libname ex 'D:\data';

**proc** **import** out=sasuser.a1 datafile="D:\data\question.xlsx" dbms=xlsx replace;

sheet='a1';

getnames=yes;

**run**;

**proc** **import** out=sasuser.a2 datafile="D:\data\question.xlsx" dbms=xlsx replace;

sheet='a2';

getnames=yes;

**run**;

**proc** **import** out=sasuser.b1 datafile="D:\data\question.xlsx" dbms=xlsx replace;

sheet='b1';

getnames=yes;

**run**;

**proc** **import** out=sasuser.b2 datafile="D:\data\question.xlsx" dbms=xlsx replace;

sheet='b2';

getnames=yes;

**run**;

**proc** **contents** data=sasuser.a1;**run**;

**proc** **contents** data=sasuser.b1;**run**;

/\*a1 表中height 是char,b1表中height 是numeric\*/

**data** sasuser.a1;

set sasuser.a1;

height1=input(height,**8.**);/\*把变量height转成字符型的，长度为8位\*/

drop height;

**run**;

/\*纵向合并a1,b1表\*/

**data** ab1;

set sasuser.a1 sasuser.b1(rename=(ht=height1 wt=weight));

rename height1=height;

**proc** **print**;

**run**;

/\*a2中，变量y2和y4是字符型的，要转成数值型的，长度为8位\*/

**data** sasuser.a2;

set sasuser.a2;

format time yymmdd10.;

y22=input(y2,**8.**);/\*把变量height转成字符型的，长度为8位\*/

y44=input(y4,**8.**);

drop y2 y4;

rename y22=y2 y44=y4;

**run**;

/\*纵向合并a2,b2表,两个表所有的变量类型都一样了\*/

**data** ab2;

set sasuser.a2 sasuser.b2;

**proc** **print**;

**run**;

/\*总结：合并前表之间变量的类型一定要一致\*/

/\*a2和b2合并，为了区分两个人录入的记录区分开，我们用临时变量标识\*/

**data** ab2;

set sasuser.a2(in=a) sasuser.b2(in=b);/\*产生临时变量a，凡是是a2中的记录，a的值均为1；产生临时变量b,凡是b2中的记录，b的值均为1\*/

a2=a;/\*把临时变量a的值赋值给变量a2\*/

b2=b;/\*把临时变量b的值赋值给变量b2\*/

**proc** **print**;

**run**;

/\*横向合并ab1和ab2\*/

**data** ab;

merge ab1 ab2;/\*注意：如果在用by语句横向合并时，如果两个数据集事先没有按id排序，一定要用proc sort分别排序才能合并\*/

by id;

drop a2 b2;

**proc** **print**;**run**;

**data** cd;

merge ab1(in=d1) ab2(in=d2);/\*产生标识两个数据集的临时变量d1和d2\*/

by id; /\*以id为索引进行合并\*/

if d1=**1** and d2=**1**; /\*保留d1和d2都为1的记录\*/

**proc** **print**;**run**;

**proc** **import** out=ef datafile="D:\data\ef.xlsx" dbms=xlsx replace;

getnames=yes;

**run**;

/\*数据对比\*/

**proc** **compare** base=ab compare=ef nosummary transpose;/\*nosummary指定用于比较的两个数据集，transpose按记录显示不一致的结果\*/

by id;

id id;

**run**;

/\*发现不一致的地方改正一下\*/

**data** ab;

set ab;

format time date9.;

**proc** **print**;

**run**;

**data** xb;

set ab;

if id=**4** then time=**'16Jun2012'd**;

**proc** **print**;**run**;

/\*数据清洗-查找和删除重复值\*/

**proc** **sort** data=xb nouniquekey out=rep;/\*nouniquekey：输出重复值 out=rep把输出的重复值保存到数据集rep中\*/

by name gender;

**proc** **print** data=rep;**run**;

**proc** **sort** data=xb nodupkey out=norep;/\* nodupkey:输出唯一值 out=norep把输出的唯一值保存到数据集norep中\*/

by name gender;

**proc** **print** data=norep;**run**;

/\*查找缺失值的万能程序\*/

**data** missing;

set xb;

array cha[\*] \_character\_;/\*利用\*号不指定cha数组中的字符型变量个数\*/

do i=**1** to dim(cha);/\*指定循环次数为数组cha中的元素数（有多少个变量）\*/

if missing (cha[i]) then output;

end;

array num[\*] \_numeric\_;/\*利用\*号不指定num数组中的数值型变量个数\*/

do i=**1** to dim(num);/\*指定循环次数为数组num中的元素数（有多少个变量）\*/

if missing(num[i]) then output;

end;

**proc** **print**;**run**;

/\*查找异常值\*/

/\*定义异常值：\*/

/\*sex:except 1 and 2 ,age:<18 or age>50 or 空值,height:<150 or >200 or 空值，weight:<40 or >100 or 空值，y1-y5:1,2,3,4,5 和空值以外的值\*/

**data** outline;

set xb;

if (age^=**.** and (age<**18** or age>**50**))|(height^=**.** and (height<**150** or height>**200**))|

(weight^=**.** and (weight<**40** or weight>**100**))|(gender not in(**1**,**2**,**.**))|y1 not in (**1**,**2**,**3**,**4**,**5**,**.**)|

y2 not in (**1**,**2**,**3**,**4**,**5**,**.**)|y3 not in (**1**,**2**,**3**,**4**,**5**,**.**)|y4 not in (**1**,**2**,**3**,**4**,**5**,**.**)|y5 not in (**1**,**2**,**3**,**4**,**5**,**.**);

**proc** **print**;

**run**;

/\*利用万能宏程序查找异常值（只要修改带有标注的地方就可以了\*/

%let data=sasuser.xb; /\*此处改为自己的数据集\*/

%let id=id; /\*此处改为数据集中表示id号的变量\*/

**%macro** outline(var=,low=,high=);

data outline;

set &data.(keep=&id. &var.);

length variable $20. reason $20.;

variable="&var.";

value=&var.;

if &var.<&low. and not missing(&var.) then do;

reason="lower";

output;

end;

else if &var.>&high. and not missing(&var.) then do;

reason="higher";

output;

end;

drop &var.;

proc append base=outliner1 data=outline;

run;

**%mend** outline;

%***outline***(var=height,low=**150**,high=**200**);\*这几行分别添加查找的变量及上下限值；

%***outline***(var=weight,low=**40**,high=**100**);

%***outline***(var=y1,low=**1**,high=**5**);

%***outline***(var=y2,low=**1**,high=**5**);

%***outline***(var=y3,low=**1**,high=**5**);

%***outline***(var=y4,low=**1**,high=**5**);

%***outline***(var=y5,low=**1**,high=**5**);

**proc** **print** data=outliner1;

**run**;

/\*调查员重新返回几个变量的值，缺失值填补\*/

**data** xb\_revised;

set sasuser.xb;

if id=**6** then y3=**1**;

if id=**7** then y5=**5**;

if id=**9** then weight=**56**;

if id=**10** then height=**162**;

**proc** **print**;**run**;

/\*还有一些缺失值，通过mi（mutiple imputation)多重填补\*/

**data** xb9;

set xb\_revised;

if id=**9** then delete;

**run**;

/\*round=1 1 1 :指定3个变量的填补值都保留整数，minimum=150 1 1 :分别指定3个变量的最小值整数，maximun=200 5 5:分别指定3个变量的最大值整数;\*/

**proc** **mi** data=xb9 out=nomissing round=**1** **1** **1** minimum=**150** **1** **1** maximun=**200** **5** **5**;

mcmc;

var height y2 y4;

**run**;

**proc** **print** data=nomissing;

**run**;

/\*上面程序已经输出了5次缺失值填补值，我们用univariate过程来输出5次填补的均值\*/

**proc** **univariate** data=nomissing noprint;/\*这5行产生height,y2,y4的5次填补的均值，输出到数据集nm\*/

class id;

var height y2 y4;

output out=nm mean=height y2 y4;

**run**;

**data** newxb;

update xb9 nm;/\*利用update语句将新数据集nm中的数据更新到旧数据集xb9中，\*/

by id;

height=round(height,**1**);/\*利用round函数将3个变量保留为整数\*/

y2=round(y2,**1**);

y4=round(y4,**1**);

**proc** **print** data=newxb;**run**;