# Dataset Integrity Check for the Follow-up of Children Diagnosed with Diabetes (JDRF-TEDDY Follow-Up) Study

**Prepared by NIDDK-CR**
**December 6, 2021**

# Contents

# 1 Standard Disclaimer

The intent of this DSIC is to provide confidence that the data distributed by the NIDDK repository is a true copy of the study data. Our intent is not to assess the integrity of the statistical analyses reported by study investigators. As with all statistical analyses of complex datasets, complete replication of a set of statistical results should not be expected in secondary analysis. This occurs for a number of reasons including differences in the handling of missing data, restrictions on cases included in samples for a particular analysis, software coding used to define complex variables, etc. Experience suggests that most discrepancies can ordinarily be resolved by consultation with the study data coordinating center (DCC), however this process is labor-intensive for both DCC and Repository staff. It is thus not our policy to resolve every discrepancy that is observed in an integrity check. Specifically, we do not attempt to resolve minor or inconsequential discrepancies with published results or discrepancies that involve complex analyses, unless NIDDK Repository staff suspect that the observed discrepancy suggests that the dataset may have been corrupted in storage, transmission, or processing by repository staff. We do, however, document in footnotes to the integrity check those instances in which our secondary analyses produced results that were not fully consistent with those reported in the target publication.

# 2 Study Background

The Environmental Determinants of Diabetes in the Young (TEDDY) study enrolled over 8,000 infants identified to have an increased genetic risk for type 1 diabetes (T1D) and were followed prospectively for 15 years for the development of T1D and islet cell antibodies. There were 400 children projected to develop T1D over the course of the study, and follow-up of these children stopped upon development of T1D. The Follow-up of Children Diagnosed with Diabetes (JDRF-TEDDY Follow-Up) study sought to understand whether these children were diagnosed at an earlier stage of T1D compared to children not enrolled in prospective studies, and to identify if they maintained the ability to produce C-peptide longer than children diagnosed through standard clinical care in the community. The JDRF-TEDDY Follow-Up study analyzed the preservation of C-peptide over time in children diagnosed with T1D through prospective studies and compared them to a group of age matched controls identified from the community. Furthermore, the JDRF-TEDDY Follow-Up study collected samples to investigate immunological changes occurring after diagnosis and how these changes may relate to earlier T1D diagnoses. The JDRF-TEDDY Follow-Up study utilized a subset of the TEDDY study population consisting of 130 participants including 120 matched cases and controls.

# 3 Archived Datasets

All SAS data files, as provided by the Data Coordinating Center (DCC), are located in the JDRF-TEDDY Follow-Up folder in the data package. For this replication, variables were taken from the enrollment.sas7bdat, base_demographics_rev.sas7bdat, registration_rev.sas7bdat, base_diagnosis_diabetes.sas7bdat, base_medical_history1.sas7bdat, base_medical_history2.sas7bdat, physical_exam.sas7bdat, diagnosis_diabetes.sas7bdat, base_specimen.sas7bdat, base_specimen_information.sas7bdat, specimen.sas7bdat, specimen_info.sas7bdat, base_test_results.sas7bdat, and test_resuts.sas7bdat datasets.

# 4 Statistical Methods

Analyses were performed to replicate results for the data published by Steck et al. [1] for Factors Associated With the Decline of C-Peptide in a Cohort of Young Children Diagnosed With Type 1 Diabetes. To verify the integrity of the data, descriptive statistics were computed. Genetic and linear mixed model results were not replicated.

# 5 Results

For Table 1 in the publication [1], Characteristics of study participants, Table A lists the variables that were used in the replication, and Table B compares the results calculated from the archived data files to the results published in Table 1. The results of the replication are within expected variation to the published results.

# 6 Conclusions

The NIDDK Central Repository is confident that the JDRF-TEDDY Follow-Up data files to be distributed are a true copy of the study data.

# 7 References

[1] Steck AK, Liu X, Krischer JP, Haller MJ, Veijola R, Lundgren M, Ahmed S, Akolkar B, Toppari J, Hagopian WA, Rewers MJ, Larsson HE. Factors Associated With the Decline of C-Peptide in a Cohort of Young Children Diagnosed With Type 1 Diabetes. The Journal of Clinical Endocrinology and Metabolism, 106(3), e1380-e1388, March 2021. doi: https://doi.org/10.1210/clinem/dgaa715 PMCID: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8244121/

**Table A: Variables used to replicate Table 1** – Characteristics of study participants

| **Table Variable** | **dataset.variable** |
|---|---|
| Female, N (%) | enrollment.sex<br>base_demographics_rev.sex |
| FDR with T1D, N (%) | enrollment.FamilyType1Diabetes<br>base_demographics_rev.AnyFamilyMemT1D |
| Age at diagnosis (years) | registration_rev.t1d_diag_agedys |
| Diabetic ketoacidosis, N (%) | base_diagnosis_diabetes.DKAstatus |
| Weight z-score at diagnosis | base_medical_history1.weight<br>base_medical_history2.weight<br>physical_exam.weight<br>physical_exam.phys_agedys |
| Height z-score at diagnosis | base_medical_history1.lengthheight<br>base_medical_history2.lengthheight<br>physical_exam.lengthheight<br>physical_exam.phys_agedys |
| BMI z-score at diagnosis | base_medical_history1.lengthheight<br>base_medical_history2.lengthheight<br>physical_exam.lengthheight<br>base_medical_history1.weight<br>base_medical_history2.weight<br>physical_exam.weight<br>physical_exam.phys_agedys |
| HbA1c at diagnosis (%) | base_diagnosis_diabetes.HemoglobinA1cResult<br>diagnosis_diabetes.HemoglobinA1cResult |
| HbA1c at diagnosis (mmol/mol) | base_diagnosis_diabetes.HemoglobinA1cResult<br>diagnosis_diabetes.HemoglobinA1cResult |
| Positive autoantibodies, N(%) | base_test_results.test_name<br>test_results.test_name<br>base_test_results.due_num<br>test_results.due_num<br>base_test_results.outcome<br>test_results.outcome |

**Table B:** Comparison of values computed in integrity check to reference article Table 1 values

| Variable | Manuscript TEDDY Case (n=57) | DSIC Case (n=57) | Diff. (n=0) | Manuscript Control (n=56) | DSIC Control (n=56) | Diff. (n=0) |
|---|---|---|---|---|---|---|
| Female, N (%) | 25 (43.9) | 25 (43.9) | 0 (0) | 30 (53.6) | 30 (53.6) | 0 (0) |
| FDR with T1D, N (%) | 9 (15.8) | 9 (15.8) | 0 (0) | 5 (8.9) | 5 (8.9) | 0 (0) |
| Age at diagnosis (years) | 6.4 ± 1.8 | 6.4 ± 1.8 | 0 ± 0 | 6.7 ± 1.9 | 6.7 (1.9) | 0 ± 0 |
| Diabetic ketoacidosis, N (%) | 0 (0.0) | 1 (1.75) | 1 (1.75) | 9 (16.1) | 7 (12.5) | 2 (3.6) |
| Weight z-score at diagnosis | 0.6 ± 0.9 | 0.6 ± 0.8 | 0 ± 0.1 | 0.1 ± 0.9 | 0.4 ± 0.8 | 0.3 ± 0.1 |
| Height z-score at diagnosis | 0.8 ± 1.0 | 0.8 ± 1.0 | 0 ± 0 | 0.6 ± 1.0 | 0.5 ± 1.0 | 0.1 ± 0.0 |
| BMI z-score at diagnosis | 0.2 ± 1.1 | 0.2 ± 1.0 | 0 ± 0.1 | -0.4 ± 1.2 | 0.08 ± 0.8 | 0.48 ± 0.4 |
| HbA1c at diagnosis (%) | 6.9 ± 1.5 | 7.1 ± 1.5 | 0.2 ± 0 | 10.2 ± 2.3 | 10.6 ± 2.3 | 0.4 ± 0 |
| HbA1c at diagnosis (mmol/mol) | 52 ± 16 | 53.6 ± 15.8 | 1.6 ± 0.2 | 88 ±25 | 92.4 ± 25.4 | 4.4 ± 0.4 |
| Positive autoantibodies, N (%) | | | | | | |
| 0-1 Ab | 14 (24.6) | 9 (15.8) | 5 (8.8) | 13 (23.2) | 12 (21.4) | 1 (1.8) |
| ≥ 2 Ab | 43 (75.4) | 48 (84.2) | 5 (8.8) | 43 (76.8) | 44 (78.6) | 1 (1.8) |

*The DCC for JDRF-TEDDY Follow-Up study provided a list of 113 PIDs used for the replication of Table 1

## Attachment A: SAS Code

```
libname dsic "X:\NIDDK\niddk-dr_studies2\JDRF\private_created_data\Data";
libname refdir "X:\NIDDK\niddk-dr_studies2\JDRF\prog_initial_analysis\sasgrowth chartscdcdata";

/*********************************************************/
/*        Dataset Integrity Check for JDRF              */
/*********************************************************/

*Calling the list of IDs for particpants included in the analysis;
DATA WORK.'JDRF Mask IDs'n;
  LENGTH
    JDRF_NIDDK_maskid   8 ;
  FORMAT
    JDRF_NIDDK_maskid BEST12. ;
  INFORMAT
    JDRF_NIDDK_maskid BEST12. ;
  INFILE DATALINES4
    DLM='7F'x
    MISSOVER
    DSD ;
  INPUT
    JDRF_NIDDK_maskid : BEST32. ;
DATALINES4;
458061
496250
273106
666890
553779
746693
969448
506469
792878
635226
564148
907131
720832
760560
249813
697214
693364
497140
637630
528288
393638
480393
655154
```

349550
950249
728793
798979
410915
637815
681846
435391
421353
489001
981205
781521
888784
690601
352077
477260
897741
913469
601476
872615
563588
664928
943799
497089
925294
895603
753071
211056
342395
499500
816831
459193
366464
616906
542015
419092
949832
370589
902019
312927
850978
210160
716366
723620
995066
292825
802146
410540

```
757134
869045
812200
441523
303413
324719
430518
313946
699157
306867
579162
762496
545733
748376
357084
978593
957708
526072
595177
696172
943235
583136
678703
464440
888764
853765
438473
351705
956368
902789
275357
524537
557149
886363
283773
380999
872588
705785
782397
425758
398642
538787
;;;;

data maskids; set work.'jdrf mask ids'n;
run;

/*************************/
```

```
/*   Calling Datasets     */
/*************************/

data base_diag; set dsic.base_diagnosis_diabetes;
run;

data diag_diab; set dsic.diagnosis_diabetes;
run;

data reg; set dsic.registration_rev;
run;

data enroll; set dsic.enrollment;
run;

data phys; set dsic.physical_exam;
run;

data base_demo; set dsic.base_demographics_rev;
run;

data hba1c; set dsic.post_hba1c;
run;

data base_med; set dsic.medical_history;
run;

data specimen; set dsic.specimen;
run;

data spec_info; set dsic.specimen_info;
run;

data test_res; set dsic.test_results;
run;

data base_spec; set dsic.base_specimen;
run;

data base_spec_info; set dsic.base_specimen_information;
run;

data base_test; set dsic.base_test_results;
run;

/*************************/
/*   Female, N(%)        */
/*************************/
```

```
proc sort data=enroll;
by JDRF_NIDDK_maskid;
run;

proc sort data=reg;
by JDRF_NIDDK_maskid;
run;

proc sort data=base_demo;
by JDRF_NIDDK_maskid;
run;

proc sort data=maskids;
by JDRF_NIDDK_maskid;
run;

data sex;
merge base_demo (in=a)
        reg (in=b)
        maskids (in=c);
by jdrf_niddk_maskid;
if c = 1;
run;

data sex1;
merge sex (in=a)
        enroll (in=b)
        maskids (in=c);
by jdrf_niddk_maskid;
if c=1;
run;

proc freq data=sex;
tables sex*casecontrol/ norow nopercent;
/*where match_ind = 1;*/
run;

proc freq data=sex1;
tables sex*casecontrol / norow nopercent;
/*where match_ind = 1;*/
run;

/*********************************/
/*     FDR with T1D, N(%)      */
/*********************************/

proc sort data=base_demo;
by JDRF_NIDDK_maskid;
```

```
run;

proc sort data=enroll;
by JDRF_NIDDK_maskid;
run;

data fdr;
merge base_demo (in=a)
        reg (in=b)
        maskids (in=c);
by jdrf_niddk_maskid;
if c=1;
run;

proc freq data=fdr;
tables AnyFamilyMemT1D*CaseControl/norow nopercent;
run;

data fdr2;
merge enroll (in=a)
        reg (in=b)
        maskids (in=c);
by jdrf_niddk_maskid;
if c=1;
run;

proc freq data=fdr2;
table FamilyType1Diabetes*CaseControl/norow nopercent;
run;

/**************************/
/*Age at diagnosis in Years*/
/**************************/
data reg1; set reg;
diag_ageyrs = (t1d_diag_agedys)/365;
run;

data reg2;
merge reg1 (in=a)
        maskids (in=b);
by jdrf_niddk_maskid;
if b=1;
run;

proc means data=reg2 n mean std;
var diag_ageyrs;
class CaseControl;
/*where match_ind = 1;*/
```

```
run;

/***********************************/
/*   Diabetic ketoacidosis, N(%)    */
/***********************************/

proc contents data=base_diag;
run;

proc contents data=diag_diab;
run;

proc freq data=base_diag;
tables DKANo_AsymptomaticPhysicianrepo DKAStatus DKAStatusPresent_Bloodketones15
DKAStatusPresent_Urineketoneslar
DKAStatusPresent_Urineketonesmod DiabetesDiagnosisMade UrineKetonesResult;
where due_num <=3;
run;

proc freq data=diag_diab;
tables DKAStatus DKAStatusPresent_Bloodketones15 DKAStatusPresent_Urineketoneslar
DKAStatusPresent_Urineketonesmod
UrineKetonesResult;
where due_num <= 3;
run;

data DKA; set base_diag diag_diab;
if urineketonesresult = "Large" or DKAstatus = "DKA" OR DKAStatusPresent_Bloodketones15 = 1 OR
DKAStatusPresent_Urineketonesmod = 1 then DKA = 1 ; else DKA = 0;
run;

proc sort data=DKA;
by JDRF_NIDDK_maskid;
run;

data DKA2;
merge enroll (in=a)
        maskids (in=b)
        reg (in=c)
        DKA (in=d);
by jdrf_niddk_maskid;
if b=1;
run;

proc contents data=dka2;
run;

proc freq data=dka2;
```

```
tables DKA*CaseControl/norow nopercent;
run;

/*************************************/
/* Weight, Height, and BMI Z-score Calc*/
/*************************************/

data base_med1; set dsic.base_medical_history1;
keep JDRF_NIDDK_maskid due_num HeightCmsDynamic1_1 HeightInchesDynamic1_1
HeightInchesDynamic2_1 HeightInchesDynamic3_1
WeightKgsDynamic1_1 WeightOuncesDynamic2_1 WeightOuncesDynamic3_1
WeightOuncesDynamic4_1 WeightPoundsDynamic1_1 WeightPoundsDynamic2_1
WeightPoundsDynamic3_1 WeightPoundsDynamic4_1 medhist_agedys;
run;

data base_med2; set dsic.base_medical_history2;
keep JDRF_NIDDK_maskid due_Num HeightCmsDynamic1_1 WeightKgsDynamic1_1 medhist_agedys;
run;

proc contents data=base_med1;
run;

proc contents data=base_med2;
run;

proc contents data=phys;
run;

proc sort data=phys;
by JDRF_NIDDK_maskid due_num;
run;

proc sort data=base_med1;
by JDRF_NIDDK_maskid due_num;
run;

proc sort data=base_med2;
by JDRF_NIDDK_maskid due_Num;
run;

data phys1;
merge phys (in=a)
        maskids (in=b);
by jdrf_niddk_maskid;
if b=1;
run;

data base_med1_1;
```

```
merge base_med1 (in=a)
        maskids (in=b);
by jdrf_niddk_maskid;
run;

data base_med2_1;
merge base_med2 (in=a)
        maskids (in=b);
by jdrf_niddk_maskid;
if b=1;
run;

data one;
merge phys1 (in=a)
        base_med1_1 (in=b)
        base_med2_1 (in=c);
by jdrf_niddk_maskid due_num;
run;

data two;
merge one (in=a)
        reg (in=b keep=jdrf_niddk_maskid CaseControl t1d_diag_agedys);
by jdrf_niddk_maskid;
if a=1;
run;

proc contents data=two;
run;

proc freq data=two;
tables due_num*CaseControl;
run;

data three; set two;
if due_num = 3;
run;

data mydata1; set three;

agemos = ((phys_agedys/365)*12);
agemos1 = ((t1d_diag_agedys/365)*12);
run;

proc contents data=mydata1;
run;

proc freq data=mydata1;
tables agemos;
```

```
run;

data mydata2; set mydata1;
if agemos = . then agemos = agemos1;
run;

proc freq data=mydata2;
tables agemos;
run;

proc sort data=mydata2;
by JDRF_NIDDK_maskid;
run;

proc contents data=base_demo;
run;

data mydata3;
merge mydata2 (in=a)
        enroll (in=b keep=sex jdrf_niddk_maskid)
        base_demo (in=c keep=sex jdrf_niddk_maskid);
by jdrf_niddk_maskid;
if a=1;
run;

proc freq data=mydata3;
tables sex;
run;

data mydata4; set mydata3;
if sex = "Male" then sex1 = 1;
if sex = "Female" then sex1 = 2;
run;

proc freq data=mydata4;
tables sex;
run;

data mydata5; set mydata4;
keep agemos CaseControl LengthHeight Weight Sex JDRF_NIDDK_maskid height sex1;
height = lengthheight;
run;

proc contents data=mydata5;
run;

proc freq data=mydata5;
tables height sex1;
```

```
run;

*final dataset for CDC SAS program;
data mydata6; set mydata5;
drop LengthHeight sex;
run;

data mydata; set mydata6;
rename sex1 = sex;
run;

proc contents data=mydata;
run;

*CDC sas program;
%include "X:\NIDDK\niddk-dr_studies2\JDRF\prog_initial_analysis\sasgrowth chartscdcdata\CDC-
source-code.sas"; run;

*looking at z-scores;
proc means data=work._cdcdata;
run;

proc sort data=work._cdcdata;
by JDRF_NIDDK_maskid;
run;

data casecontrol; set mydata;
keep JDRF_NIDDK_maskid CaseControl;
run;

proc sort data=casecontrol;
by JDRF_NIDDK_maskid;
run;

data zscores;
merge _cdcdata
            casecontrol;
by jdrf_niddk_maskid;
run;

proc means data=zscores;
class CaseControl;
run;

/*********************************/
/*    HbA1c at diagnosis (%)     */
/*********************************/
```

```
proc contents data=work.base_diag;
run;

data hba1c;
merge reg (in=a)
        base_diag (in=b)
        maskids (in=c);
by jdrf_niddk_maskid;
if c=1;
run;

*cases;
proc means data=hba1c n mean std;
var HemoglobinA1cResult _2ndHemoglobinA1cResult;
class CaseControl;
where HemoglobinA1cResult ^= .;
run;

proc contents data=work.diag_diab;
run;

data hba1c_2;
merge reg (in=a)
        diag_diab (in=b)
        maskids (in=c);
by jdrf_niddk_maskid;
if c=1;
run;

proc means data=hba1c_2 n mean std;
var HemoglobinA1cResult _2ndHemoglobinA1cResult;
class CaseControl;
where HemoglobinA1cResult ^= .;
run;

*converting to mmol/mol;
data hba1c_3; set hba1c;
a1c = 10.929 * (HemoglobinA1cResult - 2.15);
run;

proc means data=hba1c_3 n mean std;
var a1c;
class CaseControl;
run;

data hba1c_4; set hba1c_2;
a1c = 10.929 * (HemoglobinA1cResult - 2.15);
run;
```

```
proc means data=hba1c_4 n mean std;
var a1c;
class casecontrol;
run;

/*****************************/
/* Positive Autoantibodies   */
/*****************************/

*These should be in the specimen and test datasets;

proc contents data=work.base_spec;
run;
proc contents data=work.base_spec_info;
run;

proc contents data=work.specimen;
run;
proc contents data=work.spec_info;
run;

proc contents data=work.test_res;
run;
proc contents data=work.base_test;
run;

proc freq data= work.test_res;
tables test_name due_num;
run;

proc freq data=work.base_test;
tables due_Num test_name;
run;

*starting with the Base_test results dataset, narrow down to the Autoantibody tests;
data basetest1; set base_test;
if test_name = "GAD" OR test_name = "IA2A" OR test_name = "ZnT8A";
run;

*also the test_results dataset;
data test1; set work.test_res;
if test_name = "GAD" OR test_name = "IA2A" OR test_name = "ZnT8A";
run;

*reducing the test1 dataset to include only baseline and month 3 visits;
data test2; set test1;
if due_num <= 3;
```

```
run;

*checking the reduction;
proc freq data=test2;
tables due_num;
run;

proc freq data=basetest1;
tables test_name;
run;

*merging basetest with ccmask;
proc sort data=basetest1;
by JDRF_NIDDK_maskid;
run;

proc sort data=ccmask;
by JDRF_NIDDK_maskid;
run;

data basetest2;
merge basetest1 (in=a)
    ccmask (in=b);
by jdrf_niddk_maskid;
if b=1;
run;

*merging test2 with CCmask;
proc sort data=test2;
by JDRF_NIDDK_maskid;
run;

data test3;
merge test2 (in=a)
          ccmask (in=b);
by jdrf_niddk_maskid;
if b=1;
run;

*merging basetest and test3;
data basetest3;
merge test3
          basetest2;
by jdrf_niddk_maskid;
run;

*looking at results to determine positive test;
proc freq data=basetest3;
```

```
tables outcome;
run;
*just need the outcome variable;

data basetest4; set basetest3;
if outcome ^= "";
if outcome ^= "Not Reported";
run;

proc freq data=basetest4;
tables outcome;
run;

*creating a summary variable to identify # of positive autoantibodies;
data basetest5; set basetest4;
if test_name = "GAD" and outcome = "Pos" then GAD = 1; else GAD = 0;
if test_name = "IA2A" and outcome = "Pos" then IA2A = 1; else IA2A = 0;
if test_name = "ZnT8A" and outcome = "Pos" then ZnT8A = 1; else ZnT8A = 0;
run;

proc freq data=basetest5;
tables gad ia2a ZnT8A;
run;

/*
proc sort data=basetest5 out=basetest6 nodupkey;
by JDRF_NIDDK_maskid gad ia2a ZnT8A due_num;
run;
*/

*eliminating duplicate AA tests and keeping the earliest test;
data GAD; set basetest5;
where test_name = "GAD";
run;

data IA2A; set basetest5;
where test_name = "IA2A";
run;

data ZnT8A; set basetest5;
where test_name = "ZnT8A";
run;

*limiting the GAD tests to the first possible test;
proc sort data=GAD;
by JDRF_NIDDK_maskid due_num;
run;
```

```
data GAD1; set GAD;
by JDRF_NIDDK_maskid;
retain N;
if first.jdrf_niddk_maskid then N=1;
else N = N+1;
if N = 1 then output;
run;

*limiting the IA2A tests to the first possible test;
proc sort data=IA2A;
by JDRF_NIDDK_maskid due_num;
run;

data IA2A1; set IA2A;
by JDRF_NIDDK_maskid;
retain N;
if first.jdrf_niddk_maskid then N=1;
else N = N+1;
if N=1 then output;
run;

*limiting the ZnT8A tests to the first possible test;
proc sort data=ZnT8A;
by JDRF_NIDDK_maskid due_num;
run;

data ZnT8A1; Set ZnT8A;
by JDRF_NIDDK_maskid;
retain N;
if first.jdrf_niddk_maskid then N=1;
else N = N+1;
if N=1 then output;
run;

*Concatinating the three seperate test datasets;
data basetest6; set GAD1 IA2A1 ZnT8A1;
run;

proc sort data=basetest6;
by JDRF_NIDDK_maskid due_num;
run;

data basetest8; set basetest6;
by JDRF_NIDDK_maskid;
if first.jdrf_niddk_maskid then
        do;
                sumAA = 0;
                cnt = 0;
```

```
            end;
sumAA + GAD + IA2A + ZnT8A;
cnt +1;
 if last.jdrf_niddk_maskid then output;
run;

proc freq data=basetest8;
tables sumAA cnt;
run;

data basetest9; set basetest8;
drop evaluate_agedys JDRF_NIDDK_SampleMaskID receive_agedys sample_agedys specimen_id;
run;

proc print data=basetest9;
run;

proc sort data=basetest9 ;
by JDRF_NIDDK_maskid;
run;

*looking at total AA by CaseControl status;
proc freq data=basetest9;
tables sumAA*CaseControl;
run;

data basetest10; set basetest9;
if sumAA <= 1 then posAA = 1;
if sumAA >= 2 then posAA = 2;
run;

proc freq data=basetest10;
tables posAA*casecontrol;
run;
```