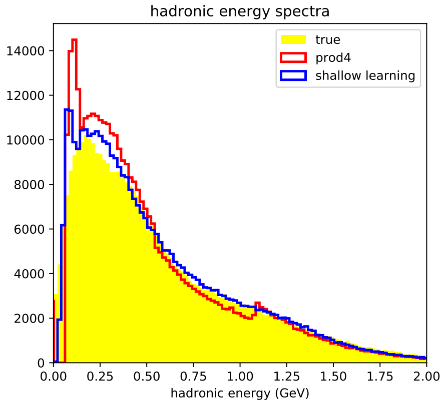
A Shallow Learning Hadronic Energy Estimator

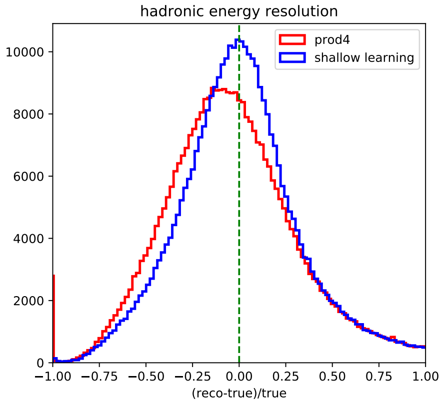
Shih-Kai Lin

Colorado State University

March 28, 2018

Some Teaser


hadronic energy spectra
true
prod4
shallow learning
14000
12000
10000
8000
6000
4000
2000
0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
hadronic energy (GeV)


hadronic energy resolution
prod4
shallow learning
10000
8000
6000
4000
2000
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
(reco-true)/true

Motivation

- NOvA has put a lot of effort into PID (classification) with the state-of-the-art machine learning techniques, but not as much in energy reconstruction (regression).
  - Except CVN regression (UCI)
- Why one more attempt at energy reconstruction besides the current prong-based one (Erica, Michael) and CVN regression?
  - It is a natural generalization to the current official spline fit.
    - In the sense that it also uses event-level variables to fit a regression function.
  - It has welcoming mathematical properties and beautiful underlying theory.
  - The nice mathematical properties are reflected in the results.
  - Better tools! There are many CVN final state particle scores available at the moment.

Shallow Learning

- As opposed to deep learning. Some authors use this term in literature.
  - I personally like it due to my initials...
- Below is why this class of methods is called shallow learning in contrast to deep learning:

deep architecture	CNN	$\longrightarrow$	many hidden layers	$\longrightarrow$	classification regression
shallow architecture	support vector machine kernel ridge regression	$\longrightarrow$	one hidden layer (feature map)	$\longrightarrow$	classification regression

- A cohort of *kernel methods* belongs to shallow architecture, among which the support vector machine was so popular that it almost killed neural network in the early 2000 before CNN took the crown.
- I will quickly go through the ideas behind kernel methods to justify the use of them for an energy estimator.

Ideas behind Kernel Methods – from the Most Basic

Linear regression:
Given N training samples, $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, N$, where $\mathbf{x}_i$'s $\in \mathbb{R}^\ell$ are predictor variables and y_i's $\in \mathbb{R}$ are target variables of training samples, find a linear function

$$f_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T\mathbf{x} \tag{1}$$

that minimizes the quadratic cost,

$$C(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}(y_i - \mathbf{w}^T\mathbf{x}_i)^2 \tag{2}$$

$\mathbf{w}$ that minimizes the cost function is readily found by solving the *normal equation*:

$$\mathbf{X}^T\mathbf{X}\mathbf{w} = \mathbf{X}^T\mathbf{y} \tag{3}$$

, where $\mathbf{X}$ is the so called *design matrix*,

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_N^T \end{pmatrix} \tag{4}$$

Ridge Regression and Dual Form

Very often, the predictor variables vary in a similar way, known as near collinearity. In such cases, $\mathbf{X}^T\mathbf{X}$ is almost singular, and the resulting $\mathbf{w}$ becomes highly sensitive to variations, leading to overfitting.
Applying Tikhonov regularization leads to ridge regression, namely, minimizing the cost function

$$C(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}(y_i - \mathbf{w}^T\mathbf{x}_i)^2 + \frac{1}{2}\alpha\|\mathbf{w}\|^2 \tag{5}$$

with the solution

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X} + \alpha\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \tag{6}$$

The cost function is convex, which guarantees a global minimum. (Very different from NN case.)
Note that $\mathbf{w}$ can be rewritten[1] as $\mathbf{w} = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \alpha\mathbf{I})^{-1}\mathbf{y}$.
For a test sample $\hat{\mathbf{x}}$, the predicted value $\hat{y} = \mathbf{w}^T\hat{\mathbf{x}} = \hat{\mathbf{x}}^T\mathbf{w} = \hat{\mathbf{x}}^T\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \alpha\mathbf{I})^{-1}\mathbf{y}$. Now, let $\mathbf{a} = (\mathbf{X}\mathbf{X}^T + \alpha\mathbf{I})^{-1}\mathbf{y}$, we arrive at the *dual form*:

$$\hat{y} = \sum_{i=1}^{N} a_i\mathbf{x}_i^T\hat{\mathbf{x}} \tag{7}$$

, i.e., instead of solving for $\mathbf{w}$, we solve for $\mathbf{a}$.

[1] See here for details.