CSE 151 Machine Learning
Instructor: Kamalika Chaudhuri
**Expectation**

For a discrete random variable $X$, the expectation is defined as:

$$E[X] = \sum_x x f_X(x)$$

For a continuous random variable $X$, the expectation is defined as:

$$E[X] = \int_{-\infty}^{\infty} x f_X(x) \, dx$$

Let $Y = r(X)$. Then $E[Y]$ can be computed as:

$$E[Y] = \int_{-\infty}^{\infty} r(x) f_X(x) \, dx \quad \text{continuous}$$

$$E[Y] = \sum_x r(x) f_X(x) \quad \text{discrete}$$
Variance

For a random variable $X$, the variance is defined as:

$$Var(X) = E[(X - E[X])^2]$$

**Property:**

$$E[(X - E[X])^2] = E[X^2] - (E[X])^2$$

Exercise: Prove the property
Independence of random variables, Covariance

Random variables $X$ and $Y$ are independent if for any two sets $A$ and $B$,
\[ P(X \in A, Y \in B) = P(X \in A)P(Y \in B) \]

For two random variables $X, Y$,
\[ \text{cov}(X,Y) = E( (X - E[X]) (Y - E[Y]) ) \]
\[ \text{cov}(X,Y) \text{ can also be written as: } \text{cov}(X,Y) = E(XY) - E(X) E(Y) \]

**Property:** If $X$ and $Y$ are independent, then
\[ \text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y) \]
\[ \text{Cov}(X,Y) = 0 \]

Does the converse hold?
Announcements

• Homework 1 due on Wed Apr 18
• Midterm 1 is on Mon Apr 23
• Material for midterm:
  • Linear algebra review, probability review
  • k-nearest neighbors
Classification

Given labeled data:

\[(x_i, y_i) \quad i = 1, \ldots, n\]

where \(y\) is \textit{discrete}, find a rule to predict \(y\) values for \textit{unseen} \(x\) feature vector and label.
Typical Classification Algorithm

Set of input examples \((x_i, y_i)\) → Classification Algorithm → Prediction Rule

New example \(x\) → Label \(y\)
Typical Classification Algorithm

Set of input examples \((x_i, y_i)\)

Classification Algorithm

Test Data

New example \(x\)

Prediction Rule

Label \(y\)

Training and test data must be **separate**!
Classification

Given labeled data \((x_i, y_i), i=1,..,n,\) where \(y\) is discrete, predict \(y\) values for unseen \(x\)

**Example 1:** Which digit in the image?

\[
\begin{array}{cccc}
0 & 1 & 2 & 3 \\
5 & 6 & 7 & 8 \\
4 & & & \\
\end{array}
\]

Label: 0,1,..,9

What are the features?

A **multiclass** classification problem
Classification

Given labeled data \((x_i, y_i), i=1,\ldots,n,\)
where \(y\) is discrete, predict \(y\) values for unseen \(x\)

**Example 1:** Which digit in the image?

Label: 0, 1, .., 9

What are the features?
Vector of pixel colors

Image

\[
\begin{array}{cccccc}
0 & 1 & 2 & 3 & 4 \\
5 & 6 & 7 & 8 & 9
\end{array}
\]

\[
\begin{array}{c}
0 \\
0
\end{array}
\] (0 for white, 1 for black)

\[
\begin{array}{cccccc}
0 & 0 & 1 & 0 & 1 & 0 & 1 \\
\ldots
\end{array}
\]
Performance Measures

Training error:
For a classifier $f$, given training data $(x_1, y_1), \ldots, (x_m, y_m)$

$$\text{Training error} = \frac{1}{m} \sum_{j=1}^{m} 1[f(x_j) \neq y_j]$$
Performance Measures

**Training error:**
For a classifier $f$, given *training* data $(x_1, y_1), \ldots, (x_m, y_m)$

$$\text{Training error} = \frac{1}{m} \sum_{j=1}^{m} 1[f(x_j) \neq y_j]$$

**Test error:**
For a classifier $f$, given *test* data $(x_1, y_1), \ldots, (x_m, y_m)$

$$\text{Test error} = \frac{1}{m} \sum_{j=1}^{m} 1[f(x_j) \neq y_j]$$
Performance Measures

**Training error:**
For a classifier \( f \), given training data \((x_1, y_1),..(x_m, y_m)\)

\[
\text{Training error} = \frac{1}{m} \sum_{j=1}^{m} 1[f(x_j) \neq y_j]
\]

**Test error:**
For a classifier \( f \), given test data \((x_1, y_1),..(x_m, y_m)\)

\[
\text{Test error} = \frac{1}{m} \sum_{j=1}^{m} 1[f(x_j) \neq y_j]
\]

Both are proxy for the **true error** of the classifier:

\[
\text{True error} = \mathbb{E}_{(X,Y) \sim D}[1[f(X) \neq Y]]
\]

where data (both training and test) are drawn from an underlying distribution \( D \)
Performance Measures

Training error:
For a classifier $f$, given training data $(x_1, y_1), \ldots, (x_m, y_m)$

$$\text{Training error} = \frac{1}{m} \sum_{j=1}^{m} 1[f(x_j) \neq y_j]$$

Test error:
For a classifier $f$, given test data $(x_1, y_1), \ldots, (x_m, y_m)$

$$\text{Test error} = \frac{1}{m} \sum_{j=1}^{m} 1[f(x_j) \neq y_j]$$

Usually, training error is less than test error.
Test error is a better measure than training error.
The Nearest Neighbor Classifier

**Given:** Labelled examples \((x_1, y_1), \ldots, (x_n, y_n)\), Predict label of a new example \(x\)

**Solution:** \(j = \arg\min_i d(x, x_i)\)  
Return \(y_j\)  
d is the distance
The Nearest Neighbor Classifier

**Given:** Labelled examples \((x_1, y_1), \ldots, (x_n, y_n)\),
Predict label of a new example \(x\)

**Solution:** \(j = \arg\min_i d(x, x_i)\)
Return \(y_j\)

d is the distance

- Original image: 
- Predicted label: 
- Another example: 
- Predicted label: 
Example 1

<table>
<thead>
<tr>
<th>x</th>
<th>Training Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

What does the classifier look like?
Example 1

Training Data

What does the classifier look like?

**Decision Boundary:**
Boundary between regions corresponding to different classes
The Nearest Neighbor Classifier

Assign to a test example the same label as its closest training example

- Red: training points
- Blue cell: region with same label as the training example in it

Voronoi diagram
Example 2

Training Data:

( (1, 1), 1)  
( (-1, -1), 1)  
( (1, -1), 2)  
( (-1, 1), 2)
Example 2

Training Data:
- (1, 1), 1
- (-1, -1), 1
- (1, -1), 2
- (-1, 1), 2
Example 3

Two classes: + and o
Classifier: Green region as o, White region as +
$1 \text{ NN Error: 6\%}$
When does 1 NN work?

Each label is a contiguous region of space. 1-NN works well in the interior, not so well close to the decision boundary.
When does it not work so well?

- Truth
- Training Data

Diagram: Red dots represent missing data, blue dots represent present data.
When does it not work so well?
How to make it more robust?

Find the k-nearest neighbors, and output the majority of their labels

**Question:** Suppose we are in a binary classification problem, and a close neighbor of q has the wrong label with probability p=0.3.

(a) What is the probability that 1-NN is correct on q?
(b) What is the probability that 3-NN is correct on q?
(c) What is the probability that \((2m+1)\)-NN is correct on q for integer m?

[Assume that each training data point is drawn independently]
k-Nearest Neighbor Classifier

**Given:** Labelled examples \((x_1, y_1), \ldots, (x_n, y_n)\),
Predict label of a new example \(x\)

**Solution:** Find \(j_1, \ldots, j_k\), indices of \(k\) closest neighbors of \(x\) in \(\{x_1, \ldots, x_n\}\)
Return majority\((y_1, \ldots, y_k)\)