PharmaSUG 2013 - Paper SP01

Doctoring Your Clinical Trial with Adaptive Randomization: SAS® Macros to Perform Adaptive Randomization

Jenna Colavincenzo, University of Pittsburgh

ABSTRACT

Adaptive randomization schemes have become increasingly common in beginning stages of clinical trials and in small clinical trials. This paper introduces two kinds of adaptive randomization schemes (treatment adaptive randomization and covariate adaptive randomization) and discusses the benefits and limitations of each. In addition, this paper demonstrates how to use SAS® macros to perform these adaptive randomization schemes in a clinical setting, and how these macros can be modified to fit your randomization needs.

BACKGROUND

In clinical trials, the entire sample of patients is not often available at the time of randomization. Thus, patients must be placed into treatment groups as they arrive to the trial. At the beginning stages of a clinical study, or for trials with small sample sizes ($n < 100$), using simple randomization can cause an extreme imbalance among the treatment groups by random chance alone (Kang et al 2008). This imbalance may be in the number of patients for each treatment or in the possible covariates across each treatment group. These situations may lead to inflation of the error variance and may cause researchers to disregard a potentially significant treatment difference when one exists.

These statistical imbalances can often be resolved by using an adaptive randomization scheme which uses the current balance across treatment groups to influence the randomization of the next patient that arrives to the trial. Two types of adaptive randomization schemes are discussed in this paper: treatment adaptive randomization and covariate adaptive randomization; however it should be noted that there are many other types of adaptive designs. The goal of each of these two designs is to balance the treatment groups as much as possible by changing the way the next patient is assigned to a treatment group. Step-by-step instructions for use of SAS® macros designed to perform each type of randomization are described below.

TREATMENT ADAPTIVE RANDOMIZATION

The limitations of many randomization designs are in detecting imbalance in treatment group sample sizes, and detecting the severity of that imbalance. This was a main influence in the creation and implementation of the adaptive biased coin design. The adaptive biased coin design was introduced in 1977 by Wei, and is based on Friedman's Urn Model (Chow and Chang 2006). The difference in treatment group sample sizes, D_N, is calculated preceding the arrival of the next patient. The difference is then scaled by the total sample size thus far, *n*. This is called the "difference ratio", D_N / n, and it weights the assignment probabilities for the next patient (Chow and Chang 2008).

The probability that the next patient is assigned to group B is *f*, a monotone increasing function of the ratio D_N / n. The probability of the next patient being assigned to group A is *g,* a monotone decreasing function such that $f + g = 1$. Thus, the severity of imbalance in treatment group sample sizes is taken into account. Choosing $f = N_A / n$ has been shown to yield statistically desirable designs, which makes $g = N_B / n$ where N_j is the number of subjects in group *j*.

The *ABCD* macro, which takes the current number in each treatment group as inputs, performs treatment adaptive randomization by calculating the difference ratio. It then uses the RANUNI function with a random seed to generate a number between negative one and one, and will assign the next patient to treatment group B if that random number is less than or equal to the difference ratio. Otherwise, if the random number is greater than the difference ratio, the next patient will be assigned to treatment group A. The following demonstrates this balanced randomization:

```
/* assign subject to B with probability x, A with prob 1-x */
%let RandNum = 2*%SYSFUNC(ranuni(0))-1; %*uniform[-1,1] random var;
%if %SYSEVALF(&RandNum <= &DiffRatio) %then %let Assign=B;
%else %let Assign=A;
```

Finally, the macro formats the treatment group assignment information and prints the results to the output window.

The adaptive biased coin design accounts for the severity of imbalance in treatment group sample sizes, so the more severe the imbalance in treatment groups the more certain it is that the next patient will be assigned to the smaller treatment group. The relationship between the difference ratio and the probability of assignment for the two treatment groups can be seen in Figure 1 below.

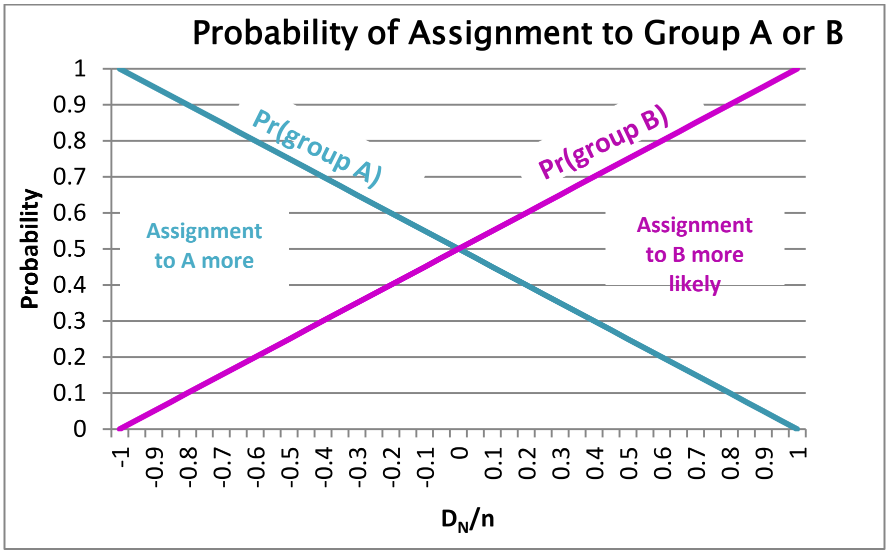

Figure 1. Assignment Probabilities for the Adaptive Biased Coin Design

Consider a case where there are currently five patients in treatment group A and 10 patients in treatment group B. The difference ratio would be -1/3 and the next patient would have a 67% chance of assignment to treatment group A. Conversely, if the groups are well balanced, this randomization scheme behaves similar to simple randomization and assigns patients to treatment groups with near equal probabilities. This design also guarantees perfectly balanced groups for the first two patient assignments. (The first patient will be completely randomized to a treatment group and the second patient will be automatically assigned to the other treatment group.) Table 1 below compares the unconditional probability of having two balanced treatment groups in using a completely randomized design (CRD) or the adaptive biased coin design (ABCD). (For odd sample sizes, two groups are in balance if the sample size of one group is within one of the other.) For the first 10 patients, the adaptive biased coin design achieves balanced treatment groups more often than complete randomization. As the number of patients increases, the adaptive biased coin design more closely mimics a complete randomization scheme.

Probability of Balance (of Two Groups) after *n* Assignments

Design					*n*				
	2	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
CRD	.500	.750	.375	.625	.313	.547	.273	.492	.246
ABCD	1	1	.667	.917	.550	.839	.479	.775	.430

Table 1. Comparison across Designs of the Probability of Equal Treatment Group Sample Sizes (Wei 1977)

The adaptive biased coin design macro has been written for two treatment groups and could be easily extended to multiple treatment groups, provided the number of patients in each group is known.

COVARIATE ADAPTIVE RANDOMIZATION

The previous design attempts to be balance the sample sizes of treatment groups. It is possible that there are other factors that the researcher wishes to keep balanced across treatment groups. For example, in a trial examining the efficacy of different blood pressure medications, it may be of interest to have nearly equal baseline blood pressures across the treatment groups. Similar to treatment adaptive randomization, covariate adaptive randomization works to balance treatment groups, while considering the covariate levels of patients within each group. It achieves this by assigning the next patient to the treatment group that will minimize the imbalance in covariate levels. The imbalance in covariate levels is measured with a *Chi-Squared goodness-of-fit* test for categorical covariates (analogously, a *two-sample t* test or *ANOVA* works for quantitative covariates) (Frane 1998). A low *Chi-Squared* statistic indicates higher balance among the covariates.

There are many methods of covariate adaptive randomization; the Frane method is used in this paper since it can control for both quantitative and qualitative covariates (Frane 1998). The steps involved in covariate adaptive randomization are highlighted in Figure 2 below.

Step 1	Temporarily assign the new patient to treatment group A.
Step 2	Calculate the Pearson's *Chi-Squared goodness-of-fit* test statistic for the covariate groups to which the new patient would belong.
Step 3	Identify the maximum *Chi-Squared* test statistic among all the covariate groups.
Step 4	Remove the patient from group A and repeat steps 1 – 3 for all other treatment groups.
Step 5	Identify the minimum *Chi-Squared* test statistic over all the identified test statistics (one from each repetition of step 1 – 3).
Step 6	Assign the new patient to the group for which the minimum *Chi-Squared* test statistic was achieved.

Figure 2. Steps of Covariate Adaptive Randomization

The two SAS® macros that together perform covariate adaptive randomization require a user-defined patient data set. This data set gives the frequency of each unique combination of covariates and group assignment thus far. The data set should be of the structure proposed below.

```
data patdat;
input cov level group $ count;
datalines;
1 1 A 4
1 2 A 8
2 1 A 5
2 2 A 7
1 1 B 5
1 2 B 3
2 1 B 2
2 2 B 6
;
```

Consider the following example: the efficacy of two blood pressure reducing drugs is being compared in patients with high blood pressure, with consideration of the covariates baseline blood pressure (hypertensive or pre-hypertensive) and age (greater than or equal to 65 years or less than 65 years). The user input data above shows that there are four patients who are pre-hypertensive taking drug A, eight patients who are hypertensive taking drug A, etc.

The first of the two macros, called *getStat*, finds the highest *Chi-Squared* statistic over all the covariates when given the treatment group and values of the two covariates. To do this, it increases the frequency for the correct value of the covariates. Then the *getStat* macro performs a *Chi-Squared goodness-of-fit* test (if requested, testing specific proportions using the TESTP= option in the TABLE statement). The ODS SELECT NONE statement, specified above the PROC FREQ, suppresses all printed output. Using ODS TRACE to locate the individual *Chi-Squared* test statistics, named *OneWayChiSq*, SAS® can store the output *Chi-Squared* test statistics into a data set called *temp* using the ODS OUPUT statement, keeping the test statistics for the correct covariate values.

```
/* run goodness-of-fit test */
proc freq data=patdat_test;
    by cov level;
    table group / chisq testp=&props;
    weight count / ZEROS;
       /* create output data set with test statistics */
     ods output OneWayChiSq = temp
       (where=(Name1="_PCHI_" and ((cov=1 & level=&level1) OR
          (cov=2 & level=&level2)) ));
run;
```

Then the largest calculated statistic is located and kept as the only observation in the data set so it is returned when the macro is called.

The second macro, *CAR*, uses the *getStat* macro for each treatment group to obtain the highest *Chi-Squared* test statistic for assignment to each treatment group. It then finds the lowest *Chi-Squared* test statistic, which corresponds to the treatment group assignment that will cause the least imbalance among the covariates.

These macros are currently used with two covariates, each with two levels, but can be adapted for more variables and/or levels easily. It can be adapted for use in quantitative covariates by changing the *Chi-Squared goodness-of-fit* test to a *t*-test from Analysis of Variance.

Consider again the example from above, which is summarized in Figure 3 below.

Distribution of First 20 Patients by Blood Pressure and Age			
		Drug A (N_A = 12)	Drug B (N_B = 8)
BASELINE BLOOD PRESSURE	Pre-Hypertensive (level 1)	4	5
	Hypertensive (level 2)	8	3
AGE	< 65 years-old (level 1)	5	2
	≥ 65 years-old (level 2)	7	6

Figure 3. Patient Characteristics for the First 20 Patients in Trial

A new patient arrives at the trial who is hypertensive and less than 65 years of age. To determine to which treatment group this new patient should be assigned, the new patient is first temporarily assigned to group A, and the *Chi-Squared goodness-of-fit* test statistics are calculated for the number of hypertensive patients across groups A and B, and for the number of under 65-year-old patients across groups A and B. This process is demonstrated in Figure 4 below.

Calculated Chi-Squared Test Statistics for New Patient				
		N (Drug A)	N (Drug B)	*Chi-Squared Test Statistic*
BASELINE BLOOD PRESSURE	Hypertensive (level 2)	9	3	3.0
AGE	< 65 years (level 1)	6	2	2.0

Figure 4. Calculating the *Chi-Squared* Test Statistics

The largest *Chi-Squared* test statistic in adding the new patient to group A is 3.0. Repeating this process, assigning the patient to B, and calculating *Chi-Squared* test statistics, the largest *Chi-Squared* test statistic when the patient is assigned to group B is 1.33. Therefore the new patient should be assigned to group B, since there will be a smaller imbalance in the covariates when the patient is in group B.

This example utilizes Frane's covariate adaptive randomization method with two treatment groups and two binary categorical covariates. However, this method can be extended for more treatment groups and more covariates and covariate categories. Also, the researcher may predetermine the proportion of allocation for the treatment groups by specifying props=, if unequal treatment group sample sizes are desired. In the example above, one could specify that there should be twice as many patients on drug A as on drug B. This could be done for numerous reasons, including ethical reasons, if one drug is potentially more effective or more practically used than another. To implement this scenario the props=(0.67, 0.33) input can be employed in the *CAR* macro.

Covariate adaptive randomization will balance the covariates across the treatment groups but there are some marked limitations of this design. Because this method determines the treatment group of the next patient, rather than increasing the probability of being assigned to that treatment group (as seen in the adaptive biased coin design), selection bias is present. Researchers may be able to determine the treatment group assignment of the next patient based on the patient characteristics alone, which will lead to a more biased method than desired. Further investigation should consider randomizing patients with increased probability rather than directly determining the group assignment of the next patient.

CONCLUSION

The adaptive biased coin design and covariate adaptive randomization are specific designs aimed at balancing certain aspects of a clinical trial. The adaptive biased coin design aims to balance treatment group sample sizes by assigning the next patient to the group with the smaller sample size with higher probability and using the total sample size to scale the difference in treatment group sample sizes. The probability of assignment to a specific group in the adaptive biased coin design thus depends on the ratio of the difference in sample sizes between treatment groups to total sample size. Covariate randomization designs aim to balance the covariates across the treatment groups by assigning the next patient to the group that causes the smallest imbalance across the covariate groups. The macros provided allow for these randomization schemes to be implemented using SAS®. The examples provided facilitate the adoption of these macros, which can be easily modified to fit a specific trial's needs.

Each adaptive randomization design provides researchers with alternatives to traditional randomization. However, the complicated probability structures involved in patient assignment to treatment groups makes estimating treatment effects more challenging than in traditional randomization designs. Often the traditional treatment effect estimates are used in these designs; however, more development is needed in methods to estimate treatment effects and variance.

REFERENCES

- Chow, Shein-Chung and Chang, Mark. 2006. Adaptive Design Methods in Clinical Trials. *Chapman and Hall/CRC Press*. New York, NY. Taylor and Francis.
- Frane, J.W. 1998. A Method of Biased Coin Randomization, Its Implementation, and Its Validation.*Drug Information Journal* 32:423-432.
- Chow, Shein-Chung, and Chang, Mark. 2008. Adaptive Design Methods in Clinical Trials – A Review. Orphanet Journal of Rare Diseases 3(11).
- Frane, J.W. 1998. A Method of Biased Coin Randomization, Its Implementation, and Its Validation.*Drug Information Journal* 32:423-432.
- Kang, Minsoo, Ragan, Brian G., and Park, Jae-Hyeon. 2008. Issues in Outcomes Research: An Overview of Randomization Techniques for Clinical Trials. *Journal of Athletic Training* 43(2):215-221.
- Wei, L. J. 1977. A Class of Designs for Sequential Clinical Trials. *Journal of the American Statistical Association* 72(358):382-386.

ACKNOWLEDGMENTS

The author wishes to thank the following: Dr. Karen McGaughey, for her perseverance and inspiration in advising this project, Rebecca Ottesen, for helping with the framework for the macros and providing advice on the paper, and The Cal Poly statistics department for its support, education, and continued inspiration.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Jenna Colavincenzo
Enterprise: University of Pittsburgh
Department of Biostatistics
Address: 310 Parran Hall
130 DeSoto Street
City, State ZIP: Pittsburgh, PA 15217
E-mail: jenna.colavincenzo@gmail.com
Web: http://www.biostat.pitt.edu/
http://www.linkedin.com/pub/jenna-colavincenzo/37/1b3/583

APPENDIX A – ADAPTIVE BIASED COIN DESIGN MACRO

```
/***************************************************************/
/* f(x)=1-x <- probability of assignment to A */
/* g(x)=x <- probability of assignment to B */
/* where x = (diff in num of subjects in each trt)/(total num of subjects)*/

%macro ABCDrandomization(NumA, NumB);
       data Assignment;
         %let numer=%EVAL(&NumA-&NumB);
         %let denom=%EVAL(&NumA+&NumB);
         %let DiffRatio = %SYSEVALF(&numer/&denom);

         /* assign subject to B with probability x, A with prob 1-x */
         %let RandNum = 2*%SYSFUNC(ranuni(0))-1; %*uniform[-1,1] random var;
         %if %SYSEVALF(&RandNum <= &DiffRatio) %then %let Assign=B;
         %else %let Assign=A;

         /* create subject num var */
         %let AssignNum = %EVAL(&NumA+&NumB+1);

         groupA = &NumA;
         groupB = &NumB;
         newGroup = "&Assign";

         label groupA = "Number in treatment group A"
                    groupB = "Number in treatment group B"
                    newGroup = "Assignment for next subject is";
       run;

       proc print data=Assignment label noobs; run;

%mend ABCDrandomization;

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

APPENDIX B – COVARIATE ADAPTIVE RANDOMIZATION MACROS

```
/***************************************************************/
/* This is the data set for the current subjects in the trial */
/* Data must be in this form */
data patdat;
input cov level group $ count;
datalines;
1 1 A 4
1 2 A 8
2 1 A 5
2 2 A 7
1 1 B 5
1 2 B 3
2 1 B 2
2 2 B 6
;
run;
/***************************************************************/

/***************************************************************/
%macro getStat(TrtGrp=A,level1=2,level2=1,props=(.5,.5));
/* increase frequencies to include new patient */
  data patdat_test;
    set patdat;
       if Group = "&TrtGrp" then do;
      if cov=1 and level=&level1 then count = count + 1;
         if cov=2 and level=&level2 then count = count + 1;
       end;
  run;

/* sort data by cov and level for proc freq */
  proc sort data=patdat_test;
    by cov level;
  run;

/* run goodness-of-fit test */
  ods select none; /* turns off printing from procedures */
  proc freq data=patdat_test;
      by cov level;
      table group / chisq testp=&props;
      weight count / ZEROS;
         /* create output data set with test statistics */
         ods output OneWayChiSq = temp
          (where=(Name1="_PCHI_" AND ((cov=1 & level=&level1) OR (cov=2 &
level=&level2)) ));
  run;

/* find highest Chi-Squared statistic */
  proc sort data=temp out=patdat_test_&TrtGrp (keep=cov level nvalue1
rename=(nvalue1=ChiSquare));
    by descending nvalue1;
  run;
%mend getStat;
/***************************************************************/

/***************************************************************/
%macro CAR(cov1_val=2,cov2_val=1,test_p=(.5,.5));
/* get highest Chi-Squared statistics when we assign */
/*   new subject to group A and group B */
  %getStat(TrtGrp=A,level1=&cov1_val,level2=&cov2_val,props=&test_p);
  %getStat(TrtGrp=B,level1=&cov1_val,level2=&cov2_val,props=&test_p);
```

```
/* combine output from macro calls above */
  data all;
   set patdat_test_A (in=A) patdat_test_B (in=B);
   GrpA = A;
   GrpB = B;
  run;

/* find lowest Chi-Squared for the two treatment groups */
  proc sort data=all;
   by ChiSquare;
  run;

/* print out the group assignment associated with the lowest Chi-Squared statistic*/
  data all;
   set all;
   if _N_ = 1;
   if GrpA = 1 then TrtGrp="A";
   else if GrpB = 1 then TrtGrp="B";
   cov1 = &cov1_val;
   cov2 = &cov2_val;
   label cov1 = "Next subject*has covariate 1*value"
             cov2 = "Next subject*has covariate 2*value"
             TrtGrp = "Assignment*for next*subject is";
   keep cov1 cov2 TrtGrp;
  run;

  ods select all; /* turn printing back on*/
  proc print split="*" noobs; run;

%mend CAR;
/***************************************************************/
```