

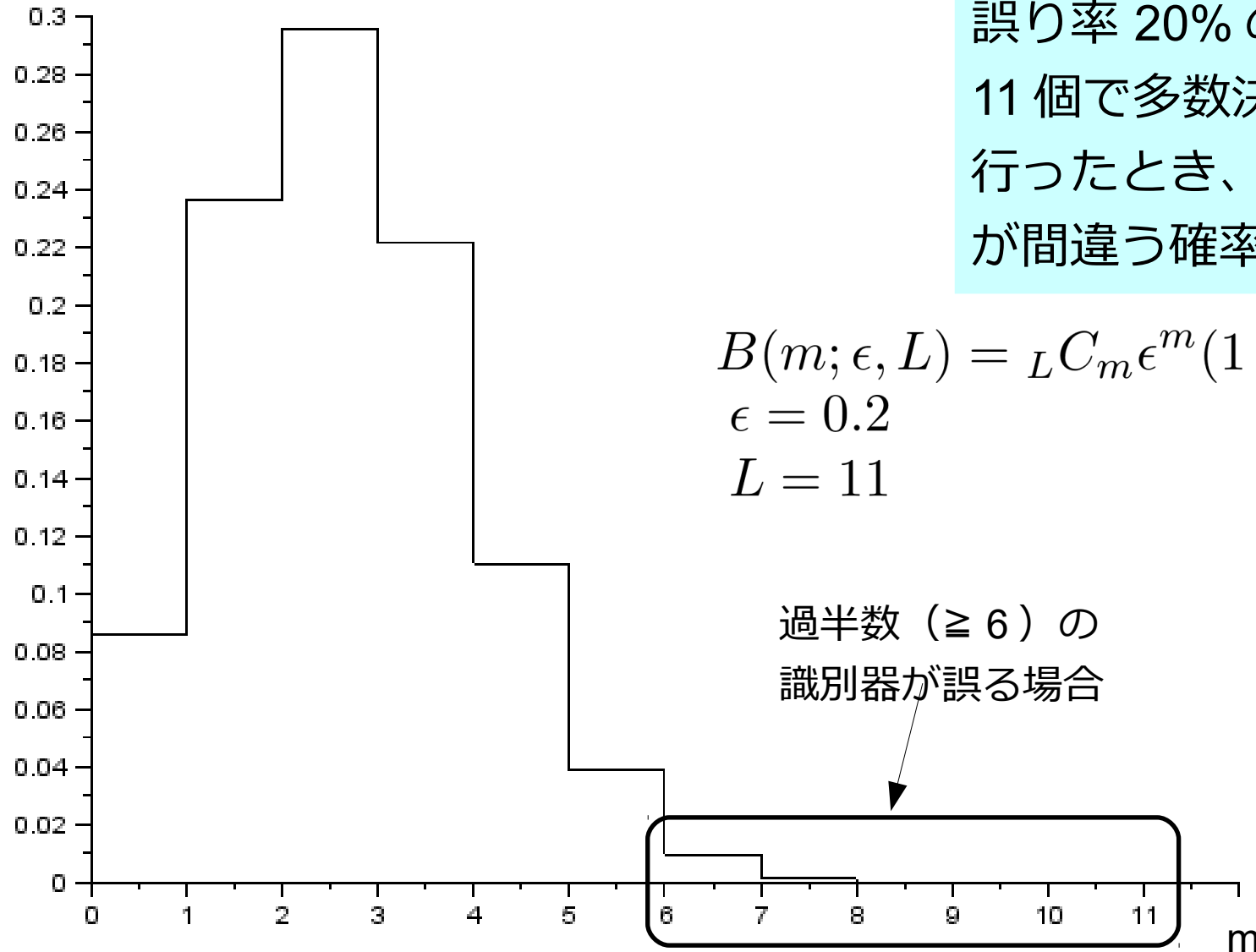
Section 4

- 高度な機械学習 (9,12,13章)
- 深層学習 (15章)

9. アンサンブル学習

- アンサンブル学習とは
 - 識別器を複数組み合わせ、それらの結果を統合することで個々の識別器よりも性能を向上させる方法
 - 個々の識別器を弱識別器とよぶ
- アイディア
 - 学習データから全く独立に L 個の識別器 (誤り率 ε , 誤りは独立) を作成
 - m 個の識別器が誤る確率は二項分布 $B(m; \varepsilon, L)$
 - $\varepsilon < 0.5$ のとき、 $m > L/2$ となる B は小さい値

9.1 なぜ性能が向上するのか



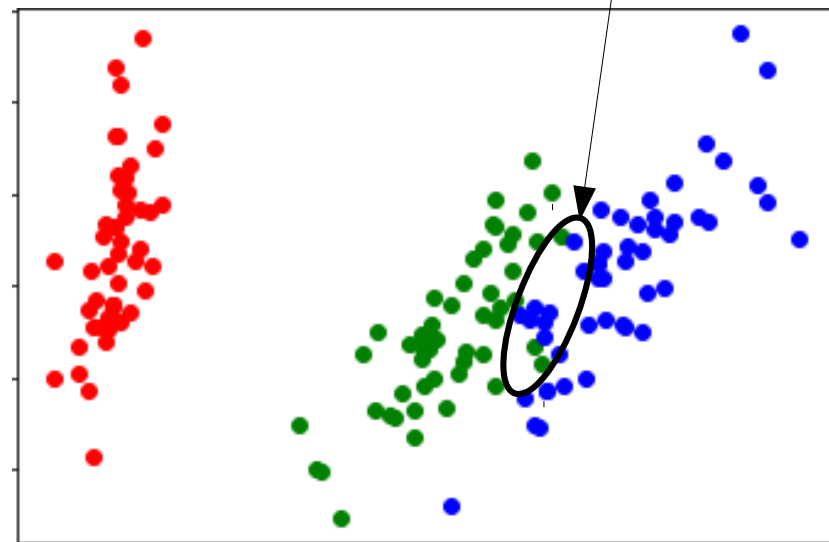
9.1 なぜ性能が向上するのか

- ここまでの議論の非現実的なところ
 - それぞれの弱識別器の誤りが独立である

➡ データの誤りやすさに差はない ✕

識別面付近のデータなど、
普通は成立しない

多くの弱識別器が誤る



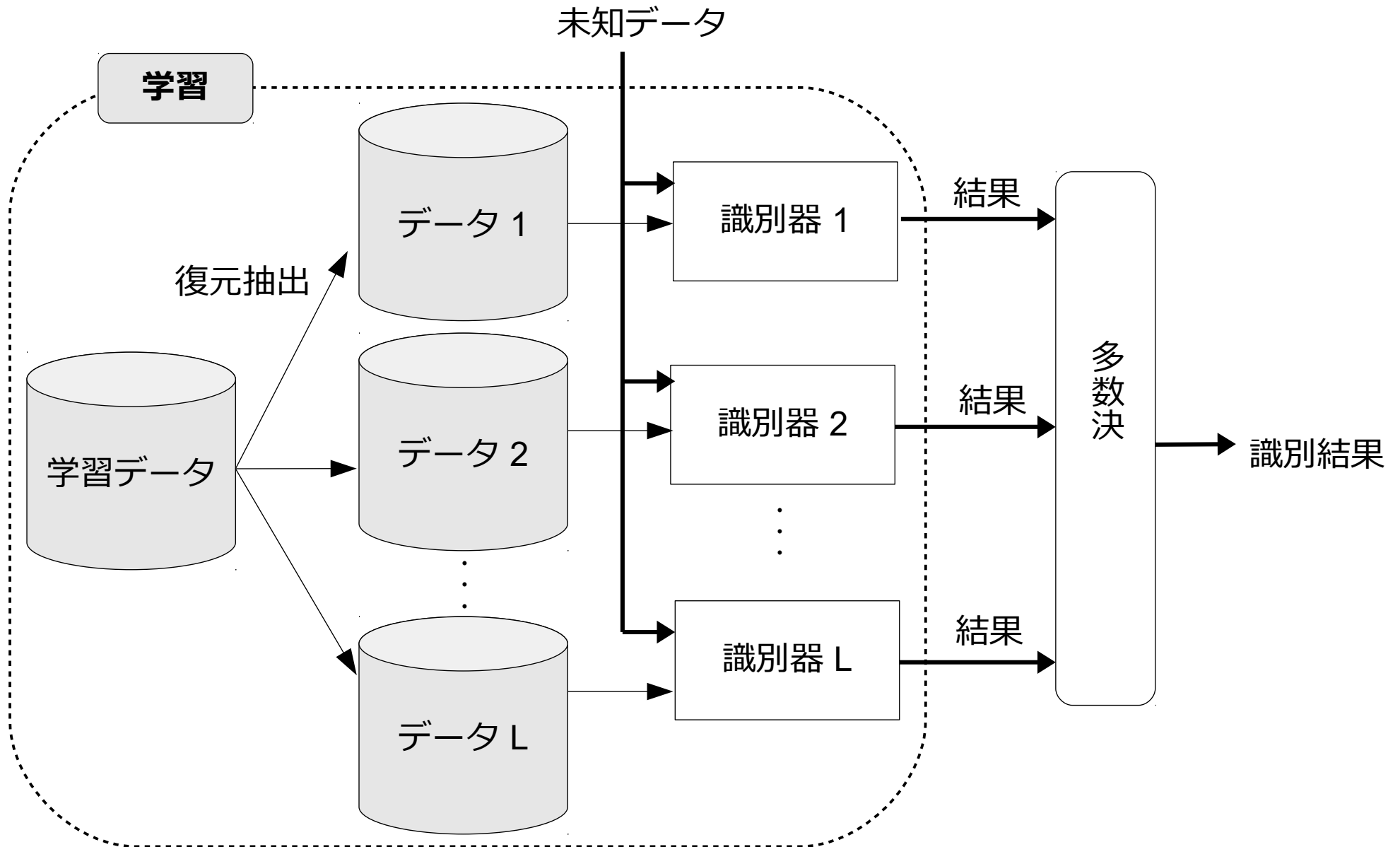
- アンサンブル学習の目標
 - なるべく異なる振る舞いをする弱識別器を作成する

9.2 バギング

- アイディア
 - 異なる学習データから作成された識別器は異なる
- Bagging とは
 - **bootstrap aggregating**
 - 元の学習データから、異なる部分集合を作成し、そのそれぞれに対して識別器を作成して、結果を統合する

9.2 バギング

- バギングの構成

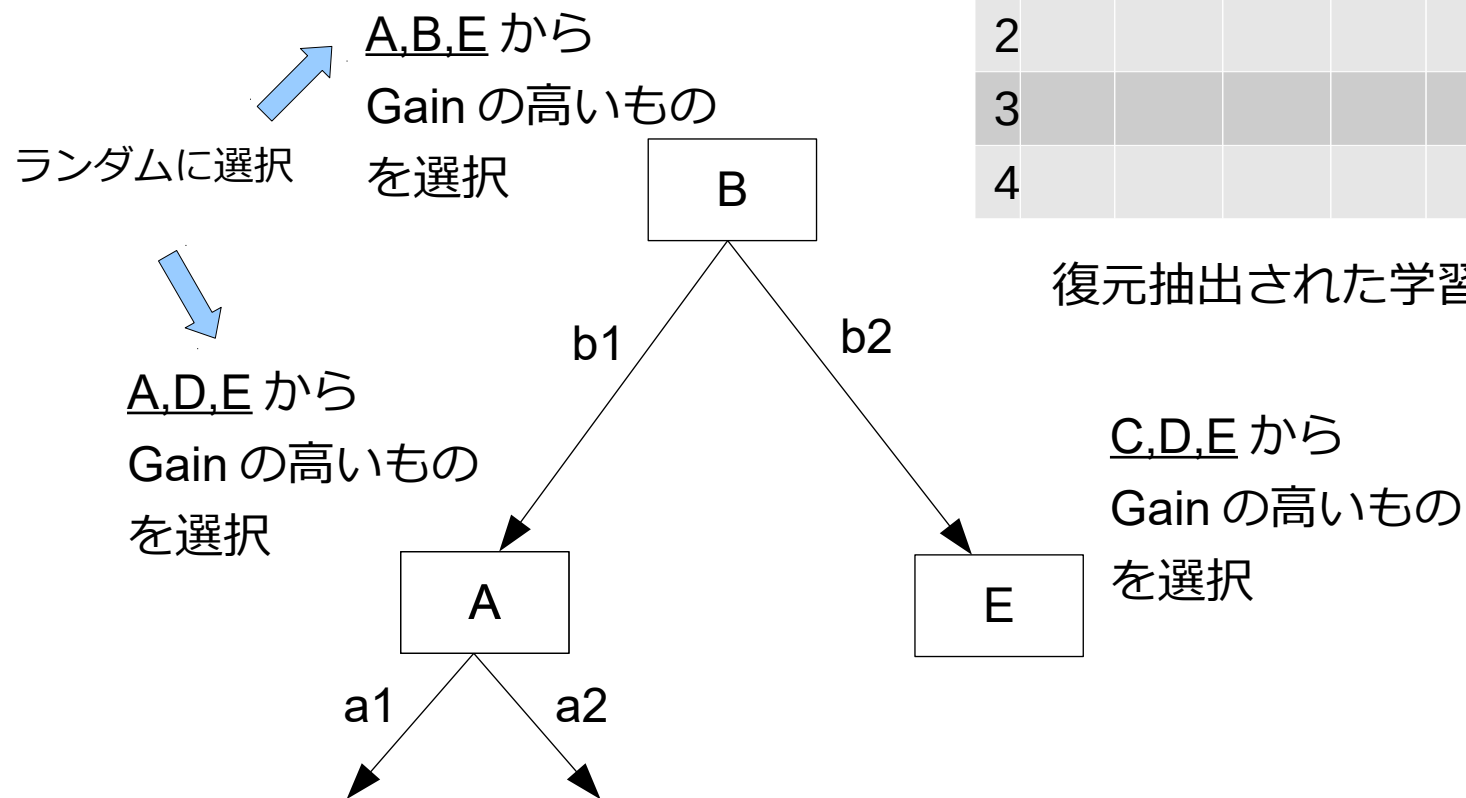


9.3 ランダムフォレスト

- アイディア
 - 識別器を作成する毎に異なる特徴を用いることで、異なった弱識別器を複数作成する
- ランダムフォレストとは
 - ランダム：乱数によって用いる特徴を制限する
 - フォレスト：森 = 複数の決定木
 - バギングのアイディアも取り入れて、学習毎にデータセットは復元抽出

9.3 ランダムフォレスト

- ランダムフォレストの学習



	A	B	C	D	E	class
1						
2						
3						
4						

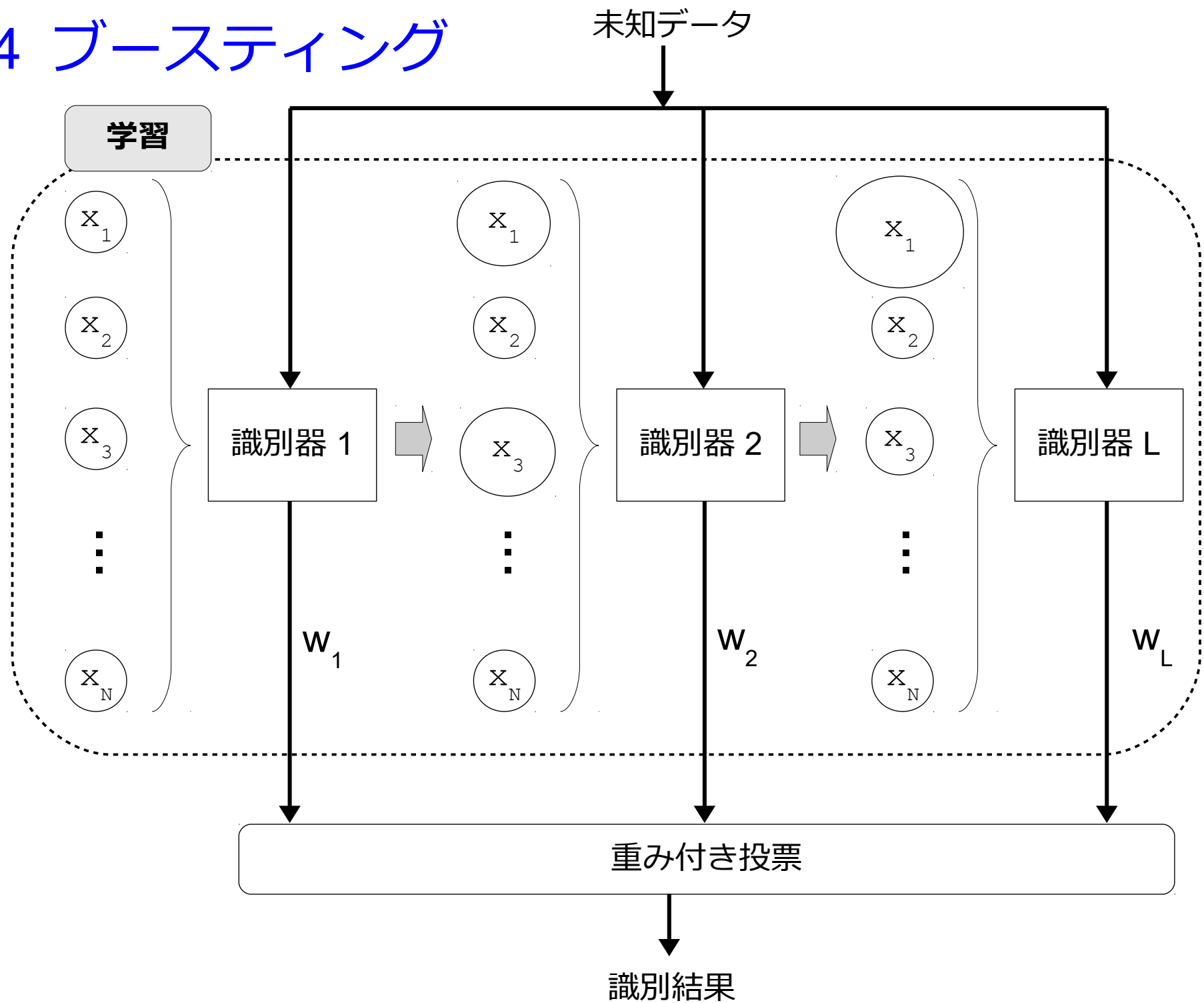
復元抽出された学習データ

これを複数回繰り返す

9.4 ブースティング

- アイディア
 - 現在の識別器が誤識別を起こすデータを正しく識別する弱識別器を逐次的に追加
 - 過学習とならないように、弱識別器として浅い決定木を用いることが多い
- Boosting とは
 - boost: (能力などを) 上昇 (向上) させる

9.4 ブースティング



12.2 系列ラベリング問題— CRF—

- 系列ラベリング問題の事例

- 形態素解析

入力	系列	で	入力	さ	れる	各	要素
出力	名詞	助詞	名詞	動詞	接尾辞	接頭辞	名詞

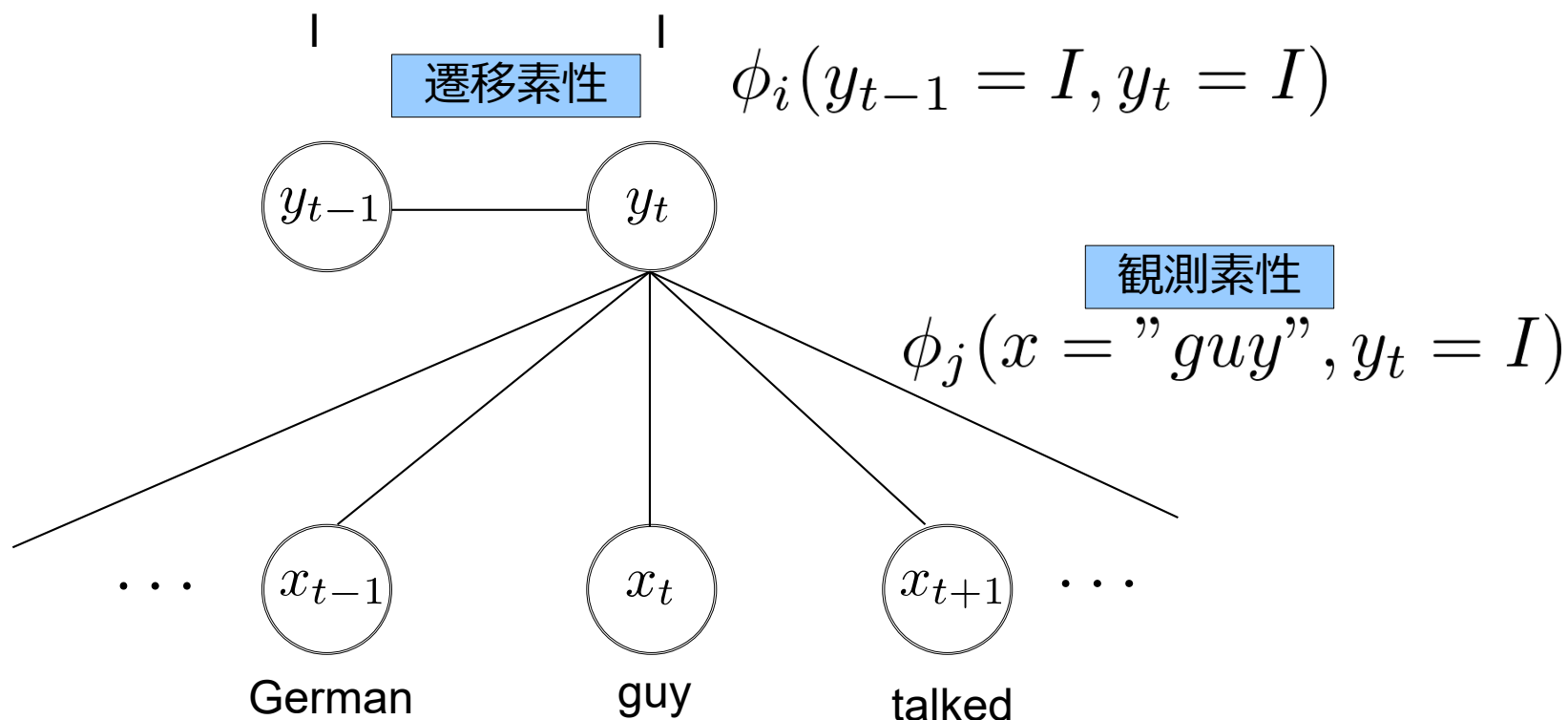
- 固有表現抽出（例：人を指す表現の抽出）

入力	Suddenly,	the	tall	German	guy	talked	to	me
出力	O	B	I	I	I	O	O	B

B: begin
I: inside
O: outside

12.2 系列ラベリング問題— CRF—

- 対数線型モデルによる系列ラベリング
 - 素性関数の導入



12.2 系列ラベリング問題— CRF—

- 対数線型モデル

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_{\mathbf{x},\mathbf{w}}} \exp(\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{y}))$$

識別モデル

あるクラスの事後確率が上がれば、他のクラスは下がる

- 出力の決定

$$\begin{aligned} \mathbf{y}^* &= \arg \max_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}) \\ &= \arg \max_{\mathbf{y}} \frac{1}{Z_{\mathbf{x},\mathbf{w}}} \exp(\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{y})) \\ &= \arg \max_{\mathbf{y}} \mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{y}) \end{aligned}$$

12.3 系列識別問題— HMM—

- 例題

- PC 操作系列による熟練度の判定

- k: キーボード、 g: マウス、 e: エラー
- 初心者の入力系列例

k e k g k e k g g k g k k e g e e k e e e g e

- 熟練者の入力系列例

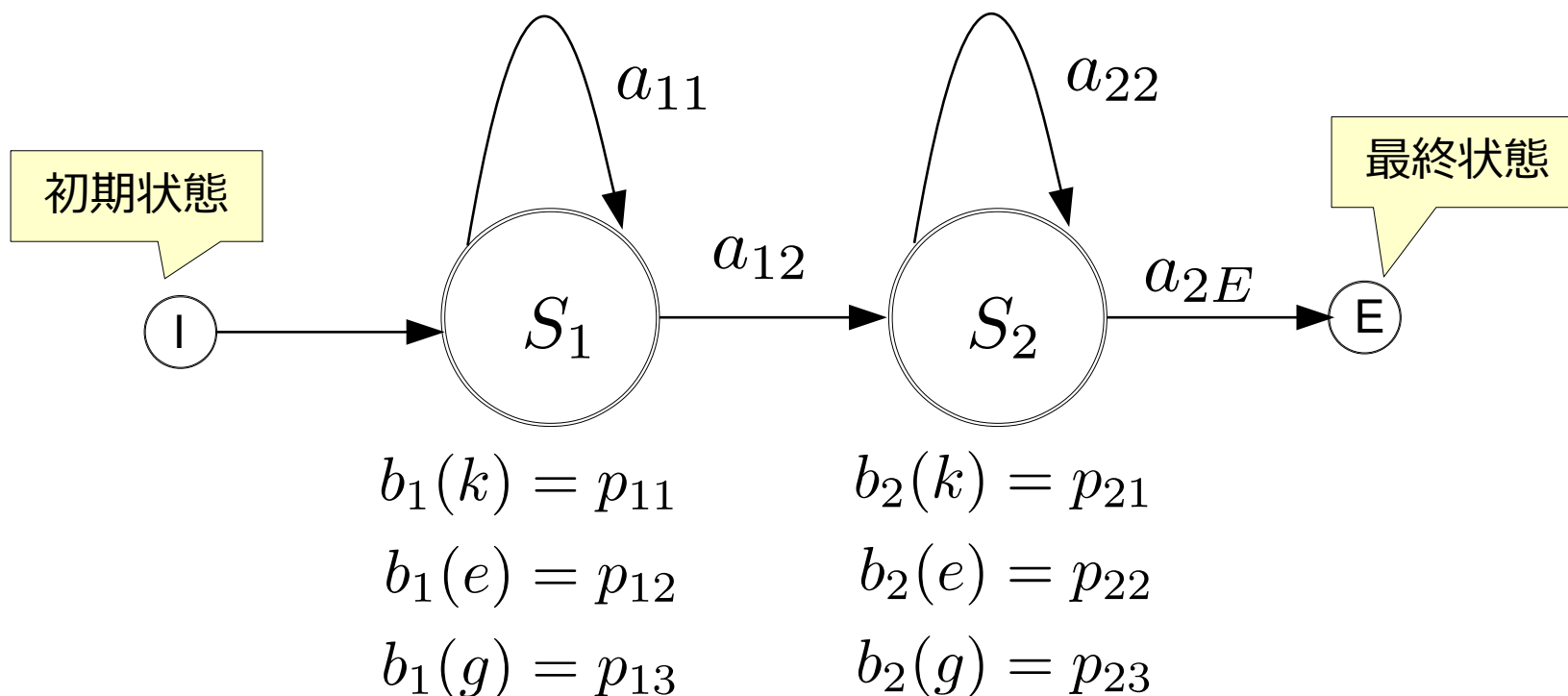
k k e k g k k k e k g k g g g e g k g

- 判定したい入力系列

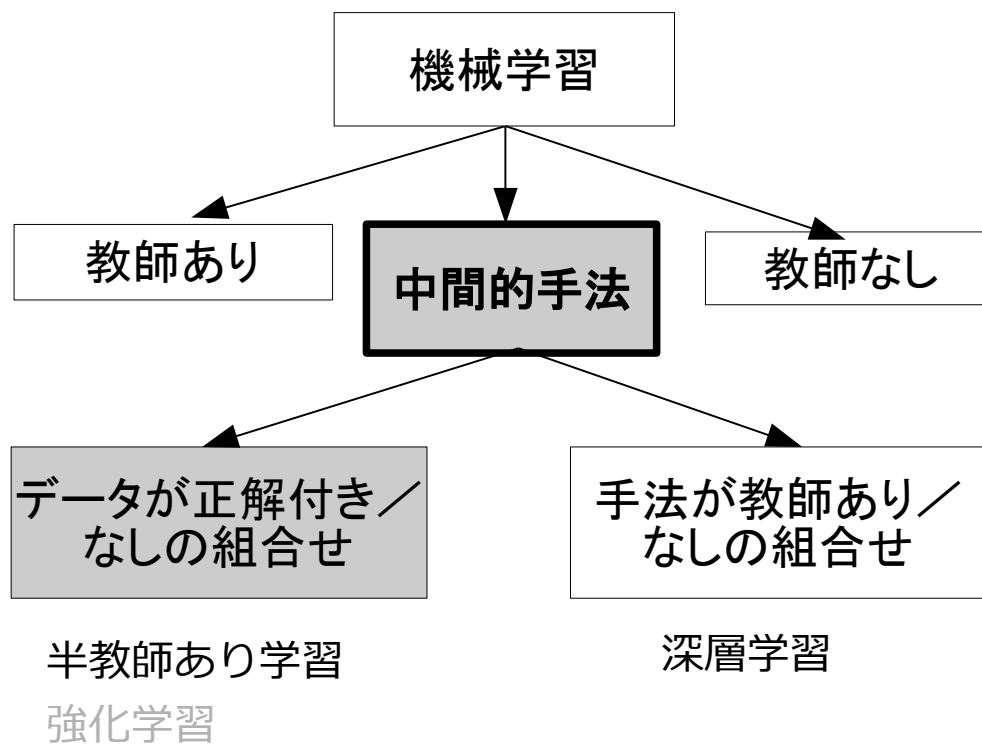
k g e k g k k g e k g e k e e k e g e k

12.3 系列識別問題—HMM—

- 不定長入力に対する尤度計算法
 - 自己遷移を持つ確率オートマトンを用いる



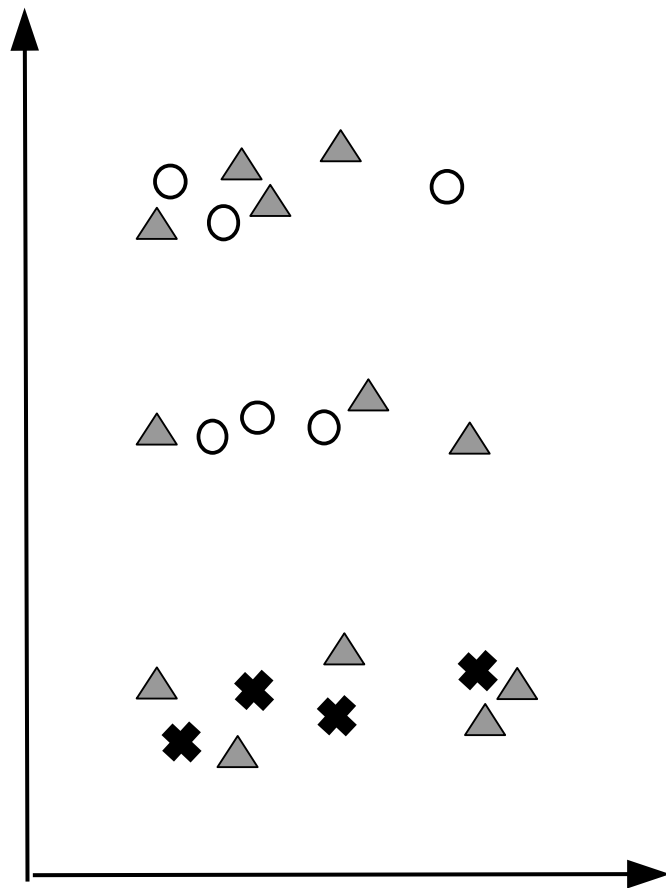
13 章 半教師あり学習



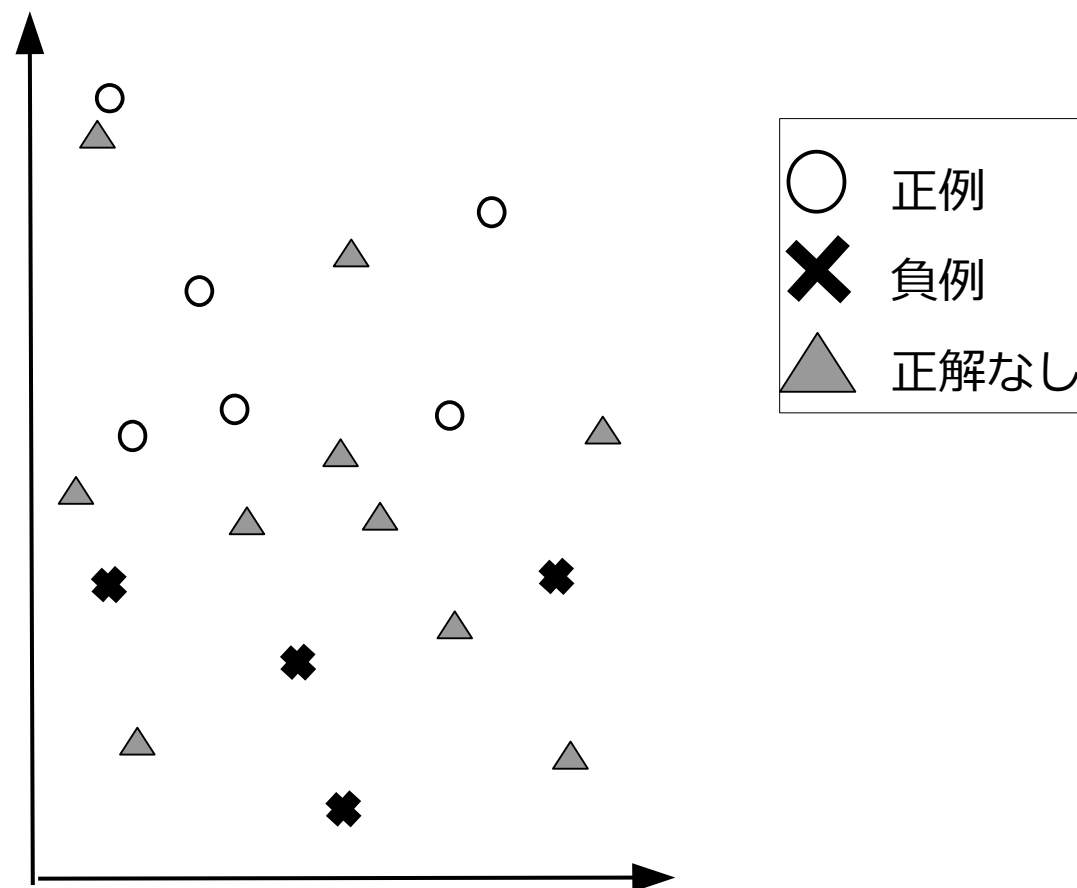
13.1 半教師あり学習とは

13.1.1 数値特徴の場合

- 半教師あり学習に適した数値特徴データの性質



半教師あり学習に適するデータ



半教師あり学習に適さないデータ

13.1.2 カテゴリカル特徴の場合

- オーバーラップ
 - 文書からの評判分析の例

Positive ○

... よかった。 ..
...
高性能 ..
... 満足

?

...
...
高性能 ..
... 満足 ..
....

?

.....
...
高性能 ..
...
... よかった。

Negative ✕

... 壊れた。 ..
...
不満 ..
...
... 買わない

?

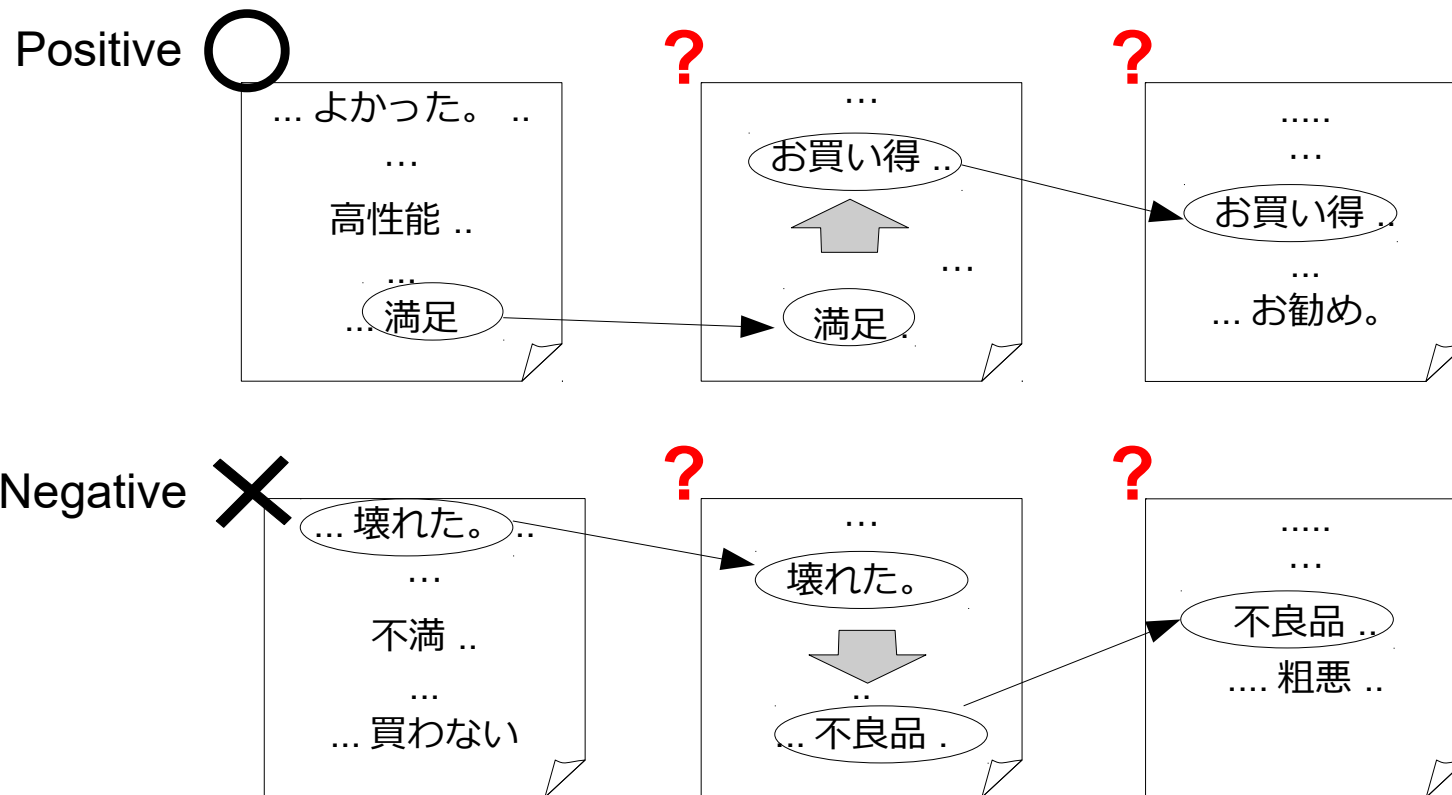
...
...
壊れた。 ..
... 買わない ..
....

?

.....
...
不満 ..
...
... 買わない

13.1.2 カテゴリカル特徴の場合

- 特徴の伝播



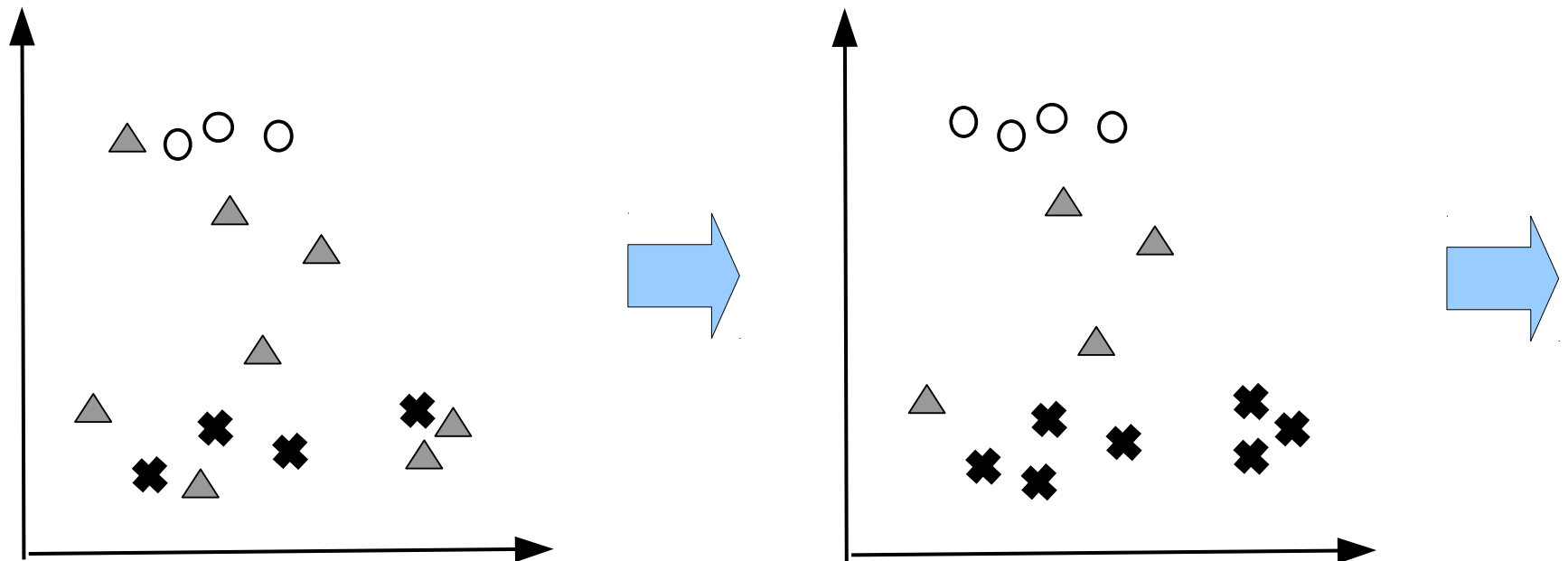
13.1.3 半教師あり学習のアルゴリズム

- 半教師あり学習の基本的な考え方
 - 正解付きデータで識別器を作成
 - 正解なしデータで識別器のパラメータを調整
- 識別器に対する要求
 - 確信度の出力：正解なしデータに対する出力を信用するかどうかの判定に必要

13.2 自己学習

- 自己学習のアルゴリズム

1. 正解付きデータで初期識別器を作成
2. 正解なしデータの識別結果のうち、確信度の高いものを、正解付きデータとみなす
3. 新しい正解付きデータで、識別器を学習
4. 2, 3 を繰り返す



ラベル伝搬法

- ラベル伝搬法の考え方
 - 特徴空間上のデータをノードとみなし、類似度に基づいたグラフ構造を構築する
 - 近くのノードは同じクラスになりやすいという仮定で、正解なしデータの予測を行う
 - 評価関数（最小化）

$$J(\mathbf{f}) = \sum_{i=1}^l (y_i - f_i)^2 + \lambda \sum_{i < j} w_{ij} (f_i - f_j)^2$$

予測値と正解
ラベルを近づける

隣接ノードの
予測値を近づける

f_i : i 番目のノードの予測値

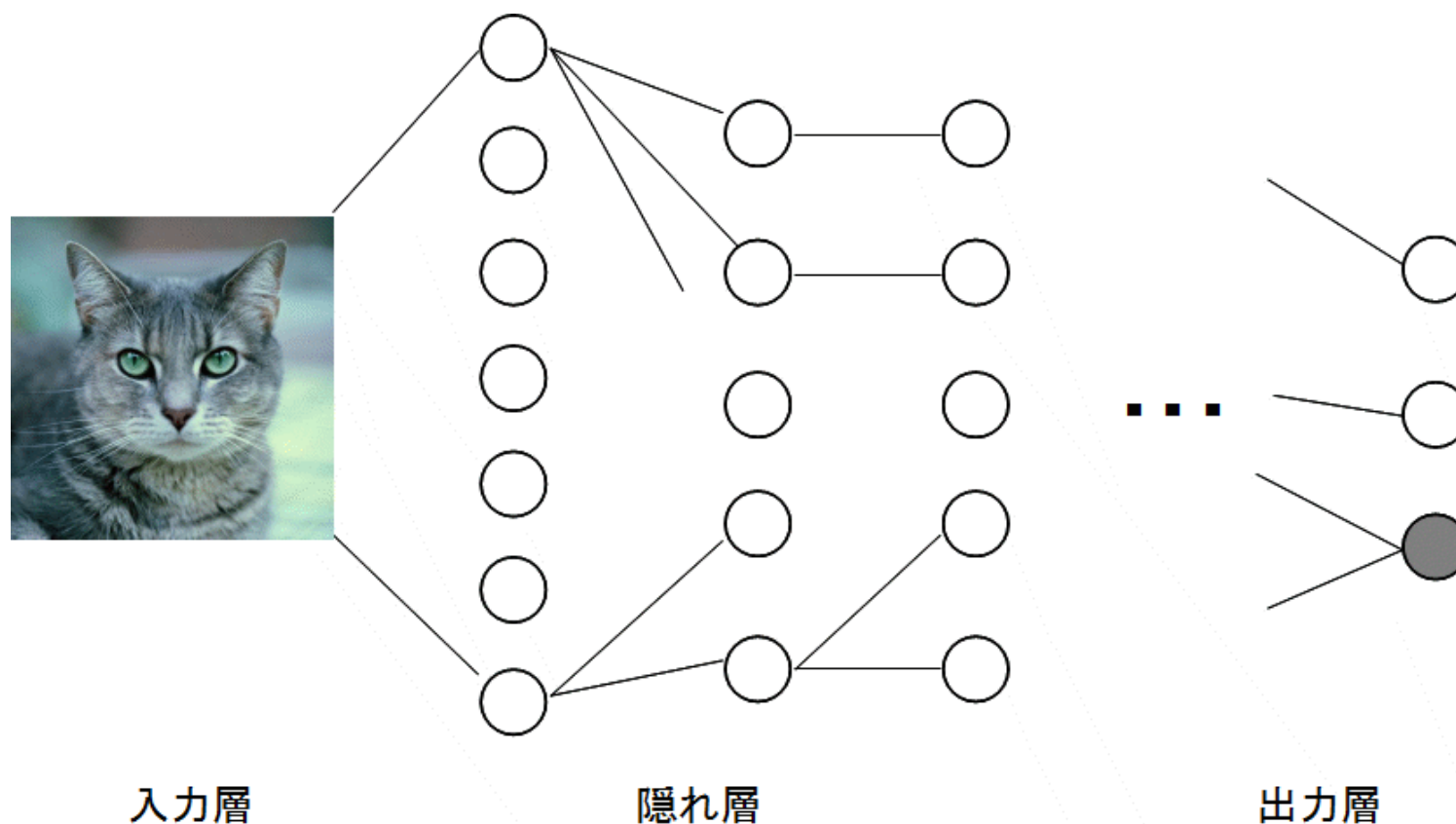
y_i : i 番目のノードの正解ラベル $\{-1, 0, 1\}$

w_{ij} : i 番目のノードと j 番目のノードの結合の有無

15 章 深層学習

15.1 深層学習とは

- 深層学習の定義のひとつ
 - 表現学習：抽出する特徴も学習する

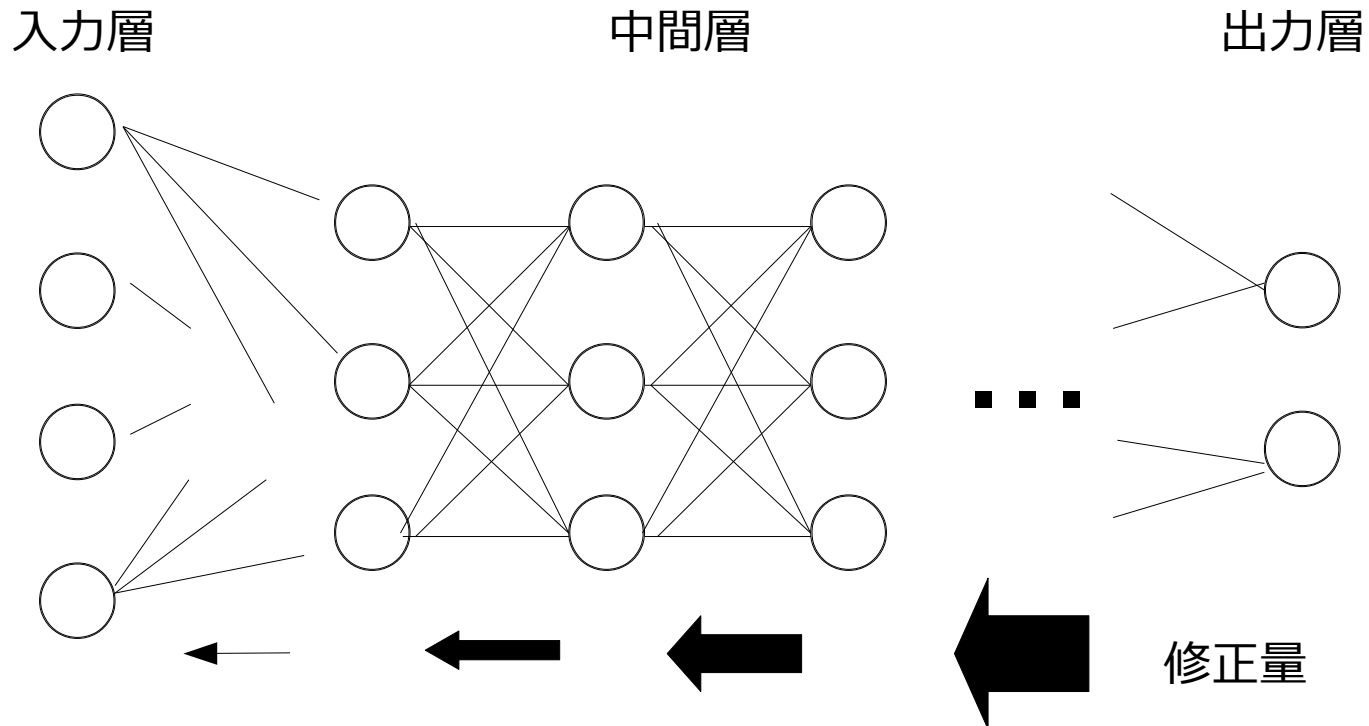


15.2 多階層ニューラルネットワークの学習

- 多階層における誤差逆伝播法の問題点
 - 修正量が消失／発散する

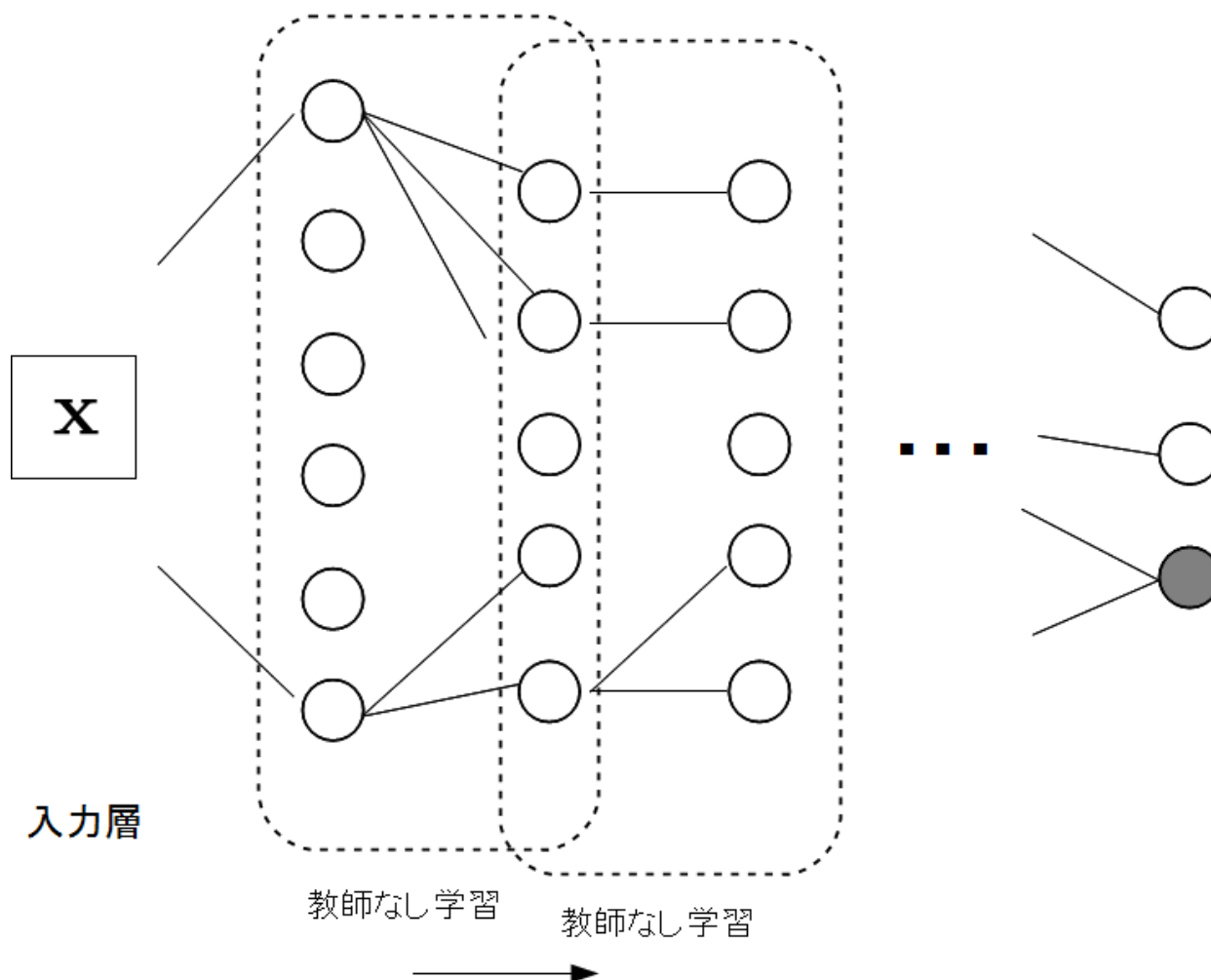
順方向：非線形

逆方向：線形



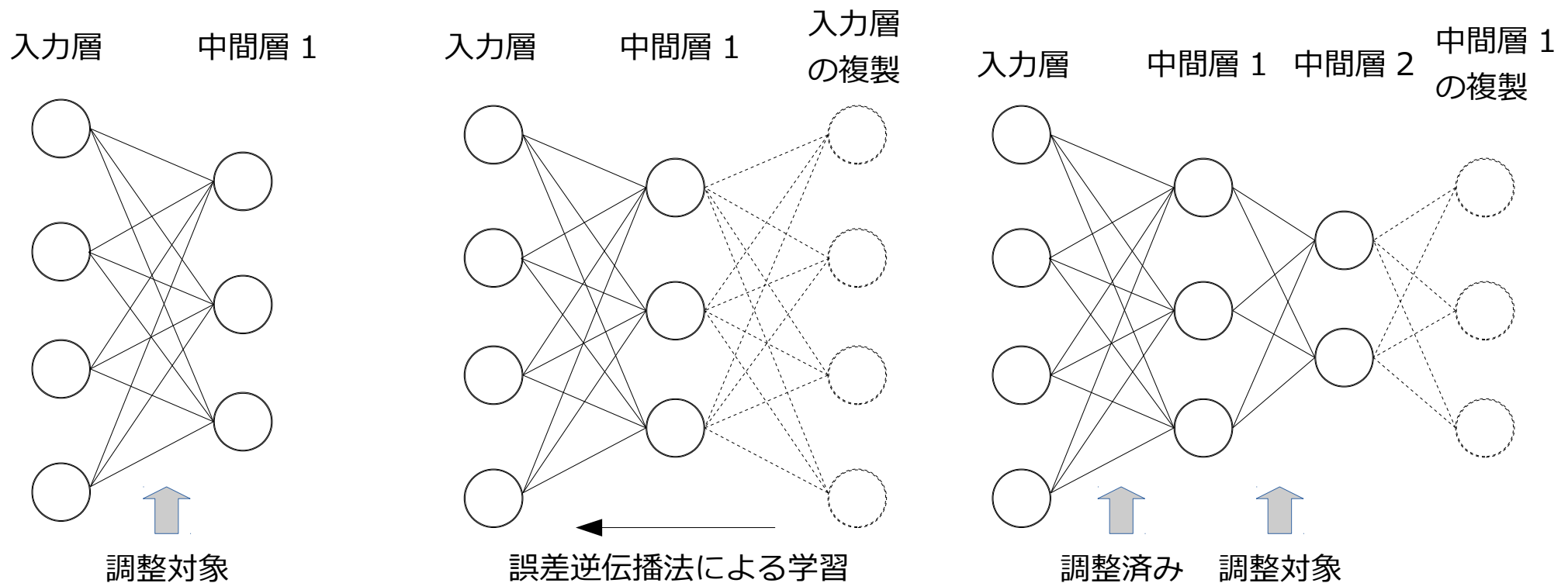
15.2 多階層ニューラルネットワークの学習

- 事前学習法のアイデア
 - 深層学習における初期パラメータ学習



15.3 Autoencoder

- autoencoder のアイデア：自己写像を行う



(a) 事前調整対象の重み

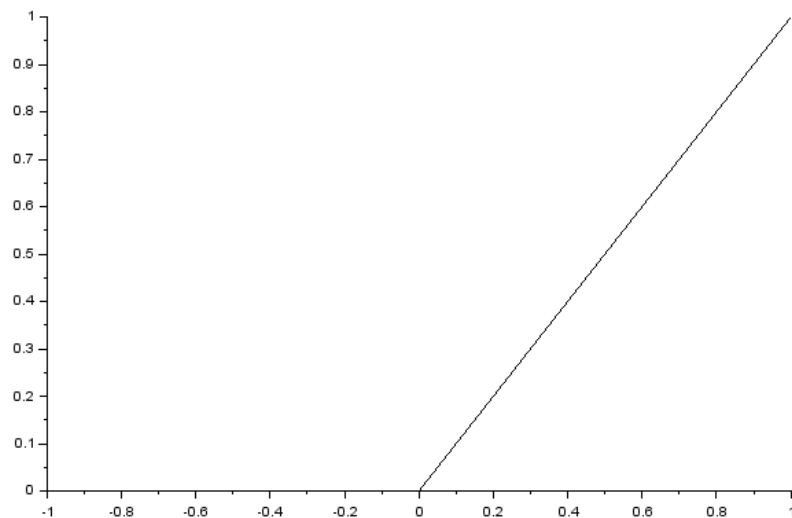
(b) オートエンコーダによる
復元学習

(c) 1 階層上の事前調整

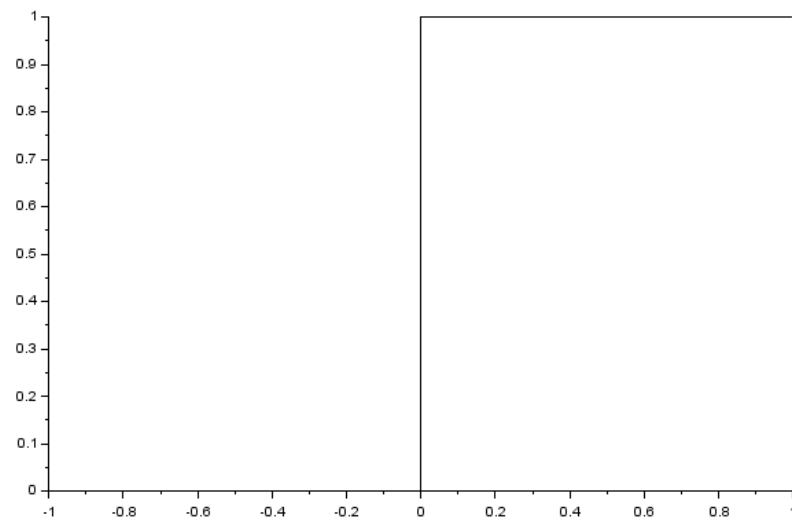
多階層学習における工夫

- 活性化関数を rectified linear 関数に ➡ RELU

$$f(x) = \max(0, x)$$



(a) rectified linear 関数

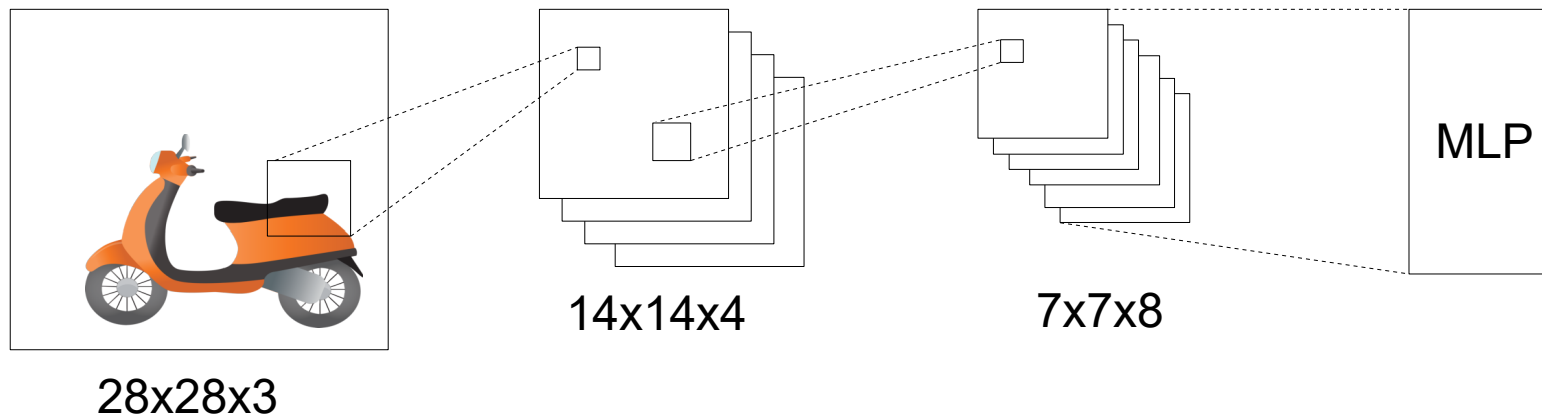


(b) (a) の導関数

- RELU の利点
 - 誤差消失が起こりにくい
 - 0 を出力するユニットが多くなる

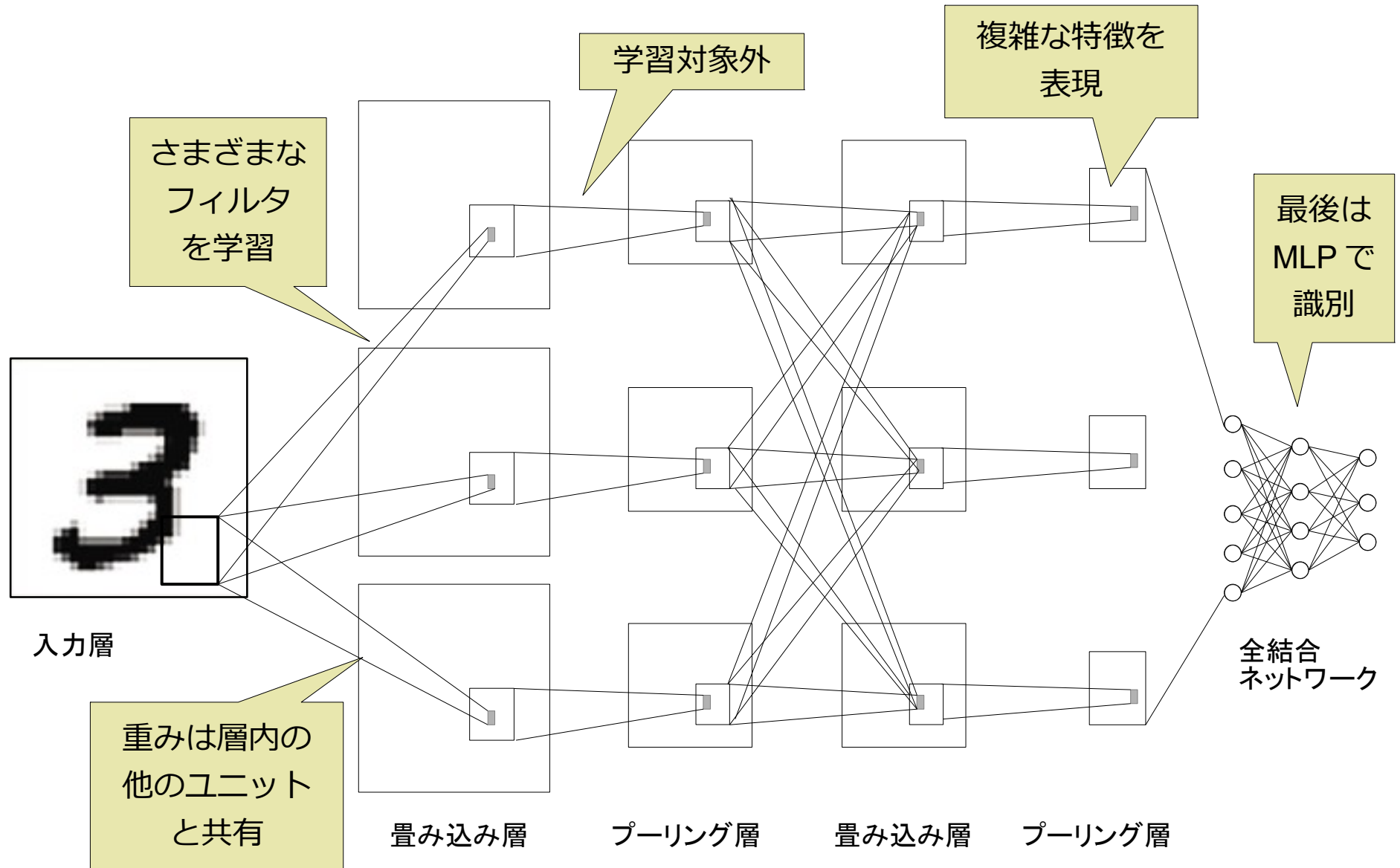
畳み込みニューラルネットワーク

- 畳み込みニューラルネットワークの構造
 - 畳み込み層とプーリング層を交互に重ねる
 - 畳み込み層はフィルタの画素数・ずらす画素数・チャンネル数の情報からなる
 - 最後は通常の MLP (Relu+softmax)



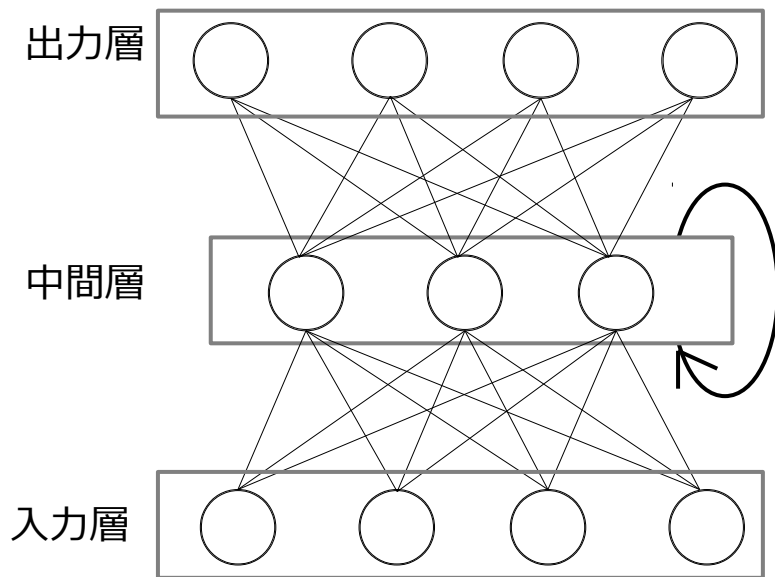
畳み込みニューラルネットワーク

- 畳み込みニューラルネットワークにおける学習

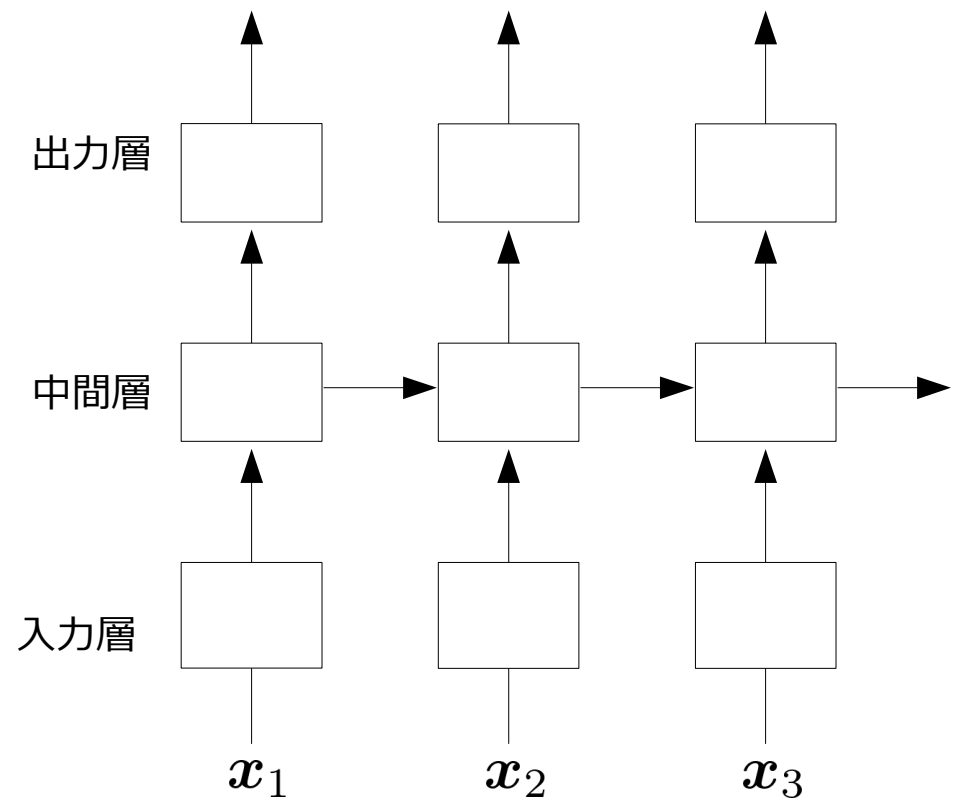


リカレントニューラルネットワーク

- 時系列信号の認識や自然言語処理に適する



(a) リカレントニューラルネットワーク

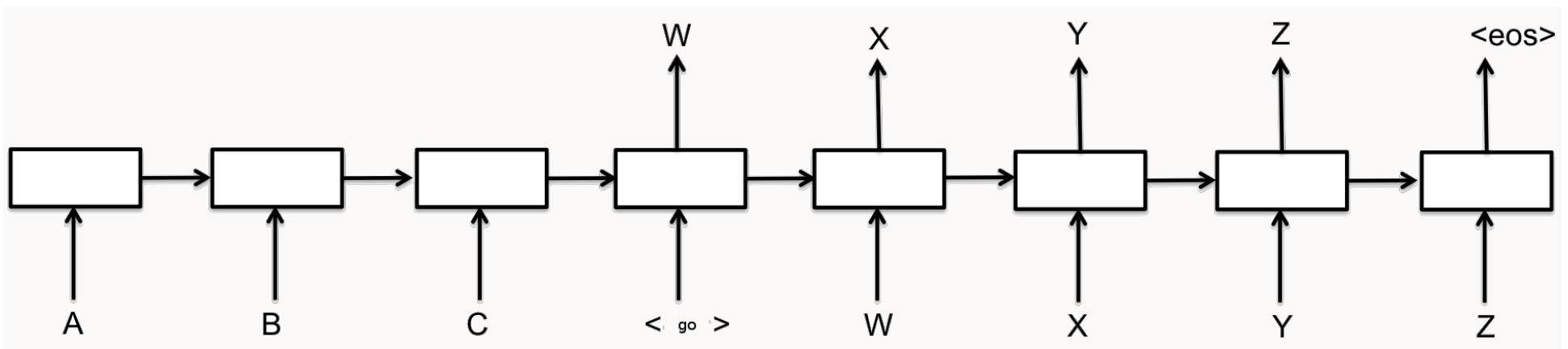


(b) 帰還路を時間方向に展開

リカレントニューラルネットワーク

- Encoder-Decoder

- 入力の内容をひとつの表現にまとめて、そこから出力を生成

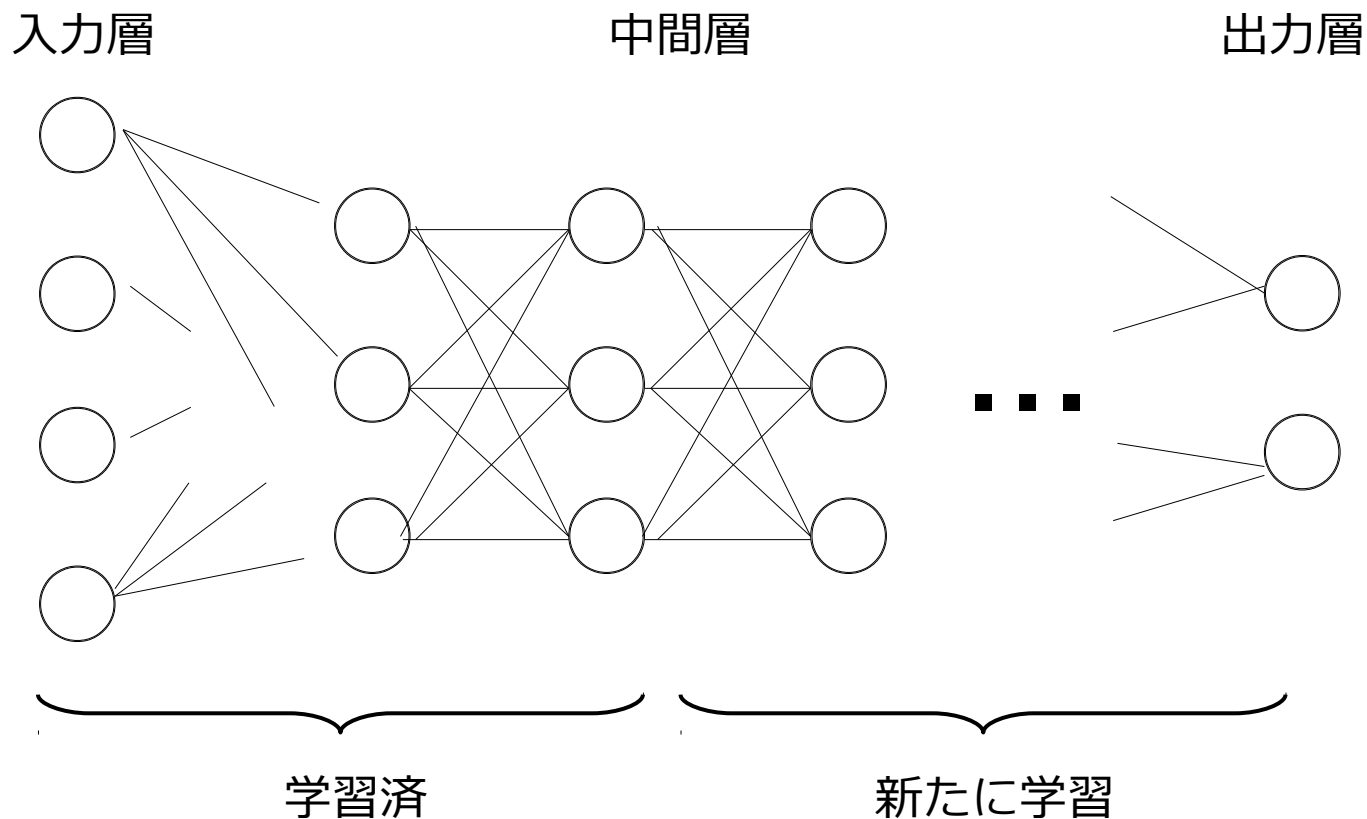


arXiv:1406.1078

深層学習の利点

- 転移学習

- 大量のデータで学習済のネットワークを用いて、少量のデータで再学習



Section4 のまとめ

- アンサンブル学習
 - 問題によっては、深層学習よりも高い性能を示すことがある
- 半教師あり学習
 - 人手によるラベリング済データの入手が難しい問題に適用可能
- 系列データの学習
 - 自然言語処理などに応用可能
- 深層学習
 - 音声・画像・自然言語など局所依存性のあるデータの識別に適する
 - 転移学習によって少量データでの学習が可能な問題もある