



Comparison of Type I Error of some Non-Parametric Tests on Multiple Regression Models Coefficients

Ali Shadrokh ¹

Statistics Department, University of Payame Noor, 19395-4697, I.R of IRAN

Abstract

Various non-parametric methods have been used to perform hypothesis test on multiple regression coefficients. In this article, at first the most important methods which has been introduced from other statisticians as proper methods, such as Kennedy, Freedman and Lane, and modified Kennedy, are explained and then, Freedman and Lane (Huh-John) method will be modified in the form of Kennedy method; finally, all aforementioned methods will be compared as simulating. At last, we look for a method that done best. So, Huh-John (2001) modify the method of Kennedy which was proposed in 1995 and showed by simulation that is called modified Huh-John method; and it has less type I error. On the other hand, Anderson as simulation (1991) and Schadrekh as theory (2011) had shown that Freedman& Lane method has lower type I error in comparison with Kennedy method. We did some modification on Freedman and Lane method that Huh-John had done on Kennedy method and compared this modified method with Freedman and Lane and Huh-John method. We conclude that Freedman and Lane modified method often has lower type I error estimation and higher power than Freedman& Lane and Huh-John method.

Keywords: Permutation tests; type I permutation error; changeability; multiple regression coefficient.



Council for Innovative Research

Peer Review Research Publishing System

Journal: JOURNAL OF ADVANCES IN MATHEMATICS

Vol.11, No.7

www.cirjam.com , editorjam@gmail.com



1. INTRODUCTION

When fundamental hypotheses to perform hypothesis test are not true on linear models regression coefficients, doing classic tests on regression coefficients will not give a reliable answer, so we should use non-parametric methods. One group of these tests is re-sampling tests including Jackknife, Bootstrap, and permutation tests.

Since most of the time, fundamental hypotheses are not true on real data, so permutation tests attain significant place. There is an agreement among statisticians about linear simple regression for hypothesis test, but different methods have been proposed for multiple regression. Important methods that are based on Permutability principle include: Freedman and Lane (1983), Kennedy (1995), Manly (1991), Ter Braak (1992). In 1999, Anderson and Legendre investigated four above methods by doing extensive simulation and showed that Freedman and Lane method and Kennedy method perform better than other. In 2010, Schaderch theoretically showed that Freedman and Lane method can perform better than Kennedy's. Huh and John modified Kennedy method in 2001 and showed that Kennedy modified method has better performance than Kennedy. Schaderch in 2010 also showed that Freedman and Lane method is better than Huh and John. Now, our goal is to modify the method of Freedman & Lane as the way that Huh and John has modified Kennedy method, as well as compare it by Freedman and Lane and Kennedy method as simulation.

2. Permutation test methods

In this section, we neglect the presence of missing data and we investigate only the modes have one dependant variable, and expand its results to multi-variable modes. The method that is identified appropriate for this mode will be proper to permutation methods with models having several dependant models. Here, for the ease of problem, we survey linear multiple regression model with two independent variables. We consider the following model for studying with two variables,

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon \quad (1)$$

We use the following statistic for testing the hypothese,

$$\begin{cases} H_0 : \beta_2 = 0 \\ H_1 : \beta_2 \neq 0 \end{cases} \quad (2)$$

where $S_{(b_2)}$ is the standard deviation of b_2 .

2.1 Permutation of remaining under reduced model

Freedman and Lane utilized a model in which the value of remaining are used as permutable units.

Assume a model in which $H_0 : \beta_2 = 0$ is true, so we can write model (1) under the title of "reduced model" as below:

$$y = \beta_0 + \beta_1 x_1 + \varepsilon' \quad (3)$$

It is evident that β_1 and ε' used in reduced model are different from β_1 and ε in complete model. There are two different methods for the use of remaining permutation in regression coefficient estimation. First method was proposed in 1983 by Freedman & Lane and the second one was in 1995 by Kennedy; we explain both here:

1) Proposed model by Freedman and Lane:

Make Y regression on X_1 and X_2 , and estimate the value of b_2 for β_2 coefficient using minimum squares method.

And by use of it, we obtain statistic t value for primary data. We call this statistic t_{Ref}

$$t_{ref} = \frac{b_2}{S_{(b_2)}}$$

- 1) Make Y regression on X_1 variable as model (3), and obtain the estimation of b_0 for β_0 and b_1 for β_1 ; then we calculate \hat{Y} :
- 2) Obtain the remaining as follow:

$$R_{Y|x_1} = Y - \hat{Y} = Y - (b_0 + b_1 x_1) \quad (4)$$



Now we permute these remaining, and show them by $R_{Y|x_1}^*$.

- 1) Add Permuted remaining to \hat{Y} in order to obtain new values of Y^* . A considerable note here is, Y^* are not permuted values of Y , but indicate the Y that are obtained from permutation of remaining.

$$Y^* = \hat{Y} + R_{Y|x_1}^* \tag{5}$$

- 2) Make the Y^* obtained from step 4 regression as the model below on both variables of X_1 and X_2 .

$$Y^* = \beta_0^* + \beta_1^* x_1 + \beta_2^* x_2 + \varepsilon \tag{6}$$

Finally obtain an estimation b_2^* for β_2^* and t^* for permutation data in the form of $t_{ref}^* = \frac{b_2^*}{S_{(b_2^*)}}$ with $(n-2)$ freedom degree

by using it.

- 1) Repeat steps 3 to 5 for $n!$ Possible permutation in order to obtain value distribution of t^* under permutations.
- 2) Calculate absolute value of Ref as the method that was explained at the first of this chapter, and decide to accept or omit zero hypotheses.

II) Proposed method by Kennedy:

The steps of Kennedy method are as follow:

- 1) First 3 steps are done like Freedman and Lane method.
- 2) Calculate regression equation of X_2 variable on X_1 as the model below

$$x_2 = \tau_0 + \tau_1 x_1 + \varepsilon'' \tag{7}$$

and obtain remaining of $R_{x_2|x_1}$ from estimation of X_2 .

- 3) Make $R_{Y|x_1}^*$ vector regression on variable of $R_{x_2|x_1}$ under the following model

$$R_{Y|x_1}^* = \beta_0 + \beta_1 R_{Y|x_1} + \varepsilon' \tag{8}$$

and calculate b_1 estimation for β_1 , and obtain t^* values from it by different permutation.

- 4) Repeat Steps 3 until 5 to obtain t^* value under permutation. (note that $R_{x_2|x_1}$ obtained in Step 4 should be fixed.)
- 5) P-value is obtained like methods described.

3. Freedman and Lane method in multiple mode:

Consider the following regression model:

$$Y = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i1} + \dots + \beta_k x_{ik} + \varepsilon_i \tag{9}$$

Where errors are came from an independent random sample uni-distributed by zero mean and σ^2 variance. Explain individual test of regression coefficients. A regression coefficient which is considered for test is β_k , and the goal of performing this test is Freedman and Lane method.

$$\begin{cases} H_0: \beta_k = 0 \\ H_1: \beta_k \neq 0 \end{cases} \tag{10}$$

Rewrite equation (8) as matrix form:

$$Y = X_1 \bar{\beta}_1 + \bar{X}_k \beta_k + \bar{\varepsilon} \quad \bar{\varepsilon} \sim F(0, \sigma^2 I_n) \tag{11}$$

Where error terms have unspecified distribution F as above:



$$X_k = \begin{pmatrix} x_{1k} \\ \vdots \\ x_{nk} \end{pmatrix} \quad B = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_{k-1} \end{pmatrix} \quad e = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_k \end{pmatrix} \quad Y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad X_1 = \begin{pmatrix} 1 & x_{11} & \dots & x_{1(k-1)} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & \dots & x_{n(k-1)} \end{pmatrix}$$

Indeed, separate the coefficient is considered for test from others, and assumes other coefficients as a vector. Now, we perform Freedman and Lane method as follow:

Step 1) We first obtain $\hat{\beta}_k$ value and also t_{Ref} statistic value from main data (non-permuted) of equation (11).

Step 2) Multiply both sides of the equation (11) by the following matrix:

$$M_{X_1} = I_n - H_1 = I_n - X_1(X_1^t X_1)^{-1} X_1^t \tag{12}$$

The following relation is achieved from above operation:

$$\tilde{Y} = \tilde{X}_k \beta_k + \tilde{\varepsilon} \tag{13}$$

Where

$$\tilde{X}_k = (I_n - H_1) X_k = M_{X_1} X_k, \quad \tilde{\varepsilon} = (I_n - H_1) \varepsilon = M_{X_1} \varepsilon \tag{14}$$

$$\tilde{Y} = (I_n - H_1) Y = M_{X_1} Y$$

We know that $M_{X_1} = I - H_{X_1}$ matrix is a symmetric and idempotent matrix. Furthermore, the matrix has the following feature:

$$M_{X_1} X_1 = 0 \tag{15}$$

By the use of minimum squares method, we estimate β_k coefficient as below:

$$\tilde{\beta}_k = \frac{\tilde{X}_k^t \tilde{Y}}{\tilde{X}_k^t \tilde{X}_k} \tag{16}$$

By simplifying this term and regarding the symmetry and idempotency of M_{X_1} , we have:

$$\tilde{\beta}_k = \frac{(M_{X_1} X_k)^t (M_{X_1} Y)}{(M_{X_1} X_k)^t (M_{X_1} X_k)} = \frac{X_k^t M_{X_1}^t M_{X_1} Y}{X_k^t M_{X_1}^t M_{X_1} X_k} = \frac{X_k^t M_{X_1} Y}{X_k^t M_{X_1} X_k} \tag{17}$$

Using $\tilde{\beta}_k$ estimation, \tilde{Y} estimation is calculated and is called $\tilde{\tilde{Y}}$.

Step 3) Calculate the remaining of subtracting $\tilde{\tilde{Y}}$ from \tilde{Y} and call it $R_{(Y|X_1)}$. Then, we permute these remaining and sowed them by $R^*_{(Y|X_1)}$.

Step 4) Add permuted remaining to $\tilde{\tilde{Y}}$ in order to obtain new values of \tilde{Y}^* . Then, we make them regress on X_1 and X_k , regression equation is in the form of below:

$$\tilde{Y}^* = \beta_1^* X_1 + \beta_k^* X_k + \tilde{\varepsilon}^* \tag{18}$$

We obtain estimator β_k^* by permuted data \tilde{Y}^* :

$$\tilde{\beta}_k^* = \frac{\tilde{X}_k^t \tilde{Y}^*}{\tilde{X}_k^t \tilde{X}_k} \tag{19}$$

and by using it we obtain $t^* = \tilde{\beta}_k^* / s(\tilde{\beta}_k^*)$.

Step 5) we repeat steps 3 and 4 and by using values of t^* s we obtain the distribution of permutation of t s and then p-value.

3.1 Criticism of the Freedman and Lane method



In Friedman and Lane's approach from the reduced regression equation based on residuals about the null hypothesis, we decided that we need to know the distribution of $\bar{\varepsilon}$. This is a point that $\bar{\varepsilon}$ in model (11) has an equal distribution with $\bar{\varepsilon}^*$ but the distribution of $\bar{\varepsilon}$ in model (14) with the distribution of $\bar{\varepsilon}^*$ is different.

3.2 Distribution of Errors and Compare them:

In multiple Friedman and Lane's approach mean and variance of the error is as follows:

$$E(\bar{\varepsilon})=E((I_n-H_1)\bar{\varepsilon})=(I_n-H_1)E(\bar{\varepsilon})=0$$

$$V(\bar{\varepsilon})=Var((I_n-H_1)\bar{\varepsilon})=(I_n-H_1)Var(\bar{\varepsilon})(I_n-H_1)^t=\sigma^2(I_n-H_1)(I_n-H_1)^t=\sigma^2(I_n-H_1) \quad (20)$$

$$= \sigma^2(I_n-H_1)$$

$$\bar{\varepsilon} \sim F(0, \sigma^2(I_n - H_1))$$

Also in equation (14) we express the mean and variance of the error, now if we consider P as a permutation matrix, we have:

$$\bar{\varepsilon}^* = p\bar{\varepsilon} \quad E(\bar{\varepsilon}^*) = E(p\bar{\varepsilon}) = pE(\bar{\varepsilon}) = 0 \quad (21)$$

$$V(\bar{\varepsilon}^*) = V(p\bar{\varepsilon}) = pV(\bar{\varepsilon})p^t = p\sigma^2(I_n-H_1)p^t = \sigma^2(I_n-H_1)$$

Therefore $\bar{\varepsilon}$ and $\bar{\varepsilon}^*$ are identically distribution. For comparing the distribution of $\bar{\varepsilon}$ and $\bar{\varepsilon}^*$ by using the permutation matrix we have:

$$\bar{\varepsilon}^* = p\bar{\varepsilon} \quad E(\bar{\varepsilon}^*) = E(p\bar{\varepsilon}) = pE(\bar{\varepsilon}) = 0 \quad (22)$$

$$V(\bar{\varepsilon}^*) = V(p\bar{\varepsilon}) = pV(\bar{\varepsilon})p^t = p(\sigma^2(I_n - H_1))p^t = \sigma^2(p(I_n - H_1)p^t) = \sigma^2(I_n - H_1^*)$$

Considering the above term, we find that the variance of $\bar{\varepsilon}^*$ is affiliated H_1^* so that by any permutation of H_1 , σ^2 would be multiply the new number and the variance would be change. Mean and variance $\bar{\varepsilon}$ with $\bar{\varepsilon}^*$ are not the same; as a result, their distribution is not equal. In view of the hypothesis $\beta_k=0$ the distribution of \tilde{y}^* will be different from the distribution of \tilde{y} . This is due to Friedman and Lane's approach that is under doubt.

3.3 Modified Freedman and Lane method:

Error expressed in Friedman and Lane method was first in the Kennedy method that Huh and John gave a way to resolve this problem; similarly we used this method to improve Friedman and Lane method. Considering that the matrix (I_n-H_1) is an idempotent matrix with rank $n^* = n-k$, (where k is the number of independent variables in the model.) For Correcting Friedman and Lane method the following steps will apply:

- Step 1) Calculate eigenvectors of matrix (I_n-H_1) and its rank.
- Step 2) Consider eigenvectors corresponding to the eigenvalues of one that their number with rank (I_n-H_1) is equivalent, as a matrix.
- Step 3) Using Gram-E.schmidt method the above matrix is converted to a orthogonal matrix.
- Step 4) Dividing each column by norm of that column it goes to a unit vector and resulting matrix will be represented by W_1 that is an $(n \times n)$ orthogonal matrix so that has the following features:

$$W_1^t W_1 = I_n \quad (23)$$

Step 5) Since the eigenvalue of matrix W_1 is one and by using theory of Spectral Decomposition presented in index, matrix (I_n-H_1) is obtained

$$(I_n-H_1) = W_1 W_1^t = W_1 W_1^t \quad (24)$$

Step 6) Product W_1^t in $\tilde{Y} = \tilde{X}_k \beta_k + \bar{\varepsilon}$ equation:

$$\tilde{Y}_{n \times 1} = \tilde{X}_{k \times 1} \beta_k + \bar{\varepsilon}_{n \times 1} \quad (25)$$



Where $\tilde{Y} = W^t_1 \tilde{Y}$ $\tilde{X}_k = W^t_1 \tilde{X}_k$ $\tilde{\varepsilon} = W^t_1 \tilde{\varepsilon}$

For mean and variance $\tilde{\varepsilon}$ we have

$$\begin{aligned}
 E(\tilde{\varepsilon}) &= E(W^t_1 \tilde{\varepsilon}) = W^t_1 E(\tilde{\varepsilon}) = 0 \\
 V(\tilde{\varepsilon}) &= V(W^t_1 \tilde{\varepsilon}) = W^t_1 V(\tilde{\varepsilon}) W_1 = W^t_1 (\sigma^2 (I_n - H_1)) W_1 = \\
 &W^t_1 (\sigma^2 (W_1 W^t_1)) W_1 = \sigma^2 W^t_1 (W_1 W^t_1) W_1 = \sigma^2 I_n I_n = \sigma^2 I_n \\
 \tilde{\varepsilon} &\sim (0, \sigma^2 I_n)
 \end{aligned}
 \tag{26}$$

To consider the distribution of permuted vector $\tilde{\varepsilon}^{*}$ namely $\tilde{\varepsilon}^{*}$, we use permutater matrix ρ , therefore

$$\begin{aligned}
 \tilde{\varepsilon}^{*} &= \rho \tilde{\varepsilon} \quad E(\tilde{\varepsilon}^{*}) = E(\rho \tilde{\varepsilon}) = \rho E(\tilde{\varepsilon}) = 0 \\
 V(\tilde{\varepsilon}^{*}) &= V(\rho \tilde{\varepsilon}) = \rho V(\tilde{\varepsilon}) \rho^t = \rho (\sigma^2 I_n) \rho^t = \sigma^2 I_n \rho \rho^t = \sigma^2 I_n I_n = \sigma^2 I_n \\
 \tilde{\varepsilon}^{*} &\sim (0, \sigma^2 I_n)
 \end{aligned}
 \tag{27}$$

So, $\tilde{\varepsilon}$ and $\tilde{\varepsilon}^{*}$ are identically distribution, in fact with this method distribution of the error terms after the permutation also are fixed, so by using new regression equation in the Freedman and Lane algorithm, this problem will disappear.

By using Sum Least Squares Error we obtain $\tilde{\beta}_k$:

$$\tilde{\beta}_k = (\tilde{x}_k^t \tilde{x}_k)^{-1} \tilde{x}_k^t \tilde{Y}
 \tag{28}$$

Now we explain how we obtained the above estimation

$$\tilde{x}_k^t \tilde{Y} = (W^t_1 \tilde{x}_k)^t (W^t_1 \tilde{Y}) = \tilde{x}_k^t W_1 W^t_1 \tilde{Y} = \tilde{x}_k^t (I_n - H_1) \tilde{Y} = \tilde{x}_k^t I_n \tilde{Y} - \tilde{x}_k^t H_1 \tilde{Y} = \tilde{x}_k^t \tilde{Y} - \tilde{x}_k^t H_1 \tilde{Y}$$

$$\text{Where } H_1 \tilde{Y} = H_1 (I_n - H_1) Y = (H_1 I_n - H_1^2) Y = H_1 I_n Y - H_1^2 Y$$

Since H_1 is self-power we have

$$H_1 \tilde{Y} = H_1 Y - H_1 Y = 0 \quad \tilde{x}_k^t \tilde{Y} = \tilde{x}_k^t \tilde{Y} - 0 = \tilde{x}_k^t \tilde{Y}
 \tag{29}$$

$$\tilde{x}_k^t \tilde{x}_k = (W^t_1 \tilde{x}_k)^t (W^t_1 \tilde{x}_k) = \tilde{x}_k^t W_1 W^t_1 \tilde{x}_k = \tilde{x}_k^t (I_n - H_1) \tilde{x}_k = \tilde{x}_k^t I_n \tilde{x}_k - \tilde{x}_k^t H_1 \tilde{x}_k$$

And also

$$H_1 \tilde{x}_k = H_1 (I_n - H_1) x_k = H_1 I_n x_k - H_1 H_1 x_k = H_1 x_k - H_1 x_k = 0$$

$$\tilde{x}_k^t \tilde{x}_k = \tilde{x}_k^t x_k - 0 = \tilde{x}_k^t x_k$$

Therefore

$$\tilde{\beta}_k = \frac{\tilde{x}_k^t \tilde{Y}}{\tilde{x}_k^t \tilde{x}_k} = \frac{\tilde{x}_k^t \tilde{Y}}{\tilde{x}_k^t x_k} = \tilde{\beta}_k
 \tag{30}$$

$$\tilde{Y}^* = \tilde{x}_k^t \beta_K^* + \tilde{\varepsilon}^*
 \tag{31}$$

So estimate β_K^* by least squares method is as follow

$$\tilde{\beta}_k^* = \frac{\tilde{x}_k^t \tilde{Y}^*}{\tilde{x}_k^t \tilde{x}_k}
 \tag{32}$$

Where \tilde{Y}^* is a $(n \times 1)$ vector that has obtained by random permutation \tilde{Y} . Because of testing of null hypothesis we obtained all $\tilde{\beta}_k^*$ s and t^* and then p-value, and decision whether hypothesis should be rejected or not.

4. Simulation

In this section we want to compare three methods: modified Kennedy (Huh and John), Freedman and Lane and modified Freedman and Lane. We investigate the result of simulation by using figures and programs exist in index written by S-PLUS software.

4.1. Simulation way:

For this purpose, we compute the estimation of type I error for all three permutation methods (Huh and John, Freedman and Lane, and modified Freedman and Lane) in terms of following 4 factors:

- 1- Sample size (n)
- 2- Correlation between two variables of X_1 and X_2 which is demonstrated by ρ
- 3- β_k regression coefficient value which is not tested.
- 4- Error variable distribution

To perform stimulation, we consider the values of random variable of X_1 and X_2 from three uniform distributions (-3, 3), exponential distribution with parameter 2, and Gama distribution with parameters1 and 2 so that one of them is symmetric and two others are skew to the right. Y vector is also generated through $Y=X\bar{\beta}+\bar{\epsilon}$ where X is a matrix that its columns are X_1 and X_2 , and $\bar{\beta}$ vector consist of the values of $\{\beta_1$ and $\beta_2\}$. Since the goal of the test is the hypothesis of $H_0: \beta_2=0$, and regarding to the fact that, type I error is possible to deny zero hypothesis when it is true, thus we must assume zero hypothesis of $H_0: \beta_2=0$ true, then β_1 will adopt selective values of $\{0, 0.5, 1, 1.5, 2, 2.5\}$.

The values of error term are also generated by three distribution of standard normal distribution, exponential distribution with parameter 2 (skew to the right), and uniform distribution (-3, 3). The correlation between two variables of X_1 and X_2 is obtained by using Choleski decomposition of correlation matrix $R_{2 \times 2}$, whose values on main diagonal are 1 and values on minor diagonal are ρ , which are here $\{0, 0.1$ and $0.9\}$. Therefore,

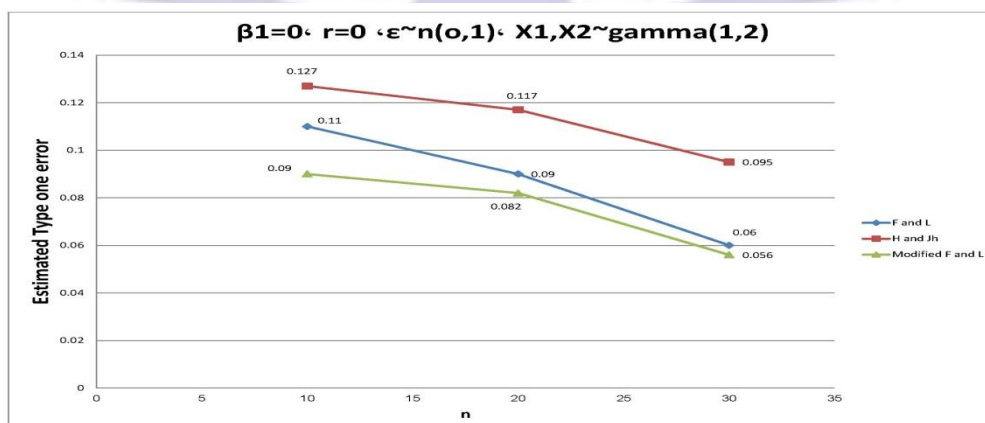
$$X_R = X \times R_C \tag{33}$$

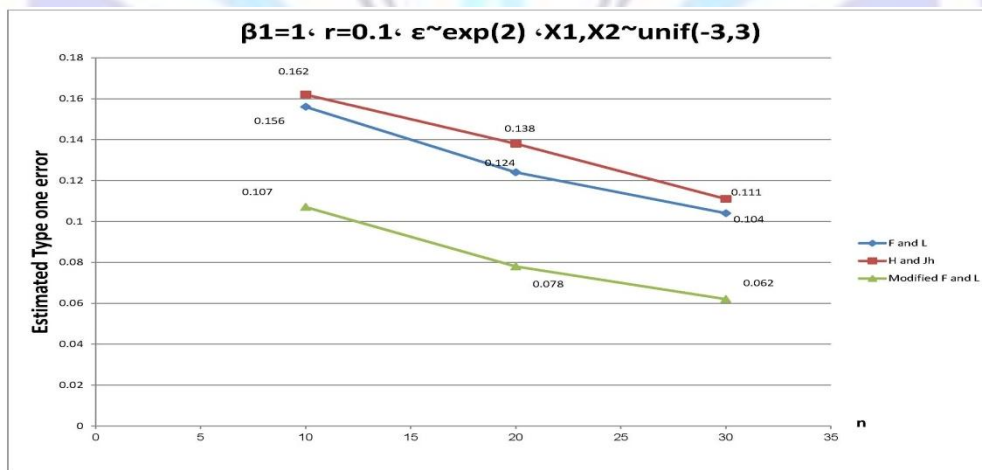
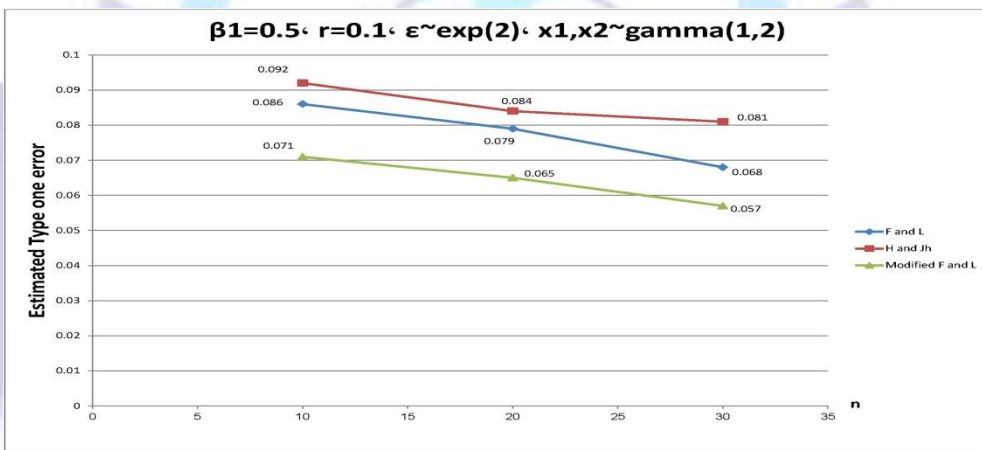
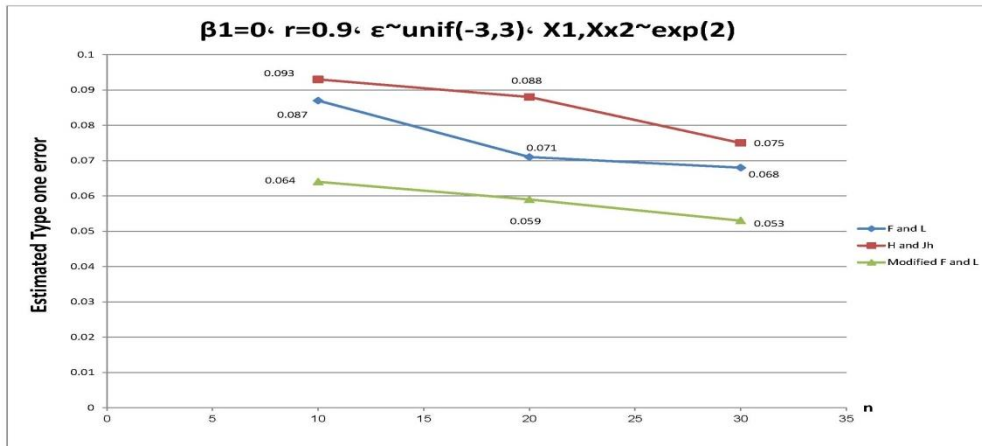
Where R_C is matrix that is achieved by Choleski decomposition of R, X_R is also a matrix whose columns are consisted of X_{1R} and X_{2R} vectors, which are correlated to each other by R and consistent with correlation coefficient of ρ . finally, Y vector is regenerated by using this matrix based on $Y=X_R\bar{\beta}+\bar{\epsilon}$ model.

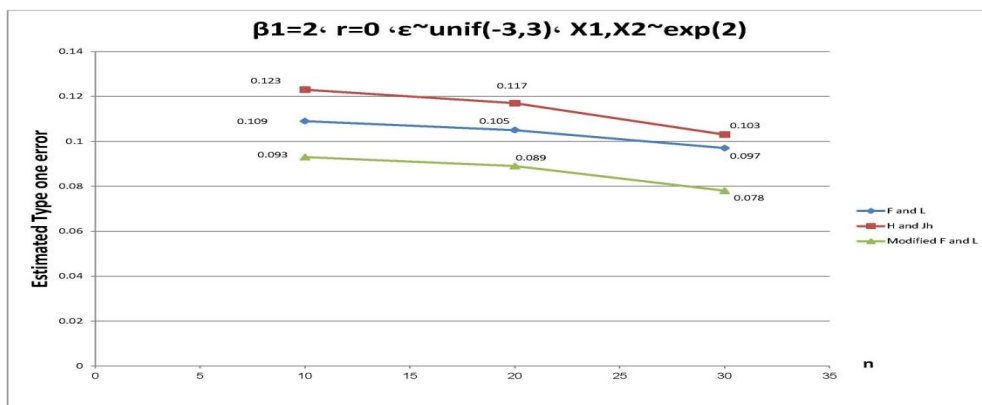
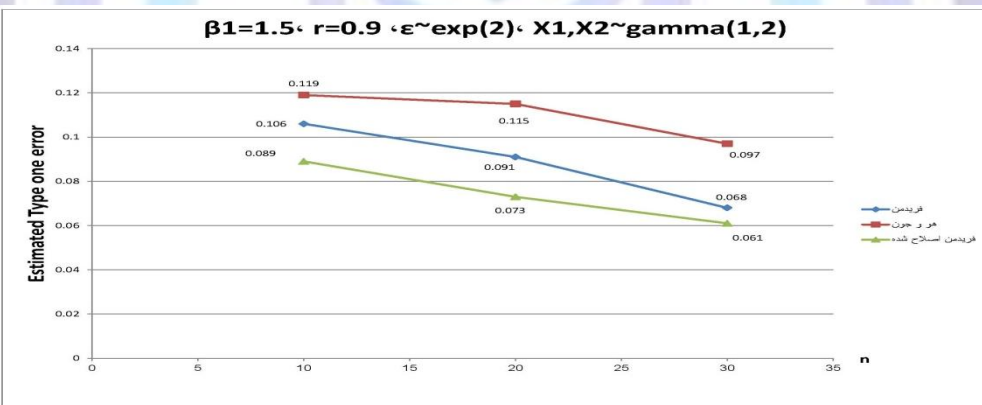
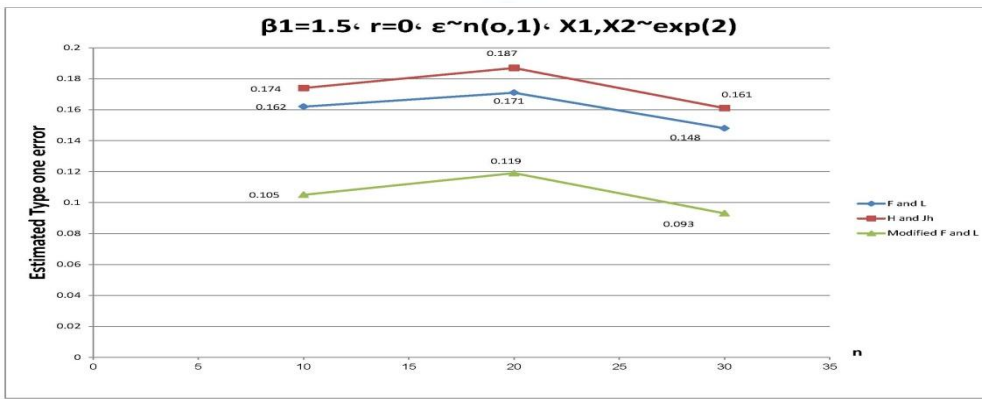
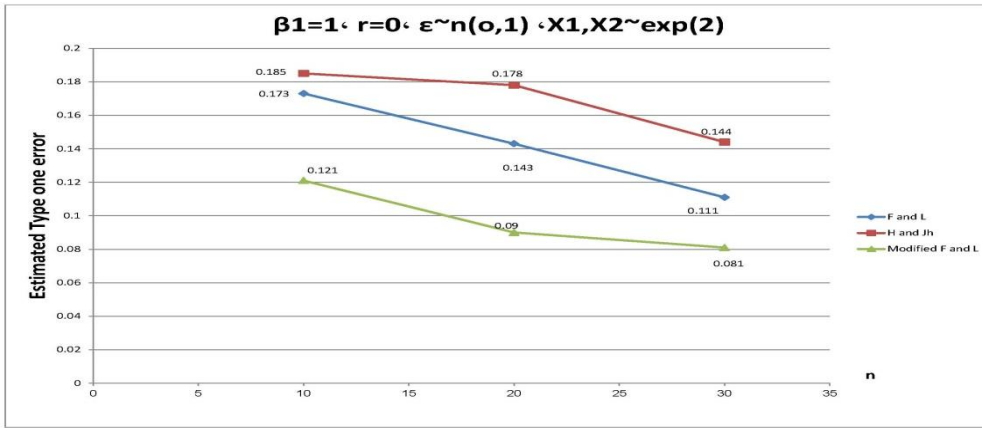
The estimation of type1 errors at significance level of 0.05 is calculated by 1000 replication for 3 methods namely, Freedman and Lane, Huh and John, and modified Freedman and Lane. Calculation method of type I error for every type and every combination that is called experimental or permutation error is such that, we obtain the proportion of times that p- values are less than significance level of $\alpha=0.05$, and consider it as type I error estimation.

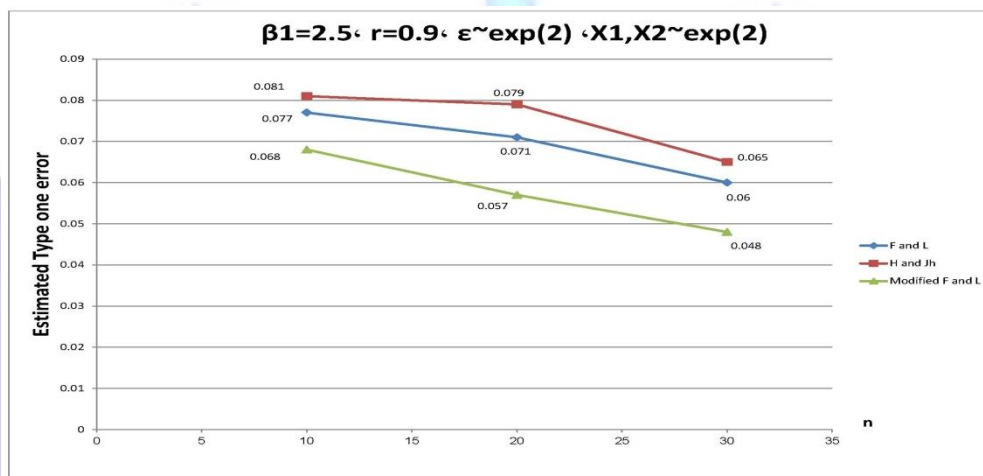
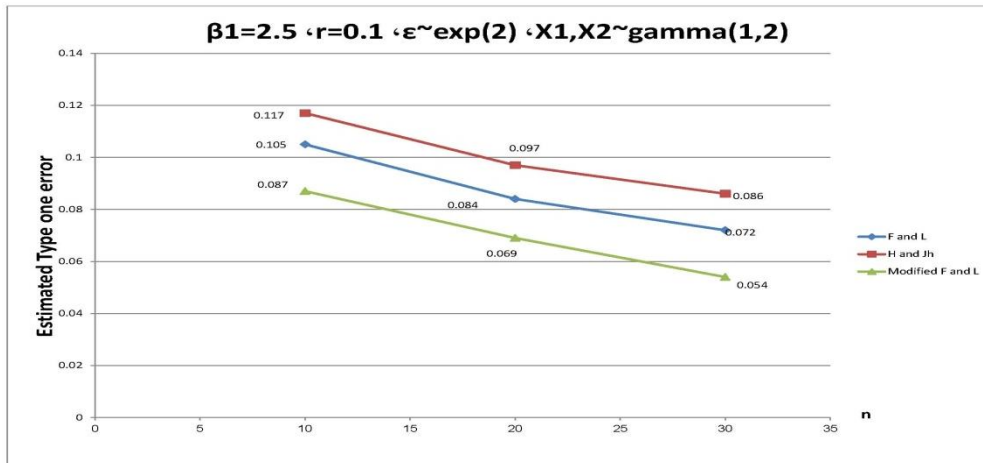
At the end, we compare the power of all three methods considering values of $\{0.5, 1,$ and $1.5\}$ for parameter β_2 . We prepare and simulate these programs in S-PLUS. Some of figures from simulation are in the below. It is sufficient to compare type I error and power of three methods in the following examples.

Fig. 2.1



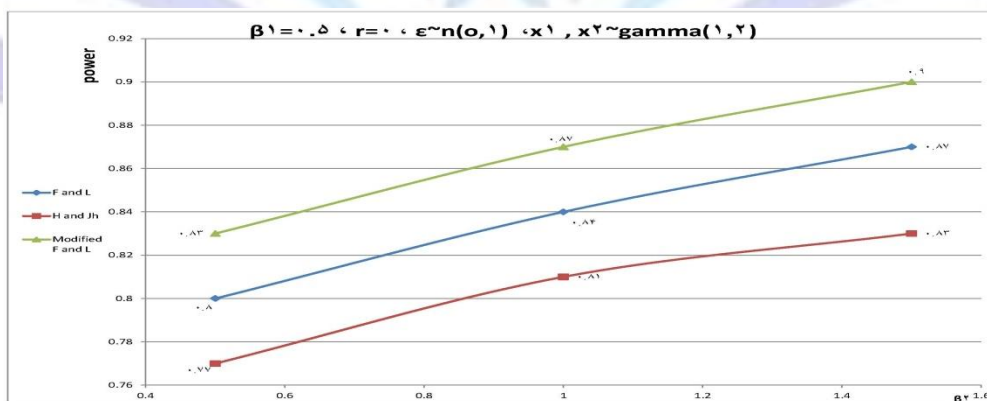


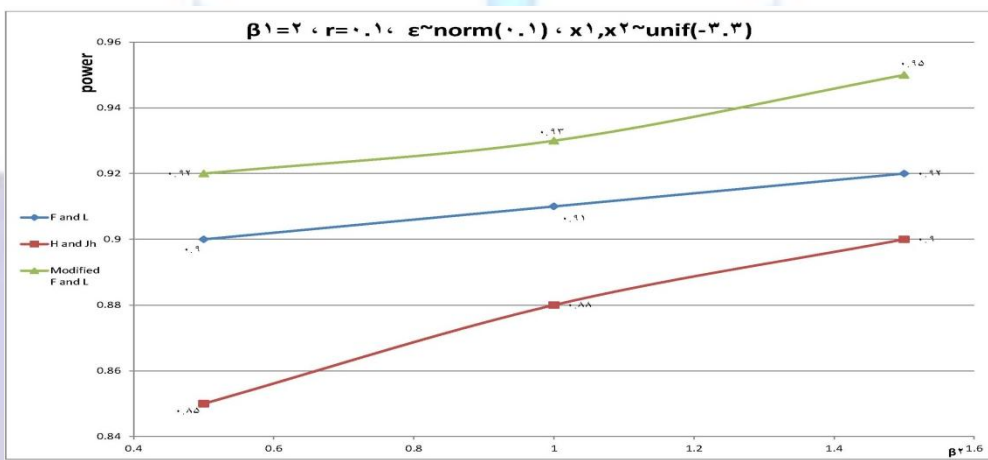
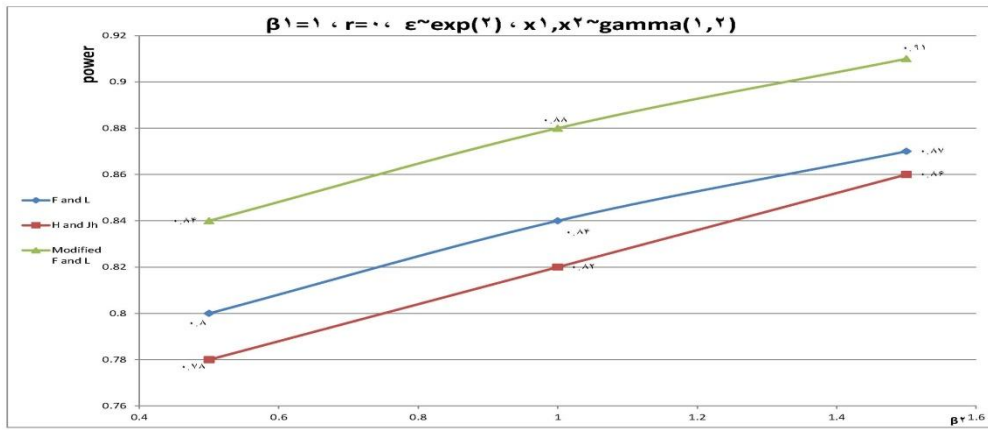




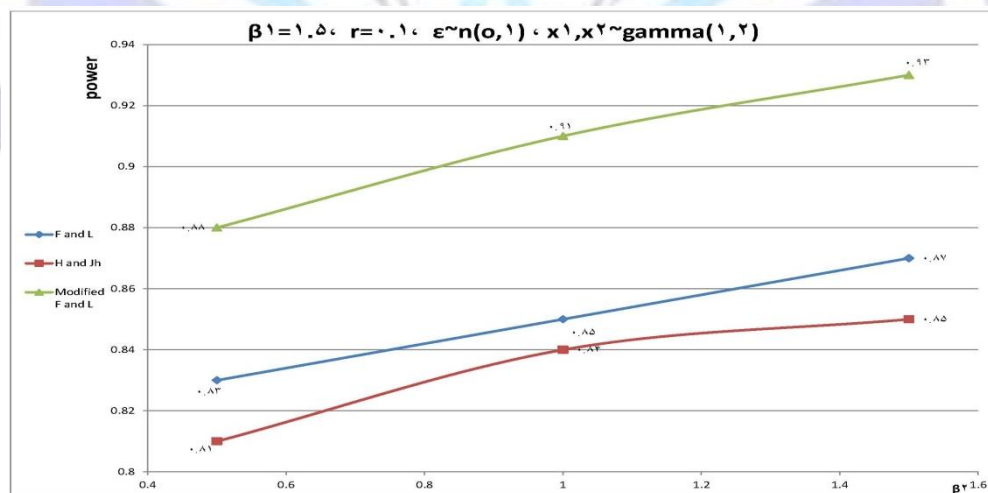
For investigating power of three mentioned methods we present some figures for $n=20$ and 30 :

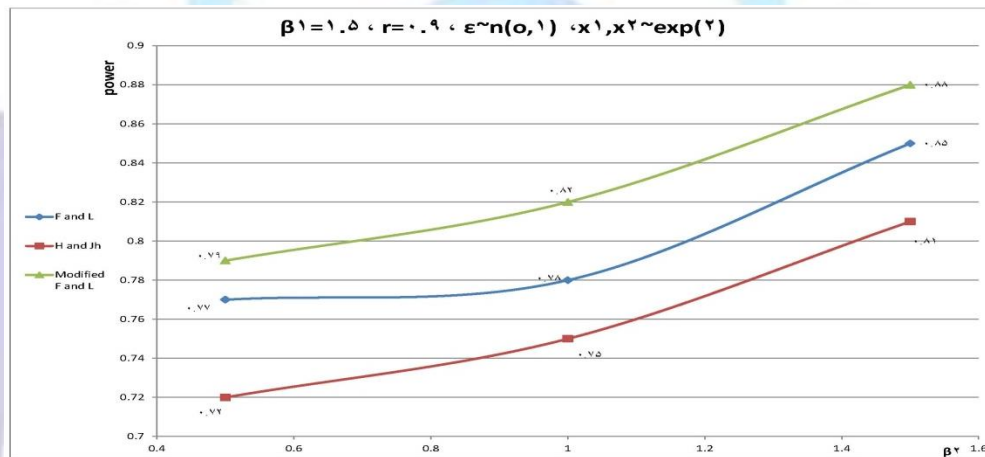
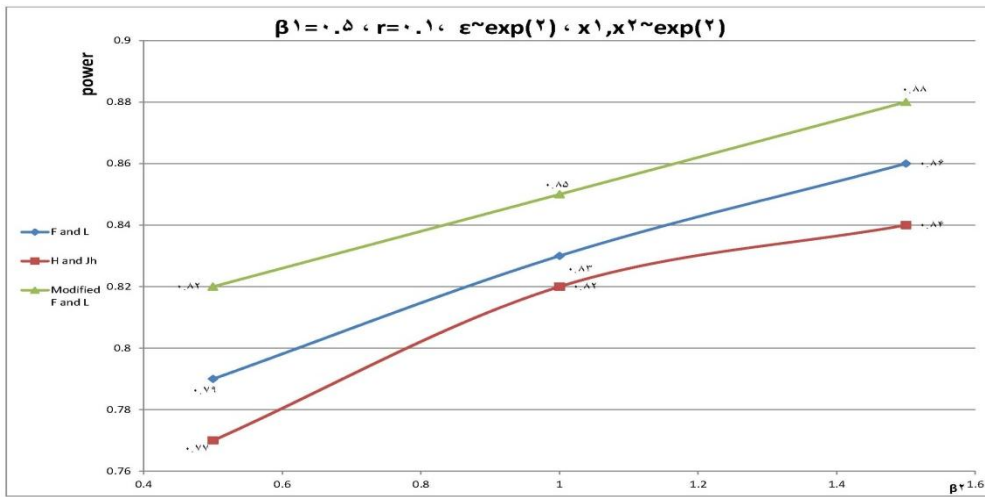
I. $n=20$





II. n=30





In total, every method that its type I error is close to $\alpha=0/05$ identified have been done better. Also, by using simulation we showed under reduced models, modified Freedman and Lane due to lower type I error and higher power is first choice and Freedman and Lane and Huh and John will be our next choice.

REFERENCES

- 1 Anderson, M.J., Legendre, P., 1999. An empirical comparison of permutation methods for tests of partial regression coefficients in a linear model, *J. Stats. Comput. Simul.*, 62, 271-303.
- 2 Hoe, M., and Jhun, M. 2001. Random permutation testing in multiple linear regression, *Commun. Statist.-Theory Meth.* 30, 10, 2023-2032.
- 3 B. Efron, 1979. Bootstrap Methods: Another Look at the Jackknife, *Annals of Statistics*, 7, 1-26.
- 4 Wolter, K.M., 1985. *Introduction to Variance Estimation*, Springer-Verma, New York.
- 5 Shadrokh,A. and d' Aubigny .G, 2010. An analytic comparison of permutation methods for test of partial regression coefficients in the linear model. *Applied mathematical sciences*, Vol. 4, no. 17-20, 857-878.



INDEX (Simulation Programs)

These three algorithms for a bivariate regression model with the hypothesis test $\beta_2=0$ have been regulated and they have the capacity that for different values of sample size and correlation coefficient (r) can change. To determine these values we marked them by (*).

A. Kennedy's Algorithm

```
alpha_matrix(nrow=1000,ncol=1)
for(i in 1:1000){
  x1_runif( $n^*$ ,-3,3)
  x1_matrix(c(x1),ncol=1)
  x2_runif( $n^*$ ,-3,3)
  x2_matrix(c(x2),ncol=1)
  x_matrix(c(x1,x2),ncol=2)
  beta_matrix(c( $\beta^*$ *,0))
  e_rexp( $n^*$ ,2)
  e_matrix(c(e),ncol=1)
  r_matrix(c(1, $r^*$ , $r^*$ *,1),ncol=2)
  sr_chol(r)
  xr_x%*%sr
  y_(xr%*%beta)+e
  reg_lm(y~x1+x2)
  h_summary(reg)$coefficient
  bref_h[3,1]
  tref_h[3,3]
  reg2_lm(y~ x1)
  x0_matrix(c(rep(1, $n^* - 1$ ),ncol=1))
  x_matrix(c(x0,x1),ncol=2)
  xt_t(x)
  z_xt%*%x
  zv_solve(z)
  e_x%*%zv
  h_e%*%xt
  i_diag( $n^*$ )
  mx_i-h
  k_matrix(nrow= $n^* - 1$ ,ncol=1)
  for(i in 1: 1000) {
    mxy_mx%*%y
    mxys_sample(mxy)
    mxx2_mx%*%x2
    reg3_lm(mxys~mxx2)
    s_summary(reg2)$coefficient
```




```
t_s[2,3]
k[i]_t
k
}
kplus_abs(k)
k1_kplus[kplus>=tref]
k1_length(k1)
pvalue_k1/1000
pvalue
}
alpha_alpha[pvalue<0.05]
halpha_length(alpha)/1000
halpha
```

B. Huh and John's Algorithm

```
alpha_matrix(nrow=1000,ncol=1)
```

```
for(i in 1:1000){
x1_runif( $n^*$ , -3,3)
x1_matrix(c(x1),ncol=1)
x2_runif( $n^*$ , -3,3)
x2_matrix(c(x2),ncol=1)
x_matrix(c(x1,x2),ncol=2)
beta_matrix(c( $\beta^*$ ,0))
e_rexp(n,2)
e_matrix(c(e),ncol=1)
r_matrix(c( $r^*$ ,  $r^*$ ),1,ncol=2)
sr_chol(r)
xr_x%*%sr
y_(xr%*% $\beta$ )+e
reg_lm(y~x1+x2)
h_summary(reg)$coefficient
bref_h[3,1]
tref_h[3,3]
reg2_lm(y~ x1)
x0_matrix(c(rep(1,  $n^*$  -1)),ncol=1)
x_matrix(c(x0,x1),ncol=2)
xt_t(x)
z_xt%*%x
zv_solve(z)
e_x%*%zv
h_e%*%xt
```



```
i_diag( $n^*$ )
mx_i-h
m_matrix(c(mx),nrow= $n^*$ )
d_eigen(m)$vectors
k_qr(d)$rank
d_d[,1:k]
td_t(d)
v_matrix(nrow= $n^*$ ,ncol=k)
v[,1:k]_d[,1:k]
  for(i in 2:k){
    s_matrix(0,  $n^*$ , 1)
    for(j in 1:(k-1)){
      s_s+(t(v[,j])%*%v[,j])/(t(v[,j])%*%v[,j])*v[,j]
    }
    v[,k]_d[,k]-s
  }
vn_matrix(nrow= $n^*$ ,ncol=k)
  for(i in 1:k){
    vn[,i]_v[,i]/sqrt(t(v[,i])%*%v[,i])
  }
tvn_t(vn)
mxy_mx%*%y
ytilde_tvn%*%mxy
mxx2_mx%*%x2
xtilde_tvn%*%mxx2
k_matrix(nrow= $n^*$ -1,ncol=1)
for(i in 1: 1000){
  ytildes_sample(ytilde)
  reg3_lm(ytildes~xtilde)
  s_summary(reg3)$coefficient
  t_s[2,3]
  k[i]_t
  k}
kplus_abs(k)
k1_kplus[kplus>=tref]
k2_length(k1)
pvalue_k2/1000
pvalue}
```



```
alpha_alpha[pvalue<0.05]
halpha_length(alpha)/1000
halpha
```

C. Freedman and Lane's Algorithm

```
alpha_matrix(nrow=1000,ncol=1)
for(i in 1:1000){
  x1_runif( $n^*$ ,-3,3)
  x1_matrix(c(x1),ncol=1)
  x2_runif( $n^*$ ,-3,3)
  x2_matrix(c(x2),ncol=1)
  x_matrix(c(x1,x2),ncol=2)
  beta_matrix(c( $Beta^*$ ),0))
  e_rexp( $n^*$ ,2)
  e_matrix(c(e),ncol=1)
  r_matrix(c(1, $r^*$ , $r^*$ ),1,ncol=2)
  sr_chol(r)
  xr_x%*%sr
  y_(xr%*%beta)+e
  reg_lm(y~x1+x2)
  h_summary(reg)$coefficient
  bref_h[3,1]
  tref_h[3,3]
  reg2_lm(y~x1)
  x0_matrix(c(rep(1, $n^* - 1$ )),ncol=1)
  x_matrix(c(x0,x1),ncol=2)
  h1_summary(reg2)$coefficients
  b_h1[1,]
  yhat_x%*%b
  k_matrix(nrow=, $n^* - 1$ ,ncol=1)
  for(i in 2: , 1000){
    R_y-yhat
    Rs_sample(R)
    ys_yhat+Rs
    reg3_lm(ys~x1+x2)
    h2_summary(reg3)$coefficients
    t_h2[2,3]
    k[i]_t
    k
  }
  kplus_abs(k)
  k1_kplus[kplus>=tref]
  k1i_length(k)
  pvaluei_k1i/1000
  pvaluei
}
```



```
alpha_alpha[pvaluei<0.05]
halpha_length(alpha)/1000
halpha
```

D. Modified Freedman and Lane's Algorithm

```
alpha_matrix(nrow=1000,ncol=1)
for(i in 1:1000){
  x1_runif( $n^*$ ,-3,3)
  x1_matrix(c(x1),ncol=1)
  x2_runif( $n^*$ ,-3,3)
  x2_matrix(c(x2),ncol=1)
  x_matrix(c(x1,x2),ncol=2)
  beta_matrix(c( $Beta^*$ ),0))
  e_rnorm( $n^*$ ,0,1)
  e_matrix(c(e),ncol=1)
  r_matrix(c(1,  $r^*$ ,  $r^*$ ),1),ncol=2)
  sr_chol(r)
  xr_x%*%sr
  y_(xr%*%beta)+e
  reg_lm(y~x1+x2)
  h_summary(reg)$coefficient
  bref_h[3,1]
  tref_h[3,3]
  x0_matrix(c(rep(1,  $n^*$  - 1)),ncol=1)
  x_matrix(c(x0,x1),ncol=2)
  xt_t(x)
  z_xt%*%x
  zv_solve(z)
  e_x%*%zv
  h_e%*%xt
  i_diag( $n^*$ )
  mx_i-h
  m_matrix(c(mx),nrow= $n^*$ )
  d_eigen(m)$vector
  k_qr(d)$rank
  d_d[,1:k]
  td_t(d)
  v_matrix(nrow= $n^*$ ,ncol=k)
  v[,1:k]_d[,1:k]
  for(i in 1:k){
    s_matrix(0,  $n^*$ ,1)
    for(j in 1:(k-1)){
      s_s+(t(v[,j])%*%v[,j])/(t(v[,j])%*%v[,j])*v[,j]
    }
    v[,k]_d[,k]-s
  }
}
```



```
vn_matrix(nrow= $n^*$ ,ncol=k)
for(i in 1:k){
vn[,i]_v[,i]/sqrt(t(v[,i])%*%v[,i])
vn
}
tvn_t(vn)
mxy_mx%*%y
ytilde_tvn%*%mxy
mxx2_mx%*%x2
xtilde2_tvn%*%mxx2
mxx1_mx%*%x1
xtilde1_tvn%*%mxx1
reg2_lm(ytilde~mxx1)
h1_summary(reg2)$coefficient
b_h1[,1]
yhat_x%*%b
k_matrix(nrow=1000,ncol=1)
for(i in 1:1000){
R_ytilde-yhat
Rtilde_sample(R)
ys_yhat+Rtilde
reg3_lm(ys~xtilde1+xtilde2)
h2_summary(reg3)$coefficients
t_h2[3,3]
k[i]_t
k
}
kplus_abs(k)
k1_kplus[kplus>=tref]
k2_length(k1)
pvalue_k2/1000
pvalue
}
alpha_alpha[pvalue<0.05]
halpha_length(alpha)/1000
halpha
```