# Using Machine Learning To Design A Robust Computerized Adaptive Test for Post Traumatic Stress Disorder Diagnosis

Ishanu Chattopadhyay (ishanu@uchicago.edu)
Robert Gibbons (ishanu@uchicago.edu)

## Data Set

TABLE I
Data Set Dimensions

| | |
|---|---|
| No. of samples | 304 |
| No. of items (features) | 211 |
| No. of categories (target labels) | 3 |

The dataset consists of 211 items to which 304 subjects responded. The responses are integer valued in the range $[0, 5]$. Each respondent is either has a positive or a negative PTSD diagnosis.

## Problem Description

Given the dataset as described above, we aim to infer a model that predicts the diagnosis given the responses to a model-directed choice of subset from the item bank. Our model has the following a priori constraint:

- At most 6 items can be used for a single subject

under which we aim to maximize performance measured by standard metrics such as the area under the receiver-operating characteristics curve (AUC).

Additionally, we require that our approach has the ability to generate a plurality of distinct tests of comparable performance, *i.e.*, two subjects taking the test are not necessarily given the same items to respond to.

## Solution: Extremely Randomized Trees

A search of the space of possible classification algorithms indicated that the *extra-trees algorithm* performs best, *i.e.* maximizes AUC under the constraint described above, while allowing for the generation of hundreds of distinct test sets. The Extra-Trees method (standing for extremely randomized trees) was proposed in,[1] with the objective of further randomizing tree building in the context of numerical input features, where the choice of the optimal cut-point is responsible for a large proportion of the variance of the induced tree.

With respect to random forests, the method drops the idea of using bootstrap copies of the learning sample, and instead of trying to find an optimal cut-point for each one of the K randomly chosen features at each node, it selects a cut-point at random. This idea is rather productive in the context of many problems characterized by a large number of numerical features varying more or less continuously: it leads often to increased accuracy thanks to its smoothing and at the same time significantly reduces computational burdens linked to the determination of optimal cut-points in standard trees and in random forests. From a statistical point of view, dropping the bootstrapping idea leads to an advantage in terms of bias,

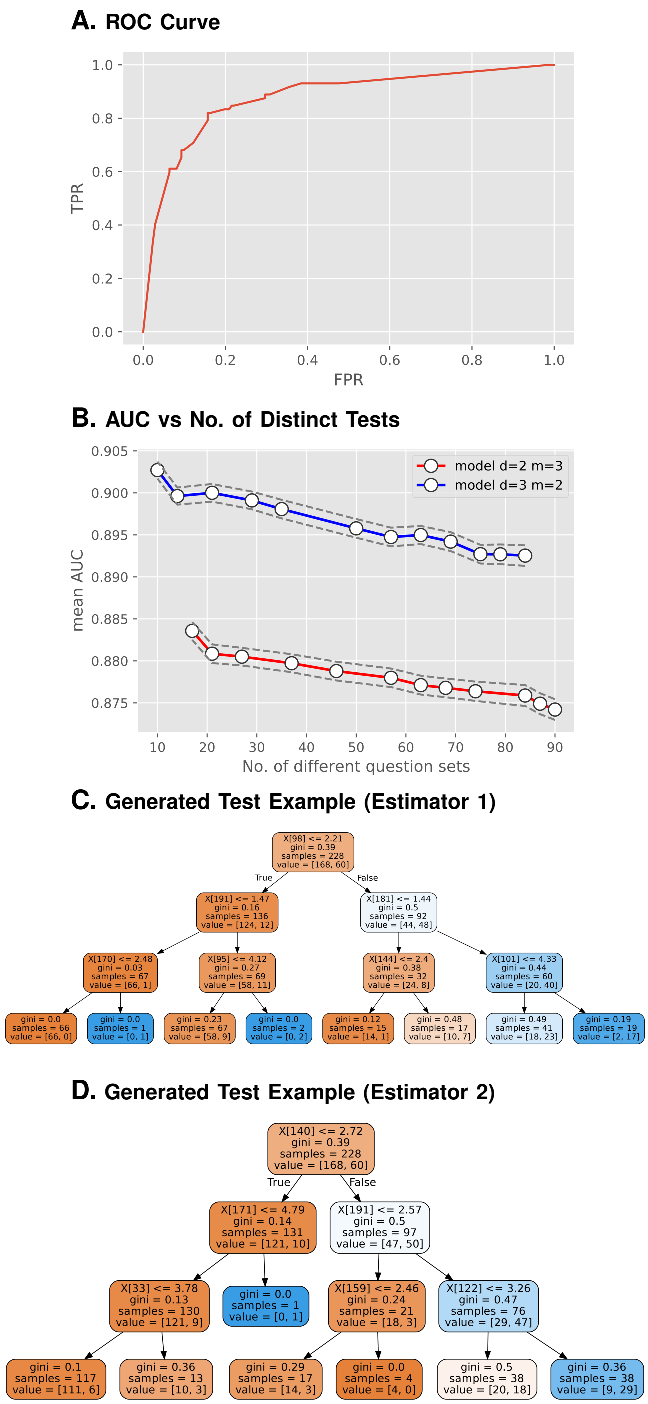


Fig. 1. Plate A shows a representative ROC curve obtained during training with random test-train split (30/70 split). Plate B shows the variation of the median performance against the number of such models obtained, which reflects the number of distinct test sets that can be generated. Plate C and D illustrate a single test, with two estimators, each being a decision tree of depth 3, implying that the maximum number of items presented: 6.

whereas the cut-point randomization has often an excellent variance reduction effect. This method has yielded state-of-

TABLE II
INFERRED MODEL PERFORMANCE CHARACTERISTICS

| | |
|---|---|
| No. of Estimators | 2 |
| Depth of each estimator | 3 |
| Items presented | 6 |
| No. of tests | 240 |
| No. of items used | 189 |
| Median AUC | 0.885 |

the-art results in several high-dimensional complex problems. From a functional point of view, the Extra-Tree method produces piece-wise multilinear approximations, rather than the piece-wise constant ones of random forests.

## RESULTS & DISCUSSION

The constraint on the number of items to be presented implies that we have the following possibilities for the number of estimators (m) and the depth of each estimator (d):

- 2 estimators, each of depth 3
- 3 estimators, each of depth 2
- 6 estimators, each of depth 1
- 1 estimator of depth 6

It turns out the that first scenario $(m = 2, d = 3)$ results in the best performance (See Fig 1, plate B). The performance characteristics of the final inferred model set is shown in Table II.

Importantly, our approach is both randomized and adaptive. The solution first randomly selects a model from the set of optimized model set, where each such model consists of 2 decision trees, each of depth 3. Then the test proceeds adaptively down the decision paths of these trees, as responses to individual presented items are entered by the test-taker.

## SOFTWARE

All software sources are available at the Github Repository link: https://github.com/zeroknowledgediscovery/zcad

## REFERENCES

[1] P. GEURTS, D. ERNST, AND L. WEHENKEL, *Extremely randomized trees*, Mach. Learn., 63 (2006), pp. 3–42.

TABLE III
EXAMPLE CONSOLE RUNS (AFTER 6 RESPONSES, CLASS PROBABILITIES ARE RETURNED.)

```
./CAD.py
 Response to item 184: 1
 Response to item 170: 2
 Response to item 58: 2
 Response to item 67: 3
 Response to item 81: 5
 Response to item 149: 4
[[0.4140625 0.5859375]]

./CAD.py
 Response to item 68: 4
 Response to item 184: 3
 Response to item 100: 5
 Response to item 41: 2
 Response to item 180: 4
 Response to item 119: 4
[[0.36842105 0.63157895]]

./CAD.py
 Response to item 134: 1
 Response to item 164: 3
 Response to item 157: 5
 Response to item 184: 2
 Response to item 81: 4
[[0.26666667 0.73333333]]

./CAD.py
 Response to item 134: 1
 Response to item 192: 2
 Response to item 127: 3
 Response to item 70: 4
 Response to item 135: 5
 Response to item 110: 3
[[0.690625 0.309375]]
```