Derivative-Free Optimization for Hyper-Parameter Tuning in Machine Learning Problems

Hiva Ghanbari

Jointed work with Prof. Katya Scheinberg

Industrial and Systems Engineering Department

Lehigh University

ICCOPT Conference, Tokyo, Japan

August 2016

Outline

Introduction

AUC Optimization via DFO-TR

Parameter Tuning of Cost-Sensitive LR

Summary

Outline

Introduction

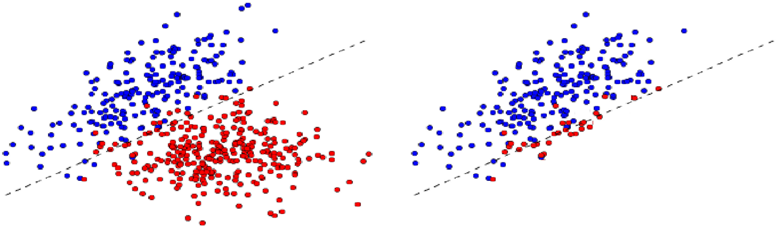
AUC Optimization via DFO-TR

Parameter Tuning of Cost-Sensitive LR

Summary

Learning From Imbalanced Data Sets

- Many real-world machine learning problems are dealing with imbalanced learning data



(a) Balanced data set (b) Imbalanced data set

Learning From Imbalanced Data Sets

Learning From Imbalanced Data Sets

- Rare positive example, as the minority class, but numerous negative ones, as the majority class
- In many applications the minority class is the more interesting and important one
- The rare class has a much higher misclassification cost compared to the majority class

Learning From Imbalanced Data Sets

- Most common methods for dealing with imbalance data sets

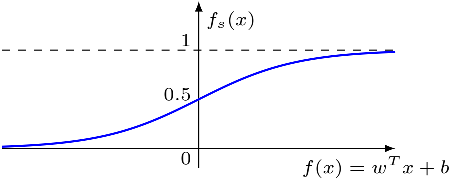
Methods	**Advantages**	**Drawbacks**
Undersampling	Independent of underlying classifier	May remove significant patterns
Oversampling	Independent of underlying classifier Can be easily implemented	Time consuming May lead to over-fitting
Cost sensitive	Minimizes the cost of misclassification	The misclassification costs are unknown

Ranking Quality of Classifier $f(x)$

- Consider linear classifier $f(x) = w^T x + b$, where

$$y_i = \begin{cases} +1, & \text{if } \ f(x_i) > 0, \\ -1, & \text{otherwise.} \end{cases}$$

- Consider function $f_s(x) = \frac{1}{1+e^{-f(x)}}$, where $f(x) = w^T x + b$.

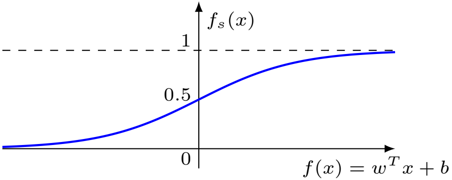


Ranking Quality of Classifier $f(x)$

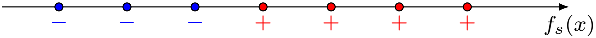
- Consider linear classifier $f(x) = w^T x + b$, where

$$y_i = \begin{cases} +1, & \text{if } \ f(x_i) > 0, \\ -1, & \text{otherwise.} \end{cases}$$

- Consider function $f_s(x) = \frac{1}{1+e^{-f(x)}}$, where $f(x) = w^T x + b$.

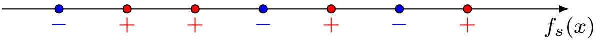


- In perfect classification, all positive examples are ranked higher than the negative ones.



Receiver Operating Characteristic(ROC) Curve

- Sorted outputs based on descending value of $f_s(x)$



- Confusion Matrix of a binary classification problem, for a specific threshold to decide when negative turns into positive example

	+ve (Predicted Class)	**-ve (Predicted Class)**
+ve (Actual Class)	True Positive (TP)	False Negative (FN)
-ve (Actual Class)	False Positive (FP)	True Negative (TN)

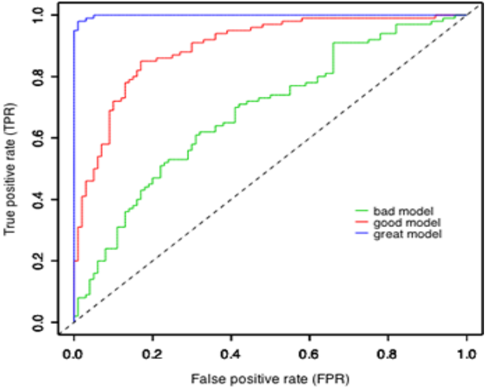
- Various thresholds result in different True Positive Rate $= \frac{TP}{TP+FN}$ and False Positive Rate $= \frac{FP}{FP+TN}$.

Definition 1

ROC curve presents the tradeoff between the true positive rate and the false positive rate, for all possible thresholds.

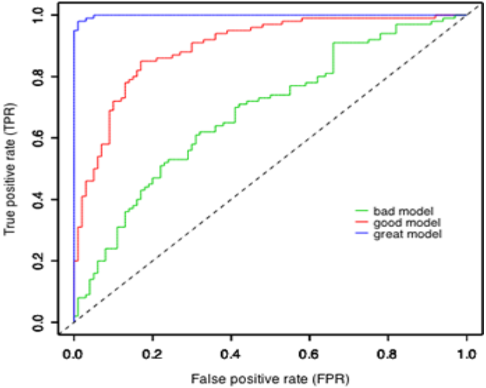
Area Under ROC Curve (AUC)

- How we can compare ROC curves?



Area Under ROC Curve (AUC)

- How we can compare ROC curves? Higher AUC $\Longrightarrow$ Better classifier



Measuring AUC

- How can we measure the value of AUC of a classifier?

Lemma 2

Consider linear classifier $f(x) = w^T x + b$, *which has classified positive set* $\mathcal{S}_+ := \{x_i^+ : i = 1, \ldots, N_+\}$ *from negative set* $\mathcal{S}_- := \{x_j^- : j = 1, \ldots, N_-\}$. *Then, the associated AUC can be obtained by "WMW" result*

$$F_{AUC}(w) = \frac{\sum_{i=1}^{N_+} \sum_{j=1}^{N_-} \mathbb{I}_w[w^T x_i^+ > w^T x_j^-]}{N_+ \cdot N_-}.$$

where

$$\mathbb{I}_w[w^T x_i > w^T x_j] = \begin{cases} +1, & \text{if } \; w^T x_i^+ > w^T x_j^-, \\ 0, & \text{otherwise.} \end{cases}$$

Probabilistic Interpretation of AUC

- We can measure the ranking quality of the classifier $f(x)$ through a probability value.

$$F_{AUC}(w) = P(w^T X_+ > w^T X_-).$$

where X_+ and X_- are randomly sampled from $\mathcal{S}_+$ and $\mathcal{S}_-$.

- If two sets $\mathcal{S}_+$ and $\mathcal{S}_-$ are randomly drawn from distributions $\mathcal{D}_1$ and $\mathcal{D}_2$, then

$$E(F_{AUC}(w)) = P(w^T \hat{X}_+ > w^T \hat{X}_-), \tag{1}$$

where $\hat{X}_+$ and $\hat{X}_-$ are randomly chosen from distributions $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively.

Ranking Loss

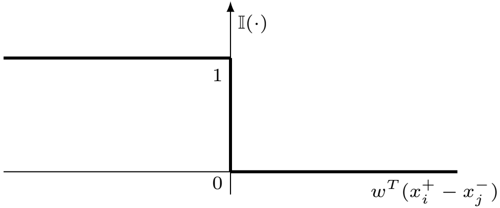
maximizing AUC function $F_{AUC}(w)$ is equivalent to minimizing *ranking loss* $1 - F_{AUC}(w)$

$$1 - F_{AUC}(w) = 1 + \frac{\sum_{i=1}^{N_+} \sum_{j=1}^{N_-} \mathbb{I}[w^T x_i^+ - w^T x_j^- \leq 0]}{N_+ \cdot N_-},$$

where

$$\mathbb{I}[w^T x_i^+ - w^T x_j^- \leq 0] = \begin{cases} +1, & \text{if } \; w^T x_i^+ - w^T x_j^- \leq 0, \\ 0, & \text{otherwise.} \end{cases}$$

as is shown below



Surrogate Convex Losses

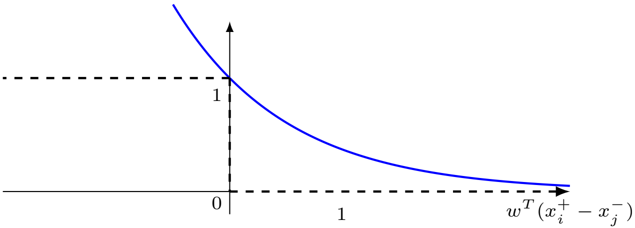
- The main challenge of minimizing ranking loss $1 - F_{AUC}(w)$ is in its non-smoothness and discontinuousness owned from indicator function $\mathbb{I}(\cdot)$.

Surrogate Convex Losses

- The main challenge of minimizing ranking loss $1 - F_{AUC}(w)$ is in its non-smoothness and discontinuousness owned from indicator function $\mathbb{I}(\cdot)$.

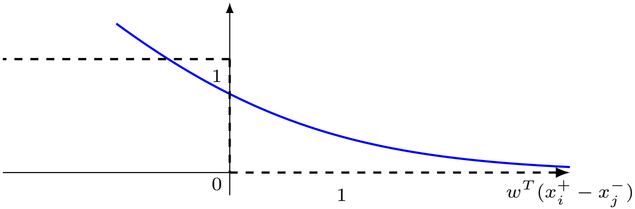
- Exponential Loss:

$$\mathbb{I}_{exp}(w, x_i^+ - x_j^-) = e^{-w^T(x_i^+ - x_j^-)}$$



Surrogate Convex Losses

- The main challenge of minimizing ranking loss $1 - F_{AUC}(w)$ is in its non-smoothness and discontinuousness owned from indicator function $\mathbb{I}(\cdot)$.
- Logistic Loss:

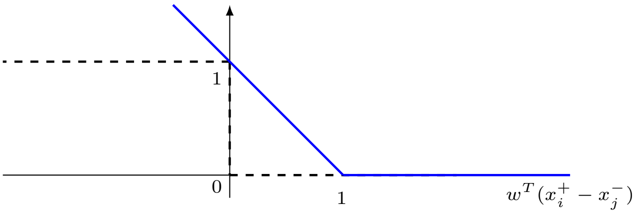


$$\mathbb{I}_{log}(w, x_i^+ - x_j^-) = \log(1 + e^{-w^T(x_i^+ - x_j^-)}),$$

Surrogate Convex Losses

- The main challenge of minimizing ranking loss $1 - F_{AUC}(w)$ is in its non-smoothness and discontinuousness owned from indicator function $\mathbb{I}(.)$.
- Hinge Loss:

$$\mathbb{I}_{hinge}(w, x_i^+ - x_j^-) = \max\{0, 1 - w^T(x_i^+ - x_j^-)\}$$



Outline

Introduction

AUC Optimization via DFO-TR

Parameter Tuning of Cost-Sensitive LR

Summary

Algorithmic Framework of DFO-TR

Algorithm 1 **Trust Region Based Derivative Free Optimization**

0. Initializations.
– Define interpolation set $\mathcal{X}$ and compute corresponding function values $\mathcal{F}_{\mathcal{X}}$.

1. **Build the model.**
– Build $Q_k(x)$ as a quadratic approximation of function f.

2. **Minimize the model within the trust region.**
– Find $\hat{x}_k$ such that $Q_k(\hat{x}_k) := \min_{x \in \mathcal{B}_k} Q_k(x)$, where $\mathcal{B}_k = \{x : \|x - x_k\| \leq \Delta_k\}$.
– Compute function $f(\hat{x}_k)$ and the ratio $\rho_k = \frac{f(x_k) - f(\hat{x}_k)}{Q_k(x_k) - Q_k(\hat{x}_k)}$.

3. **Update the interpolation set.**

4. **Update the trust region radius.**

AUC, a Smooth Function of w in Expectation

Theorem 3

If two $d-$dimensional random vectors $\hat{X}_1$ and $\hat{X}_2$ have a joint multivariate normal distribution, such that

$$\begin{pmatrix} \hat{X}_1 \\ \hat{X}_2 \end{pmatrix} \sim \mathcal{N}(\mu, \Sigma), \quad \textit{where,} \quad \mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \ \textit{and} \ \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix},$$

then the marginal distributions of $\hat{X}_1$ and $\hat{X}_2$ are normal distributions with the following properties

$$\hat{X}_1 \sim \mathcal{N}(\mu_1, \Sigma_{11}), \qquad \hat{X}_2 \sim \mathcal{N}(\mu_2, \Sigma_{22}).$$

AUC, a Smooth Function of w in Expectation

Theorem 4

Consider two random vectors $\hat{X}_1$ and $\hat{X}_2$, then for any vector $w \in \mathbb{R}^d$, we have

$$Z = w^T(\hat{X}_1 - \hat{X}_2) \sim \mathcal{N}(\mu_Z, \sigma_Z^2),$$

where

$$\mu_Z = w^T(\mu_1 - \mu_2),$$
$$\sigma_Z^2 = w^T\left(\Sigma_{11} + \Sigma_{22} - \Sigma_{12} - \Sigma_{21}\right) w.$$

Theorem 5

If two random vectors $\hat{X}_1$ and $\hat{X}_2$, respectively from the positive and the negative classes, have a joint normal distribution, then the expected value of AUC function can be defined as

$$E(F_{AUC}(w)) = \phi\left(\frac{\mu_Z}{\sigma_Z}\right),$$

where ϕ is the cumulative function of the standard normal distribution, so that $\phi(x) = e^{-\frac{1}{2}x^2}/2\pi$, for $\forall x \in \mathbb{R}$.

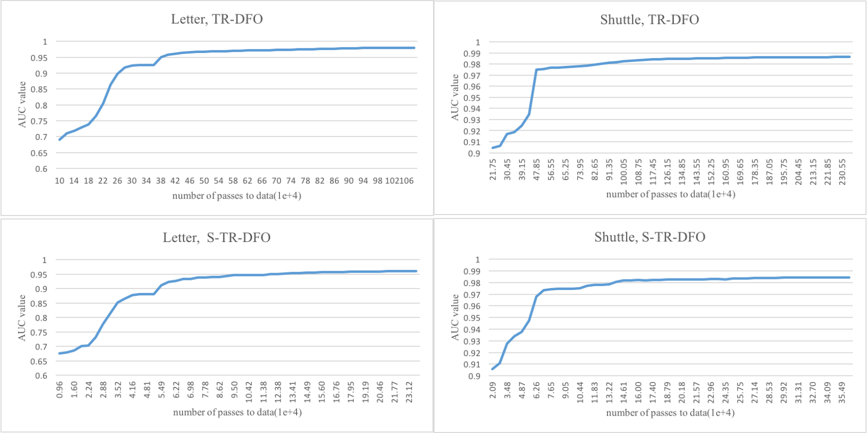
Computational Result (DFO-TR vs. Hinge)

- Performance of algorithms based on the average value of AUC and number of function evaluations,

Algorithm	**sonar**$(d = 60, N = 208)$ AUC	fevals	**fourclass**$(d = 2, N = 862)$ AUC	fevals
Hinge	0.849057 ± 0.003061	274	0.836226± 0.000923	480
DFO-TR	0.840323± 0.002453	254	0.836311± 0.000921	254
Algorithm	**svmguide1**$(d = 4, N = 3089)$ AUC	fevals	**magic04**$(d = 10, N = 4020)$ AUC	fevals
Hinge	0.989319 ± 0.000008	334	0.843085 ± 0.000208	417
DFO-TR	0.989132± 0.000007	254	0.843511±0.000213	254
Algorithm	**svmguide3**$(d = 22, N = 1243)$ AUC	fevals	**shuttle**$(d = 9, N = 3500)$ AUC	fevals
Hinge	0.793116±0.001284	368	0.988625±0.000021	266
DFO-TR	0.775246± 0.002083	254	0.987531±0.000035	254
Algorithm	**segment**$(d = 19, N = 2310)$ AUC	fevals	**ijcnn1**$(d = 22, N = 4691)$ AUC	fevals
Hinge	0.993134 ± 0.000023	753	0.930685±0.000204	413
DFO-TR	0.99567±0.00071	254	0.910897± 0.000264	254
Algorithm	**letter**$(d = 16, N = 5000)$ AUC	fevals	**poker**$(d = 10, N = 5010)$ AUC	fevals
Hinge	0.986699±0.000037	517	0.519942±0.001549	553
DFO-TR	0.985119±0.000042	254	0.520517±0.001618	254

Computational Result (Stochastic DFO-TR)

- Stochastic vs Deterministic DFO-TR



Outline

Introduction

AUC Optimization via DFO-TR

Parameter Tuning of Cost-Sensitive LR

Summary

Cost-Sensitive Logistic Regression Problem

- Biasing the classifier toward the minority class

$$F(w) := \frac{1}{N}\sum_{i=1}^{N} C(y_i)\log(1+\exp(-y_i \cdot w^T x_i)) + \frac{\lambda}{2}\|w\|_2,$$

where

$$C(y_i) = \begin{cases} C_+, & \text{if } \; y_i > 0, \\ C_-, & \text{otherwise.} \end{cases}$$

Cost-Sensitive Logistic Regression Problem

- Biasing the classifier toward the minority class

$$F(w) := \frac{1}{N}\sum_{i=1}^{N} C(y_i)\log(1+\exp(-y_i\cdot w^T x_i)) + \frac{\lambda}{2}\|w\|_2,$$

where

$$C(y_i) = \begin{cases} C_+, & \text{if } \; y_i > 0, \\ C_-, & \text{otherwise.} \end{cases}$$

GOAL $\cdots$

- Find the best value of costs C_+ and C_- and regularization parameter λ, to achieve a linear classifier $f(x) = w^T x + b$ with maximum value of AUC.

AUC as a Function of $\{C_+, C_-, \lambda\}$

- AUC is a function of w

$$F_{AUC}(w) = \frac{\sum_{i=1}^{N_+} \sum_{j=1}^{N_-} \mathbb{I}_w[w^T x_i > w^T x_j]}{N_+ \cdot N_-}.$$

- The vector w is a function of the parameters $\{C_+, C_-, \lambda\}$, so that

$$F(w) := \frac{1}{N} \sum_{i=1}^{N} C(y_i) \log(1 + \exp(-y_i \cdot w^T x_i)) + \frac{\lambda}{2} \|w\|_2,$$

AUC as a Function of $\{C_+, C_-, \lambda\}$

- AUC is a function of w

$$F_{AUC}(w) = \frac{\sum_{i=1}^{N_+} \sum_{j=1}^{N_-} \mathbb{I}_w[w^T x_i > w^T x_j]}{N_+ \cdot N_-}.$$

- The vector w is a function of the parameters $\{C_+, C_-, \lambda\}$, so that

$$F(w) := \frac{1}{N} \sum_{i=1}^{N} C(y_i) \log(1 + \exp(-y_i \cdot w^T x_i)) + \frac{\lambda}{2} \|w\|_2,$$

- $F_{AUC}(w)$ is a non-smooth function of the parameters $\{C_+, C_-, \lambda\}$.

w a Smooth Function of $\{C_+, C_-, \lambda\}$

Theorem 6

In the cost sensitive logistic regression, the vector w is a smooth function of non-zero parameters C_+ ,C_-, and λ if matrix M and added matrix $M|v$ have the same rank (one special case which satisfies this condition is when the matrix M is a full rank matrix)

$$\begin{aligned} M &= \sum_{i=1}^{N_+} \frac{\exp(y_i \cdot w^T x_i)}{(1+\exp(y_i \cdot w^T x_i))^2}(x_i x_i^T), \\ v &= \frac{1}{C_+} \sum_{i=1}^{N_+} \frac{y_i x_i}{1+\exp(y_i \cdot w^T x_i)}. \end{aligned} \tag{2}$$

Similar condition is required when M and v are constructed based on the negative class.

Computational Result

In this section we compare the following algorithms,

- **C**ost-**S**ensitive **L**ogistic **R**egression based on **C**lass **D**istribution **R**atio
- **C**ost-**S**ensitive **L**ogistic **R**egression based on **O**ptimized **R**atio

Algorithm	**sonar**	**fourclass**	**svmguide1**	**magic04**
CSLR-CDR	0.837656	0.835398	0.981900	0.841500
CSLR-OR	0.84183	0.835387	0.988035	0.843396
Algorithm	**diabetes**	**german**	**a9a**	**svmguide3**
CSLR-CDR	0.827639	0.788780	0.902774	0.774338
CSLR-OR	0.822690	0.784636	0.902868	0.797397
Algorithm	**connect-4**	**shuttle**	**HAPT**	**segment**
CSLR-CDR	0.899115	0.988924	0.999992	0.999791
CSLR-OR	0.898861	0.991161	0.999994	0.999841
Algorithm	**mnist**	**ijcnn1**	**satimage**	**vowel**
CSLR-CDR	0.994864	0.934975	0.761618	0.978475
CSLR-OR	0.995341	0.934986	0.760733	0.975405
Algorithm	**poker**	**letter**	**w1a**	**w8a**
CSLR-CDR	0.502508	0.987558	0.960954	0.960331
CSLR-OR	0.521330	0.988293	0.962368	0.961645

performance of algorithms based on average value of AUC

Outline

Introduction

AUC Optimization via DFO-TR

Parameter Tuning of Cost-Sensitive LR

Summary

Summary and Future Study

- The main challenge of optimizing AUC is in its non-smoothness and discontinuousness
- Directly Optimizing AUC function via trust region based derivative free optimization
- Tuning parameters of cost-sensitive logistic regression to obtain a classifier with maximum AUC value

References

- A. R. Conn, K. Scheinberg, and L. N.Vicente, "Introduction To Derivative-Free Optimization", *SIAM J*, 2009.
- M. Menickelly R. Chen and K. Scheinberg, "Stochastic optimization using a trust-region method and random models",
- C. X. Ling, J. Huang, and H. Zhang, "AUC: a Statistically Consistent and more Discriminating Measure than Accuracy",
- J.A. Hanley, B.J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve", *Radiology*, 143:29?36, 1982.