

# StatTestCalculator: a new general tool for statistical analysis in high energy physics

Oleg Vasilevskii<sup>1,2</sup>, Daniil Gorin<sup>1,2</sup>, Lev Dudko<sup>1,2</sup>, Emil Abasov<sup>1,2</sup>

1. Moscow State University, Faculty of Physics
2. Skobeltsyn Institute of Nuclear Physics, M.V. Lomonosov Moscow State University

# Introduction

Experiments in high energy physics always require the statistical analysis of data obtained since it has probabilistic nature.

Parameters and expected significance require complex statistical models including systematic uncertainties to be calculated accurately.

To perform these calculations one needs a profound statistical tool.

Many different tools and methods already exist such as:

Roofit and RooStats packages [1,2], Combine[3], Theta[4] and others.

Existing tools require complex setup (ROOT/CMSSW), datacards and C++/Roofit/RooStats plugins.

A new lightweighted potentially Python-based tool is needed for quick analysis and estimations which can be used for theoretical and phenomenological researches.

[1]Verkerke, Wouter, and David Kirkby. "The RooFit toolkit for data modeling." *Statistical Problems in Particle Physics, Astrophysics and Cosmology*. 2006. 186-189.

[2]Moneta L. et al. The roostats project //arXiv preprint arXiv:1009.1003. – 2010.

[3]CMS collaboration. "The CMS statistical analysis and combination tool: COMBINE." *arXiv preprint arXiv:2404.06614* (2024).

[4] Tóth, Tamás, et al. "Theta: a framework for abstraction refinement-based model checking." *2017 Formal Methods in Computer Aided Design (FMCAD)*. IEEE, 2017.

# StatTestCalculator

In this work we announce a new lightweighted software for statistical analysis - StatTestCalculator (STC). The STC functional ability allows the following:

1. Estimate the expected significance and upper limits using asymptotic formulae.
2. Calculate the exact significance and upper limits using Monte-Carlo simulations of test statistics.
3. Use normal or lognormal distributed systematic uncertainties for both signal and background events.
4. Define whether to use correlated or uncorrelated systematic uncertainty.
5. Implement user's own test statistics, formulae or systematic uncertainty distributions.
6. Plot the data, signal and background distributions.
7. Plot the resulting distributions of Monte-Carlo simulated experiments and calculations

This functions can be applied for both counting experiment data and histograms analyses.

Since the software is built on Python, it is simple to understand the functionality of the code and easy to use. The tool can be directly integrated into Jupyter notebooks and python scripts to complete a neural network pipeline.

In this report we will explore the basic concepts of statistical analysis, mathematical foundation of the methods, dive into the functionalities of the tool and investigate the applicability.

# Statistical analysis

The basic concept of statistical analysis of high energy physics data is hypothesis testing.

The data can be a distribution collected from an experiment (fig.1) or a neural network output discriminator (fig.2)

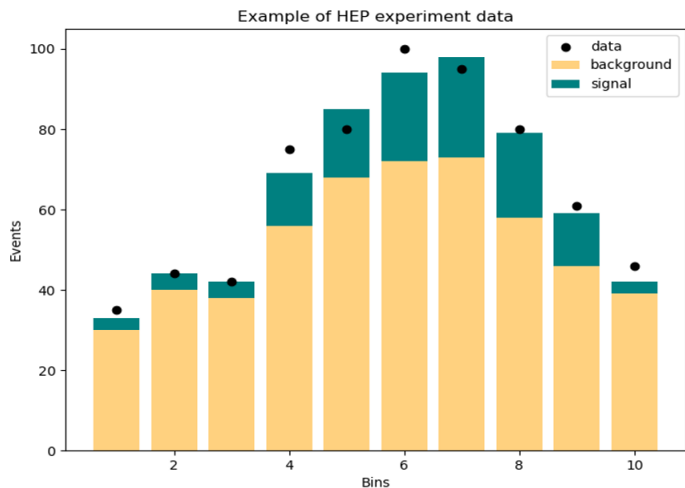


Fig.1. example of experimental data.  
Black dots – number of observed events in each bin,  
Blue – number of expected signal events,  
yellow – number of expected background events

The hypotheses can be denoted as:

Background-only (null) hypothesis:

$H_0$  – data does not contain any signal events

Singal + background (alternative) hypothesis:

$H_1$  – data contains both background and signal events

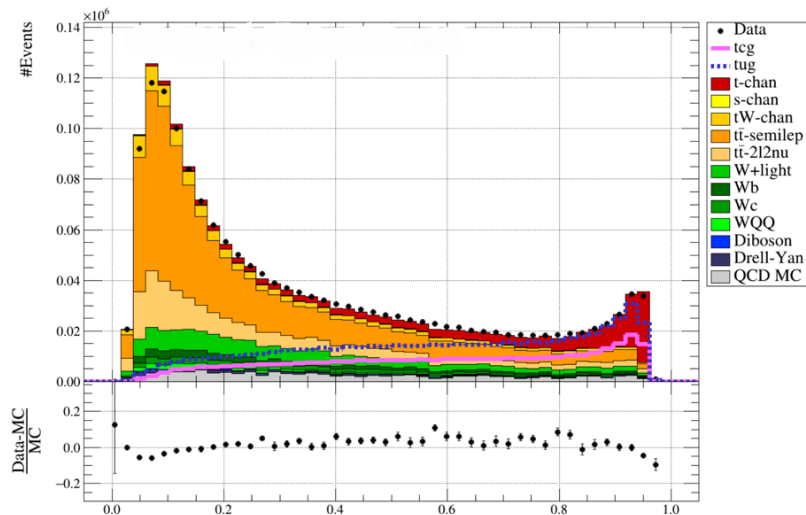


Fig. 2. Example of neural network output discriminator.  
Black dots – number of observed events,  
Bars – background events  
Dotted line – signal events

# Statistical analysis and hypothesis test

$H_0: n_i = \tau_i b_i$  - no signal is present in data(1)

$H_1: n_i = \mu s_i + \tau_i b_i$  - signal is present in data (2)

Where  $s_i, b_i$  – number of signal and background events in each bin.

$\tau = \frac{b_{exp}}{b_{obs}}, \mu = \frac{s_{exp}}{s_{obs}}$  - parameters of strength of background and signal respectively.

Likelihood under the assumption of a hypothesis:

$$L = L(H|H = H_i)$$

Consider each bin in data histogram has  $n_i$  events. Under an assumption of hypotheses.(1,2)

Here  $\tau$  represents the nuisance parameter and used for accounting of systematic uncertainty,  $\mu$  – parameter of interest which is depended on cross-section of signal process.

One can set upper limits on  $\mu$  or calculate an expected significance of discovery of a new process.

For that one should find the hypothesis that is best consistent with the data.

It can be done with test of a hypotheses:

**$H_1$  vs.  $H_0$**  for calculating expected significance.

**$H_0$  vs.  $H_1$**  for setting upper limits.

# Mathematical foundation for hypothesis tests

Consider the statistical model of the data obtained:

$$L(\mu, \tau(\vec{\theta}), \vec{\theta}) = \prod_{i=1}^N \text{Poisson}(\mu s_i + \tau_i b_i) \prod_{j=1}^M \text{Systematic}(\vec{\theta})$$

Where *Systematic* is the distribution of systematic uncertainty and  $\vec{\theta}$  represents the vector of nuisance parameters.

The general statistic for hypothesis test can be denoted as[5]:  $\lambda = \frac{L(\mu, \hat{\vec{\theta}})}{L(\hat{\mu}, \hat{\vec{\theta}})}$

From this two statistics can be derived to be used for calculating expected discovery significance and upper limits respectfully[5]:

$$q_0 = \begin{cases} -2 \ln \lambda(0) , & \hat{\mu} \geq 0 \\ 0 , & \hat{\mu} < 0 \end{cases} \quad q_\mu = \begin{cases} -2 \ln \lambda(\mu) , & \hat{\mu} \leq \mu \\ 0 , & \hat{\mu} > \mu \end{cases} ,$$

# Mathematical foundation for asymptotic formulae

Using these statistics under the assumption of Asimov dataset, the following conclusions can be made[5]:

$$Z_0 = \Phi^{-1}(1 - p_0) = \sqrt{q_0}.$$

$$Z_\mu = \Phi^{-1}(1 - p_\mu) = \sqrt{q_\mu}.$$

And the asymptotic formulae for significance of the discovery and upper limits both including and excluding systematic uncertainty can be derived[6]:

$$Z_{disc} = \sqrt{2 \left[ (s+b) \ln \left( \frac{(s+b)(1+\delta^2 b)}{b+\delta^2 b(s+b)} \right) - \frac{1}{\delta^2} \ln \left( 1 + \delta^2 \frac{s}{1+\delta^2 b} \right) \right]}. \quad Z_{disc} = \sqrt{2 \left[ (s+b) \ln \left( 1 + \frac{s}{b} \right) - s \right]}.$$

$$Z_{excl} = \sqrt{2 \left[ s - b \ln \left( \frac{b+s+x}{2b} \right) - \frac{1}{\delta^2} \ln \left( \frac{b-s+x}{2b} \right) \right] - (b+s-x) \left( 1 + \frac{1}{\delta^2 b} \right)}. \quad Z_{excl} = \sqrt{2 \left[ s - b \ln \left( 1 + \frac{s}{b} \right) \right]}.$$

Where  $\delta$  – complete relative uncertainty of the background events

[5] Cowan, Glen, et al. "Asymptotic formulae for likelihood-based tests of new physics." *The European Physical Journal C* 71 (2011): 1-19.

[6] Асимптотические формулы для оценки статистической значимости в коллайдерных экспериментах. Горин Д.Э., Василевский О.С., Дудко Л.В., Абасов Э.Э. *Ученые записки физического факультета Московского Университета*, № 1, с. 1-12, 2024.

# Monte-Carlo hypothesis test using StatTestCalculator

Consider the background events with systematic uncertainty having the distribution in general case:

$$\mathbf{b} \sim \mathbf{b} \times \mathit{Systematic}(\vec{\theta})$$

In most cases *Systematic* is either normal or lognormal distribution of nuisance parameters.

The simple way to consider all the nuisance parameters at once is to use the background strength parameter  $\tau$ :

$$\tau \sim \mathit{Normal}(\mu = 1, \sigma = \delta) \text{ or } \tau \sim \mathit{Lognormal}(\mu' = 1, \sigma' = \delta)$$

where  $\mu' = e^{\left(\mu + \frac{\sigma^2}{2}\right)}$  and  $\sigma' = \left(e^{\sigma^2} - 1\right) e^{(2\mu + \sigma^2)}$  the parameters of lognormal distributions.

# Monte-Carlo hypothesis test using StatTestCalculator

Thus, the final likelihood for each Monte-Carlo simulated experiment would be:

$$L(\mu, \vec{b}, \vec{\tau}) = \prod_{i=1}^N \frac{(\mu s_i + \tau_i b_i)^{n_i}}{n_i!} e^{-(\mu s_i + \tau_i b_i)} \prod_{i=1}^N \text{Systematic}(\mu, \sigma)$$

Where  $\tau$  is distributed according to a systematic distributions (Normal or Lognormal):

$$\tau \sim \text{Systematic} = \frac{1}{\sqrt{2\pi}\delta} e^{-\frac{(x-1)^2}{2\delta^2}} - \text{normal distribution or}$$

$$\tau \sim \text{Systematic} = \frac{e^{-\left(\frac{[\ln(x)-1]^2}{\delta^2}\right)}}{x\delta\sqrt{2\pi}} - \text{lognormal distribution}$$

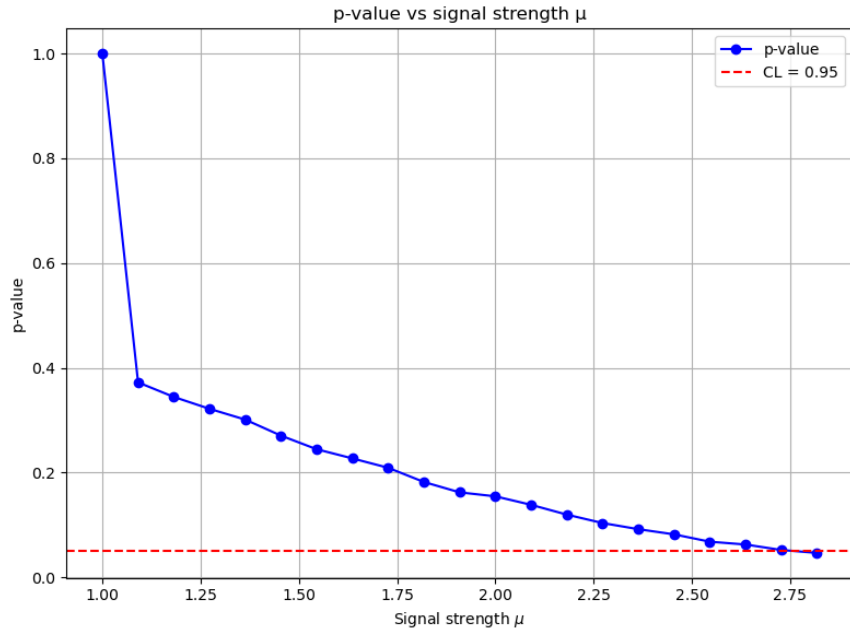
If the systematic uncertainty is considered to be correlated, the likelihood is simplified to:

$$L(\mu, \vec{b}, \tau) = \prod_{i=1}^N \frac{(\mu s_i + \tau b_i)^{n_i}}{n_i!} e^{-(\mu s_i + \tau b_i)} \times \text{Systematic}(\mu, \sigma)$$

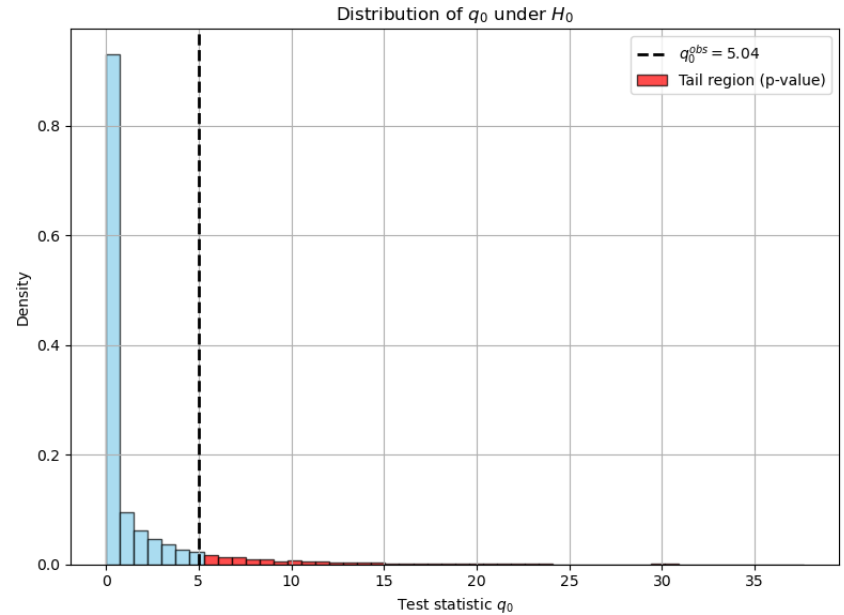
If the systematic uncertainty is considered to be neglectable, then the likelihood simplifies to:

$$L(\mu, \mathbf{b}, \tau = 1) = \prod_{i=1}^N \frac{(\mu s_i + b_i)^{n_i}}{n_i!} e^{-(\mu s_i + b_i)}$$

# Monte-Carlo hypothesis test using StatTestCalculator



The convergence of  $\mu$  under Monte-Carlo calculations



The distribution of test statistic after Monte-Carlo simulations

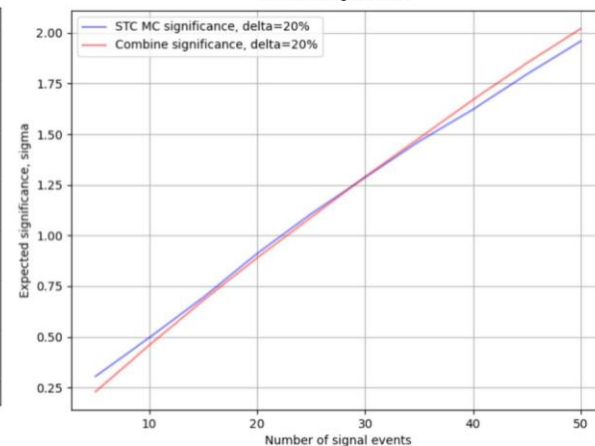
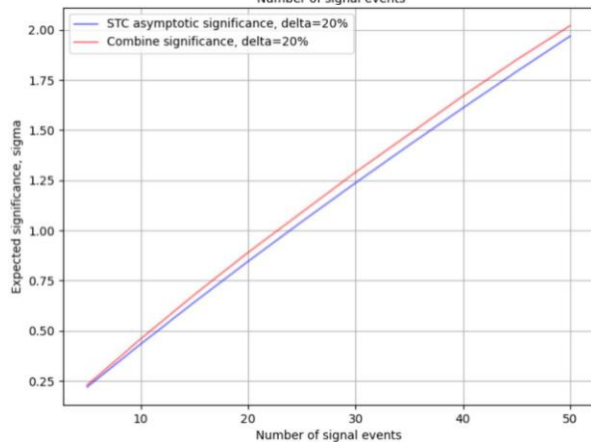
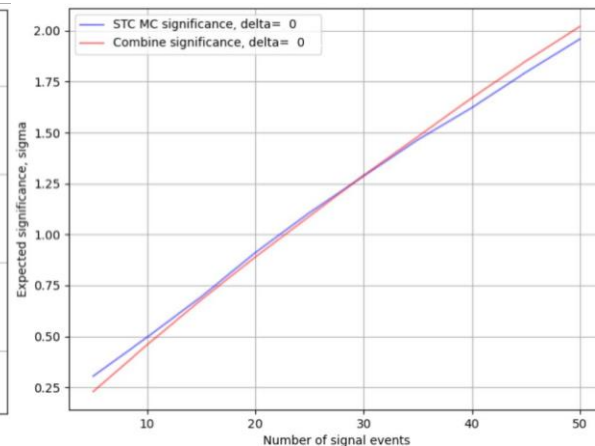
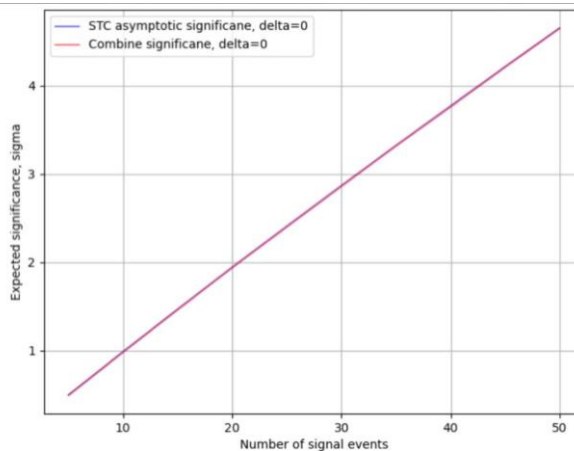
# Comparison of asymptotic, MC and reference method calculations

In order to check an accuracy of STC calculations we can set up a counting toy experiment and make calculations for different number of signal events.

Here the comparison of calculated expected significance of discovery obtained from STC and Combine as reference method is shown.

Each toy background consists of 100 events.

Each MC calculation consists of 10000 toy experiments.

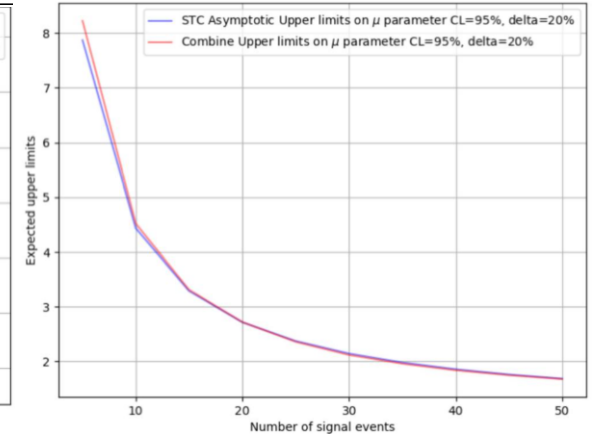
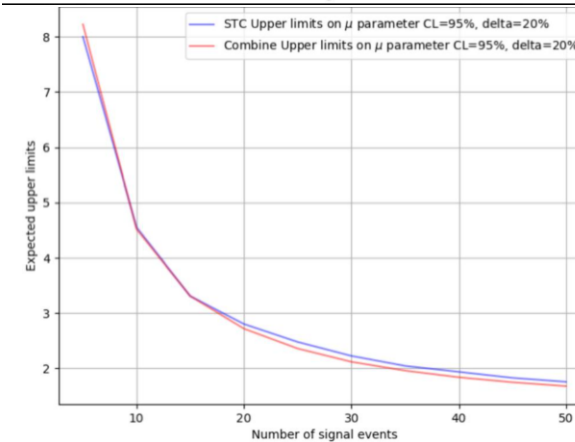
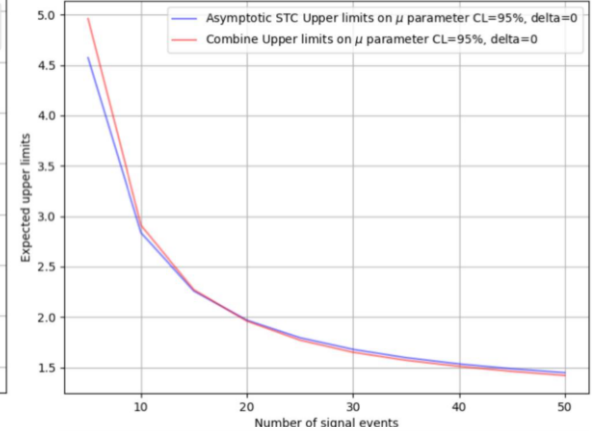
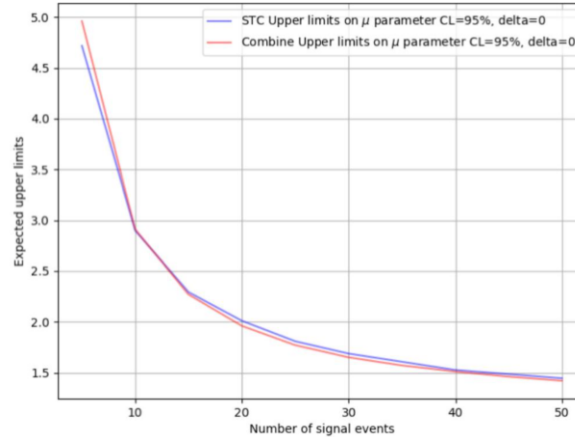


# Comparison of asymptotic, MC and reference method calculations

In order to check an accuracy of STC calculations we can set up a counting toy experiment and make calculations for different number of signal events.

Here the comparison of set upper limits obtained from STC and Combine as reference method is shown.

Each toy background consists of 100 events.  
Each MC calculation consists of 10000 toy experiments.



# Comparison of Combine, Theta, StatTestCalculator

Feature	Combine	Theta-framework	StatTestCalculator
Implementation	C++(ROOT/RooFit/RooStats) with Python/CLI glue	C++ core with Python bindings	Pure Python
Supported Methods	Asymptotic, Monte-Carlo, Bayesian MCMC, Feldman–Cousins	Asymptotic, frequentist Monte-Carlo, Bayesian	Asymptotic, frequentist and hybrid Monte-Carlo
Systematics & uncertainty	Supports any nuisance parameters and shaped systematics, correlations	Shaped systematics, nuisance parameters and correlations	General background systematics and correlations
Flexibility & variability	Implementing new systematics or distributions via RooFit & RooStats	Implementing new systematics via C++ plugins or custom distributions	Implementing any statistics, distributions or systematics via Python callable functions
Pre-requisites	C++, ROOT, CMSSW, datacards, documentation	C++, ROOT, BOOST/GSL, documentation	Python, simple documentation
Can be used for:	Complete exact analysis of full data	Complete exact analysis of full data	Complete analysis, quick estimations and neural networks pipelines

# Conclusion

The new general tool for statistical analysis in high energy physics is presented. This tool allows user to run complete statistical analysis including calculations, plotting the results or conduct quick estimations of the expected significance or upper limits.

StatTestCalculator possesses a vast functionality and variety for different analyses both general and specific. The variety of the class allows user to use custom models, uncertainties, distributions and formulae to set the method exactly to satisfy the very specific needs.

The tool, its description and tutorial is available on GitHub:

<https://github.com/skottver/stattestcalculator>