CSCI 590: Machine Learning

Lecture 7: Nonparametric methods, K-nearest neighbor classification, linear regression

Instructor: Murat Dundar

Acknowledgement:
Some of these slides are taken from course textbook website
http://research.microsoft.com/~cmbishop/prml/
Nonparametric Methods (1)

Parametric distribution models are restricted to specific forms, which may not always be suitable; for example, consider modelling a multimodal distribution with a single, unimodal model.

Nonparametric approaches make few assumptions about the overall shape of the distribution being modelled.
Nonparametric Methods (2)

**Histogram methods** partition the data space into distinct bins with widths $\Delta_i$ and count the number of observations, $n_i$, in each bin.

$$p_i = \frac{n_i}{N \Delta_i}$$

- Often, the same width is used for all bins, $\Delta_i = \Delta$.
- $\Delta$ acts as a smoothing parameter.

- In a $D$-dimensional space, using $M$ bins in each dimension will require $M^D$ bins!
Nonparametric Methods (3)

Assume observations drawn from a density $p(x)$ and consider a small region $\mathcal{R}$ containing $x$ such that

$$ P = \int_{\mathcal{R}} p(x) \, dx. $$

The probability that $K$ out of $N$ observations lie inside $\mathcal{R}$ is $\text{Bin}(K|N,P)$ and if $N$ is large

$$ K \sim NP. $$

If the volume of $\mathcal{R}$, $V$, is sufficiently small, $p(x)$ is approximately constant over $\mathcal{R}$ and

$$ P \simeq p(x)V $$

Thus

$$ p(x) = \frac{K}{NV}. $$
Kernel Density Estimation: fix $V$, estimate $K$ from the data. Let $R$ be a hypercube centred on $x$ and define the kernel function (Parzen window)

$$k((x - x_n)/h) = \begin{cases} 1, & |(x_i - x_{ni})/h| \leq 1/2, \\ 0, & \text{otherwise.} \end{cases}$$

It follows that

$$K = \sum_{n=1}^{N} k \left( \frac{x - x_n}{h} \right) \text{ and hence } p(x) = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{h^D} k \left( \frac{x - x_n}{h} \right).$$
To avoid discontinuities in \( p(x) \), use a smooth kernel, e.g. a Gaussian

\[
p(x) = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{(2\pi h^2)^{D/2}} \exp \left\{ -\frac{\| \mathbf{x} - \mathbf{x}_n \|^2}{2h^2} \right\}
\]

Any kernel such that

\[
k(u) \geq 0, \quad \int k(u) \, du = 1
\]

will work.
Nonparametric Methods (6)

Nearest Neighbour Density Estimation: fix $K$, estimate $V$ from the data. Consider a hypersphere centred on $x$ and let it grow to a volume, $V^*$, that includes $K$ of the given $N$ data points. Then

$$p(x) \simeq \frac{K}{NV^*}.$$
Nonparametric Models (7)

Nonparametric models (not histograms) requires storing and computing with the entire data set.

Parametric models, once fitted, are much more efficient in terms of storage and computation.
$K$-Nearest-Neighbours for Classification (1)

Given a data set with $N_k$ data points from class $\mathcal{C}_k$ and $\sum_k N_k = N$, we have

$$p(x) = \frac{K}{NV}$$

and correspondingly

$$p(x|\mathcal{C}_k) = \frac{K_k}{N_k V}.$$ 

Since $p(\mathcal{C}_k) = N_k/N$, Bayes’ theorem gives

$$p(\mathcal{C}_k|x) = \frac{p(x|\mathcal{C}_k)p(\mathcal{C}_k)}{p(x)} = \frac{K_k}{K}.$$
$K$-Nearest-Neighbours for Classification (2)

$K = 3$  

$K = 1$
K-Nearest-Neighbours for Classification (3)

- K acts as a smoother
- For $N \to \infty$, the error rate of the 1-nearest-neighbour classifier is never more than twice the optimal error (obtained from the true conditional class distributions).
Linear Basis Function Models (1)

Example: Polynomial Curve Fitting

\[ y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j \]
Linear Basis Function Models (2)

Generally

\[ y(x, w) = \sum_{j=0}^{M-1} w_j \phi_j(x) = w^T \phi(x) \]

where \( \phi_j(x) \) are known as basis functions.

Typically, \( \phi_0(x) = 1 \), so that \( w_0 \) acts as a bias.

In the simplest case, we use linear basis functions : \( \phi_d(x) = x_d \).
Polynomial basis functions:

$$\phi_j(x) = x^j.$$ 

These are global; a small change in $x$ affect all basis functions.
Linear Basis Function Models (4)

Gaussian basis functions:

\[ \phi_j(x) = \exp \left\{ -\frac{(x - \mu_j)^2}{2s^2} \right\} \]

These are local; a small change in \( x \) only affect nearby basis functions. \( \mu_j \) and \( s \) control location and scale (width).
Sigmoidal basis functions:

\[
\phi_j(x) = \sigma \left( \frac{x - \mu_j}{s} \right)
\]

where

\[
\sigma(a) = \frac{1}{1 + \exp(-a)}.
\]

Also these are local; a small change in \( x \) only affect nearby basis functions. \( \mu_j \) and \( s \) control location and scale (slope).
Maximum Likelihood and Least Squares (1)

Assume observations from a deterministic function with added Gaussian noise:

\[ t = y(x, w) + \epsilon \quad \text{where} \quad p(\epsilon | \beta) = \mathcal{N}(\epsilon | 0, \beta^{-1}) \]

which is the same as saying,

\[ p(t | x, w, \beta) = \mathcal{N}(t | y(x, w), \beta^{-1}). \]

Given observed inputs, \( X = \{x_1, \ldots, x_N\} \), and targets, \( t = [t_1, \ldots, t_N]^T \), we obtain the likelihood function

\[ p(t | X, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n | w^T \phi(x_n), \beta^{-1}). \]
Taking the logarithm, we get

\[ \ln p(t|w, \beta) = \sum_{n=1}^{N} \ln \mathcal{N}(t_n|w^T \phi(x_n), \beta^{-1}) \]

\[ = \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi) - \beta E_D(w) \]

where

\[ E_D(w) = \frac{1}{2} \sum_{n=1}^{N} \left( t_n - w^T \phi(x_n) \right)^2 \]

is the sum-of-squares error.
Computing the gradient and setting it to zero yields

$$\nabla_w \ln p(t|w, \beta) = \beta \sum_{n=1}^{N} \left\{ t_n - w^T \phi(x_n) \right\} \phi(x_n)^T = 0.$$ 

Solving for $w$, we get

$$w_{ML} = \left( \Phi^T \Phi \right)^{-1} \Phi^T t$$

where

$$\Phi = \begin{pmatrix}
\phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_{M-1}(x_1) \\
\phi_0(x_2) & \phi_1(x_2) & \cdots & \phi_{M-1}(x_2) \\
\vdots & \vdots & \ddots & \vdots \\
\phi_0(x_N) & \phi_1(x_N) & \cdots & \phi_{M-1}(x_N)
\end{pmatrix}.$$ 

The Moore-Penrose pseudo-inverse, $\Phi^\dagger$. 

Maximum Likelihood and Least Squares (3)
Geometry of Least Squares

Consider

\[ y = \Phi w_{ML} = [\varphi_1, \ldots, \varphi_M] w_{ML}. \]

\[ y \in S \subseteq T \quad t \in T \]

\( S \) is spanned by \( \varphi_1, \ldots, \varphi_M \).

\( w_{ML} \) minimizes the distance between \( t \) and its orthogonal projection on \( S \), i.e. \( y \).