

Nicholas Ruozzi University of Texas at Dallas

Slides adapted from David Sontag and Vibhav Gogate

Announcements

- Homework 1 available soon
- Piazza join if you haven't already
- Reminder: my office hours are 10am-11am on Tuesdays in ECSS 3.409
- Schedule and lecture notes available through the course website (see the link on eLearning)



Binary Classification

- Input $(x^{(1)}, y^{(1)}), \dots, (x^{(M)}, y^{(M)})$ with $x^{(m)} \in \mathbb{R}^n$ and $y^{(m)} \in \{-1, +1\}$
- We can think of the observations as points in \mathbb{R}^n with an associated sign (either +/- corresponding to 0/1)





Binary Classification

- Input $(x^{(1)}, y^{(1)}), \dots, (x^{(M)}, y^{(M)})$ with $x^{(m)} \in \mathbb{R}^n$ and $y^{(m)} \in \{-1, +1\}$
- We can think of the observations as points in \mathbb{R}^n with an associated sign (either +/- corresponding to 0/1)





What If the Data Isn't Separable?

- Input $(x^{(1)}, y^{(1)}), \dots, (x^{(M)}, y^{(M)})$ with $x^{(m)} \in \mathbb{R}^n$ and $y^{(m)} \in \{-1, +1\}$
- We can think of the observations as points in \mathbb{R}^n with an associated sign (either +/- corresponding to 0/1)



What is a good hypothesis space for this problem?



What If the Data Isn't Separable?

- Input $(x^{(1)}, y^{(1)}), \dots, (x^{(M)}, y^{(M)})$ with $x^{(m)} \in \mathbb{R}^n$ and $y^{(m)} \in \{-1, +1\}$
- We can think of the observations as points in \mathbb{R}^n with an associated sign (either +/- corresponding to 0/1)



What is a good hypothesis space for this problem?



Adding Features

- The idea:
 - Given the observations $x^{(1)}, \dots, x^{(M)}$, construct a feature vectors $\phi(x^{(1)}), \dots, \phi(x^{(M)})$
 - Use $\phi(x^{(1)}), \dots, \phi(x^{(M)})$ instead of $x^{(1)}, \dots, x^{(M)}$ in the learning algorithm
 - Goal is to choose ϕ so that $\phi(x^{(1)})$, ..., $\phi(x^{(M)})$ are linearly separable
 - Learn linear separators of the form $w^T \phi(x)$ (instead of $w^T x$)
- <u>Warning</u>: more expressive features can lead to overfitting!



• How can we decide between perfect classifiers?





• How can we decide between perfect classifiers?





• Define the margin to be the distance of the closest data point to the classifier





• Support vector machines (SVMs)



- Choose the classifier with the largest margin
 - Has good practical and theoretical performance





• In *n* dimensions, a hyperplane is a solution to the equation

$$w^T x + b = 0$$

with $w \in \mathbb{R}^n$, $b \in \mathbb{R}$

• The vector *w* is sometimes called the normal vector of the hyperplane





• In *n* dimensions, a hyperplane is a solution to the equation

$$w^T x + b = 0$$

• Note that this equation is scale invariant for any scalar *c*

$$c \cdot (w^T x + b) = 0$$





- The distance between a point y and a hyperplane $w^T + b = 0$ is the length of the segment perpendicular to the line to the point y
- The vector from *y* to *z* is given by

$$y - z = ||y - z|| \frac{w}{||w||}$$



Scale Invariance



- By scale invariance, we can assume that c = 1
- The maximum margin is always attained by choosing w^Tx + b = 0 so that it is equidistant from the closest data point classified as +1 and the closest data point classified as -1



Scale Invariance



We want to maximize the margin subject to the constraints that

$$y^{(i)}(w^T x^{(i)} + b) \ge 1$$

• But how do we compute the size of the margin?





$$y - z = ||y - z|| \frac{w}{||w||}$$

and

$$w^T y + b = 1$$
$$w^T z + b = 0$$

$$w^T(y-z)=1$$

and

$$w^{T}(y-z) = ||y-z|| ||w||$$

which gives

||y - z|| = 1/||w||



• This analysis yields the following optimization problem $\max_{w} \frac{1}{\|w\|}$

such that

$$y^{(i)}(w^T x^{(i)} + b) \ge 1$$
, for all i

• Or, equivalently,

 $\min_{w} \|w\|^2$

such that

$$y^{(i)}(w^T x^{(i)} + b) \ge 1$$
, for all i



$$\label{eq:such that} \begin{split} & \min_{w} \|w\|^2 \\ \text{such that} \\ & y^{(i)} \big(w^T x^{(i)} + b \big) \geq 1 \text{, for all } i \end{split}$$

- This is a standard quadratic programming problem
 - Falls into the class of convex optimization problems
 - Can be solved with many specialized optimization tools (e.g., quadprog() in MATLAB)





- Where does the name come from?
 - The set of all data points such that $y^{(i)}(w^T x^{(i)} + b) = 1$ are called support vectors



- What if the data isn't linearly separable?
 - Use feature vectors
 - Relax the constraints (coming soon)
- What if we want to do more than just binary classification (i.e., if y ∈ {1,2,3})?



Multiclass Classification











Regions correctly classified by exactly one classifier



- Compute a classifier for each label versus the remaining labels (i.e., and SVM with the selected label as plus and the remaining labels changed to minuses)
- Let $f^k(x) = w^{(k)^T}x + b^{(k)}$ be the classifier for the k^{th} label
- For a new datapoint *x*, classify it as

 $k' \in \operatorname{argmax}_k f^k(x)$

- Drawbacks:
 - If there are L possible labels, requires learning L classifiers over the entire data set
 - Doesn't make sense if the classifiers are not comparable





Regions in which points are classified by highest value of $w^T x + b$



One-Versus-One SVMs

- Alternative strategy is to construct a classifier for all possible pairs of labels
- Given a new data point, can classify it by majority vote (i.e., find the most common label among all of the possible classifiers)
- If there are L labels, requires computing $\binom{L}{2}$ different classifiers each of which uses only a fraction of the data
- Drawbacks: Can overfit if some pairs of labels do not have a significant amount of data (plus it can be computationally expensive)



One-Versus-One SVMs



Regions determined by majority vote over the classifiers

