Machine Learning (CSE 446): Perceptron Convergence

Sham M Kakade

University of Washington
cse446-staff@cs.washington.edu

Review

Happy Medium?

Decision trees (that aren't too deep): use relatively few features to classify.

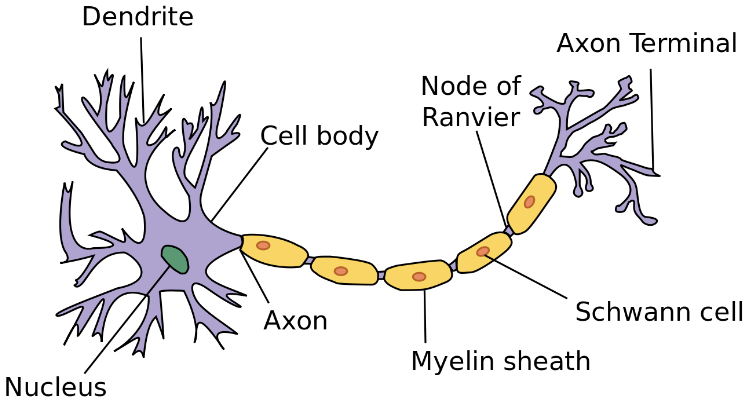
K-nearest neighbors: all features weighted equally.

Today: use all features, but weight them.

For today's lecture, assume that $y \in \{-1, +1\}$ instead of $\{0, 1\}$, and that $\mathbf{x} \in \mathbb{R}^d$.

Inspiration from Neurons

Image from Wikimedia Commons.



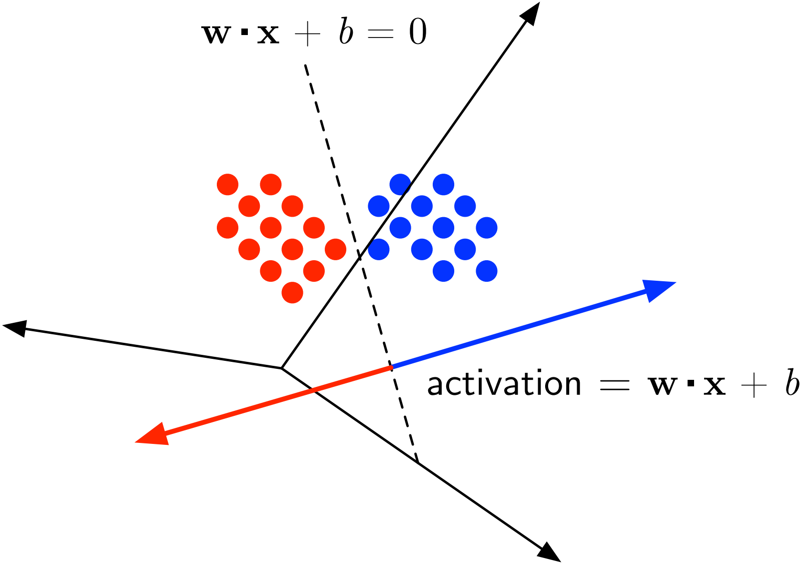
Input signals come in through dendrites, output signal passes out through the axon.

Perceptron Learning Algorithm

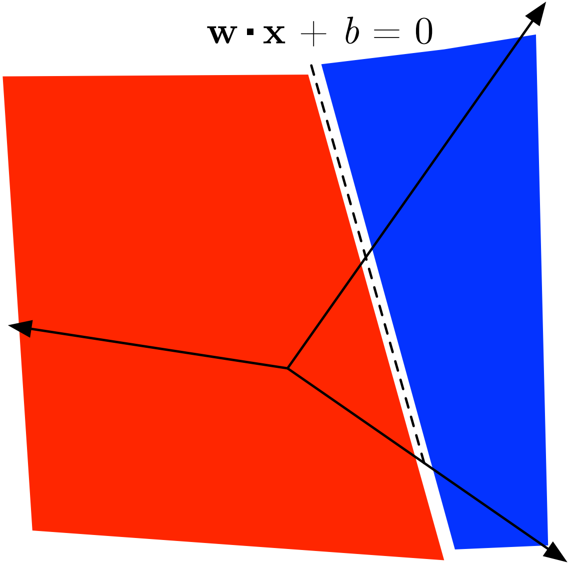
Data: $D = \langle(\mathbf{x}_n, y_n)\rangle_{n=1}^{N}$, number of epochs E
Result: weights $\mathbf{w}$ and bias b
initialize: $\mathbf{w} = \mathbf{0}$ and $b = 0$;
for $e \in \{1, \ldots, E\}$ **do**
  for $n \in \{1, \ldots, N\}$, *in random order* **do**
    # predict
    $\hat{y} = \text{sign}(\mathbf{w} \cdot \mathbf{x}_n + b)$;
    if $\hat{y} \neq y_n$ **then**
      # update
      $\mathbf{w} \leftarrow \mathbf{w} + y_n \cdot \mathbf{x}_n$;
      $b \leftarrow b + y_n$;
    end
  end
end
return $\mathbf{w}, b$

Algorithm 1: PerceptronTrain

Linear Decision Boundary



Linear Decision Boundary



Interpretation of Weight Values

What does it mean when ...

- $w_1 = 100$?
- $w_2 = -1$?
- $w_3 = 0$?

What if $\|w\|$ is "large"?

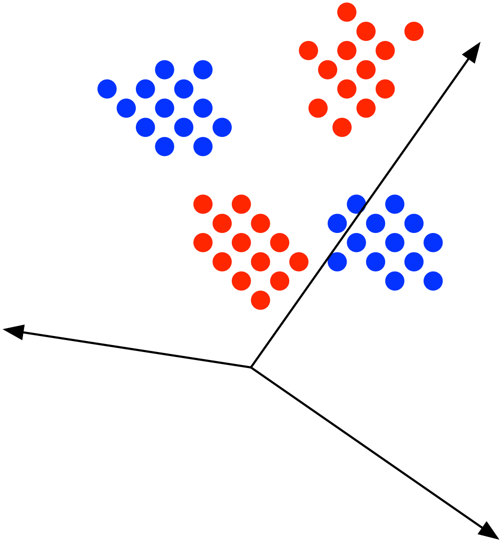
Today

What would we like to do?

- **Optimization problem:** find a classifier which minimizes the classification loss.
- The perceptron algorithm can be viewed as trying to do this...
- Problem: (in general) this is an NP-Hard problem.
- Let's still try to understand it...

This is the general approach of loss function minimization: find parameters which make our training error 'small' (and which also generalizes)

When does the perceptron not converge?



Linear Separability

A dataset $D = \langle(\mathbf{x}_n, y_n)\rangle_{n=1}^{N}$ is **linearly separable** if there exists some linear classifier (defined by $\mathbf{w}, b$) such that, for all n, $y_n = \text{sign}\left(\mathbf{w} \cdot \mathbf{x}_n + b\right)$.

If data are separable, (without loss of generality) can scale so that:

- "margin at 1", can assume for all (x, y)

$$y\left(\mathbf{w}_* \cdot \mathbf{x}\right) \geq 1$$

  (let w^* be smallest norm vector with margin 1).
- CIML: assumes $\|w^*\|$ is unit length and scales the "1" above.

Perceptron Convergence

Due to Rosenblatt (1958).

Theorem: Suppose data are scaled so that $\|\mathbf{x}_i\|_2 \leq 1$.
Assume D is linearly separable, and let be $\mathbf{w}_*$ be a separator with "margin 1".
Then the perceptron algorithm will converge in at most $\|\mathbf{w}_*\|^2$ epochs.

- Let $\mathbf{w}_t$ be the param at "iteration" t; $\mathbf{w}_0 = 0$
- "A Mistake Lemma": At iteration t

$$
\begin{aligned}
\text{If we make a mistake,} \quad \|\mathbf{w}_{t+1} - \mathbf{w}_*\|^2 &= \|\mathbf{w}_t - \mathbf{w}_*\|^2 \\
\text{If we do make a mistake,} \quad \|\mathbf{w}_{t+1} - \mathbf{w}_*\|^2 &\leq \|\mathbf{w}_t - \mathbf{w}_*\|^2 - 1
\end{aligned}
$$

- The theorem directly follows from this lemma. Why?

Proof of the “Mistake Lemma”

Proof of the "Mistake Lemma" (more scratch space)

Proof of the “Mistake Lemma” (more scratch space)

Voted Perceptron

- Suppose $\mathbf{w}^1$, $\mathbf{w}^4$, $\mathbf{w}^{10}$,$\mathbf{w}^{11}$ ... are the parameters right after we updated (e.g. after we made a mistake).
- Idea: instead of using the final $\mathbf{w}^t$ to classify, we classify with a majority vote using $\mathbf{w}^1$, $\mathbf{w}^4$, $\mathbf{w}^{10}$,$\mathbf{w}^{11}$ ...
- Why?

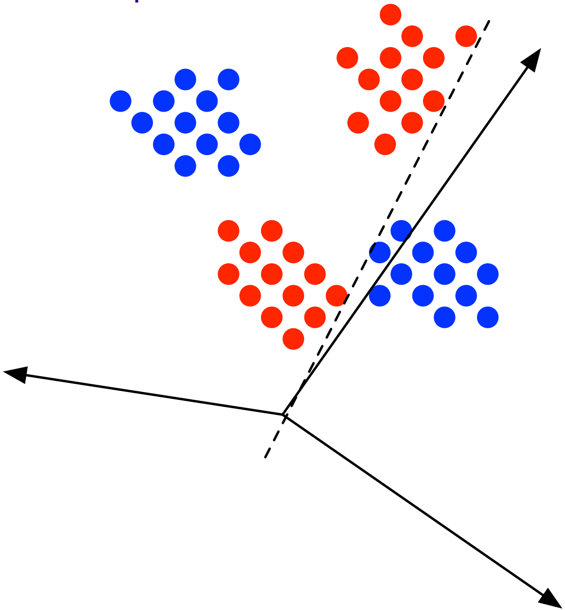
See CIML for details: Implementation and variants.

Voted Perceptron

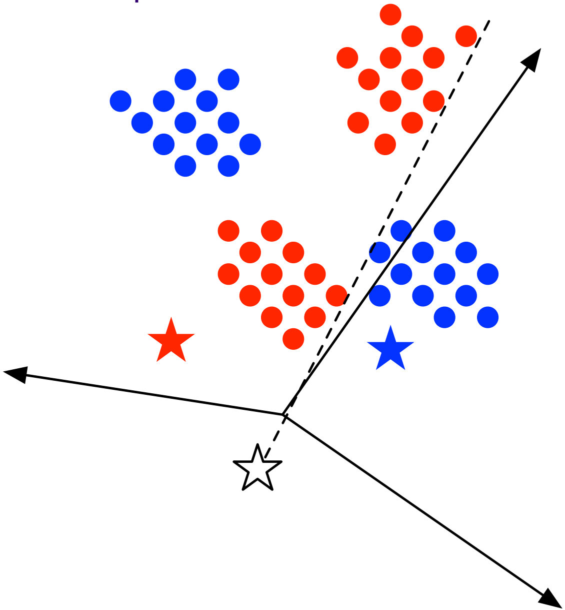
Let $\mathbf{w}^{(e,n)}$ and $b^{(e,n)}$ be the parameters after updating based on the nth example on epoch e.

$$\hat{y} = \mathsf{sign}\left(\sum_{e=1}^{E}\sum_{n=1}^{N}\mathsf{sign}(\mathbf{w}^{(e,n)} \cdot \mathbf{x} + b^{(e,n)})\right)$$

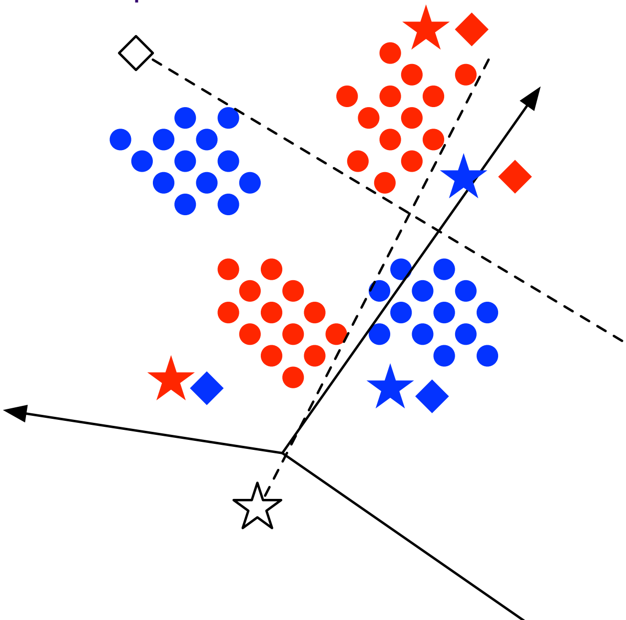
Voted Perceptron



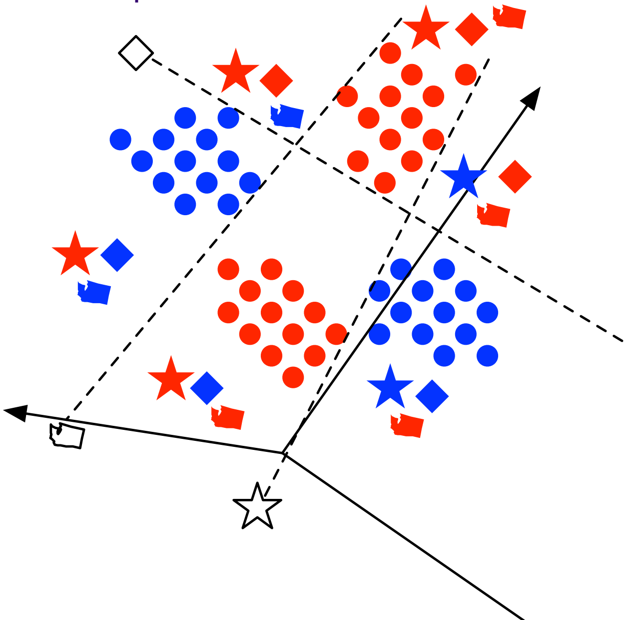
Voted Perceptron



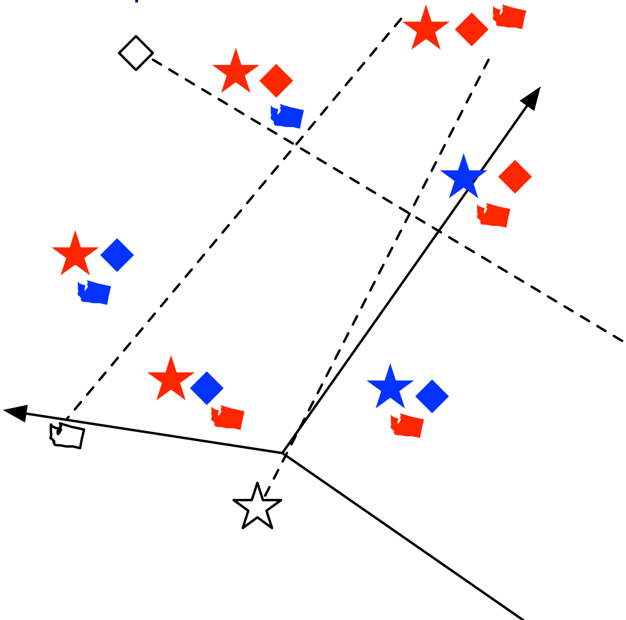
Voted Perceptron



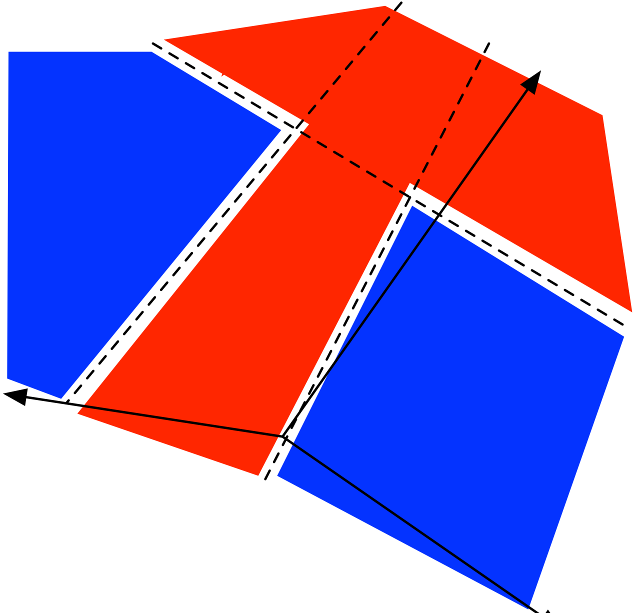
Voted Perceptron



Voted Perceptron



Voted Perceptron



References I

Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 1958.