Table 8B:

```
##############################
## Variable Set Up       #####
##############################
#####################################################################
hsig_list=c(0,0.125,0.25,0.5,1.0,2.0,4.0);
nj_list=c(20, 80);
al1_list=c(0, 1);
Simn=10000;

model=1;
ind=1;
Bias=50;
#####################################################################




#####################################################################
# Import 'metafor' Package and set seed
#####################################################################
set.seed(7);
library(tcltk)
library(metafor)


#####################################################################
# Generating data for primary data
#####################################################################
dgp <-function(al1, sigh,ind){
  #DGP: refer to step 1 in page 25;
  obs=sample(c(62,125,250,500,1000),1);
  x1=runif(obs, min = 100, max = 200);
  x2=x1+rnorm(obs, mean = 0, sd = 50);
  x3=x1+rnorm(obs, mean = 0, sd = 50);
  if (ind==0){
    z = 100 + al1*x1 + 0.5*x2 + rnorm(1,mean=0,sd=sigh)*x3+rnorm(obs,mean=0,sd=100);
  } else if (ind==1) {
    z = 100 + (al1+rnorm(1,mean=0,sd=sigh))*x1 + 0.5*x2 + rnorm(obs,mean=0,sd=100);}
  return (as.data.frame(cbind(z, x1 ,x2)))
}


```

```r
#################################################################
# Generating data for MRA study
#################################################################
dtcollection <- function(al1, sigh, ssize, Bias,ind){
  output =  matrix(0, nrow=ssize, ncol=5);
  colnames(output) <- c("id","y","al_se","mj","Significant");
  num_publ=ssize*(Bias/100);
  for(i in 1:ssize) {
    output[i,1]=i;
    if (i<=num_publ){
      while (output[i,5]==0){
        data=dgp(al1,sigh,ind);
        model_selection=sample(c(1.1, 1.2),1);
        if (model_selection==1.1) {
          out <- lm(data[,1] ~ data[,2] + data[,3]);
          output[i,4]=0;
        } else if (model_selection==1.2){
          out <- lm(data[,1] ~ data[,2])
          output[i,4]=1;
        } else { cat("Model Selection Error", "\n"); }
        output[i,2]=coefficients(out)[2];
        output[i,3]=sqrt(diag(vcov(out)))[2];
        output[i,5]=((summary(out)$coefficients[2,4]<=0.05)*(0<=summary(out)$coefficients
            [2,1]));
      }
    } else if (i>num_publ){
      data=dgp(al1,sigh,ind);
      model_selection=sample(c(1.1, 1.2),1);
      if (model_selection==1.1) {
        out <- lm(data[,1] ~ data[,2] + data[,3])
        output[i,4]=0;
      } else if (model_selection==1.2){
        out <- lm(data[,1] ~ data[,2]);
        output[i,4]=1;
      } else { cat("Model Selection Error", "\n"); }

      output[i,2]=coefficients(out)[2];
      output[i,3]=sqrt(diag(vcov(out)))[2];
      output[i,5]=((summary(out)$coefficients[2,4]<=0.05)*(0<summary(out)$coefficients[2,1])
          );
    } else { cat("Publication Bias Error", "\n"); }
  }
  return(output)
}
#################################################################
```

```
88  ##################################################################
89  # Simulation Begins
90  ##################################################################
91  if(model==0){
92    mdl=y~mj;
93  }else if (model==1) {
94    mdl=y~al_se+mj;
95  }else if (model==2) {
96    mdl=y~al_se2+mj;
97  }
98
99
100 nrow=length(hsig_list)*length(nj_list)*length(al1_list)
101 output_zero  =  matrix(0, nrow=nrow, ncol=13);
102 colnames(output_zero) <- c("MRA_Size","h_sigma","TrueEffect","FE-ERR(B0)","RE-ERR(B0)","WLS-
        ERR(B0)","I2","FE-E[B0]","RE-E[B0]","WLS-E[B0]","FE-sig(B0)","RE-sig(B0)", "WLS-sig(B0)"
        );
103 output_one  =  matrix(0, nrow=nrow, ncol=13);
104 colnames(output_one) <- c("MRA_Size","h_sigma","TrueEffect","FE-ERR(B1)","RE-ERR(B1)","WLS-
        ERR(B1)","I2","FE-E[B1]","RE-E[B1]","WLS-E[B1]","FE-sig(B1)","RE-sig(B1)", "WLS-sig(B1)"
        );
105 output_two  =  matrix(0, nrow=nrow, ncol=13);
106 colnames(output_two) <- c("MRA_Size","h_sigma","TrueEffect","FE-ERR(B2)","RE-ERR(B2)","WLS-
        ERR(B2)","I2","FE-E[B1]","RE-E[B2]","WLS-E[B2]","FE-sig(B2)","RE-sig(B2)", "WLS-sig(B2)"
        );
107
108 cnt=1;
109 start.time <- Sys.time()
110
111
112 for (k in al1_list){ al1=k;
113 for (j in nj_list) { nj=j;
114 for (l in hsig_list){
115   hsig=l;
116   output_b0 =  matrix(0, nrow=Simn, ncol=7);
117   colnames(output_b0) <- c("FE-MRA","RE_MRA","WLS-MRA", "I2","FE_Coeff","RE_Coeff","WLS_
          Coeff");
118   output_b1 =  matrix(0, nrow=Simn, ncol=7);
119   colnames(output_b1) <- c("FE-MRA","RE_MRA","WLS-MRA", "I2","FE_Coeff","RE_Coeff","WLS_
          Coeff");
120   output_b2 =  matrix(0, nrow=Simn, ncol=7);
121   colnames(output_b2) <- c("FE-MRA","RE_MRA","WLS-MRA", "I2","FE_Coeff","RE_Coeff","WLS_
          Coeff");
122   cat("Preparing row", cnt, "/", nrow, "\n");
123   pc=0;
```

```
124   for(i in 1:Simn) {
125     flag=TRUE
126     while (flag==TRUE){
127       flag=FALSE
128       MRAdata=as.data.frame(dtcollection(al1, hsig, nj, Bias, ind));
129       al_se2=MRAdata$al_se*MRAdata$al_se;
130       test=try(rma.uni(mdl,vi=al_se*al_se, intercept=TRUE, data=MRAdata, weighted=TRUE,
              method="REML", level=95, digits=5), silent =TRUE)
131       if(summary(test)[2]=="try-error") {flag=TRUE; }
132       if(flag==FALSE) { if(is.numeric(test$R2)==FALSE) {flag=TRUE;}}
133     }
134
135     reg_fe=rma.uni(mdl,vi=al_se*al_se, intercept=TRUE, data=MRAdata, weighted=TRUE, method="
          FE", level=95, digits=5)
136     output_b0[i,1]=(0==(reg_fe$ci.lb[1]<al1)*(al1<reg_fe$ci.ub[1]))
137     output_b1[i,1]=(0==(reg_fe$ci.lb[2]<0)*(0<reg_fe$ci.ub[2]))
138     output_b2[i,1]=(0==(reg_fe$ci.lb[3]<0)*(0<reg_fe$ci.ub[3]))
139     output_b0[i,5]=(coefficients(reg_fe)[1]);
140     output_b1[i,5]=(coefficients(reg_fe)[2]);
141     output_b2[i,5]=(coefficients(reg_fe)[3]);
142
143     reg_re=rma.uni(mdl,vi=al_se*al_se, intercept=TRUE, data=MRAdata, weighted=TRUE, method="
          REML", level=95, digits=5)
144     output_b0[i,2]=(0==(reg_re$ci.lb[1]<al1)*(al1<reg_re$ci.ub[1]))
145     output_b0[i,6]=(coefficients(reg_re)[1]);
146     output_b1[i,2]=(0==(reg_re$ci.lb[2]<0)*(0<reg_re$ci.ub[2]))
147     output_b1[i,6]=(coefficients(reg_re)[2]);
148     output_b2[i,2]=(0==(reg_re$ci.lb[3]<0)*(0<reg_re$ci.ub[3]))
149     output_b2[i,6]=(coefficients(reg_re)[3]);
150
151
152     reg_wls=lm (mdl, data=MRAdata,weights=(1/(al_se*al_se)))
153     output_b0[i,3]=(0==(confint(reg_wls)[1,1]<al1)*(al1<confint(reg_wls)[1,2]))
154     output_b0[i,7]=(coefficients(reg_wls)[1]);
155     output_b1[i,3]=(0==(confint(reg_wls)[2,1]<0)*(0<confint(reg_wls)[2,2]))
156     output_b1[i,7]=(coefficients(reg_wls)[2]);
157     output_b2[i,3]=(0==(confint(reg_wls)[3,1]<0)*(0<confint(reg_wls)[3,2]))
158     output_b2[i,7]=(coefficients(reg_wls)[3]);
159
160     output_b0[i,4]=reg_re$I2;
161     output_b1[i,4]=reg_re$I2;
162     output_b2[i,4]=reg_re$I2;
163     pc=pc+1;
164     if((i-1)%%(Simn/10)==0){cat(10*(i-1)/(Simn/10), ".. ") }
165   }
```

```
166   output_zero[cnt,]=c(nj, hsig, al1, mean(output_b0[,1]), mean(output_b0[,2]), mean(output_
          b0[,3]), mean(output_b0[,4])/100, mean(output_b0[,5]), mean(output_b0[,6]), mean(
          output_b0[,7]),mean((output_b0[,5]-al1)^2), mean((output_b0[,6]-al1)^2), mean((output_
          b0[,7]-al1)^2))
167   output_one[cnt,]=c(nj, hsig, al1, mean(output_b1[,1]), mean(output_b1[,2]), mean(output_b1
          [,3]), mean(output_b1[,4])/100, mean(output_b1[,5]), mean(output_b1[,6]), mean(output_
          b1[,7]),sqrt(var(output_b1[,5])), sqrt(var(output_b1[,6])), sqrt(var(output_b1[,7])))
168   output_two[cnt,]=c(nj, hsig, al1, mean(output_b2[,1]), mean(output_b2[,2]), mean(output_b2
          [,3]), mean(output_b2[,4])/100, mean(output_b2[,5]), mean(output_b2[,6]), mean(output_
          b2[,7]),sqrt(var(output_b2[,5])), sqrt(var(output_b2[,6])), sqrt(var(output_b2[,7])))
169   cat("Done!", "\n");
170   cnt=cnt+1;
171 }}}
172
173 ##################################################################
174 ##################################################################
175 ##################################################################
176
177 end.time <- Sys.time()
178 time.taken <- end.time - start.time
179 time.taken
180
181
182
183 round(output_zero, 6)
184 round(output_one, 6)
185 round(output_two, 6)
```