ARE 213 Problem Set 1B

Becky Cardinali, Yuen Ho, Sara Johns, and Jacob Lefler

Due 10/12/2020

Question 1

In Problem Set 1a, you used linear regression to relate infant health outcomes and maternal smoking during pregnancy. Please answer the following questions.

(a) Under the assumption of random assignment conditional on the observables, what are the sources of misspecification bias in the estimates generated by the linear model estimated in Problem Set 1a?

Even if our assumption of "selection on observables" holds, our estimates from Pset1a may be biased if $E[D|X]$ is not linear in X. That is our estimates from Pset1a impose a linear functional form on the relationship between Y and X, or between $E[D|X]$ and X, causing our estimates to be biased if the true relationship is not linear. To address the potential misspecification bias, we can instead use a nonparametric method to control for X, or estimate $E[D|X]$ nonparametrically.

(b) Consider a series estimator. Estimate the smoking effects using a flexible functional form for the control variables (e.g., higher order terms and interactions). What are the benefits and drawbacks to this approach?

Recall from Pset1a that our list of predtermined variables, which for convenience we'll call X, were state of residence, age of mother, hispanic origin of mother, race of mother, educ of mother, total birth order, interval since last live birth, age of father, hispanic origin of father, month of birth, plurality, previous infant 4000 or more grams, previous preterm infant, prior birth, marital status of mother, and educ of father.

In our series estimator, in addition to X above, to allow for a more flexible functional form we also include higher order terms of mother's and father's age and education. We also include interaction terms between mother's race and mother's age and splines for mother's age with knots at 18 and 35 (because pregnant women under the age of 18 or above the age of 35 may have higher health risks with pregnancy) and splines for mother's education with a knot at 12, for highschool completion. Specifically, we estimate the model:

$$birthweight_i = \alpha_i + \beta smoking_i + \sum_{j=1}^{J} \delta_j X_{ji} + \gamma_1 mother\_age_i^2 + \gamma_2 mother\_age_i^3 + \gamma_3 mother\_educ_i^2 + \gamma_4 mother\_educ_i^3 +$$

$$\gamma_5 father\_age_i^2 + \gamma_6 father\_age_i^3 + \gamma_7 father\_educ_i^2 + \gamma_8 father\_educ_i^3 + \gamma_9 mother\_race_i X mother\_age_i +$$

$$\gamma_{10} mother\_race_i X mother\_age_i^2 + \gamma_{11} I(mother\_age_i > 18)(mother\_age_i - 18)^3 +$$

$$\gamma_{12} I(mother\_age_i > 35)(mother\_age_i - 35)^3 + \gamma_{13} I(mother\_educ_i >= 12)(mother\_educ_i - 12)^3 + \epsilon_i$$

Where $X_j = (X_1, ...X_J)$ represents the J pre-determined variables listed above.

```r
# Generate higher order terms
mom_dt[,c("dmage2", "dmage3", "dfage2", "dfage3", "dmeduc2", "dmeduc3", "dfeduc2", "dfeduc3") := .(dmage^2, d

# Generate splines for mother's age above 18 and above 35
mom_dt[dmage >= 18 , dmage_adult := (dmage - 18)^3]
mom_dt[dmage < 18, dmage_adult := 0]
mom_dt[dmage > 35, dmage_ger := (dmage-35)^3]
mom_dt[dmage <= 35, dmage_ger := 0]
#View(mom_dt[,.(dmage, dmage_adult, dmage_ger)])

# Generate splines for mother's education highschool graduate or more
mom_dt[dmeduc >= 12, dmeduc_hs := (dmeduc-12)^3]
mom_dt[dmeduc < 12, dmeduc_hs := 0]
#View(mom_dt[,.(dmeduc, dmeduc_dropout, dmeduc_hs)])

lm1 <- lm(dbrwt ~ tobacco + factor(stresfip) + factor(mrace3) + factor(birmon) + factor(orfath) + factor(ormo

summary(lm1)
```

```
##
## Call:
## lm(formula = dbrwt ~ tobacco + factor(stresfip) + factor(mrace3) +
##     factor(birmon) + factor(orfath) + factor(ormoth) + factor(pre4000) +
##     factor(preterm) + factor(dmar) + dtotord + disllb + dplural +
##     dmage + dmage2 + dmage3 + dmeduc + dmeduc2 + dmeduc3 + dfage +
##     dfage2 + dfage3 + dfeduc + dfeduc2 + dfeduc3 + factor(mrace3) *
##     dmage + factor(mrace3) * dmage2 + dmage_adult + dmage_ger +
##     dmeduc_hs, data = mom_dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3269.7  -301.4    16.6   335.8  2731.5
##
## Coefficients:
##                         Estimate Std. Error  t value Pr(>|t|)
## (Intercept)           -5.450e+03  3.061e+03   -1.781 0.074993 .
## tobacco               -2.267e+02  4.661e+00  -48.640  < 2e-16 ***
## factor(stresfip)4      1.155e+02  4.011e+02    0.288 0.773410
## factor(stresfip)6      2.314e+02  2.006e+02    1.154 0.248685
## factor(stresfip)8     -2.496e+01  4.011e+02   -0.062 0.950388
## factor(stresfip)9     -5.691e+01  2.975e+02   -0.191 0.848307
## factor(stresfip)10     1.342e+02  1.316e+02    1.020 0.307886
## factor(stresfip)11     1.050e+02  4.013e+02    0.262 0.793666
## factor(stresfip)12    -2.291e+01  1.651e+02   -0.139 0.889604
## factor(stresfip)13     1.060e+02  2.198e+02    0.482 0.629648
## factor(stresfip)17     2.484e+02  2.721e+02    0.913 0.361353
## factor(stresfip)19     1.408e+01  4.012e+02    0.035 0.971999
## factor(stresfip)21     5.472e+02  2.721e+02    2.011 0.044319 *
## factor(stresfip)23    -2.630e+02  5.531e+02   -0.475 0.634437
## factor(stresfip)24     1.770e+02  1.329e+02    1.332 0.182859
## factor(stresfip)25     3.089e+02  2.721e+02    1.135 0.256330
## factor(stresfip)26     2.060e+02  3.357e+02    0.614 0.539458
## factor(stresfip)27     5.337e+02  5.530e+02    0.965 0.334502
## factor(stresfip)29     8.072e+02  5.538e+02    1.458 0.144915
## factor(stresfip)31     3.602e+02  5.529e+02    0.652 0.514724
## factor(stresfip)32     1.065e+02  5.530e+02    0.193 0.847258
## factor(stresfip)34     1.298e+02  1.275e+02    1.018 0.308560
## factor(stresfip)36     1.195e+02  1.359e+02    0.879 0.379430
## factor(stresfip)37      2.181e+02  2.006e+02    1.087 0.277049    
## factor(stresfip)38     -2.817e+02  5.529e+02   -0.509 0.610461    
## factor(stresfip)39      8.793e+01  1.302e+02    0.675 0.499422    
## factor(stresfip)40      1.699e+02  4.012e+02    0.424 0.671906    
## factor(stresfip)42      1.711e+02  1.269e+02    1.348 0.177540    
## factor(stresfip)44     -1.721e+02  5.529e+02   -0.311 0.755584    
## factor(stresfip)45     -5.243e+02  2.975e+02   -1.762 0.078042 .  
## factor(stresfip)46      5.849e+01  5.529e+02    0.106 0.915759    
## factor(stresfip)47      1.846e+02  2.537e+02    0.728 0.466813    
## factor(stresfip)51     -5.129e+00  1.729e+02   -0.030 0.976336    
## factor(stresfip)53     -1.440e+02  3.356e+02   -0.429 0.667838    
## factor(stresfip)54      9.878e+01  1.405e+02    0.703 0.481869    
## factor(stresfip)55      7.666e+01  2.975e+02    0.258 0.796678    
## factor(mrace3)2         1.497e+01  2.763e+02    0.054 0.956803    
## factor(mrace3)3        -1.229e+02  8.504e+01   -1.446 0.148316    
## factor(birmon)2        -8.113e+00  7.988e+00   -1.016 0.309815    
## factor(birmon)3        -1.646e+00  7.748e+00   -0.212 0.831780    
## factor(birmon)4         3.003e+00  7.811e+00    0.384 0.700636    
## factor(birmon)5        -4.204e+00  7.747e+00   -0.543 0.587397    
## factor(birmon)6        -1.127e+01  7.816e+00   -1.442 0.149416    
## factor(birmon)7        -1.144e+01  7.710e+00   -1.484 0.137766    
## factor(birmon)8        -3.148e+00  7.727e+00   -0.407 0.683727    
## factor(birmon)9        -5.500e+00  7.750e+00   -0.710 0.477892    
## factor(birmon)10       -1.621e+01  7.834e+00   -2.069 0.038561 *  
## factor(birmon)11       -1.114e+01  8.000e+00   -1.393 0.163671    
## factor(birmon)12       -1.348e+01  7.952e+00   -1.695 0.090160 .  
## factor(orfath)1        -4.799e+01  2.958e+01   -1.623 0.104678    
## factor(orfath)2        -3.043e+01  1.441e+01   -2.111 0.034746 *  
## factor(orfath)3         4.654e+01  6.165e+01    0.755 0.450264    
## factor(orfath)4        -4.698e+01  2.828e+01   -1.661 0.096697 .  
## factor(orfath)5        -6.640e+01  2.719e+01   -2.442 0.014593 *  
## factor(ormoth)1         2.084e+00  3.364e+01    0.062 0.950607    
## factor(ormoth)2        -1.319e+02  1.478e+01   -8.924  < 2e-16 ***
## factor(ormoth)3        -5.446e+00  6.046e+01   -0.090 0.928222    
## factor(ormoth)4        -2.763e+01  3.002e+01   -0.920 0.357464    
## factor(ormoth)5        -9.884e+01  2.684e+01   -3.683 0.000231 ***
## factor(pre4000)2       -4.514e+02  1.342e+01  -33.636  < 2e-16 ***
## factor(preterm)2        4.293e+02  1.357e+01   31.629  < 2e-16 ***
## factor(dmar)2          -4.525e+01  4.973e+00   -9.098  < 2e-16 ***
## dtotord                 3.970e+00  1.420e+00    2.796 0.005178 ** 
## disllb                 -1.654e-01  5.653e-03  -29.254  < 2e-16 ***
## dplural                -9.298e+02  9.139e+00 -101.740  < 2e-16 ***
## dmage                   1.604e+03  5.136e+02    3.122 0.001794 ** 
## dmage2                 -8.807e+01  2.871e+01   -3.068 0.002155 ** 
## dmage3                  1.620e+00  5.338e-01    3.035 0.002406 ** 
## dmeduc                  5.375e+01  2.559e+01    2.100 0.035701 *  
## dmeduc2                -7.496e+00  2.719e+00   -2.757 0.005834 ** 
## dmeduc3                 3.090e-01  9.358e-02    3.302 0.000960 ***
## dfage                   5.362e+00  7.541e+00    0.711 0.477028    
## dfage2                 -1.674e-01  2.168e-01   -0.772 0.440009    
## dfage3                  1.705e-03  2.020e-03    0.844 0.398729    
## dfeduc                 -6.676e+01  1.836e+01   -3.636 0.000277 ***
## dfeduc2                 6.383e+00  1.658e+00    3.850 0.000118 ***
## dfeduc3                -1.810e-01  4.821e-02   -3.755 0.000173 ***
## dmage_adult            -1.612e+00  5.376e-01   -2.999 0.002706 ** 
## dmage_ger              -6.147e-02  6.264e-02   -0.981 0.326447    
## dmeduc_hs              -1.205e+00  2.612e-01   -4.613 3.97e-06 ***
## factor(mrace3)2:dmage  -1.954e+01  1.907e+01   -1.025 0.305576    
## factor(mrace3)3:dmage  -7.506e+00  6.591e+00   -1.139 0.254807    
## factor(mrace3)2:dmage2  3.818e-01  3.242e-01    1.178 0.238863
## factor(mrace3)3:dmage2  1.362e-01  1.231e-01    1.106 0.268516
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 538.1 on 114526 degrees of freedom
## Multiple R-squared:  0.1549, Adjusted R-squared:  0.1543
## F-statistic:   253 on 83 and 114526 DF,  p-value: < 2.2e-16
```

A benefit to this approach is that it is straightforward and relatively easy to implement and interpret. However, a drawback to this approach is that the choice of which covariates, higher order terms, interaction terms, and knots to include is arbitrary, so our series estimator may not accurately capture the true data generating process and we run the risk of overfitting our model or omitting an important control variable.

(c) Use the LASSO to determine which covariates (and higher order terms) to include in your regression from part (b). Do you end up dropping some covariates that you had thought might be necessary to include?

We follow the procedure suggested by Belloni, Chernozhukov, and Hansen by first applying the LASSO to the equation in part (b). We also apply the LASSO to the model regressing treatment status on the remaining covariates:

$$smoking_i = \pi_0 + \sum_{j=1}^{J} \pi_j X_{ji} + \nu_1 mother\_age_i^2 + \nu_2 mother\_age_i^3 + \nu_3 mother\_educ_i^2 + \nu_4 mother\_educ_i^3 +$$

$$\nu_5 father\_age_i^2 + \nu_6 father\_age_i^3 + \nu_7 father\_educ_i^2 + \nu_8 father\_educ_i^3 + \nu_9 mother\_race_i X mother\_age_i +$$

$$\nu_{10} mother\_race_i X mother\_age_i^2 + \nu_{11} I(mother\_age_i > 18)(mother\_age_i - 18)^3 +$$

$$\nu_{12} I(mother\_age_i > 35)(mother\_age_i - 35)^3 + \nu_{13} I(mother\_educ_i >= 12)(mother\_educ_i - 12)^3 + \epsilon_i$$

Then we regress $birthweight_i$ on $smoking_i$ and all of the covariates selected by LASSO in either equation 1 or 2 above. In each case, we cross-validate the LASSO and choose lambda such that the mean cross-validated error is within 1 standard error of the minimum.

```r
#First convert factor variables in dataset to factor variables
mom_dt[,stresfip := as.factor(stresfip)]
mom_dt[,cntocpop := as.factor(cntocpop)]
mom_dt[,mrace3 := as.factor(mrace3)]
mom_dt[,dmar := as.factor(dmar)]
mom_dt[,birmon := as.factor(birmon)]
mom_dt[,orfath := as.factor(orfath)]
mom_dt[,ormoth := as.factor(ormoth)]
mom_dt[,pre4000 := as.factor(pre4000)]
mom_dt[,preterm := as.factor(preterm)]

#Lasso for lm1
x1 <- model.matrix(lm1, data = mom_dt) #create X matrix
lasso_y <- cv.glmnet(x1, mom_dt$dbrwt, family = "gaussian", standardize = TRUE, intercept = TRUE, alpha = 1,
#plot(lasso_y)
#lasso_y$glmnet.fit
coef(lasso_y, s = "lambda.1se") #Coefficients from lasso for lambda that gets error within 1 se of the minimu
```

```
## 85 x 1 sparse Matrix of class "dgCMatrix"
##                                    1
## (Intercept)             4.361765e+03
## (Intercept)             .
## tobacco                -1.909135e+02
## factor(stresfip)4       .
## factor(stresfip)6       .
## factor(stresfip)8       .
## factor(stresfip)9       .
## factor(stresfip)10      .
## factor(stresfip)11      .
## factor(stresfip)12      .
## factor(stresfip)13      .
## factor(stresfip)17      .
## factor(stresfip)19      .
## factor(stresfip)21      .
## factor(stresfip)23      .
## factor(stresfip)24      .
## factor(stresfip)25      .
## factor(stresfip)26      .
## factor(stresfip)27      .
## factor(stresfip)29      .
## factor(stresfip)31      .
## factor(stresfip)32      .
## factor(stresfip)34      .
## factor(stresfip)36      .
## factor(stresfip)37      .
## factor(stresfip)38      .
## factor(stresfip)39      .
## factor(stresfip)40      .
## factor(stresfip)42      .
## factor(stresfip)44      .
## factor(stresfip)45      .
## factor(stresfip)46      .
## factor(stresfip)47      .
## factor(stresfip)51      .
## factor(stresfip)53      .
## factor(stresfip)54      .
## factor(stresfip)55      .
## factor(mrace3)2        -1.138977e+02
## factor(mrace3)3        -1.778087e+02
## factor(birmon)2         .
## factor(birmon)3         .
## factor(birmon)4         .
## factor(birmon)5         .
## factor(birmon)6         .
## factor(birmon)7         .
## factor(birmon)8         .
## factor(birmon)9         .
## factor(birmon)10        .
## factor(birmon)11        .
## factor(birmon)12        .
## factor(orfath)1         .
## factor(orfath)2         .
## factor(orfath)3         .
## factor(orfath)4         .
## factor(orfath)5         .
## factor(ormoth)1         .
## factor(ormoth)2        -7.248294e+01
## factor(ormoth)3         .
## factor(ormoth)4         .
## factor(ormoth)5         .
## factor(pre4000)2       -3.638941e+02
## factor(preterm)2        3.182590e+02
## factor(dmar)2          -6.048703e+01
## dtotord                 .
## disllb                 -1.344746e-01
## dplural                -8.505336e+02
## dmage                   .
## dmage2                  .
## dmage3                  .
## dmeduc                  3.002534e+00
## dmeduc2                 .
## dmeduc3                 .
## dfage                   .
## dfage2                  .
## dfage3                  .
## dfeduc                  6.880839e-01
## dfeduc2                 4.817106e-05
## dfeduc3                 .
## dmage_adult             .
## dmage_ger               .
## dmeduc_hs               .
## factor(mrace3)2:dmage   .
## factor(mrace3)3:dmage   .
## factor(mrace3)2:dmage2  .
## factor(mrace3)3:dmage2  .
```

```
#Lasso for model of treatment status as a function of the other control variables
lm2 <- lm(tobacco ~ factor(stresfip) + factor(mrace3) + factor(birmon) + factor(orfath) + factor(ormoth) + fa

x2 <- model.matrix(lm2, data = mom_dt) #create X matrix
lasso_d <- cv.glmnet(x2, mom_dt$dbrwt, family = "gaussian", standardize = TRUE, intercept = TRUE, alpha = 1,
#plot(lasso_d)
#lasso_d$glmnet.fit
coef(lasso_d, s = "lambda.1se") #Coefficients from lasso for lambda that gets error within 1 se of the minimu
```

```
## 84 x 1 sparse Matrix of class "dgCMatrix"
##                                   1
## (Intercept)             4.266767e+03
## (Intercept)             .
## factor(stresfip)4       .
## factor(stresfip)6       .
## factor(stresfip)8       .
## factor(stresfip)9       .
## factor(stresfip)10      .
## factor(stresfip)11      .
## factor(stresfip)12      .
## factor(stresfip)13      .
## factor(stresfip)17      .
## factor(stresfip)19      .
## factor(stresfip)21      .
## factor(stresfip)23      .
## factor(stresfip)24      .
## factor(stresfip)25      .
## factor(stresfip)26      .
## factor(stresfip)27      .
## factor(stresfip)29      .
## factor(stresfip)31      .
## factor(stresfip)32      .
## factor(stresfip)34      .
## factor(stresfip)36      .
## factor(stresfip)37      .
## factor(stresfip)38      .
## factor(stresfip)39      .
## factor(stresfip)40      .
## factor(stresfip)42      .
## factor(stresfip)44      .
## factor(stresfip)45      .
## factor(stresfip)46      .
## factor(stresfip)47      .
## factor(stresfip)51      .
## factor(stresfip)53      .
## factor(stresfip)54      .
## factor(stresfip)55      .
## factor(mrace3)2        -9.860228e+01
## factor(mrace3)3        -1.498265e+02
## factor(birmon)2         .
## factor(birmon)3         .
## factor(birmon)4         .
## factor(birmon)5         .
## factor(birmon)6         .
## factor(birmon)7         .
## factor(birmon)8         .
## factor(birmon)9         .
## factor(birmon)10        .
## factor(birmon)11        .
## factor(birmon)12        .
## factor(orfath)1         .
## factor(orfath)2         .
## factor(orfath)3         .
## factor(orfath)4         .
## factor(orfath)5         .
## factor(ormoth)1         .
## factor(ormoth)2        -3.504671e+01
## factor(ormoth)3         .
## factor(ormoth)4         .
## factor(ormoth)5         .
## factor(pre4000)2       -3.786934e+02
## factor(preterm)2        3.329088e+02
## factor(dmar)2          -9.673474e+01
## dtotord                 .
## disllb                 -1.201598e-01
## dplural                -8.470217e+02
## dmage                   .
## dmage2                  .
## dmage3                  .
## dmeduc                  6.502188e+00
## dmeduc2                 8.127778e-04
## dmeduc3                 .
## dfage                   .
## dfage2                  .
## dfage3                  .
## dfeduc                  .
## dfeduc2                 1.308466e-01
## dfeduc3                 .
## dmage_adult             .
## dmage_ger               .
## dmeduc_hs               .
## factor(mrace3)2:dmage   .
## factor(mrace3)3:dmage  -1.201014e-01
## factor(mrace3)2:dmage2  .
## factor(mrace3)3:dmage2  .
```

Based on these results, we choose to keep the following covariates: an intercept, tobacco, both levels of mother's race, a dummy for whether the mother is Puerto Rican, a dummy for whether the mother is Hispanic "other", previous infant 4000 grams or more, previous preterm infant, marital status, interval since last live birth, plurality, mother's education and the square of mother's education, father's education and the square of father's education, a spline for mother's age with a knot at 35, and the interaction term between whether the mother is black by mother's age.

We drop a significant number of covariates that we had initially thought would be necessary to include, such as state of residence, birth month, hispanic origin of father, the cube of mother's and father's education, father's age and its higher order terms, mother's age and its higher order terms, and most of the spline and interaction terms.

After our LASSO procedure, our final model is:

$$birthweight_i = \alpha_i + \beta Smoking_i + \delta_1 mother\_race_i + \delta_2 mother\_hispanic_i == 2 + \delta_3 mother\_hispanic_i == 5 + \delta_4 pre4000_i +$$
$$\delta_5 preterm_i + \delta_6 marital\_status_i + \delta_7 last\_birth_i + \delta_8 plurality_i + \delta_9 mother\_educ_i + \delta_{10} mother\_educ_i^2 + \delta_{11} father\_educ_i +$$
$$\delta_{12} father\_educ_i^2 + \delta_{13} I(mother\_age_i > 35)(mother\_age_i - 35)^3 + \delta_{14} mother\_race_i == 3X mother\_age_i + \epsilon_i$$

```
#Create necessary dummy and interaction terms
mom_dt[, ormoth2 := ifelse(ormoth == 2, 1, 0)]
mom_dt[,ormoth2 := as.factor(ormoth2)]
mom_dt[,ormoth5 := ifelse(ormoth == 5, 1, 0)]
mom_dt[,ormoth5 := as.factor(ormoth5)]
mom_dt[,mrace3Xdmage := ifelse(mrace3 == 3, 3*dmage, 0)]

#New model with covariates chosen from 2-stage Lasso above
lm3 <- lm(dbrwt ~ tobacco + factor(mrace3) + factor(ormoth2) + factor(ormoth5) + factor(pre4000) + factor(pre
summary(lm3)
```

```
##
## Call:
## lm(formula = dbrwt ~ tobacco + factor(mrace3) + factor(ormoth2) +
##     factor(ormoth5) + factor(pre4000) + factor(preterm) + factor(dmar) +
##     disllb + dplural + dmeduc + dmeduc2 + dfeduc + dfeduc2 +
##     dmage_ger + mrace3Xdmage, data = mom_dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3255.8  -301.9    16.0   336.7  2755.8
##
## Coefficients:
##                    Estimate Std. Error  t value Pr(>|t|)
## (Intercept)       4.400e+03  4.679e+01   94.042  < 2e-16 ***
## tobacco          -2.249e+02  4.629e+00  -48.581  < 2e-16 ***
## factor(mrace3)2  -2.180e+02  1.154e+01  -18.898  < 2e-16 ***
## factor(mrace3)3  -2.463e+02  2.051e+01  -12.010  < 2e-16 ***
## factor(ormoth2)1 -1.582e+02  1.078e+01  -14.681  < 2e-16 ***
## factor(ormoth5)1 -1.405e+02  2.313e+01   -6.073 1.26e-09 ***
## factor(pre4000)2 -4.526e+02  1.340e+01  -33.779  < 2e-16 ***
## factor(preterm)2  4.281e+02  1.356e+01   31.568  < 2e-16 ***
## factor(dmar)2    -5.342e+01  4.547e+00  -11.748  < 2e-16 ***
## disllb           -1.753e-01  4.545e-03  -38.579  < 2e-16 ***
## dplural          -9.299e+02  9.139e+00 -101.744  < 2e-16 ***
## dmeduc            1.061e+01  6.551e+00    1.620    0.105
## dmeduc2          -2.130e-01  2.469e-01   -0.863    0.388
## dfeduc           -3.598e+00  6.234e+00   -0.577    0.564
## dfeduc2           2.528e-01  2.336e-01    1.082    0.279
## dmage_ger        -1.928e-01  3.544e-02   -5.440 5.33e-08 ***
## mrace3Xdmage      3.619e-01  2.630e-01    1.376    0.169
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 538.5 on 114593 degrees of freedom
## Multiple R-squared:  0.1533, Adjusted R-squared:  0.1532
## F-statistic:  1297 on 16 and 114593 DF,  p-value: < 2.2e-16
```

Question 2

Describe the propensity score approach to the problem of estimating the average causal effect of smoking when the treatment is randomly assigned conditional on the observables. How does it reduce the dimensionality problem of multivariate matching?

We know that if we condition on observables, we will get a consistent estimate of the ATE under the selection on observables assumption. However, if the observables are high dimensional, it might be difficult to find a comparison unit with the same values of the observables. From lecture, we know that it is sufficient instead to condition on the propensity score. Using the propensity score allows us to compare treated and control units with the same probability of being treated, controlling for any differences that are consistently related to the probability of treatment. The propensity score does not require that all values of the observables be the same and so avoids problems of multidimensionality.

Try a few ways to estimate the effects of maternal smoking on birthweight:

(a) First create the propensity score. For our purposes let's use a logit specification. First specify the logit using all of the "predetermined" covariates (don't include interactions). Next, include only those "predetermined" covariates that enter significantly in the first logit specification. How comparable are the propensity scores? If they are similar does this imply that we have the "correct" set of covariates in the logit specification used for our propensity score?

```r
# get prop score using all predetermined variables
prop_all <- glm(tobacco ~ factor(stresfip) + dmage + factor(mrace3) + dmeduc +
                  dtotord + disllb + dfage + factor(birmon) + factor(orfath) + factor(ormoth) +
                  factor(dmar) + dfeduc + dplural + factor(pre4000) + factor(preterm),
                family=binomial(link='logit'), data = mom_dt)
mom_dt[, prop_score_all := fitted(prop_all)]

# take a look at the output to see which are significant - omitting because takes up a lot of space
# summary(prop_all)
# only need to take out state of residence. birth month has a few months that are significant so will keep

# get prop score using significant variables from previous logit
prop_sig <- glm(tobacco ~ dmage + factor(mrace3) + dmeduc +
                  dtotord + disllb + dfage + factor(birmon) + factor(orfath) + factor(ormoth) +
                  factor(dmar) + dfeduc + dplural + factor(pre4000) + factor(preterm),
                family=binomial(link='logit'), data = mom_dt)
mom_dt[, prop_score_sig := fitted(prop_sig)]

# how different are the prop scores?
prop_score_diff <- mom_dt$prop_score_all - mom_dt$prop_score_sig
summary(prop_score_diff)
```

```
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -0.2918444  0.0003145  0.0005001  0.0000000  0.0007028  0.2901933
```

```r
# a few outliers but not very different
```

In the first logit, we include state of residence, mother's age, mother's race, mother's hispanic origin, mother's education, birth order, interval since last birth, father's age, father's Hispanic origin, father's education, birth month, plurality, previous heavy child, and previous preterm. This is a combination of the "predetermined" variables we selected and discussed in the last problem set and the ones discussed in the problem set solutions. We find that all coefficients are significant, except for state of residence. We remove that from the next regression and the propensity scores, which are the fitted values of the logit, do not change very much. This does not guarantee we are estimating the logit with the correct set of covariates. There could be important covariates that we do not have in this dataset that we would want to use in the logit specification. The fact that our propensity scores do not change much between the two specifications just gives us confidence that state of residence was not an important regressor to include. Moving forward, we will use the propensity scores from the second specification.

(b) Control directly for the estimated propensity scores using a regression analysis, and estimate an average treatment effect. State clearly the assumptions under which your estimate is correct.

```
# run regression with prop_score
prop_reg <- lm(dbrwt ~ tobacco + prop_score_sig, data = mom_dt)
```

Controlling directly for the propensity score, we get the ATE is -225.1 and is statistically significant at the 99.9% level. The assumption under which this estimate is consistent is unconfoundedness and homogeneous treatment effects. If we instead believed that there were heterogeneous treatment effects (that varied with X), then this regression does not provide meaningful results.

(c) As discussed in class, one can use the estimated propensity scores to reweight the outcomes of non-smokers and estimate the average treatment effect. Compute an estimate of the average treatment effect and the "effect of the treatment on the treated" by appropriate reweighting of the data.

```
# create propensity weights
mom_dt[,prop_weights := ifelse(tobacco == 1,
                          1/prop_score_sig,
                          1/(1 - prop_score_sig))]
# normalize the weights
mom_dt[,norm_prop_weights := ifelse(tobacco == 1,
                               prop_weights/sum(mom_dt[tobacco == 1, prop_weights]),
                               prop_weights/sum(mom_dt[tobacco == 0, prop_weights]))]
# estimate ATE
tau_ipw <- sum((mom_dt$tobacco*mom_dt$dbrwt)*(mom_dt$norm_prop_weights) -
                 ((1 - mom_dt$tobacco)*mom_dt$dbrwt)*(mom_dt$norm_prop_weights))

# estimate TOT - use formula from section
tot_y1 <- sum(mom_dt$tobacco*mom_dt$dbrwt) / sum(mom_dt$tobacco)
tot_y0 <- sum(mom_dt$prop_score_sig * (1 - mom_dt$tobacco) * mom_dt$dbrwt /
        (1 - mom_dt$prop_score_sig)) /
  sum(mom_dt$prop_score_sig * (1 - mom_dt$tobacco) /
        (1 - mom_dt$prop_score_sig))

tau_tot <- tot_y1 - tot_y0
```

We have from lecture that the ATE using inverse propensity weighting is

$$\hat{\tau}_{ATE} = \Bigg(\sum^{N} \frac{D_i Y_i}{\hat{p}(X_i)} / \sum^{N} \frac{D_i}{\hat{p}(X_i)}) - (\sum^{N} \frac{(1 - D_i) Y_i}{1 - \hat{p}(X_i)} / \sum^{N} \frac{1 - D_i}{1 - \hat{p}(X_i)} \Bigg)$$

and using this formula, we get that the ATE is -224.87. We have from section that the TOT using inverse propensity weighting is

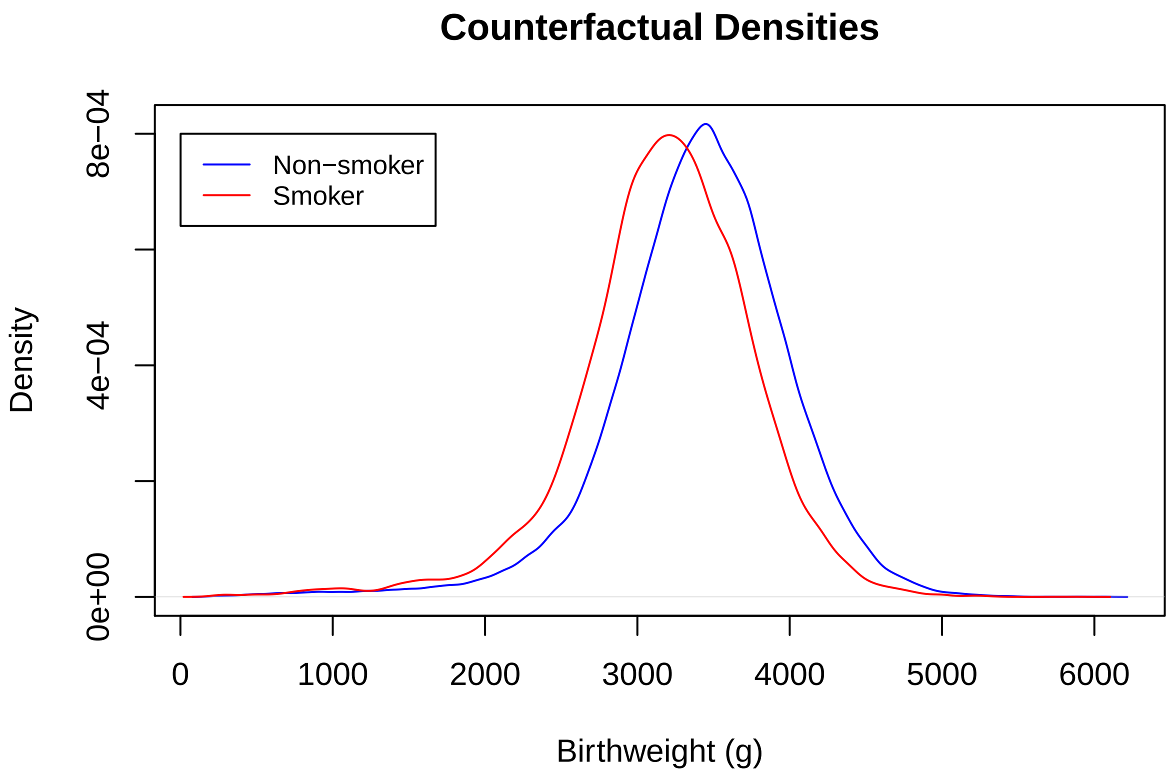
$$\hat{\tau}_{TOT} = \left(\sum^{N} D_i Y_i / \sum^{N} D_i - (\sum^{N} \frac{\hat{p}(X_i)(1-D_i)Y_i}{1-\hat{p}(X_i)} / \sum^{N} \frac{\hat{p}(X_i)(1-D_i)}{1-\hat{p}(X_i)} \right)$$

and using this formula, we get that the TOT is -226.59.

(d) Estimate the counterfactual densities relevant for the above part with a kernel density estimator. That is, estimate the density of birthweight (or log birthweight) if everyone smoked and again if no one smoked. Hint: Consider directly applying the Hirano, Imbens, and Ridder propensity score reweighting scheme in the context of estimating the densities of the treated and control groups (rather than the means of the treated and control groups). Stata has very useful preprogrammed commands. In addition to using the preprogrammed Stata command to compute/graph the kernel density over the entire range of birthweight, please also calculate by hand the kernel estimator at birthweight equals 3,000 grams (and provide the code you wrote that shows the calculation of the kernel estimator at this single point). Play around with a bandwidth starting with half the default Stata bandwidth. Choose the same bandwidth for all the pictures, and produce a (beautiful, production quality) figure depicting both densities.

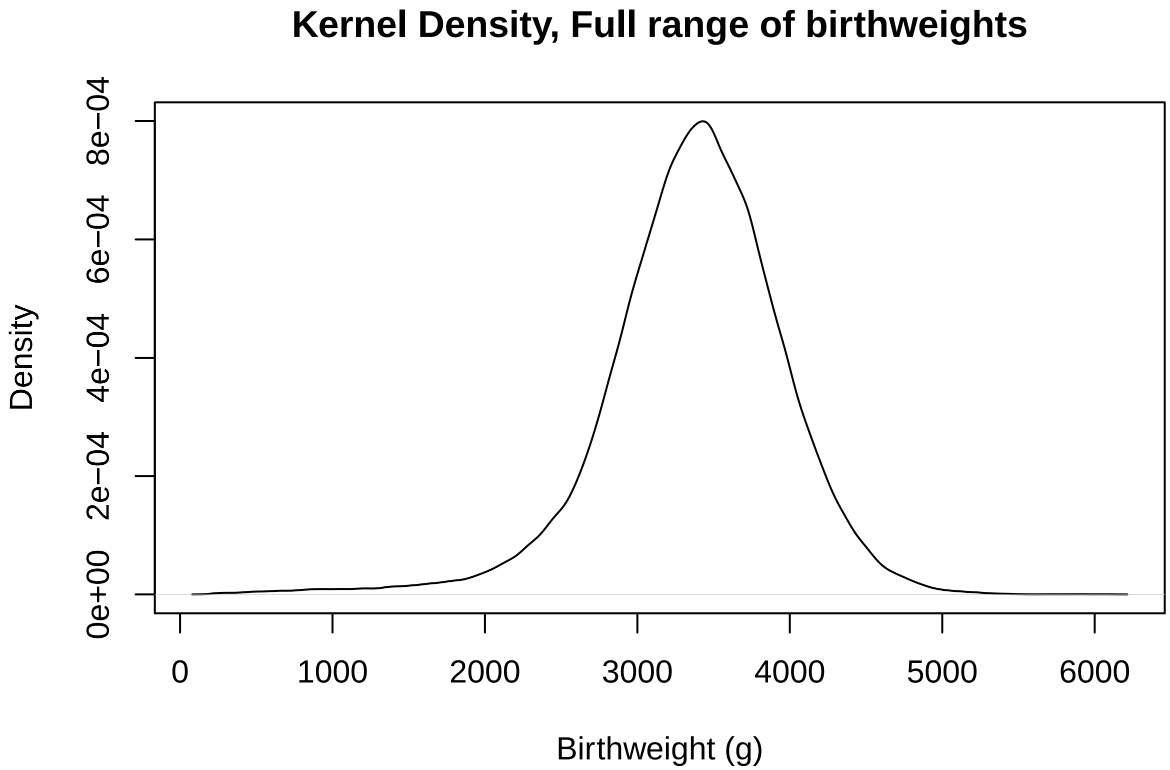
```
d0 <- density(mom_dt[tobacco==0,dbrwt], kernel="gaussian", bw="sj",
              adjust=1, weights=mom_dt[tobacco==0,norm_prop_weights])
d1 <- density(mom_dt[tobacco==1,dbrwt], kernel="gaussian", bw="sj",
              adjust=1, weights=mom_dt[tobacco==1,norm_prop_weights])

plot(d0, col="blue", main="Counterfactual Densities",
     xlab = "Birthweight (g)")
lines(d1, col="red")
legend(1, 0.0008, legend=c("Non-smoker", "Smoker"),
       col=c("blue","red"), lty=1, cex=0.8)
```



In order to get the counterfactual densities, we weight the treated and control groups by the normalized inverse propensity scores, similar to our calculation above. We use the density() function with a Gaussian kernel and a bandwidth using the methods of Sheather & Jones (1991) which uses pilot estimation of derivatives. This results in a bandwidth of 49.83 for the control and a bandwidth of 68.9 for the treated. As we expect from our estimates of the ATE, the density of the birthweights for the treated group is shifted to the left of the control group (lower birthweights).

```r
d_all <- density(mom_dt$dbrwt, kernel="gaussian", bw="sj", adjust=1)
plot(d_all, col="black", main="Kernel Density, Full range of birthweights",
     xlab = "Birthweight (g)")
```



```r
# kernel estimator at 3000 g
# use bandwidth selected above: h = 49
h <- d_all$bw
ke_3000 <- 1 / (nrow(mom_dt) * h * sqrt(2*pi)) * sum(exp(-0.5 * (((3000-mom_dt$dbrwt)/h)^2 )))
```

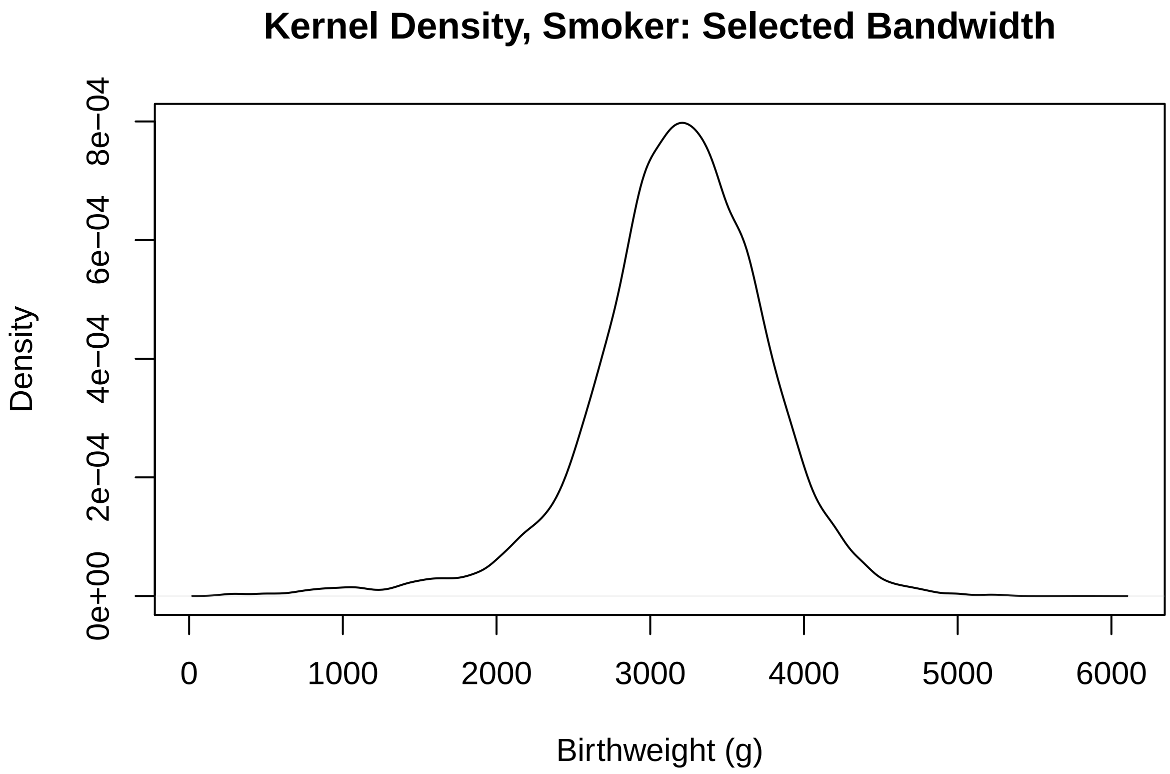
We plot the kernel density over the entire range of birthweights, using the Gaussian kernel and selecting the bandwidth using the methods of Sheather & Jones (1991) which uses pilot estimation of derivatives. For the entire range of birthweights, this gives us a bandwidth of 49.05. We then calculate the kernel estimator by hand at a weight of 3000 grams using the Gaussian kernel and this bandwidth. Our formula is

$$\hat{f}(x) = \frac{1}{nh}\frac{1}{\sqrt{2\pi}}\sum_{i}^{n} e^{-1/2(\frac{x-x_i}{h})^2}$$

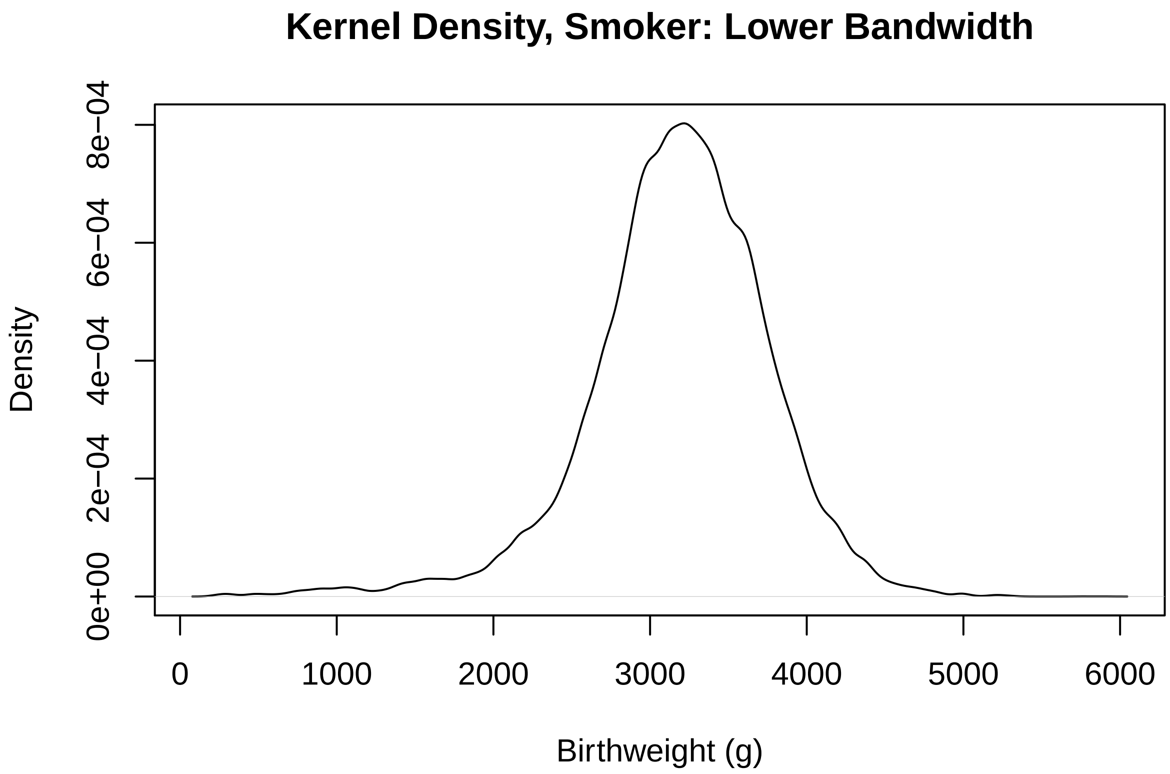
where $x = 3,000$, $n = 114610$, and $h = 49.05$. We get that the density at 3000g is 5.4×10^{-4}, which visually corresponds to our plot of the density.

(e) Take one of your densities and display an estimate of the density using different bandwidths as well as the one you settled on. What happens with bigger (smaller) bandwidths?

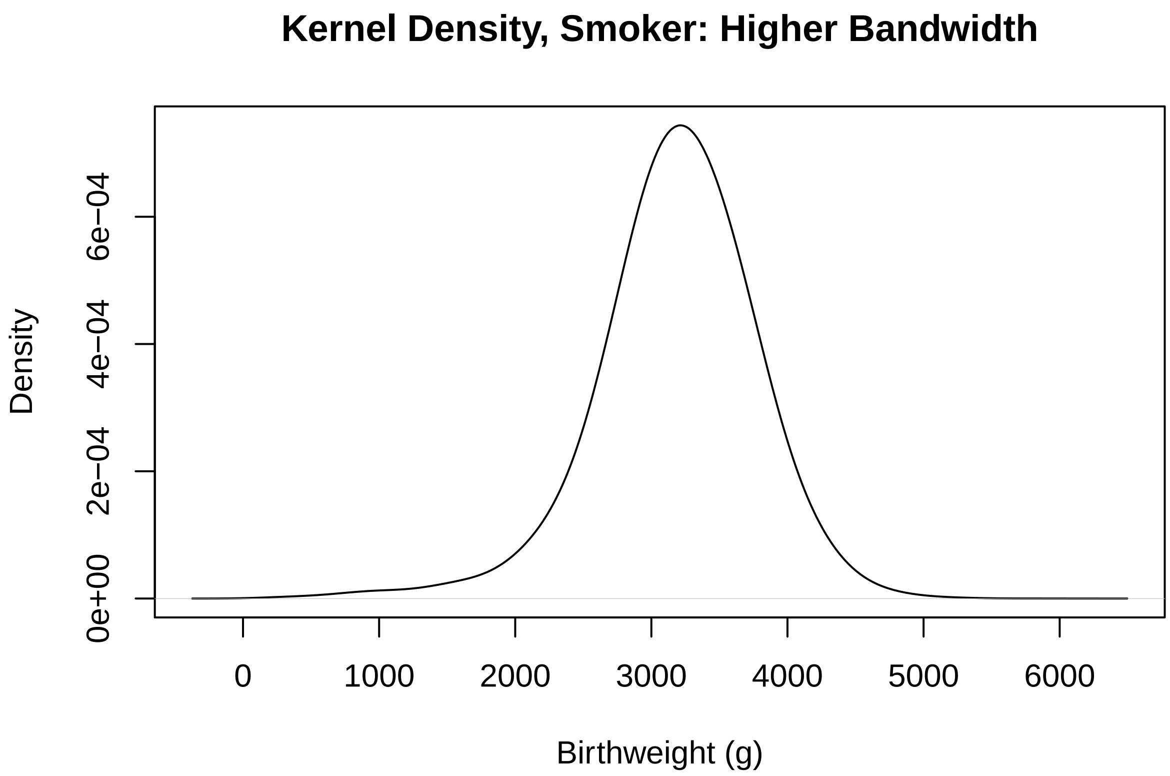
```
# plot bandwidth selected
plot(d1, col="black", main="Kernel Density, Smoker: Selected Bandwidth",
     xlab = "Birthweight (g)")
```



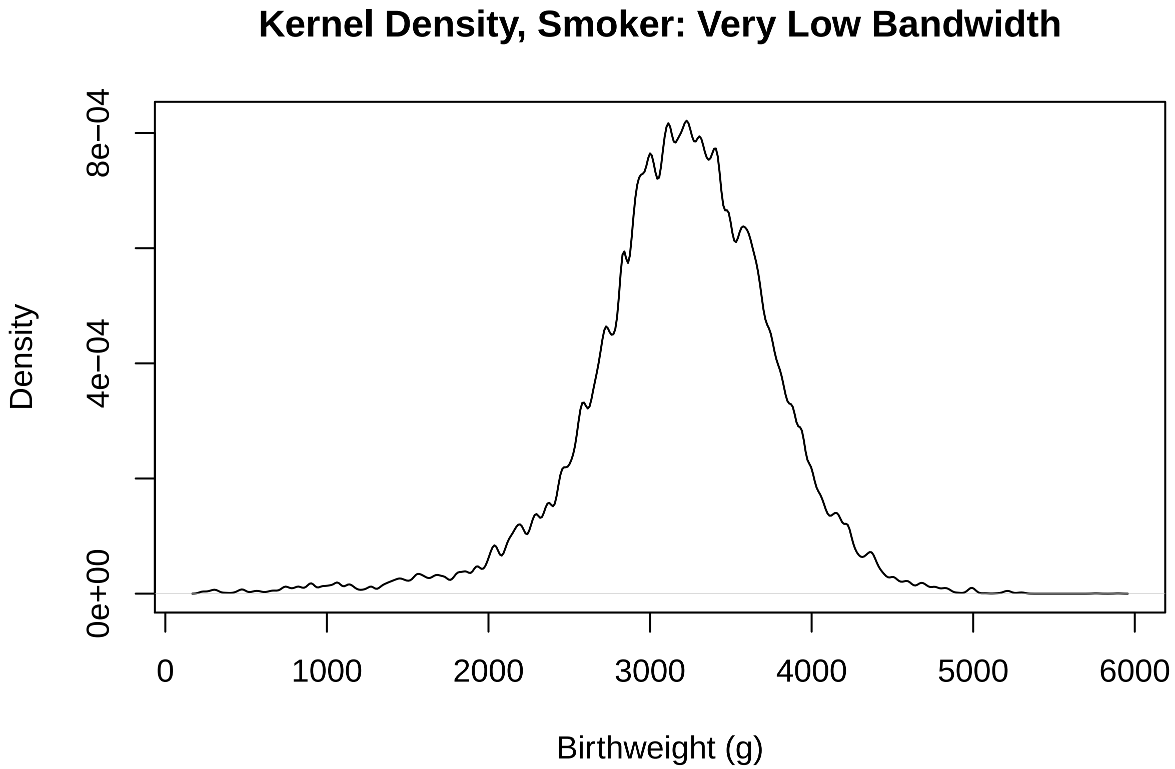
```
# bw of treated group is 68.9, what if we used 49.8 like control?
d1_low <- density(mom_dt[tobacco==1,dbrwt], kernel="gaussian", bw=49.8, adjust=1,
                  weights=mom_dt[tobacco==1,norm_prop_weights])
plot(d1_low, col="black", main="Kernel Density, Smoker: Lower Bandwidth",
     xlab = "Birthweight (g)")
```



```
# what if we raised it?
d1_high <- density(mom_dt[tobacco==1,dbrwt], kernel="gaussian", bw=200, adjust=1,
                   weights=mom_dt[tobacco==1,norm_prop_weights])
plot(d1_high, col="black", main="Kernel Density, Smoker: Higher Bandwidth",
     xlab = "Birthweight (g)")
```



```
# what if we made it very low
d1_verylow <- density(mom_dt[tobacco==1,dbrwt], kernel="gaussian", bw=20, adjust=1,
                      weights=mom_dt[tobacco==1,norm_prop_weights])
plot(d1_verylow, col="black", main="Kernel Density, Smoker: Very Low Bandwidth",
     xlab = "Birthweight (g)")
```



We use the treated group counterfactual density and try a few different bandwidths. First, we plot our selected bandwidth, 68.9. Next, since the treated group is smaller than the control group, the bandwidth selected had been larger than that of the control group, so we plot the density had we used the same bandwidth as the control, ~ 49.8. Next we crank up the bandwidth to 200 and finally we drop it to 20. The higher the bandwidth, the smoother the density. At a bandwidth of 20, the density becomes a lot choppier. This is what we expect as a lower bandwidth in a Gaussian kernel gives less weight to observations further from x, making $\hat{f}(x)$ more sensitive to observations closer to x, producing less smoothing.

(f) What are the benefits of the weighting approach (from part c)? What are the potential drawbacks? Pay particular attention to to the issue of people with extremely high and extremely low values of the propensity score.

The benefits of the weighting approach in part c are that we can get consistent estimates of the ATE and TOT that are also efficient. One potential drawback is that we have to first estimate the propensity score, which we are not certain we estimate correctly. Additionally, we run into problems if there is insufficient overlap in the treatment and control distributions of the covariates. This in turn would cause insufficient overlap in the propensity score itself, where there would be ranges of $p(X)$ that contain many estimated scores from the treatment group but not from the control group, or vice versa. If the propensity score $p(X_i)$ gets close to zero or one, then the weights become enormous, making our estimation in part (c) above very sensitive to outliers. Specifically, since people with extremely high propensity scores would mostly be found in the treated group and people with extremely low propensity scores would mostly be found in the control group, any observations with extremely high propensity scores found in the control group, or with extremely low propensity scores found in the treatment group, would receive an enormous weight since our weighting scheme above balances the propensity score across treated and control group. Our estimation would then be very sensitive to a few observations, or outliers. To address this concern, we can trim observations above or below certain propensity scores to improve overlap between treatment and control, as in Imbens (2007).

(g) Present your findings and interpret the results on the relationship between birthweight and smoking. For the estimates in parts (b) and (c), consider which of the following conditions must hold in order for that estimate to be valid: (A1) The treatment effect heterogeneity is linear in the propensity score. (A2) The treatment effect heterogeneity is not linear in the propensity score. (A3) The decision to smoke is completely randomly assigned. (A4) Conditional on the exogenous variables the decision to smoke is randomly assigned.

Our findings are included with the relevant sections above. All of our results are consistent with one another in finding that smoking causes lower birthweights – as indicated by our estimates of the ATE, the TOT, and the counterfactual densities of birthweight. In part (b), we include the propensity score in a regression. In order for this estimate to be valid, we need assumptions A1 and A4 to hold (assuming we also include an interaction term between treatment status and the propensity score to allow for heterogeneous treatment effects). In part (c), we again need assumption A4 to hold.

Question 3

A potentially more informative way to describe how birth weight affects smoking is to estimate the "non-parametric" conditional mean of birth weight as a function of the estimated probability of smoking, separately for smokers and non-smokers on the same graph. To do so, divide the data from smokers into 100 approximately equally spaced bins based on the estimated propensity score. Do the same for nonsmokers. Use the blocking estimator we discussed in class. Interpret your findings and relate them to the results in (2b).

```r
mom_dt[, block := as.numeric(cut(prop_score_sig, 100))] # Divide propensity scores into 100 bins.
mom_dt[, low_bw := ifelse(dbrwt < 2500, 1, 0)] # Indicator for low birth weight

# Divide smokers and non-smokers into propensity score bins, calculate mean birthweights and low birthweight
tau_k_dt <- mom_dt[,.(mean_bw = mean(dbrwt), mean_low = mean(low_bw), n = .N), by = c("block", "tobacco")]
tau_k_dt <- dcast(tau_k_dt, block ~ tobacco, value.var=c("mean_bw", "mean_low", "n"))

N <- sum(tau_k_dt[,n_0],na.rm = TRUE) + sum(tau_k_dt[,n_1],na.rm = TRUE) # calculate sample size
tau_k_dt[, weight := (n_0 + n_1) / N] # calculate weight of each bin

# Calculate blocking estimator (weighted average of each bin's treatment effect).
tau_k_dt[, tau_k_hat := mean_bw_1 - mean_bw_0] # difference within each bin
tau_k_dt[, tau_weighted := tau_k_hat*weight] # weight each "bin effect"
print(sum(tau_k_dt[,tau_weighted],na.rm = TRUE)) # sum up to find treatment effect and print
```

```
## [1] -222.751
```

We calculate a treatment effect within each bin and then sum a weighted average of these within bin treatment effects to find our blocking estimator treatment effect estimate of about -223 grams.

Question 4

Low birth weight births (less than 2,500 grams) are considered particularly undesirable since they comprise a large share of infant deaths. Redo question 3 using an indicator for low birth weight births as the outcome of interest. Interpret your findings.

```r
# Calculate blocking estimator (weighted average of each bin's treatment effect).
tau_k_dt[, tau_k_hat := mean_low_1 - mean_low_0] # difference within each bin
tau_k_dt[, tau_weighted := tau_k_hat*weight] # weight each "bin effect"
print(sum(tau_k_dt[,tau_weighted],na.rm = TRUE)) # sum up to find treatment effect and print
```

```
## [1] 0.03970231
```

```r
rm(N)
rm(tau_k_dt)
```

We estimate that smoking causes babies to be about 4% more likely to be born weighing less than 2500g.

Question 5

Let's link matching back to regression. Consider the conditional expectation function $E[\ \mathit{birthweight} \mid X]$, where X contains the following variables: `rectype pldel3 cntocpop stresfip dmage mrace3 dmar adequacy csex dplural`.

(a) Develop a regression that you are confident estimates $E[\ \mathit{birthweight} \mid X]$ as $N \to \infty$? Why are you confident that your regression gets the CEF right?

A regression of birthweight on a saturated model for the discrete regressors (i.e. a model with a separate parameter for every possible combination of values that the set of regressors can take on) will produce asymptotically consistent estimates of the CEF, assuming that as $N \to \infty$ we would add in more dummy variables for any additional new combinations of the variables in X that appear in the data.

We are confident that this regression gets the CEF right because having a saturated model for discrete regressors is a sufficient condition for the CEF to be linear. Then by the Regression-CEF Theorem, as long as the CEF is linear the regression of Y on X estimates the CEF. Formally, if $E[y_i|x_i] = x_i\gamma$, then $\gamma = E[x_i'x_i]^{-1}E[x_i'y_i]$, which the regression coefficient will converge to.

To see that we can estimate a fully saturated model, note from the codebook that all of our regressors are indeed discrete:

- `rectype` is record type (resident or nonresident): a discrete variable with values $\in \{1, 2\}$
- `pldel3` is place or facility of birth: a discrete variable with values $\in \{1, 2\}$
- `cntocpop` is population size of county of occurrence: a discrete variable with values $\in \{0, 1, 2, 3\}$
- `stresfip` is state of residence: a discrete variable with values $\in \{0, ..., 55\}$
- `dmage` is age of mother: a discrete variable with values $\in \{12, 13, ..., 49\}$
- `mrace3` is race of mother: a discrete variable with values $\in \{1, 2, 3\}$
- `dmar` is marital status of mother: a discrete variable with values $\in \{1, 2\}$
- `adequacy` is adequacy of care index: a discrete variable with values $\in \{1, 2, 3\}$
- `csex` is child sex: a discrete variable with values $\in \{1, 2\}$
- `dplural` is plurality (single, twin, triplet, etc): a discrete variable with values $\in \{1, 2, 3, 4\}$

(b)*Now run the regression you propose above, but add the treatment (your binary smoking variable) as the righthand side variable of interest. Prove that if the treatment effect of smoking on birthweight is independent of the covariates in X, then exact matching and your regression estimate the same thing. You may assume the conditional indpendence assumption holds given the variables in X listed above.

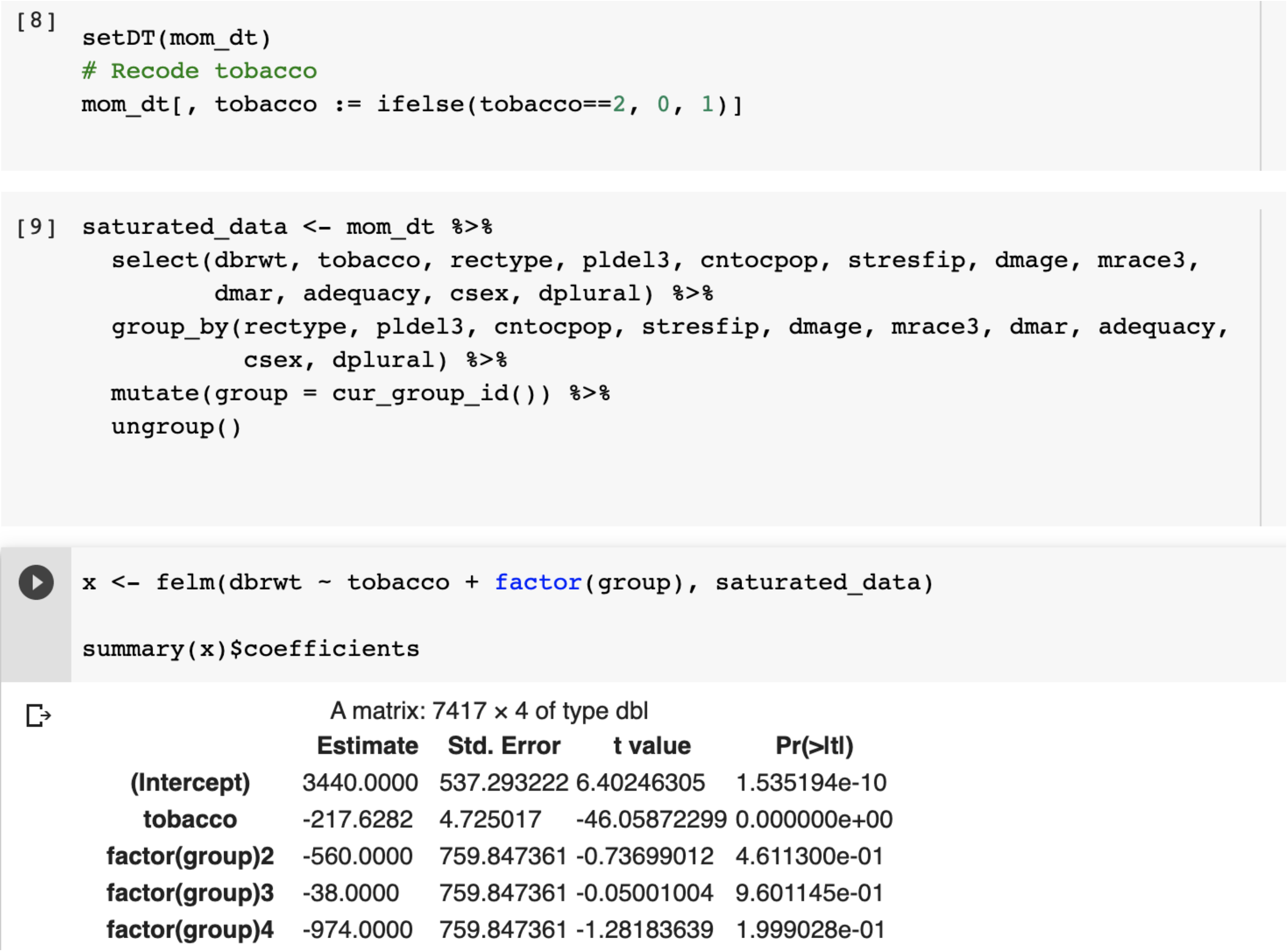
We estimate the saturated model from (a) with our treatment variable (smoking during pregnancy) also on the right hand side. Specifically, we estimate $y_i = \alpha + \tau D_i + \beta g(X_i) + \epsilon_i$, where D_i represents treatment status and $g(X_i)$ captures all of the parameters for every possible combination of values in our other regressors.

```r
# the following code came from Arthur's tip on the Slack channel

saturated_data <- mom_dt %>%
  # take a subset of the data with only the variables we are interested in
  select(dbrwt, tobacco, rectype, pldel3, cntocpop, stresfip, dmage, mrace3,
         dmar, adequacy, csex, dplural) %>%
  # combinations of the variables in X
  group_by(rectype, pldel3, cntocpop, stresfip, dmage, mrace3, dmar, adequacy,
          csex, dplural) %>%
  # give each group a unique id
  mutate(group = cur_group_id()) %>%
  ungroup()

# the output (from running on Google Colab) is included below
# the line below takes a long time to run, so is commented out

# felm(dbrwt ~ tobacco + factor(group), saturated_data)
```



```
[8]  setDT(mom_dt)
     # Recode tobacco
     mom_dt[, tobacco := ifelse(tobacco==2, 0, 1)]
```

```
[9]  saturated_data <- mom_dt %>%
       select(dbrwt, tobacco, rectype, pldel3, cntocpop, stresfip, dmage, mrace3,
              dmar, adequacy, csex, dplural) %>%
       group_by(rectype, pldel3, cntocpop, stresfip, dmage, mrace3, dmar, adequacy,
                csex, dplural) %>%
       mutate(group = cur_group_id()) %>%
       ungroup()
```

```
x <- felm(dbrwt ~ tobacco + factor(group), saturated_data)

summary(x)$coefficients
```

```
                  A matrix: 7417 × 4 of type dbl
                  Estimate   Std. Error  t value      Pr(>|t|)
  (Intercept)     3440.0000  537.293222  6.40246305   1.535194e-10
      tobacco     -217.6282  4.725017    -46.05872299 0.000000e+00
factor(group)2    -560.0000  759.847361  -0.73699012  4.611300e-01
factor(group)3    -38.0000   759.847361  -0.05001004  9.601145e-01
factor(group)4    -974.0000  759.847361  -1.28183639  1.999028e-01
```

Figure 1: 5(b) regression output

Our model estimates an ATE of $\tau = -217.6$, that is maternal smoking is associated with a decrease of 217.6 grams in birth weight, ceteris paribus.

Note that our regression estimate and an exact matching estimator produce the same result under the assumptions that the treatment effect of smoking on birthweight is independent of the covariates and that the conditional independence assumption holds given our variables in X. To see this, note that from Lecture Notes 2C we can define a generic matching estimator as:

$$\hat{\tau}_M = \frac{1}{N_T} \sum_{i \in \{D_i = 1\}} [y_i - \sum_{j \in \{D_j = 0\}} w_i(j) y_j]$$

For exact matching, we match observations that have the exact same values for each covariate, that is we set the weights $w_i(j) = 1$ if $X_i = X_j$, and to 0 otherwise. Formally, we can define the weights for exacting matching as $w_i(j) = \frac{\mathbf{1}\{X_i = X_j\}}{\sum_{j\in\{D_j=0\}} \mathbf{1}\{X_i = X_j\}}$, so that $\sum_j w_j(j)y_i = \bar{y}_c$ the control mean, so $\hat{\tau}_M$ is just the difference in means between the treatment and control groups.

$$\hat{\tau}_M = \frac{1}{N_T} \sum_{i\in\{D_i=1\}} [y_i - \sum_{j\in\{D_j=0\}} w_i(j)y_j]$$

$$= \frac{1}{N_T} \sum_{i\in\{D_i=1\}} [y_i - \sum_{j\in\{D_j=0\}} \frac{\mathbf{1}\{X_i = X_j\}}{\sum_{j\in\{D_j=0\}} \mathbf{1}\{X_i = X_j\}} y_j]$$

Substituting y_i with the conditional expectation of y given X_i, D_i, gives us:

$$= \frac{1}{N_T} \sum_{i\in\{D_i=1\}} [E[y_i|D_i = 1, g(X_i)] - \sum_{j\in\{D_j=0\}} \frac{\mathbf{1}\{X_i = X_j\}}{\sum_{j\in\{D_j=0\}} \mathbf{1}\{X_i = X_j\}} E[y_j|D_j = 0, g(X_j)]]$$

Under the assumptions of unconfoundedness, conditional independence, and a constant treatment effect, and because our CEF is linear under a fully saturated model, we know that we can consistently estimate $[E[y_i|D_i, g(X_i)]$ with the regression $y_i = \alpha + \tau D_i + \beta g(X_i) + \epsilon_i$. Substituting into our expression above yields:

$$= \frac{1}{N_T} \sum_{i\in\{D_i=1\}} [(\alpha + \tau + \beta g(X_i)) - \sum_{j\in\{D_j=0\}} \frac{\mathbf{1}\{X_i = X_j\}}{\sum_{j\in\{D_j=0\}} \mathbf{1}\{X_i = X_j\}} (\alpha + \beta g(X_j))]$$

and since we are only interested in $X_i = X_j$ for exact matching,

$$= \frac{1}{N_T} \sum_{i\in\{D_i=1\}} [(\alpha + \tau + \beta g(X_i)) - \sum_{j\in\{D_j=0|X_i=X_j\}} \frac{1}{\sum_{j\in\{D_j=0\}} \mathbf{1}\{X_i = X_j\}} (\alpha + \beta g(X_j))]$$

$$= \frac{1}{N_T} \sum_{i\in\{D_i=1\}} [(\alpha + \tau + \beta g(X_i)) - \frac{\sum_{j\in\{D_j=0|X_i=X_j\}} 1}{\sum_{j\in\{D_j=0|X_i=X_j\}} \mathbf{1}\{X_i = X_j\}} (\alpha + \beta g(X_j))]$$

$$= \frac{1}{N_T} \sum_{i\in\{D_i=1\}} [(\alpha + \tau + \beta g(X_i)) - (\alpha + \beta g(X_j))]$$

$$= \frac{1}{N_T} \sum_{i\in\{D_i=1\}} \tau$$

$$= \frac{1}{N_T} N_T * \hat{\tau}$$

$$= \hat{\tau}$$

As we can see, under the assumption that the treatment effect of smoking on birthweight is independent of the covariates and that the conditional independence assumption holds, exact matching and our regression model both consistently estimate the ATE $\hat{\tau}$.

(c)*Develop a weighted version of the exact matching estimator that estimates the same thing as the regression above (regardless of whether the treatment effect is independent of covariates).

The following was discussed with and inspired by Abdou. Under the assumption that the treatment effect is independent of the covariates, a simple difference in means between the treatment and control groups (as discussed in part b above) is sufficient to consistently estimate the ATE. However, if we relax the assumption that the treatment effect is independent of the covariates, then we are essentially allowing for selection on the observables, and we need to adjust our weights accordingly. Intuitively, we want to place less weight on observations in cells where there is little overlap between treatment and control, and more weight on observations in cells with more overlap between treatment and control.

Because we have exact matching, each treatment and control observation will belong to only one cell, where we define a cell k as all observations for which $X_i = X_j$. Define $n_{ik} = |\{j : j \in k \;\&\; D_i = D_j\}| = |\{j : X_i = X_j \;\&\; D_i = D_j\}|$ as the number of observations with the same treatment status as i in a given cell k, D_{ik} being the treatment status of i in cell k. Since we also need to account for within-cell overlap of the treatment status in our weights, we also define σ_k as the within-cell variance of treatment status. Then we define our weights as:

$$w_{ik} = \frac{n_{ik}\sigma_k N_c}{\sum_{k=1}^{K} \sum_{i \in (D_i = c)} n_{ik}\sigma_k} \forall c \in 0, 1$$

which are normalized to sum to N_T and N_C for treatment and control observations respectively, where N_T is the total number of treatment observations and N_C is the total number of control observations in our sample. Note that this weighting scheme gives 0 weight to observations in cells with insufficient overlap in treatment status, that is cells with only treatment or only control observations.

$$\sum_{D_i = c} w_i = N_c \text{ for } c \in \{0, 1\}$$

Then, using the weights defined above, we can estimate the ATE as:

$$\tau_M = \frac{1}{N_T} \sum_{D_i = 1} y_{ik} w_{ik} - \frac{1}{N_C} \sum_{D_i = 0} y_{ik} w_{ik}$$

(d)*Estimate the weighted matching estimator you propose. Compare it to the regression estimate from part (b). Are they similar?

```
#Generate weights
saturated_data <- as.data.table(saturated_data) #set data as a data.table
N_T <- nrow(saturated_data[tobacco == 1]) #number of treatment observations
N_C <- nrow(saturated_data[tobacco == 0]) #number of control observations

saturated_data[,n:= .N, by = .(group, tobacco)] #generate n_jk, which is the number of observations in each g
saturated_data[,vark := var(tobacco), by = .(group)] #generate var_k, which is the variance of the treatment
saturated_data[,nvark := n*vark] #generate a variable = n*vark
saturated_data[,matching_weights := ifelse(tobacco == 1, nvark*N_T/sum(saturated_data[tobacco == 1 & !is.na(n

# estimate ATE
tau_M <- (1/N_T)*sum(saturated_data[tobacco == 1 & !is.na(matching_weights), matching_weights*dbrwt]) - (1/N_

# [old way of estimating ATE when weights added up to 1; we can delete the code before submission, this is ju
#saturated_data[tobacco == 1,ywT := sum(dbrwt*matching_weights), by=.(group)] #within each group, generate th
#saturated_data[tobacco == 0,ywC := sum(dbrwt*matching_weights), by=.(group)]#within each group, generate the
#saturated_data <- saturated_data[,ywT := na.locf0(ywT, fromLast = TRUE), by = .(group)][,ywT := na.locf(ywT)
#saturated_data <- saturated_data[,ywC := na.locf0(ywC, fromLast = TRUE), by = .(group)][,ywC := na.locf(ywC)
#saturated_data_groups <- unique(saturated_data, by = "group") #create sub-table that includes 1 observation
#tau_M <- sum(saturated_data_groups[!is.na(ywT) & !is.na(ywC),ywT-ywC]) #By our formula for the ATE for our p
```

As discussed in part (c) above, we estimate weights:

$$w_{ik} = \frac{n_{ik}\sigma_k N_c}{\sum_{k=1}^{K} \sum_{i \in (D_i = c)} n_{ik}\sigma_k} \forall c \in 0, 1$$

And then use our weights to estimate the ATE:

$$\tau_M = \frac{1}{N_T} \sum_{D_i = 1} y_{ik} w_{ik} - \frac{1}{N_C} \sum_{D_i = 0} y_{ik} w_{ik}$$

Using this procedure, we estimate τ_M = -258.13, which is less than what we estimated in part (b).

(e)*Is the sample size of your regression the same as the sample size of your matching estimator, or does the regression have more observations? If the regression has more observations, why don't these extra observations influence the treatment effect estimate?

The sample size of the regression and the sample size of the matching estimator are not the same. The regression has more observations. This is because for the matching estimator, if a given cell doesn't have both a control and treatment observation, then that cell is dropped because we cannot estimate the treatment effect within that cell. But the regression doesn't drop any observations, it just gives 0 weight to the observations where there is not an exact match of covariates in treatment and control, so these "extra" observations do not influence our estimate of the ATE.

(f)*Compute a standard error for your matching estimator using the formula from Imbens (2015). Specifically, note that your matching estimator should have a form $\frac{1}{N_t}\sum_{d_i=1} w_i y_i - \frac{1}{N_c}\sum_{d_i=0} w_i y_i$, where $\sum_{d_i=1} w_i = N_t$ and $\sum_{d_i=0} w_i = N_c$. Then the conditional variance is approximately $\sum_i (\frac{d_i}{N_t^2} + \frac{1-d_i}{N_c^2}) w_i^2 \hat{\sigma}^2_{d_i}(x_i)$, where $\hat{\sigma}^2_{d_i}(x_i) = \frac{1}{2}(y_i - y_{nn(i)})^2$, and $y_{nn(i)}$ is the nearest neighbor to observation i with the *same* treatment status. Figure out the implicit weights w_i in your estimator from part (d), and compute the conditional variance. Is it close to your regression coefficient variance?

```r
# get treat/control data
treat_data <- mom_dt[tobacco==1,c("rectype", "pldel3", "cntocpop", "stresfip", "dmage", "mrace3", "dmar", "ad
control_data <- mom_dt[tobacco==0,c("rectype", "pldel3", "cntocpop", "stresfip", "dmage", "mrace3", "dmar", "

treat_data <- dummy_cols(treat_data, select_columns = c("cntocpop", "stresfip", "mrace3", "dmar"))
control_data <- dummy_cols(control_data, select_columns = c("cntocpop", "stresfip", "mrace3", "dmar"))

treat_data <- treat_data[,!c("cntocpop", "stresfip", "mrace3", "dmar")]
control_data <- control_data[,!c("cntocpop", "stresfip", "mrace3", "dmar")]

# get nearest neighbor match
match1 <- get.knn(data=treat_data, k=1, algorithm = "kd_tree")
match1dt <- data.table(sapply(match1,c))
match1dt$id <- as.numeric(rownames(match1dt))

match0 <- get.knn(data=control_data, k=1, algorithm = "kd_tree")
match0dt <- data.table(sapply(match0,c))
match0dt$id <- as.numeric(rownames(match0dt))

# merge to original data
saturated_data$main_id <- as.numeric(rownames(saturated_data))
treat_ids <- saturated_data[tobacco==1, c("main_id")]
treat_ids$id <- as.numeric(rownames(treat_ids))
control_ids <- saturated_data[tobacco==0, c("main_id")]
control_ids$id <- as.numeric(rownames(control_ids))

# merge ids
match1dt <- merge(match1dt, treat_ids, by = "id")
match0dt <- merge(match0dt, control_ids, by = "id")

# merge matched ids
match1dt <- merge(match1dt, treat_ids, by.x="nn.index", by.y = "id")
match0dt <- merge(match0dt, control_ids, by.x="nn.index", by.y = "id")
matchdt <- rbind(match1dt, match0dt)

matchdt <- matchdt[,4:5]
colnames(matchdt) <- c("main_id", "match_id")

saturated_data <- merge(saturated_data, matchdt, by = "main_id")
```

```
#Merge birthweight information of matched observations to main dataset
birthweights <- saturated_data[,.(match_id = main_id, match_dbrwt = dbrwt)] #create sub dataset of ID and bir
saturated_data <- merge(saturated_data, birthweights, by = "match_id")

# Calculate conditional variance
saturated_data[,diff := dbrwt - match_dbrwt]
saturated_data[,sigma_d := 0.5*(diff^2)]
cond_var <- sum(((1/N_T^2)+(1/N_C^2))*saturated_data[!is.na(matching_weights), matching_weights^2*sigma_d])
cond_se <- sqrt(cond_var)
```

We calculated the weights directly in part (d) above as:

$$w_{ik} = \frac{n_{ik}\sigma_k N_c}{\sum_{k=1}^{K}\sum_{i\in(D_i=c)} n_{ik}\sigma_k} \forall c \in 0,1$$

To calculate the Imbens standard errors, we use get.knn from the FNN to find the nearest neighbor match for each observation with the same treatment status. We can then calculate the conditional variance using the formula above, which we find is equal to 178.045 which implies a standard error of 13.343. This is much larger than the SE of the regression.

Question 6

Concisely and coherently summarize your overall results, providing some intuition. Write it like you would the conclusion of a paper. In this summary, describe whether you think your best estimate of the effects of smoking is credibly identified. State why or why not.

We have used various methods to estimate the effect of smoking on birth weight. We have assumed throughout that mothers with comparable observable characteristics have the same treated and untreated outcomes. These various estimates have, for the most part, been consistently around -220g (smoking reduces birthweight by 220g). One exception, however, is the matching estimator in Question 6, which produced a larger estimate with larger standard errors. Intuitively, this may be because the matching estimator must remove mothers from the sample who do not have a good match in the other group. If the treatment effect is a function of observable characteristics, then we may have removed some mothers with a smaller treatment effect from our sample, increasing our estimated treatment effect.

Despite the relatively small differences between different estimators, most of our estimates have been close to the others. All of these estimators, however, depend on the conditional independence assumption, which there is reason to doubt. Smoking could be correlated with the mother's background in ways that violate the conditional independence assumption (for example, it could be correlated with quality of mother's schooling or with familial support). In this case, it is important to remember that even the best estimator cannot make up for a lack of exogenous variation in treatment assignment.