Introductory Applied Machine Learning

Naïve Bayes

Victor Lavrenko and Nigel Goddard
School of Informatics

Overview

- Naïve Bayes classifier
  - components and their function
  - independence assumption
  - dealing with missing data
- Continuous example
- Discrete example
- Pros and cons

Bayesian classification

- Goal: learning function $f(x) \rightarrow y$
  - y ... one of k classes (e.g. spam/ham, digit 0-9)
  - $x = x_1 \dots x_n$ – values of attributes (numeric or categorical)
- Probabilistic classification:
  - most probable class given observation: $\hat{y} = \arg\max_y P(y|x)$
- Bayesian probability of a class:

$$P(y|x) = \frac{\overbrace{P(x|y)}^{\text{class model}}\overbrace{P(y)}^{\text{prior}}}{\underbrace{\sum_{y'} P(x|y')P(y')}_{\text{normalizer } P(x)}}$$

Bayesian classification: components

$$P(y|x) = \frac{P(x|y)P(y)}{\sum_{y'} P(x|y')P(y')}$$

Example:
y … patient has Avian flu
x … observed symptoms

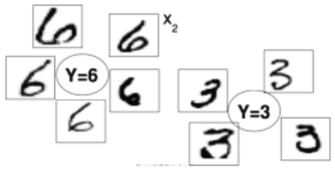
- $P(y)$: prior probability of each class
  - encodes how which classes are common, which are rare
  - apriori much more likely to have common cold than Avian flu
- $P(x|y)$: class-conditional model
  - describes how likely to see observation x for class y
  - assuming it's Avian flu, do the symptoms look plausible?
- $P(x)$: normalize probabilities across observations
  - does not affect which class is most likely (**arg max**)

Bayesian classification: normalization

Normalizer: $P(x) = \sum_{y'} P(x|y')P(y')$

- an "outlier" has a low probability under every class

P (X=x_1 | Y=3) < P(X=x_2 | Y=3)

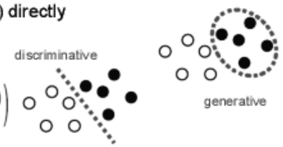


normalizer makes $P(Y{=}3|X{=}x_1)$ comparable to non-outliers

Naïve Bayes: a generative model

- A complete probability distribution for each class
  - defines likelihood for any point x
  - P(class) via P(observation) $\quad P(y|x) \propto P(x|y)P(y)$
  - can "generate" synthetic observations
    - will share many properties of the original data
- Not all probabilistic classifiers do this
  - possible to estimate $P(y|x)$ directly
  - e.g. logistic regression:

$$P(y|x) = \frac{1}{z_y} \exp\left(\sum_i \lambda_i g_i(y,x)\right)$$



Independence assumption

- Compute $P(x_1 \dots x_n|y)$ for every observation $x_1 \dots x_n$
  - class-conditional "counts", based on training data
  - problem: may not have seen every $x_1 \dots x_n$ for every y
    - digits: 2^{400} possible black/white patterns (20x20)
    - spam: every possible combination of words: $2^{10{,}000}$
  - often have observations for individual x_i for every class
- idea: assume $x_1 \dots x_n$ conditionally independent given y

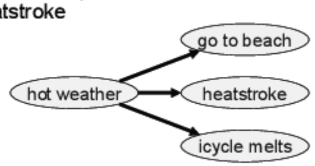
$$P(x_1 \dots x_d|y) = \underbrace{\prod_{i=1}^{d} P(x_i|x_1 \dots x_{i-1}, y)}_{\text{chain rule (exact)}} = \underbrace{\prod_{i=1}^{d} P(x_i|y)}_{\text{independence}}$$

Conditional independence

- Probabilities of going to the beach and getting a heat stroke are not independent: $P(B,S) > P(B)\,P(S)$
- May be independent if we know the weather is hot

$P(B,S|H) = P(B|H)\,P(S|H)$

- Hot weather "explains" all the dependence between beach and heatstroke
- In classification:
  - class value explains all the dependence between attributes



Example © Amos Storkey, 2007

Overview

- Naïve Bayes classifier
- Continuous example
  - general concepts
  - working example
  - example of failure
- Discrete example
  - general concepts
  - problems with Naïve Bayes
- Pros and cons

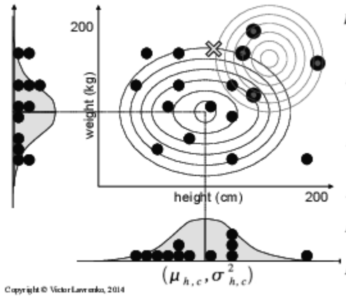
$$P(y|x) = \frac{\overbrace{P(x|y)}^{\text{class model}}\overbrace{P(y)}^{\text{prior}}}{\underbrace{\sum_{y'} P(x|y')P(y')}_{\text{normalizer } P(x)}}$$

Continuous example

- Distinguish children from adults based on size
  - classes: {*a*,*c*}, attributes: height [cm], weight [kg]
  - training examples: $\{h_i, w_i, y_i\}$, 4 adults, 12 children
- Class probabilities: $P(a)=\frac{4}{4+12}=0.25 \; ; P(c)=0.75$
- Model for adults:
  - height ~ Gaussian with mean, variance $\begin{cases} \mu_{h,a}=\frac{1}{4}\sum_{i:y_i=a} h_i \\ \sigma^2_{h,a}=\frac{1}{4}\sum_{i:y_i=a}(h_i-\mu_{h,a})^2 \end{cases}$
  - weight ~ Gaussian $(\mu_{w,a}, \sigma^2_{w,a})$
  - assume height and weight independent
- Model for children: same, using $(\mu_{h,c}, \sigma^2_{h,c}), (\mu_{w,c}, \sigma^2_{w,c})$

Continuous example

$P(a)=\frac{4}{4+12}=0.25 \; ; P(c)=0.75$



$$p(h_x|c)=\frac{1}{\sqrt{2\pi\sigma^2_{h,c}}}\exp-\frac{1}{2}\left(\frac{(h_x-\mu_{h,c})^2}{\sigma^2_{h,c}}\right)$$

$$p(w_x|c)=\frac{1}{\sqrt{2\pi\sigma^2_{w,c}}}\exp-\frac{1}{2}\left(\frac{(w_x-\mu_{w,c})^2}{\sigma^2_{w,c}}\right)$$

$$p(h_x|a)=\frac{1}{\sqrt{2\pi\sigma^2_{h,a}}}\exp-\frac{1}{2}\left(\frac{(h_x-\mu_{h,a})^2}{\sigma^2_{h,a}}\right)$$

$$p(w_x|a)=\frac{1}{\sqrt{2\pi\sigma^2_{w,a}}}\exp-\frac{1}{2}\left(\frac{(w_x-\mu_{w,a})^2}{\sigma^2_{w,a}}\right)$$

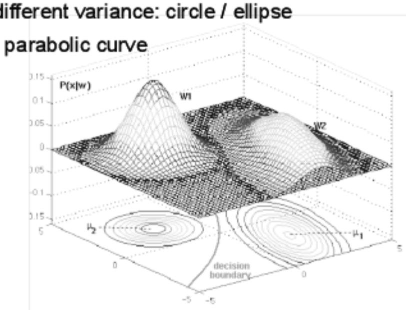
$$P(x|a)=p(h_x|a)\,p(w_x|a)$$

$$P(x|c)=p(h_x|c)\,p(w_x|c)$$

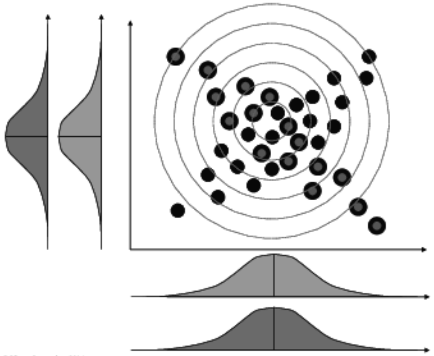
$$P(a|x)=\frac{P(x|a)P(a)}{P(x|a)P(a)+P(x|c)P(c)}$$

Decision boundary

- Different means, same variance: straight line / plane
- Same mean, different variance: circle / ellipse
- General case: parabolic curve



Problems with Naïve Bayes



Discrete example: spam

- Separate spam from valid email, attributes = words

D1: "send us your password"	spam
D2: "send us your review"	ham
D3: "review your password"	ham
D4: "review us"	spam
D5: "send your password"	spam
D6: "send us your account"	spam

new email: "review us now"

P (spam) = 4/6 P (ham) = 2/6

spam	ham	
2/4	1/2	password
1/4	2/2	review
3/4	1/2	send
3/4	1/2	us
3/4	1/2	your
1/4	0/2	account

$$P(\text{review us}\,|\text{spam})=P(0,1,0,1,0,0\,|\text{spam})=(1-\tfrac{2}{4})(\tfrac{1}{4})(1-\tfrac{3}{4})(\tfrac{3}{4})(1-\tfrac{3}{4})(1-\tfrac{1}{4})$$

$$P(\text{review us}\,|\text{ham})=P(0,1,0,1,0,0\,|\text{ham})=(1-\tfrac{1}{2})(\tfrac{2}{2})(1-\tfrac{1}{2})(\tfrac{1}{2})(1-\tfrac{1}{2})(1-\tfrac{0}{2})$$

$$P(\text{ham}|\text{review us})=\frac{0.0625\times 2/6}{0.0625\times 2/6+0.0044\times 4/6}=0.87$$ (note identical example)

Problems with Naïve Bayes

- Zero-frequency problem
  - any mail containing "account" is spam: P(account|ham) = 0/2
  - solution: never allow zero probabilities
    - Laplace smoothing: add a small positive number to all counts: $P(w|c)=\frac{num(w,c)+\epsilon}{num(c)+2\epsilon}$
    - may use global statistics in place of ε: *num(w) / num*
  - very common problem (Zipf's law: 50% words occur once)
- Assumes word independence
  - every word contributes independently to P(spam|email)
  - fooling NB: add lots of "hammy" words into spam email

Overview

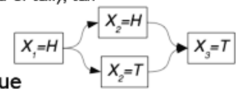
- Naïve Bayes classifier
- Continuous example
- Discrete example
- Pros and Cons
  - dealing with missing data
  - computational cost and incremental updates

Missing data

- Suppose don't have value for some attribute X_i
  - applicant's credit history unknown
  - some medical test not performed on patient
  - how to compute $P(X_1=x_1 \ldots X_j=? \ldots X_d=x_d \mid y)$
- Easy with Naïve Bayes
  - ignore attribute in instance where its value is missing $P(x_1 \ldots \boxed{X_j} \ldots x_d|y)=\prod_{i\neq j}^{d} P(x_i|y)$
  - compute likelihood based on observed attribtues
  - no need to "fill in" or explicitly model missing values
  - based on conditional independence between attributes

Missing data (2)

- Ex: three coin tosses: Event = $\{X_1=H, X_2=?, X_3=T\}$
  - event = head, unknown (either head or tail), tail
  - event = {H,H,T} + {H,T,T}
  - *P*(event) = P(H,H,T) + P(H,T,T)



- General case: X_j has missing value

$$P(x_1 \ldots \boxed{x_j} \ldots x_d|y)=P(x_1|y)\cdots \boxed{P(x_j|y)} \cdots P(x_d|y)$$

$$\sum_{\boxed{x_j}} P(x_1 \ldots \boxed{x_j} \ldots x_d|y)=\sum_{x_j} P(x_1|y)\cdots \boxed{P(x_j|y)} \cdots P(x_d|y)$$

$$=P(x_1|y)\cdots \boxed{\sum_{x_j} P(x_j|y)} \cdots P(x_d|y)$$

$$=P(x_1|y)\cdots \boxed{1} \cdots P(x_d|y)$$

Summary

- Naïve Bayes classifier $P(y|x) = \frac{P(x|y)P(y)}{\sum_{y'} P(x|y')P(y')}$
  - explicitly handles class priors
  - “normalizes” across observations: outliers comparable
  - assumption: all dependence is “explained” by class label
- Continuous example
  - unable to handle correlated data
- Discrete example
  - fooled by repetitions
  - must deal with zero-frequency problem
- Pros:
  - handles missing data
  - good computational complexity
  - incremental updates

Computational complexity

- One of the fastest learning methods
- $O(nd+cd)$ training time complexity
  - c … number of classes
  - n … number of instances
  - d … number of dimensions (attributes)
  - both learning and prediction
  - no hidden constants (number of iterations, etc.)
  - testing: O(ndc)
- $O(dc)$ space complexity
  - only decision trees are more compact

Incremental updates

- Allows incremental updates: $O(d)$ insertion / deletion
- Bernoulli: store raw counts instead of probabilities
  - new example of class c:
    - n_{cd} += x_d for each d in example, n_c += 1, n += 1
  - when need to classify:
    - $P(x_d=1 \mid c) = (n_{cd} + \varepsilon) / (n_c + 2\varepsilon)$
    - $P(c) = n_c / n$
- Gaussian: store partial sums instead of mean/variance
  - s_{cd} += x_d s^2_{cd} += x_d^2
  - when need to classify:

    mean $= s_{cd} / n$ variance $= s^2_{cd} / n -$ mean^2

General structure for Naïve Bayes

- **Task**
  - c-class classification ($c \geq 2$)
- **Model structure**
  - $c \times d$ independent distributions
  - continuous: Gaussian, discrete: Bernoulli
- **Score function**
  - class-conditional likelihood
- **Optimization / search method**
  - analytic solution
- Book: section 4.2