

臨床試験のデータ加工における SAS/Proc SQLの活用

—事例：データセット併合と図表作成—

持田製薬株式会社

岩崎 晋也*

高田 康行

矢島 勉

発表内容

- Proc SQLについて
- Proc SQLの活用事例の紹介
- Proc SQLへの提案
- まとめ

Proc SQLについて

■ データステップの拡張

<主な機能>

- ・SASデータセットの併合
- ・SASデータセットからの抽出 など

■ Proc SQLの活用事例の紹介

- ・データセット併合
- ・図表作成のためのデータ抽出

データセット併合

■2つのデータセットの併合

臨床検査の測定値
と
臨床検査の基準値

事例1

測定値データセット

被験者	性別	項目番号	項目名	時点	測定値
1	M	1	TC	0	256
1	M	1	TC	8	237
1	M	1	TC	16	284
1	M	1	TC	24	309
1	M	2	HDL	0	245
1	M	2	HDL	8	254
1	M	2	HDL	16	261
1	M	2	HDL	24	266

key key

基準値データセット

項目番号	項目名	性別	下側基準値	上側基準値
1	TC	.	150	220
2	HDL	M	40	85
2	HDL	F	40	95

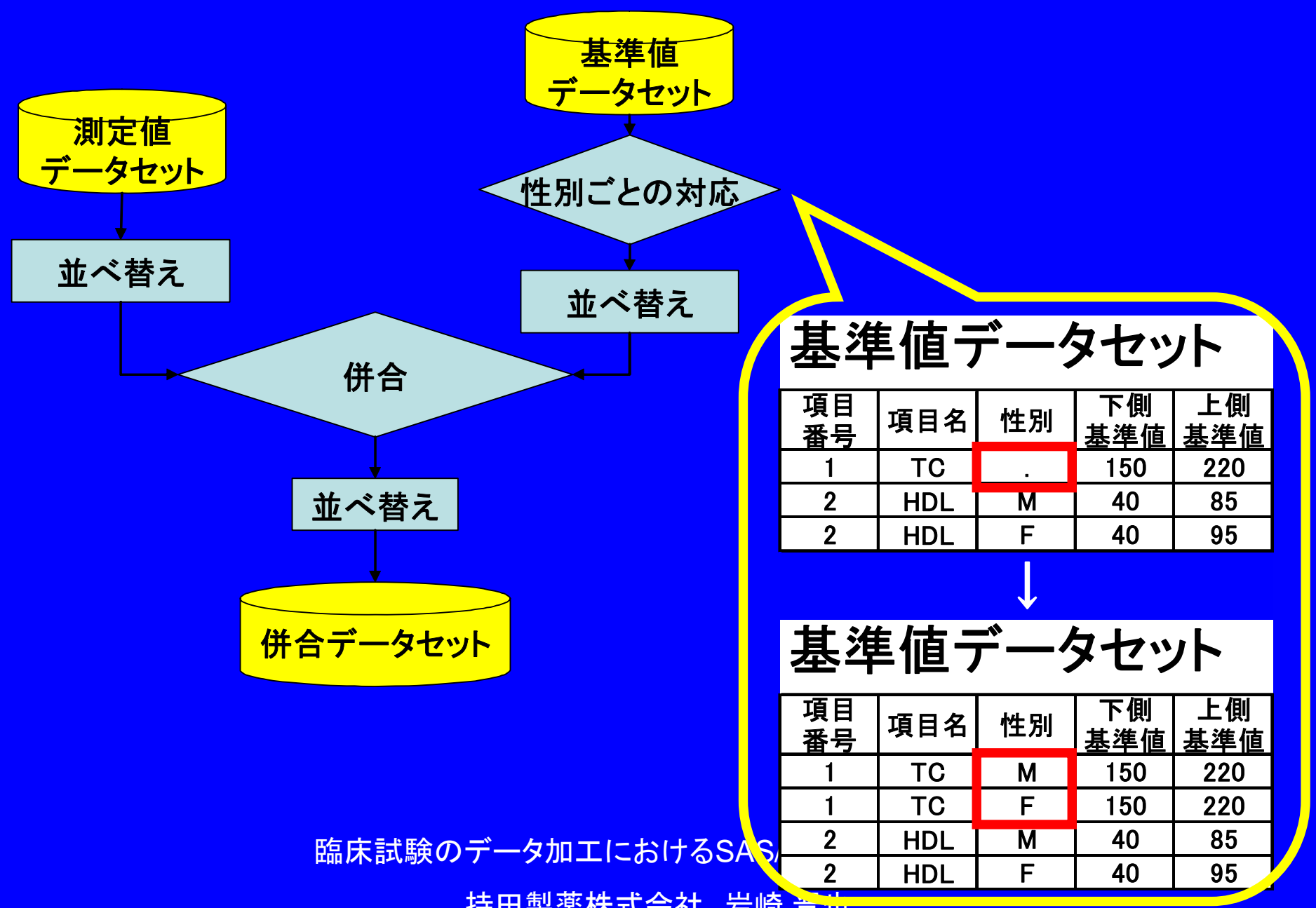
key

key

併合データセット

被験者	性別	項目番号	項目名	時点	測定値	下側基準値	上側基準値
1	M	1	TC	0	256	150	220
1	M	1	TC	8	237	150	220
1	M	1	TC	16	284	150	220
1	M	1	TC	24	309	150	220
1	M	2	HDL	0	245	40	85
1	M	2	HDL	8	254	40	85
1	M	2	HDL	16	261	40	85
1	M	2	HDL	24	266	40	85

事例1



Proc SQLを使わない場合

- ①測定値データセットの並べ替え
Proc SORT data=測定値データセット;
by 項目番号 性別;
run ;
- ②併合させるための基準値データセットの加工
(性別ごとに基準値を作成)
Data 基準値データセット2; set
基準値データセット;
if 性別ごとに基準値がある場合 then そのまま出力;
if 性別ごとに基準値がない場合 then 性別ごとに基準値を出力;
run ;
- ③加工した基準値データセット
(基準値データセット2)の並べ替え
Proc SORT data=基準値データセット2;
by 項目番号 性別;
run ;
- ④併合
Data 併合データセット;
merge測定値データセット 基準値データセット2;
by 項目番号 性別;
keep被験者 項目番号 項目名 測定値 下側基準値 上側基準値;
run ;
- ⑤併合データセットの並べ替え
Proc SORT data=併合データセット;
by 被験者 項目番号;
run ;
- 臨床試験のデータ加工におけるSAS/Proc SQLの活用
- 持田製薬株式会社 岩崎 晋也



事例1

作成するデータセットの名称を指定

Proc SQLを使う場合



Proc SQL ;

作成するデータセットに保存する変数を指定

create table

併合データセット as

元のデータセットを指定

select

被験者, 測定値データセット.性別, 項目番号, 項目名,
測定値, 下側基準値, 上側基準値

from

測定値データセット left join 基準値データセット

on

測定値データセット.項目番号 =

基準値データセット.項目番号

where

基準値データセット.性別 = . or

測定値データセット.性別 = 基準値データセット.性別

order by

被験者, 項目番号

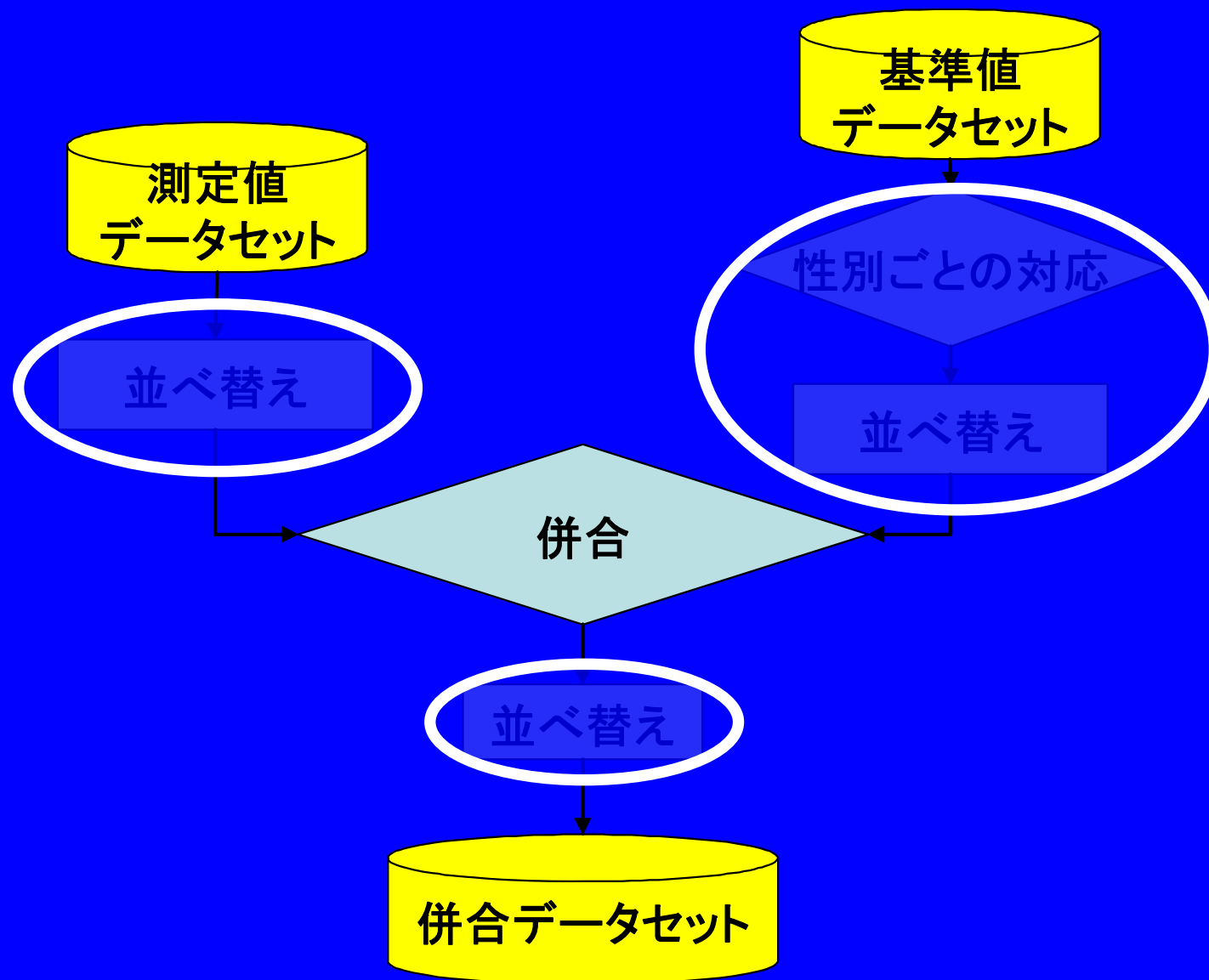
併合の際のkeyを指定

併合後の並べ替えを指定

データ抽出の条件を指定

臨床試験のデータ加工におけるSAS/Proc SQLの活用

事例1



臨床試験のデータ加工におけるSAS/Proc SQLの活用

持田製薬株式会社 岩崎 晋也

Proc SQLの利点①

- 事前の並べ替え不要
- 併合と同時に並べ替え可能

並べ替えと併合の繰り返し

■記述の繰り返しが必要

- ・データセット名 ・変数名 など

■不具合が発生した場合

- ・発生箇所特定の労力が大
- ・不具合の修正箇所が多数

Proc SQLの利点②

■ データステップ

1 : N 対応

のみ併合可能

key		key	
被験者	性別	被験者	項目名
1	M	1	TC
2	F	1	HDL
3	M	2	TC
4	F	2	HDL
5	M	3	TC
6	F	3	HDL
7	M	4	TC
8	F	4	HDL

■ Proc SQL

M : N 対応

でも併合可能

被験者	時点
1	0
1	8
1	16
1	24
2	0
2	8
2	16
2	24

臨床試験のデータ加工におけるSAS/Proc SQLの活用

key

M : N 対応のマッチマージ

key

被験者	項目名
1	TC
1	HDL
2	TC
2	HDL
3	TC
3	HDL
4	TC
4	HDL



被験者	時点
1	0
1	8
1	16
1	24
2	0
2	8
2	16
2	24

key

<データステップ>

被験者	項目名	時点
1	TC	0
1	HDL	8
1	HDL	16
1	HDL	24
2	TC	0
2	HDL	8
2	HDL	16
2	HDL	24

key

<Proc SQL>

被験者	項目名	時点
1	TC	0
1	TC	8
1	TC	16
1	TC	24
1	HDL	0
1	HDL	8
1	HDL	16
1	HDL	24
2	TC	0
2	TC	8
2	TC	16
2	TC	24
2	HDL	0
2	HDL	8
2	HDL	16
2	HDL	24

key

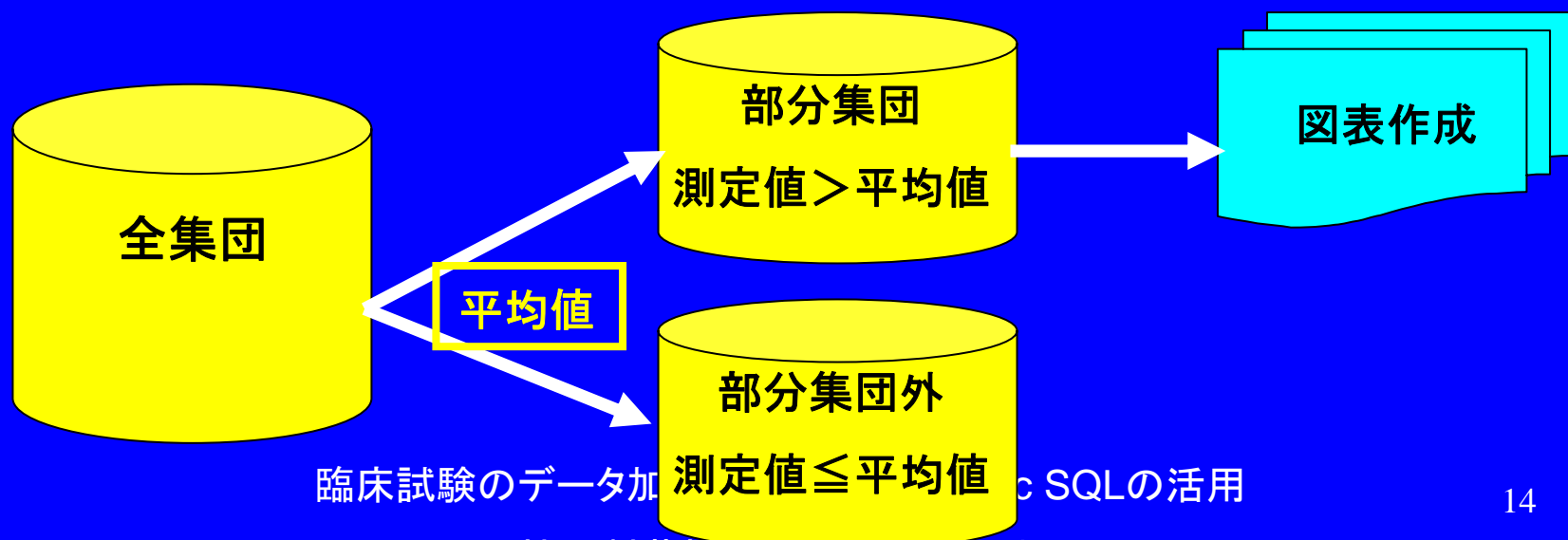
臨床試験のデータ加工におけるSAS/Proc SQLの活用

図表作成のためのデータ抽出

- 部分集団の図表作成
- 部分集団の抽出

部分集団の
抽出が必要

測定値 > 全被験者の平均値



事例2

測定値データセット

被験者	項目番号	項目名	時点	測定値
1	1	TC	0	256
1	1	TC	8	237
1	1	TC	16	284
1	1	TC	24	309
2	1	TC	0	245
2	1	TC	8	254
2	1	TC	16	261
2	1	TC	24	266



抽出データセット

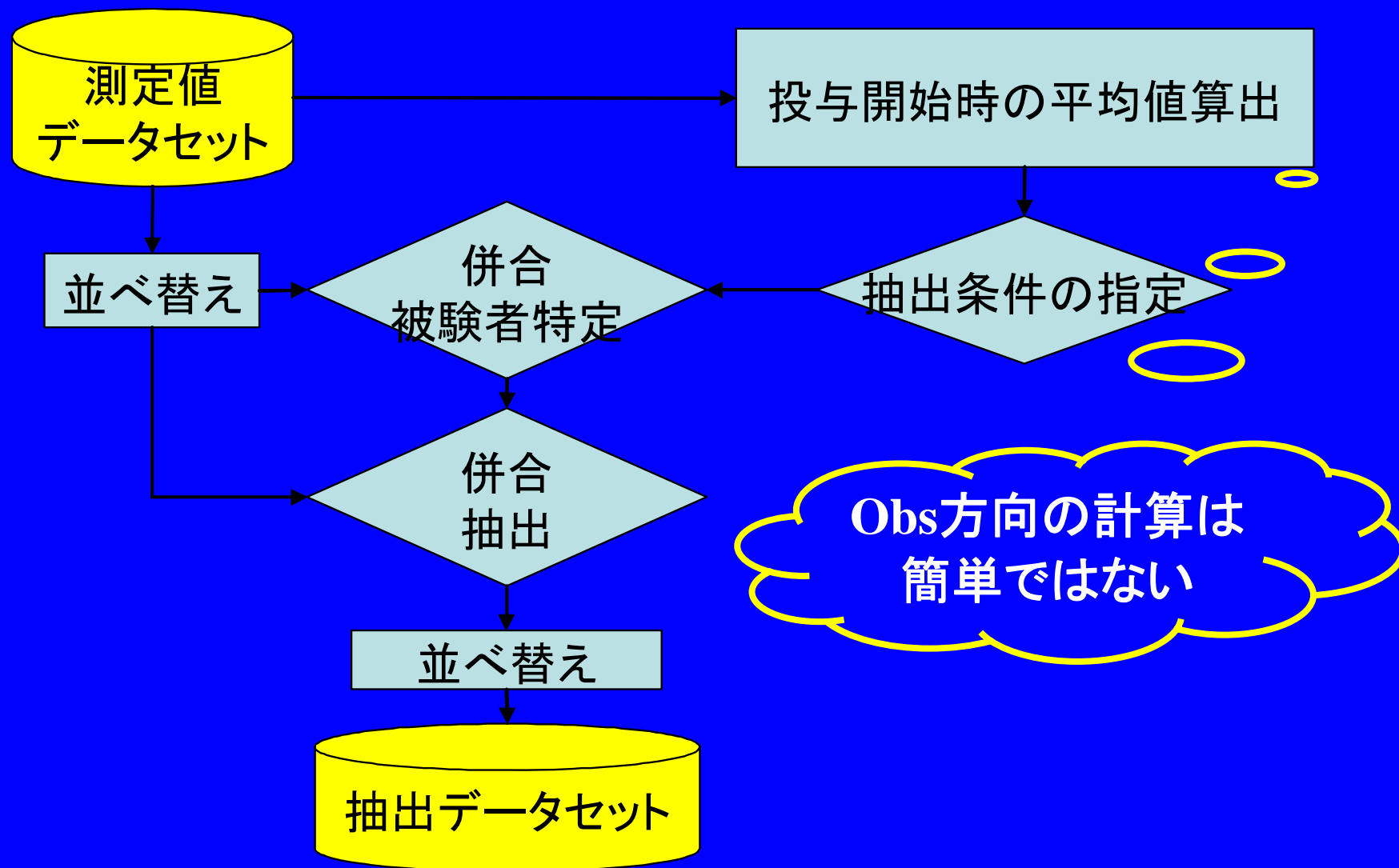
被験者	項目番号	項目名	時点	測定値	平均値
1	1	TC	0	256	250.095
1	1	TC	8	237	250.095
1	1	TC	16	284	250.095
1	1	TC	24	309	250.095
2	1	TC	0	245	250.095
2	1	TC	8	254	250.095
2	1	TC	16	261	250.095
2	1	TC	24	266	250.095

平均値は250.095

測定値 > 平均値
抽出する

測定値 ≤ 平均値
抽出しない

事例2



Proc SQLを使わない場合

①測定値データセットの並べ替え
Proc SORT data=測定値データセット;
by 被験者 時点;
run;

②測定値データセットから投与開始時の平均値を算出
Proc MEANS data = 測定値データセット;
where 時点 = 0;
var 測定値;
output out =
PMEAN mean = MEAN;
run;

③算出した平均値をマクロ化
data _null_ ; set PMEAN;
call
symput("PMEAN",MEAN);
run;

④測定値が平均値よりも高い被験者を特定
data SUBJ ; set 測定値データセット;
where 時点 = 0 and
測定値 > &PMEAN. ;
keep 被験者 ;
run;

⑤併合後、測定値が平均値よりも高い被験者を抽出
data 抽出データセット;
merge 測定値データセット SUBJ(in = a);
by 被験者;
if a;





Proc SQLを使う場合

```
Proc SQL ;  
  create table 抽出データセット as  
  select  
  from  
  where  
  where  
  被験者 in
```

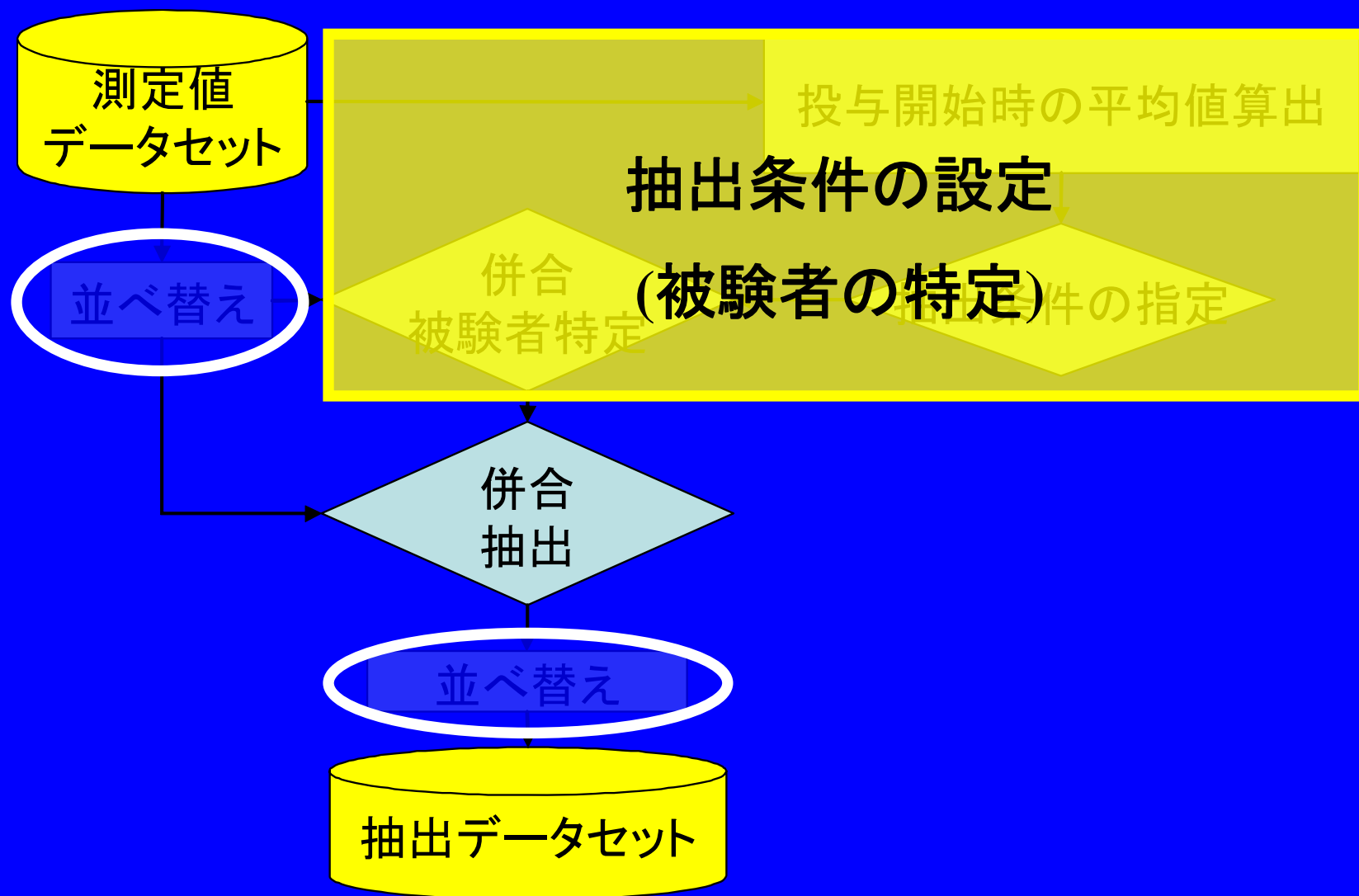
抽出条件の設定
(被験者特定)

(select
 from
 where
 having
 被験者
 測定値データセット
 時点 = 0
 測定値 > avg(測定値))

```
order by 被験者, 時点
```

集計するための関数を用いる場合のデータ抽出の条件を指定

事例2



Proc SQLの利点③

■ Obs方向の計算が可能

→測定値とObs方向の計算値の比較が可能

Proc SQLの留意点

■ 複雑なデータ加工をする場合

- Proc SQL

- 1プログラムにまとめるデータ加工

- 複雑な全体構造になる可能性

- データステップ

- 1つ1つ準備作業をしながらのデータ加工

Proc SQLへの提案

■ 臨床試験で用いる集計するための関数

- ・平均値
- ・標準偏差
- ・最小値
- ・最大値
- ・25%点
- ・中央値
- ・75%点 など

■ 集計するための関数の充実

- ・Proc MEANS
- ・Proc UNIVARIATE など

まとめ

- Proc SQLの活用事例の紹介

- プログラム行数の節約

 - シンプルなプログラム

 - ヒューマンエラーの減少

ご清聴、ありがとうございました