Weighted Classification Cascades for Optimizing Discovery Significance

Lester Mackey[†]

Collaborators: Jordan Bryan[†] and Man Yue Mo

[†]Stanford University

December 13, 2014

Hypothesis Testing in High-Energy Physics

Goal: Given a collection of **events** (high-energy particle collisions) and a definition of "interesting" (e.g., Higgs boson produced), detect whether any interesting events occurred

- Interesting events = signal events
- Other events (e.g., no Higgs produced) = background events

Why? To test predictions of physical models

- Standard Model of physics predicts existence of elementary particles and various modes of particle decay
  - **Claim:** Higgs bosons exist and often decay into tau particles
- To substantiate claim experimentally, must distinguish
  - Higgs to tau tau decay events (signal events)
  - Other events with similar characteristics (background events)

Hypothesis Testing in High-Energy Physics

Goal: Given a collection of **events** (high-energy particle collisions), test whether any signal events occurred

How?

- Event represented as features (momenta and energy) of particles produced by collision
  - Ideally: Test based on distributions of signal and background
  - Signal and background event distributions complex and difficult to characterize explicitly: hinders development of analytical test
- Identify relatively signal-rich selection region by training classifier on labeled training data
- Test new dataset for signal by counting events in selection region and computing (approximate) "significance value" or p-value under Poisson likelihood ratio test

Approximate Median Significance (AMS)

How to estimate significance of new event data?

- **Dataset** $\mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ with event feature vectors $x_i \in \mathcal{X}$ and labels $y_i \in \{-1, 1\} = \{$background, signal$\}$
- **Classifier** $g : \mathcal{X} \to \{-1, 1\}$ assigning labels to events $x \in \mathcal{X}$
- **True positive count** $s_{\mathcal{D}}(g) = \sum_{i=1}^{n} \mathbb{I}[g(x_i) = 1, y_i = 1]$
- **False positive count** $b_{\mathcal{D}}(g) = \sum_{i=1}^{n} \mathbb{I}[g(x_i) = 1, y_i = -1]$
- Approximate Median Significance (AMS) (Cowan et al., 2011)

$$\mathrm{AMS}_2(g, \mathcal{D}) = \sqrt{2\left((s_{\mathcal{D}}(g) + b_{\mathcal{D}}(g)) \log\left(\frac{s_{\mathcal{D}}(g) + b_{\mathcal{D}}(g)}{b_{\mathcal{D}}(g)}\right) - s_{\mathcal{D}}(g)\right)}$$

- Approximates $1 - p$-value quantile of Poisson model test statistic
- Measures significance in units of standard deviation or σ's
  - Typically $> 5\sigma$ needed to declare signal discovery significant

Approximate Median Significance (AMS)

Training goal: Select classifier g to maximize AMS_2 on future data

Standard two-stage approach

- Withhold fraction of training events
- **Stage 1:** Train any standard classifier on remaining events
- **Stage 2:** Order held-out events by classifier scores and select new classification threshold to minimize AMS_2 on held-out data
- **Pros:** Requires only standard classification tools; works with any classifier
- **Con:** Stage 2 prone to overfitting, may require hand tuning
- **Con:** Stage 1 ignores AMS_2 objective, optimizes classification error

This talk: A more direct approach to optimizing training AMS_2 that only requires standard classification tools and works with any classifier supporting class weights

Weighted Classification Cascades

Algorithm (Weighted Classification Cascade for Maximizing AMS_2)

- **initialize signal class weight:** $u_0^{\mathrm{SIG}} > 0$
- **for** $t = 1$ **to** T
  - **compute background class weight:** $u_{t-1}^{\mathrm{BAC}} \leftarrow e^{u_{t-1}^{\mathrm{SIG}}} - u_{t-1}^{\mathrm{SIG}} - 1$
  - **train any weighted classifier:**

    $g_t \leftarrow$ approximate minimizer of weighted classification error

    $$b_{\mathcal{D}}(g)\, u_{t-1}^{\mathrm{BAC}} + \tilde{s}_{\mathcal{D}}(g)\, u_{t-1}^{\mathrm{SIG}}$$

    (where $\tilde{s}_{\mathcal{D}}(g) = \sum_{i=1}^{n} \mathbb{I}[y_i = 1] - s_{\mathcal{D}}(g) =$ false negative count)
  - **update signal class weight:** $u_t^{\mathrm{SIG}} \leftarrow \log(s_{\mathcal{D}}(g_t)/b_{\mathcal{D}}(g_t) + 1)$
- **return** g_T

Advantages

- Reduces optimizing AMS_2 to series of classification problems
- Can use any weighted classification procedure
- AMS_2 improves if g_t decreases weighted classification error

Questions: Where does this come from? Why should this work?

The Difficulty of Optimizing AMS

Approximate Median Significance (squared and halved)

$$\frac{1}{2}\mathrm{AMS}_2^2(g, \mathcal{D}) = (s_{\mathcal{D}}(g) + b_{\mathcal{D}}(g)) \log\left(\frac{s_{\mathcal{D}}(g) + b_{\mathcal{D}}(g)}{b_{\mathcal{D}}(g)}\right) - s_{\mathcal{D}}(g)$$

- True positive count $s_{\mathcal{D}}(g) = \sum_{i=1}^n \mathbb{I}[g(x_i) = 1, y_i = 1]$
- False positive count $b_{\mathcal{D}}(g) = \sum_{i=1}^n \mathbb{I}[g(x_i) = 1, y_i = -1]$

$\frac{1}{2}\mathrm{AMS}_2^2$ is

- Combinatorial, as a function of indicator functions
- Non-decomposable across events, due to logarithm
- Convex in $(s_{\mathcal{D}}(g), b_{\mathcal{D}}(g))$, bad for maximization

Linearizing AMS with Convex Duality

Observation:

$$\begin{aligned}\frac{1}{2}\mathrm{AMS}_2^2(g,\mathcal{D}) &= b_{\mathcal{D}}(g) f_2\left(\frac{s_{\mathcal{D}}(g)}{b_{\mathcal{D}}(g)}\right) = b_{\mathcal{D}}(g)\sup_u u\frac{s_{\mathcal{D}}(g)}{b_{\mathcal{D}}(g)} - f_2^*(u) \\ &= \sup_u u\, s_{\mathcal{D}}(g) - f_2^*(u)\, b_{\mathcal{D}}(g) \\ &= -\inf_u u\, \tilde{s}_{\mathcal{D}}(g) + f_2^*(u)\, b_{\mathcal{D}}(g) - u\textstyle\sum_{i=1}^n \mathbb{I}[y_i = 1]\end{aligned}$$

- where $f_2(t) = (1+t)\log(1+t) - t$ is convex
- f_2 admits **variational representation** $f_2(t) = \sup_u ut - f_2^*(u)$ in terms of **convex conjugate** $f_2^*(u) \triangleq \sup_t tu - f_2(t) = e^u - u - 1$
- Since false negative count $\tilde{s}_{\mathcal{D}}(g) = \sum_{i=1}^n \mathbb{I}[y_i = 1] - s_{\mathcal{D}}(g)$

Optimizing AMS with Coordinate Descent

Take-away

$$-\frac{1}{2}\mathrm{AMS}_2^2(g,\mathcal{D}) = \inf_u u\,\tilde{s}_{\mathcal{D}}(g) + (e^u - u - 1)\,b_{\mathcal{D}}(g) - u\sum_{i=1}^n \mathbb{I}[y_i = 1]$$

- Maximizing AMS_2 is equivalent to minimizing weighted error $R_2(g,u,\mathcal{D}) \triangleq u\,\tilde{s}_{\mathcal{D}}(g) + (e^u - u - 1)\,b_{\mathcal{D}}(g) - u\sum_{i=1}^n \mathbb{I}[y_i = 1]$ over classifiers g and signal class weight u jointly

Optimize $R_2(g,u,\mathcal{D})$ with coordinate descent

- Update g_t for fixed u_{t-1}: train weighted classifier
- Update u_t for fixed g_t: closed form, $u = \log(s_{\mathcal{D}}(g_t)/b_{\mathcal{D}}(g_t) + 1)$
- AMS_2 increases whenever a new g_{t+1} achieves smaller weighted classification error with respect to u_t than its predecessor g_t: $-\frac{1}{2}\mathrm{AMS}_2(g_{t+1})^2 \le R_2(g_{t+1},u_t) < R_2(g_t,u_t) = -\frac{1}{2}\mathrm{AMS}_2(g_t)^2$
- **Minorization-maximization** algorithm (like EM)

Optimizing Alternative Significance Measures

Simpler Form of AMS: $\mathrm{AMS}_3(g, \mathcal{D}) = s_{\mathcal{D}}(g)/\sqrt{b_{\mathcal{D}}(g)}$

- Approximates $\mathrm{AMS}_2 = \mathrm{AMS}_3 \times \sqrt{1 + O((s/b)^3)}$ when $s \ll b$
- Amenable to weighted classification cascading
  $\frac{1}{2}\mathrm{AMS}_3^2(g, \mathcal{D}) = b_{\mathcal{D}}(g) f_3\left(\frac{s_{\mathcal{D}}(g)}{b_{\mathcal{D}}(g)}\right)$ for convex $f_3(t) = (1/2)t^2$
- (Can also support uncertainty in b: $b_{\mathcal{D}}(g) \leftarrow b_{\mathcal{D}}(g) + \sigma_b$)

Algorithm (Weighted Classification Cascade for Maximizing AMS_3)

- **for** $t = 1$ **to** T
  - **compute background class weight:** $u_{t-1}^{\mathrm{BAC}} \leftarrow (u^{\mathrm{SIG}})^2/2$
  - **train any weighted classifier:**
    $g_t \leftarrow$ approximate minimizer of weighted classification error

$$b_{\mathcal{D}}(g)\, u_{t-1}^{\mathrm{BAC}} + \tilde{s}_{\mathcal{D}}(g)\, u_{t-1}^{\mathrm{SIG}}$$

  - **update signal class weight:** $u_t^{\mathrm{SIG}} \leftarrow s_{\mathcal{D}}(g_t)/b_{\mathcal{D}}(g_t)$

HiggsML Challenge Case Study

Cascading in the Wild

- So far, recipe for turning classifier into training AMS maximizer
- Must be coupled with effective regularization strategies to ensure adequate test set generalization
- Team `mymo` incorporated two practical variants of cascading into HiggsML challenge solution, placing 31st out of 1800 teams

Cascading Variant 1

- Fit each classifier g_t using XGBoost implementation of gradient tree boosting[1]
- To curb overfitting, computed true and false positive counts on held-out dataset $\mathcal{D}_{\text{val}}$ and updated the class weight parameter u_t^{sig} using $s_{\mathcal{D}_{\text{val}}}(g_t)$ and $b_{\mathcal{D}_{\text{val}}}(g_t)$ in lieu of $s_{\mathcal{D}}(g_t)$ and $b_{\mathcal{D}}(g_t)$

[1] `https://github.com/tqchen/xgboost`

HiggsML Challenge Case Study

Cascading in the Wild

- So far, recipe for turning classifier into training AMS maximizer
- Must be coupled with effective regularization strategies to ensure adequate test set generalization
- Team `mymo` incorporated two practical variants of cascading into HiggsML challenge solution, placing 31st out of 1800 teams

Cascading Variant 2

- Maintained single persistent classifier, the complexity of which grew on each cascade round
- Developed a customized XGBoost classifier that, on cascade round t, introduced a single new decision tree based on the gradient of the round t weighted classification error
- In effect, each classifier g_t was warm-started from the prior round classifier g_{t-1}

HiggsML Challenge Case Study

Cascading in the Wild

- So far, recipe for turning classifier into training AMS maximizer
- Must be coupled with effective regularization strategies to ensure adequate test set generalization
- Team `mymo` incorporated two practical variants of cascading into HiggsML challenge solution, placing 31st out of 1800 teams

Final Solution

- Ensemble of cascade procedures of each variant and several non-cascaded (standard two-stage / hand-tuned) XGBoost, random forest, and neural network models
- Ensemble of all non-cascade models yielded a private leaderboard score of 3.67 (roughly 198th place)
- Each cascade variant alone yielded 3.65
- Incorporating the cascade models into ensemble yielded 3.72594

Beyond the HiggsML Challenge

Next Steps

- More comprehensive, controlled empirical evaluation of cascading
- More extensive exploration of strategies for ensuring good generalization

Thanks!

References I

Cowan, Glen, Cranmer, Kyle, Gross, Eilam, and Vitells, Ofer. Asymptotic formulae for likelihood-based tests of new physics. *The European Physical Journal C-Particles and Fields*, 71(2):1–19, 2011.