MEMORANDUM

TO: ALEXIS. J. DIAMOND

FROM: WAHOME BRIAN GITHIRE.

SUBJECT: STATISTICAL ANALYSIS OF THE LALONDE 3 WAYS PROGRAM.

DATE: SATURDAY, FEBRUARY 17TH.

CLASS: CS112-SESSION

EXECUTIVE SUMMARY.

A detailed review of data acquired from our Lalonde program was deemed necessary to determine any strong associations between the 12 observed variables (Listed in the code) and our outcome variable, re78(Real earnings, 1978). To do this, we employed multivariable regression analysis on a subset of the data that had a degree and another that had no degree. We conducted a similar analysis on the whole data set to try and validate such relationships, if any.

Other methods used were random forests with bootstrap aggregation, logistics regression, 10-fold and LOOCV model cross validation. All of these aimed at formalizing variable associations, validating these relationships and informing us of the most significant impact variables. (For elevated consideration in future interventions).

Assuming a linear relationship between our outcome variable (re78) and the predictor variables was a necessary abstraction to simplify the complex nature of the job sector our program targeted. Our primary predictor was treat.

Multiple analyses that incorporated it indicate that it bore statistical significance on re78. Treat showed no statistical significance in the multivariable models for nodegr ==1 (P-Value: 13.003%) and nodegr==0 (P-Value: 12.1%). It should be noted however, it was very significant when the full dataset was considered. (P-Value: 0.8%). It should be noted; a model of the confidence intervals indicate that treat always had a positive impact on re78 supporting the idea statistical insignificance does not imply practical insignificance.

The variable black was highly significant in the models with nodegr==1 (P-Value: 0.94%) and the full dataset (P-Value: 0.70%). In all 3 cases, including nodegr == 0, it had a negative effect on re78. Other significant variables were u75 in the nodegr==1 data set and educ in the full dataset. The irony is, in both cases the impact was positive.

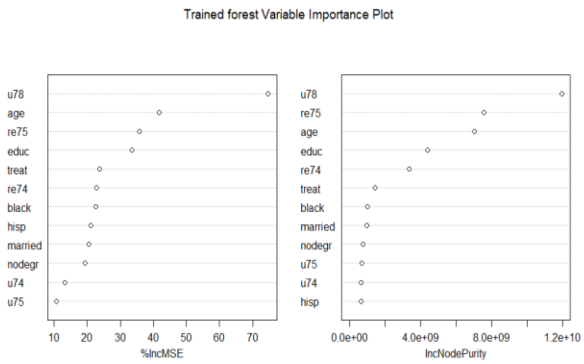
We could make the following real-life inferences following these observations:

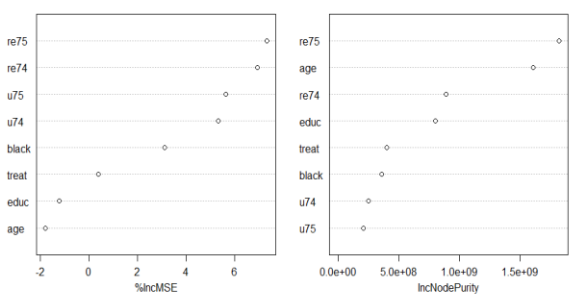
- Your earnings in 1978 could be negatively impacted if you are black. This could be attributed to racism and discrimination in the job field.
- Taking part in the job training program would assuredly give your earnings a boost.
- Being unemployed in the previous years does not spell doom on your earnings in 1978. If anything, the effect is positive. Perhaps the unemployed individuals found jobs within the year hence a positive impact of the variables on re78.

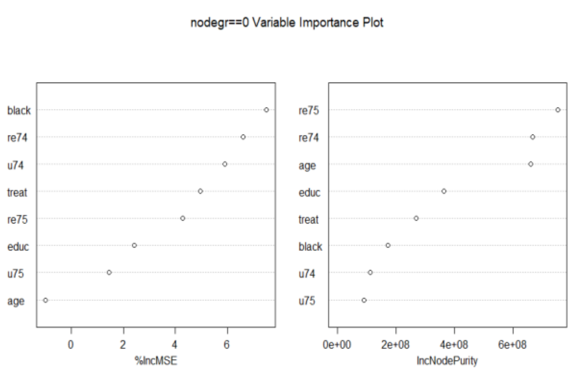
```
Call:
glm(formula = re78 ~ treat + re74 + re75 + u74 + u75 + age + 
    educ + black, data = lalonde)

Deviance Residuals: 
   Min      1Q  Median      3Q     Max  
 -9921   -4401   -1577    3111   53183  

Coefficients:
              Estimate Std. Error t value Pr(>|t|)   
(Intercept)  5.963e+02  2.316e+03   0.257  0.79694   
treat        1.655e+03  6.303e+02   2.625  0.00897 **
re74         1.215e-01  8.637e-02   1.407  0.16023   
re75         1.602e-02  1.445e-01   0.111  0.91180   
u74          1.334e+03  1.172e+03   1.138  0.25570   
u75         -1.048e+03  1.014e+03  -1.034  0.30182   
age          5.191e+01  4.439e+01   1.169  0.24287   
educ         3.935e+02  1.744e+02   2.256  0.02456 * 
black       -2.262e+03  8.342e+02  -2.712  0.00696 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 42185667)

    Null deviance: 1.9526e+10  on 444  degrees of freedom
Residual deviance: 1.8393e+10  on 436  degrees of freedom
AIC: 9086.9

Number of Fisher Scoring iterations: 2
```

```
Call:
glm(formula = re78 ~ treat + re74 + re75 + u74 + u75 + age + 
    educ + black, data = Degree1)

Deviance Residuals: 
   Min      1Q  Median      3Q     Max  
 -7559   -4174   -1563    2871   52761  

Coefficients:
              Estimate Std. Error t value Pr(>|t|)   
(Intercept)  3.830e+03  2.833e+03   1.352   0.1773   
treat        1.067e+03  7.035e+02   1.517   0.1303   
re74         3.089e-02  9.654e-02   0.320   0.7492   
re75        -7.099e-02  1.527e-01  -0.465   0.6424   
u74          1.288e+03  1.250e+03   1.031   0.3034   
u75         -2.372e+03  1.111e+03  -2.136   0.0334 * 
age          5.172e+01  4.723e+01   1.095   0.2743   
educ         2.227e+02  2.279e+02   0.977   0.3293   
black       -2.444e+03  9.358e+02  -2.612   0.0094 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 38934262)

    Null deviance: 1.3823e+10  on 347  degrees of freedom
Residual deviance: 1.3199e+10  on 339  degrees of freedom
AIC: 7080.6

Number of Fisher Scoring iterations: 2
```

```
Call:
glm(formula = re78 ~ treat + re74 + re75 + u74 + u75 + age + 
    educ + black, data = Degree0)

Deviance Residuals: 
   Min      1Q  Median      3Q     Max  
-11223   -4332   -1863    2934   25681  

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.848e+04  1.227e+04  -1.506    0.136
treat        2.395e+03  1.528e+03   1.568    0.121
re74         2.909e-01  2.121e-01   1.371    0.174
re75         7.049e-01  4.592e-01   1.535    0.128
u74          1.689e+03  3.092e+03   0.546    0.586
u75          2.867e+03  2.472e+03   1.160    0.249
age          9.528e+00  1.284e+02   0.074    0.941
educ         1.672e+03  1.031e+03   1.622    0.108
black       -2.049e+03  1.888e+03  -1.085    0.281

(Dispersion parameter for gaussian family taken to be 51087249)

    Null deviance: 5483177032  on 96  degrees of freedom
Residual deviance: 4495677946  on 88  degrees of freedom
AIC: 2007.5

Number of Fisher Scoring iterations: 2
```



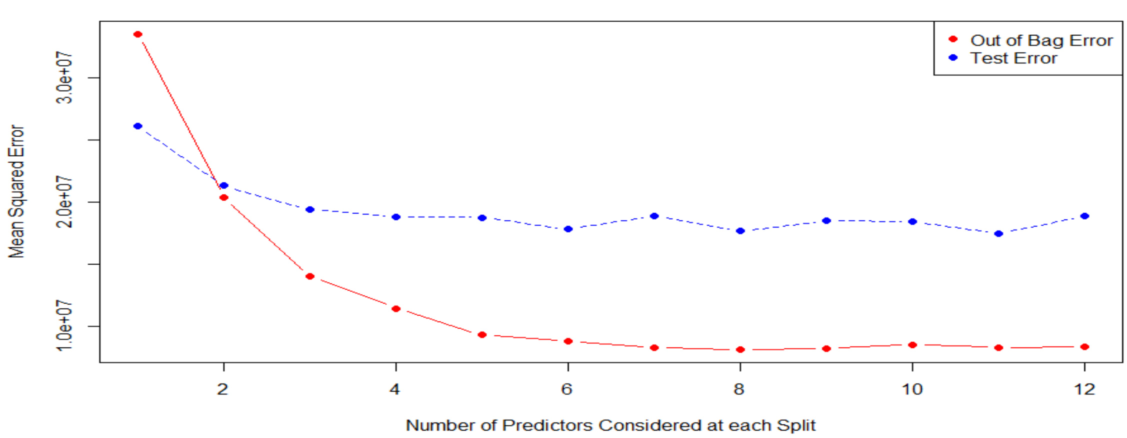




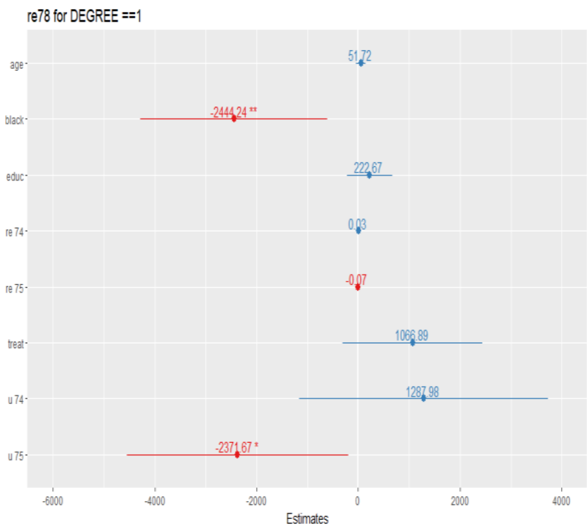
The random forests technique was applied on models like the ones above. The idea was to estimate the treatment effects using another modelling technique to have a comparison criterion. The beauty of random forests is that atop estimating the treatment. Their plots give us a hierarchy of the variables based on importance in the model. To validate the **models,** cross validation techniques, i.e. LOOCV and 10-fold cross validation were done to determine what degree of the prediction polynomial would be necessary to have a well working model. Bootstrapping was incorporated to reduce the variance of the model. A plot of the same was then used to cross check the outcomes of the cross validations. The plots and inferences are as follows:

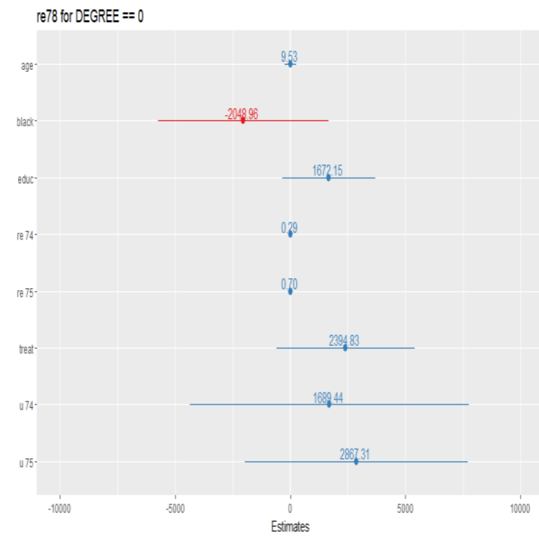
Inferences:

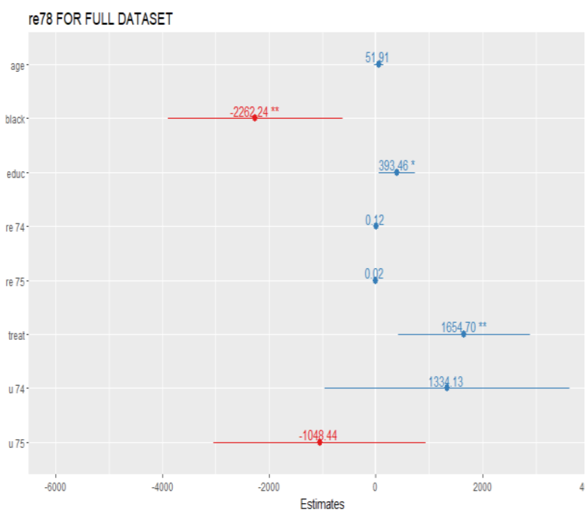
From the above, we note that for the trained set (Bootstrapping applied), nodegr == 1 and nodegr==0, the most important variables are **u78, re75, u78** and black respectively. The model seems to have and OOB error minimized when about **4-6 degrees** polynomials are used. (Cross validation results supporting this are appended in the appendix sections.)

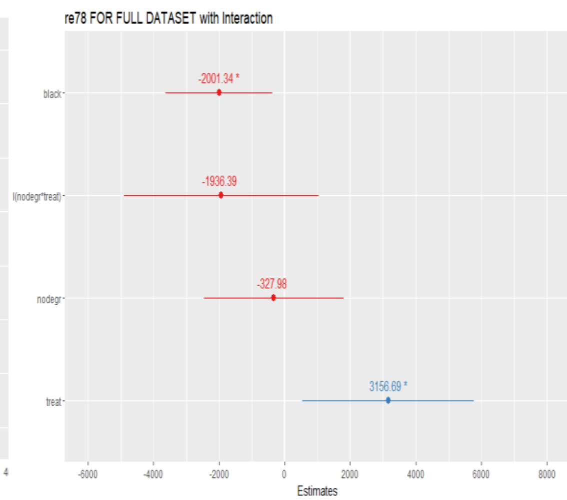


The following confidence interval plots and summaries were generated from the three models, including the model that factored in the interaction **I(nodegr*treat)**:









The plots above indicate the 95% confidence intervals for the different variables as well as the effect estimate. Statistically significant variables are also denoted by an Asterix next to the effect estimate. For the **nodegr==1**, **nodegr==0**, **full dataset** and **full data set with Interaction terms**, the treatment effects were as follows: **Estm: 1066.89, CI .95(-204.5044, 2384.6931)**, **Estm: 2394.83, CI .95(380.8884, 5210.2969)**, **Estm: 1654.70, CI .95(536.5745, 2753.8879), Estm: 3156.69, CI .95(1082.934, 5784.671).** Note, the effect estimates are in $.

SUMMARY AND RECOMMENDATIONS.

Given the multivariable nature of all our models, our regressions allow for different treatment effects for different individuals. The most prominent example is the variation in response based on degree. Person with no high school degree will be impacted more compared to the graduants(See effects). The regressions therefore **do not require a constant treatment effect for all individuals.**

Expanding on the inference above, it was noted that the program was more effective for individuals in with a high school degree than those without. The effect amongst the individuals with a degree was higher. The variability explained by the individual models was highest in the model incorporating the treatment term **I(nodegr*treat).** The interaction term was also very significant in this model. This further supports that our model allows for different treatment effects for different individuals and emphasizes the significance of the degree effect on re78.

The analytics board recommends that future training programs target individuals with no high school degree given the effectiveness of the program. The program should also target black individuals to enhance diversity and give them a much-needed boost given the negative effect their skin color has on their earnings. The positive effect the training has would serve as a counterweight to the skin color.

APPENDIX.

MODEL SUMMARIES.

Degree holders' linear regression summary:

```
Call:
glm(formula = re78 ~ treat + re74 + re75 + u74 + u75 + age +
    educ + black, data = Degree1)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
 -7559   -4174   -1563    2871   52761

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.830e+03  2.833e+03   1.352   0.1773
treat        1.067e+03  7.035e+02   1.517   0.1303
re74         3.089e-02  9.654e-02   0.320   0.7492
re75        -7.099e-02  1.527e-01  -0.465   0.6424
u74          1.288e+03  1.250e+03   1.031   0.3034
u75         -2.372e+03  1.111e+03  -2.136   0.0334 *
age          5.172e+01  4.723e+01   1.095   0.2743
educ         2.227e+02  2.279e+02   0.977   0.3293
black       -2.444e+03  9.358e+02  -2.612   0.0094 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 38934262)

    Null deviance: 1.3823e+10  on 347  degrees of freedom
Residual deviance: 1.3199e+10  on 339  degrees of freedom
AIC: 7080.6
```

Degree holders' confidence intervals (Generated by simulation):

```
> simulationsConfidencesD1<-data.frame(quantile(simD1.coef.df$`(Intercept)` , c(0.025, 0.975)),
+                                      quantile(simD1.coef.df$`treat` , c(0.025, 0.975)),
+                                      quantile(simD1.coef.df$`re74` , c(0.025, 0.975)),
+                                      quantile(simD1.coef.df$`re75` , c(0.025, 0.975)),
+                                      quantile(simD1.coef.df$`u74` , c(0.025, 0.975)),
+                                      quantile(simD1.coef.df$`u75` , c(0.025, 0.975)),
+                                      quantile(simD1.coef.df$`age` , c(0.025, 0.975)),
+                                      quantile(simD1.coef.df$`educ` , c(0.025, 0.975)),
+                                      quantile(simD1.coef.df$`black` , c(0.025, 0.975))
+ )
> colnames(simulationsConfidencesD1) <- c("(Intercept)","treat", "re74", "re75", "u74", "u75", "age", "educ","black")
> simulationsConfidencesD1
      (Intercept)     treat       re74       re75       u74       u75      age      educ      black
2.5%    -323.2472 -204.5044 -0.1813949 -0.3814968 -1138.826 -4474.9364 -42.3067 -297.5161 -4476.0985
97.5%   9883.9133 2384.6931  0.2089532  0.2194474  4255.027  -415.2916 123.9915  514.1133  -859.3165
```

Non-Degree holder's linear regression summary:

```
Call:
glm(formula = re78 ~ treat + re74 + re75 + u74 + u75 + age +
    educ + black, data = Degree0)

Deviance Residuals:
   Min       1Q   Median       3Q      Max
-11223    -4332    -1863     2934    25681

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.848e+04  1.227e+04  -1.506    0.136
treat        2.395e+03  1.528e+03   1.568    0.121
re74         2.909e-01  2.121e-01   1.371    0.174
re75         7.049e-01  4.592e-01   1.535    0.128
u74          1.689e+03  3.092e+03   0.546    0.586
u75          2.867e+03  2.472e+03   1.160    0.249
age          9.528e+00  1.284e+02   0.074    0.941
educ         1.672e+03  1.031e+03   1.622    0.108
black       -2.049e+03  1.888e+03  -1.085    0.281

(Dispersion parameter for gaussian family taken to be 51087249)

    Null deviance: 5483177032  on 96  degrees of freedom
Residual deviance: 4495677946  on 88  degrees of freedom
AIC: 2007.5

Number of Fisher Scoring iterations: 2
```

Non-Degree holders' confidence intervals (Generated by simulation):

```
> #Simulated confint for model Deg == 1.
> simD0.coef.df=as.data.frame(simD0.coef)
> simulationsConfidencesD0<-data.frame(quantile(simD0.coef.df$`(Intercept)` , c(0.025, 0.975)),
+                                      quantile(simD0.coef.df$`treat` , c(0.025, 0.975)),
+                                      quantile(simD0.coef.df$`re74` , c(0.025, 0.975)),
+                                      quantile(simD0.coef.df$`re75` , c(0.025, 0.975)),
+                                      quantile(simD0.coef.df$`u74` , c(0.025, 0.975)),
+                                      quantile(simD0.coef.df$`u75` , c(0.025, 0.975)),
+                                      quantile(simD0.coef.df$`age` , c(0.025, 0.975)),
+                                      quantile(simD0.coef.df$`educ` , c(0.025, 0.975)),
+                                      quantile(simD0.coef.df$`black` , c(0.025, 0.975))
+                                      )
> colnames(simulationsConfidencesD0)<- c("(Intercept)","treat", "re74", "re75", "u74", "u75", "age", "educ","black")
> simulationsConfidencesD0
      (Intercept)     treat        re74      re75      u74      u75       age      educ      black
2.5%    -36211.08  380.8884 -0.06368697 0.0754625 -3629.615 -2146.559 -257.5248  264.7986 -5375.4004
97.5%    -2131.58 5210.2969  0.64164855 1.5293864  8072.685  7268.640  235.8450 3285.4472   896.6046
```

Full dataset linear regression summary:

```
Call:
glm(formula = re78 ~ treat + re74 + re75 + u74 + u75 + age +
    educ + black, data = lalonde)

Deviance Residuals:
   Min       1Q   Median       3Q      Max
 -9921    -4401    -1577     3111    53183

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.963e+02  2.316e+03   0.257  0.79694
treat        1.655e+03  6.303e+02   2.625  0.00897 **
re74         1.215e-01  8.637e-02   1.407  0.16023
re75         1.602e-02  1.445e-01   0.111  0.91180
u74          1.334e+03  1.172e+03   1.138  0.25570
u75         -1.048e+03  1.014e+03  -1.034  0.30182
age          5.191e+01  4.439e+01   1.169  0.24287
educ         3.935e+02  1.744e+02   2.256  0.02456 *
black       -2.262e+03  8.342e+02  -2.712  0.00696 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 42185667)

    Null deviance: 1.9526e+10  on 444  degrees of freedom
Residual deviance: 1.8393e+10  on 436  degrees of freedom
AIC: 9086.9

Number of Fisher Scoring iterations: 2
```

Full dataset confidence intervals. (Generated by simulation):

```
> #SIMULATIONS Full dataset
> simFull <-sim(modelFull)
> simFull.coef <- coef(simFull)
> simFull.coef.df <- as.data.frame(simFull.coef)
> #simulated confint for model Full dataset.
> simulationsConfidencesFull<-data.frame(quantile(simFull.coef.df$`(Intercept)` , c(0.025, 0.975)),
+                                 quantile(simFull.coef.df$`treat` , c(0.025, 0.975)),
+                                 quantile(simFull.coef.df$`re74` , c(0.025, 0.975)),
+                                 quantile(simFull.coef.df$`re75` , c(0.025, 0.975)),
+                                 quantile(simFull.coef.df$`u74` , c(0.025, 0.975)),
+                                 quantile(simFull.coef.df$`u75` , c(0.025, 0.975)),
+                                 quantile(simFull.coef.df$`age` , c(0.025, 0.975)),
+                                 quantile(simFull.coef.df$`educ` , c(0.025, 0.975)),
+                                 quantile(simFull.coef.df$`black` , c(0.025, 0.975))
+                                 )
>
> colnames(simulationsConfidencesFull)<- c("(Intercept)","treat", "re74", "re75", "u74", "u75", "age", "educ","black")
> simulationsConfidencesFull
      (Intercept)     treat         re74       re75       u74        u75       age      educ      black
2.5%    -3960.462  536.5745 -0.001138137 -0.2488424 -532.8489 -2926.8327 -31.79247  18.46344 -3831.0442
97.5%    4794.210 2753.8879  0.306920124  0.2934257 3666.9742   438.6132 134.12495 693.09692  -692.4347
```

Full dataset with interaction term linear regression summary:

```
Call:
glm(formula = re78 ~ I(nodegr * treat) + treat + nodegr + black,
    data = lalonde)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
 -9640   -4154   -1727    2918   54933

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)         6483.5     1202.8   5.390 1.15e-07 ***
I(nodegr * treat)  -1936.4     1516.0  -1.277   0.2022
treat               3156.7     1332.8   2.369   0.0183 *
nodegr              -328.0     1088.5  -0.301   0.7633
black              -2001.3      831.5  -2.407   0.0165 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 42515954)

    Null deviance: 1.9526e+10  on 444  degrees of freedom
Residual deviance: 1.8707e+10  on 440  degrees of freedom
AIC: 9086.4

Number of Fisher Scoring iterations: 2
```

Full dataset with interaction term confidence intervals. (Generated by simulation):

```
> #Simulated confint for FULL.
> simFull.interaction.coef.df=as.data.frame(simFull.interaction.coef)
> simulationsConfidencesFull.interaction<-data.frame(quantile(simFull.interaction.coef.df$`I(nodegr * treat)` , c(0.025,0.975)),
+                                        quantile(simFull.interaction.coef.df$`nodegr` , c(0.025,0.975)),
+                                        quantile(simFull.interaction.coef.df$`treat` , c(0.025,0.975)),
+                                        quantile(simFull.interaction.coef.df$`black` , c(0.025,0.975))
+                                        )
> colnames(simulationsConfidencesFull.interaction)<- c("I(nodegr * treat)", "nodegr", "treat", "black")
> simulationsConfidencesFull.interaction
      I(nodegr * treat)    nodegr    treat     black
2.5%          -5020.683 -2203.161 1082.934 -3591.3410
97.5%           382.344  2124.671 5784.671  -476.3084
```

Logit model for degree holders' summary:

```
> logitModD0 <- glm(formula = u78 ~ treat + re74 + re75 + u74 + u75 + re78 + age + educ + black,
+                  data=logitDegree0, family=binomial(link="logit"))
>
>
>
> #SIMULATIONS DEG 1
> logitSimD1 <-sim(logitModD1)
> logitSimD1.coeff <- coef(logitSimD1)
> logitSimD1.coef.df <- as.data.frame(logitSimD1.coeff)
> #simulated confint for model Full dataset.
> simulationsConfidencesLogD1<-data.frame(quantile(logitSimD1.coef.df$`(Intercept)` , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`treat` , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`re74`  , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`re75`  , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`u74`   , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`u75`   , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`re78`  , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`age`   , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`educ`  , c(0.025,0.975)),
+                                  quantile(logitSimD1.coef.df$`black` , c(0.025,0.975))
+                                  )
>
> colnames(simulationsConfidencesLogD1)<- c("(Intercept)","treat", "re74", "re75", "u74", "u75","re78", "age", "educ","black")
> display(logitModD1)
glm(formula = u78 ~ treat + re74 + re75 + u74 + u75 + re78 +
    age + educ + black, family = binomial(link = "logit"), data = logitDegree1)
            coef.est coef.se
(Intercept)   814.38 48922.02
treat           6.61 10236.26
re74           -0.02     1.10
re75            0.05     4.71
u74          -105.51 19709.91
u75          -151.92 25964.29
re78           -0.62    10.28
age             0.02    41.22
educ           -5.26  3323.79
black        -484.61 10703.30
---
  n = 348, k = 10
  residual deviance = 0.0, null deviance = 435.8 (difference = 435.8)
> simulationsConfidencesLogD1
      (Intercept)     treat      re74      re75      u74      u75      re78      age      educ     black
2.5%   -102874.61 -21877.79 -1.484534 -8.891861 -36778.09 -46187.32 -20.17055 -84.94728 -5791.988 -19540.07
97.5%    95914.16  18985.15  1.981245  9.723815  33744.45  56597.48  14.78462  68.20035  6504.534  18048.19
```

Logit model for non degree holders' summary:

```
> #SIMULATIONS LOGIT DEG 0
> logitSimD0 <-sim(logitModD0)
> logitSimD0.coeff <- coef(logitSimD0)
> logitSimD0.coef.df <- as.data.frame(logitSimD0.coeff)
> #simulated confint for model Full dataset.
> simulationsConfidencesLogD0 <- data.frame(quantile(logitSimD0.coef.df$`(Intercept)` , c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`treat`, c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`re74` , c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`re75` , c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`u74`  , c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`u75`  , c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`re78` , c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`age`  , c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`educ` , c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`black`, c(0.025,0.975)),
+                                     quantile(logitSimD0.coef.df$`re78` , c(0.025,0.975))
+                                     )
>
> colnames(simulationsConfidencesLogD0)<- c("(Intercept)","treat", "re74", "re75", "u74", "u75", "re78", "age", "educ","black",
"u78")
> display(logitModD0)
glm(formula = u78 ~ treat + re74 + re75 + u74 + u75 + re78 +
    age + educ + black, family = binomial(link = "logit"), data = logitDegree0)
            coef.est  coef.se
(Intercept)    -56.21 171215.76
treat            5.65  24171.60
re74             0.00      2.62
re75            -0.01     13.93
u74             50.49  93639.70
u75            -37.95 102302.94
re78            -0.06      9.35
age             -0.68   1390.43
educ             6.06  14701.96
black           10.70  13663.24
---
  n = 97, k = 10
  residual deviance = 0.0, null deviance = 112.8 (difference = 112.8)
> simulationsConfidencesLogD0
      (Intercept)     treat      re74      re75       u74       u75      re78      age      educ     black
2.5%    -260669.3 -35748.13 -5.403027 -23.28069 -166130.5 -185580.6 -16.56064 -2484.880 -26164.80 -25243.48
97.5%    303005.5  49241.88  4.365936  27.44118  169234.9  188612.1  16.14443  2369.505  23244.71  20452.54
            u78
2.5%  -16.56064
97.5%  16.14443
>
```

Bootstrapping with random forests to estimate test error and oob error.

```
> #Random Forests generated with a sample half the size of the observations.
> #Bootstrap aggregation was applied here.
> lalondeTraining <- sample(1:nrow(lalonde),1000, replace = TRUE)
> lalondeTraining.rf <- randomForest(re78 ~ .,data = lalonde , subset = lalondeTraining,importance=TRUE)
> lalondeTraining.rf

Call:
 randomForest(formula = re78 ~ ., data = lalonde, importance = TRUE,      subset = lalondeTraining)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 4

          Mean of squared residuals: 10996936
                    % Var explained: 77.49
> oob.err <- double(12)
> test.err <- double(12)
>
> for(mtry in 1:12)
+ {
+   rf <- randomForest(re78 ~ . , data = lalonde , subset = lalondeTraining, mtry = mtry, ntree=230)
+   oob.err[mtry] = rf$mse[230] #Error of all Trees fitted
+
+   lalondePred <- predict(rf,lalonde[-lalondeTraining,]) #Predictions on Test Set for each Tree
+   test.err[mtry]= with(lalonde[-lalondeTraining,], mean( (re78 - lalondePred)^2)) #Mean Squared Test Error
+   cat(mtry," ") #printing the output to the console
+ }
1  2  3  4  5  6  7  8  9  10  11  12  >
> test.err
 [1] 26107407 21319929 19404067 18847981 18775189 17815322 18920953 17677842 18509240 18462742 17486203 18871707
> oob.err
 [1] 33537294 20364163 14034990 11465287  9314384  8789738  8316608  8145847  8209734  8551547  8314422  8351076
>
> #Plot test error and OOB error to see if they correlate and where error is minimized
>
> matplot(1:mtry , cbind(oob.err,test.err), pch=19 , col=c("red","blue"),type="b",
+         ylab="Mean Squared Error",xlab="Number of Predictors Considered at each Split")
> legend("topright",legend=c("Out of Bag Error","Test Error"),pch=19, col=c("red","blue"))
> importance(lalondeTraining.rf)
```

Variable importance indices for random forests.

```
> importance(lalondeTraining.rf)
          %IncMSE IncNodePurity
age      41.72983    7033316838
educ     33.61209    4406892365
black    22.70113     999246676
hisp     21.10896     640794935
married  20.44267     948317917
nodegr   19.28793     748280943
re74     22.78984    3367475995
re75     35.77767    7576228534
u74      13.18191     653530679
u75      10.69317     668956514
treat    23.79340    1426265859
u78      74.73420   11970063035
>
> #Random Forests with bagging here
> bagNoDegSetVars <- randomForest(re78 ~ treat +re74 +re75+u74+u75 + age + educ + black,
+                              data = Degree0, ntree = 540, importance=TRUE, replace= TRUE)
>
> importance(bagNoDegSetVars)
         %IncMSE IncNodePurity
treat  4.9624954     269130915
re74   6.6149104     665904205
re75   4.2749218     752960335
u74    5.9127404     112143344
u75    1.4369614      90863560
age   -0.9894604     659028211
educ   2.4054621     362939428
black  7.4977398     172805263
> bagDegSetVars <- randomForest(re78 ~ treat +re74 +re75+u74+u75 + age + educ + black,
+                              data = Degree1, ntree = 540, importance=TRUE)
> importance(bagDegSetVars)
         %IncMSE IncNodePurity
treat  0.3797234     402897587
re74   6.9243898     893766497
re75   7.3213137    1825919460
u74    5.3159722     250838748
u75    5.6315646     210166998
age   -1.7834376    1613073158
educ  -1.2155954     803262807
black  3.1152270     359936140
> #Plotted the variable importance
>
> varImpPlot(lalondeTraining.rf, main = "Trained forest Variable Importance Plot")
> varImpPlot(bagNoDegSetVars, main = "nodegr==0 Variable Importance Plot")
> varImpPlot(bagDegSetVars, main = "nodegr==1 Variable Importance Plot")
>
> #Apply LOOCV
>
```

__MSE calculation using LOOCV and 10-FOLD CV to evaluate model efficacy.__

```
> #LOOCV
>
> modelD0 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = Degree0)
> modelD1 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ + black, data = Degree1)
> modelFull = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ + black, data = lalonde)
>
>
> MSE_LOOCVD0 <- NULL
> MSE_LOOCVD1 <- NULL
> MSE_LOOCVFULL <- NULL
>
> for (i in 1:10){
+   modelD0.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = Degree0)
+   modelD1.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = Degree1)
+   modelFULL.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = lalonde)
+   MSE_LOOCVD0[i] <- cv.glm(Degree0 ,modelD0.loop)$delta[1]
+   MSE_LOOCVD1[i] <- cv.glm(Degree1 ,modelD1.loop)$delta[1]
+   MSE_LOOCVFULL[i] <- cv.glm(lalonde ,modelFULL.loop)$delta[1]
+ }
> #MSE D0
> MSE_LOOCVD0
 [1] 5.966524e+07 4.764272e+07 4.887511e+07 5.010982e+07 6.842141e+09 5.550075e+10 8.534327e+12 1.277024e+14
 [9] 5.941772e+13 1.594283e+18
> #MSE D1
> MSE_LOOCVD1
 [1] 40038018 40027987 40211836 40484347 40409095 41095884 41261812 47198711 55601508 75388280
> #MSE FULL
> MSE_LOOCVFULL
 [1]  44115390  44675308  45166390  45516868  47182142  47885775  57850401  54888432 114685153 208270559
>
```

```
> #K-Fold: Performed a 10 fold cross validation.
>
> modelD0 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = Degree0)
> modelD1 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = Degree1)
> modelFull = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = lalonde)
>
>
> MSE_10F_D0 <- NULL
> MSE_10F_D1 <- NULL
> MSE_10F_FULL <- NULL
>
> for (i in 1:10){
+   modelD0.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = Degree0)
+   modelD1.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = Degree1)
+   modelFULL.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = lalonde)
+   MSE_10F_D0[i] <- cv.glm(Degree0 ,modelD0.loop, K = 10)$delta[1]
+   MSE_10F_D1[i] <- cv.glm(Degree1 ,modelD1.loop, K = 10)$delta[1]
+   MSE_10F_FULL[i] <- cv.glm(lalonde ,modelFULL.loop, K = 12)$delta[1]
+ }
> #MSE D0
> MSE_10F_D0
 [1] 6.413358e+07 4.768488e+07 4.705209e+07 1.100387e+10 3.913095e+08 2.905566e+10 9.074421e+12 1.082694e+14
 [9] 2.093434e+13 2.085975e+18
> #MSE D1
> MSE_10F_D1
 [1]  40003729  39914620  40052595  40624994  40823613  40838170  41896209 379238530  53474974  81976598
> #MSE FULL
> MSE_10F_FULL
 [1]  43987604  44706210  45252990  45579542  47036380  46796936  58057358  54387081  52501620 233106514
>
```

CODE ALMIGHTY!

GIST LINK: https://gist.github.com/GitWahome/d3c8241222b2460dac835d8eab1838df

Pasted in here in case the link fails.

```
################################################################################################
##############################

"Loaded the Matching library and the lalonde dataset"

library(Matching)

library(arm)

#?lalonde

data(lalonde)


################################################################################################
#################################

################################################################################################
################################

#QUESTION 1

################################################################################################
#################################

################################################################################################
################################

#Stratify the data wrt degree

Degree1 = lalonde[lalonde$nodegr == 1,]

Degree0 = lalonde[lalonde$nodegr == 0,]


################################################################################################
###############################

#Regression for re78  vs treat, educ, age, re74, u74, u75 and re75: DEGREE == 0

modelD1 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ + black, data = Degree1)


#Details of statistical significance: For Degree == 1
```

```
summary(modelD1)

display(modelD1)

#SIMULATIONS DEG == 0


simD1 <- sim(modelD1)

simD1.coef <- coef(simD1)


#Simulated confint for model Deg == 1.

simD1.coef.df <- as.data.frame(simD1.coef)

simulationsConfidencesD1<-data.frame(quantile(simD1.coef.df$`(Intercept)` , c(0.025, 0.975)),

                          quantile(simD1.coef.df$`treat` , c(0.025, 0.975)),

                          quantile(simD1.coef.df$`re74` , c(0.025, 0.975)),

                          quantile(simD1.coef.df$`re75` , c(0.025, 0.975)),

                          quantile(simD1.coef.df$`u74` , c(0.025, 0.975)),

                          quantile(simD1.coef.df$`u75` , c(0.025, 0.975)),

                          quantile(simD1.coef.df$`age` , c(0.025, 0.975)),

                          quantile(simD1.coef.df$`educ` , c(0.025, 0.975)),

                          quantile(simD1.coef.df$`black` , c(0.025, 0.975))

)

colnames(simulationsConfidencesD1) <- c("(Intercept)","treat", "re74", "re75", "u74", "u75", "age", "educ","black")

simulationsConfidencesD1

###################################################################################################
#################################


# RELEVANT PLOT For #1, Degree 1

library(sjPlot)

plot_model(modelD1, show.loess.ci = T, show.values = T, show.summary = T, title ="re78 for DEGREE ==1")

plot_model(modelD1, show.values = T, show.ci = T, title ="re78 for DEGREE ==1")
```

```
####################################################################################################
#################################

#Regression for re78  vs treat, educ, age, re74, u74, u75 and re75: DEGREE == 0

modelD0 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = Degree0)

#Details of statistical significance: For Degree == 1

summary(modelD0)

display(modelD0)

#SIMULATIONS DEG == 0

simD0 <-sim(modelD0)

simD0.coef <- coef(simD0)

#Simulated confint for model Deg == 1.

simD0.coef.df=as.data.frame(simD0.coef)

simulationsConfidencesD0<-data.frame(quantile(simD0.coef.df$`(Intercept)` , c(0.025, 0.975)),

                    quantile(simD0.coef.df$`treat` , c(0.025, 0.975)),

                    quantile(simD0.coef.df$`re74` , c(0.025, 0.975)),

                    quantile(simD0.coef.df$`re75` , c(0.025, 0.975)),

                    quantile(simD0.coef.df$`u74` , c(0.025, 0.975)),

                    quantile(simD0.coef.df$`u75` , c(0.025, 0.975)),

                    quantile(simD0.coef.df$`age` , c(0.025, 0.975)),

                    quantile(simD0.coef.df$`educ` , c(0.025, 0.975)),

                    quantile(simD0.coef.df$`black` , c(0.025, 0.975))

                    )

colnames(simulationsConfidencesD0)<- c("(Intercept)","treat", "re74", "re75", "u74", "u75", "age", "educ","black")

simulationsConfidencesD0
```

```
#####################################################################################################
##################################


# RELEVANT PLOT For #1, Degree 0

library(sjPlot)

plot_model(modelD0, show.loess.ci = T, show.values = T, show.summary = T, title ="re78 for DEGREE == 0")

plot_model(modelD0, show.values = T, show.ci = T, title ="re78 for DEGREE == 0")


#####################################################################################################
##################################


#####################################################################################################
##################################

#Run the same simulations for the whole data set just to affirm the inferences.


modelFull = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ + black, data = lalonde)


#Details of statistical significance: For Full dataset.

summary(modelFull)

display(modelFull)


#SIMULATIONS Full dataset

simFull <-sim(modelFull)

simFull.coef <- coef(simFull)

simFull.coef.df <- as.data.frame(simFull.coef)

#simulated confint for model Full dataset.

simulationsConfidencesFull<-data.frame(quantile(simFull.coef.df$`(Intercept)` , c(0.025, 0.975)),

                    quantile(simFull.coef.df$`treat` , c(0.025, 0.975)),

                    quantile(simFull.coef.df$`re74` , c(0.025, 0.975)),

                    quantile(simFull.coef.df$`re75` , c(0.025, 0.975)),
```

```
                    quantile(simFull.coef.df$`u74` , c(0.025, 0.975)),

                    quantile(simFull.coef.df$`u75` , c(0.025, 0.975)),

                    quantile(simFull.coef.df$`age` , c(0.025, 0.975)),

                    quantile(simFull.coef.df$`educ` , c(0.025, 0.975)),

                    quantile(simFull.coef.df$`black` , c(0.025, 0.975))

                    )


colnames(simulationsConfidencesFull)<- c("(Intercept)","treat", "re74", "re75", "u74", "u75", "age", "educ","black")

simulationsConfidencesFull

################################################################################################
#################################


# RELEVANT PLOT For #1, FULL

library(sjPlot)

plot_model(modelFull, show.loess.ci = T, show.values = T, show.summary = T, title ="re78 FOR FULL DATASET")

plot_model(modelFull, show.values = T, show.ci = T, title ="re78 FOR FULL DATASET")


################################################################################################
#################################


################################################################################################
#################################

################################################################################################
#################################

#QUESTION 2

################################################################################################
#################################

################################################################################################
#################################

#Regression for re78  vs I(nodegr*treat): FULL

modelFull.interaction = glm(formula = re78 ~ I(nodegr*treat) + treat +nodegr + black, data = lalonde)
```

```
#Details of statistical significance: For Degree == 1

summary(modelFull.interaction)

display(modelFull.interaction)


#SIMULATIONS FULL

simFull.interaction <-sim(modelFull.interaction)

simFull.interaction.coef <- coef(simFull.interaction)


#Simulated confint for FULL.

simFull.interaction.coef.df=as.data.frame(simFull.interaction.coef)

simulationsConfidencesFull.interaction<-data.frame(quantile(simFull.interaction.coef.df$`I(nodegr * treat)` ,
c(0.025,0.975)),

                          quantile(simFull.interaction.coef.df$`nodegr` , c(0.025,0.975)),

                          quantile(simFull.interaction.coef.df$`treat` , c(0.025,0.975)),

                          quantile(simFull.interaction.coef.df$`black` , c(0.025,0.975))

                          )

colnames(simulationsConfidencesFull.interaction)<- c("I(nodegr * treat)", "nodegr", "treat", "black")

simulationsConfidencesFull.interaction

#####################################################################################################
##################################


# RELEVANT PLOT For #2, FULL

library(sjPlot)

plot_model(modelFull.interaction, show.loess.ci = T, show.values = T, show.summary = T, title ="re78 FOR FULL
DATASET with Interacttion")

plot_model(modelFull.interaction, show.values = T, show.ci = T, title ="re78 FOR FULL DATASET with Interaction")


#####################################################################################################
##################################
```

```
################################################################################################
#################################

################################################################################################
#################################

#QUESTION 3

################################################################################################
#################################

################################################################################################
#################################

#Generate u78 based on the re74/u74 and re75/u75 criteria

lalonde$u78[lalonde$re78<=0]<-1

lalonde$u78[lalonde$re78>0]<-0


#Stratify the data wrt degree

logitDegree1 = lalonde[lalonde$nodegr == 1,]

logitDegree0 = lalonde[lalonde$nodegr == 0,]


logitModD1 <- glm(formula = u78 ~ treat + re74 + re75 + u74 + u75 + re78 + age + educ + black,

            data=logitDegree1, family=binomial(link="logit"))

logitModD0 <- glm(formula = u78 ~ treat + re74 + re75 + u74 + u75 + re78 + age + educ + black,

            data=logitDegree0, family=binomial(link="logit"))


#SIMULATIONS DEG 1

logitSimD1 <-sim(logitModD1)

logitSimD1.coeff <- coef(logitSimD1)

logitSimD1.coef.df <- as.data.frame(logitSimD1.coeff)
```

```
#simulated confint for model Full dataset.

simulationsConfidencesLogD1<-data.frame(quantile(logitSimD1.coef.df$`(Intercept)` , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`treat` , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`re74`  , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`re75`  , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`u74`   , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`u75`   , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`re78`  , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`age`   , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`educ`  , c(0.025,0.975)),

                        quantile(logitSimD1.coef.df$`black` , c(0.025,0.975))

                        )


colnames(simulationsConfidencesLogD1)<- c("(Intercept)","treat", "re74", "re75", "u74", "u75","re78", "age",
"educ","black")

display(logitModD1)

simulationsConfidencesLogD1


#SIMULATIONS LOGIT DEG 0

logitSimD0 <-sim(logitModD0)

logitSimD0.coeff <- coef(logitSimD0)

logitSimD0.coef.df <- as.data.frame(logitSimD0.coeff)

#simulated confint for model Full dataset.

simulationsConfidencesLogD0 <- data.frame(quantile(logitSimD0.coef.df$`(Intercept)` , c(0.025,0.975)),

                        quantile(logitSimD0.coef.df$`treat`, c(0.025,0.975)),

                        quantile(logitSimD0.coef.df$`re74` , c(0.025,0.975)),

                        quantile(logitSimD0.coef.df$`re75` , c(0.025,0.975)),

                        quantile(logitSimD0.coef.df$`u74`  , c(0.025,0.975)),

                        quantile(logitSimD0.coef.df$`u75`  , c(0.025,0.975)),
```

```
                    quantile(logitSimD0.coef.df$`re78` , c(0.025,0.975)),

                    quantile(logitSimD0.coef.df$`age`  , c(0.025,0.975)),

                    quantile(logitSimD0.coef.df$`educ` , c(0.025,0.975)),

                    quantile(logitSimD0.coef.df$`black`, c(0.025,0.975)),

                    quantile(logitSimD0.coef.df$`re78` , c(0.025,0.975))

                    )


colnames(simulationsConfidencesLogD0)<- c("(Intercept)","treat", "re74", "re75", "u74", "u75", "re78", "age",
"educ","black", "u78")

display(logitModD0)

simulationsConfidencesLogD0




################################################################################################
#################################

################################################################################################
#################################

#QUESTION 4

################################################################################################
#################################

################################################################################################
#################################

require(randomForest)

set.seed(101)


################################################################################################
#################################

#Random Forests generated with a sample half the size of the observations.

#Bootstrap aggregation was applied here.

lalondeTraining <- sample(1:nrow(lalonde),1000, replace = TRUE)

lalondeTraining.rf <- randomForest(re78 ~ .,data = lalonde , subset = lalondeTraining,importance=TRUE)
```

```
lalondeTraining.rf

oob.err <- double(12)

test.err <- double(12)


for(mtry in 1:12)

{

 rf <- randomForest(re78 ~ . , data = lalonde , subset = lalondeTraining, mtry = mtry, ntree=230)

 oob.err[mtry] = rf$mse[230] #Error of all Trees fitted


 lalondePred <- predict(rf,lalonde[-lalondeTraining,]) #Predictions on Test Set for each Tree

 test.err[mtry]= with(lalonde[-lalondeTraining,], mean( (re78 - lalondePred)^2)) #Mean Squared Test Error

 cat(mtry," ") #printing the output to the console

}


test.err

oob.err


#Plot test error and OOB error to see if they correlate and where error is minimized


matplot(1:mtry , cbind(oob.err,test.err), pch=19 , col=c("red","blue"),type="b",

     ylab="Mean Squared Error",xlab="Number of Predictors Considered at each Split")

legend("topright",legend=c("Out of Bag Error","Test Error"),pch=19, col=c("red","blue"))

importance(lalondeTraining.rf)


#Random Forests with bagging here

bagNoDegSetVars <- randomForest(re78 ~ treat +re74 +re75+u74+u75 + age + educ + black,

                    data = Degree0, ntree = 540, importance=TRUE, replace= TRUE)
```

```
importance(bagNoDegSetVars)

bagDegSetVars <- randomForest(re78 ~ treat +re74 +re75+u74+u75 + age + educ + black,

                     data = Degree1, ntree = 540, importance=TRUE)

importance(bagDegSetVars)

#Plotted the variable importance


varImpPlot(lalondeTraining.rf, main = "Trained forest Variable Importance Plot")

varImpPlot(bagNoDegSetVars, main = "nodegr==0 Variable Importance Plot")

varImpPlot(bagDegSetVars, main = "nodegr==1 Variable Importance Plot")


#Apply LOOCV


library(ISLR)

library(boot)

set.seed(1)



####################################################################################################
##################################

####################################################################################################
#################################

#LOOCV AND K FOLD

####################################################################################################
##################################

####################################################################################################
##################################

#LOOCV


modelD0 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = Degree0)

modelD1 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ + black, data = Degree1)
```

```
modelFull = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ + black, data = lalonde)


MSE_LOOCVD0 <- NULL
MSE_LOOCVD1 <- NULL
MSE_LOOCVFULL <- NULL

for (i in 1:10){
  modelD0.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = Degree0)
  modelD1.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = Degree1)
  modelFULL.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = lalonde)
  MSE_LOOCVD0[i] <- cv.glm(Degree0 ,modelD0.loop)$delta[1]
  MSE_LOOCVD1[i] <- cv.glm(Degree1 ,modelD1.loop)$delta[1]
  MSE_LOOCVFULL[i] <- cv.glm(lalonde ,modelFULL.loop)$delta[1]
}
#MSE D0
MSE_LOOCVD0
#MSE D1
MSE_LOOCVD1
#MSE FULL
MSE_LOOCVFULL

#K-Fold: Performed a 10 fold cross validation.

modelD0 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = Degree0)
modelD1 = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = Degree1)
modelFull = glm(formula = re78 ~ treat +re74 +re75+u74+u75 + age + educ+black, data = lalonde)
```

```
MSE_10F_D0 <- NULL

MSE_10F_D1 <- NULL

MSE_10F_FULL <- NULL


for (i in 1:10){

  modelD0.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = Degree0)

  modelD1.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = Degree1)

  modelFULL.loop <- glm(formula = re78 ~ poly(treat +re74 +re75+u74+u75 + age + educ+black, i), data = lalonde)

  MSE_10F_D0[i] <- cv.glm(Degree0 ,modelD0.loop, K = 10)$delta[1]

  MSE_10F_D1[i] <- cv.glm(Degree1 ,modelD1.loop, K = 10)$delta[1]

  MSE_10F_FULL[i] <- cv.glm(lalonde ,modelFULL.loop, K = 12)$delta[1]

}

#MSE D0

MSE_10F_D0

#MSE D1

MSE_10F_D1

#MSE FULL

MSE_10F_FULL
```