# Identifying fermionic decay signals of the Higgs boson with classification algorithms

Mansell, Georgie Stephenson, Ant

January 19, 2021

**Abstract**

...

## 1 Introduction

After the theoretical discovery of the Higgs mechanism and the associated particle, the Higgs boson, experimental confirmation of the prediction became a much sought-after prize in High Energy Physics. Although the theory was more or less accepted by the theoretical community since its inception in the 1960s, experimental evidence was required for verification.

Once CERN's Large Hadron Collider (LHC) emerged on the scene, with the necessary power to probe higher energy regimes of collisions, vast amounts of data started to be collected. In order to verify the theoretical predictions of particle physics, this data needed to be anaylsed carefully to try and distinguish signal of as-yet unknown particle decays vs effectively "proven" existing decays. The accepted threshold for a discovery in particle physics is the "gold-standard" 5-sigma rule. As a result, any announcement of a new experimental discovery (such as the Higgs boson) was required to meet this threshold.

For the Higgs boson in particular, various decay mechanisms were proposed that could be used to demonstrate its existence by comparing the rate of the by-products of those decays vs background mechanisms (that do not involve the Higgs). In 2013, when the Higgs was experimentally discovered by CERN, the evidence was provided by *bosonic* decays from the Higgs to the following pairs, $\gamma\gamma$, $WW$ and $ZZ$.

The goal here is to instead examine the coupling of the Higgs to fermions, to verify that their mass can likewise be explained by the Higgs mechanism. Specifically, the aim is to analyse the decay $H \to \tau\tau$; i.e. the decay of the Higgs to a pair of tau-flavour leptons. (The other candidates, electrons and muons fall outside the energy range of the LHC due to their lighter mass).

In this document we look at the problem of improving the statistical significance of the experimental results collected by implementing classification models to identify signal events. We organise the report by providing a brief overview of the dataset and some of its key properties in 2. After this we introduce our proposed methods for studying the problem and generating classification predictions in 3. We go onto analyse the results we obtained from running experiments on the data with our models in order to try and pick the best performing model from our

subset of trials, being careful to fairly assess this from two key metrics (4). Finally we summarise our key findings and overall score on a hold-out test set of data in 5.

# 2 Data

## 2.1 Structure

We have access to a dataset comprising roughly of primitive covariates, derived covariates, target labels and auxiliary data (weights, subset labels).

## 2.2 Class Imbalance

Since the events we are looking for in the data are rare, if we were to naïvely include rows of signal and background events at the actually observed signal:noise ratio (of approximately 1:1000) we would have a highly imbalanced dataset. As a result, we need to adopt a strategy to address this by either modifying our dataset such that standard classification algorithms can still be of use, or modifying the algorithms to take this into account. Fortunately, the dataset includes a column of weights which account for the imbalance. In particular, the ratio of signal:background rows in the training data is approximately 1:1.9 with associated weights of $O(10^{-3})$ and $O(1)$ for the signal and background rows respectively.

## 2.3 Missing Data

Early examinations of the dataset revealed the presence of a significant quantity of missing data. Elements of the data matrix recorded as missing were labelled by a value of -999. Before embarking on any model building or even feature engineering we analysed the missing values to try and ascertain whether there was any pattern to their locations. From the definitions of covariates in the dataset it was possible to infer potential causes of missing data. In particular, the estimated mass of the Higgs ($DER\_mass\_MMC$) "(may be undefined if the topology of the event is too far from the expected topology)", indicates that this feature may be expected to be labelled as missing. Similarly, a set of the features mention a dependence on the number of jets measured in the interaction:

$DER\_deltaeta\_jet\_jet$ undefined if $PRI\_jet\_num \leq 1$

$DER\_mass\_jet\_jet$ undefined if $PRI\_jet\_num \leq 1$

$DER\_prodeta\_jet\_jet$ undefined if $PRI\_jet\_num \leq 1$

$PRI\_jet\_subleading\_pt$ undefined if $PRI\_jet\_num \leq 1$

$PRI\_jet\_subleading\_phi$ undefined if $PRI\_jet\_num \leq 1$

$PRI\_jet\_subleading\_eta$ undefined if $PRI\_jet\_num \leq 1$

$DER\_lep\_eta\_entrality$ undefined if $PRI\_jet\_num \leq 1$

$PRI\_jet\_leading\_pt$ undefined if $PRI\_jet\_num = 0$

$PRI\_jet\_leading\_eta$ undefined if $PRI\_jet\_num = 0$

$PRI\_jet\_leading\_phi$ undefined if $PRI\_jet\_num = 0$

So we can see that the missing data is explained by the combination of an unexpected topology and the number of jets observed. This implies that the physical data generating process is different in each of these regimes and that we ought to treat them differently in accordance. As such we chose to attempt to build a separate model for each regime.

# 3 Method

## 3.1 Performance Metrics

### 3.1.1 AMS

The AMS metric is the Approximate Median discovery Significance; an approximation of the *significance* defined by

$$\begin{aligned} Z &= \Phi^{-1}(1-p) \\ &= \sqrt{q_0} \\ &= \sqrt{2\left(n \ln \frac{n}{\mu_b} - n + \mu_b\right)} \end{aligned}$$

where $n$ is the (unknown) number of events in some search region $\mathcal{G}$, $\Phi^{-1}$ is the inverse Normal CDF, $q_0$ is a test statistic given by Wilks' Theorem and $\mu_b$ is the (unknown) expected number of background events. We replace $n \to s + b$ and $\mu_b \to b$ with $s, b$ the estimator of the expected number of signal and background events respectively.

See [1] for more details.

Since this is the formal objective we aim to maximise, it might make sense to try and optimise it directly. A little analysis however reveals that the function is non-convex and therefore a poor choice. See A for the full calculation.

### 3.1.2 ROC and AUC

A ROC (Receiver Operating Characteristic) curve plots the true positive rate against the false positive rate for a binary classifier with a varying decision threshold. By doing this we can show the ability of the classifier to dscriminate between the two classes and attain a metric for the performance that we can use to compare models. The worst possible model is given by a line at $y = x$ and any curve above this line is an improvement.

The AUC (Area Under the [ROC] Curve) provides a compression of the assessment the ROC curve provides by integrating the ROC curve and giving a value in $[0, 1]$ that represents an overall view of the discriminatory power of the model. A value of 0.5 corresponds to the worst model ($y = x$) whilst 1.0 would represent a perfect model.

Note that for our problem, the actual ratio of signal:background events is very small, which would cause a problem in interpreting the AUC reliably. In the actual dataset, this problem is mitigated by providing comparable numbers of rows with signal labels as to rows with background labels, each with an associated weight, such that the overall weighted events of signal and background match the actual observed data.

## 3.2 Logistic Regression

We chose to use a logistic regression as a baseline model to try and solve our classification task. The logic for this was two-fold; we would have a robust, flexible model to compare other modelling approaches to, as well as retaining the capacity to augment it with additional features which could include other models as sub-models within the logistic regression model.

## 3.3 SVM

## 3.4 Feature Engineering

In order to improve the performance of our model, we attempted to carry out some feature engineering to extract as much information as possible from the dataset.

### 3.4.1 Redundancy

By considering the basic physics of the beam it is possible to see that there is some redundancy amongst the (primitive) features in the dataset. We can exploit this fact and reduce our initial feature space slightly, by transforming the redundant features into a set of new derived features that contain the same information. More specifically, the redundancy comes from the consideration that the physical phenomena should be invariant to certain symmetries; a rotation about the beam ($z$) axis and reflections in the $xy$-plane.

In particular, we defined the following new features:

$$
\begin{aligned}
PRI\_lep\_phi - PRI\_tau\_phi &= (PRI\_lep\_phi - PRI\_tau\_phi) \mod 2\pi \\
PRI\_met\_phi - PRI\_tau\_phi &= (PRI\_met\_phi - PRI\_tau\_phi) \mod 2\pi \\
PRI\_jet\_leading\_phi - PRI\_tau\_phi &= (PRI\_jet\_leading\_phi - PRI\_tau\_phi) \mod 2\pi \\
PRI\_jet\_subleading\_phi - PRI\_tau\_phi &= (PRI\_jet\_subleading\_phi - PRI\_tau\_phi) \mod 2\pi \\
PRI\_tau\_eta &= \text{sign(PRI\_tau\_eta)PRI\_tau\_eta} \\
PRI\_lep\_eta &= \text{sign(PRI\_tau\_eta)PRI\_lep\_eta} \\
PRI\_jet\_leading\_eta &= \text{sign(PRI\_tau\_eta)PRI\_jet\_leading\_eta} \\
PRI\_jet\_subleading\_eta &= \text{sign(PRI\_tau\_eta)PRI\_jet\_subleading\_eta}
\end{aligned}
$$

### 3.4.2 Higher-Order Effects

From the exploratory analysis of the data, in particular the visualisation of the principal components, it appears that the classes are not linearly separable in the base feature space. As a result, we hope that perhaps by including non-linear transformations of our features, such as polynomials and RBF centroids, we might

be able to model some of the non-linear elements of the relationship between the classes and the features.

**RBF Centroids** To attempt to capture generic higher order interaction-type effects, we implemented a set of RBF centroid features. The implementation works as follows

**Algorithm 1:** Augment covariate matrix with RBF centroid features

1. Generate heuristic estimate of RBF hyperparameter $s$.
2. Select $n_c$ points from $\mathbf{X}_{train}$, $\mathbf{X}_c = \{\mathbf{x}_i^{train}, \ldots, \mathbf{x}_{n_c}^{train}\}$.
3. **for** *i=1:*$n_c$ **do**
   - Calculate $\mathbf{k}_{RBF}(\mathbf{x}_i^{train}, \mathbf{X}_{train}; s)$
   - $\mathbf{X}_{train} \leftarrow \mathbf{X}_{train} \oplus \mathbf{k}_{RBF}(\mathbf{x}_i^{train}, \mathbf{X}_{train}; s)$
4. Fit model.
5. **for** *i=1:*$n_c$ **do**
   - Calculate $\mathbf{k}_{RBF}(\mathbf{x}_i^{train}, \mathbf{X}_{test}; s)$
   - $\mathbf{X}_{test} \leftarrow \mathbf{X}_{test} \oplus \mathbf{k}_{RBF}(\mathbf{x}_i^{train}, \mathbf{X}_{test}; s)$

#### 3.4.3 Polynomial Transformations

In order to try and model non-linear relationships between our features and the target labels, we decided to try and include polynomial transformations of the features.

#### 3.4.4 Interactions

Although we considered implementing interactions to further augment our covariates, we chose not to pursue this avenue in the end. The reason for this is that the number of combinations of pairwise interactions for our feature set is .... Including all of these is likely to significantly increase the risk of overfitting our model, and without a principled method to either choose to only include a subset of interactions, or a way to remove most of them, it seems preferable to skip this option. Additionally, the large increase in feature space would also lead to a sizeable increase in computation time, as the Hessian dimensions would increase by ... and hence the time by approximately ... (as our algorithm uses Newton's method which scales as $O(d^3)$ for a $d$-dimensional feature space).

## 4 Results

### 4.1 Model Selection

To make decisions regarding the relative performance of our candidate models we used a 10-fold cross-validation procedure to train and test each of the permutations of our features and models. By running experiments that carried out the procedure, we were able to generate summary outputs that concisely captured a representation of performance. More precisely, we appended rows of information to a *.csv* file for each experiment that could then be subsequently analysed to compare performance. Note that each experiment was run with an unspecified random seed, so we

Table 1: Results table of the top 5 experiments

| output | $n_{rbf}$ | lambda | poly | auc | $mad_{auc}$ | ams | $mad_{ams}$ | $scaled_{auc}$ | $scaled_{ams}$ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 0.000 | 3 | 0.81 | 0.063 | 1.907 | 0.855 | 12.886 | 2.230 |
| 2 | 1 | 2.154 | 3 | 0.81 | 0.061 | 1.898 | 0.856 | 13.180 | 2.217 |
| 3 | 2 | 2.154 | 3 | 0.81 | 0.061 | 1.898 | 0.856 | 13.180 | 2.217 |
| 4 | 3 | 2.154 | 3 | 0.81 | 0.061 | 1.898 | 0.856 | 13.180 | 2.217 |
| 5 | 4 | 2.154 | 3 | 0.81 | 0.061 | 1.898 | 0.856 | 13.180 | 2.217 |

expect some degree of random variation between experiments even with no change to parameters, simply due to the change in data partitioning. We can use this scale of variance to aid identification of significant performance improvements. 10 fold cross-validation was picked as a compromise between speed of computation and in-sample vs out-of-sample variance; computation time scales as $O(KNd)$ for $K$ folds, $N$ data points and $d$ features, whilst we expect the variance of our (averaged over folds) performance estimate to scale as $\frac{1}{K}$. In effect, by picking (for example) 10 folds rather than 5, we are trading an increase in computation time of a factor of two for a decrease in estimator variance of the same factor.

### 4.2 Predictions

To actually calculate AMS scores, we need to convert out probabilistic outputs into binary labels which means we need to pick a decision rule, i.e. a threshold over which we assign a label 1 (or $s$) vs 0 (or $b$).

## 5 Conclusion

# Appendices

## A AMS

Is the AMS metric convex?

$$\text{AMS} = \sqrt{2\left((s+b)\log(1+\frac{s}{b}) - s\right)}$$

In the documentation, the implication is that in general, we expect $b \gg s$ giving the approximate AMS as AMS $\sim \frac{\mathrm{s}}{\sqrt{\mathrm{b}}}(1+\mathcal{O}(\frac{\mathrm{s}}{\mathrm{b}}))$.

If we directly check the derivatives of the approximate AMS in the large $b$ regime:

$$\begin{aligned}
\partial_s^2 \frac{s}{\sqrt{b}} &= 0 \\
\partial_b^2 \frac{s}{\sqrt{b}} &= \frac{3}{4}\frac{s}{b^{5/2}} > 0 \\
\partial_{bs}^2 &= -\frac{1}{2b^{3/2}} < 0
\end{aligned}$$

which implies non-convexity (and non-concavity).

To check more carefully though, calculate the terms of the Hessian of the original and then apply the approximation:

$$\begin{aligned}
\partial_s \text{AMS} &= \text{AMS}^{-1}\log\left(1+\frac{\mathrm{s}}{\mathrm{b}}\right) \\
\partial_b \text{AMS} &= \text{AMS}^{-1}\left(\log(1+\frac{\mathrm{s}}{\mathrm{b}}) - \frac{\mathrm{s}}{\mathrm{b}}\right) \\
\partial_s^2 \text{AMS} &= \text{AMS}^{-1}\left(\frac{1}{\mathrm{b}+\mathrm{s}} - \text{AMS}^{-2}\log(1+\frac{\mathrm{s}}{\mathrm{b}})^2\right) \\
\partial_b^2 \text{AMS} &= \text{AMS}^{-1}\left(\frac{\mathrm{s}^2}{\mathrm{b}^2(\mathrm{s}+\mathrm{b})} - \text{AMS}^{-2}(\log(1+\frac{\mathrm{s}}{\mathrm{b}}) - \frac{\mathrm{s}}{\mathrm{b}})^2\right) \\
\partial_{sb}^2 \text{AMS} &= -\text{AMS}^{-1}\left(\frac{\mathrm{s}}{\mathrm{b}(\mathrm{s}+\mathrm{b})} - \text{AMS}^{-2}\log(1+\frac{\mathrm{s}}{\mathrm{b}})(\log(1+\frac{\mathrm{s}}{\mathrm{b}}) - \frac{\mathrm{s}}{\mathrm{b}})\right)
\end{aligned}$$

and then verify whether any of them are ever $< 0$.

Using the AMS approximation, and $\log\left(1+\frac{s}{b}\right) \sim \frac{s}{b} - \frac{1}{2}\left(\frac{s}{b}\right)^2$ we get the following

$$\begin{aligned}
\partial_s^2 \text{AMS} &\sim \frac{\sqrt{b}}{s}\left[\frac{1}{(b+s)} - \frac{b}{s^2}(\frac{s^2}{b^2} - \frac{s^3}{b^3})\right] \\
&= \frac{\sqrt{b}}{s(s+b)} - \frac{1}{s\sqrt{b}} + \frac{1}{b^{3/2}} \\
&\gtrsim \epsilon
\end{aligned}$$

Since in this regime of large $b$ this term is negative, the Hessian cannot be positive definite and hence the metric is non-convex. If the Hessian is *concave* though, we can simply optimize $-$AMS, so we need to check the other terms.

$$\begin{aligned}\partial_b^2 \text{AMS} &\sim \frac{\sqrt{b}}{s}\left[\frac{s^2}{b^2(s+b)} - \frac{b}{s^2}(\frac{1}{4}\frac{s^4}{b^4})\right] \\ &= \frac{s}{b^{3/2}(s+b)} - \frac{1}{4}\frac{s}{b^{5/2}} \\ &\gtrsim \epsilon\end{aligned}$$

$$\begin{aligned}\partial_{bs}^2 \text{AMS} &\sim -\frac{\sqrt{b}}{s}\left[\frac{s}{b(s+b)} + \frac{b}{s^2}(\frac{s}{b} - \frac{1}{2}\frac{s}{b^2})(-\frac{1}{2}\frac{s^2}{b^2})\right] \\ &= \frac{1}{2b^{3/2}}(1 - \frac{1}{2}s) - \frac{1}{\sqrt{b}(s+b)} \\ &< 0\end{aligned}$$

where we take $\epsilon$ to be some suitably small positive number.

If we decide to take some tolerance, $\epsilon \sim 10^{-6}$ at which to ignore higher order terms in $\frac{s}{b}$ we can check specifically that the claims on the Hessian above hold in that regime.

So, let's assume we can ignore terms of $O(\frac{s}{b})^3$ then for $\epsilon \sim 10^{-6}$ a ratio of $\frac{s}{b} \sim 10^{-2}$ would suffice; so to test this, let us choose $s = 1$ and $b = 100$ for simplicity. With this, we find that $\partial_s^2 \text{AMS} \sim +10^{-6}$, $\partial_b^2 \text{AMS} \sim +10^{-6}$ and $\partial_{bs}^2 \text{AMS} \sim -10^{-4}$ which satisfies the claims above. Finally, to verify that this is a reasonable assertion, if we check the ratio of $s : b$ in the data (using the weight vector) we find a ratio of approximately $10^{-3}$, which means our assumption was conservative.

# B Modelling

## B.1 Newton's Method for Logistic Regression

From the conditional distribution under the logistic regression model for a single point $\mathbf{x} \in R^d$, $y \in \{+1, -1\}$ and coefficients $\boldsymbol{\beta} \in R^d$.

$$\begin{aligned}p(y \mid \mathbf{x}, \boldsymbol{\beta}) &= \sigma(f(\mathbf{x}; \boldsymbol{\beta}) \cdot y) \\ &= (1 + \exp(y\langle \mathbf{x}, \boldsymbol{\beta}\rangle))^{-1}\end{aligned}$$

We want to minimise the negative loglikelihood:

$$\begin{aligned}l(\mathbf{x}, y, \boldsymbol{\beta}) &= \log(1 + \mathrm{e}^{y\langle \mathbf{x}, \mathbf{x}\rangle}) \\ \partial_{\beta_a} l &= \frac{y x_a \mathrm{e}^{y\langle \mathbf{x}, \boldsymbol{\beta}\rangle}}{1 + \mathrm{e}^{y\langle \mathbf{x}, \boldsymbol{\beta}\rangle}} \\ \partial_{\beta_a \beta_b}^2 l &= \frac{y^2 x_a x_b \mathrm{e}^{y\langle \mathbf{x}, \boldsymbol{\beta}\rangle}}{(1 + \mathrm{e}^{y\langle \mathbf{x}, \boldsymbol{\beta}\rangle})^2}\left(1 + \mathrm{e}^{y\langle \mathbf{x}, \boldsymbol{\beta}\rangle} - \mathrm{e}^{y\langle \mathbf{x}, \boldsymbol{\beta}\rangle}\right) \\ &= \frac{x_a x_b \mathrm{e}^{y\langle \mathbf{x}, \boldsymbol{\beta}\rangle}}{(1 + \mathrm{e}^{y\langle \mathbf{x}, \boldsymbol{\beta}\rangle})^2} \\ &= x_a x_b w(x)\end{aligned}$$

where in the last line we define a weight function $w(x) = \frac{\mathrm{e}^{-x}}{(1+\mathrm{e}^{-x})^2}$ (and the elementwise vector version, $\mathbf{w}(\mathbf{x}) = \frac{\mathrm{e}^{-\mathbf{x}}}{(1+\mathrm{e}^{-\mathbf{x}})^2}$).

To implement Newton's method we want to try and simplify the expressions. First define the logistic and logit functions as

$$\gamma(x) = (1 + \mathrm{e}^{-x})^{-1}$$
$$\gamma^{-1}(p) = \log\left(\frac{p}{1-p}\right)$$

and then write out the derivatives in terms of the logistic function:

$$\partial_{\beta_a} l(y = +1) = x_a \gamma(\langle \mathbf{x}, \boldsymbol{\beta} \rangle)$$
$$\partial_{\beta_a} l(y = -1) = -x_a (1 - \gamma(\langle \mathbf{x}, \boldsymbol{\beta} \rangle)$$

If we then map $y \in \{1, -1\}$ to $y' \in \{0, 1\}$ we can condense these into a single expression

$$\partial_{\beta_a} l(y) = -x_a (y - \gamma(\langle \mathbf{x}, \boldsymbol{\beta} \rangle))$$

and rewrite the loss as $l(y') = \log(1 + \mathrm{e}^{(-1)^{y'} \langle \mathbf{x}, \boldsymbol{\beta} \rangle})$.

Now for our entire dataset $D$, assumed to be IID, we have a total loss, gradient and Hessian

$$\begin{aligned} L &= \sum_{i \in D} l_i = \sum_{i \in D} \log(1 + \mathrm{e}^{y_i \langle \mathbf{x}_i, \boldsymbol{\beta} \rangle}) \\ \partial_{\boldsymbol{\beta}} L &= -\sum_{i \in D} \mathbf{x}_i (y_i - \gamma(\langle \mathbf{x}_i, \boldsymbol{\beta} \rangle)) \\ &= -\mathbf{X}^T (\mathbf{y} - \gamma(\mathbf{X}\boldsymbol{\beta})) \\ \boldsymbol{H} &= \sum_{i \in D} \mathbf{x}_i \mathbf{x}_i^T w(\mathbf{x}_i) \\ &= \mathbf{X}^T \boldsymbol{W} \mathbf{X} \end{aligned}$$

with $\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_n^T \end{pmatrix} \in R^{n \times d}$, $\boldsymbol{W} = \mathrm{diag}(\mathbf{w}(\mathbf{x})) \in R^{n \times n}$.

### B.1.1 Regularisation

We can add constraints to our minimisation problem, to help constrain the coefficients and effectively implement $L_2$ regularisation by formulating the following convex optimisation problem.

$$\begin{aligned} \min. \quad & L \\ \text{subject to} \quad & \|\boldsymbol{\beta}\|^2 \leq c \end{aligned}$$

From this we can construct the Lagrangian $\mathcal{L} = L + \lambda(\|\boldsymbol{\beta}\|^2 - c)$ and hence augment the gradient and Hessian we calculated above as

$$\partial_{\boldsymbol{\beta}} \mathcal{L} = \partial_{\boldsymbol{\beta}} L + 2\lambda\boldsymbol{\beta}$$
$$\tilde{\boldsymbol{H}} = \boldsymbol{H} + 2\lambda \mathbf{I}$$

Since we have a free parameter $c$ that determines the optimal value $\lambda^\star(c)$ of our dual variable, we can equivalently ignore $c$ and treat $\lambda$ as a parameter to be tuned.

#### B.1.2 Newton's method

**Algorithm 2:** Newton's method

1. Initialise parameters; $\boldsymbol{\beta}_0$, $\lambda_0$, $\epsilon$, $L_0 = L(\boldsymbol{\beta}_0)$, $m$
2. **for** *i=1:m* **do**
   - Compute $\partial_{\boldsymbol{\beta}} L$ and $\boldsymbol{H}$
   - Calculate Newton step $\Delta\boldsymbol{\beta} = -\boldsymbol{H}^{-1}\partial_{\boldsymbol{\beta}} L$
   - Compute new loss $L$
   - Test stopping criterion: $|L - L_0| < \epsilon$
3. Return $\hat{\boldsymbol{\beta}}$.

## References


[1] Claire Adam-Bourdarios, Glen Cowan, Cécile Germain, Isabelle Guyon, Balàzs Kégl, and David Rousseau. The Higgs boson machine learning challenge. In Glen Cowan, Cécile Germain, Isabelle Guyon, Balázs Kégl, and David Rousseau, editors, *Proceedings of the NIPS 2014 Workshop on High-energy Physics and Machine Learning*, volume 42 of *Proceedings of Machine Learning Research*, pages 19–55, Montreal, Canada, 13 Dec 2015. PMLR.