CSC 411: Lecture 09: Naive Bayes

Class based on Raquel Urtasun & Rich Zemel’s lectures

Sanja Fidler

University of Toronto

Feb 8, 2015
Today

- Classification – Multi-dimensional (Gaussian) Bayes classifier
- Estimate probability densities from data
- Naive Bayes classifier
Two approaches to classification:

- **Discriminative** classifiers estimate parameters of decision boundary/class separator directly from labeled examples
  - learn $p(y|x)$ directly (logistic regression models)
  - learn mappings from inputs to classes (least-squares, neural nets)

- **Generative approach**: model the distribution of inputs characteristic of the class (Bayes classifier)
  - Build a model of $p(x|y)$
  - Apply Bayes Rule
Aim to diagnose whether patient has diabetes: classify into one of two classes (yes $C=1$; no $C=0$)

Run battery of tests

Given patient's results: $\mathbf{x} = [x_1, x_2, \cdots, x_d]^T$ we want to update class probabilities using Bayes Rule:

$$p(C|\mathbf{x}) = \frac{p(\mathbf{x}|C)p(C)}{p(\mathbf{x})}$$

More formally

$$\text{posterior} = \frac{\text{Class likelihood} \times \text{prior}}{\text{Evidence}}$$

How can we compute $p(\mathbf{x})$ for the two class case?

$$p(\mathbf{x}) = p(\mathbf{x}|C = 0)p(C = 0) + p(\mathbf{x}|C = 1)p(C = 1)$$
Classification: Diabetes Example

- Last class we had a single observation per patient: white blood cell count

\[ p(C = 1 | x = 48) = \frac{p(x = 48 | C = 1)p(C = 1)}{p(x = 48)} \]

- Add second observation: Plasma glucose value
- Now our input \( x \) is 2-dimensional
Gaussian Discriminant Analysis (Gaussian Bayes Classifier)

- Gaussian Discriminant Analysis in its general form assumes that \( p(x|t) \) is distributed according to a multivariate normal (Gaussian) distribution.

- Multivariate Gaussian distribution:
  \[
  p(x|t = k) = \frac{1}{(2\pi)^{d/2}|\Sigma_k|^{1/2}} \exp \left[ -(x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) \right]
  \]
  
  where \( |\Sigma_k| \) denotes the determinant of the matrix, and \( d \) is dimension of \( x \)

- Each class \( k \) has associated mean vector \( \mu_k \) and covariance matrix \( \Sigma_k \)

- Typically the classes share a single covariance matrix \( \Sigma \) ("share" means that they have the same parameters; the covariance matrix in this case):
  \( \Sigma = \Sigma_1 = \cdots = \Sigma_k \)
Multivariate Data

- Multiple measurements (sensors)
- \( d \) inputs/features/attributes
- \( N \) instances/observations/examples

\[
X = \begin{bmatrix}
  x_1^{(1)} & x_2^{(1)} & \cdots & x_d^{(1)} \\
  x_1^{(2)} & x_2^{(2)} & \cdots & x_d^{(2)} \\
  \vdots & \vdots & \ddots & \vdots \\
  x_1^{(N)} & x_2^{(N)} & \cdots & x_d^{(N)}
\end{bmatrix}
\]
Multivariate Parameters

- **Mean**
  \[ \mathbb{E}[\mathbf{x}] = [\mu_1, \ldots, \mu_d]^T \]

- **Covariance**
  \[ \Sigma = \text{Cov}(\mathbf{x}) = \mathbb{E}[(\mathbf{x} - \mu)(\mathbf{x} - \mu)^T] = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{12} & \sigma_2^2 & \cdots & \sigma_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2 \end{bmatrix} \]

- **Correlation** = \( \text{Corr}(\mathbf{x}) \) is the covariance divided by the product of standard deviation
  \[ \rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j} \]
Multivariate Gaussian Distribution

- $\mathbf{x} \sim \mathcal{N}(\mu, \Sigma)$, a Gaussian (or normal) distribution defined as
  \[
p(x) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp \left[ -\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right]
  \]

- Mahalanobis distance $((\mathbf{x} - \mu_k)^T \Sigma^{-1} (\mathbf{x} - \mu_k))$ measures the distance from $\mathbf{x}$ to $\mu$ in terms of $\Sigma$
- It normalizes for difference in variances and correlations
Bivariate Normal

$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\Sigma = 0.5 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\Sigma = 2 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Figure: Probability density function

Figure: Contour plot of the pdf
Bivariate Normal

\[ \text{var}(x_1) = \text{var}(x_2) \quad \text{var}(x_1) > \text{var}(x_2) \quad \text{var}(x_1) < \text{var}(x_2) \]

**Figure:** Probability density function

**Figure:** Contour plot of the pdf
Bivariate Normal

\[ \Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \]

\[ \Sigma = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix} \]

\[ \Sigma = \begin{pmatrix} 1 & 0.8 \\ 0.8 & 1 \end{pmatrix} \]

Figure: Probability density function

Figure: Contour plot of the pdf
Bivariate Normal

\[ \text{Cov}(x_1, x_2) = 0 \quad \text{Cov}(x_1, x_2) > 0 \quad \text{Cov}(x_1, x_2) < 0 \]

**Figure:** Probability density function

**Figure:** Contour plot of the pdf
Gaussian Discriminant Analysis (Gaussian Bayes Classifier)

- GDA (GBC) decision boundary is based on class posterior:

\[
\log p(t_k|\mathbf{x}) = \log p(\mathbf{x}|t_k) + \log p(t_k) - \log p(\mathbf{x})
\]

\[
= -\frac{d}{2} \log(2\pi) - \frac{1}{2} \log |\Sigma_k| - \frac{1}{2} (\mathbf{x} - \mu_k)^T \Sigma_k^{-1} (\mathbf{x} - \mu_k) + \log p(t_k) - \log p(\mathbf{x})
\]

- Decision: take the class with the highest posterior probability
Decision Boundary

Likelihoods

Posterior for $t_1$

discriminant: $P(t_1|x) = 0.5$
Decision Boundary when Shared Covariance Matrix
Learn the parameters using maximum likelihood

\[
\ell(\phi, \mu_0, \mu_1, \Sigma) = - \log \prod_{n=1}^{N} p(x^{(n)}, t^{(n)}|\phi, \mu_0, \mu_1, \Sigma)
\]

\[
= - \log \prod_{n=1}^{N} p(x^{(n)}|t^{(n)}, \mu_0, \mu_1, \Sigma)p(t^{(n)}|\phi)
\]

What have we assumed?
More on MLE

- Assume the prior is Bernoulli (we have two classes)

\[ p(t|\phi) = \phi^t (1 - \phi)^{1-t} \]

- You can compute the ML estimate in closed form

\[
\phi = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}[t^{(n)} = 1] \\
\mu_0 = \frac{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = 0] \cdot x^{(n)}}{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = 0]} \\
\mu_1 = \frac{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = 1] \cdot x^{(n)}}{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = 1]} \\
\Sigma = \frac{1}{N} \sum_{n=1}^{N} (x^{(n)} - \mu_{t^{(n)}})(x^{(n)} - \mu_{t^{(n)}})^T
\]
Gaussian Discriminative Analysis vs Logistic Regression

- If you examine $p(t = 1|\mathbf{x})$ under GDA, you will find that it looks like this:

$$
p(t|\mathbf{x}, \phi, \mu_0, \mu_1, \Sigma) = \frac{1}{1 + \exp(-\mathbf{w}^T \mathbf{x})}
$$

where $\mathbf{w}$ is an appropriate function of $(\phi, \mu_0, \mu_1, \Sigma)$

- So the decision boundary has the same form as logistic regression!

- When should we prefer GDA to LR, and vice versa?
GDA makes stronger modeling assumption: assumes class-conditional data is multivariate Gaussian

If this is true, GDA is asymptotically efficient (best model in limit of large N)

But LR is more robust, less sensitive to incorrect modeling assumptions

Many class-conditional distributions lead to logistic classifier

When these distributions are non-Gaussian, in limit of large N, LR beats GDA
What if $x$ is high-dimensional?

- For Gaussian Bayes Classifier, if input $x$ is high-dimensional, then covariance matrix has many parameters.
- Save some parameters by using a shared covariance for the classes.
- Any other idea you can think of?
Naive Bayes

- **Naive Bayes** is an alternative generative model: Assumes features independent given the class

\[
p(x|t = k) = \prod_{i=1}^{d} p(x_i|t = k)
\]

- Assuming likelihoods are Gaussian, how many parameters required for Naive Bayes classifier?

- Important note: Naive Bayes does not assume a particular distribution
Naive Bayes Classifier

Given

- prior $p(t = k)$
- assuming features are conditionally independent given the class
- likelihood $p(x_i | t = k)$ for each $x_i$

The decision rule

$$ y = \arg \max_k p(t = k) \prod_{i=1}^{d} p(x_i | t = k) $$

- If the assumption of conditional independence holds, NB is the optimal classifier
- If not, a heavily regularized version of generative classifier
- What’s the regularization?
- Note: NB’s assumptions (cond. independence) typically do not hold in practice. However, the resulting algorithm still works well on many problems, and it typically serves as a decent baseline for more sophisticated models
**Gaussian Naive Bayes** classifier assumes that the likelihoods are Gaussian:

\[
p(x_i | t = k) = \frac{1}{\sqrt{2\pi \sigma_{ik}}} \exp \left[ -\frac{(x_i - \mu_{ik})^2}{2\sigma_{ik}^2} \right]
\]

(this is just a 1-dim Gaussian, one for each input dimension)

- Model the same as Gaussian Discriminative Analysis with diagonal covariance matrix
- Maximum likelihood estimate of parameters

\[
\begin{align*}
\mu_{ik} &= \frac{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = k] \cdot x_i^{(n)}}{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = k]} \\
\sigma_{ik}^2 &= \frac{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = k] \cdot (x_i^{(n)} - \mu_{ik})^2}{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = k]}
\end{align*}
\]
Decision Boundary: Shared Variances (between Classes)

variances may be different
Same variance across all classes and input dimensions, all class priors equal

Classification only depends on distance to the mean. Why?
In this case: $\sigma_{i,k} = \sigma$ (just one parameter), class priors equal (e.g., $p(t_k) = 0.5$ for 2-class case)

Going back to class posterior for GDA:

$$
\log p(t_k|x) = \log p(x|t_k) + \log p(t_k) - \log p(x)
$$

$$
= -\frac{d}{2} \log(2\pi) - \frac{1}{2} \log |\Sigma_k^{-1}| - \frac{1}{2} (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) + \log p(t_k) - \log p(x)
$$

where we take $\Sigma_k = \sigma^2 I$ and ignore terms that don’t depend on $k$ (don’t matter when we take max over classes):

$$
\log p(t_k|x) = -\frac{1}{2\sigma^2} (x - \mu_k)^T (x - \mu_k)
$$
You have examples of emails that are spam and non-spam

How would you classify spam vs non-spam?

Think about it at home, solution in the next tutorial