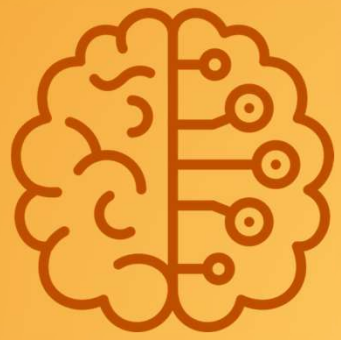
CS 4375 – Introduction to Machine Learning

Logistic Regression

Erick Parolin

THE UNIVERSITY OF TEXAS AT DALLAS



[Based on the slides from Dr. Vibhav Gogate and Dr. Nicholas Ruozzi]

Previously…

- **Classification:** find a function $h: X \rightarrow Y$ from training sample.
  - h approximates the real function f
- **Naïve Bayes** approximates the function by $P(Y|X)$ based on Bayesian approach: $\boldsymbol{P(Y|X) \propto P(X|Y)P(Y)}$
  - **X** is the feature vector
  - Likelihood $\boldsymbol{P(X|Y)}$ and prior $\boldsymbol{P(Y)}$ terms are got from data, assuming **conditional independence**
- What if we make a functional model for probability directly from data?
  - Assume a particular functional form and learn $\mathbf{P(Y|X)}$ from data

Logistic Regression – Functional Form

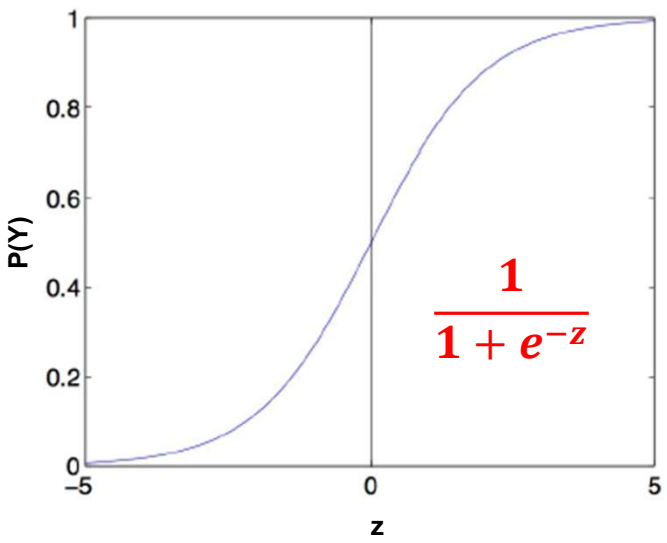
Assume a Functional Form for Learning P(Y|X) is Logistic Function

- Two classes functional form: $Y \in \{0, 1\}$
- For convenience, let $-z = w_0 + \sum_{i=1}^{n} w_i X_i$

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

Since $P(Y = 1|X) + P(Y = 0|X) = 1$, we have:

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



Form of the logistic function. In Logistic Regression, P(Y|X) is assumed to follow this form.

Logistic Regression – Functional Form

Statistical Point of View

- Logistic Regression is a **Generalized Linear Model (GLM)**. Instead of assuming that the mean of the response variable is a linear combinations of the parameters and the predictor variables, Logistic Regression assumes that

$$logit(E[Y|X]) = \ln\left(\frac{P(Y=1|X)}{P(Y=0|X)}\right) = \beta_0 + \sum_i \beta_i X_i$$

or

$$E[Y|X] = P(Y|X) = logit^{-1}(\beta_0 + \sum_i \beta_i X_i) = \frac{1}{1+e^{-(\beta_0+\sum_i \beta_i X_i)}}$$

Logit is the "**link**" function for the GLM, and it is convenient to convert the probabilities to a continuous variables so we can use linear prediction

Nomenclature Mapping

$$\beta_0 + \sum_i \beta_i X_i = w_0 + \sum_{i=1}^{n} w_i X_i$$

$$\frac{P(Y=1|X)}{P(Y=0|X)} = \frac{p}{1-p} = odds = \exp(-z)$$

$$\ln\left(\frac{p}{1-p}\right) = logit(p) = z$$

Logistic Regression – Classification

Classification Rule

- Given **W**, we can classify a new point **X** by assigning the label 1 if $\boldsymbol{P}(\boldsymbol{Y} = 1 \,|\, \boldsymbol{X}) > \boldsymbol{P}(\boldsymbol{Y} = 0 \,|\, \boldsymbol{X})$ and 0 otherwise.
- In practice,

$$\frac{P(Y = 1|X)}{P(Y = 0|X)} > 1 \rightarrow Y = 1$$

- Applying log on both sides, we have

$$w_0 + \sum_{i=1}^{n} w_i X_i > 0 \rightarrow Y = 1$$

Well, it turns out that this is a linear classification rule!!!

Logistic Regression – Learning W

How to Learn the Weights?

- Maximize the conditional likelihood

$$W \leftarrow \arg\max_{W} \prod_j P(Y^{(j)}|X^{(j)}, \boldsymbol{W})$$

$$W \leftarrow \arg\max_{W} \sum_j \boldsymbol{ln}\, P(Y^{(j)}|X^{(j)}, \boldsymbol{W})$$

Note that P(Y|X) is computed using $\boldsymbol{W} = [\boldsymbol{w_0}, \boldsymbol{w_1}, \ldots, \boldsymbol{w_n}]$, which is the weight vector to be learned

- **For convenience, let $l(W)$ be**

$$l(W) = \sum_j \boldsymbol{ln}\, P(Y^{(j)}|X^{(j)}, \boldsymbol{W}) = \sum_j Y^{(j)}\, ln\, P\left(Y^{(j)} = 1 \middle| X^{(j)}, \boldsymbol{W}\right) + (1 - Y^{(j)})\, ln\, P\left(Y^{(j)} = 0 \middle| X^{(j)}, \boldsymbol{W}\right)$$

Examples where Y=1 will account $P\left(Y^{(j)} = 1 \middle| X^{(j)}, \boldsymbol{W}\right)$, while examples where Y=0 will account $P\left(Y^{(j)} = 0 \middle| X^{(j)}, \boldsymbol{W}\right)$.

Logistic Regression – Learning W

How to Learn the Weights?

$$l(W) = \sum_j Y^{(j)} \ln P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right) + (1 - Y^{(j)}) \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \ln P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right) - Y^{(j)} \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right) + \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \ln \frac{P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right)}{P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)} + \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \left(w_0 + \sum_{i=1}^{n} w_i X_i^{(j)}\right) + \ln\left(\frac{1}{1 + \exp\left(w_0 + \sum_{j=1}^{n} w_i X_i^{(j)}\right)}\right)$$

$$= \sum_j Y^{(j)} \left(w_0 + \sum_{i=1}^{n} w_i X_i^{(j)}\right) - \ln\left(1 + \exp\left(w_0 + \sum_{j=1}^{n} w_i X_i^{(j)}\right)\right)$$

Logistic Regression – Learning W

How to Learn the Weights?

$$l(W) = \sum_j Y^{(j)} \ln P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right) + (1 - Y^{(j)}) \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \ln P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right) - Y^{(j)} \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right) + \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \ln \frac{P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right)}{P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)} + \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \left(w_0 + \sum_{i=1}^{n} w_i X_i^{(j)}\right) + \ln\left(\frac{1}{1 + \exp\left(w_0 + \sum_{j=1}^{n} w_i X_i^{(j)}\right)}\right)$$

$$= \sum_j \boldsymbol{Y^{(j)}} \left(\boldsymbol{w_0} + \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{w_i X_i^{(j)}}\right) - \boldsymbol{\ln}\left(\mathbf{1} + \exp\left(\boldsymbol{w_0} + \sum_{\boldsymbol{j=1}}^{\boldsymbol{n}} \boldsymbol{w_i X_i^{(j)}}\right)\right)$$

This is concave on W!!!
Then now it's easy: just take the derivatives and solve!

Logistic Regression – Learning W

How to Learn the Weights?

$$l(W) = \sum_j Y^{(j)} \ln P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right) + (1 - Y^{(j)}) \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \ln P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right) - Y^{(j)} \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right) + \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \ln \frac{P\left(Y^{(j)} = 1 \middle| X^{(j)}, W\right)}{P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)} + \ln P\left(Y^{(j)} = 0 \middle| X^{(j)}, W\right)$$

$$= \sum_j Y^{(j)} \left(w_0 + \sum_{i=1}^{n} w_i X_i^{(j)}\right) + \ln\left(\frac{1}{1 + \exp\left(w_0 + \sum_{j=1}^{n} w_i X_i^{(j)}\right)}\right)$$

$$= \sum_j \boldsymbol{Y^{(j)}} \left(\boldsymbol{w_0} + \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{w_i X_i^{(j)}}\right) - \boldsymbol{\ln}\left(\mathbf{1} + \mathbf{exp}\left(\boldsymbol{w_0} + \sum_{\boldsymbol{j=1}}^{\boldsymbol{n}} \boldsymbol{w_i X_i^{(j)}}\right)\right)$$

No closed-form solution to maximize *l(W)!!!*

Logistic Regression – Learning W

Gradient Ascent

- Apply gradient ascent to maximize the conditional likelihood

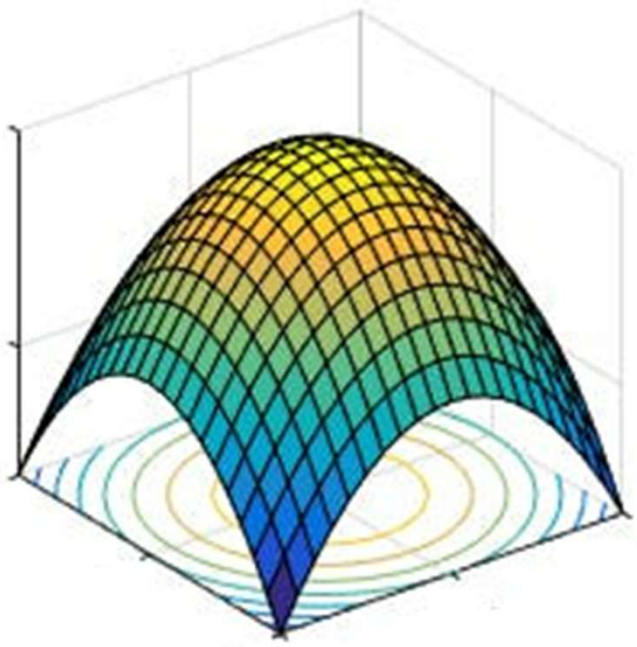
$$l(W) = \sum_j Y^{(j)} \left(w_0 + \sum_{i=1}^{n} w_i X_i^{(j)} \right) - ln \left(1 + exp \left(w_0 + \sum_{j=1}^{n} w_i X_i^{(j)} \right) \right)$$

The i^{th} component of the vector gradient has the form:

$$\frac{\partial l(W)}{\partial w_i} = \sum_j X_i^{(j)} (Y^{(j)} - \hat{P}(Y^{(j)} = 1 | Y^{(j)}, W))$$

Weights are update based on the gradients (γ is the learning rate):

$$w_i \leftarrow w_i + \gamma \sum_j X_i^{(j)} (Y^{(j)} - \hat{P}(Y^{(j)} = 1 | Y^{(j)}, W))$$



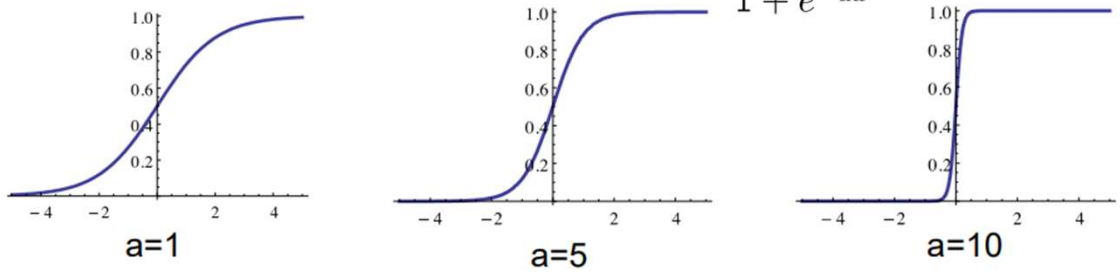
Note: $Y^{(j)} - \hat{P}(Y^{(j)} = 1 | Y^{(j)}, W)$ is the **prediction error** (difference between the observed value and the predicted probability)

Logistic Regression – Regularization

Overfitting

- Maximum likelihood can lead to severe overfitting if complex models are trained using data sets of limited size.
- Maximum likelihood prefers higher weights:

$$\frac{1}{1+e^{-ax}}$$



The higher the likelihood of (properly classified) examples close to decision boundary ⇨ The larger influence of corresponding features on decision

Consequence: overfitting!!

Logistic Regression – Regularization

Regularization: Penalize High Weights

Idea: Define priors on W

- Normal distribution, zero mean, identity covariance
- "Pushes" parameters towards zero

$$P(W) = \prod_i \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{w_i^2}{2\sigma^2}\right)$$

- **Regularization:** Helps avoid very large weights and overfitting
- Regularization distributes weights and avoids the model to rely on few weights associated to particular features, making it more robust.

Logistic Regression – Regularization

MAP Estimation as Regularization

Remember (MAP):
$P(\theta|D) \propto P(D|\theta)P(\theta)$

Adding the **prior** to the estimation:

$$W \leftarrow \arg\max_W \prod_j P(Y^{(j)}|X^{(j)}, W) \cdot \boldsymbol{P(W)}$$

The log-MAP objective with this Gaussian prior is then

$$l(W) = \sum_j ln\, P\left(Y^{(j)}\middle|X^{(j)}, W\right) + ln\, P(W)$$

Taking the log of the Prior (Gaussian), we have:

$$P(W) = -\frac{n}{2}\, ln(2\pi\sigma^2) - \sum_{i=1}^{n} \frac{w_i^2}{2\sigma^2}$$, where n is the number of weights/attributes

… ignoring the constant term $-\frac{n}{2}\, ln(2\pi\sigma^2)$, *which does not depend on w, we are left with* $-\sum_{i=1}^{n} \frac{w_i^2}{2\sigma^2}$

Logistic Regression – Regularization

MAP Estimation as Regularization

Adding the **prior** to the estimation:

$$W \leftarrow \arg\max_W \prod_j P(Y^{(j)}|X^{(j)}, W) \cdot \boldsymbol{P(W)}$$

Quadratic penalty (**L2-Regularizer**): drives weights towards zero

The log-MAP objective with this Gaussian prior is then

$$l(W) = \sum_j ln\, P\left(Y^{(j)}\middle|X^{(j)}, W\right) + ln\, P(W) = \sum_j ln\, P\left(Y^{(j)}\middle|X^{(j)}, W\right) - \frac{\lambda}{2} \sum_{i=1}^{n} \boldsymbol{w_i^2}$$

Remember the log of Gaussian was $-\sum_{i=1}^{n} \frac{w_i^2}{2\sigma^2}$

If we set the regularization parameter as $\boldsymbol{\lambda = \frac{1}{\sigma^2}}$ then we have the L2 regularization term in logistic regression.

In practice, the prior variance is often related to λ in the regularization term. So different values of λ correspond to different assumed variances in the Bayesian setting.

$\boldsymbol{\lambda}$ controls the strength of regularization

Logistic Regression – Regularization

MAP Estimation as Regularization

Adding the **prior** to the estimation:

$$W \leftarrow \arg\max_W \prod_j P(Y^{(j)}|X^{(j)}, W) \cdot \boldsymbol{P(W)}$$

The log-MAP objective with this Gaussian prior is then

$$l(W) = \sum_j ln\, P\left(Y^{(j)}\middle|X^{(j)}, W\right) + ln\, P(W) = \sum_j ln\, P\left(Y^{(j)}\middle|X^{(j)}, W\right) - \frac{\boldsymbol{\lambda}}{2} \sum_{i=1}^{n} \boldsymbol{w_i^2}$$

Quadratic penalty (**L2-Regularizer**): drives weights towards zero

$\boldsymbol{\lambda}$ controls the strength of regularization

Then, we have:

$$\frac{\partial l(W)}{\partial w_i} = \sum_j X_i^{(j)} (Y^{(j)} - \hat{P}(Y^{(j)} = 1|Y^{(j)}, W)) - \boldsymbol{\lambda w_i}$$

Adds a negative linear term to the gradients

Weight update rule:

$$w_i \leftarrow w_i + \gamma \sum_j X_i^{(j)} (Y^{(j)} - \hat{P}(Y^{(j)} = 1|Y^{(j)}, W)) - \boldsymbol{\gamma \lambda w_i}$$

NB vs Logistic Regression

Naïve Bayes versus Logistic Regression

Naïve Bayes (Generative Classifier)	**Logistic Regression (Discriminative Classifier)**
Assume functional form for $P(Y\|X)$, assumes **cond. independence** $P(Y)$ Estimate **likelihood** and **prior** from training data	Assume functional form for $P(Y\|X)$, **no assumptions**
Gaussian Naïve Bayes for continuous features	Handles discrete and continuous features
Bayes Rule to compute $P(Y\|X) \propto P(X\|Y)P(Y)$	**Directly** calculate $P(Y\|X)$
Indirect computation	
Can also generate a sample of the data	Can't generate data sample.

NB vs Logistic Regression

By the way… Remember Gaussian Naïve Bayes (GNB)?

Let's assume the following for GNB:

- Y is boolean, governed by a **Bernoulli** distribution, with parameter $\pi = P(Y = 1)$
- $X = \langle X_i, \ldots, X_n \rangle$, where each X_i is a **continuous random variable**
- For each X_i, $\mathbf{P}(\boldsymbol{X_i}|\boldsymbol{Y} = \boldsymbol{y_k})$ **is a Gaussian distribution** of the form $\mathcal{N}(\mu_{i,k}, \sigma_i)$
- For all i and j≠i, X_i and X_j are **conditionally independent** given Y

By deriving the parametric form of P(Y|X) that follows from this set of GNB assumptions, we will have:

$$P(Y = 0|X) = \frac{1}{1+\exp(w_0+\sum_{i=1}^{n} w_i X_i)} \quad \text{and} \quad P(Y = 1|X) = \frac{exp(w_0+\sum_{i=1}^{n} w_i X_i)}{1+exp(w_0+\sum_{i=1}^{n} w_i X_i)}$$

NB vs Logistic Regression

By the way… Remember Gaussian Naïve Bayes (GNB)?

- Under these assumptions, GNB implies the parametric form of P(Y|X) used in Logistic Regression.

- But only in special case (GNB with class-independent variances $\mathcal{N}(\mu_{i,\mathrm{k}}, \boldsymbol{\sigma_i})$)

- Besides, Logistic Regression makes no assumptions about P(X|Y) in learning!!!

GNB vs Logistic Regression

GNB vs LR Comparison:

When number of training examples tend to infinite:

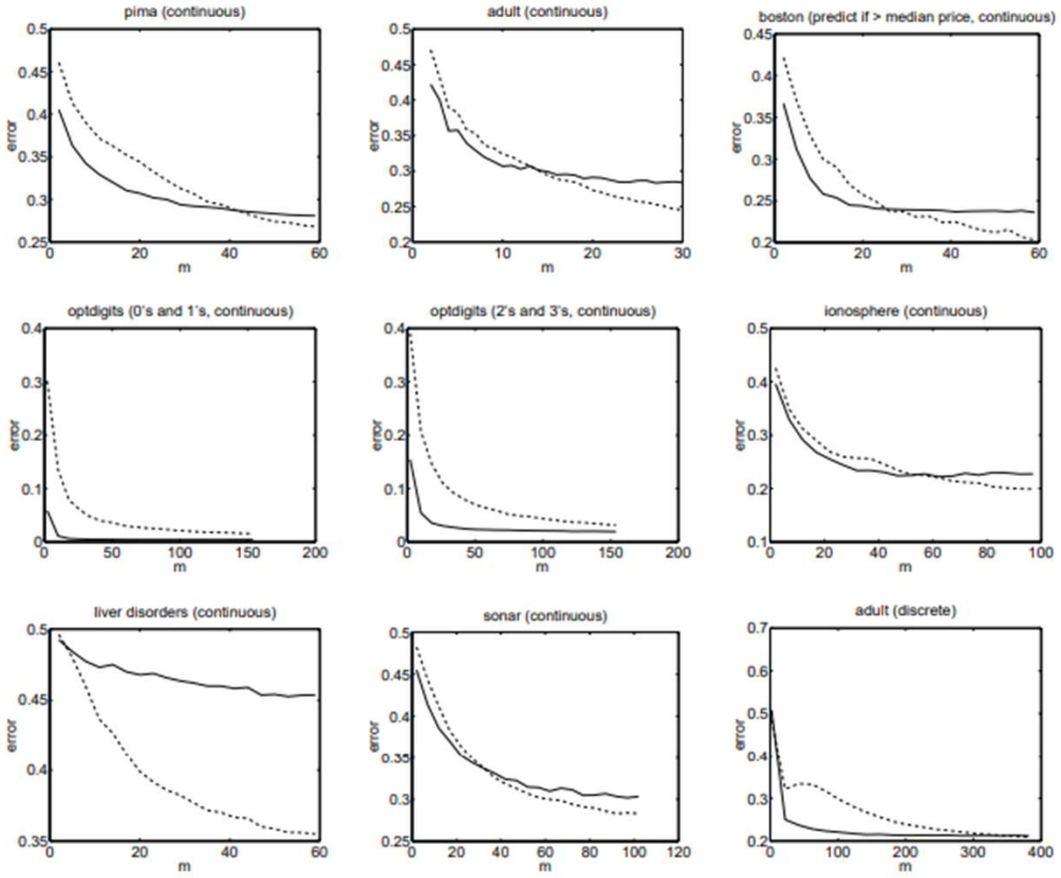
- GNB (with class independent variances) and LR produce identical classifiers
- When independence assumption on data does not hold, LR tend to produce better results.

Convergence rate of parameter estimates (where n = # of attributes in X):

- Naïve Bayes needs O(log n) samples
- Logistic Regression needs O(n) samples
  - **Logistic Regression parameter estimates converge more slowly, requiring order n examples.**

GNB vs Logistic Regression

GNB vs LR Emprirical Comparison (UCI datasets):



Naïve bayes

Logistic Regression

Sample size m

[Ng & Jordan, 2002]

Logistic Regression

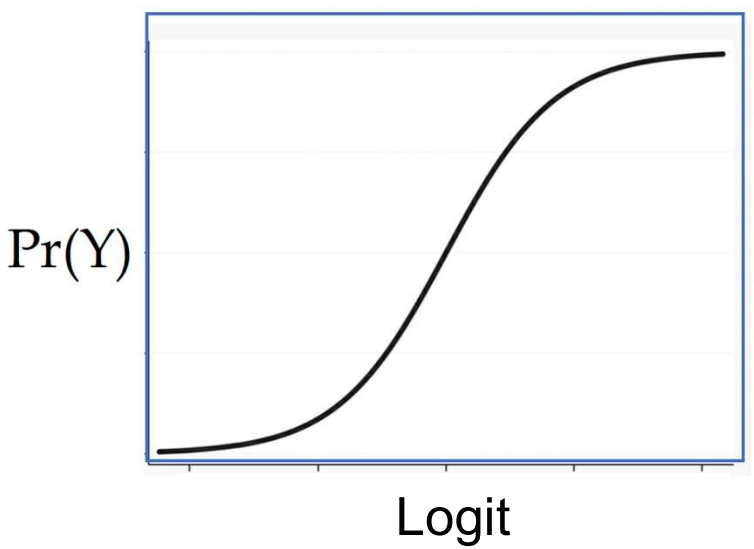
What you Need to Know about Logistic Regression

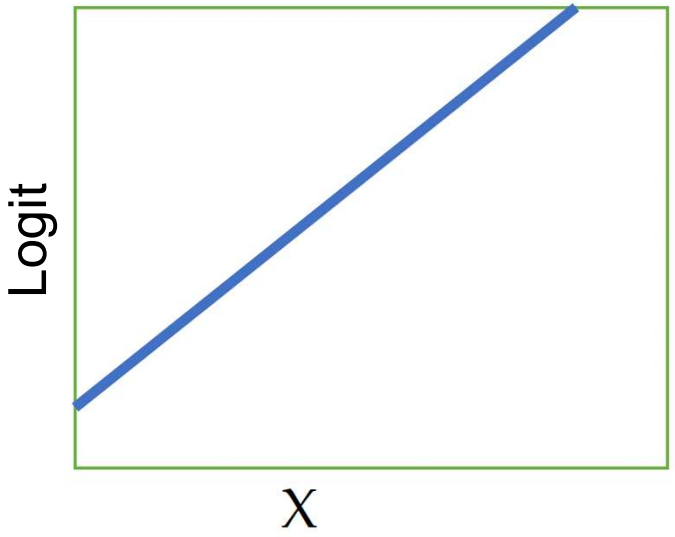
- Gaussian Naïve Bayes with class-independent variances is "representationally" equivalent to LR
- In general, NB and LR make different assumptions
  - NB: Features independent given class assumption on P(X|Y)
  - LR: Functional form of P(Y|X), no assumption on P(X|Y)
- LR is a linear classifier: decision rule is a hyperplane
- Great advantage: Interpretability!!!
- LR optimized by conditional likelihood
  - no closed-form solution
  - Concave shape → global optimum with gradient ascent
  - Maximum conditional a posteriori corresponds to regularization
- Convergence rates
  - GNB (usually) needs less data
  - LR (usually) gets to better solutions in the limit

Readings

Machine Learning by Tom Mitchell, Chapter "GENERATIVE AND DISCRIMINATIVE CLASSIFIERS: NAIVE BAYES AND LOGISTIC REGRESSION"

Appendix





Appendix

3.1 Form of $P(Y|X)$ for Gaussian Naive Bayes Classifier

Here we derive the form of $P(Y|X)$ entailed by the assumptions of a Gaussian Naive Bayes (GNB) classifier, showing that it is precisely the form used by Logistic Regression and summarized in equations (16) and (17). In particular, consider a GNB based on the following modeling assumptions:

- Y is boolean, governed by a Bernoulli distribution, with parameter $\pi = P(Y = 1)$
- $X = \langle X_1 \ldots X_n \rangle$, where each X_i is a continuous random variable
- For each X_i, $P(X_i|Y = y_k)$ is a Gaussian distribution of the form $N(\mu_{ik}, \sigma_i)$
- For all i and $j \neq i$, X_i and X_j are conditionally independent given Y

Appendix

Note here we are assuming the standard deviations σ_i vary from attribute to attribute, but do not depend on Y.

We now derive the parametric form of $P(Y|X)$ that follows from this set of GNB assumptions. In general, Bayes rule allows us to write

$$P(Y=1|X) = \frac{P(Y=1)P(X|Y=1)}{P(Y=1)P(X|Y=1)+P(Y=0)P(X|Y=0)}$$

Dividing both the numerator and denominator by the numerator yields:

$$P(Y=1|X) = \frac{1}{1+\frac{P(Y=0)P(X|Y=0)}{P(Y=1)P(X|Y=1)}}$$

or equivalently

$$P(Y=1|X) = \frac{1}{1+\exp(\ln\frac{P(Y=0)P(X|Y=0)}{P(Y=1)P(X|Y=1)})}$$

Because of our conditional independence assumption we can write this

$$\begin{aligned} P(Y=1|X) &= \frac{1}{1+\exp(\ln\frac{P(Y=0)}{P(Y=1)}+\sum_i \ln\frac{P(X_i|Y=0)}{P(X_i|Y=1)})} \\ &= \frac{1}{1+\exp(\ln\frac{1-\pi}{\pi}+\sum_i \ln\frac{P(X_i|Y=0)}{P(X_i|Y=1)})} \end{aligned} \tag{19}$$

Appendix

Note the final step expresses $P(Y=0)$ and $P(Y=1)$ in terms of the binomial parameter π.

Now consider just the summation in the denominator of equation (19). Given our assumption that $P(X_i|Y=y_k)$ is Gaussian, we can expand this term as follows:

$$\begin{aligned}
\sum_i \ln \frac{P(X_i|Y=0)}{P(X_i|Y=1)} &= \sum_i \ln \frac{\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(\frac{-(X_i-\mu_{i0})^2}{2\sigma_i^2}\right)}{\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(\frac{-(X_i-\mu_{i1})^2}{2\sigma_i^2}\right)} \\
&= \sum_i \ln \exp\left(\frac{(X_i-\mu_{i1})^2-(X_i-\mu_{i0})^2}{2\sigma_i^2}\right) \\
&= \sum_i \left(\frac{(X_i-\mu_{i1})^2-(X_i-\mu_{i0})^2}{2\sigma_i^2}\right) \\
&= \sum_i \left(\frac{(X_i^2-2X_i\mu_{i1}+\mu_{i1}^2)-(X_i^2-2X_i\mu_{i0}+\mu_{i0}^2)}{2\sigma_i^2}\right) \\
&= \sum_i \left(\frac{2X_i(\mu_{i0}-\mu_{i1})+\mu_{i1}^2-\mu_{i0}^2}{2\sigma_i^2}\right) \\
&= \sum_i \left(\frac{\mu_{i0}-\mu_{i1}}{\sigma_i^2}X_i + \frac{\mu_{i1}^2-\mu_{i0}^2}{2\sigma_i^2}\right) \qquad (20)
\end{aligned}$$

Appendix

Note this expression is a linear weighted sum of the X_i's. Substituting expression (20) back into equation (19), we have

$$P(Y=1|X) = \frac{1}{1+\exp(\ln\frac{1-\pi}{\pi} + \sum_i \left(\frac{\mu_{i0}-\mu_{i1}}{\sigma_i^2}X_i + \frac{\mu_{i1}^2-\mu_{i0}^2)}{2\sigma_i^2}\right))} \tag{21}$$

Or equivalently,

$$P(Y=1|X) = \frac{1}{1+\exp(w_0 + \sum_{i=1}^n w_i X_i)} \tag{22}$$

where the weights $w_1 \ldots w_n$ are given by

$$w_i = \frac{\mu_{i0}-\mu_{i1}}{\sigma_i^2}$$

and where

$$w_0 = \ln\frac{1-\pi}{\pi} + \sum_i \frac{\mu_{i1}^2-\mu_{i0}^2}{2\sigma_i^2}$$

Also we have

$$P(Y=0|X) = 1 - P(Y=1|X) = \frac{\exp(w_0 + \sum_{i=1}^n w_i X_i)}{1+\exp(w_0 + \sum_{i=1}^n w_i X_i)} \tag{23}$$