Second-Order Perceptron

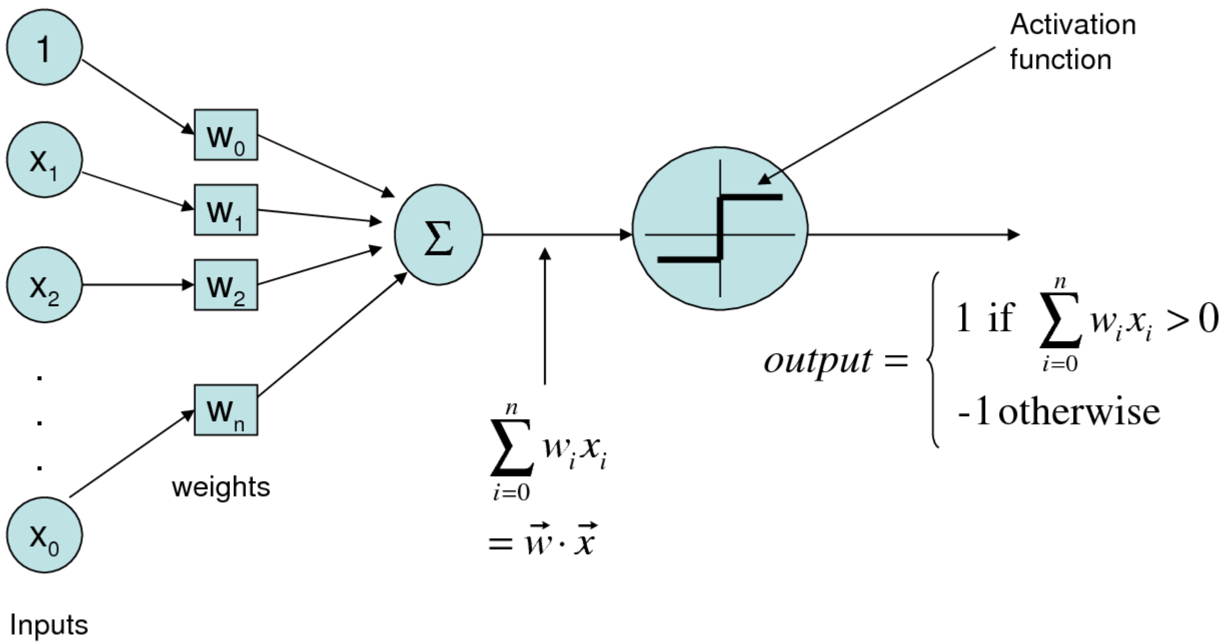
Nicolò Cesa-Bianchi, Alex Conconi, Claudio Gentile: A Second-Order Perceptron Algorithm. SIAM J. Comput. 34(3): 640-668 (2005)

Outline

- Introduction: Perceptron
- Algorithm: Second-order perceptron
- Analysis: Mistake bounds
- Simulations
- Conclusions

Perceptron Algorithm (F. Rosenblatt, 1958)

- One of the oldest machine learning algorithm
- Online algorithm for learning a linear thre or



Perceptron Algorithm (F. Rosenblatt, 1958)

- Goal: find a linear classifier with small error

Initialize $\mathbf{w}_0 = \mathbf{0}$
for $t = 1, 2, \ldots$ **do**
  Observe $\mathbf{x}_t$ and predict $\hat{y}_t = \mathrm{sgn}(\mathbf{w}_{t-1}^T \mathbf{x}_t)$
  Update
  - If $\mathbf{w}_{t-1}^T \mathbf{x}_t y_t \leq 0$, then $\mathbf{w}_t = \mathbf{w}_{t-1} + \mathbf{x}_t y_t$
  - Otherwise $\mathbf{w}_t = \mathbf{w}_{t-1}$

end for

If no error, keeping the same; otherwise, update.

Perceptron Mistake Bound

- Consider w* separate the data: $\mathbf{w}_*^T \mathbf{x}_i y_i > 0$
- Define margin

$$\gamma = \frac{\min_i \left| \mathbf{w}_*^T \mathbf{x}_i \right|}{\left\| \mathbf{w}_* \right\|_2 \sup_i \left\| \mathbf{x}_i \right\|_2}$$

The larger, the more confidence

Norm of $\mathbf{x}$: the larger, the larger mistake bound

- The number of mistakes perceptron makes is at most γ^{-2}

Proof of Perceptron Mistake Bound

[Novikoff, 1963]

Proof: Let $\mathbf{v}_k$ be the hypothesis before the k-th mistake. Assume that the k-th mistake occurs on the input example $(\mathbf{x}_i, y_i)$.

$$\gamma = \frac{\min_i |\mathbf{w}_*^T \mathbf{x}_i|}{\|\mathbf{w}_*\|_2 \sup_i \|\mathbf{x}_i\|_2}$$

First,

$$\begin{aligned}
\|\mathbf{v}_{k+1}\|^2 &= \|\mathbf{v}_k + y_i \mathbf{x}_i\|^2 \\
&= \|\mathbf{v}_k\|^2 + 2y_i(\mathbf{v}_k^T \mathbf{x}_i) \\
&\quad + \|\mathbf{x}_i\|^2 \\
&\leq \|\mathbf{v}_k\|^2 + R^2 \\
&\leq kR^2 (R := \sup_i \|\mathbf{x}\|_2)
\end{aligned}$$

Second,

$$\begin{aligned}
\mathbf{v}_{k+1} &= \mathbf{v}_k + y_i \mathbf{x}_i \\
\mathbf{v}_{k+1}^T \mathbf{u} &= \mathbf{v}_k^T \mathbf{u} + y_i \mathbf{x}_i^T \mathbf{u} \\
&\geq \mathbf{v}_k^T \mathbf{u} + \gamma R
\end{aligned}$$

$$\mathbf{v}_{k+1}^T \mathbf{u} \geq k\gamma R.$$

Hence,

$$\sqrt{k} R \geq \|\mathbf{v}_{k+1}\| \geq \mathbf{v}_{k+1}^T \mathbf{u} \geq k\gamma R$$

$$k \leq \gamma^{-2}$$

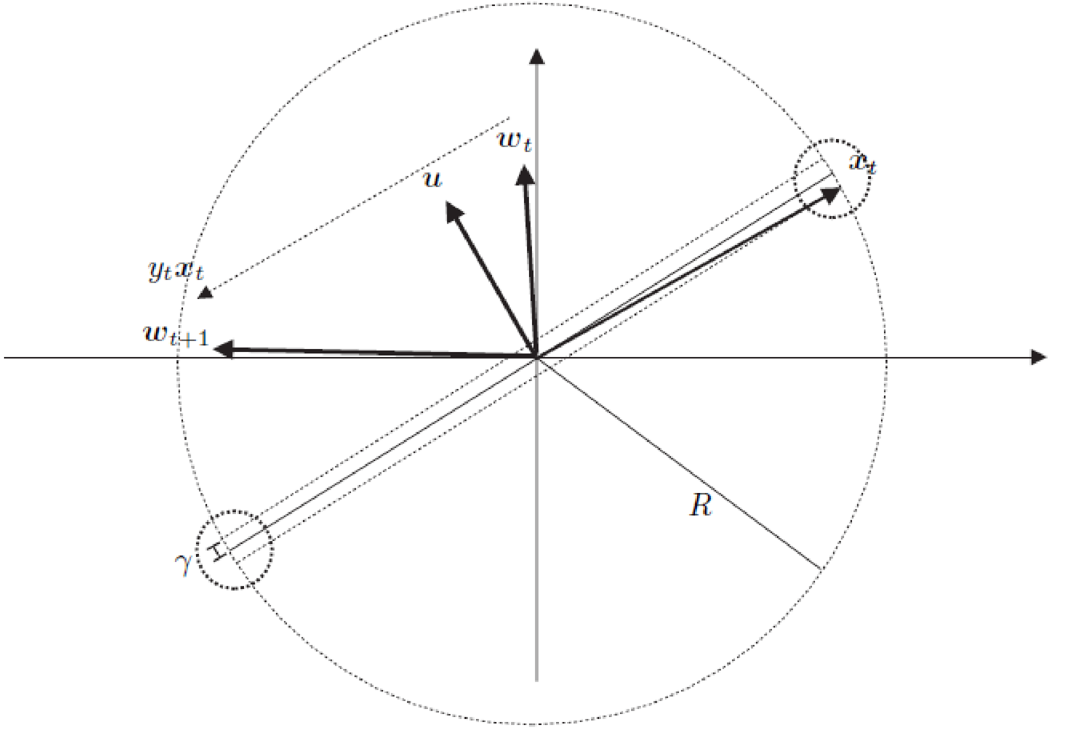
Problem of Perceptron

- Mistake bound

$$(R/\gamma)^2$$

$$R = \max_{1 \le s \le t} \|\boldsymbol{x}_s\|,$$

$$\gamma = \min_{1 \le s \le t} \left| \boldsymbol{u}^\top \boldsymbol{x}_t \right|$$



How to Incorporate Second-order Information?

- Intuitive idea: Whitened perceptron
  - Construct the correlation ma $M = \sum_{t=1}^{T} \boldsymbol{x}_t \boldsymbol{x}_t^\top$
  - Run standard Perceptron on

$$(M^{-1/2}\boldsymbol{x}_1, y_1), (M^{-1/2}\boldsymbol{x}_2, y_2), \ldots, (M^{-1/2}\boldsymbol{x}_T, y_T)$$

- Properties
  - Not incremental (instance available, label is hidden)
  - Make correlation mat to identity

$$\sum_{t=1}^{T}\left(M^{-1/2}\boldsymbol{x}_t\right)\left(M^{-1/2}\boldsymbol{x}_t\right)^\top = \sum_{t=1}^{T} M^{-1/2}\boldsymbol{x}_t \boldsymbol{x}_t^\top M^{-1/2} = M^{-1/2} M M^{-1/2} = I_n.$$

  - Mistake bound approaches 2

$$\frac{1}{\gamma^2}\max_t\left(\boldsymbol{x}_t^\top M^{-1}\boldsymbol{x}_t\right)\left(\boldsymbol{u}^\top M \boldsymbol{u}\right) = 1 + \frac{R^2 T - \gamma^2 T}{\sum_{t=1}^{T}\|\boldsymbol{x}_t\|^2 - \gamma^2 T}$$

Second-order Perceptron: Basic Form

- Algorithm

Parameter: $a > 0$.

Initialization: $X_0 = \emptyset$; $\boldsymbol{v}_0 = \boldsymbol{0}$; $k = 1$.

Repeat for $t = 1, 2, \ldots$:

1. get instance $\boldsymbol{x}_t \in \mathbb{R}^n$;
2. set $S_t = [\, X_{k-1} \; \boldsymbol{x}_t \,]$;
3. predict $\widehat{y}_t = \mathrm{SGN}(\boldsymbol{w}_t^\top \boldsymbol{x}_t) \in \{-1, +1\}$, where $\boldsymbol{w}_t = \left(a I_n + S_t\, S_t^\top\right)^{-1} \boldsymbol{v}_{k-1}$;
4. get label $y_t \in \{-1, +1\}$;
5. if $\widehat{y}_t \neq y_t$ then:

$$\boldsymbol{v}_k = \boldsymbol{v}_{k-1} + y_t\, \boldsymbol{x}_t,$$
$$X_k = S_t,$$
$$k \leftarrow k + 1.$$

Not a linear-threshold predictor

$$\boldsymbol{w}_t = \operatorname*{argmin}_{\boldsymbol{v}} \left(\sum_{s \in \mathcal{M}_{t-1}} \left(\boldsymbol{v}^\top \boldsymbol{x}_s - y_s\right)^2 + a\, \|\boldsymbol{v}\|^2 \right)$$

X_k: store the mis-classified instances

Analysis

- Theorem

THEOREM 3.1. *The number m of mistakes made by the second-order Perceptron algorithm of Figure 3.1, run on any finite sequence $\mathcal{S} = ((\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots)$ of examples, satisfies*

$$m \le \inf_{\gamma > 0} \min_{\|\boldsymbol{u}\|=1} \left(\frac{D_\gamma(\boldsymbol{u}; \mathcal{S})}{\gamma} + \frac{1}{\gamma} \sqrt{(a + \boldsymbol{u}^\top X_m X_m^\top \boldsymbol{u}) \sum_{i=1}^{n} \ln(1 + \lambda_i / a)} \right),$$

where $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of $X_m X_m^\top$.

$$D_\gamma(\boldsymbol{u}; (\boldsymbol{x}, y)) = \max\{0, \gamma - y\, \boldsymbol{u}^\top \boldsymbol{x}\}$$

$$D_\gamma(\boldsymbol{u}; \mathcal{S}) = \sum_{t=1}^{T} D_\gamma(\boldsymbol{u}; (\boldsymbol{x}_t, y_t))$$

Sketched Proof

Let $A_0 = aI_n$ and $A_k = aI_n + X_k X_k^\top$

$$\boldsymbol{v}_k^\top A_k^{-1}\boldsymbol{v}_k \le \boldsymbol{v}_{k-1}^\top A_{k-1}^{-1}\boldsymbol{v}_{k-1} + \boldsymbol{x}_t^\top A_k^{-1}\boldsymbol{x}_t$$

$$\begin{aligned}
\boldsymbol{v}_m^\top A_m^{-1}\boldsymbol{v}_m &\le \sum_{k=1}^m \boldsymbol{x}_t^\top A_k^{-1}\boldsymbol{x}_t \\
&= \sum_{k=1}^m \left(1 - \frac{\det(A_{k-1})}{\det(A_k)}\right) \\
&\le \sum_{k=1}^m \ln\frac{\det(A_k)}{\det(A_{k-1})} \\
&= \sum_{i=1}^n \ln\left(1 + \frac{\lambda_i}{a}\right)
\end{aligned}$$

$$\begin{aligned}
\sqrt{\boldsymbol{v}_m^\top A_m^{-1}\boldsymbol{v}_m} &= \left\| A_m^{-1/2}\boldsymbol{v}_m \right\| \\
&\ge \frac{\left(A_m^{-1/2}\boldsymbol{v}_m\right)^\top \boldsymbol{z}}{\|\boldsymbol{z}\|} \\
&= \frac{\boldsymbol{v}_m^\top \boldsymbol{u}}{\sqrt{\boldsymbol{u}^\top A_m \boldsymbol{u}}} \\
&= \frac{\boldsymbol{v}_m^\top \boldsymbol{u}}{\sqrt{a + \boldsymbol{u}^\top X_m X_m^\top \boldsymbol{u}}} \\
&\ge \frac{\gamma\, m - D_\gamma(\boldsymbol{u}; \mathcal{S})}{\sqrt{a + \boldsymbol{u}^\top X_m X_m^\top \boldsymbol{u}}},
\end{aligned}$$

Extension-Kernel

THEOREM 3.3. *With the notation of Figure 3.1, let $\widetilde{\boldsymbol{y}}_t$ be the k-component vector whose first $k-1$ components are the labels y_i where the algorithm has made a mistake up to trial $t-1$ and whose last component is 0. Then, for all $\boldsymbol{x}_t \in \mathbb{R}^n$, we have*

$$\boldsymbol{v}_{k-1}^\top \left(aI_n + S_t\, S_t^\top\right)^{-1} \boldsymbol{x}_t = \widetilde{\boldsymbol{y}}_t^\top \left(aI_k + G_t\right)^{-1} \left(S_t^\top \boldsymbol{x}_t\right),$$

where $G_t = S_t^\top S_t$ is a $k \times k$ (Gram) matrix and I_k is the k-dimensional identity matrix.

COROLLARY 3.4. *The number m of mistakes made by the dual second-order Perceptron algorithm with kernel K, run on any finite sequence $\mathcal{S} = ((\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots)$ of examples, satisfies*

$$m \le \inf_{\gamma > 0} \min_{\|f\|_K = 1} \left(\frac{D_\gamma(f;\mathcal{S})}{\gamma} + \frac{1}{\gamma} \sqrt{\left(a + \sum_{t \in \mathcal{M}} f(\boldsymbol{x}_t)^2\right) \sum_i \ln\left(1 + \lambda_i / a\right)} \right).$$

The numbers λ_i are the nonzero eigenvalues of the kernel Gram matrix with entries $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, where $i, j \in \mathcal{M}$ and $\mathcal{M}$ is the set of indices of mistaken trials.

Extension-Adaptive Parameter

Parameter sequence: $\{a_k\}_{k=1,2,\ldots}$, $a_{k+1} > a_k > 0$, $k = 1, 2, \ldots$.
Initialization: $X_0 = \emptyset$; $\boldsymbol{v}_0 = \mathbf{0}$; $k = 1$.
Repeat for $t = 1, 2, \ldots$:

1. get instance $\boldsymbol{x}_t \in \mathbb{R}^n$;
2. set $S_t = [\, X_{k-1}\ \boldsymbol{x}_t \,]$;
3. predict $\widehat{y}_t = \textsc{sgn}(\boldsymbol{w}_t^\top \boldsymbol{x}_t) \in \{-1, +1\}$,
   where $\boldsymbol{w}_t = \left(a_k I_n + S_t S_t^\top\right)^{-1} \boldsymbol{v}_{k-1}$;
4. get label $y_t \in \{-1, +1\}$;
5. if $\widehat{y}_t \neq y_t$, then

$$\boldsymbol{v}_k = \boldsymbol{v}_{k-1} + y_t \boldsymbol{x}_t,$$
$$X_k = S_t,$$
$$k \leftarrow k + 1.$$

Extension-Pseudoinverse

Initialization: $X_0 = \emptyset$; $\boldsymbol{v}_0 = \mathbf{0}$; $k = 1$.

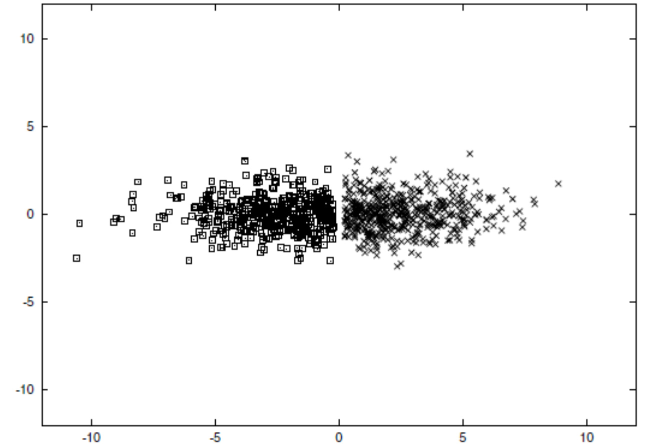
Repeat for $t = 1, 2, \ldots$:

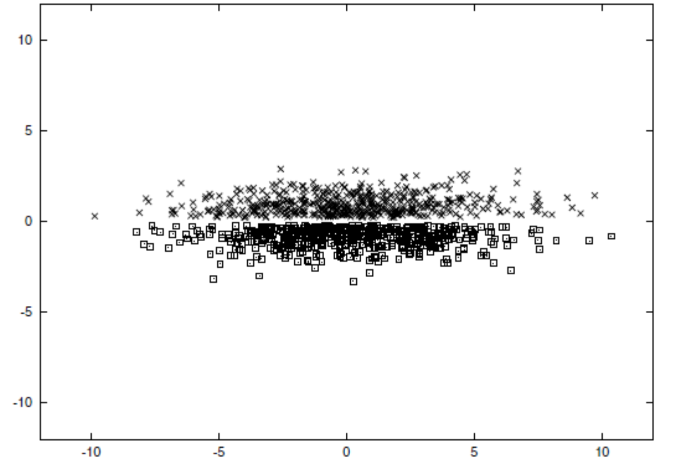
1. get instance $\boldsymbol{x}_t \in \mathbb{R}^n$;
2. set $S_t = [\, X_{k-1}\ \boldsymbol{x}_t \,]$;
3. predict $\widehat{y}_t = \text{SGN}(\boldsymbol{w}_t^\top \boldsymbol{x}_t) \in \{-1, +1\}$,
   where $\boldsymbol{w}_t = \left(S_t\, S_t^\top\right)^+ \boldsymbol{v}_{k-1}$;
4. get label $y_t \in \{-1, +1\}$;
5. if $\widehat{y}_t \neq y_t$, then:

$$\boldsymbol{v}_k = \boldsymbol{v}_{k-1} + y_t\, \boldsymbol{x}_t,$$
$$X_k = S_t,$$
$$k \leftarrow k + 1.$$

Simulations

- Linearly separable Gaussian data with 100 attributes
  - Correlation matrix: **a dominant** eigenvalue, eight times bigger than the others
  - Data 1: hyperplane is orthogonal to the eigenvector with the dominant eigenvalue
  - Data 2: hyperplane is orthogonal to the eigenvector with the first non-dominant eigenvalue





Simulation

- Procedure (Randomly repeat 5 times)
  - Train two epochs on 9,000 examples
  - Test on 3,000 examples
- Results

Algorithm	Mistakes, 1st dataset	Mistakes, 2nd dataset
Perceptron	30.20 (6.24)	29.80 (8.16)
second-order Perceptron, $a = 1$	9.60 (2.94)	5.60 (2.80)
second-order Perceptron, $a = 10$	10.60 (2.58)	3.20 (1.47)
second-order Perceptron, $a = 50$	14.00 (4.36)	10.40 (6.05)

Conclusions

- Second-order Perceptron algorithm
  - Online binary classification exploiting spectral properties
  - Prove the best known mistake bound for kernel-based linear threshold classifiers
  - Two variants for replacing inverse of correlation matrix

Q & A

Lemma

LEMMA D.1. *Let A be an arbitrary $n \times n$ positive semidefinite matrix, let $\boldsymbol{x}$ be an arbitrary vector in $\mathbb{R}^n$, and let $B = A - \boldsymbol{x}\,\boldsymbol{x}^\top$. Then*

$$\boldsymbol{x}\,A^+\,\boldsymbol{x} = \begin{cases} 1 & \textit{if } \boldsymbol{x} \notin \operatorname{span}(B), \\ 1 - \frac{\det_{\neq 0}(B)}{\det_{\neq 0}(A)} < 1 & \textit{if } \boldsymbol{x} \in \operatorname{span}(B), \end{cases} \tag{D.5}$$

where $\det_{\neq 0}(M)$ denotes the product of the nonzero eigenvalues of matrix M. Note that $\det_{\neq 0}(M) = \det(M)$ when M is nonsingular.