

情報意味論 (第6回) ベイズ学習

慶應義塾大学工学部
櫻井 彰人

目次

- Bayes 定理
- MAP と ML
- Bayes 最適分類器, Gibbs アルゴリズム
- クラスの推定か確率の推定か
- Naïve Bayes

Bayes の定理

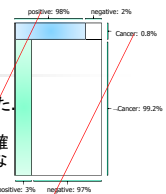
$$P(A | B) = \frac{P(B | A) P(A)}{P(B)}$$



$$P(A, B) = P(A | B) P(B) \\ = P(B | A) P(A)$$

例 (Mitchell Chap. 6.2)

ある患者がガンの検査を受けたところ結果が陽性であった。この患者には、本当に病変があるのだろうか？
なお、当該検査は、本当に病変があるときに陽性となる確率は 98% を誇る。また、病変がないときに正しく陰性となる確率は 97% である。



さらに、全人口に対するこのガンをもつ率は .008 である。

$$\begin{aligned} P(\text{cancer}) &= .008 & P(\neg\text{cancer}) &= .992 \\ P(+ | \text{cancer}) &= .98 & P(- | \text{cancer}) &= .02 \\ P(+ | \neg\text{cancer}) &= .03 & P(- | \neg\text{cancer}) &= .97 \\ P(+) &= P(+ | c^+r) P(c^+r) + P(+ | -c^+r) P(-c^+r) = .0376 \\ P(\text{cancer} | +) &= \frac{P(+ | \text{cancer}) P(\text{cancer})}{P(+)} = .209 \end{aligned}$$

例 (Mitchell Exercise 6.1)

2回目の検査(2回は統計的に独立とする)を受け、その結果も陽性であったとしよう。ガンである事後確率はどうなるであろうか？

$$\begin{aligned} P(\text{cancer}) &= .008 & P(\neg\text{cancer}) &= .992 \\ P(+ | \text{cancer}) &= .98 & P(- | \text{cancer}) &= .02 \\ P(+ | \neg\text{cancer}) &= .03 & P(- | \neg\text{cancer}) &= .97 \\ P(+_1+_2) &= P(+_1+_2 | c^+r) P(c^+r) + P(+_1+_2 | -c^+r) P(-c^+r) = .00858 \\ P(\text{cancer} | +_1+_2) &= \frac{P(+_1+_2 | \text{cancer}) P(\text{cancer})}{P(+_1+_2)} = .896 \end{aligned}$$

良く使う公式

乗法の公式 (実は、条件付確率の定義！):

$$P(A \wedge B) = P(A | B) P(B) = P(B | A) P(A)$$

参考: 和事象に対しては

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$$

全確率の公式:

$$P(B) = \sum_{i=1}^n P(B, A_i) = \sum_{i=1}^n P(B | A_i) P(A_i)$$

仮説選択に関して教えてくれること

$$P(h | D) = \frac{P(D | h) P(h)}{P(D)}$$

$P(h)$ = 仮説 h の事前確率

$P(D)$ = 訓練データ D の生起確率

$P(h|D)$ = D が与えられたときの h の事後確率

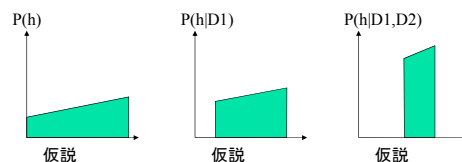
$P(D|h)$ = h が与えられたときの D の生起確率

データ D を生成したらしい仮説 h を選択することができる！

注: 条件付確率は因果関係(もしあれば)を反映するわけではない

注: "仮説の生起確率" を考えることができるのだろうか？

ノイズがないときの事後確率の進展



目次

- Bayes 定理
- MAP と ML
- Bayes 最適分類器, Gibbs アルゴリズム
- クラスの推定か確率の推定か
- Naïve Bayes

MAP推定

$$P(h | D) = \frac{P(D | h) P(h)}{P(D)}$$

データが所与のとき、必要とするのは、最もありうべき仮説であろう。

事後確率最大仮説 (Maximum a posteriori hypothesis) h_{MAP} :

$$\begin{aligned} h_{MAP} &= \arg \max_{h \in H} P(h | D) \\ &= \arg \max_{h \in H} \frac{P(D | h) P(h)}{P(D)} \\ &= \arg \max_{h \in H} P(D | h) P(h) \end{aligned}$$

ML推定

全ての i, j について $P(h_i) = P(h_j)$ と仮定すれば、より簡単化でき、最尤Maximum

Likelihood (ML) 仮説 を選ぶことになる

$$h_{MAP} = \arg \max_{h \in H} P(D | h) P(h)$$

$$h_{ML} = \arg \max_{h \in H} P(D | h)$$

ML推定の一つの解釈

- 現実世界では、事前確率分布は、未知か、計算不能か、存在しないと思われる
 - 例えば、文書における単語の生起頻度の事前分布はあるのだろうか？ 年齢、社会的背景、人口分布で大きく異なりうる
- 事前確率分布が存在しないとしたら、尤度最大化は自然な考え

尤度最大化は、各仮説の生起確率がすべて等しいとした場合と等価である。つまり仮説の事前確率分布が一様であるとの仮定と等価である。妥当か？

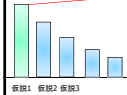
目次

- Bayes 定理
- MAP と ML
- Bayes 最適分類器, Gibbs アルゴリズム
- クラスの推定か確率の推定か
- Naïve Bayes

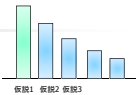
未知事例の最もありうる分類

- これまで、事例 D のもとでの最もありうる仮説を求めてきた (例: h_{MAP})。
- 未知事例の最もありうる (最も確率が高い) 分類結果はどうなるのでしょうか？

- $h_{MAP}(x)$ は最もありうる分類ではない！
 - 次の例で、 x のもっともありうる類別は？
 - 3仮説: $P(h_1|D)=0.4, P(h_2|D)=0.3, P(h_3|D)=0.3$
 - 新事例: $h_1(x)=+, h_2(x)=-, h_3(x)=-$



Bayes 最適な分類器



$$\arg \max_{c_j \in \{+, -\}} \sum_{h_i \in H} P(c_j | h_i) P(h_i | D)$$

注: Bayes 最適な分類器は H に含まれるとは限らない
 注: 論文にはうまくくと報告されているのだが、試してみるとMAPやMLと変わらない場合がある。どのような場合にそうなるか、興味のあるところである
 注: 実行可能か? 見るからに時間がかかりそう

例 (Mitchell Chap. 6.7)

$$\begin{array}{lll} P(h_1 | D) = .4 & P(- | h_1) = 0 & P(+ | h_1) = 1 \\ P(h_2 | D) = .3 & P(- | h_2) = 1 & P(+ | h_2) = 0 \\ P(h_3 | D) = .3 & P(- | h_3) = 1 & P(+ | h_3) = 0 \end{array}$$

それゆえ:

$$\sum_{h_i \in H} P(+ | h_i) P(h_i | D) = .4$$

$$\sum_{h_i \in H} P(- | h_i) P(h_i | D) = .6$$

そして:

$$\arg \max_{c_j \in \{+, -\}} \sum_{h_i \in H} P(c_j | h_i) P(h_i | D) = -$$

Gibbs 分類器 – 速度向上

1. 仮説を $P(h|D)$ に従ってランダムに選ぶ
2. 新事例をこれに従い分類する

慶賀: もし仮説を事前分布 $P(h)$ に従ってランダムに選ぶと,

$$E[\text{error}_{\text{Gibbs}}] \leq 2E[\text{error}_{\text{BayesOptimal}}]$$

(詳細は “Mitchell Machine Learning Chap. 6.8”)

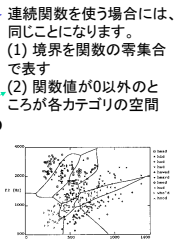
仮説の個数が多くて、ベイズ最適な分類器が計算できないときに有用

目次

- Bayes 定理
- MAP と ML
- Bayes 最適分類器, Gibbs アルゴリズム
- クラスの推定か確率の推定か
- Naïve Bayes

学習目標値には2通り

- カテゴリ: 分類問題
 - (説明変数の)空間を部分空間に分割
 - 分割する境界を得る
 - カテゴリに数値を対応させる
 - 連続値: 回帰問題
 - 離散値も連続値の一部と考える
- ただし、不連続関数になるので、回帰問題化には注意が必要
誤り数最小化とするのが妥当



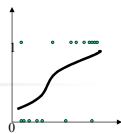
ところが

- カテゴリ値の場合 (境界で0をとる連続関数を探すとき)
 - 関数値が0に近い=境界に近い=判断に迷い
 - 仮に、推定の確信度合いを、0から1の実数で表せば、カテゴリを表す部分空間で
 - 中ほど=判断に自信=値は1に近い、
 - 境界に近い=判断に迷い=値は0に近い
- すると、回帰問題と考えられる。
 - 値は、カテゴリに振った番号



見方を変えて

- 確率とサンプル点の個数
 - ある点での(あるクラスに属するという)割合を、確定値ではなく、確率で考えることにすれば、その確率が、その付近にあるサンプル点の個数に現れると考える
- 確信度合いとサンプル点の個数
 - ある点での(今いる部分空間にいるという)確信度合いはその付近にあるサンプル点の個数に比例すると考える



まとめると

| 目標値 | カテゴリ値 | 連続値 | |
|--------|-----------------------------|---------------|--------------------|
| 考え方と方法 | 誤り数最小化(境界の推定) | 目標値の回帰(誤差最小化) | |
| | カテゴリ値を目標値とした回帰(誤差最小化) | | 出力値を丸めてカテゴリ値と解釈 |
| | | | 出力値とカテゴリ値の違いを確率と解釈 |
| | 入力点を確率分布のサンプルと考え、分布推定(密度推定) | | |

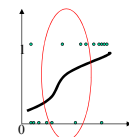
ニューラルネット学習方法2通り

- 分類**

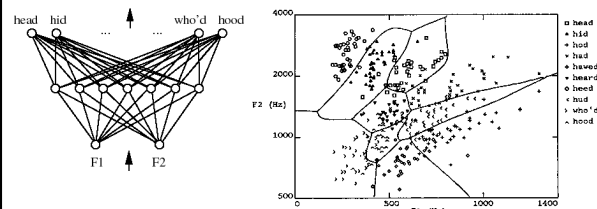
教えるのは分類クラス。しかし出力値を確率と解釈可能

回帰

教えるのは分類クラス。出力値は丸めて分類クラスと解釈
- 標準シグモイド関数を用いると、出力値は0と1の間
- これが可能なのは、過学習しにくいからである(右図) online 学習なら resampling していることとほぼ同じ



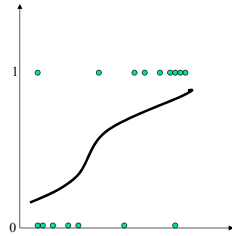
再掲: クラスと境界



出力値は、実数であり、シグモイド関数を出力関数とすると、例えば[0,1]とすることができる。クラスごとに出力素子を用意すれば、そのクラスへの所属確率を出力しているとも解釈できる。
教師値等は、クラス種別でもよいが、その「クラスらしさ」ともできる。後者の場合、確率の学習(回帰)とも考えることができる。

確率を予測するように学習する

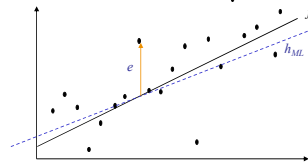
クラス種別を教えたのに、確率を学習してしまうとは？



online学習で、頻度を学習する。
個数を数える。

実数値の学習(回帰分析)

ところで、回帰分析とは？



回帰分析の統計的解釈

学習事例: $\langle x_i, d_i \rangle$ 但し

$$d_i = f(x_i) + e_i$$

e_i はノイズ = iid なる正規分布に従う確率変数
で、平均=0 かつ分散は有限とする

iid=independent, identically distributed

ならば (予想): 次のスライドで証明 random variable

$$h_{ML} = \arg \min_{h \in H} \sum_{i=1}^m (d_i - h(x_i))^2$$

回帰分析の統計的解釈(証明)

$$\begin{aligned} h_{ML} &= \arg \max_{h \in H} \ln p(D | h) \\ &= \arg \max_{h \in H} \ln \prod_{i=1}^m e^{-\frac{1}{2} \left(\frac{d_i - h(x_i)}{\sigma} \right)^2} \\ &= \arg \max_{h \in H} \sum_{i=1}^m -\frac{1}{2} \left(\frac{d_i - h(x_i)}{\sigma} \right)^2 \\ &= \arg \max_{h \in H} \sum_{i=1}^m -(d_i - h(x_i))^2 \\ &= \arg \min_{h \in H} \sum_{i=1}^m (d_i - h(x_i))^2 \end{aligned}$$

確率の予測には二乗誤差は不適

- 例: 生存確率を患者データから学習しよう

$$\begin{aligned} h_{ML} &= \arg \max_{h \in H} \ln p(D | h) && d_i \text{ は } 0 \text{ or } 1 \text{ (または} \\ &= \arg \max_{h \in H} \ln \prod_{i=1}^m P(d_i | h, x_i) P(x_i) && \text{所属確率)} \\ &= \arg \max_{h \in H} \sum_{i=1}^m \ln [P(d_i | h, x_i) P(x_i)] \\ &= \arg \max_{h \in H} \sum_{i=1}^m \ln (h(x_i)^{d_i} (1 - h(x_i))^{1-d_i} P(x_i)) \\ &= \arg \max_{h \in H} \sum_{i=1}^m d_i \ln h(x_i) + (1 - d_i) \ln (1 - h(x_i)) \end{aligned}$$

注: cross entropy $H(p, q) = -\sum_x p(x) \log q(x) = H(p) + D_{KL}(p \| q)$

目次

- Bayes 定理
- MAP と ML
- Bayes 最適分類器, Gibbs アルゴリズム
- クラスの推定か確率の推定か
- Naïve Bayes

Naïve Bayes 分類器

- 単純だが(だから?)よく知られた分類方法
 - 単純な割には高精度
 - 単純なだけに、高速
- Bayes 定理 + 仮定 **条件付独立**
 - 実際には成り立たないことが多い仮定
 - それにも関わらず、実際にはしばしばうまくいく
- 成功事例:
 - 文書分類
 - 診断

Bayes 定理を使う場合の課題

- 変数 x の属性 $\langle a_1, \dots, a_n \rangle$ が与えられたとき, x が属するクラス c を最尤推定するには?

$$c_{MAP} = \arg \max_{c_j \in C} P(c_j | a_1, a_2, \dots, a_n)$$

$$= \arg \max_{c_j \in C} \frac{P(a_1, a_2, \dots, a_n | c_j) P(c_j)}{P(a_1, a_2, \dots, a_n)}$$

$$= \arg \max_{c_j \in C} P(a_1, a_2, \dots, a_n | c_j) P(c_j)$$

- 問題: 大量のデータが $P(a_1 \dots a_n | c_j)$ を推定するのに必要. パラメータ数が膨大 ($\prod |A_i|$) (2値属性の場合、属性数が n なら 2^n 個)だから

Naïve Bayes 分類器

- **Naïve Bayes の仮定**: 属性同士は、属するクラスが所与なら、独立
 - $P(a_1, \dots, a_n | c_j) = P(a_1 | c_j) P(a_2 | c_j) \dots P(a_n | c_j)$
 - **条件付独立性** (クラスが所与の時)
 - 推定すべきパラメータ数の削減: $\prod |A_i| (=O(2^n)) \rightarrow \sum |A_i| (=O(n))$
- この仮定のもと, c_{MAP} は

$$c_{NB} = \arg \max_{c_j \in C} P(c_j) \prod_i P(a_i | c_j)$$

Naïve Bayes: アルゴリズム

学習(事例集合)

事例がクラス c_j に属する確率

$$P^*(c_j) = P(c_j) \text{ の推定値}$$

クラス c_j に属する事例の i 番目の属性の属性値が a_i である確率

$$P^*(a_i | c_j) = P(a_i | c_j) \text{ の推定値}$$

分類(x)

$$c_{NB} = \arg \max_{c_j \in C} \hat{P}(c_j) \prod_i \hat{P}(a_i | c_j)$$

Naïve Bayes: 推定

- どうやって $P(c_j)$ と $P(a_i | c_j)$ を推定するか?
 - 統計学が教える標準的な方法
 - サンプルの頻度から確率を推定する
 - $P(c)$ の推定値は $\text{count}(c) / N$
 - $P(A|B)$ の推定値は $\text{count}(A \wedge B) / \text{count}(B)$
 - 例: 100 事例. 内訳 70 + と 30 -
 - $P(+)=0.7$ かつ $P(-)=0.3$
 - 70 個の正例のなかに, 35 個で $a_1=\text{SUNNY}$
 - $P(a_1=\text{SUNNY} | +)=0.5$

例

| Day | Outlook | Temperature | Humidity | Wind | Play Tennis |
|-------|----------|-------------|----------|--------|-------------|
| Day1 | Sunny | Hot | High | Weak | No |
| Day2 | Sunny | Hot | High | Strong | No |
| Day3 | Overcast | Hot | High | Weak | Yes |
| Day4 | Rain | Mild | High | Weak | Yes |
| Day5 | Rain | Cool | Normal | Weak | Yes |
| Day6 | Rain | Cool | Normal | Strong | No |
| Day7 | Overcast | Cool | Normal | Strong | Yes |
| Day8 | Sunny | Mild | High | Weak | No |
| Day9 | Sunny | Cool | Normal | Weak | Yes |
| Day10 | Rain | Mild | Normal | Weak | Yes |
| Day11 | Sunny | Mild | Normal | Strong | Yes |
| Day12 | Overcast | Mild | High | Strong | Yes |
| Day13 | Overcast | Hot | Normal | Weak | Yes |
| Day14 | Rain | Mild | High | Strong | No |

$$P(Y) = 9/14,$$

$$P(\text{sunny} | Y) = 2/9,$$

$$P(\text{cool} | Y) = 3/9,$$

$$P(\text{high} | Y) = 3/9,$$

$$P(\text{strong} | Y) = 3/9$$

Naïve Bayes: 例

- 例の *PlayTennis*, と新事例
<Outk=sun, Temp=cool, Humid=high, Wind=strong>

- 計算したいのは:

$$c_{NB} = \arg \max_{c_j \in C} \hat{P}(c_j) \prod_i \hat{P}(a_i | c_j)$$

- $\hat{P}(Y)\hat{P}(sun|Y)\hat{P}(cool|Y)\hat{P}(high|Y)\hat{P}(strong|Y) = 0.005$
 $\hat{P}(N)\hat{P}(sun|N)\hat{P}(cool|N)\hat{P}(high|N)\hat{P}(strong|N) = 0.021$

$$\Rightarrow c_{NB} = No$$

Naïve Bayes: 条件付独立は必須か?

- もし仮定が成り立たなかったら?
 - i.e. if $P(a_1, \dots, a_n | c_j) \neq P(a_1 | c_j) P(a_2 | c_j) \dots P(a_n | c_j)$
- それでも、下記の(弱い)条件が成り立つ限り、予測値は Bayes 予測値と等価:

$$\arg \max_{c_j \in C} P(a_1 | c_j) P(a_2 | c_j) \dots P(a_n | c_j) P(c_j)$$

$$= \arg \max_{c_j \in C} P(a_1, a_2, \dots, a_n | c_j) P(c_j)$$

- しかし、予測時に求める 確率は 0 や 1 に極めて近い非現実的な値になりうる

Naïve Bayes: ある問題

- もしも、あるクラス c_j で属性値 a_i が観測されなかったら?
 - 推定値 $P(a_i | c_j) = 0$ なぜなら $\text{count}(a_i \wedge c_j) = 0$?
 - 影響は甚大: これが 0 だと積は 0!
- 解: Laplace correction を用いる
 - $\hat{P}(a_i | c_j) = \frac{n_c + mp}{n + m}$
 - n 訓練例数. 但し $c = c_j$
 - n_c 訓練例数. 但し $c = c_j$ かつ $a = a_i$
 - p 事前確率(の推定値) $P^*(a_i | c_j)$ (通常は一様分布)
 - m “仮想” 事例数(しばしば、当該属性 a の属性値の個数を用いる)
 - $m=1$ とする方法がある。その方が結果がよいことがある

文書分類



- 文書分類とは:
 - 文書(メール、ニュース、webページ等々。それらの一段落ということも、また、一文ということも)を分類すること
 - 分りやすいのは、メールがスパムか否かの分類
 - ブログを、スブログか否かに分類する、とうい課題もある
 - ニュースが(ある人にとって)興味のあるものか否かを分類する、というのものもある。さらに、
 - ある商品の評判を(良い評判も悪い評判も)集めるにも「分類」が必要。そして、良い評判と悪い評判とに分ける。
 - 信頼できる評価か信用できない評価かに分けるのも、文書分類
 - アンケート調査のうち、自由記述文の分類。
 - コールセンターでの、QAの分類
- Naive Bayes が結構うまくいく
 - どうやって Naive Bayes を用いるか?
 - ポイント: どう事例(すなわち、1文書)を表現するか? 属性は何か?

文書の表現方法

- Bag-of-words, すなわち、袋一杯の単語 or 袋詰めめの単語
 - ある文書を、それぞれの単語が何回現れたかで表現する。
 - "Bag" で、もとの文書のどこにあったかを忘れることを表している。
 - また、単語の連なりも考えないことを表している。
 - 例えば、仮に、「慶應」「義塾」「大学」がそれぞれ単語なら、「慶應義塾大学」も「慶應大学義塾」も「義塾慶應大学」も同じと考えることになる。
 - 「何を単語とするか」が結構重要。文書ごとによって変わってはいけない。
 - 英語であれば、dog と dogs といったような語形変化は無視した方がよい。
 - 文書分類に役立ちそうもない単語は考えない
 - 日本語で言えば、助詞(は、が、も、や、...)がその代表。
 - 英語で言えば、前置詞がその代表
 - こういった、文法機能を持ち、単語単独では意味のない単語を機能語という
 - ノイズの可能性が高い単語は考えない。
 - 文書集合内で、出現頻度が極めて低い(一回等)もの

文書の表現方法(続)

- 表現自体が naive Bayes 的
 - ベイズ推論とは直接には関係しないので、naive Bayes ではないが、naive な表現であることは間違いない。
- しかし、naive Bayes 的に、文書の出現確率を書くことができる。
- 文書の属するクラスごとに、文書内にある特定の単語が出現する確率 $P(w_1 | c_j), P(w_2 | c_j), \dots, P(w_n | c_j)$ が決まっているとすると、 w_1, w_2, \dots, w_n が文書中に含まれる単語であるとき、そのような文書が出現する確率を次のように書く

$$P(\text{doc} | c_j) = P(w_1 | c_j)^{\text{TF}(w_1)} P(w_2 | c_j)^{\text{TF}(w_2)} \dots P(w_n | c_j)^{\text{TF}(w_n)}$$
 ただし $\text{TF}(w)$ は単語 w の doc における出現度数(term frequency)

出現確率をこう書けば naive Bayes といえよう

Naïve Bayes による文書分類

- ある文書 doc につき

$$c_{NB} = \arg \max_{c_j \in C} P(c_j) \prod_{w_k \in \text{Voc}} P(w_k | c_j)^{TF(w_k, \text{doc})}$$

ただし、 $TF(w_k, \text{doc}) = \text{doc}$ 中の w_k の出現度数、Voc は全単語(考えている全単語)集合とした

- 単語の出現確率については、Laplace correction が必須。そこで、下記の推定値を使用; ただし、 $n_j = \text{クラス } c_j \text{ 中の全単語出現度数}$ 、 $n_{k,j} = \text{クラス } c_j \text{ 中の単語 } w_k \text{ 出現度数}$

$$P(w_k | c_j) = \frac{n_{k,j} + 1}{n_j + |\text{Voc}|}$$

Twenty News Groups (Joachims 1996)

- 各グループ1000の訓練文書
- 新規の文書を、もとのnewsgroupに割振る

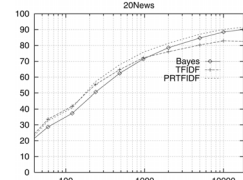
| | |
|-----------------------------------|--------------------|
| comp.graphics | misc.forsale |
| comp.os.ms-windows.misc | rec.autos |
| comp.sys.ibm.pc.hardware | rec.motorcycles |
| comp.sys.mac.hardware | rec.sport.baseball |
| comp.windows.x & rec.sport.hockey | rec.sport.hockey |
| alt.atheism | sci.space |
| soc.religion.christian | sci.crypt |
| talk.religion.misc | sci.electronics |
| talk.politics.mideast | sci.med |
| talk.politics.misc | |
| talk.politics.guns | |

T. Joachims. *A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization*. In Proceedings of the 14th International Conference on Machine Learning, Nashville, TN, 1997, pp.143-151.

Twenty News Groups (Joachims 1996)

- Naive Bayes: 89% 分類正解率
 - 頻出単語上位100個 (the and of ...) は除去
 - このように文法機能を担う単語や、文書を類別するのに有効でない単語を stop words として除去するのが普通
 - 頻度が3回に満たない単語は除去
 - 残った単語は、約 38,500 語

ただし、この正解率は高すぎ。20 Newsgroupsの各投稿には、分類に非常に役立つ subject フィールドがある。今ではこれらは除去することになっているが、当時では、除去せずに、分類実験をした可能性がある。



精度対訓練データ数 (1/3はテスト用にとりおいた)

20 Newsgroups: Rでは?

- データが多すぎて、Rのパッケージに含まれる naive Bayes 分類器は使えない。
 - データ行列(さきほどのRプログラムでは、xy, tt といった行列)が巨大になる(行数が文書数(約2,000)、列数が単語数(約40,000))。
 - しかし、非零要素は非常に少ないので、スパース行列表示を用いなければならない。
 - それでもオーバーヘッドが大きい。
 - それなら自分でプログラムを書いてしまおう。
- なお、Weka にもスパース行列が表現できて、原理的には取り扱える。しかし、大きなメモリが必要で、しかも遅い。

20 Newsgroups: データ

- "20 Newsgroups" というサイトにあり
 - <http://people.csail.mit.edu/jrennie/20Newsgroups/>
- 前処理(単語の切り出し等)が終わって、単語の個数のデータに編集が終わったものを用いる。Matlab で使いやすい形になっている。
 - 20news-bydate-matlab.tgz
- このうち、train.data, train.label, test.data, test.label を用いる。
- プログラムは資料として web 頁に掲載しておきます。
- 結果のうち、confusion matrix を次頁に示します。
- 正解率は、約78.2%です。

```
> cm
      correct
predicted 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
1 237  3  3  0  0  0  0  1  0  4  2  0  2 10  3  7  2 12  7 47
2  0 299 33  8  8 42  9  1  1  1  0  5 16  7  8  2  0  1  1  3
3  0  7 208 15 10  8  4  0  0  0  1  0  1  0  1  0  0  0  0  0
4  0 12 58 306 38 10 49  2  0  1  0  1 28  3  0  0  0  0  0  0
5  0  7 11 21 275  2 21  0  0  1  0  2  8  0  0  1  1  0  0  0
6  1 21 30  2  3 306  1  1  0  2  0  1  3  0  2  0  0  1  0  1
7  0  1  0  4  4  1 227  5  1  3  1  1  1  1  1  0  0  2  0  0
8  0  3  2  6  4  0 32 356 25  3  1  0  9  3  0  0  2  2  1  0
9  0  1  2  0  1  2  5  4 353  1  0  0  2  0  1  0  1  1  0  1
10 0  0  2  0  1  1  0  2 245  4  0  0  2  0  0  1  1  0  0  0
11 1  0  1  1  0  0  1  0  0 16 381  0  0  0  1  0  1  0  0  0
12 1 16 17  5  5 10  3  1  1  2 1361 45  0  3  1  3  4  3  1
13 1  4  1 23 16  0 11  4  1  2  0  3 260  3  4  0  0  0  0  0
14 2  3  4  0  7  0  2  0  1  0  2  2 6324  4  1  1  0  3  3
15 3  6  4  1  2  3  3  2  0  0  1  0  3  333  0  2  0  7  5
16 43  4  5  0  0  1  3  0  1  3  2  2  6 16 5377  3  7  2  69
17 3  0  0  0  3  1  1  5  4  1  0  7  0  3  1  2 324  3  95 19
18 9  0  0  0  0  1  3  1  2  2  1  0  2  6  2  2 323  5  5
19 7  2  9  0  6  2  6  9  5  9  3  8  0 10 24  1 16 21 184  8
20 10  0  1  0  0  0  1  1  0  1  0  1  0  1  1  1  4  0  1  90
```


NewsWeeder (Lang 1995)

- 目標概念 “usenet articles that I find interesting” を学習する
- ユーザはネットニュースを読むときに、興味深さの点数をつける
- 点数のついた文書を訓練例とする
- 点数を自動的につけた文書のうち上位 10% に興味深い文書が含まれる割合は、ユーザが普通に読む文書集合に含まれる割合の 3~4倍高かった

Lang, K (1995). *NewsWeeder: Learning to Filter News*. Proceedings of the 12th International Conference on Machine Learning, 331-339, Lake Tahoe, CA.

まとめ: Bayes 学習

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- 学習アルゴリズムの俯瞰像:
 - ML: $P(D|h)$ の最大化
 - MAP: $P(h|D) \propto P(D|h)P(h)$ の最大化
 - Bayes 最適分類器: $P(c|D) = \int P(c|h)P(h|D) dh$
- Gaussian ノイズ下の回帰:
 - ⇔ 二乗誤差の最小化
- 二値事象の確率の学習
 - ⇔ cross-entropy の最小化
- Naive Bayes: 乱暴な仮説だが実用的
 - 例えば、文書分類