

# Approximating and decomposing likelihood ratios on mixture models using machine learning

K Cranmer<sup>1</sup>, J Pavez<sup>2</sup>, G Louppe<sup>1</sup> and W Brooks<sup>3</sup>

<sup>1</sup> Physics Department, New York University, New York, NY 10003, U.S.A.

<sup>2</sup> Informatics Department, Universidad Técnica Federico Santa María, 1240 Av. España, Valparaíso, Chile

<sup>3</sup> Physics Department, Universidad Técnica Federico Santa María, 1240 Av. España, Valparaíso, Chile

E-mail: [juan.pavezs@alumnos.usm.cl](mailto:juan.pavezs@alumnos.usm.cl)

**Abstract.** Likelihood ratio test are a key tool in many fields of science. In order to evaluate the likelihood ratio the likelihood function is needed. However, it is common in fields like high energy physics to have complex simulations that describe the distribution while not having a description of the likelihood that can be directly evaluated. In this setting it is impossible or computationally expensive to evaluate the likelihood. It is possible to construct an equivalent version of the likelihood ratio that can be easily evaluated by using discriminative classifiers. We show how this can be used to approximate the likelihood ratio when the underlying distribution is a weighted sum of probability distributions (e.g. signal plus background model). We demonstrate how the results can be considerably improved by decomposing the test and use a set of classifiers in a pairwise manner on the components of the mixture model and in which way this can be used to estimate the unknown coefficients of the model, such as the signal contribution.

## 1. Introduction

In High Energy Physics (HEP) and many other fields hypothesis testing is a key tool when reporting results from an experiment. Likelihood ratio tests are the main technique for hypothesis testing and they are the most powerful statistic for simple hypothesis testing. For composite hypothesis testing the profile or generalized likelihood ratio test is commonly used. When computing the likelihood ratio the data distribution  $p(x|\theta)$  must be evaluated, where  $\theta$  are parameters of the probability distribution. However, it is common in HEP to have physics simulations that allow to sample high dimensional vectors from the distribution  $p(x|\theta)$  while not having a description that can be directly evaluated. Commonly it is impossible or computationally expensive to evaluate the likelihood ratio in this setting.

A common use of likelihood ratios in HEP is signal process identification. In this task the hypothesis testing procedure is used to evaluate the signal process significance by contrasting the only background (null) hypothesis versus the signal plus background (alternative) hypothesis. In this setting, the underlying distribution can be seen as a signal and background mixture model defined as

$$p(x|\mu, \nu) = \mu p_s(x|\nu) + (1 - \mu) p_b(x|\nu), \quad (1)$$

where  $p_x(x|\nu)$  correspond to the signal distribution and  $p_b(x|\nu)$  is the background distribution, both parametrized by nuisance parameters  $\nu$  which describe uncertainties in the underlying

physics predictions or response of measurement devices. The parameter  $\mu$  is the mixture coefficient corresponding to the signal component of the distribution. In this case the generalized likelihood ratio test takes the form of

$$\lambda(D) = \prod_{e=1}^n \frac{p(x_e|\mu=0, \hat{\nu})}{p(x_e|\hat{\mu}, \hat{\nu})}, \quad (2)$$

where  $D$  is a data set of i.i.d observations  $x_e$ ,  $\hat{\nu}$  is the conditional maximum likelihood estimator for  $\nu$  under the null hypothesis  $\theta_0$  ( $\mu = 0$ ) and  $\hat{\nu}, \hat{\mu}$  are the maximum likelihood estimators for  $\nu$  and  $\mu$ . This approach has been used extensively to assert the discovery of new particles in HEP [2], such as in the discovery of the Higgs boson [3, 4].

As previously mentioned, the original distributions for signal and background can only be approximated by forward simulations. Most of the likelihood ratio tests at the LHC are made on the distribution of a single feature that discriminate between signal and background observations. For this, the simulated data is used together with interpolations algorithms in order to approximate the parametrized model and then use it in the hypothesis testing procedure [5].

Noticeably, it has been shown that a discriminative classifier trained to classify between signal and background can be used to obtain an equivalent likelihood ratio test [1]. Given a classifier trained to learn a monotonic function of the per event ratio  $p(x_e|\theta_0)/p(x_e|\theta_1)$  (this is the usual case since many of the commonly used classifiers learn to approximate some monotonic function of the regression function), the likelihood ratio test on the conditional distributions of the score is equivalent to the original likelihood ratio.

In this work we show how these results can be used to approximate the likelihood ratio when the underlying distribution is a weighted sum of probability distributions (mixture model). We also show that by training a set of classifiers in a pairwise manner on the components of the mixture model it is possible to improve considerably the results of the approximation.

## 2. Decomposed likelihood ratio test for mixture models

A generalized version of the signal and background mixture model of eq. (1) for several components is

$$p(x|\theta) = \sum_{i=1}^k w_i(\theta) p_i(x|\theta), \quad (3)$$

where  $w_i(\theta)$  are the mixture coefficients for each one of the components parametrized by  $\theta$ . In [1] it is shown that the likelihood ratio between two mixture models

$$\frac{p(x|\theta_0)}{p(x|\theta_1)} = \frac{\sum_{i=1}^k w_i(\theta_0) p_i(x|\theta_0)}{\sum_{j=1}^{k'} w_j(\theta_1) p_j(x|\theta_1)}, \quad (4)$$

is equal to the composition of pairwise ratios for each one of the components which is equivalent to the composition of ratios on the score distribution of pairwise trained classifiers

$$\frac{p(x|\theta_0)}{p(x|\theta_1)} = \sum_{i=1}^k \left[ \sum_{j=1}^{k'} \frac{w_j(\theta_1)}{w_i(\theta_0)} \frac{p_j(x|\theta_1)}{p_i(x|\theta_0)} \right]^{-1} = \sum_{i=1}^k \left[ \sum_{j=1}^{k'} \frac{w_j(\theta_1)}{w_i(\theta_0)} \frac{p_j(s_{i,j}(x; \theta_0, \theta_1)|\theta_1)}{p_i(s_{i,j}(x; \theta_0, \theta_1)|\theta_0)} \right]^{-1}. \quad (5)$$

In the case that the only free parameters of the mixture model are the coefficients  $w_i(\theta)$ , then each distribution  $p_i(s_{i,j}(x; \theta_0, \theta_1)|\theta)$  is independent of  $\theta$  and can be pre-computed and used after in the evaluation of the likelihood ratio. Moreover, when numerator and denominator correspond to the same distribution the values can be directed replaced by 1 avoiding unnecessary computations.

Also, for a two-class classifier the values of  $s_{j,i}(x; \theta_0, \theta_1)$  for one of the classes can be replaced by the values of  $s_{i,j}(x; \theta_0, \theta_1)$  for the opposing class, then it is only necessary to train the classifiers for  $i < j$ . This saves a lot of computation time and avoid possible variance that can be introduced by differences between  $s_{i,j}(x; \theta_0, \theta_1)$  and  $s_{j,i}(x; \theta_0, \theta_1)$  due to imperfect training. In the case of only background hypothesis versus signal plus background hypothesis it is common that the signal coefficient  $w_j(\theta_1)$  is a very small number compared to the background coefficients under the alternate hypothesis. In this conditions a classifier trained on data from the full mixture model will have a lot of problems to identify the signal since most of the useful discriminative data will lay in a small region of the feature space while the decomposed model will not face this issues.

It is possible to estimate the signal and background coefficients by using maximum likelihood on the ratios. This can be done by keeping the denominator fixed and estimate the parameters on the numerator using the maximum likelihood method.

The complete algorithm to approximate the likelihood ratio using pairwise trained classifiers can be separated into three independent stages: classifier training, score distribution estimation and composition formula computation. Since this steps are independent, each one can be solved as a different problem. In the first step any classifier satisfying the monotonic requirement can be used. In the second stage the probability distribution of the score on data from  $\theta_0$  or  $\theta_1$  can be estimated using any univariate density estimation technique such as histograms or kernel density estimation [6, 7].

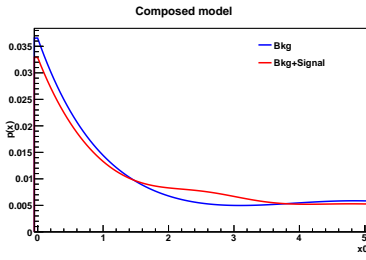
### 3. Experiments

In this section two examples of how the method works on data generated from known distributions will be presented. In both cases we study how the decomposition formula works using pairwise trained classifiers and we compare the results to the true (in this case known) likelihood ratio and to the likelihood ratios obtained by training a classifier in data from the full mixture model and replacing the original ratio with the ratio of the conditional score distributions. All studies were conducted using a simple multilayer perceptron model. This classifier shows a good tradeoff between quality of the ratios and simplicity of the model (good results were also obtained using boosted decision trees, logistic regression and support vector machines). The probability models were implemented with `Roofit` probabilistic programming language and the classifiers were implemented using `Theano` (a framework to build neural network models) and `scikit-learn` (a general framework for machine learning in python) [6, 8, 9] (the code is available for replication of the results at <https://github.com/jgpavez/systematics>).

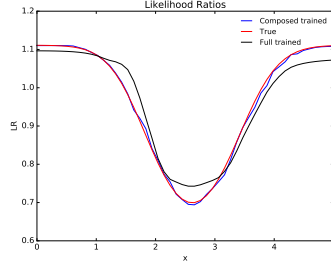
First, we present a simple case in which each component is an univariate distribution. We consider a mixture model consisting of three distributions, where  $p_0(x)$  and  $p_1(x)$  are univariate gaussian distributions while  $p_2(x)$  is a decaying exponential. The mixture models are composed by the weighted sum of those distributions where  $p_0(x)$  correspond to the signal component. The mixture models with coefficients  $W(\theta_0) = \{0., 0.3, 0.7\}$  for the only background hypothesis and  $W(\theta_1) = \{0.1, 0.27, 0.63\}$  for the background plus signal hypothesis are shown in Figure 1.

Three neural networks were trained on 200000 examples sampled from the pairs of single distributions and one on 200000 examples sampled from the full model (is important to use the same amount of training data in order to allow a fair comparison). The distribution of the score is estimated using histograms (the number of bins were carefully chosen in order to allow a good approximation while minimizing poisson fluctuations). The composed ratio using eq. (5), the ratio estimated using a classifier trained on data from the full model and the true density ratio are shown in Figure 2 and Figure 3 for different values of the signal contribution (0.1 and 0.05) while keeping the ratio between the background contributions fixed.

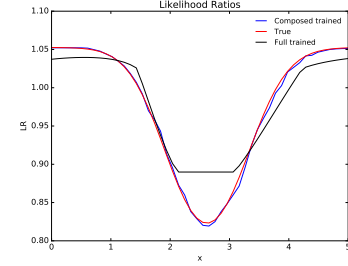
It can be seen that the ratios obtained by the composition method are better (closer to the



**Figure 1.** The mixture models.

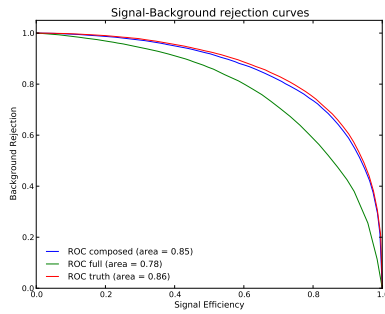


**Figure 2.** Density ratio comparison for a signal weight of 0.1.

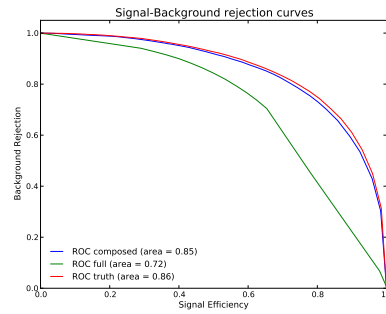


**Figure 3.** Density ratio comparison for a signal weight of 0.05.

104 true ratios) than the obtained using a model trained on data from the full mixture model and  
 105 this become clearer when the signal contribution is smaller.



**Figure 4.** Background rejection versus signal efficiency curve comparison for a signal weight of 0.1.



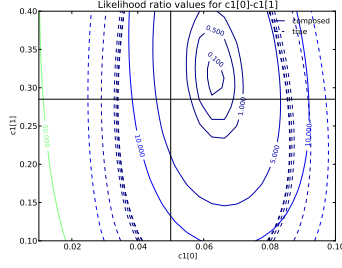
**Figure 5.** Background rejection versus signal efficiency curve comparison for a signal weight of 0.05.

106 The same experiments are repeated but now considering a much harder mixture model  
 107 consisting of three distributions, each one composed by the sum of three 10-dimensional  
 108 multivariate gaussian distributions. In this case datasets of size 500000 were used to train  
 109 all the models. This is more similar to what can be found in real physics experiments. To  
 110 evaluate the estimated ratios the background rejection versus signal efficiency curves on data  
 111 from  $\theta_0$  and  $\theta_1$  are used, employing the density ratio as discriminative variable. Again, the values  
 112 for each one of the three cases (decomposed, full and true) are shown for a signal contribution  
 113 of 0.1 and 0.05 in Figure 4 and Figure 5.

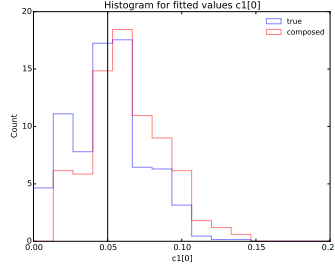
114 The values of each one of the coefficients of the mixture model can be estimated by using the  
 115 method of maximum likelihood as explained in Section 2. In Figure 6 the contour plot for the  
 116 likelihood ratio values given the signal and one background weight, obtained using the estimated  
 117 density ratios and the true density ratios is shown. Histogram for the fitted values (estimated  
 118 and true) of each one of the coefficients on 80 different pseudo-datasets of size 1000 are shown  
 119 in Figure 7 and Figure 8. It can be seen on the histograms that the estimations are unbiased.

## 120 4. Conclusions

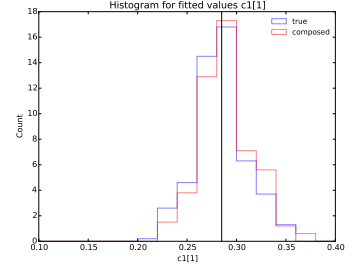
121 We have shown the power of using discriminative classifiers in order to approximate the likelihood  
 122 ratio. In the case of mixture models we proved that the decomposed version of the ratio can



**Figure 6.** Likelihood ratio values given signal ( $c1[0]$ ) and bkg. ( $c1[1]$ ) weights.



**Figure 7.** Histogram of fitted values for the signal weight ( $c1[0]$ ).



**Figure 8.** Histogram of fitted values for the bkg. weight ( $c1[1]$ ).

greatly improve the quality of the results. Using the same method it was shown how to estimate the unknown parameters of the model. Initial experiments have been conducted on simulated data from different Higgs production mechanisms obtaining encouraging results. An open source Python package was implemented allowing to easily use the presented method (the package can be found at <https://github.com/diana-hep/car1>).

## 5. Acknowledgments

KC and GL are supported by the DIANA grant ACl-14503. JP was partially supported by the Scientific and Technological Center of Valparaiso (CCTVal) under Fondecyt grant BASAL FB0821. JP would like to thank Hector Allende and Carlos Valle for helpful discussions.

## References

- [1] Cranmer K 2015 *ArXiv e-prints* <http://arxiv.org/abs/1506.02169> (*Preprint* 1506.02169)
- [2] Cowan G, Cranmer K, Gross E and Vitells O 2010 *Eur.Phys.J.* **C71** 1554 (*Preprint* 1007.1727) URL <http://arxiv.org/abs/1007.1727>
- [3] The ATLAS Collaboration 2012 *Phys.Lett.* **B716** 1–29 (*Preprint* 1207.7214)
- [4] The CMS Collaboration 2012 *Phys.Lett.* **B716** 30–61 (*Preprint* 1207.7235)
- [5] Cranmer K, Lewis G, Moneta L, Shibata A and Verkerke W 2012 CERN-OPEN-2012-016, <http://inspirehep.net/record/1236448>
- [6] Verkerke W and Kirkby D P 2003 *eConf* **C0303241** MOLT007 (*Preprint* physics/0306116)
- [7] Cranmer K S 2001 *Comput. Phys. Commun.* **136** 198–207 (*Preprint* hep-ex/0011057)
- [8] Bergstra J, Breuleux O, Bastien F, Lamblin P, Pascanu R, Desjardins G, Turian J, Warde-Farley D and Bengio Y 2010 Theano: a CPU and GPU math expression compiler *Proceedings of the Python for Scientific Computing Conference (SciPy)* oral Presentation
- [9] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M and Duchesnay E 2011 *Journal of Machine Learning Research* **12** 2825–2830