インターネット掲示板の特定スレッドおける時系列テキストデータに対するテキストマイニングとその考察

電気通信大学システム工学科
早川敦士

概 要

今日、データマイニングの需要が今まで以上に上がっている。対象とされるデータの分野は様々であり、活用されている業種の多種に渡る。インターネット上には、数値データよりも圧倒的に量が多いものがテキストデータである。その中でも、時間系列ごとに取得することができ、規模の大きいデータとして、インターネット掲示板がある。インターネット掲示板の中で最大級であるものの一つに 2 ちゃんねるというサイトがある。この掲示板では、政治や経済などのカテゴリ毎に分類されているのに加え、スレッド毎にテーマを絞った発言が蓄積されている。また、それぞれのスレッドでは活発に発言が行われていて、内容の深いものが多く見受けられる。今回は、このように時系列で蓄積されているスレッドを一つに絞り、分析を行い考察をしていきたいと思う。その対象として、「侵略！イカ娘」という作品について発言が行われているスレッドに限定することによって、深く分析が行えるようにした。当然、他のスレッドに対しても同様に分析を行っていくことができる。

目 次

1　初めに

インターネット上に大量に蓄積されているテキストデータの一つとして、2 ちゃんねるのような掲示板では、過去長い年月のレスを取得することが可能である。ミラーサイトと呼ばれる元データと同等のものを閲覧することができるサイトが有志によって提供されている。そのサイトの一つに `http://app.xrea.jp/dat/` があり、今回はこれを利用することによって、データを取得し、分析を行なっていった。S-PLUS は、多くの分析手法を利用することができ、さらにプログラミングをすることによって、分析を自由自在に行うことが出来るので、統計解析をする上で非常に有用なソフトウェアである。今回、分析する手法は、折れ線グラフによる可視化、相互相関関数から時系列に相関関係を求める事、主成分分析による分析の三つの手法を利用した。それぞれの分析では、予め用意されている機能も十分であるが、プログラミングが可能な点を利用して、既存の方法に工夫を加えた分析を行った。また、自然言語の処理やデータの取得に対しては、python と MeCab を使用して行っている。データを収集し、整形したのち、csv 形式で出力したのち S-PLUS を使用した。csv 形式のデータであれば、S-PLUS の強力な統計解析を使用することが出来る。また、S-PLUS では csv ファイルを処理する際にラベル名がアスキー文字である必要がある。そのため、python で前処理を行う段階で、それぞれの単語を word1,word2,word3 のように命名していき、実際の語と対応表を作成している。

2　データ収集と整形

先に述べた用にデータの収集と整形については python と MeCab を使用している。MeCab は形態素解析を行うソフトウェアで、文章を単語ごとに区切り、各々の品詞を推定することが出来る。また、辞書として ipadic を使用した。2 ちゃんねるのミラーサイトから、html 形式で過去のスレッドに遡り、一番初めのスレッドから順にデータの収集を行った。収集期間は、2010/3/26〜2011/10/18 である。取得したデータに対して、次に示す順で整形した。

1. １レスあたりに含まれる記号の割合が 5 割を超えるものは、アスキーアートと判断し、分析対象から外した。
2. URL を取り除いた。
3. MeCab による品詞の判定で名詞と形容詞と判断されたものを抽出し、それぞれの語の基本形を抽出した。
4. 抽出した語の出現頻度を求める。

語の出現頻度を求める時に、データを時系列として扱うために、年月ごとに求めている。また、語の出現頻度の相対度数を求めることにより、月ごとに異なるデータ量を平等に扱えるようにした。名詞と形容詞の種類は、すべての月を合計すると、27623 種類存在した。これらの全てのデータを扱うと、計算量が非常に大きくなることを考慮して、出現頻度の高い 100 種類の語に限定して分析を行った。この操作を加えることによって、全テキストデータの特徴的な語に限定して行うことが出来るので、分析を行うときに有用な情報を対象とすることができ、全テキストに対して一度しか出現しないような語を取り除くことが出来るため、効率的に分析を行うことができる。また、この１００語が占める割合は、約 44.3%あった。一見すると、50%を超えておらず分析対象として不十分と思われるかもしれないが、助詞や助動詞、句読点などの分析から除外した語が大量に存在することを考慮すれば、元のテキストの特徴を十分に捉えていると考えることができる。

3　折れ線グラフによって月ごとの特徴を捉える

一般的に言えることであるが、どの分野や話題に対しても、流行り廃りが存在する。他には、導入期・成長期・成熟期・衰退期といった時間的な流れがある。語の出現頻度を月ごとに捉え、折れ線グラフにより可視化することにより、これらを認識することが出来る。

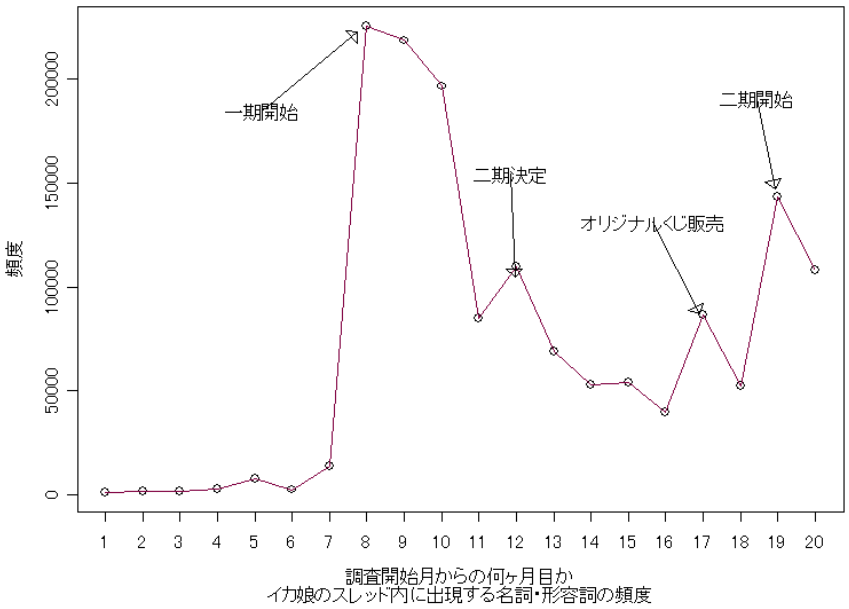

図 1: イカ娘のスレッドに出現する名詞・形容詞の頻度

図 1 に、イカ娘のスレッドに出現する名詞・形容詞の頻度を時系列にグラフに表現した。この図において、語の出現頻度が上がる月がいくつかあることが分かる。これらの月に注目し調査を行うと、その要因を知ることが出来る。この図から、アニメの放送が非常に大きく影響していることが分かる。それまで、週刊誌にて連載されていた作品であり、継続的に発言が行われていることを読みとることが出来るが、テレビでの放送によって、急激に注目が高まった事を認識することが出来る。初めの盛り上がりはアニメ一期の放送であり、これは三ヶ月で放送が終了し最終回を迎えた。それにともなって、発言数が減っている。しかしながら、テレビ放送前に比較すると比較にならないほど、発言数が増えている。二つ目の盛り上がりについては、アニメ二期放送の決定の情報があった月であった。これによって、一時的に盛り上がりが見られるが一時的なものであり、その後、現象傾向にある。さらに、データの取得から 17ヶ月目つまり 2011 年 7 月に企業によるキャンペーンによって、再び注目され、アニメ二期の放送にいたる。このような傾向を獲得することが出来る。

作図を行う時に、plot 関数を使用することが出来るがそれだけでは、点のみのプロットであったり、線のみである。lines という関数を組み合わせ、色を変更することによって、印象が深くなるように工夫した。また、矢印を追加することによって、折れ線グラフの特徴の理由を付加している。この時に、locator(1) という関数を使用することによって、マウス操作で対話的に矢印を追加することが出来る。

4　出現する語の関係を調査する

4.1　2 種類の特徴的な語を折れ線グラフで可視化する

「侵略!イカ娘」という作品の主人公である「イカ娘」という登場人物は、多くのファンから「可愛い」ものとして捉えられている。それを決定付けるものとして、「イカ娘 ネタバレ」と検索エンジンでクエリを投げると、タイトルが「単語記事: ネタバレ:イカ娘かわいい」という記事が見つかる。そこで、「イカ」「可愛い」という単語を同じグラフ上に時系列に表現し、それぞれの語の関係を見ていくことにした。これを図 2 に示した。この図によって、それぞれの語が多くの月において同じように推移していることが分かる。しかしながら、調査月から 17ヶ月目において、他とは違う挙動を示している。この月に起きた事柄を調査すると、オリジナルくじというキャンペーンが行われいることが判明した。その景品は、「侵略!イカ娘」という作品に関わるものであった。頻度の推移が伸びていることから、これが注目されていた事が分かる。しかしながら、この作品に対して重要な因子である「可愛い」という要素が抜けているということを知ることが出来る。
作図する上で、スケールの違う 2 変数を表現する点でいくつかの工夫をした。2 変数のスケール両方を考慮した軸を一つとると、片方の変数の推移を示した折れ線グラフが潰れた形になってしまう。しかしながら、それぞれを別の軸に取ることによって、それぞれの変数の時系列の推移を観察しやすくした。また、変数毎に色を変化させ、それに対応するように軸の色も変化させることによって、読み手に誤解を与えないように可視化している。

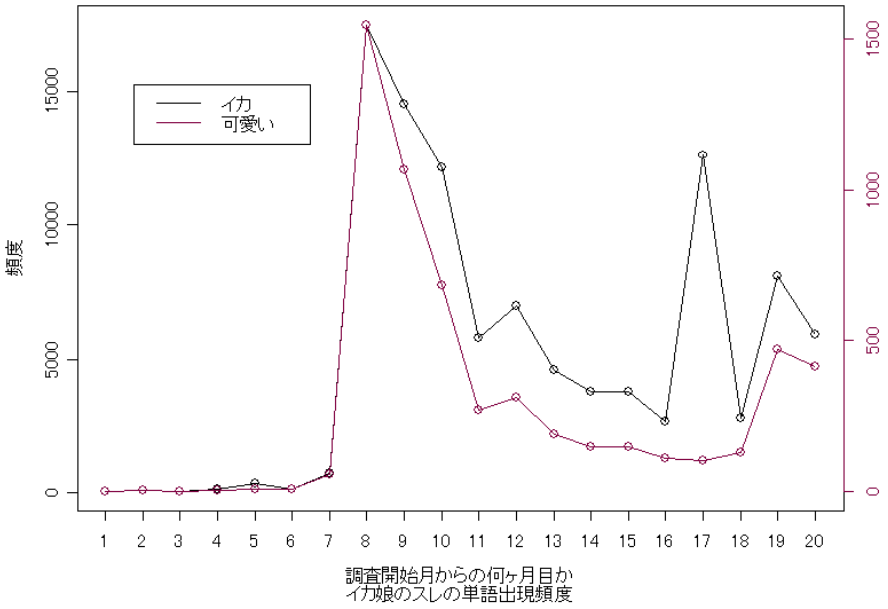

図 2: イカ娘のスレの単語出現頻度

4.2 相互相関関数によって、時系列に相関具合いを求める

2 変数の相関を求める時には、相関係数を利用するが、時系列に相関を調べる時には別の手段によって相関を求める。それが相互相関関数である。これは、次のような数奇によって求めることが出来る。離散的なデータを対象にする場合、

$$(f \star g)[n] \stackrel{\text{def}}{=} \sum_{m=-\infty}^{\infty} f^{\star}[m] g[n+m] \tag{1}$$

n は時間を示し、m は時間のズレ (lag) を意味している。変数が一つの場合は自己相関関数と呼ばれていて。S-PLUS では acf という関数を使うことによって求めることが出来るのだが、相互相関関数を求める関数が事前に用意されていなかった。そのため、これを求める関数を自ら実装することによって、分析を可能にした。図 2 で示した「イカ」と「可愛い」という語に対して、相互相関関数を求めて、その相関具合いを確認する。その図が**??**である。

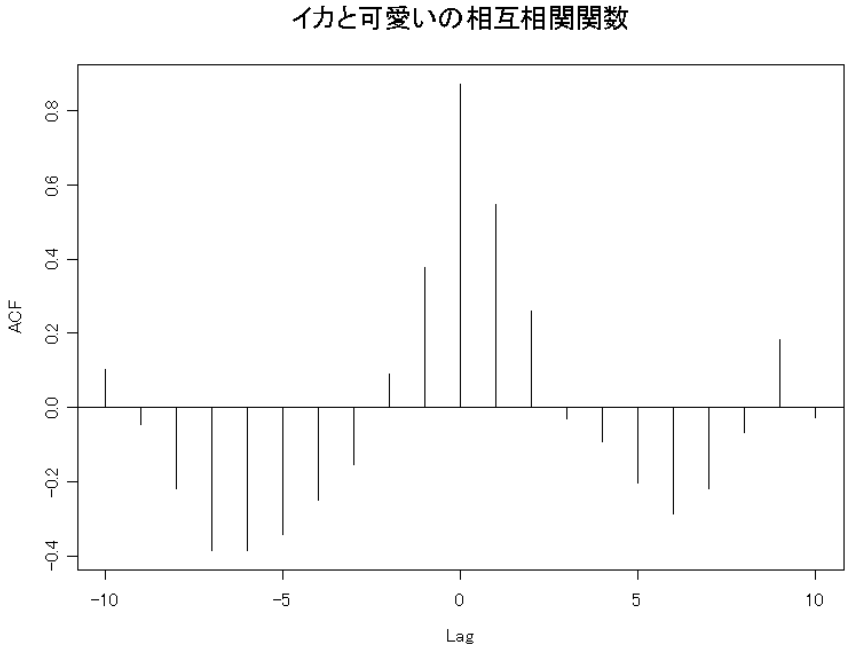

図 3: イカと可愛いの相互相関関数

図**??**の縦軸は、相関具合いを示していて、-1 から 1 の値を取る。1 に近いほど正の相関が強いことを示していて、-1 に近いほど負の相関が強いことを意味している。これによって、折れ線グラフから人の目によって、それぞれの変数に相関があるかどうかを行なっていた判断を定量的に行うことができ、他の変数との比較を容易に行うことが出来る。

4.3 相互相関関数を用いて、関係の強い語を抽出する

表 1 ：相互相関の高いまたは低いもの

0.9以上	-0.8以下
ネタ 、キャラ	こと 、差
作品 、声	こと 、違い
平成 、年月日	こと 、反省
平成 、発売	こと 、点
平成 、特典	人 、イカ
悪い 、差	日 、そう
悪い 、違い	日 、原作
悪い 、点	平成 、感じ
悪い 、反省	平成 、原作
差 、反省	年月日 、原作
差 、点	違い 、こと
違い 、差	発売 、原作
違い 、反省	ん 、イカ
違い 、点	特典 、原作
発売 、日	
発売 、年月日	
発売 、特典	
方 、的	
反省 、点	
特典 、年月日	

自作した相互相関関数によって、算出されたものに対して関係の強いものを抽出したものが表 1 である。その際に、Lag が前後一ヶ月まで許容し、それが 0.9 以上もしくは 0.8 以下のものを抽出した。マイニングという特性上、探索的にこれらの閾値を設定することによって、いくつかに絞り出すことが出来た。これらの中にあるいくつかの変数の組み合わせに対して、相互相関関数を使用して、グラフをプロットした。それが図 4 である。「ネタ」と「キャラ」という語の組み合わせでは、非常に強い相互相関があることが分かる。左右対称に広がるように描かれていることから、それぞれが持続的に同じような場面で使われていると分かる。「悪い」と「反省」という組み合わせでは、lag が 0 の部分に強く相互相関が現れいてその他では非常に相互相関が弱いので、局所的にかつ同じ場面で使われていて、他の時間ではあまり使われていないことが分かる。「原作」と「発売」では、相互相関が負の値を示していて、左右に広がるようにあることから、それぞれが同じような場面ではあまり使われていないことが分かる。「イカ」と「人」という組み合わせでは、全く別の動きを示していて、「イカ」と「人」が別の存在であることをここから読み取ることが出来る。

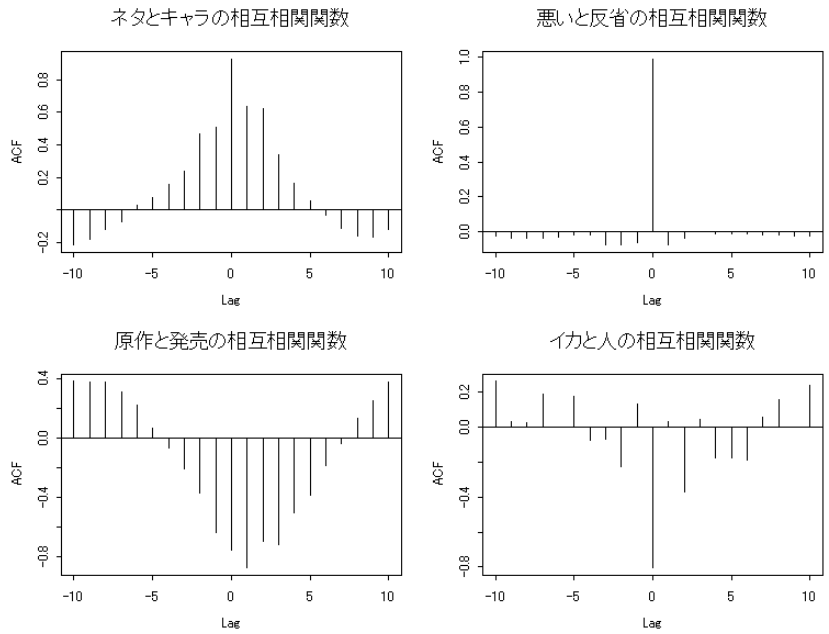

図 4: 相互相関関数による可視化

5　主成分分析による考察

主成分分析を行うことによって、多くの変数の意味を合わせ持つ主成分を求める事によって、テキスト情報の特徴を掴むことが出来る。月ごとのデータに対して主成分分析を行うのであるが、それぞれの月に含まれる語の頻度が大きく異なる。各月を平等の重みにするためには、それぞれの相対度数を利用する方法が相応しいと言える。なぜなら、全体に占める割合をとるので、テキストの規模の影響を受けることがないからである。その結果を図 5 に示した。

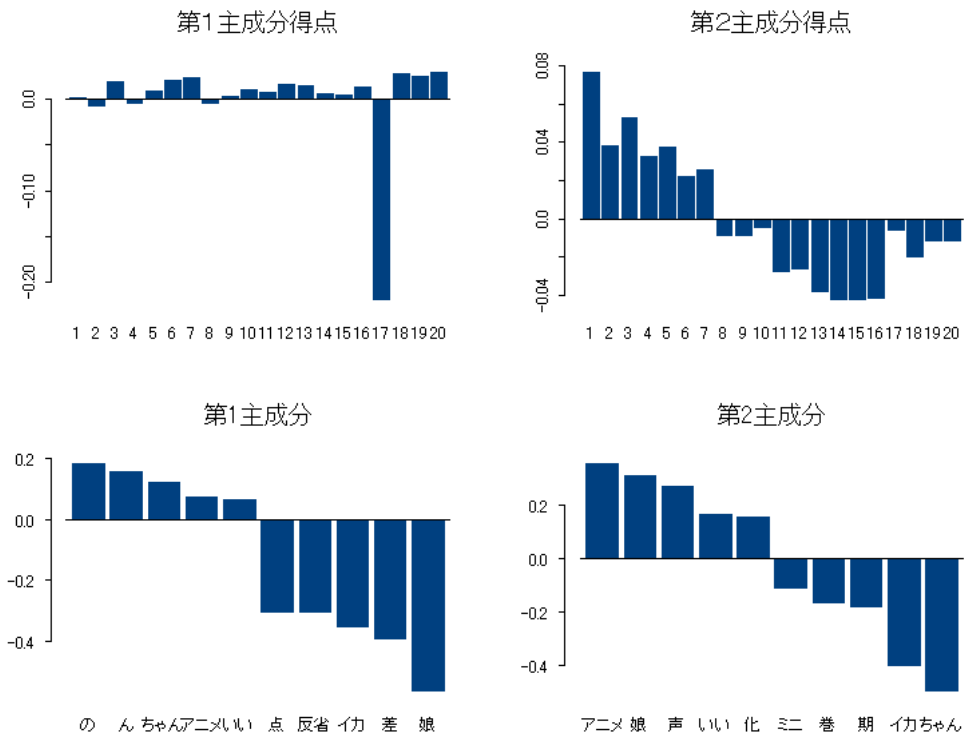

図 5: 主成分分析による第一第二の主成分得点と上位 5 個下位 5 個の因子

表 2: 標準偏差と寄与率・累積寄与率の表

	第一主成分	第2主成分	第3主成分
標準偏差	0.0525	0.0345	0.0173
寄与率	0.5507	0.2374	0.0601
累積寄与率	0.5507	0.7881	0.8482

まず、寄与率を確認する。第 2 主成分までで累積寄与率が 0.7881 を示していることから、第 2 主成分までで十分な精度を持っていると分かる。次に、第 1 主成分を見ると、下位 5 個に「反省」「点」や「イカ」「娘」という語が含まれていることより、第 1 主成分がイカ娘に対するプラスイメージであると解釈できる。すると、主成分得点が圧倒的に小さい 17ヶ月目に行われたオリジナルくじによるキャンペーンが上手く行われていなかったと判断出来る。第 2 主成分では、「アニメ」「化」「声」「いい」という語より、アニメ化に対する期待度と解釈することが出来る。すると、一期の放送開始まで高い期待が寄せられていることが分かり、放送開始にしたがって主成分得点が減少していることが理解出来る。ニ期放送前では、再び増加し、期待が寄せられていたことが分かる。

6　終わりに

インターネット上の掲示板にあるテキストデータを時系列に分析する解析例を示した。時系列に分析することによって、それぞれの月毎に現れる特徴を発見することが出来、スレッドの盛り上がり具合いも知ることが出来た。その盛り上がりが急上昇する点に置いては、それぞれに理由があることが分かった。つまり、施策に対して、どの程度の効果があるかを判断する材料に一つになると言えるだろう。また、それぞれの語の相互相関を知ることによって、それぞれの変数がどのような関係にあるかを知ることが出来るので、期待する効果を上げるにはどのようなアクションを起こすことが効果的であるかを知る手がかりになる。また、対象とするテキスト全体が時系列でどのような特徴があるかを知るときに、主成分分析を用いることによって、影響度の強い説明変数からその主成分の意味を解釈することができるので、新たに得られた主成分で主成分得点をみるとそれが時系列でどの部分にどのように影響があり推移したを知ることが出来るので、問題の再確認や知見の獲得を行うことが出来る。

参考文献

[1] 水田正弘・山本義郎・南弘征・田澤司,S-PLUS によるデータマイニング入門, 森北出版株式会社

[2] JIN'S PAGE,http://www1.doshisha.ac.jp/~mjin/R/

[3] 単語記事:　ネタバレ:イカ娘かわいい,http://dic.nicovideo.jp/a/%E3%83%8D%E3%82%BF%E3%83%90%E3%83%AC%3A%E3%82%A4%E3%82%AB%E5%A8%98%E3%81%8B%E3%82%8F%E3%81%84%E3%81%84

A　折れ線グラフによって月ごとの特徴を捉える時に使用

```
zikeiretu <- structure(c(985, 1812, 1624, 2688, 7683, 2592, 13899,
 225664, 218541, 196558, 85159, 109738, 68796, 53159, 54207, 39708,
 86626, 52512, 143180, 108365), .Names = c(‘‘1’’, ‘‘2’’, ‘‘3’’, ‘‘4’’,
‘‘5’’, ‘‘6’’, ‘‘7’’, ‘‘8’’, ‘‘9’’, ‘‘10’’, ‘‘11’’, ‘‘12’’, ‘‘13’’,
 ‘‘14’’, ‘‘15’’, ‘‘16’’, ‘‘17’’, ‘‘18’’, ‘‘19’’, ‘‘20’’))
par(xaxt=’’n’’)
plot(zikeiretu,type=’’p’’,ylab=’’ 頻度’’,xlab=’’ 調査開始月からの何ヶ月目
 か’’,sub=’’ イカ娘のスレッド内に出現する名詞・形容詞の頻度’’)
lines(zikeiretu,col=3)
par(xaxt=’’s’’)
axis(1,at = c(1:20))

from <- locator(1)
to <- locator(1)
arrows(from$x,from$y,to$x,to$y)
text(from,’’ 一期開始’’)

from <- locator(1)
to <- locator(1)
arrows(from$x,from$y,to$x,to$y)
text(from,’’ 二期開始’’)

from <- locator(1)
to <- locator(1)
arrows(from$x,from$y,to$x,to$y)
text(from,’’ 二期決定’’)

from <- locator(1)
to <- locator(1)
arrows(from$x,from$y,to$x,to$y)
text(from,’’ オリジナルくじ販売’’)
```

B　イカと可愛いの折れ線グラフ

```
par(xaxt=''n'')
plot(c(1:20),ika$word90,type=''p'',col=1,xlab='' 調査開始月からの何ヶ月目
 か'',ylab='' 頻度'',sub='' イカ娘のスレの単語出現頻度'')
par(new=T)
plot(c(1\maketitle20),ika$word90,type=''l'',col=1,ylab='''',xlab='''')
par(new=T)
plot(c(1:20),ika$word29,type=''p'',col=3,xlab='''',ylab='''',labels=FALSE,axes=FALSE)
par(new=T)
plot(c(1\maketitle20),ika$word29,type=''l'',col=3,xlab='''',ylab='''',
labels=FALSE,axes=FALSE)
axis(side=4,col=3)
par(xaxt=''s'')
axis(1,at = c(1:20))
legend(locator(1),c(``イカ'','' 可愛い''),lty=1,col=c(1,3))
```

C　実装した相互相関関数

```
myccf <- function(a,b,mainname){
aa <- ts(a)
bb <- ts(b)
x <- ts.intersect(aa,bb)
xx <- acf(x, plot=F)
dat <- c(rev(xx$acf[-1, 1, 2]), xx$acf[, 2, 1])
d <- data.frame(lag=round(c(rev(xx$lag[-1, 1, 2]), xx$lag[, 2,1]),3),acf=dat)
plot(d$lag,d$acf,type="n",ylab="ACF",xlab="Lag",main=mainname)
abline(h=0)
for ( i in 1:length(d$acf)){
 lines(c(d$lag[i],d$lag[i]),c(0,d$acf[i]))
}
return(d)
}
```

D　相互相関関数による分析

```
wariai100 <- read.table("100_wariais.txt",sep=",",head=T)
sortlist <- order(wariai100$word101,decreasing=FALSE)
wariai100 <- wariai100[sortlist,]
#rownames(wariai100) <- c(1:nrow(wariai100))



for ( i in 2:101){
  for ( j in 2:101){
    a <- myccf(wariai100[i],wariai100[j],mainname =
paste(names(wariai100[i]),"&",names(wariai100[j])))
    if ( max(a$acf[c(10,11,12)]) >= 0.8 && i != j){
      cat(paste(names(wariai100[i]),"&"
                , names(wariai100[j]),"\n"))
    }
    if ( min(a$acf[c(10,11,12)]) <= -0.8 && i != j){
      cat(paste(names(wariai100[i]),"|"
                , names(wariai100[j]),"\n"))

    }
  }
}
par(mfrow=c(2,2))
myccf(wariai100$word10,wariai100$word58,mainname="ネタとキャラの相互相関関数")
myccf(wariai100$word54,wariai100$word73,mainname="悪いと反省の相互相関関数")
myccf(wariai100$word100,wariai100$word68,mainname="原作と発売の相互相関関数")
myccf(wariai100$word89,wariai100$word29,mainname="イカと人の相互相関関数")
```

E　主成分分析とその作図

```
wariai100 <- read.table("100_wariais.txt",sep=",",head=T)
sortlist <- order(wariai100$word101,decreasing=FALSE)
wariai100 <- wariai100[sortlist,]

par(mfrow=c(2,2))
ika.pc<-princomp(wariai100[1:100],cor=F)
barplot(ika.pc$scores[,1],main='' 第 1 主成分得
点'',names=paste(1\maketitle20))
barplot(ika.pc$scores[,2],main='' 第 2 主成分得点'',names=paste(1:20))

sortlist <- order(ika.pc$coef[,1],decreasing=FALSE)
dat <- ika.pc$coef[,1]
barplot(c(dat[sortlist[100:96]],dat[sortlist[5:1]]),las=2,names=c(``の
'','' ん'','' ちゃん'','' アニメ'','' いい'','' 点'','' 反省'','' イ
カ'','' 差'','' 娘''),main='' 第 1 主成分'')

sortlist <- order(ika.pc$coef[,2],decreasing=FALSE)
dat <- ika.pc$coef[,2]
barplot(c(dat[sortlist[100:96]],dat[sortlist[5:1]]),las=2,names=c(``アニ
メ'','' 娘'','' 声'','' いい'','' 化'','' ミニ'','' 巻'','' 期'','' イカ'','' ちゃ
ん''),main='' 第 2 主成分'')
```