Deep Learning Exotic Hadrons

[Pentaquark case study]

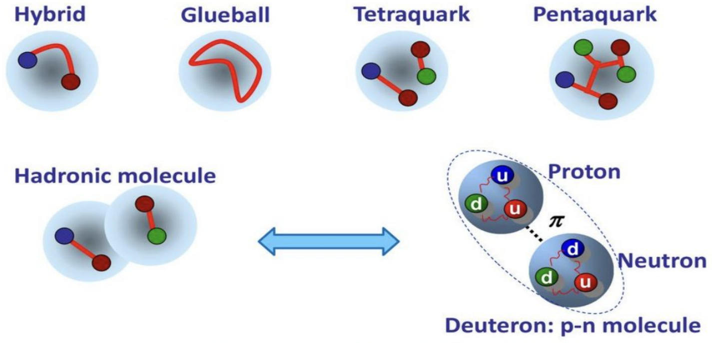
Lawrence Ng
Florida State University
January 26, 2022
Seminar

Overview

- Exotic hadrons
- $P_c(4312)^+$ pentaquark candidate
  - Microscopic nature
- Neural networks
  - Training set
  - Architecture
  - Feature impact
  - Results

Exotic Hadrons

- Existence predicted by Gell-mann and Zweig in 1964
- Systems of quark and gluons beyond the conventional meson ($q\bar{q}$) and baryon (qqq) states



https://indico.cern.ch/event/900972/attachments/2051939/3451258/ExoticStates_20200616.pdf

- Probing exotics are important in the study of non-perturbative QCD
  - Internal structure and dynamics
- Determining the nature of the state by studying the line shape

Pentaquarks

LHCb observed several peaks in $\Lambda_b^0 \rightarrow J/\psi p K^-$

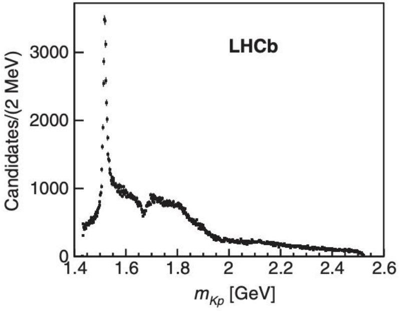
Phys. Rev. Lett. 122, 222001 June 2019

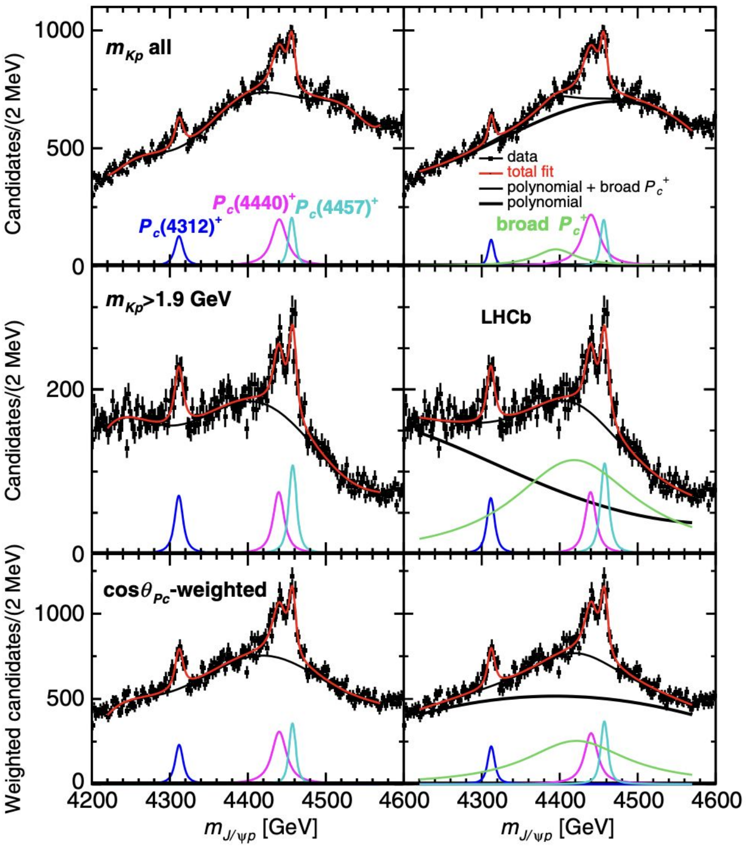
Significant $\Lambda^* \rightarrow pK^-$ populate the spectrum

Prominent peak = Λ(1520)

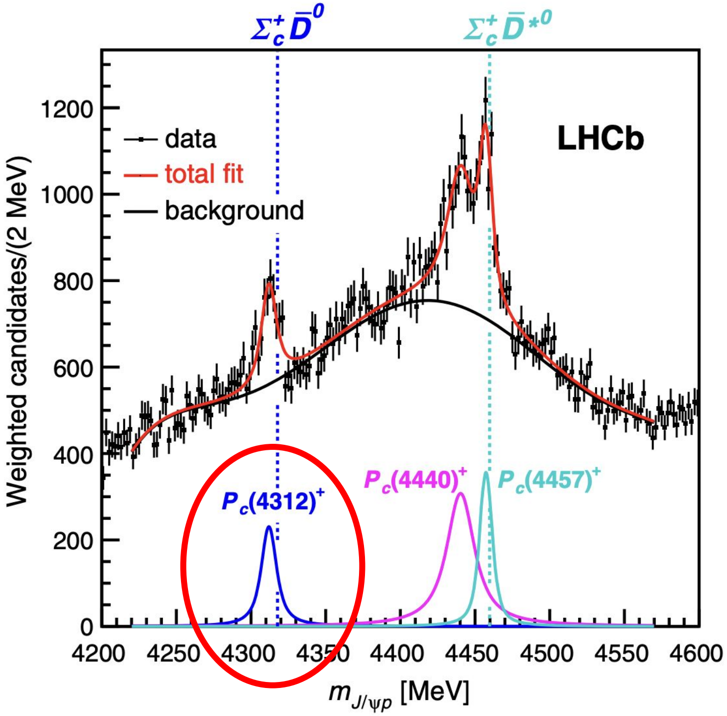
High-order polynomial vs low-order + Breit-Winger for $P_c(4380)^+$

Focus on isolated $P_c(4312)^+$





Lineshape → Microscopic origins



Top down:

Develop a microscopic model and fit to data

1. Assigns physical interpretation
2. Biased to assumed dynamics

Bottom up:

Develop minimally biased amplitudes based on basic principles

1. Harder to deduce the nature but possible
2. Less bias

Theoretical Model

Assume $P_c(4312)^+$ has a well-defined spin contributing to a single partial wave

- Intensity distribution

$$\frac{dN}{d\sqrt{s}} = \rho(s)[|F(s)|^2 + B(s)]$$

B(s) smooth contribution from other partial waves, ρ = phase space factor

$$F(s) = P_1(s)\,T_{11}(s)$$

P_1 smooth contribution for the production of $J/\psi p K^-$
T_{11} = $J/\psi p$ to $J/\psi p$ scattering

$$(T^{-1})_{ij} = M_{ij} - ik_i\delta_{ij} \qquad i, j = 1, 2 \text{ corresponding to the } J/\psi\, p \text{ and the } \Sigma_c^+ \bar{D}^0 \text{ channels}$$

k_i are the momenta

$$M_{ij}(s) = m_{ij} - c_{ij}s$$

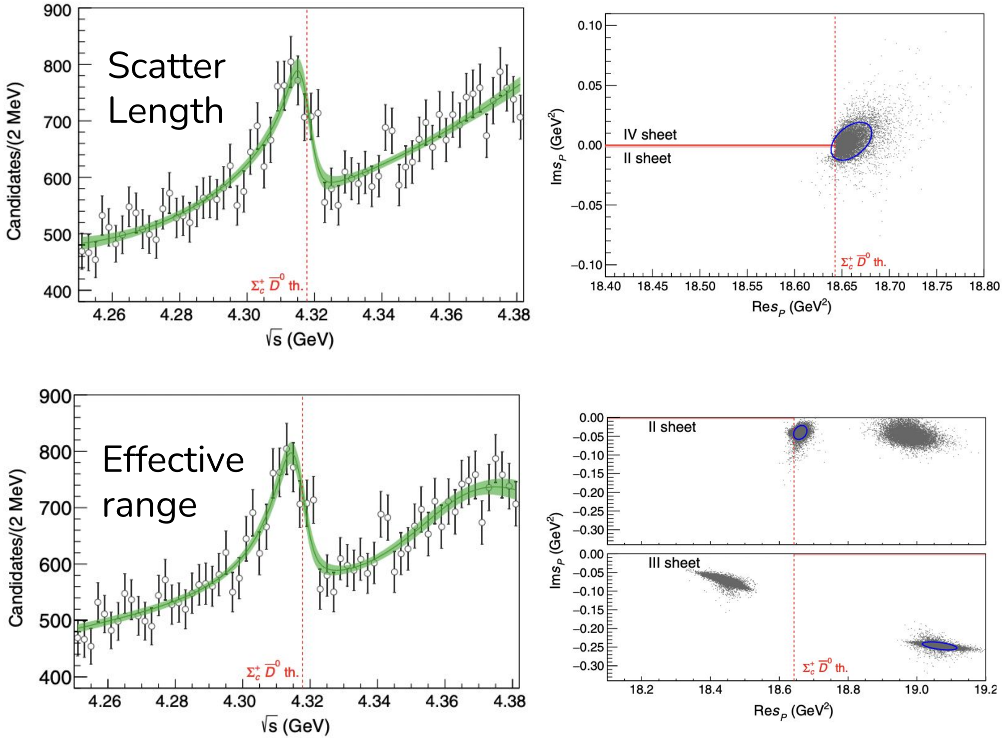
c_{ij}=0 (Scattering length approx) $c_{ij}\neq 0$ (Effective range approx)

Results

Minimally biased: Reaction amplitudes respecting S-matrix principles
Two channels: $J/\psi p$ (channel 1) and $\Sigma_c^+ \bar{D}^0$ (channel 2)

Location of the poles of T when channels decouple determine the nature

Scattering length / Effective range approximation both suggest $P_c(4312)^+$ is virtual (unbound) state - not strong enough to bind Σ_c^+ and $\bar{D}^0$ to form a molecule



Phys. Rev. Lett. 123, 092001 August 2019

Neural Networks

Use those minimally biased amplitudes to develop a training set

Alternative tool to analyze and interpret data as opposed to a standard χ^2 fit for a single hypothesis

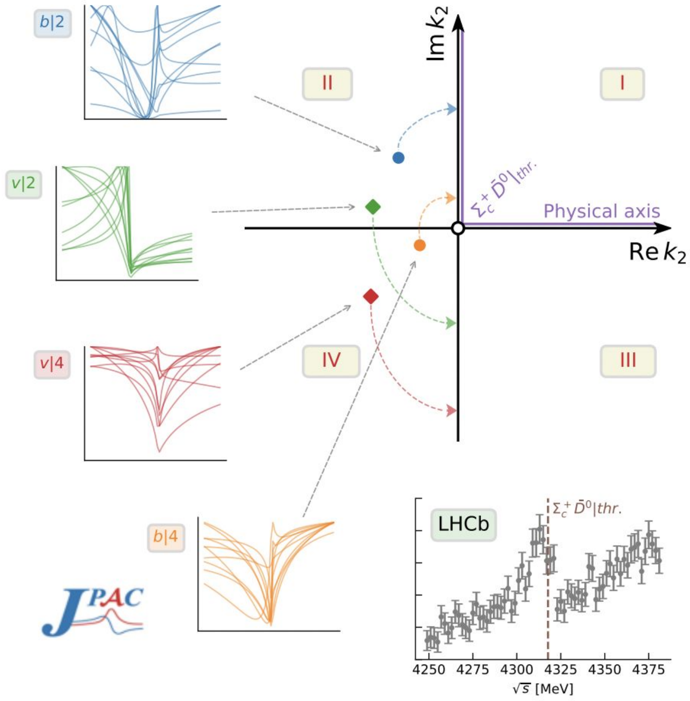
- Multi-class prediction
- Understand the impact of lineshape features to the class assignment

<u>Outline for the rest of the talk</u>

1. Training set
2. Neural Network Intro
3. Feature impact and explainable AI
4. Results

Training set

$$T(s) = \frac{M_{22} - ik_2}{(M_{11} - ik_1)(M_{22} - ik_2) - M_{12}^2}$$



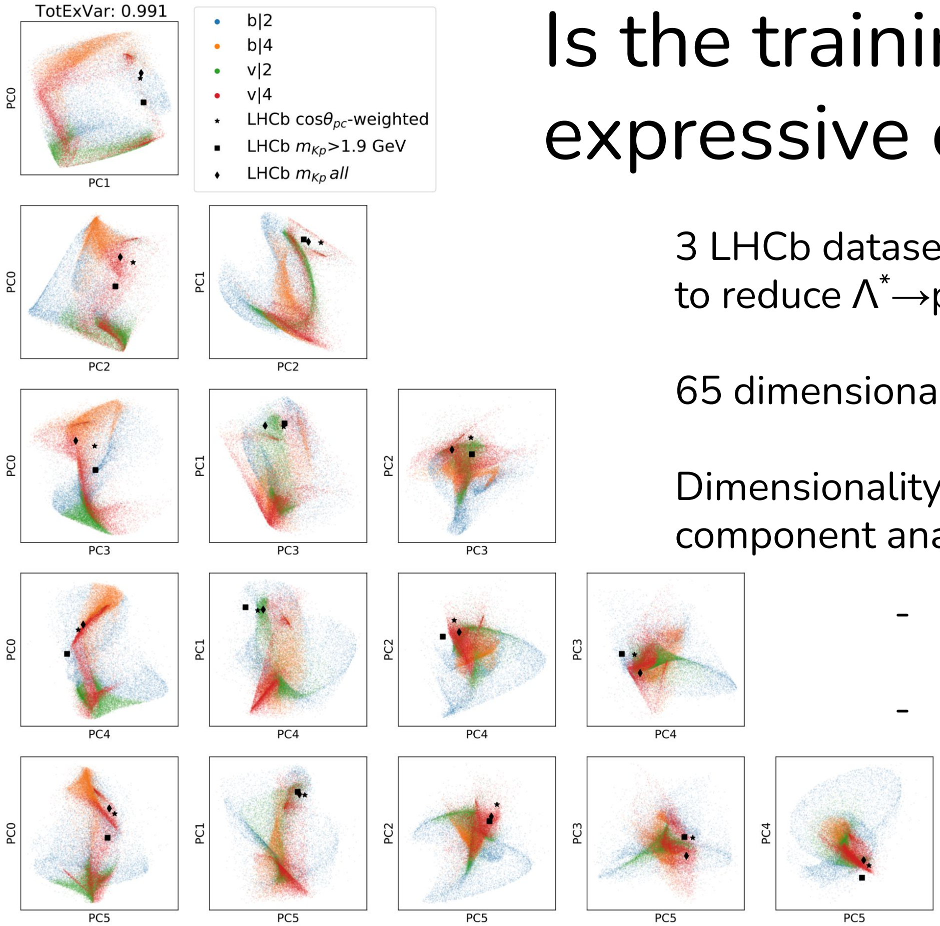
T(s) encodes dynamics of J/ψp rescattering
Poles = zeros of denominator

Complex momentum plane split into 4 sheets
Poles can only lay on II and IV sheet

Migration of poles when channels decouple ($M_{12} \to 0$)
$M_{22} < 0$ = bound state in $\Sigma_c^+ \overline{D}^0$ channel
$M_{22} > 0$ = virtual (unbound) state

Data generated for wide range of model parameters and over a larger energy range

Classify:
{bound, II}, {bound, IV}, {virtual, II}, {virtual, IV}
Inputs:
Spectrum (incorporating noise and resolution)



Is the training set / model expressive enough?

3 LHCb datasets for the $P_c(4312)^+$ where 2 attempts to reduce $\Lambda^* \rightarrow pK^-$ contributions

65 dimensional input (intensities at specific energies)

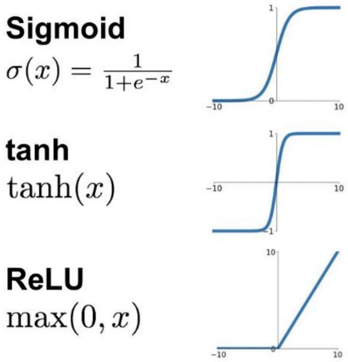
Dimensionality reduction for visualization (Principal component analysis) ~ Rotation

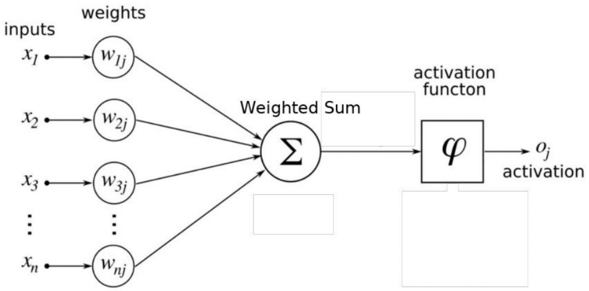
- Reduces input to 6 dimensions while retaining 99% of the variance
- Training set encompasses LHCb datasets

Deep Forward Neural Networks

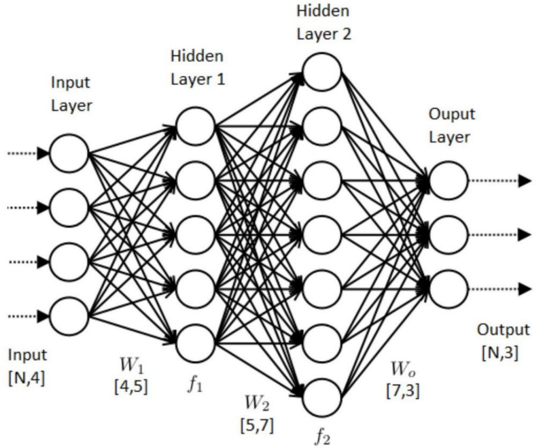
The objective of NN is to approximate a function by composing simpler affine functions followed by a non-linearity, φ

$y = f^n(f^{n-1}(\ldots f^1(x)))$ $\quad f^m(x) = \varphi(W_m x + b_m)$





https://www.nersc.gov/assets/GPUs-for-Science-Day/22-mustafa-mustafa.pdf



Loss Function Optimization

Maximum likelihood

Minimizing the dissimilarity between the empirical distribution and the model distribution **= minimize the cross-entropy**

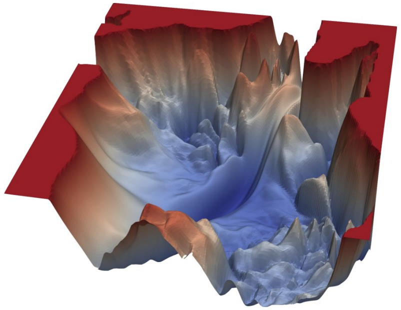
Binary Classification: Cross entropy between the empirical distribution and Bernoulli distribution -> Binary cross entropy loss

Regression: Cross entropy between the empirical distribution and Gaussian distribution -> Mean squared error loss

Matching output layer with loss function

1. Linear+MSE = Regression
2. Sigmoid+BCE = Binary classification
3. Softmax+MCCE = Multi-class classification

Optimization performed through variations of Gradient Descent



VGG-56 loss landscape: arXiv:1712.09913

Network Architecture

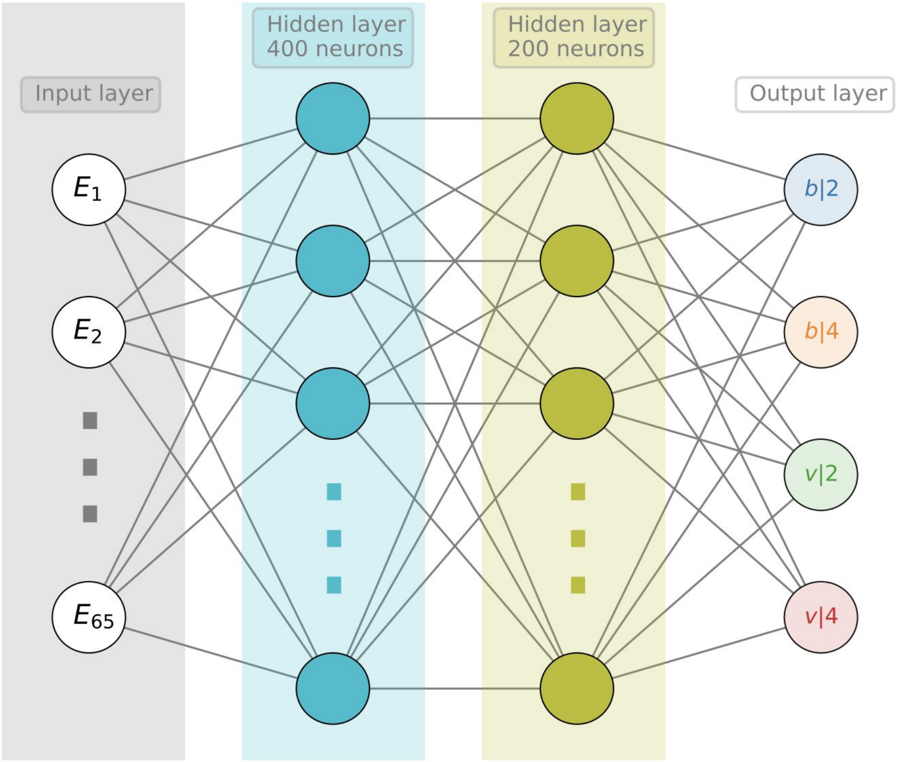
Fixed input size (65) + 2 hidden layers (ReLU activation) +
Output layer size (4, for each class)

Dropout included in between hidden layers

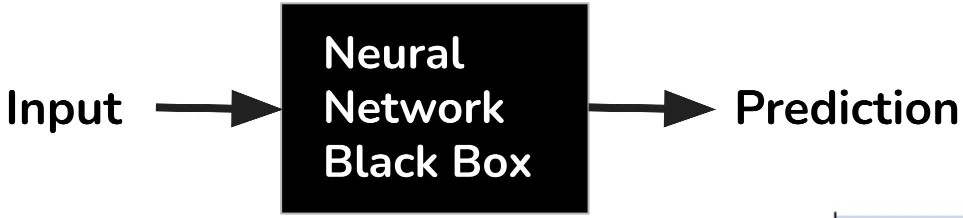
- Randomly zero nodes with some probability [0.2, 0.5]
- Prevents overfitting (regularization)
- Allows determination of classification probability uncertainty

Softmax output + Multiclass cross-entropy

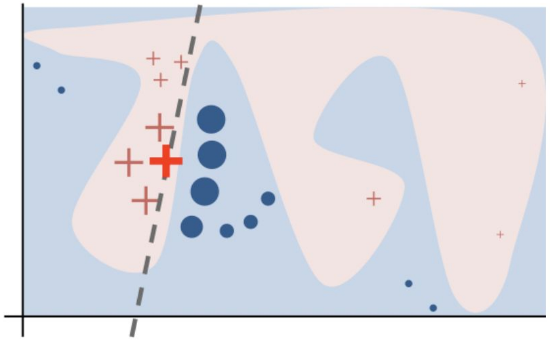
Optimized stochastically with Adam, batch-size 1024



Explainable AI



Create a **local** surrogate model (i.e. linear regression) that models the changes in prediction for small tweaks in the input
LIME - Locally Interpretable Model-agnostic Explanations



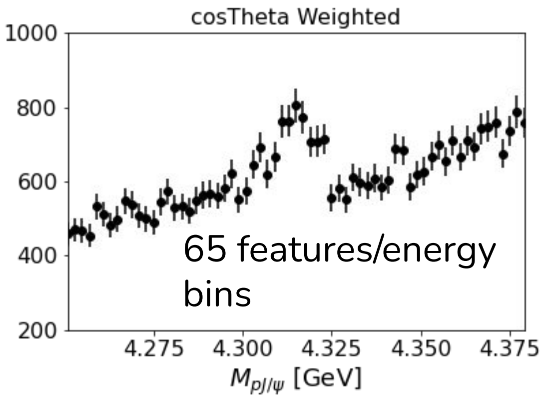
Simply query neural network response with different features active
SHAP - Shapley Additive EXplantions

- Theoretical guarantees

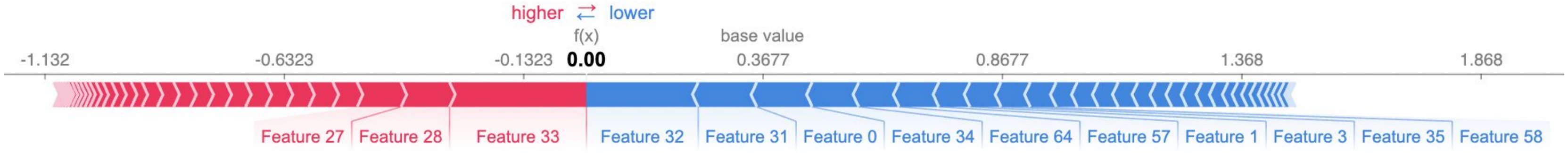
Game Theory: Determining player contribution

- How to split money among a group of players?
  - Determining contributions of a feature to the loss function
- Fairness:
  - Additivity - Sum of values = total money
  - Consistency - More contribution = more money
- Only ONE fair way of doing this - Shapely values
  - Lloyd Shapely won a nobel prize in economics
  - His father, Harlow Shapely - Astronomer - first to determine correct position of the sun in the Milky way
  - Harlow's student, Georges Lemaitre - first derived Hubble's Law and first estimation of hubble constant in 1927

Shapely Values



- ❖ Average marginal contribution across all feature coalitions
- ❖ Coalition - Set of features of any size
- ❖ Marginal Contribution - Changes to prediction with feature included in a coalition
- ❖ Additive local explanations
  - ➢ For **each** spectrum we can determine how much each energy bin contributes to the overall response (classification probability)



Determining the energy/input window

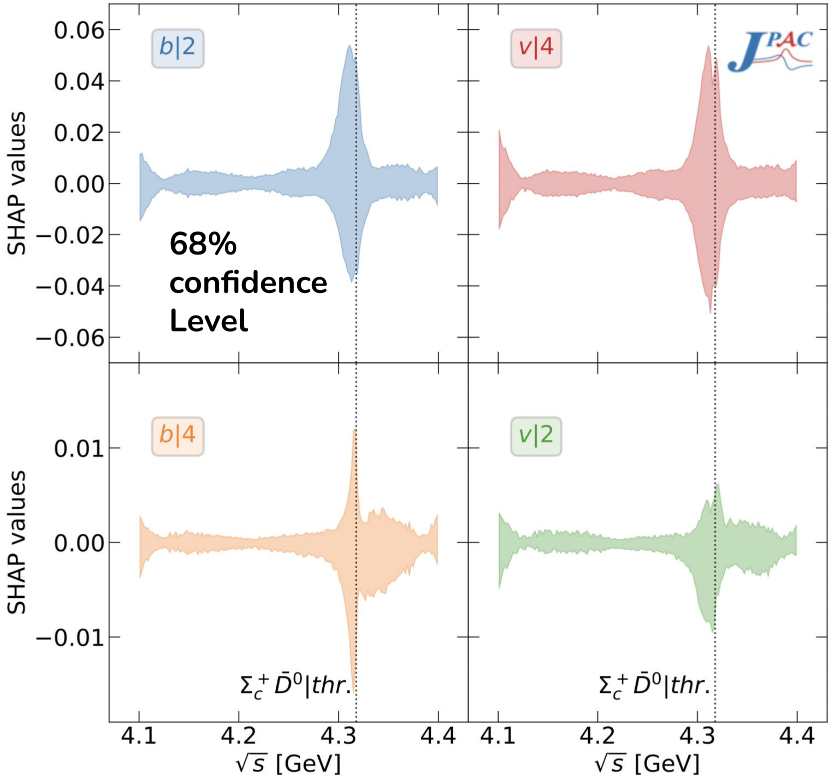
Typically the input window depends on some heuristic

Train a network using a wider energy/feature window = [4.1, 4.4] GeV
Use **SH**apley **A**dditive **E**xplanations (**SHAP python package**)
Determines feature importance

+(-) SHAP values push a network to predict into(outside) a given class

Large abs(SHAP) = high feature importance

Select energy region → [4.251, 4.379] GeV
+ Retrain



Network Performance

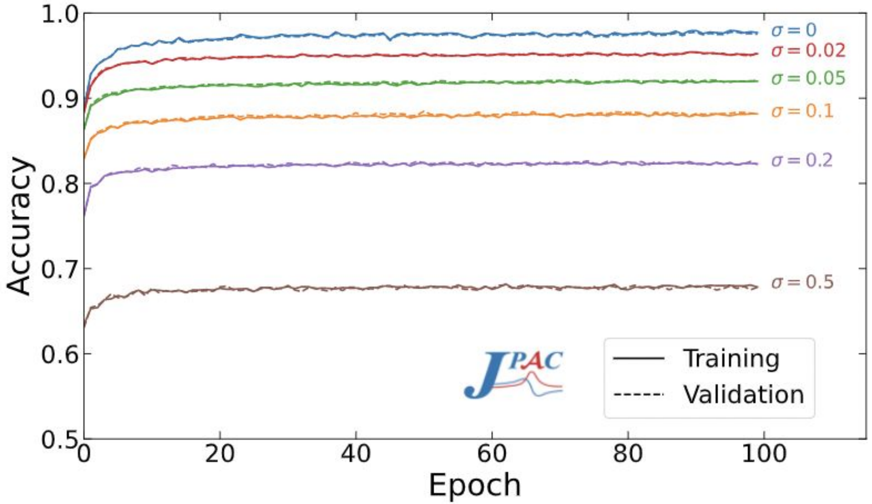
Network trained using various amounts of noise

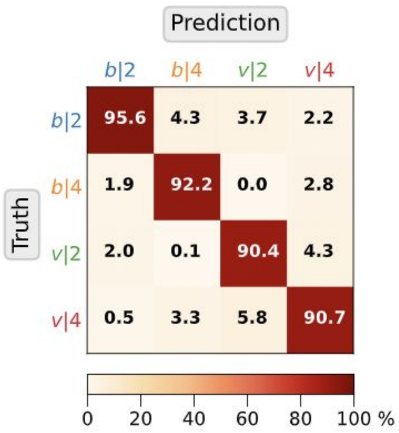
- LHCb data ~ 5% noise
- Network saturates > 90% accuracy

Confusion matrix for 5% noise normalized column-wise

Probability point estimates for LHCb data

	$b\|2$	$b\|4$	$v\|2$	$v\|4$
$\cos\theta_{P_c}$-weighted	0.6%	$<$ 0.01%	1.1%	98.3%
$m_{Kp} > 1.9\,\mathrm{GeV}$	1.4%	$<$ 0.1%	1.6%	97.0%
m_{Kp} all	5.4%	$<$ 0.1%	21.0%	73.6%





Exploring Uncertainties

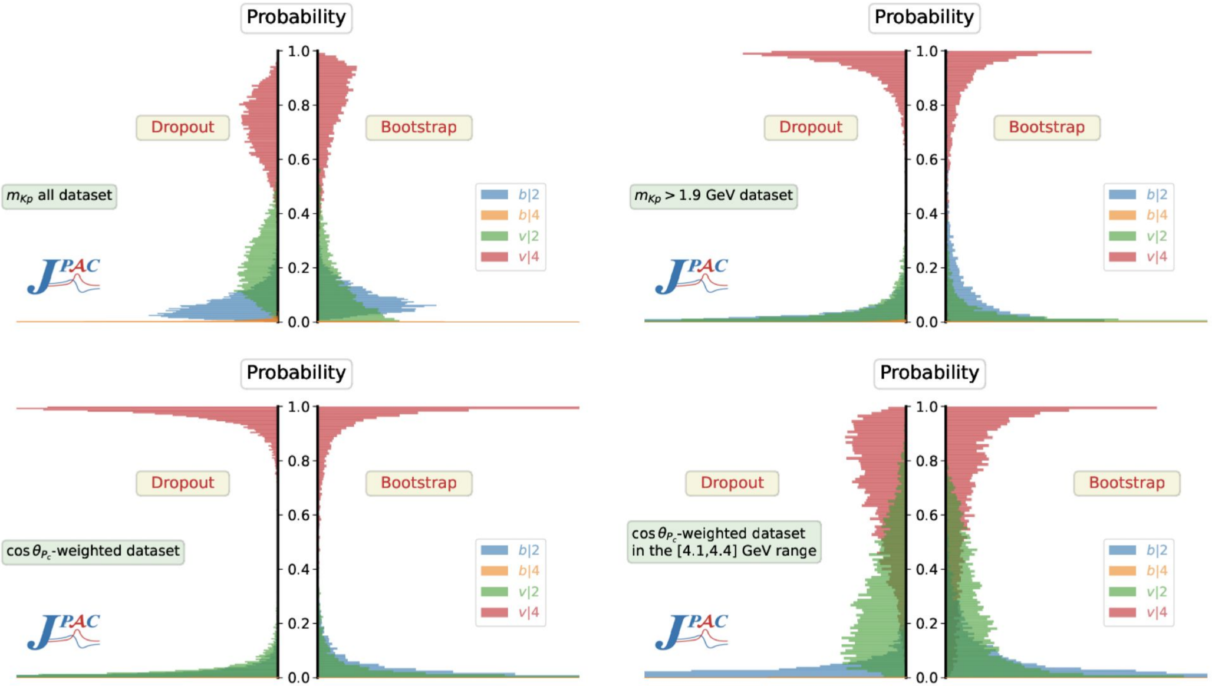
Dropout

Pass 5000x LHCb central values through the network with dropout turned on
Approximate deep Gaussian process a Bayesian probabilistic model
arXiv:1506.02142

Bootstrap

Resample LHCb data around its uncertainties, pass through network with dropout off

Good agreement between approaches



Uncertainties on the softmax probabilities strongly favor v|4 class

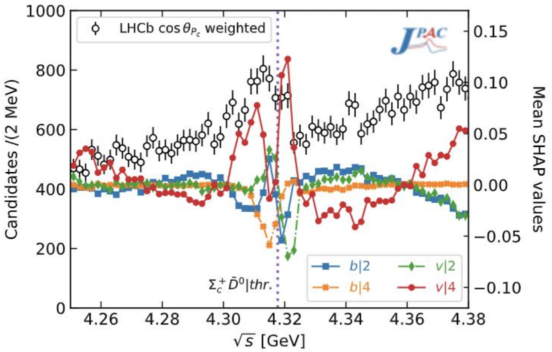
Feature attribution in LHCb data

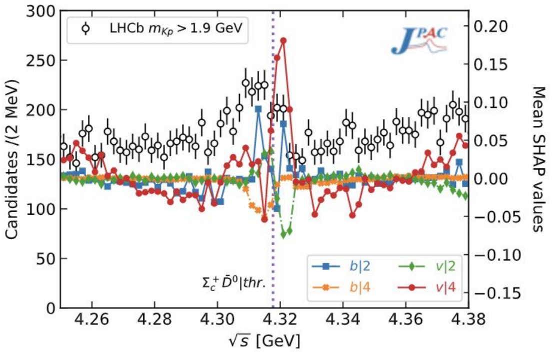
Bootstrapped LHCb data - determine Shapley values

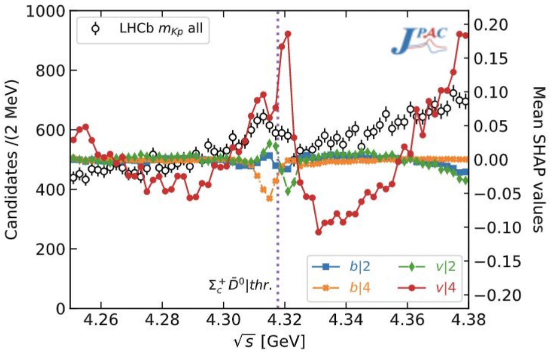
Features right above threshold favors v|4 and rejects v|2

Below threshold features reject both b|4 and favors v|4

Not removing the Λ^* resonances greatly affects classification impact above threshold







Future outlook

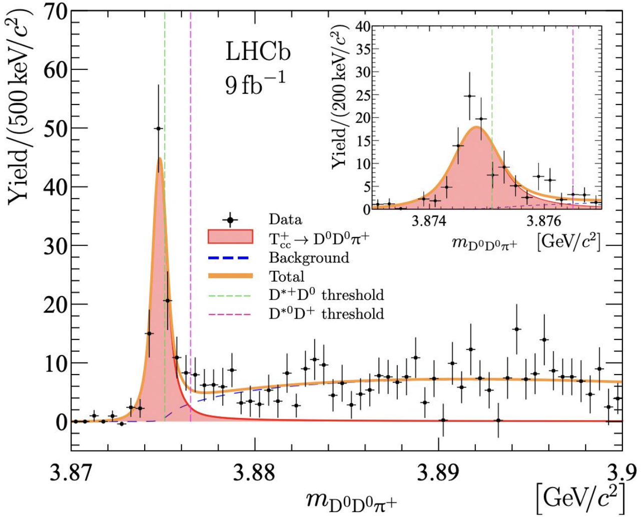
Current analysis assumes scatter length approximation

- Extend to effective range to incorporate genuine pentaquark states

Study other exotic hadron candidates

- $a_0(980)/f_0(980)$ meson-meson molecule?
- $T_{cc}^{+}(3875)$: Tetraquark candidate with minimal quark content $cc\bar{u}\bar{d}$

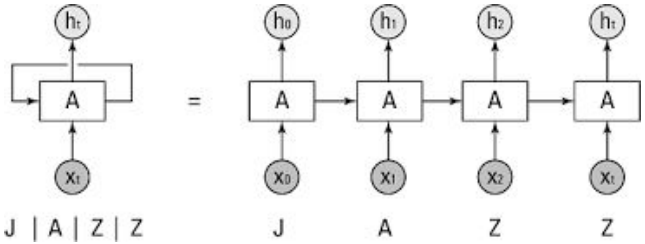
LHCb-PAPER-2021-031



Future outlook

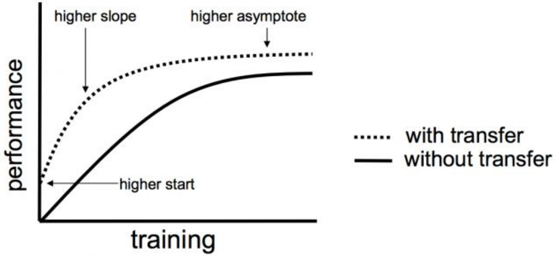
Currently, the network has a fixed input (65 energy bins)
Sequence Learning

- Natural representation for spectra
- RNN, LSTM, Transformer



Transfer Learning

- Reuse parts of a pretrained network to improve learning process with other models



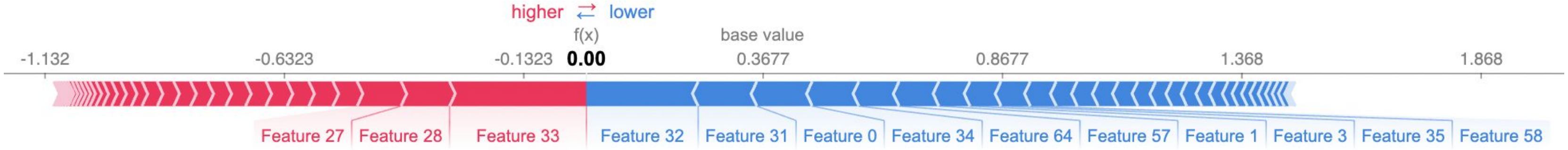
Conclusion

- Prescription to develop a neural network to investigate exotic lineshapes
- Incorporates noise and resolution
- Shapley values to determine regions importance
- Prediction uncertainty quantification through dropout and bootstrapping
- Pentaquark case study favors virtual state interpretation of $P_c(4312)^+$
- Numerous future prospects

Backup

Shapely Values

- ❖ Additive local explanations
  - ➢ For **each** spectrum we can determine how much each energy bin contributes to the overall response (classification probability)



- ❖ **Force plot** on a given model for class: 2|-
  - ➢ Base values = probability of average input
  - ➢ f(x) is output probability
  - ➢ Feature 33 (4.317 GeV) strongest feature increasing probability
  - ➢ Feature 32 (4.315 GeV) strongest feature decreasing probability
  - ➢ Lots of small contributions from tails, total contribution is very significant

Notes

Two channel amplitude coupling

$J/\psi p$ (channel 1) and $\Sigma_c^+ \bar{D}^0$ (channel 2)

$$\frac{dN}{d\sqrt{s}} = \rho(s)[|F(s)|^2 + B(s)]$$

Differential cross section

ρ = phase space

F(s) = P_c with definite spin, single wave

B(s) = bkgnd, all other partial waves

$$F(s) = P_1(s)T_{11}(s),$$

P_1 = production of $J/\psi pK^-$

T_{11} = $J/\psi p$ to $J/\psi p$ scattering (resonance masses included here)

T_{11} can only have poles on II and IV sheet and not on III

$$(T^{-1})_{ij} = M_{ij} - ik_i\delta_{ij}$$

Here $k_i = \sqrt{s - s_i}$ with $s_1 = (m_\psi + m_p)^2$

$s_2 = (m_{\Sigma_c^+} + m_{\bar{D}^0})^2$;

$$M_{ij}(s) = m_{ij} - c_{ij}s$$

M - effective range expansion

Case A: cij = 0 - scattering length approx

Two pairs of conjugate poles

Case B: cij floating - 4 poles - description of genuine pentaquark states

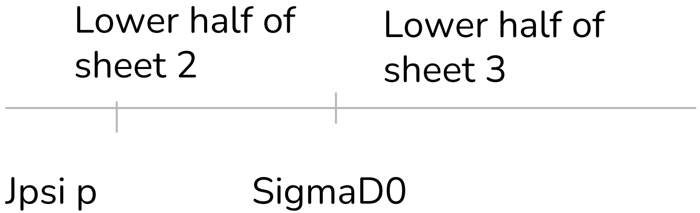
Both cases show a pole

Case A: pole on 4th sheet

Case B: pole on 2nd sheet

Both suggest a virtual state

Poles here will be breit-wigners



Moving to the upper half of sheet 2 past the higher threshold - enters upper half of sheet 4

Pole here will be a cusp

Both attractive and repulsive potentials can produce poles.

Coupling of two channels is turned off

the pole could either move to the real axis of the physical sheet below the heavier threshold, thus representing a bound molecule

move onto the real axis of the unphysical sheet, corresponding to an unbound, virtual state

Case A: $c_{ij} = 0$

Citations

LHCb paper

https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.122.222001

JPAC Pentaquark paper

https://journals.aps.org/prl/pdf/10.1103/PhysRevLett.123.092001

More SHAP notes

Exponential in time to get shap values for combinatorial subsets. How to nullify features when calculating the marginal contribution

LIME creates a surrogate model of the data distribution at a local point. Locality defined by a kernel - distance metric. LIME = SHAP under a specific kernel

KernelExplainer - Works for all models but only an approximation (not all subsets made) - Need to determine some way to fill missing values (some background data - like median)
TreeExplainer - if model is tree based then we can extract SHAP values exactly in polynomial time (using memory). Missing values a dealt with by traversing both paths at a branch point
DeepExplainer - For deep learning - calculates SHAP values in local/small parts of the network (i.e. a single linear neuron) and uses DeepLIFT to backprop the local SHAP values to calculate the SHAP value

Be careful with highly correlated data. When subsets are created we cannot really control what correlated variables are shown and some SHAP values might seem low. SHAP is additive so we can sum contributions of those correlated variables

Can also compare with random dataset

Resources:
https://github.com/slundberg/shap
https://slundberg.github.io/shap/notebooks/plots/decision_plot.html
https://christophm.github.io/interpretable-ml-book/shap.html
https://www.actuaries.digital/2021/02/05/explainable-ml-a-peek-into-the-black-box-through-shap/

https://www.youtube.com/watch?v=0yXtdkIL3Xk