

情報意味論(9)

(簡単に)事例ベースアプローチ

櫻井彰人

慶應義塾大学理工学部

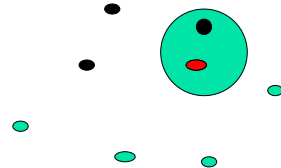
事例ベース学習

- キーアイデア
 - 訓練データ $\langle x_i, f(x_i) \rangle$ を全て憶えていよう(とりあえずは、何も、または、あまりしない)
 - 問い合わせがあったら、その時点で、しよう
- この類に属する方法
 - 最近傍法 (Nearest neighbor)
 - k -Nearest neighbor
 - Locally weighted regression
 - Radial basis functions
- Lazy 対 eager

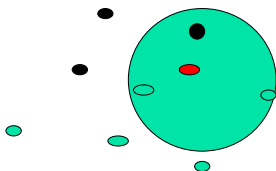
最近傍法

- 最近傍法 (Nearest neighbor)
 - 問合せ x_q に対し、最近接の x_n を見つけ、 $f(x_q) \leftarrow f(x_n)$ とする
- k -Nearest neighbor
 - k 個の最近接データの間で、多数決
 - k 個の最近接データの間で、平均値

1-Nearest Neighbor



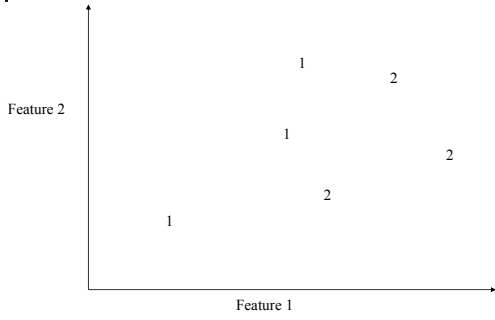
3-Nearest Neighbor



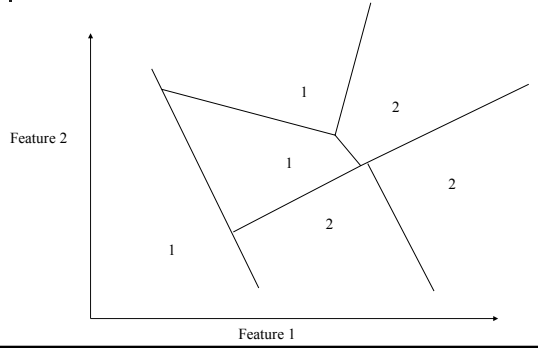
最近傍法の特徴

- いつ使うか
 - 属性が R^n の点とみなせる
 - 大体20個以下の属性
 - 大量の訓練データ
- 長所
 - 学習が速い
 - 複雑な目標関数も可能
 - (訓練データがもつ)情報を失うことがない
- 短所
 - 問合せ時、遅い
 - 無関係な属性によって、簡単に、ごまかされる

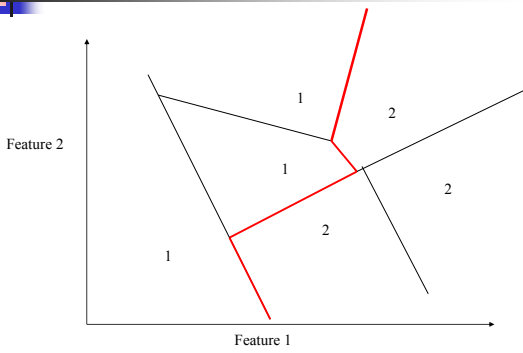
幾何的解釈



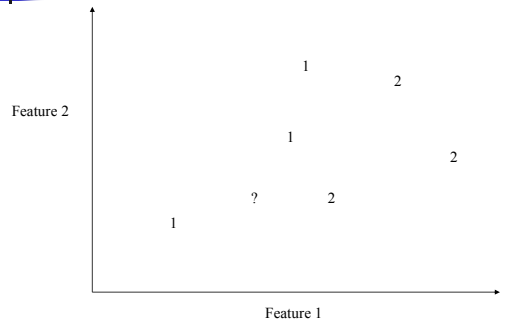
境界



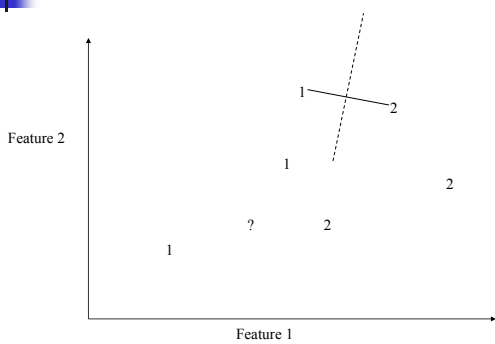
境界



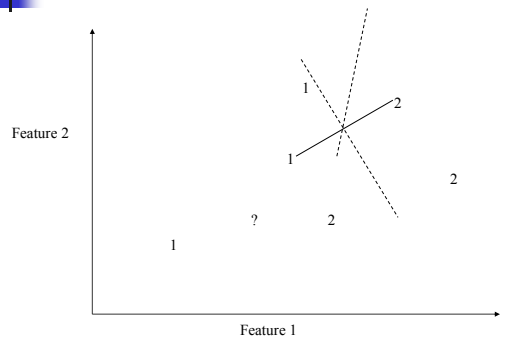
境界を描く



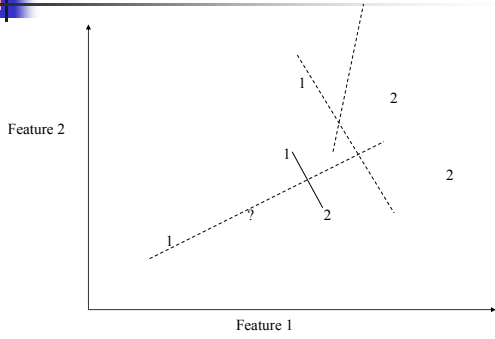
境界を描く



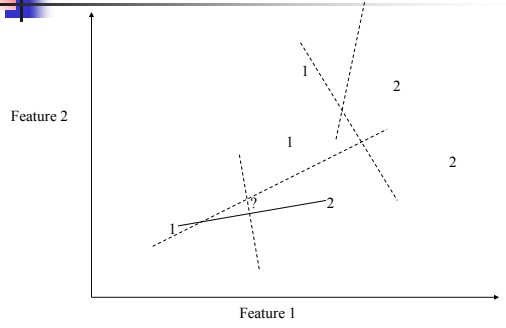
境界を描く



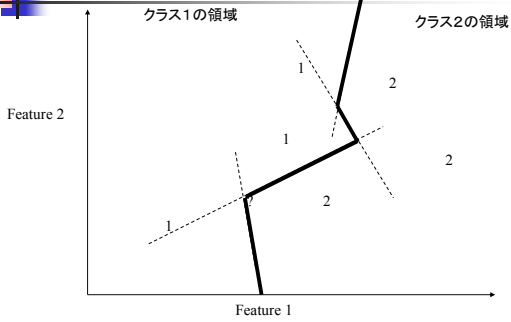
境界を描く



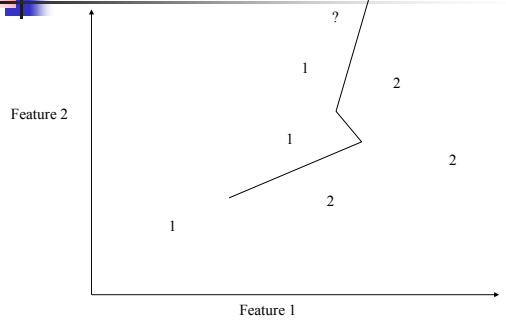
境界を描く



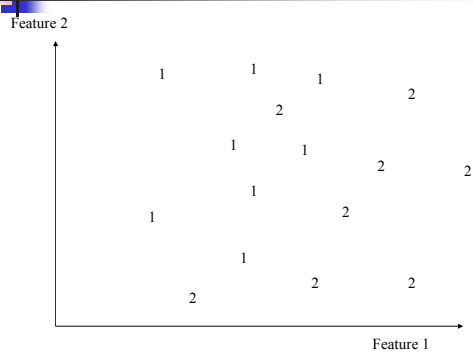
境界を描く



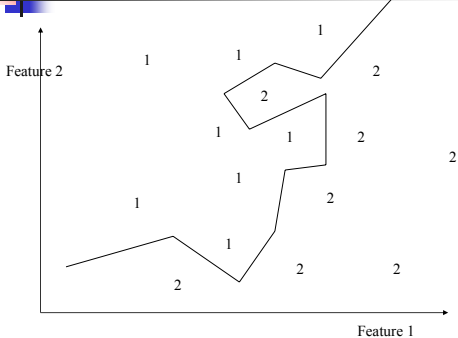
1-NNの幾何的解釈



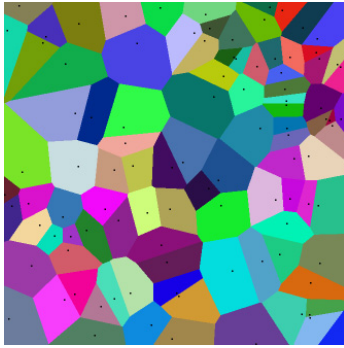
データ点が多いとき



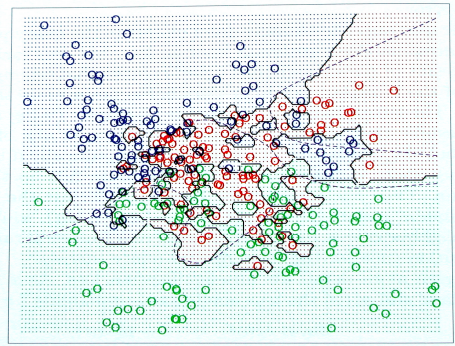
境界は複雑となる



Voronoi

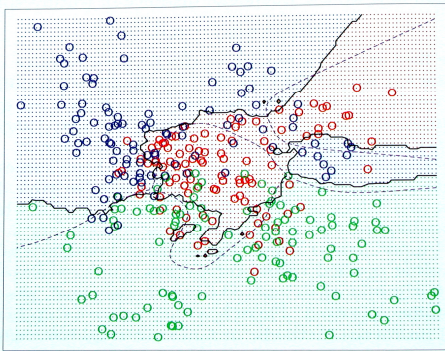


1-Nearest Neighbor



From Hastie, Tibshirani, Friedman 2001 p418

15-Nearest Neighbors



From Hastie, Tibshirani, Friedman 2001 p418

From Hastie, Tibshirani, Friedman 2001 p419

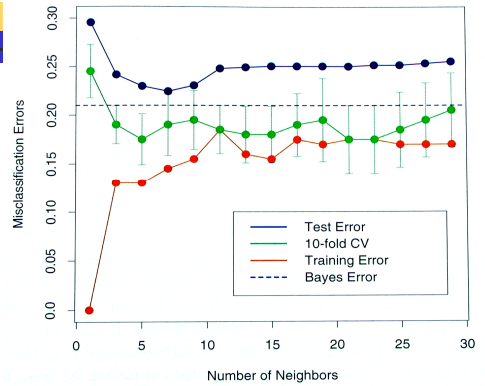


Table 6. Results summary of TC systems on Reuters versions 1–4.

System	Reuters version 1	Reuters version 2	Reuters version 3	Reuters version 4
WORD	—	.15 (Scut)	.31 (Pcut)	.29 (Pcut)
kNN	—	.69 (Scut)	.85 (Scut)	.82 (Scut)
LLSF	—	—	.85 (Scut)	.81 (Scut)
NNets.PARC (perceptron)	—	—	—	.82 (Pcut)
CLASS1 (perceptron)	—	—	.80	—
RIPPER (DNF)	—	.72 (Scut)	.80 (Scut)	—
SWAP-1 (DNF)	—	—	.79	—
DTree IND	—	.67 (Pcut)	—	—
DTree C4.5	—	—	.79 (F ₁)	—
CHARADE (DNF)	—	—	.78	—
EXPERTS (n-gram)	—	.75 (Scut)	.76 (Scut)	—
Rocchio	—	.66 (Scut)	.75 (Scut)	—
NaiveBayes	—	.65 (Pcut)	.71	—
CONSTRUE (Exp. Sys.)	.90	—	—	—

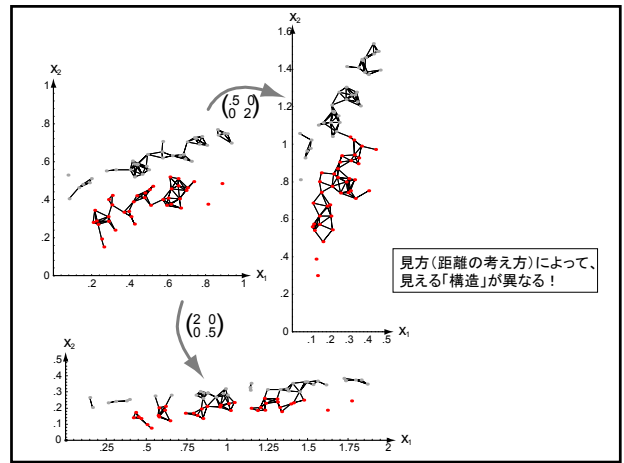
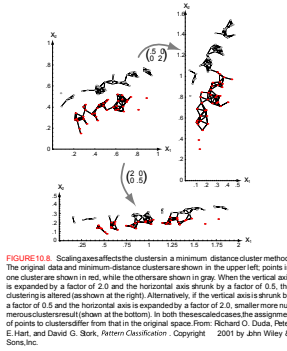
Yiming Yang, An Evaluation of Statistical Approaches to Text Categorization, Information Retrieval, vol.1, 69-90 (1999)

		#1	#2	#3	#4	#5
		# of documents	14,704	10,657	9,410	4,652
		# of training documents	4,746	3,480	3,002	3,299
		# of categories	135	93	92	90
System	Type	Results reported by	.750	.719	.690	.752
Word	microaveraging	[Buckley et al. 1998]				.752
	probabilistic	[Dochterman 1998]				.720
	probabilistic	[Lam et al. 1997]	.443 (M _{F1})			
NaiveBayes	Doc	[Lewis 1992]	.600			.747
	probabilistic	[Li and Yamashita 1999]				.773
	probabilistic	[Li and Yamashita 1999]				.773
kNN	Doc	[Yang and Liu 1999]				.884
	probabilistic	[Yang and Liu 1999]				.794
	decision trees	[Dochterman 1998]				
c4.5	Doc	[Dochterman 1998]				.884
	decision trees	[Dochterman 1998]				.794
	Doc	[Lewis and Singer's 1994]	.670	.805		
Rocchio	Doc	[Cohen and Singer 1999]	.683	.811		.820
	decision rules	[Cohen and Singer 1999]	.793	.759		.827
	Doc	[Li and Yamashita 1999]				.820
CHARADE	Doc	[Li and Yamashita 1999]				.738
	decision rules	[Mouliner and Grosz 1994]				.783 (F ₁)
	Doc	[Mouliner et al. 1994]				.835
Lear	Doc	[Yang 1995]				.810
	regression	[Yang and Liu 1999]	.747	.833		.849
	Doc	[Dochterman et al. 1997]				.822
Wapnet-Roc	Doc	[Lam and Ho 1996]				.822
	batch linear	[Cohen and Singer 1999]	.660	.718		.776
	Doc	[Dochterman et al. 1997]				.717
Rocchio	batch linear	[Dochterman 1998]				.711
	batch linear	[Lam and Ho 1996]				.759
	batch linear	[Li and Yamashita 1999]				.625
NNets	Doc	[Li and Yamashita 1999]				.802
	neural network	[Yang and Liu 1999]				.838
	neural network	[Yang and Liu 1999]				.838
CharW	Doc	[Li and Ho 1996]				.820
	Doc	[Dochterman 1998]				.838
	Doc	[Lam and Ho 1996]				.820
k-NN	Doc	[Yang and Liu 1999]	.690	.852	.820	.820
	Doc	[Yang and Liu 1999]				.668
	Doc	[Buckley et al. 1998]				.820
SWAP	Doc	[Yang and Liu 1999]				.876
	Doc	[Li and Yamashita 1999]				.841
	Doc	[Yang and Liu 1999]				.855
ANALOC	Doc	[Cohen and Singer 2000]				.878
	Doc	[Cohen and Singer 2000]				.878
	Doc	[Buckley et al. 1998]	.842 (M _{F1})			.860
CONSTRUE	Doc	[Lam et al. 1997]				.850
	Doc	[Lam et al. 1997]				.850

Table 6. Comparative results among different classifiers obtained on five different versions of the Reuters collection. Unless otherwise noted, entries indicate the microaveraged break point; within parentheses, "M" indicates macroaveraging and "F₁" indicates use of the F₁ measure. Boldface indicates the best performer on the collection.

Fabrizio Sebastiani, Machine learning in automated text categorization, ACM Computing Surveys, vol.34, no.1, 1-47 (2002)

距離の問題



次元の呪い

- 20個の属性で記述されるが、その内、たった2属性のみが意味ある場合を考える
- 次元の呪い:
 - k -NNなら、他の18属性の値でどんな結論も出うる
- 解決方法
 - j 番目の属性に z_j の荷重を。 z_j は予測誤差最小となるように選択
 - cross-validationを用いて自動的に z_j を決定

Locally weighted regression

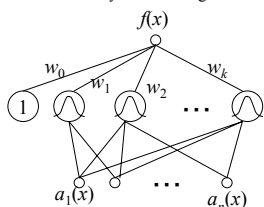
- k -NN は各問合せ x_q で f の局所近似を構成していた
- x_q の周囲で $f(x)$ の近似関数を明示的に構成したらどうだろうか?
 - k -NNに線型回帰したら?
 - 2次回帰では?
 - 区分回帰したら?
- 最小化すべき誤差にもいくつかの候補が

$$E_1(x_q) = \frac{1}{2} \sum_{x \in x_q, \text{ } \neq k\text{-NN}} (f(x) - \hat{f}(x_q))^2$$

$$E_2(x_q) = \frac{1}{2} \sum_{x \in D} (f(x) - \hat{f}(x_q))^2 K(d(x_q, x))$$

Radial Basis Function Network

- 局所近似の線型結合による大域近似
- 神経回路網の一種
- distance-weighted regression に類似
 - lazyではなくeagerであるが



$$f(x) = w_0 + \sum_{u=1}^k w_u K_u(d(x_u, x))$$

$K_u(d(x_u, x))$ の一例

$$K_u(d(x_u, x)) \equiv e^{-\frac{1}{2\sigma^2} d(x_u, x)^2}$$

RBFの学習

- $K_u(d(x_u, x))$ の x_u の定め方
 - 事例空間に一樣にばら撒く
 - 事例を使用(事例の分布が反映)
- 荷重の学習(K_u は正規分布とする)
 - 各 K_u の分散(と平均)を定める
 - 例えば、EMを使用
 - K_u を固定したまま、線型出力部分を学習
 - 線型回帰で高速に



Lazy 対 eager

- Lazy: 事例からの一般化をしないている。問合せがあったときに考える
 - k-Nearest Neighbor
- Eager: 問合せ前に予め一般化しておく
 - 「学習」アルゴリズム、ID3, 回帰, RBF,,,
- 違いはあるか？
 - Eager学習は全域的な近似を作成
 - Lazy学習は局所近似を大量に作成
 - 同じ仮説空間を使うなら、lazyの方が複雑な関数を作成
 - over-fittingの可能性
 - 柔軟(複雑なところと単純なところの組合せ)



まとめ

- 事例ベースアプローチ
 - 大域的な構造を仮定しない
 - どんな場合にも使える
 - 雑音に弱い(大域構造を用いた平滑化ができない)
 - 次元の呪い