Lecture 9: Logistic Regression

Discriminative vs. Generative Classification
Given:

- Class prior $P(Y)$
- $d$ conditionally independent features $X_1, \ldots, X_d$ given the class label $Y$
- For each $X_i$ feature, we have the conditional likelihood $P(X_i | Y)$

Naïve Bayes Decision rule:

$$f_{NB}(x) = \arg \max_y P(x_1, \ldots, x_d | y) P(y)$$

$$= \arg \max_y \prod_{i=1}^{d} P(x_i | y) P(y)$$
Last time... Naïve Bayes Algorithm for discrete features

\[ f_{NB}(x) = \arg \max_y \prod_{i=1}^{d} P(x_i|y)P(y) \]  

We need to estimate these probabilities!

**Estimators**

For Class Prior

\[ \hat{P}(y) = \frac{\{ \# j : Y(j) = y \}}{n} \]

For Likelihood

\[ \frac{\hat{P}(x_i, y)}{\hat{P}(y)} = \frac{\{ \# j : X_i(j) = x_i, Y(j) = y \}/n}{\{ \# j : Y(j) = y \}/n} \]

**NB Prediction for test data:**

\[ X = (x_1, \ldots, x_d) \]

\[ Y = \arg \max_y \hat{P}(y) \prod_{i=1}^{d} \frac{\hat{P}(x_i, y)}{\hat{P}(y)} \]
Last time... Text Classification

MEDLINE Article

MeSH Subject Category Hierarchy

- Antagonists and Inhibitors
- Blood Supply
- Chemistry
- Drug Therapy
- Embryology
- Epidemiology
- ...

How to represent a text document?
Last time... Bag of words model

Typical additional assumption:

Position in document doesn’t matter:

\[ P(X_i=x_i|Y=y) = P(X_k=x_i|Y=y) \]

- “Bag of words” model – order of words on the page ignored
  The document is just a bag of words: i.i.d. words
- Sounds really silly, but often works very well!

⇒ \( K(50000-1) \) parameters to estimate

The probability of a document with words \( x_1, x_2, \ldots \)

\[
\prod_{i=1}^{\text{LengthDoc}} P(x_i|y) = \prod_{w=1}^{W} P(w|y)^{\text{count}_w}
\]
### Last time... Bag of words model

\[ \hat{P}(c) = \frac{N_c}{\sum c} \]

\[ \hat{P}(w|c) = \frac{N_{count(w,c)} + 1}{\sum c + |V|} \]

#### Priors:
- \( P(c) = \frac{3}{4} \)
- \( P(j) = \frac{1}{4} \)

#### Conditional Probabilities:
- \( P(\text{Chinese}|c) = \frac{5+1}{8+6} = \frac{6}{14} = \frac{3}{7} \)
- \( P(\text{Tokyo}|c) = \frac{0+1}{8+6} = \frac{1}{14} \)
- \( P(\text{Japan}|c) = \frac{0+1}{8+6} = \frac{1}{14} \)
- \( P(\text{Chinese}|j) = \frac{1+1}{3+6} = \frac{2}{9} \)
- \( P(\text{Tokyo}|j) = \frac{1+1}{3+6} = \frac{2}{9} \)
- \( P(\text{Japan}|j) = \frac{1+1}{3+6} = \frac{2}{9} \)

### Choosing a class:

\[
\begin{align*}
P(\text{cl}d5) & \propto \frac{3}{4} * (\frac{3}{7})^3 * \frac{1}{14} * \frac{1}{14} \\ & \approx 0.0003
\end{align*}
\]

\[
\begin{align*}
P(\text{jd}5) & \propto \frac{1}{4} * (\frac{2}{9})^3 * \frac{2}{9} * \frac{2}{9} \\ & \approx 0.0001
\end{align*}
\]
Last time... What if features are continuous?

e.g., character recognition: $X_i$ is intensity at $i^{th}$ pixel

Gaussian Naïve Bayes (GNB):

$$P(X_i = x \mid Y = y_k) = \frac{1}{\sigma_{ik} \sqrt{2\pi}} e^{-\frac{(x-\mu_{ik})^2}{2\sigma_{ik}^2}}$$

Different mean and variance for each class $k$ and each pixel $i$.

Sometimes assume variance
- is independent of $Y$ (i.e., $\sigma_i$),
- or independent of $X_i$ (i.e., $\sigma_k$)
- or both (i.e., $\sigma$)

$$\hat{\mu}_{MLE} = \frac{1}{N} \sum_{j=1}^{N} x_j$$

$$\hat{\sigma}^2_{unbiased} = \frac{1}{N - 1} \sum_{j=1}^{N} (x_j - \hat{\mu})^2$$
Logistic Regression
Recap: Naïve Bayes

- NB Assumption: \( P(X_1 \ldots X_d|Y) = \prod_{i=1}^{d} P(X_i|Y) \)

- NB Classifier:
  \[
  f_{NB}(x) = \arg \max_y \prod_{i=1}^{d} P(x_i|y)P(y)
  \]

- Assume parametric form for \( P(X_i|Y) \) and \( P(Y) \)
  - Estimate parameters using MLE/MAP and plug in
Gaussian Naïve Bayes (GNB)

• There are several distributions that can lead to a linear boundary.

• As an example, consider Gaussian Naïve Bayes:

\[
P(X_i | Y = y) = \frac{1}{\sqrt{2\pi\sigma_{i,y}^2}} e^{-\frac{(X_i - \mu_{i,y})^2}{2\sigma_{i,y}^2}}
\]

Gaussian class conditional densities

• What if we assume variance is independent of class, i.e. \( \sigma_{i,0}^2 = \sigma_{i,1}^2 \)
GNB with equal variance is a Linear Classifier!

\[
P(X_i | Y = y) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(X_i - \mu_i, y)^2}{2\sigma_i^2}}
\]

Decision boundary:

\[
\prod_{i=1}^{d} P(X_i | Y = 0)P(Y = 0) = \prod_{i=1}^{d} P(X_i | Y = 1)P(Y = 1)
\]
GNB with equal variance is a Linear Classifier!

Decision boundary:

\[
\prod_{i=1}^{d} P(X_i \mid Y = 0) P(Y = 0) = \prod_{i=1}^{d} P(X_i \mid Y = 1) P(Y = 1)
\]

\[
\log \frac{P(Y = 0) \prod_{i=1}^{d} P(X_i \mid Y = 0)}{P(Y = 1) \prod_{i=1}^{d} P(X_i \mid Y = 1)} = \log \frac{1 - \pi}{\pi} + \sum_{i=1}^{d} \log \frac{P(X_i \mid Y = 0)}{P(X_i \mid Y = 1)}
\]
GNB with equal variance is a Linear Classifier!

\[
P(X_i|Y = y) = \frac{1}{\sqrt{2\pi\sigma^2_i}} e^{-\frac{(X_i - \mu_i,y)^2}{2\sigma^2_i}}
\]

Decision boundary:

\[
\prod_{i=1}^{d} P(X_i|Y = 0)P(Y = 0) = \prod_{i=1}^{d} P(X_i|Y = 1)P(Y = 1)
\]

\[
\log \frac{P(Y = 0) \prod_{i=1}^{d} P(X_i|Y = 0)}{P(Y = 1) \prod_{i=1}^{d} P(X_i|Y = 1)} = \log \frac{1 - \pi}{\pi} + \sum_{i=1}^{d} \log \frac{P(X_i|Y = 0)}{P(X_i|Y = 1)}
\]

\[
= \log \frac{1 - \pi}{\pi} + \sum_{i} \frac{\mu_{i,1} - \mu_{i,0}}{2\sigma^2_i} + \sum_{i} \frac{\mu_{i,0} - \mu_{i,1}}{\sigma^2_i}X_i =: w_0 + \sum_{i} w_i X_i
\]

\[
\text{Constant term} \quad \text{First-order term}
\]
Gaussian Naive Bayes (GNB)

\[ X = (x_1, x_2) \]
\[ P_1 = P(Y = 0) \]
\[ P_2 = P(Y = 1) \]
\[ p_1(X) = p(X|Y = 0) \sim \mathcal{N}(M_1, \Sigma_1) \]
\[ p_2(X) = p(X|Y = 1) \sim \mathcal{N}(M_2, \Sigma_2) \]
Generative vs. Discriminative Classifiers

• Generative classifiers (e.g. Naïve Bayes)
  - Assume some functional form for $P(X,Y)$ (or $P(X|Y)$ and $P(Y)$)
  - Estimate parameters of $P(X|Y)$, $P(Y)$ directly from training data

• But $\arg \max_Y P(X|Y) \ P(Y) = \arg \max_Y P(Y|X)$

• Why not learn $P(Y|X)$ directly? Or better yet, why not learn the decision boundary directly?

• Discriminative classifiers (e.g. Logistic Regression)
  - Assume some functional form for $P(Y|X)$ or for the decision boundary
  - Estimate parameters of $P(Y|X)$ directly from training data
Logistic Regression

Assumes the following functional form for $P(Y \mid X)$:

$$P(Y = 1 \mid X) = \frac{1}{1 + \exp(w_0 + \sum_i w_i X_i)}$$

Logistic function applied to linear function of the data

Logistic function (or Sigmoid):

$$\frac{1}{1 + \exp(-z)}$$

Features can be discrete or continuous!
Logistic Regression is a Linear Classifier!

Assumes the following functional form for $P(Y|X)$:

$$P(Y = 1|X) = \frac{1}{1 + \exp(w_0 + \sum_i w_i X_i)}$$

Decision boundary:

$$P(Y = 0|X) \geq P(Y = 1|X)$$

$$w_0 + \sum_i w_i X_i \geq 0$$

(Linear Decision Boundary)
Logistic Regression is a Linear Classifier!

Assumes the following functional form for $P(Y\mid X)$:

$$P(Y = 1\mid X) = \frac{1}{1 + \exp(w_0 + \sum_i w_i X_i)}$$

$$\Rightarrow P(Y = 0\mid X) = \frac{\exp(w_0 + \sum_i w_i X_i)}{1 + \exp(w_0 + \sum_i w_i X_i)}$$

$$\Rightarrow \frac{P(Y = 0\mid X)}{P(Y = 1\mid X)} = \exp(w_0 + \sum_i w_i X_i) \quad 0 \ll 1$$

$$\Rightarrow w_0 + \sum_i w_i X_i \quad 0 \ll 0$$
Logistic Regression for more than 2 classes

- Logistic regression in more general case, where \( Y \in \{y_1, \ldots, y_K\} \)

\[
P(Y = y_k|X) = \frac{\exp(w_{k0} + \sum_{i=1}^{d} w_{ki}X_i)}{1 + \sum_{j=1}^{K-1} \exp(w_{j0} + \sum_{i=1}^{d} w_{ji}X_i)}
\]

for \( k < K \)

for \( k = K \) (normalization, so no weights for this class)

\[
P(Y = y_K|X) = \frac{1}{1 + \sum_{j=1}^{K-1} \exp(w_{j0} + \sum_{i=1}^{d} w_{ji}X_i)}
\]
Training Logistic Regression

We’ll focus on binary classification:

\[
P(Y = 0|X, w) = \frac{1}{1 + exp(w_0 + \sum_i w_i X_i)}
\]

\[
P(Y = 1|X, w) = \frac{exp(w_0 + \sum_i w_i X_i)}{1 + exp(w_0 + \sum_i w_i X_i)}
\]

How to learn the parameters \(w_0, w_1, \ldots, w_d\)?

Training Data \(\{(X^{(j)}, Y^{(j)})\}_{j=1}^n\)

\(X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})\)

Maximum Likelihood Estimates

\(\hat{w}_{MLE} = \arg\max_w \prod_{j=1}^n P(X^{(j)}, Y^{(j)} | w)\)
Training Logistic Regression

We’ll focus on binary classification:

\[
P(Y = 0|X, w) = \frac{1}{1 + e^{w_0 + \sum_i w_i X_i}}
\]

\[
P(Y = 1|X, w) = \frac{e^{w_0 + \sum_i w_i X_i}}{1 + e^{w_0 + \sum_i w_i X_i}}
\]

How to learn the parameters \(w_0, w_1, \ldots, w_d\)?

Training Data

\[
\{(X^{(j)}, Y^{(j)})\}_{j=1}^n
\]

\[
X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})
\]

Maximum Likelihood Estimates

\[
\hat{w}_{MLE} = \arg \max_w \prod_{j=1}^n P(X^{(j)}, Y^{(j)} | w)
\]

But there is a problem...

Don’t have a model for \(P(X)\) or \(P(X|Y)\) — only for \(P(Y|X)\)
Training Logistic Regression

How to learn the parameters $w_0, w_1, \ldots, w_d$?

Training Data $\{(X^{(j)}, Y^{(j)})\}_{j=1}^n$ $X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})$

Maximum (Conditional) Likelihood Estimates

$$\hat{w}_{MCLE} = \arg \max_w \prod_{j=1}^n P(Y^{(j)} | X^{(j)}, w)$$

Discriminative philosophy — Don’t waste effort learning $P(X)$, focus on $P(Y|X)$ — that’s all that matters for classification!
Expressing Conditional log Likelihood

\[ l(W) = \sum_l Y^l \ln P(Y^l = 1|X^l, W) + (1 - Y^l) \ln P(Y^l = 0|X^l, W) \]

\[ P(Y = 0|X) = \frac{1}{1 + \exp(w_0 + \sum_{i=1}^n w_i X_i)} \]

\[ P(Y = 1|X) = \frac{\exp(w_0 + \sum_{i=1}^n w_i X_i)}{1 + \exp(w_0 + \sum_{i=1}^n w_i X_i)} \]

\( Y \) can take only values 0 or 1, so only one of the two terms in the expression will be non-zero for any given \( Y^l \)
Expressing Conditional log Likelihood

\[ l(W) = \sum_{l} Y^l \ln P(Y^l = 1|X^l, W) + (1 - Y^l) \ln P(Y^l = 0|X^l, W) \]
Expressing Conditional log Likelihood

\[
l(W) = \sum_l Y^l \ln P(Y^l = 1|X^l, W) + (1 - Y^l) \ln P(Y^l = 0|X^l, W)
\]

\[
l(W) = \sum_l Y^l \ln \frac{P(Y^l = 1|X^l, W)}{P(Y^l = 0|X^l, W)} + \ln P(Y^l = 0|X^l, W)
\]

\[
P(Y = 0|X) = \frac{1}{1 + \exp(w_0 + \sum_{i=1}^n w_i X_i)}
\]

\[
P(Y = 1|X) = \frac{\exp(w_0 + \sum_{i=1}^n w_i X_i)}{1 + \exp(w_0 + \sum_{i=1}^n w_i X_i)}
\]
Expressing Conditional log Likelihood

\[
l(W) = \sum_{l} Y^l \ln P(Y^l = 1|X^l, W) + (1 - Y^l) \ln P(Y^l = 0|X^l, W)
\]

\[
= \sum_{l} Y^l \ln \frac{P(Y^l = 1|X^l, W)}{P(Y^l = 0|X^l, W)} + \ln P(Y^l = 0|X^l, W)
\]

\[
= \sum_{l} Y^l (w_0 + \sum_{i} w_i X_i^l) - \ln (1 + \exp (w_0 + \sum_{i} w_i X_i^l))
\]
Maximizing Conditional log Likelihood

\[
\max_w l(w) \equiv \ln \prod_j P(y^j | x^j, w)
\]

\[
= \sum_j y^j (w_0 + \sum_i^d w_i x^j_i) - \ln (1 + \exp(w_0 + \sum_i^d w_i x^j_i))
\]

**Bad news:** no closed-form solution to maximize \(l(w)\)

**Good news:** \(l(w)\) is concave function of \(w\)! concave functions easy to optimize (unique maximum)
Optimizing concave/convex functions

- Conditional likelihood for Logistic Regression is concave
- Maximum of a concave function = minimum of a convex function

Gradient Ascent (concave) / Gradient Descent (convex)

Gradient:
\[ \nabla_w l(w) = \left[ \frac{\partial l(w)}{\partial w_0}, \ldots, \frac{\partial l(w)}{\partial w_n} \right]' \]

Update rule:
\[ \Delta w = \eta \nabla_w l(w) \]

Learning rate, \( \eta > 0 \)

\[ w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \frac{\partial l(w)}{\partial w_i} \bigg|_t \]
Gradient Ascent for Logistic Regression

Gradient ascent algorithm: iterate until change < \( \varepsilon \)

\[
w_0^{(t+1)} \leftarrow w_0^{(t)} + \eta \sum_j [y^j - \hat{P}(Y^j = 1 \mid x^j, w^{(t)})]
\]

For \( i = 1, \ldots, d \),

\[
w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \sum_j x_i^j [y^j - \hat{P}(Y^j = 1 \mid x^j, w^{(t)})]
\]

repeat

- Gradient ascent is simplest of optimization approaches
  - e.g. Newton method, Conjugate gradient ascent, IRLS (see Bishop 4.3.3)
Effect of step-size $\eta$

Large $\eta \rightarrow$ Fast convergence but larger residual error
Also possible oscillations

Small $\eta \rightarrow$ Slow convergence but small residual error
Naïve Bayes vs. Logistic Regression

- Representation equivalence
  - But only in a special case!!! (GNB with class-independent variances)
- But what’s the difference???
Naïve Bayes vs. Logistic Regression

- Representation equivalence
  - But only in a special case!!! (GNB with class-independent variances)
- But what’s the difference???
- LR makes no assumption about $P(X|Y)$ in learning!!!
- Loss function!!!
  - Optimize different functions! Obtain different solutions
Naïve Bayes vs. Logistic Regression

Consider Y Boolean, $X_i$ continuous $X=<X_1 \ldots X_d>$

Number of parameters:
- NB: $4d+1$ $\pi, (\mu_{1,y}, \mu_{2,y}, \ldots, \mu_{d,y}), (\sigma^2_{1,y}, \sigma^2_{2,y}, \ldots, \sigma^2_{d,y})$ $y=0,1$
- LR: $d+1$ $w_0, w_1, \ldots, w_d$

Estimation method:
- NB parameter estimates are uncoupled
- LR parameter estimates are coupled
Generative vs. Discriminative

Given infinite data (asymptotically),

If conditional independence assumption holds, Discriminative and generative NB perform similar.

\[ \epsilon_{\text{Dis}, \infty} \sim \epsilon_{\text{Gen}, \infty} \]

If conditional independence assumption does NOT hold, Discriminative outperforms generative NB.

\[ \epsilon_{\text{Dis}, \infty} < \epsilon_{\text{Gen}, \infty} \]
Generative vs. Discriminative

[Ng & Jordan, NIPS 2001]

Given finite data \((n \text{ data points, } d \text{ features})\),

\[
\epsilon_{\text{Dis},n} \leq \epsilon_{\text{Dis},\infty} + O\left(\sqrt{\frac{d}{n}}\right)
\]

\[
\epsilon_{\text{Gen},n} \leq \epsilon_{\text{Gen},\infty} + O\left(\sqrt{\frac{\log d}{n}}\right)
\]

Naïve Bayes (generative) requires \(n = O(\log d)\) to converge to its asymptotic error, whereas Logistic regression (discriminative) requires \(n = O(d)\).

Why? “Independent class conditional densities”

• parameter estimates not coupled – each parameter is learnt independently, not jointly, from training data.
Naïve Bayes vs. Logistic Regression

Verdict

Both learn a linear decision boundary. Naïve Bayes makes more restrictive assumptions and has higher asymptotic error, BUT converges faster to its less accurate asymptotic error.
Experimental Comparison (Ng-Jordan’01)

UCI Machine Learning Repository 15 datasets, 8 continuous features, 7 discrete features

- pima (continuous)
- adult (continuous)
- boston (predict if > median price, continuous)
- optdigits (0’s and 1’s, continuous)
- optdigits (2’s and 3’s, continuous)
- ionosphere (continuous)

More in the paper...

Naïve Bayes

Logistic Regression
What you should know

• LR is a linear classifier
  - decision rule is a hyperplane

• LR optimized by maximizing conditional likelihood
  - no closed-form solution
  - concave ! global optimum with gradient ascent

• Gaussian Naïve Bayes with class-independent variances representationally equivalent to LR
  - Solution differs because of objective (loss) function

• In general, NB and LR make different assumptions
  - NB: Features independent given class! assumption on \( P(X|Y) \)
  - LR: Functional form of \( P(Y|X) \), no assumption on \( P(X|Y) \)

• Convergence rates
  - GNB (usually) needs less data
  - LR (usually) gets to better solutions in the limit