CSCI 590: Machine Learning

Lecture 11:
Perceptron algorithm, probabilistic generative models, probabilistic discriminative models

Instructor: Murat Dundar

Acknowledgement:
These slides are prepared using the course textbook
http://research.microsoft.com/~cmbishop/prml/
Another example of a linear discriminant model is the perceptron of Rosenblatt (1962)

\[ y(x) = f \left( w^T \phi(x) \right) \]

where the nonlinear activation function \( f(\cdot) \) is given by a step function of the form

\[ f(a) = \begin{cases} +1, & a \geq 0 \\ -1, & a < 0 \end{cases} \]
Perceptron Algorithm (2)

Targets: $t = +1$ for class $C_1$ and $t = -1$ for class $C_2$.

Error Function: Total number of misclassified patterns?

Does not work because of discontinuities. Methods based on optimizing $w$ using the gradient of the error function cannot be applied, because the gradient is zero almost everywhere.
Perceptron Algorithm (3)

**Perceptron Criterion:**

\[ E_P(w) = - \sum_{n \in M} w^T \phi_n t_n \]

where M denotes the set of all misclassified samples.

**Stochastic gradient descent:**

\[ w^{(\tau+1)} = w^{(\tau)} - \eta \nabla E_P(w) = w^{(\tau)} + \eta \phi_n t_n \]
Perceptron Algorithm (4)

Perceptron Algorithm:

- for each sample $x_n$ evaluate $y(x_n)$
- is $x_n$ correctly classified?
  - yes: do nothing
  - no:
    - if $t_n = 1$ add $\phi(x_n)$ to the current estimate of $w$
    - if $t_n = -1$ subtract $\phi(x_n)$ from the current estimate of $w$
Perceptron Algorithm (4)

The contribution to the error from a misclassified pattern will be reduced

\[-w^{(\tau+1)T} \phi_n t_n = -w^{(\tau)T} \phi_n t_n - (\phi_n t_n)^T \phi_n t_n < -w^{(\tau)T} \phi_n t_n\]

The total error may still increase because the change in \(w\) may cause the contribution of other samples to the error function to increase.

*Perceptron convergence theorem:* If there exists an exact solution, i.e., the classes are linearly separable, then the perceptron learning algorithm is guaranteed to find an exact solution in a finite number of steps.
For a two-class classification problem the posterior probability for class $C_1$ is

$$p(C_1|x) = \frac{p(x|C_1)p(C_1)}{p(x|C_1)p(C_1) + p(x|C_2)p(C_2)} = \frac{1}{1 + \exp(-a)} = \sigma(a)$$

$$a = \ln \frac{p(x|C_1)p(C_1)}{p(x|C_2)p(C_2)}$$

Logistic Sigmoid Function

$$\sigma(a) = \frac{1}{1 + \exp(-a)}$$
Symmetry property: \( \sigma(-a) = 1 - \sigma(a) \)

Inverse function: 
\[
a = \ln \left( \frac{\sigma}{1 - \sigma} \right)
\]

Inverse function is also known as the logit function and represents the log of the ratio of probabilities \( \ln \left[ \frac{p(C_1/x)}{p(C_2/x)} \right] \).
Probabilistic Generative Models (3)

For the case of \( K > 2 \) classes

\[
p(C_k | x) = \frac{p(x | C_k) p(C_k)}{\sum_j p(x | C_j) p(C_j)}
\]

\[
= \frac{\exp(a_k)}{\sum_j \exp(a_j)}
\]

\[
a_k = \ln p(x | C_k) p(C_k)
\]
Probabilistic Generative Models (4)

Continuous Inputs: Class conditional densities are Gaussian with a shared covariance matrix

\[
p(x|C_k) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_k)^T \Sigma^{-1} (x - \mu_k) \right\}
\]

\[
p(C_1|x) = \sigma(w^T x + w_0)
\]

To find \( w^T x + w_0 \) we evaluate \( \ln \frac{p(x|C_1)p(C_1)}{p(x|C_2)p(C_2)} \)

\[
w = \Sigma^{-1}(\mu_1 - \mu_2)
\]

\[
w_0 = -\frac{1}{2} \mu_1^T \Sigma^{-1} \mu_1 + \frac{1}{2} \mu_2^T \Sigma^{-1} \mu_2 + \ln \frac{p(C_1)}{p(C_2)}
\]
Probabilistic Generative Models (5)

Maximum Likelihood Solution: We have a dataset \( \{x_n, t_n\} \), \( n = 1, \ldots, N \), \( t_n = \{1, -1\} \)

\[
p(x_n, C_1) = p(C_1)p(x_n | C_1) = \pi \mathcal{N}(x_n | \mu_1, \Sigma)
\]

\[
p(x_n, C_2) = p(C_2)p(x_n | C_2) = (1 - \pi) \mathcal{N}(x_n | \mu_2, \Sigma)
\]
Probabilistic Generative Models (6)

Thus the likelihood function is given by

\[
p(t|\pi, \mu_1, \mu_2, \Sigma) = \prod_{n=1}^{N} \left[ \pi \mathcal{N}(x_n | \mu_1, \Sigma) \right]^{t_n} \left[ (1 - \pi) \mathcal{N}(x_n | \mu_2, \Sigma) \right]^{1-t_n}
\]

(4.71)

where \( t = (t_1, \ldots, t_N)^T \). As usual, it is convenient to maximize the log of the likelihood function. Consider first the maximization with respect to \( \pi \). The terms in the log likelihood function that depend on \( \pi \) are

\[
\sum_{n=1}^{N} \left\{ t_n \ln \pi + (1 - t_n) \ln(1 - \pi) \right\}.
\]

(4.72)

Setting the derivative with respect to \( \pi \) equal to zero and rearranging, we obtain

\[
\pi = \frac{1}{N} \sum_{n=1}^{N} t_n = \frac{N_1}{N} = \frac{N_1}{N_1 + N_2}
\]

(4.73)
Now consider the maximization with respect to $\mu_1$. Again we can pick out of the log likelihood function those terms that depend on $\mu_1$ giving

$$\sum_{n=1}^{N} t_n \ln \mathcal{N}(x_n | \mu_1, \Sigma) = -\frac{1}{2} \sum_{n=1}^{N} t_n (x_n - \mu_1)^T \Sigma^{-1} (x_n - \mu_1) + \text{const.} \quad (4.74)$$

Setting the derivative with respect to $\mu_1$ to zero and rearranging, we obtain

$$\mu_1 = \frac{1}{N_1} \sum_{n=1}^{N} t_n x_n \quad (4.75)$$

which is simply the mean of all the input vectors $x_n$ assigned to class $C_1$. By a similar argument, the corresponding result for $\mu_2$ is given by

$$\mu_2 = \frac{1}{N_2} \sum_{n=1}^{N} (1 - t_n) x_n \quad (4.76)$$

which again is the mean of all the input vectors $x_n$ assigned to class $C_2$.  

Finally, consider the maximum likelihood solution for the shared covariance matrix $\Sigma$. Picking out the terms in the log likelihood function that depend on $\Sigma$, we have

$$
-\frac{1}{2} \sum_{n=1}^{N} t_n \ln |\Sigma| - \frac{1}{2} \sum_{n=1}^{N} t_n (x_n - \mu_1)^T \Sigma^{-1} (x_n - \mu_1)
$$

$$
-\frac{1}{2} \sum_{n=1}^{N} (1 - t_n) \ln |\Sigma| - \frac{1}{2} \sum_{n=1}^{N} (1 - t_n) (x_n - \mu_2)^T \Sigma^{-1} (x_n - \mu_2)
$$

$$
= -\frac{N}{2} \ln |\Sigma| - \frac{N}{2} \text{Tr} \left\{ \Sigma^{-1} S \right\}
$$

(4.77)
where we have defined

\[ S = \frac{N_1}{N} S_1 + \frac{N_2}{N} S_2 \] (4.78)

\[ S_1 = \frac{1}{N_1} \sum_{n \in C_1} (x_n - \mu_1)(x_n - \mu_1)^T \] (4.79)

\[ S_2 = \frac{1}{N_2} \sum_{n \in C_2} (x_n - \mu_2)(x_n - \mu_2)^T. \] (4.80)

Using the standard result for the maximum likelihood solution for a Gaussian distribution, we see that \( \Sigma = S \), which represents a weighted average of the covariance matrices associated with each of the two classes separately.
Probabilistic Discriminative Models (1)

Logistic Regression

\[ p(C_1|\phi) = y(\phi) = \sigma \left( w^T \phi \right) \]
\[ p(C_2|\phi) = 1 - p(C_1|\phi) \]

Probabilistic generative model with Gaussian class densities had \( \frac{M(M+1)}{2} + 2M + 2 \) parameters. In contrast, logistic regression has only \( M \) parameters.

For large values of \( M \) there is a clear advantage working with logistic regression.
Maximum Likelihood for Logistic Regression

For a data set \( \{ \phi_n, t_n \} \), where \( t_n \in \{0,1\} \) and \( \phi_n = \phi(x_n) \), with \( n=1,\ldots,N \), the likelihood function can be written

\[
p(t|w) = \prod_{n=1}^{N} y_n^{t_n} \{1 - y_n\}^{1-t_n}
\]

Negative log likelihood (cross-entropy function)

\[
E(w) = -\ln p(t|w) = -\sum_{n=1}^{N} \{t_n \ln y_n + (1 - t_n) \ln(1 - y_n)\}
\]
Taking the gradient of the error function with respect to $w$:

$$\nabla E(w) = \sum_{n=1}^{N} (y_n - t_n) \phi_n$$

We used the fact that:

$$\frac{d\sigma}{da} = \sigma(1 - \sigma)$$
Maximum likelihood can exhibit severe over-fitting for data sets that are linearly separable.

This arises because the maximum likelihood solution occurs when $y_n = t_n$ for all samples. This occurs when the sigmoid function saturates, i.e., $w^T \phi_n \to \pm \infty$. 
In the case of the linear regression models discussed the maximum likelihood solution, on the assumption of a Gaussian noise model, leads to a closed-form solution.

This was a consequence of the quadratic dependence of the log likelihood function on the parameter vector $\mathbf{w}$.

For logistic regression, there is no longer a closed-form solution, due to the nonlinearity of the logistic sigmoid function.
The departure from a quadratic form is not substantial.

The error function is convex and hence has a unique minimum.

The error function can be minimized by an efficient iterative technique based on the *Newton-Raphson* iterative optimization scheme, which uses a local quadratic approximation to the log likelihood function.
\[ w^{(\text{new})} = w^{(\text{old})} - H^{-1} \nabla E(w). \] (4.92)

where \( H \) is the Hessian matrix whose elements comprise the second derivatives of \( E(w) \) with respect to the components of \( w \).

\[
\nabla E(w) = \sum_{n=1}^{N} (y_n - t_n) \phi_n = \Phi^T(y - t)
\]

\[
H = \nabla \nabla E(w) = \sum_{n=1}^{N} y_n(1 - y_n) \phi_n \phi_n^T = \Phi^T R \Phi
\]
Probabilistic Discriminative Models(8)

The Newton-Raphson update formula for the logistic regression model then becomes

\[
\begin{align*}
\mathbf{w}^{(\text{new})} &= \mathbf{w}^{(\text{old})} - (\Phi^T \mathbf{R} \Phi)^{-1} \Phi^T (\mathbf{y} - \mathbf{t}) \\
&= (\Phi^T \mathbf{R} \Phi)^{-1} \left\{ \Phi^T \mathbf{R} \mathbf{w}^{(\text{old})} - \Phi^T (\mathbf{y} - \mathbf{t}) \right\} \\
&= (\Phi^T \mathbf{R} \Phi)^{-1} \Phi^T \mathbf{R} \mathbf{z}
\end{align*}
\]

\[
\mathbf{z} = \Phi \mathbf{w}^{(\text{old})} - \mathbf{R}^{-1} (\mathbf{y} - \mathbf{t}) \quad R_{nn} = y_n (1 - y_n)
\]

Compare this with linear regression:

\[
\nabla E(\mathbf{w}) = \sum_{n=1}^{N} (\mathbf{w}^T \phi_n - t_n) \phi_n = \Phi^T \Phi \mathbf{w} - \Phi^T \mathbf{t}
\]

\[
\mathbf{H} = \nabla \nabla E(\mathbf{w}) = \sum_{n=1}^{N} \phi_n \phi_n^T = \Phi^T \Phi
\]

\[
\begin{align*}
\mathbf{w}^{(\text{new})} &= \mathbf{w}^{(\text{old})} - (\Phi^T \Phi)^{-1} \left\{ \Phi^T \Phi \mathbf{w}^{(\text{old})} - \Phi^T \mathbf{t} \right\} \\
&= (\Phi^T \Phi)^{-1} \Phi^T \mathbf{t}
\end{align*}
\]
The update formula takes the form of a weighted least-squares solution.

Because the weighing matrix $\mathbf{R}$ is not constant but depends on the parameter vector $\mathbf{w}$, unlike the least square solution to the linear regression problem there is no closed-form solution.

We apply the equations iteratively, each time using the new weight vector $\mathbf{w}$ to compute a revised weighing matrix $\mathbf{R}$.

For this reason, the algorithm is known as *iterative reweighted least squares*, or *IRLS* (Rubin, 1983).
Probit regression:

For a broad range of class-conditional distributions, described by the exponential family, the resulting posterior class probabilities are given by a logistic (or softmax) transformation acting on a linear function of the feature variables.

However, not all choices of class-conditional density give rise to such a simple form for the posterior probabilities (for instance, if the class-conditional densities are modelled using Gaussian mixtures).
The generalized linear model based on an inverse probit activation function is known as probit regression.

Inverse Probit function:

\[ \Phi(a) = \int_{-\infty}^{a} \mathcal{N}(\theta | 0, 1) \, d\theta \]

Results obtained by probit regression is usually similar to those of logistic regression.

However, in the case of outliers they behave differently. The tails of a sigmoid function decay asymptotically like \( \exp(-x) \) whereas those of a probit function decay like \( \exp(-x^2) \), which makes probit more sensitive to outliers.