STN

SAS Technical News

For Higher
Customer Satisfaction,
We Bridge
the SAS System
Between
Customer's World.

SPRING 2011

特集 …… 01
SASから始めるSQL

SAS Academic News …… 09
- 事例紹介
- コラム「SAS四方山話」

Q&A …… 18
SASトレーニングのお知らせ …… 21
最新リリース情報 …… 23
SASユーザー総会のお知らせ …… 24

SQLは長い歴史を持つデータベース操作言語で、現在も多くのリレーショナルデータベースで標準的に使用されています。SAS以外のリレーショナルデータベースでSQLを使用していた方は、その知識をSASで直接生かすことができます。まだSQLに触れたことがない方も、これからSQLを学ぶ利点はあります。DATAステップとは異なる構文、異なるデータ処理方法を持つ言語を学ぶことで、プログラミングの幅を広げることができます。例えば、複数のテーブルから情報を取得するような処理はSQLを利用すると簡易に記述できるため、その箇所だけSQLで書くケースがあります。処理の内容によってDATAステップとSQLを使い分けることで、プログラムの開発効率をスマートに上げることができます。

特集

SASから始めるSQL

本特集ではSQLの基本であるデータの抽出と、代表的な処理であるテーブルの結合およびサブクエリについてサンプルプログラムと図を交えてご説明します。

1 SQLとは

SQLとはリレーショナルデータベース管理システム(RDBMS)において、データの抽出や更新を行うための言語です。標準規格として多くのデータベースでサポートされており、SASでもサポートしています。
SASでSQLを使うと、DATAステップとほぼ同等の処理をデータセットや外部データベースのテーブルに対して行うことができます。また、SQL文の中でSAS関数、データセットオプション、マクロなどSAS特有の機能も使用することができます。
SQLとDATAステップにて使用する用語の名称が異なるものがあります。主な用語の違いについては以下の表をご覧ください。

SQLとDATAステップ用語比較表

SAS DATAステップにおける名称	SQLにおける名称
データセット	テーブル
オブザベーション	行
変数	カラム(列)

2 SQLプロシジャ

SQLプロシジャはBase SAS® に含まれるプロシジャです。SASでSQLを使用する場合、次のようにSQLステートメントをSQLプロシジャの中に記述します。

```
PROC SQL;
  < SQLステートメント>;
                        /* QUITステートメントで終わる点に注意! */
QUIT;
```

SQLステートメントは一つのまとまった処理を実行するための命令文です。複数の句によって構成され、必ずセミコロンで終わります。
また、下記のように一つのSQLプロシジャの中に複数の独立したSQLステートメントを記述することも可能です。

```
PROC SQL;
  < SQLステートメント>;
  < SQLステートメント>;
  < SQLステートメント>;
QUIT;
```

3 データの抽出と加工

SQLで最も多く使用されるデータの抽出処理について説明します。データの抽出とは、既存のテーブルからデータを読み込む処理です。DATAステップの「データセットの読み込み」と同じ処理です。
SQLでデータの抽出処理を行うには、次のようなSQLステートメントを記述します。

```
                                        /* クエリの基本構文*/
PROC SQL;
  SELECT カラム名
    FROM ライブラリ名.テーブル名
      WHERE 抽出条件
        ORDER BY カラム名
  ;
QUIT;
```

上記のようにSELECT句から始まるSQLステートメントをクエリと呼ぶことがあります。クエリには必ずSELECT句とFROM句が必要です。それ以外の句は処理の内容に応じて追加します。
この章ではクエリにおけるさまざまな句の使い方について、サンプルプログラムを用いて説明します。なお、サンプルプログラムではsashelp.classを用います。これはSASのsashelpライブラリ配下に標準で格納されているデータセット(テーブル)です。
sashelp.classに格納されているデータは以下の通りです。なお、SAS® 9.2から文字データが日本語に変更されています。

sashelp.classの内容

OBS	Name	Sex	Age	Height	Weight
1	アルフレッド	男子	14	69	112.5
2	アリス	女子	13	56.5	84
3	バーバラ	女子	13	65.3	98
4	キャロル	女子	14	62.8	102.5
5	ヘンリー	男子	14	63.5	102.5
6	ジェームズ	男子	12	57.3	83

〈7行目以降省略〉

3.1 データの抽出 (SELECT句、FROM句)

クエリの最も簡単な形は、次のようにSELECT句とFROM句のみを含みます。

```
                                        /* 最も簡単なクエリ */
PROC SQL;
  SELECT name, weight, age
    FROM sashelp.class
  ;
QUIT;
```

このクエリは、SELECT句で指定したカラム(name、weight、age)を、FROM句で指定したテーブル(sashelp.class)から抽出し、結果をアウトプットウィンドウに表示します。カラムはSELECT句で指定した順番で表示されます。

名前	体重(ポンド)	年齢
アルフレッド	112.5	14
アリス	84	13
バーバラ	98	13
キャロル	102.5	14
ヘンリー	102.5	14
ジェームズ	83	12

〈7行目以降省略〉

SQLプロシジャの中でSELECT句を使うと、抽出結果はアウトプットウィンドウに表示されます。SELECT句単体では、新しいテーブルは作成されません。
また、次のようにSELECT句でアスタリスク(*)を指定するとすべてのカラムが抽出されます。

```
PROC SQL;
                                        /* すべてのカラムを抽出する */
  SELECT *
    FROM sashelp.class
  ;
QUIT;
```

SELECT文の前にCREATE TABLE ASステートメントを追加することで、クエリの結果をテーブルに出力できます。大きいテーブルを扱う場合は抽出結果をテーブルに出力することが有用です。

```
PROC SQL;
                                        /* 新規テーブルを作成 */
  CREATE TABLE table1 AS
    SELECT *
      FROM sashelp.class
  ;
QUIT;
```

上記プログラムを実行するとworkライブラリの中にテーブルtable1が作成されますが、他の恒久ライブラリの中にテーブルを出力することもできます。

```
                                        /* tempライブラリの割り当て */
LIBNAME temp "c:¥temp";

PROC SQL;
                                        /* 新規テーブルを作成 */
  CREATE TABLE temp.table1 AS
    SELECT *
      FROM sashelp.class
  ;
QUIT;
```

3.2 データの並べ替え（ORDER BY句）

特定のカラムの値に応じて行を並べ替えるにはORDER BY句を用います。下記の例では、カラムageに含まれる値に応じて行が昇順で並べ替えられます。

```
PROC SQL;
  CREATE TABLE sorted AS
    SELECT age, name, height
      FROM sashelp.class
                          /* ageによってデータを並べ替える */
        ORDER BY age
  ;
QUIT;
```

デフォルトではデータは昇順で並べ替えられます。降順で並べ替えるにはカラムの後に「DESC」を記述してください。

並べ順キーワードについて

カラムの後に「ASC」を記述することで、昇順で並べ替えることを明示的に指定できます。

ASC: 昇順

DESC: 降順

```
                                  /* 降順で並べ替える */
ORDER BY age DESC
```

結果テーブル（age降順）

OBS	Age	Name	Height
1	16	フィリップ	72
2	15	ロナルド	67
3	15	メアリー	66.5
4	15	ウィリアム	66.5
5	15	ジャネット	62.5
6	14	ヘンリー	63.5

〈7行目以降省略〉

SORTプロシジャと同様に、複数のカラムの値を対象とした並べ替えを行うこともできます。次のようにORDER BY句で複数のカラムを指定します。

```
      /* ageを降順で、heightを昇順（デフォルト値）で並べ替える */
ORDER BY age DESC, height
```

結果テーブル（age降順、height昇順）

OBS	Age	Name	Height
1	16	フィリップ	72
2	15	ジャネット	62.5
3	15	メアリー	66.5
4	15	ウィリアム	66.5
5	15	ロナルド	67
6	14	キャロル	62.8

〈7行目以降省略〉

SQLでは、クエリを実行すると行の順序が変わってしまう場合があります。行の順序を保つ必要がある場合はORDER BY句を指定してください。

3.3 抽出条件の指定（WHERE句）

WHERE句でデータの抽出条件を指定できます。WHERE句の条件に合致した行がテーブルから抽出されます。

SQLのWHERE句では条件式を記述します。条件式は真・偽について評価され、真の場合に限り該当の行が抽出されます。条件式では比較演算子、論理演算子、または特殊演算子を用いることができます。それぞれにおける代表的な演算子を例に挙げて、説明します。

● 比較演算子

SAS DATAステップとほぼ同様の比較演算子を使うことができます。例えばカラムageが15より大きい行を抽出するには、次のようなプログラムを記述します。

```
PROC SQL;
  CREATE TABLE condition1 AS
    SELECT age, name, height
      FROM sashelp.class
                                  /* 抽出条件を指定 */
        WHERE age > 15
  ;
QUIT;
```

結果テーブル（age > 15）

OBS	Age	Name	Height
1	16	フィリップ	72

● 論理演算子

DATAステップのWHEREステートメントと同じように、「AND」「OR」「NOT」論理演算子が使用できます。以下のサンプルプログラムでは、カラムsexが「男子」ではない行を抽出します。なお、二つの条件を明示的に区切るためにカッコを記述しています。

```
PROC SQL;
  CREATE TABLE condition2 AS
    SELECT age, name, height
      FROM sashelp.class
                                  /* 男子を除外する */
        WHERE sex NE "男子"
  ;
QUIT;
```

結果テーブル（男子を除外）

OBS	Age	Name	Height
1	13	アリス	56.5
2	13	バーバラ	65.3
3	14	キャロル	62.8
4	12	ジェーン	59.8
5	15	ジャネット	62.5
6	11	ジョイス	51.3
7	14	ジュディー	64.3
8	12	ルイーズ	56.3
9	15	メアリー	66.5

● 特殊演算子

INやCONTAINSなどDATAステップのWHEREステートメントと共通する特殊演算子の他、EXISTSやANYなどSQL特有の演算子を使うことができます。以下、CONTAINS特殊演算子を使ったサンプルプログラムをご紹介します。

下記プログラムを実行すると、カラムnameの中に「ア」もしくは「イ」が含まれている行が抽出されます。

```
PROC SQL;
  CREATE TABLE condition3 AS
    SELECT age, name, height
      FROM sashelp.class
                                  /* アもしくはイを含む行 */
        WHERE (name CONTAINS "ア") OR (name CONTAINS "イ")
  ;
QUIT;
```

結果テーブル（nameにアもしくはイを含む）

OBS	Age	Name	Height
1	14	アルフレッド	69
2	13	アリス	56.5
3	11	ジョイス	51.3
4	12	ルイーズ	56.3
5	15	メアリー	66.5
6	15	ウィリアム	66.5

3.4 カラム名のリネーム（AS句）

クエリの中にAS句を記述することで、カラム名のリネームを行うことができます。以下のサンプルプログラムを参照してください。カラムageをage2にリネームして新規テーブルrenamedを作成します。

なお、リネーム処理は、抽出元テーブルsashelp.classのカラム名には影響を与えません。

```
PROC SQL;
  CREATE TABLE renamed AS
                                  /* ageをage2にリネーム */
      SELECT name, age AS age2
        FROM sashelp.class
  ;
QUIT;
```

3.5 重複データの削除（DISTINCT句）

クエリの中でDISTINCT句を指定することで、重複する行の削除を行うことができます。例えば、sashelp.classに含まれる変数ageの値の種類や範囲を調べたい場合、次のようにDISTINCT句を指定します。

```
PROC SQL;
  CREATE TABLE unique AS
                    /* ageの値が重複している行を削除する */
      SELECT DISTINCT age
        FROM sashelp.class
  ;
QUIT;
```

4 テーブルの結合

前章では単体のテーブルからデータを抽出する方法をご紹介しました。本章では複数のテーブルからデータを抽出し、共通カラムを使ってデータを結合する「テーブルの結合」について説明します。結合の中でも特に使用頻度が高いと思われる内部結合と外部結合について説明します。

説明にあたって、例として文房具店の販売データを用います。販売データはテーブル商品表と、テーブル売上明細表で構成されます。テーブル商品表とテーブル売上明細表はそれぞれ異なる情報を保持していますが、両テーブルに共通してカラムIDが含まれる点にご注目ください。通常、テーブルの結合を行う場合はIDのような共通カラムが必要となります。

商品表（one）

OBS	id	item	price
1	1	消しゴム	100
2	2	ボールペン	120
3	3	鉛筆	50
4	4	ノート	210
5	5	ガムテープ	200
6	6	万年筆	1200
7	7	カレンダー	400

売上明細表（two）

OBS	id	amount	date
1	4	4	2011/01/01
2	2	3	2011/01/03
3	7	1	2011/01/12
4	1	6	2011/01/15
5	6	4	2011/01/18
6	94	3	2011/01/22

サンプルテーブルの作成方法

以下のDATAステップを実行し、サンプルテーブル商品表、売上明細表を作成します。

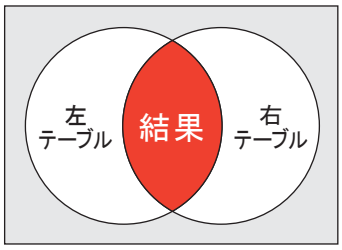
```
DATA one;
  INPUT id item : $15. price;
  DATALINES;
1 消しゴム 100
2 ボールペン 120
3 鉛筆 50
4 ノート 210
5 ガムテープ 200
6 万年筆 1200
7 カレンダー 400
;
RUN;

DATA two;
  INPUT id amount date : yymmdd10.;
  FORMAT date yymmdds10.;
  DATALINES;
4 4 2011/01/01
2 3 2011/01/03
7 1 2011/01/12
1 6 2011/01/15
6 4 2011/01/18
94 3 2011/01/22
;
RUN;
```

4.1 テーブルの内部結合

内部結合とは、二つのテーブルから指定したカラムの値が一致する行を抽出する処理です。

上述の文房具店の例を用いると、テーブル商品表とテーブル売上明細表の両方に含まれる商品だけを抽出したい場合に内部結合を使用します。

具体的には、次のような結果テーブルを想定しています。テーブル商品表とテーブル売上明細表のカラムidの値が一致する行だけを出力します。



テーブル売上明細表6行目のカラムIDの値は商品表には存在しないため、結果テーブルに出力されません。

また、テーブル商品表3行目のカラムIDの値はテーブル売上明細表に存在しないため、これも出力されません。

結果テーブル(内部結合)

OBS	id	item	price	amount	date
1	1	消しゴム	100	6	2011/01/15
2	2	ボールペン	120	3	2011/01/03
3	4	ノート	210	4	2011/01/01
4	6	万年筆	1200	4	2011/01/18
5	7	カレンダー	400	1	2011/01/12

内部結合によって上記結果を得るには、次のようなSQL文を記述します。FROM句の後に、二つのテーブルをJOIN句で繋げて記述します。また、ON句を使って共通カラムidを等式で繋いでいます。

```
PROC SQL;
  CREATE TABLE join1 AS
    SELECT one.id, one.item, one.price, two.amount, two.date
                                                  /* 内部結合 */
      FROM one JOIN two
        ON one.id=two.id
  ;
QUIT;
```

結合を行う場合、カラムを「テーブル名.カラム名」という形で記述することがあります。これは、二つのテーブルにおいて同じ名前のカラムがある場合、どちらのテーブルのカラムであるかを区別するためです。

4.2 テーブルの外部結合

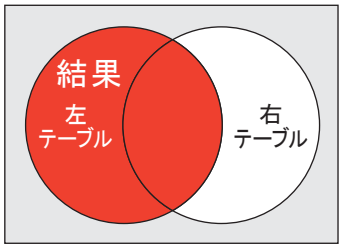
外部結合には、「左外部結合」「右外部結合」「完全外部結合」の3種類が存在します。

左外部結合とは、左のテーブルに存在するすべての行と、左と右に共通して含まれる行を抽出します。

右外部結合はその逆です。完全外部結合は、両テーブルのすべての行を抽出します。

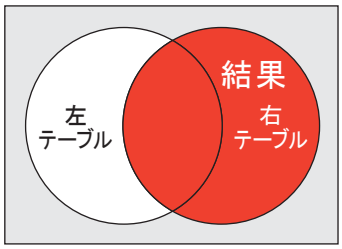
それぞれをベン図で表すと次のようになります。

左外部結合

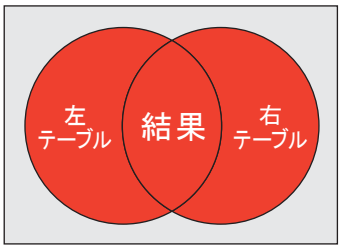


今回は、左外部結合について解説します。

右外部結合



完全部結合



文房具店の例を用いると、商品表を左テーブル、売上明細表を右テーブルとして左外部結合を行った場合、次のような結果テーブルが出力されます。

結果テーブル (左外部結合)

OBS	id	item	price	amount	date
1	1	消しゴム	100	6	2011/01/15
2	2	ボールペン	120	3	2011/01/03
3	3	鉛筆	50	.	.
4	4	ノート	210	4	2011/01/01
5	5	ガムテープ	200	.	.
6	6	万年筆	1200	4	2011/01/18
7	7	カレンダー	400	1	2011/01/12

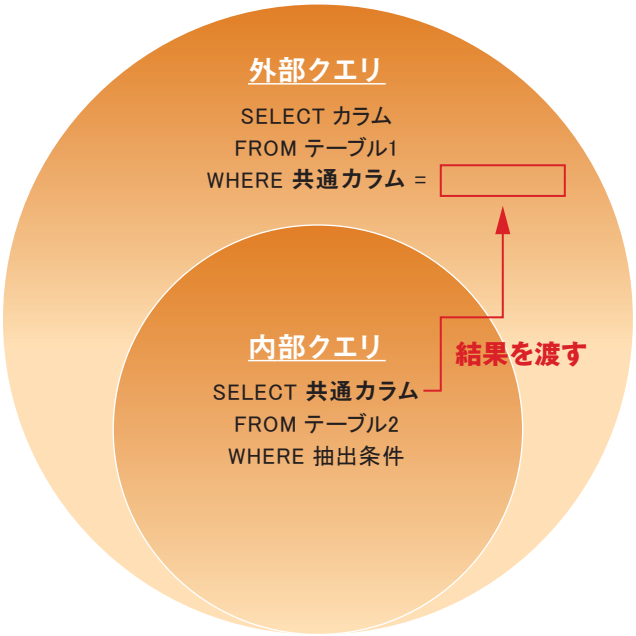
ベースとなる商品表のデータは全て出力されます。これは売上明細表から商品表にマッチするデータだけを抽出して結合するイメージです。
左外部結合の構文は以下の通りです。左テーブルと右テーブルをLEFT OUTER JOIN句で繋ぎます。また、内部結合の時と同じようにON句を使って共通カラムを繋ぎます。

```
PROC SQL;
  CREATE TABLE join2 AS
    SELECT one.ID, one.ITEM, one.price, two.amount ,two.date
                                        /* 左外部結合 */
      FROM one LEFT OUTER JOIN two
        ON one.id=two.id
  ;
QUIT;
```

なお、右外部結合の場合はRIGHT OUTER JOIN句を、完全外部結合の場合はFULL OUTER JOIN句を記述します。

5 サブクエリ

サブクエリとは、クエリの結果を別のクエリに渡して処理させる方法です。サブクエリを使うことで、複数のテーブルをまたぐ複雑なクエリを効率よく記述することができます。
以下の図を参照してください。外側の円がベースとなるクエリで、内側の円がサブクエリを表します。最初に内側クエリが実行され、その抽出結果を受け取った外側クエリが実行されます。



5.1 サブクエリの結果が1行だけの場合

第4章で使用した文房具店の販売データに関するテーブルを使って、サブクエリの使い方をご紹介します。

商品表 (one)

OBS	id	item	price
1	1	消しゴム	100
2	2	ボールペン	120
3	3	鉛筆	50
4	4	ノート	210
5	5	ガムテープ	200
6	6	万年筆	1200
7	7	カレンダー	400

売上明細表 (two)

OBS	id	amount	date
1	4	4	2011/01/01
2	2	3	2011/01/03
3	7	1	2011/01/12
4	1	6	2011/01/15
5	6	4	2011/01/18
6	94	3	2011/01/22

例題として、売上(カラムamount)が5個より大きい商品の商品名(カラムitem)と価格(カラムprice)を抽出する処理の実現方法を考えます。求める結果は次の通りです。

結果テーブル (amount > 5)

OBS	id	item	price
1	1	消しゴム	100

このような処理を行うにはどうすればよいでしょうか？　ご覧のとおり、商品表には売上の情報は含まれていません。また、売上明細表には商品名、価格のデータは含まれていません。
一つの選択肢として内部結合を使う方法が考えられます。二つのテーブルを内部結合で一つにまとめて得られたテーブルに対してカラムamountの値が5個より大きい行を抽出するクエリを記述することが考えられます。確かにこの方法によって求める結果は得られます。しかし、別のテーブルから1行のデータを抽出するためだけにテーブル結合を実行し、新たなテーブルを作成するのは効率よくありません。
このような処理はサブクエリを使うことで効率よく行うことができます。次の構文を参照してください。

```
PROC SQL;
  CREATE TABLE subquery AS
    SELECT *
      FROM one
        WHERE one.id =
          (
                                        /* 1行のみ抽出される */
            SELECT two.id
              FROM two
                WHERE amount > 5
          )
  ;
QUIT;
```

開き括弧から閉じ括弧までの構文がサブクエリです。

```
                                    /* サブクエリの内容 */
 SELECT two.id
   FROM two
     WHERE amount > 5
```

サブクエリのWHERE句を満たす1行(amountが6、idが1の行)が外側クエリに返されます。
サブクエリの結果を受け取った外側クエリは、次のようなクエリを実行します。最終結果として、1行だけのデータを持つテーブルが作成されます。

```
                                    /* 外側クエリの内容 */
   SELECT *
     FROM one
       WHERE one.id =1
```

5.2 サブクエリの結果が1行より多い場合

前の例題では、サブクエリに合致する行が1行だけでした。今回はサブクエリに合致する行が1行より多い場合を想定します。
例えば、売上(カラムamount)が3個より大きい商品の商品名(カラムitem)と価格(カラムprice)を抽出するにはどうすれば良いでしょうか?
サブクエリ内の「amount>5」という記述を「amount>3」に変えてクエリを実行すると次のようなエラーが発生します。

```
                                    /* エラーが発生する */
PROC SQL;
  CREATE TABLE err AS
    SELECT *
      FROM one
        WHERE one.id =
          (
                                    /* 複数行が抽出される */
            SELECT two.id
              FROM two
                WHERE amount > 3
          )
  ;
QUIT;
```

サブクエリの結果が複数の場合、エラーが発生

```
ERROR: 複数の行に対して、サブクエリが評価されます。

2536  QUIT;
NOTE: エラーが発生したため、このステップの処理を中止しました。
NOTE: PROCEDURE SQL 処理 (合計処理時間):
       処理時間              0.02 秒
       CPU 時間              0.00 秒
```

エラーの原因は、サブクエリから受け渡される複数の値を外側クエリの(WHERE句の)等式で扱うことができないためです。今回、サブクエリに合致する行が複数(カラムidが4、1もしくは6の値の時)存在しますが、これらがWHERE句に返されると次のような無効な条件式になってしまいます。

```
                                    /* 無効な条件式*/
 WHERE one.id = 4、1、6
```

このような問題はIN演算子を使うことで解決できます。IN演算子は右辺に複数の値を持つ式を評価することができる特殊な演算子です。次のように=演算子をIN演算子に置き換えます。

```
PROC SQL;
  CREATE TABLE subquery2 AS
    SELECT *
      FROM one
                                    /* =の代わりにINを使う */
        WHERE one.id IN
          (
                                    /* 複数行が抽出される */
            SELECT two.id
              FROM two
                WHERE amount > 3
          )
  ;
QUIT;
```

結果テーブル (AMOUNT > 3)

OBS	id	item	price
1	1	消しゴム	100
2	4	ノート	210
3	6	万年筆	1200

6 終わりに

本特集ではSQLの基本から応用まで広く説明しました。記事の都合上省略せざるを得なかった句や、紹介しきれなかったテクニック、SQL独自の動作などがありますが、本特集がそれらを学ぶための第一歩となれば幸いです。

(参考資料)
SAS 9.2 SQL Procedure User's Guide
http://support.sas.com/documentation/cdl/en/sqlproc/62086/HTML/default/titlepage.htm

DATA Step vs. PROC SQL: What's a neophyte to do?
http://www2.sas.com/proceedings/sugi29/269-29.pdf

Efficiency Techniques for Beginning PROC SQL Users
http://www2.sas.com/proceedings/sugi29/127-29.pdf

Intermediate PROC SQL
http://www2.sas.com/proceedings/sugi23/Advtutor/p35.pdf

SAS Academic News

SAS アカデミック・ニュース

事例紹介、第4章SAS® Enterprise Guide® と基本統計の算出をご紹介いたします。
SAS Enterprise Guideには、通常メニューとウィザードメニューが用意されています。
通常メニューとは、各種統計手法による計算結果を算出でき、ウィザードメニューとは
ガイド機能に沿って操作し出力を得ることができます。本文では、これらの
メニューの使い分けや使用方法について説明しています。
SAS四方山話では、新企画として医学研究をテーマに4話にわたり、統計学を
もちいた研究デザインとその重要性についてご紹介いたします。

事例紹介

コラム
「SAS四方山話」

事例紹介

高柳 良太
國學院大學 経済学部および人間総合科学大学 人間科学部 兼任講師

第4章 SAS Enterprise Guideと基本統計の算出

今回は、SAS Enterprise Guide（以下、EG）の「記述統計」メニュー内で実行できる各種統計量の算出機能について、説明したいと思います。

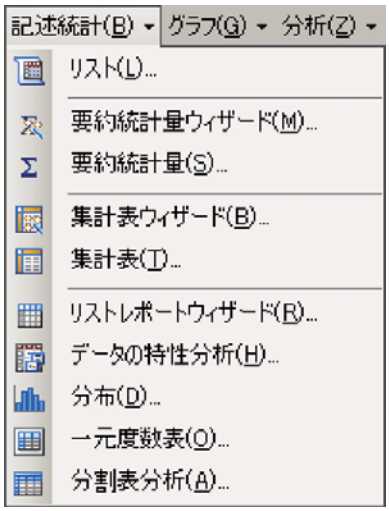
データ解析に慣れていない人に説明をしていると、データを入力したら、もしくは入力済みのデータを手に入れたら、即分析ができると考えている人が多いことに気づかされます。もちろん、大学での統計学や調査の講義では、そうではないことをこちらが教えていかないといけないので、学生たちがそう考えているのはある意味仕方がないでしょう。また、あまりほめられた話ではないですが、私が授業用に用意したデータに入力間違いがあることがあります。しかし、学生たちは先生が配るものが間違っているとは思っていないのです。もちろん、入力間違いはよくないですし、配るものに誤りがあるのは教育上よくはないことです。ただ、自分がミスしている可能性を考えもしない、得られたデータに間違いがあるかもしれないことを疑いもしないという姿勢は、データ解析をする場合に問題があるようにも思います。教える側としては、そういう部分も留意していかないといけないと痛感させられます。

データのチェック以外にも、手に入れたデータの傾向を把握することは大切なことです。しかし、データ解析の指導をしていたり、相談を受けたりするときに、このことがどうも忘れられているような気がしてなりません。根本的に、なにをしようと思ってデータを取ったのかという視点が欠けていることが多く、手法にばかり気を取られ、得たデータがどういう情報を持っているのかということを気にかけていない例が多いようにも思います。

統計学の授業などで統計量について説明する場合、教えているのが数学専攻の学生ではないこともあり、それぞれの統計量がどういう役割を持っているのか具体的に示してあげることが大切だと考えています。もちろん各統計量を計算できるようになることは大切ですが、それ以上にその統計量の意味を理解してもらうことが大切だと思います。得られた統計量から、データの傾向などを把握し、後に分析をかけた際にだいたいどんな結果になるかを予測できるようになる。データ解析の講義や指導では、もちろん個々の分析方法を習得するということも大切ですが、データ解析の全体が見通せる、方略を立てることができるようになることも大切です。

そのようなことを指導するのにも、EGの基本統計の算出機能は非常に役に立ちます。基本的な統計値の算出はExcelでもできますが、変数としてデータを管理できるという点で、EGには大きなメリットがあります。初学者や学生など、Excelしか使ったことがない人には、EGに抵抗を示す人もいますが、このメリットを知ってもらうだけでもだいぶ違います。

EGは、手軽にデータの傾向を把握するためのメニューが用意されています。基本的な統計量の算出は、[記述統計]メニューに集約されています。



EGの記述統計メニュー 一覧

前置きが長くなりましたが、今回はこの「記述統計」でできる基本統計の算出について説明をしていきたいと思います。

リスト

レポートのリスト表示

行番号	Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7-1	Q7-2	Q7-3	Q7-4	Q7-5	Q7-6	Q8	Q9	Q10	Q11
1	経済	1	1	2	5	2	40	1	.	.	1	.	.	2	2	2	10000
2	経済	1	4	1	7	1	80	1	.	1	.	.	.	1	5	1	30000
3	経済	1	2	2	6	3	60	.	1	.	.	.	.	4	2	1	10000
4	経済	2	5	1	6	1	75	1	.	.	.	.	.	5	1	2	30000
5	経済	1	10	2	3	4	40	.	1	.	.	.	.	5	1	1	15000
6	経済	1	6	1	4	3	60	1	.	.	.	.	.	6	3	1	50000
7	経済	1	6	1	4	1	40	1	.	1	.	.	.	1	2	2	10000
8	経済	1	6	1	6	1	40	.	1	1	.	.	.	2	3	2	10000

リスト出力例

「リスト」は、この出力例のように入力データを一覧出力します。後述のリストレポートとは異なります。

「リスト」では一覧表示したい変数を、「データ」画面の「タスクの役割」で「リスト変数」に指定します。

「リスト」の「データ」画面（抜粋）

「タスクの役割」で「合計」に指定した変数は、リスト末尾に合計が出力されます。マルチアンサーの変数を指定すると、回答数の把握ができます。「グループ変数」に属性となる変数を指定すれば、属性ごとのリストを出力することができます。

要約統計量

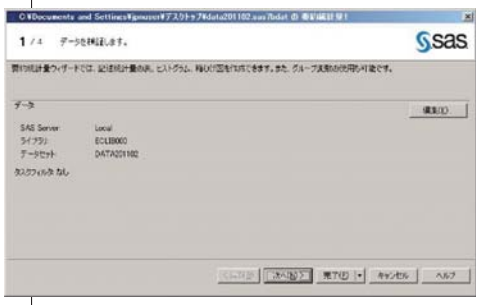
要約統計量
結果

MEANS プロシジャ

変数	平均	標準偏差	最小値	最大値	N
Q4	6.0125000	1.8724772	0	12.0000000	80
Q6	55.8125000	27.0775229	10.0000000	150.0000000	80
Q11	18287.50	16058.59	0	100000.00	80

要約統計量出力例

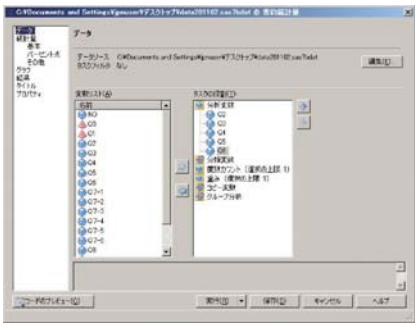
EGの「記述統計量」メニューには、「要約統計量ウィザード」と「要約統計量」の2種類の要約統計量メニューがあります。「要約統計量ウィザード」はウィザード（ガイド機能）メニュー従って指定を行えば、出力を得ることができます。



要約統計量ウィザード開始画面 下部にある「次へ」ボタンで進んでいく

ウィザードは質問に答えていく形式なので、EGに慣れていない人でも簡単に使うことができます。

「要約統計量」は、他のEGと同様のメニューです。講義等で説明をしながら使う場合や、操作に慣れている場合はこちらが便利だと思います。



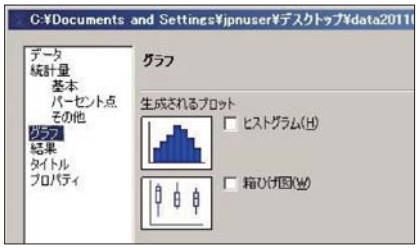
要約統計量メニュー

「要約統計量ウィザード」も「要約統計量」でも、出力される内容は同じです。どちらも出力したい統計量やグラフを任意に指定することができます。「要約統計量ウィザード」では、ウィザードの3番目のメニューで「編集」ボタンから統計量の指定ができ、同じ画面内ヒストグラムなどのグラフ作成の指定ができます。

要約統計量ウィザードの3番目　統計量やグラフ出力の指定ができる

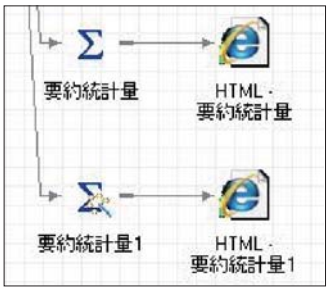
「要約統計量」では、統計量の指定は「統計量」から、グラフの指定は「グラフ」から行うことができます。

「要約統計量」メニューの画面（抜粋）
左「統計量」／下「グラフ」



EGの「要約統計量」では、統計量の出力指定を明確、かつ柔軟に行うことができます。統計学の講義等で統計量の説明をするときなど、意識的にクリックさせて統計量を出力させ、そして統計量について理解させるといった使い方をすることもできます。

同じ内容のものでウィザードメニューとそうでないものがあるときは、メニューやプロセスフローで表示されるウィザード側のアイコンに、語源（魔法使い）にちなんでキラッ！と光る魔法の杖が描かれています。後述の「集計表」も、同様にウィザード側のアイコンがキラッ！としています。もちろん、出力自体は変わりありません。



プロセスフローにて上が「要約統計量」、
下が「要約統計量ウィザード」のアイコン

集計表

「集計表」は属性などを分類変数に指定し、分類変数ごとに統計値を算出するものです。

集計表

		Mean	StdDev	Median	Min	Max	N
Q0							
経営	Q2	5.60	2.24	6.00	1.00	11.00	40
	Q3	2.08	1.23	2.00	1.00	5.00	40
	Q4	6.35	1.69	6.00	2.00	10.00	40
	Q5	2.05	0.99	2.00	1.00	4.00	40
	Q6	51.50	28.38	47.50	10.00	150.00	40
経済	Q2	5.38	2.13	6.00	1.00	12.00	40
	Q3	1.80	1.26	1.00	1.00	5.00	40
	Q4	5.68	2.00	6.00	0.00	12.00	40
	Q5	2.60	1.35	2.00	1.00	5.00	40
	Q6	60.13	25.33	52.50	25.00	120.00	40
Q2		5.49	2.18	6.00	1.00	12.00	80
Q3		1.94	1.25	1.00	1.00	5.00	80
Q4		6.01	1.87	6.00	0.00	12.00	80
Q5		2.33	1.21	2.00	1.00	5.00	80
Q6		55.81	27.08	50.00	10.00	150.00	80

「集計表」の出力例

「集計表」も、「要約統計量」と同様にウィザードと通常のメニューがあります。「要約統計量」でも分類変数を指定すれば属性毎の統計量を算出できますが、「集計表」は表作成の自由度が高く、より複雑な表形式での統計量出力が可能です。

上「集計表ウィザード」の画面／下「集計表」の画面

ネスト（入れ子）になった集計表など、「集計表ウィザード」でも「集計表」でも複雑な帳票を作成することができます。「集計表ウィザード」「集計表」ともに作成できる表は基本的に変わりがありませんが、「集計表」のほうがより細かく表の形式を指定することが可能です。

リストレポート

「リストレポート」は、ウィザードメニューのみが用意されています。「リストレポート」と前述の「リスト」の違いは、「リストレポート」は文字変数ごとに同種の行を要約したリストが出力されます。

リストレポート

	Q4	Q6	Q7-1	Q7-2	Q7-3	Q7-4	Q7-5	Q7-6	Q11
Q0	MEAN	MEAN	N	N	N	N	N	N	MEAN
経営	6.35	51.5	19	28	14	3	6	1	18775
経済	5.675	60.125	21	24	11	3	1	0	17800
合計			40	52	25	6	7	1	

リストレポート出力例

「リストレポート」は文字変数を属性のカテゴリとして、カテゴリ別の統計量を出力します。初期設定では、すべての数値変数の合計（Sum）が出力されるようになっています。ウィザードの2番目で、「編集」ボタンをクリックし「列の割り当て」から、出力する変数と統計量の種類を指定することができます。数値変数を文字変数のように分割するカテゴリとして使用したい場合は、「編集」ボタンの「列の割り当て」から「選択した列」の内容を「すべての値を表示する」に変更します。こうすることで、文字変数の同様の扱いとなります。

「リストレポート」の「リストの定義」画面
「編集」ボタンのメニューと「列の割り当て」

なお「リストレポート」は、文字変数の同じカテゴリで集計（要約）されるようになっています。これを「リスト」と同様にすべてのデータが表示されるようにしたい場合は、上図の「編集」ボタンのメニューから「テーブルのプロパティ」をクリックします。初期設定ではここが「同種の行を要約する（集計データ）」になっていますが、「すべての行を表示する（詳細データ）」を選択すると「リスト」と同じような出力になります。

データの特性分析

データに文字変数と数値変数が混在している場合、「データの特性分析」を使うと、変数の種類を気にすることなく一度でデータの傾向を把握できます。

カテゴリ変数の要約: ECLIB000.DATA201102

制限: 30 変数の最頻値

Variable	Label	Value	Frequency Count	Percent of Total Frequency
Q0		経営	40	50
		経済	40	50
Q1		1	56	70
		2	24	30

改ページ

数値変数の要約: ECLIB000.DATA201102

Variable	Label	N	NMiss	Total	Min	Mean	Median	Max	StdMean
NO		80	0	13640	101	170.50	170.5	240	5.77
Q10		80	0	102	1	1.28	1.0	2	0.05
Q11		80	0	1463000	0	18287.50	12000.0	100000	1795.40
Q2		80	0	439	1	5.49	6.0	12	0.24
Q3		80	0	155	1	1.94	1.0	5	0.14
Q4		80	0	481	0	6.01	6.0	12	0.21
Q5		80	0	186	1	2.33	2.0	5	0.14
Q6		80	0	4455	10	55.81	50.0	150	3.03
Q8		80	0	215	1	2.69	2.0	6	0.21
Q9		80	0	172	1	2.15	2.0	6	0.13
Q7-1		40	40	40	1	1.00	1.0	1	0.00
Q7-2		52	28	52	1	1.00	1.0	1	0.00
Q7-3		25	55	25	1	1.00	1.0	1	0.00
Q7-4		6	74	6	1	1.00	1.0	1	0.00
Q7-5		7	73	7	1	1.00	1.0	1	0.00
Q7-6		1	79	1	1	1.00	1.0	1	

「データの特性分析」出力例

「データの特性分析」では、文字変数はカテゴリごとの度数が算出されます。数値変数の場合は各種の統計量が算出されます。出力はN、欠損値、合計、最小値、平均値、最大値、中央値、最大値、標準偏差です。

「データの特性分析」画面

「データの特性分析」は「リストレポート」と同じくウィザードです。こちらはほぼオプション指定はなく、数値変数の統計値もありません。数値データと文字データを意識しなくても一気に出力が出るので、講義のようなときに使用するのには向かないかもしれませんが、業務等で簡便に結果を出したいときは重宝するでしょう。

分布

特定の分布と見なせるかどうかの検定を行います。また、ヒストグラム、確率プロットなども指定できます。これらについては次回詳しく説明したいと思います。

一元度数表

度数表は、その名の通り度数を集計するものです。度数のほかにパーセント、累積度数、累積パーセントも出力できます。また、棒グラフの出力も指定できます。

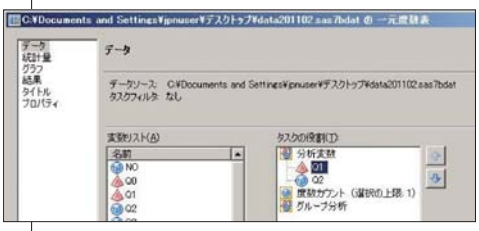
一元度数表
結果
FREQ プロシジャ

Q1	度数	パーセント	累積度数	累積パーセント
1	56	70.00	56	70.00
2	24	30.00	80	100.00

Q2	度数	パーセント	累積度数	累積パーセント
1	3	3.75	3	3.75
2	3	3.75	6	7.50
3	8	10.00	14	17.50
4	11	13.75	25	31.25
5	11	13.75	36	45.00
6	24	30.00	60	75.00
7	11	13.75	71	88.75
8	4	5.00	75	93.75
10	2	2.50	77	96.25
11	2	2.50	79	98.75
12	1	1.25	80	100.00

「一元度数表」出力例

「一元度数表」は、「データ」画面の「タスクの役割」で「分析変数」に指定した変数の度数を集計します。文字データになっている変数はもちろんですが、数値データの変数でも「分析変数」に指定すれば度数が算出されます。変数のプロパティを、数値から文字に変更する必要はありません。



「一元度数表」の「データ」画面(抜粋)

Excel等で入力したデータをインポートすると、数値であればSASの数値データとなります。そのデータで「データの特性分析」などを実行すると、統計量は算出されますが、度数は算出されません。その場合でも「一元度数表」で分析変数に指定すれば、数値変数でも度数が算出されます。

例えば順序変数として順位相関などを出したい場合など、データの種類を文字変数にしてしまうと、相関分析で変数として選択できなくなります。そのような変数でも、変数の種類を数値のままにしておいても、「一元度数表」を利用すれば数値のままで度数を算出できます。もちろん種類を変えていないので、相関等を行う場合でも分析変数として選択が可能です。

分割表分析

クロス集計表の算出を行います。χ二乗検定などもこのメニューから行います。こちらも「分布」とあわせて次回の説明にしたいと思います。

マルチアンサーの集計

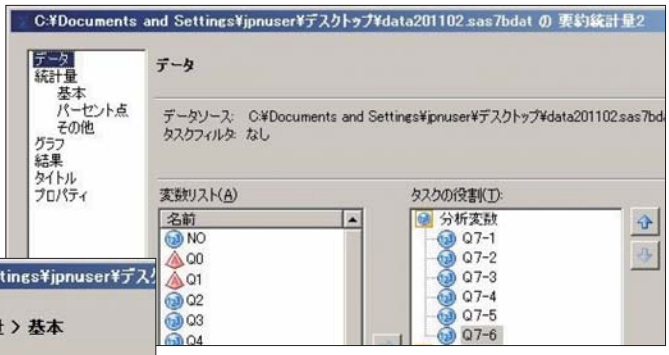
いわゆる複数回答のことですが、EGでマルチアンサーの集計も可能です。ちょっとマニアックな指定が必要な部分がありますが、複数回答データの比率を把握することができます。

集計表

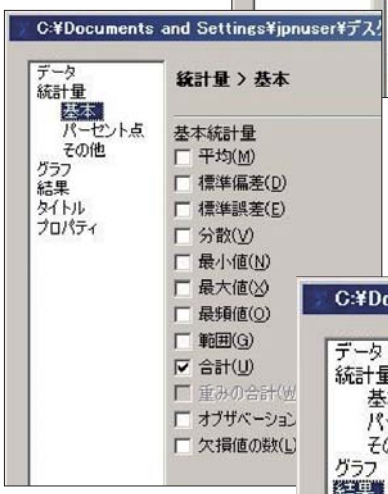
	Sum	PctSum
Q7-1_Sum	40.00	50.00
Q7-2_Sum	52.00	65.00
Q7-3_Sum	25.00	31.25
Q7-4_Sum	6.00	7.50
Q7-5_Sum	7.00	8.75
Q7-6_Sum	1.00	1.25

複数回答の割合を出した「集計表」出力
この場合Sumが度数PctSumが%

まず、マルチアンサーの集計となっている変数部分のみを、「要約統計量」で集計します。このとき、算出するのは「合計」だけです。また、出力結果をデータセットとしてファイルにして保存します。この保存したデータを使用して、「集計表」で分析をします。なお、ここでの集計結果を表示してもあまり意味がないので、データセットに保存する際に統計量表示のオプションを外してもよいでしょう。



「要約統計量」の「データ」画面(抜粋)
マルチアンサーの変数を「分析変数」に指定



「要約統計量」の「統計量」画面の
「基本」画面(抜粋)
「合計」だけをチェック

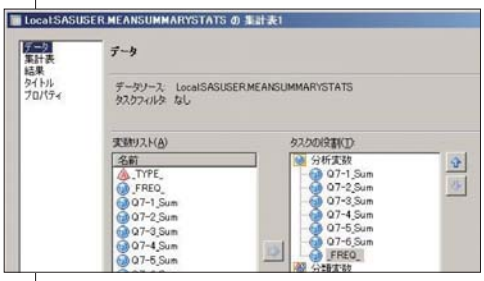
「要約統計量」の「結果」画面(抜粋)
「統計量をデータセットに保存する」にチェック

次に、今「要約統計量」で保存したデータセットを開きます。

	TYPE	_FREQ_	Q7-1_Sum	Q7-2_Sum	Q7-3_Sum	Q7-4_Sum	Q7-5_Sum	Q7-6_Sum
1	0	80	40	52	25	6	7	1

「要約統計量」で保存されたデータセット

このデータセットを、「集計表」で集計します。このデータセットには、算出された回答数合計が変数（「_FREQ_」）となって追加されていますが、それも分析変数に含めて集計します。

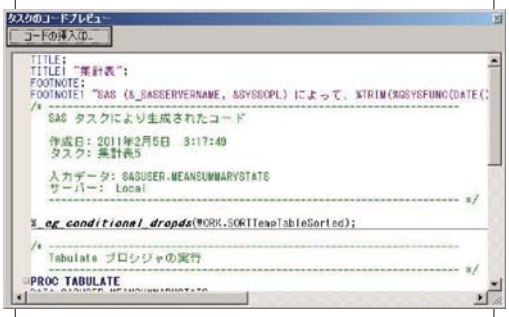


「集計表」の「データ」画面（抜粋）「_FREQ_」も分析変数に指定

次に集計表を指定します。ここで集計したい変数を指定しますが、合計である「_FREQ_」は集計表には指定しません。

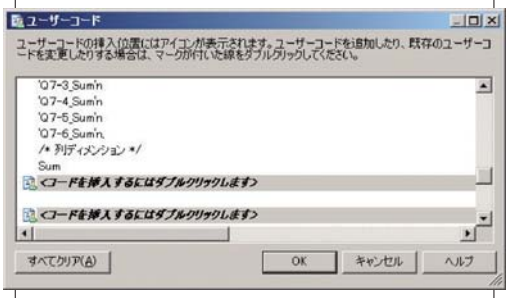
「集計表」の「集計表」画面（抜粋）
図のようにマルチアンサーの変数を指定する「_FREQ_」は指定しない。左下に「コードのプレビュー」ボタンがある

マルチアンサーの比率を出す場合、SASのプログラムコードを使います。使うといってもほんの少しだけです。「集計表」画面の左下に「コードのプレビュー」というボタンがあります。EGは、画面上はGUIで分析が実行されていますが、実際にはメニュー指定にあわせてSASプログラムが実行されています。その実行前のプログラムコードを、この「コードのプレビュー」で見ることができます。



「タスクのコードプレビュー」画面
上に「コードの挿入」ボタンがある

このプレビュー画面で、一番上にある「コードの挿入」ボタンをクリックすると「ユーザーコード」画面が表示され、コードを追加することができます。



「ユーザーコード」画面

「ユーザーコード」画面を下方向にスクロールしていくと、末尾の方に

```
                    /*列ディメンション*/
Sum
<コードを挿入するにはダブルクリックします>
```

という箇所があります。ここで <コードを挿入するにはダブルクリックします> をダブルクリックすると「ユーザーコードの入力」というウインドウが表示されます。

「ユーザーコードの入力」画面
これはコードを入力した状態　右に「OK」ボタンがある

ここで、

```
PctSum<_FREQ_>
```

と記述し、「OK」をクリックします。この「PctSum」は、レポートの合計比のことです。それを回答数合計の変数である「_FREQ_」で求めよというコードになっています。

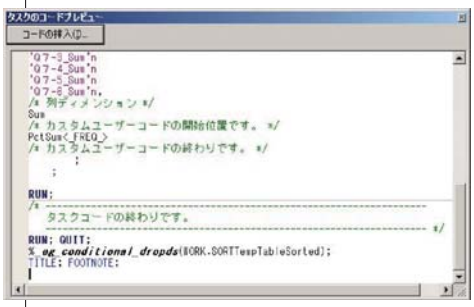
「OK」をクリックすると、「ユーザーコード」画面にコードが追加されます。前述の部分の <コードを挿入するにはダブルクリックします> が消えて

```
                    /*列ディメンション*/
Sum
PctSum<_FREQ_>
```

となっています。

コードが追加された「ユーザーコード」画面
真中下に「OK」ボタンがある

ここで「OK」ボタンをクリックして、コードプレビュー画面に戻ります。



コードが追加された「タスクのコードプレビュー」画面
右上に小さな「×」ボタンがある

プレビュー画面上では

```
                    /*列ディメンション*/
Sum
```

の下に

```
/* カスタムユーザーコードの開始位置です。 */
PctSum<_FREQ_>
  /* カスタムユーザーコードの終わりです。 */
```

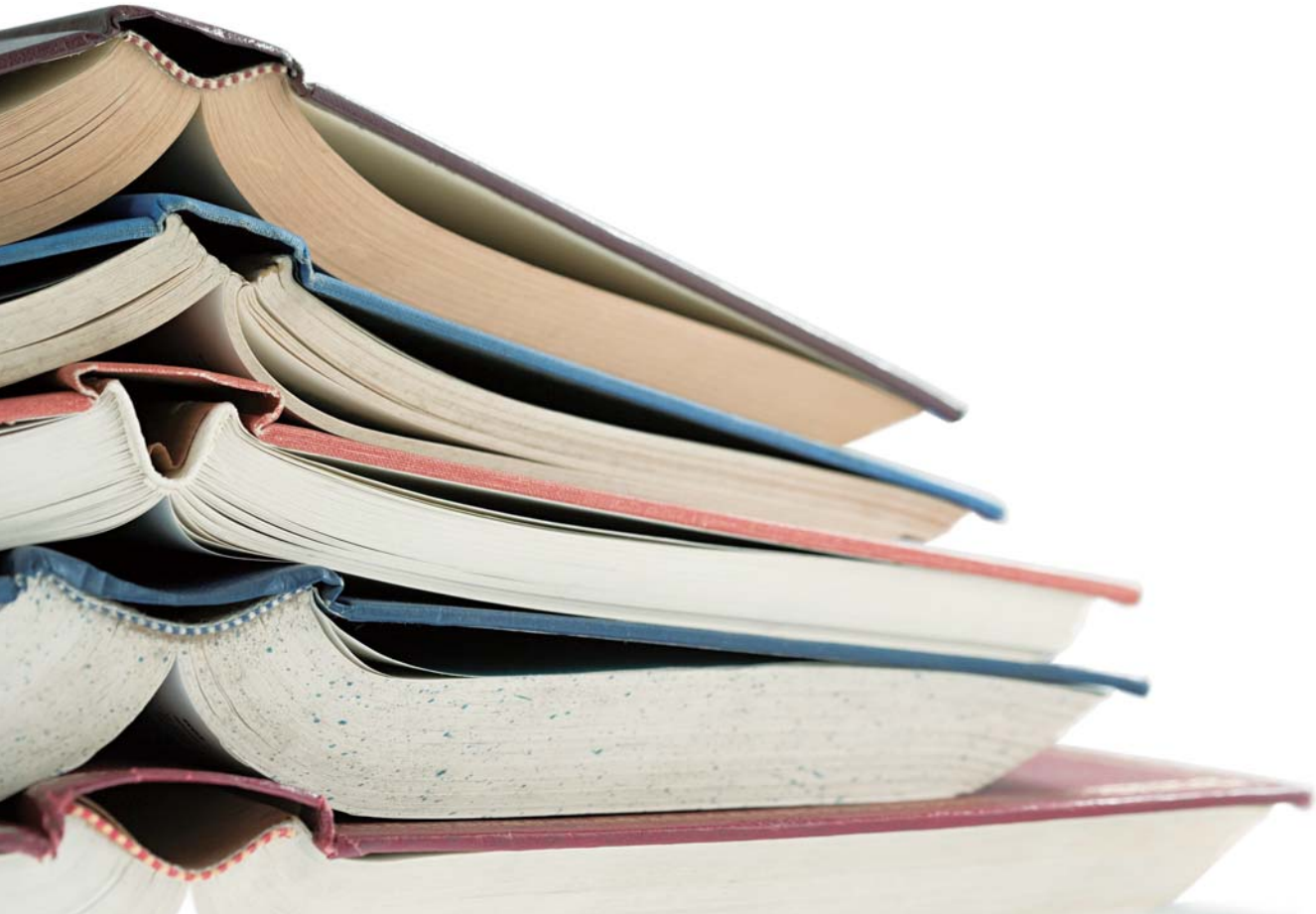
という記述が追加されているはずです。追加されていれば、右上の「×」をクリックしてウインドウを閉じて、集計表画面に戻ります。そして集計表画面で「実行」をクリックすると、マルチアンサーの回答数での割合が出力されます。

SAS プログラムというと敷居が高いという人はいるかもしれませんが、このようなちょっとしたカスタマイズに利用することができます。GUI メニューは確かに便利なのですが、微妙な部分がどうしても用意できないということがあります。そういう場合に、少しだけでいいので SAS プログラムの知識があると、このようなちょっと便利なことができます。

集計等で、もうちょっと便利にできないかなという場合、GUI にはメニューがなくても、このようにプログラム側から制御ができます。今回はその一例です。普段はプログラムコードを使うことがなくても、そういうことができるということを知っているだけで、EG の使い方も変わってきます。「できない」と思っていることが、ひょっとしたらできるかもしれないのです。

一から全部プログラムで分析を実行するのは、慣れていない人には大変ですし、業務がある人には勉強する時間も取りにくいでしょう。ただ、ちょっと一工夫というか、ほんの一手間かけるだけで、さらに EG を便利に使うことができます。プログラムだからといって毛嫌いせず、ほんの少しだけでいいから理解してもらえればよいなと思います。

今回は基本的な統計量の出力と、それに関連して SAS プログラムの非常に簡単な利用方法の説明をしました。次回は今回触れなかった「分布」と「分割表分析」の説明をしたいと思います。



コラム「SAS四方山話」

大橋　渉
千葉大学医学部疾患プロテオミクス寄附研究部門　データベース・インフォマティクス担当
ヤンセンファーマ株式会社　研究開発本部　臨床統計部　統計解析グループ　マネージャー　医学博士

第13回 医学研究のススメ（1）

まずは本年のテーマ

本年度は久々に筆者本来の医薬関連について、これから4回分にて述べさせていただきたく存じます。もちろん、他分野の方が読んでもそれなりに楽しんでいただけるように努力致しますので、何卒ご理解ください。筆者の経験則ではありますが、他分野の作法を知っているということは、単に視野を広げられるだけではありません。他分野におけるデータの見方やアプローチなどは、必ずご自身の分野における研究や業務にも役立ちますので、時にはご自身の分野について、より深く理解することにもつながります。

統計学はどのような分野にも必要な知識であり、英語などと同様に共通のツールである、と筆者は考えておりますので、絶対に「ムダな知識」になることはありません。例えば、統計学とは一番縁遠いであろうと思われる哲学や文学にも統計学が用いられていた、などという事実もありますが、他分野の方には（当該分野の方にも?）案外知られていなかったりします。「私は文科系出身なので統計学は…」などとおっしゃる方も多いのですが、統計学はいわゆる文科系の分野にも深く浸透しております。少々古いフレーズではありますが、「統計学は皆兄弟」のような広い心で統計学と向き合ってみてください。

調査・研究のススメ

皆様が何かを知りたいと思ったときに、それを明らかにするための方法は何でしょうか。昔だったら図書館で調べるのが一般的であったと思われますが、今であればインターネットのGoogleなどで検索をするという方も多いでしょう。他の方法も駆使して色々と探してみても見つからない場合には、知ること自体を諦めるか、もしくは**ご自身でさらに調べるか、自ら実験するか、の方法で確認する以外にありません**。普通であれば、Googleに聞いても分からないということであれば「ガッカリ」なのでしょうけれども、逆に研究の場合は「シテヤッタリ！」なのです。**まだ誰も答えを出していない問題に対して、ご自身が疑問を持ったということは、研究活動において極めて重要な「新規性」の問題をクリアしたことになるからです**＊。新規性が認められない論文が（まともな雑誌に）採択されることはまずありませんので、考えようによっては、研究上の一番大きな問題をクリアできたと言っても良いほどなのです。既に答えが出ている事柄であれば、調べてその結果を用いれば良いだけの話であり、おそらくGoogleなどの検索でも何らかの手がかりがヒットするのは間違いありません。また医学系の論文であれば、有名な「pubmed（http://www.ncbi.nlm.nih.gov/pubmed/）」で検索すれば出てくると思われますので、むしろヒットしなかったときこそ喜べるわけです＊＊。

以上のように、まずは研究を開始する前に、以前に自身の問題意識と同様の研究（先行研究と呼びます）がなされていないかを調べるのは、研究開始前の儀式（常識）といえるでしょう。こちらを調べることで、自身の問題意識もよりクリアになってくることは間違いありませんので、出来る限り念入りに調べてみることをお勧めします。Pubmedのtopページで色々とKeywordを入力して調べてみるのも方法ですが、その分野の研究者の方などに尋ねてみるのも、より確かな方法です。

医学研究の種類

知りたいことは見つかりました。続いてその「知りたいこと」の方向性や深さ、目的について考えてみましょう。

医学研究の場合ですと、基本的に「ヒト」を対象として実施されるものはすべてが臨床研究になります。マウスなどの実験動物を相手に実験をしている段階は**前臨床**と呼ばれ、まだヒトを対象としてはおりません。

一方、ヒトそのものではなくとも、ヒト由来の細胞や組織（流行の遺伝子解析等）を用いた研究は全て**臨床研究**になります。また、**ヒトの集団を見る場合にはそれらは疫学研究**になります。さらに疫学研究のうち、対象集団を複数群に分けて、それぞれの群に異なった治療や薬剤を与えて**比較を行う研究は臨床試験**です。また臨床試験のうち、厚生労働省から医薬品（医療機器）の承認を得ることを目的として行われるものが**治験**です。基本的に、内側に行くほど規則・規制が厳しくなっていきます。参考までに、一番内側にある**治験に関しては、日本国内はもちろんですが、世界的にもSASによる統計処理結果以外は認められておりません**。

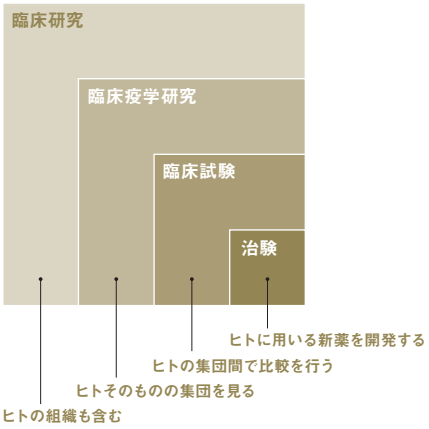


図1:臨床研究の種類と包含関係
（臨床研究人材教育コンソーシアムのHPより）

検証か探索か ～デザインの重要性～

1) 探索的研究とは

筆者が各所で主張しているもののひとつに、**研究デザインの重要性**があります。2009年のSASユーザー会でもお話させていただいただきましたが、筆者が受けてきた統計相談のうちの多くが研究終了後の「事後相談」であり、研究開始前のデザインに関する相談は全体の僅か30%弱程度でした。早い話が、既に収集されたデータを持参してきて、「**このデータから何かを言ってくれ！**」というような要望が全体の70%強程度ということなのですが、残念ながらほとんどのデータについては、その「何か」を言うことは不可能です。しかも**求められることの多くは検定（つまり仮説の検証です）**なのですが、適切なデザインがなされていない（**そもそも仮説がない！**）調査・研究のデータを解析しても、（一応p値は出ますが）それは適切な結果とは言えません。せめてもの「何か」として、次回の研究につなげるために調査・研究項目の**探索**ができました…と言えれば、まだ上出来な方です。

確かに無計画に集められたデータから何かが言えることも、時にはないわけではありませんが、**そこには一番大切であるはずの研究者の疑問といった問題意識は存在しません**。たとえ問題意識の発掘を目的とした**探索的研究**であっても、瓢箪から駒のような探索ではない、やはり何らかの疑問を持って臨む必要があります。

2) 検証的研究とは

探索的調査・研究を経て得られた仮説を検証するためには、必ず比較対照が必要になります。しかも、その比較対照としてどのような集団を集めるかは研究デザインの段階で明確に定められているため、研究の趣旨に合致しない被験者は除外されることになります。たとえば新薬Aと既存薬Bの効果を比較したい場合には、**それぞれの治療群A,Bの集団は可能な限り均一にしなければなりません**。他の要素が入り込む余地がある限りは、純粋に新薬Aの効果を確認することができませんので、たとえ本当に新薬Aの効果があったとしても、それを立証出来ないことになります注）。数字とグループ分けのフラグさえあれば検定自体は可能ですが、適切なデザインをされていない調査・研究から得られたp値や検定統計量には何の意味もありませんので、求めたところでもあくまで参考程度としてください。

注）比較可能性は？

例えば治療法Aと治療法Bが施された患者を100人ずつ比較して、A群は5年後に60人が、Bは同じく30人が生存していたという結果がありました。$\chi 2$検定により$p<0.001$と高度に有意になったので、私が開発した治療法AはBよりも優れていると言えます！
（拍手喝采…！）

ところが、

- **治療法A群の平均年齢は60歳、B群は80歳だった**
- **治療法A群は過去に手術歴なし、B群は平均1.5回の手術歴ありだった**

のような場合、余命や手術歴（=既に他の病気あり）が5年後の生存率に影響している可能性が非常に高いです。実際に治療法Aが優れていたとしても、このデザインではそれを立証することはできません。本当に効果があった場合は、非常にもったいないことになってしまいます。

図1の臨床試験よりも内側は比較対照を持ちますので、すべて検証的研究になります。検証的な研究には必ず検証するための仮説が必要であり、その仮説を検証するために各種検定を行います。その、検証したい項目のことを**評価項目（Endpoint）**と呼び、特にその調査・研究において一番知りたい項目を**主要評価項目（Primary Endpoint）**と呼びます。もしも、主要評価項目が存在しなければ仮説が設定できませんので、そもそも検定が成立しません。検定が成立しなければ、仮説検証型の研究デザインが成立しないのです。

3) サンプル数の設定

仮説検証型のデザインを行うためには、**仮説の設定根拠が必要です**。この場合の仮説の設定とは、対照群と比較して、反応率や生存割合などの評価項目にどの程度の差が見込まれるかの見積もりが必要ということです。そうすることによって、ようやく研究に必要な**必要最低限のサンプル数**を計算することができるのです。基本的には**過去の先行研究を前述のpubmedなどから検索してくるか、探索的な研究などを行って設定する**かのいずれかです。

「ちょっと待ってください、必要最低限って…サンプル数なんて多ければ多いほどいいのではないですか？」

…というご意見もあるかと存じますが、それこそちょっと待ってください。本当にそれでいいのでしょうか。ここで少し、研究を下記の観点から考えてみましょう。

調査・研究の実施可能性

①時間的制約：
限られた時間内で実施できるのか？

②経済的制約：
限られた予算内で実施できるのか？

③倫理的制約：
法的・心情的・倫理的観点から実施は可能なのか？

④技術的制約：
実施方法が存在するのか？

まずは①ですが、例えば視聴率のデータを求めるのに1年もかかってしまえば、そのデータは価値を失ってしまいます。医薬品も同様、特許期間の間しか独占的に販売することができませんので、企業として利益を確保するためにはできるだけ早く売り出す必要があります***。また、サンプル数が増えればそれだけ時間も経費も要しますので、②を考えれば、**サンプル数は必要最低限で行うのが良いことになります**。

③は上記以上に重要な問題であり、たとえば開発中の新薬であればどのような副

作用があるかわかりません。そのような「**得体の知れない薬」を多くの人に投与するということは、それだけ多くの人を副作用の危険にさらしてしまうことになりますので**、明らかに倫理的問題があります(④はまた別の機会に)。

また、右記の図3はおなじみのt検定の検定統計量ですが、こちらのnは両方とも、大きくなればなるほど、検定統計量Tの値は大きくなります。ということは、例えば身長差1mmなど、明らかに意味がない差であったとしても、サンプル数さえ大きければTの値は大きくなるということです。結果として、医学的や常識的に全く意味のない差であっても、統計的検定上は有意になってしまうということに他なりません。

$$T = \frac{\bar{X}_A - \bar{X}_B}{S\sqrt{1/n_A + 1/n_B}}$$

X_A:A群の平均値
X_B:B群の平均値
S :A群およびB群の標準偏差(分岐は等しい)
n_A: A群の例数
n_B: B群の例数

図2: t検定の検定統計量は…nが大きくなれば大きくなりやすい!

SASによるサンプル数の設定

SASでは、上記のt検定をはじめとした各種統計手法を用いた検定に対応したサンプル数の設定機能があり、power procedureにより算出されます。下の囲みは、主要評価項目についておなじみのt検定を行う場合の症例数設定のプログラムです。Ver9.1以降に搭載されたものゆえ、出力はまだ英語のみですが、他にもχ2検定やLog-rank検定など、多くの手法による解析に対応した症例数設定が可能になっています。この事例ではnを求めるプログラムを紹介させて頂きましたが、例えばnを固定して検出力を空白にすることも可能です。n数が決まっていてpower(検出力)が不明な場合には、power=.とすれば検出力を求めることができます。

```
<症例数設定 Power Procedure>
proc power;
  twosamplemeans test=diff (平均値の差の検定=t検定)
  meandiff    =1 (見込まれる平均値の差)
  stddev      =5 (見込まれる標準偏差)
  alpha       =0.05 (有意水準)
  power       =0.8 (検出力)
  ntotal      = .;
```

$$N = \frac{2\{z_{\alpha/2} + z_{\beta}\}^2 SD^2}{\Delta^2}$$

N :必要サンプル数
SD:個体間のバラツキの大きさ
Δ :見込まれる平均値の差
Zα/2:有意水準から算出する値(通常5%で1.96)
Zβ:検出力から算出する値(通常80%で0.84)
****

図3: t検定による必要症例数の設定プログラムと公式

実際に図3のプログラムを実行しますと、図4の出力が得られます。見込まれる平均値の差が1と小さく、標準偏差が5と大きいため必要なnの値は大きくなります。図3の公式を見ても、Δ(分母)が大きければ大きいほど、SD(分子の一部)は小さければ小さいほど必要症例数は小さくなることがわかります。

以上、医学研究におけるサンプル数は「多ければ多いほど良い」というものではないことが、ご理解いただけたかと存じます。アンケートなどは特に「こんなに集まった」と歓喜する場面が良く見受けられますが、臨床試験ではさまざまな事情から、必要最低限しか集めないようにしているのです。

次号は引き続き他の手法による症例数設計のプログラムと、論文の見方等についてご説明させていただきます。

*しかし、どれほど新規性があったとしても、その研究を行うこと自体に意味や価値があるか否かというのはまた別問題です。まずは第一関門というだけですので、誤解なきよう。

**通常、英文論文であればpubmedには掲載されますが、中には掲載されない雑誌もあることはあります。

***あくまで特許期間は**出願後受理された時点から20年**ですので、実際に独占的な利益を得られるのは発売開始後の話です。特例による5年間の延長が得られたとしても、実際に発売出来る期間はせいぜい5年～10年ですので、売り出すのは1日も早い方が良いのです。

****図3における式は一般的に用いられる近似公式となります。POWERプロシジャにて用いている手法、O'Brien and Muller (1993)、とほぼ同じ結果となりますが、厳密には異なります。また、近似公式では各群にて必要とされるサンプル数となっていますが、POWERプロシジャでは2群併せて必要なサンプル数が出力されています。

※各群におけるサンプル数を求める場合には、NTOTAL=オプションではなくNPERGROUP=オプションを用います。

```
The POWER Procedure
Two-sample t Test for Mean Difference

Fixed Scenario Elements

Distribution                Normal
Method                       Exact
Alpha                         0.05
Mean Difference                  1
Standard Deviation               5
Nominal Power                  0.8
Number of Sides                  2
Null Difference                  0
Group 1 Weight                   1
Group 2 Weight                   1

Computed N Total

Actual      N
 Power    Total

0.801      788
```

(t検定による2つの平均値の差)
正規分布
有意水準
平均値の差
標準偏差
両側検定の場合2(片側で1)
必要な症例数

図4 SASによる出力

Q&A

- グラフ枠の非表示
- 時点ごとの平均推移グラフの作成
- ODS のアウトプットに使用されているスタイルを調べたい
- INFILEステートメントにて複数の文字を区切り文字として指定する
- 残存したWORKライブラリの削除について
- EXPORTプロシジャにてCSVファイルを作成する際、変数名出力を制御する方法
- ODSで出力されるファイルを1つのファイルへ圧縮する方法

Q 散布図を作成していますが、グラフの枠(右側、上側の線)が表示されます。この枠を非表示とできますか。

A GPLOTプロシジャでの散布図作成の場合、PLOTステートメントのNOFRAMEオプションを用います。

例

```
PROC GPLOT DATA=sashelp.class;
                          /* NOFRAME オプションの追加 */
  PLOT weight*height=sex / NOFRAME;
  SYMBOL1 I=none V=dot C=red;
  SYMBOL2 I=none V=dot C=blue;
RUN;
QUIT;
```

散布図の作成はSGPLOTプロシジャにても行うことができますが、枠の非表示に対応しているオプションはありません。このため、ユーザー定義のスタイルを作成、適用し、枠を非表示とします。

例

```
                          /* ユーザー定義のスタイル作成 */
PROC TEMPLATE;
  DEFINE STYLE styles.mystyle;
    PARENT=styles.default;
                        /* FRAMEBORDER を OFF に設定 */
    STYLE GRAPHWALLS FROM GRAPHWALLS / FRAMEBORDER=OFF;
  END;
RUN;
                              /* 作成したスタイル適用 */
ODS LISTING STYLE=styles.mystyle;
                                    /* 散布図の作成 */
PROC SGPLOT DATA=sashelp.class;
  SCATTER Y=weight X=height / GROUP=sex;
RUN;
```

Q 時点ごとの平均値の推移グラフにおいて、エラーバーも併せて表示するため、以下のSYMBOLステートメントを用いました。

SYMBOL1 I=STD2TJ V=DOT C=red;

各オブザベーションの値に対してシンボルが表示されますが、平均値のみシンボルにて表示することはできますでしょうか。

A SYMBOLステートメントにて直接、対応している指定はありません。しかしながら、ANNOTATE機能を用いて、GPLOTプロシジャにて作成したグラフにシンボルを追加することができます。ただし、この場合、事前に平均値を算出しておく必要があります。

例

```
                                  /* 事前に平均値を算出 */
PROC MEANS DATA=test NWAY NOPRINT;
  CLASS time;
  VAR resp;
  OUTPUT OUT=out(DROP=_TYPE_ _FREQ_) MEAN=resp;
RUN;
                            /* ANNOTATE データセットを作成 */
DATA anno;
  SET out;
  RETAIN XSYS YSYS '2' WHEN 'A' COLOR 'red';
  FUNCTION="SYMBOL"; TEXT="dot"; X=time; Y=resp;OUTPUT;
RUN;
                          /* GPLOT プロシジャにてグラフ作成 */
PROC GPLOT DATA=test ANNOTATE=anno;
  PLOT resp*time;
  SYMBOL1 I=STD2TJ V=none C=red;
RUN;
QUIT;
```

また、他の方法としてはSGPLOTプロシジャを用いることができます。この場合、事前に平均値を算出しておく必要はありません。

例

```
                        /* SGPLOT プロシジャにてグラフ作成 */
PROC SGPLOT DATA=test;
  VLINE time / RESPONSE=resp STAT=MEAN LIMITS=BOTH MARKERS
                                          /* シンボルの指定 */
               MARKERATTRS=(SYMBOL="CircleFilled" COLOR="red")
                                              /* 線の指定 */
               LINEATTRS=(COLOR="red");
RUN;
```

Q TEMPLATE プロシジャでスタイルを作成する際、どのスタイルエレメントがアウトプットのどの部分に反映されているのかを簡単に調べる方法はありますか。

A ODS TAGSETS.STYLE_DISPLAY を使用すると、スタイルエレメントを確認するための特殊なアウトプットを生成することができます。

例

```
ODS TAGSETS.STYLE_DISPLAY
   FILE="c:¥temp¥stylecheck.html" STYLE=STYLES.RTF;

TITLE 'TITLE のメッセージ ';
FOOTNOTE 'FOOTNOTE のメッセージ ';

PROC REPORT DATA=sashelp.class nowd;
   COLUMN  ( sex name age height weight );
   DEFINE  sex    / GROUP;
   DEFINE  name   / GROUP;
   DEFINE  age    / ORDER;
   DEFINE  height / ORDER;
   DEFINE  weight / ORDER;
RUN;

ODS TAGSETS.STYLE_DISPLAY CLOSE;
```

このプログラムで作成された HTML ファイルを Internet Explorer で開き、ウィンドウ内の任意の場所をクリックすると、その場所で使用されているスタイルエレメントの定義が以下のような形で表示されます。

例

```
STYLE Data /
   FONT_FACE = "'MS PMincho', 'Times Roman', serif"
   FONT_SIZE = 10pt
   FONT_WEIGHT = medium
   FONT_STYLE = roman
   FOREGROUND = cx000000
;
```

Q 複数の連結文字が区切り文字として入っているファイルを正常にインポートするにはどうしたら良いでしょうか。

A SAS® 9.1ではINFILEステートメントのDLMオプションにて区切り文字を指定することが可能です。しかし、DLMオプションは単一文字を指定するオプションであるため、複数の連結文字を区切り文字を指定することができません。この場合はDATAステップにて追加のロジックを記述することで可能になります。一例として区切り文字を他の文字に変換した後、区切られた文字をSAS変数に代入する方法が考えられます。
右の例ではその一例を示します。

SAS 9.1にて複数の連結文字を区切り文字に指定する例:

```
DATA DLMTEST;
  INFILE DATALINES;
  INPUT;
                        /*「{sep}」を「/」文字に置き換える */
  _infile_=TRANWRD(_INFILE_,"{sep}","/");

          /* 区切り文字で区切った一番目の文字を数値に変換 */
  N1=INPUT(SCAN(_infile_,1,"/"),32.);
  LENGTH N2 $ 8;

          /* 区切り文字で区切った二番目の文字を変数に代入 */
  N2=SCAN(_infile_,2,"/");

          /* 区切り文字で区切った三番目の文字を変数に代入 */

  N3=INPUT(SCAN(_infile_,3,"/"),32.);
DATALINES;
123{sep}sep{sep}789
0{sep}ABCXYZ{sep}456
run;
```

SAS® 9.2では、複数の連結文字列を区切り文字として指定するDLMSTRオプションが追加されています。区切り文字に大文字小文字が含まれる場合は、DLMOPTオプションに「I」を指定します。

SAS 9.2にて複数の連結文字を区切り文字に指定する例:

```
DATA dlmtest;
    /* DLMSTR、DLMSOPT オプションを使用して文字列を区切る */
  INFILE DATALINES DLMSTR="{sep}" DLMSOPT="I";
  INPUT n1 n2 $ n3;
DATALINES;
123{sep}sep{sep}789
0{Sep}ABCXYZ{SEP}456
RUN;
```

Q Unix版SASでは、CLEANWORKというツールにて、残存したWORKライブラリの削除を行うことができますが、Windows版SASに同様のツールはありますか？

A SAS® 9.2より、標準でWindows ディスククリーンアップツールのCOMフックとして、残存したWORKライブラリを削除可能なツールがインストールされます。

例）
Windowsエクスプローラを開き、ドライブを右クリック→プロパティ、"ディスクのクリーンアップ"を選択し、"SAS Temporary Files"という項目がありますので、こちらにチェックをいれ、OKボタンを押していただくことで、残存しているWORKライブラリを削除できます。

Q EXPORTプロシジャを使用して、CSVファイルを作成しています。デフォルトではCSVファイルの先頭行に変数名が出力されますが、これをデータ行から出力するように変更できますか？

A SAS® 9.2の新機能であるPUTNAMES=NOオプションを指定することで、データ行から出力されます。

例

```
PROC EXPORT DATA=sashelp.class
            OUTFILE='c:¥temp¥noheader_92.csv'
            DBMS=CSV REPLACE;
  PUTNAMES=NO;
RUN;
```

SAS® 8またはSAS® 9.1.3を使用している場合は、以下のようなDATAステップを実行します。

例

```
DATA _NULL_;
  SET sashelp.class;
  FILE "c:¥temp¥noheader_913.csv" DSD;
  PUT name sex age height weight;
RUN;
```

Q 1つのプログラムの中でODS HTML、ODS RTFステートメントを使用しています。たくさんのグラフや結果を出力しているため、ファイル数も多く、ファイルサイズも大きく管理が大変です。これらのファイルを一つのファイルにまとめるような方法はありますか？

A SAS® 9.2では、出力されたグラフや結果を1つのZIP形式の圧縮ファイルへまとめられるODS PACKAGEステートメントが追加されました。これにより、出力する結果、グラフなどが多く、ファイルサイズが大きい場合でも圧縮して1つのファイルとして出力できます。

例

```
                                    /* パッケージ化開始 */

ODS PACKAGE OPEN;

出力内容
                /* 作成した内容を ZIP 形式の圧縮ファイルへ出力 */
ODS PACKAGE PUBLISH ARCHIVE
PROPERTIES(ARCHIVE_NAME=" ファイル名 " ARCHIVE_PATH=" 出力パス ");

                                    /* パッケージ化終了 */
ODS PACKAGE CLOSE;

                                    /* サンプルデータ作成 */
PROC SORT DATA = sashelp.class OUT = sample;
  BY age;
RUN;

ODS LISTING CLOSE;

                                    /* パッケージ化開始 */
ODS PACKAGE OPEN;

ODS HTML PACKAGE;
PROC GPLOT DATA = sample;
  PLOT height * weight;
  BY age;
RUN;
QUIT;
ODS HTML CLOSE;
                /* 作成した内容を ZIP 形式の圧縮ファイルへ出力 */
ODS PACKAGE PUBLISH ARCHIVE
PROPERTIES(ARCHIVE_NAME="Sample.zip" ARCHIVE_PATH="c:¥temp");
ODS PACKAGE CLOSE;
```

上記の例では、変数AGEごとに出力される結果をSample.zipファイルとして出力します。

ODS PACKAGEステートメントの詳細に関しましては、ドキュメントをご確認ください。
http://support.sas.com/rnd/base/ods/odspackages/statement.html

SAS Training

SAS トレーニングのお知らせ

プラットフォーム系3試験の日本語版提供開始、BI試験練習問題にチャレンジ！

2010年12月よりSASのプラットフォーム系の3試験：SAS BI Content Development for SAS® 9（以下、BI試験）、SAS Data Integration Development for SAS® 9（以下、DI試験）、SAS Platform Administration for SAS 9®（以下、PA試験）が日本語で受験できるようになりました。
今回はこれら3試験の大まかな位置づけを紹介し、BI試験のサンプル問題を掲載します。
※試験範囲の詳細などは下記Webページより確認可能です。
www.sas.com/jp/training/certify/index.html

試験の位置付け

企業では膨大なデータが日々記録されており、そのデータから何らかの価値のある知見を見つけ出し、今後の予測や経営上の判断を下したいという要望があります。
弊社ではこの要望に応える製品群として、SAS Analytics Platformを提供しています。SAS Educationでは業務上の役割（ジョブロール）に従って主に使用するPlatform製品を図1のようにまとめており、日本語化された3試験の試験範囲もこのジョブロールに基づいています。

SAS Analytics Platformのシステムは多くの製品を組み合せるため、複雑に感じるかもしれません。そこで、今回は企業内の膨大なデータから価値ある情報を提供する過程を、食材（データ）からおいしい料理（分析結果）を作る過程とみなして解説します。

■美食家：Business User

シェフが作成したおいしい料理を食べるお客さんです。ただ、単に食べて満足するだけではなく、「なぜこのような料理が出来上がったのか、その理由がわかる」、「さらにおいしい料理になるには、どのような材料を使った方がよいのか」などの判定も可能な美食家のイメージです。
実務上では、作成された分析結果・指標から経営判断を行う方や、簡単な分析レポートを作成する担当者に該当します。

■シェフ：BI Content Developer

食材を適切に調理して、おいしい料理を作る人です。もちろん味見もして作った料理の味判定も可能です。BI試験では、上記2つの役割を試験範囲としています。
業務上ではビジネスユーザーから要求されるレポートの作成や、ビジネスユーザーが分析の切り口として使用するさまざまなコンテンツ（Information Map、Stored Process、OLAPキューブ、経営ダッシュボード）を提供します。

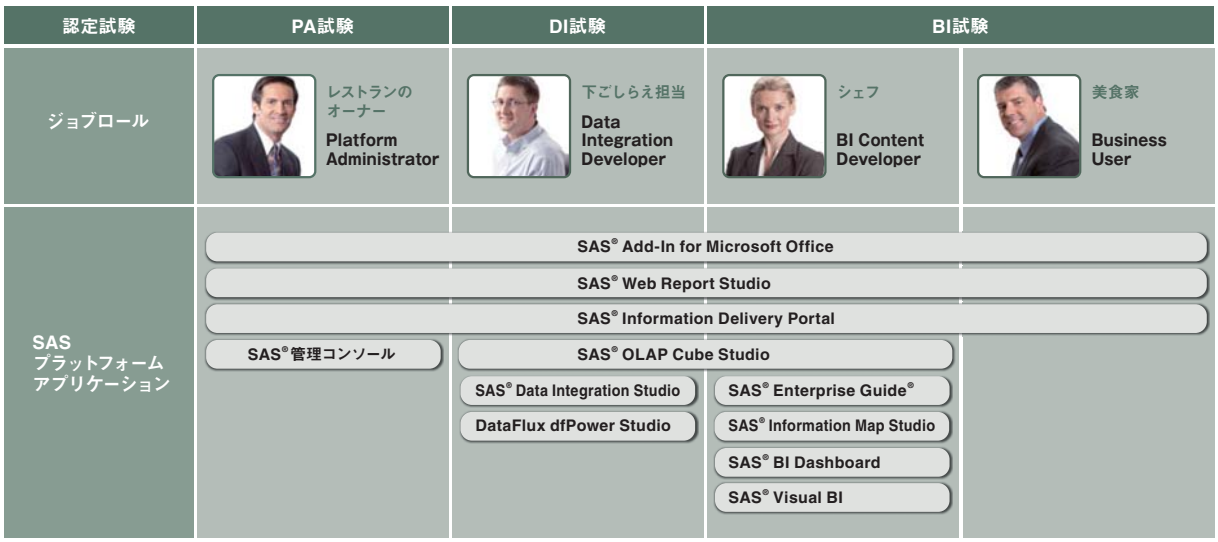
■下ごしらえ担当：Data Integration Developer

このレストランには、非常に大量な食材があるため、適切な食材の下ごしらえを行う担当がいます。各地に存在する食材を厳選し、膨大な食材を煮込んで特製スープを作るような作業を担当します。この役割に対する試験がDI試験です。
実際では分析の目的に適した形式にデータの結合・変換などを行う、データウェアハウスやデータマートの作成者です。

■レストランのオーナー：Platform Administrator

レストラン全体の支配人です。誰にどのような業務を担当させるのか、役割分担などを指揮します。いわゆる全体的な管理を行う役割であり、PA試験が該当します。
システム上ではサーバー、データ、ユーザーなど、SASリソース全体の管理を行います。

図1：認定試験別のジョブロールと主なアプリケーション



BI試験の練習問題にチャレンジ！

以下のSASプラットフォーム・アプリケーションの説明として間違っているものを選択しなさい。

A.

AMO（SAS Add-In for Microsoft Office）を使用して直接OLAPキューブの情報をドリルダウンできる。

B.

WRS（SAS Web Report Studio）はWebベースのレポート作成アプリケーションである。

C.

IDP（SAS Information Delivery Portal）をクライアントPCにインストールすると、企業内の情報を効率的に検索できる。

D.

EG（SAS Enterprise Guide）を使用するとレポート作成だけでなく、ストアドプロセスの作成も可能である。

選択肢A～Cの3製品はビジネスユーザー向けのアプリケーションとして位置付けています。これらはすべてテーブルの結合やフィルタリングなどのシステム的な知識がなくても、普段のPC操作で使い慣れている、WebブラウザやMicrosoft Office製品を使用してSASの分析結果にアクセス可能です。IDPはWRSと同様にWebアプリケーションであるためクライアントPCにインストールする必要はありません。従ってCが正解です。
ちなみにIDPは分析レポートのPortal（入口）となるサイトです。PortalのWebページからWRSへのリンクも作成できます。

［ 正解：C ］

プログラミング系の試験対策はe-Learningで！

・プログラミング系の2試験ではe-Learning形式の練習問題を提供しています。
・受験直前の力だめしや試験対策にご活用ください。
・実際の試験配分を考慮しているため、弱点の把握にも役立ちます。

e-Learning名	アクセス期限	申込み価格
SAS®認定プロフェッショナル模擬試験: SAS Base Programmer for SAS®9	180日間	7,350円
SAS®認定プロフェッショナル模擬試験: SAS Advanced Programmer for SAS®9	180日間	7,350円

新規トレーニング・コースの開催のご案内

●「事例によるデータ分析法入門」コース（1日間）

［日　　程］
2011年6月6日（月） 10:00 ～ 17:00（東京会場）
2011年6月20日（月） 10:00 ～ 17:00（大阪会場）
［価　　格］
57,750円（税込）／チケット捺印数1
［受講対象］
データ分析法の習得に興味のある方
マーケティング担当者、営業企画・営業管・経営計画等担当者
［学習内容］
ビジネス分析の基礎となる知識を、身近な事例と共にご紹介します。
・基本統計量（平均、分散、標準偏差）
・各種分布（t、χ2、F）
・平均値の差の検定
・相関分析（相関関係、相関係数）
・回帰分析（単回帰、重回帰）
事例については、SAS® Enterprise Guide® を使用して、実際に分析結果を確認します。
［Web Page］
www.sas.com/jp/training/course/da_intro_cs.html

●「Microsoft OfficeによるSAS分析入門」コース（1日間）

［日　　程］
2011年7月4日（月） 10:00 ～ 17:00（東京会場）
［価　　格］
57,750円（税込）／チケット捺印数1
［受講対象］
Excelスプレッド・シートを使用してビジネス分析を行いたい方
［学習内容］
よく知られたMicrosoft Excelのインターフェイスを通して、SASの分析のパワーを使用する方法を学習します。
本コースでは、SAS® Add-In for Microsoft Officeの機能を紹介します。SAS Add-Inを使用することで効果のあるようなシナリオを通して、ビジネスの問題を解決します。
・スプレッド・シート内でSAS分析タスクを実行するためのSASツールバーの操作
・グラフの作成
・要約タスクの実行や箱ひげ図の描画によるデータ分析
・線形回帰分析の実行
・クラスタ分析によるデータの探索
［Web Page］
www.sas.com/jp/training/course/mso_ai.html

新人研修キャンペーンのご案内

毎年好評いただいております新人研修キャンペーンを今年も期間限定で実施します。
お得な「Basic Plan」を6月30日までに「新人研修キャンペーン」専用申込書にてお申し込みをいただくと20%割引でご受講いただけます。
[Web Page]
www.sas.com/jp/training/newcomer.html

SAS Institute Japan株式会社では、今後も多岐にわたったトレーニングコースを追加していく予定です。

コース内容・日程等の詳細は、順次弊社Webサイトに公開しますので、以下のURLをご参照ください。
http://www.sas.com/jp/training/

その他、トレーニングに関する情報については、上記のURLをご参照いただくか、下記トレーニング担当までお問い合わせください。

トレーニング担当

T E L: 03-6434-3690　F A X: 03-6434-3691
E-mail: JPNTraining@sas.com

Latest Releases

最新リリース情報

PCプラットフォーム

Windows版	SAS 9.1.3 / 9.2
64-bit Windows(Itanium)版	SAS 9.1.3 / 9.2

メインフレームプラットフォーム

IBM版(OS/390,z/OS)	SAS 9.1.3 / 9.2

UNIXプラットフォーム

SunOS/Solaris版	SAS 9.1.3 / 9.2
HP-UX版	SAS 9.1.3 / 9.2
HP-UX(Itanium)版	SAS 9.1.3 / 9.2
AIX版	SAS 9.1.3 / 9.2
Linux(Intel) 版	SAS 9.1.3 / 9.2
IBM版(OS/390,z/OS)	SAS 9.1.3 / 9.2

SAS Technical News入手

SAS Technical Newsは、右記のURLから入手できます。 **http://www.sas.com/jp/periodicals/technews/index.html**

SASユーザー総会

アカデミア／テクノロジー&ソリューションセッション

毎年、多くのお客様にご参加いただいておりますSASユーザー総会は、今年で30回目を迎えます。これも日頃からSASをご愛顧いただいておりますお客様のおかげであり、心より感謝申し上げます。
今回30回目開催を記念し、また西日本地区お客様からの熱いご要望にお応えするべく、初の関西(神戸)開催が決定いたしました。
是非ともこの機会に進化を続けるSASをご体感いただきたく、皆様のご参加をお待ちしております。

また、同時にお客様からSASを使っての論文やプレゼンテーション等も募集しております。
奮ってご応募くださいますようお願いいたします。

■日程

2011年7月28日(木)～29日(金)
※SASユーザー総会前後に特別セッションを企画しております。
7月27日は基礎講座、7月30日はSAS Institute Japanエデュケーション部主催の「SAS Learning Session 2011」を予定しています。

■会場

神戸国際会議場
住所:〒650-0046 神戸市中央区港島中町6-9-1
アクセス:ポートライナーで神戸空港より8分、三宮駅より10分

■内容

内容の詳細は、5月以降にSASホームページにてご案内を予定しております。
・お客様による発表(論文、プレゼンテーション、ポスター)
・基礎講座(分析、データマネジメント、製品基礎など)
・SAS製品ハンズオン、製品/ソリューション紹介セッション
・優秀発表表彰式、懇親会

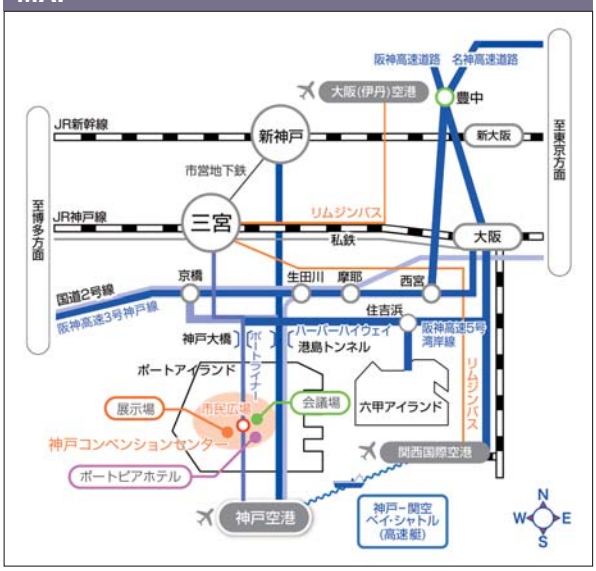
■お申し込み

2011年の発表募集、参加募集ともにSASホームページにて5月頃ご案内を予定しております。
昨年開催の内容はこちらをご参照ください。
http://www.sas.com/offices/asiapacific/japan/usergroups/

■お問い合わせ先

SASマーケティング本部
T E L: 03-6434-3702　E-mail: jpnsaswg@sas.com

MAP



発行:SAS Institute Japan株式会社

STN SAS Technical News SPRING 2011

■テクニカルニュースに関するお問い合わせ先
テクニカルサポートグループ
TEL:03-6434-3680　FAX:03-6434-3681

SAS Institute Japan株式会社

本社
〒106-6111
東京都港区六本木6-10-1
六本木ヒルズ森タワー 11F
Tel 03(6434)3000
Fax 03(6434)3001

大阪支店
〒530-0004
大阪市北区堂島浜1-4-16
アクア堂島西館 12F
Tel 06(6345)5700
Fax 06(6345)5655

www.sas.com/jp

このカタログに記載された内容は改良のため、予告なく仕様・性能を変更する場合があります。あらかじめご了承ください。SASロゴ、The Power to Knowは米国SAS Institute Inc.の登録商標です。その他記載のブランド、商品名は、一般の各社の登録商標です。
Copyright©2011, SAS Institute Inc.All rights reserved.