

# 月震の震源域を推定する 機械学習手法の検討

菊池 栞<sup>1</sup> 山田竜平<sup>2</sup> 山本幸生<sup>3</sup> 横山昌平<sup>4</sup> 石川博<sup>5</sup>

<sup>1</sup>首都大学東京システムデザイン学部 4年

<sup>2</sup>国立天文台

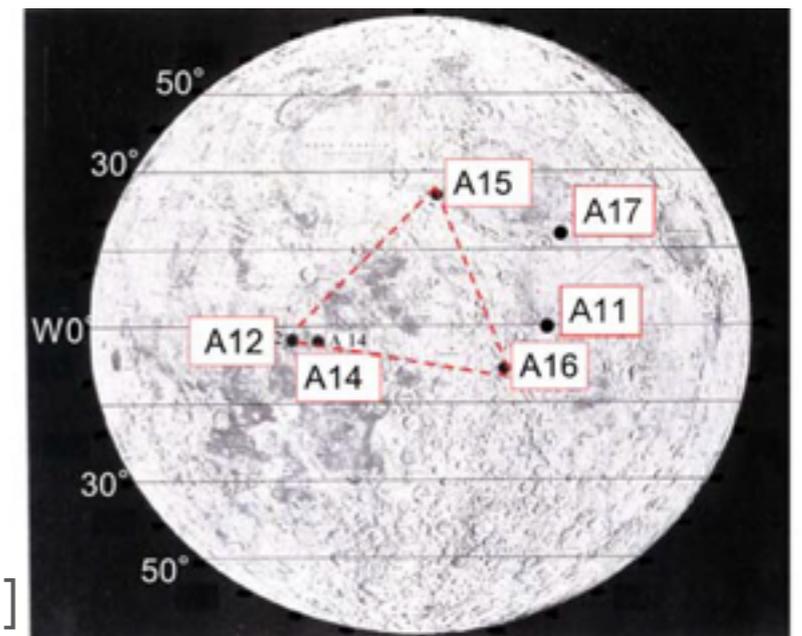
<sup>3</sup>ISAS/JAXA

<sup>4</sup>静岡大学情報学部

<sup>5</sup>首都大学東京システムデザイン研究科

## はじめに

- **月震の震源の性質**
  - 同じ震源の月震は，波形が似ている
  - 3点で観測できれば震源の特定が可能
- **震源が特定されていない月震が多数存在**
  - 3点で観測できるイベントが少ない
    - 月震波の振幅が小さいため
  - 月震波にノイズが多い



# 目的

- **月震の分類に適した機械学習手法の検討**
  - 機械学習を用いて，従来手法で震源が分類できていないイベントの分類を目指す
- **前発表との違い**
  - 前発表  
今までの分類の検証と細分化
  - 本研究  
今まで分類された月震を用いて，未分類の月震を分類することを目指す

# クラス分類

分類済みのデータ

- A1
- A6
- A8
- A18
- A20

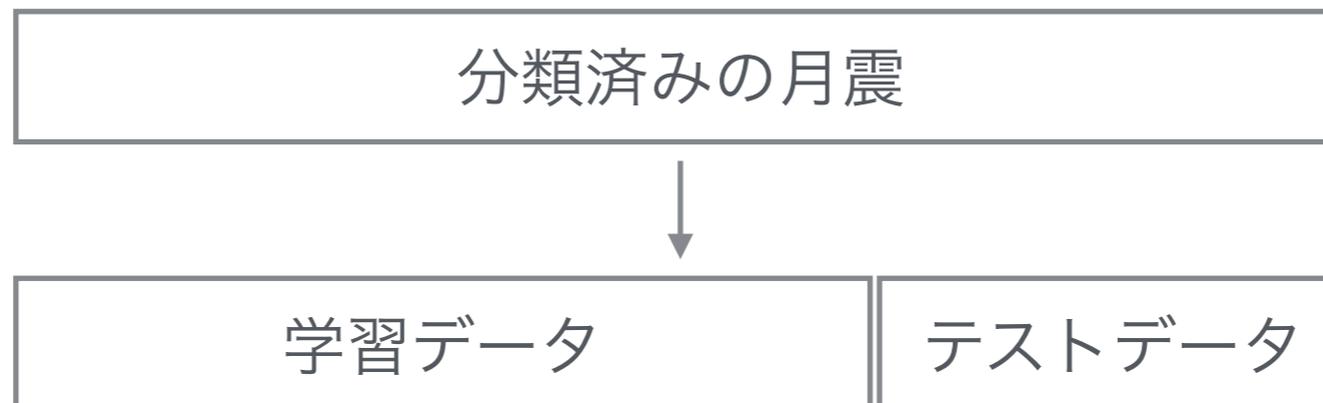


未分類のデータ



# 手法の概要

- **未分類の月震を分類するには、分類済みの月震の分類を再現できなければいけない**
  - 分類済みの月震を用いてF値が1に近い分類器を作ることを目指す



学習データで分類器を作り、  
テストデータで性能を評価する

# データセット

- データセット

Apollo12のデータを使用

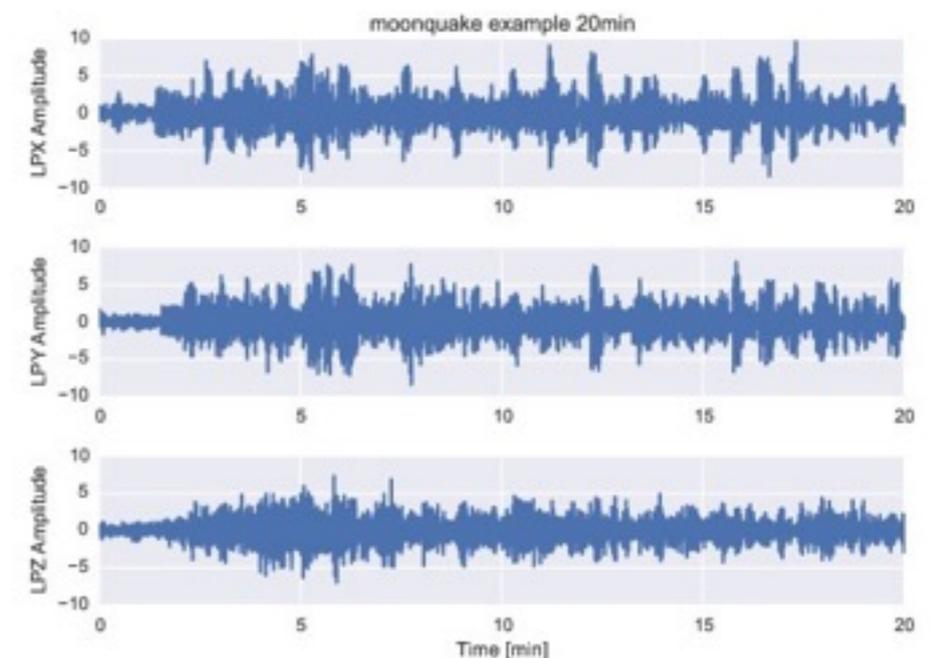
イベント数が30以上ある5震源を選択

月震の3成分 (LPX, LPY, LPZ)をそれぞれ使用

波形からPSDを計算 (前発表と同様)

使用した震源ごとのイベント数

震源名	A1	A6	A8	A18	A20	合計
イベント数	173	32	39	36	48	331



# 比較方法

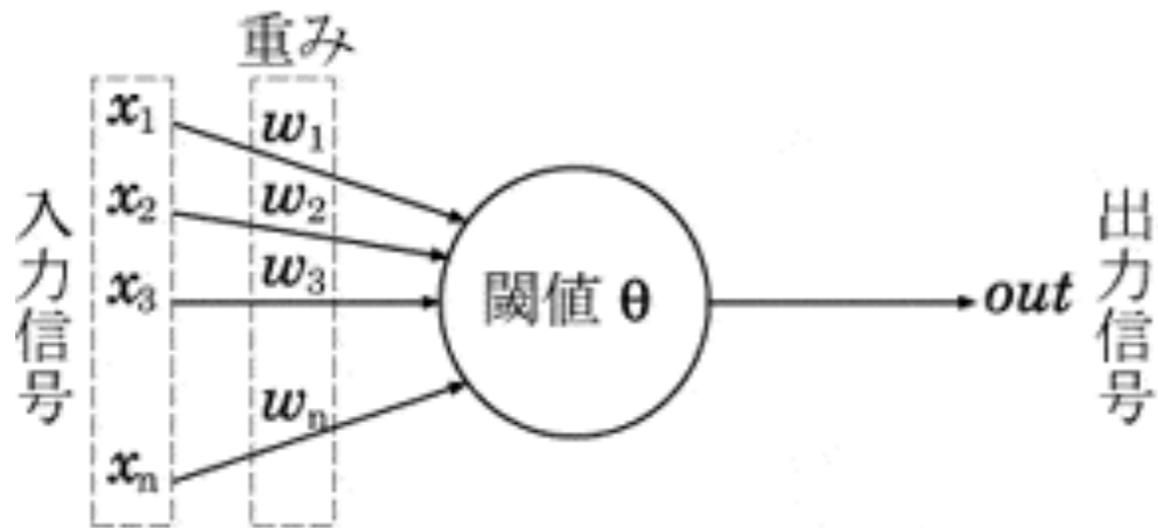
- イベント長と， PSDのサンプリング数の各条件で， 分類手法を比較する
- 月震分類に適した条件 (イベント長, サンプリング数)を検討する
  - イベント長 (min) : {5, 10, 15, 20, 25, 30}
  - サンプリング数 : {256, 512, 1024, 2048}
- 比較する分類手法
  - Neural Network (NN)
  - ロジスティック回帰
  - Random Forest
  - Support Vector Machine-Linear (SVM-Linear)
  - Support Vector Machine-RBF (SVM-RBF)
  - AdaBoost [弱分類器 : 決定木]

# Neural Network (NN)

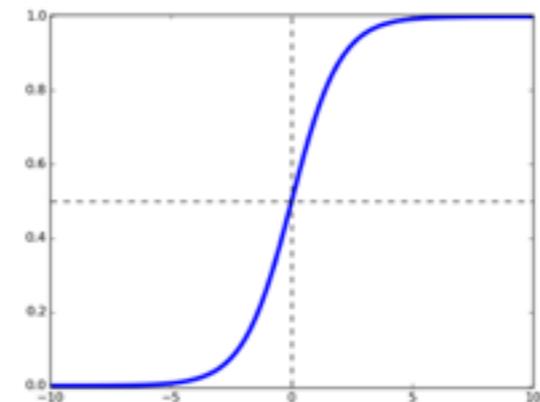
- **多くの分野で応用, 注目されている手法**
  - 画像認識, 音声認識, 言語処理など
  - AI分野での活用も始まっている
  - Deep Learning (深層学習)もこの一種
- **中間層で有効な特徴量を抽出できる**
  - その特徴量を抽出できる理由は解明されていない

# Neural Network (NN)

## ニューロンモデル



## 活性化関数



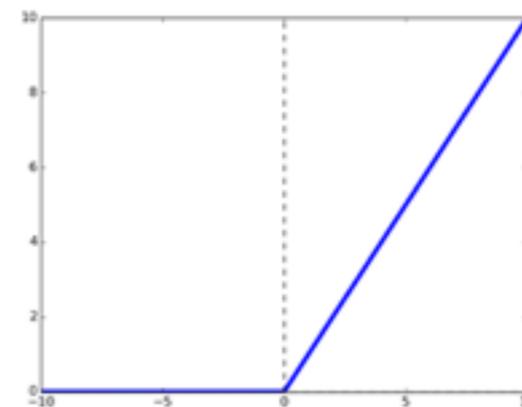
sigmoid

$$net = m_1x_1 + m_2x_2 + \dots + m_nx_n$$

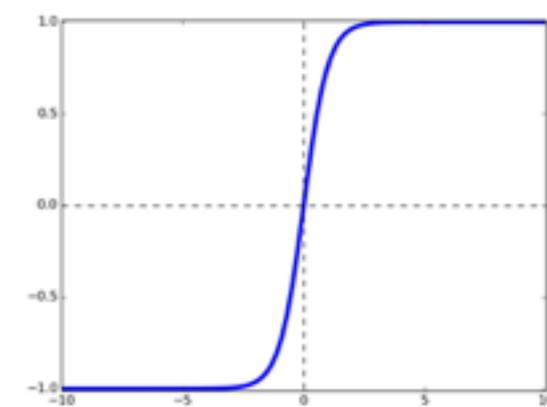
$$out = f(net - \theta)$$

$f$  : 活性化関数

最初, 重みはランダムで決定する



ReLU

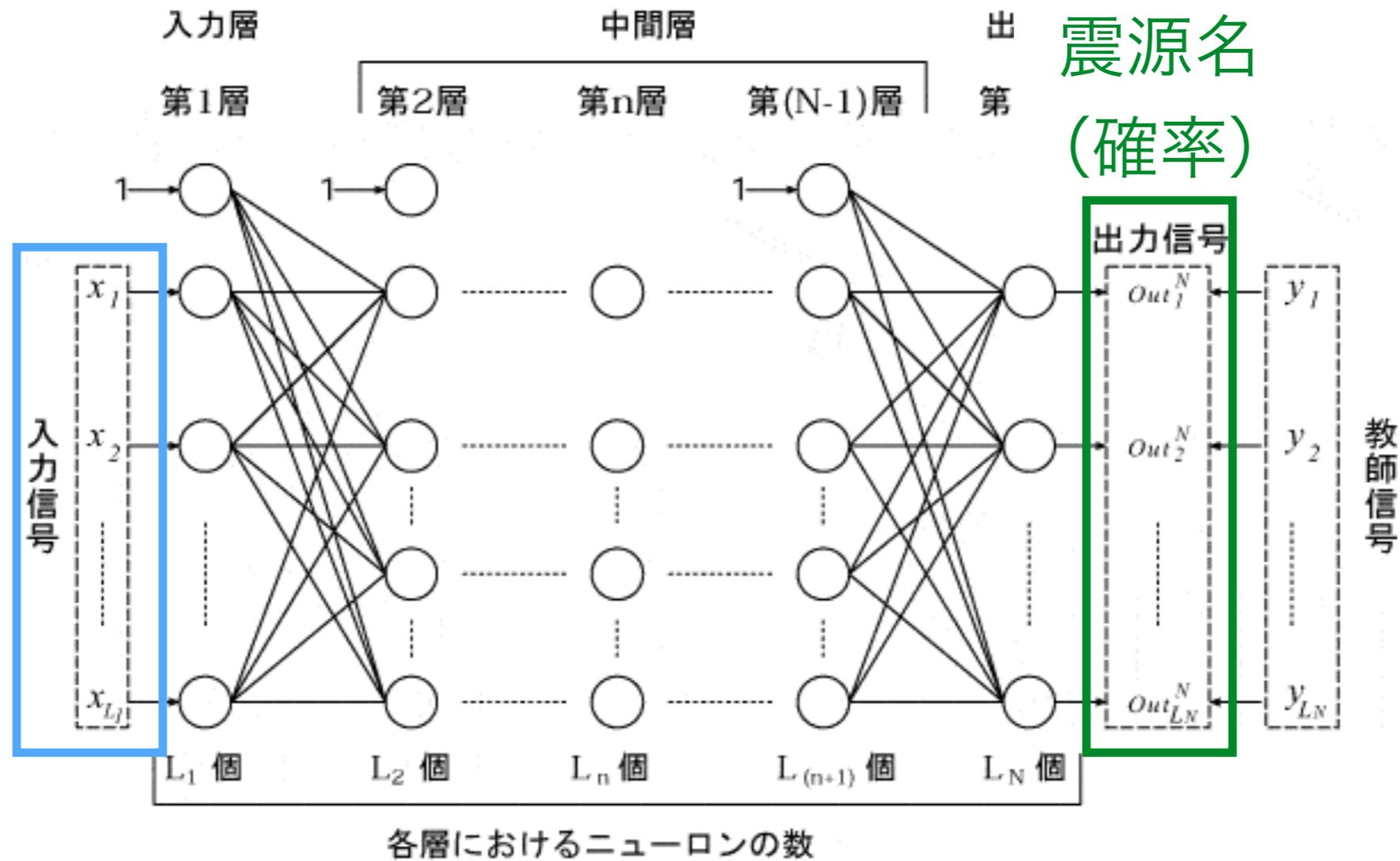


tanh

# Neural Network (NN)

学習

月震波  
(PSD)

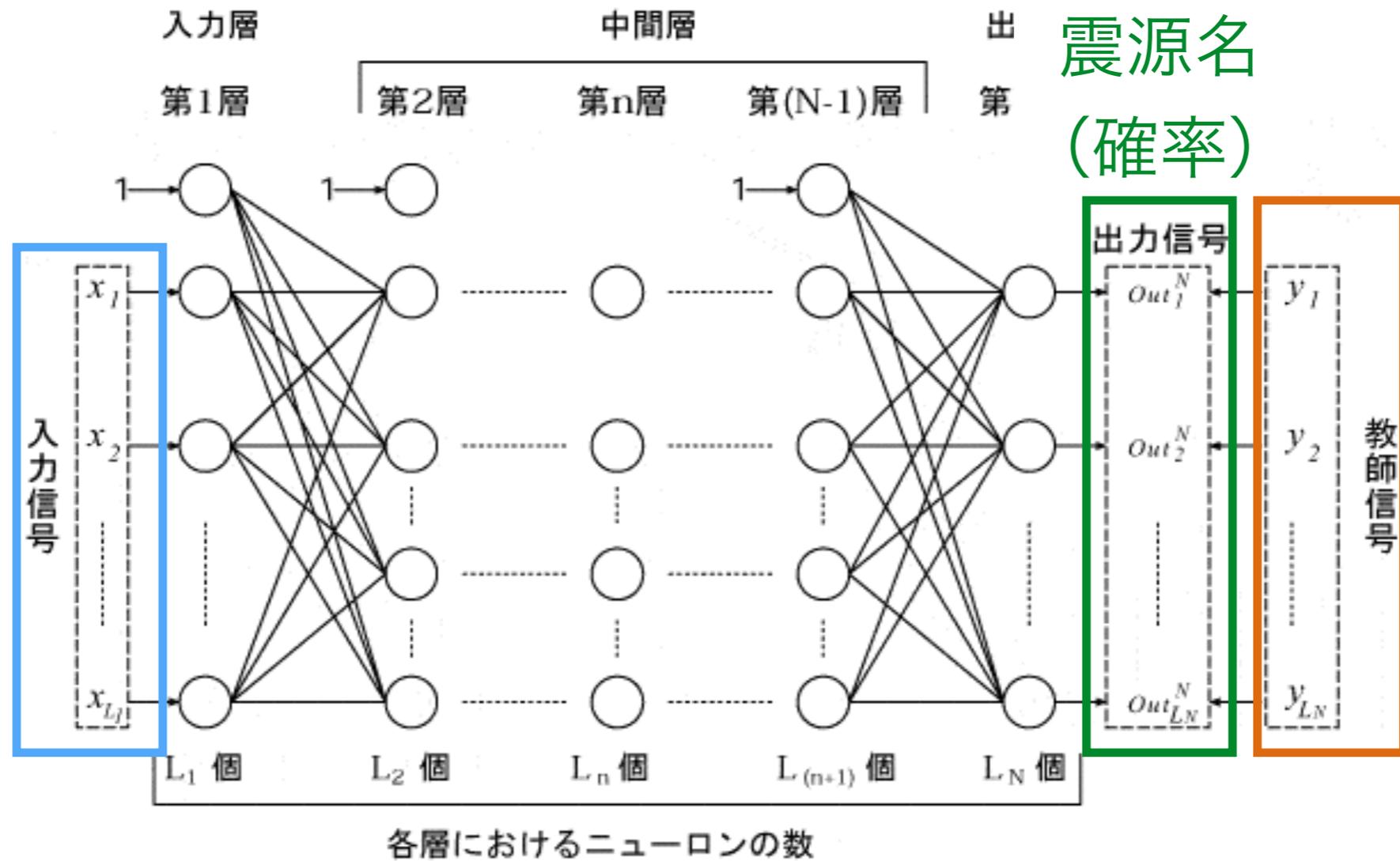


入力信号を用いて，計算を行いながら，  
出力信号 (震源名) を出力

# Neural Network (NN)

学習

月震波  
(PSD)

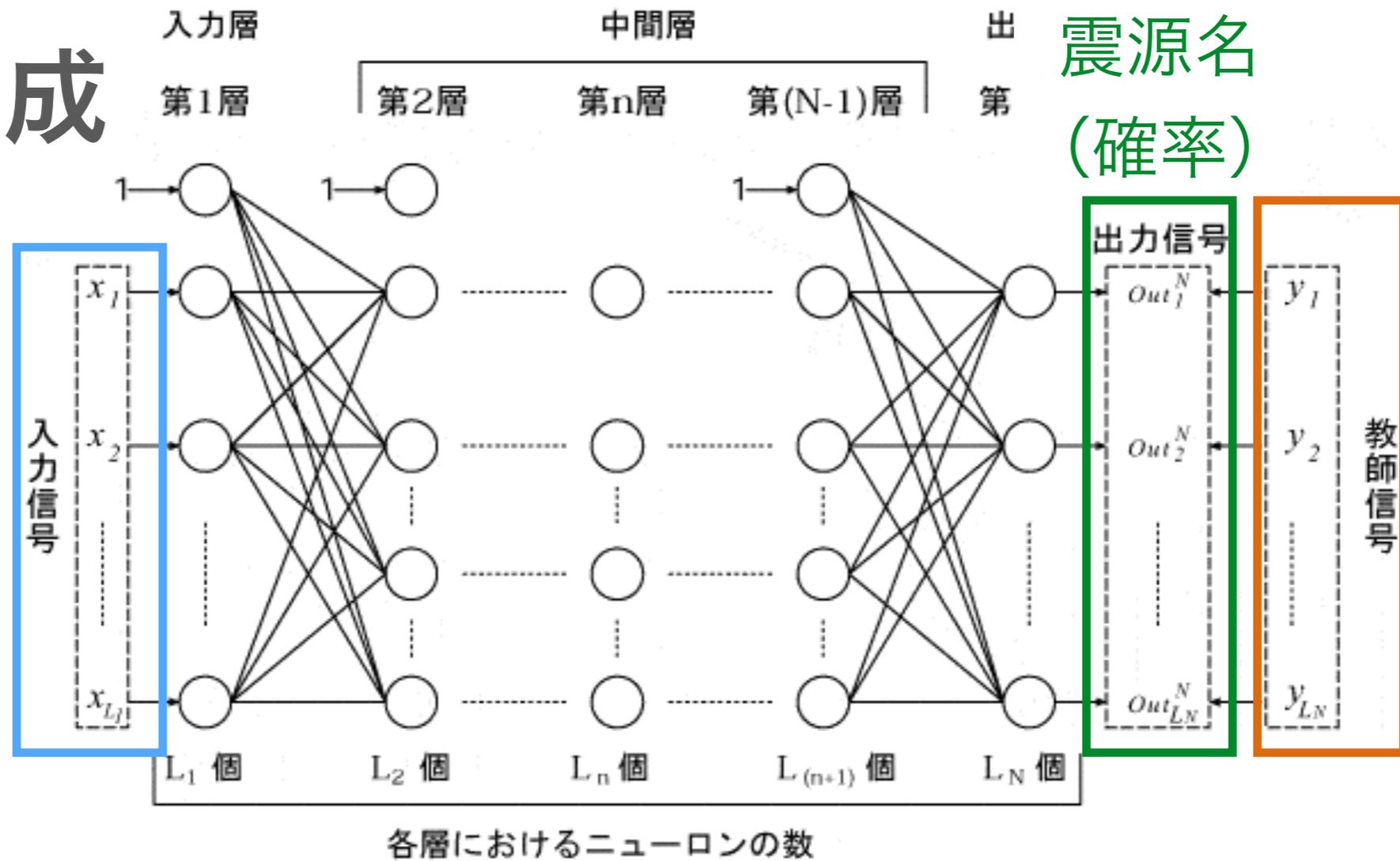


出力信号と教師信号の誤差を計算し、  
その誤差を小さくするように重みの更新を行う

# Neural Network (NN)

## 分類器完成

月震波  
(PSD)

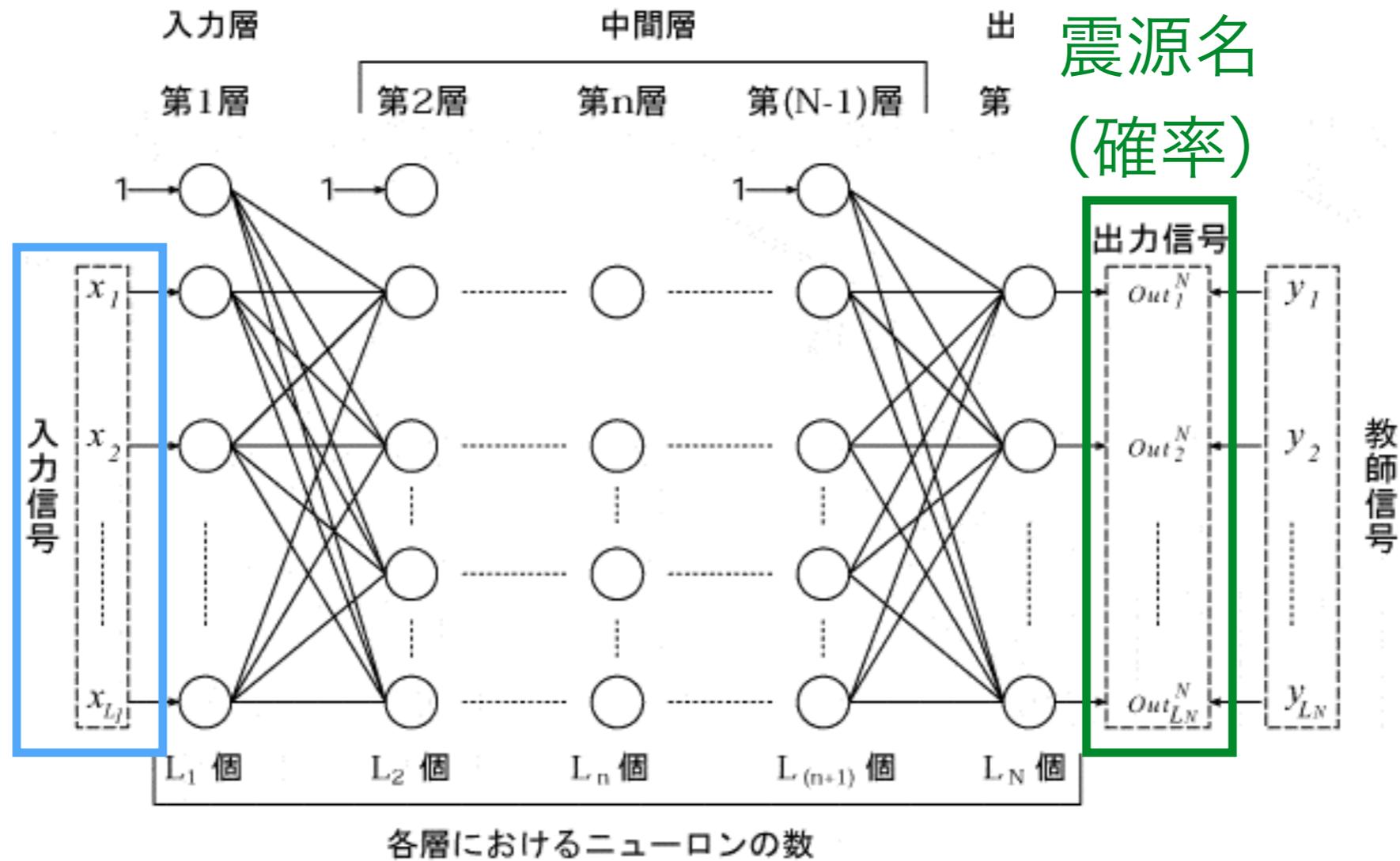


これらを繰り返し行うことで、月震波を入力すると、それがどの震源なのかを表現する関数を作ることができる

# Neural Network (NN)

分類

月震波  
(PSD)



分類したい月震波を入力し、

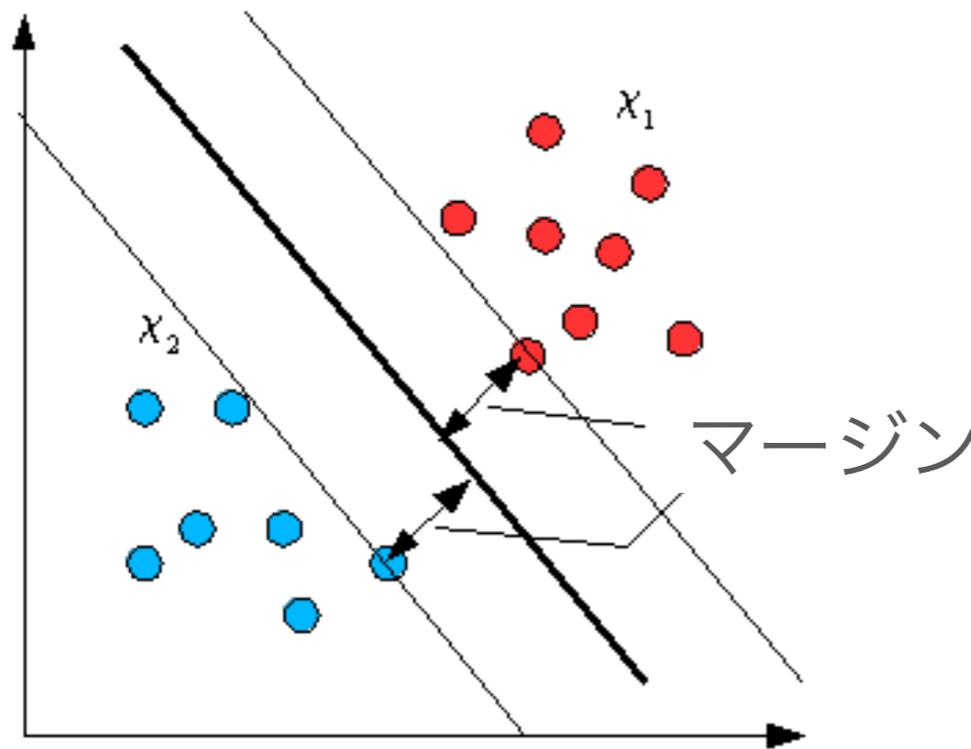
出力された確率が最も高い震源を、予測された震源とする

# Support Vector Machine (SVM)

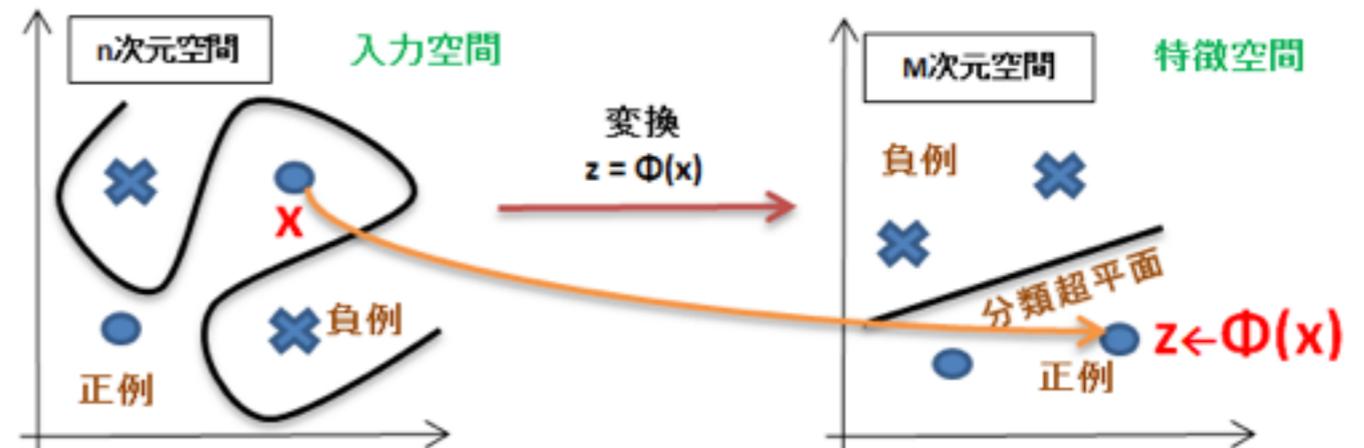
- **教師あり学習で分類すると言ったらこの手法**
  - 精度が高く, チューニングが簡単で使いやすい
  - 次元数が多くても精度が高い
- **固定長であるデータに用いられる**
  - 画像認識, 文字認識など

# Support Vector Machine (SVM)

マージンを最大化し，汎化性能を高める



線形カーネル  
(SVM-Linear)  
直線で分類



RBFカーネル  
(SVM-RBF)  
曲線で分類

分類する線を引いて，クラスを決定する

# 結果

NNを用いて各条件で分類した結果のF値 LPZを使用

サンプリング数	イベント長 (min)					
	5	10	15	20	25	30
256	0.81	0.82	0.83	0.83	0.81	0.84
512	0.83	0.94	0.92	0.91	0.88	0.87
1024	0.64	0.93	0.89	0.93	0.9	0.87
2048	0.95	0.93	0.99	0.99	0.96	0.94

このような条件ごとに、各手法、各月震成分で分類を行う

# 手法と月震成分の比較

それぞれのサンプリング数とイベント長で分類した際の各手法のF値

	平均			最大		
	LPX	LPY	LPZ	LPX	LPY	LPZ
Neural Network	0.78	0.81	0.88	0.87	0.90	0.99
ロジスティック回帰	0.33	0.75	0.81	0.33	0.86	0.92
Random Forest	0.55	0.56	0.56	0.71	0.67	0.77
SVM-Linear	0.70	0.79	0.84	0.79	0.88	0.95
SVM-RBF	0.70	0.71	0.76	0.80	0.83	0.92
AdaBoost	0.38	0.39	0.44	0.49	0.55	0.68

NNのF値が最も高い結果になった

LPZ成分のデータを用いた時、分類性能が一番高かった

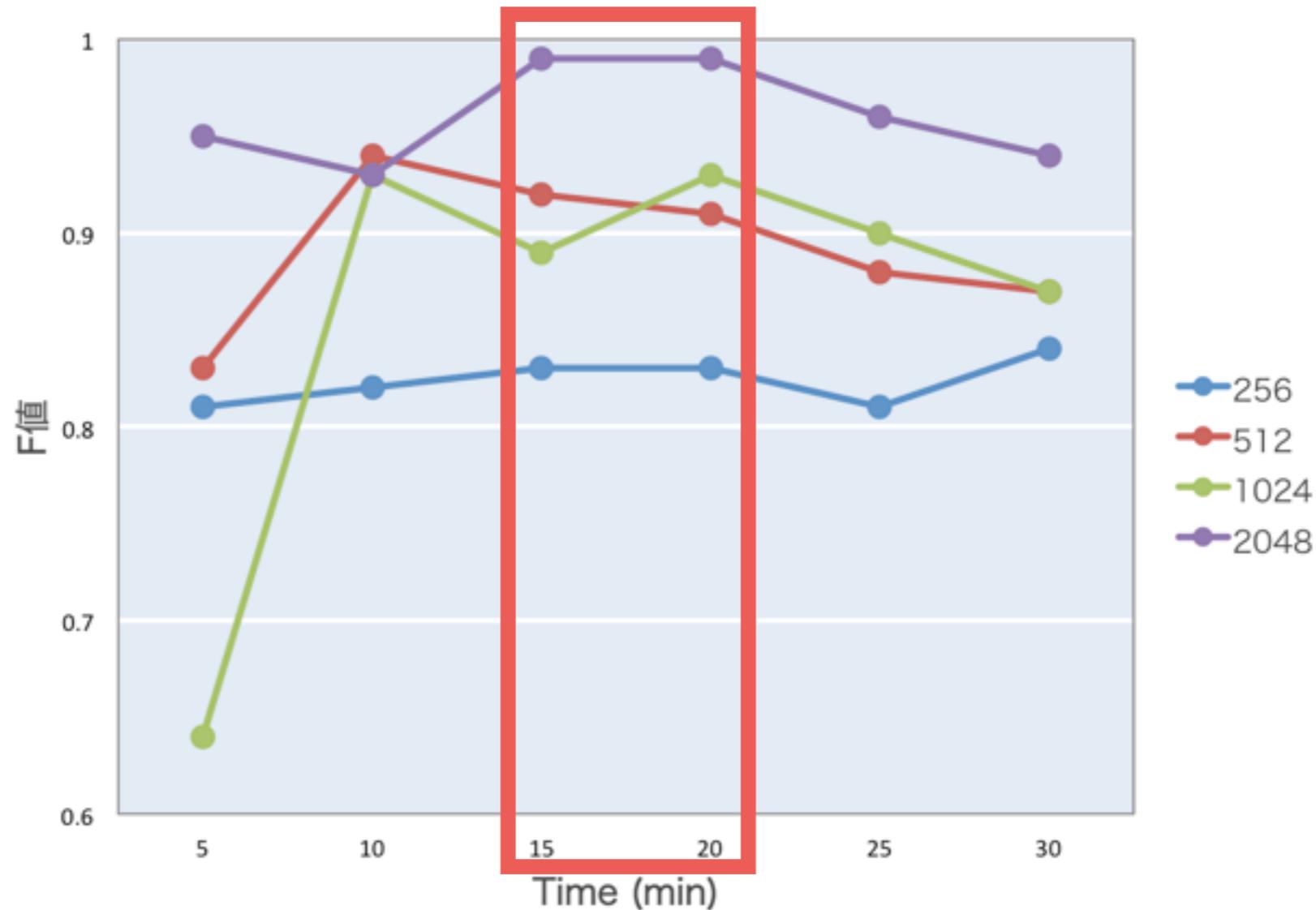
# 手法の比較

それぞれのサンプリング数とイベント長での、F値が最も高い手法 (LPZを使用)

サンプリング数	Time (min)					
	5	10	15	20	25	30
256	SVM-L	NN	NN	NN	NN	NN
512	NN	NN	NN	SVM-L	NN	NN
1024	NN	NN	NN	NN	NN	NN
2048	NN	NN	NN	NN	NN	NN

高いF値を観測したのはNNが多かった

# 特徴量の比較結果



NNにおけるサンプリング数とイベント長のそれぞれの分類結果によるF値  
 イベント長が15, 20分の時にF値は高く, 他の時間では低かった  
 PSDはサンプリング数が増加するごとにF値が上昇した

# 考察

- **Neural Networkが高性能だった理由**
  - 中間層で有効な特徴量が抽出できたために、性能が高くなったと考えられる
- **SVM-LinearがSVM-RBFより性能が高かった理由**
  - 月震データが線形で高精度に分類可能だったため
  - 少ないデータ数のため、RBFでの複雑な境界面のパラメータが上手く設定できなかったため

## まとめ

- 月震において，NNの分類性能が一番高かった
- イベント長が15, 20分の時，F値が高かった
- PSDサンプリング数が多いほど，F値が高かった

## 今後の課題

- **3方向の月震成分を同時に考慮した分類**
  - 現在は1方向の成分のみで分類
  - 分類性能の向上が見込まれる
- **イベント数の少ない震源も含めた分類**
  - イベント数が少ない震源が多いため

# 今後の展望

- NNを用いることで、未分類のイベントに確率付きの震源ラベルを付与することができる
  - NNは出力が確率となるため
- NNを用いた月震分類に有効な特徴量の解析

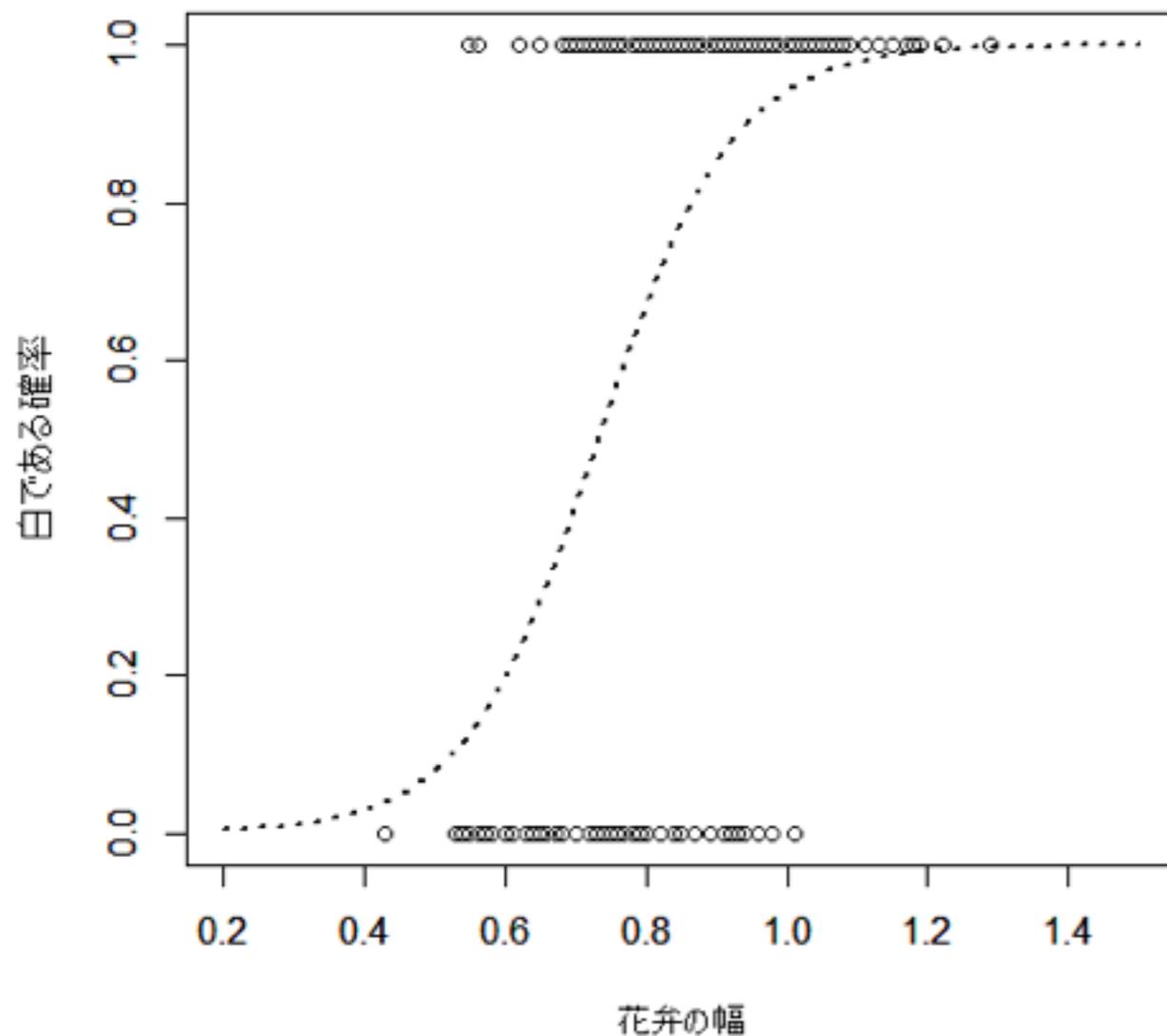


# 予備スライド

# ロジスティック回帰

- 線形分類を行う手法
- 被説明変数が0-1になるような場合に用いられ、出力を確率とすることができる  
例：「買った」「買ってない」の予測
- 医療統計などによく用いられる

# ロジスティック回帰



ロジスティック関数の例

$$y = \frac{1}{1 + \exp [-(b_1x_1 + b_2x_2 + \dots + b_ix_i + b_0)]}$$

x : 各次元の変数

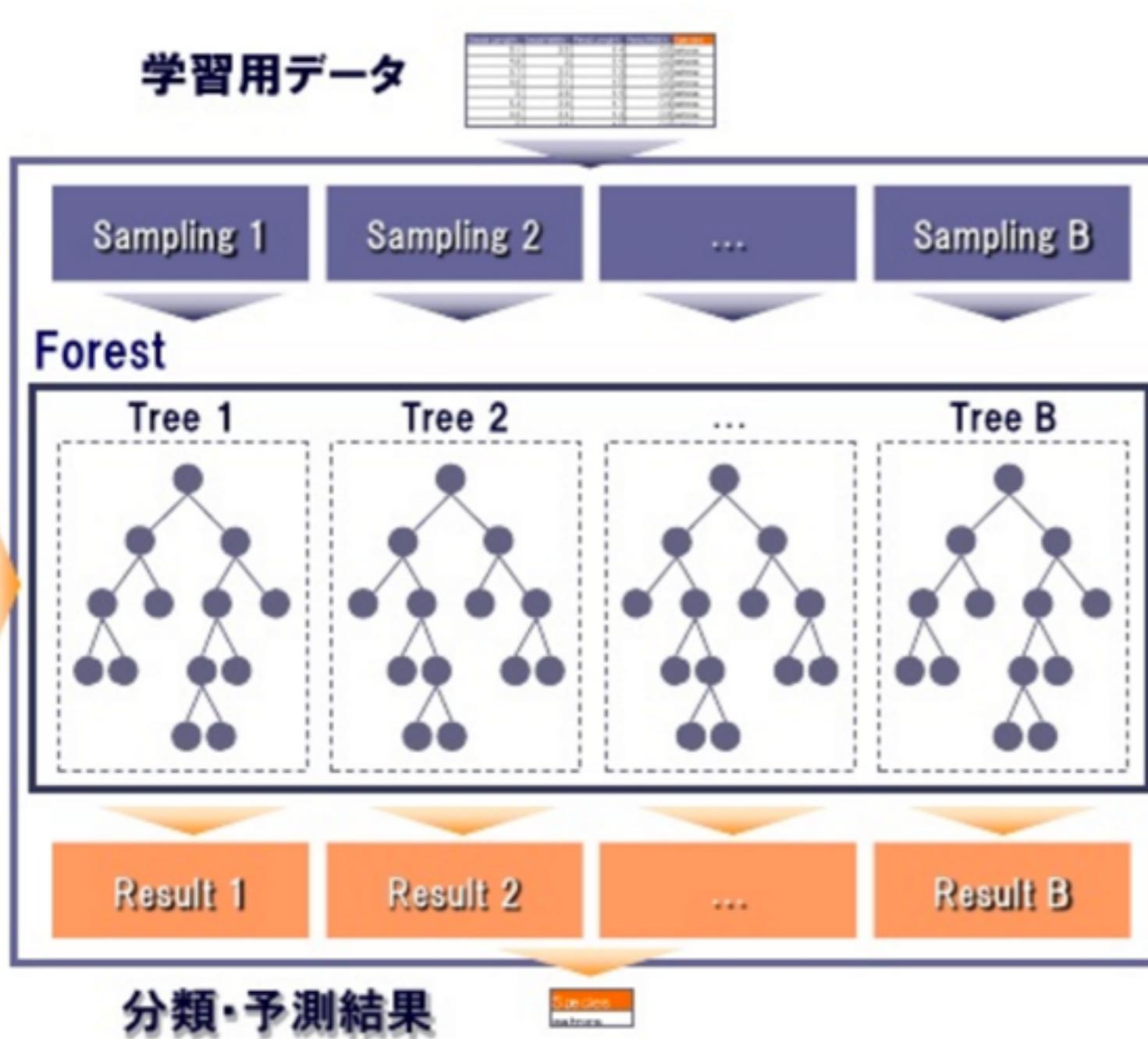
b : 対数尤度を最大にする係数

出力で確率を計算する

# Random Forest

- **アンサンブル学習の一種**  
弱分類器を複数組合わせて強分類器を作る  
本手法での弱分類器は決定木
- **次元数が多くても上手く働く**
  - 意味のある変数がノイズより少ない場合  
上手くいかない

# Random Forest



学習データからランダムに  
データを選択する



決定木を生成

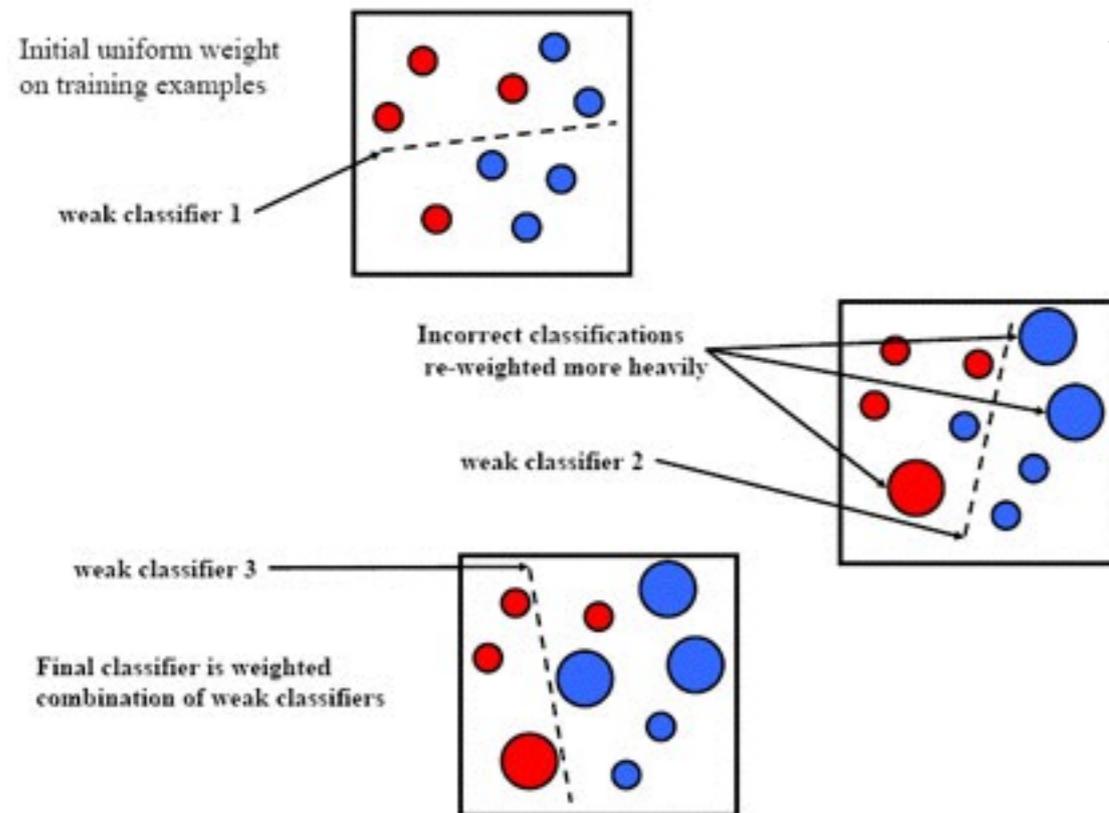


各決定木で予測し、  
多数決でクラスを判断する

# AdaBoost

- アンサンブル学習の一種
  - 今回は弱分類器を決定木とした
- 画像認識などの分野で用いられる
- ノイズの多いデータや、異常値に影響を受ける

# AdaBoost



$$H(x) = \text{sign}(\alpha_1 h_1(x) + \alpha_2 h_2(x) + \alpha_3 h_3(x))$$

複数の弱分類器を用いて分類を行う



間違っ分て分類されたイベントの  
重みを重くする



この重みを基に、もう一度  
全弱分類器で分類を行う