知的情報処理システム特論　第14回

二宮　崇

今日の講義の予定

- 生成モデルと識別モデル
- 確率的識別モデル
  - 最大エントロピーモデル(多クラスロジスティック回帰、対数線形モデル)
  - パラメータ推定
  - 自然言語処理での素性ベクトル
- 教科書
  - Yusuke Miyao (2006) From Linguistic Theory to Syntactic Analysis: Corpus-Oriented Grammar Development and Feature Forest Model, Ph.D Thesis, University of Tokyo
  - Jun'ichi Kazama (2004) Improving Maximum Entropy Natural Language Processing by Uncertainty-aware Extensions and Unsupervised Learning, Ph.D. Thesis, University of Tokyo
  - 北研二(著) 辻井潤一(編) 言語と計算4 確率的言語モデル 東大出版会
  - Jorge Nocedal, Stephen Wright (1999) "Numerical Optimization" Springer, 1st edition 1999, 2nd edition 2006
  - Cristopher M. Bishop "PATTERN RECOGNITION AND MACHINE LEARNING" Springer, 2006
  - Nello Cristianini and John Shawe-Taylor "An Introduction to Support Vector Machines and other kernel-based learning methods", Cambridge University Press, 2000.

Generative Model and Discriminative Model

生成モデルと識別モデル

生成モデルから識別モデルへ

- 生成モデル
  - HMM
    - 状態遷移による生成
    - 状態遷移の確率は一つ前の状態のみに依存
    - 任意の文sと状態列qに対する同時確率p(s,q)を計算できる

生成モデルの改良の先に

- 条件部を詳細化
  - HMM: $p(q_j | q_{j-1}, q_{j-2})$
- 素性（特徴）という考え方

  条件部をどんどんリッチにして線形補間をとればよいのでは？→HMMの状態xの確率
  $p(x)=p(x | x$周辺の状況$)$

生成モデルの問題点

- 生成モデルの問題点
  - 独立性の仮定
    - HMMは一つ前の状態にのみ依存する(マルコフ性)
    - 二つ前の状態と今の状態は何らかの関係があるかもしれないけど、独立した事象として扱われている。

生成モデルの問題点

- 生成モデル

$$\tilde{t} = \arg\max_{t} p(t \mid s;\theta) = \arg\max_{t} p(s,t;\theta)$$

  - しかし、

$$p(s,t;\theta) = p(t \mid s;\theta)\, p(s;\theta)$$

  であるから$p\,(s;\theta)$ が計算できてしまう分、冗長なモデルになっている。間接的に分類問題を解いている。

  - 独立性を仮定した事象に分解しないと同時確率を計算することができない

識別モデル

- 識別モデル

  直接

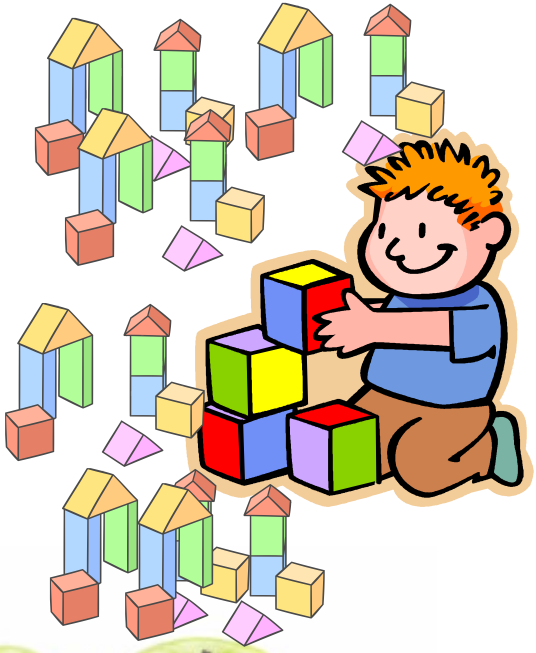
$$\tilde{t} = \arg\max_{t} p(t \mid s; \theta)$$

  を解く

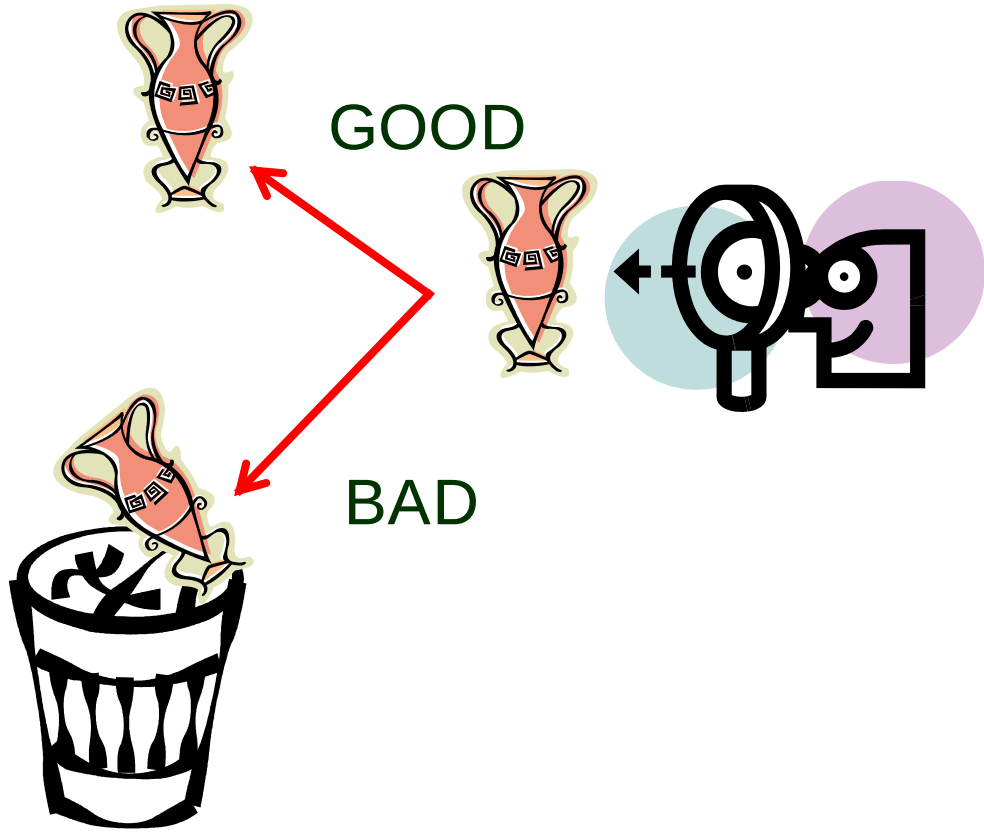
  - 独立な事象を仮定しない
  - 「条件部の確率」をモデルにいれない

生成モデルと識別モデル (イメージ)

生成モデル



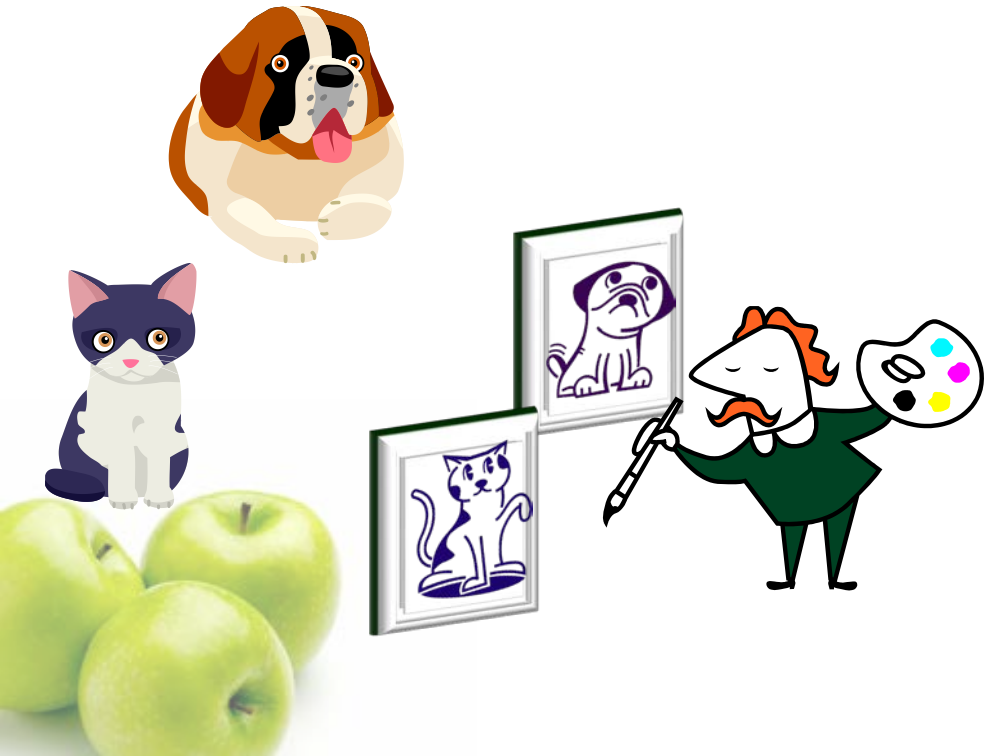
識別モデル



生成モデルと識別モデル（イメージ２）

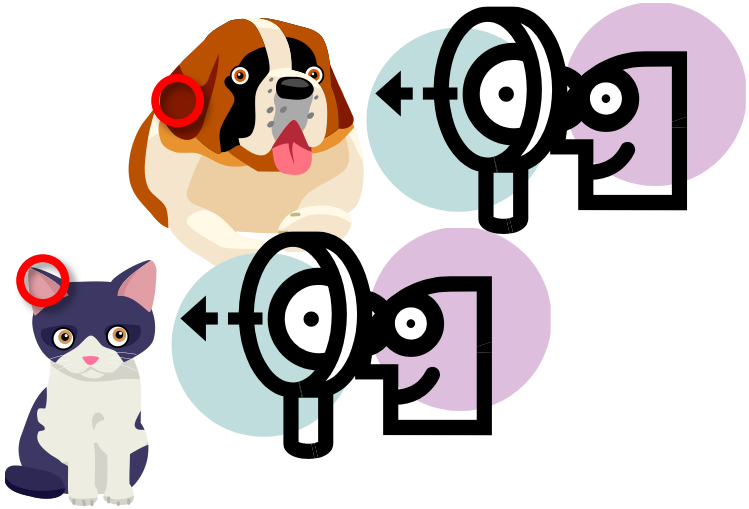
生成モデル

- 絵を描いて全体像の比較



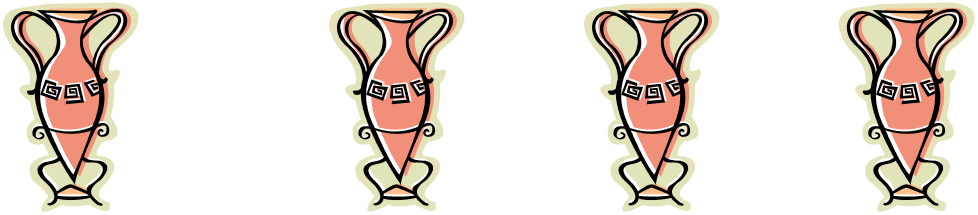
識別モデル

- それぞれの特徴を比較
  - 鼻の位置
  - 耳の形
  - 体の大きさ
  - 舌の表面

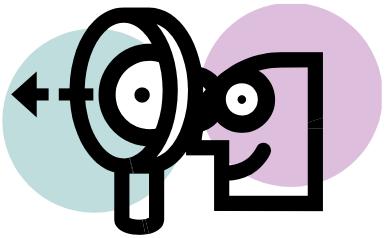


識別するための訓練

- 教師付学習
  - 良い例と悪い例を与えて、どこに注目すればより良く識別出来るのか学習



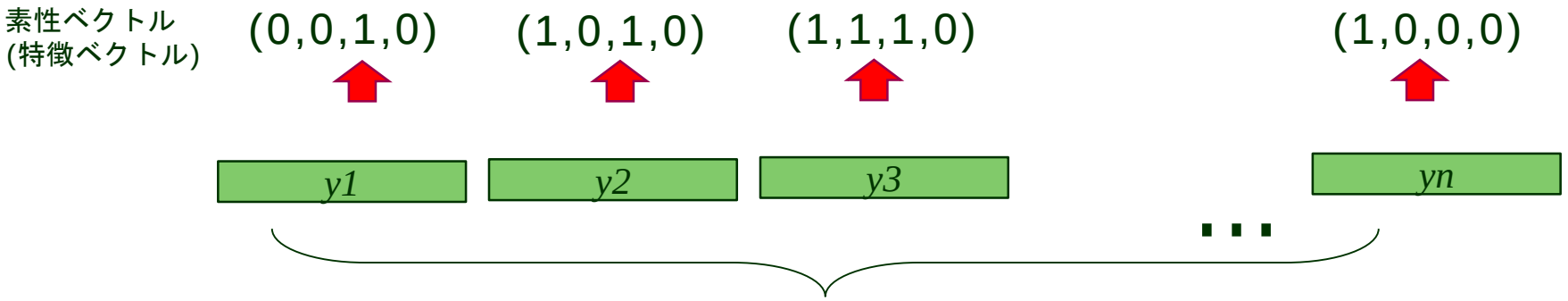
good examples



bad examples

識別モデル

$p(y \mid x)$ $x =$ “A blue eye girl with white hair and skin walked”

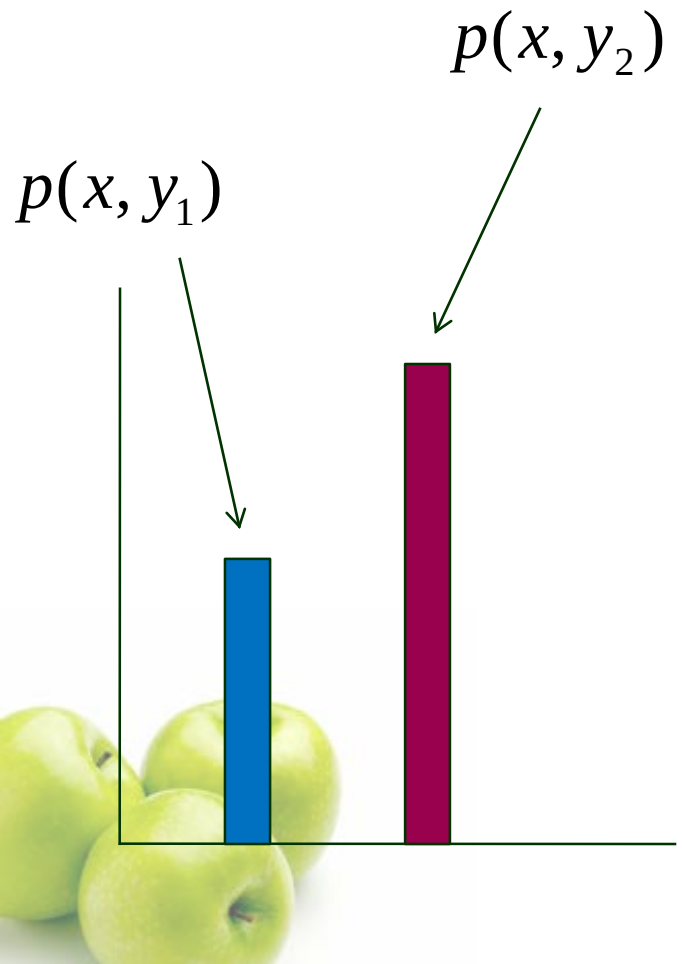


Xに対する全ての可能な品詞列集合

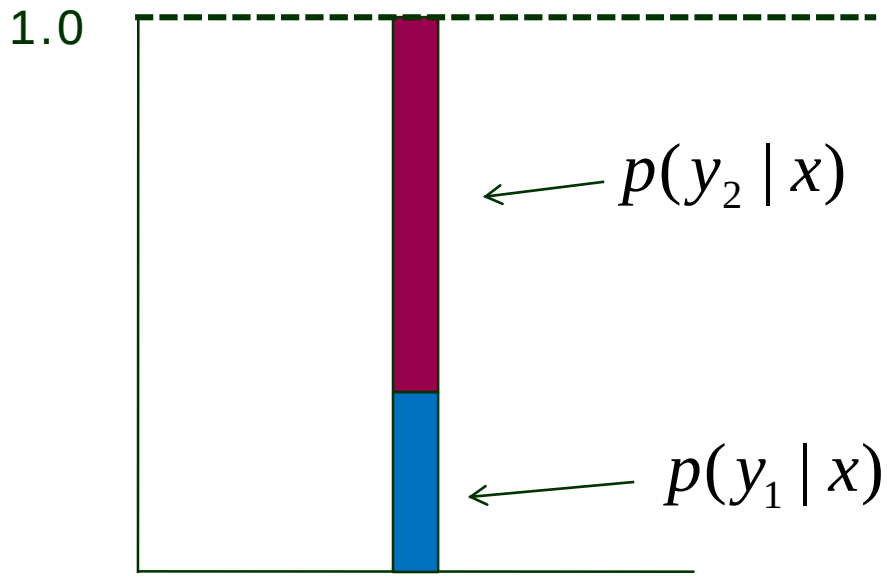
$p(y3|x)$ はy1,y2,y3,..,ynからy3を選択する確率

生成モデルと識別モデル

生成モデル



識別モデル



どちらが優秀なのか？

Tom Minka (2005) Discriminative models, not discriminative training, MSR-TR-2005-144, 1 p.

xを入力(文)、yを出力(構文木)としたときのパラメータ推定

- 生成モデル

$$p(x, y; \theta) = p(y \mid x; \theta)\, p(x; \theta)$$

- 識別モデル

$$p(y \mid x; \theta)$$

$\theta = \theta'$

- 一般モデル

$$p(x, y; \theta, \theta') = p(y \mid x; \theta)\, p(x; \theta')$$

最大エントロピーモデル、多クラスロジスティック回帰、対数線形モデル

確率的識別モデル

問題設定

- x: 入力
- y: 出力
- 訓練データ
  - (x_i, y_i) i=1,...,N
  - 例
    - xは文で、yはxに対する正解の構文木
    - xは競馬情報で、yは1位の馬
- 問題
  - ある未知の入力xに対する出力yの予測

素性関数

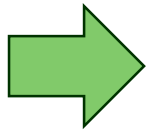
- 入力や出力から特徴を抽出する素性関数 (feature function) を複数定義
  - $f_j(x, y)$ j=1,...,M
  - 注意
    - 人手で定義
    - Mは特にいくつでもかまわないが、増やした分だけ計算時間・空間がかかったり、overfittingしてしまう
    - 良い素性関数をできるだけたくさん見つける、ということが人間がしなくてはいけない重要な仕事
- 素性ベクトル (または特徴ベクトル, feature vector)
  - $(f_1(x,y), f_2(x,y), ..., f_M(x, y))$

全体の流れ(1/2)

- Estimation (推定、パラメータ推定)
  - 各素性 f_j に対する重み λ_j を学習

訓練データ

入力	出力
x_1	y_1
x_2	y_2
…	…
x_N	y_N



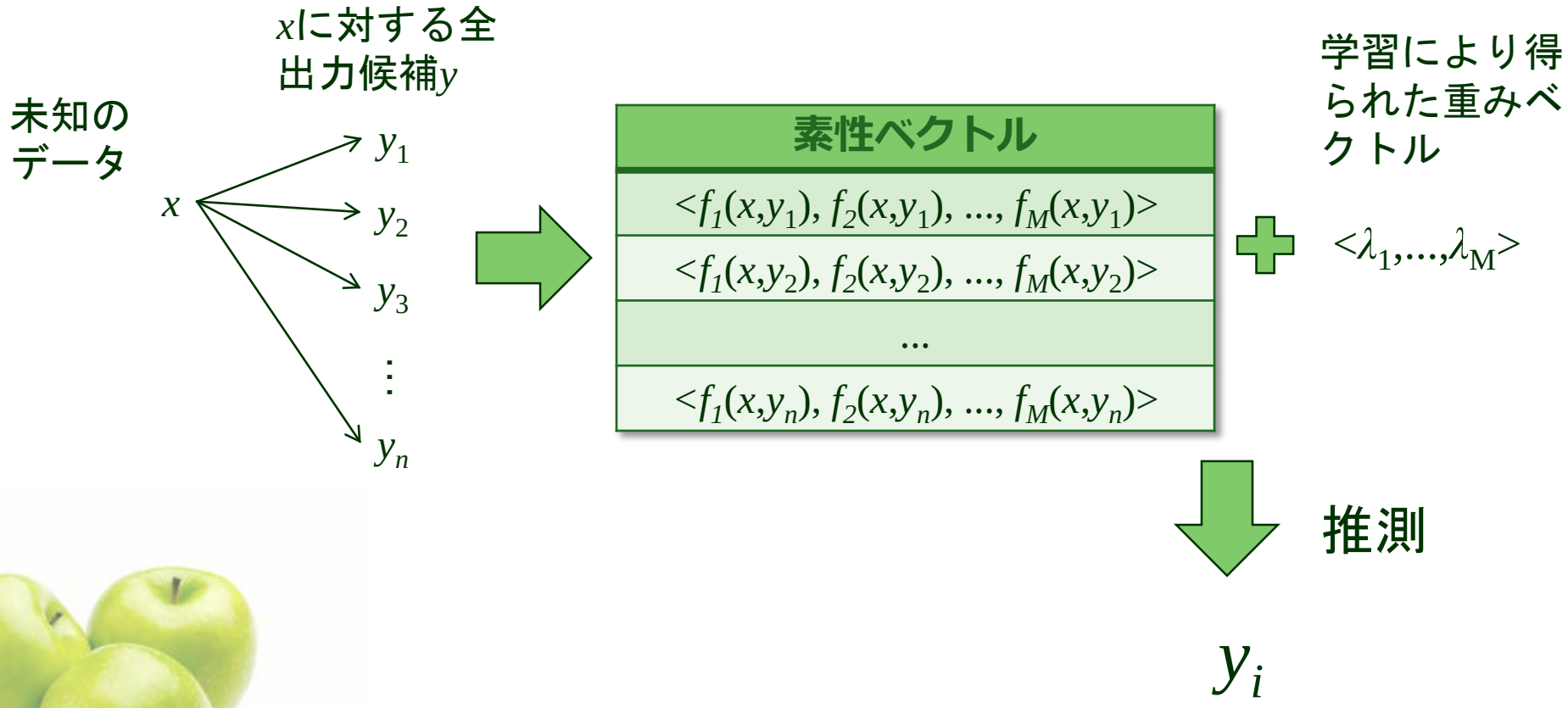
素性ベクトル
$<f_1(x_1,y_1), f_2(x_1,y_1), \ldots, f_M(x_1,y_1)>$
$<f_1(x_2,y_2), f_2(x_2,y_2), \ldots, f_M(x_2,y_2)>$
…
$<f_1(x_N,y_N), f_2(x_N,y_N), \ldots, f_M(x_N,y_N)>$

学習

$<\lambda_1, \lambda_2, \ldots, \lambda_M>$

全体の流れ(2/2)

- Inference (推測、推定)
  - 未知のデータxに対する出力yの推定



最大エントロピーモデル (Maximum Entropy model)
多クラスロジスティック回帰 (Multi-class Logistic Regression)
対数線形モデル (Log-linear Model)

- 確率モデル

重み

素性関数

$$p(y \mid x;\lambda) = \frac{1}{Z(x,\lambda)} \exp\left(\sum_j \lambda_j f_j(x,y)\right)$$

ただし

分配関数
(Partition function)

$$Z(x,\lambda) = \sum_{y' \in Y(x)} \exp\left(\sum_j \lambda_j f_j(x,y')\right)$$

直感的理解

- スコアの対数=各素性の(値×重み)の和
- p(y | x)= (xyのスコア)/(xに対する候補集合y'のスコアの和)

$$s(x,y,\lambda)=\exp\left(\sum_j \lambda_j f_j(x,y)\right)$$ とおくと、

$$p(y\mid x;\lambda)=\frac{1}{Z(x,\lambda)}\exp\left(\sum_j \lambda_j f_j(x,y)\right)=\frac{s(x,y,\lambda)}{\sum_{y'\in Y(x)} s(x,y',\lambda)}$$

となる。ちなみに、

$$s(x,y,\lambda)=e^{\lambda_1 f_1(x,y)+\lambda_2 f_2(x,y)+\cdots+\lambda_M f_M(x,y)}=e^{\lambda_1 f_1(x,y)}e^{\lambda_2 f_2(x,y)}\cdots e^{\lambda_M f_M(x,y)}$$

$$\log s(x,y,\lambda)=\sum_j \lambda_j f_j(x,y)$$

パラメータ推定

パラメータ推定

- 訓練データに対する対数尤度

$$\log\left(\prod_{i=1}^{N} p(y_i \mid x_i;\lambda)\right)$$

$$= \sum_{i=1}^{N} \log p(y_i \mid x_i;\lambda)$$

$$= \sum_{i=1}^{N} \log \frac{1}{Z(x_i,\lambda)} \exp\left(\sum_j \lambda_j f_j(x_i,y_i)\right)$$

$$= -\sum_{i=1}^{N} \log Z(x_i,\lambda) + \sum_{i=1}^{N}\sum_j \lambda_j f_j(x_i,y_i)$$

$$\log ab = \log a + \log b$$

$$\log_e \exp(x) = \log_e e^x = x$$

Zはパラメータを含むexpの足し算になっているから、これの極値を求めるのは難しい…

パラメータ推定

- GIS (Generalized Iterative Scaling)
- IIS (Improved Iterative Scaling)
- 勾配に基づく数値最適化
  - 最急降下法 (steepest decent method)
  - 共役勾配法 (Conjugate Gradient, CG; Fletcher & Reeves 1964)
  - BFGS (L-BFGS) (Nocedal 1980)
  - 自然言語処理では、経験的に勾配ベースのアルゴリズムの方がIISより非常に速く収束するため、勾配ベースのアルゴリズムが望ましい (Malouf 2002)
- オンラインアルゴリズム
  - パーセプトロンなど

パラメータ推定: 勾配ベースのアルゴリズム

- 目的関数

$$L(\lambda) = \log\left(\prod_{i=1}^{N} p(y_i \mid x_i; \lambda)\right) = -\sum_{i=1}^{N} \log Z(x_i, \lambda) + \sum_{i=1}^{N} \sum_{j} \lambda_j f_j(x_i, y_i)$$

- 勾配

$$\begin{aligned}
\frac{\partial L(\lambda)}{\partial \lambda_j} &= \sum_{i=1}^{N} f_j(x_i, y_i) - \sum_{i=1}^{N} \frac{1}{Z(x_i, \lambda)} \frac{\partial Z(x_i, \lambda)}{\partial \lambda_j} \\
&= \sum_{i=1}^{N} f_j(x_i, y_i) - \sum_{i=1}^{N} \frac{1}{Z(x_i, \lambda)} \sum_{y \in Y(x_i)} f_j(x_i, y) \exp\left(\sum_{j} \lambda_j f_j(x_i, y)\right) \\
&= \sum_{i=1}^{N} f_j(x_i, y_i) - \sum_{i=1}^{N} \sum_{y \in Y(x_i)} p(y \mid x_i; \lambda) f_j(x_i, y)
\end{aligned}$$

$$\mathbf{g} = \nabla L(\lambda) = \left\langle \frac{\partial L(\lambda)}{\partial \lambda_1}, \cdots, \frac{\partial L(\lambda)}{\partial \lambda_n} \right\rangle$$

パラメータ推定: 最急降下法

- パラメータ更新式

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \alpha^{(k)}\mathbf{g}^{(k)}$$

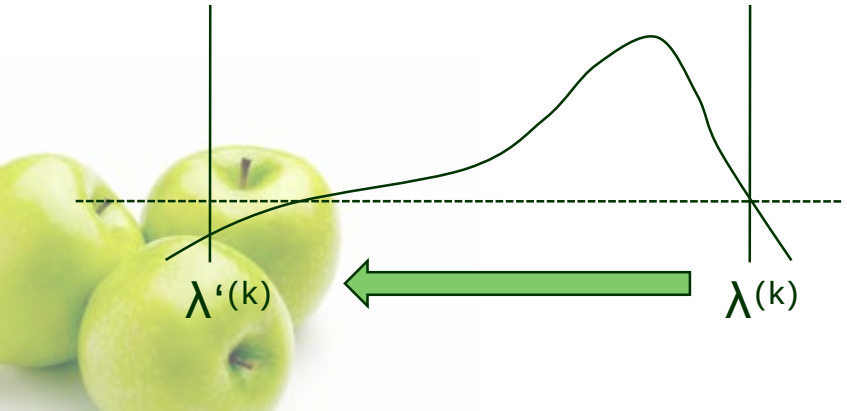
- αは適当な小さな値もしくは一次元最適化(直線探索 ともいう) (one-dimensional line search) で決定
- 収束が非常に遅い

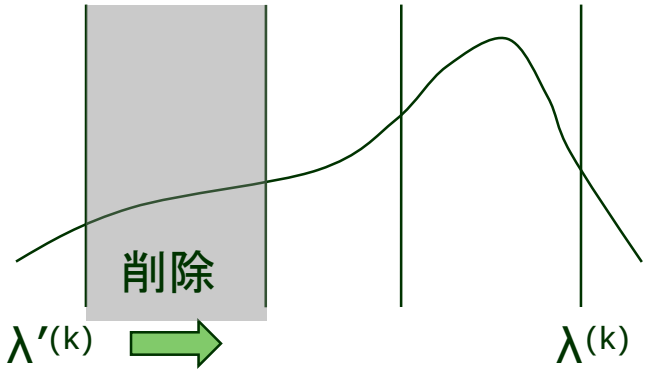
黄金分割にすると、L(λ)の計算が2回ではなくて1回で済む

一次元最適化

1. 候補領域の決定
あるステップ幅を**g**方向に2乗しながら探索し、L(λ′)<L(λ)になったところで候補領域の決定

2. 候補領域を3分割(黄金分割)し、2つの中間点のL(λ)を計算し、その大小を比較することにより、左か右の領域を候補領域から削除。2.を繰り返す。





パラメータ推定: 共役勾配法 Conjugate Gradient (CG)

- 更新式

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}$$

$$\mathbf{d}^{(k)} = -\mathbf{g}^{(k)} + \beta_{FR}\mathbf{d}^{(k-1)}$$

$$\beta_{FR} = \frac{\mathbf{g}^{(k)}\mathbf{g}^{(k)}}{\mathbf{g}^{(k-1)}\mathbf{g}^{(k-1)}}$$

- αは1次元最適化(one-dimensional line search)で求める
- 毎回、直交する方向に探索している
- n次元なら、n回の繰り返しで終了

パラメータ推定: 準ニュートン法

- 多次元のニュートン法
  - ヘシアンの逆行列の計算が重い…
- 準ニュートン法
  - ヘシアン逆行列を近似する
  - BFGS (Broyden 1970, Fletcher 1970, Goldfarb 1970, Shanno 1970)が有名。ただし、$|\lambda|^2$のサイズの行列を扱うので、巨大な次元の素性ベクトルには向かない
  - Limited-memory BFGS (L-BFGS) (Nocedal 1980)は少ないメモリでヘシアン逆行列を近似する。最大エントロピー法ではよく使われる。

パーセプトロン (Perceptron)

- 最大エントロピー法の問題点
  - Z(正解候補集合のスコアの和)の計算が重い
- パーセプトロン
  - 訓練データx_iに対しy_iを出力する確率が、正解候補集合$Y(x_i)$のどの要素の確率よりも高ければ良い

$$\log p(y_i \mid x_i;\lambda) - \log p(y' \mid x_i;\lambda) \text{を大きく} \left(y' = \arg\max_{y \in Y(x_i)} p(y \mid x_i;\lambda)\right)$$

$$\Leftrightarrow \sum_j \lambda_j f_j(x_i, y_i) - \sum_j \lambda_j f_j(x_i, y') \text{を大きく}$$

- 訓練データの正解と現在のパラメータで推測される最も確率の高い答えとだけ比較
- 実装もアルゴリズムも簡単！
- 最大エントロピーより性能は落ちるけど、メモリー使用量や学習時間の点で非常に有利

パーセプトロン: アルゴリズム

Input: training data $D=\langle x_1,y_1\rangle \ldots \langle x_N,y_N\rangle$, feature functions $f=\{f_j\}$, initial parameters $\lambda=\{\lambda_j\}$

Output: optimal parameters λ

loop until λ converges

  foreach $\langle x,y\rangle \in D$

    $z' := \text{argmax}_z\ p(z|x;\lambda)$

    if($y \neq z'$)

      foreach $f_j \in f$

        $\lambda_j := \lambda_j + f_j(x, y) - f_j(x, z')$

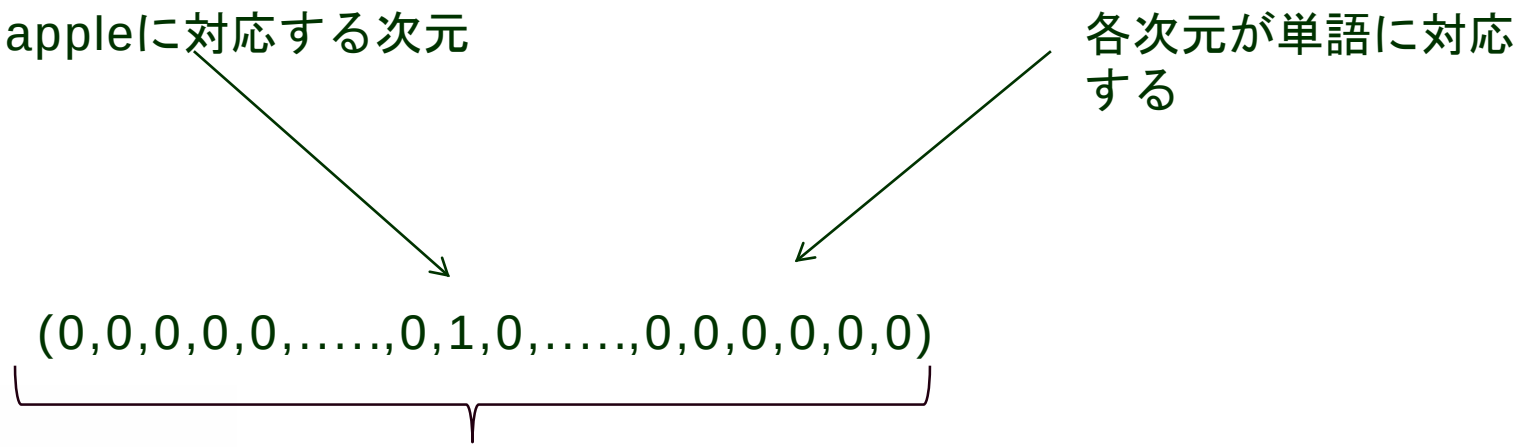
自然言語処理の識別モデル

識別モデルのいいところ

- 独立性を仮定していない
  - (戦略として) 思いつく限りいろんな素性をいれる
  - 訓練データに対してより良い予測ができる
  - 逆にoverfittingする可能性がある
    - c.f. 正規分布の事前分布によるMAP推定でoverfittingを緩和
- 自然言語処理の場合、疎なベクトルになる
- 疎なベクトルなら数百万次元ぐらい扱える

素性に関する注意その1

- 単語の素性と素性値
  - 例: 今みている単語が``apple''であった時の素性値



(訓練データに出現した)単語の数だけ次元がある！

素性に関する注意その2

- 素性の組み合わせ
  - 最大エントロピー法(ロジスティック回帰)では、素性同士の共起情報が別素性として自動的に組み込まれるわけではない
    - 一つ前の単語が”the”で、今見ている単語が”flies”。一つ前の品詞が動詞で、今見ている品詞が冠詞などなど。
    - SVM: 多項式カーネル
  - 素性の組み合わせを手で指示する。自動的に組み合わせる研究もある。c.f. 素性選択

まとめ

- 識別モデル
  - 最大エントロピーモデル(多クラスロジスティック回帰、対数線形モデル)
- 識別モデルのパラメータ推定
  - 勾配ベースのアルゴリズム
  - パーセプトロン
- 自然言語処理での素性ベクトル

今までの講義資料
http://aiweb.cs.ehime-u.ac.jp/~ninomiya/iips/