インターネット計測とデータ解析 第6回

長 健二朗

2012年5月18日

前回のおさらい

多様性と複雑さ

- ロングテール
- Web アクセスとコンテンツ分布
- べき乗則と複雑系
- 演習: べき乗則解析

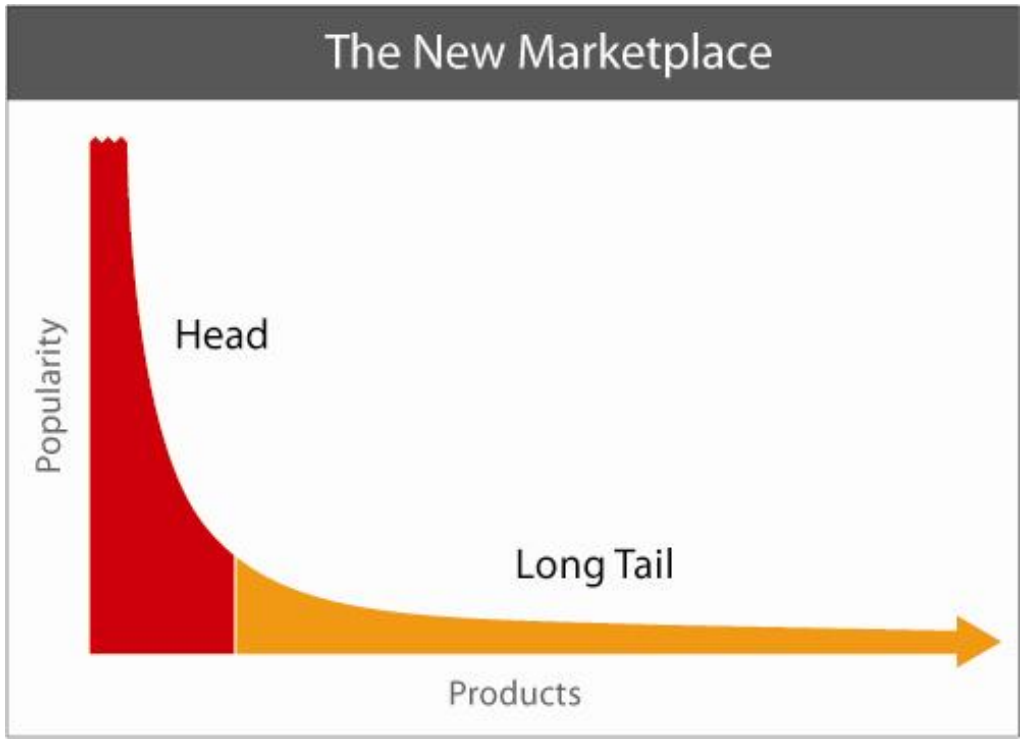
今日のテーマ

相関

- オンラインお勧めシステム
- 距離と類似度
- 相関係数
- 演習: 相関

オンラインお勧めシステム

- EC サイトにおけるロングテールのユーザの潜在ニーズ
  - ユーザの嗜好に合った商品を提示して購買に繋げる
- レコメンダーパッケージによる導入コスト低下で普及



source: http://longtail.com/

お勧めシステムの技術

- ユーザの行動を観察して有用な情報を予測して自動的に提示
- EC サイト: 購買履歴や閲覧履歴からお勧め商品を自動的に提示
- EC サイトだけでなく検索予測、かな漢字変換などへの応用も

データベースの作り方

- アイテムベース: アイテムごとに情報をまとめる
- ユーザベース: ユーザごとに情報をまとめる
- 実際にはこれらを組み合わせて使う

お勧めシステムの予測手法

- コンテンツベース:
  - ユーザが過去に利用したアイテムから類似アイテムを推薦
    - アイテムの属性分類
    - 機械学習クラスタリングによるグループ化
    - ノウハウのルール化
  - 比較的狭い範囲での推薦になりがち、意外性が低い
- 協調フィルタリング: amazon をはじめ広く利用されている
  - 購買履歴から顧客間の類似度を計算
  - 類似したユーザの実績から共通度の高いアイテムを推薦
  - 特徴: 個別のアイテムに関する情報は使わない
  - 思いがけない発見 (serendipity) の可能性
- 単純ベイズ分類器: スパム判定と同じ原理
  - アイテムやユーザに関する個別の多様な情報から確率計算、機械学習する

協調フィルタリング (collaborative filtering)

- 複数のアルゴリズムが存在
- シンプルなユーザ間相関分析
  - ユーザ間の相関をとり類似ユーザを抽出
  - 類似ユーザの類似度を重みに各アイテムの合計点数を計算

例: ユーザの購買履歴

	item						
user	a	b	c	d	e	f	···
A	1		1		1		···
B			1	1			···
C	1	1					···
D	1		1		1		···
···							···

A と相関高いユーザから A の持っていないアイテムのスコアを計算

	similarity	item						
user	σ	a	b	c	d	e	f	···
A	1	1		1		1		···
S	0.88		0.88		-		0.88	···
C	0.81		0.81		-		-	···
K	0.75		-		-		-	···
F	0.73		0.73		0.73		0.73	···
score			2.50		0.73		1.61	···

距離について

いろいろな距離

- ユークリッド距離 (Euclidean distance)
- 標準化ユークリッド距離 (standardized Euclidean distance)
- ミンコフスキー距離 (Minkowski distance)
- マハラノビス距離 (Mahalanobis distance)

類似度

- バイナリベクトルの類似度
- n 次元ベクトルの類似度

距離の性質

空間上の 2 点 (x, y) 間の距離 $d(x, y)$:
非負性 (positivity)

$$d(x, y) \geq 0$$

$$d(x, y) = 0 \Leftrightarrow x = y$$

対称性 (symmetry)

$$d(x, y) = d(y, x)$$

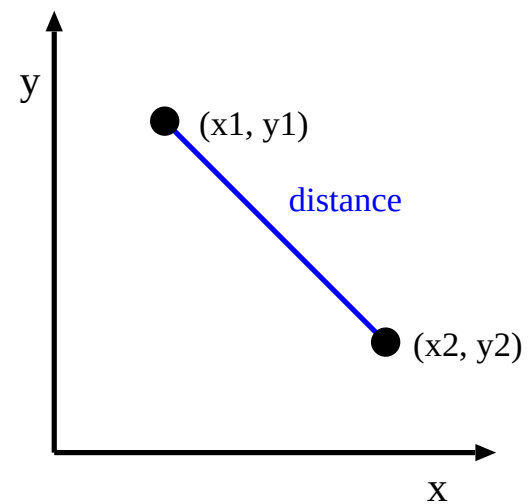
三角不等式 (triangle inequality)

$$d(x, z) \leq d(x, y) + d(y, z)$$

ユークリッド距離 (Euclidean distance)

普通に距離といえばユークリッド距離を指す
n 次元空間での 2 点 (x, y) の距離

$$d(x, y) = \sqrt{\sum_{k=1}^{n} (x_k - y_k)^2}$$

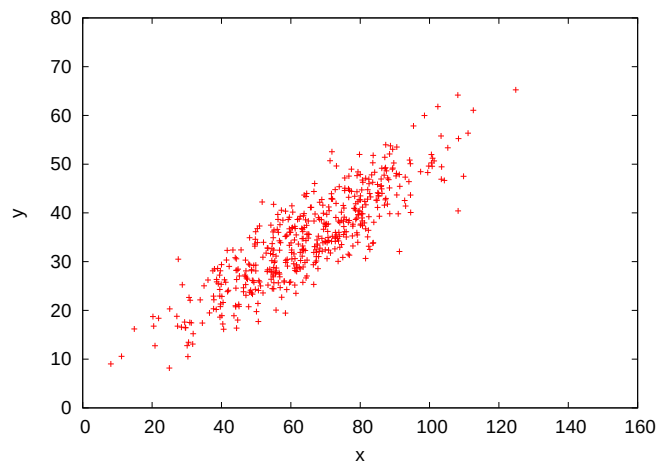


euclidean distance in 2-dimensional space

標準ユークリッド距離 (standardized Euclidean distance)

- 変数間でばらつきの大きさが異なると、距離が影響を受ける
- そこで、ユークリッド距離を各変数の分散で割って正規化

$$d(x,y) = \sqrt{\sum_{k=1}^{n}(\frac{x_k}{s_k} - \frac{y_k}{s_k})^2} = \sqrt{\sum_{k=1}^{n}\frac{(x_k - y_k)^2}{s_k^2}}$$



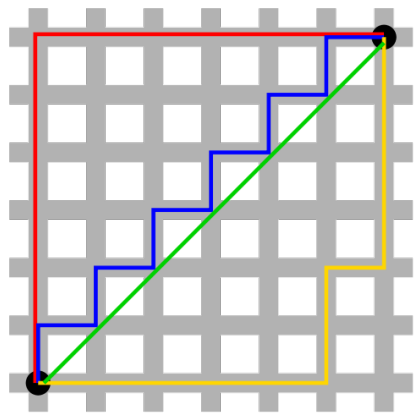
ミンコフスキー距離 (Minkowski distance)

ユークリッド距離を一般化

- パラメータ r が大きいほど、次元軸にとらわれない移動 (斜め方向のショートカット) を重視する距離

$$d(x, y) = \left(\sum_{k=1}^{n} |x_k - y_k|^r\right)^{\frac{1}{r}}$$

- $r = 1$: マンハッタン距離
  - ハミング距離: 2 つの文字列間の同じ位置の文字の不一致数
  - 例えば、111111 と 101010 のハミング距離は 3
- $r = 2$: ユークリッド距離



Manhattan distance vs. Euclidean distance

マハラノビス距離 (Mahalanobis distance)

変数間に相関がある場合に、相関を考慮した距離

$$mahalanobis(x, y) = (x - y)\Sigma^{-1}(x - y)^T$$

ここで Σ^{-1} は共分散行列の逆行列

類似度

類似度

- ふたつのデータの似ている度合の数値表現

類似度の性質
非負性 (positivity)

$$0 \leq s(x, y) \leq 1$$

$$s(x, y) = 1 \Leftrightarrow x = y$$

対称性 (symmetry)

$$s(x, y) = s(y, x)$$

三角不等式 (triangle inequality) は一般に類似度には当てはまらない

バイナリベクトルの類似度

Jaccard 係数

- 1 の出現が少ないバイナリベクトル同士の類似度に使われる
- 文書中に出現する単語から文書の類似度を示す場合など
- 多くの単語は両方ともに出現しない $\Rightarrow$ これらは考慮しない
- 2 つのベクトルの各要素の対応関係を表のように集計

		vector y	
		1	0
vector x	1	n_{11}	n_{10}
	0	n_{01}	n_{00}

Jaccard 係数は以下で表される

$$J = \frac{n_{11}}{n_{11} + n_{10} + n_{01}}$$

n 次元ベクトルの類似度

一般のベクトルの類似度

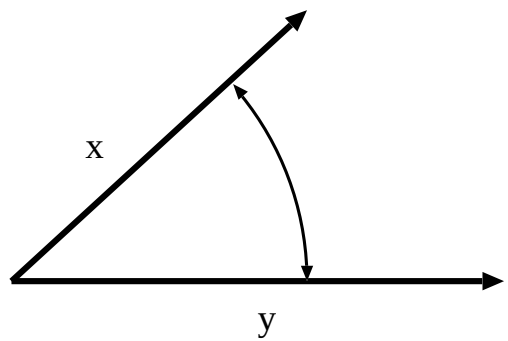
- 文書の類似度で出現頻度も考慮する場合など

コサイン類似度

- ベクトルの x, y の cosine を取る、向きが一致:1、直交:0、向きが逆:-1
- ベクトルの長さで正規化 ⇒ 大きさは考慮しない

$$cos(x, y) = \frac{x \cdot y}{\|x\|\|y\|}$$

$x \cdot y = \sum_{k=1}^{n} x_k y_k$: ベクトルの積
$\|x\| = \sqrt{\sum_{k=1}^{n} x_k^2} = \sqrt{x \cdot x}$: ベクトルの長さ



コサイン類似度の例題

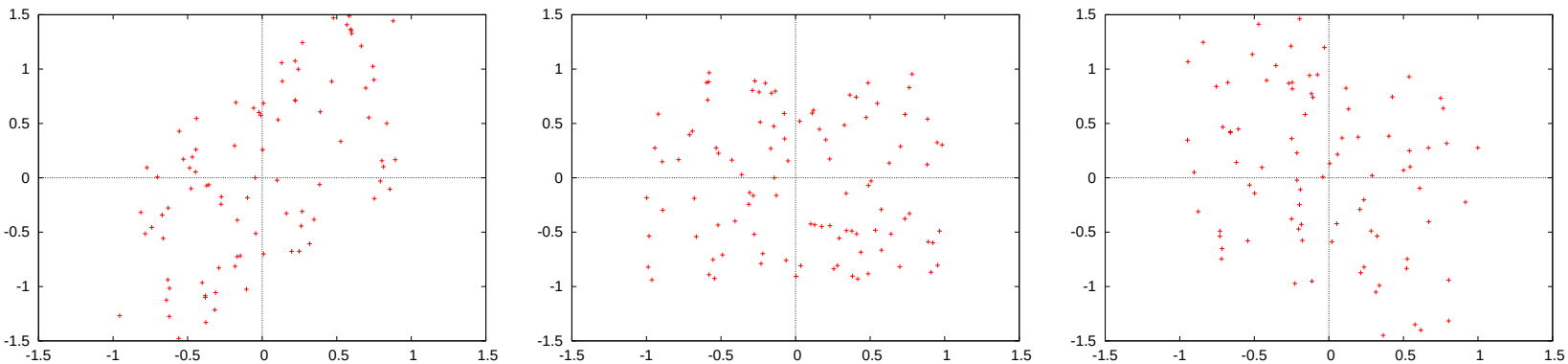
$$x = 3 \quad 2 \quad 0 \quad 5 \quad 0 \quad 0 \quad 0 \quad 2 \quad 0 \quad 0$$
$$y = 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 2$$

$$x \cdot y = 3 * 1 + 2 * 1 = 5$$
$$\|x\| = \sqrt{3 * 3 + 2 * 2 + 5 * 5 + 2 * 2} = \sqrt{42} = 6.481$$
$$\|y\| = \sqrt{1 * 1 + 1 * 1 + 2 * 2} = \sqrt{6} = 2.449$$

$$cos(x, y) = \frac{5}{6.481 * 2.449} = 0.315$$

散布図と相関係数

- 散布図は 2 つの変数の関係を見るのに有効
  - X 軸: 変数 X
  - Y 軸: それに対応する変数 Y の値
- 散布図で分かる事
  - X と Y に関連があるか
    - 無相関、正の相関、負の相関
  - 外れ値の存在があるか
- 相関係数: 相関の方向 (正負) と強さを表す量



例: (左) 正の相関 0.7 (中) 無相関 0.0 (右) 負の相関 -0.5

相関 (correlation)

- 共分散 (covariance):

$$\sigma_{xy}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

- 相関係数 (correlation coefficient):

$$\rho_{xy} = \frac{\sigma_{xy}^2}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- 相関係数は共分散を正規化したもの。 -1 から 1 の値を取る。
- 相関係数は外れ値の影響を大きく受ける。 散布図と併用し、外れ値を確認する必要。
- 相関関係と因果関係
  - 相関関係が因果関係を示すとは限らない。
    - 未知の第 3 の共通の要因が存在する場合
    - 単なる偶然

相関係数の計算 (1)

偏差平方和 (sum of squares)

$$\begin{aligned}
\sum_{i=1}^{n}(x_i - \bar{x})^2 &= \sum_{i=1}^{n}(x_i^2 - 2x_i\bar{x} + \bar{x}^2) \\
&= \sum_{i=1}^{n} x_i^2 - 2\bar{x}\sum_{i=1}^{n} x_i + n\bar{x}^2 \\
&= \sum_{i=1}^{n} x_i^2 - 2\bar{x}\cdot n\bar{x} + n\bar{x}^2 \\
&= \sum_{i=1}^{n} x_i^2 - n\bar{x}^2 = \sum_{i=1}^{n} x_i^2 - \frac{(\sum_{i=1}^{n} x_i)^2}{n}
\end{aligned}$$

偏差積和 (sum of products)

$$\begin{aligned}
\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) &= \sum_{i=1}^{n}(x_i y_i - x_i\bar{y} - \bar{x}y_i + \bar{x}\bar{y}) \\
&= \sum_{i=1}^{n} x_i y_i - \bar{x}\sum_{i=1}^{n} y_i - \bar{y}\sum_{i=1}^{n} x_i + n\bar{x}\bar{y} \\
&= \sum_{i=1}^{n} x_i y_i - \bar{x}\cdot n\bar{y} - \bar{y}\cdot n\bar{x} + n\bar{x}\bar{y} \\
&= \sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y} = \sum_{i=1}^{n} x_i y_i - \frac{(\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} y_i)}{n}
\end{aligned}$$

相関係数の計算 (2)

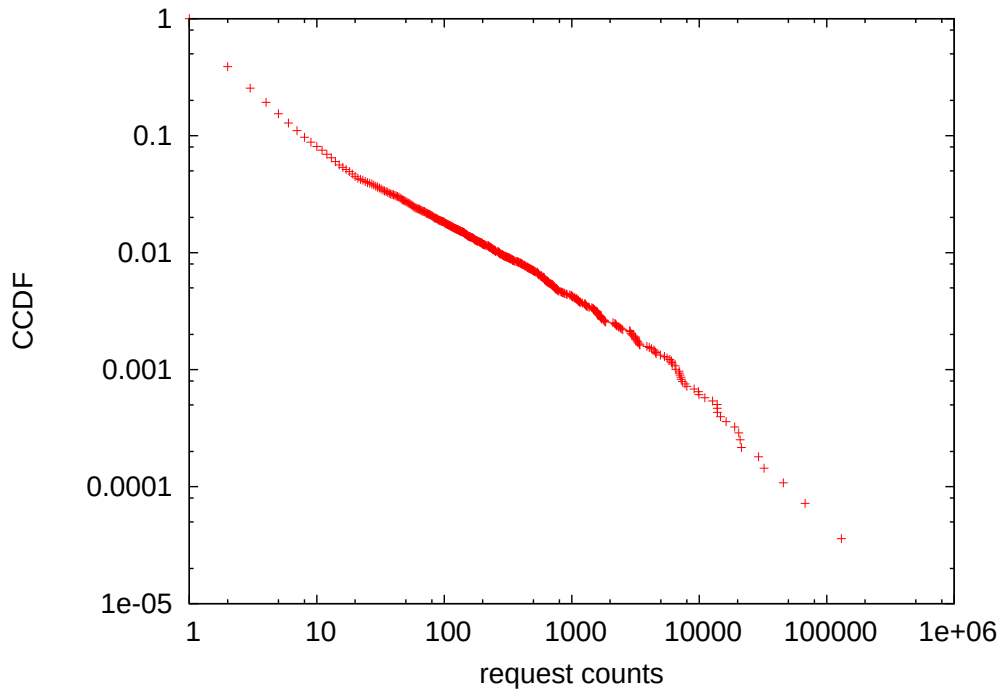
相関係数 (correlation coefficient)

$$
\begin{aligned}
\rho_{xy} &= \frac{\sigma_{xy}^2}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} \\
&= \frac{\sum_{i=1}^n x_i y_i - n\bar{x}\bar{y}}{\sqrt{(\sum_{i=1}^n x_i^2 - n\bar{x}^2)(\sum_{i=1}^n y_i^2 - n\bar{y}^2)}} \\
&= \frac{\sum_{i=1}^n x_i y_i - \frac{(\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i)}{n}}{\sqrt{(\sum_{i=1}^n x_i^2 - \frac{(\sum_{i=1}^n x_i)^2}{n})(\sum_{i=1}^n y_i^2 - \frac{(\sum_{i=1}^n y_i)^2}{n})}}
\end{aligned}
$$

前回の演習: CCDF のプロット

JAIST サーバのアクセスログから、コンテンツ毎のアクセス数分布を CCDF でプロットする

```
% ./count_contents.rb sample_access_log > contents.txt
% ./make_ccdf.rb contents.txt > ccdf.txt
```



コンテンツ毎のアクセス数を集計するスクリプト

```
# output: URL req_count byte_count
# regular expression for apache combined log format
#   host ident user time request status bytes referer agent
re = /^(\S+) (\S+) (\S+) \[(.*?)\] "(.*?)" (\d+) (\d+|-) "(.*?)" "(.*?)"/
# regular expression for request: method url proto
req_re = /(\w+) (\S+) (\S+)/
contents = Hash.new([0, 0])
count = parsed = 0
ARGF.each_line do |line|
  count += 1
  if re.match(line)
    host, ident, user, time, request, status, bytes, referer, agent = $~.captures
    # ignore if the status is not success (2xx)
    next unless /2\d{2}/.match(status)
    if req_re.match(request)
      method, url, proto = $~.captures
      # ignore if the method is not GET
      next unless /GET/.match(method)
      parsed += 1
      # count contents by request and bytes
      contents[url] = [contents[url][0] + 1, contents[url][1] + bytes.to_i]
    else
      # match failed.  print a warning msg
      $stderr.puts("request match failed at line #{count}: #{line.dump}")
    end
  else
    $stderr.puts("match failed at line #{count}: #{line.dump}") # match failed.
  end
end
contents.sort_by{|key, value| -value[0]}.each do |key, value|
  puts "#{key} #{value[0]} #{value[1]}"
end
$stderr.puts "# #{contents.size} unique contents in #{parsed} successful GET requests"
$stderr.puts "# parsed:#{parsed} ignored:#{count - parsed}"
```

前回の演習: アクセス数を CCDF に変換するスクリプト

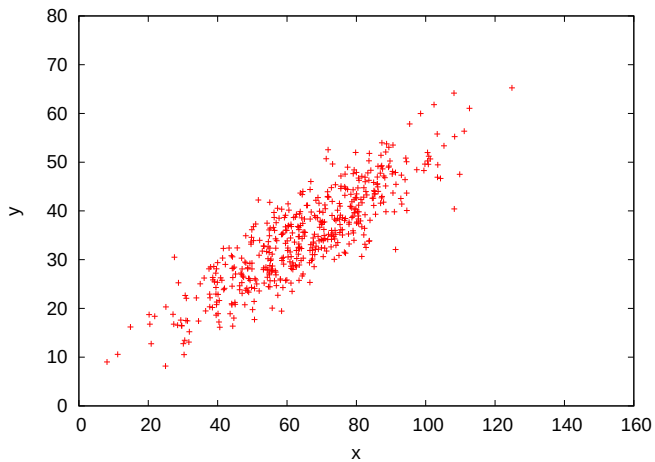
```
#!/usr/bin/env ruby

re = /^\S+\s+(\d+)\s+\d+/

n = 0
counts = Hash.new(0)
ARGF.each_line do |line|
  if re.match(line)
    counts[$1.to_i] += 1
    n += 1
  end
end

cum = 0
counts.sort.each do |key, value|
  comp = 1.0 - Float(cum) / n
  puts "#{key} #{value} #{comp}"
  cum += value
end
```

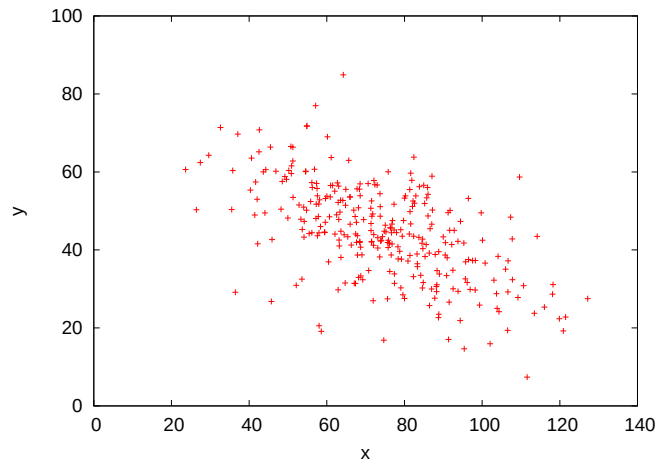
前回の演習: コンテンツ毎のアクセス数を集計するスクリプト

```
set logscale
set xlabel "request counts"
set ylabel "CCDF"

plot "ccdf.txt" using 1:3 notitle with points
```

今回の演習: 相関係数の計算

- データの相関係数を計算する
  - correlation-data-1.txt, correlation-data-2.txt





data-1:r=0.87 (left), data-2:r=-0.60 (right)

演習: 相関係数の計算スクリプト

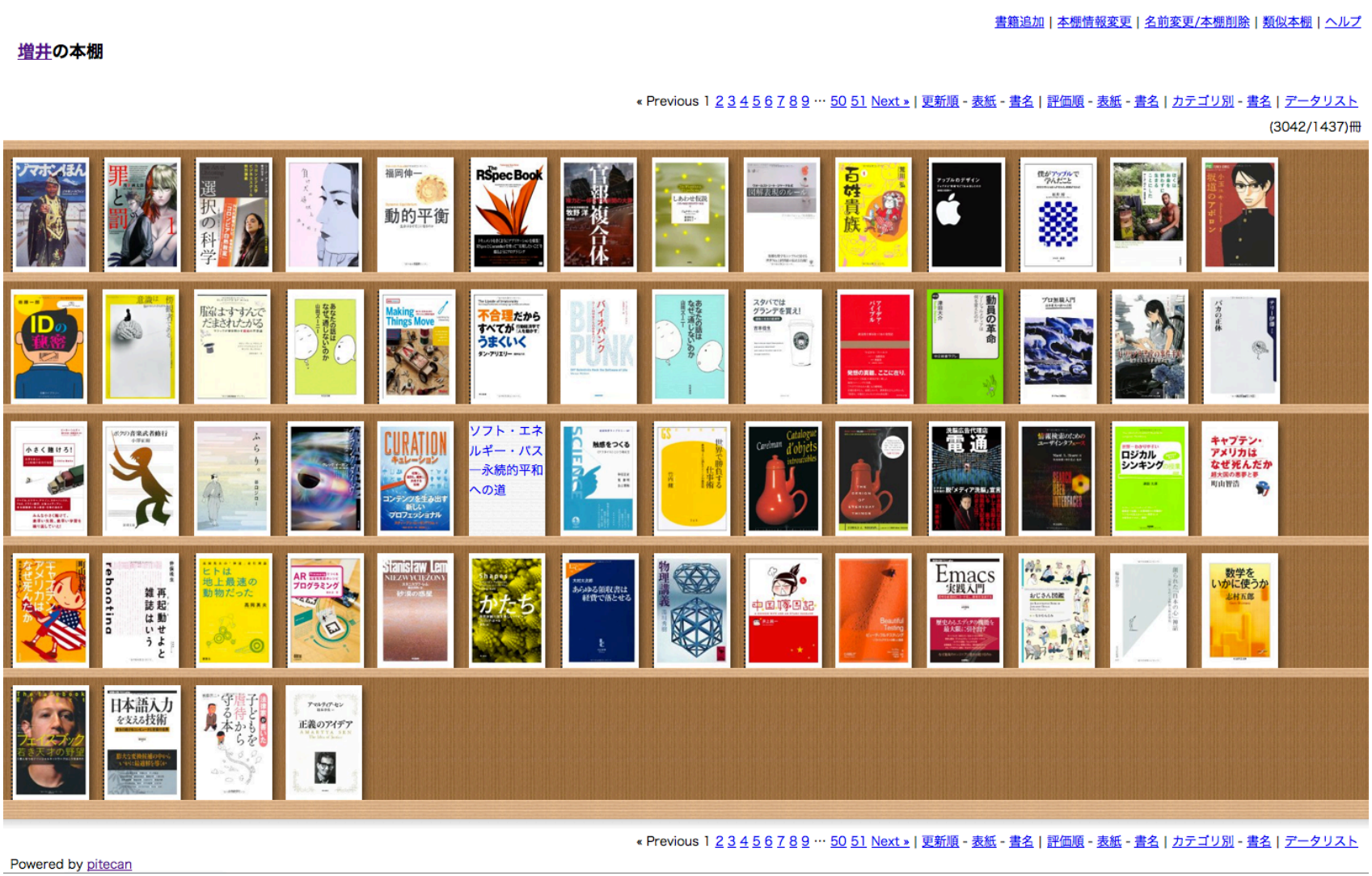
```
#!/usr/bin/env ruby

# regular expression for matching 2 floating numbers
re = /([-+]?\d+(?:\.\d+)?)\s+([-+]?\d+(?:\.\d+)?)/

sum_x = 0.0 # sum of x
sum_y = 0.0 # sum of y
sum_xx = 0.0 # sum of x^2
sum_yy = 0.0 # sum of y^2
sum_xy = 0.0 # sum of xy
n = 0 # the number of data

ARGF.each_line do |line|
    if re.match(line)
      x = $1.to_f
      y = $2.to_f
      sum_x += x
      sum_y += y
      sum_xx += x**2
      sum_yy += y**2
      sum_xy += x * y
      n += 1
    end
end

r = (sum_xy - sum_x * sum_y / n) /
  Math.sqrt((sum_xx - sum_x**2 / n) * (sum_yy - sum_y**2 / n))

printf "n:%d r:%.3f\n", n, r
```

トピック: 本棚.org

増井俊之先生の本棚.org

- 自分の持つ本のリストを共有するサイト

  http://www.hondana.org/



トピック: 本棚演算

本棚演算: 本棚.org のデータに演算を適用し、協調フィルタリング的に本の情報を集める

- 本棚演算のページ
  `http://www.pitecan.com/Enzan/`
  - 2007 年のデータと ruby で書かれたソースコードも
  - 日本語は EUC コード
- 本棚演算を解説した UnixMagazine の原稿
  `www.pitecan.com/UnixMagazine/PDF/if0512.pdf`
- MySQL のデータも公開されている
  - 最新データがダウンロード可能 (約 80MB)
  - 文字コードは UTF-8

まとめ

相関

- オンラインお勧めシステム
- 距離と類似度
- 相関係数
- 演習: 相関

次回予定

第 7 回 多変量解析 (5/25)

- データセンシング
- 線形回帰
- 主成分分析
- 演習: 線形回帰