情報意味論 (第2回)

慶應義塾大学理工学部
櫻井 彰人

補足: 機械学習環境

- Weka: **説明済み**
- Yale: yet another learning environment
  - http://www-ai.cs.uni-dortmund.de/SOFTWARE/YALE/index.html
- **掲示板**
  - http://www.kdkeys.net/forums/

なぜ機械学習か？

様々な意味で「計算能力が向上」
- データベースマイニング: データを知識に
- 自動カスタマイズプログラム: ニュースのフィルタ, 適応的な監視カメラ
- 行動の学習: ロボットの計画, 制御の最適化, 決定支援
- プログラム困難なアプリケーション: 自動運転, 音声認識

人間の学習や教育のよりよい理解を求めて
- 認知科学: 知識獲得の理論 (e.g., 実践を通じて)
- パフォーマンス向上: 推論・推測, 推薦システム

時は今、、、
- 学習アルゴリズムや理論の最近の進歩は目覚しい
- 様々なソースから大量のオンラインデータが提供される
- 計算機は安価・高速
- 機械学習を用いた事業が発生・成長 (e.g., データマイニング/KDD)

ニューラルネットワーク

Autonomous Learning Vehicle In a Neural Net (ALVINN): Pomerleau *et al*
Navlab-5 に到り終了 (1995). 高速道路を 70mph で. "No Hands Across America"
http://www-2.cs.cmu.edu/afs/cs/user/tjochem/www/nhaa/nhaa_home_page.html

関連領域

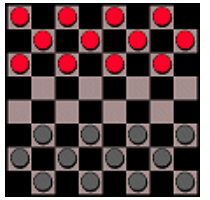
認知科学: 言語獲得、推論の学習
統計学: バイアス vs. 分散, 信頼区間, 仮説検定
ベイズの方法: ベイズの定理、欠測値の推定
人工知能: 記号表現、計画、知識を用いた学習
計算の複雑さの理論: PAC 学習、VC次元、誤差限界
制御理論: 最適化、動的計画、予測の学習
情報理論: エントロピー, MDL, 情報源符号化
神経科学: 人工神経回路網、脳 (大脳、小脳、視床下部)
哲学: オッカムの剃刀, 帰納的一般化
心理学: 練習の冪法則(Power Law of Practice) 発見的学習

学習課題とは

学習 = あるタスクにおいて経験により改善すること
- あるタスク T において, 改善がなされること
- 但し、ある性能指標 P に対してであり,
- 経験 E に基づいてなされるものとする

例: チェッカーでの学習
- T: チェッカーゲームを行うこと
- P: 世界選手権での勝率
- E: 自分自身とゲームを行う機会

問題仕様の具体化
- *経験* とは何か?
- 学習されるべきものを *具体的* に
- その *表現* は?
- 具体的な学習アルゴリズム?



補足:
- チェッカーのゲーム例 http://www.h4ns.net/checkers/index.html
- 試してみよう http://www.zipschocolatechip.com/puzzles/checkers/checkers.htm

例:チェッカー

練習経験の型
- 直接的か間接的か?
- 教師がいるかいないか?
- 手に関する知識 (e.g., 序盤の定石/終盤の詰め)?

問題: 練習経験は性能目標を達するために十分か?

ソフトウェアの設計
- 前提条件: 規則にあった手 legal move の生成器は存在する
- 作るべきもの: 手の生成器, 評価器, パラメータ付きの目標関数

目標関数の選択
- *ChooseMove: Board → Move* (行動選択関数, or *方策 policy*)
- *V: Board → R* (盤面の評価関数)
- 理想の V; その近似 $\hat{V}$
- 学習過程の目的: V の operational (実際に演算可能) な記述 (近似)

チェッカーの目標関数

ありうる定義
- b が詰んだ勝った盤面であれば, $V(b) = 100$
- b が詰んで負けた盤面であれば, $V(b) = -100$
- b が引分けの最終盤面であれば, $V(b) = 0$
- b が最終盤面でないとき, $V(b) = V(b')$ となるのは, b' が最良の最終盤面で, b から始めて最終盤面 b' まで最善の手を尽くして到達できる場合
- *値としては間違っていないが, operational (実際に演算可能) ではない*

目標関数の記述方法
- 計算規則の集合?
- ニューラルネットワーク?
- 盤面の特徴 feature の多項式 (e.g., 線型, 2次関数の組合わせ等)
- 等々

学習される関数
- bp, rp = 黒ピース, 赤ピースの個数; bk, rk = 黒キング, 赤キングの個数;
- bt, rt = 取られそうになっている黒ピース, 赤ピースの個数

$$\hat{V}(b) = w_0 + w_1 bp(b) + w_2 rp(b) + w_3 bk(b) + w_4 rk(b) + w_5 bt(b) + w_6 rt(b)$$

チェッカーの訓練手続き

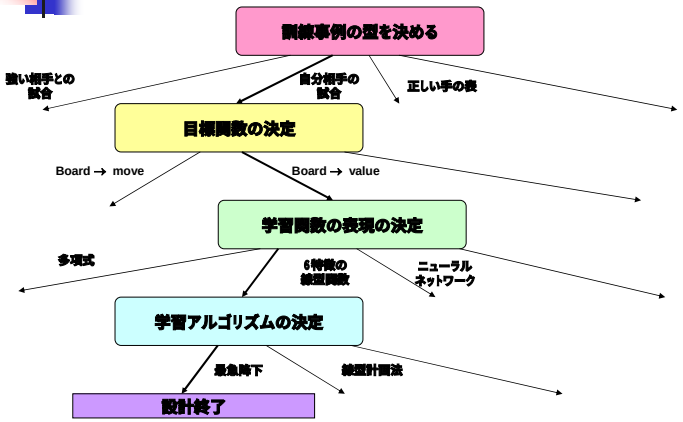
訓練例をえて
- $V(b)$ 目標関数
- $\hat{V}(b)$ 学習関数
- $V_{train}(b)$ 訓練値

訓練値を得る規則:
- $V_{train}(b) \leftarrow \hat{V}(Successor(b))$

荷重変更の規則
- 自乗平均最小化 (LMS) 荷重更新規則:
  - REPEAT
    - 訓練例 b をランダムに選択
    - $error(b)$ を計算
      - $error(b) = V_{train}(b) - \hat{V}(b)$
    - それぞれの盤面特徴 f_i に対し、その荷重 w_i を更新:
      - $w_i \leftarrow w_i + c \cdot f_i \cdot error(b)$
      - 但し c は正の小さい定数で、学習係数
      - これを発展させると、TD学習になる

Arthur Samuel(1901-1990) : Playing checkers at SAIL, with teletype ~1970

チェッカー学習の設計順序



機械学習の課題

どんなアルゴリズムなら、関数近似がうまくいくか？

学習システムの設計要素は、精度にどのように影響するか
- 訓練事例数
- *仮説表現* の複雑度

学習課題の性質は、精度にどのように影響するか?
- データにノイズがある
- データの発生もとが複数

学習可能性の理論的限界は?

事前知識を役立てるにはどうすべきか?

生物の学習システムからどんなヒントが得られるか?

学習システム自身が表現を変えることができるか?

何を学習するか

- 分類関数
  - 関数の形が分かっている場合: パラメータ推定 ("fitting")
  - 概念学習 (e.g., 椅子、机、顔、テニスをするに最適な天候)
  - 診断、予知: 医療、リスク評価, 詐欺行為, 機械システム
- モデル
  - 地図 (未知領域の探索)
  - 分布関数 (質問応答)
  - 言語モデル (e.g., オートマトン/文法)
- スキル
  - ゲーム
  - 計画立案
  - 推論方法
- パターン認識のためのクラスターの定義
  - 物体の形
  - 関数または分類階層 taxonomy
- *実は、分類に帰着される問題は多い*

学習の方法

- **教師付き学習** supervised learning
  - **学習結果**: **分類関数, モデル他何でも**
  - **入出力**: examples $\langle x, f(x) \rangle \rightarrow$ approximation $\hat{f}(x)$
  - **どうやって?** **教師が正解を示す**
- **教師なし学習** unsupervised learning
  - **学習結果**: **クラスタ, ベクトル量子化** (*codebook* **ともいう**)
  - **入出**: observations $x \times$ distance metric $d(x_1, x_2) \rightarrow$ discrete codebook $f(x)$
  - **どうやって?** **データの不均一性, 粗密, 但し, 近さ・遠さの距離関数が必要**
- **教師なし, 報酬あり**: **強化学習** reinforcement learning
  - **学習結果**: **行動方策 (状態の観測値から行動への関数)**
  - **入出力**: state/reward sequence $\{\langle s_i, r_i \rangle : 1 \le i \le n\} \rightarrow$ policy $p : s \rightarrow a$
  - **どうやって? 選択した行動の結果得られる(遅延) 報酬. 報酬と観測状態 (部分観測可能な場合も含む) とからモデルを更新する**

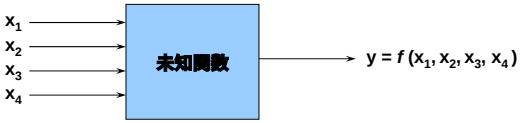
(教師付き) 概念学習

- **所与**: **ある未知関数** f **の訓練例** $<x, f(x)>$
- **目的**: f **のよい近似**
- **例 (概念学習以外)**
  - **病名診断**
    - x = **患者の特徴 (病歴, 症状, 検査結果)**
    - f = **病名 (または推奨する治療方法)**
  - **リスク評価**
    - x = **消費者の特徴, 契約者 (人口統計的情報, 事故履歴)**
    - f = **リスク水準 (期待コスト)**
  - **自動運転**
    - x = **車のフロントから見える道路の画像**
    - f = **ハンドルを回すべき角度**
  - **品詞付け**
  - **不正アクセス・侵入検知**
  - Web **ログ解析**

「学習」の枠組み

- **所与**: **ある未知関数** f **の訓練例** $<x, f(x)>$
- **目的**: f **のよい近似**
- **道具**: **仮説空間, 探索順序, 評価関数**
  - **仮説空間**
    - **仮説**: f **を近似する解候補**
    - **保持する仮説は, 一時には一つのことが多い. 複数の場合もある**
    - **離散空間のことも連続空間のこともある**
  - **探索順序**
    - **一時には一つの仮説しかチェックできない**
    - **探す順序によって, 解の良さや元まるまでの時間が異なる**
  - **評価関数**
    - **仮説と未知関数との違いを数値化するもの**
    - **未知関数は未知なので, 近似値を用いることになる**

学習問題



訓練例	x_1	x_2	x_3	x_4	y
0	0	1	1	0	0
1	0	0	0	0	0
2	0	0	1	1	1
3	1	0	0	1	1
4	0	1	1	0	0
5	1	1	0	0	0
6	0	1	0	1	0

- x_i: t_i, y: t, f: $(t_1 \times t_2 \times t_3 \times t_4) \rightarrow$ t
- **学習関数**: **集合**{ $(t_1 \times t_2 \times t_3 \times t_4 \times t)$ }$\rightarrow (t_1 \times t_2 \times t_3 \times t_4) \rightarrow$ t

仮説空間: 無制限の場合

- $|A \to B| = |B|^{|A|}$
- $|H^4 \to H| = |\{0,1\} \times \{0,1\} \times \{0,1\} \times \{0,1\} \to \{0,1\}| = 2^{2^4} = 65536$ 個の関数値あり
- 完全に無知で学習は可能か?
  - その場合、全ての入出力の組を知らないといけない
  - 7 例の後、まだ f には $2^9 = 512$ 個 (65536中) の可能な組がある

訓練例	x_1	x_2	x_3	x_4	y
0	0	0	0	0	?
1	0	0	0	1	?
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	?
8	1	0	0	0	?
9	1	0	0	1	1
10	1	0	1	0	?
11	1	0	1	1	?
12	1	1	0	0	0
13	1	1	0	1	?
14	1	1	1	0	?
15	1	1	1	1	?

概念学習

- 概念:
  - 対象集合を (より大きな集合の部分集合として) 特徴付ける記述
  - 特に、ブール値関数で記述できるもの
- 概念学習:
  - あるブール値関数を、その入力値と出力値の対からなる訓練例から推定すること
- 例: EnjoySport

Sky	AirTemp	Humid	Wind	Water	Forecast	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

- これらを一般化した概念は何だろう

仮説の表現

- 仮説の表現はいろいろありうる
  - 実は、仮説空間自体もいろいろありうる
- (これから説明する概念学習では) 各仮説 h は属性値に対する制約の連言 (and)
- 各属性に対する制約は、
  - 特定値 (Water = Warm)
  - don't care (Water = ?)
  - どんな値も許さず (Water=∅)
- 例

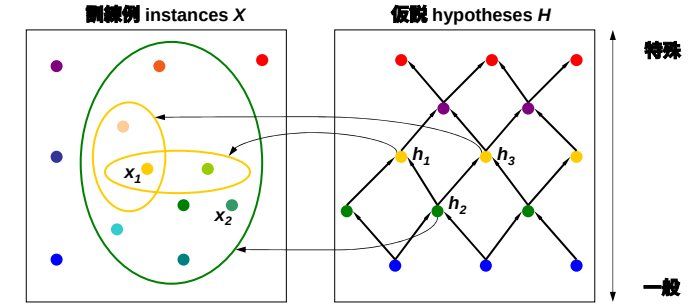
< Sky: *Sunny*, AirTemp: ?, Humid: ?, Wind: *Strong*, Water: ?, Forecast: *Same* >

概念学習の基本

- 下記の所与条件のもと
  - 事例集合 X: 次の属性で表現される *Sky, AirTemp, Humidity, Wind, Water, Forecast*
  - 目標関数 c: EnjoySport: $X \to \{0,1\}$
  - 仮説空間 H: リテラルの連言。例えば、
    - , ?, *Cold*, *High*, ?, ?, ? Ú
  - 訓練例 D: 目標関数の正例及び負例
- 次を求める:
  - H の中の h で D 中の全ての x に対して、$h(x) = c(x)$ となるもの

暗黙の前提

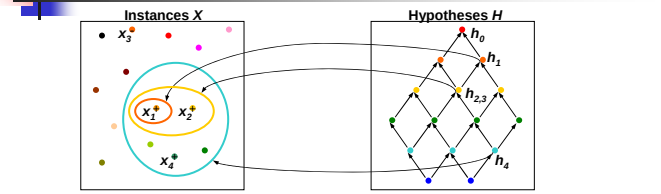
- 十分大きな訓練事例において目標関数を近似する仮説は、また、他の未観測の事例に対しても目標関数を近似する
- 統計的枠組みのもとで、「高い確率で正しい」ということがわかる。
- 確率が導入できない場合でも、「真の仮説は十分短い」という前提のもと、正しい



Find-S アルゴリズム

- h として H の中で最も特殊なものを選ぶ
- 正の訓練事例の各々の x に対し
  - h 中の属性制約 a_i に対し
    - もし、a_i が x によって満たされるなら何もしない
    - そうでなければ、a_i を x によって満足される，より一般的な制約に改めよ
- h を出力する

Find-S の仮説空間探索



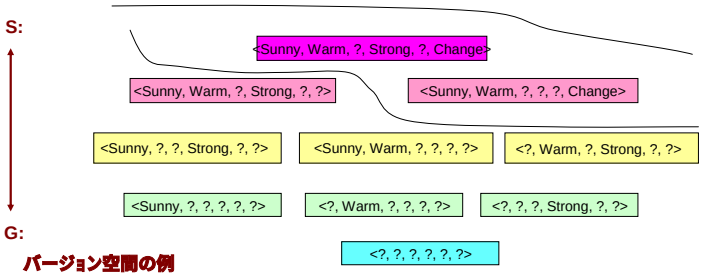
- *Find-S* の短所
  - 真の目標概念に到達したかが不明: 可能な解に何があるか分からない
  - 最も特殊な h を選んでしまう: もっと一般的な h を選んではいけないか？
  - 訓練事例が整合的でないかが分からない
  - 仮説空間 H に依存し、最も特殊であっても解が複数個ありうる

Find-S の短所克服

- 候補除去 (candidate-elimination) アルゴリズム
  - 訓練事例に整合的な仮説を全て求めよう
    (勿論、全てを枚挙することはしない)
- 整合的な仮説
  - 仮説 h が目標関数 c の訓練事例集合 D と整合する (consistent) であるというのは、D 中の任意の訓練例 $\langle x, c(x)\rangle$ に対して、$h(x)=c(x)$ となることを言う。
  - Consistent$(h,D)=(\forall\langle x,c(x)\rangle\in D)\ h(x)=c(x)$
- バージョン空間
  - (仮説空間 H と訓練例集合 D に関する) バージョン空間 $VS_{H,D}$ とは、D 中の全ての訓練例と整合する仮説 $h\in H$ の集合のことである
  - $VS_{H,D}\equiv\{h\in H|\ \text{Consistent}(h,D)\}$

List-Then-Eliminate アルゴリズム

- $VS \leftarrow H$ の全仮説からなるリスト
- 各訓練事例 $\langle x,c(x)\rangle$ につき、VS から $h(x)\neq c(x)$ となる h を除く
- VS を出力する



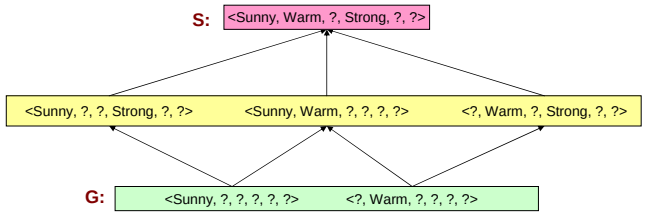
バージョン空間の例

バージョン空間の表現

- 仮説空間 (一般の場合) H
  - 有限交叉性を持つ半束 (半順序 $x \preceq_p y$ は、y はより特殊か等しい; $\perp \equiv$ most general)
  - どの二つの仮説にも *最大下界 greatest lower bound* (GLB) が存在する
  - $VS_{H,D}\equiv$ 整合的な仮説の部分集合
- 定義: 一般化側境界 General Boundary
  - $VS_{H,D}$ の一般化側境界 G : 最も一般な要素の集合
  - 最も一般な要素 $\equiv VS_{H,D}$ の極小要素 ≡ "必要条件の集合"
- 定義: 特殊化側境界 Specific Boundary
  - $VS_{H,D}$ の特殊化側境界 S: 最も特殊な要素の集合
  - 最も特殊 $\equiv VS_{H,D}$ の極大要素 ≡ "十分条件の集合"
- バージョン空間
  - バージョン空間内の全ての要素は S と G の間にある
  - $VS_{H,D}\equiv\{\ h\in H\mid(\exists\ s\in S).\ (\exists\ g\in G).\ g\preceq_p h\preceq_p s\ \}$
    但し、$x\preceq_p y\equiv$ x は y より一般的か等しい (正確には、より特殊ではない)

バージョン空間の例

- 特殊化側境界(S)と一般化側境界(G)とに挟まれた領域



候補削除アルゴリズム (1/3)

- G ← H 中で「一般化が極大」である仮説の集合
  - {<?, … , ?>} と記す
- S ← H 中で「特殊化が極大」である仮説の集合
  - {<O, … , O>} と記す
- 各訓練事例 d 毎に次のことを行なう

候補削除アルゴリズム (2/3)

- d が正例であるとき
  - d と不整合な仮説を G から除去
  - S中のd と不整合な仮説 s につき
    - s をS から除去
    - s の一般化で、一般化が極小となる h を全て S に加える
      - h は d と整合的であり、
      - G の中に h より一般なものがある

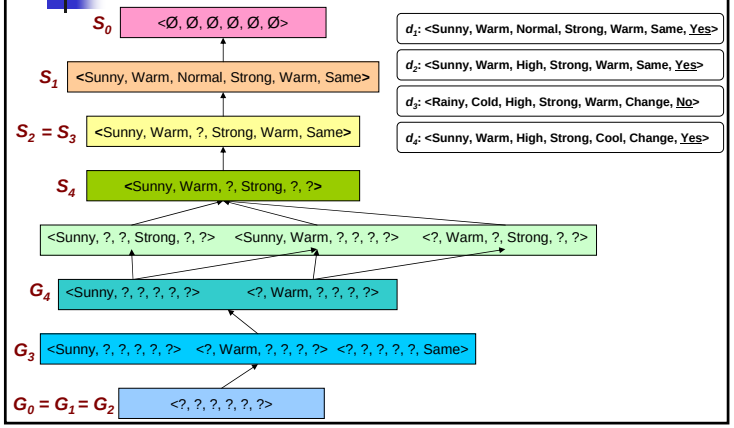
      (これらは$VS_{H,D}$中の最大下界, すなわち交わり, $s \vee d$ である)
    - Sから、S 中の他の仮説より一般な仮説をすべて除去する

候補削除アルゴリズム (3/3)

- d が負例であるとき
  - d と不整合な仮説を S から除去
  - G 中の d と不整合な仮説 g につき
    - g を G から除去
    - g の特殊化で特殊化が極小である h を全て G に加える
      - h は g と整合的であり、
      - S の中に h より特殊なものがある

      (これらは$VS_{H,D}$中の最小上界, すなわち和, $g \wedge d$ である)
    - G から、G 中の他の仮説より特殊な仮説をすべて除去する

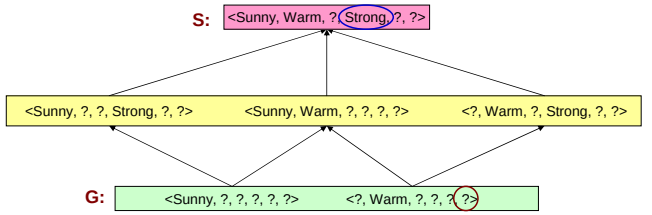
動きのトレース



補足

- 真の解に収束するか？
  - 訓練例に誤りなく、真の解が仮説空間内にあるならば、yes
  - 訓練例に誤りがあれば、バージョン空間は空集合となる
- 質問 query が許される場合
- 未知事例に対する予測

最もよい次の質問と未知事例



- 学習者が（もし質問できるとしたら）何を聞くのが最良か?
- 次のものはどう分類されるべきか?
  - <Sunny, Warm, Normal, Strong, Cool, Change>
  - <Rainy, Cold, Normal, Light, Warm, Same>
  - <Sunny, Warm, Normal, Light, Warm, Same>

帰納的飛躍の正当化

- 例: 次の帰納的一般化は正当化できるか？どうして？
  - 正例: <Sunny, Warm, Normal, Strong, Cool, Change, Yes>
  - 正例: <Sunny, Warm, Normal, Light, Warm, Same, Yes>
  - 帰納結果 *S*: <Sunny, Warm, Normal, ?, ?, ?>
- 未知データが分類できると考えるのは何故?
  - 例えば, <Sunny, Warm, Normal, Strong, Warm, Same>
  - （新しい事例に）どれだけ情報があれば、予測してもよいとするのか？

概念学習のまとめ

- 仮説空間 *H* における探索としての 概念学習
  - 学習: 正しい仮説を見出すこと
  - 複数あると: 最も特殊なもの or 全部
- 仮説空間 *H* をより一般的なものからより特殊なものへと順序つける
  - 半順序
  - *H* 中に上限と下限がある → 束
- バージョン空間を用いた候補削除アルゴリズム
  - *S* および *G* 境界が、学習者が持つ不確実さ uncertainty を表現
  - バージョン空間を用いると、未知事例に対して予測を行うことができる
- （バージョン空間があると）有用な質問が可能