6.034 Introduction to Artificial Intelligence

Machine learning and applications
Problems we will cover

• Computational biology
  - cancer classification
  - functional classification of genes

• Information retrieval
  - document classification/ranking

• Recommender systems
  - predicting user preferences (e.g., movies)
What are we trying to do?

- The goal is to find the right method for the right problem (matching task)

### Methods
- SVMs
- Boosting
- K-means
- ...

### Problems
- ...

This paper shows that the accuracy of learned text augmenting a small number of labeled training documents with a large pool of unlabeled classifiers can be improved by...
Cancer classification

• We’d like to automatically classify tissue samples according to whether there’s evidence of cancer or the type of tumor cells they contain

• What features to extract?
  - visual features due to different types of staining
  - how active different genes are in the cells (gene expression)
Gene expression

(Orphanides et al. 2002)
Measuring gene expression

- Basic cDNA micro-array technology

control sample (e.g., tumor)

cDNA microarray → Tissue profile

\[
\begin{bmatrix}
1.2 \\
0.5 \\
1.0 \\
2.3 \\
\ldots
\end{bmatrix}
\]

\[\text{genes}\]

Measuring gene expression

- Basic cDNA micro-array technology

control sample (e.g., tumor)

cDNA microarray

Tissue profile

\[
\begin{bmatrix}
0.18 \\
-0.69 \\
0.00 \\
0.83 \\
\ldots
\end{bmatrix}
\]

genes
Cancer classification

Tissues (with known tumor type)

Genes

(Golub et al. 1999)
Machine learning problem

Another way to motivate the choice of the maximal margin separator is to see that it reduces the "variance" of the hypothesis class. Recall that a hypothesis has large variance if small changes in the data result in a very different hypothesis. With a maximal margin separator, we can wiggle the data quite a bit without affecting the separator. Placing the separator very close to positive or negative points is a kind of overfitting; it makes your hypothesis very dependent on details of the input data.

Let's see if we can figure out how to find the separator with maximal margin as suggested by this picture.

First we have to define what we are trying to optimize. Clearly we want to use our old definition of margin, but we'll have to deal with a couple of issues first. Note that we're using the $w, b$ notation instead of $\bar{w}$ because we will end up giving $b$ special treatment in the future.

Remember that any scaling of $w$ and $b$ defines the same line; but it will result in different values of $\gamma$. To get the actual geometric distance from the point to the separator (called the geometric margin), we need to divide $\gamma$ through by the magnitude of $w$.

The next issue is that we have defined the margin for a point relative to a separator but we don't want to just maximize the margin of some particular single point. We want to focus on one point on each side of the separator, each of which is closest to the separator. And we want to place the separator so that the it is as far from these two points as possible. Then we will have the maximal margin between the two classes.

Since we have a degree of freedom in the magnitude of $w$ we're going to just define the margin for each of these points to be 1. (You can think of this 1 as having arbitrary units given by the magnitude of $w$.)

You might be worried that we can't possibly know which will be the two closest points until we know what the separator is. It's a reasonable worry, and we'll sort it out in a couple of slides.
Machine learning problem

- micro-array measurements are very noisy
- each training example is of very high dimension (e.g., ~ 10,000 genes)
- there are relatively few labeled tissue samples (only tens per class)
- some labels may be wrong
This one seems safer, no?

Another way to motivate the choice of the maximal margin separator is to see that it reduces the "variance" of the hypothesis class. Recall that a hypothesis has large variance if small changes in the data result in a very different hypothesis. With a maximal margin separator, we can wiggle the data quite a bit without affecting the separator. Placing the separator very close to positive or negative points is a kind of overfitting; it makes your hypothesis very dependent on details of the input data.

Let's see if we can figure out how to find the separator with maximal margin as suggested by this picture.

First we have to define what we are trying to optimize. Clearly we want to use our old definition of margin, but we'll have to deal with a couple of issues first. Note that we're using the $w, b$ notation instead of $\hat{w}$, because we will end up giving $b$ special treatment in the future.

Remember that any scaling of $w$ and $b$ defines the same line; but it will result in different values of $\gamma$. To get the actual geometric distance from the point to the separator (called the geometric margin), we need to divide $\gamma$ through by the magnitude of $w$.

The next issue is that we have defined the margin for a point relative to a separator but we don't want to just maximize the margin of some particular single point. We want to focus on one point on each side of the separator, each of which is closest to the separator. And we want to place the separator so that it is as far from these two points as possible. Then we will have the maximal margin between the two classes.

Since we have a degree of freedom in the magnitude of $w$ we're going to just define the margin for each of these points to be 1. (You can think of this 1 as having arbitrary units given by the magnitude of $w$.)

You might be worried that we can't possibly know which will be the two closest points until we know what the separator is. It's a reasonable worry, and we'll sort it out in a couple of slides.
SVM training

- SVMs are trained by solving a quadratic programming problem

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \\
\text{subject to} & \quad \alpha_i \geq 0, \quad \sum_{i=1}^{n} y_i \alpha_i = 0 \\
(\text{where is } w_0?)
\end{align*}
\]
Back to the problem

- High dimensionality $\Rightarrow$ linear kernel
  \[ K(x_i, x_j) = (x_i^T x_j + 1) \]
- Noise in the measurements $\Rightarrow$ feature selection (use only a relevant subset of the genes)
- Outliers $\Rightarrow$ adjust the kernel to increase resistance to outliers
Feature selection / ranking

- We can rank genes according to how much they seem to be related to the classification task

\[ R(\text{gene}_i) = \frac{\mu_i^+ - \mu_i^-}{\sigma^+ + \sigma^-} \]

- mean value across +1 tissues
- mean value across -1 tissues
- stdv across +1 tissues
- stdv across -1 tissues
• Suppose the expression levels of all the 10,000 genes in each tissue sample are drawn at random from some distribution (e.g., normal)

• Based on 5 such expression vectors for each class, can we find a gene that is perfectly correlated with the labels?

• The chance of this happening is 100%

• What if we have had instead 10 such vectors per class? The probability drops to 1%
Dealing with outliers

- We should make the linear decision boundary resistant to outliers (e.g., due to mislabeled samples)
Dealing with outliers

• One way to increase resistance to outliers is to add a diagonal term to the kernel function so that each example appears more similar to itself than before.

\[
K \leftarrow \begin{bmatrix}
K(x_1, x_1) + \lambda & \cdots & K(x_1, x_n) \\
\cdots & \cdots & \cdots \\
K(x_n, x_1) & \cdots & K(x_n, x_n) + \lambda
\end{bmatrix}
\]
The effect of lambda

\[ \lambda = 0 \]
The effect of lambda

\[ \lambda = 2 \]
The effect of lambda

\[ \lambda = 4 \]
The effect of lambda

\[ \lambda = 8 \]
The effect of lambda

\[ \lambda = 16 \]
Results

- AML vs MML distinction
  - training set: 27 ALL and 11 AML
  - test set: 20 ALL and 14 AML

- The SVM classifier achieves perfect classification of the test samples
Problems we will cover

• Computational biology
  - cancer classification
  - functional classification of genes

• Information retrieval
  - document classification/ranking

• Recommender systems
  - predicting user preferences (e.g., movies)
Functional classification of genes

• We don’t know what most genes do
• Given known roles for some genes, we would like to predict the function of all the remaining genes

ribosomal genes

unannotated “genes”

\[
\begin{align*}
F2N1.3 \\
T18A10.9 \\
F5J6.12 \\
\ldots
\end{align*}
\]

\[
\begin{align*}
YLA003W \\
YPL037C \\
\ldots
\end{align*}
\]
How can only ~20,000 genes specify a complex mammal?
How can only ~20,000 genes specify a complex mammal?
How can only ~20,000 genes specify a complex mammal?

Machine learning problem

- Dimensionality no longer very high (# of tissue samples/conditions)
- Can use other kernels, e.g., radial basis kernel
- New problem: there are much more negatively labeled genes than positive
Imbalanced classes

\[ \lambda = 2 \]
Imbalanced classes

\[ \lambda = 4 \]
Imbalanced classes

\[ \lambda = 8 \]
Imbalanced classes

\[ \lambda = 16 \]
Imbalanced classes

- In order to ensure that the classifier pays attention to the positive class, we increase (proportionally) resistance to negative examples

\[
K \leftarrow \begin{bmatrix}
K(x_1, x_1) + \lambda \left(\frac{n^+}{n}\right) & \cdots & K(x_1, x_n) \\
\cdots & \cdots & \cdots \\
K(x_n, x_1) & \cdots & K(x_n, x_n) + \lambda \left(\frac{n^-}{n}\right)
\end{bmatrix}
\]

freq. of positive examples  freq. of negative examples
Differential resistance
Differential resistance

\[ \lambda = 1 \]
Functional annotation of genes

• SVMs perform very well (though there are other comparable methods)

• Learning methods can identify incorrectly annotated genes, predict functional roles for uncharacterized genes, as well as guide further experimental effort

• Used in many contexts; based on profiles, text, and/or sequence
  - e.g., understanding developmental roles of genes (lineage specific genes)
  - etc.
Problems we will cover

• Computational biology
  - cancer classification
  - functional classification of genes

• Information retrieval
  - document classification/ranking

• Recommender systems
  - predicting user preferences (e.g., movies)