Multivariate $\nu_\mu/\overline{\nu}_\mu$ Hadronic Energy Regression with CVN Output Features

Shih-Kai Lin

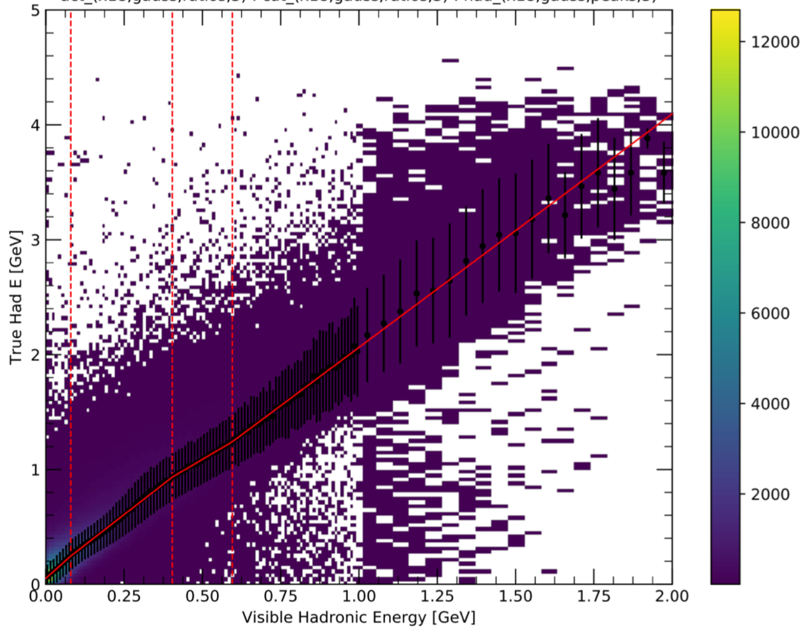
Colorado State Univertisy

September 21, 2018

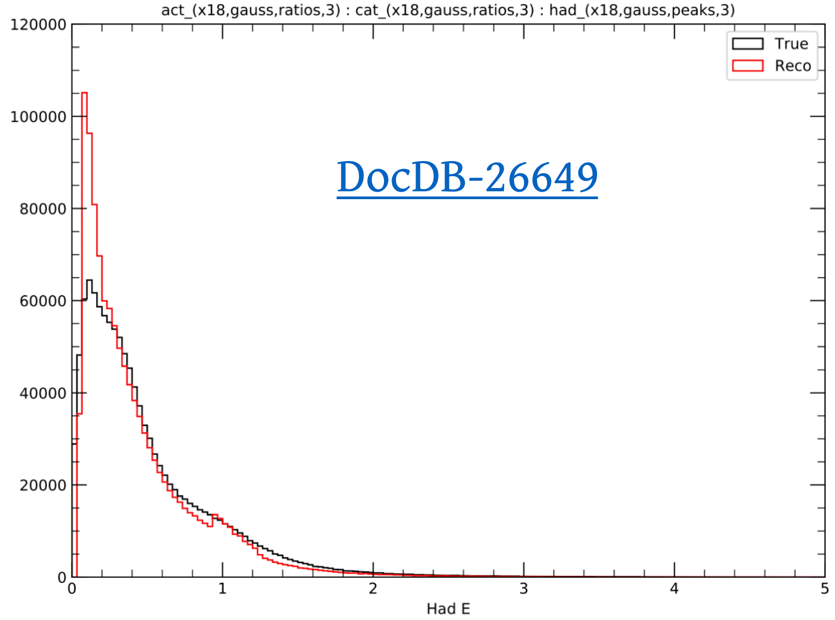
Reco Parallel, NOvA Collaboration Meeting

Motivation

Intro 4, Hadronic Energy Fit, Spline fit

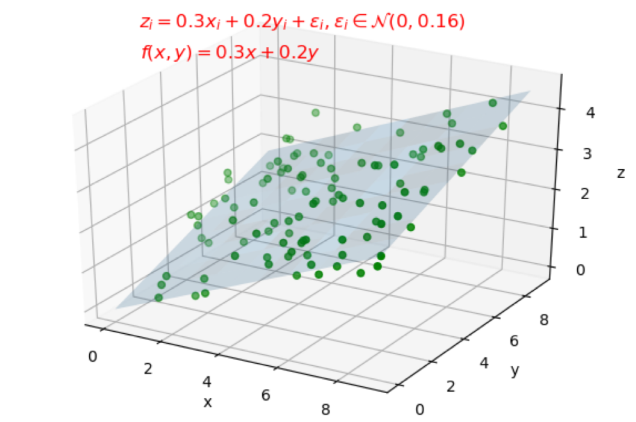


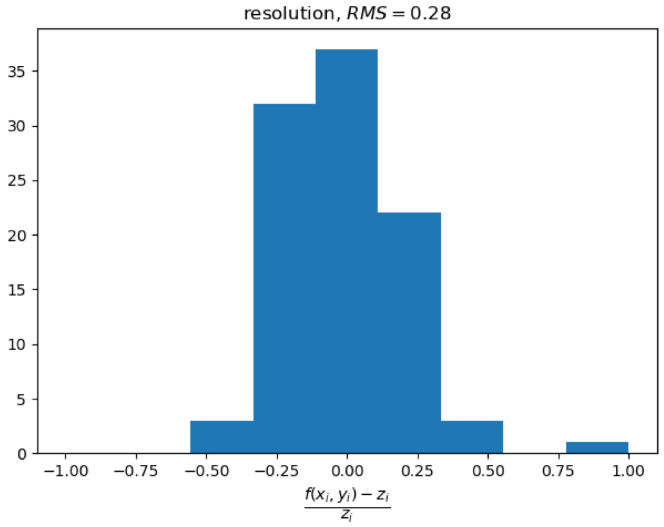
Motivation 3, Spiky Reco Hadronic Energy



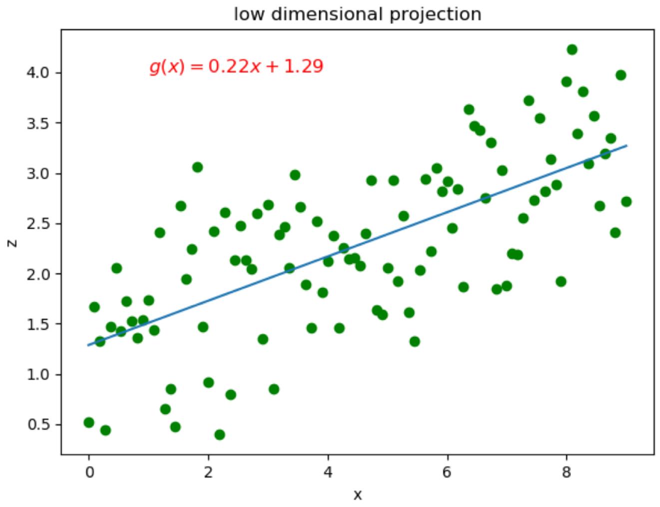
- I was in a NuMu meeting and listening to Dmitrii's talk about $\bar{\nu}_\mu$ CC energy reconstruction.
- The official way is using this spline fit from visible hadronic energy to true hadronic energy.
- This could lead to small peaks in reco spectrum around the kinks of the splines.
- Greg said, what if we had more handles…

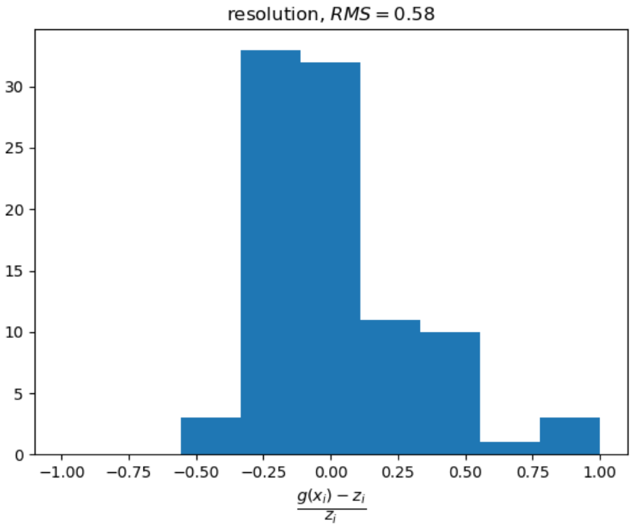
More Handles: An Illustration





- More handles means more variables.
- Suppose our variable is 2D, and the best fit is the plane.
- The RMS of resolution if we fit a 2D surface is 0.28.



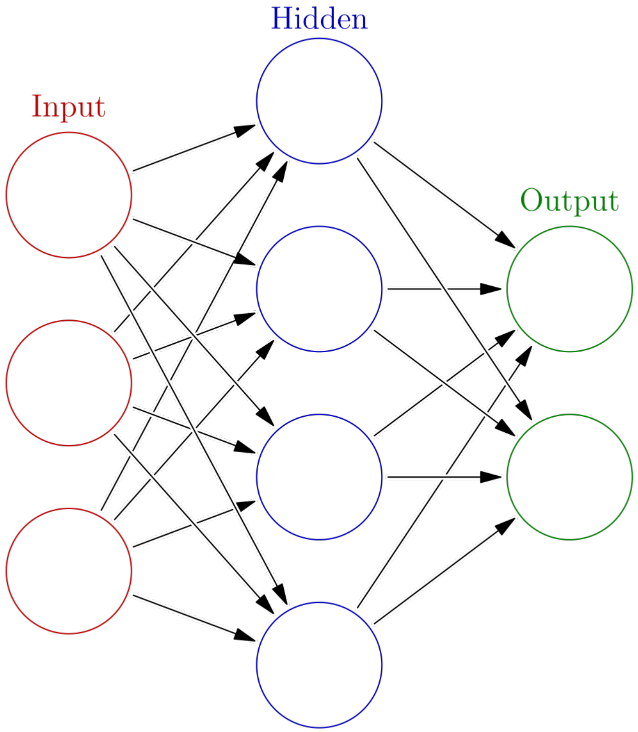


- If y variable is not utilized, we project data points to xz plane.
- The real spread caused by slope in y is now regarded as merely random fluctuation.
- The RMS of resolution if we fit a 1D line is larger at 0.58.
- The mean of resolution is also more biased.

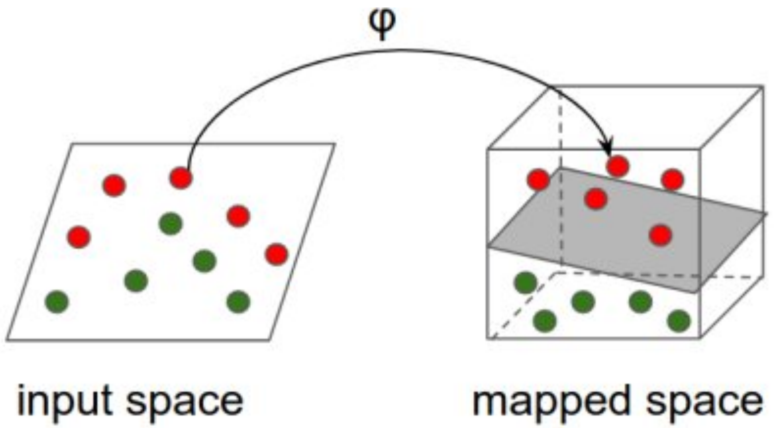
Two Classes of Nonlinear Multivariate Regression on the Market

Of course our problems belong to nonlinear regression!

Neural Networks



Kernel Methods



- Two of the very popular classes of methods.
- Support vector machine, one of the kernel methods, so dominated early 2000s research that almost killed neural network[1].
- Kernel methods have some appealing properties that I decided to try them out.

1 http://www.cs.cmu.edu/~10701/slides/Perceptron_Reading_Material.pdf. Of course we all know what happens later…

Kernel Methods

Idea for achieving nonlinearity: feature map (yeah, this term is overloaded)

- Maps input space to high dimensional space (even infinitely dimensional), do linear regression/classification there, and go back.
- Utilizes the so called kernel trick to avoid working out expensive feature map explicitly.
- Choosing a kernel function $K(\boldsymbol{x}, \boldsymbol{x}')$, the regression surface takes the form

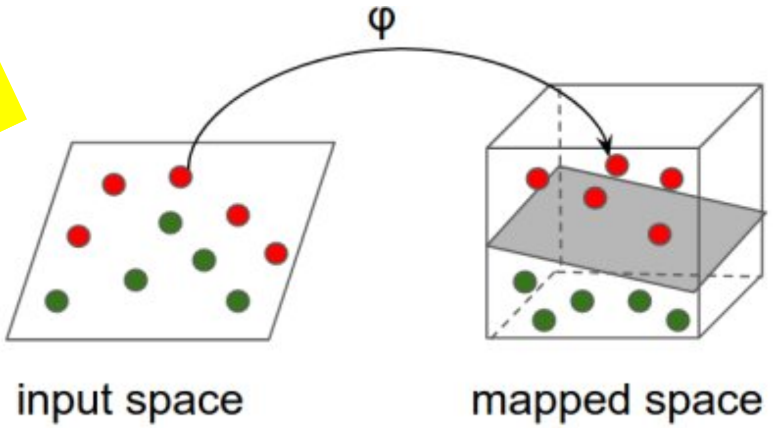
number of training samples

$$f(\boldsymbol{x}) = \sum_{k=1}^{n} a_i \, K(\boldsymbol{x_i}, \boldsymbol{x})$$

multivariate training data

RBF

	Kernel Ridge Regression	Support Vector Machine
Popular kernel function	$\exp(-\gamma\|\|\boldsymbol{x}-\boldsymbol{x}'\|\|^2)$	$\exp(-\gamma\|\|\boldsymbol{x}-\boldsymbol{x}'\|\|^2)$
Loss function	$L[f] = \sum_{i=1}^{n} \frac{(y_i - f(\boldsymbol{x_i}))^2}{{y_i}^2} + \alpha\|\|f\|\|2$	Refer to here
Hyper parameters	α, γ	C, γ
this work	✓	✗



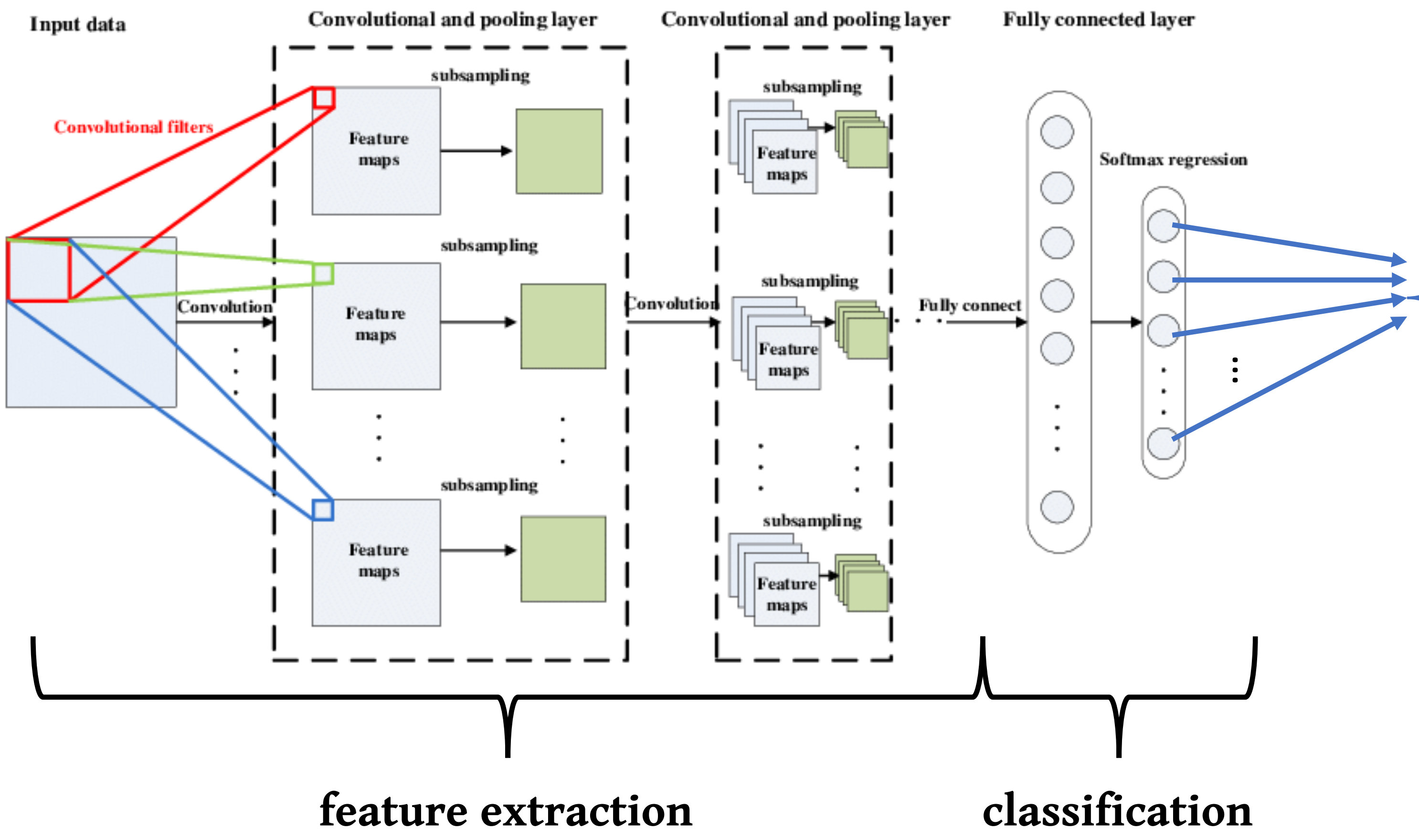
Hands-on

Time to get hands dirty:

- datasets
  `prod_caf_R17-11-14-prod4reco.d_nd_genie_nonswap_fhc_nova_v08_period3_v1`
- cuts
  `kNumuCutND2018&&kIsNumuCC`
- weights
  - `kXSecCVWgt2018*kPPFXFluxCVWgt`
- variables
  regressor
  - `kNumuHadVisE` and CVN particle final state scores for p, n, π^0, and $\pi^\pm$

  target
  - always `kTrueE` (true neutrino energy) - `kMuE` (prod4 reco muon energy)

Additional Handles / Feature Selection

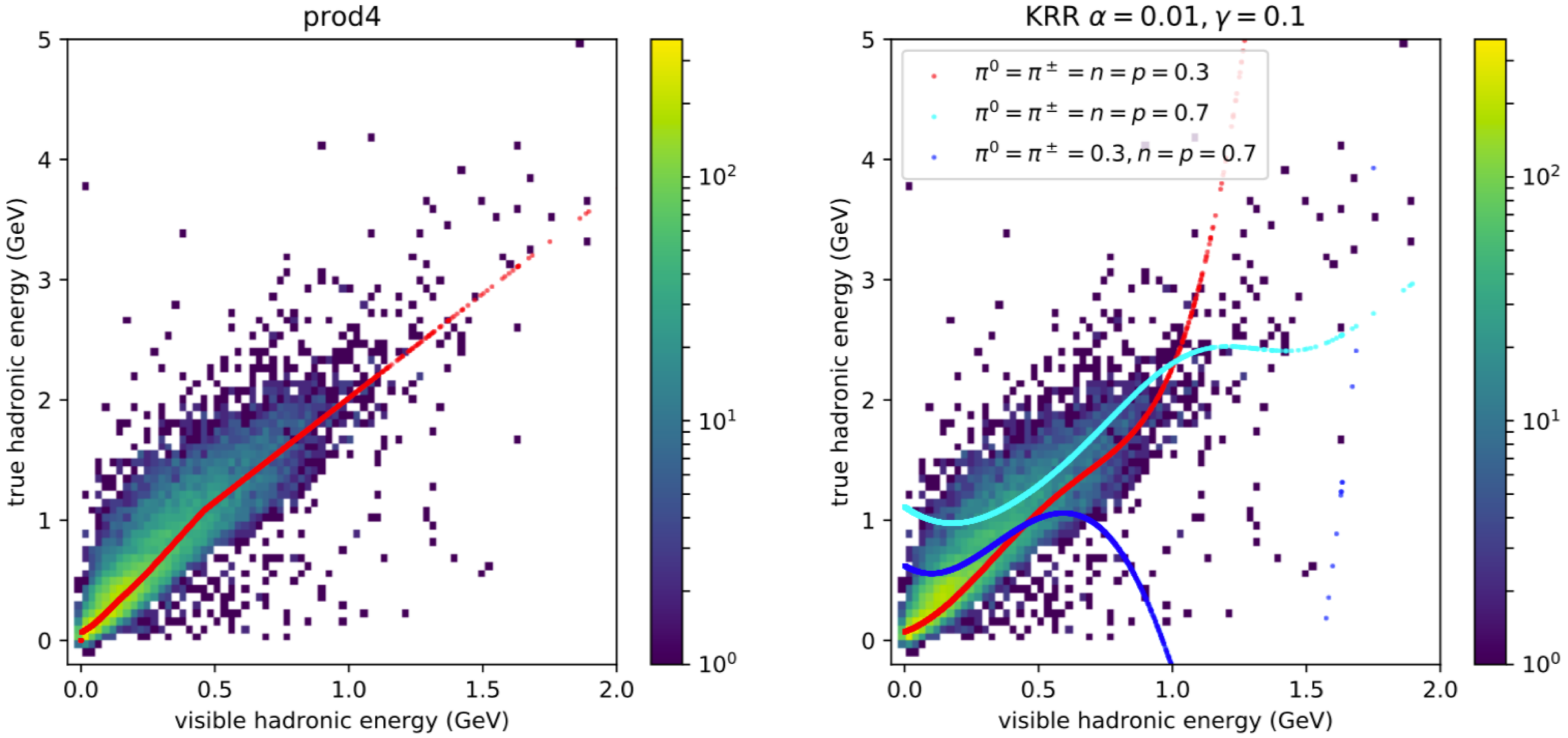


- proton score
- neutron score
- π^0 score
- π^+ score
- π^- score

- Utilize CVN final state score.
- For each event, I requested these five scores from CVN output.

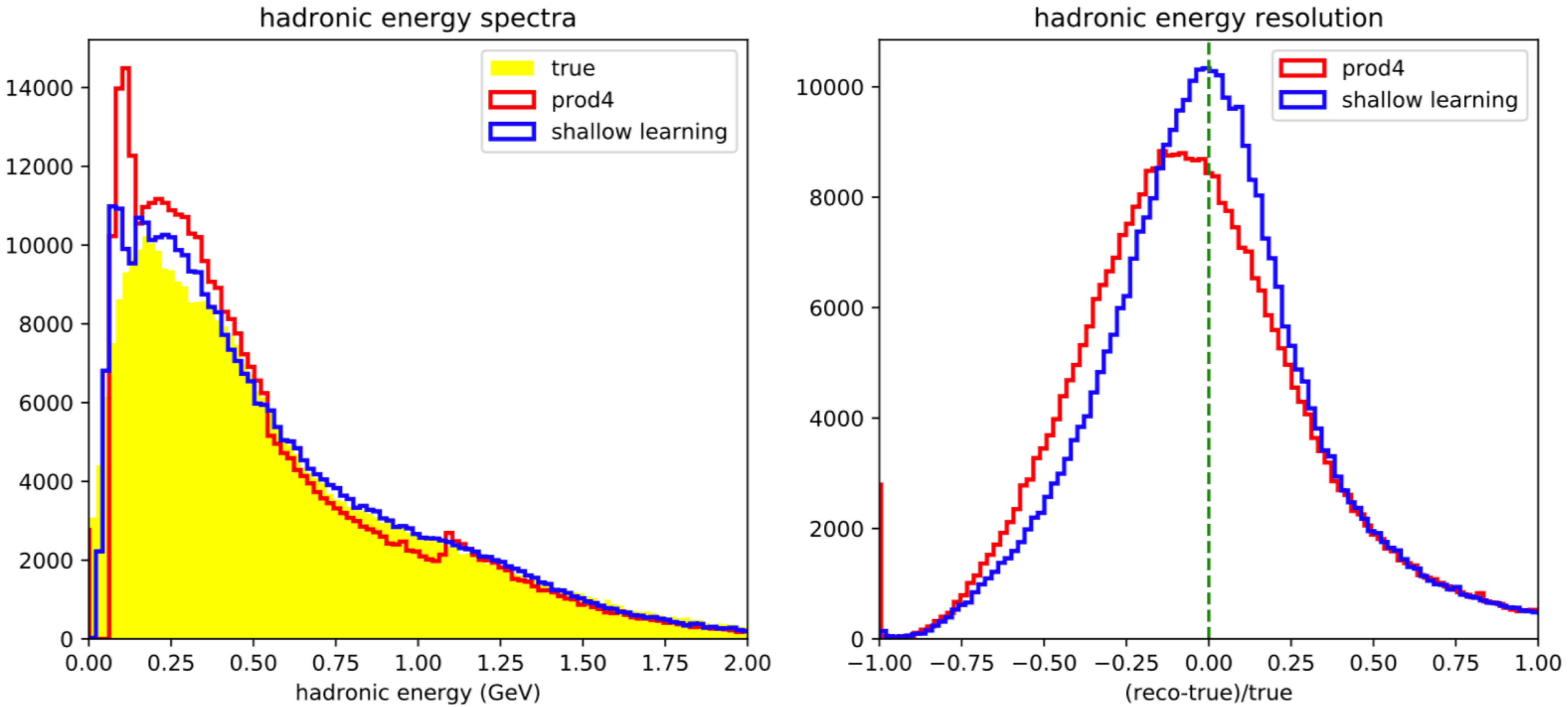
Illustration only. Not CVN structure.

Results



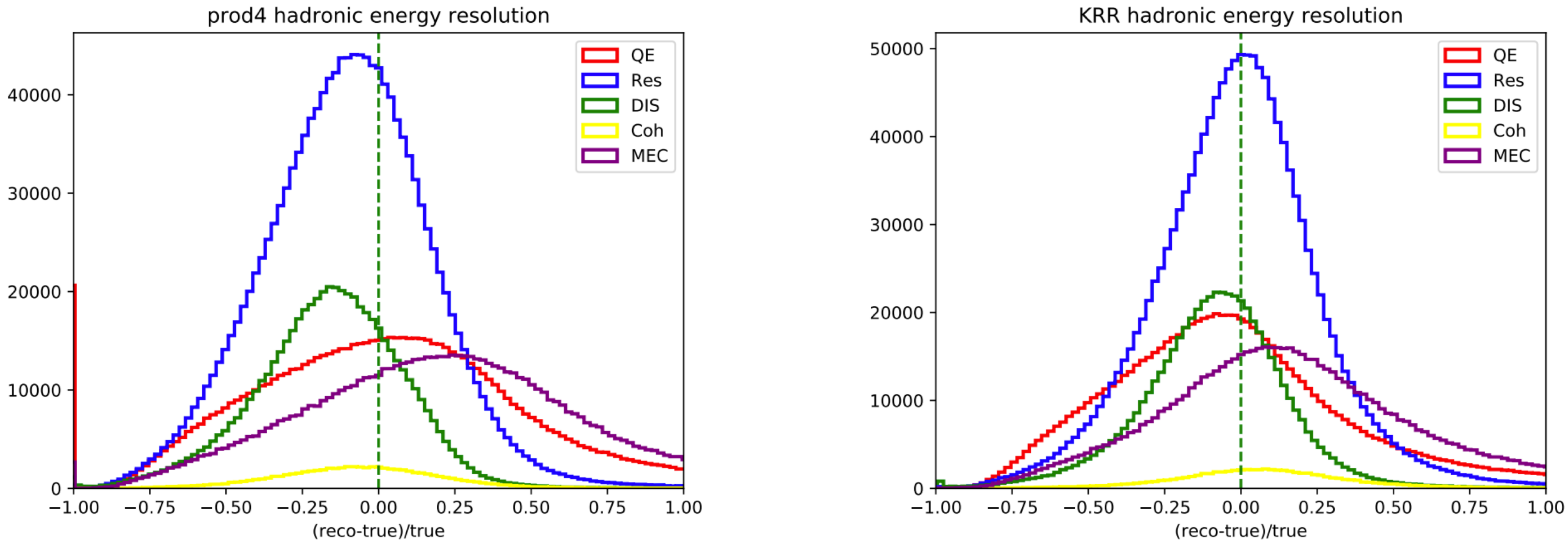
- Do the regression and obtain the regression hypersurface.
- We can try to "visualize" the hypersurface by coordinate curves.
- Don't panic about the weird curves. Those far off portions are simply not occupied by any events.
- The regression surface picks up spread that would otherwise be thought of as random fluctuation.

Results



	integral	mean	RMS	skewness	kurtosis
prod4	351169	-0.057	0.35	0.27	0.24
KRR	351803	-0.010	0.32	0.28	0.38

Breakdown by Interaction Mode



Performs better in all modes. CVN final state scores combined do carry some kind of interaction mode information.

Comparison with Other Methods

Compare with prod4

pros	cons
• Direct extension to prod4. Just adding more handles. • Very smooth with RBF kernel. • No kinks due to spline junctions. • Could incorporate quantiles into so called "categorical variables" and do regression at once. • Complementary to other methods such as BPF.	• Computationally heavy. It is $O(N^3)$ in time and $O(N^2)$ in memory.

Compare with CNN Regression

pros	cons
• Beautiful and transparent theory. • Adapt to changes in CVN structure since it is a posteriori method.	• Might not perform as well as CNN regression. • A posteriori method feels inferior to the integrated one. • No GPU support yet. Lack of scalability as of now.