Model-Independent Search using Interpretable Semi-Supervised Classifier Tests

Purvasha Chakravarti

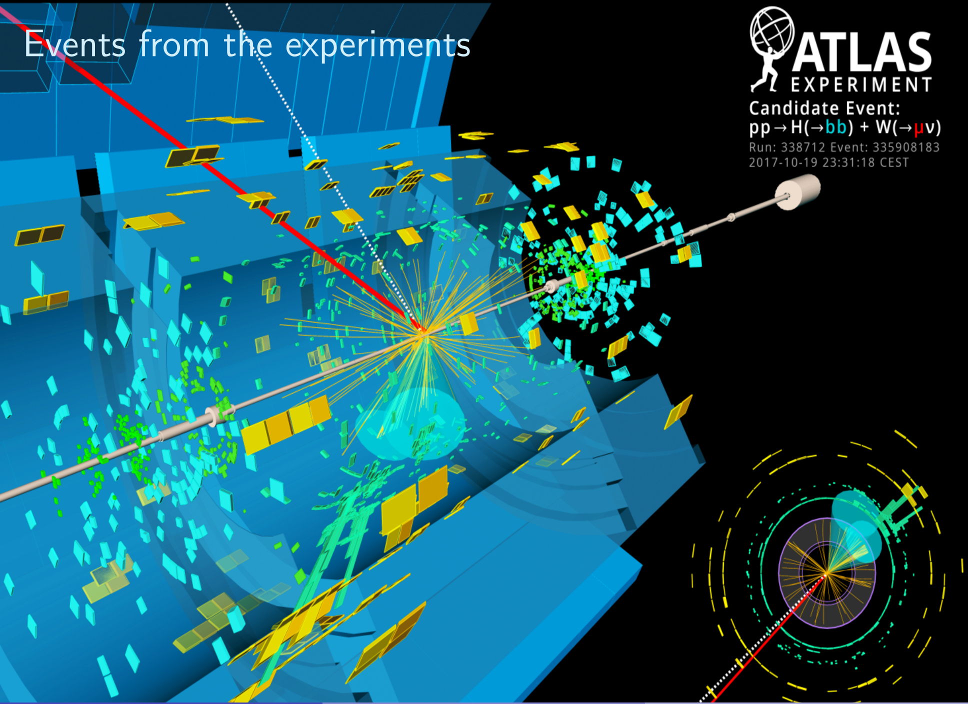
Department of Statistical Science
University College London

p.chakravarti@ucl.ac.uk

Systematic Effects and Nuisance Parameters in Particle Physics Data Analyses, Banff International Research Station
April 27, 2023

Joint work with Mikael Kuusela, Jing Lei and Larry Wasserman
Department of Statistics & Data Science
Carnegie Mellon University

Events from the experiments



Experimental data

Experimental data are generated from one of the two processes:
Background (p_b) - refers to the known physics (SM).
Signal (p_s) - represents an unknown possible particle or interaction not accounted for in the SM.

Experimental data

Experimental data are generated from one of the two processes:
Background (p_b) - refers to the known physics (SM).
Signal (p_s) - represents an unknown possible particle or interaction not accounted for in the SM.

Experimental data density: $q = (1 - \lambda)p_b + \lambda p_s$, $\lambda \in [0, 1]$.
No signal: $\lambda = 0$ or equivalently $q = p_b$, where λ is the signal strength.

Experimental data

Experimental data are generated from one of the two processes:
Background (p_b) - refers to the known physics (SM).
Signal (p_s) - represents an unknown possible particle or interaction not accounted for in the SM.

Experimental data density: $q = (1 - \lambda)p_b + \lambda p_s$, $\lambda \in [0, 1]$.
No signal: $\lambda = 0$ or equivalently $q = p_b$, where λ is the signal strength.

Testing for signal can be formulated as:

$$H_0 : \lambda = 0 \quad \text{versus} \quad H_1 : \lambda > 0.$$

Experimental data

Experimental data are generated from one of the two processes:
Background (p_b) - refers to the known physics (SM).
Signal (p_s) - represents an unknown possible particle or interaction not accounted for in the SM.

Experimental data density: $q = (1 - \lambda)p_b + \lambda p_s$, $\lambda \in [0, 1]$.
No signal: $\lambda = 0$ or equivalently $q = p_b$, where λ is the signal strength.

Testing for signal can be formulated as:

$$H_0 : \lambda = 0 \quad \text{versus} \quad H_1 : \lambda > 0.$$

This is equivalent to a two-sample testing problem

$$H_0 : q = p_b \quad \text{versus} \quad H_1 : q \neq p_b.$$

Objectives

- Model-Independent Signal Detection: Detect signal without assuming a signal model.

Objectives

- Model-Independent Signal Detection: Detect signal without assuming a signal model.

- Semi-Supervised Classifier Tests: Use a semi-supervised classifier to handle the high-dimensionality of the data.

Objectives

- Model-Independent Signal Detection: Detect signal without assuming a signal model.
- Semi-Supervised Classifier Tests: Use a semi-supervised classifier to handle the high-dimensionality of the data.
- Interpretability:

Objectives

- Model-Independent Signal Detection: Detect signal without assuming a signal model.
- Semi-Supervised Classifier Tests: Use a semi-supervised classifier to handle the high-dimensionality of the data.
- Interpretability:
    - Signal Strength Estimation: Estimate the signal strength in the data.

Objectives

- Model-Independent Signal Detection: Detect signal without assuming a signal model.
- Semi-Supervised Classifier Tests: Use a semi-supervised classifier to handle the high-dimensionality of the data.
- Interpretability:
  - Signal Strength Estimation: Estimate the signal strength in the data.
  - Active Subspace Methods: Characterize the signal and find subspaces that influence the classifier.

Model-dependent supervised methods (assume a signal model)

Two sources of data are at hand:

- Background + signal (MC simulations) sample - labelled observations

$$\text{Background:} \quad X_1, \ldots, X_{m_b} \sim p_b$$
$$\text{Signal:} \quad Y_1, \ldots, Y_{m_s} \sim p_s$$

Model-dependent supervised methods (assume a signal model)

Two sources of data are at hand:

- Background + signal (MC simulations) sample - labelled observations

$$\text{Background:} \quad X_1, \ldots, X_{m_b} \sim p_b$$
$$\text{Signal:} \quad Y_1, \ldots, Y_{m_s} \sim p_s$$

- Background + possible signal (experimental) sample - unlabelled observations

$$\text{Experimental:} \quad W_1, \ldots, W_n \sim q = (1-\lambda)p_b + \lambda p_s$$

Model-dependent supervised methods (assume a signal model)

Two sources of data are at hand:

- Background + signal (MC simulations) sample - labelled observations

$$\text{Background:} \quad X_1, \ldots, X_{m_b} \sim p_b$$
$$\text{Signal:} \quad Y_1, \ldots, Y_{m_s} \sim p_s$$

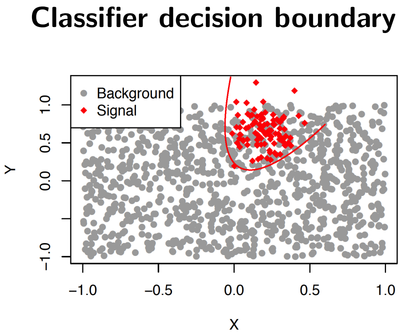
- Background + possible signal (experimental) sample - unlabelled observations

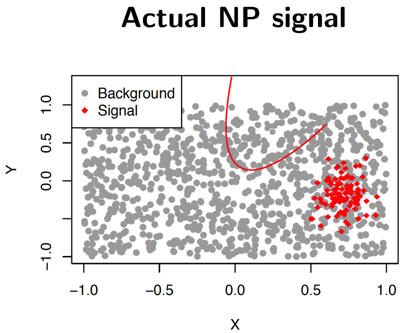
$$\text{Experimental:} \quad W_1, \ldots, W_n \sim q = (1-\lambda)p_b + \lambda p_s$$

Test $H_0 : \lambda = 0$ vs $H_1 : \lambda > 0$.

Train a classifier (h) to separate signal from background.

Motivation for model-independent methods: systematically misspecified signal





Model-independent semi-supervised methods (don't assume a signal model)

Two sources of data are at hand:

- Background (MC simulations) sample - labelled observations

$$\text{Background:} \quad X_1, \ldots, X_{m_b} \sim p_b$$

- Background + possible signal (experimental) sample - unlabelled observations

$$\text{Experimental:} \quad W_1, \ldots, W_n \sim q = (1-\lambda)p_b + \lambda p_s$$

Model-independent semi-supervised methods (don't assume a signal model)

Two sources of data are at hand:

- Background (MC simulations) sample - labelled observations

$$\text{Background:} \quad X_1, \ldots, X_{m_b} \sim p_b$$

- Background + possible signal (experimental) sample - unlabelled observations

$$\text{Experimental:} \quad W_1, \ldots, W_n \sim q = (1 - \lambda)p_b + \lambda p_s$$

Train a semi-supervised classifier (h) to separate experimental from background.

Model-independent semi-supervised methods (don't assume a signal model)

Two sources of data are at hand:

- Background (MC simulations) sample - labelled observations

$$\text{Background:} \quad X_1, \ldots, X_{m_b} \sim p_b$$

- Background + possible signal (experimental) sample - unlabelled observations

$$\text{Experimental:} \quad W_1, \ldots, W_n \sim q = (1-\lambda)p_b + \lambda p_s$$

Train a semi-supervised classifier (h) to separate experimental from background.

Note: Here p_b is a simulator for SM background events, p_s is an unspecified signal distribution and the signal strength is λ. We only have access to $X's$ and $W's$; i.e., we have no direct access to p_b, q, p_s or λ.

Signal detection via semi-supervised classifiers

We have:

- Background: $X_1, \ldots, X_{m_b} \sim p_b$
- Experimental: $W_1, \ldots, W_n \sim q = (1-\lambda)p_b + \lambda p_s$
- A semi-supervised classifier (h) that separates $X_1, \ldots, X_{m_b}$ from $W_1, \ldots, W_n$.

We want to test $H_0 : \lambda = 0$ vs $H_1 : \lambda > 0$ or equivalently $H_0 : q = p_b$ vs $q \neq p_b$ (Two-sample testing).

Signal detection via semi-supervised classifiers

We have:

- Background: $X_1, \ldots, X_{m_b} \sim p_b$
- Experimental: $W_1, \ldots, W_n \sim q = (1-\lambda)p_b + \lambda p_s$
- A semi-supervised classifier (h) that separates $X_1, \ldots, X_{m_b}$ from $W_1, \ldots, W_n$.

We want to test $H_0 : \lambda = 0$ vs $H_1 : \lambda > 0$ or equivalently $H_0 : q = p_b$ vs $q \neq p_b$ (Two-sample testing).

Recent approach: use classifiers to perform the test in high-dimensional spaces (e.g., Kim et al. (2019, 2021))
Idea: If the classifier is able to distinguish between the two samples, then there is a difference in the two distributions.

Likelihood Ratio Test statistic

- $X_1, \ldots, X_{m_b} \sim p_b$ and $W_1, \ldots, W_n \sim q = (1-\lambda)p_b + \lambda p_s$.
- Test $H_0 : \lambda = 0$ vs $H_1 : \lambda > 0$.
- Likelihood Ratio of the experimental data W_i's:

$$\frac{\mathcal{L}_q(\lambda)}{\mathcal{L}_q(0)} = \prod_i \psi(W_i), \quad \psi = q/p_b,$$

where $q = (1-\lambda)p_b + \lambda p_s$.

Likelihood Ratio Test statistic

- $X_1, \ldots, X_{m_b} \sim p_b$ and $W_1, \ldots, W_n \sim q = (1-\lambda)p_b + \lambda p_s$.
- Test $H_0 : \lambda = 0$ vs $H_1 : \lambda > 0$.
- Likelihood Ratio of the experimental data W_i's:

$$\frac{\mathcal{L}_q(\lambda)}{\mathcal{L}_q(0)} = \prod_i \psi(W_i), \quad \psi = q/p_b,$$

  where $q = (1-\lambda)p_b + \lambda p_s$.

- Goal: Estimate the ratio ψ using the classifier h instead of estimating q and p_b individually.

Likelihood Ratio Test statistic

- The classifier output (experimental membership probability) h, using Bayes' rule can be written as:

$$h(z) = \frac{n\psi(z)}{n\psi(z) + m_b},$$

where m_b and n are the number of background and experimental events respectively.

Likelihood Ratio Test statistic

- The classifier output (experimental membership probability) h, using Bayes' rule can be written as:

$$h(z) = \frac{n\psi(z)}{n\psi(z) + m_b},$$

where m_b and n are the number of background and experimental events respectively.

- We can estimate $\widehat{\psi}(z) = \frac{m_b h(z)}{n(1-h(z))}$.

Likelihood Ratio Test statistic

- The classifier output (experimental membership probability) h, using Bayes' rule can be written as:

$$h(z) = \frac{n\psi(z)}{n\psi(z) + m_b},$$

where m_b and n are the number of background and experimental events respectively.

- We can estimate $\widehat{\psi}(z) = \frac{m_b h(z)}{n(1-h(z))}$.

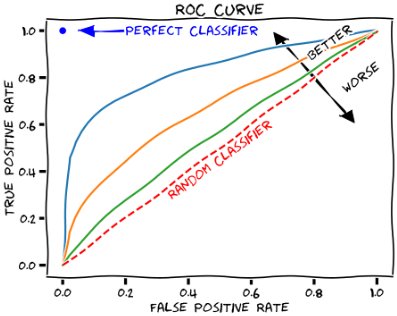
- So, LRT statistic $\text{LRT} = 2\sum_i \log \widehat{\psi}(W_i)$.

Classifier performance based test statistics

- $H_0 : \lambda = 0$ vs $H_1 : \lambda > 0$ is equivalent to $H_0 : q = p_b$ vs $H_1 : q \neq p_b$

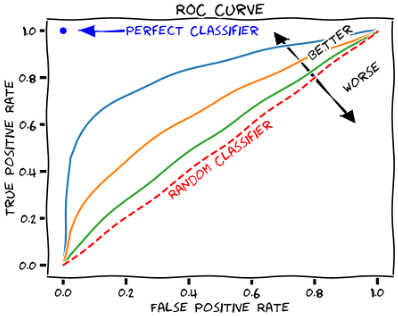
Classifier performance based test statistics

- $H_0 : \lambda = 0$ vs $H_1 : \lambda > 0$ is equivalent to $H_0 : q = p_b$ vs $H_1 : q \neq p_b$



Classifier performance based test statistics

- $H_0 : \lambda = 0$ vs $H_1 : \lambda > 0$ is equivalent to $H_0 : q = p_b$ vs $H_1 : q \neq p_b$



1. Area Under the Curve (AUC) Statistic: $\hat{\theta}$
   Test $H_0 : \theta = 0.5$ vs $H_1 : \theta > 0.5$.

2. Misclassification Error Statistic: $\widehat{MCE}$
   Test $H_0 : MCE = 0.5$ vs $H_1 : MCE < 0.5$.

Calibration of the tests to control Type I error

Under the null both $X's$ and $W's$ are samples from the same distribution p_b. For all the statistics we have different ways of estimating the null distribution:

- Asymptotic
- Nonparametric Bootstrap
- Permutation

Calibration of the tests to control Type I error

Under the null both $X's$ and $W's$ are samples from the same distribution p_b. For all the statistics we have different ways of estimating the null distribution:

- Asymptotic: We can derive and use the asymptotic distribution for each of the test statistics; e.g., for AUC (Newcombe, 2006) under H_0

$$\frac{\hat{\theta} - 0.5}{\sqrt{Var_0(\hat{\theta})}} \rightsquigarrow N(0, 1),$$

  where $Var_0(\hat{\theta})$ can be estimated under H_0.

- Nonparametric Bootstrap: Randomly sample with replacement from the X's and W's combined and randomly label them as either X's or W's.
- Permutation: Randomly permute the class labels of the X's and W's.

Power of detecting a well-specified signal

Power to detect signal in 50 experiments (in percentage) in the Kaggle's Higgs Boson Machine Learning Challenge at $\alpha = 0.05$..

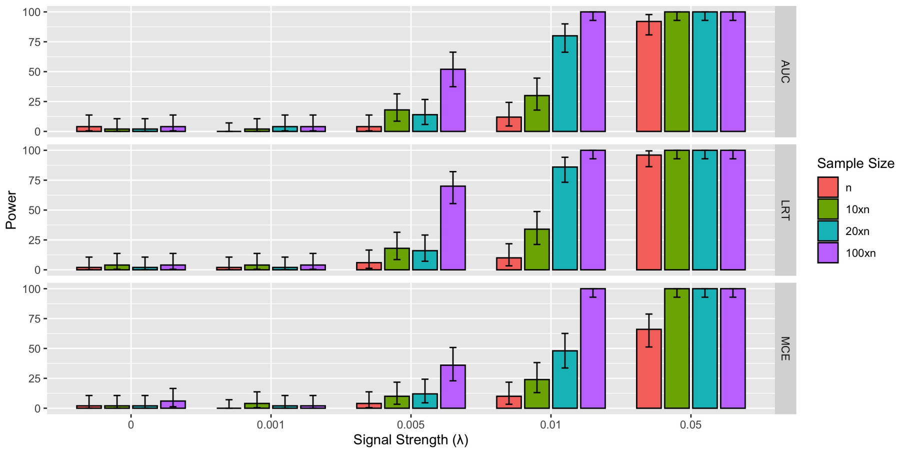
	Model	Method	Signal Strength (λ)					
			0.15	0.1	0.07	0.05	0.01	0
Signal Labels	Supervised LRT	Asymptotic	100	100	96	62	18	6
		Permutation	100	98	98	86	6	0
	Supervised Score	Permutation	94	92	100	92	24	12
NO Signal Labels	Semi-Supervised LRT	Asymptotic	100	98	74	38	6	2
		Permutation	100	98	72	38	6	2
	Semi-Supervised AUC	Asymptotic	100	98	70	32	6	2
		Permutation	100	98	68	32	4	2
		Slow Perm	100	100	94	56	8	4
	Semi-Supervised MCE	Asymptotic	100	96	52	28	6	6
		Slow Perm	100	98	86	58	6	2

Power of detecting a misspecified signal

Power to detect signal in 50 experiments (in percentage) in the Kaggle's Higgs Boson Machine Learning Challenge at $\alpha = 0.05$.

	Model	Method	Signal Strength (λ) 0.15	0.1	0.07	0.05	0.01	0
Signal Labels	Supervised LRT	Asymptotic	2	10	2	8	6	4
		Permutation	0	0	0	0	2	0
	Supervised Score	Permutation	0	0	0	0	2	8
NO Signal Labels	Semi-Supervised LRT	Asymptotic	100	100	100	82	4	4
		Permutation	100	100	100	82	4	2
	Semi-Supervised AUC	Asymptotic	100	100	100	78	8	4
		Permutation	100	100	100	80	8	2
		Slow Perm	100	100	100	100	10	4
	Semi-Supervised MCE	Asymptotic	100	100	100	66	6	4
		Slow Perm	100	100	100	98	8	2

Power with increasing sample size



Power of the asymptotic model-independent tests for increasing sample sizes, where $n = 2 \times 10^4$.

Interpreting the semi-supervised classifier

To understand the signal that the semi-supervised classifier has identified, we need to understand the semi-supervised classifier.

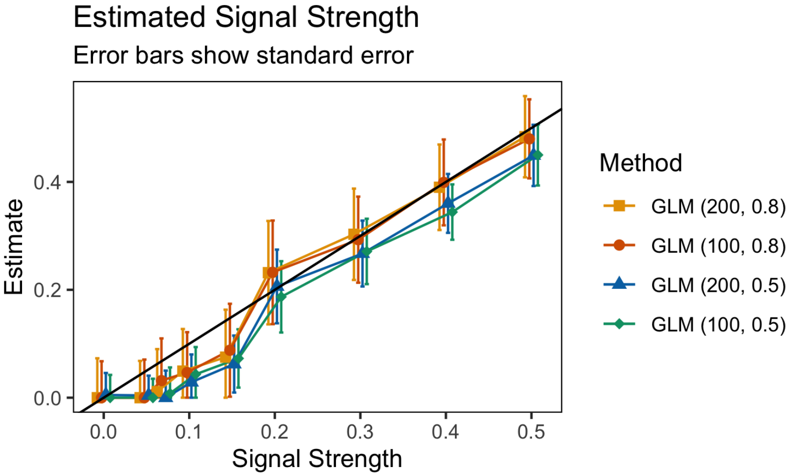
The trouble is that the classifier is trained to separate the experimental from the background and not the signal from the background..

We consider the following:

- Signal Strength Estimation: Estimate the signal strength in the data.
- Active Subspace Methods: Characterize the signal and find subspaces that influence the classifier.

Signal strength (λ) estimation

We estimate the signal strength λ from the classifier using the Neyman–Pearson quantile transform.

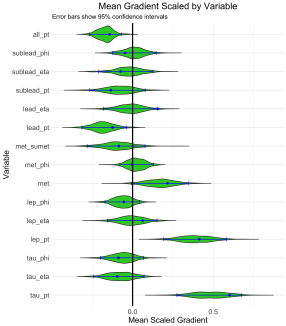


Active subspace of the classifier for $\lambda = 0.15$

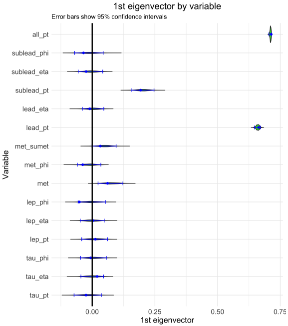
We use the active subspace of the classifier to identify variable combinations that help separate the signal from the background.

The vectors capture the variable dependencies that influence the classifier.

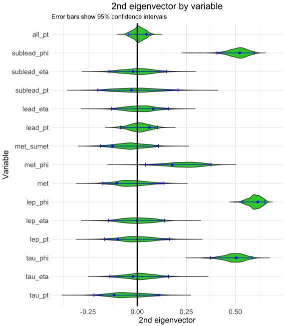
Mean Gradient T-Stat



First Eigenvector



Second Eigenvector



Discussion: Incorporating systematics

So far, I haven't spoken about systematics at all!

The methods proposed here assume that the background samples $X_1, \ldots, X_{m_b}$ come from the "true" background distribution p_b.

But X's are MC simulations which are likely to be systematically misspecified.

Discussion: Incorporating systematics

So far, I haven't spoken about systematics at all!
The methods proposed here assume that the background samples $X_1, \ldots, X_{m_b}$ come from the "true" background distribution p_b.
But X's are MC simulations which are likely to be systematically misspecified.

Important question: Are the "signals" found true signals or differences between the true background and a misspecified background?
Answer: Right now, we don't know!

Discussion: Incorporating systematics

So far, I haven't spoken about systematics at all!
The methods proposed here assume that the background samples $X_1, \ldots, X_{m_b}$ come from the "true" background distribution p_b.
But X's are MC simulations which are likely to be systematically misspecified.

Important question: Are the "signals" found true signals or differences between the true background and a misspecified background?
Answer: Right now, we don't know!

We can still use the methods to:

- Identify and characterize regions of high-dimensional space where the background is mismodelled.
- Perform pilot analysis to guide future model-independent searches.

Discussion: Incorporating systematics

Let $\gamma \in \Gamma$ be the nuisance parameter. Then we want to test:

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

This is an open problem that needs new methodology.

Discussion: Incorporating systematics

Let $\gamma \in \Gamma$ be the nuisance parameter. Then we want to test:

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

This is an open problem that needs new methodology.

D'Agnolo et al. (2021b) makes a significant contribution in incorporating systematics into high-dimensional two-sample testing (Gaia's talk!). See also D'Agnolo and Wulzer (2019); D'Agnolo et al. (2021a).

Discussion: Incorporating systematics

Let $\gamma \in \Gamma$ be the nuisance parameter. Then we want to test:

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

This is an open problem that needs new methodology.

D'Agnolo et al. (2021b) makes a significant contribution in incorporating systematics into high-dimensional two-sample testing (Gaia's talk!). See also D'Agnolo and Wulzer (2019); D'Agnolo et al. (2021a).

We additionally use the AUC and the MCE test statistics and estimate the LRT using a semi-supervised high-dimensional classifier.
Interesting to see how we can incorporate systematics to the tests.

Conclusions

- Model-Independent Detection: Model-independent searches may have more power to find unexpected or misspecified signals.

Conclusions

- Model-Independent Detection: Model-independent searches may have more power to find unexpected or misspecified signals.
- Semi-Supervised Classifier Tests:
  - High-dimensional semi-supervised classifiers that separate experimental data from the background can be used for signal detection.

Conclusions

- Model-Independent Detection: Model-independent searches may have more power to find unexpected or misspecified signals.

- Semi-Supervised Classifier Tests:
  - High-dimensional semi-supervised classifiers that separate experimental data from the background can be used for signal detection.
  - Explored using LRT, AUC and MCE statistics to perform the test - AUC and MCE perform better than LRT.

Conclusions

- Model-Independent Detection: Model-independent searches may have more power to find unexpected or misspecified signals.

- Semi-Supervised Classifier Tests:
  - High-dimensional semi-supervised classifiers that separate experimental data from the background can be used for signal detection.
  - Explored using LRT, AUC and MCE statistics to perform the test - AUC and MCE perform better than LRT.
  - Explored various calibration methods (asymptotic, bootstrap and permutation).

Conclusions

- Model-Independent Detection: Model-independent searches may have more power to find unexpected or misspecified signals.
- Semi-Supervised Classifier Tests:
  - High-dimensional semi-supervised classifiers that separate experimental data from the background can be used for signal detection.
  - Explored using LRT, AUC and MCE statistics to perform the test - AUC and MCE perform better than LRT.
  - Explored various calibration methods (asymptotic, bootstrap and permutation).
- Interpretability:
  - Signal Strength Estimation

Conclusions

- Model-Independent Detection: Model-independent searches may have more power to find unexpected or misspecified signals.
- Semi-Supervised Classifier Tests:
  - High-dimensional semi-supervised classifiers that separate experimental data from the background can be used for signal detection.
  - Explored using LRT, AUC and MCE statistics to perform the test - AUC and MCE perform better than LRT.
  - Explored various calibration methods (asymptotic, bootstrap and permutation).
- Interpretability:
  - Signal Strength Estimation
  - Active Subspace Methods

Conclusions

- Model-Independent Detection: Model-independent searches may have more power to find unexpected or misspecified signals.
- Semi-Supervised Classifier Tests:
  - High-dimensional semi-supervised classifiers that separate experimental data from the background can be used for signal detection.
  - Explored using LRT, AUC and MCE statistics to perform the test - AUC and MCE perform better than LRT.
  - Explored various calibration methods (asymptotic, bootstrap and permutation).
- Interpretability:
  - Signal Strength Estimation
  - Active Subspace Methods
- **Open question:** How to incorporate background systematics?

Thank you!

Model-Independent Detection of New Physics Signals Using Interpretable Semi-Supervised Classifier Tests. (arXiv:2102.07679)

References

Chakravarti, P., Kuusela, M., Lei, J., and Wasserman, L. (2021). Model-independent detection of new physics signals using interpretable semi-supervised classifier tests. *arXiv preprint arXiv:2102.07679*.

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

D'Agnolo, R. T., Grosso, G., Pierini, M., Wulzer, A., and Zanetti, M. (2021b). Learning new physics from an imperfect machine. *arXiv preprint arXiv:2111.13633*.

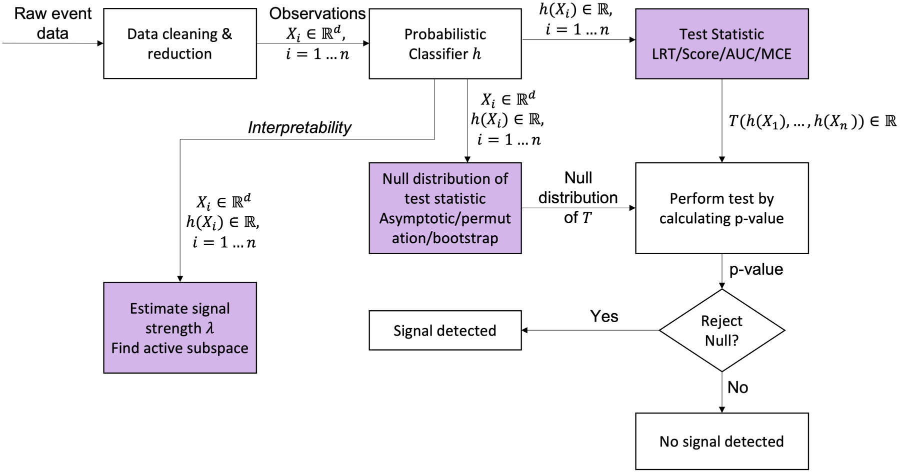
D'Agnolo, R. T. and Wulzer, A. (2019). Learning new physics from a machine. *Physical Review D*, 99(1):015014.

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

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

Newcombe, R. G. (2006). 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.

Flowchart of signal detection procedure



Model-dependent supervised methods test statistics

- Likelihood Ratio on the W_i's for $H_0 : \lambda = 0$ vs $H_1 : 0 < \lambda < 1$:

$$\frac{\mathcal{L}_q(\lambda)}{\mathcal{L}_q(0)} = \prod_i [(1-\lambda) + \lambda\psi(W_i)], \quad \psi = p_s/p_b,$$

where ψ can be estimated using a classifier trained on signal and background MC simulations, p_s and p_b are the signal and background models and λ is the signal strength.

1. Likelihood Ratio Test Statistic:

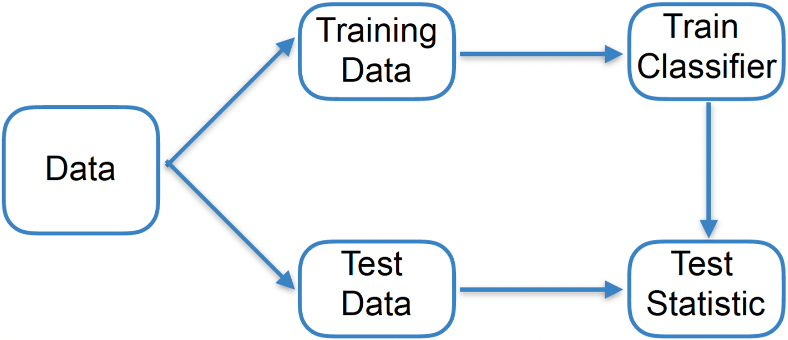
$$\text{LRT} = 2\sum_i \log\left((1-\hat{\lambda}_{\text{MLE}}) + \hat{\lambda}_{\text{MLE}}\hat{\psi}(W_i)\right)$$

2. Score Test Statistic:

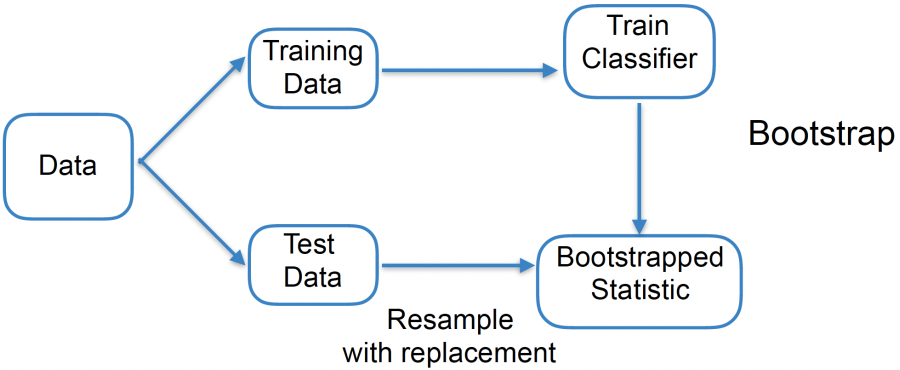
$$S = \frac{1}{N}\sum_{i=1}^{N}\widehat{\psi}(W_i).$$

- Asymptotic method for first, permutation and bootstrap methods for both.

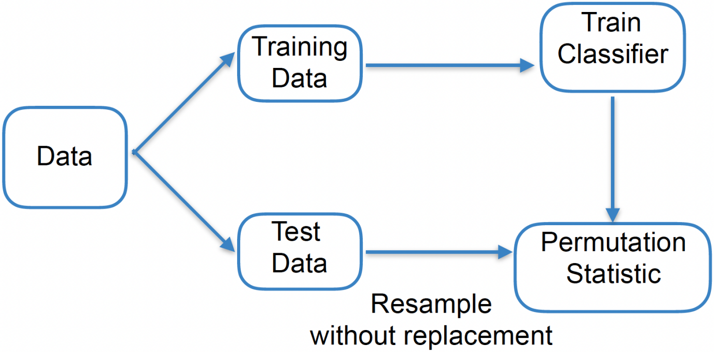
Calibration methods



Calibration methods

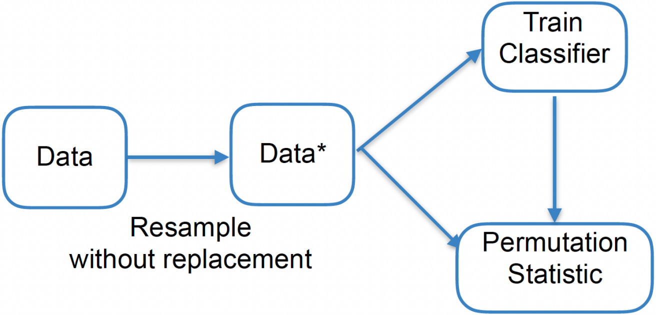


Calibration methods



Permutation

Calibration methods



In-sample
Slow
Permutation

Kaggle's Higgs boson challenge [1]

- Data provided by ATLAS on CERN Open Data Portal.
- 15 variables.
- Transverse momentum and energy as well as angles of resulting particles and jets of particles in a collision event.
- 80,806 background events and 84,221 signal events.
- Create experimental data in 50 simulations with varying signal strength, λ.
- Compare power of the methods in detecting the Higgs boson.

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

Signal strength (λ) estimation

We define a Neyman-Pearson Quantile Transform:

$$\rho(w) = \mathbb{P}_{X \sim p_b}\left(h(X) \geq h(w)\right),$$

where h is the semi-supervised classifier.
If g_q is the density of $\rho(W)$ when $W \sim q$ (the experimental density), then we show that:

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

So we can estimate:

$$\hat{\lambda} = \widehat{g_q}(1).$$

To estimate g_q we first estimate $\rho(\cdot)$ for the experimental data W_i:

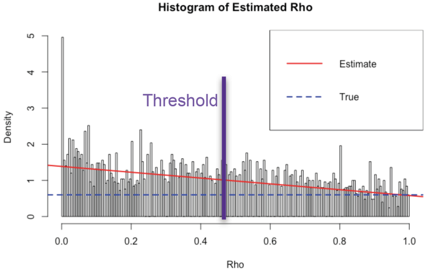
$$\hat{\rho}(W_i) = \frac{1}{m_b} \sum_{j=1}^{m_b} \mathbb{I}\{\tilde{h}(X_j) \geq \tilde{h}(W_i)\}$$

Signal strength (λ) estimation

We define a Neyman-Pearson Quantile Transform:

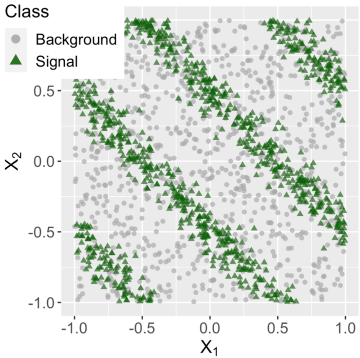
$$\rho(w) = \mathbb{P}_{X\sim p_b}(h(X) \geq h(w)) \to \hat{\rho}(W_i) = \frac{1}{m_b}\sum_{j=1}^{m_b} \mathbb{I}\{\tilde{h}(X_j) \geq \tilde{h}(W_i)\}$$

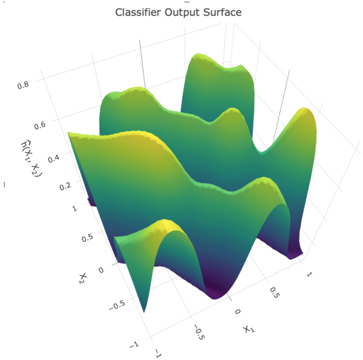
1. If g_q is the density of $\rho(W)$ when $W \sim q$, then $\hat{\lambda} = \widehat{g_q}(1)$.
2. Estimate density of $\hat{\rho}(W_i)$'s using histograms.
3. Fit a Poisson regression model above threshold T to estimate $\widehat{g_q}(1)$.



Identifying the active subspace that explains the classifier $\tilde{h}$

2D toy example.

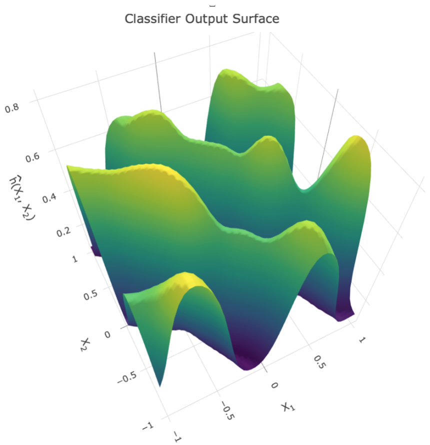




Identifying the active subspace that explains the classifier $\tilde{h}$

1. Consider the gradients of the classifier surface:

$$\frac{\nabla_z h(z)}{\sqrt{Var(\nabla_z h(z))}}$$

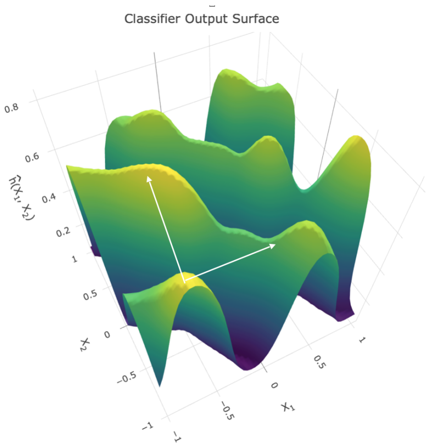


Identifying the active subspace that explains the classifier $\tilde{h}$

1. Consider the gradients of the classifier surface:

$$\frac{\nabla_z h(z)}{\sqrt{Var(\nabla_z h(z))}}$$

2. The gradients explains changes in the classifier surface.



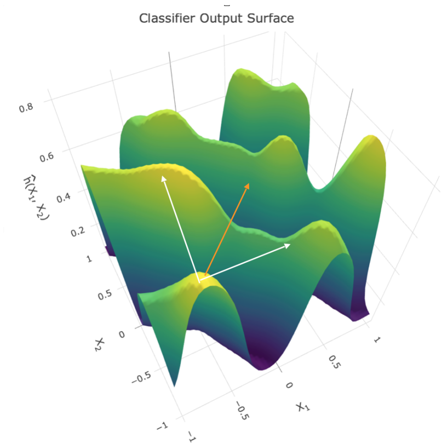
Identifying the active subspace that explains the classifier $\tilde{h}$

1. Consider the gradients of the classifier surface:

$$\frac{\nabla_z h(z)}{\sqrt{Var(\nabla_z h(z))}}$$

2. The gradients explains changes in the classifier surface.

3. Perform PCA on gradients resulting in directions in which the gradient varies the most.



Identifying the active subspace that explains the classifier $\tilde{h}$

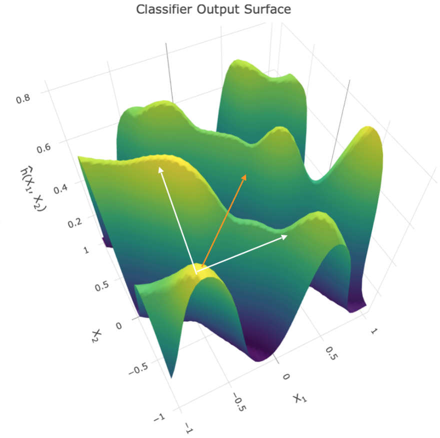
1. Consider the gradients of the classifier surface:

$$\frac{\nabla_z h(z)}{\sqrt{Var(\nabla_z h(z))}}$$

2. The gradients explains changes in the classifier surface.

3. Perform PCA on gradients resulting in directions in which the gradient varies the most.

4. Mean of the gradients gives direction of change.



Active Subspace of $h(\cdot)$

For experimental data $W_1, \ldots, W_N$,

- $\frac{\nabla_{\mathbf{z}} h(\mathbf{z})}{\sqrt{Var(\nabla_{\mathbf{z}} h)}}$ - $T_j = \frac{\widehat{\nabla_{\mathbf{z}} h(W_j)}}{\sqrt{\widehat{Var}(\nabla_{\mathbf{z}} h(W_j))}}$ using a local linear smoother on h.

Active Subspace of $h(\cdot)$

For experimental data $W_1, \ldots, W_N$,

- $\frac{\nabla_{\mathbf{z}} h(\mathbf{z})}{\sqrt{Var(\nabla_{\mathbf{z}} h)}}$ - $T_j = \frac{\widehat{\nabla_{\mathbf{z}} h(W_j)}}{\sqrt{\widehat{Var(\nabla_{\mathbf{z}} h(W_j))}}}$ using a local linear smoother on h.
- Perform Principal Component Analysis (PCA) or sparse PCA on $H = (T_1, T_2, \ldots, T_N)^T$.

Active Subspace of $h(\cdot)$

For experimental data $W_1, \ldots, W_N$,

- $\frac{\nabla_{\mathbf{z}} h(\mathbf{z})}{\sqrt{Var(\nabla_{\mathbf{z}} h)}}$ - $T_j = \frac{\widehat{\nabla_{\mathbf{z}} h(W_j)}}{\sqrt{\widehat{Var(\nabla_{\mathbf{z}} h(W_j))}}}$ using a local linear smoother on h.
- Perform Principal Component Analysis (PCA) or sparse PCA on $H = (T_1, T_2, \ldots, T_N)^T$.
- Let $\mathbf{m}_1, \mathbf{m}_2, \ldots$ be the leading eigenvectors - $\hat{\mathbf{m}}_1, \hat{\mathbf{m}}_2, \ldots$.

Active Subspace of $h(\cdot)$

For experimental data $W_1, \dots, W_N$,

- $\frac{\nabla_{\mathbf{z}} h(\mathbf{z})}{\sqrt{Var(\nabla_{\mathbf{z}} h)}}$ - $T_j = \frac{\widehat{\nabla_{\mathbf{z}} h(W_j)}}{\sqrt{\widehat{Var(\nabla_{\mathbf{z}} h(W_j))}}}$ using a local linear smoother on h.
- Perform Principal Component Analysis (PCA) or sparse PCA on $H = (T_1, T_2, \dots, T_N)^T$.
- Let $\mathbf{m}_1, \mathbf{m}_2, \dots$ be the leading eigenvectors - $\hat{\mathbf{m}}_1, \hat{\mathbf{m}}_2, \dots$.
- $\mathbb{E}\left[\frac{\nabla_{\mathbf{z}} h(\mathbf{z})}{\sqrt{Var(\nabla_{\mathbf{z}} h)}}\right], \mathbf{m}_1, \mathbf{m}_2$ capture the changes in the classifier surface - $\overline{T} = \frac{1}{N}\sum_{j=1}^{N} T_j,\ \hat{\mathbf{m}}_1, \hat{\mathbf{m}}_2$.