Lecture 3

Spring 2013

Three Topics:

(I) Receiver Operator Characteristic (ROC) curves
   Precision and Recall Curves.
   * What if we do not fix the loss function?

(II) The Curse of Dimensionality
   * Why human intuition about geometry are misleading in high-dimensional spaces.
   * Why we do not have enough data in high-dimensions. And what we can do about it

III The Bias-Variance Dilemma
   * A classic statistical perspective about generalization.
(2) **Topic I**: Precision/Recall and ROC curve.

What if we do not know the loss function? Or if we want a more “sophisticated” decision process, i.e., we don’t want to diagnose people as “cancer” or “non-cancer.” Instead we want to separate them into groups for further testing.

**Example: Detecting Cats in the Pascal Challenge**

*Data:* 20,000 images

1,000 cats.

Images can contain 0 cats, 1 cat, or more cats.

The positions of cats are specified by bounding boxes surrounding them.

**Task:** Determine the boxes which contain cats.

- There are 1,000 boxes that contain cats.
- There are $20,000 \times 1,000 = 20,000,000$ total boxes.
- There are $\frac{1,000}{20,000} = 0.05$ boxes containing cats per image.

So, many more “negatives” (boxes without cats) than “positives” (boxes with cats).
Suppose the dataset has \( n_1 \) targets and \( n_2 \) background - i.e. \( \{(x_i, y_i): i = 1 \leq N\} \)

\[
\begin{align*}
n_1 &= \sum_{i=1}^{N} I(y_i = 1) \\
n_2 &= \sum_{i=1}^{N} I(y_i = -1)
\end{align*}
\]

We have a set of decision rules \( \delta_T(\cdot) \)
which are parameterized by a threshold \( T \).

i.e. each decision rule is of form

\[
\delta_T(x) = \begin{cases} 
1 & \text{if} \quad \hat{f}(x) > T \\
-1 & \text{if} \quad \hat{f}(x) < T
\end{cases}
\]

Let \( m_1^T \) = true +ve with \( \delta_T = \) number of targets (rats) detected by \( \delta_T \)

\( m_2^T \) = false +ve with \( \delta_T = \) number of backgrounds (non-rats) detected (incorrectly) by \( \delta_T \)

i.e.

\[
\begin{align*}
m_1^T &= \sum_{i=1}^{N} I(\delta_T(x_i) = 1) I(y_i = 1) \\
m_2^T &= \sum_{i=1}^{N} I(\delta_T(x_i) = -1) I(y_i = -1)
\end{align*}
\]

Precision - at threshold \( T = \frac{m_1^T}{m_1^T + m_2^T} = \frac{\text{No. True +ve's}}{\text{No. +ve's}} \)

Recall - at threshold \( T = \frac{m_1^T}{n_1^T} = \frac{\text{Proportion of Targy that are detected}}{\text{No. of Targy}} \)

\[
\begin{align*}
\text{Precision} & \uparrow \\
\text{Recall} & \downarrow \\
\text{Typical shape} & \approx \text{Plot curve - each point corresponds to} \\
\text{Precision/Recall at a value of } T & \text{Trade-off: High Precision \& Low Recall} \\
\text{Low Precision \& High Recall}
\end{align*}
\]
ROC curves

Bayes Decision Theory. \( R(x) = \sum L(x|y) P(x|y) \)

For binary \( y \in Y \) (i.e., \( y \in \{\pm 1\} \), \( x \)

The Bayes Rule \( t = \arg \max_{y \in Y} K(x) \)

reduces to thresholding the log-likelihood ratio, i.e., it is of form:

\[ L_t(x) = 1, \] \( \text{if} \) \( \log \frac{P(y=1|x)}{P(y=-1|x)} > T \)

\[ L_t(x) = -1, \] otherwise.

The threshold \( T \) is a function of the prior \( P(y) \)
and the loss function \( L(x|y) \). Hence changing the prior, or the loss function, corresponds to changing \( T \).

So \( \log \frac{P(y=1|x)}{P(y=-1|x)} \) is an example of the function \( f(x) \) (previous page).

Changing \( T \) will alter the false-\( +ve \)'s, the true-\( -ve \)'s, the false-\( -ve \)'s, the true-\( +ve \)'s.
ROC curve. Plot \( P(\alpha_T=1 \mid y=1) \)

\[ P(\alpha_T=1 \mid y=1) = \sum_{x} P(\alpha_T(x)=1 \mid x) P(x \mid y=1) \]

\[ = \sum_{x} \frac{P(x \mid y=1)}{P(x \mid y=1)} \]

\( \tilde{y}_i : \log \frac{P(x,y=1)}{P(x,y=1)} > T \)

\( P(\alpha_T=-1 \mid y=1) = \sum_{x} P(\alpha_T(x)=-1 \mid x) P(x \mid y=-1) \)

\[ = \sum_{x} \frac{P(x \mid y=-1)}{P(x \mid y=-1)} \]

\( \tilde{y}_i : \log \frac{P(x,y=-1)}{P(x,y=-1)} > T \)

Each point on the curve comes from a value of \( T \).

Proportion of true negatives

\( \Rightarrow T = \infty \)

\( \Rightarrow P(\alpha_T=1 \mid y=-1) \rightarrow 0 \) false negatives

\( \Rightarrow \) false positives

Rule:

\( \alpha(x) = \begin{cases} 1 & \text{if } \frac{\log P(x,y=1)}{P(x,y=-1)} > T \\ 0 & \text{otherwise} \end{cases} \)

So if \( T = \infty \), then all data is classified as positive.

\( \Rightarrow \) \( P(\alpha_T=1 \mid y=-1) = P(\alpha_T=1 \mid y=1) = 1 \)

\( \Rightarrow \) \( T = \infty \), all data is classified as negative. \( P(\alpha_T=1 \mid y=-1) = \Phi(\tilde{y}_i) \).

Bayes decision is given by a specific point \( T^* \) on the curve.
The Curse of Dimensionality

The examples of Bayes Decision theory are misleading because they are given in low-dimensional spaces (1-dim, or 2-dim)

Many pattern classification tasks occur in high dimensional spaces. In these spaces our geometric intuitions are often wrong.

E6. Consider the volume of a sphere of radius \( r = 1 \) in \( D \) dimensions. What fraction of its volume lies in the region between \( 1 - \varepsilon < r < 1 \)?

\[
V_D(r) = K_D r^D
\]

\[
V_D(1) - V_D(1 - \varepsilon) = 1 - (1 - \varepsilon)^D
\]

\[
\frac{V_D(1) - V_D(1 - \varepsilon)}{V_D(1)}
\]

For large \( D \), the volume fraction tends to 1 even for small \( \varepsilon \). Most of the volume is at the boundary.

\[
\ln l \propto -e^{-x^2/2\delta^2}
\]

\[
\ln Z = \frac{1}{\sqrt{2\pi} \delta} e^{-x^2/(2\delta^2)}
\]

\[
P(x_1, x_2) = \frac{1}{Z} e^{-\frac{(x_1^2 + x_2^2)}{2\delta^2}}
\]

Let \( r = \sqrt{x_1^2 + x_2^2} \).

Then \( P(r) = \frac{r}{2\pi \delta^2} e^{-\frac{r^2}{2\delta^2}} \).

In higher dimensions:

\[
P(r) = r^{D-1} e^{-\frac{r^2}{2\delta^2}}
\]

So in high dimensions, most of the probability mass of the Gaussian is concentrated on a thin shell away from the center of the Gaussian.
Learning probability distribution in high dimensions can require a lot of data. E.g., Gaussian distribution in D dimensions.

- Mean: \( \mu \) in D dimensions.
- Covariance: \( \Sigma = \frac{1}{D(D+1)} \) in D dimensions.

This is \( O(D^2) \), not too bad.

But suppose we represent the data by a histogram with B bins per dimension.

- \( B \) bins in \( D = 1 \)
- \( B^2 \) bins in \( D = 2 \)
- \( B^D \) bins in \( D \) dimensions.

Exponential growth:

Requires exponential amount of data to learn the distribution.
How to deal with the curse of dimensionality?

In practice, data typically lies on some low-dimensional surface in the high dimensional space.

So the effective dimension of the data may be a lot smaller than the dimension of the space.

- Dimension Reduction Methods attempt to reduce the dimension by seeking the low-dimensional surface. (Not always easy!)
- Modeling, if we can guess distributions for the data (e.g. Gaussian) then dependence on the dimension is not too bad.
- Concentrate on the decision boundary — there may be enough data to learn the decision boundary even if we cannot learn the distributions.
Bias and Variance

This is a classical statistics perspective on generalization. First we need to introduce some statistics terminology.

Suppose we want to estimate a continuous quantity \( \theta \) — e.g., the mean/variance of a Gaussian distribution (more about this in the next lecture), or the parameters of a regression line (see below) — then statisticians use an estimator.

The estimator is based on a set \( X = \{ x_i; i = 1 \text{ to } N \} \) of examples — drawn from an unknown distribution \( P(x) \).

\[
P(X) = \prod_{i=1}^{N} P(x_i)
\]

The task is to estimate a property \( \theta \) by an estimator \( \hat{\theta} = g(X) \). E.g., like a classification rule, but based on the set \( X \), and \( \theta \) is continuous.

For example: let \( \theta = (\mu, \sigma) \) be the mean and variance of the data (data is one-dimensional in this example), then

\[
\hat{\mu}(X) = \frac{1}{N} \sum_{i=1}^{N} x_i, \quad \hat{\sigma}^2(X) = \frac{1}{N} \sum_{i=1}^{N} x_i - \hat{\mu}(X)^2
\]

\[
g(X) = (\hat{\mu}(X), \hat{\sigma}^2(X))
\]

Note: that the estimator is a function of the set \( X \), this will be important later.
To evaluate the estimator, we want to measure how much it differs from the correct $\theta$. It is attractive to use a quadratic error (this helps the analysis) $(g(x) - \theta)^2$ - but this depends on the data set $X$.

So we need to get the expected error with respect to the set $X$,

$$\Gamma(g, \theta) = \mathbb{E}_X (g(x) - \theta)^2 = \int (g(x) - \theta)^2 p(x) dx$$

**Mean square error.**

$$b \theta(g) = \mathbb{E}_X g(x) - \theta$$, bias of estimator

If $b \theta(g) = 0$ for all $\theta$, then $g$ is an unbiased estimator of $\theta$.

E.g. consider $\hat{\mu}(x) = \frac{1}{N} \sum_{i=1}^{N} x_i$ estimator of the mean.

$$\mathbb{E}_X \hat{\mu}(x) = \mathbb{E}_X \left[ \frac{1}{N} \sum_{i=1}^{N} x_i \right] = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_X x_i$$

$$= \frac{1}{N} N \mu = \mu$$

Hence $\hat{\mu}(x)$ is an unbiased estimator of $\mu$. mean of the distribution that generated the data.

($\hat{\mu}(x)$ will depend on $X$, but on average it gives you the right answer). We can compute the variance of the estimator - i.e., how much it varies in estimating $\mu$. 

...
Process $P(x)$

$X_1 = \left\langle x_1, \ldots, x_n \right\rangle$

$X_2 = \left\langle x_{n+1}, \ldots, x_{2n} \right\rangle$

$X_3 = \left\langle x_{2n+1}, \ldots, x_{3n} \right\rangle$

possible datasets

g(x_1), g(x_2), g(x_3),$ these will vary.

Calculate the mean $\Rightarrow \text{e.g. } E_X(g(x))$

Calculate the variance $\Rightarrow \text{Var}(d(x)) = E_X\left\{ \left( g(x) - E_X(g(x)) \right)^2 \right\}$.

For the estimator $\hat{\mu}(x)$ of the mean, we know that $E_X(\hat{\mu}(x)) = \mu$ unbiased previous page.

$\text{Var}_X(\hat{\mu}) = \text{Var}_X\left( \frac{1}{n} \sum_{i=1}^{n} x_i \right) = \frac{1}{N} \sum_{i=1}^{N} \text{Var}_x(x_i) \overset{\text{p.i.i.d.}}{=} \frac{N \sigma^2}{N} = \frac{\sigma^2}{N}$

Hence, the variance of the estimator tends to zero as $N \to \infty$ will fall off rate $1/N$.

Note: this $1/N$ fall-off rate is true for any linear estimator - i.e. $g(x)$ is a linear function of the elements $x_1, \ldots, x_n$ in set $X$.

Next, consider the bias of $\hat{\sigma}^2(x)$ $\Rightarrow \frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{\mu})^2$

By similar analysis (to analysis of $\hat{\mu}(x)$) we find that

$E_X[\hat{\sigma}^2(x)] = \frac{1}{N} \left( \sum_{i=1}^{N} E_x(x_i^2) - N \mu^2 \right) = \frac{N-1}{N} \left( \mu^2 \right) = \frac{\mu^2 - \sigma^2}{N}$

(cuses $\sum_{i=1}^{N} (x_i - \mu)^2 = \sum_{i=1}^{N} x_i^2 - N \mu^2$) biased, but asymptotically unbiased.
\[(13) \quad \text{Bias Variance Dilemma: Adjusting Model Complexity}\]

Dataset \(X = \{(x_i, y_i) : i = 1 \text{ to } n\}\) sampled from \(P(x, y) = p(y|x)p(x)\) \(y\) is continuous valued.

Let \(g(x)\) be an estimator of \(y\)

Claim 1: \[\frac{\langle (y - g(x))^2 \rangle_{P(y|x)}}{P(y|x)} = (y - \langle y \rangle_{P(y|x)}^2) + (\langle y \rangle_{P(y|x)} - g(x))^2\]

Here \(\langle f(y, x) \rangle_{P(y|x)} = \frac{\sum f(y, x)P(y|x)}{\sum P(y|x)}\)

Proof: Write \((y - g(x))^2 = (y - \langle y \rangle_{P(y|x)})^2 + (\langle y \rangle_{P(y|x)} - g(x))^2\)

\[= (y - \langle y \rangle_{P(y|x)})^2 + (\langle y \rangle_{P(y|x)} - g(x))^2 + 2(y - \langle y \rangle_{P(y|x)})(\langle y \rangle_{P(y|x)} - g(x))\]

Take expectations with respect to \(P(y|x)\)

\[\langle (y - g(x))^2 \rangle_{P(y|x)} = \langle (y - \langle y \rangle_{P(y|x)})^2 \rangle + \langle (\langle y \rangle_{P(y|x)} - g(x))^2 \rangle\]

because \(\langle (\langle y \rangle_{P(y|x)} - g(x))^2 \rangle\) is independent of \(y\) and \(\langle (y - \langle y \rangle_{P(y|x)})^2 \rangle\) depends on \(g(.)\) and the data \(X\)

\[\langle (y \rangle_{P(y|x)} - g(x))^2 \]
Next we study the expectation \( \mathbb{E}( (y_{\text{pred}} - g(x))^2 \) with respect to \( P(x) \)

i.e., how it depends on the particular sample \( X = (x_1, \ldots, x_n) \) from \( P(x) \).

Claim II \[
\mathbb{E}( (y_{\text{pred}} - g(x))^2 \mid P(x) = \mathbb{E}( (y_{\text{pred}} - \mathbb{E}(y_{\text{pred}} \mid P(x))^2 + \mathbb{E}(g(x) - \mathbb{E}(g(x) \mid P(x))^2)
\]

Result: Bias - Variance

The first term depends on the \textit{bias} - the difference between the best estimate \( \mathbb{E}(y_{\text{pred}} \mid P(x) \) (if we knew the distribution) and the expectation of our estimate \( g(x) \).

The second term is the \textit{variance} of the estimate \( g(x) \)

- i.e., how much it depends on the sample set \( X \) (no label)

Proof: Write \[
(\mathbb{E}(y_{\text{pred}} - g(x))^2
= (\mathbb{E}(y_{\text{pred}} - \mathbb{E}(y_{\text{pred}}) + \mathbb{E}(y_{\text{pred}}) - g(x))^2
= (\mathbb{E}(y_{\text{pred}} - \mathbb{E}(y_{\text{pred}}))^2 + (\mathbb{E}(y_{\text{pred}}) - g(x))^2
+ 2(\mathbb{E}(y_{\text{pred}} - \mathbb{E}(y_{\text{pred}})(\mathbb{E}(y_{\text{pred}}) - g(x))

Take expectation w.r.t. \( P(x) \)

\[
\mathbb{E}( (y_{\text{pred}} - g(x))^2 \mid P(x) = \mathbb{E}( (y_{\text{pred}} - \mathbb{E}(y_{\text{pred}}))^2 + \mathbb{E}(g(x) - \mathbb{E}(g(x))^2)

\]

Because \( \mathbb{E}(y_{\text{pred}} - g(x))^2 \mid P(x) \) is independent of \( X \)

and \( \mathbb{E}(g(x) - \mathbb{E}(g(x))^2 \mid P(x) \)
What does this mean? Spring 2013.

Distribution $P(x)$

Dataset $X_1 = \langle x_1, \ldots, x_n \rangle$, $X_2 = \langle x_{n+1}, \ldots, x_{2n} \rangle$, \ldots, $X_m = \langle x_{m(n-1)+1}, \ldots, x_{mn} \rangle$

For each data we get an estimate of $y$:

$g(x_1), g(x_2), \ldots, g(x_m)$

The mean estimate is $\bar{g} = \frac{1}{m} \sum_{i=1}^{m} g(x_i)$

the variance is $\text{Var}(g) = \frac{1}{m} \sum_{i=1}^{m} (g(x_i) - \bar{g})^2$

To get good generalization, we want the variance to be small, so that it is not sensitive to the data we have trained the classifier on.

Ideally we want to have a classifier $g(\cdot)$ which has small bias and variance.

In practice, there is often a trade-off between bias and variance.

A complex classifier can give a good fit to the data (compared to a simple classifier) but can have high variance because it over-fits the data. So it gives different results on different datasets.
The data is generated by:
\[ f(x) = 2 \sin((1.5)x) \]
\[ \epsilon \sim \mathcal{N}(0,1) \]

Fit an order 1 polynomial (straight line) to the data...

Fit order 3 poly. Better but more variance

A more complex model gives better fit to the data (i.e., to underlying model)
\[ \rightarrow \text{reduces bias} \]

But small changes in dataset lead to bigger variability in fitted model
\[ \rightarrow \text{increases variance.} \]

Low orders - risk of underfitting
High orders - risk of overfitting

To get a small error, we should have the proper inductive bias and have large enough dataset so that variability is constrained by data.

Note: take very high-variance models, use average (lale)
What to do? How to test generalization? Spring 2013

Cross-validation → divide dataset into two parts as training & validation set.
Train models of different complexity and test their error on the validation set.
As model complexity increases, error on training set decreases. But error on validation set decreases too.

\[ \text{regularization} \]
\[ \text{augmented error function} \]
\[ \varepsilon' = \text{error on data} + \lambda \cdot \text{model complexity} \]
(A optimized using cross-validation)

Structural risk minimization (Vapnik) (VC dim)
- also penalize model complexity.

Minimum Description Length (Rissanen)
Penalize complexity by cost of encoding model.

Bayesian Model Selection: if some prior knowledge
\[ P(\text{model} | \text{data}) = \frac{P(\text{data} | \text{model}) P(\text{model})}{P(\text{data})} \]
(gives higher prob to simpler models)