A Statistician's View on Model-Independent Searches of New Physics at the Large Hadron Collider

Mikael Kuusela

Department of Statistics and Data Science,
Carnegie Mellon University

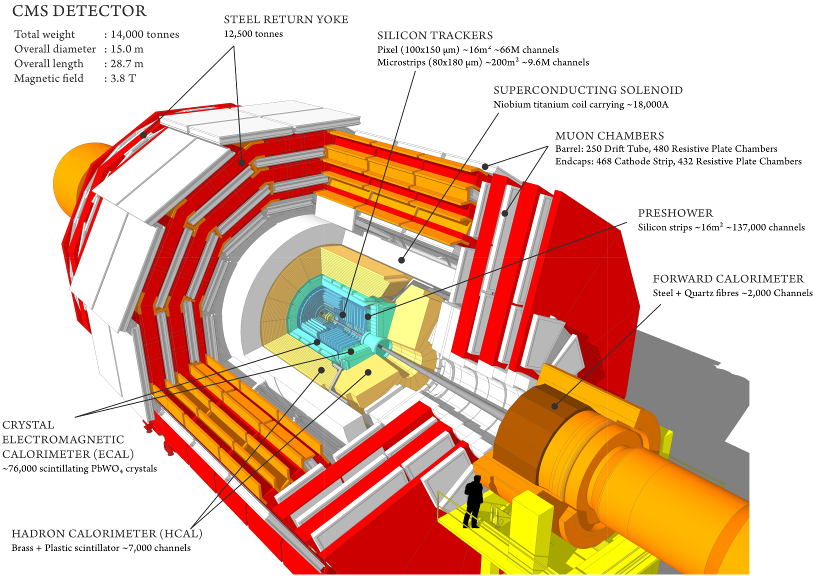
AISSAI Anomaly Detection Workshop
Clermont-Ferrand, France

March 4, 2024

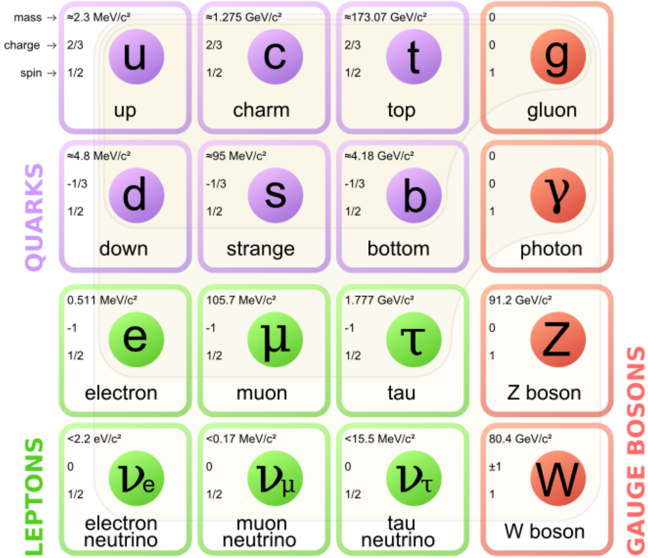
Joint work with: Purvasha Chakravarti, Jing Lei and Larry Wasserman

CERN and the Large Hadron Collider (LHC)

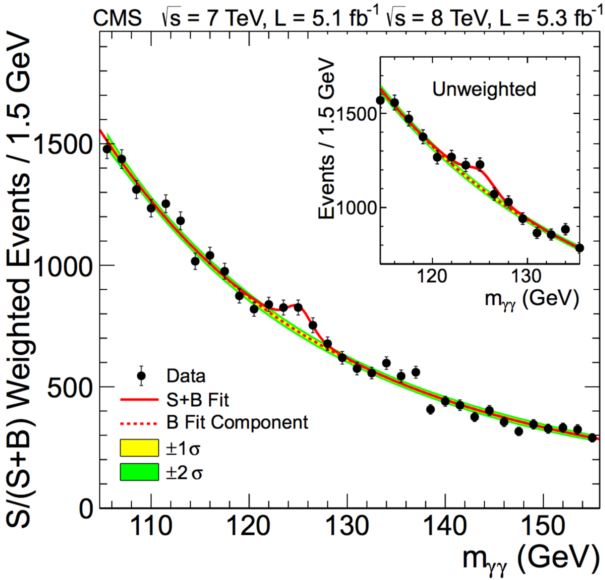
CMS experiment at the LHC



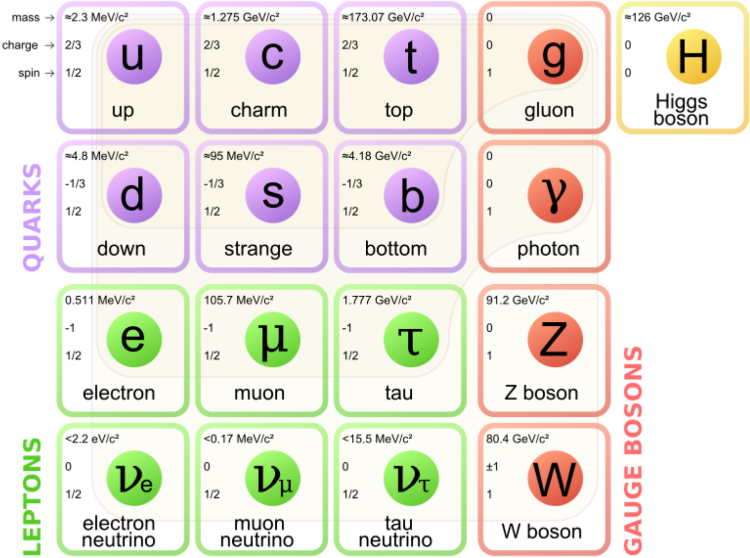
The Standard Model of particle physics (until 2012)



Discovery of the Higgs boson



The Standard Model of particle physics (since 2012)



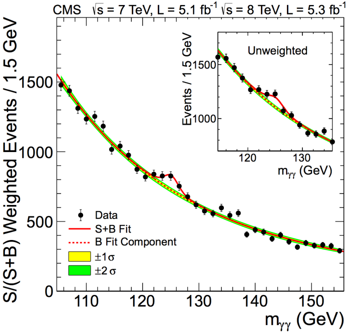
Hypothesis testing for discovery of new physics

Searches of new phenomena at the LHC usually boil down to testing for the presence of a signal distribution over a background of known SM physics:

- Known physics: $p_b(z)$
- New signal: $p_s(z)$
- Nature: $q(z) = (1-\lambda)p_b(z) + \lambda p_s(z)$

Want to test $H_0 : \lambda = 0$ vs. $H_1 : \lambda > 0$

If one rejects H_0 at a high enough significance level, then one might proceed to claim discovery of new physics



Model-dependent classifier-based tests

Most of these tests are done in the model-dependent mode, where the test statistic is optimized to have power for detecting a specific signal

Relevant datasets:

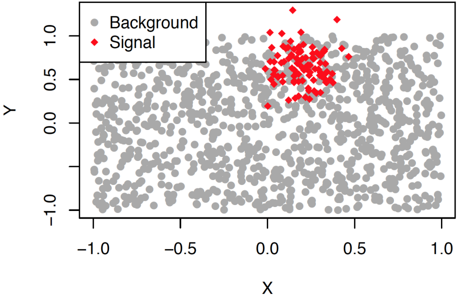
$$
\begin{aligned}
\text{Training background:} \quad & \mathcal{X} = \{X_1, \ldots, X_{m_b}\}, \qquad && X_i \sim p_b \\
\text{Training signal:} \quad & \mathcal{Y} = \{Y_1, \ldots, Y_{m_s}\}, \qquad && Y_i \sim p_s \\
\text{Experimental data:} \quad & \mathcal{W} = \{W_1, \ldots, W_n\}, \qquad && W_i \sim q = (1-\lambda)p_b + \lambda p_s
\end{aligned}
$$

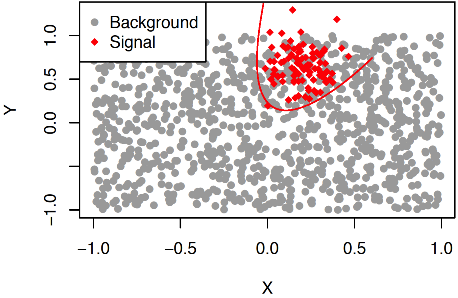
Basic idea: use $\mathcal{X}$ and $\mathcal{Y}$ to find the optimal test for detecting p_s

When the data space is high-dimensional, this is usually done using machine learning classifiers:

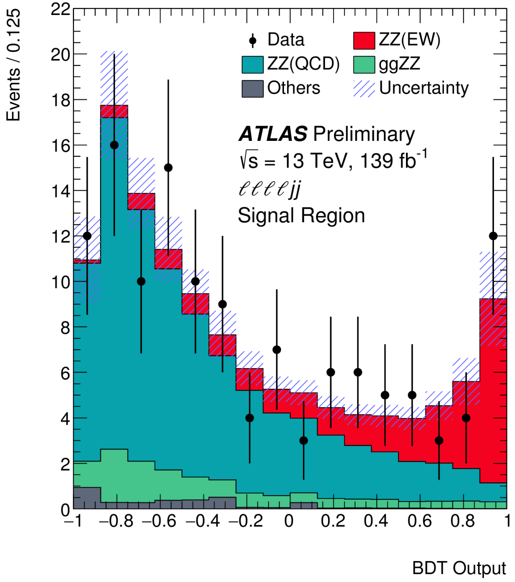
1. Train a supervised classifier to separate $\mathcal{X}$ from $\mathcal{Y}$
2. Use the classifier output to test for the presence of signal in $\mathcal{W}$

Classifier training





Classifier output



Some options for the test:

- Counting experiment in the highest purity output bin(s)
- Cut on the classifier output; test using the resulting signal-enriched sample
- LRT: Use the connection of the classifier output to the likelihood ratio
- ...

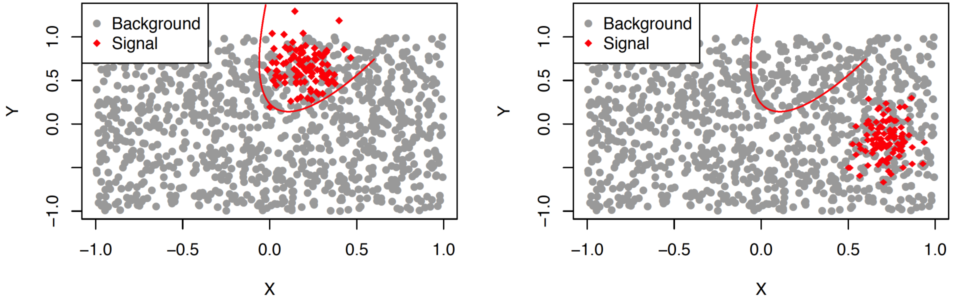
Testing when the signal is misspecified

To perform these tests, we need to assume that we can reliably simulate data from both p_b and p_s

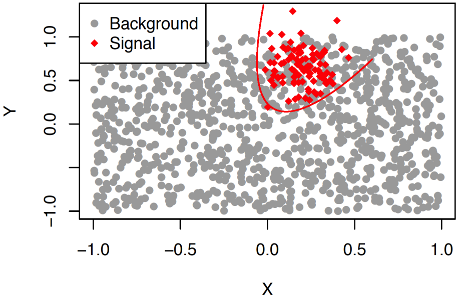
However, when either or both of these simulators are unreliable / systematically misspecified / unavailable, we need to consider alternative strategies for performing the test

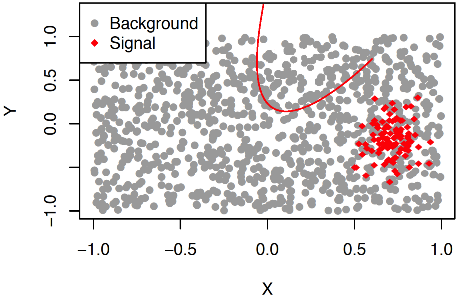
Specifically, if the test is optimized for a misspecified p_s, it may have little to no power for detecting an actual signal

Systematically misspecified signal



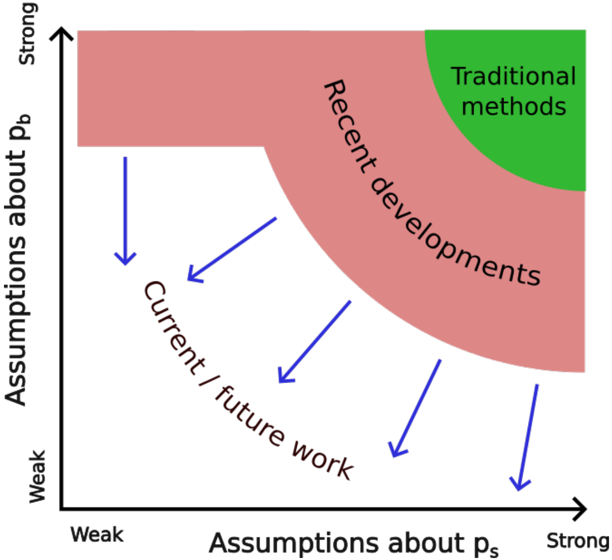
Systematically misspecified signal



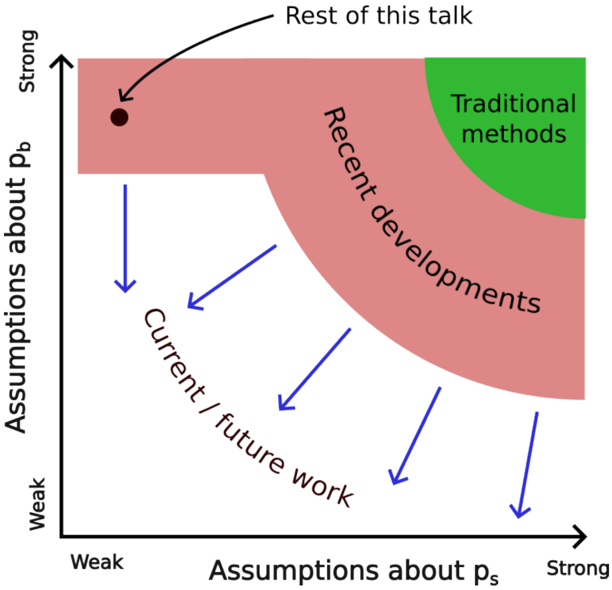


⇒ How to obtain an omnibus test that has power for a wide range of signals, even in high-dimensional situations?

Landscape of model-independent methods



Landscape of model-independent methods



Model-independent search

For the rest of this talk, we focus on a particular variant of model-independent searches of new physics

We assume that we have a reliable sample from p_b but we do not assume access to a training sample from p_s

$\rightarrow$ Provides sensitivity for unexpected or misspecified signals

Available datasets:

$$\begin{aligned} \text{Training background:} \quad & \mathcal{X} = \{X_1, \ldots, X_{m_b}\}, \qquad && X_i \sim p_b \\ \text{Experimental data:} \quad & \mathcal{W} = \{W_1, \ldots, W_n\}, \qquad && W_i \sim q = (1-\lambda)p_b + \lambda p_s \end{aligned}$$

We only have access to $\mathcal{X}$ and $\mathcal{W}$; i.e., no direct access to p_b, q, p_s or λ

Task 1: We want to understand if $\mathcal{W}$ shows evidence for the presence of p_s

Task 2: We want to understand what λ and p_s look like

Related problems in statistics and ML

The model-independent search problem is closely related to a number of problems studied in statistics and machine learning

Specifically, it can be seen as an example of:

1. Two-sample testing (e.g., Kim et al. (2019, 2021)):
   $X_i \overset{\text{iid}}{\sim} p_1,\ Y_i \overset{\text{iid}}{\sim} p_2$, is $p_1 = p_2$?
2. Collective anomaly detection (e.g., Chandola et al. (2009)):
   Is there a *collection* of data points which taken together deviate from the anticipated data?

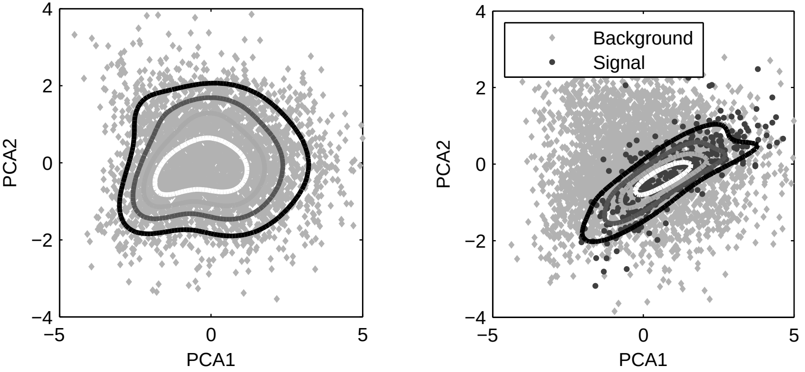
Notice that

model independent search $\neq$ outlier detection

Each signal event is typically indistinguishable from the background on its own; it is the collection of many signal events that defines the excess

Model-independent searches in low-dimensional spaces

In Kuusela et al. (2012) and Vatanen et al. (2012), we used Gaussian mixture models to first fit the background sample and then, given the background model, fit any anomalous signal present in the experimental sample



(a) Background model $p_b(z)$ (b) Signal model $p_s(z)$

This approach works fine in 2–3 dimensions but does not really scale to higher dimensions

Model-independent search using a semi-supervised classifier

What to do when the data space has more than just a couple of dimensions?

$\rightarrow$ Use machine learning classifiers!

Basic idea: Train a classifier h to separate the background data $\mathcal{X}$ from the experimental data $\mathcal{W}$

- Under H_0, the classifier should not be able to separate $\mathcal{X}$ from $\mathcal{W}$
- So if the classifier is able to differentiate between these two samples, then that provides evidence for the presence of p_s

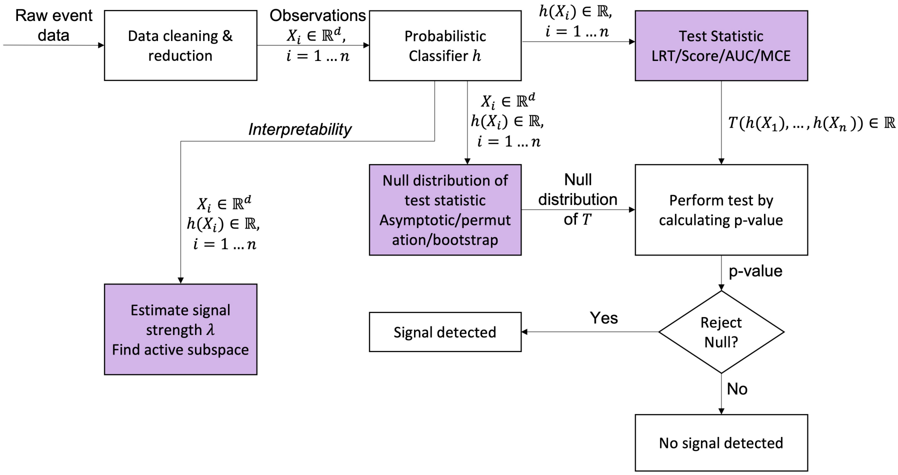
This basic strategy is closely related to recent work by D'Agnolo and Wulzer (2019) and D'Agnolo et al. (2021, 2022); see also Kim et al. (2019, 2021) for a similar approach in the two-sample testing literature

Our contributions

In Chakravarti et al. (2023), we make the following contributions:

1. We investigate various ways of obtaining a test statistic from the trained classifier $\widehat{h}$ as well as various ways of calibrating the tests
2. We propose a way to estimate the signal strength λ based on $\widehat{h}$
3. We propose a way to interpret $\widehat{h}$ using active subspaces

Overview of the approach



Classifier-based test statistics

Test statistics based on a classifier $\widehat{h}$ that is trained to separate the experimental data $\mathcal{W}$ from the background data $\mathcal{X}$:

1. Likelihood Ratio Test Statistic:

$$\mathrm{LRT} = 2\sum_i \log \widehat{\psi}(W_i),$$

where $\widehat{\psi}(z) = \frac{m_b}{n}\frac{\widehat{h}(z)}{1-\widehat{h}(z)}$ is a classifier-based estimate of the density ratio $\psi = q/p_b$

2. Area Under the Curve (AUC) Test Statistic:

$$\widehat{\theta} = \frac{1}{m_b\ n}\sum_i \sum_j \mathbb{I}\left\{\widehat{h}(W_j) > \widehat{h}(X_i)\right\}$$

Test $H_0 : \theta = 0.5$ versus $H_1 : 0.5 < \theta < 1$.

3. Misclassification Error (MCE) Test Statistic:

$$\widehat{\mathrm{MCE}} = \frac{1}{2}\left[\frac{1}{m_b}\sum_i \mathbb{I}\left\{\widehat{h}(X_i) > \pi\right\} + \frac{1}{n}\sum_j \mathbb{I}\left\{\widehat{h}(W_j) < \pi\right\}\right],\ \pi = n/(n+m_b)$$

Test $H_0 : \mathrm{MCE} = 0.5$ versus $H_1 : \mathrm{MCE} < 0.5$.

Calibration of the tests

In order to control the Type I error, we need to obtain the distribution of the test statistics under the null $H_0 : \lambda = 0$

Notice that, under the null, both $\mathcal{X}$ and $\mathcal{W}$ are samples from p_b

Three approaches:

1. Asymptotics: Can derive the asymptotic distribution for each of the test statistics; for example, for AUC, Newcombe (2006) showed that

$$\frac{\widehat{\theta} - 0.5}{\sqrt{V_0(\widehat{\theta})}} \rightsquigarrow N(0,1),$$

   for a certain $V_0(\widehat{\theta})$ under the null

2. Nonparametric bootstrap: Sample with replacement from $\mathcal{X} \cup \mathcal{W}$ and randomly label as either X's or W's

3. Permutation: Randomly permute the class labels in $\mathcal{X} \cup \mathcal{W}$

In-sample vs. out-of-sample evaluations

In practice, we need to be careful with in-sample vs. out-of-sample evaluation of the classifier $\widehat{h}$

- For each calibration method, we use half of the data to train the classifier and the other half to evaluate and calibrate the test statistics (sample splitting)
- For the permutation method, we also consider a variant where the classifier is evaluated in-sample, which requires retraining the classifier for each permutation cycle

Kaggle Higgs boson data

We compare these methodological choices using the Kaggle Higgs boson challenge dataset[1]

- Simulated $H \to \tau\tau$ events in ATLAS
- We select events with two jets and only consider primitive features (transverse momenta, MET, angles,...)
- 15 variables after accounting for rotational symmetry in ϕ
- 80,806 background events; 84,221 signal events
- Generate 50 "replicates" by sampling without replacement $m_b = 40{,}403$ background events, $m_s = 20{,}403$ signal events and $n = 40{,}403$ experimental events from the original samples
- We use Random Forest as the classifier h throughout

[1] https://www.kaggle.com/c/higgs-boson

Power of detecting a signal

Power of detecting a well-specified signal in the Kaggle Higgs boson data

		Signal Strength (λ)						
Model	Method	0.15	0.1	0.07	0.05	0.03	0.01	0
Supervised LRT	Asymptotic	100	100	96	62	18	18	6
	Bootstrap	100	96	78	58	6	0	0
	Permutation	100	98	98	86	28	6	0
Supervised Score	Bootstrap	64	66	74	50	18	0	0
	Permutation	94	92	100	92	80	24	12
Semi-Supervised LRT	Asymptotic	100	98	74	38	16	6	2
	Bootstrap	100	98	48	10	2	2	0
	Permutation	100	98	72	38	16	6	2
	Slow Perm	82	8	0	4	2	0	4
Semi-Supervised AUC	Asymptotic	100	96	78	32	14	4	2
	Bootstrap	100	98	70	32	20	6	2
	Permutation	100	98	68	32	20	4	2
	Slow Perm	100	100	94	56	20	8	4
Semi-Supervised MCE	Asymptotic	100	92	60	28	14	2	2
	Bootstrap	100	96	52	28	16	6	4
	Permutation	100	96	52	30	14	6	6
	Slow Perm	100	98	86	58	16	6	2

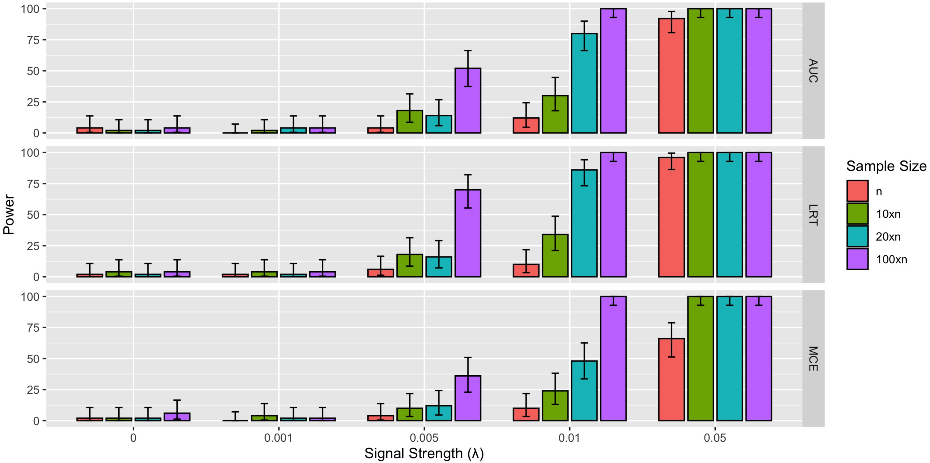
Power of detecting a signal

Power of detecting a misspecified signal in the Kaggle Higgs boson data

		Signal Strength (λ)						
Model	Method	0.15	0.1	0.07	0.05	0.03	0.01	0
Supervised LRT	Asymptotic	2	10	2	8	8	6	4
	Bootstrap	0	0	0	0	0	0	0
	Permutation	0	0	0	0	0	2	0
Supervised Score	Bootstrap	0	0	0	0	0	0	0
	Permutation	0	0	0	0	0	2	8
Semi-Supervised LRT	Asymptotic	100	100	100	82	18	4	4
	Bootstrap	100	100	100	60	4	2	0
	Permutation	100	100	100	82	18	4	2
	Slow Perm	100	100	78	22	2	4	6
Semi-Supervised AUC	Asymptotic	100	100	100	78	16	8	4
	Bootstrap	100	100	100	82	20	10	0
	Permutation	100	100	100	80	20	8	2
	Slow Perm	100	100	100	100	34	10	4
Semi-Supervised MCE	Asymptotic	100	100	100	66	24	6	4
	Bootstrap	100	100	100	62	16	6	4
	Permutation	100	100	100	62	14	6	4
	Slow Perm	100	100	100	98	22	8	2

Signal misspecified by transforming $\texttt{tau\_pt}^* = \texttt{tau\_pt} - 0.7\,(\texttt{tau\_pt} - \min(\texttt{tau\_pt}))$

Power as a function of sample size



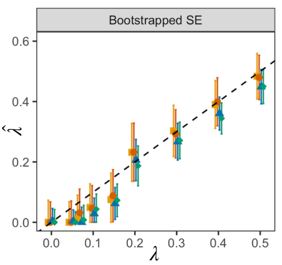
Power of the asymptotic model-independent tests for increasing sample sizes

Interpreting the semi-supervised classifier

We may want to be able to analyze the trained semi-supervised classifier $\widehat{h}$ to learn about the properties of the potential signal

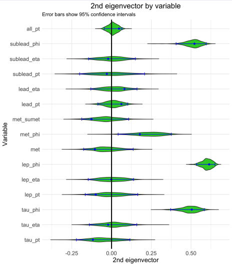
Signal strength

We estimate the signal strength λ from the classifier $\widehat{h}$ using the Neyman–Pearson quantile transform



Variable importance

We use the *active subspace* of the classifier to identify variable combinations that help separate the signal from the background



See the backups or Chakravarti et al. (2023) for more on these two approaches

Incorporating background systematics

The previous approach assumes that the training background sample $\mathcal{X}$ comes from the true background p_b

However, in practice the simulator for $\mathcal{X}$ is likely to be systematically misspecified

So the "signals" found might simply be due to background mismodeling
$\rightarrow$ Need to find a way to relax our assumptions about p_b

We can learn from techniques developed for model-dependent searches: template morphing, nuisance parameters, two-point systematics,...

For example, it might be possible to parameterize the systematics so that $p_b = p_b(\gamma)$, where $\gamma \in \Gamma$ is a nuisance parameter; we could then test

$$H_0 : q \in \{p_b(\gamma) : \gamma \in \Gamma\} \text{ vs. } H_1 : q \notin \{p_b(\gamma) : \gamma \in \Gamma\}$$

D'Agnolo et al. (2022) is an important first contribution on this, but further work is needed to incorporate nuisance parameters into other model-independent tests, including the test statistics discussed here

Conclusions

- Model-independent searches may be able to increase the sensitivity of LHC for unexpected or misspecified signals
  - Has received increased attention in recent years due to the absence of major new signals in model-dependent searches
- Model-independent searches in HEP are collective anomaly detection problems, but usually not outlier detection problems
- A whole spectrum of model-independent search techniques have been developed that differ in the strength of assumptions made about the signal and background distributions
- Machine learning classifiers are key for performing these searches in high-dimensional spaces
- Here we focused on a particular variant[2] of a classifier-based test and explored the effect of the choice of the test statistic, calibration method and in-sample vs. out-of-sample evaluation
- Current / future work: further relaxing assumptions about p_b and/or p_s

[2] P. Chakravarti, M. Kuusela, J. Lei, and L. Wasserman, Model-independent detection of new physics signals using interpretable semi-supervised classifier tests, The Annals of Applied Statistics, 17(4):2759–2795, 2023

References I

P. Chakravarti, M. Kuusela, J. Lei, and L. Wasserman, Model-independent detection of new physics signals using interpretable semi-supervised classifier tests, The Annals of Applied Statistics, 17(4):2759–2795, 2023.

V. Chandola, A. Banerjee, and V. Kumar, Anomaly detection: A survey, ACM Computing Surveys, 41:15:1–15:58, 2009.

R. T. D'Agnolo and A. Wulzer, Learning new physics from a machine, Physical Review D, 99(1):015014, 2019.

R. T. D'Agnolo, G. Grosso, M. Pierini, A. Wulzer, and M. Zanetti, Learning multivariate new physics, The European Physical Journal C, 81(1):1–21, 2021.

R. T. D'Agnolo, G. Grosso, M. Pierini, A. Wulzer, and M. Zanetti, Learning new physics from an imperfect machine, The European Physical Journal C, 82(275):1–37, 2022.

I. Kim, A. B. Lee, J. Lei, et al., Global and local two-sample tests via regression, Electronic Journal of Statistics, 13(2):5253–5305, 2019.

I. Kim, A. Ramdas, A. Singh, and L. Wasserman, Classification accuracy as a proxy for two-sample testing, The Annals of Statistics, 49(1):411 – 434, 2021.

References II

M. Kuusela, T. Vatanen, E. Malmi, T. Raiko, T. Aaltonen, and Y. Nagai. Semi-supervised anomaly detection–towards model-independent searches of new physics. In Journal of Physics: Conference Series, volume 368, page 012032. IOP Publishing, 2012.

R. G. Newcombe, Confidence intervals for an effect size measure based on the mann–whitney statistic. part 2: asymptotic methods and evaluation, Statistics in Medicine, 25(4):559–573, 2006.

T. Vatanen, M. Kuusela, E. Malmi, T. Raiko, T. Aaltonen, and Y. Nagai. Semi-supervised detection of collective anomalies with an application in high energy particle physics. In The 2012 International Joint Conference on Neural Networks (IJCNN), pages 1–8. IEEE, 2012.

Backup

Estimating the signal strength

Given a trained semi-supervised classifier $\widehat{h}$, how can we estimate the signal strength λ?

If we know that $p_s(z) = 0$ for some known z, then this is simple

Since

$$\psi(z) = \frac{q(z)}{p_b(z)} = \left(\frac{1-\pi}{\pi}\right)\left(\frac{h(z)}{1-h(z)}\right),$$

we obtain

$$\widehat{\lambda} = 1 - \left(\frac{1-\pi}{\pi}\right)\left(\frac{\widehat{h}(z)}{1-\widehat{h}(z)}\right),$$

for any z with $p_s(z) = 0$

However, in the model-independent setting, we may not know when $p_s(z) = 0 \rightarrow$ What to do?

Estimating the signal strength

Need to assume $\inf_z p_s(z)/p_b(z) = 0$ for identifiability; assume also $p_b, q > 0$ everywhere, for simplicity

Define the Neyman–Pearson Quantile Transform of z as:

$$\rho(z) = P_{X\sim p_b}\left(\frac{q(X)}{p_b(X)} \geq \frac{q(z)}{p_b(z)}\right) = P_{X\sim p_b}\left(\psi(X) \geq \psi(z)\right) = P_{X\sim p_b}\left(h(X) \geq h(z)\right)$$

Let g_q be the density function of $\rho(Z)$ when $Z \sim q$

Then it can be shown that g_q is monotonically decreasing and

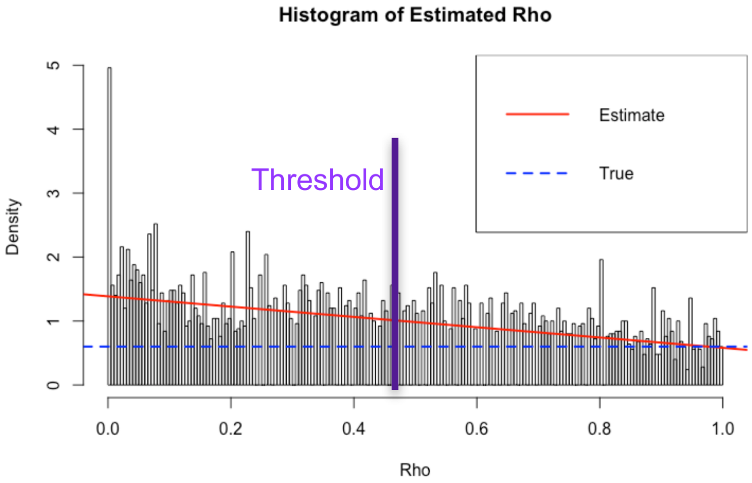
$$g_q(1) = 1 - \lambda$$

which allows us to estimate λ using $\widehat{\lambda} = 1 - \widehat{g_q}(1)$

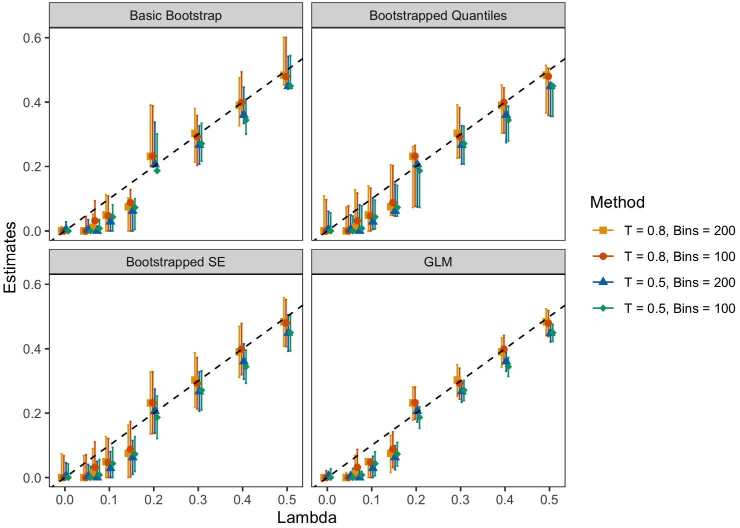
$\rightarrow$ We need to estimate a monotone density at its boundary

Estimating the signal strength

In practice, we form a histogram of $\rho(W_i)$ and estimate $g_q(1)$ using a Poisson regression on bins close to 1



Estimating the signal strength



Estimated λ vs. true λ with various uncertainty estimates

Active subspaces for interpreting the classifier

The fitted classifier surface $\widehat{h}$ contains information about how the experimental data $\mathcal{W}$ differs from the background data $\mathcal{X}$

How do we extract this information from $\widehat{h}$?

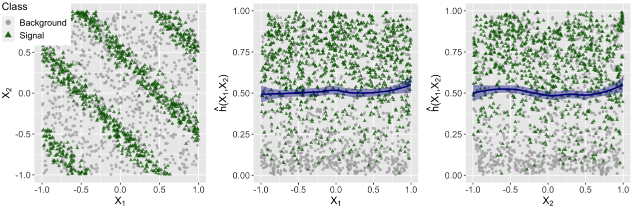
Could look at $\widehat{h}$ as a function of each input variable

But this might not reveal information contained in variable dependencies

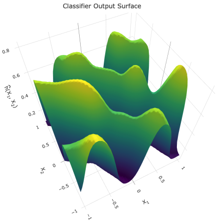
We propose to look at the *active subspace* of $\widehat{h}$ instead

Basic idea: perform PCA on the gradients $\nabla\widehat{h}(z)$ to reveal those directions in which the classifier surface changes the most

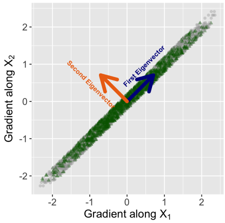
Active subspaces for interpreting the classifier



(a) X_1 versus X_2, $\widehat{h}(X_1, X_2)$ versus X_1 and $\widehat{h}(X_1, X_2)$ versus X_2



(b) Smoothed Classifier Surface



(c) PCA of the Standardized Gradients

Active subspaces for interpreting the classifier

In practice, we look at the gradients of

$$H(z) := \operatorname{logit}(\widehat{h}(z)) = \log\left(\widehat{h}(z)/(1-\widehat{h}(z))\right)$$

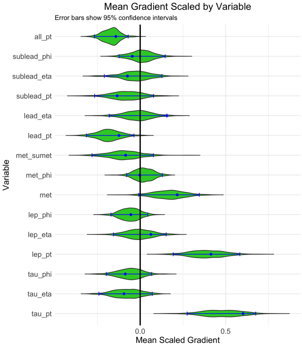
which are estimated by fitting a local linear regression on $H(Z_i)$ where $Z_i \in \mathcal{X} \cup \mathcal{W}$

Furthermore, we standardize the gradients by their estimated standard errors: $G(z) = \frac{\widehat{\nabla H(z)}}{\sqrt{\widehat{\mathrm{Var}}(\widehat{\nabla H(z)})}}$

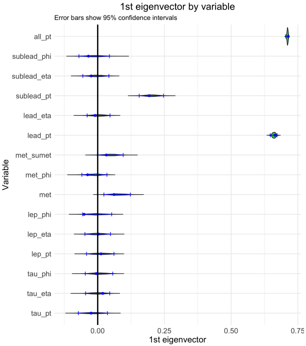
We then perform PCA on $G(Z_i)$: the mean of $G(Z_i)$ describes the slope of $H(z)$ and the principal components of $G(Z_i)$ capture the variation of $H(z)$ around the slope

Uncertainty estimates using bootstrapping

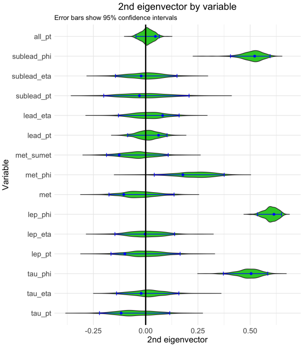
Active subspaces for interpreting the classifier



(a) Mean Gradient



(b) First Eigenvector



(c) Second Eigenvector

Density Ratios and Classifiers

In general, given two densities p and q and samples

$$X_1, \ldots, X_n \sim p$$

$$Y_1, \ldots, Y_n \sim q$$

Give labels:

$$\begin{array}{c|cccccc} & X_1 & \ldots & X_n & Y_1 & \ldots & Y_n \\ Z & 1 & \ldots & 1 & 0 & \ldots & 0 \end{array}$$

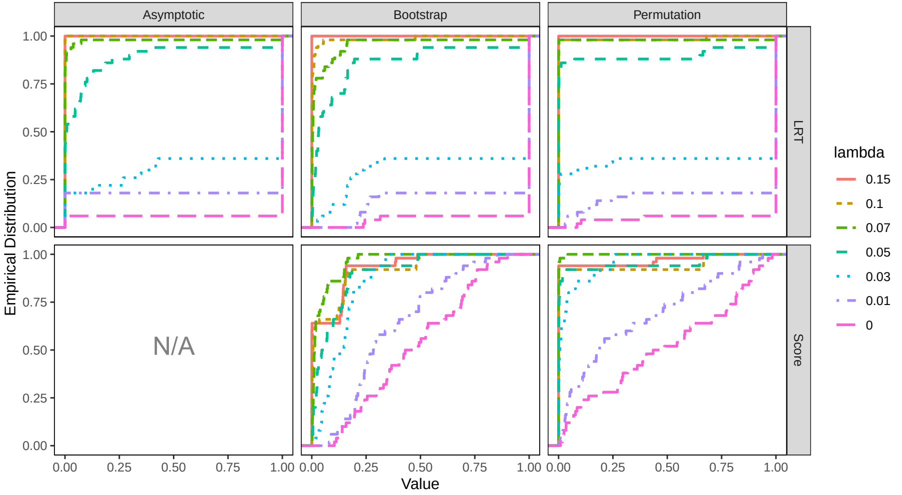
Classifier ψ:

$$\psi(u) = P(Z = 1|u) = \frac{p}{p+q}$$

and so

$$\frac{p}{q} = \frac{\psi}{1-\psi}.$$

p-value distributions for the supervised tests



p-value distributions for the semi-supervised tests