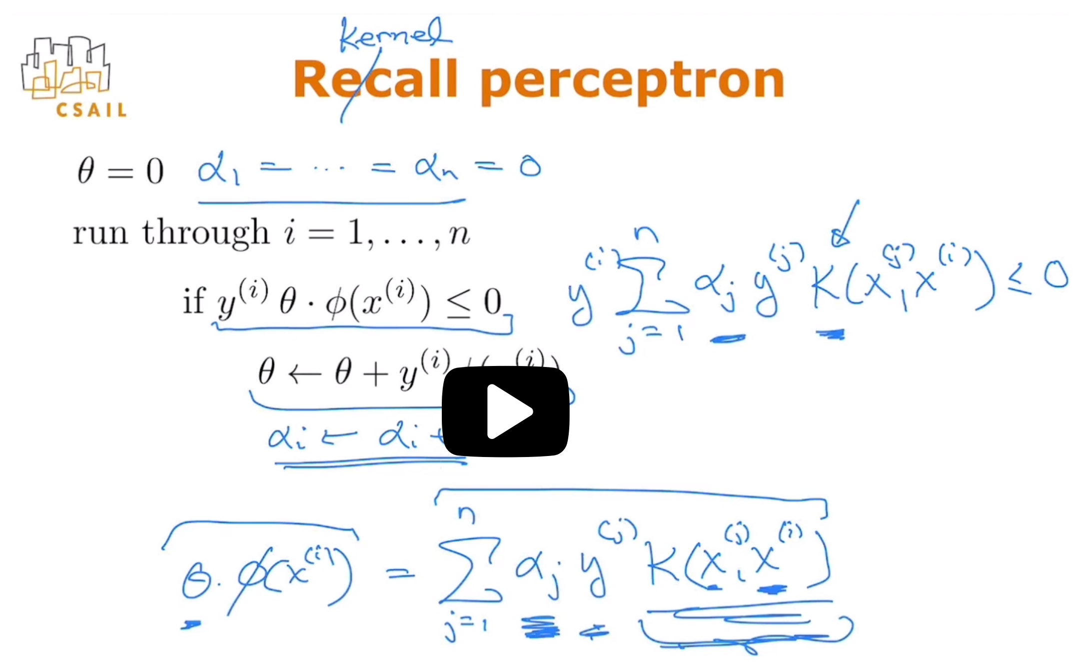
5. The Kernel Perceptron Algorithm

Computational Efficiency



7:34 / 7:34 1.0x

here as a kind of similarity measure.

How similar the j-th example is to the i-th example.

So our predicted value here is now

how important the j-th example is.

Its label times how similar the example

we wish to make a prediction on and the j-th training example.

All right.

That is now the kernel perceptron algorithm.

End of transcript. Skip to the start.

Video
Download video file

Transcripts
Download SubRip (.srt) file
Download Text (.txt) file

xuetangX.com 学堂在线

How the Kernel Perceptron Algorithm Works: Initalization

1/1 point (graded)
Recall that the original Perceptron Algorithm is given as the following:

Perceptron $\left(\left\{\left(x^{(i)}, y^{(i)}\right), i = 1, \ldots, n\right\}, T\right)$:

initialize $\theta = 0$ (vector);
for $t = 1, \ldots, T$,
for $i = 1, \ldots, n$,
if $y^{(i)}\left(\theta \cdot x^{(i)}\right) \leq 0$,
then update $\theta = \theta + y^{(i)} x^{(i)}$.

In the lecture, it was introduced that we can always express θ as

$$\theta = \sum_{j=1}^{n} \alpha_j y^{(j)} \phi\left(x^{(j)}\right)$$

where values of $\alpha_1, \ldots, \alpha_n$ may vary at each step of the algorithm. In other words, we can reformulate the algorithm so that we somehow initialize and update α_j's, instead of θ.

The reformulated algorithm, or **kernel perceptron** , can be given in the following form:

Kernel Perceptron $\left(\{(x^{(i)}, y^{(i)}), i = 1, \ldots, n, T\}\right)$

Initialize $\alpha_1, \alpha_2, \ldots, \alpha_n$ to some values;
for $t = 1, \ldots, T$
for $i = 1, \ldots, n$
if $(\text{Mistake Condition Expressed in } \alpha_j)$
Update α_j appropriately

Look at the initialization statement of the algorithm. Which of the following is an equivalent way to initialize $\alpha_1, \alpha_2, \ldots, \alpha_n$, if we want the same result as initializing $\theta = 0$?

- $\alpha_1 = \ldots = \alpha_n = \theta$
- $\alpha_1 = \ldots = \alpha_n = 1$
- $\alpha_1 = \ldots = \alpha_n = 0$ ✔
- $\alpha_1 = \ldots = \alpha_n = -1$

Solution:

Since $\theta = \sum_{j=1}^{n} \alpha_j y^{(j)} \phi(x^{(j)})$, setting $\alpha_j = 0$ for all j leads to $\theta = 0$.

Submit You have used 1 of 1 attempt

Answers are displayed within the problem

How the Kernel Perceptron Algorithm Works: The Update

1/1 point (graded)
As in the previous problem, our goal is to correctly reformulate the original perceptron algorithm. In other words, we want the algorithm to be about updating α_j's instead of θ.

Kernel Perceptron $\left(\{(x^{(i)}, y^{(i)}), i = 1, \ldots, n, T\}\right)$

initialize $\alpha_1, \alpha_2, \ldots, \alpha_n$ to some values;
for $t = 1, \ldots, T$
for $i = 1, \ldots, n$
if $(\text{Mistake Condition Expressed in } \alpha_j)$
Update α_j appropriately

Now look at the line "**Update α_j appropriately**" in the above algorithm. Remember that we express θ as

$$\theta = \sum_{j=1}^{n} \alpha_j y^{(j)} \phi(x^{(j)})$$

Assuming that there was a mistake in classifying the ith data point i.e.

$$y^{(i)}(\theta \cdot x^{(i)}) \leq 0$$

which of the following conditions about $\alpha_1, \ldots, \alpha_n$ is equivalent to

$$\theta = \theta + y^{(i)} \phi(x^{(i)}),$$

the update condition of the original algorithm?

- $\alpha_i = \alpha_i + 1$ ✔
- $\alpha_i = \alpha_i - 1$
- $\alpha_j = \alpha_j + 1$ for all $j \in 1, \ldots, n$

Solution:

Expand θ in the last equation and it turns out only α_i gets updated:

$$\alpha_i y^{(i)} \phi\left(x^{(i)}\right) + y^{(i)} \phi\left(x^{(i)}\right) = \left(\alpha_i + 1\right) y^{(i)} \phi\left(x^{(i)}\right).$$

Submit You have used 1 of 1 attempt

Answers are displayed within the problem

How the Kernel Perceptron Algorithm Works: The Mistake Condition

1/1 point (graded)

Kernel Perceptron $\left(\left\{\left(x^{(i)}, y^{(i)}\right), i = 1, \ldots, n, T\right\}\right)$
initialize $\alpha_1, \alpha_2, \ldots, \alpha_n$ to some values;
for $t = 1, \ldots, T$
for $i = 1, \ldots, n$
if $(\text{Mistake Condition Expressed in } \alpha_j)$
Update α_j appropriately

Now look at the line "**Mistake Condition Expressed in** α_j" in the above algorithm. Remember that we express θ as

$$\theta = \sum_{j=1}^{n} \alpha_j y^{(j)} \phi\left(x^{(j)}\right)$$

Which of the following conditions is equivalent to $y^{(i)}\left(\theta \cdot \phi\left(x^{(i)}\right)\right) \leq 0$? Remember from the video lecture above that given feature vectors $\phi(x)$ and $\phi(x')$, we define the Kernel function K as

$$K(x, x') = \phi(x)\, \phi(x').$$

- $y^{(i)} \sum_{j=1}^{n} \alpha_j y^{(j)} K\left(x^j, x^i\right) \leq 0$ ✔
- $y^{(i)} \sum_{j=1}^{n} \alpha_i y^{(j)} K\left(x^j, x^i\right) \leq 0$
- $y^{(i)} \sum_{j=1}^{n} \alpha_j y^{(i)} K\left(x^j, x^i\right) \leq 0$
- $y^{(i)} \sum_{j=1}^{n} \alpha_j y^{(j)} \phi\left(x^{(j)}\right) \leq 0$

Solution:

Substitute θ with $\sum_{j=1}^{n} \alpha_j y^{(j)} \phi\left(x^{(j)}\right)$ in $y^{(i)}\left(\theta \cdot \phi\left(x^{(i)}\right)\right) \leq 0$.

Submit You have used 1 of 1 attempt

Answers are displayed within the problem

Discussion

Show Discussion

Topic: Unit 2 Nonlinear Classification, Linear regression, Collaborative Filtering (2 weeks):Lecture 6. Nonlinear Classification / 5. The Kernel Perceptron Algorithm

© All Rights Reserved