# Directly and Efficiently Optimizing Prediction Error and AUC of Linear Classifiers

Hiva Ghanbari

Joint work with
Prof. Katya Scheinberg

Industrial and Systems Engineering Department

LEHIGH UNIVERSITY

US & Mexico Workshop on
Optimization and its Applications

Huatulco, Mexico

January 2018

# Outline

# Outline

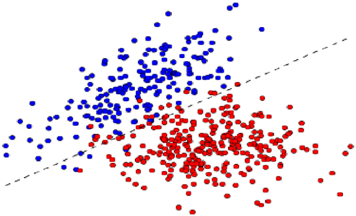
Introduction

Directly Optimizing Prediction Error

Directly Optimizing AUC

Numerical Analysis

Summary



- Given a finite sample data set $\mathcal{S}$ of $n$ (input, label) pairs, e.g.,

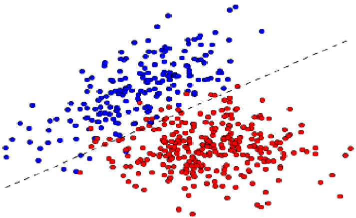
$$\mathcal{S} := \{(x_i, y_i) \ : \ i = 1, \cdots, n\}, \ \textit{where} \ \ x_i \in \mathbb{R}^d \ \textit{and} \ \ y_i \in \{+1, -1\}.$$

- We are interested in Binary Classification Problem in supervised learning
- Binary Classification Problem $\Rightarrow$ Discrete valued output $+1$ or $-1$
- We are interested in linear classifier (predictor) $f(x; w) = w^T x$ so that

$$f : \mathcal{X} \to \mathcal{Y},$$

  where $\mathcal{X}$ denote the space of input values and $\mathcal{Y}$ the space of output values.

# Supervised Learning Problem



**How good is this classifier?**

- Prediction Error
- Area Under ROC Curve (AUC)

- Given a finite sample data set $\mathcal{S}$ of $n$ (input, label) pairs, e.g.,

$$\mathcal{S} := \{(x_i, y_i)\ :\ i = 1, \cdots, n\},\ \textit{where}\ \ x_i \in \mathbb{R}^d\ \textit{and}\ \ y_i \in \{+1, -1\}.$$

- We are interested in Binary Classification Problem in supervised learning
- Binary Classification Problem $\Rightarrow$ Discrete valued output $+1$ or $-1$
- We are interested in linear classifier (predictor) $f(x; w) = w^T x$ so that

$$f : \mathcal{X} \rightarrow \mathcal{Y},$$

  where $\mathcal{X}$ denote the space of input values and $\mathcal{Y}$ the space of output values.

# Outline

# Expected Risk (Prediction Error)

- In $\mathcal{S}$, each $(x_i, y_i)$ is an i.i.d. observation of the random variables $(X, Y)$.
- $(X, Y)$ has an unknown joint probability distribution $P_{X,Y}(x, y)$ over $\mathcal{X}$ and $\mathcal{Y}$.
- The *expected risk* associated with a linear classifier $f(x; w) = w^T x$ for zero-one loss function is defined as

$$\begin{aligned} R_{0-1}(f) &= \mathbb{E}_{\mathcal{X},\mathcal{Y}}\left[\ell_{0-1}\left(f(X;w), Y\right)\right] \\ &= \int_{\mathcal{X}} \int_{\mathcal{Y}} P_{X,Y}(x,y)\ell_{0-1}\left(f(x;w), y\right) dydx \end{aligned}$$

where

$$\ell_{0-1}\left(f(x;w), y\right) = \begin{cases} +1 & \text{if } \; y \cdot f(x;w) < 0, \\ 0 & \text{if } \; y \cdot f(x;w) \geq 0. \end{cases}$$

- The joint probability distribution $P_{X,Y}(x, y)$ is unknown
- The *empirical risk* of the linear classifier $f(x; w)$ for zero-one loss function over the finite training set $\mathcal{S}$ is of the interest, e.g.,

$$R_{0-1}(f; \mathcal{S}) = \frac{1}{n} \sum_{i=1}^{n} \ell_{0-1}(f(x_i; w), y_i).$$

# Empirical Risk Minimization

- The joint probability distribution $P_{X,Y}(x,y)$ is unknown
- The *empirical risk* of the linear classifier $f(x;w)$ for zero-one loss function over the finite training set $\mathcal{S}$ is of the interest, e.g.,

$$R_{0-1}(f;\mathcal{S}) = \frac{1}{n}\sum_{i=1}^{n} \ell_{0-1}\left(f(x_i;w), y_i\right).$$

- Utilizing the logistic regression loss function instead of 0-1 loss function, results

$$R_{log}(f;\mathcal{S}) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1+\exp(-y_i \cdot f(x_i;w))\right),$$

- Practically

$$\min_{w\in\mathbb{R}^d}\left\{F_{log}(w) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1+\exp(-y_i \cdot f(x_i;w))\right) + \lambda\|w\|^2\right\}.$$

# Alternative Interpretation of the Prediction Error

- We can interpret prediction error as a probability value:

$$
\begin{aligned}
F_{error}(w) &= R_{0-1}(f) \\
&= \mathbb{E}_{\mathcal{X},\mathcal{Y}}\left[\ell_{0-1}(f(X;w),Y)\right] \\
&= P(Y \cdot w^T X < 0).
\end{aligned}
$$

- We can interpret prediction error as a probability value:

$$
\begin{aligned}
F_{error}(w) &= R_{0-1}(f) \\
&= \mathbb{E}_{\mathcal{X},\mathcal{Y}}\left[\ell_{0-1}\left(f(X;w),Y\right)\right] \\
&= P(Y \cdot w^T X < 0).
\end{aligned}
$$

If the true values of the prior probabilities $P(Y = +1)$ and $P(Y = -1)$ are known or obtainable from a trivial calculation, then

**Lemma 1**

*Expected risk can be interpreted in terms of the probability value, so that*

$$
\begin{aligned}
F_{error}(w) &= P(Y \cdot w^T X < 0) \\
&= P\left(Z^+ \leq 0\right) P\left(Y = +1\right) + \left(1 - P\left(Z^- \leq 0\right)\right) P\left(Y = -1\right),
\end{aligned}
$$

*where*

$$Z^+ = w^T X^+, \quad \textit{and} \quad Z^- = w^T X^-,$$

*for $X^+$ and $X^-$ as random variables from positive and negative classes, respectively.*

# Data with Any Arbitrary Distribution

- Suppose $(X_1, \cdots, X_n)$ is a multivariate random variable.
- For a given mapping function $g(\cdot)$ we are interested in the c.d.f of

$$Z = g\left(X_1, \cdots, X_n\right).$$

- If we define a region in space $\{\mathcal{X}_1 \times \cdots \times \mathcal{X}_n\}$ such that $g(x_1, \cdots, x_n) \leq z$, then we have

$$\begin{aligned} F_Z(z) &= P(Z \leq z) \\ &= P(g(X) \leq z) \\ &= P\left(\{x_1 \in \mathcal{X}_1, \cdots, x_n \in \mathcal{X}_n : g(x_1, \cdots, x_n) \leq z\}\right) \\ &= \int_{\{x_1 \in \mathcal{X}_1, \cdots, x_n \in \mathcal{X}_n : g(x_1, \cdots, x_n) \leq z\}} \cdots \int f_{X_1, \cdots, X_n}(x_1, \cdots, x_n) dx_1 \cdots dx_n. \end{aligned}$$

# Data with Normal Distribution

Assume

$$X^+ \sim \mathcal{N}\left(\mu^+, \Sigma^+\right) \quad \text{and} \quad X^- \sim \mathcal{N}\left(\mu^-, \Sigma^-\right).$$

# Data with Normal Distribution

Assume

$$X^+ \sim \mathcal{N}\left(\mu^+, \Sigma^+\right) \quad \text{and} \quad X^- \sim \mathcal{N}\left(\mu^-, \Sigma^-\right).$$

**Why Normal?**

- The family of multivariate Normal distributions is closed under linear transformations.

**Theorem 2 (Tong (1990))**

*If $X \sim \mathcal{N}(\mu, \Sigma)$ and $Z = CX + b$, where $C$ is any given $m \times n$ real matrix and $b$ is any $m \times 1$ real vector, then $Z \sim \mathcal{N}\left(C\mu + b, C\Sigma C^T\right)$.*

- Normal Distribution has a smooth c.d.f.

**Theorem 3**

*Suppose that*

$$X^+ \sim \mathcal{N}\left(\mu^+, \Sigma^+\right) \quad \textit{and} \quad X^- \sim \mathcal{N}\left(\mu^-, \Sigma^-\right).$$

*Then,*

$$F_{error}(w) = P(Y=+1)\left(1 - \phi\left(\mu_{Z^+}/\sigma_{Z^+}\right)\right) + P(Y=-1)\phi\left(\mu_{Z^-}/\sigma_{Z^-}\right),$$

*where*

$$\mu_{Z^+} = w^T\mu^+, \quad \sigma_{Z^+} = \sqrt{w^T\Sigma^+ w}, \quad \textit{and}$$
$$\mu_{Z^-} = w^T\mu^-, \quad \sigma_{Z^-} = \sqrt{w^T\Sigma^- w},$$

*in which $\phi$ is the c.d.f of the standard normal distribution, e.g.,*

$$\phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}t^2)dt, \quad \textit{for} \quad \forall x \in \mathbb{R}.$$

**Theorem 3**

*Suppose that*

$$X^+ \sim \mathcal{N}\left(\mu^+, \Sigma^+\right) \quad \textit{and} \quad X^- \sim \mathcal{N}\left(\mu^-, \Sigma^-\right).$$

*Then,*

$$F_{error}(w) = P(Y = +1)\left(1 - \phi\left(\mu_{Z^+}/\sigma_{Z^+}\right)\right) + P(Y = -1)\phi\left(\mu_{Z^-}/\sigma_{Z^-}\right),$$

*where*

$$\mu_{Z^+} = w^T\mu^+, \quad \sigma_{Z^+} = \sqrt{w^T\Sigma^+ w}, \quad \textit{and}$$
$$\mu_{Z^-} = w^T\mu^-, \quad \sigma_{Z^-} = \sqrt{w^T\Sigma^- w},$$

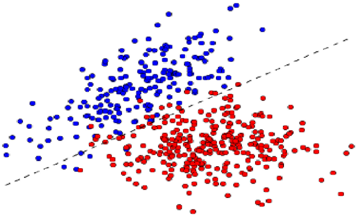
*in which $\phi$ is the c.d.f of the standard normal distribution, e.g.,*

$$\phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}t^2)dt, \quad \textit{for} \quad \forall x \in \mathbb{R}.$$

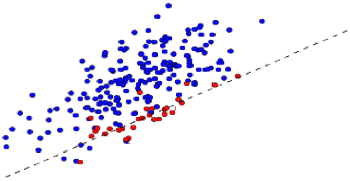
Prediction error is a smooth function of $w \Rightarrow$ we can compute the gradient and ...

# Outline

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



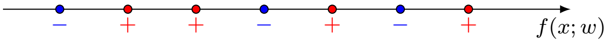
(a) Balanced data set



(b) Imbalanced data set

# Receiver Operating Characteristic (ROC) Curve

- Sorted outputs based on descending value of $f(x;w) = w^T x$



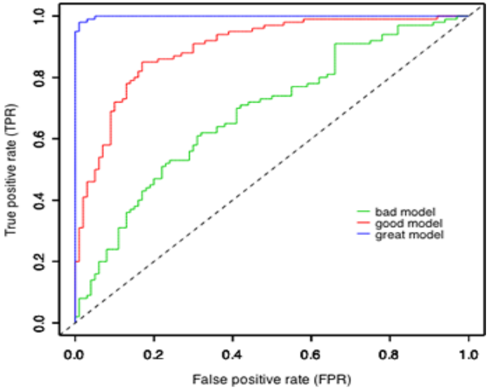

| | **Predicted Positive** | **Predicted Negative** |
|---|---|---|
| **Actual Positive** | True Positive (TP) | False Negative (FN) |
| **Actual Negative** | False Positive (FP) | True Negative (TN) |

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

- ROC curve presents the tradeoff between the TPR and the FPR, for all possible thresholds.

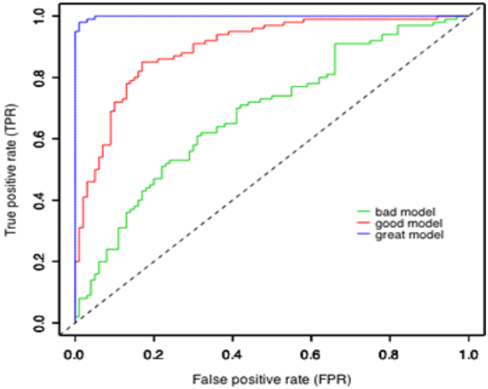
# Area Under ROC Curve (AUC)

- How we can compare ROC curves?



# Area Under ROC Curve (AUC)

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



# An Unbiased Estimation of AUC Value

An unbiased estimation of the AUC value of a linear classifier can be obtained via Wilcoxon-Mann-Whitney (WMW) statistic result *(Mann and R.Whitney (1947))*, e.g.,

$$AUC\left(f;\mathcal{S}^+,\mathcal{S}^-\right) = \frac{\sum_{i=1}^{n^+}\sum_{j=1}^{n^-} \mathbb{1}\left[f(x_i^+;w) > f(x_j^-;w)\right]}{n^+ \cdot n^-}.$$

where

$$\mathbb{1}\left[f(x_i^+;w) > f(x_j^-;w)\right] = \begin{cases} +1 & \text{if } \ f(x_i^+;w) > f(x_j^-;w), \\ 0 & \text{otherwise.} \end{cases}$$

in which $\mathcal{S} = \mathcal{S}^+ \cup \mathcal{S}^-$.

The indicator function $\mathbb{1}[\cdot]$ can be approximate with:

- Sigmoid surrogate function, *Yan et al. (2003)*,
- Pairwise exponential loss or pairwise logistic loss, *Rudin and Schapire (2009)*,
- Pairwise hinge loss, *Steck (2007)*,

$$F_{hinge}(w) = \frac{\sum_{i=1}^{n^+} \sum_{j=1}^{n^-} \max\left\{0, 1 - \left(f(x_j^-; w) - f(x_i^+; w)\right)\right\}}{n^+ \cdot n^-}.$$

# Measuring AUC Statistically

- Let $\mathcal{X}^+$ and $\mathcal{X}^-$ denote the space of the positive and negative input values,
- Then $x_i^+$ is an i.i.d. observation of the random variable $X^+$ and $x_j^-$ is an i.i.d. observation of the random variable $X^-$,
- If the joint probability distribution $P_{X^+,X^-}(x^+, x^-)$ is known, the actual associated AUC value of a linear classifier $f(x; w) = w^T x$ is defined as

$$
\begin{aligned}
AUC(f) &= \mathbb{E}_{\mathcal{X}^+,\mathcal{X}^-}\left[\mathbb{1}\left[f(X^+; w) > f(X^-; w)\right]\right] \\
&= \int_{\mathcal{X}^+}\int_{\mathcal{X}^-} P_{X^+,X^-}(x^+, x^-)\mathbb{1}\left[f(x^+; w) > f(x^-; w)\right] dx^- dx^+.
\end{aligned}
$$

**Lemma 4**

*We can interpret AUC value as a probability value:*

$$\begin{aligned} F_{AUC}(w) &= 1 - AUC(f) \\ &= 1 - \mathbb{E}_{\mathcal{X}^+,\mathcal{X}^-}\left[\mathbb{1}\left[f\left(X^+;w\right) > f\left(X^-;w\right)\right]\right] \\ &= 1 - P\left(Z<0\right), \end{aligned}$$

*where*

$$Z = w^T\left(X^- - X^+\right),$$

*for $X^+$ and $X^-$ as random variables from positive and negative classes, respectively.*

**Theorem 5**

*If two random variables $X^+$ and $X^-$ have a joint multivariate normal distribution, such that*

$$\begin{pmatrix} X^+ \\ X^- \end{pmatrix} \sim \mathcal{N}(\mu, \Sigma),$$

$$\textit{where} \quad \mu = \begin{pmatrix} \mu^+ \\ \mu^- \end{pmatrix} \quad \textit{and} \quad \Sigma = \begin{pmatrix} \Sigma^{++} & \Sigma^{+-} \\ \Sigma^{-+} & \Sigma^{--} \end{pmatrix},$$

*then the AUC function can be defined as*

$$F_{AUC}(w) = 1 - \phi\left(\frac{\mu_Z}{\sigma_Z}\right),$$

*where*

$$\mu_Z = w^T\left(\mu^- - \mu^+\right) \quad \textit{and}$$

$$\sigma_Z = \sqrt{w^T\left(\Sigma^{--} + \Sigma^{++} - \Sigma^{-+} - \Sigma^{+-}\right)w},$$

*and is the c.d.f of the standard normal distribution.*

# AUC as a Smooth Function

**Theorem 5**

*If two random variables $X^+$ and $X^-$ have a joint multivariate normal distribution, such that*

$$\begin{pmatrix} X^+ \\ X^- \end{pmatrix} \sim \mathcal{N}(\mu, \Sigma),$$

$$\textit{where} \quad \mu = \begin{pmatrix} \mu^+ \\ \mu^- \end{pmatrix} \quad \textit{and} \quad \Sigma = \begin{pmatrix} \Sigma^{++} & \Sigma^{+-} \\ \Sigma^{-+} & \Sigma^{--} \end{pmatrix},$$

*then the AUC function can be defined as*

$$F_{AUC}(w) = 1 - \phi\left(\frac{\mu_Z}{\sigma_Z}\right),$$

*where*

$$\mu_Z = w^T\left(\mu^- - \mu^+\right) \quad \textit{and}$$

$$\sigma_Z = \sqrt{w^T\left(\Sigma^{--} + \Sigma^{++} - \Sigma^{-+} - \Sigma^{+-}\right)w},$$

*and is the c.d.f of the standard normal distribution.*

AUC is a smooth function of $w \Rightarrow$ we can compute the gradient.

# Outline

# Computational Settings

- We have used *gradient descent with backtracking line search* as the optimization method.
- The algorithm is implemented in *Python 2.7.11* and computations are performed on the *COR@L computational cluster*.
- We have used both *artificial data sets* and *real data sets*.
- We used *five-fold cross-validation* with the *train-test* framework.

- Artificial data points with normal distribution are generated randomly.

| Name | $d$ | $n$ | $P^+$ | $P^-$ | $out\%$ |
|---|---|---|---|---|---|
| $data_1$ | 500 | 5000 | 0.05 | 0.95 | 0 |
| $data_2$ | 500 | 5000 | 0.35 | 0.65 | 5 |
| $data_3$ | 500 | 5000 | 0.5 | 0.5 | 10 |
| $data_4$ | 1000 | 5000 | 0.15 | 0.85 | 0 |
| $data_5$ | 1000 | 5000 | 0.4 | 0.6 | 5 |
| $data_6$ | 1000 | 5000 | 0.5 | 0.5 | 10 |
| $data_7$ | 2500 | 5000 | 0.1 | 0.9 | 0 |
| $data_8$ | 2500 | 5000 | 0.35 | 0.65 | 5 |
| $data_9$ | 2500 | 5000 | 0.5 | 0.5 | 10 |

| Data | $F_{error}(w)$ Minimization Exact moments | | $F_{error}(w)$ Minimization Approximate moments | | $F_{log}(w)$ Minimization | |
|---|---|---|---|---|---|---|
| | Accuracy± std | Time (s) | Accuracy ± std | Time (s) | Accuracy ± std | Time (s) |
| $data_1$ | 0.9965±0.0008 | 0.25 | 0.9907±0.0014 | 1.04 | 0.9897±0.0018 | 3.86 |
| $data_2$ | 0.9905±0.0023 | 0.26 | **0.9806**±0.0032 | 0.86 | 0.9557±0.0049 | 13.72 |
| $data_3$ | 0.9884±0.0030 | 0.03 | **0.9745**±0.0037 | 1.28 | 0.9537±0.0048 | 15.79 |
| $data_4$ | 0.9935±0.0017 | 0.63 | 0.9791±0.0034 | 5.51 | 0.9782±0.0031 | 10.03 |
| $data_5$ | 0.9899±0.0026 | 5.68 | **0.9716**±0.0048 | 10.86 | 0.9424±0.0055 | 28.29 |
| $data_6$ | 0.9904±0.0017 | 0.83 | **0.9670**±0.0058 | 5.18 | 0.9291±0.0076 | 25.47 |
| $data_7$ | 0.9945±0.0019 | 4.79 | 0.9786±0.0028 | 32.75 | 0.9697±0.0031 | 43.20 |
| $data_8$ | 0.9901±0.0013 | 9.96 | 0.9290±0.0045 | 119.64 | 0.9263±0.0069 | 104.94 |
| $data_9$ | 0.9899±0.0028 | 1.02 | 0.9249±0.0096 | 68.91 | 0.9264±0.0067 | 123.85 |

# Real Data Sets Information

These data sets can be downloaded from LIBSVM website [1] and UCI machine learning repository [2].

| Name | AC | $d$ | $n$ | $P^+$ | $P^-$ |
|---|---|---|---|---|---|
| fourclass | $[-1, 1]$, real | 2 | 862 | 0.35 | 0.65 |
| svmguide1 | $[-1, 1]$, real | 4 | 3089 | 0.35 | 0.65 |
| diabetes | $[-1, 1]$, real | 8 | 768 | 0.35 | 0.65 |
| shuttle | $[-1, 1]$, real | 9 | 43500 | 0.22 | 0.78 |
| vowel | $[-6, 6]$, int | 10 | 528 | 0.09 | 0.91 |
| magic04 | $[-1, 1]$, real | 10 | 19020 | 0.35 | 0.65 |
| poker | $[1, 13]$, int | 11 | 25010 | 0.02 | 0.98 |
| letter | $[0, 15]$, int | 16 | 20000 | 0.04 | 0.96 |
| segment | $[-1, 1]$, real | 19 | 210 | 0.14 | 0.86 |
| svmguide3 | $[-1, 1]$, real | 22 | 1243 | 0.23 | 0.77 |
| ijcnn1 | $[-1, 1]$, real | 22 | 35000 | 0.1 | 0.9 |
| german | $[-1, 1]$, real | 24 | 1000 | 0.3 | 0.7 |
| landsat satellite | $[27, 157]$, int | 36 | 4435 | 0.09 | 0.91 |
| sonar | $[-1, 1]$, real | 60 | 208 | 0.5 | 0.5 |
| a9a | binary | 123 | 32561 | 0.24 | 0.76 |
| w8a | binary | 300 | 49749 | 0.02 | 0.98 |
| mnist | $[0, 1]$, real | 782 | 100000 | 0.1 | 0.9 |
| colon-cancer | $[-1, 1]$, real | 2000 | 62 | 0.35 | 0.65 |
| gisette | $[-1, 1]$, real | 5000 | 6000 | 0.49 | 0.51 |

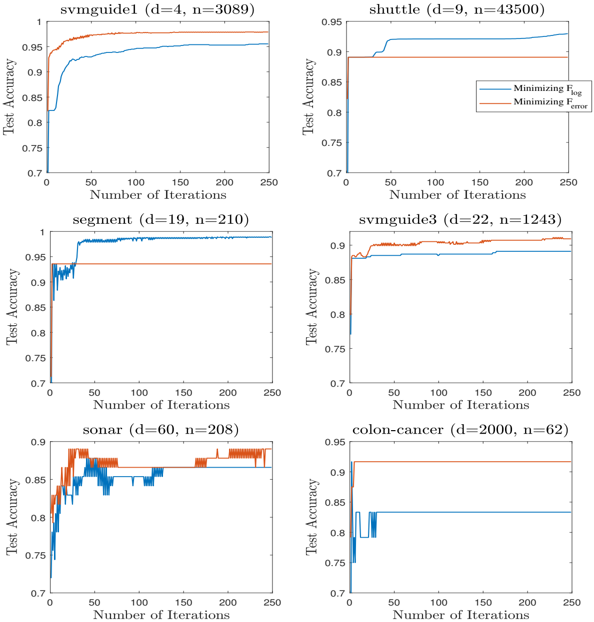
[1] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html

[2] http://archive.ics.uci.edu/ml/

# Optimizing $F_{error}(w)$ vs. $F_{log}(w)$ on Real Data

| Data | $F_{error}(w)$ Minimization | | $F_{log}(w)$ Minimization | |
|---|---|---|---|---|
| | Accuracy± std | Time (s) | Accuracy ± std | Time (s) |
| fourclass | 0.8782±0.0162 | 0.02 | 0.8800±0.0147 | 0.12 |
| svmguide1 | **0.9735**±0.0047 | 0.42 | 0.9506±0.0070 | 0.28 |
| diabetes | 0.8832±0.0186 | 1.04 | 0.8839±0.0193 | 0.13 |
| shuttle | 0.8920±0.0015 | 0.01 | **0.9301**±0.0019 | 4.05 |
| vowel | 0.9809±0.0112 | 0.91 | 0.9826±0.0088 | 0.11 |
| magic04 | 0.8867±0.0044 | 0.66 | 0.8925±0.0041 | 1.75 |
| poker | 0.9897±0.0008 | 0.17 | 0.9897±0.0008 | 10.96 |
| letter | 0.9816±0.0015 | 0.01 | 0.9894±0.0009 | 4.51 |
| segment | 0.9316±0.0212 | 0.17 | **0.9915**±0.0101 | 0.36 |
| svmguide3 | **0.9118**±0.0136 | 0.39 | 0.8951±0.0102 | 0.17 |
| ijcnn1 | 0.9512±0.0011 | 0.01 | 0.9518±0.0011 | 4.90 |
| german | 0.8780±0.0125 | 1.09 | 0.8826±0.0159 | 0.62 |
| landsat satellite | 0.9532±0.0032 | 0.01 | 0.9501±0.0049 | 3.30 |
| sonar | **0.8926**±0.0292 | 0.49 | 0.8774±0.0380 | 0.92 |
| a9a | 0.9193±0.0021 | 0.98 | 0.9233±0.0020 | 11.45 |
| w8a | 0.9851±0.0005 | 0.36 | 0.9876±0.004 | 24.16 |
| mnist | 0.9909±0.0004 | 3.79 | 0.9938±0.0004 | 136.83 |
| colon-cancer | **0.9364**±0.0394 | 15.92 | 0.8646±0.0555 | 1.20 |
| gisette | 0.9782±0.0025 | 310.72 | 0.9706±0.0036 | 136.71 |

# Optimizing $F_{error}(w)$ vs. $F_{log}(w)$ on Real Data



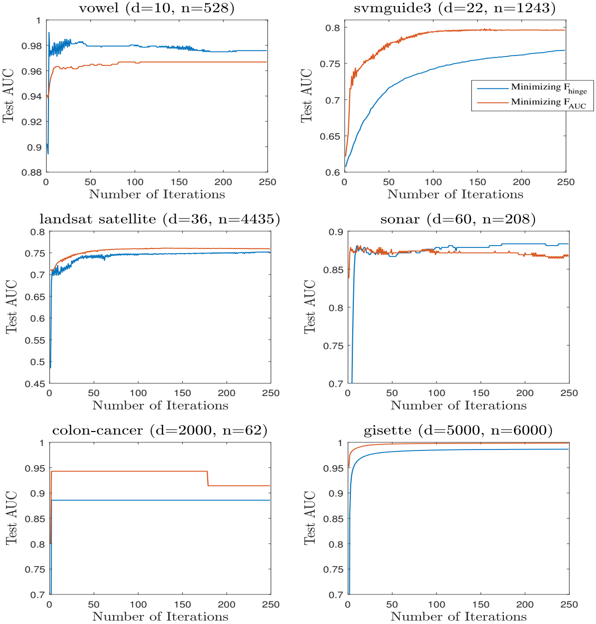
# Optimizing $F_{AUC}(w)$ vs. $F_{hinge}(w)$ on Artificial Data

| Data | $F_{AUC}(w)$ **Minimization** **Exact moments** | | $F_{AUC}(w)$ **Minimization** **Approximate moments** | | $F_{hinge}(w)$ **Minimization** | |
|---|---|---|---|---|---|---|
| | AUC± std | Time (s) | AUC ± std | Time (s) | AUC ± std | Time (s) |
| $data_1$ | 0.9972±0.0014 | 0.01 | **0.9941**±0.0027 | 0.23 | 0.9790±0.0089 | 5.39 |
| $data_2$ | 0.9963±0.0016 | 0.01 | **0.9956**±0.0018 | 0.22 | 0.9634±0.0056 | 159.23 |
| $data_3$ | 0.9965±0.0015 | 0.01 | **0.9959**±0.0018 | 0.24 | 0.9766±0.0041 | 317.44 |
| $data_4$ | 0.9957±0.0018 | 0.02 | **0.9933**±0.0022 | 0.83 | 0.9782±0.0054 | 23.36 |
| $data_5$ | 0.9962±0.0011 | 0.02 | **0.9951**±0.0013 | 0.80 | 0.9589±0.0068 | 110.26 |
| $data_6$ | 0.9962±0.0013 | 0.02 | **0.9949**±0.0015 | 0.82 | 0.9470±0.0086 | 275.06 |
| $data_7$ | 0.9965±0.0021 | 0.08 | **0.9874**±0.0034 | 4.61 | 0.9587±0.0092 | 28.31 |
| $data_8$ | 0.9966±0.0008 | 0.07 | **0.9929**±0.0017 | 4.54 | 0.9514±0.0051 | 104.16 |
| $data_9$ | 0.9962±0.0014 | 0.08 | **0.9932**±0.0020 | 4.54 | 0.9463±0.0085 | 157.62 |

# Optimizing $F_{AUC}(w)$ vs. $F_{hinge}(w)$ on Real Data

| Data | $F_{AUC}(w)$ **Minimization** | | $F_{hinge}(w)$ **Minimization** | |
|---|---|---|---|---|
| | AUC± std | Time (s) | AUC ± std | Time (s) |
| fourclass | 0.8362±0.0312 | 0.01 | 0.8362±0.0311 | 6.81 |
| svmguide1 | 0.9717±0.0065 | 0.06 | 0.9863±0.0037 | 35.09 |
| diabetes | 0.8311±0.0311 | 0.03 | 0.8308±0.0327 | 12.48 |
| shuttle | 0.9872±0.0013 | 0.11 | 0.9861±0.0017 | 2907.84 |
| vowel | 0.9585±0.0333 | 0.12 | **0.9765**±0.0208 | 2.64 |
| magic04 | 0.8382±0.0071 | 0.11 | 0.8419±0.0071 | 1391.29 |
| poker | 0.5054±0.0224 | 0.11 | 0.5069±0.0223 | 1104.56 |
| letter | 0.9830±0.0029 | 0.12 | 0.9883±0.0023 | 121.49 |
| segment | 0.9948±0.0035 | 0.21 | 0.9992±0.0012 | 4.23 |
| svmguide3 | **0.8013**±0.0420 | 0.34 | 0.7877±0.0432 | 23.89 |
| ijcnn1 | 0.9269±0.0036 | 0.08 | 0.9287±0.0037 | 2675.67 |
| german | 0.7938±0.0292 | 0.14 | 0.7919±0.0294 | 32.63 |
| landsat satellite | **0.7587**±0.0160 | 0.43 | 0.7458±0.0159 | 193.46 |
| sonar | 0.8214±0.0729 | 0.88 | **0.8456**±0.0567 | 2.15 |
| a9a | 0.9004±0.0039 | 0.92 | 0.9027±0.0037 | 15667.87 |
| w8a | 0.9636±0.0055 | 0.54 | 0.9643±0.0057 | 5353.23 |
| mnist | 0.9943±0.0009 | 0.64 | 0.9933±0.0009 | 28410.2393 |
| colon-cancer | **0.8942**±0.1242 | 2.50 | 0.8796±0.1055 | 0.05 |
| gisette | **0.9957**±0.0015 | 31.32 | 0.9858±0.0029 | 3280.38 |

# Optimizing $F_{AUC}(w)$ vs. $F_{hinge}(w)$ on Real Data



# Outline

# Summary

- We proposed some conditions under which the expected error and AUC are smooth functions.
- Any gradient-based optimization method can be applied to directly optimize these functions.
- These new proposed approaches work efficiently without perturbing the unknown distribution of the real data sets.
- Studying data distributions may lead to new efficient approaches.

# References


- H. B. Mann and D. R.Whitney. On a test whether one of two random variables is stochastically larger than the other. *Ann. Math. Statist, 18:50-60, 1947.*
- Y.L. Tong. The multivariate normal distribution. *Springer Series in Statistics, 1990.*
- G. Casella and R.L. Berger. Statistical Inference. *Pacific Grove, CA: Duxbury, 2, 2002.*

Thanks for your attention!