

テキストマイニングの実習2

— 形態素解析を利用した集計と分析 —

2015/7/9

ビジネス科学研究科
経営システム科学専攻

演習 — 頻出語を確認する

- 形態素解析器 MeCab の出力例

```
$ mecab  
子供の運動会でとても良い映像が撮影できた  
子供 名詞,一般,*,*,*,*,子供,コドモ,コドモ,,  
の 助詞,連体化,*,*,*,*,の,ノ,ノ,,  
運動会 名詞,一般,*,*,*,*,運動会,ウンドウカイ,ウンドーカイ,,  
で 助詞,格助詞,一般,*,*,*,で,デ,デ,,  
とても副詞,助詞類接続,*,*,*,*,とても,トテモ,トテモ,,  
良い 形容詞,自立,*,*,形容詞・アウオ段,基本形,良い,ヨイ,ヨイ,よい/良い,  
映像 名詞,一般,*,*,*,*,映像,エイゾウ,エイゾー,,  
が 助詞,格助詞,一般,*,*,*,が,ガ,ガ,,  
撮影 名詞,サ変接続,*,*,*,*,撮影,サツエイ,サツエイ,,  
でき 動詞,自立,*,*,一段,連用形,できる,デキ,デキ,でき/出来,  
た 助動詞,*,*,*,特殊・タ,基本形,た,タ,タ,,  
EOS
```

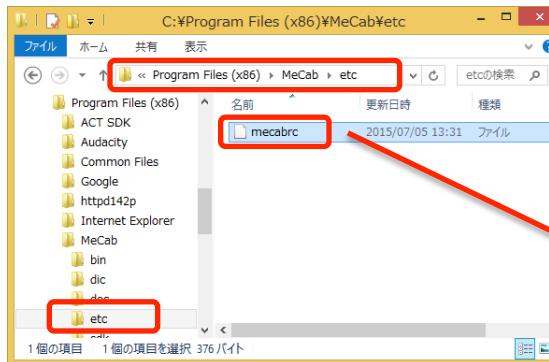
- 単語 <tab> 品詞1,品詞2,品詞3,品詞4,活用型,活用系,基本形,読み,発音
- EOS: End of Sentence
- 出力フォーマットの変更が可能

演習 — MeCabのインストール

- Windows の場合
 - 設定ファイルの書き換え (1/2)

重要!!

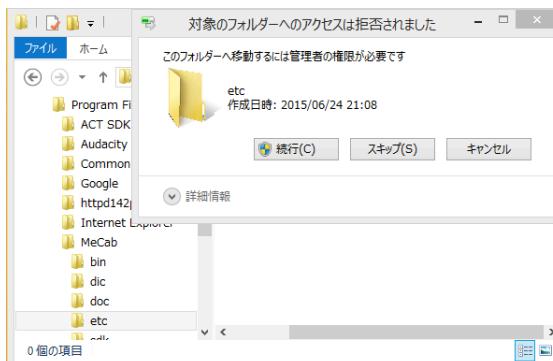
32bit版 Windows の場合は,全ての
「C:\Program Files (x86)」を
「C:\Program Files」
に読み替えてください



- エクスプローラで「C:\Program Files (x86)\MeCab\etc」を開き、「mecabrc」をデスクトップに移動する



- 次頁の内容に従って「mecabrc」ファイルを編集して保存する

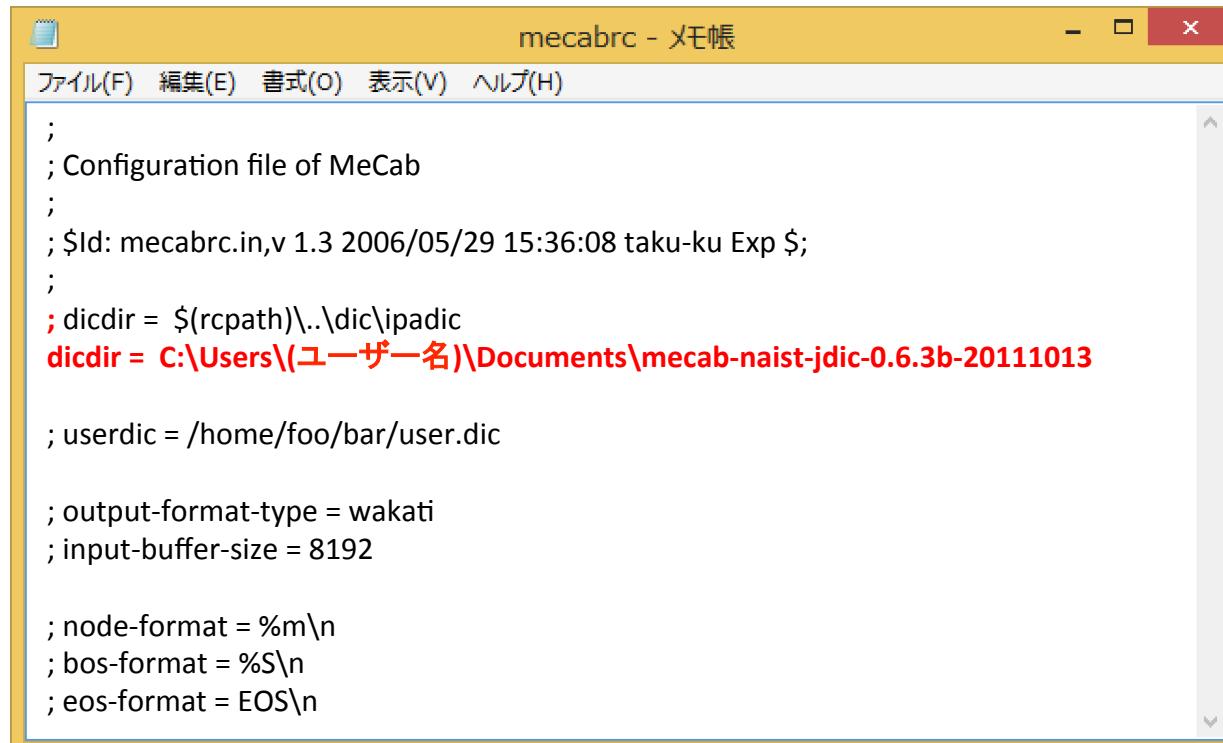


- 「mecabrc」を「C:\Program Files (x86)\MeCab\etc」に戻す

注: 右の画面が出たら [続行] をクリック

演習 — MeCabのインストール

- Windows の場合
 - 設定ファイルの書き換え (2/2)



```
; Configuration file of MeCab
; $Id: mecabrc.in,v 1.3 2006/05/29 15:36:08 taku-ku Exp $;
; dicdir = $(rcpath)\..\dic\ipadic
dicdir = C:\Users\ユーザー名\Documents\mecab-naist-jdic-0.6.3b-20111013

; userdic = /home/foo/bar/user.dic

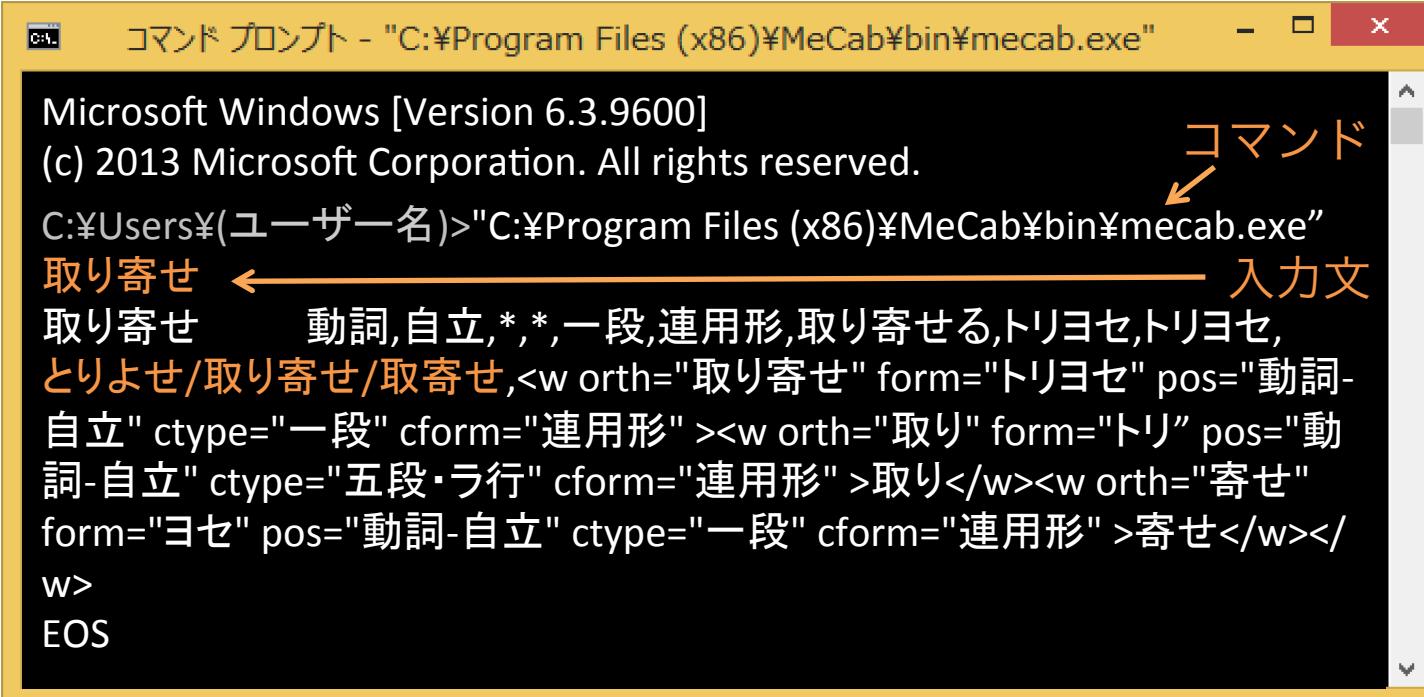
; output-format-type = wakati
; input-buffer-size = 8192

; node-format = %m\n
; bos-format = %S\n
; eos-format = EOS\n
```

※ 6行目をコメントアウトし, 7行目に NAIST-Jdic のパスを追加する

演習 — MeCabの動作確認

- Windows(64bit)の場合
 - 形態素解析の実行 (以下のような出力がされればOKです)



コマンド プロンプト - "C:\Program Files (x86)\MeCab\bin\mecab.exe"

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\ユーザー名>"C:\Program Files (x86)\MeCab\bin\mecab.exe"
取り寄せ ←————— 入力文
取り寄せ 動詞,自立,*,*一段,連用形,取り寄せる,トリヨセ,トリヨセ,
とりよせ/取り寄せ/取寄せ,<w orth="取り寄せ" form="トリヨセ" pos="動詞-
自立" ctype="一段" cform="連用形" ><w orth="取り" form="トリ" pos="動
詞-自立" ctype="五段・ラ行" cform="連用形" >取り</w><w orth="寄せ"
form="ヨセ" pos="動詞-自立" ctype="一段" cform="連用形" >寄せ</w><
/w>
EOS
```

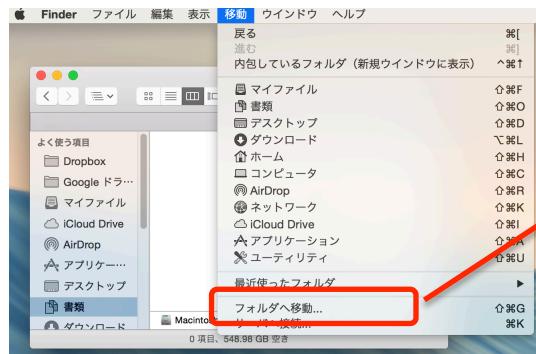
↑ コマンド

※ NAIST-Jdic 表記ゆれにも対応 → 「とりよせ/取り寄せ/取寄せ」

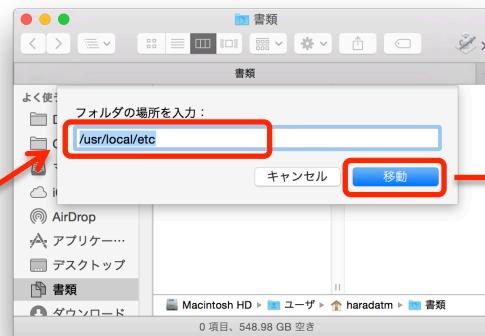
演習 — MeCabのインストール

• Mac の場合

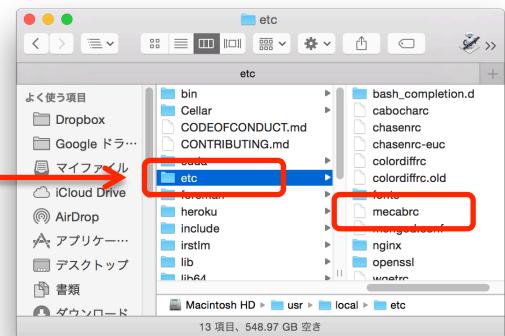
– 設定ファイルの書き換え (1/2)



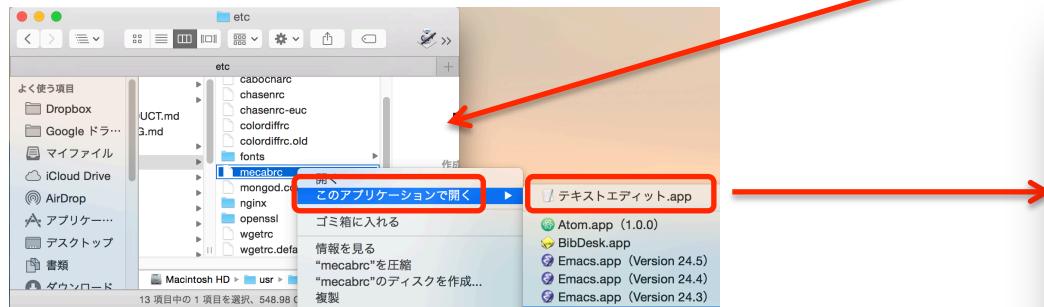
①Finderを開き、メニューから
[移動]→[フォルダへ移動...]



②「/usr/local/etc」を
入力して [移動]



③「/usr/local/etc」を開き
「mecabrc」を見つける



④「mecabrc」上で右クリック→[テキストエディット]で開く

```
; Configuration file of MeCab
; $Id: mecabrc.in,v 1.3 2006/05/29 15:36:08 taku-ku Exp $;
dicdir = /usr/local/lib/mecab/dic/ipadic
userdic = /home/foo/bar/user.dic
output-format-type = wakati
input-buffer-size = 8192
node-format = %m\n
bos-format = %S\n
eos-format = EOS\n
```

演習 — MeCabのインストール

- Mac の場合
 - 設定ファイルの書き換え (2/2)



```
;; Configuration file of MeCab
; $Id: mecabrc.in,v 1.3 2006/05/29 15:36:08 taku-ku Exp $;
; dicdir = /usr/local/lib/mecab/dic/ipadic
dicdir = /usr/local/lib/mecab/dic/naist-jdic

; userdic = /home/foo/bar/user.dic

; output-format-type = wakati
; input-buffer-size = 8192

; node-format = %m\n
; bos-format = %S\n
; eos-format = EOS\n
```

※ 6行目をコメントアウトし, 7行目に NAIST-Jdic のパスを追加する

演習 — MeCabの動作確認

- Mac の場合
 - 形態素解析の実行 (以下のような出力がされればOKです)

```
$ cd ~/Documents  
$ mecab  
取り寄せ ← コマンド  
取り寄せ ← 入力文  
取り寄せ 動詞,自立,*,*,一段,連用形,取り寄せる,トリヨセ,トリヨセ,  
とりよせ/取り寄せ/取寄せ,<w orth="取り寄せ" form="トリヨセ" pos="動詞-自  
立" ctype="一段" cform="連用形" ><w orth="取り" form="トリ" pos="動詞-自  
立" ctype="五段・ラ行" cform="連用形" >取り</w><w orth="寄せ" form="ヨセ"  
pos="動詞-自立" ctype="一段" cform="連用形" >寄せ</w></w>  
EOS
```

※ NAIST-Jdic 表記ゆれにも対応 → 「とりよせ/取り寄せ/取寄せ」

データの公開場所

- <https://github.com/haradatm/gssm-201507>

The screenshot shows a GitHub repository named `gssm-201507`. It displays two main branches: `master` and `03-samples`.

Master Branch (second commit):

- `00-slides`: second commit
- `01-data`: first commit
- `02-tools`: first commit
- `03-samples`: second commit
- `.gitignore`: first commit
- `README.md`: first commit

03-samples Branch:

- `cmdline-mac.txt`: first commit
- `cmdline-win.txt`: first commit
- `lecture-2.pdf`: second commit
- `lecture-2.r`: second commit
- `lecture-2_summary.xlsx`: first commit
- `lecture-34-py1.png`: second commit
- `lecture-34-r1.pdf`: second commit
- `lecture-34-r2.pdf`: second commit
- `lecture-34.py`: second commit
- `lecture-34.r`: second commit
- `lecture-3_frequency.xlsx`: second commit
- `lecture-3_text-words.zip`: first commit
- `lecture-3_tf-idf.xlsx`: second commit

Annotations:

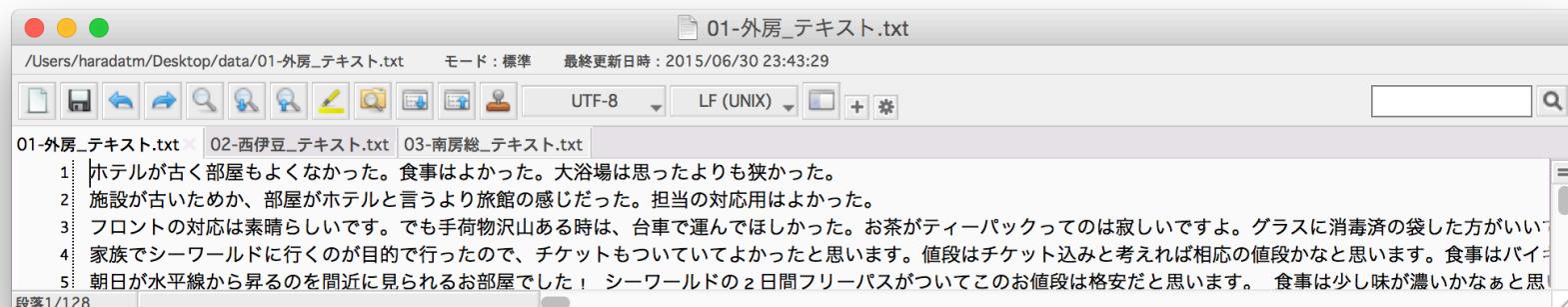
- A red box highlights the `01-data` folder in the master branch.
- A red box highlights the `03-samples` folder in the master branch.
- A red box highlights the `rakuten-eval.xlsx` file in the `01-data` folder of the master branch.
- A red box highlights the `cmdline-mac.txt` and `cmdline-win.txt` files in the `03-samples` folder of the master branch.
- A blue dashed circle highlights the `lecture-34-py1.png` through `lecture-3_tf-idf.xlsx` files in the `03-samples` folder of the master branch.
- Red text annotations:
 - "変更なし: 実行コマンド例" (No change: Execution command example) points to the `cmdline` files.
 - "変更なし: 演習用データ" (No change: Practice data) points to the `rakuten-eval` files.
 - "追加: スライド26-29 のサンプルコードと実行結果" (Added: Sample code and execution results for slides 26-29) points to the `lecture-34` files.

演習 — 頻出語を確認する

- 風呂の評価が1~2点の口コミを抜き出す

- シート名 [データ (6317)] を開く
- フィルター機能で,A列を [01-外房] のみに絞り込む
- フィルターで機能で,J列を [1,2] のみに絞り込む
- テキストエディタで新しいファイル(ウィンドウ)を開く
- D列の2行目以降を選択し,テキストエディタに貼り付ける
- ファイル名 「01-外房_テキスト.txt」 で保存する

※ 一旦, A列のチェックを外した後, 同様に [02-西伊豆] [03-南房総]についても②~⑥を繰り返す



演習 — 頻出語を確認する

- MeCab を実行する: Mac の場合

```
$ cd (作業ディレクトリ)
```

```
$ mecab -b 819200 -F"%f[6]\t%f[0]\t%f[1]\n" -U"%m\tUNK\n" -E"EOS\tEOS\n"  
01-外房_テキスト.txt > 01-外房_単語.txt
```

```
$ mecab -b 819200 -F"%f[6]\t%f[0]\t%f[1]\n" -U"%m\tUNK\n" -E"EOS\tEOS\n"  
02-西伊豆_テキスト.txt > 02-西伊豆_単語.txt
```

```
$ mecab -b 819200 -F"%f[6]\t%f[0]\t%f[1]\n" -U"%m\tUNK\n" -E"EOS\tEOS\n"  
03-南房総_テキスト.txt > 03-南房総_単語.txt
```

画面の都合上折り返し

注: バッファ・サイズには “-b 819200” を指定してください

【フォーマットを変更するパラメータ】

-F	… 項目の並び	例) -F“%f[6]\t%f[0]\t%f[1]\n”	→ 子供<tab>名詞<tab>一般
-U	… 未知語の表示法	例) -U“%m\tUNK\n”	→ 単語<tab>UNK<tab><tab>
-E	… EOSの表示法	例) -E“EOS\tEOS\n”	→ EOS<tab>EOS

演習 — 頻出語を確認する

- MeCab を実行する: Windows の場合

```
コマンド プロンプト

C:\$Users\$(ユーザー名)> "C:\Program Files (x86)\MeCab\bin\mecab.exe"
-b 819200 -F"%f[6]\t%f[0]\t%f[1]\n" -U"%m\tUNK\n" -E"EOS\tEOS\n"
01-外房_テキスト.txt > 01-外房_単語.txt

C:\$Users\$(ユーザー名)> "C:\Program Files (x86)\MeCab\bin\mecab.exe"
-b 819200 -F"%f[6]\t%f[0]\t%f[1]\n" -U"%m\tUNK\n" -E"EOS\tEOS\n"
02-西伊豆_テキスト.txt > 02-西伊豆_単語.txt

C:\$Users\$(ユーザー名)> "C:\Program Files (x86)\MeCab\bin\mecab.exe"
-b 819200 -F"%f[6]\t%f[0]\t%f[1]\n" -U"%m\tUNK\n" -E"EOS\tEOS\n"
03-南房総_テキスト.txt > 03-南房総_単語.txt
```

画面の都合上折り返し

注1: バッファ・サイズには “-b 819200” を指定してください

注2: フォーマット変更のパラメータは前頁(Macの場合)と同様です

演習 — 頻出語を確認する

• 単語分割の結果を EXCEL に貼り付ける

- ① EXCEL の新しいシートの1行目に列名を入力する
→ 列名は左から順に [エリア] [形態素] [品詞1] [品詞2] [単語]
- ② ファイル「01-外房_単語.txt」をテキストエディタで開く
- ③ 形態素解析結果の全体を選択し,EXCEL に貼り付ける
→ 貼り付け先: ①のシート上の B列の2行目
- ④ A列2行目に [01-外房] と入力し,行末までコピーする
- ⑤ E列2行目に式 [=B2&" ("&C2&" ")] を入力し,行末までコピー
→ ④および⑤は、貼り付けたデータが存在する最終行までコピー
- ⑥ シート名を [01-外房] に変更する

	A	B	C	D	E
1	エリア	形態素	品詞1	品詞2	単語
2	01-外房	ホテル	名詞	一般	ホテル (名詞)
3	01-外房	が	助詞	格助詞	が (助詞)
4	01-外房	古く	副詞	助詞類接続	古く (副詞)

※ 同様に [02-西伊豆] [03-南房総] についても②～⑨を繰り返す

演習 — 頻出語を確認する

• 単語の出現頻度を集計する

- ① シート [01-外房] を開き, A~E 列を選択してピボットを作成
→ 新しいシートに作成し, シート名を [集計] に変更

※ 同様に [02-西伊豆] [03-南房総] についても①を行う
→ ピボットは, 比較ができるように同じシート [集計] 上に作成する

A	B	C	D	E	F	G	H
1 エリア	01-外房	エリア	02-西伊豆	エリア	03-南房総		
2 品詞1	(複数の項目)	品詞1	(複数の項目)	品詞1	(複数の項目)		
3 品詞2	(複数の項目)	品詞2	(複数の項目)	品詞2	(複数の項目)		
4							
5 データの個数: 形態素		データの個数: 形態素		データの個数: 形態素			
6 行ラベル	計	行ラベル	計	行ラベル	計		
7 部屋 (名詞)	121	部屋 (名詞)	92	部屋 (名詞)	99		
8 風呂 (名詞)	99	風呂 (名詞)	84	風呂 (名詞)	77		
9 良い (形容詞)	82	良い (形容詞)	75	良い (形容詞)	54		
10 食事 (名詞)	74	ない (形容詞)	67	食事 (名詞)	3		
11 ない (形容詞)	67	食事 (名詞)	54	宿泊	3		
12 ホテル (名詞)	54	残念 (名詞)	50	良い	8		
13 宿泊 (名詞)	50	温泉 (名詞)	42	利用	7		
14 人 (名詞)	50	宿泊 (名詞)	40	料理	5		
15 残念 (名詞)	42	宿 (名詞)	38	ホテル	9		
16 朝食 (名詞)	40	料理 (名詞)	36	満足	8		
17 利用 (名詞)	38	露天風呂 (名詞)	36	ない	8		
18 海 (名詞)	36	利用 (名詞)		夕食	6		
19				残念	3		
20							

- ② フィルター機能を使い, 品詞情報で不要語削除する

- 品詞1を [形容詞, 名詞, UNK] のみに絞る
- 品詞2から [数, 非自立] のチェックを外す

演習 — 語と語の結びつきを確認する

• 品詞情報をを使った不要語削除する

- ① シート [01-外房] を開く
- ② フィルター機能で,C列を [形容詞,名詞,UNK,EOS] に絞る
- ③ フィルター機能で,D列から [数,非自立] のチェックを外す
- ④ データのあるセル全体を選択し,新しいシートに貼り付ける
- ⑤ シート名を [01-外房 (2)] に変更する

A	B	C	D	E
1 エリア	形態素	品詞1	品詞2	単語
2 01-外房	ホテル	名詞	一般	ホテル (名詞)
5 01-外房	部屋	名詞	一般	部屋 (名詞)
7 01-外房	よい			
11 01-外房	食事			
13 01-外房	よい			
17 01-外房	浴場			
23 01-外房	狭い			
26 01-外房	EOS			
27 01-外房	施設			

A	B	C	D	E
1 エリア	形態素	品詞1	品詞2	単語
2 01-外房	ホテル	名詞	一般	ホテル (名詞)
3 01-外房	部屋	名詞	一般	部屋 (名詞)
4 01-外房	よい	形容詞	自立	よい (形容詞)
5 01-外房	食事	名詞	サ変接続	食事 (名詞)
6 01-外房	よい	形容詞	自立	よい (形容詞)
7 01-外房	浴場	名詞	一般	浴場 (名詞)
8 01-外房	狭い	形容詞	自立	狭い (形容詞)
9 01-外房	EOS	EOS		EOS (EOS)
10 01-外房	施設	名詞	サ変接続	施設 (名詞)

演習 — 語と語の結びつきを確認する

- 近くに出現(=共起)する単語の組みを作る
 - ⑥ 新しいシートの1行目にあるC列より右の列名を変更する
→ [品詞1(前)] [品詞2(前)] [単語(前)] [品詞1(後)] [品詞2(後)] [単語(後)]
[単語(前)-単語(後)]
 - ⑦ 新しいシートの C～E列の3行目以降を行末まで選択し,
右側にある同じシートの F列2行目 に貼り付ける
 - ⑧ I列2行目に式 [=E2&"-"&H2"] を入力し,行末までコピーする

※ 同様に [02-西伊豆] [03-南房総] についても②～⑧を繰り返す

	A	B	C	D	E	F	G	H	I	J
1	エリア	形態素	品詞1(前)	品詞2(前)	単語(前)	品詞1(後)	品詞2(後)	単語(後)	単語(前)-単語(後)	
2	01-外房	ホテル	名詞	一般	ホテル (名詞)	名詞	一般	部屋 (名詞)	ホテル (名詞)	部屋 (名詞)
3	01-外房	部屋	名詞	一般	部屋 (名詞)	形容詞	自立	よい (形容詞)	部屋 (名詞)-よい (形容詞)	
4	01-外房	よい	形容詞	自立	よい (形容詞)	名詞	サ変接続	食事 (名詞)	よい (形容詞)	食事 (名詞)
5	01-外房	食事	名詞	サ変接続	食事 (名詞)	形容詞	自立	よい (形容詞)	食事 (名詞)-よい (形容詞)	
6	01-外房	よい	形容詞	自立	よい (形容詞)	名詞	一般	浴場 (名詞)	よい (形容詞)	浴場 (名詞)
7	01-外房	浴場	名詞	一般	浴場 (名詞)	形容詞	自立	狭い (形容詞)	浴場 (名詞)-狭い (形容詞)	
8	01-外房	狭い	形容詞	自立	狭い (形容詞)	EOS		EOS (EOS)	狭い (形容詞)	EOS (EOS)
9	01-外房	EOS	EOS		EOS (EOS)	名詞	サ変接続	施設 (名詞)	EOS (EOS)-施設 (名詞)	
10	01-外房	施設	名詞	サ変接続	施設 (名詞)	形容詞	自立	古い (形容詞)	施設 (名詞)-古い (形容詞)	

演習 — 語と語の結びつきを確認する

• 単語の組みの出現頻度を集計する

- ① シート [01-外房 (2)] を開き,A~I列を選択し,ピボットを作る
→ 新しいシートに作成し, シート名を [共起] に変更

※ 同様に [02-西伊豆] [03-南房総] についても①を行う
→ ピボットは, 比較ができるように同じシート [共起] 上に作成する

	A	B	C	D	E	F	G	H
1	エリア	01-外房	エリア	02-西伊豆	エリア	03-南房総		
2	品詞1(前)	(複数の項目)	品詞1(前)	(複数の項目)	品詞1(前)	(複数の項目)		
3	品詞2(前)	(すべて)	品詞2(前)	(すべて)	品詞2(前)	(すべて)		
4	品詞1(後)	形容詞	品詞1(後)	形容詞	品詞1(後)	形容詞		
5	品詞2(後)	(すべて)	品詞2(後)	(すべて)	品詞2(後)	(すべて)		
6								
7	データの個数: 形態素		データの個数: 形態素		データの個数: 形態素			
8	行ラベル		行ラベル		行ラベル			
9	風呂 (名詞)-ない (形容詞)	2	風呂 (名詞)-狭い (形容詞)	4	風呂 (名詞)-狭い (形容詞)	4		
10	風呂 (名詞)-狭い (形容詞)	2	風呂 (名詞)-良い (形容詞)	3	露天風呂 (名詞)-ない (形容詞)	4		
11	風呂 (名詞)-広い (形容詞)	2	露天風呂 (名詞)-良い (形容詞)	3	風呂 (名詞)-古い (形容詞)	3		
12	風呂 (名詞)-大きい (形容詞)	2	風呂 (名詞)-広い (形容詞)	2	風呂 (名詞)-良い (形容詞)	2		
13	風呂 (名詞)-良い (形容詞)	2	水風呂 (名詞)-無い (形容詞)	1	風呂 (名詞)-古い (形容詞)	1		
14	露天風呂 (名詞)-ない (形容詞)	2	風呂 (名詞)-でかい (形容詞)	1	風呂 (名詞)-小さい (形容詞)	1		
15	水風呂 (名詞)-ない (形容詞)	1	露天風呂 (名詞)-寒い (形容詞)	1	露天風呂 (名詞)-寒い (形容詞)	1		
16	風呂 (名詞)-すごい (形容詞)	1	露天風呂 (名詞)-寒い (形容詞)	1	風呂 (名詞)-大きい (形容詞)	1		
17	風呂 (名詞)-ぬるい (形容詞)	1	露天風呂 (名詞)-広い (形容詞)	1	風呂 (名詞)-大きい (形容詞)	1		
18	風呂 (名詞)-遠い (形容詞)	1	露天風呂 (名詞)-熱い (形容詞)	1	風呂 (名詞)-古い (形容詞)	1		
19	風呂 (名詞)-寒い (形容詞)	1	露天風呂 (名詞)-良い (形容詞)	1	露天風呂 (名詞)-新しい (形容詞)	1		
20	風呂 (名詞)-高い (形容詞)	1	総計		風呂 (名詞)-大きい (形容詞)	1		
21	風呂 (名詞)-小さい (形容詞)	1			風呂 (名詞)-古い (形容詞)	1		
22	風呂 (名詞)-熱い (形容詞)	1			風呂 (名詞)-大きい (形容詞)	1		
23	風呂 (名詞)-怖い (形容詞)	1			風呂 (名詞)-新しい (形容詞)	1		
24	風呂場 (名詞)-強い (形容詞)	1			風呂 (名詞)-大きい (形容詞)	1		
25	風呂場 (名詞)-狭い (形容詞)	1			風呂 (名詞)-新しい (形容詞)	1		
26	露天風呂 (名詞)-ぬるい (形容詞)	1			風呂 (名詞)-大きい (形容詞)	1		
27	露天風呂 (名詞)-良い (形容詞)	1			風呂 (名詞)-新しい (形容詞)	1		
28	総計	25						

ピボットテーブルの構成:

- 行ラベル: 風呂 (名詞)-ない (形容詞), 風呂 (名詞)-狭い (形容詞), 風呂 (名詞)-良い (形容詞), 風呂 (名詞)-大きい (形容詞), 風呂 (名詞)-古い (形容詞), 風呂 (名詞)-小さい (形容詞), 風呂 (名詞)-寒い (形容詞), 風呂 (名詞)-大きい (形容詞), 風呂 (名詞)-新しい (形容詞), 風呂 (名詞)-大きい (形容詞)
- 列ラベル: エリア, 品詞1(前), 品詞2(前), 品詞1(後), 品詞2(後)
- 値: データの個数: 形態素

ピボットテーブルのフィルター:

- 行ラベル: 風呂
- 品詞1(後): 形容詞

- ② フィルター機能を使い, 関心のある語(名詞)と評価(形容詞)の共起を確認する

- 行ラベルで [“風呂”を含む] に絞り込む
- 品詞1(後)を [形容詞] のみに絞り込む

演習課題

- 課題1
 - 各エリアの口コミデータ中に発生する単語の出現頻度を集計し、エリアによって高頻度の単語がどのように違うかを比較する
 - 3つのエリアの特徴が分かるか否かを考察する
- 課題2
 - 各エリアの口コミデータ中に発生する名詞「風呂」と形容詞の組みの出現頻度を集計し、エリアによって高頻度の組みがどのように違うかを比較する
 - 3つのエリアの特徴が分かるか否かを考察する
- 課題3
 - 3エリア全体で「風呂」のユーザー評価が低い(評価点1-2)口コミと高い(評価点4-5)口コミを課題1,2の方法を用いて比較する
 - 両者の特徴が分かるか否かを考察する

演習 — チャレンジ課題 (1 / 2)

• 単語出現頻度で文書ベクトルを作る

- ① 新しいシートを開いて, [01-外房 (2)] [02-西伊豆 (2)] [03-南房総 (2)] の A~E 列を縦方向に並べて貼り付ける
→ 新しいシートに作成し, シート名を [全エリア] に変更
- ② F列2行目に [1] を F列3行目に式 [=IF(E2="EOS (EOS)",F2+1,F2)] を入力し, F列3行目から行末まで式をコピーする

	A	B	C	D	E	F
1	エリア	形態素	品詞1(前)	品詞2(前)	単語(前)	文書ID
2	01-外房	ホテル	名詞	一般	ホテル (名詞)	1
3	01-外房	部屋	名詞	一般	部屋 (名詞)	1
4	01-外房	よい	形容詞	自立	よい (形容詞)	1

- ③ A~E 列を選択して新しいシートにピボットを作成する

→ 文章IDごとに[単語(前)] の
[データの個数] を合計する

→ フィルター機能を使って,
[品詞1(前)] の [EOS] のみ
チェックを外す

	A	B	C	D	E	F	G
品詞1(前)	(複数の項目)						lecture-3_frequency.xlsx
品詞2(前)	(すべて)						
データの個数: 単語(前)	列ラベル	□					
行ラベル	部屋 (名詞)	風呂 (名詞)	良い (形容詞)	食事 (名詞)	ない (形容詞)	宿泊 (名詞)	ホテル (名詞)
01-外房	121	99	82	74	67	50	54
1	1			1			1
2		1					1
3					1		1
4		1	1		1	1	1
5		1	2		1	1	1

演習 — チャレンジ課題 (2/2)

• 単語重要度(TF・IDF)で比較する (1/3)

- ① 新しい EXCEL ファイルを開き、前頁のシート [全エリア] のすべてのデータを値として (=数式でなく) コピーする
→ シート名を [全エリア] に変更する
- ② H列1行目に列名 [文書ID-単語], 2行目に式 [=G2&“-”&F2] を入力し、H列2行目から行末まで式をコピーする
- ③ A～H列まで選択し、新しいシートに値としてコピーする
→ シート名を [DF 計算用] に変更する
- ④ シート [DF 計算用] の A～H列を選択し、H列昇順で並べ替える
- ⑤ I列1行目に列名 [DF], 2行目に式 [=IF(H1<>H2,1,0)] を入力し、I列2行目から行末まで式をコピーする

The screenshot shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H	I
1	NO.	エリア	形態素	品詞(前)	品詞2(前)	単語(前)	文書ID	文書ID-単語	DF
2	8	01-外房	EOS	EOS		EOS (EOS)	1	1-EOS (EOS)	1
3	1	01-外房	ホテル	名詞	一般	ホテル (名詞)	1	1-ホテル (名詞)	1
4	3	01-外房	よい	形容詞	自立	よい (形容詞)	1	1-よい (形容詞)	1
5	5	01-外房	よい	形容詞	自立	よい (形容詞)	1	1-よい (形容詞)	0

The formula bar shows =IF(H1<>H2,1,0). The cell H2 is selected. The ribbon tabs at the bottom are '全エリア' (selected), 'DF 計算用', 'Sheet1', 'Sheet2', and '+'. The status bar shows '標準表示'.

演習 — チャレンジ課題 (2/2)

• 単語重要度(TF・IDF)で比較する (2/3)

- ⑥ A~I列を選択して新しいシートにピボットテーブルを作成し、DF値を集計する

→ シート名を [DF 計算用 (ピボット)] に変更する

- ⑦ 新しいシートを開いて、シート [全エリア] のすべてのデータを値としてコピーする

→ シート名を [エリアTF 計算用] に変更する

- ⑧ H列1行目に列名 [エリア-単語], 2行目に式 [=G2&"-"&F2] を入力し、H列2行目から行末まで式をコピーする

- ⑨ I列1行目に列名 [TF], 2行目に式 [=COUNTIF(H:H,H2)] を入力し、I列2行目から行末まで式をコピーする ※この計算は時間がかかります

	A	B
3	合計 : DF	
4	行ラベル	計
5	EOS (EOS)	344
6	部屋 (名詞)	195
7	風呂 (名詞)	167
8	食事 (名詞)	147
9	良い (形容詞)	130
10	窓沿 (名詞)	111

	A	B	C	D	E	F	G	H	I
1	NO.	エリア	形態素	品詞1(前)	品詞2(前)	単語(前)	文書ID	エリア-単語	TF
2		1 01-外房	ホテル	名詞	一般	ホテル (名詞)		1 01-外房-ホテ	54
3		2 01-外房	部屋	名詞	一般	部屋 (名詞)		1 01-外房-部屋	121
4		3 01-外房	よい	形容詞	自立	よい (形容詞)		1 01-外房-よい	23
5		4 01-外房	食事	名詞	サ変接続	食事 (名詞)		1 01-外房-食事	74

演習 — チャレンジ課題 (2/2)

• 単語重要度(TF・IDF)で比較する (2/3)

- ⑩ 新しいシートを開いて、シート [エリアTF 計算用] のすべてのデータを値としてコピーする

→ シート名を [エリアTF-IDF] に変更する

- ⑪ J～M列の1行目に以下の列名を、2行目に以下の式を入力し、J～M列の2行目から行末まで式をコピーする

列名 [DF] → 式 [=VLOOKUP(F2,'DF 計算用 (ピボット)'!A:B,2, FALSE)]

列名 [総文書数 D] → 式 [=MAX(G:G)]

列名 [IDF=ln(D/DF)] → 式 [=LN(K2/J2)]

列名 [TF*IDF] → 式 [=I2*L2]

J1	A	B	C	D	E	F	G	H	I	J	K	L	M
1	NO.	エリア	形態素	品詞1(前)	品詞2(前)	単語(前)	文書ID	文書ID-単語	TF	DF	総文書数 D	IDF=ln(D/DF)	TF*IDF
2	1	01-外房	ホテル	名詞	一般	ホテル (名詞)		1 01-外房-ホテ	54	78	344	1.484	80.132
3	2	01-外房	部屋	名詞	一般	部屋 (名詞)		1 01-外房-部屋	121	195	344	0.568	68.685
4	3	01-外房	よい	形容詞	自立	よい (形容詞)		1 01-外房-よい	23	41	344	2.127	48.923
5	4	01-外房	食事	名詞	サ変接続	食事 (名詞)		1 01-外房-食事	74	147	344	0.850	62.915

演習 — チャレンジ課題 (2/2)

• 単語重要度(TF・IDF)で比較する (3/3)

⑫ A～E 列を選択して新しいシートにピボットを作成する

→ [エリア]ごとに[TF*IDF]を
平均する

→ フィルター機能を使って、
[品詞1(前)]の [EOS] のみ
チェックを外す

A	B	C	D	E	F	G	H	I	J	K
品詞1(前) (複数の項目)										
品詞2(前) (すべて)										
平均 : TF*IDF	列ラベル									
行ラベル	良い (形容詞)	ない (形容詞)	ホテル (名詞)	風呂 (名詞)	人 (名詞)	部屋 (名詞)	シーワルド (未知語)	利用 (名詞)	子供 (名詞)	宿 (名詞)
01-外房	79.795	76.391	80.132	71.542	79.607	68.685	58.527	54.955	72.320	60.542
02-西伊豆	72.983	71.830	43.034	60.702	41.396	52.223		49.171	38.287	65.731
03-南房総	46.709	41.046	57.873	55.644	39.804	56.197		67.971	48.923	29.406
(空白)										
総計	69.556	66.995	64.198	63.332	59.918	59.868	58.527	58.443	56.977	56.852

※単語重要度による文書ベクトル

⑬ 同様に A～E 列を選択して新しいシートにピボットを作成する

→ [単語(前)]ごとに[TF*IDF]を
平均する

→ フィルター機能を使って、
[エリア]を[01-外房]に
絞り込む

A	B	C	D	E	F	G	H	I
エリア	01-外房	エリア	02-西伊豆	エリア	03-南房総			
平均 : TF*IDF		平均 : TF*IDF		平均 : TF*IDF				
行ラベル	計	行ラベル	計	行ラベル	計			
ホテル (名詞)	80.132	露天 風呂 (名詞)	74.860	利用 (名詞)	67.971			
良い (形容詞)	79.795	温泉 (名詞)	72.988	料理 (名詞)	65.638			
人 (名詞)	79.607	良い (形容詞)	72.983	ベット (名詞)	61.956			
ない (形容詞)	76.391	ない (形容詞)	71.830	宿泊 (名詞)	59.949			
子供 (名詞)	72.320	宿 (名詞)	65.731	ホテル (名詞)	57.873			
風呂 (名詞)	71.542	風呂 (名詞)	60.702	部屋 (名詞)	56.197			
朝食 (名詞)	69.191	時 (名詞)	59.787	夕食 (名詞)	56.018			

※ [02-西伊豆] [03-南房総] についても⑬を繰り返し横並びにする

分析を進める

- 単語ランキング
 - 分析対象を増やす
 - 評価ポイントは1-2 (今回) → 範囲を広げる
※ ただし,外房の対象口コミ127件 → 形態素解析結果 20,215行
- 共起ランキング
 - 評価表現をまとめる
 - ポジティブ: 良い, 広い, 大きい, 美味しい など
 - ネガティブ: ない, 狹い, ぬるい, 通り など
 - 今回は bi-gram → 少し離れた単語も考慮する
 - 係り受けを考慮する

スクリプト言語を使う

- R の場合
 - RMeCab を使用する
 - R 上で MeCab を実行するライブラリ
<http://rmecab.jp/wiki/index.php?RMeCab>
- Python の場合
 - mecab-python を使用する
 - Python 上で MeCab を実行するライブラリ
[Mac] <https://mecab.googlecode.com/svn/trunk/mecab/doc/bindings.html>
[Windows] <http://aidiary.hatenablog.com/entry/20101121/1290339360>
 - 通常, Numpy, Scipy, scikit-learn, matplotlib 等も必要

R — RMeCab を使う

- R の強力な統計的手法やツールをテキストデータの分析に活用できる

```
# RMeCab のインストール
install.packages ("RMeCab", repos = "http://rmecab.jp/R")
library(RMeCab)

# ファイルを読み込む
path<- "rakuten-eval-utf8.txt"
data<-read.table(path,header=T,sep='\t',row.names=NULL)

# テキストをエリアごとに結合する
text1 = ""
text2 = ""
text3 = ""
for (i in 1:nrow(data)) {
  if (data[i,"エリア"] == "01-外房") {
    text1<-paste(text1,data[i,"テキスト"],sep="\n")
  } else if (data[i,"エリア"] == "02-西伊豆") {
    text2<-paste(text2,data[i,"テキスト"],sep="\n")
  } else if (data[i,"エリア"] == "03-南房総") {
    text3<-paste(text3,data[i,"テキスト"],sep="\n")
  }
}
text<-rbind(
  data.frame(area="01", text=text1),
  data.frame(area="02", text=text2),
  data.frame(area="03", text=text3)
)
```

lecture-34.r

```
# テキスト列を形態素解析する (エリアごと)
result<-docMatrixDF(text[,2],pos=c("名詞","形容詞","動詞"))
colnames(result)[1]<- "01-外房"
colnames(result)[2]<- "02-西伊豆"
colnames(result)[3]<- "03-南房総"

# 行方向に集計した重みで、データ降順にソートする
result<-as.data.frame(result[order(apply(result,1,sum),decreasing=T),])

# 行列を転置する (単語は上位150件残す)
t<-t(result)[,1:150]

# 主成分分析
pc<-prcomp(t,scale=TRUE)

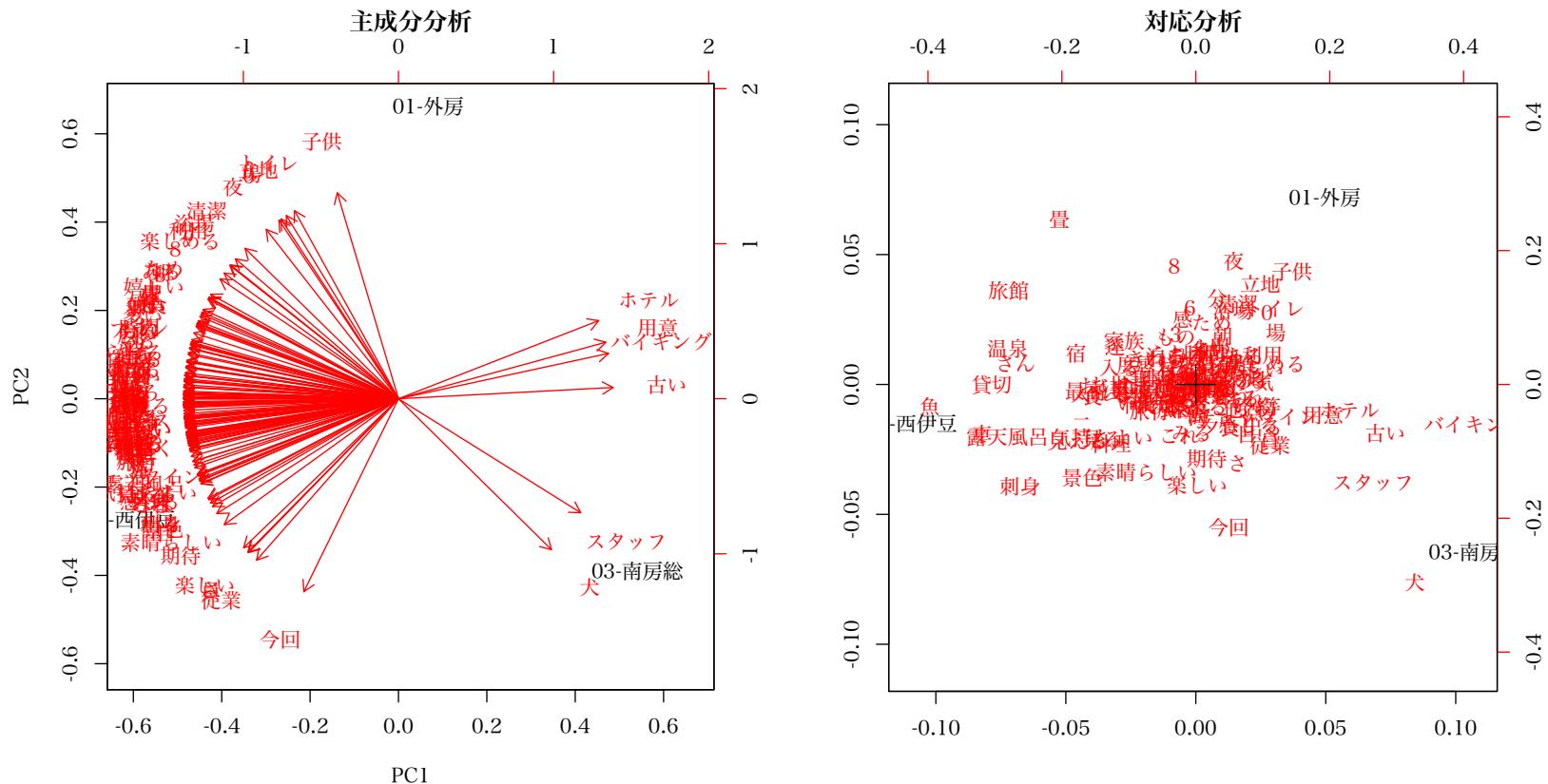
# 対応分析
library(MASS)
ca<-corresp(t,n=2)

# プロット
par(family="serif")
biplot(pc, main="主成分分析")
biplot(ca, main="対応分析")
```

lecture-34.r

R — RMeCab を使う

- 主成分分析や対応分析を使ったマップ



対象(エリアや施設)と変数(単語)の関係を視覚化

Python — mecab-python を使用

- 強力な機械学習ライブラリを活用できる

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

import sys, re
reload(sys)
sys.setdefaultencoding('utf-8')

path = "rakuten-eval-utf8.txt"

# テキストをエリアごとに結合する
text1 = ""
text2 = ""
text3 = ""

for i, line in enumerate(open(path, 'r')):
    line = unicode(line).strip()
    area = line.split('\t')[0]
    if area == u'01-外房': text1 += line.split('\t')[3]
    elif area == u'02-西伊豆': text2 += line.split('\t')[3]
    elif area == u'03-南房総': text3 += line.split('\t')[3]
    else: continue
texts = [text1, text2, text3]

# テキスト列を形態素解析する(エリアごと)
import MeCab
tagger = MeCab.Tagger()
data = []
for text in texts:
    encoded_text = text.encode('utf-8')
    node = tagger.parseToNode(encoded_text)
    terms = []
    while node:
        feature = re.split('[\s,]', (node.feature.decode('utf-8')).strip())
        if feature[0] == u'名詞' or feature[0] == u'形容詞' or feature[0] == u'動詞':
            terms.append(feature[6])
        node = node.next
    data.append(u'\t'.join(terms))

lecture-34.py
```

```
# 単語頻度ベクトルを作る
from sklearn.feature_extraction.text import CountVectorizer
def splitter(text): return text.split('\t')
vectorizer = CountVectorizer(analyzer=splitter, min_df=1, max_features=150)
features = vectorizer.fit_transform(data)

# 主成分分析
from sklearn.preprocessing import StandardScaler
Xs = StandardScaler().fit_transform(features.toarray())
from sklearn.decomposition import PCA
pca = PCA(n_components=2)
Xr = pca.fit_transform(Xs)
loadings = pca.components_.transpose()

# プロットの準備
import matplotlib.pyplot as plt
plt.figure()
plt.title(u"主成分分析")

# 主成分得点のプロット
X, Y = Xr[:, 0], Xr[:, 1]
ax1 = plt.subplot(1,1,1)
ax1.scatter(X, Y, edgecolors="none", facecolors="none", label="none")
for x, y, l in zip(X, Y, [u"01-外房", u"02-西伊豆", u"03-南房総"]):
    ax1.text(x, y, l, ha='center', va="center", size=10, color="black")

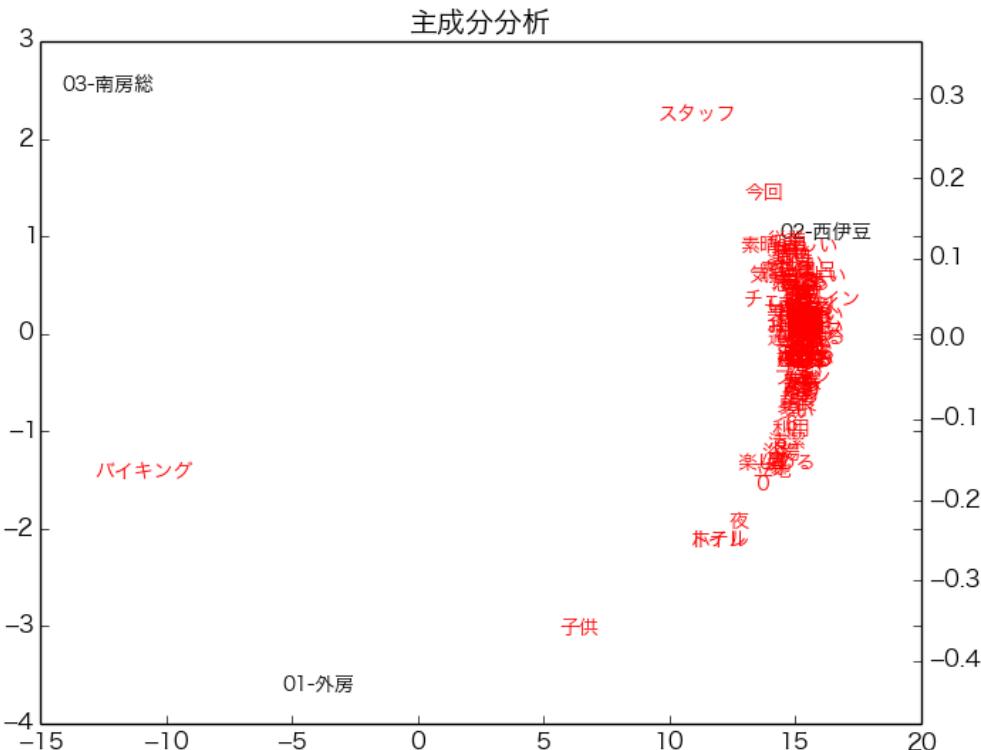
# 主成分負荷量のプロット
l1, l2 = loadings[:,0], loadings[:,1]
ax2 = ax1.twiny().twinx()
ax2.set_xlim(min(l1)/0.75, max(l1)/0.75)
ax2.set_ylim(min(l2)/0.75, max(l2)/0.75)
ax2.set_xticks([])
ax2.scatter(l1, l2, edgecolors="none", facecolors="none", label="1")
for x, y, l in zip(l1, l2, vectorizer.get_feature_names()):
    ax2.text(x, y, l, ha='center', va="center", size=10, color="red")

plt.savefig('lecture-34-py1.png')
plt.show()

lecture-34.py
```

Python — mecab-python を使用

- 主成分分析を使ったマップ



対応分析

?

Python から Rのデータ操作ができるライブラリもある: Pandas 等

係り受け解析とは

- 修飾関係(係り元→係り先)を推定する
 - 文の意味を理解することができる
- 3つの役割
 - 単語を文節にまとめ上げる
 - 文節間の係り受け関係を見つける
 - 固有表現を抽出する
- 代表的なツール
 - CaboCha (<http://taku910.github.io/cabocha/>)
 - KNP (<http://nlp.ist.i.kyoto-u.ac.jp/?KNP>)

CaboCha による係り受け解析

- 係り受け解析の実行 (Mac の場合)

```
$ cabocha ← コマンド  
子供の運動会でとても良い映像が撮影できた ← 入力文  
子供の-D  
運動会で-----D  
    とても-D |  
    良い-D |  
    映像が-D  
    撮影できた
```

文節まとめ上げ



係り受け解析

共起に比べてより正確な意味を取ることができる

参考書

(R と事例)

- [1] 金明哲. "テキストデータの統計科学入門." 岩波書店, 2009.
- [2] 石田基広. "RMeCab によるテキスト解析. R によるテキストマイニング入門." 森北出版, 2008, 51-82.

(他のツールと事例)

- [3] 那須川哲哉. "テキストマイニングを使う技術/作る技術: 基礎技術と適用事例から導く本質と活用法." 東京電機大学出版局, 2006.
- [4] 上田隆穂, 黒岩祥太, 戸谷圭子. "テキストマイニングによるマーケティング調査." 講談社, 2005.

(統計解析が中心)

- [5] 前田忠彦; 山崎誠. 言語研究のための統計入門. くろしお出版株式会社, 東京, 2013.