LAMPIRAN

Lampiran 1

Program S-Plus

Mencari Invers Matriks:

```
> x1<-c(8,1,1,1)
> x2<-c(1,8,-2,1)
> x3<-c(1,-2,8,1)
> x4<-c(1,1,1,8)
> A<-matrix(c(x1,x2,x3,x4),ncol=4)
> A
     [,1] [,2] [,3] [,4]
[1,]    8    1    1    1
[2,]    1    8   -2    1
[3,]    1   -2    8    1
[4,]    1    1    1    8

> solve(A)
            [,1]  [,2]  [,3]        [,4]
[1,]  0.13142857 -0.02 -0.02 -0.01142857
[2,] -0.02000000  0.14  0.04 -0.02000000
[3,] -0.02000000  0.04  0.14 -0.02000000
[4,] -0.01142857 -0.02 -0.02  0.13142857

> x<-c(17.42437,10.04146,11.78176,14.18427)
> q<-matrix(c(x),ncol=1)
> q
         [,1]
[1,] 17.42437
[2,] 10.04146
[3,] 11.78176
[4,] 14.18427
```

Perkalian invers matriks dengan kolom vektor

```
> solve(A)%*%q
         [,1]
[1,] 1.691490
[2,] 1.244902
[3,] 1.418932
[4,] 1.228618
```

Lampiran 2

Program SAS untuk RAK dengan dua data hilang

```
data rak;
input perlk kelp Y;
card;
1 1 1.39642
2 1 1.35277
3 1 1.21244
2 2 1.43527
3 2 1.16189
1 3 1.67630
2 3 1.43875
3 3 1.28062
1 4 1.62481
3 4 1.37477
1 5 1.33041
2 5 1.42829
3 5 1.35277
;
proc anova data=rak;
class perlk kelp;
model Y=perlk kelp;
run;
```

Out put program SAS

Analysis of Variance Procedure
Class Level Information

Class	Levels	Values
PERLK	3	1 2 3
KELP	5	1 2 3 4 5

Number of observations in data set = 13

HASIL ANALISIS RANCANGAN ACAK KELOMPOK LENGKAP

Analysis of Variance Procedure

Dependent Variable: Y

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	6	0.19482561	0.03247093	3.87	0.0620
Error	6	0.05029032	0.00838172		
Corrected Total	12	0.24511593			

R-Square	C.V.	Root MSE	Y Mean
0.794830	6.588093	0.09155173	1.38965462

Source	DF	Anova SS	Mean Square	F Value	Pr > F
PERLK	2	0.12141398	0.06070699	7.24	0.0251
KELP	4	0.07341163	0.01835291	2.19	0.1868

Lampiran 3

Program SAS untuk RAK dengan dua data hilang yang sudah diestimasi

```
data rak;
input perlk kelp Y;
card;
1 1 1.39642
2 1 1.35277
3 1 1.21244
1 2 1.43072
2 2 1.43527
3 2 1.16189
1 3 1.67630
2 3 1.43875
3 3 1.28062
1 4 1.62481
2 4 1.55836
3 4 1.37477
1 5 1.33041
2 5 1.42829
3 5 1.35277
;
proc anova data=rak;
class perlk kelp;
model Y=perlk kelp;
run;
```

Out put program SAS

```
                    Analysis of Variance Procedure
                       Class Level Information

                     Class     Levels    Values

                     PERLK          3    1 2 3

                     KELP           5    1 2 3 4 5

                Number of observations in data set = 15
```

HASIL ANALISIS RANCANGAN ACAK KELOMPOK LENGKAP

Analysis of Variance Procedure

Dependent Variable: Y

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	6	0.21394847	0.03565808	4.89	0.0218
Error	8	0.05838174	0.00729772		
Corrected Total	14	0.27233022			

R-Square	C.V.	Root MSE	Y Mean
0.785621	6.086085	0.08542668	1.40363933

Source	DF	Anova SS	Mean Square	F Value	Pr > F
PERLK	2	0.12725017	0.06362509	8.72	0.0098
KELP	4	0.08669830	0.02167457	2.97	0.0889

Lampiran 4

Program SAS untuk RAL (untuk menghitung JKB/P dari RAK dengan dua data hilang)

```
data ral;
input kelp Y @@;
card;
1 1.39642 1 1.35277 1 1.21244
2 1.43527 2 1.16189
3 1.67630 3 1.43875 3 1.28062
4 1.62481 4 1.37477
5 1.33041 5 1.42829 5 1.35277
;
proc anova data=ral;
class kelp;
model Y=kelp;
run;
```

Out put program SAS

```
                    Analysis of Variance Procedure
                       Class Level Information

                    Class     Levels     Values

                    KELP         5       1 2 3 4 5

              Number of observations in data set = 13
```

HASIL ANALISIS RANCANGAN ACAK KELOMPOK LENGKAP

Analysis of Variance Procedure

Dependent Variable: Y

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	4	0.07341163	0.01835291	0.86	0.5293
Error	8	0.17170430	0.02146304		
Corrected Total	12	0.24511593			

R-Square	C.V.	Root MSE	Y Mean
0.299498	10.54238	0.14650269	1.38965462

Source	DF	Anova SS	Mean Square	F Value	Pr > F
KELP	4	0.07341	0.01835291	0.86	0.5293

Lampiran 5

Program SAS untuk RAK dengan empat data hilang

```
data rak;
input perlk kelp Y;
card;
1 1 1.39642
3 1 1.21244
1 2 1.65529
2 2 1.43527
3 2 1.16189
1 3 1.67630
3 3 1.28062
1 4 1.62481
2 4 1.31529
2 5 1.42829
3 5 1.35277
;
proc anova data=rak;
class perlk kelp;
model Y=perlk kelp;
run;
```

Out put program SAS

Analysis of Variance Procedure
Class Level Information

Class	Levels	Values
PERLK	3	1 2 3
KELP	5	1 2 3 4 5

Number of observations in data set = 11

HASIL ANALISIS RANCANGAN ACAK KELOMPOK LENGKAP

Analysis of Variance Procedure

Dependent Variable: Y

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	6	0.26748941	0.04458157	4.42	0.0861
Error	4	0.04038876	0.01009719		
Corrected Total	10	0.30787817			

R-Square	C.V.	Root MSE	Y Mean
0.868816	7.113101	0.10048477	1.41267182

Source	DF	Anova SS	Mean Square	F Value	Pr > F
PERLK	2	0.22776617	0.11388309	11.28	0.0227
KELP	4	0.03972324	0.00993081	0.98	0.5062

Lampiran 6

Program SAS untuk RAK dengan empat data hilang yang sudah diestimasi

```
data rak;
input perlk kelp Y;
card;
1 1 1.39642
2 1 1.24490
3 1 1.21244
1 2 1.65529
2 2 1.43527
3 2 1.16189
1 3 1.67630
2 3 1.41893
3 3 1.28062
1 4 1.62481
2 4 1.31529
3 4 1.22862
1 5 1.69149
2 5 1.42829
3 5 1.35277
;
proc anova data=rak;
class perlk kelp;
model Y=perlk kelp;
run;
```

Out put program SAS

```
                         Analysis of Variance Procedure
                            Class Level Information

                          Class      Levels    Values

                          PERLK           3    1 2 3

                          KELP            5    1 2 3 4 5

                     Number of observations in data set = 15

HASIL ANALISIS RANCANGAN ACAK KELOMPOK LENGKAP

                         Analysis of Variance Procedure

Dependent Variable: Y

Source            DF     Sum of Squares       Mean Square    F Value    Pr > F

Model              6       0.41394664        0.06899111       16.51     0.0004

Error              8       0.03343571        0.00417946

Corrected Total   14       0.44738235

                  R-Square          C.V.          Root MSE            Y Mean

                  0.925264      4.590808        0.06464877        1.40822200

Source            DF         Anova SS       Mean Square    F Value    Pr > F

PERLK              2       0.33868789        0.16934395      40.52    0.0001

KELP               4       0.07525875        0.01881469       4.50    0.0338
```

Lampiran 7

Program SAS untuk RAL (untuk menghitung JKB/P dari RAK dengan empat data hilang)

```
data ral;
input kelp Y @@;
card;
1 1.39642 1 1.21244
2 1.65529 2 1.43527 2 1.16189
3 1.67630 3 1.28062
4 1.62481 4 1.31529
5 1.42829 5 1.35277
;
proc anova data=ral;
class kelp;
model Y=kelp;
run;
```

Out put program SAS

```
                    Analysis of Variance Procedure
                       Class Level Information

                     Class     Levels    Values

                     KELP         5      1 2 3 4 5

                Number of observations in data set = 11

HASIL ANALISIS RANCANGAN ACAK KELOMPOK LENGKAP

                    Analysis of Variance Procedure

Dependent Variable: Y

Source            DF      Sum of Squares     Mean Square    F Value    Pr > F

Model              4        0.03972324        0.00993081      0.22     0.9165

Error              6        0.26815493        0.04469249

Corrected Total   10        0.30787817

                  R-Square        C.V.        Root MSE          Y Mean

                  0.129023      14.96497     0.21140598      1.41267182

Source            DF        Anova SS         Mean Square    F Value    Pr > F

KELP               4      0.03972324        0.00993081      0.22     0.9165
```

Lampiran 8

Program Pascal untuk dua data hilang

```
PROGRAM YATES_APPROXIMATION;
USES CRT;
VAR    A,B,N0 : BYTE;
       BL : CHAR;
       I,ITER,J,K,L,M,N : INTEGER;
         TOTAL,Y : REAL;
       DATA,DT,TI,BI,C,X : ARRAY [1..10,1..10] OF REAL;
       BE,NB,NC,SUMB,SUMC,TE : ARRAY [1..10] OF REAL;
BEGIN
       CLRSCR;
       WRITE('NUMBER OF GROUP : ');READLN(A);
       WRITE('NUMBER OF TREATMENT : ');READLN(B);

         {ENTERING DATA EXPERIMENT PROCESS}
       CLRSCR; N0 := 0;
       FOR I := 1 TO A DO
       BEGIN
             SUMB[I] := 0; NB[I] := B;
             FOR J := 1 TO B DO
             BEGIN
                   WRITE('THE EXPERIMENT DATA FOR GROUP ',I,'
                   TREATMENT ',J,' : ');
                   READLN(DATA[I,J]);
                   SUMB[I] := SUMB[I] + DATA[I,J];
                   IF DATA[I,J] = 0 THEN
                   BEGIN
                         NB[I] := NB[I] - 1;
                         N0 := N0 + 1;
                   END;
                   CLRSCR;
             END;
       END;

       FOR I := 1 TO B DO
       BEGIN
             SUMC[I] := 0; NC[I] := A;
             FOR J := 1 TO A DO
             BEGIN
                   SUMC[I] := SUMC[I] + DATA[J,I];
                   IF DATA[J,I] = 0 THEN NC[I] := NC[I] - 1;
             END;
       END;

       TOTAL := 0;
       FOR I := 1 TO B DO TOTAL := TOTAL + SUMC[I];

       CLRSCR;
       WRITELN('APPROXIMATION WILL BE PROCESSED');
       WRITE('NOW, ENTER THE NUMBER OF ITERATION : ');READLN(ITER);
       WRITELN;WRITELN;

       K := 0;
       FOR I := 1 TO A DO
```

```
        FOR J := 1 TO B DO
        BEGIN
              IF DATA[I,J] = 0 THEN
              BEGIN
                    K := K + 1; TE[K] := SUMC[J]; BE[K] := SUMB[I];
                    DT[K,1] := DATA[I,J];
                    WRITE('   X',J,I,'      ');
                    TI[K,1] := SUMC[J] / NC[J]; BI[K,1] := SUMB[I] /
                    NB[I];
                    C[K,1] := ( TI[K,1] + BI[K,1] ) / N0;
              END;
        END;WRITELN;

        L := 1;
        REPEAT
        Y := C[L,1];
          L := L + 1;
        UNTIL DT[L,1] = 0;

        J := 2;
        REPEAT
              I := K;
              REPEAT
                    X[I,J] := ( B*TE[I] + A*BE[I] - TOTAL - Y ) /
                    ( ( A - 1 )*( B - 1 ) );
                    Y := X[I,J];
                    M := J; N := J;
                    REPEAT
                          N := N - 1;
                          IF X[I,M] = X[I,N] THEN
                          BEGIN
                                Y := X[I,M];
                                N := 1;
                          END;
                    UNTIL N = 1;
                    I := I - 1;
              UNTIL I = 0;
              J := J + 1;
        UNTIL J = ITER + 1;

        FOR J := 2 TO ITER DO
        BEGIN
        FOR I := 1 TO K DO
              WRITE(X[I,J]:7:5,'      ');
        WRITELN;
          END;
          WRITELN;WRITELN;
          WRITELN('BECAUSE OF THE LAST ITERATION HAS A SAME VALUE');
          WRITELN('WITH THE PREVIOUS ITERATION THEN THE PROCESS IS
          FINISHED');
        WRITELN;WRITELN;
        WRITE('PRESS ANY KEY TO CONTINUE .......');READ(BL);
        IF BL = '' THEN CLRSCR;


          READLN;
  END.
```

Lampiran 9

Output program pascal untuk metode Yates dengan cara aproksimasi

```
C:\TPW\BUNGA.EXE
APPROXIMATION WILL BE PROCESSED
NOW, ENTER THE NUMBER OF ITERATION : 2


   X12         X24
1.43028     1.56186


BECAUSE OF THE LAST ITERATION HAS A SAME VALUE
WITH THE PREVIOUS ITERATION THEN THE PROCESS IS FINISHED


PRESS ANY KEY TO CONTINUE .......
```

```
C:\TPW\BUNGA.EXE
APPROXIMATION WILL BE PROCESSED
NOW, ENTER THE NUMBER OF ITERATION : 3


   X12         X24
1.43028     1.56186
1.43071     1.55842


BECAUSE OF THE LAST ITERATION HAS A SAME VALUE
WITH THE PREVIOUS ITERATION THEN THE PROCESS IS FINISHED


PRESS ANY KEY TO CONTINUE ......._
```

```
C:\TPW\BUNGA.EXE
APPROXIMATION WILL BE PROCESSED
NOW, ENTER THE NUMBER OF ITERATION : 4


   X12        X24
1.43028     1.56186
1.43071     1.55842
1.43072     1.55836


BECAUSE OF THE LAST ITERATION HAS A SAME VALUE
WITH THE PREVIOUS ITERATION THEN THE PROCESS IS FINISHED


PRESS ANY KEY TO CONTINUE .......
```

```
C:\TPW\BUNGA.EXE
APPROXIMATION WILL BE PROCESSED
NOW, ENTER THE NUMBER OF ITERATION : 5


   X12        X24
1.43028     1.56186
1.43071     1.55842
1.43072     1.55836
1.43072     1.55836


BECAUSE OF THE LAST ITERATION HAS A SAME VALUE
WITH THE PREVIOUS ITERATION THEN THE PROCESS IS FINISHED


PRESS ANY KEY TO CONTINUE ......._
```

Lampiran 10

Program Pascal untuk empat data hilang

```
program mrtode_aproksimasi;
uses wincrt;
const nk=5;
      np=3;
var i,j,k,n,m,mandeg:byte;
    BL:char;
    total:real;
    kond1,kond2,kond3,kond4:boolean;
    b,t,bn,kn,br,tr,x,x21,x23,x34,x15:array[1..10] of real;
    c,data:array[1..10,1..10] of real;
begin
 clrscr;
 for i:=1 to nk do
  begin
   bn[i]:=np;
   for j:=1 to np do
    begin
     clrscr;
     write('masukkan data kelompok ',i,' perlakuan ',j,' :
     ');readln(data[i,j]);
     b[i]:=b[i]+data[i,j];
     if data[i,j]=0 then bn[i]:=bn[i]-1;

    end;
  end;clrscr;
 writeln('aproksimasi akan diproses');
 write('masukkan banyaknya iterasi yang akan diproses :
 ');readln(mandeg);
 writeln;writeln;
 writeln('  x21  ','     ','  x23  ','     ','  x34  ','     ','
 x15  ');
 total:=0;
 for i:=1 to np do
  begin
   kn[i]:=nk;
   for j:=1 to nk do
    begin
     t[i]:=t[i]+data[j,i];
     if data[j,i]=0 then kn[i]:=kn[i]-1;
    end;
    total:=total+t[i];
  end;
 for i:=1 to nk do br[i]:=b[i]/bn[i];
 for j:=1 to np do tr[j]:=t[j]/kn[j];
 k:=0;
 for i:=1 to nk do
  for j:=1 to np do
   if data[i,j]=0 then
    begin
     k:=k+1;
     c[i,j]:=(br[i]+tr[j])/2;
     x[k]:=c[i,j];
    end;
 x21[1]:=x[1];
```

```
x23[1]:=x[2];
x34[1]:=x[3];
x15[1]:=(np*t[1]+nk*b[5]-total-x21[1]-x23[1]-x34[1])/((np-1)*(nk-
1));
i:=2;
repeat
 x21[i]:=(np*(t[2]+x23[i-1])+nk*b[1]-total-x15[i-1]-x23[i-1]-
         x34[i-1])/((np-1)*(nk-1));
 x23[i]:=(np*(t[2]+x21[i])+nk*b[3]-total-x21[i]-x15[i-1]-x34[i-
         1])/((np-1)*(nk-1));
 x34[i]:=(np*t[3]+nk*b[4]-total-x21[i-1]-x23[i]-x15[i-1])/((np-
         1)*(nk-1));
 x15[i]:=(np*t[1]+nk*b[5]-total-x21[i-1]-x23[i-1]-x34[i])/((np-
         1)*(nk-1));
 writeln(x21[i]:7:5,'     ',x23[i]:7:5,'     ',x34[i]:7:5,'
         ',x15[i]:7:5);
 m:=i;n:=i;
 repeat
  n:=n-1;
  if x21[m]=x21[n] then
   begin
    n:=0;
    kond1:=true;
   end;
  if x23[m]=x23[n] then
   begin
    n:=0;
    kond2:=true;
   end;
  if x34[m]=x34[n] then
   begin
    n:=0;
    kond3:=true;
   end;
  if x15[m]=x15[n] then
   begin
    n:=0;
    kond4:=true;
   end;
  until n=0;
  i:=i+1;
  if kond1=true then x21[i]:=x21[m];
  if kond2=true then x23[i]:=x23[m];
  if kond3=true then x34[i]:=x34[m];
  if kond4=true then x15[i]:=x15[m];
until i=mandeg+2;writeln;writeln;

writeln('karena nilai dari aproksimasi terakhir sudah sama dengan
nilai sebelumnya');
writeln('maka sudah diperoleh nilai aproksimasi dari data hilang
');
WRITELN;
WRITE('PRESS ANY KEY TO CONTINUE .......');READ(BL);
IF BL = '' THEN CLRSCR;

readln
end.
```

Lampiran 11

Output program pascal untuk dua data hilang

```
C:\TPW\TAMBAHAN.EXE
aproksimasi akan diproses
masukkan banyaknya iterasi yang akan diproses : 1


  x21          x23          x34          x15
1.23650      1.40424      1.22143      1.67732


karena nilai dari aproksimasi terakhir sudah sama dengan nilai sebelumny
maka sudah diperoleh nilai aproksimasi dari data hilang

PRESS ANY KEY TO CONTINUE .......
```

```
C:\TPW\TAMBAHAN.EXE
aproksimasi akan diproses
masukkan banyaknya iterasi yang akan diproses : 2


  x21          x23          x34          x15
1.23650      1.40424      1.22143      1.67732
1.24390      1.42135      1.23114      1.69406


karena nilai dari aproksimasi terakhir sudah sama dengan nilai sebelumny
maka sudah diperoleh nilai aproksimasi dari data hilang

PRESS ANY KEY TO CONTINUE .......
```

```
C:\TPW\TAMBAHAN.EXE
aproksimasi akan diproses
masukkan banyaknya iterasi yang akan diproses : 3


 x21          x23          x34          x15
1.23650      1.40424      1.22143      1.67732
1.24390      1.42135      1.23114      1.69406
1.24487      1.41829      1.22850      1.69133


karena nilai dari aproksimasi terakhir sudah sama dengan nilai sebelumny
maka sudah diperoleh nilai aproksimasi dari data hilang

PRESS ANY KEY TO CONTINUE ......._
```

```
C:\TPW\TAMBAHAN.EXE
aproksimasi akan diproses
masukkan banyaknya iterasi yang akan diproses : 4


 x21          x23          x34          x15
1.23650      1.40424      1.22143      1.67732
1.24390      1.42135      1.23114      1.69406
1.24487      1.41829      1.22850      1.69133
1.24478      1.41894      1.22864      1.69157


karena nilai dari aproksimasi terakhir sudah sama dengan nilai sebelumny
maka sudah diperoleh nilai aproksimasi dari data hilang

PRESS ANY KEY TO CONTINUE .......
```

```
C:\TPW\TAMBAHAN.EXE
aproksimasi akan diproses
masukkan banyaknya iterasi yang akan diproses : 5


  x21          x23          x34          x15
1.23650      1.40424      1.22143      1.67732
1.24390      1.42135      1.23114      1.69406
1.24487      1.41829      1.22850      1.69133
1.24478      1.41894      1.22864      1.69157
1.24489      1.41892      1.22863      1.69150


karena nilai dari aproksimasi terakhir sudah sama dengan nilai sebelumny
maka sudah diperoleh nilai aproksimasi dari data hilang

PRESS ANY KEY TO CONTINUE .......
```

```
C:\TPW\TAMBAHAN.EXE
aproksimasi akan diproses
masukkan banyaknya iterasi yang akan diproses : 6


  x21          x23          x34          x15
1.23650      1.40424      1.22143      1.67732
1.24390      1.42135      1.23114      1.69406
1.24487      1.41829      1.22850      1.69133
1.24478      1.41894      1.22864      1.69157
1.24489      1.41892      1.22863      1.69150
1.24490      1.41893      1.22862      1.69149


karena nilai dari aproksimasi terakhir sudah sama dengan nilai sebelumny
maka sudah diperoleh nilai aproksimasi dari data hilang

PRESS ANY KEY TO CONTINUE ......._
```

```
C:\TPW\TAMBAHAN.EXE
aproksimasi akan diproses
masukkan banyaknya iterasi yang akan diproses : 7


  x21          x23          x34          x15
1.23650      1.40424      1.22143      1.67732
1.24390      1.42135      1.23114      1.69406
1.24487      1.41829      1.22850      1.69133
1.24478      1.41894      1.22864      1.69157
1.24489      1.41892      1.22863      1.69150
1.24490      1.41893      1.22862      1.69149
1.24490      1.41893      1.22862      1.69149


karena nilai dari aproksimasi terakhir sudah sama dengan nilai sebelumny
maka sudah diperoleh nilai aproksimasi dari data hilang

PRESS ANY KEY TO CONTINUE ......._
```

Lampiran 12

$F_{0.05}(r_1, r_2)$

r_2 \ r_1	1	2	3	4	5	6	7	8	9	10	12	15	20	24	30	40	60	120	∞
1	161.4	199.5	215.7	224.6	230.2	234.0	236.8	238.9	240.5	241.9	243.9	245.9	248.0	249.1	250.1	251.1	252.2	253.3	254.3
2	18.51	19.00	19.16	19.25	19.30	19.33	19.35	19.37	19.38	19.40	19.41	19.43	19.45	19.45	19.46	19.47	19.48	19.49	19.50
3	10.13	9.55	9.28	9.12	9.01	8.94	8.89	8.85	8.81	8.79	8.74	8.70	8.66	8.64	8.62	8.59	8.57	8.55	8.53
4	7.71	6.94	6.59	6.39	6.26	6.16	6.09	6.04	6.00	5.96	5.91	5.86	5.80	5.77	5.75	5.72	5.69	5.66	5.63
5	6.61	5.79	5.41	5.19	5.05	4.95	4.88	4.82	4.77	4.74	4.68	4.62	4.56	4.53	4.50	4.46	4.43	4.40	4.36
6	5.99	5.14	4.76	4.53	4.39	4.28	4.21	4.15	4.10	4.06	4.00	3.94	3.87	3.84	3.81	3.77	3.74	3.70	3.67
7	5.59	4.74	4.35	4.12	3.97	3.87	3.79	3.73	3.68	3.64	3.57	3.51	3.44	3.41	3.38	3.34	3.30	3.27	3.23
8	5.32	4.46	4.07	3.84	3.69	3.58	3.50	3.44	3.39	3.35	3.28	3.22	3.15	3.12	3.08	3.04	3.01	2.97	2.93
9	5.12	4.26	3.86	3.63	3.48	3.37	3.29	3.23	3.18	3.14	3.07	3.01	2.94	2.90	2.86	2.83	2.79	2.75	2.71
10	4.96	4.10	3.71	3.48	3.33	3.22	3.14	3.07	3.02	2.98	2.91	2.85	2.77	2.74	2.70	2.66	2.62	2.58	2.54
11	4.84	3.98	3.59	3.36	3.20	3.09	3.01	2.95	2.90	2.85	2.79	2.72	2.65	2.61	2.57	2.53	2.49	2.45	2.40
12	4.75	3.89	3.49	3.26	3.11	3.00	2.91	2.85	2.80	2.75	2.69	2.62	2.54	2.51	2.47	2.43	2.38	2.34	2.30
13	4.67	3.81	3.41	3.18	3.03	2.92	2.83	2.77	2.71	2.67	2.60	2.53	2.46	2.42	2.38	2.34	2.30	2.25	2.21
14	4.60	3.74	3.34	3.11	2.96	2.85	2.76	2.70	2.65	2.60	2.53	2.46	2.39	2.35	2.31	2.27	2.22	2.18	2.13
15	4.54	3.68	3.29	3.06	2.90	2.79	2.71	2.64	2.59	2.54	2.48	2.40	2.33	2.29	2.25	2.20	2.16	2.11	2.07
16	4.49	3.63	3.24	3.01	2.85	2.74	2.66	2.59	2.54	2.49	2.42	2.35	2.28	2.24	2.19	2.15	2.11	2.06	2.01
17	4.45	3.59	3.20	2.96	2.81	2.70	2.61	2.55	2.49	2.45	2.38	2.31	2.23	2.19	2.15	2.10	2.06	2.01	1.96
18	4.41	3.55	3.16	2.93	2.77	2.66	2.58	2.51	2.46	2.41	2.34	2.27	2.19	2.15	2.11	2.06	2.02	1.97	1.92
19	4.38	3.52	3.13	2.90	2.74	2.63	2.54	2.48	2.42	2.38	2.31	2.23	2.16	2.11	2.07	2.03	1.98	1.93	1.88
20	4.35	3.49	3.10	2.87	2.71	2.60	2.51	2.45	2.39	2.35	2.28	2.20	2.12	2.08	2.04	1.99	1.95	1.90	1.84
21	4.32	3.47	3.07	2.84	2.68	2.57	2.49	2.42	2.37	2.32	2.25	2.18	2.10	2.05	2.01	1.96	1.92	1.87	1.81
22	4.30	3.44	3.05	2.82	2.66	2.55	2.46	2.40	2.34	2.30	2.23	2.15	2.07	2.03	1.98	1.94	1.89	1.84	1.78
23	4.28	3.42	3.03	2.80	2.64	2.53	2.44	2.37	2.32	2.27	2.20	2.13	2.05	2.01	1.96	1.91	1.86	1.81	1.76
24	4.26	3.40	3.01	2.78	2.62	2.51	2.42	2.36	2.30	2.25	2.18	2.11	2.03	1.98	1.94	1.89	1.84	1.79	1.73
25	4.24	3.39	2.99	2.76	2.60	2.49	2.40	2.34	2.28	2.24	2.16	2.09	2.01	1.96	1.92	1.87	1.82	1.77	1.71
26	4.23	3.37	2.98	2.74	2.59	2.47	2.39	2.32	2.27	2.22	2.15	2.07	1.99	1.95	1.90	1.85	1.80	1.75	1.69
27	4.21	3.35	2.96	2.73	2.57	2.46	2.37	2.31	2.25	2.20	2.13	2.06	1.97	1.93	1.88	1.84	1.79	1.73	1.67
28	4.20	3.34	2.95	2.71	2.56	2.45	2.36	2.29	2.24	2.19	2.12	2.04	1.96	1.91	1.87	1.82	1.77	1.71	1.65
29	4.18	3.33	2.93	2.70	2.55	2.43	2.35	2.28	2.22	2.18	2.10	2.03	1.94	1.90	1.85	1.81	1.75	1.70	1.64
30	4.17	3.32	2.92	2.69	2.53	2.42	2.33	2.27	2.21	2.16	2.09	2.01	1.93	1.89	1.84	1.79	1.74	1.68	1.62
40	4.08	3.23	2.84	2.61	2.45	2.34	2.25	2.18	2.12	2.08	2.00	1.92	1.84	1.79	1.74	1.69	1.64	1.58	1.51
60	4.00	3.15	2.76	2.53	2.37	2.25	2.17	2.10	2.04	1.99	1.92	1.84	1.75	1.70	1.65	1.59	1.53	1.47	1.39
120	3.92	3.07	2.68	2.45	2.29	2.17	2.09	2.02	1.96	1.91	1.83	1.75	1.66	1.61	1.55	1.55	1.43	1.35	1.25
∞	3.84	3.00	2.60	2.37	2.21	2.10	2.01	1.94	1.88	1.83	1.75	1.67	1.57	1.52	1.46	1.39	1.32	1.22	1.00

Column header: Degrees of Freedom for the Numerator (r_1)

IV. Percentage Points of the *F* Distribution (*continued*)

Lampiran 13

II. Percentage Points of the t Distribution[a]

ν \ α	.40	.25	.10	.05	.025	.01	.005	.0025	.001	.0005
1	.325	1.000	3.078	6.314	12.706	31.821	63.657	127.32	318.31	636.62
2	.289	.816	1.886	2.920	4.303	6.965	9.925	14.089	23.326	31.598
3	.277	.765	1.638	2.353	3.182	4.541	5.841	7.453	10.213	12.924
4	.271	.741	1.533	2.132	2.776	3.747	4.604	5.598	7.173	8.610
5	.267	.727	1.476	2.015	2.571	3.365	4.032	4.773	5.893	6.869
6	.265	.727	1.440	1.943	2.447	3.143	3.707	4.317	5.208	5.959
7	.263	.711	1.415	1.895	2.365	2.998	3.499	4.019	4.785	5.408
8	.262	.706	1.397	1.860	2.306	2.896	3.355	3.833	4.501	5.041
9	.261	.703	1.383	1.833	2.262	2.821	3.250	3.690	4.297	4.781
10	.260	.700	1.372	1.812	2.228	2.764	3.169	3.581	4.144	4.587
11	.260	.697	1.363	1.796	2.201	2.718	3.106	3.497	4.025	4.437
12	.259	.695	1.356	1.782	2.179	2.681	3.055	3.428	3.930	4.318
13	.259	.694	1.350	1.771	2.160	2.650	3.012	3.372	3.852	4.221
14	.258	.692	1.345	1.761	2.145	2.624	2.977	3.326	3.787	4.140
15	.258	.691	1.341	1.753	2.131	2.602	2.947	3.286	3.733	4.073
16	.258	.690	1.337	1.746	2.120	2.583	2.921	3.252	3.686	4.015
17	.257	.689	1.333	1.740	2.110	2.567	2.898	3.222	3.646	3.965
18	.257	.688	1.330	1.734	2.101	2.552	2.878	3.197	3.610	3.922
19	.257	.688	1.328	1.729	2.093	2.539	2.861	3.174	3.579	3.883
20	.257	.687	1.325	1.725	2.086	2.528	2.845	3.153	3.552	3.850
21	.257	.686	1.323	1.721	2.080	2.518	2.831	3.135	3.527	3.819
22	.256	.686	1.321	1.717	2.074	2.508	2.819	3.119	3.505	3.792
23	.256	.685	1.319	1.714	2.069	2.500	2.807	3.104	3.485	3.767
24	.256	.685	1.318	1.711	2.064	2.492	2.797	3.091	3.467	3.745
25	.256	.684	1.316	1.708	2.060	2.485	2.787	3.078	3.450	3.725
26	.256	.684	1.315	1.706	2.056	2.479	2.779	3.067	3.435	3.707
27	.256	.684	1.314	1.703	2.052	2.473	2.771	3.057	3.421	3.690
28	.256	.683	1.313	1.701	2.048	2.467	2.763	3.047	3.408	3.674
29	.256	.683	1.311	1.699	2.045	2.462	2.756	3.038	3.396	3.659
30	.256	.683	1.310	1.697	2.042	2.457	2.750	3.030	3.385	3.646
40	.255	.681	1.303	1.684	2.021	2.423	2.704	2.971	3.307	3.551
60	.254	.679	1.296	1.671	2.000	2.390	2.660	2.915	3.232	3.460
120	.254	.677	1.289	1.658	1.980	2.358	2.617	2.860	3.160	3.373
∞	.253	.674	1.282	1.645	1.960	2.326	2.576	2.807	3.090	3.291

ν = degrees of freedom.

[a] Adapted with permission from *Biometrika Tables for Statisticians*, Vol. 1, 3rd edition, by E. S. Pearson and H. O. Hartley, Cambridge University Press, Cambridge, 1966.