

ソフトウェア開発プロジェクトデータの 統計解析

門田 暁人

EASE プロジェクト(<http://empirical.jp>)

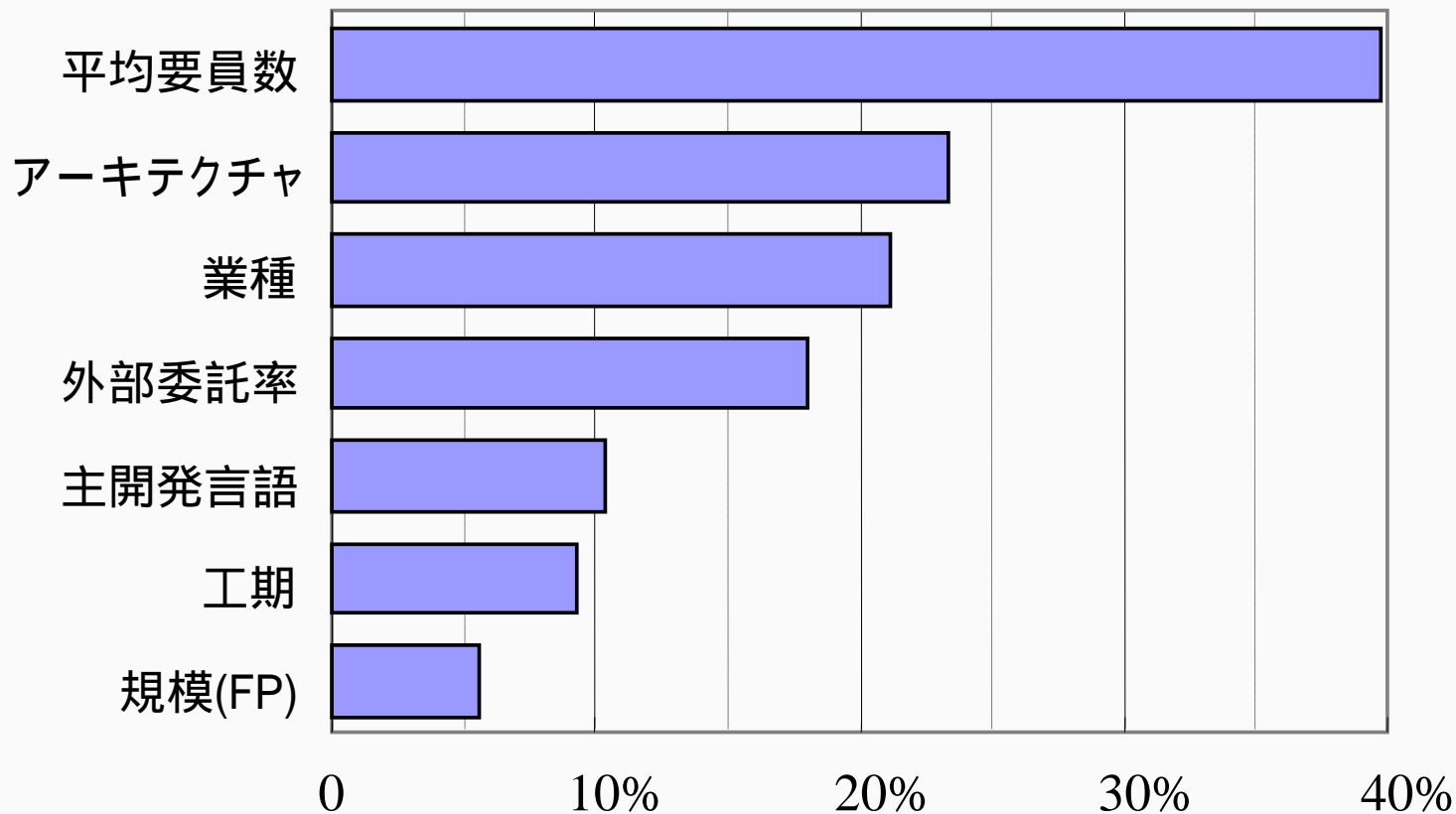
奈良先端科学技術大学院大学 情報科学研究科

分析(統計解析)の目的

- データ間の関係を調べる。
 - 視覚的に
 - 散布図, ヒストグラム, 箱ひげ図, 平行座標プロットなど
 - 定量的に
 - t検定, カイ二乗検定, 分散分析, 無相関検定など
 - 相関係数, クラメールのV, 回帰曲線など
- データの予測(見積もり)を行う。
 - 重回帰分析, 協調フィルタリング, マハラノビスタグチ法など
- 大量のデータの中から隠された関係を発見する。
 - アソシエーション分析(相関ルール分析)

分析結果の例(1)

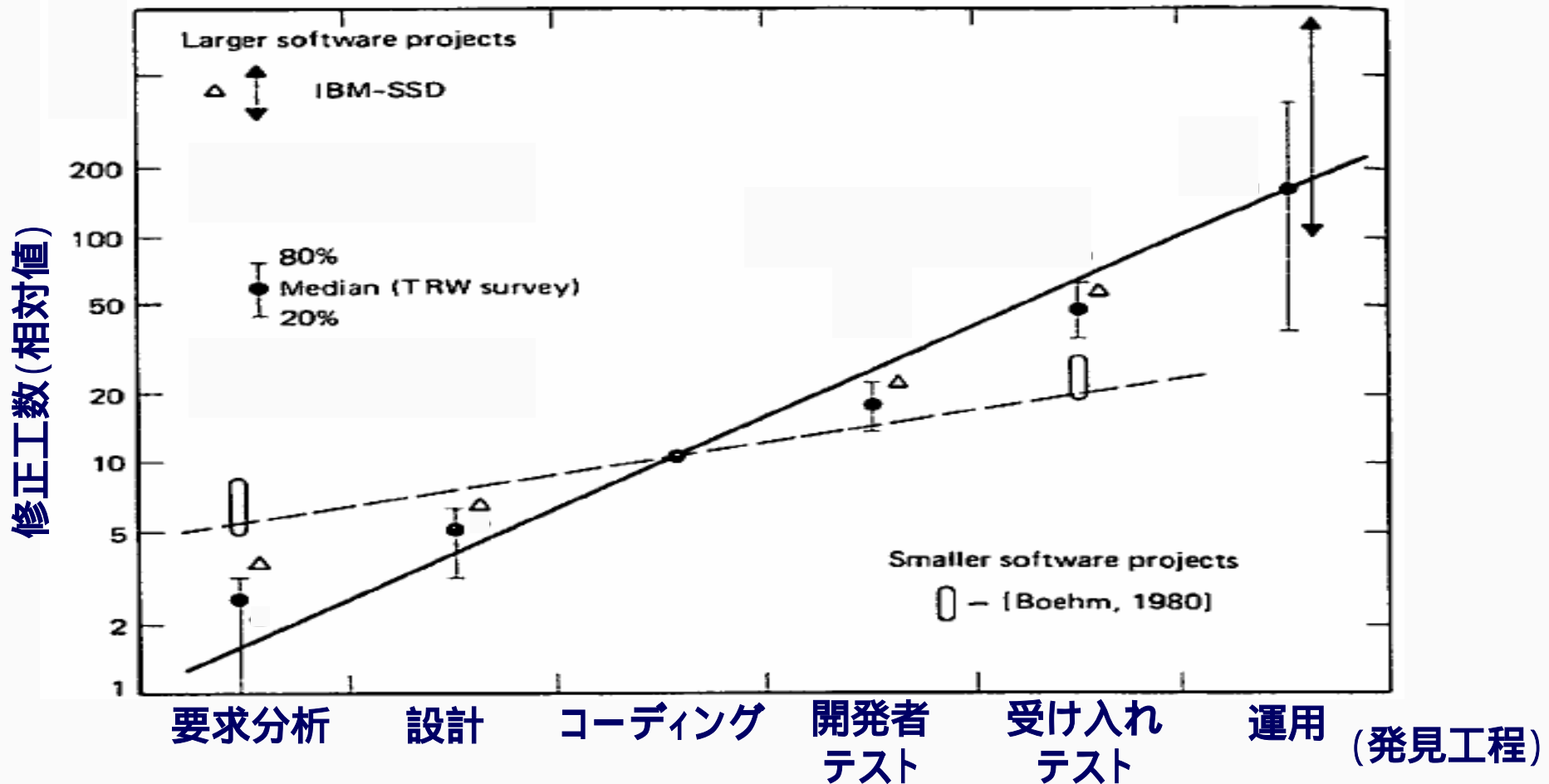
開発の生産性(規模÷工数)に対する寄与率



出典: ソフトウェア開発データ白書2006, 9.2章, pp.160-163, 日経BP

分析結果の例(2)

障害の発見工程—修正工数の関係



出典: B.W. Boehm, Software Engineering Economics, Prentice-Hall, 1981.

ソフトウェア開発プロジェクトデータの例

(注) このデータは架空のものです。

プロジェクトID	開発種別	業種	アーキテクチャ	開発言語(第1言語)	OS
1	a: 新規開発	a: 銀行	a: クライアントサーバ	d: VISUAL BASIC	q: WINDOWS NT
2	a: 新規開発	a: 銀行	b: スタンドアロン	f: PL/I	c: MVS
3	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
4	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
5	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
6	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
7	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
8	a: 新規開発	a: 銀行	c: 混合	d: VISUAL BASIC	c: MVS
...
73	b: 改修・保守	...	c: 混合	c: COBOL	...

要求仕様_明確度合	開発期間(月数)	ピーク要員数	FP計測手法	規模(FP)	開発工数(人時)
	15	15	a: IFPUG	556	24690
c: ややあいまい	8	6	a: IFPUG	80	825
	6	1	a: IFPUG	77	758
a: 非常に明確	4	6	a: IFPUG	255	2119
	6	0	a: IFPUG	349	2741
d: 非常にあいまい	1	3	a: IFPUG	69	1090
b: かなり明確	4	11	a: IFPUG	375	1855
	6		a: IFPUG	271	1747
b: かなり明確	12	4	a: IFPUG	439	2007
	4	9	b: NFGMA	197	999

プロジェクトデータの特徴 - 変数の種類

プロジェクトID	開発種別	業種	アーキテクチャ	開発言語(第1言語)	OS
1	a: 新規開発	a: 銀行	a: クライアントサーバ	d: VISUAL BASIC	g: WINDOWS NT
2	a: 新規開発	a: 銀行	b: スタンドアロン	f: PL/I	c: MVS
3	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
4	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
5	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
6	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
7	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
8	a: 新規開発	a: 銀行	c: 混合	d: VISUAL BASIC	c: MVS
...
73	b: 改修・保守	c: 混合	c: 混合	c: COBOL	c: COBOL

質的データ
(名義尺度)

要求仕様_明確度合	開発期間(月数)	ピーク要員数	FP計測手法	規模(FP)	開発工数(人時)
c: ややあいまい	15	15	a: IFPUG	556	24690
a: 非常に明確	8	6	a: IFPUG	80	825
d: 非常にあいまい	6	1	a: IFPUG	77	758
b: かなり明確	4	6	a: IFPUG	255	2119
b: かなり明確	6	0	a: IFPUG	349	2741
b: かなり明確	3	3	a: IFPUG	66	1090
b: かなり明確	11	11	a: IFPUG	1855	1747
b: かなり明確	4	4	a: IFPUG	2007	2007

質的データ
(順序尺度)

量的データ
(間隔尺度・比尺度)

プロジェクトデータの特徴 - 欠損値の存在

プロジェクトID	開発種別	業種	アーキテクチャ	開発言語(第1言語)	OS
1	a: 新規開発	a: 銀行	a: クライアントサーバ	d: VISUAL BASIC	g: WINDOWS NT
2	a: 新規開発	a: 銀行	b: スタンドアロン	f: PL/I	c: MVS
3	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
4	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
5	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
6	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
7	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
8	a: 新規開発	a: 銀行	c: 混合	d: VISUAL BASIC	c: MVS
...
73	b: 改修・保守			c: COBOL	

欠損値

要求仕様_明確度合	開発種別	業種	開発言語	開発員数	FP計測手法	規模(FP)	開発工数(人時)
				15	a: IFPUG	556	24690
c: ややあいまい				8	a: IFPUG	80	825
				6	a: IFPUG	77	758
a: 非常に明確				4	a: IFPUG	255	2119
				6	a: IFPUG	349	2741
d: 非常にあいまい				1	a: IFPUG	69	1090
b: かなり明確				4	a: IFPUG	375	1855
				6	a: IFPUG	271	1747
b: かなり明確				12	a: IFPUG	439	2007

統計分析における欠損値の取り扱い

- 欠損値をなくす

- 多くの統計手法(重回帰分析などの予測手法や,数量化理論,主成分分析など)では,欠損値のないデータセットを予め作成することが求められる.
- 欠損値処理法
 - 平均値挿入法:欠損値に対して,当該変数の平均値を挿入する.
 - k-NN法:類似するk個のプロジェクトから値を類推して埋める.

- 欠損値をそのままにしておく

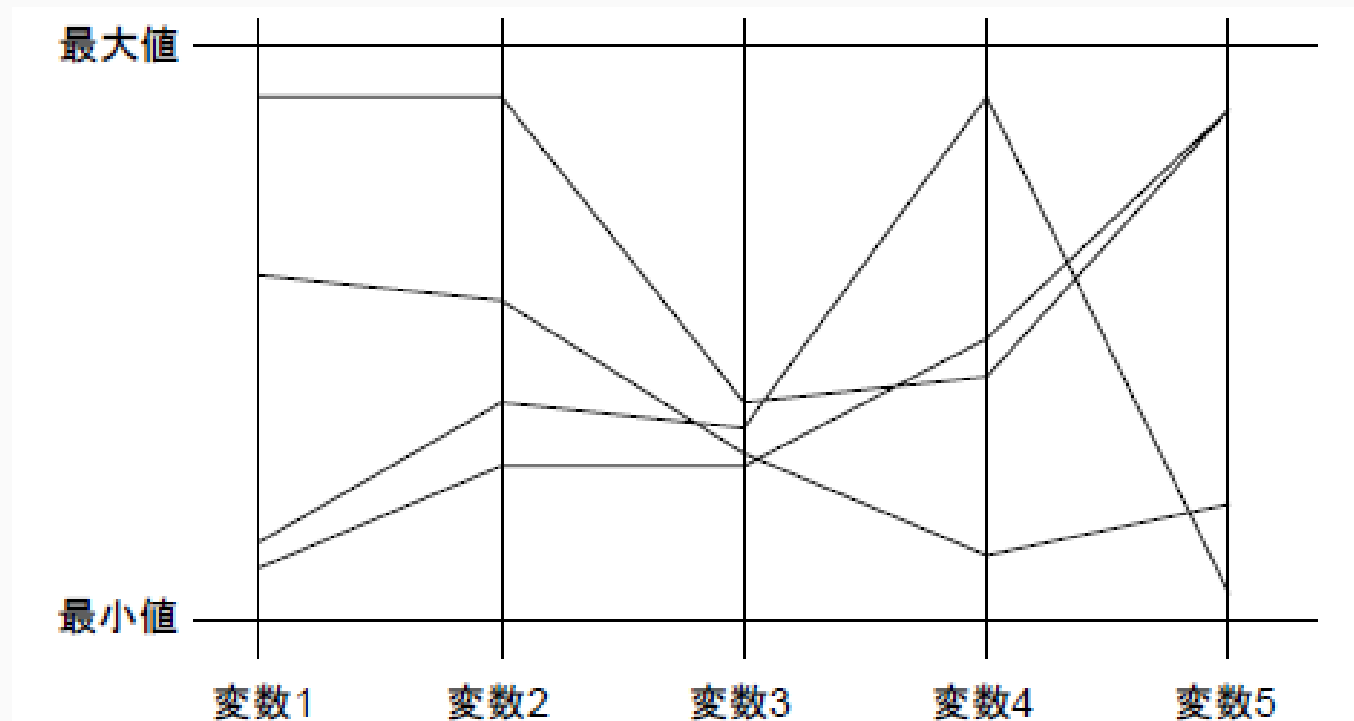
- 欠損値を埋めない方がよいことも多い.
 - 基礎統計量(平均値,中央値,分散など)の算出
 - 散布図,箱ひげ図,ヒストグラムなどのグラフ
 - 相関係数,クラメールのVなどの値
 - アソシエーション分析 など

分析技術

- 平行座標プロット
- 分散分析, 寄与率
- カイ二乗検定, クラメールのV

平行座標プロット(PCP)

- 多変数間の関連を発見するために、複数の変数の関係を視覚化したもの。
 - 軸の最下部が最小値，最上部が最大値を表す。
 - 各プロジェクトの値は軸上にプロットされ，線で結ばれる。



平行座標プロット(PCP) - デモ

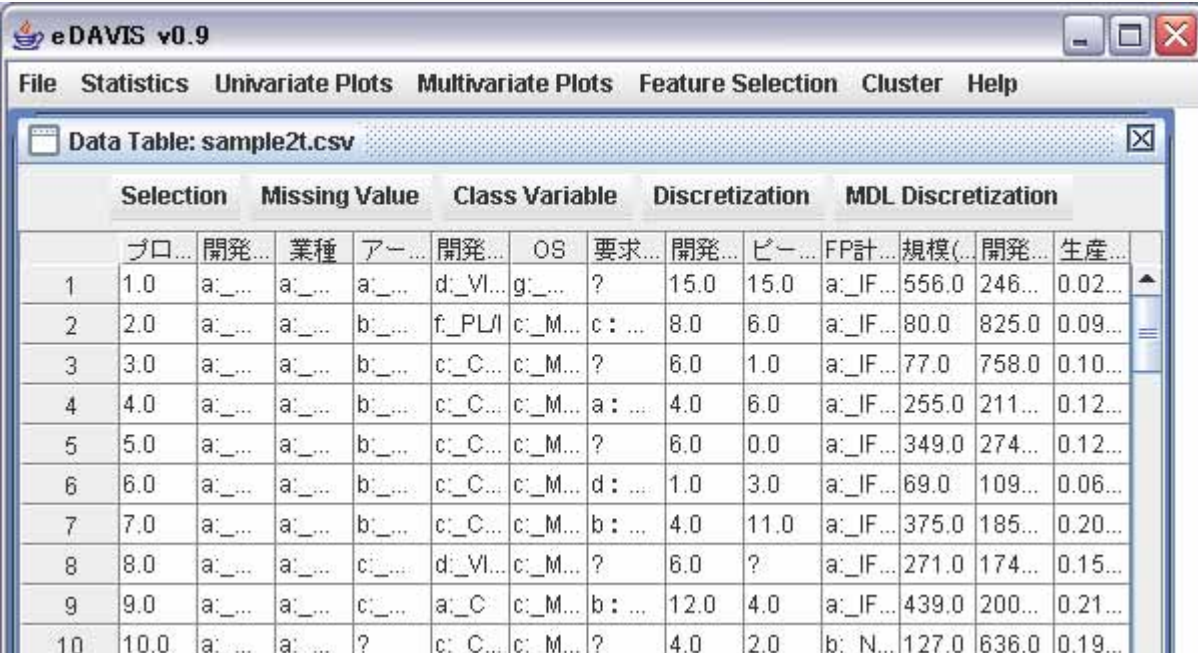
- デモ

- カナダの企業で収集された < 77プロジェクト × 11変数 > の
定量データ

平行座標プロット(PCP) - DAVIS

- DAVIS

- 韓国の成均館(ソングングァン)大学の許文烈氏が開発したフリーのデータ視覚化ツール
- Java VM上で動作する.
- <http://stat.skku.ac.kr/myhuh/>



eDAVIS v0.9

File Statistics Univariate Plots Multivariate Plots Feature Selection Cluster Help

Data Table: sample2t.csv

	Selection	Missing Value	Class Variable	Discretization	MDL Discretization									
	プロ...	開発...	業種	アー...	開発...	OS	要求...	開発...	ピー...	FP計...	規模(...	開発...	生産...	
1	1.0	a:_...	a:_...	a:_...	d:_VI...	g:_...	?	15.0	15.0	a:_IF...	556.0	246...	0.02...	
2	2.0	a:_...	a:_...	b:_...	f:_PL/I	c:_M...	c: ...	8.0	6.0	a:_IF...	80.0	825.0	0.09...	
3	3.0	a:_...	a:_...	b:_...	c:_C...	c:_M...	?	6.0	1.0	a:_IF...	77.0	758.0	0.10...	
4	4.0	a:_...	a:_...	b:_...	c:_C...	c:_M...	a: ...	4.0	6.0	a:_IF...	255.0	211...	0.12...	
5	5.0	a:_...	a:_...	b:_...	c:_C...	c:_M...	?	6.0	0.0	a:_IF...	349.0	274...	0.12...	
6	6.0	a:_...	a:_...	b:_...	c:_C...	c:_M...	d: ...	1.0	3.0	a:_IF...	69.0	109...	0.06...	
7	7.0	a:_...	a:_...	b:_...	c:_C...	c:_M...	b: ...	4.0	11.0	a:_IF...	375.0	185...	0.20...	
8	8.0	a:_...	a:_...	c:_...	d:_VI...	c:_M...	?	6.0	?	a:_IF...	271.0	174...	0.15...	
9	9.0	a:_...	a:_...	c:_...	a:_C	c:_M...	b: ...	12.0	4.0	a:_IF...	439.0	200...	0.21...	
10	10.0	a:_...	a:_...	?	c:_C...	c:_M...	?	4.0	2.0	b:_N...	127.0	636.0	0.19...	

平行座標プロット(PCP) - DAVIS

• 入力

- CSV形式のファイル
- 空白セルを「欠損であることを示す値」で予め埋めておく
- 変数によっては対数変換(log)を予め行っておく

要求仕様_明確度合	ピーク要員数	FP計測手法	規模(FP)	開発工数(人時)	
	15	15	a: IFPUG	556	24690
c: ややあいまい	8	6	a: IFPUG	80	825
	6	1	a: IFPUG	77	758
a: 非常に明確	4	6	a: IFPUG	255	2119
	6	0	a: IFPUG	349	2741
d: 非常にあいまい	1	3	a: IFPUG	69	1090
b: かなり明確	4	11	a: IFPUG	375	1855
	6		a: IFPUG	271	1747
b: かなり明確	12	4	a: IFPUG	439	2007

“e: 欠損値”
で埋める

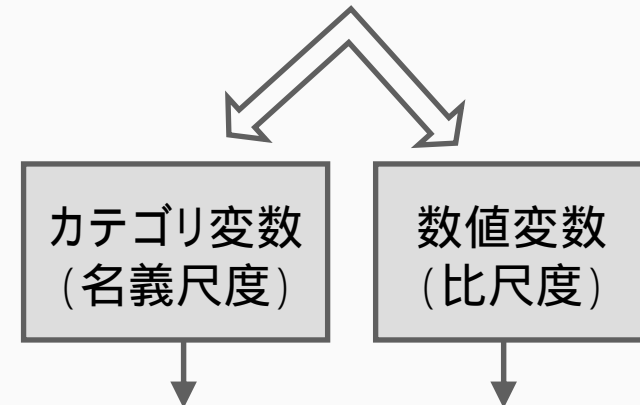
“-5”で埋める

- 多変数の関連を見るのに役立つ。
 - ただし, 8変数ぐらいまでが限界。
- 定量的な分析は別途行う必要がある。
 - 箱ひげ図, t検定, 分散分析

変数間の関係の有無と強さ

- **数値変数 数値変数**
 - 無相関検定, 相関係数 (順位相関係数)
- **カテゴリ変数 数値変数**
 - 分散分析, 寄与率
 - t検定, 平均値の差
- **カテゴリ変数 カテゴリ変数**
 - カイ二乗検定, クラメールのV

関連はあるか?
関連の強さは?



プロジェクト ID	業種	生産性 (FP ÷ 人時)
1	銀行	0.0225
2	製造業	0.0970
3	銀行	0.1016
4	銀行	0.1203
5	製造業	0.1273
6	銀行	0.1835
7	銀行	0.2022
8	公共	0.1551

(一元配置)分散分析

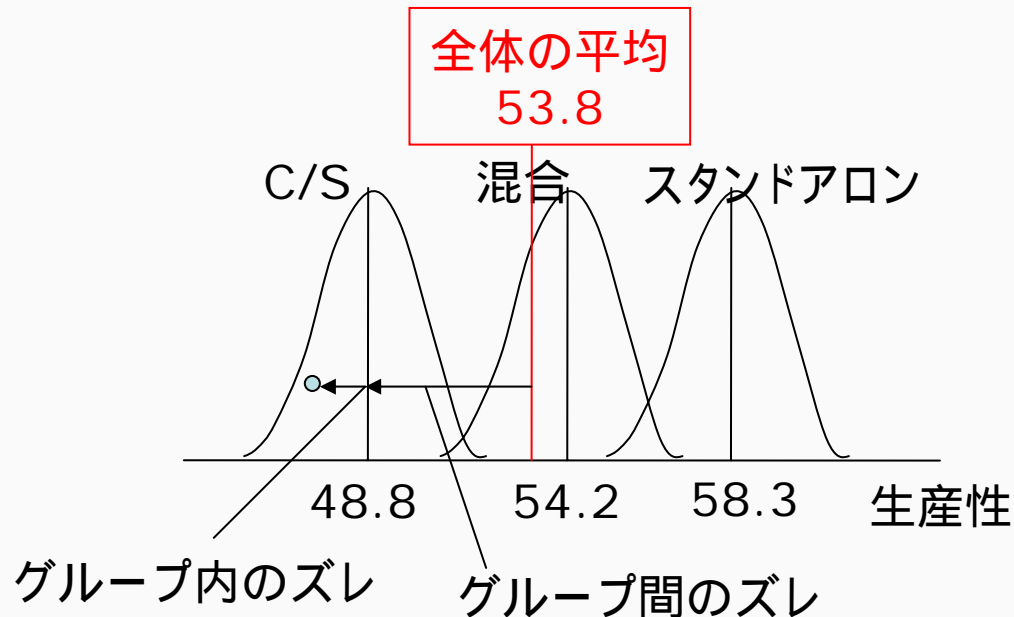
- 2変数間の関連の有無を検定する手法
 - 質的データ(業種,アーキテクチャなど)
 - 量的データ(生産性,開発工数など)
 - 帰無仮説の例:
各アーキテクチャの生産性の平均値は等しい
- p 値
 - 2変数間の関連の有無を検定するための値.
 - 例えば, $p = 0.05$ ならば有意水準5%で仮説は棄却される (= 2変数間に関連がある).
- 寄与率
 - 2変数間の関連の大きさを表す尺度

分散分析 - 寄与率

- 寄与率

$$\text{寄与率} = \frac{\text{グループ間変動}}{\text{グループ内変動} + \text{グループ間変動}}$$

- 0から1の間の実数値を取る.



- ・グループ間のズレが大きいと, 寄与率は大きくなる.
- ・より厳密には, 調整済み寄与率²を用いる方が良い.

分散分析 - 分析事例

• SECとEASEの共同研究プロジェクト

- ソフトウェア開発データ白書2005年に記載の1009プロジェクトのうちの211件（新規開発プロジェクト）

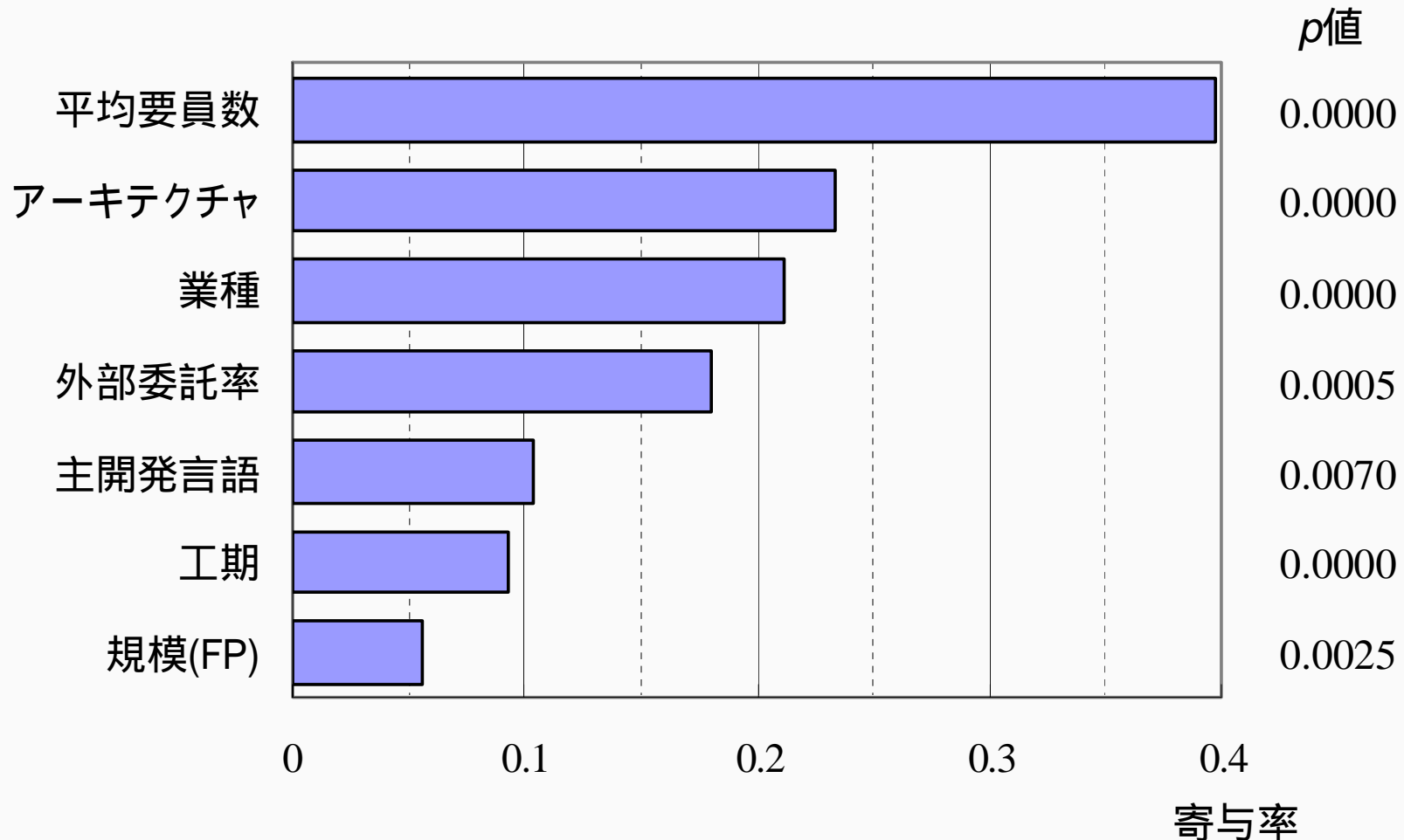
分析データのイメージ

説明変数							目的変数
業種	アーキテクチャ	主開発言語	規模 (FP)	工期 (月数)	平均要員数	外部委託率	生産性 (FP÷人時)
銀行	C/S	PL/I	上位	上位	上位	上位	0.0225
	スタンドアロン	C	下位	中位	中位	下位	0.0970
銀行	C/S	COBOL	下位	中位	下位	下位	0.1016
銀行		Visual Basic	中位		中位	中位	0.1203
製造業	スタンドアロン		中位	下位	中位	中位	0.1273
銀行	混合	C++		下位	上位	上位	0.1835
銀行	C/S	COBOL	上位	下位	下位	下位	0.2022
公共	混合	Java	中位	上位		下位	0.1551

量的データは、値の大きさに応じて下位25%、中位、上位25%に分類した。

出典：ソフトウェア開発データ白書2006, 9.2章, pp.160-163, 日経BP

分散分析 - 分析事例



全て $p < 0.05$ なので、有意水準5%で生産性との関連あり。

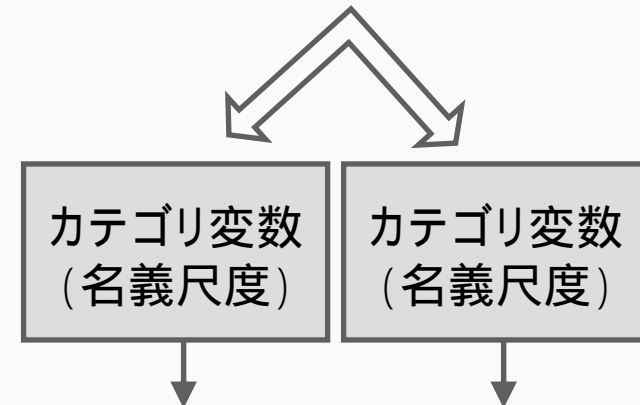
出典：ソフトウェア開発データ白書2006, 9.2章, pp.160-163, 日経BP

- 寄与率と p 値から, 目的変数(生産性など)と各変数との関連を概観できる.
- より詳細な分析を行うには, 箱ひげ図を書いたり, 散布図を書く必要がある.
- 統計的手法である以上, ある程度のデータ数が必要
 - カテゴリが細かすぎる場合に, 集約する必要がある.
- 説明変数同士の関連に注意を払う必要がある.
 - 外部委託率, 規模ともに生産性と関連があるが, 外部委託率と規模の間にも関連がある.

変数間の関係の有無と強さ

- **数値変数 数値変数**
 - 無相関検定, 相関係数 (順位相関係数)
- **カテゴリ変数 数値変数**
 - **分散分析, 寄与率**
 - t検定無, 平均値の差
- **カテゴリ変数 カテゴリ変数**
 - **カイ二乗検定, クラメールのV**

関連はあるか?
関連の強さは?



プロジェクトID	業種	主開発言語
1	銀行	PL/I
2	製造業	C
3	銀行	COBOL
4	銀行	COBOL
5	製造業	C
6	銀行	PL/I
7	銀行	COBOL
8	公共	Java

- 質的データ間の独立性(関連の有無)を検定する。
 - 帰無仮説の例:
業種と主開発言語には関連がない(=独立である)

プロジェクト ID	業種	主開発言語
1	銀行	PL/I
2	製造業	C
3	銀行	COBOL
4	銀行	COBOL
5	製造業	C
6	銀行	PL/I
7	銀行	COBOL
8	公共	Java

クロス集計表に変換する。

業種	言語			
	PL/I	C	COB OL	Java
銀行	2	0	3	0
製造業	0	2	0	0
公共	0	0	0	1

- 質的データ間の関連の強さを表す尺度
 - 質的データ版の「相関係数」ともいえる尺度
 - 量的データは、予め質的データへ変換しておく(大, 中, 小)
 - 0から1の値を取る. 1に近いほど変数間の関連が強い.

- ツール
 - 青木繁伸: 統計電卓(CGI) 独立性の検定(カイ二乗検定),
http://aoki2.si.gunma-u.ac.jp/calculator/chi_sq_test.html

カイ二乗検定 & クラメールのV - 分析事例

・ 分析対象のプロジェクト

- ソフトウェア開発データ白書2005年に記載の1009プロジェクトのうち211件（新規開発プロジェクト）

分析データのイメージ

業種	アーキテクチャ	主開発言語	規模 (FP)	工期 (月数)	平均要員数	外部委託率	生産性 (FP÷人時)
銀行	C/S	PL/I	上位	上位	上位	上位	下位
	スタンドアロン	C	下位	中位	中位	下位	下位
銀行	C/S	COBOL	下位	中位	下位	下位	下位
銀行		Visual Basic	中位		中位	中位	中位
製造業	スタンドアロン		中位	下位	中位	中位	中位
銀行	混合	C++		下位	上位	上位	上位
銀行	C/S	COBOL	上位	下位	下位	下位	上位
公共	混合	Java	中位	上位		下位	上位

量的データは、値の大きさに応じて下位25%、中位、上位25%に分類した。

カイ二乗検定 & クラメールのV - 分析事例

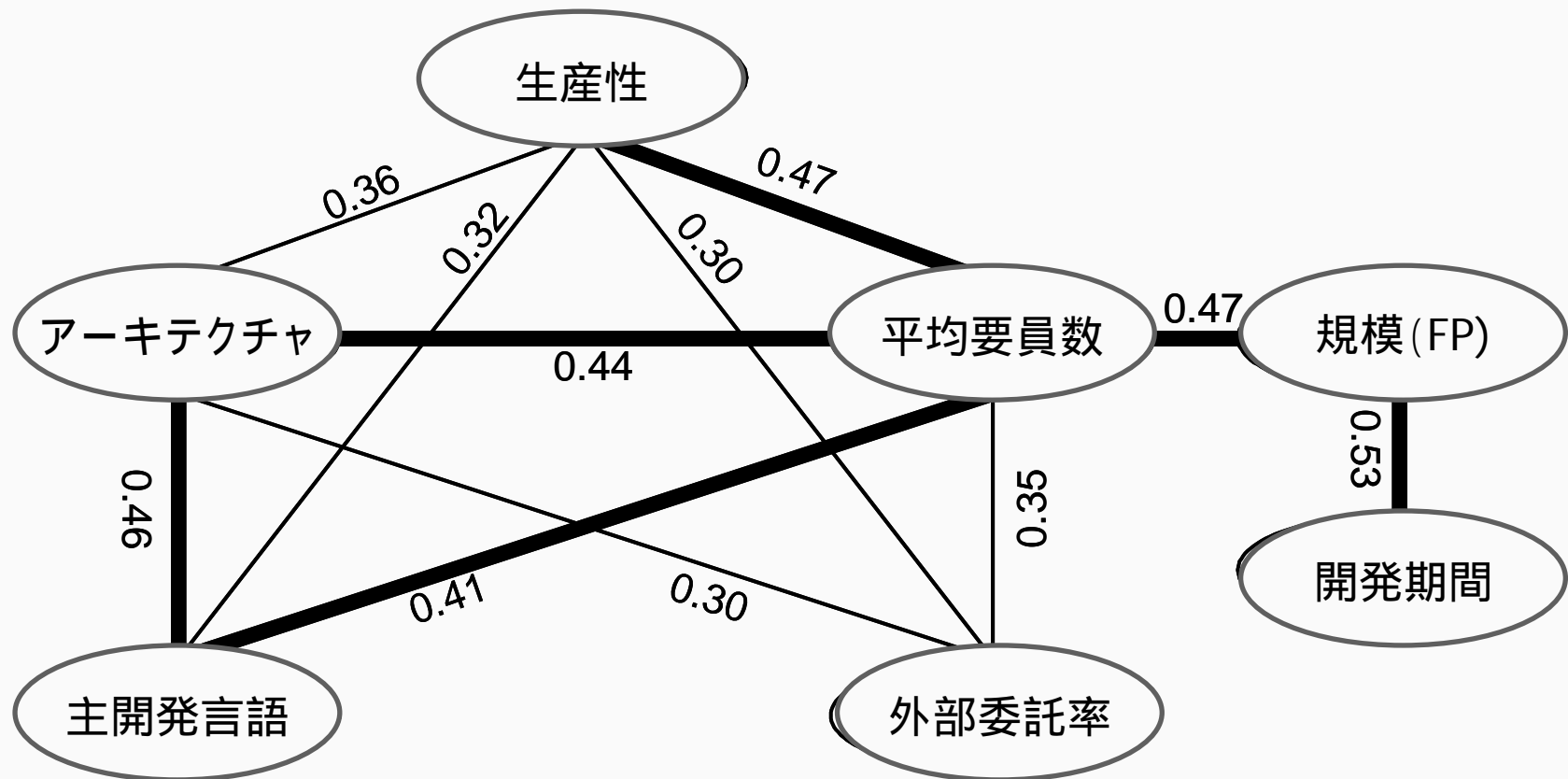
	外部委託率	平均要員数	工期	規模(FP)	業種	アーキテクチャ
平均要員数	0.35 0%					
工期	0.22 10%	0.26 0%				
規模(FP)	0.23 7%	0.47 0%	0.53 0%			
業種	0.31 13%	0.26 1%	0.18 32%	0.18 33%		
アーキテクチャ	0.30 4%	0.44 0%	0.24 0%	0.17 12%	0.20 13%	
主開発言語	0.31 16%	0.41 0%	0.28 0%	0.27 1%	0.28 1%	0.46 0%

- ・ 各マスの上段がクラメールのV, 下段がp値(パーセント表示)
- ・ 太字は有意水準5%で関連あり(p値 5%)

出典: ソフトウェア開発データ白書2006, 9.2章, pp.160-163, 日経BP

カイ二乗検定 & クラメールのV - 分析事例

クラメールのVに基づく関連グラフ



それぞれの2変数間の関係については,さらなる分析(箱ひげ図など)が必要

統計手法のまとめ

- **並行座標プロット**
 - データを概観したり, 仮説を立てるのに役立つ
- **分散分析, 寄与率**
 - カテゴリ変数 - 数値変数の関係を調べるのに役立つ.
- **カイ二乗検定, クラメールのV**
 - カテゴリ変数 - カテゴリ変数の関係を調べるのに役立つ.

参考文献

- 多変数データの統計解析, ソフトウェア開発データ白書2006, 9.2章, pp.160-163, 日経BP.
- M. Tsunoda, A. Monden, H. Yadohisa, N. Kikuchi, and K. Matsumoto, "Productivity analysis of Japanese enterprise software development projects," In Proc. 3rd Int'l Workshop on Mining Software Repositories (MSR2006), May 2006.
- 角田, 門田, 宿久, 菊地, 松本, "外部委託率に着目したソフトウェアプロジェクトの生産性分析,"ソフトウェアサイエンス研究会, 信学術報, No.SS2006-11, pp.19-24, April 2006.