Discriminative vs. Generative Learning

CSE 4334 / 5334 Data Mining
Spring 2019

Won Hwa Kim

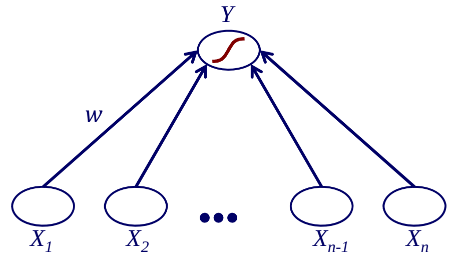
Part of the contents borrowed from Prof. Mark Craven / Prof. David Page Jr. at UW-Madison

Goals for this lecture

You should understand the following concepts

- logistic regression
- the relationship between logistic regression and naïve Bayes
- the relationship between discriminative and generative learning
- when discriminative/generative is likely to learn more accurate models

Logistic regression



- the same as a single layer **neural net** with a sigmoid in which the weights are trained to minimize

$$E(\boldsymbol{w}) = -\sum_{d \in D} \ln P\left(y^{(d)} | \boldsymbol{x}^{(d)}\right)$$

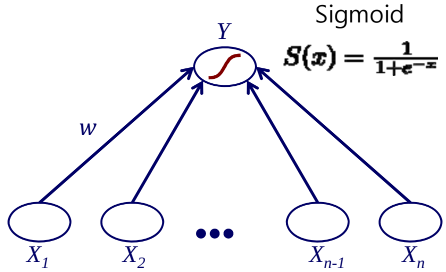
$$= \sum_{d \in D} -y^{(d)} \ln\left(o^{(d)}\right) - \left(1 - y^{(d)}\right) \ln\left(1 - o^{(d)}\right)$$

- the name is a misnomer since LR is used for classification

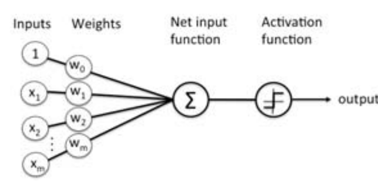
Logistic regression

Sigmoid

$$S(x) = \frac{1}{1+e^{-x}}$$



Perceptron



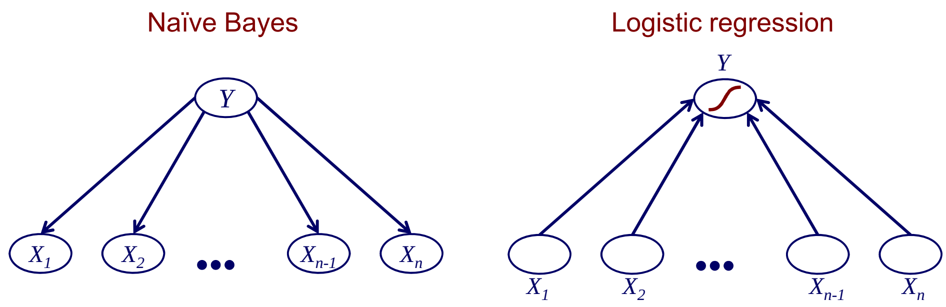
Schematic of Rosenblatt's perceptron.

- the same as a single layer **neural net** with a sigmoid

$$f(x) = \frac{1}{1 + e^{-\left(w_0 + \sum_{i=1}^{n} w_i x_i\right)}}$$

- the name is a misnomer since LR is used for classification

Naïve Bayes and Logistic regression



What's the difference?

- direction of the arrows?
- whether feature/variable names are inside the ovals or outside?
- sigmoid function?
- something else?

Naïve Bayes revisited

consider naïve Bayes for a binary classification task

$$P(Y=1 \mid x_1, \ldots, x_n) = \frac{P(Y=1)\prod_{i=1}^{n} P(x_i \mid Y=1)}{P(x_1, \ldots, x_n)}$$

expanding denominator

$$= \frac{P(Y=1)\prod_{i=1}^{n} P(x_i \mid Y=1)}{P(Y=1)\prod_{i=1}^{n} P(x_i \mid Y=1) + P(Y=0)\prod_{i=1}^{n} P(x_i \mid Y=0)}$$

dividing everything by numerator

$$= \frac{1}{1 + \dfrac{P(Y=0)\prod_{i=1}^{n} P(x_i \mid Y=0)}{P(Y=1)\prod_{i=1}^{n} P(x_i \mid Y=1)}}$$

Naïve Bayes revisited

Sigmoid

$$S(x) = \frac{1}{1+e^{-x}}$$

$$P(Y=1 \mid x_1, \ldots, x_n) = \frac{1}{1 + \dfrac{P(Y=0)\prod_{i=1}^{n} P(x_i \mid Y=0)}{P(Y=1)\prod_{i=1}^{n} P(x_i \mid Y=1)}}$$

applying $\exp(\ln(a)) = a$

$$= \frac{1}{1 + \exp\left(\ln\left(\dfrac{P(Y=0)\prod_{i=1}^{n} P(x_i \mid Y=0)}{P(Y=1)\prod_{i=1}^{n} P(x_i \mid Y=1)}\right)\right)}$$

applying $\ln(a/b) = -\ln(b/a)$

$$= \frac{1}{1 + \exp\left(-\ln\left(\dfrac{P(Y=1)\prod_{i=1}^{n} P(x_i \mid Y=1)}{P(Y=0)\prod_{i=1}^{n} P(x_i \mid Y=0)}\right)\right)}$$

Naïve Bayes revisited

Sigmoid

$$S(x) = \frac{1}{1+e^{-x}}$$

$$P(Y=1 \mid x_1, \ldots, x_n) = \frac{1}{1 + \exp\left(-\ln\left(\dfrac{P(Y=1)\prod_{i=1}^{n} P(x_i \mid Y=1)}{P(Y=0)\prod_{i=1}^{n} P(x_i \mid Y=0)}\right)\right)}$$

converting log of products to sum of logs

$$P(Y=1 \mid x_1, \ldots, x_n) = \frac{1}{1 + \exp\left(-\ln\left(\dfrac{P(Y=1)}{P(Y=0)}\right) - \sum_{i=1}^{n} \ln\left(\dfrac{P(x_i \mid Y=1)}{P(x_i \mid Y=0)}\right)\right)}$$

Does this look familiar?

Naïve Bayes revisited

Sigmoid

$$S(x) = \frac{1}{1+e^{-x}}$$

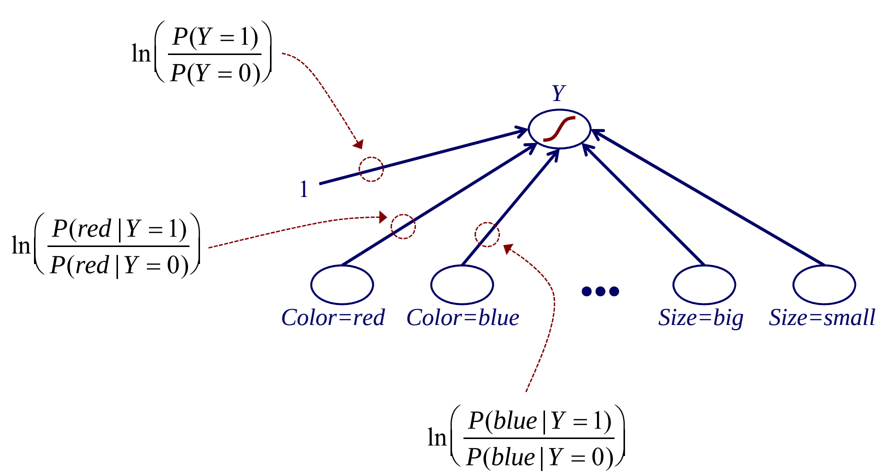
naïve Bayes

$$P(Y=1 \mid x_1, \ldots, x_n) = \frac{1}{1+e^{-\left(\ln\left(\frac{P(Y=1)}{P(Y=0)}\right) + \sum_{i=1}^{n} \ln\left(\frac{P(x_i|Y=1)}{P(x_i|Y=0)}\right)\right)}}$$

logistic regression

$$f(x) = \frac{1}{1+e^{-\left(w_0 + \sum_{i=1}^{n} w_i x_i\right)}}$$

Naïve Bayes as a neural net



weights correspond to log ratios

Naïve Bayes vs. Logistic regression

- they have the same functional form, and thus have the same hypothesis space bias (recall our discussion of inductive bias)
- Do they learn the same models?

  In general, **no**. They use different methods to estimate the model parameters.

  Naïve Bayes is a generative approach, whereas LR is a discriminative one.

Generative vs. discriminative learning

generative approach

learning: estimate $P(Y)$ and $P(X_1, \ldots, X_n \mid Y)$

classification: use Bayes' Rule to compute $P(Y \mid X_1, \ldots, X_n)$

discriminative approach

learn $P(Y \mid X_1, \ldots, X_n)$ directly

Naïve Bayes vs. Logistic regression

asymptotic comparison (# training instances ⟶ ∞)

- when conditional independence assumptions made by NB are correct, NB and LR produce identical classifiers

when conditional independence assumptions are incorrect

- logistic regression is less biased; learned weights may be able to compensate for incorrect assumptions (e.g. what if we have two redundant but relevant features)
- therefore LR expected to outperform NB when given lots of training data

Naïve Bayes vs. Logistic regression

non-asymptotic analysis [Ng & Jordan, *NIPS* 2001]

- consider convergence of parameter estimates; how many training instances are needed to get good estimates

  naïve Bayes: $O(\log n)$

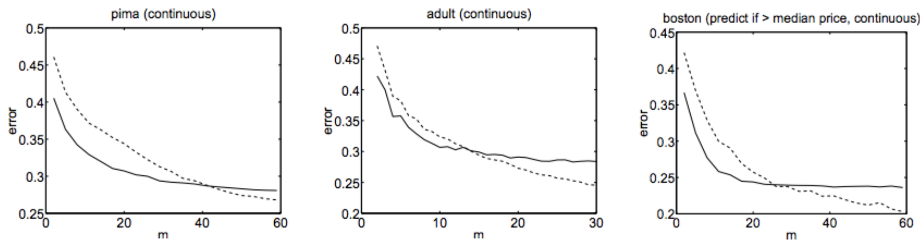
  logistic regression: $O(n)$

  n = # features

- naïve Bayes converges more quickly to its (perhaps less accurate) asymptotic estimates
- therefore NB expected to outperform LR with small training sets

Naïve Bayes vs. Logistic regression

-------- logistic regression
——— naïve Bayes

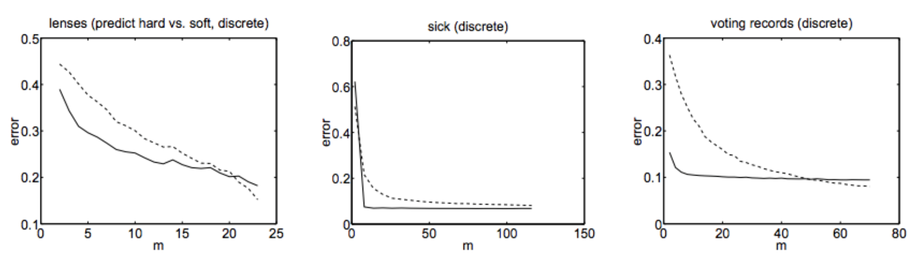


size of training set

Ng and Jordan compared learning curves for the two approaches on 15 data sets (some w/discrete features, some w/continuous features)

Naïve Bayes vs. Logistic regression

-------- logistic regression
——— naïve Bayes



general trend supports theory

- NB has lower predictive error when training sets are small
- the error of LR approaches or is lower than NB when training sets are large

Discussion

- NB/LR is one case of a pair of generative/discriminative approaches for the same model class
- if modeling assumptions are valid (e.g. conditional independence of features in NB) the two will produce identical classifiers in the limit (# training instances $\rightarrow \infty$)
- if modeling assumptions are not valid, the discriminative approach is likely to be more accurate for large training sets
- for small training sets, the generative approach is likely to be more accurate because parameters converge to their asymptotic values more quickly (in terms of training set size)
- Q: How can we tell whether our training set size is more appropriate for a generative or discriminative method? A: Empirically compare the two.