MadAnalysis 5 v1.8
Normal-Mode reference card

July 24, 2019

website: `http://madanalysis.irmp.ucl.ac.be/`

references: arXiv:1206.1599[hep-ph], arXiv:1405.3982[hep-ph], arXiv:1407.3278[hep-ph]

1 Starting the MadAnalysis 5 interpreter

The MadAnalysis 5 command line interface can be started by typing in a shell

```
bin/ma5 [options] [script]
```

where the potential options (`[options]`) are given in the table below.

Short	Full	Description
`-P`	`--partonlevel`	Parton-level mode.
`-H`	`--hadronlevel`	Hadron-level mode.
`-R`	`--recolevel`	Reconstructed-level mode.
`-E`	`--expert`	Expert mode.
`-v`	`--version`	Displays the current MadAnalysis 5 version number.
	`--release`	
`-b`	`--build`	Builds of the SampleAnalyzer library.
`-f`	`--forced`	Skips MadAnalysis 5 confirmation messages.
`-s`	`--script`	Executes a script containing all analysis commands and exits the program. The file containing the script has to be provided as `[script]`. Removing the `-s` option prevents MadAnalysis 5 from exiting.
`-h`	`--help`	Print all the available commands.
`-i`	`--installcard`	Generates a file with information on the MadAnalysis 5 dependencies, `installation_card.dat`, that is located in the `madanalysis/input` folder.
`-d`	`--debug`	Debug mode.

In case of installation issues, the usage of the dependencies can be tuned by editing the file: `madanalysis/input/installation_options.dat`
More information can be found in the original MadAnalysis 5 manual [?].

2 Installation of optional packages

MadAnalysis 5 is interfaced to several high-energy physics packages and can be linked to a variety of external plugins. Any of those can be installed by typing in the MadAnalysis 5 interpreter,

```
install [package]
```

where the different choices for `[package]` are given in the table below.

Package	Description
`PAD`	The MadAnalysis 5 Public Analysis Database of recasted LHC analyses relying on Delphes 3 for the simulation of the detector effects.
`PADforMA5tune`	The MadAnalysis 5 Public Analysis Database of recasted LHC analyses relying on the old tuned version of Delphes 3 for the simulation of the detector effects.
`delphes`	The current release of Delphes 3.
`delphesMA5tune`	The old tuned version of Delphes 3.
`fastjet`	The FastJet and FastJetContrib packages.
`samples`	Some test Monte Carlo samples.
`zlib`	The ZLib library allowing to handle gzipped compressed files.

3 Commands available from the MadAnalysis 5 interpreter

A small set of commands, to be typed from the MadAnalysis 5 interpreter, are related to console actions.

Command	Description
`quit` `EOF`	Exits MadAnalysis 5.
`help`	Displays the list of available commands.
`help [command]`	Displays details about a specific command.
`history`	Displays the history of all commands that have been typed by the user.
`reset`	Clears the MadAnalysis 5 memory as when the program is started.
`restart`	Restarts MadAnalysis 5.
`shell [command]` `![command]`	Executes a Unix command from the interpreter.
`#[text]`	Indicates a comment.

In order to design an analysis in the MadAnalysis 5 framework, the user can rely on the following commands.

Command	Description
`define [(multi)particle label] = [ID1] [ID2] ...`	Creates a new (multi)particle label attached to the provided PDG codes `[ID1]`, `[ID2]`, ...
`define_region [label1] [label2] ...`	Creates one or more signal regions named `[label1]`, `[label2]`, ...
`import [label]`	Imports a sample or a UFO model.
`open [folder]`	Opens a report from the folder `[folder]`. If the folder is unspecified, the last created report is open.
`plot [obs]([part1] [part2] ...) nbins min max [ [opts] ] { [regs] }`	Defines a histogram with the distribution in the observable `[obs]`. Its computation may require to combine the momenta of several objects `[part1]`, `[part2]`, ... Any other parameter is optional. The `nbins`, `min` and `max` quantities respectively correspond to the number of bins, and the upper and lower bound of the x-axis of the histogram. The options `[opts]` (see below) allow the user to tune the display of the histogram, and `[regs]` indicates to which signal region one needs to attach this histogram.
`reject [criterion]` `select [criterion]`	Defines a selection cut that leads to the rejection (or selection) of an event if the `[criterion]` condition is satisfied.
`reject ([part]) [criterion]` `select ([part]) [criterion]`	Defines a selection cut that leads to the rejection (or selection) of an object candidate if the `[criterion]` condition is satisfied.
`remove [object]`	Deletes an object (a (multi)particle label, a region, a histogram or a cut).
`resubmit`	Adjusts the last generated C++ code relatively to the commands issued after the last `submit` command, and executes it.
`set [obj] = [val]`	Sets an attribute of a specific object to a given value.
`submit [folder]`	Generates, compiles and executes the C++ code corresponding to the current analysis either in a folder named `[folder]`, if specified, or in an arbitrary folder otherwise.
`swap main.selection[n1] main.selection[n2]`	Swaps the analysis steps number `[n1]` and `[n2]`.

Throughout the analysis, information can be printed to the screen by means of the following commands.

Command	Description
`display [object]`	Displays the properties of a specific object.
`display_datasets`	Displays the list of all defined datasets.
`display_multiparticles`	Displays the list of all defined multiparticle labels.
`display_particles`	Displays the list of all defined particle labels.
`display_regions`	Displays the list of all defined signal regions.

4 Properties of the main object

The `main` object of MadAnalysis 5 allows to setup varied options (see the table below) to impact the code on run time. They can be modified and displayed by using the `set` and `display` commands introduced above.

Command	Description
`main.currentdir`	Folder containing MadAnalysis 5.
`main.fastsim.package`	The package used for the fast-simulation of the detector response. The available choices are `fastjet`, `delphes`, `delphesMA5tune` and `none`.
`main.fom.formula`	Formula to be used to calculate the figure of merit in the cutflow charts. Denoting by S and B the number of signal and background events, the available choices are 1 (S/B), 2 ($S/\sqrt{B}$), 3 ($S/\sqrt{B}$), 4 ($S/\sqrt{S+B}$) and 5 ($S/\sqrt{S+B+x_B^2}$). For this last formula, the x_B parameter is specified via `main.fom.x`.
`main.graphic_render`	Package to use for figure generation. The available choices are `root`, `matplotlib` and `none`.
`main.isolation.algorithm`	Algorithm to be used for particle isolation. The available choices are `cone` (no activity in a cone of radius specified by `main.isolation.radius`) and `sumpt` (the scalar sum of the transverse momenta of all particles lying in a given cone around the candidate must be smaller than `main.isolation.sumPT` and the ratio of the transverse energy of all particles in this cone to the transverse momentum of the candidate must be smaller than `main.isolation.ET_PT`).
`main.lumi`	Integrated luminosity, in fb^{-1}, to use for histogram and cutflow normalization.
`main.normalize`	Way in which histograms have to be normalized. The available choices are `none` (each event counts for 1), `lumi` (normalization to the integrated luminosity without taking into account the event weights), `lumi_weight` (as `lumi` but with the event weights).
`main.outputfile`	Name of the output file to write events onto.
`main.recast`	Switching the recasting mode `on` and `off`.
`main.stacking_method`	Way in which the contributions of the different datasets to a histogram are displayed. The available choices are `normalize2one` (the integral of each contribution equals 1), `stack` (each contribution is stacked) and `superimpose` (each contribution is superimposed).

5 Observables to be used for histograms and cuts

As shown in the tables above, the definition of a histogram or of a cut condition necessitates to provide an observable that could depend on the momentum of one or more particles or objects. We list in the tables below all observables supported by MadAnalysis 5, and begin with those that do not depend on the momenta of any object. They are therefore called without any argument.

Symbol	Description
ALPHA_QCD	Value of the QCD coupling constant.
ALPHA_QED	Value of the electromagnetic coupling constant.
ALPHA_T	The α_T variable [?].
MEFF	Effective mass being defined as the sum of the transverse momentum of all final-state objects and the missing transverse energy.
MET	Missing transverse energy.
MHT	Missing transverse energy defined from the jet activity only.
NPID	Particle content (PDG code distribution).
NAPID	Particle content (PDG code distribution in absolute value).
SQRTS	Partonic center-of-mass energy.
SCALE	Energy scale of the event.
TET	Scalar sum of the transverse energy of all final-state objects.
THT	Scalar sum of the transverse energy of all final-state jets.
WEIGHTS	Event weights.

The set of observables provided in the following table can be used to study the properties of a given object or particle, and thus requires to provide one four-momentum or one combination of four-momenta as an argument of the observable function.

Symbol	Description
ABSETA	Absolute value of the pseudorapidity.
BETA	Velocity $\beta = v/c$ (relatively to the speed of light).
E	Energy.
EE_HE	Ratio of the electromagnetic energy to the hadronic energy (for a jet).
ET	Transverse energy.
ETA	Pseudorapidity.
GAMMA	Lorentz-factor.
HE_EE	Ratio of the hadronic energy to the electromagnetic energy (for a jet).
M	Invariant mass.
MT	Transverse mass.
MT_MET	Transverse mass of the system comprised of the object and the missing momentum.
NTRACKS	Number of tracks (inside a jet).
P	Magnitude of the three-momentum.
PHI	Azimuthal angle.
PT	Transverse momentum.
PX	x-component of the momentum.
PY	y-component of the momentum.
PZ	z-component of the momentum.
R	Position in the (η, ϕ) plane.
Y	Rapidity.

Three additional observables involving two objects are also available, the methods given in the table below taking thus two arguments separated by a comma.

Symbol	Description
DELTAR	Angular distance, in the transverse plane, between the objects.
DPHI_0_PI	Angular distance in azimuth between the objects. The bounds for the angle are $[0, \pi]$.
DPHI_0_2PI	Angular distance in azimuth between the objects. The bounds for the angle are $[0, 2\pi]$.

For all arguments of any of the above observable, any sequence of momenta separated with spaces will lead to a sum of these momenta before computing the observable. For instance,

```
plot M(e+ e-)
```

allows for the computation of the invariant of an electron-positron system.

6 Options for histograms

The command `plot` accept varied options `[opts]`,

```
plot [obs]([part1] [part2] ...) nbins min max [ [opts] ] { [regs] }
```

to be provided between squared brackets. The list of all available choices is given in the table below.

Symbol	Description
`Eordering`	Sorts the objects in increasing energy.
`ETordering`	Sorts the objects in increasing transverse energy.
`ETAordering`	Sorts the objects in increasing pseudorapidity.
`Pordering`	Sorts the objects in increasing three-momentum magnitude.
`PTordering`	Sorts the objects in increasing transverse momentum.
`PXordering`	Sorts the objects in increasing momentum x-component.
`PYordering`	Sorts the objects in increasing momentum y-component.
`PZordering`	Sorts the objects in increasing momentum z-component.
`allstate`	Considers all (initial-state, final-state and intermediate-state) objects in the events.
`finalstate`	Considers only final-state objects (default).
`initialstate`	Considers only initial-state objects.
`interstate`	Considers only objects that are neither initial-state nor final-state objects.
`logX`	Logarithmic scale for the x-axis.
`logY`	Logarithmic scale for the y-axis.
`normalize2one`	Normalizes the histogram to 1.
`stack`	Stacks the contributions of different datasets in the histogram.
`superimpose`	Superimposes the contributions of different datasets in the histogram.

7 Options for datasets

Once one or more samples have been imported as a dataset, properties that will impact the display of the subsequent contribution in histograms can be modified by using the `set` command. For instance, a dataset named `defaultset` can be imported as

```
import <path-to-sample> as defaultset
```

and its properties can be modified by typing in

```
set defaulset.<property> = <value>
```

The list of available properties is given in the table below, together with the allowed values.

Symbol	Description
backcolor	Background color in a histogram. The available colors are auto, black, blue, cyan, green, grey, none (transparent), orange, purple, red, white and yellow. The color can be made lighter or darker by adding an explicit ±1, ±2 or ±3.
backstyle	Background texture in a histogram. The available values are dline (diagonal-lines), dotted (dots), hline (horizontal lines), solid (uniform color) and vline (vertical lines).
linecolor	Color of the histogram lines. The available colors are the same as for the backcolor attribute.
linestyle	Style of the histogram lines. The available values are dash-dotted, dashed, dotted and solid.
linewidth	Width of the histogram lines, given as an integer smaller than 10.
title	Name of the dataset (for histogram legends).
type	background or signal nature of a given sample (for figure-of-merit calculations).
weight	Reweights each histogram entry with a constant factor. The value has to be a floating-point number.
weighted_events	Allows MadAnalysis 5 to ignore the weights of the events (property to be set to true or false).
xsection	This overwrites the event sample cross section. The value has to be given in pb.
scale_variation	This attaches to the dataset cross section a theory error associated with the variation of the unphysical scales. The value has to lie in the [0,1] interval and the error band is symmetric around the central value.
pdf_variation	This attaches to the dataset cross section a theory error associated with the variation of the parton densities. The value has to lie in the [0,1] interval and the error band is symmetric around the central value.
scale_up_variation	This attaches to the dataset cross section a theory error associated with the variation of the unphysical scales. The value has to lie in the [0,1] interval and this parameter concerns the upper enveloppe of the error band.
scale_down_variation	This attaches to the dataset cross section a theory error associated with the variation of the unphysical scales. The value has to lie in the [0,1] interval and this parameter concerns the lower enveloppe of the error band.
pdf_up_variation	This attaches to the dataset cross section a theory error associated with the variation of the parton densities. The value has to lie in the [0,1] interval and this parameter concerns the upper enveloppe of the error band.
pdf_down_variation	This attaches to the dataset cross section a theory error associated with the variation of the parton densities. The value has to lie in the [0,1] interval and this parameter concerns the lower enveloppe of the error band.

8 Using FastJet through MadAnalysis 5

In order to activate the usage of FastJet through MadAnalysis 5, the program has to be started in the reconstructed mode and the first command to be typed in the interpreter has to be

```
set main.fastsim.package = fastjet
```

This allows for various option of the `main.fastsim` object, tuning the properties of the jet algorithm that has to be employed. Those options are set by typing in

```
set main.fastsim.<property> = <value>
```

the list of all available properties being presented, together with the allowed values, in the following table.

Symbol	Description
`algorithm`	Sets up the jet algorithm to use. The allowed values are `antikt` [?], `Cambridge` [?, ?], `cdfjetclu` [?], `cdfmidpoint` [?], `genkt` [?], `gridjet` [?], `kt` [?, ?], `none` and `siscone` [?].
`areafraction`	Controls the size of the cones in the CDF midpoint algorithm.
`exclusive_id`	Exclusive mode for jet reconstruction. If set to `false`, electrons muons and photons issued from hadron decays are included into the electron, muon and photon collections respectively.
`input_ptmin`	Soft protojet threshold in the siscone algorithm.
`iratch`	Switching on ratcheting for the CDF jet clustering algorithm.
`npassmax`	Number of iterations in the siscone algorithm.
`overlap`	Fraction of overlapping momentum used to combine protojets in the siscone and CDF reconstruction algorithms.
`p`	p parameter of the generalized k_T algorithm.
`ptmin`	Threshold for the transverse momentum of the reconstructed jets.
`radius`	Radius parameter relevant for most jet clustering algorithms.
`seed`	Threshold parameter used in the constituent merging procedure of the CDF reconstruction algorithms.
`spacing`	Grid spacing in the grid jet algorithm.
`ymax`	Maximum rapidity value in the grid jet algorithm.

In addition, basic detector simulation effects can be mimicked through setting up the parameters included in the table below.

Symbol	Description
`bjet_id.efficiency`	b-tagging efficiency, as a float.
`bjet_id.exclusive`	Allows several b-jets to be issued from a single B-hadron.
`bjet_id.matching_dr`	Angular distance parameter matching a b-jet with a B-hadron.
`bjet_id.misid_cjet`	Mistagging rate of a c-jet as a b-jet, as a float.
`bjet_id.misid_ljet`	Mistagging rate of a light jet as a b-jet, as a float.
`tau_id.efficiency`	Tau-tagging efficiency, as a float.
`tau_id.misid_ljet`	Mistagging rate of a light jet as a hadronic tau, as a float.

9 Using Delphes through MadAnalysis 5

In order to activate the usage of Delphes through MadAnalysis 5, the program has to be started in the reconstructed mode and the first command to be typed in the interpreter has to be

```
set main.fastsim.package = delphes
```

The properties of the simulation of the detector can then be adjusted through

```
set main.fastsim.<property> = <value>
```

the allowed choices being given in the table below.

Symbol	Description
`detector`	Determines which detector card to use. The card can be further modified on run time).
`output`	Saves the output Root file (`true` or `false`).
`rootfile`	Name of the output file.
`pileup`	Specifies the path to the input pile-up event file.
`skim_genparticles`	If set to `true`, the generator-level particles are not stored in the output file.
`skim_tracks`	If set to `true`, the track collection is not stored in the output file.
`skim_towers`	If set to `true`, the collection of calorimetric towers is not stored in the output file.
`skim_eflow`	If set to `true`, the collection of particle-flow towers is not stored in the output file.

10 Multipartonic matrix element merging

MadAnalysis 5 can be used to double check the merging procedure of event samples related to a given hard process but with matrix elements featuring a different final-state jet multiplicity. MadAnalysis 5 has to be started in the hadronic mode, and the check is then performed by typing in

```
set main.merging.check = true
```

Two extra options are available,

```
set main.merging.ma5_mode = <true or false>
set main.merging.njets = <integer>
```

the first one indicating to extract the number of extra hard jets from the process identifier, and the second one setting up the maximum number of extra jets to consider.

11 Super-fast detector simulation with MadAnalysis 5

MadAnalysis 5 offers the user options to parameterize a super-fast simulation of a detector response when event reconstruction is achieved through FastJet. The user should first initialize FastJet, through MadAnalysis 5, as shown in section 8. However, as more advanced b-tagging and tau-tagging capabilities are available, default values for all tagging options of the FastJet interface must be enforced to avoid redundancies.

The super-fast detector simulation module (denoted below as the super-fastsim package of MadAnalysis 5) offers two ways to treat jets, which can be set as

```
set main.fastsim.jetrecomode = <value>
```

Two choices exist for the value `<value>` of the `jetrecomode` attribute,

Symbol	Description
`jets`	The jet reconstruction probability distribution and smearing functions are applied after the clustering (default).
`constituents`	The jet reconstruction probability distribution and smearing functions are applied to each of the jet constituents before their clustering with FastJet.

The super-fastsim package of MadAnalysis 5 is shipped with three submodules, for which dedicated examples will be provided below. Their respectively address object reconstruction, the smearing of the momenta of the various reconstructed objects and the tagging of a given object as such (tagging efficiencies) or as a different object (mistagging rates). These modules are

Symbol	Description
`reco_efficiency`	Generates a probability distribution relying on a user-defined piecewise function. This distribution describes the efficiency to reconstruct a given object.
`smearer`	Generates a Gaussian smearing function relying on a user-defined piecewise function that reflects the standard deviation of the Gaussian. The four-momentum of the object that the function is attached to is then smeared.
`tagger`	Generates a probability distribution relying on a user-defined piecewise function describing the probabily to (mis)tag an object.

The syntax to be employed to make use of the `reco_efficiency` submodule reads

```
define reco_efficiency <obj> <function> [<domain>]
```

In this expression, `<obj>` consists in the object for which the reconstruction efficiency is given. The `reco_efficiency` submodule can be used for for the reconstruction of jets (`j`), hadronic taus (`ta`), muons (`mu`), electrons (`e`) and photons (`a`). The piecewise function on which the objet reconstruction probability distribution is built from is given through the `<function>` and `<domain>` arguments, the former being mandatory and the latter optional. One call to `define reco_efficiency` has to be made for each subdomain of the piecewise function. The function `<function>` can depend on any observable supported by MadAnalysis 5 (see section 5). All methods implemented in the `cmath` Python package are supported, so that the code understands for instance all trigonometric, logarithmic and cyclometric functions. Numbers can be inputted either in their traditional form or following the scientific format. The domain `<domain>` should contains one or more inequalities (connected by the logical operators `and` and `or`) and the analytical form of the inequalities can rely on any function (supported by the `cmath` Python package) of the observables available in MadAnalysis 5. In the case where the domain is not provided, the efficiency is understood to be applicable in all cases.

The syntax to be employed to make use of the `smearer` submodule reads

```
define smearer <obj> with <comp> <function> [<domain>]
```

In this expression, `<obj>` consists in the object whose momentum has to be smeared. The `smearer` submodule can be used for for jets (`j`), hadronic taus (`ta`), muons (`mu`), electrons (`e`) and photons (`a`). The smearing proceeds by modifying the component of the object four-momentum `<comp>` according to a Gaussian distribution specified by the user. In practice, the mean of the distribution is taken to be the initial value of `<comp>` and its standard deviation stems form the piecewise function provided by the user `<function>`. As above, the `smearer` module has to be called for each of the subdomain of the function. The syntax for providing the function and the domain information is identical to what has been described for the `reco_efficiency` module. The momentum component `<comp>` can be the energy (`E`), the transverse momentum (`PT`) or any of the three-momentum components (`PX`, `PY` and `PZ`) of the smeared object.
The `tagger` submodule can be used following the syntax

```
define tagger <true_obj > as <reco_obj> <function> [<domain>]
```

This indicates to the super-fastsim how to tag an object `<true_obj>` as an object `<reco_obj>` with a given probability specified by a piecewise function. This function and its domain are provided according to the rules described above, and the allowed tagging and mistagging pairs are given in the table below.

Symbol	Description
`b as b`	Efficiency to tag a true b-jet as a b-jet.
`c as b`	Mistagging rate of a true c-jet as a b-jet.
`j as b`	Mistagging rate of a true light jet as a b-jet.
`c as c`	Efficiency to tag a true c-jet as a c-jet.
`j as c`	Mistagging rate of a true light jet as a c-jet.
`b as c`	Mistagging rate of a true b-jet as a c-jet.
`ta as ta`	Efficiency to tag a true hadronic tau as a hadronic tau.
`j as ta`	Mistagging rate of a true light jet as a hadronic tau.
`e as j`	Mistagging rate of a true electron as a jet.
`e as mu`	Mistagging rate of a true electron as a muon.
`e as a`	Mistagging rate of a true electron as a photon.
`mu as j`	Mistagging rate of a true muon as a jet.
`mu as e`	Mistagging rate of a true muon as an electron .
`mu as a`	Mistagging rate of a true muon as a photon.
`a as j`	Mistagging rate of a true photon as a jet.
`a as e`	Mistagging rate of a true photon as an electron .
`a as mu`	Mistagging rate of a true photon as a muon.

12 Recasting LHC analyses with MadAnalysis 5

A collection of LHC analyses have been reimplemented in the MadAnalysis 5 framework and are available from the public page http://madanalysis.irmp.ucl.ac.be/wiki/PublicAnalysisDatabase. The so-called recasting module of MadAnalysis 5 allows one to download a commented C++ code associated with each of these analyses, together with information allowing to extract bounds on a given new physics signal. This download is achieved by typing in

```
install delphes
```

```
install delphesMA5tune
install PAD
install PADForMA5tune
```

The first two commands trigger the installation of the current release of Delphes as well as of its old tuned version, whilst the last two commands install the two versions of the MadAnalysis 5 Public Analysis Database of recasted LHC analyses. The first version (`PAD`) relies on the current release of Delphes for the simulation of the detector reponse, whilst the second one (`PADForMA5tune`) relies on the tuned version of Delphes. More information can be found in section 2. In order to switch on the recasting module of MadAnalysis 5, the user needs to type in

```
set main.recast = on
```

the defaul value being `off`. The user must then import a hadron-level dataset stored under the HepMC or StdHep format, as detailed in section 7. The options allowing to setup the dataset cross section, together with the associated theory uncertainties, are crucial for the good functioning of the recasting module and the extraction of a correct limit on the signal.
The module moreover allows the user to add systematic uncertainties on the signal and extrapolate the results to differnt luminosities by typing in

```
set main.recast.add.<property> = <value>
```

where the implemented properties and possible values are given below.

Symbol	Description
`systematics`	Adds systematic uncertainty to the signal, the value of the attribute being in the [0,1] interval. A single value indicates that the uncertainties are symmetric and two values separated by a space or a comma indicate asymmetric errors. The user can add several systematic uncertainties that will be combined in quadrature.
`extrapolated_luminosity`	Indicates that the results will have to be extrapolated to other lumiosities, to be provided in fb^{-1} and as floating-point numbers. Multiple luminosities can be added, separated by a space or a comma. A separate output is generated for each luminosity value.

The module offers the user a few additional properties that can be set as

```
set main.recast.<property> = <value>
```

The list of available options is given below.

Symbol	Description
`CLs_numofexps`	Number of toy experiments to be used in the CLs calculations (the default value is 100000).
`card_path`	Path of the recasting card containing the list of analyses to reinterpret (if not provided, a default card is generated ny MadAnalysis 5).
`store_root`	Boolean indicatign whether the root file outputted by Delphes should be stored (the default value is `false`).
`THerror_combination`	Indicates how the theoretical uncertainties should be combined, *i.e.*, either quadratically (`quadratic`) or linearly (`linear`, default).
`error_extrapolation`	Indicates how the uncertainties should be scaled when the results are extrapolated to other luminosities. The possible options are `linear` (*i.e.*, as for systematic uncertainties; default) or `sqrt` (*i.e.*, as for statistical uncertainties).