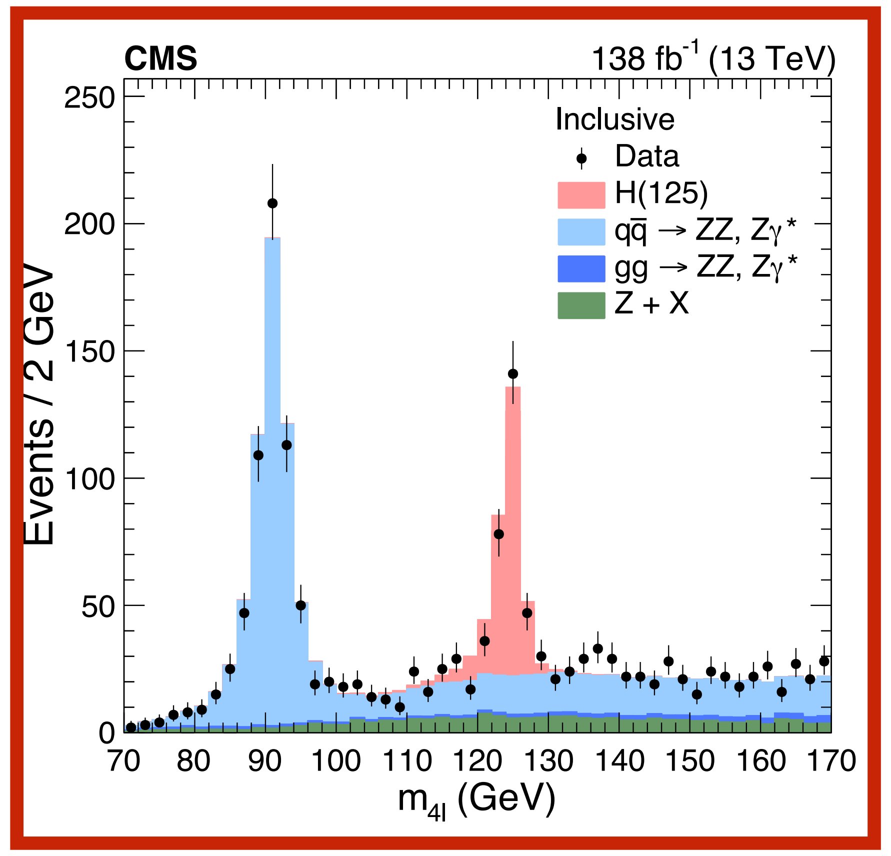


$$L(\vec{\mu};\vec{\nu}) = \prod_n p\left(x_n; \sum_i \mu_i S_{i,n}(\vec{\nu}) + \sum_k B_k(\vec{\nu})\right) \cdot \prod_i p(y_i;\nu_i)$$

Parameters of Interest parametrise the physics model



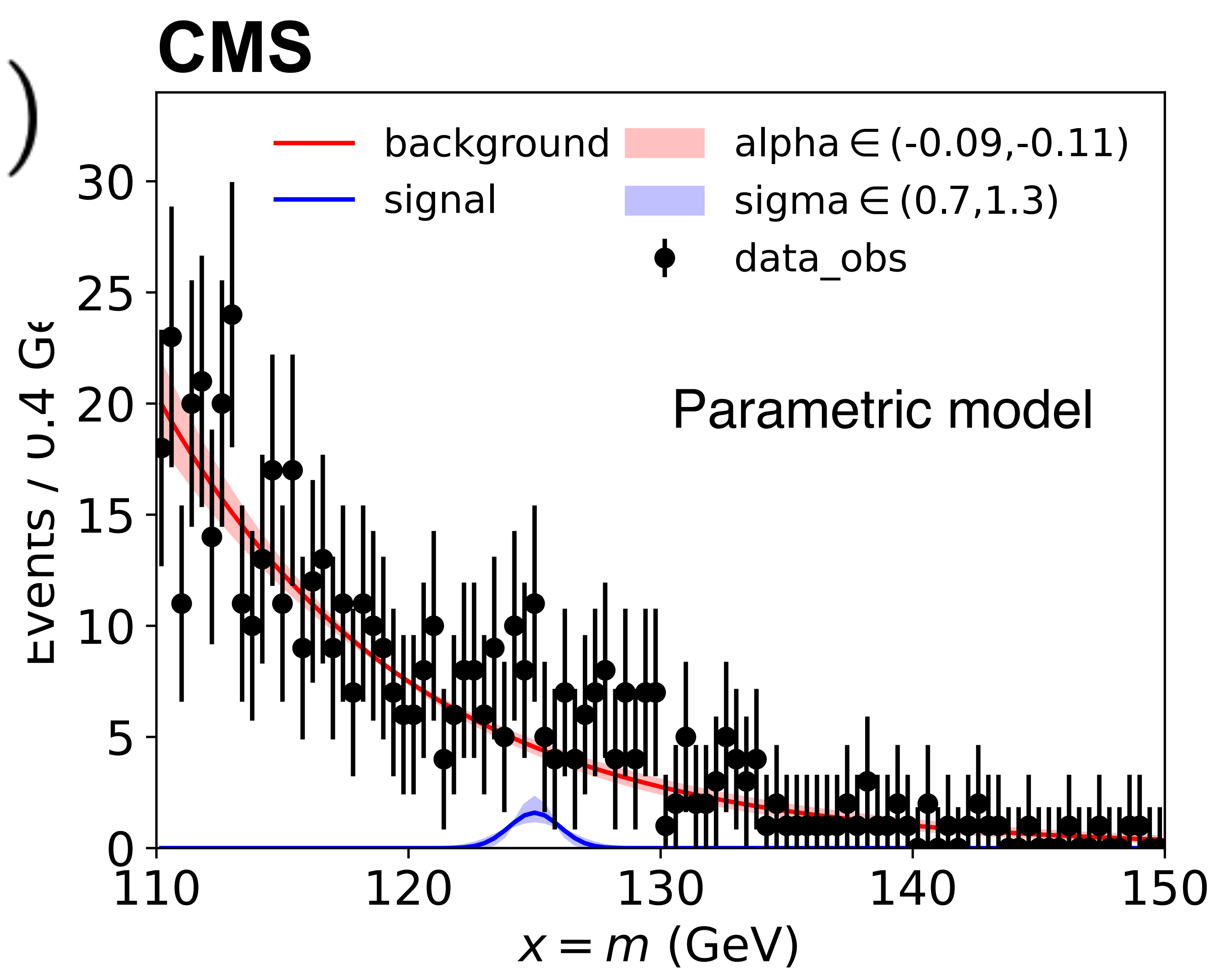
expectations decompose into signal and background contributions



Supported models

$$L(\vec{\mu};\vec{\nu}) = \prod_n p(x_n; \sum_i \mu_i S_{i,n}(\vec{\nu}) + \sum_k B_k(\vec{\nu})) \cdot \prod_i p(y_i;\nu_i)$$

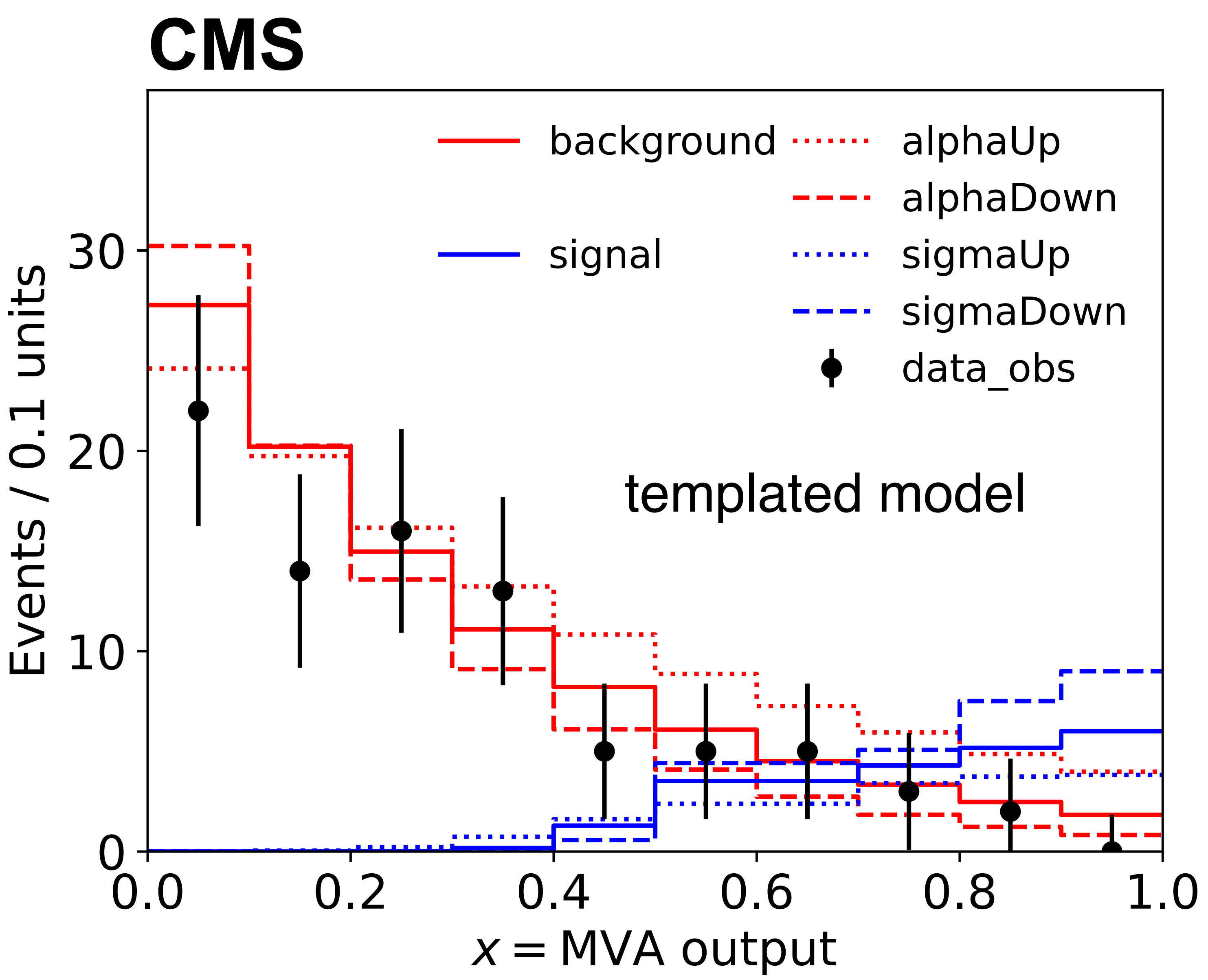
$$p(\vec{x};\vec{\mu},\vec{\nu}) = \sum_p \frac{\lambda_p(\vec{\mu},\vec{\nu}) f_p(x;\vec{\mu},\vec{\nu})}{\sum_p \lambda_p(\vec{\mu},\vec{\nu})}$$



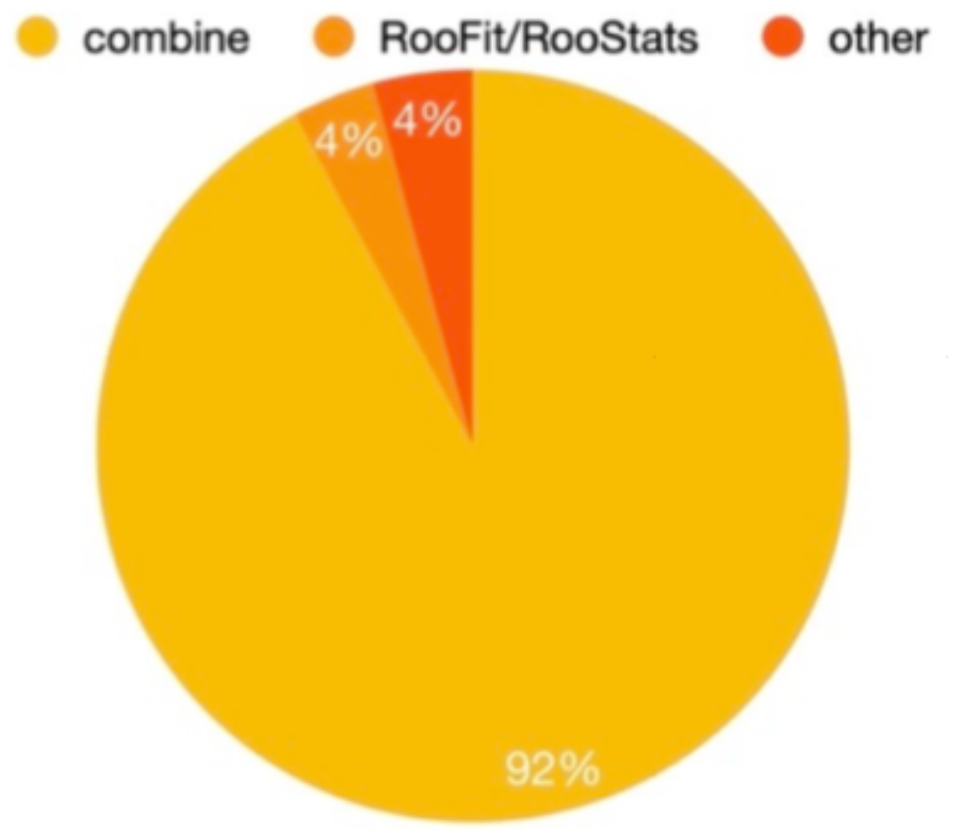
Statistical models can be both parametric (unbinned), binned, or simple counting analyses.

Different statistical models can be combined in the same likelihood function

$$p(x;\vec{\mu},\vec{\nu}) = \prod_{b=1}^{N_B} \mathcal{P}\left(n_b; \lambda_b(\vec{\mu},\vec{\nu})\right)$$



Combine



From Statistics Committee Questionnaires 2021-2022

Combine is the most used statistical analysis software in CMS

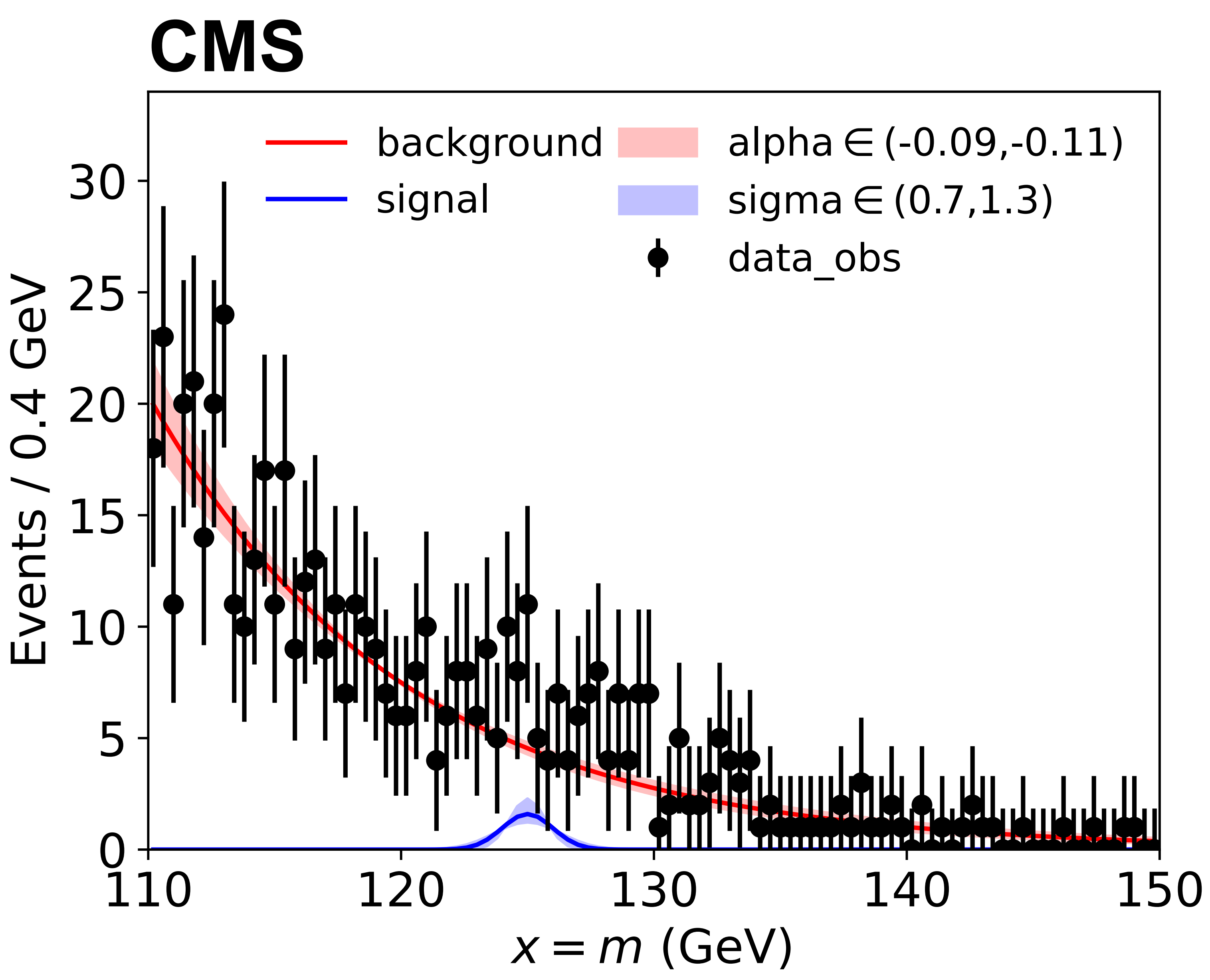
built around ROOT, RooFit and RooStats:
• Encapsulate the statistical model in a human-readable configuration file, called the ***datacard***.
• Builds pre-defined statistical models: counting, parametric unbinned and binned, template-based
• Statistical models are flexible and can be custom-designed, allowing a wide range of relations between POIs and input PDFs
• Command-line interface to RooFit/RooStats methods.
• Powerful for combinations, scales well with model complexity
• Provides workflow for statistical procedures recommended by the CMS Statistics Committee
• Provides an extensive toolset for validation
• Supported with extensive documentation and tutorials: https://cms-analysis.github.io/HiggsAnalysis-CombinedLimit/latest/
• Available as docker container for cross-platform usage

Described in Comput.Softw.Big Sci. 8 (2024) 1, 19

The datacard

Combine specifies the construction of the model through **datacards**

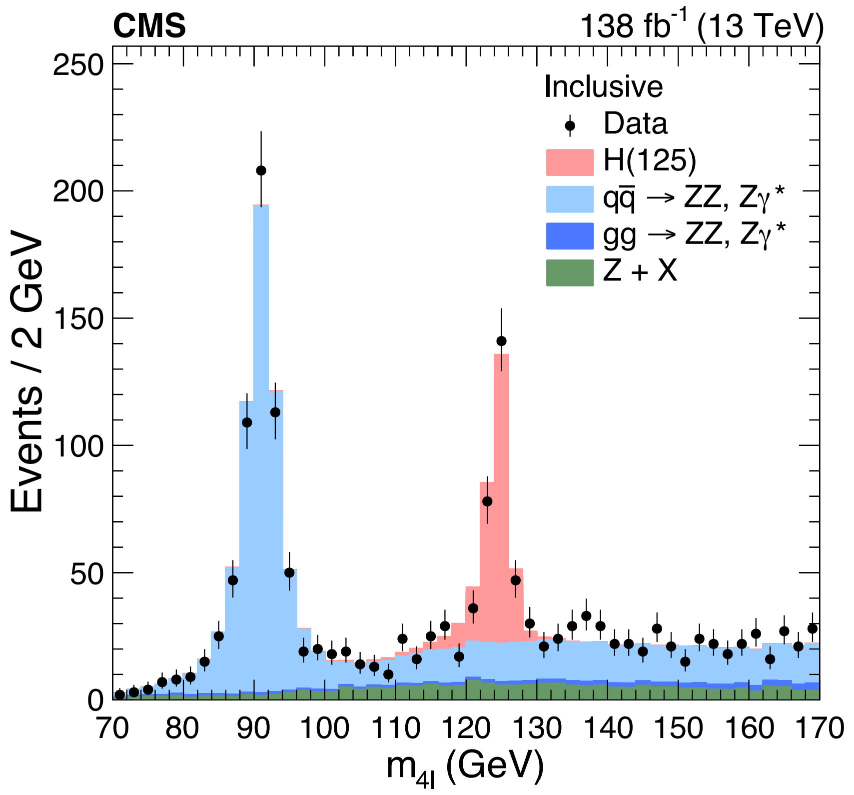
The datacards describes the content of the plots we publish in our papers, including informations on yields and systematic uncertainties

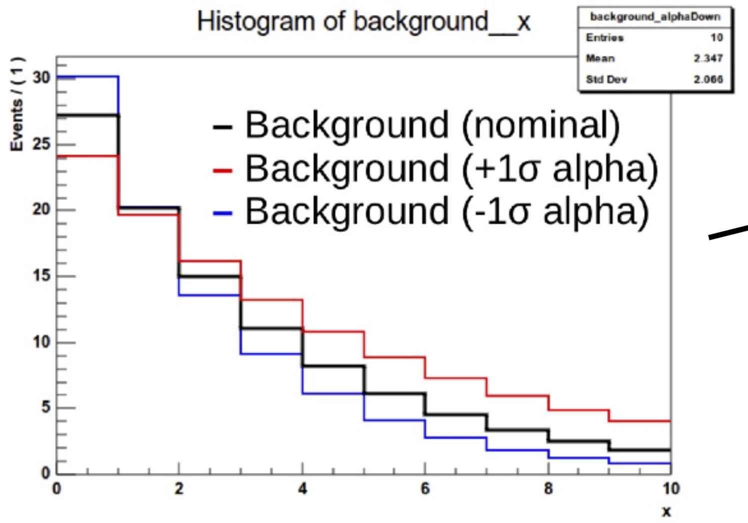



```
imax 1
jmax 1
kmax 2
# ---------------
shapes data_obs   bin1 parametric-analysis-datacard-input.root w:data_obs
shapes signal     bin1 parametric-analysis-datacard-input.root w:sig
shapes background bin1 parametric-analysis-datacard-input.root w:bkg
# ---------------
bin                 bin1
observation         567
# ---------------
bin                 bin1   bin1
process             signal background
process             0      1
rate                10     1
# ---------------
lumi         lnN   1.1     -
sigma        param 1.0     0.1
alpha        flatParam
bkg_norm     flatParam
```

Object name	Type	Description
m	RooRealVar	The invariant mass observable.
data_obs	RooDataSet	Invariant mass of each event in the observed data.
sig	RooGaussian	Normal pdf describing the probability distribution of the invariant mass for the signal process.
bkg	RooExponential	Exponential pdf describing the probability distribution of the invariant mass for the background process.
MH	RooRealVar	Mean of the signal pdf.
sigma	RooRealVar	Standard deviation of the signal pdf.
alpha	RooRealVar	Slope parameter for the background pdf.
bkg_norm	RooRealVar	Rate multiplier for the total background contribution.

What is in a Combine model

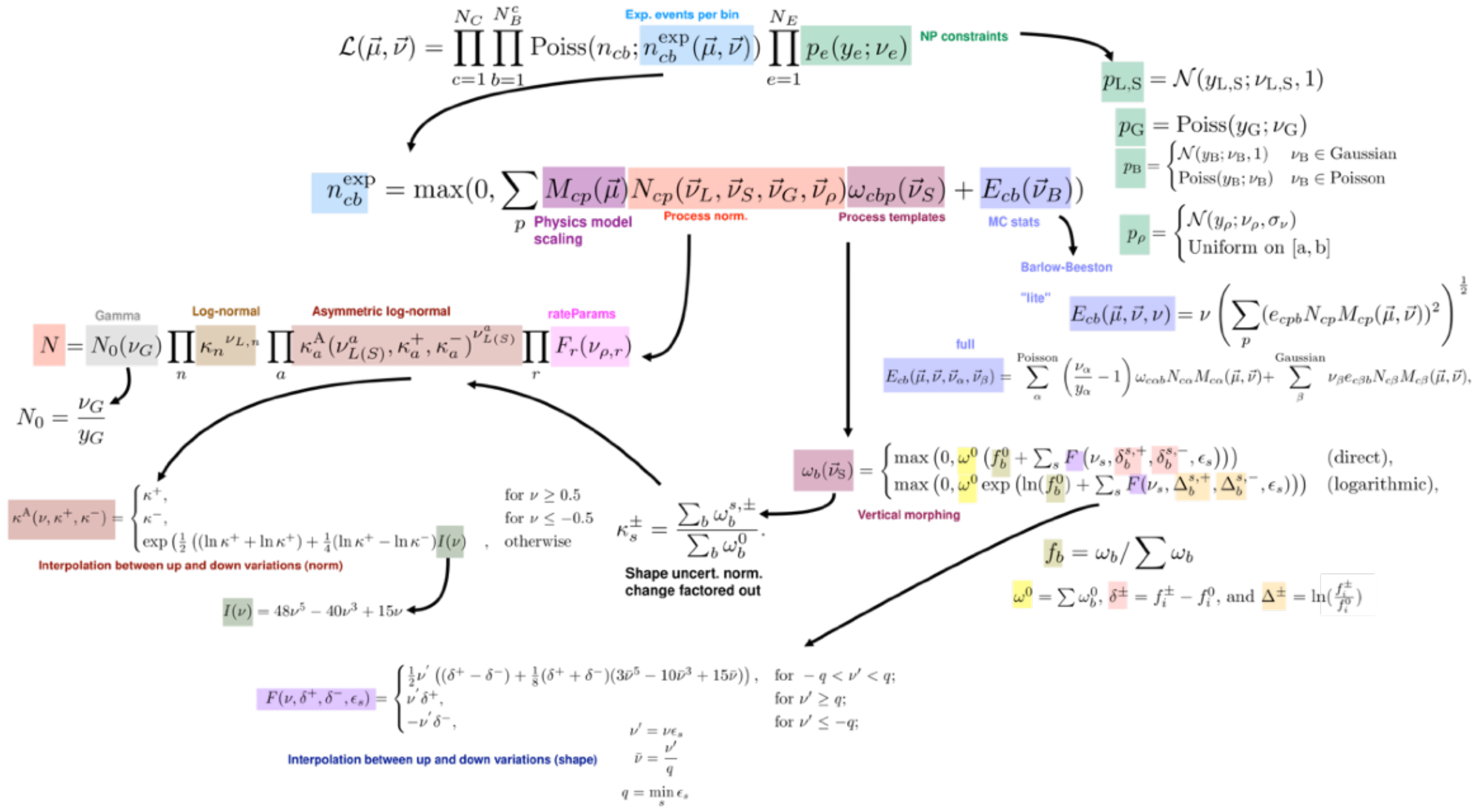




Inputs:
Signal and background distributions
Systematic uncertainties
(described in the datacard)

Combine provides several methods for parameter estimation, limit settings, likelihood maximisation etc..

likelihood function

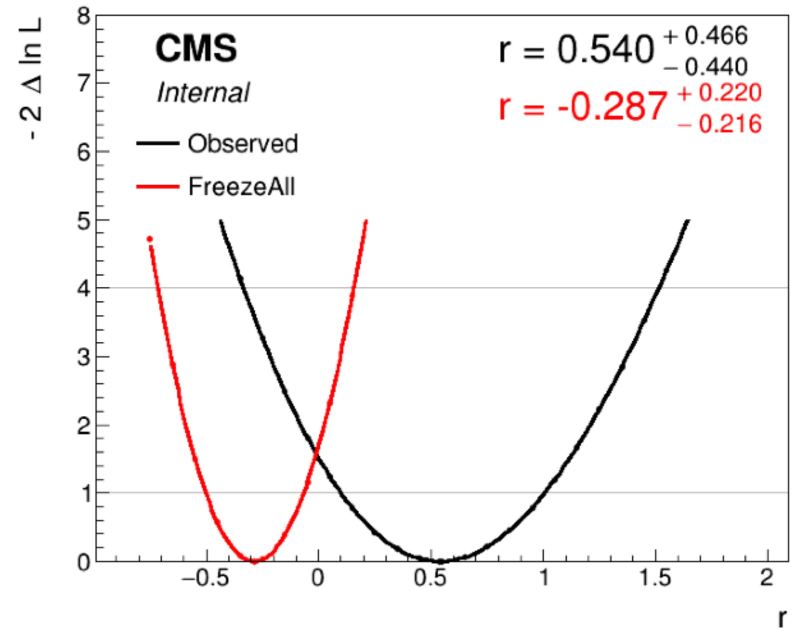


$$p(x_n; \sum_i \mu_i S_{i,n}(\vec{\nu}) + \sum_k B_k(\vec{\nu}))$$

description of the Physics model

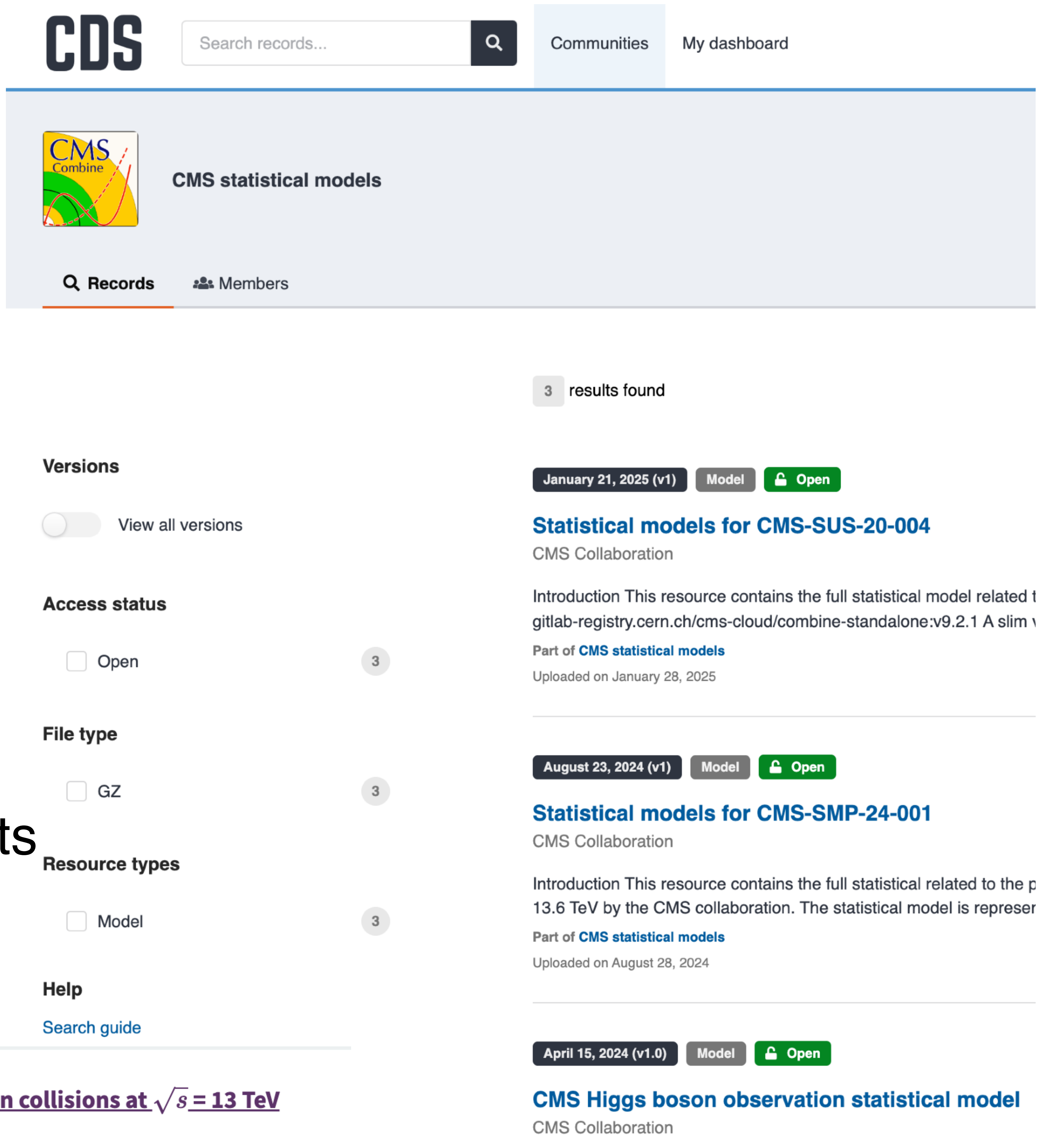
Statistical procedure

Results



Where to find the CMS statistical models

- CMS will publish its statistical models on CDS under a CC4.0 licence: https://repository.cern/communities/cms-statistical-models/records?q=&l=list&p=1&s=10&sort=newest
- HEPData publication entries will also link to the statistical model whenever a model is available



What you will find in the statistical model webpage:

- An introduction to the results
- Which combine software version was used
- The commands used to produce the results
- The datacards with the human-readable description of the inputs
- A table with the systematic uncertainties and their descriptions.
- Any auxiliary material needed to reproduce a result

(Re)interpreting: how to use the models

Combine allows for “easy” re-interpretation of previous results, such as:

- Changing/freezing POIs within the model parameters. For example promoting the Higgs mass to be a POI. (—freezeParameters, —redefineSignalPOIs)
- Changing the behaviour of some nuisances (—freezeParameters)
- Adding multiplicative factor to process yields (rateParam)

- General case: you can rebuild the likelihood using a different Physics Model. Either already present in Combine or a custom made one

```
    def getHiggsSignalYieldScale(self, production, decay, energy):
        name = "c7_XSBRscal_%s_%s_%s" % (production, decay, energy)
        if self.modelBuilder.out.function(name) == None:
            if production in ["ggH", "qqH", "ggZH", "tHq", "tHW"]:
                XSscal = ("@0", "Scaling_%s_%s" % (production, energy))
            elif production == "WH":
                XSscal = ("@0*@0", self.kappa_W)
            elif production == "ZH":
                XSscal = ("@0*@0", self.kappa_Z)
            elif production == "ttH":
                XSscal = ("@0*@0", "kappa_t")
            elif production == "bbH":
                XSscal = ("@0*@0", "kappa_b")
            else:
                raise RuntimeError("Production %s not supported" % production)
            BRscal = decay
            if not self.modelBuilder.out.function("c7_BRscal_" + BRscal):
                raise RuntimeError("Decay mode %s not supported" % decay)
            if decay == "hss":
                BRscal = "hbb"
            if production == "ggH" and (decay in self.add_bbH) and energy in ["7TeV", "8TeV", "13TeV", "14TeV"]:
                b2g = "CMS_R_bbH_ggH_%s_%s[%g]" % (decay, energy, 0.01)
                b2gs = "CMS_bbH_scaler_%s" % energy
                self.modelBuilder.factory_(
                    'expr::%s("(%s + @1*@1*@2*@3)*@4", %s, kappa_b, %s, %s, c7_BRscal_%s)' % (name, XSscal[0], XSscal[1], b2g, b2gs, BRscal
                )
            else:
                self.modelBuilder.factory_('expr::%s("%s*@1", %s, c7_BRscal_%s)' % (name, XSscal[0], XSscal[1], BRscal))
            print("[LHC-HCG Kappas]", name, production, decay, energy, ": ", end=" ")
            self.modelBuilder.out.function(name).Print("")
        return name
```

```
text2workspace.py -P HiggsAnalysis.CombinedLimit.HiggsCouplings_ICHEP12:cVcF 125.5/comb.txt -m 125.5 -o comb_kVkF.root
combine comb_kVkF.root -m 125.5 -M MultiDimFit --algo singles
```

```
 --- MultiDimFit ---
best fit parameter values and profile-likelihood uncertainties:
   CV :    +0.946   -0.120/+0.113 (68%)
   CF :    +0.497   -0.170/+0.203 (68%)
Done in 3.09 min (cpu), 3.09 min (real)
```

combination of different results

The description of nuisance parameters is also meant to help in combinations

It is possible to combine any 2 analyses, as long as they are sensitive to the same POIs

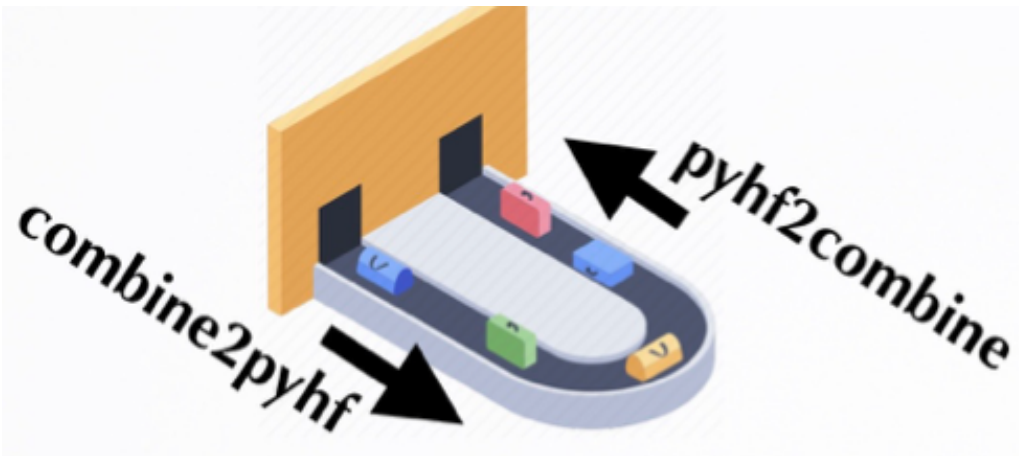
Caveats:

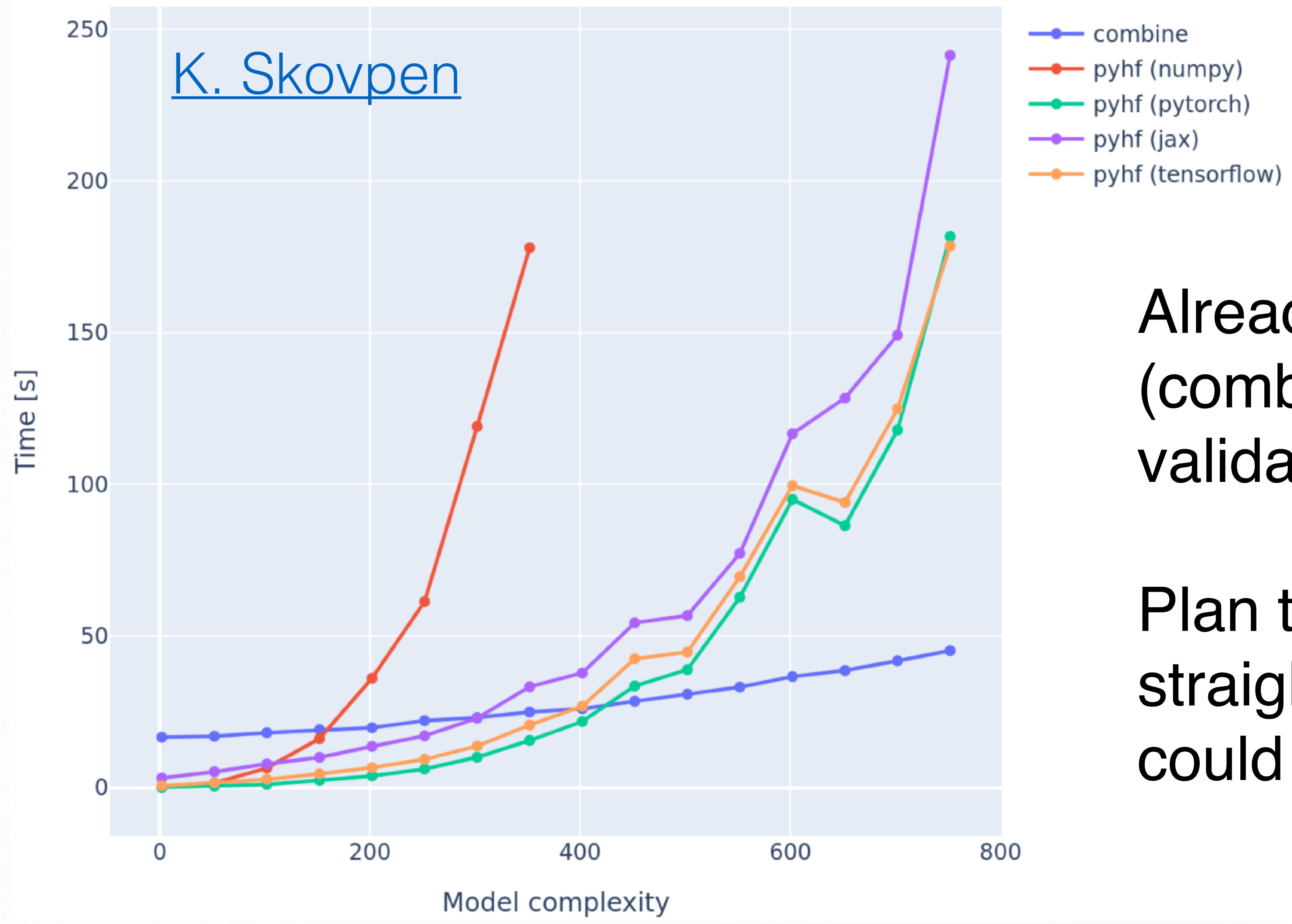
- Analyses phase spaces should be mutually exclusive to avoid double-counting
- No partial correlation of nuisances. Nuisance parameters are either fully correlated (same name) or uncorrelated (different names)

	class	description
CMS_eff_b	btag	efficiency uncertainty for standard b jets for all years.
CMS_vhbb_statZjLF_Wenu2_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Wenu_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Wmunu	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Wmunu2	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Wmunu2_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Wmunu_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_ZnunuHighPt_8TeV	custom	uncertainties due to the limited MC simulation size in 0 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_ZnunuLowPt_8TeV	custom	uncertainties due to the limited MC simulation size in 0 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Znunu_1_7TeV	custom	uncertainties due to the limited MC simulation size in 0 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Znunu_2_7TeV	custom	uncertainties due to the limited MC simulation size in 0 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Wenu2	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statsTop_Wenu	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statsTop_Wenu2_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statsTop_Wenu_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statsTop_Wmunu	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statsTop_Wmunu2	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statsTop_Wmunu2_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statsTop_Wmunu_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_stats_TT_ZeeHighPt_8TeV	custom	uncertainties due to the limited MC simulation size in 2 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_stats_TT_ZeeLoose_7TeV	custom	uncertainties due to the limited MC simulation size in 2 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_stats_TT_ZeeLowPt_8TeV	custom	uncertainties due to the limited MC simulation size in 2 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_stats_TT_ZeeTight_7TeV	custom	uncertainties due to the limited MC simulation size in 2 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statsTop_Wenu2	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_stats_TT_ZmmHighPt_8TeV	custom	uncertainties due to the limited MC simulation size in 2 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjLF_Wenu	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statZjHF_Znunu_1_7TeV	custom	uncertainties due to the limited MC simulation size in 0 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statWjHF_Znunu_2_7TeV	custom	uncertainties due to the limited MC simulation size in 0 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statWjLF_Wenu	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statWjLF_Wenu2_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statWjLF_Wenu_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statWjLF_Wmunu	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statWjLF_Wmunu2	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty
CMS_vhbb_statWjLF_Wmunu2_8TeV	custom	uncertainties due to the limited MC simulation size in 1 lepton VH(bb) channel, implemented as a shape uncertainty

Compatibility with other frameworks

Combine is fully based on ROOT/RooFit/RooStats. Should be able to interact with other RooStats-based frameworks





Already available: Combine<->pyHF conversion tool (combine2pyhf and pyhf2combine). Developed and extensively validated in the context of the ATLAS+CMS tttt EFT combination

Plan to apply HS3 standards in the next versions of combine. Not straightforward due to some custom classes in Combine, but they could be ported into ROOT.

Combine performances are powerful, especially for complex models. CMS Higgs combination in 2022 (Nature 607 (2022) 60-68) had 900 categories, STXS1.2 POI +EFT, 8000+ NP in total.

- 16GB+ to build the likelihood model, 10GB+ to perform the fit, Runs in 24-48 hours!

When will you find the models: CMS plans

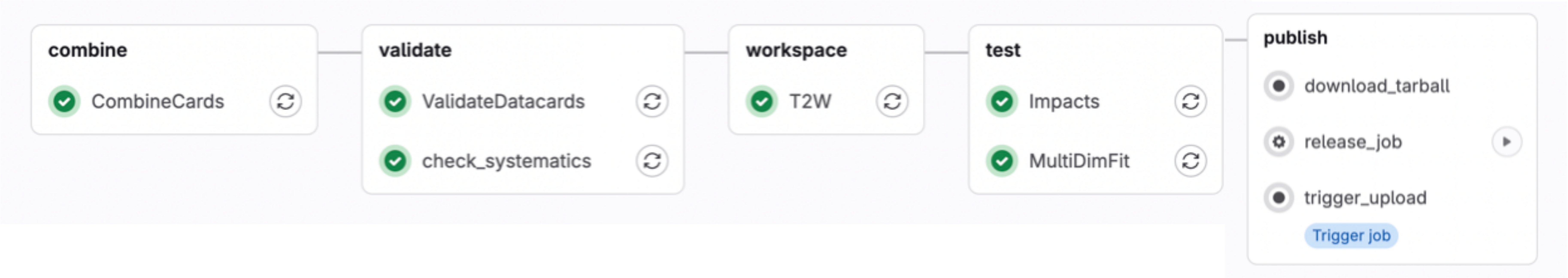
At the moment, only a handful of statistical model are available on CDS

CMS official strategy is that by default all analyses will publish their statistical model

We are in the transition/ramp-up period. There is some inertia from when a strategy is decided to when is implemented, and different physics group will be ready on different timescales.

We do not plan to release models for older, already published, analyses. Although motivated exceptions are possible

We are implementing tools to make the whole procedure as automated, standardised and easy as possible, taking as much advantage as possible from CI systems. Hopefully this will help in having statistical model published quickly



Conclusions

CMS released the first statistical models implemented in Combine and is in process of releasing more statistical models.

- Developed a standard pipeline to increase publication efficiency
- Standardised naming of systematic uncertainty to facilitate publication
- For models with several BSM signals, only a subset of signal hypotheses will be published with instructions on how to interpolate over the model phase space

Combine tool is public: documentation + standalone container

- Self-documenting statistical model building
- Extensive toolset for statistical inference
- Constantly improving documentation, ensuring compatibility with the latest ROOT versions

Working towards compatibility with other formats:

- Combine <—> pyHF conversion already available and validated.
- Work started to implement HS3 (HEP Statistics Serialization Standard).