ベイジアンフィルタ

尾藤 正人
ウノウ株式会社

ベイズ推測

- 過去に起きた事象を元に未来を予測する
- 実は単なる条件付き確率
- 事前確率から事後確率を求める

確率のおさらい

P(A) – Aが起きる確率
P(A∩B) – AとBが起きる確率(同時確率)
P(A|B) – BでAが起きる確率(条件付き確率)

ベイズの定理

条件付き確率では次が成り立つ

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

この式を変形して得られるのがベイズの定理

$P(A|B) = P(B|A)P(A)/P(B)$

ベイズの定理

データDから仮説Hが成り立つ確率

$P(H|D) = P(D|H)P(H)/P(D)$

P(H) – 事前確率

P(D|H) – 尤度

P(H|D) – 事後確率

事前確率と尤度から事後確率を求めるのがベイズの定理

ベイズの定理

メールをhamとspamに分類する例
H – カテゴリham, S – カテゴリspam
M – メール

ベイズの定理より
P(H|M) = P(M|H)P(H)/P(M) – ham確率
P(S|M) = P(M|S)P(S)/P(M) – spam確率

確率が高い最もらしい方に分類される

ベイズの定理

$P(H|M) = P(M|H)P(H)/P(M)$
$P(S|M) = P(M|S)P(S)/P(M)$

P(M)は一定なので比較するだけなら考えなくてよい
P(M)を定数とみなして

$P(H|M) \propto P(M|H)P(H)$
$P(S|M) \propto P(S|H)P(S)$

ベイズの定理

$P(H|M) \propto P(M|H)P(H)$
$P(S|M) \propto P(M|S)P(S)$

事前確率 - P(H),P(S)
単なる統計データ

尤度 - P(M|H),P(M|S)
各カテゴリでメールMが生起される確率
計算できないので近似する
近似の違いがアルゴリズムの違い

Paul Graham方式

w – メールMに含まれるトークン
P(w) – wのspam確率
b – wがspamに登場する回数
g – wがhamに登場する回数
nbad – wが含まれるspamメールの総数
ngood – wが含まれるhamメールの総数

$$P(w) = \frac{b/nbad}{2g/ngood + b/nbad}$$

P(w) = 0.99 (ngood = 0)

P(w) = 0.01 (nbad = 0)

Paul Graham方式

$$P(w) = \frac{b/nbad}{2g/ngood + b/nbad}$$

P(w) = 0.99 (ngood = 0)

P(w) = 0.01 (nbad = 0)

2*gとしているのはバイアスをかけるため
P(w)は b+2*g >= 5 の時のみ計算
wがspamにのみ含まれる場合 - P(w) = 0.99
wがhamにのみ含まれる場合 - P(w) = 0.01

Paul Graham方式

各トークンのspam確率から結合確率を求める
0.5から最も離れている15のP(w)を抽出

$$P(M)=\frac{\prod P(w)}{\prod P(w)+\prod (1-P(w))}$$

この時
spam – P(M) >= 0.9
ham – P(M) < 0.9

Gary Robinson方式

w – メールMに含まれるトークン
f(w) – wのspam確率
P(w) – Paul Graham方式で求める
n – wが含まれるメールの総数
x – 任意の値 or P(w)の平均値
s – 任意の値(参考値:1)

$$f(w)=\frac{sx+nP(w)}{s+n}$$

Gary Robinson方式

$$P = 1 - ((1 - f(w_1))(1 - f(w_2)) \cdots (1 - f(w_n)))^{1/n}$$

$$Q = 1 - f(w_1) f(w_2) \cdots f(w_n)^{1/n}$$

$$S = (P - Q)/(P + Q)$$

$$S_2 = (1 + S)/2$$

PはspamらしさQはhamらしさを表す
Sがspamかどうかを表す、ただし -1 <= S <= 1
S2は0 <= S2 <= 1 にしただけのもの

Naive Bayes

文書Xをカテゴリに分類される確率を考える

ベイズの定理より
$P(C|X) \propto P(X|C)P(C)$

P(C)は容易に計算可能

P(X|C)は計算できないので近似する

Naive Bayes

P(X|C)を各トークンの生起確率で近似する
w – 文書Xに含まれるトークン

$$P(X|C) \approx \prod_{w \in X} P(w|C) \prod_{w \notin X} (1 - P(w|C))$$

$$P(w|C) = \frac{w\text{を含むカテゴリ}C\text{の文書数}}{\text{カテゴリ}C\text{の文書数}}$$

Naive Bayes

各カテゴリにおけるP(C|X)を計算する
最も確率の高いカテゴリCに分類される

トークンの抽出

- 意味のある単語を取り出す
  - 空白区切りの言語の場合は処理が楽
  - 日本語のような場合は形態素解析が必要
  - 多言語化が難しい
  - なんとなく精度が高そう
- bi-gram, tri-gram
  - 2,3文字ずつ機械的に取り出す
  - 言語に依存しないので多言語化が容易
  - データ量が多い