GENIE Systematic Uncertainties with the Multi-universe Approach in CAFAna

1. Overview

The standard approach for drawing the NOvA systematic error band from GENIE cross section models is to vary the GENIE tunable physics parameters by N σ's, $N \in \{-2, -1, 1, 2\}$, input this number through the shift parameter of the CAFAna Spectrum class, and obtain shifted spectra as the boundaries of the error band. There are tens of tunable GENIE parameters, hereafter referred to as GENIE knobs, but only a few of them having the biggest effects are varied. This leaves somewhat arbitrary where to leave out the knobs.

A consistent way of treating the GENIE systematic uncertainties across all analyses, known as the multi-universe approach, was proposed, which varies all the knobs at the same time. One can think of a universe as a collection of physics parameters. Different universes have different values for these physics parameters. One advantage of this approach is it enables bin-to-bin correlation studies which could be in turn followed by dimensionality reduction analyses such as principal component analysis (PCA) [1].

In this note details of the implementation of this approach under the CAFAna framework is given, followed by sanity checks of this approach compared with the conventional way. Then some known issues are discussed. In the end the computing resources consumed by this class is presented.

2. Implementation

The code can be found, under NOvA's offline repository, in `CAFAna/XSec/GenieMultiverseSyst.[h,cxx]`. There are several classes defined in the files, where `MultiverseSpectra` takes on a central role.

Since there are tens of GENIE tunable knobs, to switch the knobs on and off easily, a so called knob configuration file is input into the class with a default name of `knob_config.txt` and a default path of the current directory.

With the development of theories, new knobs could be added into GENIE. To account for the possible difference in the tunable parameters used in different production datasets, a script is offered to make a configuration file based on the underlying GENIE versions of the production datasets, located in `CAFAna/XSec/Utilities/make_template_knob_config.py`. To use this scrip, simply do:

(i) setup nova

(ii) `$ python make_template_knob_config.py -d <sam_defname>`

This script generates a complete set of GENIE knobs with the name of a knob followed by a number. The names are extracted from the source file `$NUTOOLS_DIR/source/NuReweight/ReweightLabels.h`, with the leading "`fReweight`" or "`kReweight`" stripped. The number following a knob name assumes 3 numbers, namely 0, 1, or 2. 0 means it is disabled, like it does not exist at all. If it is 1, a random number is drawn from a normal distribution $\mathcal{N}(0,1)$, and used to construct a `SystShifts` object as an input argument to the `Spectrum` class. If it is 2, it means the knob chooses between two alternative models. In this case, a random number is drawn from $\{0,1\}$ with equal probability, where 0 represents the nominal model, and 1 represents the alternative model. Note that even for this kind of knobs mode 1 can still be used. In this case CAFAna does a linear evolution from the weight of the first model to that of the second model.

Note that to make the results reproducible, the random seed sequence starts from 1001, leaving the first 1000 numbers to the flux multi-universe [2].

A default configuration file is available in `CAFAna/XSec/Utilities/knob_config.txt`. By and large it enables all available knobs, and uses mode 2 whenever the knob is for alternative models. However, some knobs tune the same physical parameter. For example, `MaCCQE` tunes the normalization and shape for CCQE at the same time, and `NormCCQE` and `MaCCQEshape` tune normalization and shape of CCQE, respectively. To avoid double counting, the three knobs should not be enabled at the same time. According to the former studies [‡], decision was made to enable separate knobs and disable the combined one.

[‡] Please refer to page 15 in DocDB-15214.

2.1. Interface

The interface is designed to mimic that of the `Spectrum` class as close as possible, with two more constructor arguments. One is the number of universes one would like to generate, and the other is the pathname of the knob configuration file. Please consult the doxygen page for detailed information.

Once you have created a multi-universe object and filled them with the `SpectrumLoader::Go()` function call, you can start to extract the needed information, namely, the error band.

To obtain the spectrum above the nominal spectrum, use `MultiverseSpectra::UpperSigma()`. Similarly, `MultiverseSpectra::LowerSigma()` gives you the spectrum below the nominal one. The boundaries of the error band are obtained in the following way. For each bin of the spectrum, each universe has a different number of events. If one plots the distribution of number of events in a bin, usually it is not symmetric. For example, Figure 1 shows the reconstructed neutrino energy with all universes overlaid. If one takes the bin between the blue dashed lines and projects the counts to the y-axis, Figure 2 results.

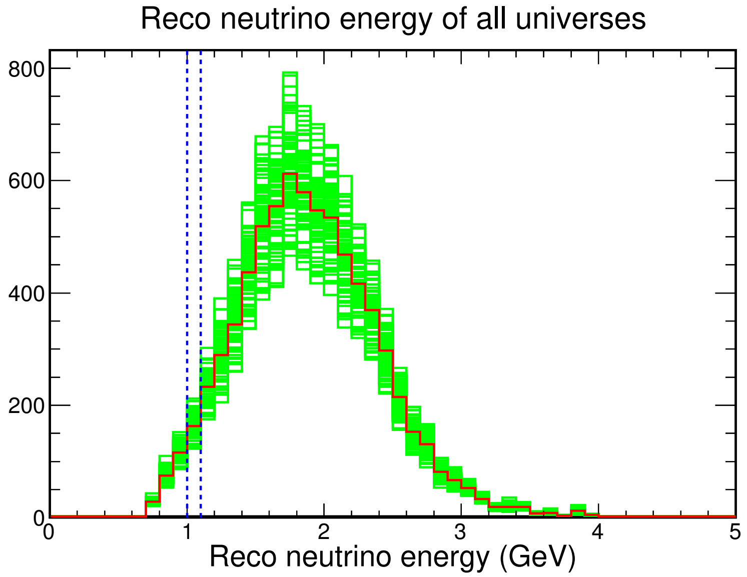

Figure 1. Reconstructed neutrino energy with all universes overlaid. The red histogram represents the nominal universe. To draw the error band, for each bin, project the counts to the y-axis and infer from the distribution.

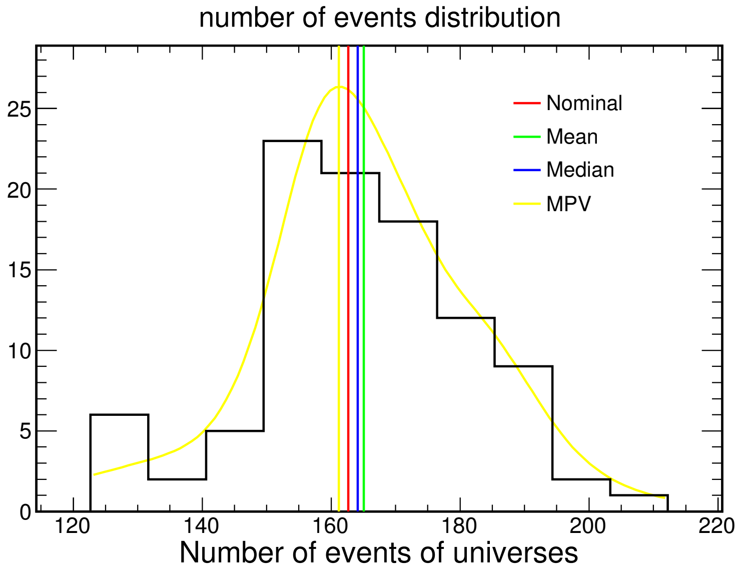

Figure 2. Different choices for the center of the distribution. The nominal value is in red. The mean and the median are drawn from the unbinned sequence of values. For the most probable value (yellow), it is estimated by the Kernel Density Estimation (KDE) algorithm. The yellow curve is the continuous distribution reconstructed by the algorithm.

The class offers several options for the center of the distribution to draw the spread of the distribution from. Figure 2 shows the options to draw the spread from. The default option is set to the nominal value. Once a center is chosen, the upper boundary is drawn from the center such that 34% of the universes are enclosed in the center line and the upper bound. The lower boundary is obtained in the same way. Figure 3 shows the boundaries determined by this procedure. Note that the number of universes outside the upper and lower lines are not equal since, in general, the median of the distribution is not the same as the central value you decide to draw the band from. The resulting error band obtained with this approach is shown in Figure 4.

3. Performance

It is observed that generating the same number of spectra with this class is significantly slower than generating usual spectra. An investigation was conducted

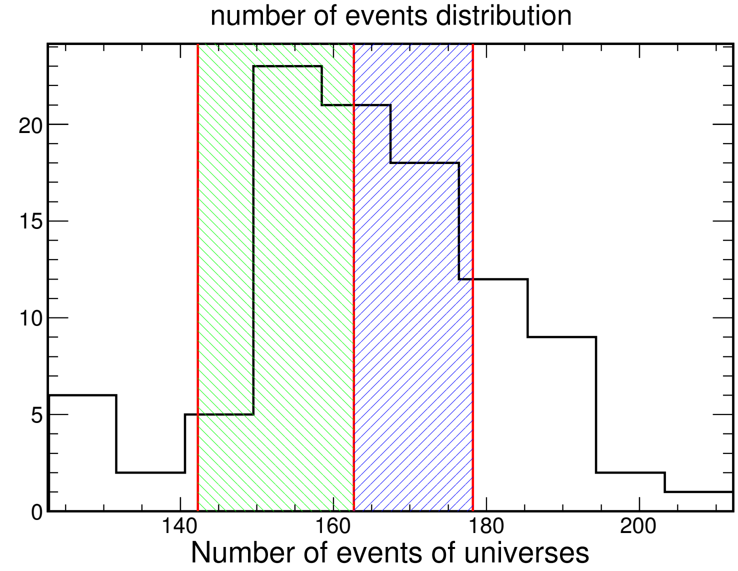

Figure 3. The upper and lower boundaries of the error band. The boundaries are determined such that the blue and green shaded areas each contain 34% of the total number of universes.

trying to identify the time consuming processes of this class.

`CAFAna` incorporates the Google Performance Tools to analyze runtime performance of `CAFAna` scripts. To invoke this function, simply do:

```
$ cafe --perf <name_of_script>
```

After cafe finishes the script, the perftool is then executed, and a pdf file recording the computing resources used by the stages of the script is generated. The pdf file with the filename `perf_all_knobs.pdf` also included in this document shows the analysis result of a multi-universe script with all knobs enabled. The file `perf_one_knob.pdf`, on the other hand, shows the result of the same script running over the same number of universes and files, but with only one knob enabled. The `caf::Proxy::GetValue` function call uses significantly less time when enabling much fewer knobs, indicating that the performance hit might be simply due to the shear number of enabled knobs.

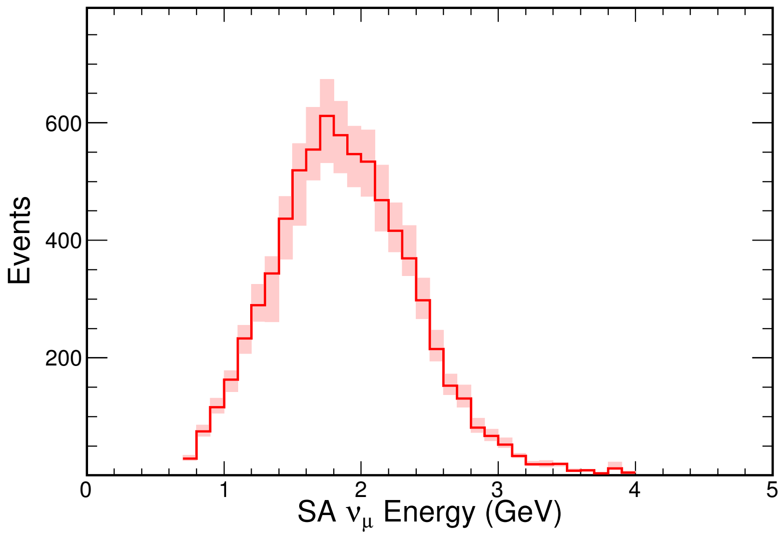

Figure 4. Error band obtained by this multi-universe approach.

3.1. Knobs with larger effects on the error band

The observation drawn from the google perf tools leads to the idea that a limited list of knobs with non-negligible contribution to the systematic error band might improve the CPU time significantly. A study of single knob effect on the muon kinematic variables is used to evaluate the size of effect of individual knobs. Table 3.1 lists the details of this study.

dataset	`prod_caf_R17-03-01-prod3reco.d_nd_genie_nonswap_fhc_nova_v08_full_v1`
variables	`kTrueMuKE & kTrueMuCostheta` defined in `NDAna/numucc_inc`
cuts	`kAllNumuCCCuts` defined in `NDAna/numucc_inc`
weights	`kPPFXFluxCVWgt * kXSecCVWgt2017`
number of universes	100

Table 1. Details of the single knob effect study.

After the spectrum and the error band of each single knob is obtained, the width of the band is divided by the nominal value bin by bin, and fit to a constant polynomial. The fit value then serves as the size of the error of this knob. Figure 5 shows the effects of individual knobs obtained by the above process. If a cut is made at 1%, the selected knobs are NormCCQE, MaCCQEshape, MaCCQE, VecCCQEshape, NormCCRES, MaCCRES, MvCCRES, RvpCC2pi, RvnCC1pi, RvnCC2pi, FormZone, MFP_pi, MFP_N, and CCQEPauliSupViaKF. We can repeat this process for the muon kinetic energy, where the effects of single knobs are shown in Figure 6. The two variable give slightly different results, and the union of the two selected knob sets is used as the list of significant knobs.

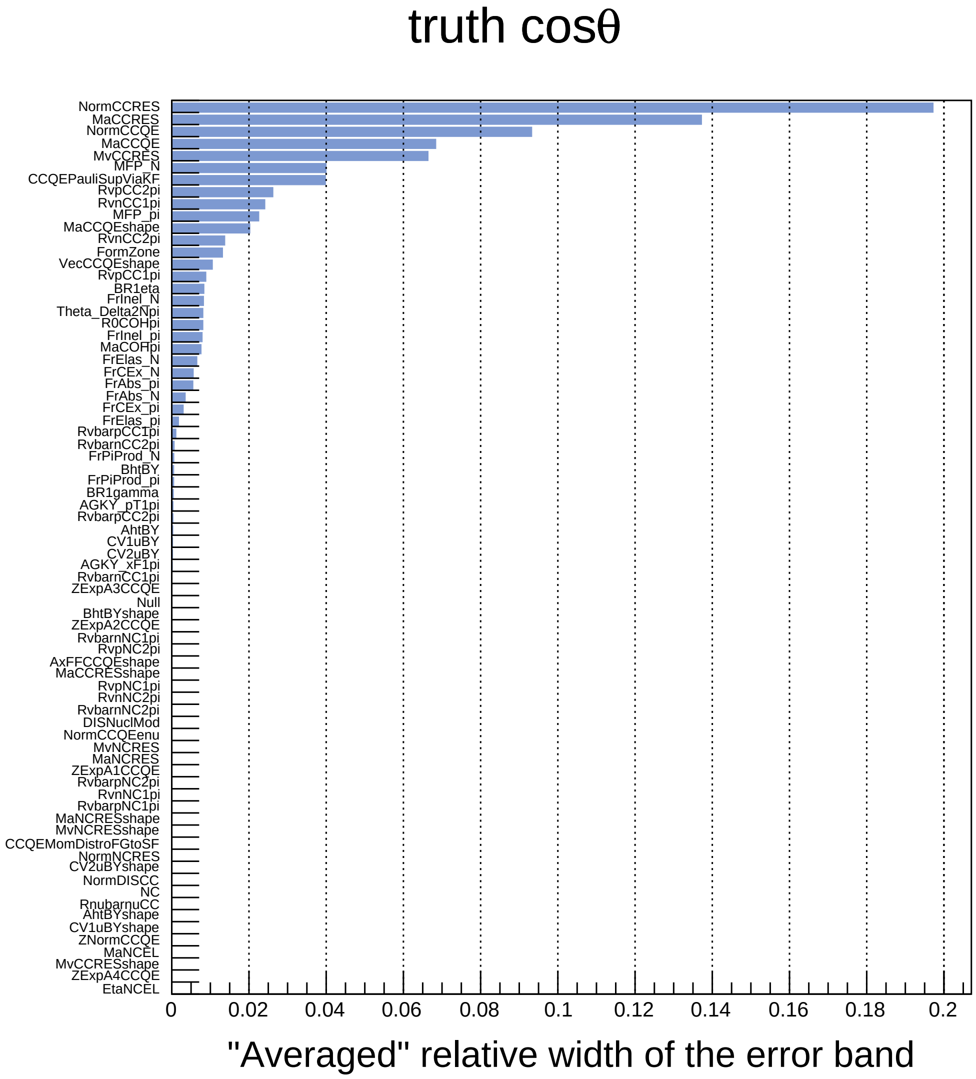

Figure 5. Single knob effect for the muon truth angle.

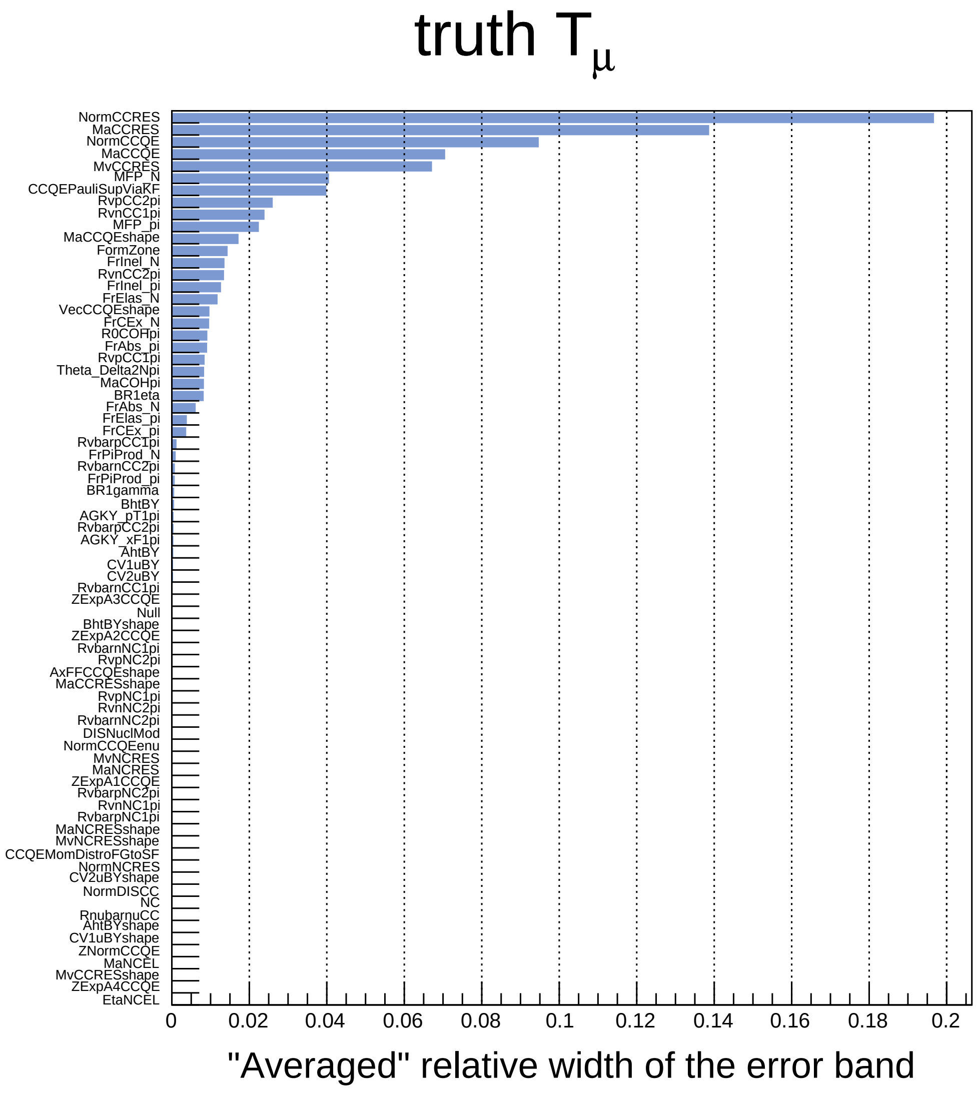

Figure 6. Single knob effect for the muon truth kinetic energy.

References

[1] Bannanje Nitish Nayak, PPFX Systematics, https://nova-docdb.fnal.gov:441/cgi-bin/ShowDocument?docid=18996

[2] Leonidas Aliaga Soplin, G4NuMI/PPFX in NOvA framework, https://nova-docdb.fnal.gov:441/cgi-bin/ShowDocument?docid=16397