LARGE MARGIN CLASSIFIERS

David Kauchak
CS 158 – Fall 2023

10/3/23

Assignment 5
- Experiments

Assignment 6: due Friday (10/13)

Next class: Meet in Edmunds 105

Midterm: out and due by the end of the day Friday

Course feedback
- Thanks!
- We’ll go over it at the beginning of next class

Which hyperplane?

Two main variations in linear classifiers:
- which hyperplane they choose when the data is linearly separable
- how they handle data that is not linearly separable

Linear approaches so far

Perceptron:
- separable:
- non-separable:

Gradient descent:
- separable:
- non-separable:
Linear approaches so far

Perceptron:
- separable:
  - finds some hyperplane that separates the data
- non-separable:
  - will continue to adjust as it iterates through the examples
  - final hyperplane will depend on which examples it saw recently

Gradient descent:
- separable and non-separable
  - finds the hyperplane that minimizes the objective function (loss + regularization)

Which hyperplane is this?

- Large margin classifiers
  Choose the line where the distance to the nearest point(s) is as large as possible

- Large margin classifiers
  The margin of a classifier is the distance to the closest points of either class
  Large margin classifiers attempt to maximize this
Support vectors
For any separating hyperplane, there exist some set of “closest points.”
These are called the support vectors.
For n dimensions, there will be at least n+1 support vectors.

Measuring the margin
The margin is the distance to the support vectors, i.e. the “closest points,” on either side of the hyperplane.

Measuring the margin
The margin is the distance to the support vectors, i.e. the “closest points,” on either side of the hyperplane.

What are the equations for the margin lines?
Measuring the margin

We know they’re the same distance apart (otherwise, they wouldn’t be support vectors!)

\[ w \cdot x_i + b = -c \]

What is \( c \)?

Depends! If we scale \( w \), we vary the constant without changing the separating hyperplane.

Larger \( w \) result in larger constants.

Smaller \( w \) result in smaller constants.
Measuring the margin

For now, let's assume \( c = 1 \).

\[
\begin{align*}
wx_i + b &= 1 \\
\end{align*}
\]

Distance from the hyperplane

How far away is this point from the hyperplane?

Distance from the hyperplane

Distance from the hyperplane

Distance from the hyperplane

How far away is this point from the hyperplane?

Distance from the hyperplane

Distance from the hyperplane

\[
\begin{align*}
d &= \sqrt{(-1)^2 + (-2)^2} = \sqrt{5}
\end{align*}
\]
Distance from the hyperplane

How far away is the point from the hyperplane?

$$d(x) = w \cdot x + b$$

$$w = (1, 2)$$

$$d((1, 1)) = 1 \times 1 + 2 \times 1 + 0 = 3?$$

Distance from the hyperplane

How far away is the point from the hyperplane?

$$d(x) = w \cdot x + b$$

$$w = (2, 4)$$

$$d((1, 1)) = 2 \times 1 + 4 \times 1 + 0 = 6?$$
Distance from the hyperplane

How far away is this point from the hyperplane?

\[ w = (1, 2) \]

\[ d(x) = \frac{w \cdot x + b}{\|w\|} \]

length normalized weight vectors

Distance from the hyperplane

How far away is this point from the hyperplane?

\[ w = (1, 2) \]

\[ d(x) = \frac{w \cdot x + b}{\|w\|} \]

\[ = \frac{\sqrt{5} + 0}{\sqrt{5}} \]

\[ = 1.34 \]

Distance from the hyperplane

The magnitude of the weight vector doesn’t matter

\[ w = (2, 4) \]

\[ d(x) = \frac{w \cdot x + b}{\|w\|} \]

length normalized weight vectors

Distance from the hyperplane

The magnitude of the weight vector doesn’t matter

\[ w = (0.5, 1) \]

\[ d(x) = \frac{w \cdot x + b}{\|w\|} \]

length normalized weight vectors
Measuring the margin

For now, let's just assume \( c = 1 \).

\[
\begin{align*}
    w \cdot x_i + b &= -1 \\
    w \cdot x_i + b &= 1
\end{align*}
\]

What is this distance?

Large margin classifier setup

Select the hyperplane with the largest margin where the points are classified correctly and outside the margin!

Setup as a constrained optimization problem:

\[
\begin{align*}
    \max_{w,b} & \quad \text{margin}(w,b) \\
    \text{subject to:} & \quad y_i (w \cdot x_i + b) \geq 1 \quad \forall i
\end{align*}
\]

what does this say?
Large margin classifier setup

Select the hyperplane with the largest margin where the points are classified correctly and outside the margin!

Setup as a constrained optimization problem:

\[
\max_{w,b} \frac{1}{\|w\|}
\quad \text{subject to:}
\]
\[
y_i(w \cdot x_i + b) \geq 1 \quad \forall i
\]

Maximizing the margin

\[
\min_{w,b} \|w\|
\quad \text{subject to:}
\]
\[
y_i(w \cdot x_i + b) \geq 1 \quad \forall i
\]

Maximizing the margin is equivalent to minimizing \(\|w\|\)
(subject to the separating constraints)

Measuring the margin

For now, let’s just assume \(c = 1\).

Claim: it does not matter what \(c\) we choose for the SVM problem. Why?

\[
w \cdot x_i + b = -1
\]
\[
w \cdot x_i + b = 1
\]
Measuring the margin

\[ w \cdot x_i + b = -c \]

What is this distance?

\[ w \cdot x_i + b = c \]

Maximizing the margin

\[
\begin{align*}
\min_{w,b} & \|w\| \\
\text{subject to:} & \quad y_i(w \cdot x_i + b) \geq c \quad \forall i \\
\end{align*}
\]

vs.

\[
\begin{align*}
\min_{w,b} & \|w\| \\
\text{subject to:} & \quad y_i(w \cdot x_i + b) \geq 1 \quad \forall i \\
\end{align*}
\]

Learn the exact same hyperplane just scaled by a constant amount

Because of this, often see it with \( c = 1 \)
Maximizing the margin: the real problem

\[
\begin{align*}
\min_{w,b} & \quad \|w\|^2 \\
\text{subject to: } & \quad y_i(w \cdot x_i + b) \geq 1 \quad \forall i
\end{align*}
\]

Why the squared?

Support vector machine problem

\[
\begin{align*}
\min_{w,b} & \quad \|w\|^2 \\
\text{subject to: } & \quad y_i(w \cdot x_i + b) \geq 1 \quad \forall i
\end{align*}
\]

This is a version of a quadratic optimization problem

Maximize/minimize a quadratic function

Subject to a set of linear constraints

Many, many variants of solving this problem (we’ll see one in a bit)

Minimizing \(\|w\|\) is equivalent to minimizing \(\|w\|^2\)

The sum of the squared weights is a convex function!
Soft Margin Classification

What about this problem?

subject to:

We’d like to learn something like this, but our constraints won’t allow it.

Slack variables

What effect does this have?

subject to:

slack penalties

subject to:

slack variables (one for each example)
Slack variables

\[ y_i (w \cdot x_i + b) \geq 1 - \xi_i \quad \forall i \]

subject to:

\[ \min_{w,b} \|w\|^2 + C \sum_i \xi_i \]

penalized by how far from "correct"

\[ \sum_i \xi_i \geq 0 \]

allowed to make a mistake

trade-off between margin maximization and penalization

Soft margin SVM

\[ \min_{w,b} \|w\|^2 + C \sum_i \xi_i \]

subject to:

\[ y_i (w \cdot x_i + b) \geq 1 - \xi_i \quad \forall i \]

\[ \xi_i \geq 0 \]

Still a quadratic optimization problem!

Demo

https://cs.stanford.edu/~karpathy/svmjs/demo/

Solving the SVM problem

49

50

51

52
Given the optimal solution, $w, b$:

Can we figure out what the slack penalties are for each point?

What do the margin lines represent wrt $w, b$?

What are the slack values for points outside (or on) the margin AND correctly classified?
10/3/23

**Understanding the Soft Margin SVM**

\[ y_i(w \cdot x_i + b) = 1 \]

\[ \min_{w,b} \|w\|^2 + C \sum \xi_i \]

subject to:
\[ y_i(w \cdot x_i + b) \geq 1 - \xi_i \quad \forall i \]
\[ \xi_i \geq 0 \]

What are the slack values for points inside the margin AND classified correctly?

What are the slack values for points that are incorrectly classified?

Difference from the point to the margin. Which is?
\[ \xi_i = 1 - y_i(w \cdot x_i + b) \]
Understanding the Soft Margin SVM

\[ y_i(w \cdot x_i + b) = 1 \]

subject to:
\[ y_i(w \cdot x_i + b) \geq 1 - \xi_i \quad \forall i \]
\[ \xi_i \geq 0 \]

\[ \min_{w,b} \|w\|^2 + C \sum_i \xi_i \]

"distance" to the hyperplane plus the "distance" to the margin

\[ y_i(w \cdot x_i + b) = 1 \]

subject to:
\[ y_i(w \cdot x_i + b) \geq 1 - \xi_i \quad \forall i \]
\[ \xi_i \geq 0 \]

"distance" to the hyperplane plus the "distance" to the margin

\[ -y_i(w \cdot x_i + b) \]

Why ?
Understanding the Soft Margin SVM

\[ y_i (w \cdot x_i + b) = 1 \]

subject to:

\[ \min_{w,b} \|w\|^2 + C \sum \xi_i \]

\[ y_i (w \cdot x_i + b) \geq 1 - \xi_i \quad \forall i \]

\[ \xi_i \geq 0 \]

"distance" to the hyperplane plus the "distance" to the margin

\[ -y_i (w \cdot x_i + b) \]

\[ \xi_i = \begin{cases} 
0 & \text{if } y_i (w \cdot x_i + b) \geq 1 \\
1 - y_i (w \cdot x_i + b) & \text{otherwise}
\end{cases} \]
Understanding the Soft Margin SVM

\[ \xi_i = \begin{cases} 
0 & \text{if } y_i(w \cdot x_i + b) \geq 1 \\
1 - y_i(w \cdot x_i + b) & \text{otherwise} 
\end{cases} \]

\[ \xi_i = \max(0, 1 - y_i(w \cdot x_i + b)) \]
\[ \xi_i = \max(0, 1 - yy') \]

Does this look familiar?

Hinge loss!

0/1 loss: \( l(y, y') = I[yy' \leq 0] \)

Hinge: \( l(y, y') = \max(0, 1 - yy') \)

Exponential: \( l(y, y') = \exp(-yy') \)

Squared loss: \( l(y, y') = (y - y')^2 \)

Understanding the Soft Margin SVM

\[
\begin{align*}
\min_{w, b} & \quad \|w\|^2 + C \sum_i \xi_i \\
\text{subject to} & \quad y_i(w \cdot x_i + b) \geq 1 - \xi_i \quad \forall i \\
& \quad \xi_i \geq 0
\end{align*}
\]

Do we need the constraints still?

Understanding the Soft Margin SVM

\[
\begin{align*}
\min_{w, b} & \quad \|w\|^2 + C \sum_i \xi_i \\
\text{subject to} & \quad y_i(w \cdot x_i + b) \geq 1 - \xi_i \quad \forall i \\
& \quad \xi_i \geq 0
\end{align*}
\]

\[
\begin{align*}
\min_{w, b} & \quad \|w\|^2 + C \sum_i \max(0, 1 - y_i(w \cdot x_i + b)) \\
\end{align*}
\]

Unconstrained problem!
Understanding the Soft Margin SVM

\[
\min_{w,b} \|w\|^2 + C \sum_{i=1}^{n} \text{loss}(y_i, y_i')
\]

Does this look like something we’ve seen before?

\[
\arg\min_{w,b} \sum_{i=1}^{n} \text{loss}(y_i, y_i') + \lambda \text{regularizer}(w,b)
\]

Gradient descent problem!

Soft margin SVM as gradient descent

One way to solve the soft margin SVM problem is using gradient descent

\[
\min_{w,b} \sum_{i=1}^{n} \text{loss}(y_i, y_i') + \lambda \text{regularizer}(w,b)
\]

hinge loss

L2 regularization

Gradient descent SVM solver

- pick a starting point \(w\)
- repeat until loss doesn’t decrease in all dimensions:
  - pick a dimension
  - move a small amount in that dimension towards decreasing loss (using the derivative)

\[
w_i = w_i - \eta \frac{d}{dw_i} (\text{loss}(w) + \text{regularizer}(w,b))
\]

\[
w_j = w_j + \eta \sum_{i=1}^{n} y_i x_i [y_i (w \cdot x + b) < 1] - \eta \lambda w_j
\]

Finds the largest margin hyperplane while allowing for a soft margin
Support vector machines: 2013

One of the most successful (if not the most successful) classification approach:

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>2013</th>
<th>2016</th>
<th>2019</th>
<th>2022</th>
</tr>
</thead>
<tbody>
<tr>
<td>decision tree</td>
<td>About 2,180,000</td>
<td>About 2,480,000</td>
<td>About 3,000,000</td>
<td>About 3,670,000</td>
</tr>
<tr>
<td>Support vector machine</td>
<td>About 1,940,000</td>
<td>About 2,430,000</td>
<td>About 3,620,000</td>
<td>About 3,290,000</td>
</tr>
<tr>
<td>k nearest neighbor</td>
<td>About 746,000</td>
<td>About 875,000</td>
<td>About 1,380,000</td>
<td>About 2,200,000</td>
</tr>
<tr>
<td>perceptron algorithm</td>
<td>About 84,000</td>
<td>About 104,000</td>
<td>About 153,000</td>
<td>About 230,000</td>
</tr>
</tbody>
</table>

Trends over time

[Chart showing trends over time]