データ科学2

第8回:分類学習(3)

担当:尾崎 知伸

ozaki.tomonobu@nihon-u.ac.jp

#### 欠損値の扱い

- あるデータの属性に欠損がある場合に、どのように学習し、予測するのか?
  - 実データを対象とすると、どうしても欠損のあるデータが出てきます。
  - もちろん排除すればよいのですが、排除せずにそのまま利用することも考えられます。

#### • 学習時:

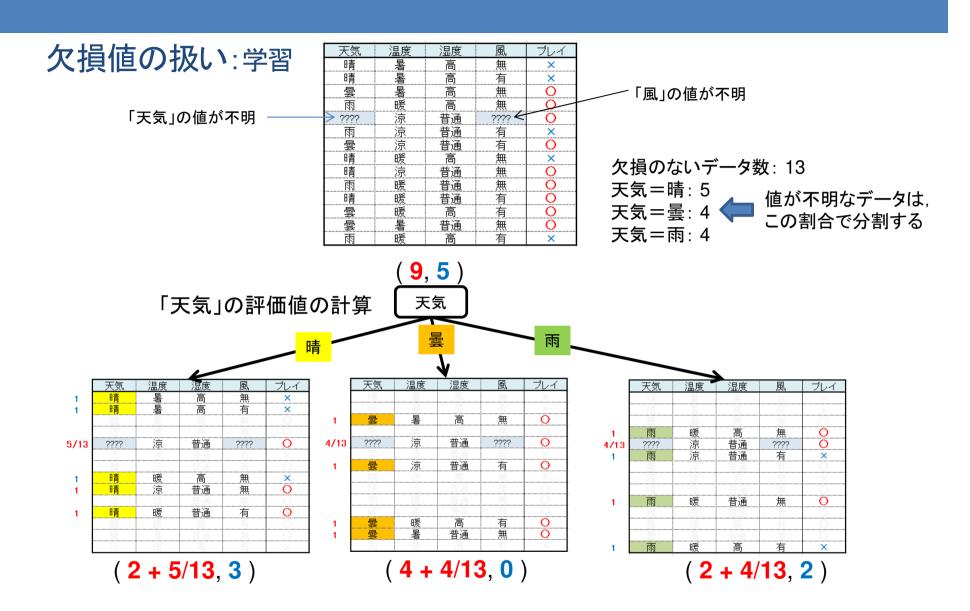
- 既知データの相対頻度に従い、欠損を持つ事例を確率的に分割
  - 分割されたそれぞれを、『重み(<sub>重要度</sub>)1未満の事例』と考える
- あくまで説明変数(属性)の欠損のみを考える、クラスの欠損は対象外

#### • 予測時:

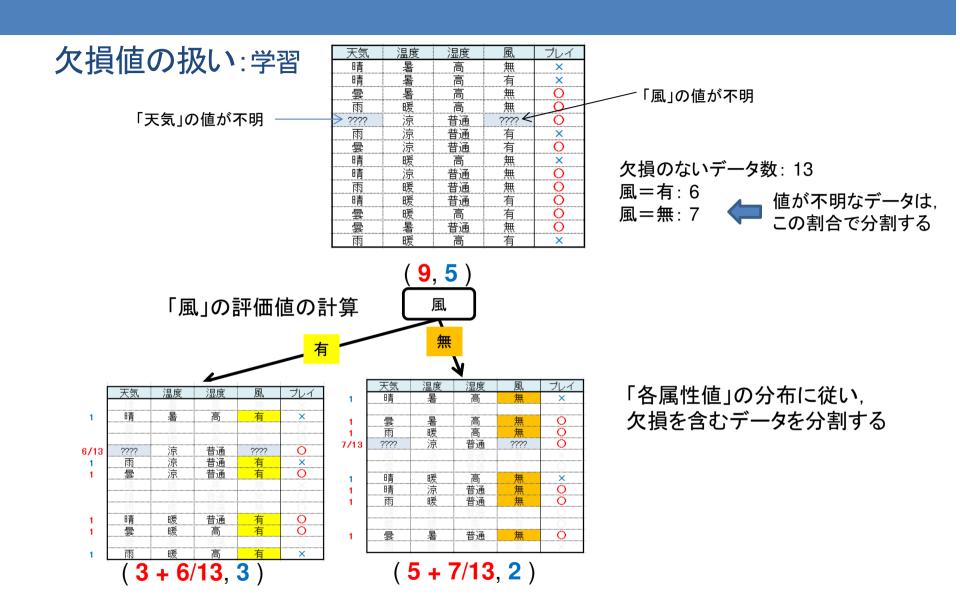
学習時と同様,欠損値の確率分布に従って,分類を進める.

ある事例において, 属性の値が不明

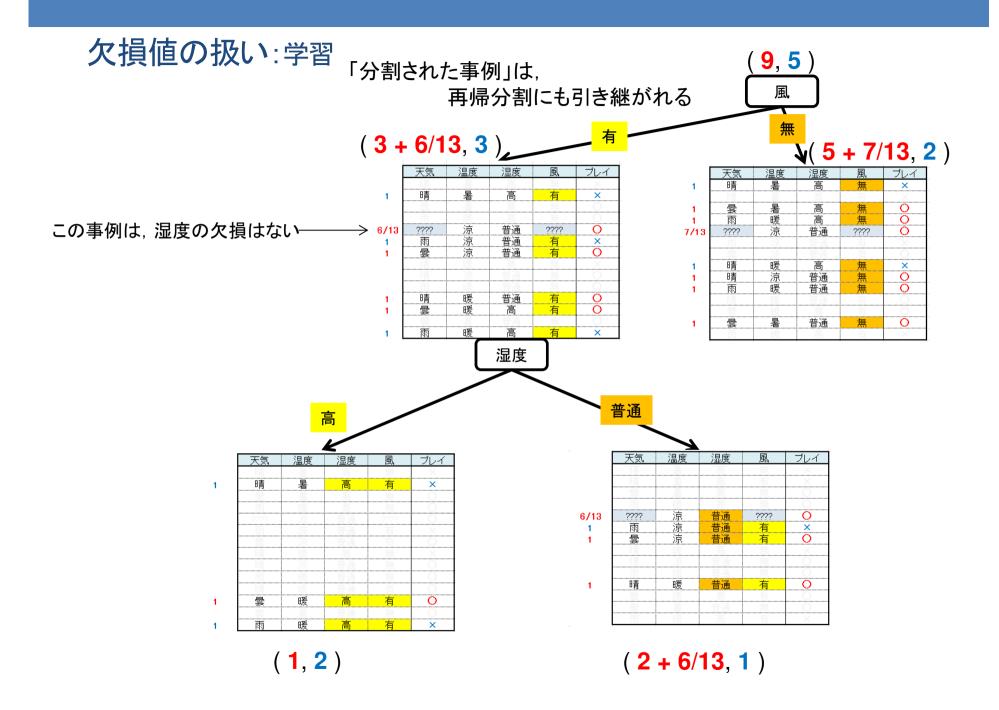
天気	温度	湿度	風	ブレイ
晴	暑	高	無	×
晴	暑	高	有	×
雲	暑	高	無	0
晴 晴 索 雨 ????? 雨 雲 晴	暑暑暖凉凉凉暖凉暖;暖;暖;暖;暖;	高 高 高 普通 普通 普通 音通	無 有 無 無	0
????	涼	普通	<i>→</i> ????	0
雨	涼	普通	有	×
雲	凉	普通	有	0
晴	暖	高	無	×
晴	涼	普通	無	0
雨	暖	普通	無	0
晴	暖	普通	有	0
雲		高	有	0
雨 晴 雲 雲 雨	暖 暑 暖	普通 普通 普通 高 普通 高	有 有 無 無 無 有 有 無	× 0 0 × 0 × 0 0
雨	暖	高	有	×

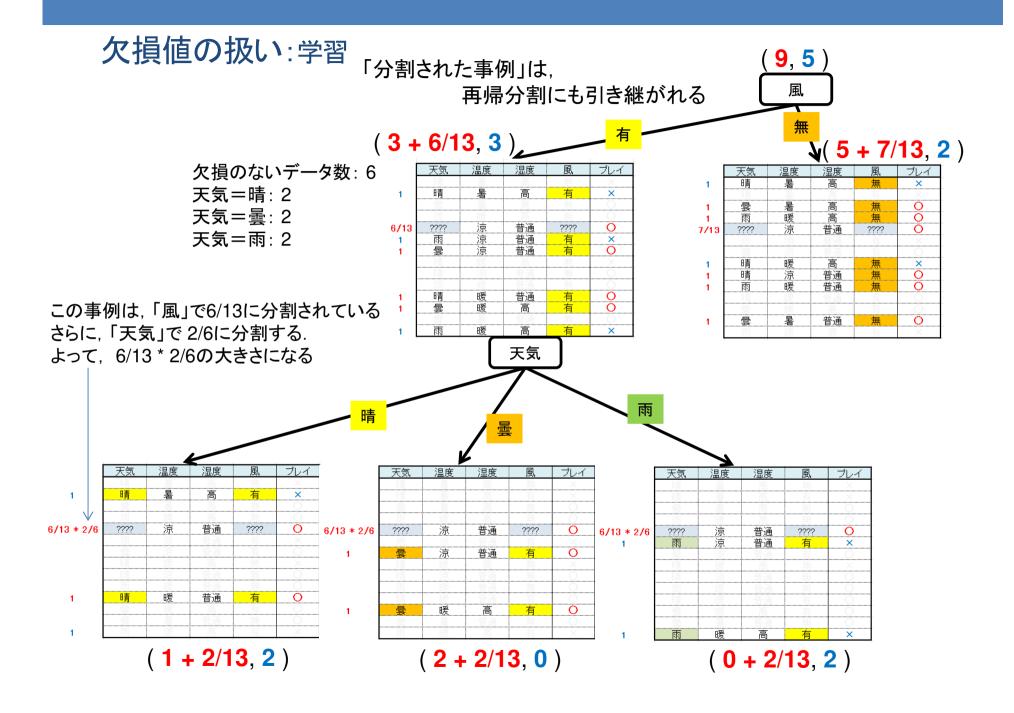


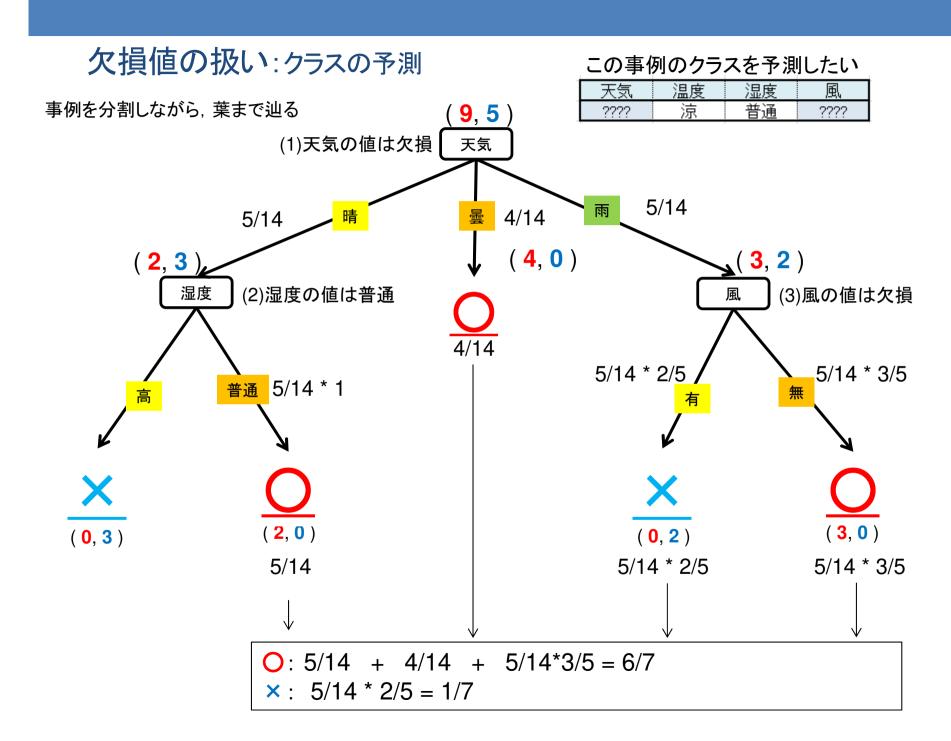
分割前の(9,5), 分割後の(2+5/13, 3), (4+4/13, 0), (2+4/13, 2)で評価値を計算



分割前の(9,5), 分割後の(3+6/13, 3), (5+7/13, 0)で評価値を計算







## 汎化能力と枝刈り

学習手法 蓄積データ

分類器・モデル

決定木の目的は、『クラス未知の事例のクラスを予測する』こと

新しいデータ

予測クラス

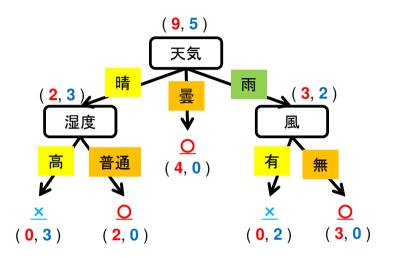
- 汎化(はんか)能力が重要. 過学習しては困る
- 汎化能力:未学習(未知)のデータに対して、正しいクラスを予測する能力
- 過学習:訓練データに過度に適合し、未知データに対する予測能力(汎化能力)が失われてい る状態
  - 学習データだけでない 枝分かれが多くなりすぎ→ノイズや例外値に由来しているものがあるかも。 テストデータに対する テストデータに対する 予測能力が重要
  - 細かすぎる条件は、ルールとしての妥当性が疑問
- 決定木における 汎化能力の向上 の方法
  - ・訓練データに過適合している木を「簡略化」する=枝刈り(Pruning)
  - (確率的に)誤差の範囲であると認められた場合には、細かい枝は払ってしまう。
  - 枝刈りによって、決定木が簡潔になり、しかも、分類精度は向上する。
- 事前枝刈り:途中で、木の成長をやめる
  - 該当する訓練例が少ない場合に木の成長を止める
    - 少ないデータに合わせると、過学習を起こす可能性が高い。
- 事後枝刈り:後処理として,作成した木を簡略化する
  - 一般に、事前枝刈りと比較し、事後枝刈りの方が、効果が大きい。
  - ・評価値同様、いくつかの事後枝刈り手法(基準)が提案されている
    - 誤り削減枝刈り、コスト複雑度枝刈り、悲観的枝刈り、など。

「事前」と「事後」 大きく2種類の枝刈りがある

### 前回資料

# 停止条件: どの様な場合に、分割する必要がない or 分割が出来ないのか?

- 1. すべての事例が、同一クラスに属するとき
  - 目的は、「未知の事例」のクラスの予測
  - 予測するクラスが一意に決まるので、分割の必要はない
- 2. 分割する属性がないとき
  - (離散属性では)既に分割テストで利用された 属性は、利用できない。
    - 利用できる属性は、徐々に減っていく
  - ・ 全属性を利用してしまった場合は、分割が出来ない
    - そのときの予測クラスは?→マジョリティクラス(多数決)



仮に属性が「天気」しかない場合

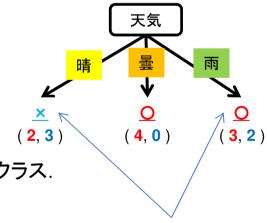
- ・ 3. 条件に当てはまる事例数が少ないとき
  - 目的は、「未知の事例」のクラスの予測 ∠
  - 訓練データを完璧に分類しても、予測が出来なければ無意味
    - ※同様の趣旨で、出来た決定木を小さく簡略化する (=事後枝刈り、Post-pruning)

事例数が少なくなったら、そこで打ち切る→予測クラスはマジョリティクラス.

- 4. 分割に対して、良い属性が見つからない。
  - 情報利得や情報利得比が小さい

→分割してもしなくても、結果がほとんど変わらない

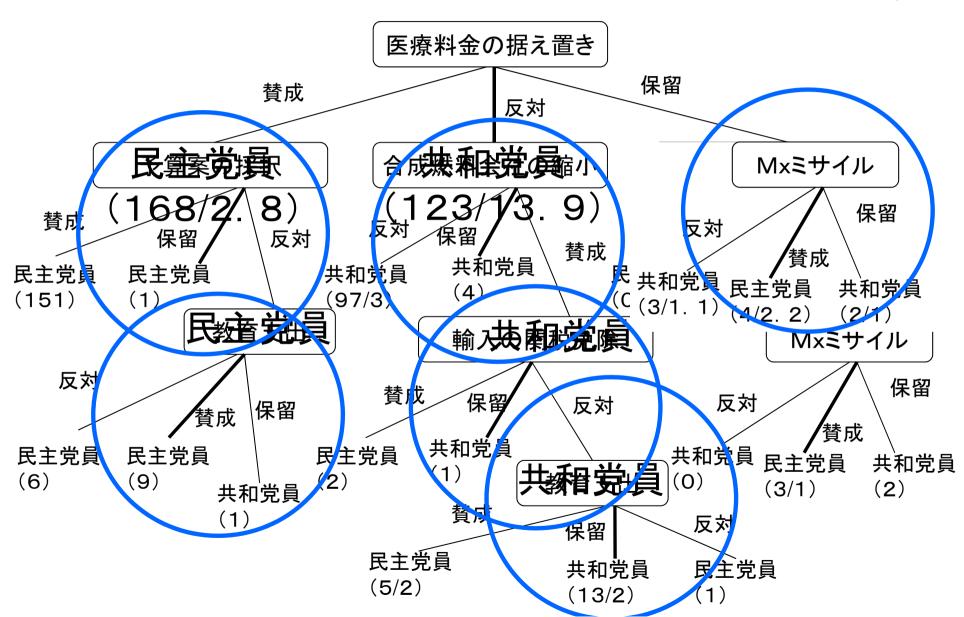
事前枝刈り(Pre-pruning)



多数決でクラスを決める

# 事後枝刈りのイメージ

データ: 1984年度第2四半期連邦会議から16の主要問題の投票を Jeff Schlimmerによって記録されたもの (California Univ.Irvine Data Library 所属)



# 誤り削減枝刈り(reduced-error pruning)

検証用のデータを準備する

決定木に対する 誤り削減に基づく枝刈りアルゴリズム

**入力**:学習済みの決定木 T. 検証データ D∠

**出力**: 枝刈り後の決定木 *T'* 

葉(Leaf)以外の ノードに限定

```
for all T のノード N, ルートから遠いものから順に do T_N \leftarrow N をルートとする部分木 D_N \leftarrow D 中で,T_N によってカバーされるデータ if accurary (T_N, D_N) \leq \text{majority}(D_N) then T_N を,リーフ majority_class (D_N) に置き換え end if end for return 枝刈り後の決定木 T'
```

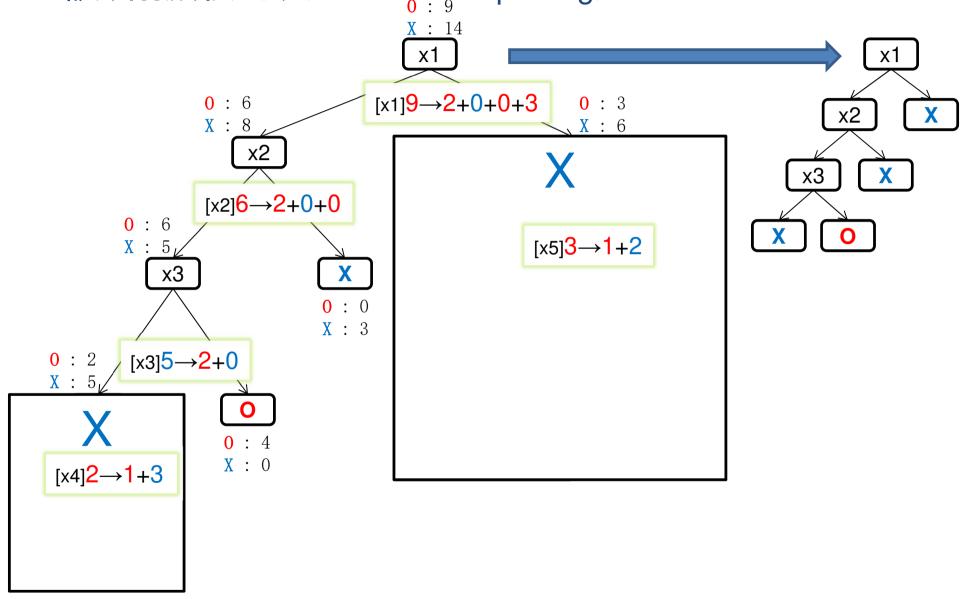
荒木著、『フリーソフトで始める機械学習入門』、森北出版、2014より

 $accurary(T_N, D_N)$ :決定木  $T_N$  を用いてデータ  $D_N$  を分類した際の正解率

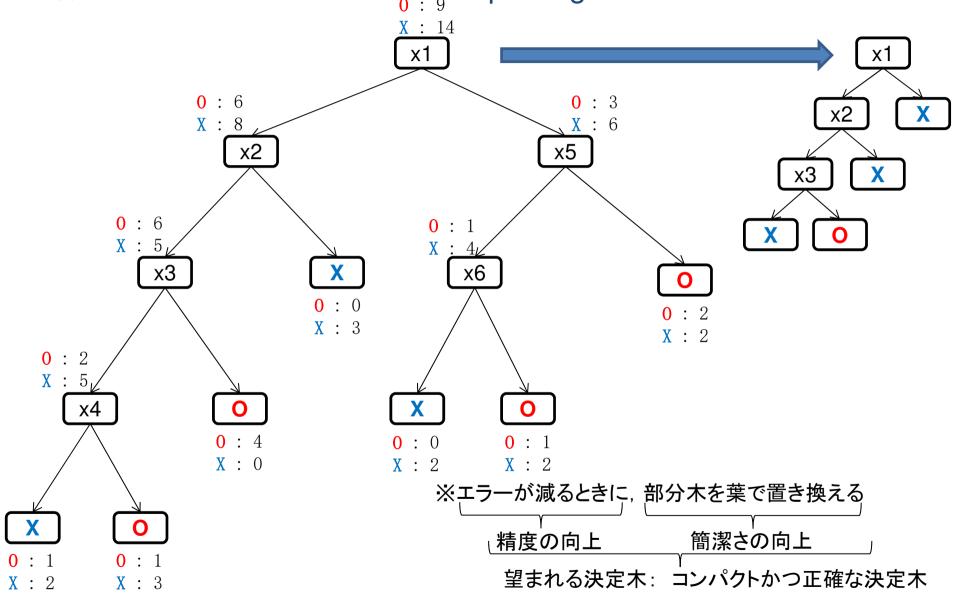
 $\mathsf{majority}(D_N)$ : データ  $D_N$  中の最頻クラスの割合

majority\_class( $D_N$ ): データ  $D_N$  中の最頻クラス

# 誤り削減枝刈り(reduced-error pruning)



# 誤り削減枝刈り(reduced-error pruning)



http://www.math.tau.ac.il/~mansour/ml-course-02/scribe10.ps より一部改変

# 悲観的枝刈り(pessimistic pruning)

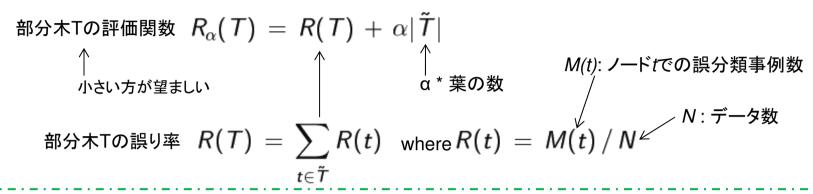
- 誤り削減枝刈りでは、評価用データが必要であった。
- 悲観的枝刈り(訓練例をサンプルだと思うことで、評価用データを必要としない)
  - 葉ノードに含まれるn個のデータを母集団から取り出した標本とみなす。
  - ・ 標本(訓練データ)のエラー率から、 裏にある母集団のエラー率を統計的に推定する
  - 推定したエラー率に基づいて枝刈りを行う
- ・ 誤り削減枝刈りも 悲観的枝刈りも、エラー率が小さくなる場合に木を剪定する
  - 両者の違いは、エラー率をどのように計算するか。
    - ・ 誤り削減枝刈り → 評価用データを利用して, エラー率を計算
    - ・ 悲観的枝刈り → 訓練用データ+統計的推定を利用して, エラー率を計算
- ・母集団のエラー率の推定
  - シンプルな方法:各葉ノードのエラー率を(誤分類事例数+0.5)/データ数とする
  - アドバンストな方法:2項分布を用いる.

# コスト複雑度枝刈り(cost-complexity pruning)

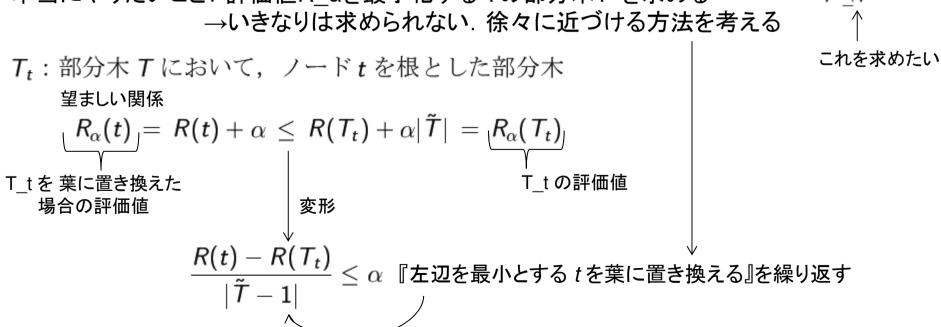
- 各ノードを根とする部分木の評価値を、以下のように定義する(形式的定義は次ページ)
  - ・コスト評価関数=訓練データによる部分木のエラーコスト + α\*部分木の葉の数
  - この値が、一定の閾値を超える部分木を枝刈りの対象とする。
- 何をやっているのか?(向かっている先を把握しよう)
  - やりたいこと:精度が高く(正確),かつ,簡潔な決定木が欲しい。
  - 一般に、木の簡潔さと精度はトレードオフの関係にある
    - 木を簡潔にすれば、精度が下がる / 精度を上がれば、木が複雑になる。
  - コスト評価関数が意味するところは...
    - (1)訓練データを使った見かけ上のエラーコストと(2)部分木の葉の数の和
      - → 不正確さ(精度の逆)と複雑さ(簡潔さの逆) の和
    - 両者の和が小さい → 簡潔 かつ 精度が高い
      - 和が大きい部分木→複雑 or 不正確 → よって、枝刈りの対象とする
    - αは、「簡潔さ」と「正確さ」のどちらを、どの程度重視するか表している。

# コスト複雑度枝刈り(cost-complexity pruning)

 $\tilde{T}$ : 部分木 T に含まれるリーフノードの集合



本当にやりたいこと: 評価値R αを最小化する Tの部分木T'を求める



### 決定木の評価

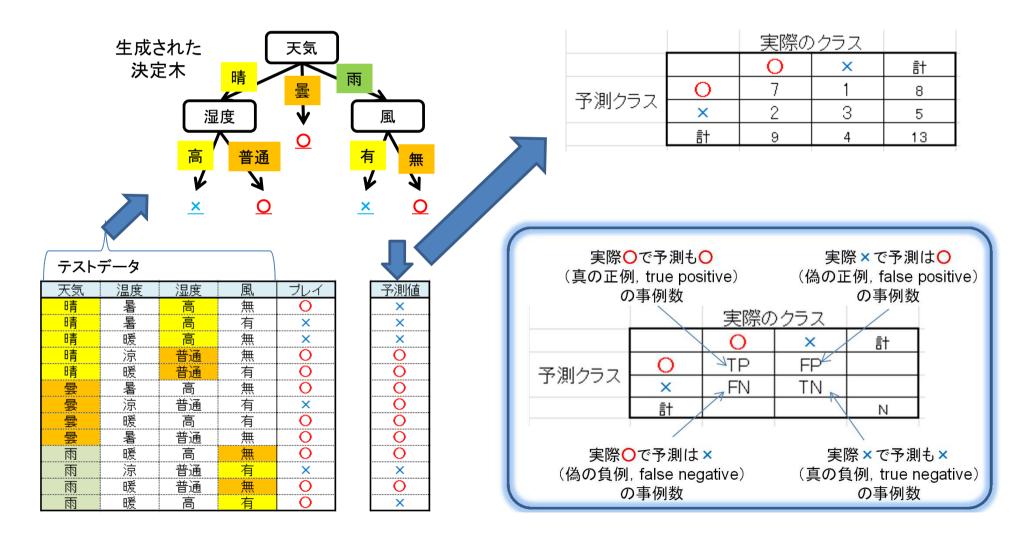
- 訓練データから生成された決定木は、どの程度良いものなのか?
- どの様にすれば、性能を評価することができるのか? そもそも性能とは?
- → 最も基本的な評価指標は「どれだけ予測が正解したか」(精度, Accuracy)
  - (訓練データとは別に)テストデータを準備する.
    - テストデータ中の各事例は、実際にはクラスの値が既知である。
  - 訓練データで作成した決定木を用いて、テストデータ中の各事例のクラスを予測
    - 実際には、各テスト例のクラスが分かるが、それを隠して予測してみるということ
  - 予測が当たった割合(精度, Accuracy)を計算
- ・ 性能の評価指標は、精度だけではない
  - 分割表による解析: 精度・再現率・適合率・F値, その他
  - 統計的検定: χ^2(カイジジョウ)検定

精度 = 予測が当たったテスト例の数 テスト例の総数

- 訓練データやテストデータが偏っているかも?一回の評価で大丈夫?
  - ・ 複数回の評価を行い、その平均・分散を考える → リサンプリングによる評価
    - 交差検定
    - ブーストラップ法

### 分割表(Contingency Table)

- 簡略化のため、「2クラス」問題を対象とします。
  - 2クラス:クラス属性(目的変数)の取り得る値が2値のいずれか.「正」「負」と呼ぶ。
- 構築された決定木を用いて、テストデータのクラスを予測し、その結果を集計する
  - ・ (1)実際のクラス と (2)予測クラス の値に従って, テストデータを4グループに分ける→分割表



### 分割表(Contingency Table)

TPに着目した評価値

		実際の	実際のクラス		
		0	×	計	
マミルムニョ	0	7	1	8	
予測クラス	×	2	З	5	
	計	9	4	13	

- 再現率(recall):
  - ・ 正例の内, 正しく正例と予測された割合
  - ・ 再現率が高い→漏れがない

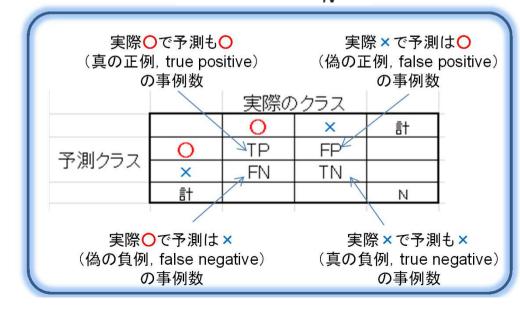
recall =  $\frac{TP}{TP + FN} = 7/9$ 

- · 適合率(precision):

  - ・ 適合率が高い→予測が正確
- F値(F-Measure): recallとprecisionの調和平均
  - 一般に、再現率と適合率はトレードオフの関係にある
    - 一方が大きくなれば、他方が小さくなる
    - 調和平均で考える

$$F - measure = \frac{1}{\frac{1}{2}(\frac{1}{recall} + \frac{1}{precision})}$$
$$= \frac{2 \cdot recall \cdot precision}{recall + precision}$$

$$Accuracy = \frac{TP + TN}{N} = 10/13$$



### (参考)幾何平均と調和平均

• 幾何平均: (相乗平均, Geometric Mean)

$$= \sqrt[n]{x_1 \cdots x_n} = \sqrt[n]{\prod_{i=1}^n x_i}$$

- ・「率の伸び縮み」の平均を計算する際に利用される.
- (Ex)売上 5万 が, その後2月で, 10万 (= 2倍), 100万(= 10倍)となった. 平均の「伸び率」は?
  - (2+10)/2=6倍ではない。
    - 仮に 6倍だとすると、5 x 6 x 6 = 180万 となり、正しくない
  - ・ (2 \* 10)^0.5 ≒4.5 が正解.
- (Ex. 投資収益率) 1年目 +9%, 2年目-15%, 3年目12%  $\sqrt[3]{(1+0.09)} \times \sqrt{1+(-0.15)} \times (1+0.12) = 1.0124$ 
  - 年平均、1.24%の収益率
- 調和平均: (Harmonic Mean)
  - [(個々のデータの逆数)の平均 ] の逆数  $=\frac{1}{\frac{1}{n}\sum_{i=1}^{n}\frac{1}{x_{i}}}$  「速度の平均」などに利用される
  - (Ex) 行きが時速30km. 帰りが時速40km. 平均の時速は?
    - (30 + 40)/2 = 35 (km)ではない.
    - 時速= 距離/時間(時間 = 距離/時速)
      - 距離をAとする/ 時間 = (行きは A/30(時間)+帰りはA/40時間)
      - (平均)時速は 2A / (A/30 + A/40)
  - ・ (Ex) F-measure ・ Recall (再現率) と Precision (適合率) の調和平均  $= \frac{1}{\frac{1}{2}(\frac{1}{30} + \frac{1}{40})}$
- ※ 算術平均 >= 幾何平均 >= 調和平均 が成り立つ

## 有効性の指標

感度(Sensitivity)	$\frac{TP}{TP+FN}$
特異度(Specificity)	$\frac{TN}{FP+TN}$
偽陽性率(False Positive Rate)	$\frac{FP}{FP+TN}$
真陰性率(True Negative Rate)	$\frac{FN}{TP+FN}$
陽性的中度(Positive Predictive Value)	$\frac{TP}{TP+FP}$
陰性的中度(Negative Predictive Value)	$\frac{TN}{FN+TN}$
検出率(Detection Rate)	$\frac{TP}{N}$
元田浩他, 『データマイニングの基礎』, オーム社, 2006 の表5.	17を改変

実際○で予測も○ (真の正例, true positive)

実際×で予測は〇 (偽の正例, false positive)

の事例数

の事例数 実際のクラス × 計  $\sqrt{A}$ FP 予測クラス × , FN ΤN 計 Ν

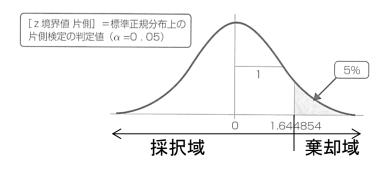
実際○で予測は× (偽の負例, false negative) の事例数

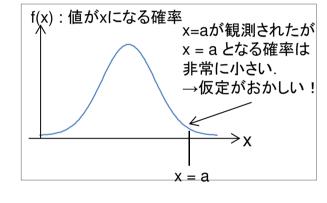
実際×で予測も× (真の負例, true negative) の事例数

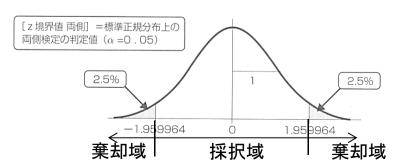
※それぞれの指標の意味を考えてみよう

## 統計的な検定 (x^2検定に向けて)

- データが確率分布に従っていることを利用し、統計的な意味で、信頼性があるか調べる
- ・『2つの対立する仮説を立てて、それを統計的に検証すること』
  - 帰無仮説: (ある意味で)成立して欲しくない仮説
  - 対立仮説:(ある意味で)成立して欲しい仮説
  - 検定では、帰無仮説を仮定し、それが否定(棄却)されるかどうかを判断する
- 検定の一般的な手順(簡略化バージョン)
  - 仮説を立てる: 帰無仮説と対立仮説を立てる.
  - 検定統計量を求める:
    - 検定に利用する統計量(=検定統計量)を決める.
  - 検定統計量から採択/棄却を判断する
    - 検定統計量は確率変数.ある確率分布に従う.
    - 検定統計量を確率分布に照らし合わせ、帰無仮説の棄却/採択の判断をする
      - 棄却:帰無仮説を否定する / 採択:帰無仮説を否定しない
    - 棄却/採択の判断には、有意水準を用いる。
      - 有意水準:棄却の基準となる確率.確率分布の端からの面積







# χ<sup>2</sup>(カイジジョウ)検定

- 実際のクラス と 予測クラス の「独立性」(独立:両者には関係がない、無関係)の検定
  - 独立(予測したものが、実際と無関係)では困る!
  - 帰無仮説:両者は独立である/対立仮説:両者は独立ではない とし. 帰無仮説を棄却したい
- 期待度数(期待値)と観測度数の"ズレ"を用いて検定を行う
  - 期待度数: 実クラスと予測クラスが独立の場合に得られる値
  - 観測度数: 実際に観測された値
- 期待度数の計算(実際の値から確率を考える)
  - Rp=0となる確率: n0/N, Rp=1となる確率: n1/N
  - Ra=0となる確率: m0/N, Ra=1となる確率: m1/N
  - Rp と Raが独立と仮定すると. . . (掛け算になる)
    - Rp=0 かつ Ra=0 となる確率: n0/N \* m0/N
    - Rp=0 かつ Ra=1 となる確率: n0/N \* m1/N
    - Rp=1 かつ Ra=0 となる確率: n1/N \* m0/N
    - Rp=1 かつ Ra=1 となる確率: n1/N \* m1/N

 $R_a$ 期待度数 観測度数 計 計 0 0  $\frac{n_0 m_0}{N}$  $\frac{n_0 m_1}{N}$  $R_p = 0$  $X_{00}$  $n_0$  $n_1 m_0$  $\underline{n_1 m_1}$  $x_{10}$  $x_{11}$  $n_1$  $n_1$ 計 Ν Ν  $m_0$  $m_1$  $m_0$  $m_1$ 期待度数: 事例数(N) \* 確率 両者のずれを、比較することで、

予測クラス

観測度数を使って,期待度数を計算する

RaとRp が独立であることを否定する

実際のクラス

TN

TP

FΝ

N \* n1/N \* m0/N

## 期待度数の計算

		-		
	0	×	計	
0	7	1	8	
×	2	ß	5	
計	9	4	13	
	$\sim$			
	O	×	計	-
	7	×		_
0		×	計 8	- 0 * 0 / 12
O ×	*	<b>X 1</b> 3		8 * 9 / 13
×	9	1	8	8 * 9 / 13

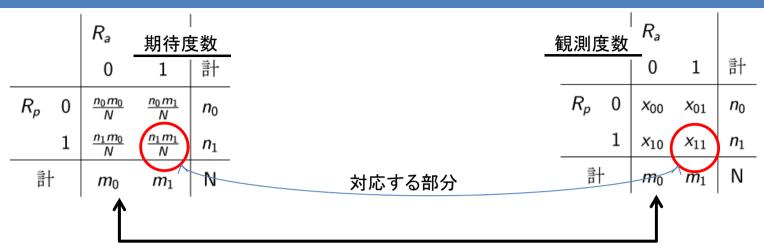
		-	
	0	×	計
0	7	1	8
×	1	3	5
計	9	4	13

				_
	0	×	計	
0	7	1	8	8 * 4 / 13
×	2	4	5	0 4/13
計	9	4	13	

		-		
	0	×	計	
0	7	1	8	5 * 4 / 13
×	2	<u> </u>	5	5 4/13
計	9	4	13	_

5 * 9 / 13	_		ı	ı
		0	×	計
	0	8*9/13	8*4/13	8
-		540 /12	E44/10	_

13



両者のずれを、比較することで、RaとRp が『独立であることを否定する』

#### 検定統計量(x^2値)

$$\sum_{p \in \{0,1\}} \sum_{a \in \{0,1\}} \frac{\left( \text{観測度数}_{pa} - \text{期待度数}_{pa} \right)^2}{\text{期待度数}_{pa}} = \sum_{p \in \{0,1\}} \sum_{a \in \{0,1\}} \frac{\left( x_{pa} - \frac{n_p m_a}{N} \right)^2}{\frac{n_p m_a}{N}}$$

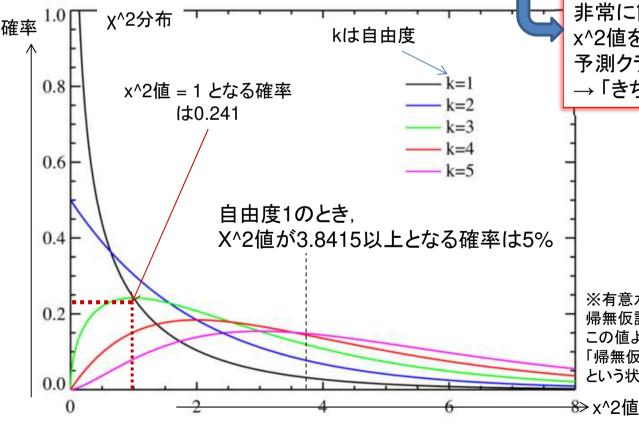
#### 基本的なアイディア

- 期待度数: 予測クラスと実クラスが独立の場合に得られる値
- 期待度数と観測度数値が近い → 予測クラスと実クラスは独立
- 期待度数と観測度数とのずれが大きい → 予測クラスと実クラスは独立であるとは言えない
- 検定統計量(x^2値)が大きい → 「予測クラスと実クラスは独立である」を棄却

$$\sum_{p \in \{0,1\}} \sum_{a \in \{0,1\}} \frac{\left(観測度数_{pa} - 期待度数_{pa}\right)^{2}}{期待度数_{pa}} = \sum_{p \in \{0,1\}} \sum_{a \in \{0,1\}} \frac{\left(x_{pa} - \frac{n_{p}m_{a}}{N}\right)^{2}}{\frac{n_{p}m_{a}}{N}}$$

この値は、自由度1のx^2分布に 従うことが知られている

- 検定統計量(x^2値)が大きい → 「予測クラスと実クラスは独立である」を棄却
- 予測クラスと実クラスが独立の場合に、x^2値が3.841以上となる確率は5%
- 有意水準を5%とした場合、x^2値が3.841以上であれば帰無仮説(独立である)を棄却できる.



非常に簡単に言えば.

x^2値を計算して、3.841以上であれば、 予測クラスと実クラスは独立ではない. → 「きちんと予測能力がある」と考える

> ※先に有意水準を決めてから、 x^2値を確認するのが正しい

有意水準 5% 3.841 有意水準 1% 6.635 有意水準 0.1% 10.828

#### ※有意水準:

帰無仮説を棄却する基準

この値より小さければ、「偶然ではない」と考える. 「帰無仮説が正しいのに、偶然x^2値が大きくなった」 という状況の「偶然」をどこまで考えるかということ

$$\sum_{p \in \{0,1\}} \sum_{a \in \{0,1\}} \frac{\left( 観測度数_{pa} - 期待度数_{pa} \right)^2}{ 期待度数_{pa}} = \sum_{p \in \{0,1\}} \sum_{a \in \{0,1\}} \frac{\left( x_{pa} - \frac{n_p m_a}{N} \right)^2}{\frac{n_p m_a}{N}}$$
 この値は、自由度1の $\chi^2$  分布に 従うことが知られている

- 注意事項: ∠
  - ・ データ数が少ない(各セルの値が5未満など)場合は、X^2分布への近似が悪くなる
  - ・ 検定の精度も悪くなるので、別の方法を利用する方が良い

# $\chi^2$ (カイジジョウ)検定:自由度と3クラス以上への拡張

- 自由度:変数のうち独立に選べるものの数
  - ・今回の場合、分割表で、各「計」が与えられているとき、 「いくつのセルの値を自由に設定できるか」ということ
  - 2x2の場合: 各「計」が与えられている→ n0, n1, m0, m1 の値は与えられている
    - このとき、x00 の値を決めると、x01,x10,x11の値は自動的に決まる
      - ・ 自由に値を決めることができるセルは、1つのみ→自由度1

		$R_a$		
		0	1	計
$R_p$	0	x <sub>00</sub>	<i>x</i> <sub>01</sub>	$n_0$
	1	<i>x</i> <sub>10</sub>	<i>x</i> <sub>11</sub>	$n_1$
計	-	$m_0$	$m_1$	N

- 3クラス以上を対象としたx^2検定
  - 基本は同じ. 期待値と観測のずれを合計
    - クラス数が n のとき, 自由度は (n-1)\*(n-1)となる.

$$\sum_{p \in \{0,1,2\}} \frac{\left(\text{観測度数}_{pa} - \text{期待度数}_{pa}\right)^{2}}{\text{期待度数}_{pa}}$$

$$= \sum_{p \in \{0,1,2\}} \sum_{a \in \{0,1,2\}} \frac{\left(x_{pa} - \frac{n_{p}m_{a}}{N}\right)^{2}}{\frac{n_{p}m_{a}}{N}}$$

	R <sub>a</sub>	観測	則度数	<u>女</u>	
			則度数 2	l	
$R_{p}$ 0	x <sub>00</sub>	<i>x</i> <sub>01</sub>	x <sub>02</sub> x <sub>12</sub> x <sub>22</sub>	$n_0$	
1	x <sub>10</sub>	<i>x</i> <sub>11</sub>	<i>x</i> <sub>12</sub>	$n_1$	
2	x <sub>20</sub>	<i>x</i> <sub>21</sub>	<b>x</b> <sub>22</sub>	<b>n</b> <sub>2</sub>	
			<i>m</i> <sub>2</sub>		

		Ra	期待度数		_
		0	1	2	計
$R_p$	0	n <sub>0</sub> m <sub>0</sub> N	$\frac{n_0 m_1}{N}$	<u>n₀m₂</u> N	n <sub>0</sub>
	1	$\frac{n_1 m_0}{N}$	$\frac{n_1 m_1}{N}$	$\frac{n_1 m_2}{N}$	$n_1$
	2	n <sub>2</sub> m <sub>0</sub> N	$\frac{n_2 m_1}{N}$	$\frac{n_2m_2}{N}$	$n_1$
計		<i>m</i> <sub>0</sub>	$m_1$	$m_2$	N

### リサンプリングによる評価:交差検定法(Cross Validation)

- 評価方法に対する疑問
  - 一回の評価で大丈夫なのか?準備したテストデータが偏っているのでは?
     →データを変えながら、何回も評価して、検証をして、その平均を考えよう!
- 何度も評価するには、何度もテストデータが必要→ データを分割し、役割を変えて再利用
- ・ 交差検定法:上記の考えに基づく評価方法の一つ

#### K-fold 交差検定の手順

- 1 データ集合 D を, k 個の均等なデータセット  $\{D_1, D_2, \dots, D_k\}$  に分割
- 2 各  $D_i$  (1  $\leq i \leq k$ ) に対し、以下を行う
  - 2.1  $D \setminus D_i$  を訓練データとして、決定木  $T_i$  を生成する
  - 2.2  $D_i$  をテストデータとして、 $T_i$  を評価し、評価値  $v_i$  を得る
- 3  $v_i$  (1  $\leq i \leq k$ ) の記述統計量 (平均,分散) を求める

特別な基準で分割を行った場合に対し、それぞれ名前がついている

クラス分布を崩さない様に分割する場合を, 「層化交差検定法(Stratified Cross Validation)と呼ぶ

Dを, k = |D| 個に分割する場合を, Leave-one out法と呼ぶ

## リサンプリングによる評価:交差検定(Cross Validation)

