ソフトウェア工学III
プロジェクト特性データの分析II
——統計分析（検定）——

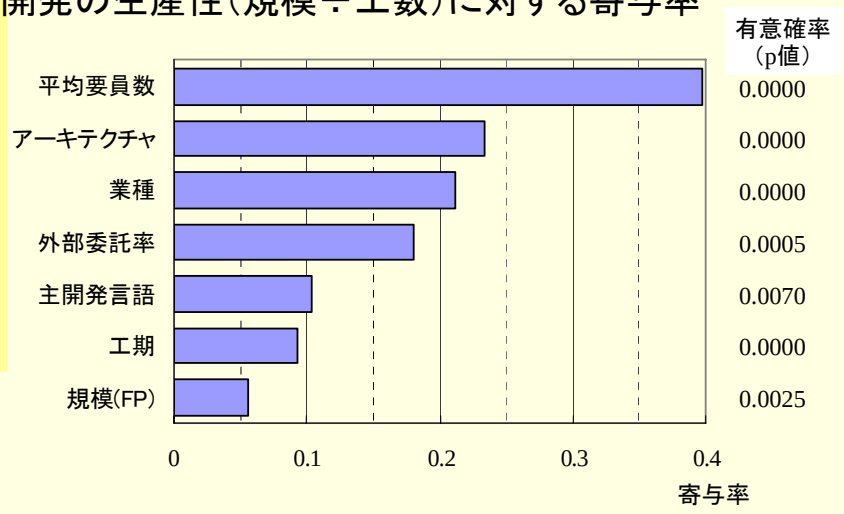
ソフトウェア工学講座
門田暁人
akito-m@is.naist.jp
B303室，内線5311

分析の目的

- 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） 分散分析と寄与率

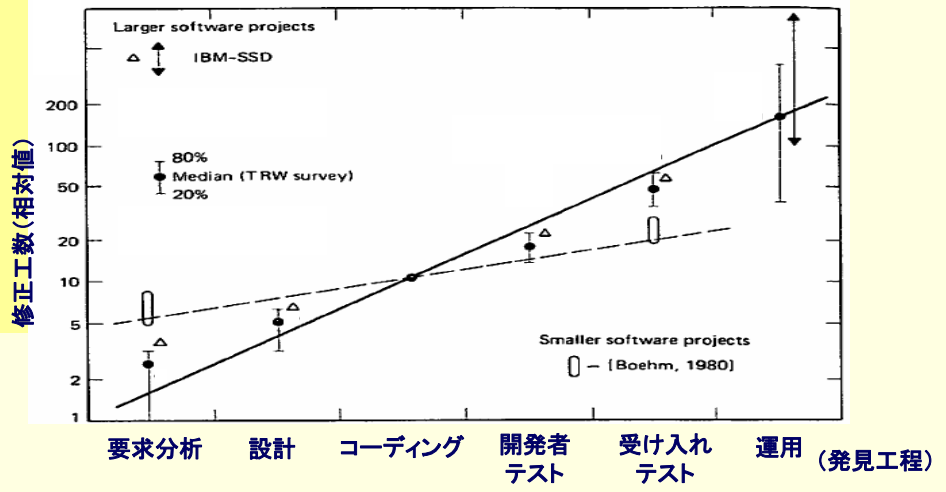
開発の生産性（規模÷工数）に対する寄与率



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

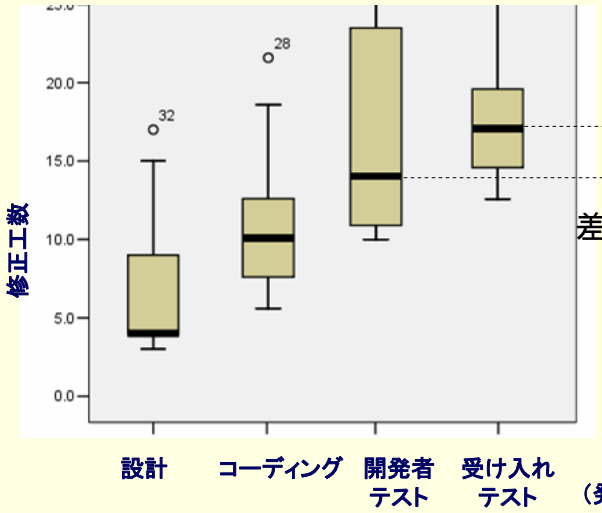
分析結果の例（2）

障害（バグ）の発見工程ー修正工数の関係



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

分析結果の例（2） t検定



差があると言えるか？

↓

t検定

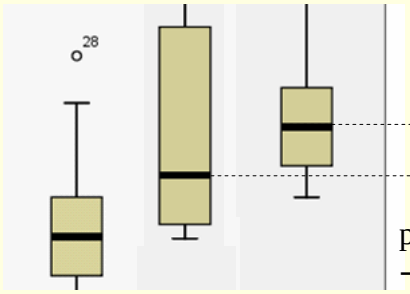
検定（statistical test）とは

- （統計的）仮説検定
  - ある仮説が正しいかどうかを統計的に判定する方法
    - 一般的な仮説（命題）：
      - バグの発見が遅くなるほど修正により多くの工数がかかる．
    - 仮説検定の対象となる**帰無仮説**：
      - バグの発見工程と修正工数は関係がない．→分散分析など
      - 「結合試験で発見されたバグの修正工数」は「総合試験で発見されたバグの修正工数」と差があるとはいえない．→t検定など
      - OSとプログラミング言語は関係がない．→カイ二乗検定など
    - 検定により帰無仮説が棄却されれば，対立仮説が支持される．
      - バグの発見工程と修正工数は関係がないとはいえない．
        →バグの発見工程と修正工数は統計的に有意な関係がある．

（帰無仮説が棄却できなかった場合は，帰無仮説が支持されるわけではない）

検定における有意確率（p値）と有意水準

- 検定を行うと，p値が算出される．
  - p値＜有意水準（例えば0.05）ならば，帰無仮説が棄却される．
    →有意水準5%で関係あり，有意差あり，が言える．
  - （通常は帰無仮説を棄却したいので，p値は小さいほど嬉しい．p値が小さい→有意差があった！）



p = 0.00132
→有意差あり！

2群の差の検定

例：JavaとC++では，生産性に差があるか？

- パラメトリック検定（標本の母集団が正規分布のとき）
  - 平均値の差の検定（t検定）
- ノンパラメトリック検定（正規分布を仮定しない）
  - 代表値（中央値）の差の検定
    - マン・ホイットニーのU検定（ウィルコクソンの順位和検定）

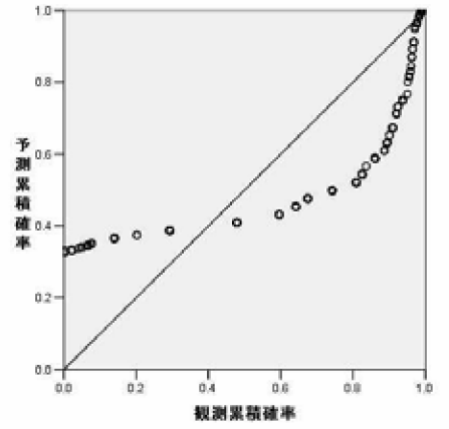
ソフトウェア工学データの多くは，正規分布に従わないため，
ノンパラメトリック検定を使うことが望ましい．
（もしくは対数変換してからパラメトリック検定を行う）

Web上のツール（2群の差の検定）

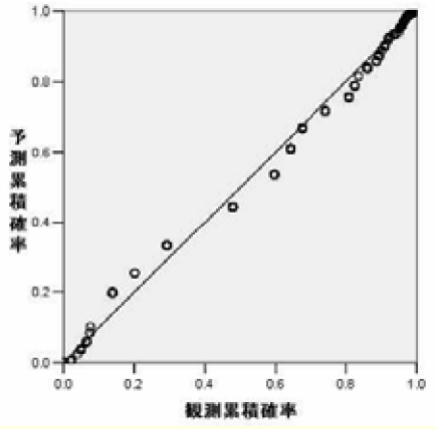
- JavaScript による検定
  - 群馬大学社会情報学部の青木繁伸教授による
  - t検定
    - http://aoki2.si.gunma-u.ac.jp/JavaScript/t-test.html
  - マン・ホイットニーのU検定
    - http://aoki2.si.gunma-u.ac.jp/JavaScript/permtest4.html

おまけ：正規分布に従っているかどうかの判断方法

- 正規Q－Qプロットを用いて視覚的に判断する.



バグ修正工数の分布



$\log_{10}$(バグ修正工数)の分布

データが正規分布に従っている場合は直線上に点が乗る.

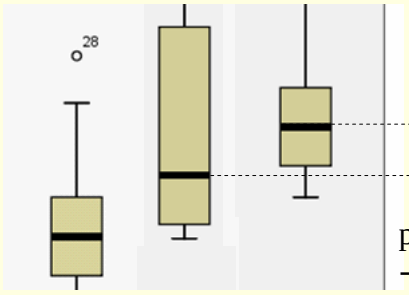
例題

- project1.csvにおいて，次の2群について，それぞれ生産性の平均値を求めよ.
  - ピーク要員数が5人以下のプロジェクト
  - ピーク要員数が10人以上のプロジェクト
- この2群の間で生産性に差があると言えるか？
  有意水準5%で検定せよ（マンホイットニーのU検定）

例題 （手順）

1. ピーク要員数が5人以下のプロジェクトの生産性の値の集合Aを作る. {0.1015, 0.7302, 0.1704,…}
   - 生産性が欠損値となっているプロジェクトは削除する.
   - ピーク要員数がゼロとなっているプロジェクトも削除する.
2. 同様に，ピーク要員数が10人以上のプロジェクトの生産性の値の集合Bを作る. {0.08147, 0.0637, ….}
3. ピーク要員数が6人～9人，もしくは，欠損値となっているプロジェクトは無視する.
4. AとBについて，それぞれ平均値を求める.
5. AとBを用いてマンホイットニーのU検定を行う.

留意点



p = 0.00132
→有意差あり！

- 検定では，差の有無だけを判定する．
- p値が小さい＝差が大きいとは限らない．
- 差の大きさを見たい場合，2群間の平均値の差を算出すればよい．
  - 単なる差ではなく分散を考慮した式もある．→effect size

（一元配置）分散分析

- 3群以上の間で平均値に差があるかどうかを判定する．
  - 帰無仮説の例：
    - 各業種（銀行，製造業，公共）の生産性の平均値は等しい
- 差の大きさは寄与率として表すことができる．

関連はあるか？
関連の強さは？

カテゴリ変数（名義尺度）	数値変数（比尺度）

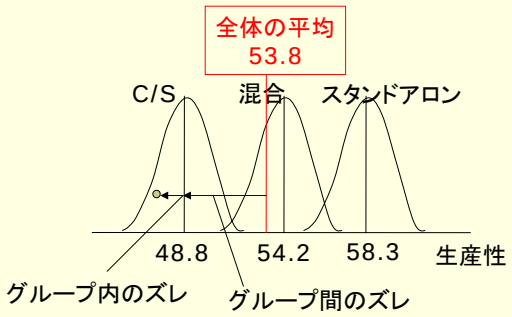
プロジェクトID	業種	生産性（FP÷人時）
1	銀行	0.0225
2	製造業	0.0970
3	銀行	0.1016
4	銀行	0.1203
5	製造業	0.1273
6	銀行	0.1835
7	銀行	0.2022
8	公共	0.1551

分散分析　ー　寄与率

- 寄与率

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

  - 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ω^2を用いる方が良い.

分散分析　ー　分析事例

- 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）

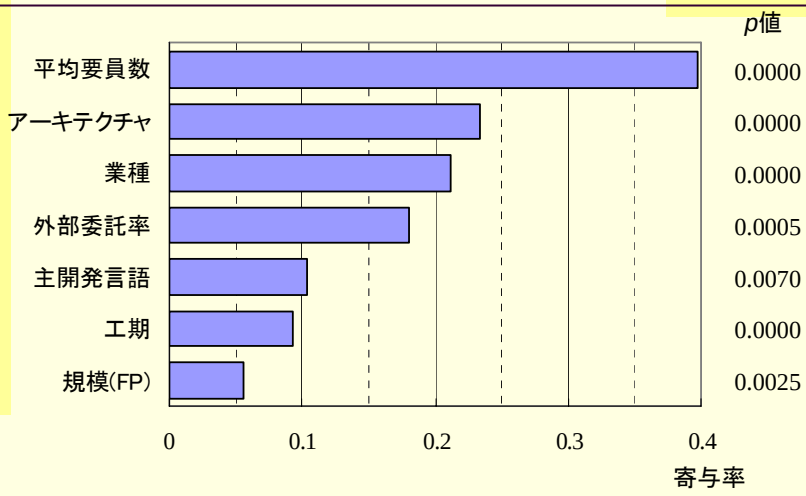
分析データのイメージ

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　ー　分析事例

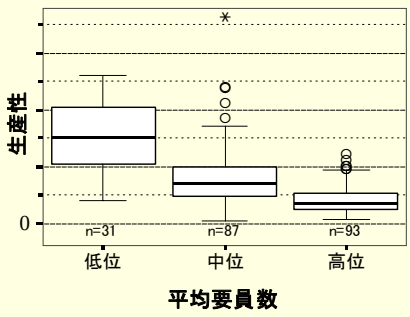


全て$p \leqq 0.05$なので，有意水準5％で生産性との関連あり.

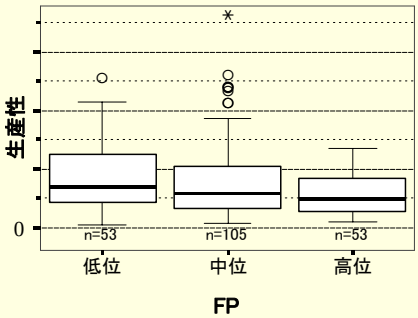
出典：ソフトウェア開発データ白書2006, 9.2章, pp.160-163, 日経BP

分散分析　ー　分析事例

- 寄与率の大きな変数のほうが，グループ間で生産性の差が大きい.

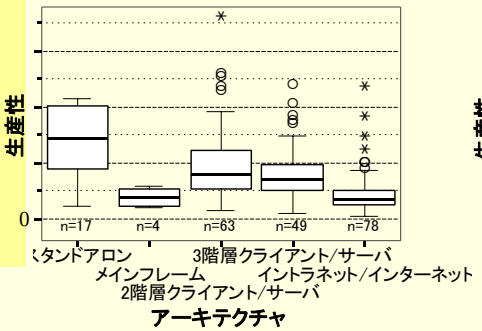


寄与率 0.39

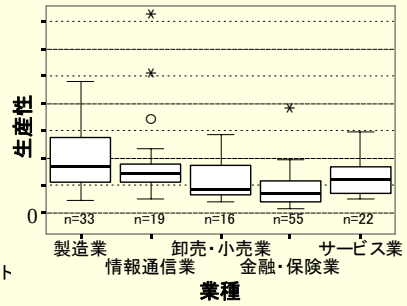


寄与率 0.05

分散分析　－　分析事例

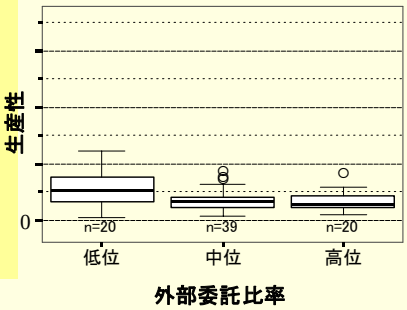


寄与率 0.23

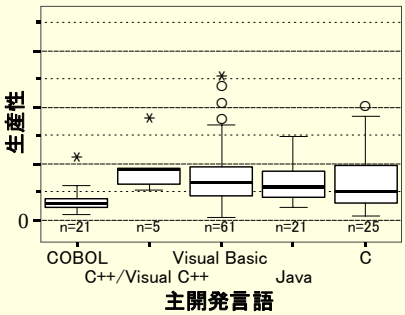


寄与率 0.21

分散分析　－　分析事例



寄与率 0.18



寄与率 0.10

分散分析に用いる表の作り方

アーキテクチャ	生産性
C/S	0.0225
スタンドアロン	0.0970
C/S	0.1016
	0.1203
スタンドアロン	0.1273
混合	0.1835
C/S	0.2022
混合	0.1551

アーキテクチャで並び替える

アーキテクチャ	生産性
C/S	0.0225
C/S	0.1016
C/S	0.2022
スタンドアロン	0.0970
スタンドアロン	0.1273
混合	0.1835
混合	0.1551
	0.1203

アーキテクチャが欠損値のため分析対象外とする

カテゴリごとの列にする

アーキテクチャ		
C/S	スタンドアロン	混合
0.0225	0.0970	0.1835
0.1016	0.1273	0.1551
0.2022		

3群以上の差の検定

- パラメトリック検定（標本の母集団が正規分布のとき）
  - 分散分析
- ノンパラメトリック検定（正規分布を仮定しない）
  - クラスカル・ウォリス検定
- JavaScript による検定
  - 群馬大学社会情報学部の青木繁伸教授による
    - 分散分析
      - http://aoki2.si.gunma-u.ac.jp/JavaScript/onewayANOVA.html
    - クラスカル・ウォリス検定
      - http://aoki2.si.gunma-u.ac.jp/JavaScript/kwtest.html

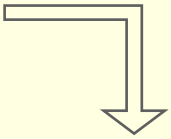
例題

- project1.csvにおいて，業種間で生産性の平均値に差があるといえるか，有意水準5%で検定せよ．
  （分散分析またはクラスカル・ウォリス検定）
  - 「その他」業種を除く

カイ二乗検定

- 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

クラメールのV

- 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
- 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　ー　分析事例

- 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

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　ー　分析事例

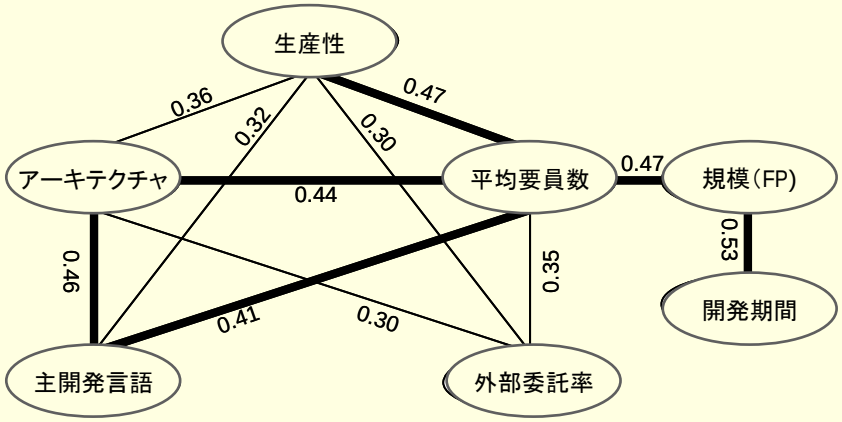
	外部委託率	平均要員数	工期	規模(FP)	業種	アーキテクチャ
平均要員数	**0.35** 0%					
工期	0.22 10%	**0.26** 0%				
規模(FP)	0.23 7%	**0.47** 0%	**0.53** 0%			
業種	0.31 13%	**0.26** 1%	0.18 32%	0.18 33%		
アーキテクチャ	**0.30** 4%	**0.44** 0%	**0.24** 0%	0.17 12%	0.20 13%	
主開発言語	0.31 16%	**0.41** 0%	**0.28** 0%	**0.27** 1%	**0.28** 1%	**0.46**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

クラメールのVに基づく関連グラフ



例題

- project1.csvにおいて，アーキテクチャと業種とに関連があるといえるか，有意水準5%で検定せよ．（カイ二乗検定）
  - 「その他」業種を除く

演習レポート課題

- レポート課題
  - プロジェクト特性データ（project1.csv，project2.csv，canada.csvのいずれか）を分析して，何らかの知見を2つ以上示せ．
    - データの可視化，および，検定
    - 必要に応じて，Weka，DAVIS，および，青木繁伸先生のWebページにある統計ツールなどを使うこと．
  - 分析結果だけでなく，分析の過程の説明，結果の考察も含めてレポートにまとめよ．書式は問わない．
  - 本日の3つの例題で示した内容を実施してもよい．

演習課題 ー 提出期限

- レポート提出期限
  - 2006年12月22日（金）2限
  - 希望者はレポート課題の内容を発表すること
    - レポートの内容について説明する.
    - レポート用紙をスクリーンに映す，もしくは，パワーポイント等のプレゼン資料を用いる.
  - 時間が余れば，提出されたレポートの中からいくつかを講義中に（門田が）紹介します.
- 連絡先
  - 門田暁人 akito-m@is.naist.jp
  - B303室，内線5311

題材

- 79プロジェクト，10変数のデータセット project1.csv
  - 演習用に作成した典型的なもの（架空のデータ）
  - 欠損値を含む
  - 欠損値にマイナスの値を埋めたもの project2.csv
- 77プロジェクト，11変数のデータセット canada.csv
  - カナダのソフトウェア企業のもの（実データ）
  - 欠損値なし

ソフトウェア工学IIIのWebサイトよりダウンロードしてください.

canada.csvに含まれる変数

- case-ID：プロジェクトID
- Duration：開発期間（月数）…. 1ヶ月＝約160時間
- ExpEquip：開発チーム経験年数
- ExpProjMan：プロジェクトマネージャ経験年数
- Transactions：トランザクション数
- RawFPs：規模（調整前FP）
- Adj Factor：FP調整係数
- Adj FPs：規模（調整後FP）
- Dev Env：開発環境
- Year Fin：開発終了年
- Entities：エンティティ数
- ActualEffort：開発工数（人時）

導出可能な変数

生産性＝Adj FPs ÷ ActualEffort （1人時で開発可能な規模）
平均要員数＝ActualEffort ÷ (Duration × 160)