Queen Mary
University of London

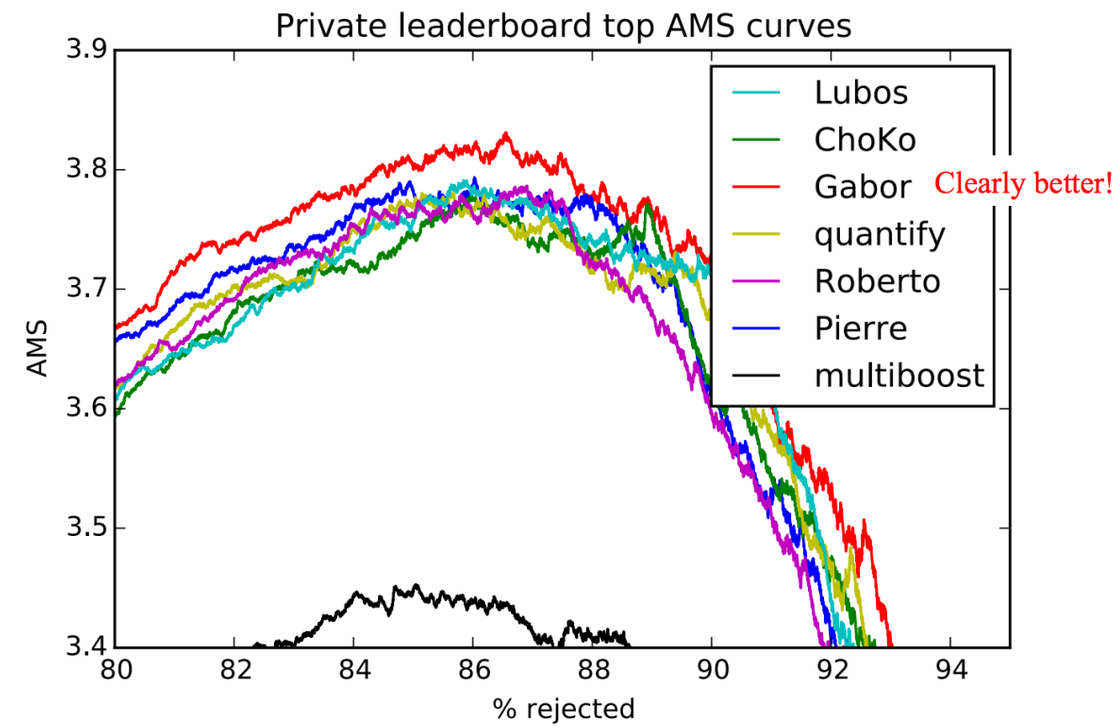
Cross Validation for Machine Learning with TMVA

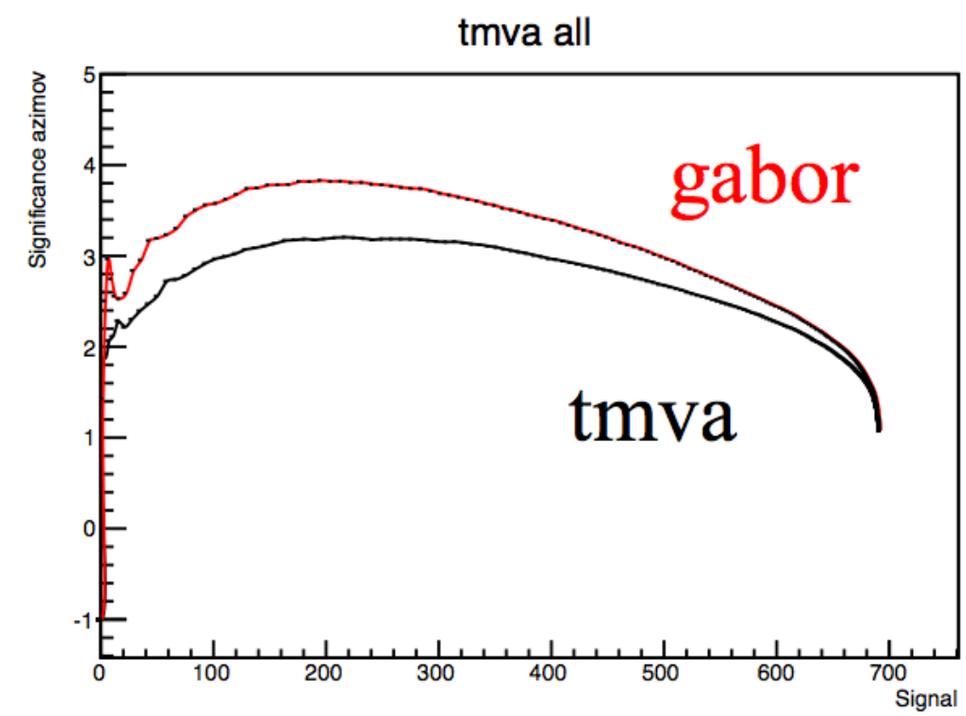
Agni Bethani, Adrian Bevan, Rodrigo Gamboa Goni
Jon Hays, Lewis Millward, Tom Stevenson

LHC Machine Learning WG
4th December 2015

Machine Learning
Cross Validation with TMVA

- Cross validation motivated by issue of generalisation.
- Need confidence that the trained MVA is robust and the performance on unseen samples can be accurately predicted.
- Winner of 2014 Higgs Machine Learning Challenge used *k*-fold cross validation and bagging to obtain better MVA performance than other entries.
- Investigating use of cross validation for VH MVA.





Taken from slides by D. Rousseau
http://indico.cern.ch/event/382895/

- In machine learning validation techniques are required for model selection and performance estimation.

Model Selection:

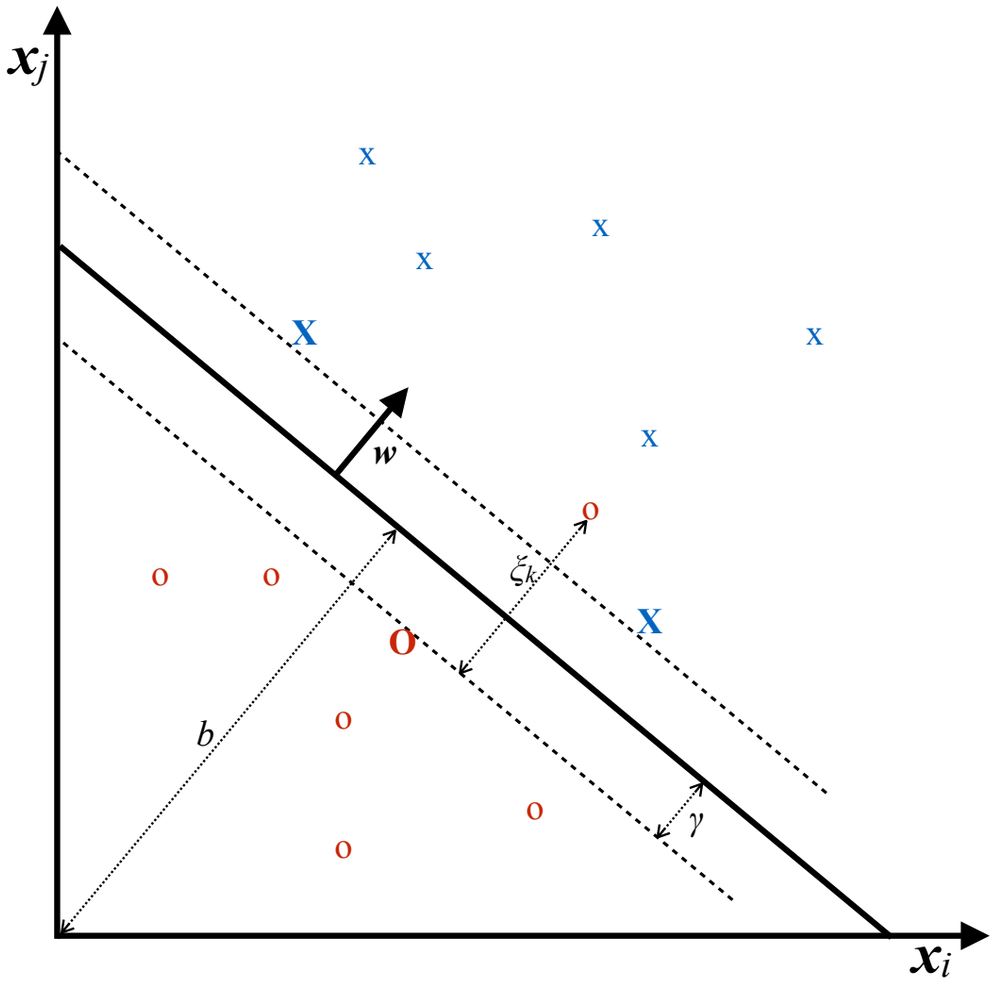
- Most methods have at least one free parameter e.g.
  - BDT - #trees, min node size, etc.
  - MLP - #neurons, #layers, weight vectors, etc.
  - SVM - kernel function, kernel parameters, cost, etc.
- How are these parameters of models “optimally” selected?

Performance Estimation:

- How does the chosen model perform?
- Usually true error rate is used (misclassification rate for the entire dataset).

Machine Learning

Aside: Support Vector Machines (SVMs) in TMVA



- Part of Motivation for cross-validation:
  - Written by Marcin Wolter and Andrzej Zemla.
  - We have been expanding implementation of SVMs in TMVA.

- Only points closest to the decision boundary are used to define the optimally separating hyper surface:
  - These are the support vectors.
  - Capitalised points in picture.

$$K(\mathbf{x},\mathbf{z}) = \exp\left(-\Gamma \parallel \mathbf{x}-\mathbf{z} \parallel^2\right) \quad (1)$$

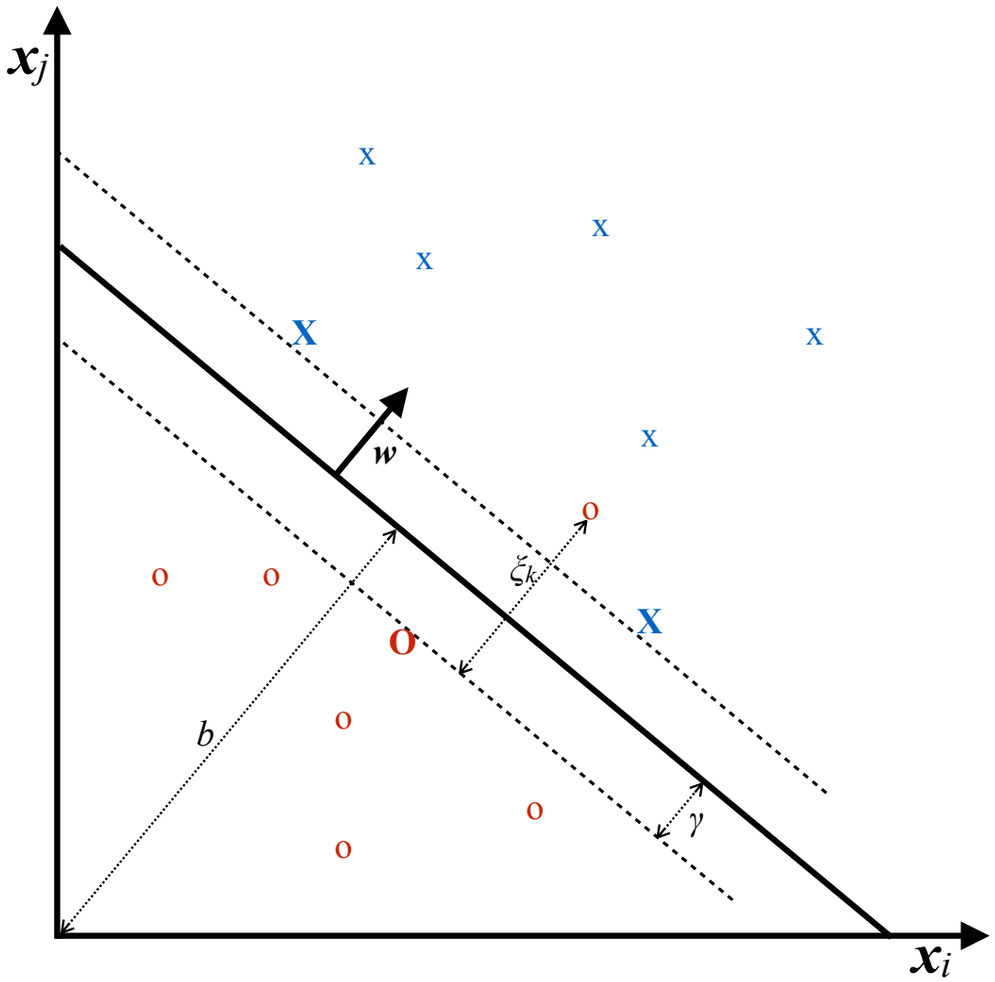
$$K(\mathbf{x},\mathbf{z}) = \prod_i \left(-\Gamma_i \parallel \mathbf{x}_i-\mathbf{z}_i \parallel^2\right) \quad (2)$$

$$K(\mathbf{x},\mathbf{z}) = (\mathbf{x}\cdot\mathbf{z}+\theta)^d \quad (3)$$

- ξ_k allows for possibility of mis-classification:
  - Means algorithm can be computationally expensive.

Machine Learning

Aside: Support Vector Machines (SVMs) in TMVA



$$K(\mathbf{x},\mathbf{z}) = \exp\left(-\Gamma \parallel \mathbf{x}-\mathbf{z} \parallel^2\right) \quad (1)$$

$$K(\mathbf{x},\mathbf{z}) = \prod_i \left(-\Gamma_i \parallel \mathbf{x}_i-\mathbf{z}_i \parallel^2\right) \quad (2)$$

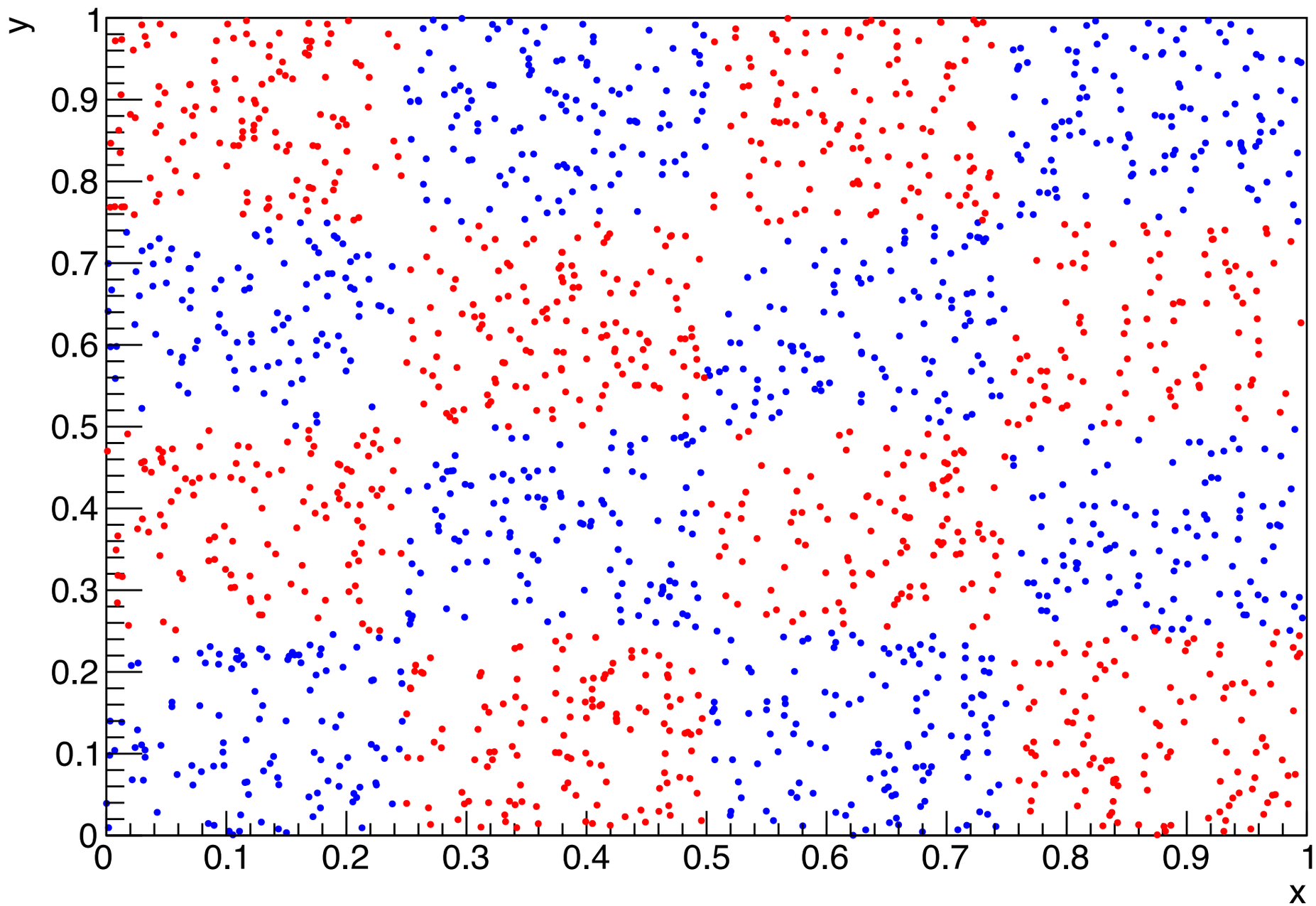
$$K(\mathbf{x},\mathbf{z}) = (\mathbf{x}\cdot\mathbf{z}+\theta)^d \quad (3)$$

- Original Implementation Kernel:
  - Radial Basis Function (RBF) (1)
- Added additional kernel functions:
  - “Multi-Gaussian” (2)
  - Polynomial (3) (re-enabled)
  - Product/Sum of kernels
- Implemented parameter optimisation:
  - Kernel parameters
  - Cost parameter
- Some additional functionality added:
  - Weighted cost
  - etc.

Machine Learning

Aside: Support Vector Machines (SVMs) in TMVA

- Checkerboard dataset example



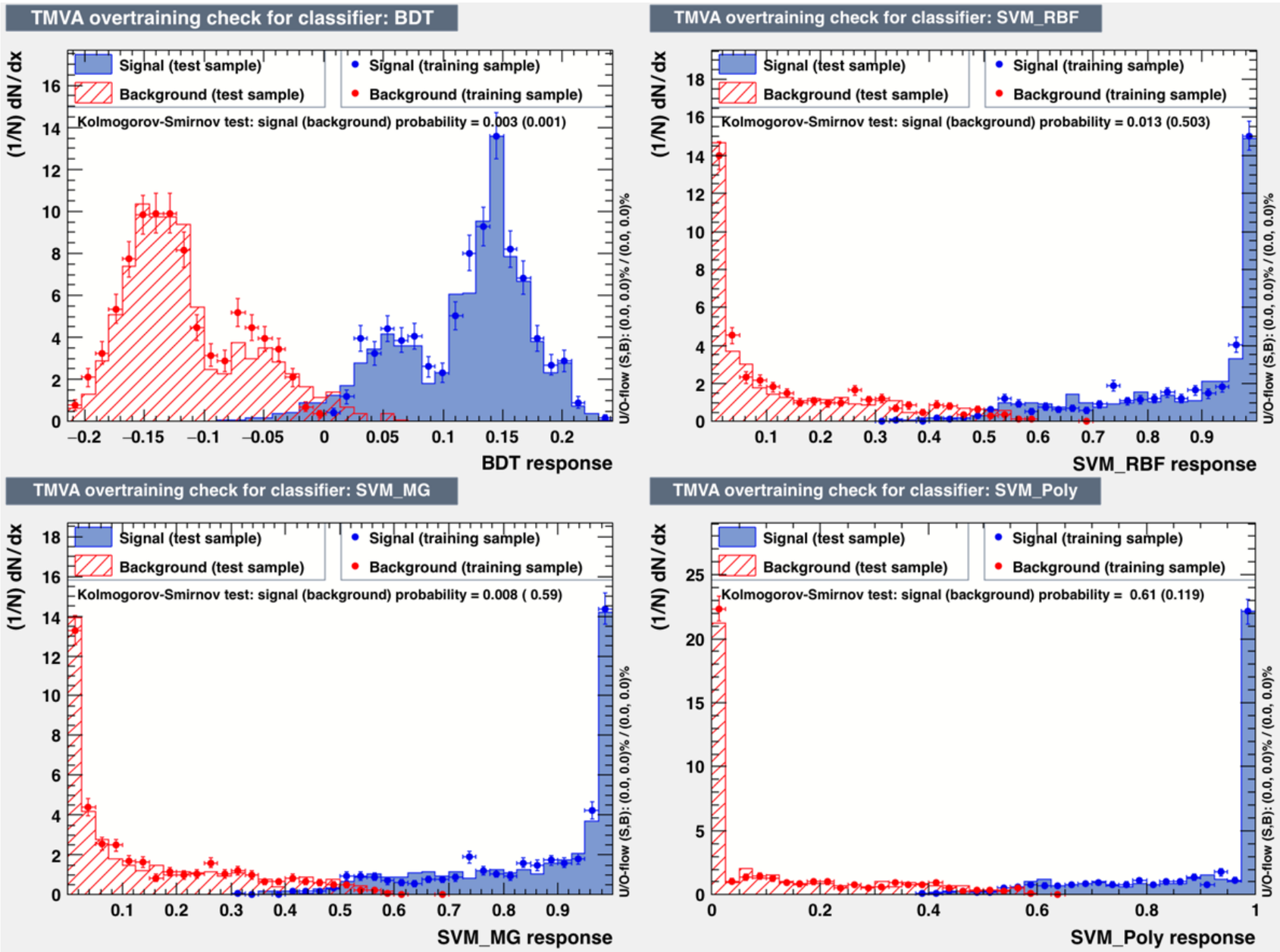
Machine Learning

Aside: Support Vector Machines (SVMs) in TMVA

- Checkerboard dataset example - (fully optimised MVA Hold-out method)

BDT

SVM Multi-Gaussian (2)



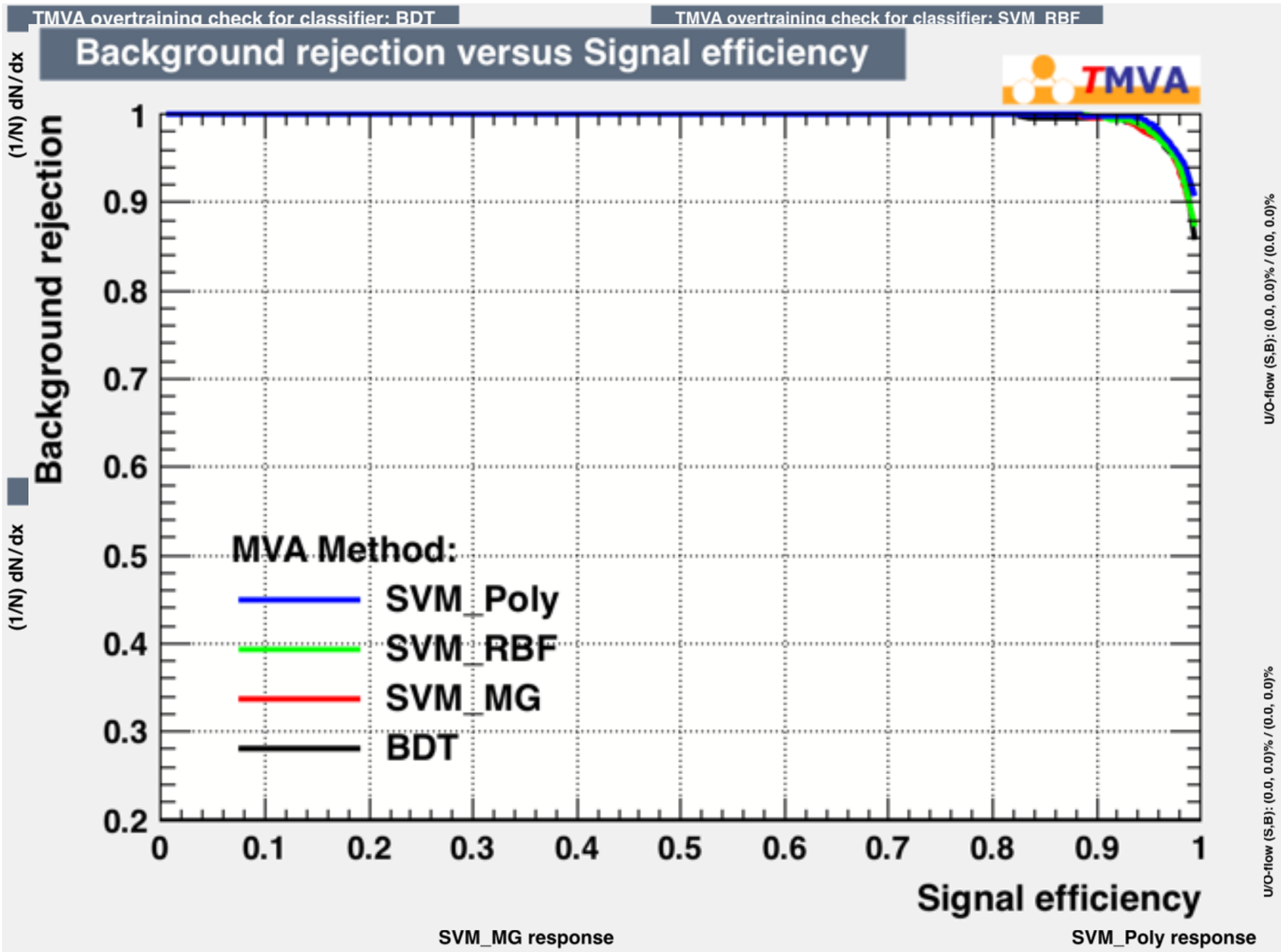
SVM RBF Kernel (1)

SVM Polynomial Kernel (3)

Machine Learning

Aside: Support Vector Machines (SVMs) in TMVA

- Checkerboard dataset example - (fully optimised MVA Hold-out method)



- Back to cross validation and issues of Model Selection and Error Estimation.

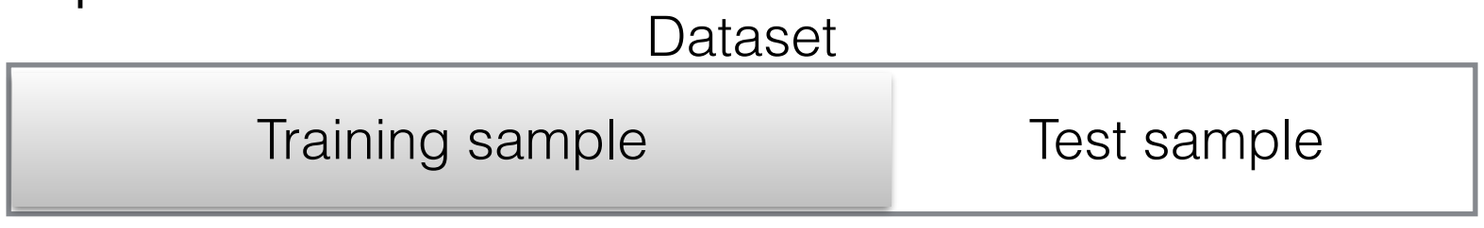
- For an unlimited dataset these issues are trivial, simply iterate through parameters and find model with lowest error rate.

- In reality datasets are smaller than we would like.

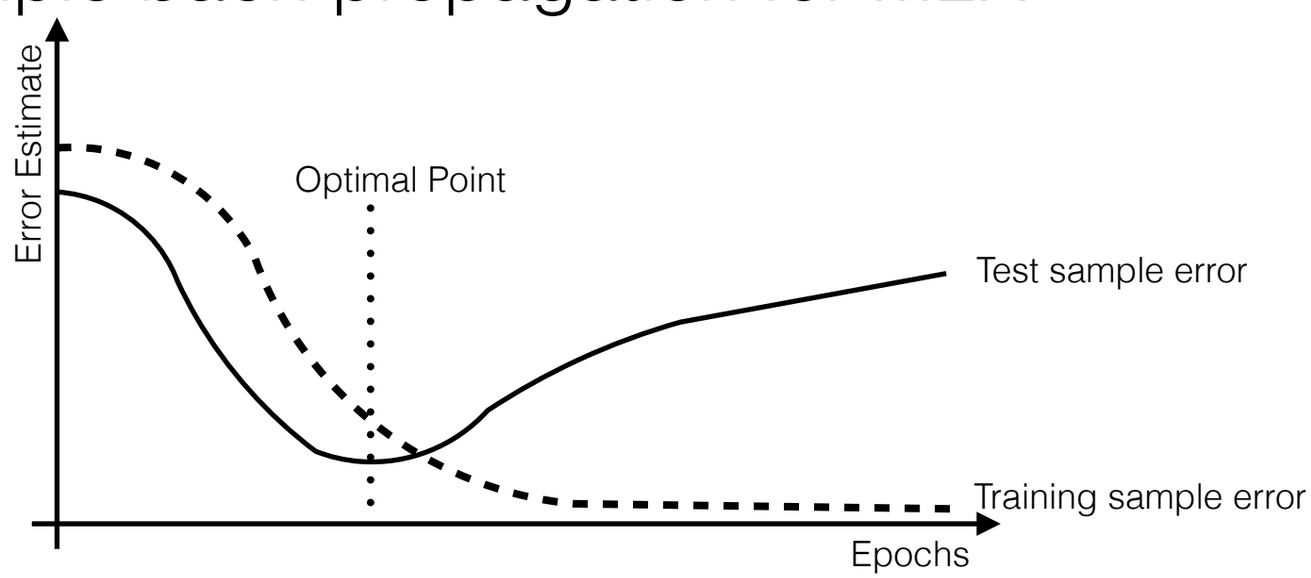
- Naïvely use whole dataset to select and train classifier and to estimate error.
  - Leads to overfitting/overtraining as classifier learns dataset and performs worse on unseen data.
  - Overfitting more distinct for classifiers with large number of tuneable parameters.
  - Also gives overly optimistic estimation of error rate.

Queen Mary
University of London

- Potential way to overcome these issues is use hold-out technique, splitting the dataset into training and test subsamples.



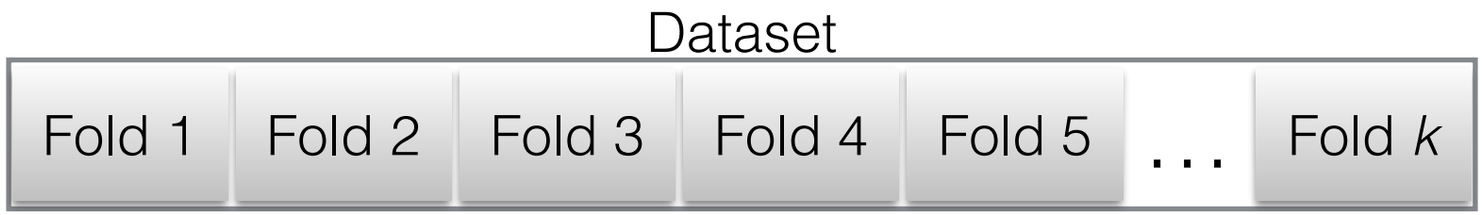
- Can use these datasets to select "optimal" parameters, for example back-propagation for MLP.



- However may give misleading error estimate depending on how the data is split.

- May not be able to reserve a large portion of data for testing, so hold-out method may not be viable.

- Instead can use *k*-fold cross-validation:



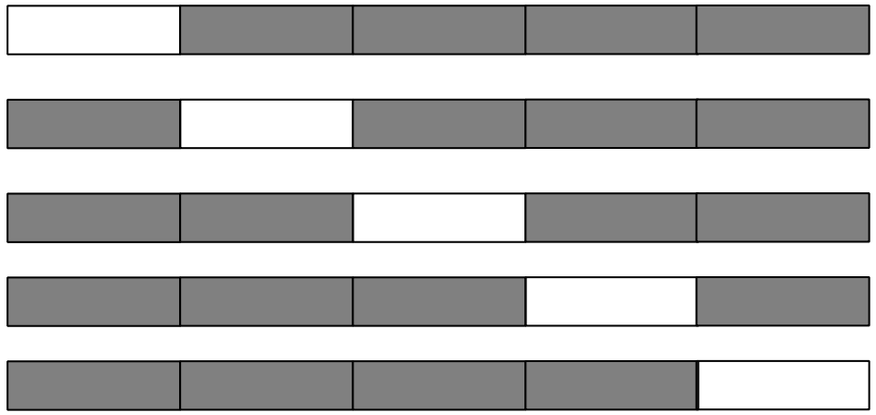
  - Split the dataset into *k* randomly sampled independent subsets (folds).
  - Train classifier with *k*-1 folds and test with remaining fold.
  - Repeat k times.

- Advantage of using the whole dataset for testing and training.
- True error rate is then estimated using average error rate:

$$E = \frac{1}{k}\sum_{i=1}^{k} E_i.$$

1) Reserve a test sample from the data $\Omega \to \Omega' \subset \Omega$ (if one wants to validate generalisation beyond the *k*-fold cross validation step).
2) Randomly split the remaining data into k sub samples:
$$\Omega' \to \Omega_i, i = 1, 2, \ldots k \quad .$$
3) Cycle through training *k* times, each time leaving one sub-sample out.

e.g. 5-fold cross validation: train 5 times dropping out one sub-sample at a time:



Use average MVA parameter configuration obtained from the *k*-folds.

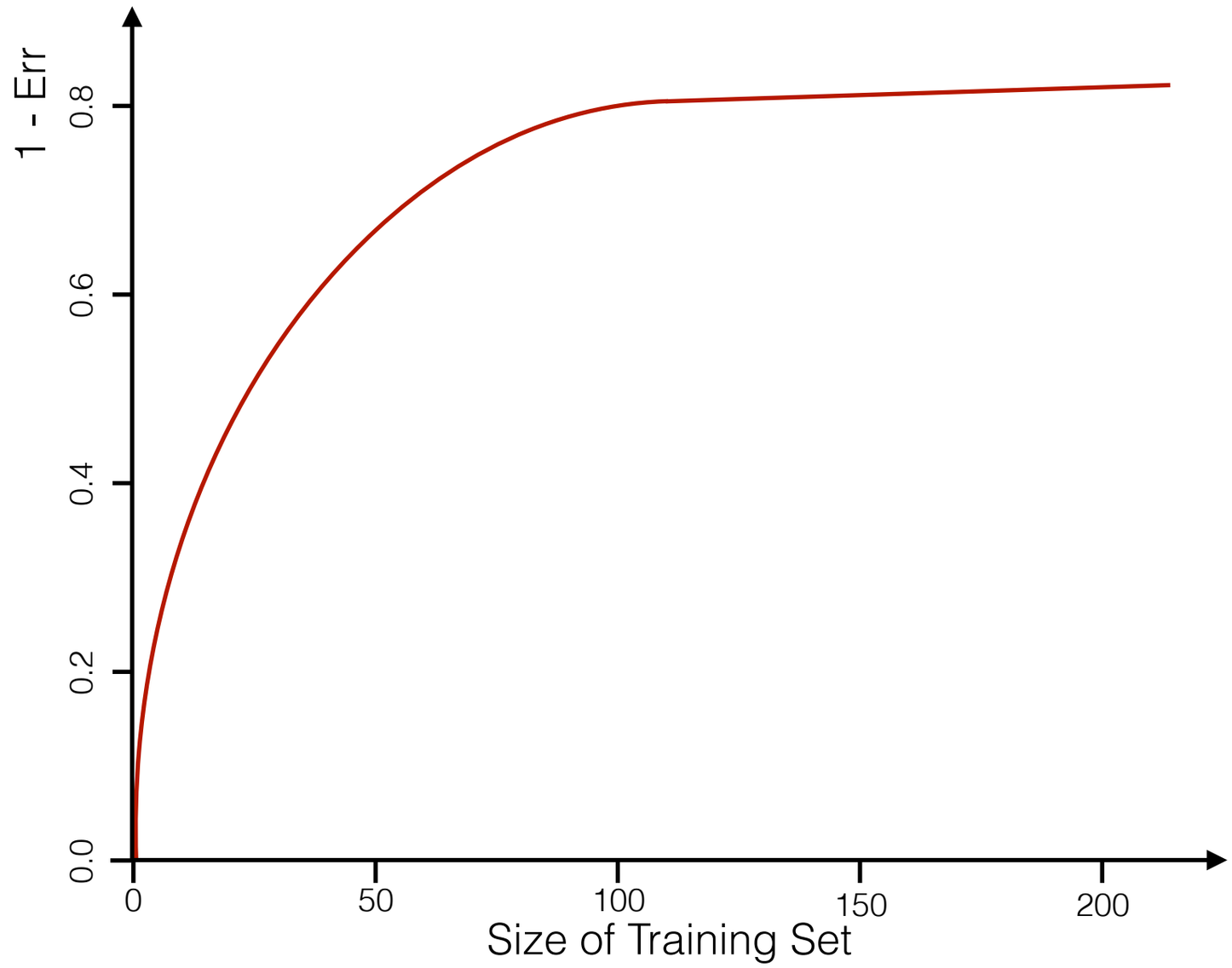
limiting cases: *k*=2: this gives the hold out cross validation method
k=N(data): this gives the leave-one-out cross validation method.

Courtesy of Adrian Bevan

- How many folds???
- Large number of folds gives good estimate of average error rate (bias of the estimator is small, approximately unbiased for leave-one-out). However the variance of the estimator is large and computational time is long.
- Reverse is true for small number of folds.
- In reality choice is motivated by the size of the dataset, i.e. sparse dataset need to go to extreme of leave-one-out method to train on as much data as possible.

Queen Mary
University of London

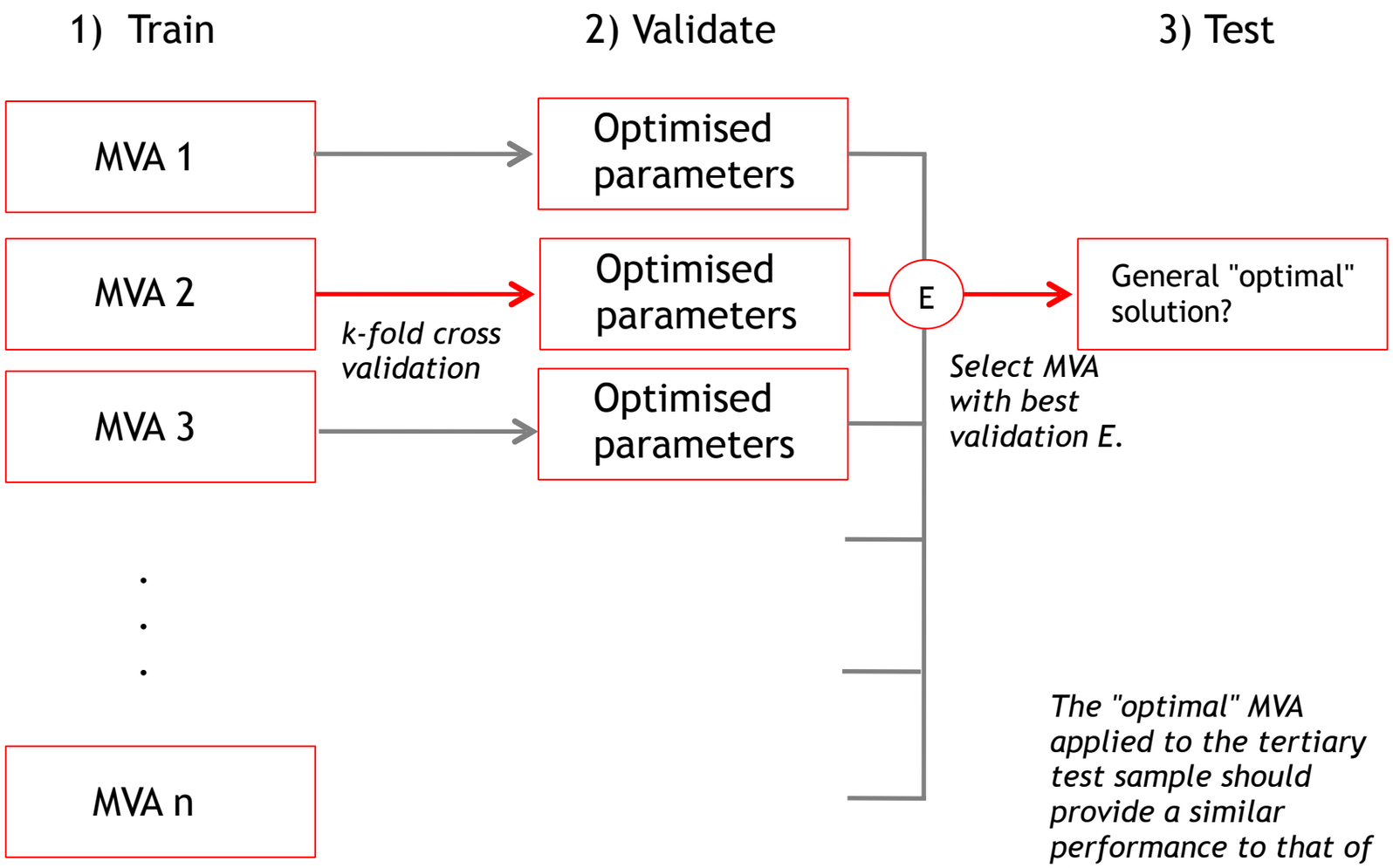
- Hypothetical example:
  - For sample size of 200, 5 fold CV will estimate the error with similar performance on training set of 160 to that of the full sample.
  - However for sample of 50, 5 fold CV will give a larger error than not using CV.



- Common choices are between 5 & 10 folds, however *k* should be determined for the given problem.

- Ideally 3 statistically independent dataset; training, validation and testing.
- Training and validation sets used to choose classifier/ model and tune parameters.
- Test set used to assess performance of final fully trained classifier.
- Avoids bias from using the same sample for model selection and parameter tuning.

Machine Learning
Cross Validation with TMVA



MVA algorithms with different generic architectures, and weight parameters to be determined.

Each optimised MVA has a corresponding validation error E. This FOM can be used to determine the "optimal" MVA to use with the test sample.

The "optimal" MVA applied to the tertiary test sample should provide a similar performance to that of the training and validation sample if the MVA is to be considered general.

If not; start the process again.

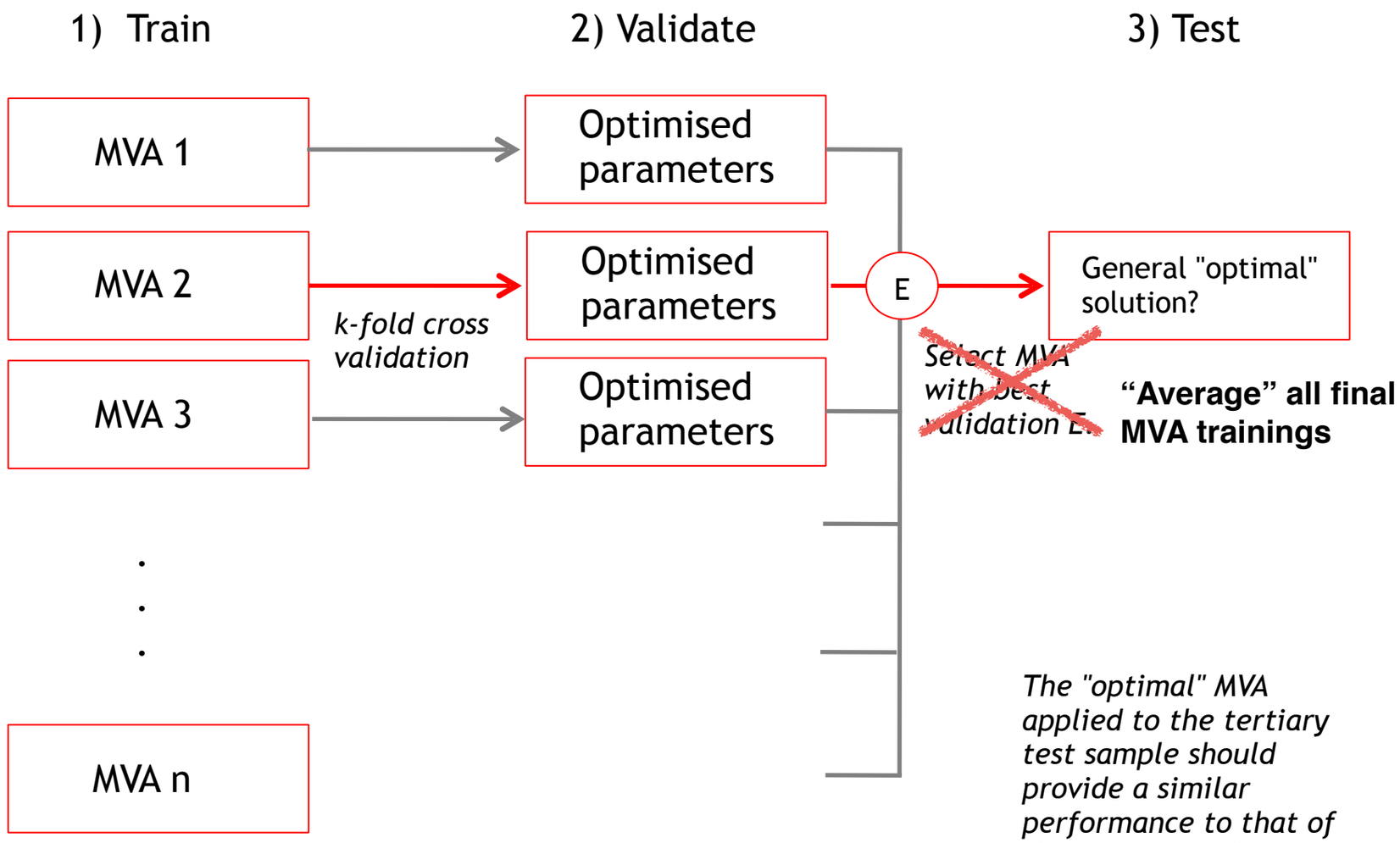
Courtesy of Adrian Bevan

- Taking the “best” performing MVA doesn’t necessarily give the desired output.
  - e.g. some pathologies in distributions.

- Also involves throwing away a large number of trainings.

- Take the aggregated output of all the final trained MVAs on the test sample in some form of average.

Machine Learning
Cross Validation with TMVA



MVA algorithms with different generic architectures, and weight parameters to be determined.

Each optimised MVA has a corresponding validation error E. This FOM can be used to determine the "optimal" MVA to use with the test sample.

The "optimal" MVA applied to the tertiary test sample should provide a similar performance to that of the training and validation sample if the MVA is to be considered general.

If not; start the process again.

Courtesy of Adrian Bevan

Queen Mary
University of London

- Developed tool to perform above processes.

- Currently used on top of TMVA.

- Basic functionality:
  - Provide input signal and background files.
  - Splits into the 3 samples:
    - Training, validation and test.
  - Splits into k-folds.
  - Performs optimisation, training and testing.
  - Produces “best” performing MVA and the average of the final trainings as output.

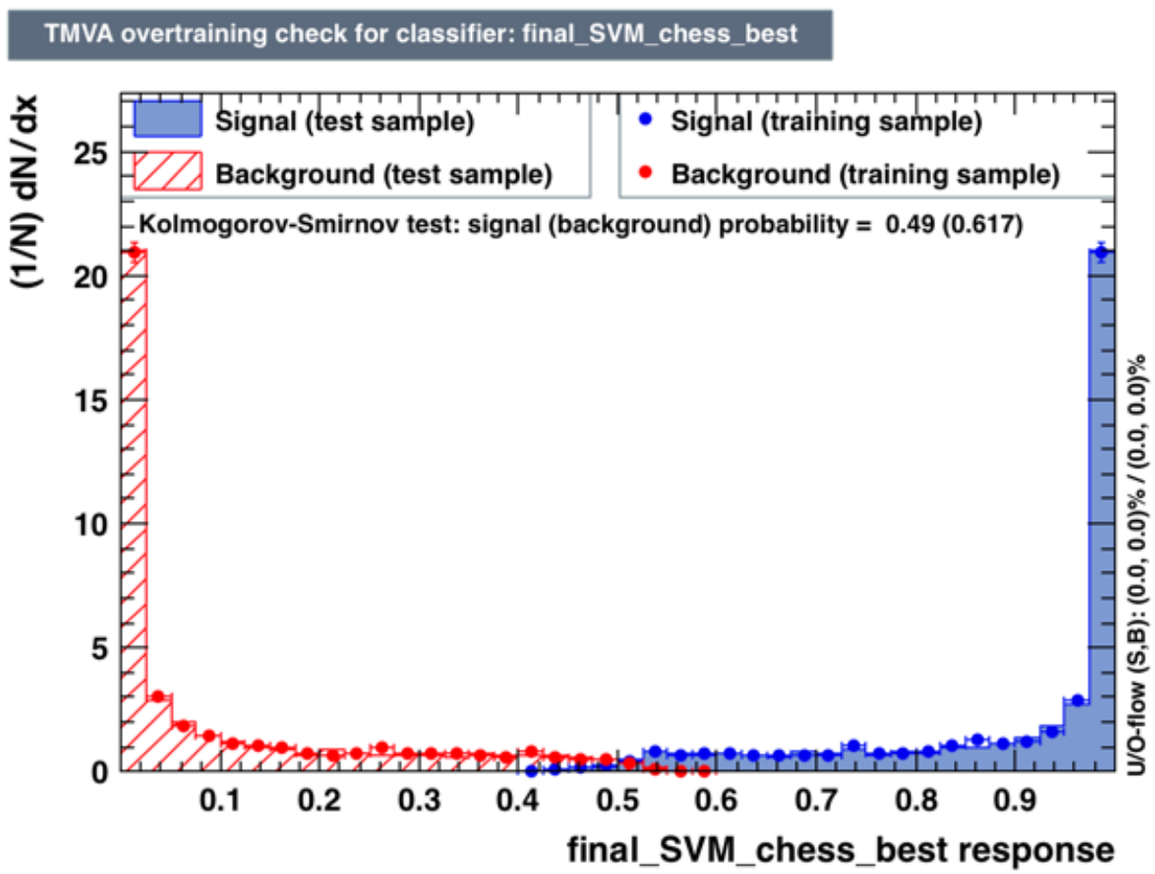
- Some current issues:
  - How to take the average?
  - Average output file not compatible with TMVAGui.
  - Integration into TMVA.

Machine Learning
Cross Validation with TMVA

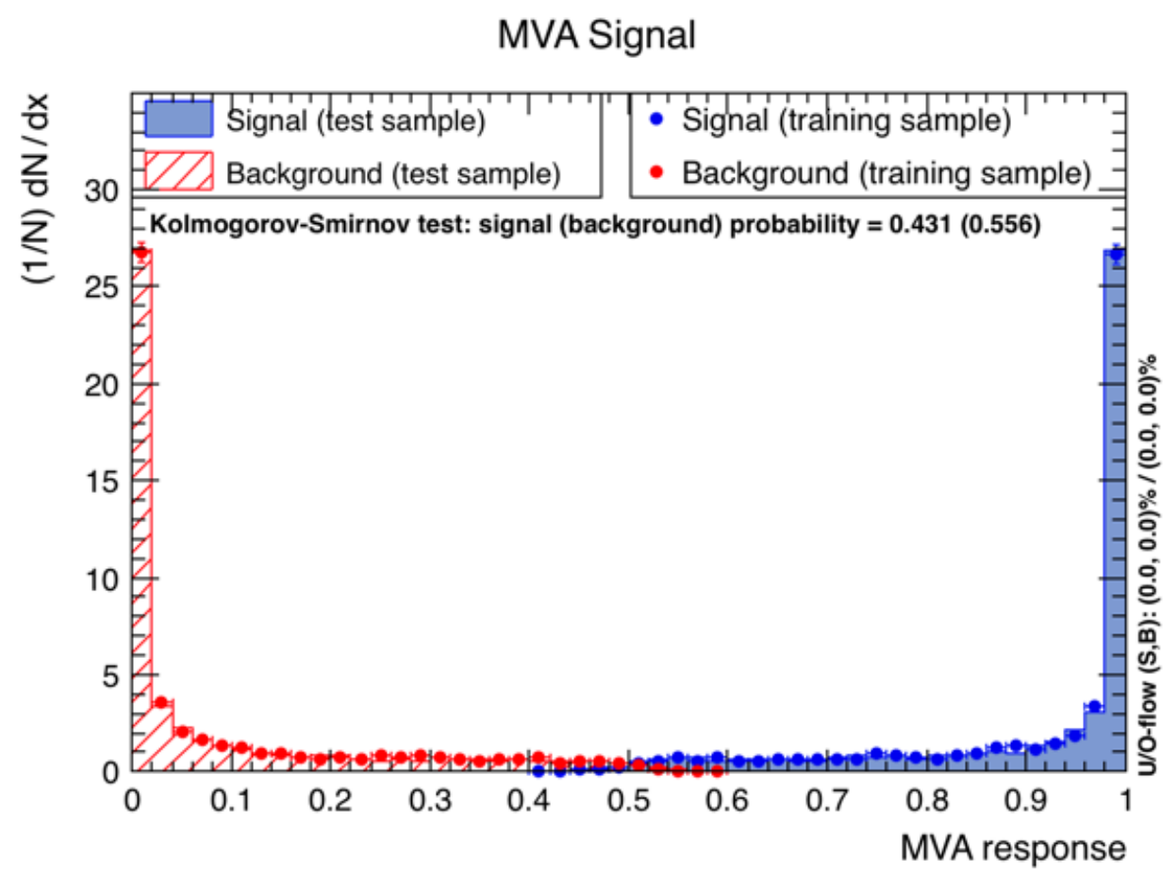
Queen Mary University of London

- 4-fold cross validation on checkerboard - SVM RBF

4-fold Best



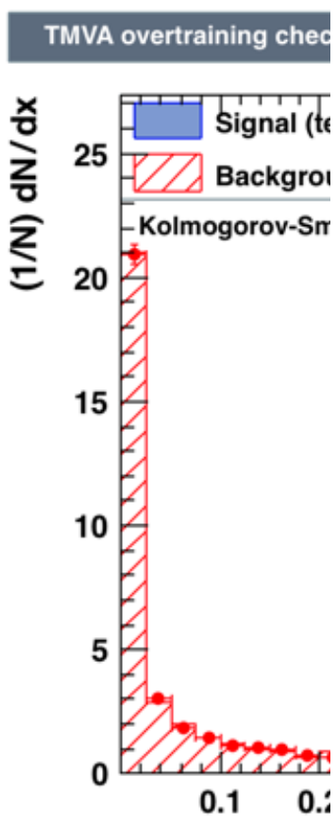
4-fold Average

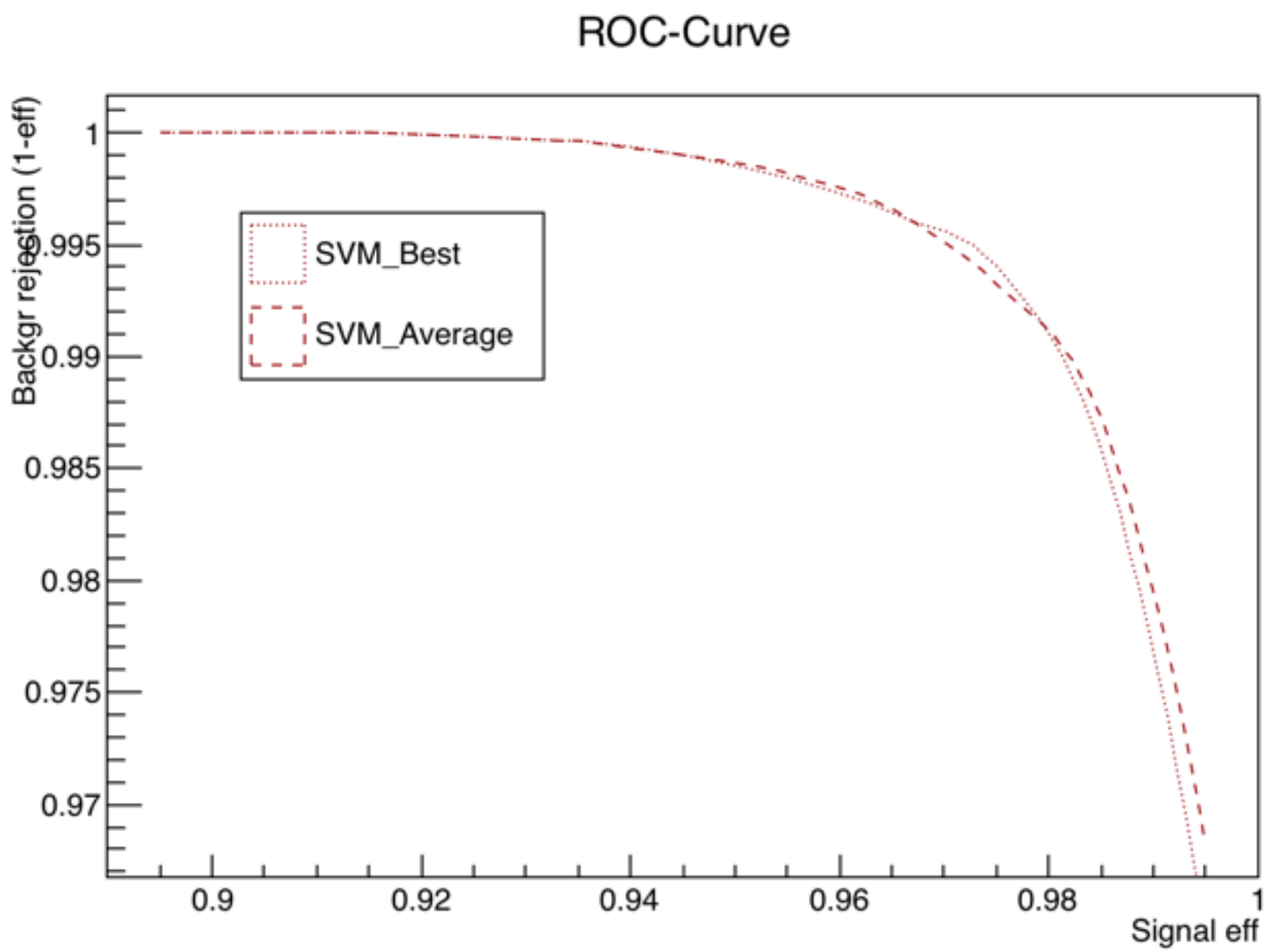


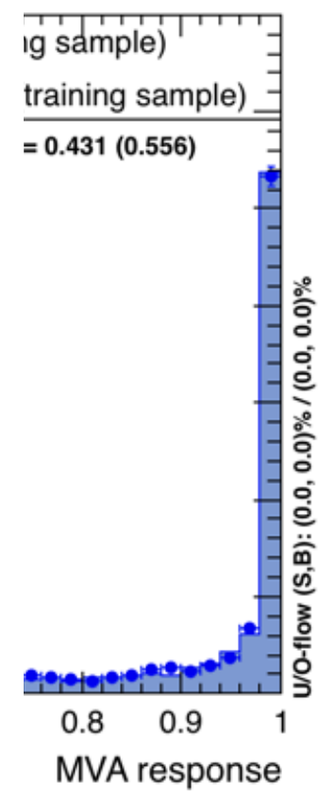
Machine Learning
Cross Validation with TMVA

Queen Mary
University of London

- 4-fold cross validation on checkerboard - SVM RBF





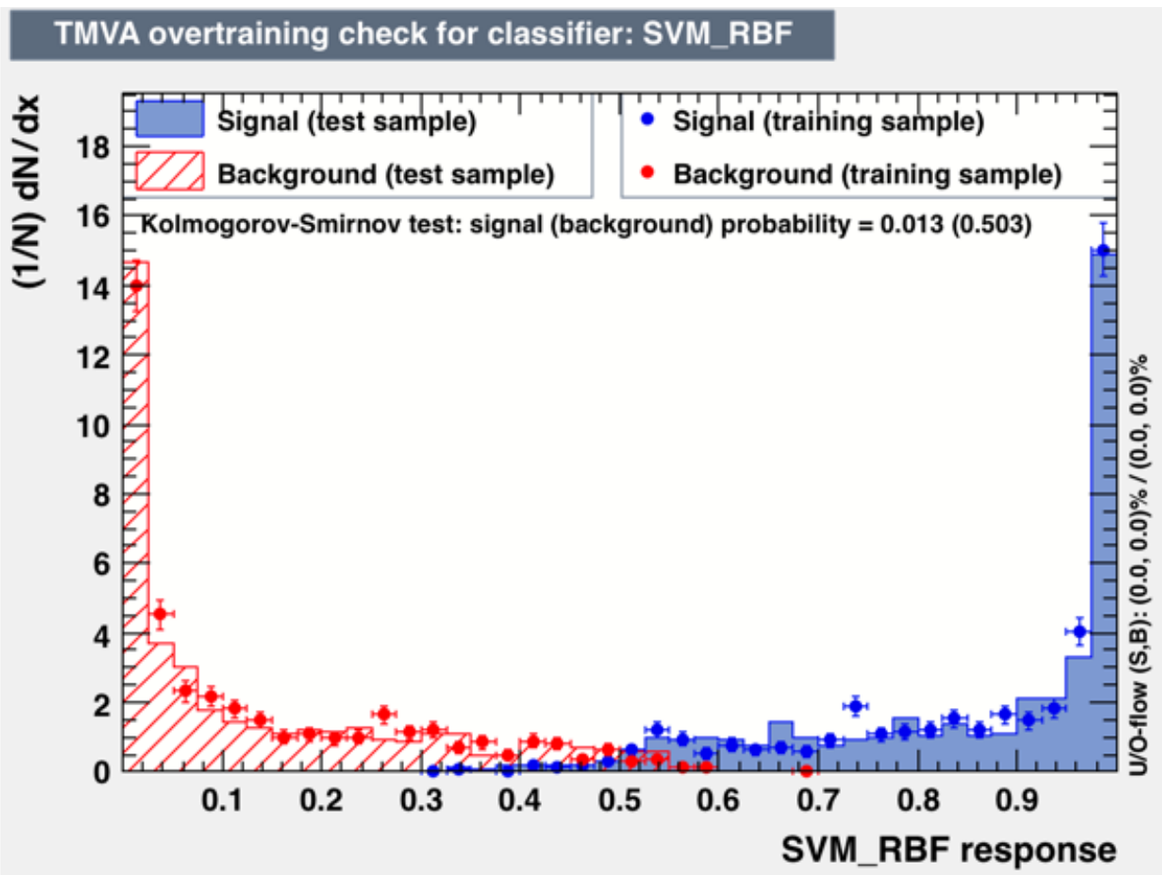


Machine Learning
Cross Validation with TMVA

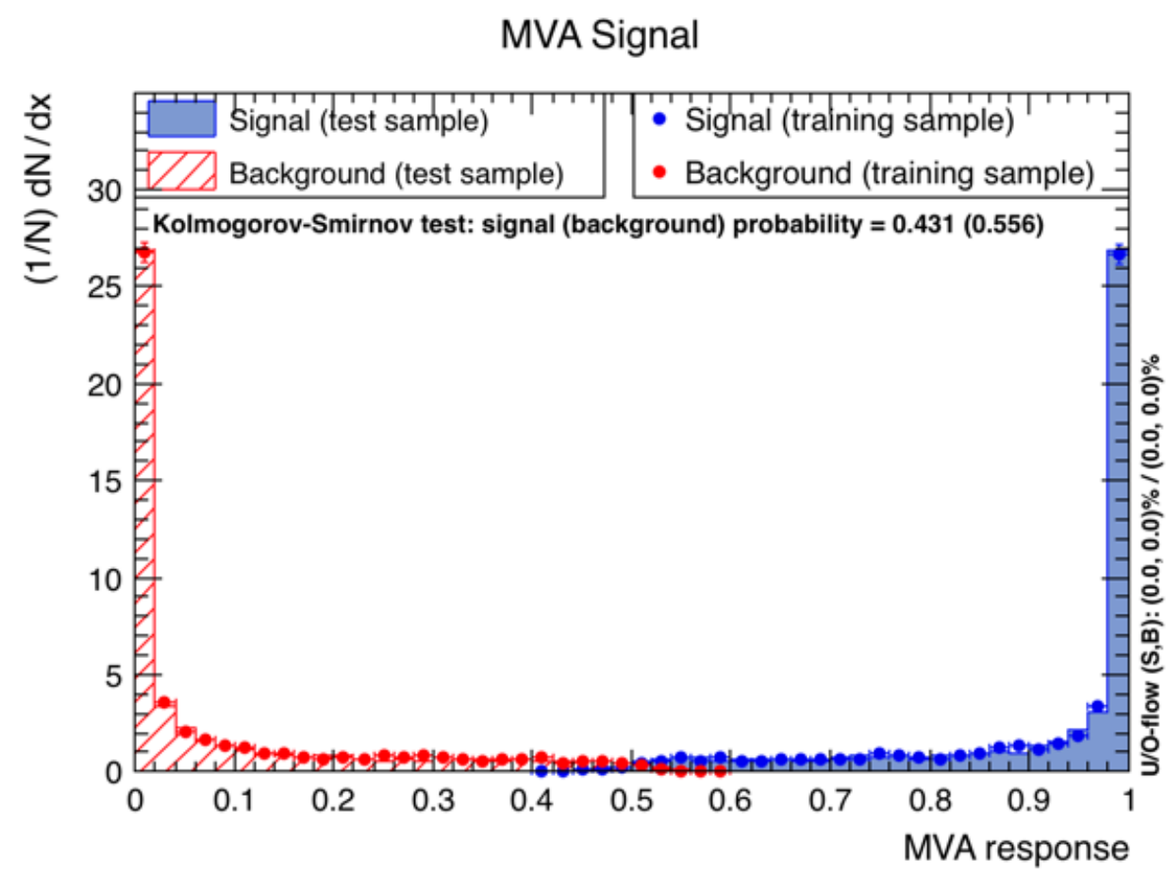
Queen Mary University of London

- 4-fold cross validation on checkerboard - SVM RBF

Hold-out

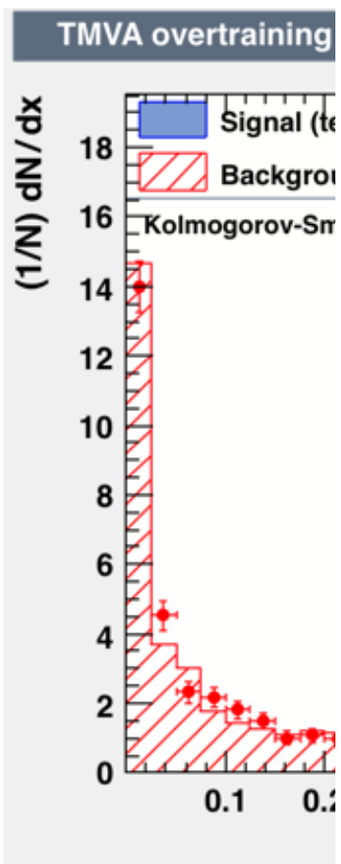


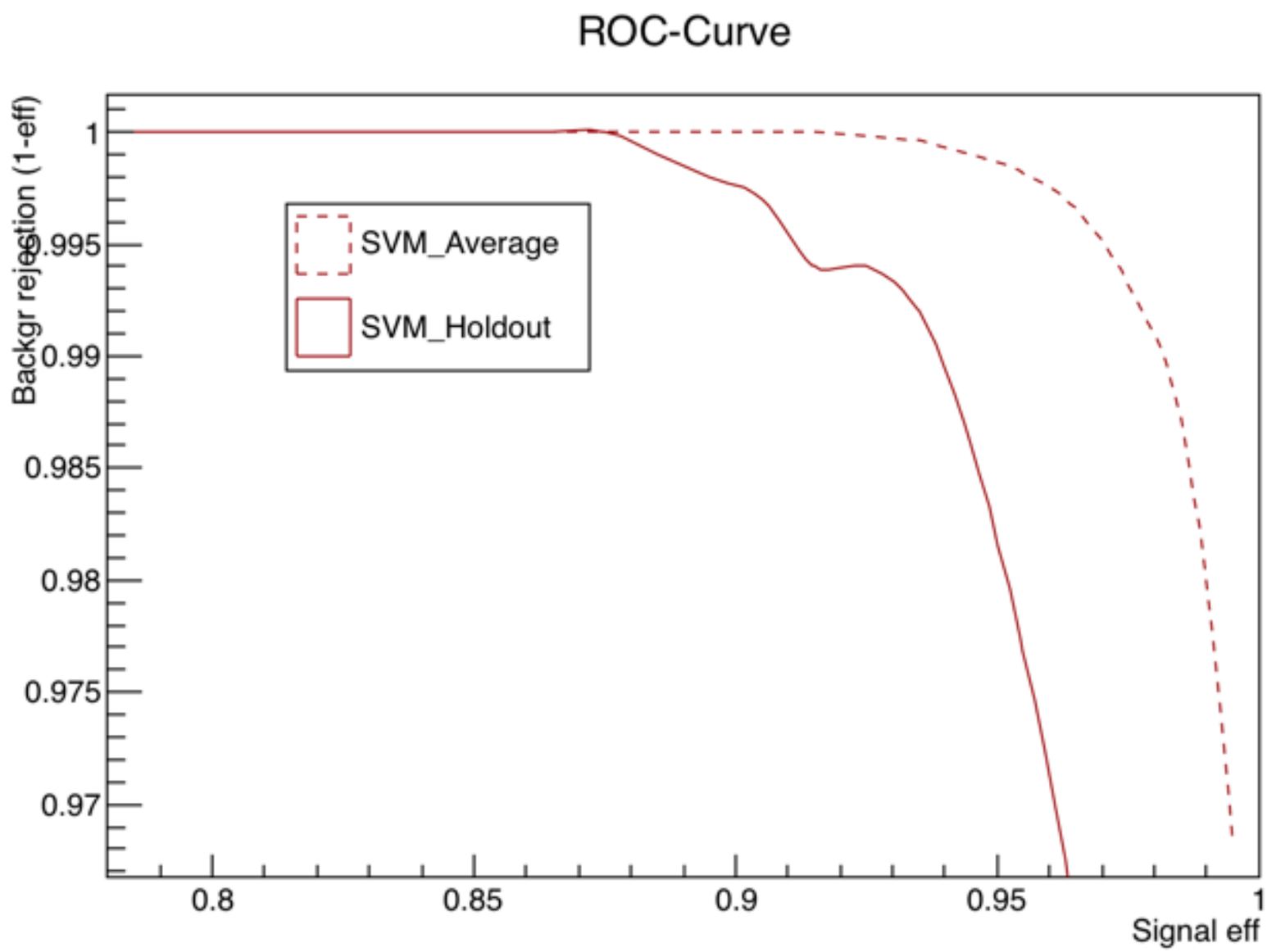
4-fold Average

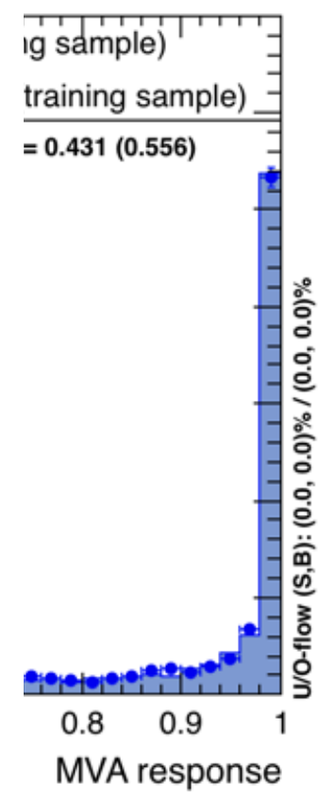


Machine Learning
Cross Validation with TMVA

- 4-fold cross validation on checkerboard - SVM RBF





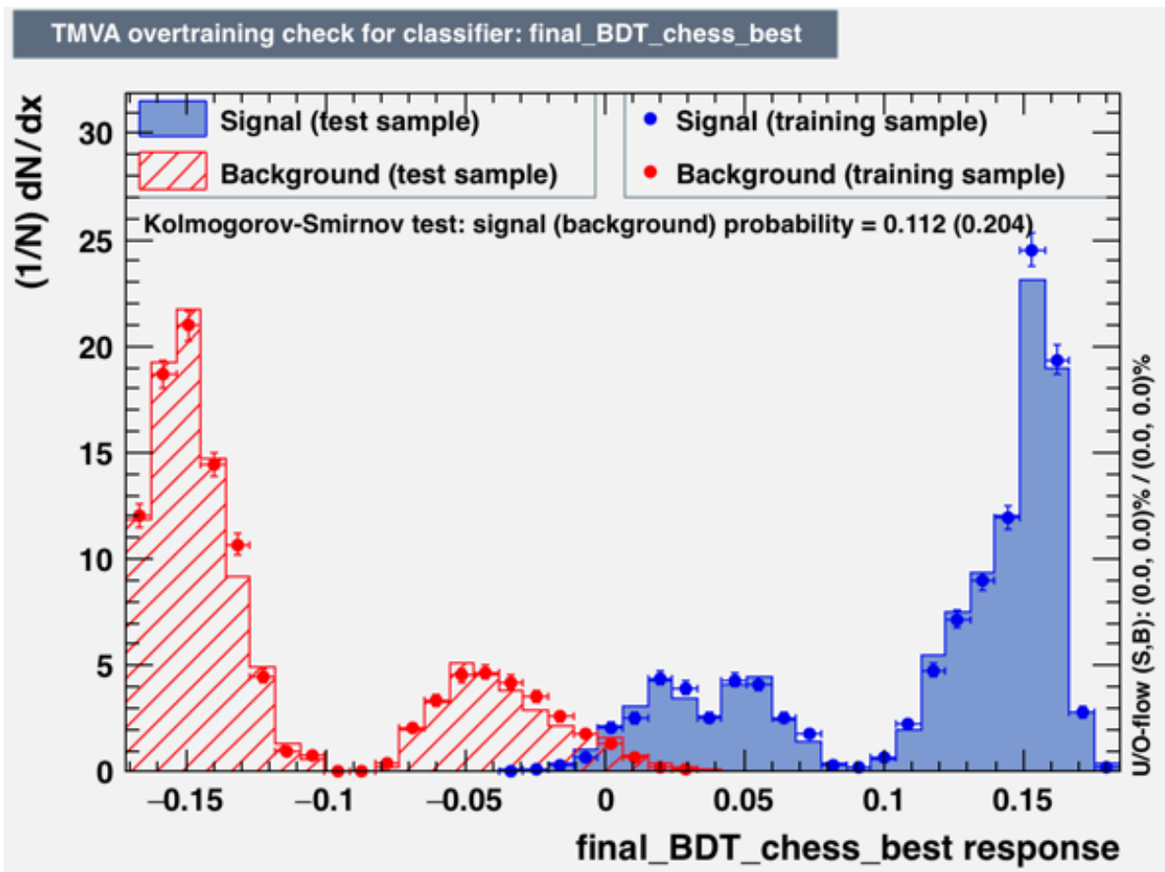


Machine Learning
Cross Validation with TMVA

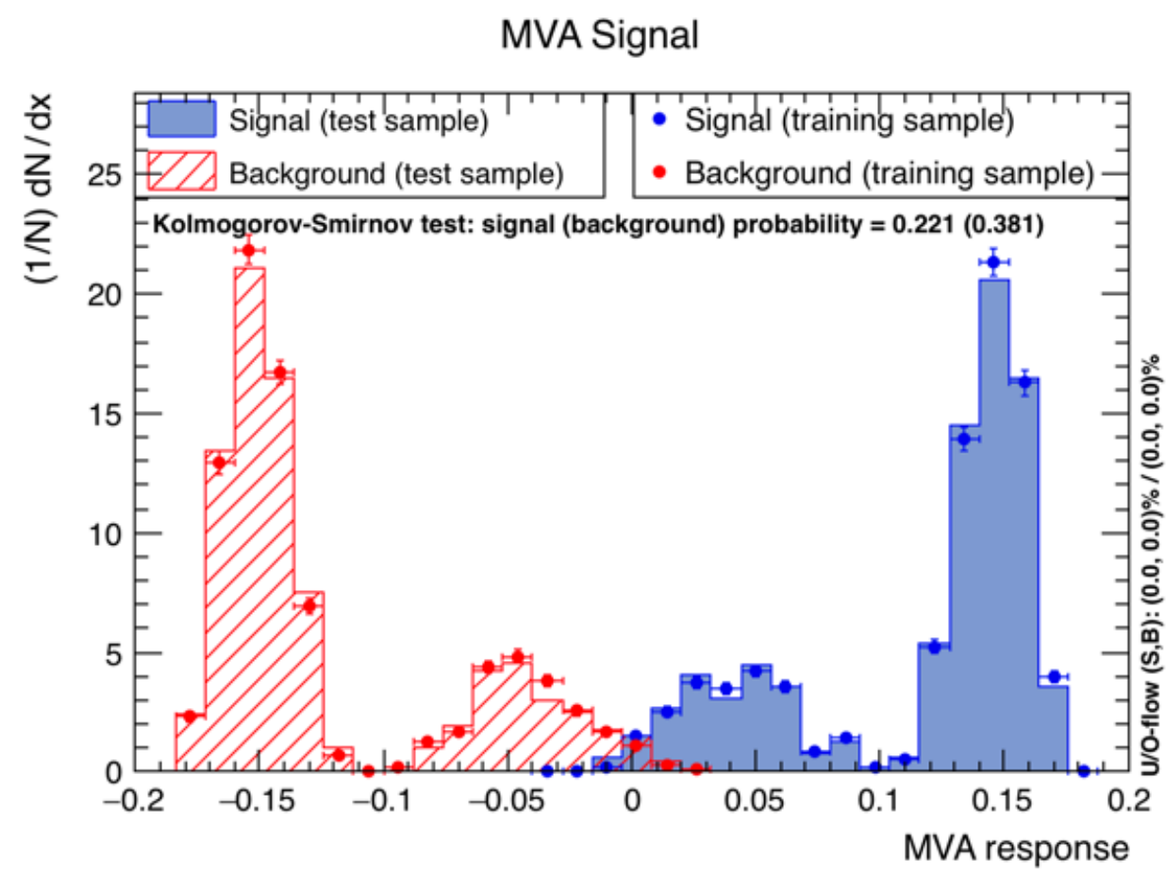
Queen Mary University of London

- 4-fold cross validation on checkerboard - BDT

4-fold Best



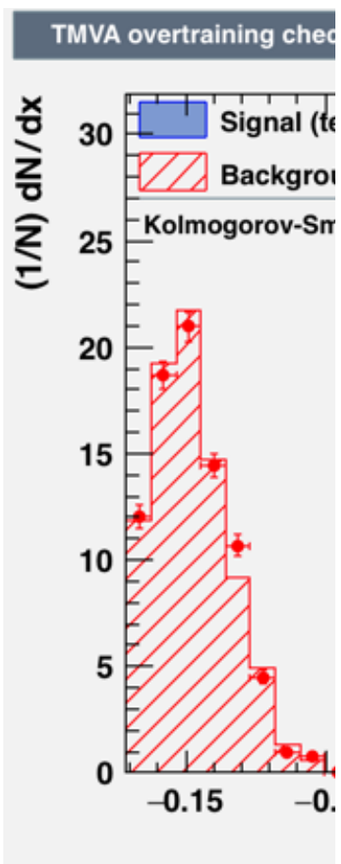
4-fold Average

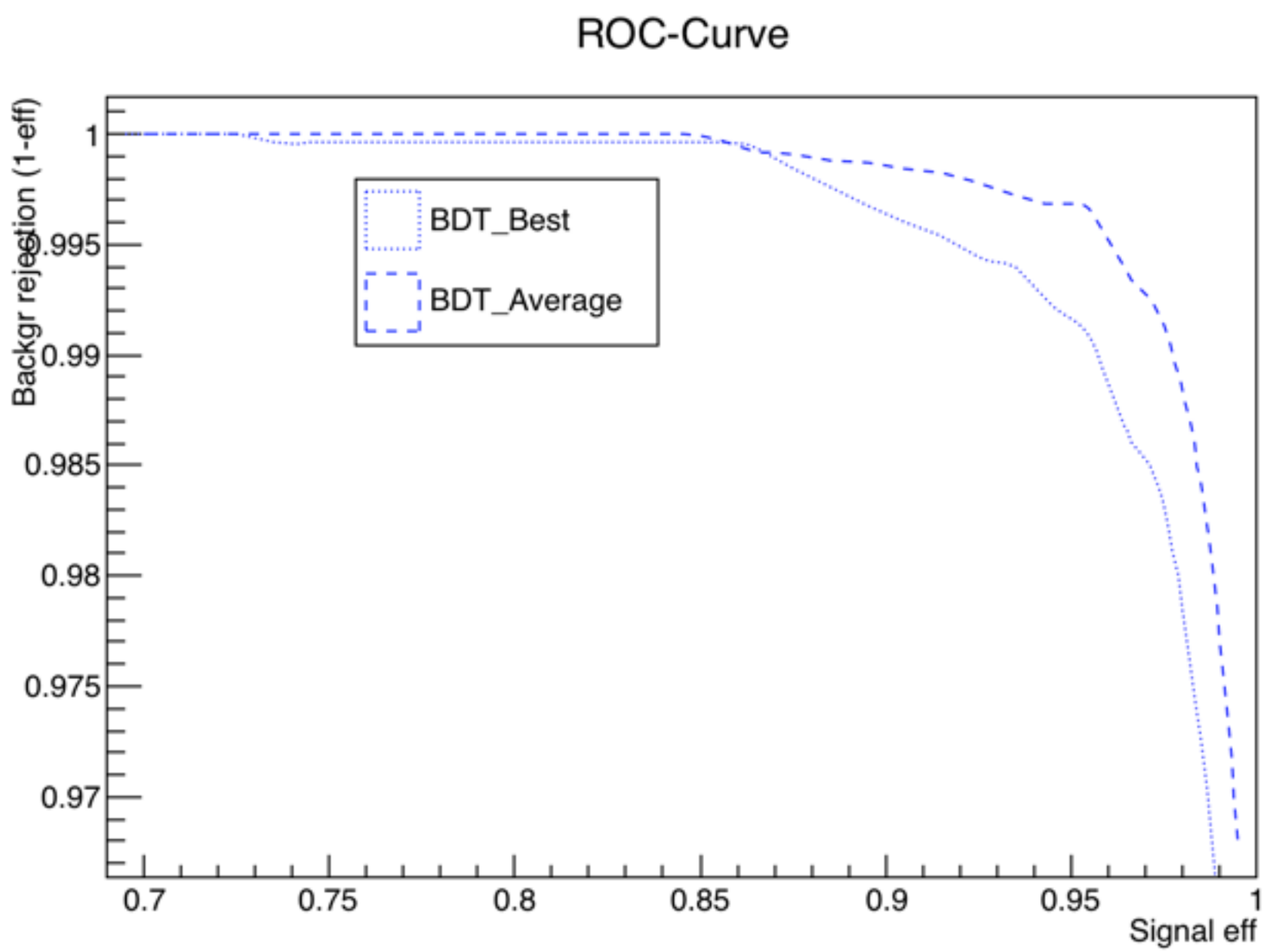


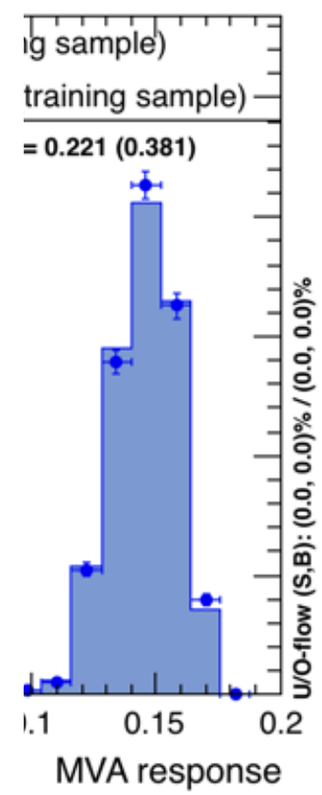
Machine Learning
Cross Validation with TMVA

Queen Mary University of London

- 4-fold cross validation on checkerboard - BDT



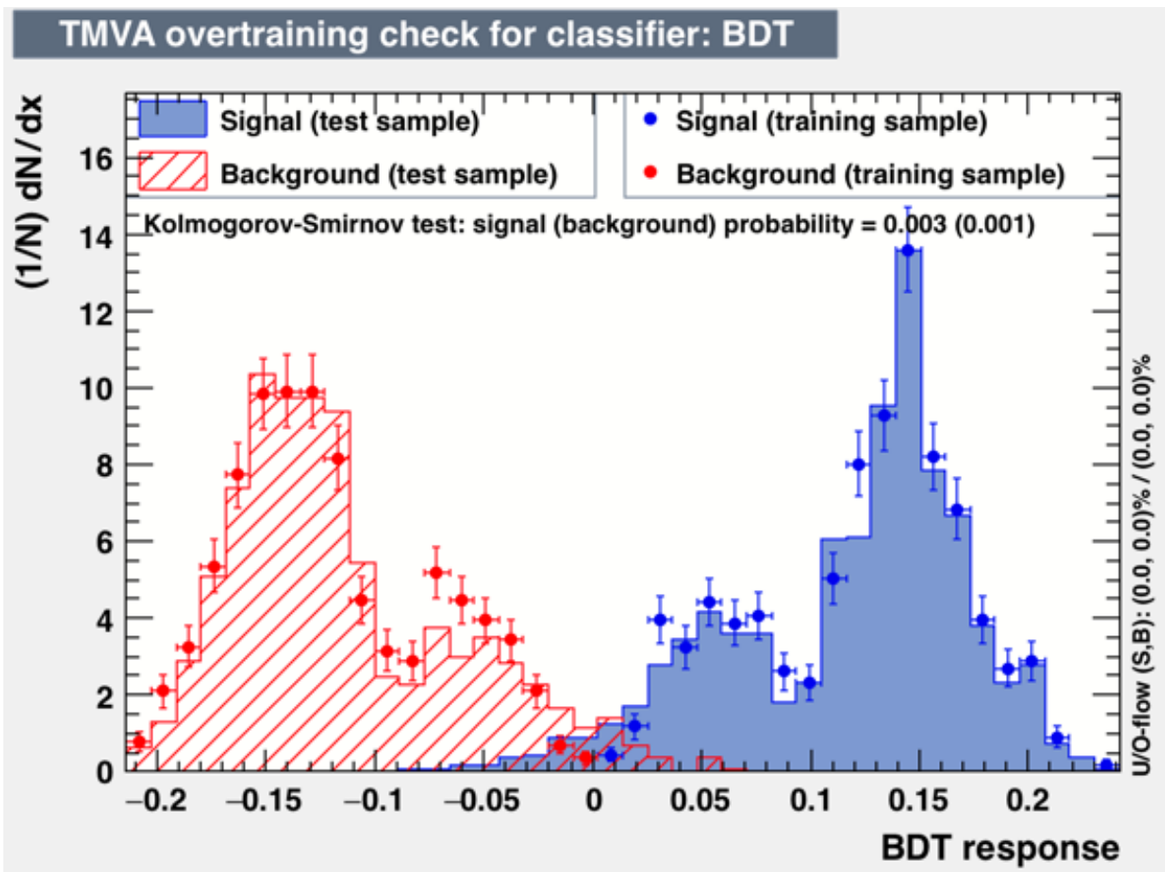




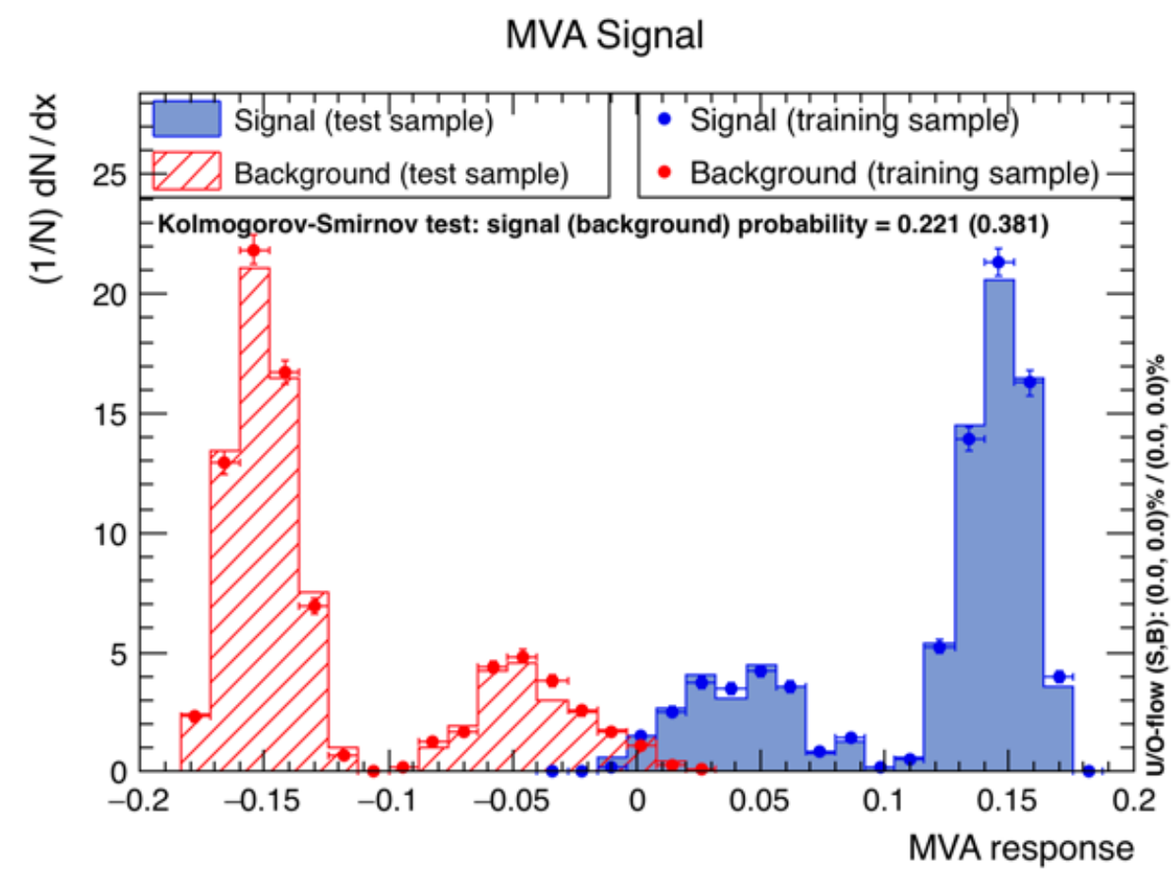
Machine Learning
Cross Validation with TMVA

- 4-fold cross validation on checkerboard - BDT

Hold-out



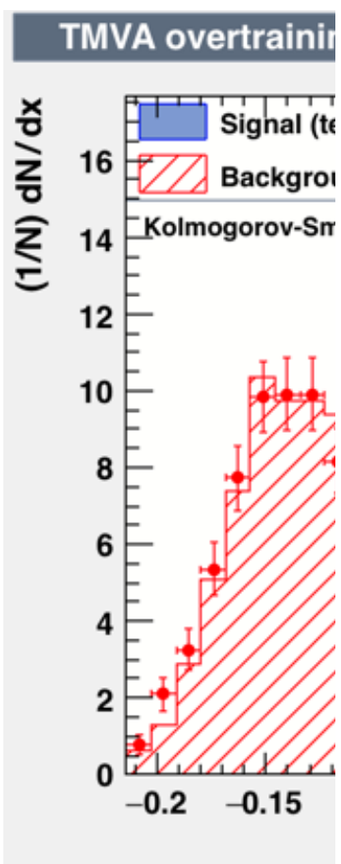
4-fold Average

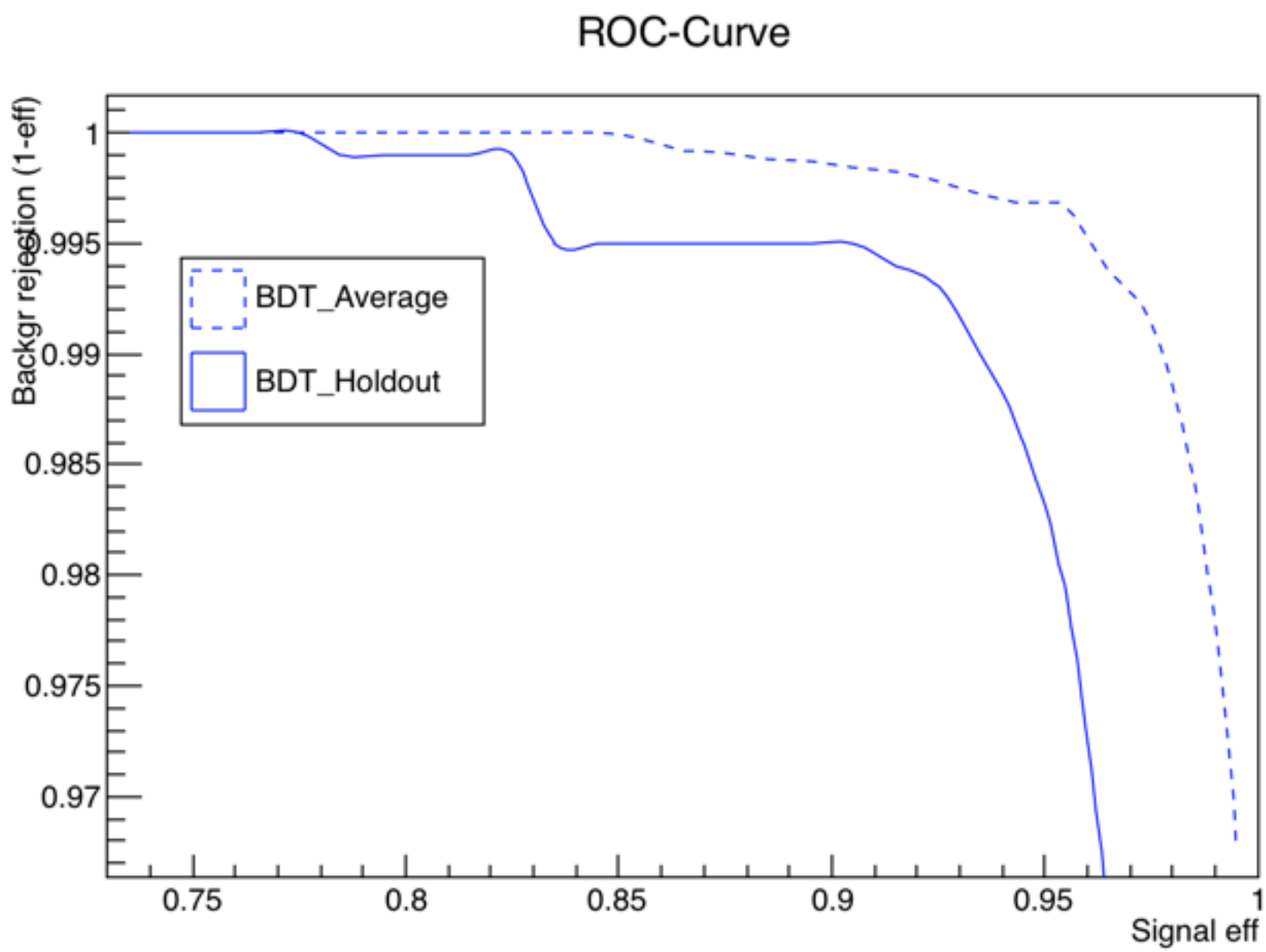


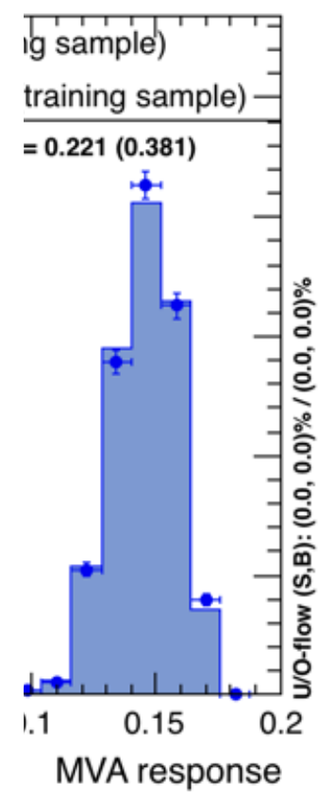
Machine Learning
Cross Validation with TMVA

Queen Mary University of London

- 4-fold cross validation on checkerboard - BDT



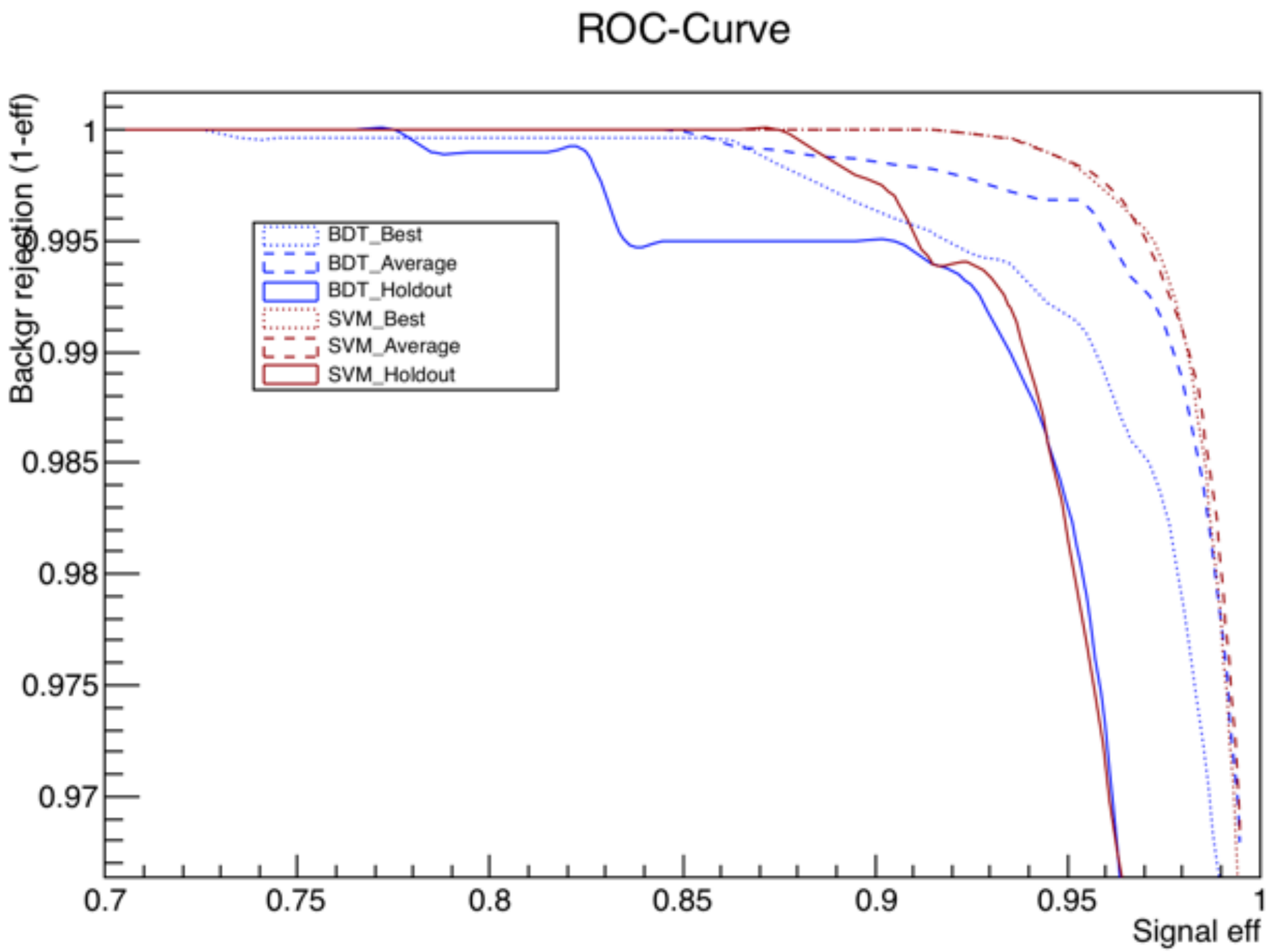




Machine Learning
Cross Validation with TMVA

Queen Mary University of London

- ROC curves for all trainings



Queen Mary
University of London

Summary and Moving Forward

- Cross validation is used in the machine learning community to improve performance and robustness against overfitting/overtraining.
- *k*-fold cross validation allows for the whole dataset to be used for training and testing of MVA classifiers.
- The optimal number of folds needs to be determined for each problem.
- A multistage training/validation/testing process have been detailed.
- Cross validation tool (soon) available as a macro in TMVA to try out and give us feedback/suggestions/improvements!
- Looking at comparison between training and test MVA distributions:
  - Want to quantify similarity.
  - Currently TMVA has binned KS test but this has flaws.
  - Considering other methods.

- Details of this talk can be found in ATL-COM-PHYS-2015-1084.

- Implementation of extensions to SVMs in TMVA detailed in:
https://indico.cern.ch/event/444465/contribution/8/attachments/1152580/1655222/2015-Sept-SVMs.pdf
and will be rolled back into TMVA in the near future.

Thanks for listening!