集合知プログラミング

9章

高度な分類手法：
カーネルメソッドとSVM

発表者：林華

はじめに

- 決定木、ベイジアン分類器、ニューラルネットワーク（既）
- 線形分類器、カーネルメソッド、SVM （未）
- 主なデータセット：デートサイトでの人々のマッチングに関するものだ。
- データセットを使い、既知分類器弱点を提示
- データセットを作る方法と利用

9.1matchmakerデータセット

あるオンラインデートサイトの情報データセット

- 年齢
- 喫煙するか？
- 子供がほしい？
- 興味があるもの？
- 住所

データセットの例

39,yes,no,skiing:knitting:dancing,220 W 42nd St New York
NY,43,no,yes,soccer:reading:scrabble,824 3rd Ave New York NY,0

23,no,no,football:fashion,102 1st Ave New York NY,30,no,no,snowboarding:knitting:computers:shopping:tv:travel,151 W 34th St New York NY,1

50,no,no,fashion:opera:tv:travel,686 Avenue of the Americas New York
NY,49,yes,yes,soccer:fashion:photography:computers:camping:movies:tv,824 3rd Ave New York NY,0

年齢のみ組み合わせ情報

24,30,1

30,40,1

22,49,0

43,39,1

23,30,1

23,49,0

…

データをセット

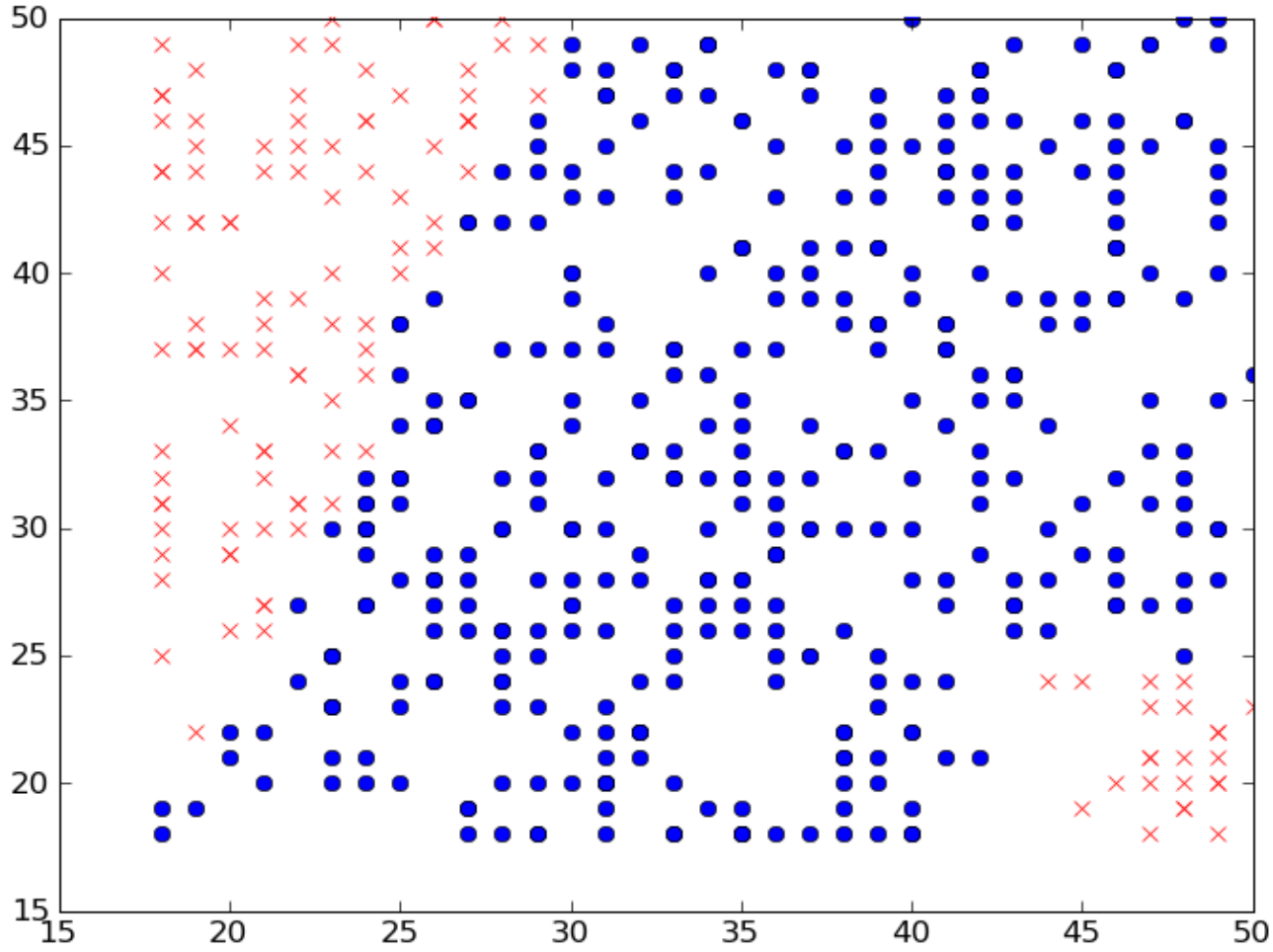
```
>>> import advancedclassify
>>>
    agesonly=advancedclassify.loadmatch('agesonly.csv',allnum=True)
>>> matchmaker=advancedclassify.loadmatch('matchmaker.csv')
>>> agesonly[0].data
[24.0, 30.0]
>>> agesonly[0].match
1
>>> matchmaker[0].data
['39', 'yes', 'no', 'skiing:knitting:dancing', '220 W 42nd St New York NY',
    '43', 'no', 'yes', 'soccer:reading:scrabble', '824 3rd Ave New York
    NY']
>>> matchmaker[0].match
0
```

マッチ情報の可視化処理

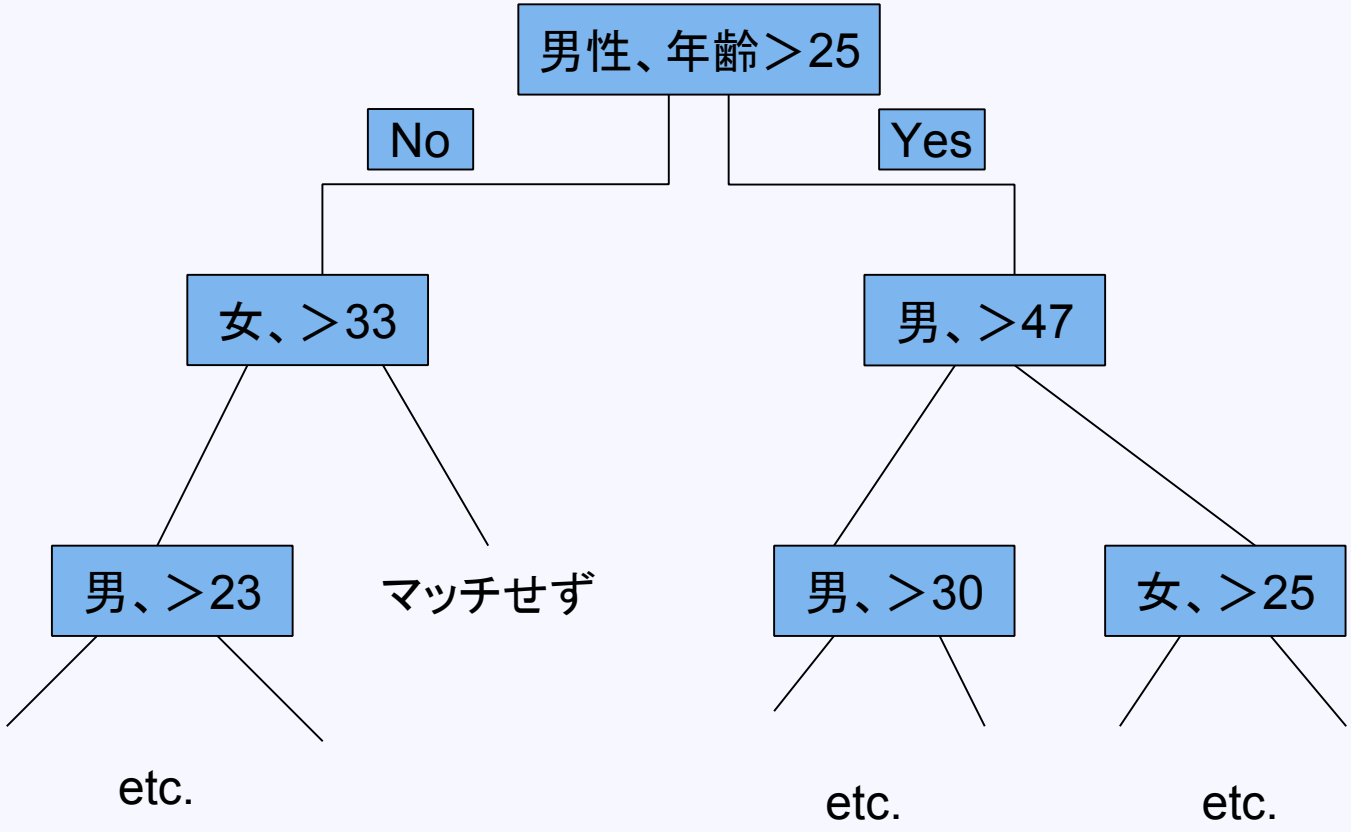
- matplotlibをインストール
- 関数plotagematchesを追加
- 実行:

```
>>> reload(advancedclassify)
<module 'advancedclassify' from
   'advancedclassify.py'>
>>> advancedclassify.plotagematches(agesonly)
```

マッチ情報の可視化結果



9.2決定木による分類器



決定木の弱点と対策

1.理由を説明するに役に立たない

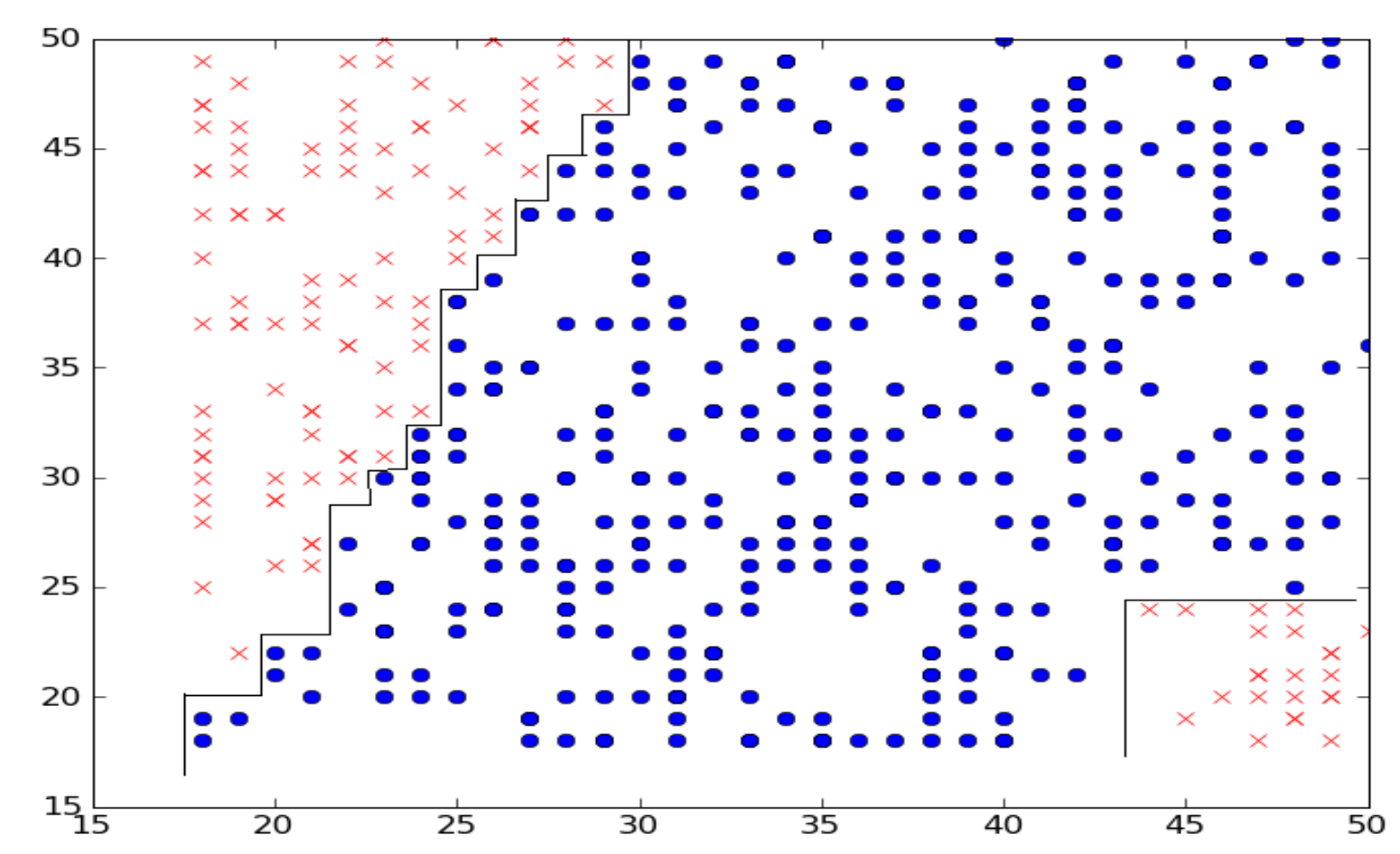
2.他の変数についても考慮する際、さらに混乱

決定境界だけを作る

1.データはどのように分けられるかわかる

2.複雑な数字の入力をクラスに分けられる

境界線を入れる



9.3基本的な線形分類

線形分類とは:

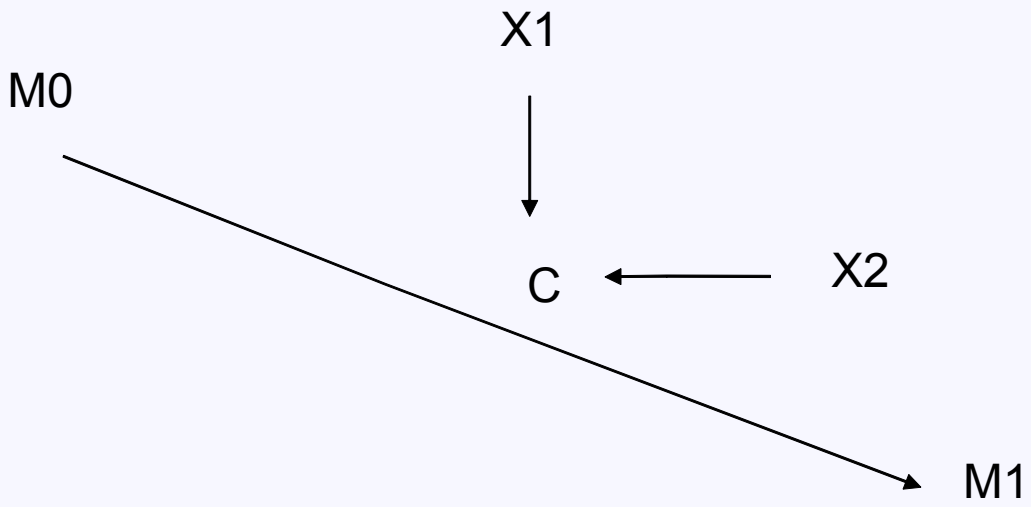
1.それぞれのクラス中のすべてのデータの平均値を探し、そのクラスの中心を表現する点を作り上げる。

2.新たなデータに対しては、どの中心に近いかで分類を行う。

まず、平均値を求める

```
>>> reload(advancedclassify)
<module 'advancedclassify' from 'advancedclassify.pyc'>
>>> avgs=advancedclassify.lineartrain(agesonly)
>>> avgs[0][0]
26.914529914529915
>>> avgs
{0: [26.914529914529915, 35.888888888888886], 1:
    [35.480417754569189, 33.0156657 96344649]}
>>> reload(advancedclassify)
>>> result=advancedclassify.lineartrain(agesonly)
>>> result
[{0: [26.914529914529915, 35.888888888888886], 1:
    [35.480417754569189, 33.0156579634 4649]}, {0: 117, 1: 383}]
```

新たなデータを分類



class=sign((X-(M0+M1)/2)・(M0-M1))

=sign(X・M0-X・M1+(M0・M0-M1・M1)/2)

結果

```
>>> reload(advancedclassify)
<module 'advancedclassify' from 'advancedclassify.py'>
>>> avgs=advancedclassify.lineartrain(agesonly)
>>> advancedclassify.dpclassify([30,30],avgs)
1
>>> advancedclassify.dpclassify([30,25],avgs)
1
>>> advancedclassify.dpclassify([30,40],avgs)
0
>>> advancedclassify.dpclassify([48,20],avgs)
1
```

なぜ間違った？原因は