Social Behaviour Dynamics: Week 2

Noémi Schuurman, based on materials of Ellen Hamaker

February 2022

In this lab we follow the analyses discussed in the paper by Schafer and Kang (2008) that are used to estimate causal effects and control for confounders. There are nine techniques they cover; below, all of them will be included, but note that some of them are optional (they are included for the sake of completeness, but they are not exam material).

We will also use the data from Schafer and Kang, which are in the data file called SchaferKangData.dat. These are simulated data, and hence the authors (and we) know what the correct answer to the question ``What is the effect of dieting on emotional distress?" actually is. Hence, the purpose of the exercises here is to obtain a deeper understanding and hands-on experience with the diverse techniques.

Make sure to compare the results you get throughout the exercises to those reported in Table 6 in Schafer and Kang. For null hypothesis tests, you can use a significance level of 0.05 throughout.

1. Data and R-packages

In this practical you will make use of various R-packages. If you haven't already, install the following packages:

```
# install.packages("tableone")
# install.packages("MatchIt")
# install.packages("survey")
```

Load the data, which are in the datafile called SchaferKangData.dat. Take a look at the data set. See Table 3 in Schafer and Kang for a description of the variables.

```
df <- read.table("SchaferKangData.dat", header=T)
df[1:10,]
```

```
##    DISTR.1 BLACK NBHISP GRADE SLFHLTH SLFWGHT WORKHARD GOODQUAL PHYSFIT PROUD
## 1     0.47     0      0    10       4       4        2        2       2     2
## 2     0.05     0      0     8       2       3        1        1       2     1
## 3     0.63     0      0     8       2       3        2        2       2     2
## 4     0.16     1      0     9       1       4        2        1       2     1
## 5     1.79     1      0     9       4       4        1        2       2     2
## 6     0.32     0      0    10       1       3        1        1       2     1
## 7     1.11     0      0    10       2       3        1        1       2     1
## 8     0.32     0      0     9       3       4        2        1       3     2
## 9     0.26     0      0    10       1       4        2        2       2     2
## 10    0.89     1      0    10       2       2        2        2       1     1
##    LIKESLF ACCEPTED FEELLOVD DIET DISTR.2
## 1        5        2        2    1    0.40
## 2        1        2        1    1    0.09
## 3        3        2        2    0    0.69
## 4        2        2        1    1    0.08
## 5        3        3        2    0    1.41
## 6        4        1        1    0    0.52
## 7        2        2        2    0    1.32
## 8        2        3        3    0    0.29
## 9        1        3        2    0    0.43
## 10       1        1        1    0    1.76
```

2. Prima Facie Effect: Not controlling for any variable.

When a randomized controlled trial (RCT) has been conducted, the treatment groups should not differ on any (pre-treatment) covariate due to random assignment. In that case, the ACE can be computed by taking the difference in means between the two groups for the outcome variable (not controlling for any variable). This is also referred to as the *prima facie effect*.

2.1 Method 1: *Compare means*

Although the current data are not generated by a RCT scenario, we will nevertheless consider this naive estimation approach for the causal effect.

▶ Compare the means between the groups on the outcome variable (see Equation (5) in S&K; use for instance `t.test()` in R).

```
t.test1 <- t.test(DISTR.2 ~ DIET, df)
# Difference in means between the groups:
m1 <- t.test1$estimate[2]
m0 <- t.test1$estimate[1]
m1 - m0 #Note: the result of this command may carry on the header "mean for group 1", bu
t it really is the difference in means.
```

```
## mean in group 1
##      0.05961424
```

```
# Standard error
t.test1$stderr
```

```
## [1] 0.01533978
```

```
# p-value
t.test1$p.value
```

```
## [1] 0.0001054363
```

```
# The mean difference is the "prima facie" estimate of the average causal effect of diet
ing on distress. The results indicate that the average of the girls who did diet is (0.7
03-0.645=)0.059 higher than the average of the girls who did not diet ($SE=0.015$). This
difference is statistically significant.
```

2.2 Investigating covariates - potential confounding

The mean difference above is based on the assumption that there is no confounding, so we can simply take that difference to estimate the mean causal effect. However, we have a set of observed covariates, and if our data mimic a true RCT, there should be no mean differences between the two groups on these covariates. We can evaluate this, for example, with the *standardized mean difference*, that is:

$$\Delta Z = \frac{(\bar{Z}|X=1) - (\bar{Z}|X=0)}{\sqrt{((S^2|X=1) + (S^2|X=0))/2}}$$

where

- $\bar{Z}|X = x$ is the mean on covariate Z in group x
- $S^2|X = x$ is the variance in group x

- $X = 1$ is the diet group; $X = 0$ is the non-diet group

The standardized mean difference is an effect size measure to evaluate how large a difference is between two means, and is also referred to in the literature as 'Cohen's d'. Of course, one could consider other ways of evaluating practically important differences between group means, we'll use the SMD here to get the main idea. For this purpose, use a "cut-off of 0.1" to indicate whether the difference is concerning or not.

▶ Determine the standardized mean difference between the girls who did diet versus the girls who did not diet on the first covariate, that is: DISTR.1 (emotional distress at wave 1).

```
df1 <- df[ which(df$DIET == 1), ]
df0 <- df[ which(df$DIET == 0), ]

(mean(df1$DISTR.1) - mean(df0$DISTR.1))/(sqrt( (var(df1$DISTR.1)+var(df0$DISTR.1))/2 ))
```

```
## [1] 0.2142174
```

▶ Instead of computing these standardized mean differences yourself, you can also use the function `CreateTableOne()` from the package `tableone` . Run the code below.

```
library(tableone)
table1 <- CreateTableOne(vars=c("DISTR.1","BLACK", "NBHISP", "GRADE",
                      "SLFHLTH", "SLFWGHT", "WORKHARD", "GOODQUAL",
                          "PHYSFIT", "PROUD", "LIKESLF", "ACCEPTED",
                      "FEELLOVD"), strata="DIET", data=df,
                        test=FALSE)
```

▶ Obtain the table with standardize mean differences using `print(table1, smd=TRUE)` , and comment on the results.

```
print(table1, smd=TRUE)
```

```
##                        Stratified by DIET
##                         0           1           SMD
##   n                     4780        1220
##   DISTR.1 (mean (SD))  0.62 (0.42) 0.71 (0.45)  0.214
##   BLACK (mean (SD))    0.26 (0.44) 0.17 (0.38)  0.197
##   NBHISP (mean (SD))   0.15 (0.35) 0.15 (0.36)  0.021
##   GRADE (mean (SD))    9.16 (1.39) 9.37 (1.34)  0.152
##   SLFHLTH (mean (SD))  2.20 (0.93) 2.35 (0.91)  0.171
##   SLFWGHT (mean (SD))  3.19 (0.76) 3.84 (0.70)  0.895
##   WORKHARD (mean (SD)) 2.14 (0.91) 2.05 (0.85)  0.093
##   GOODQUAL (mean (SD)) 1.80 (0.67) 1.84 (0.71)  0.055
##   PHYSFIT (mean (SD))  2.24 (0.93) 2.53 (0.93)  0.320
##   PROUD (mean (SD))    1.76 (0.77) 1.86 (0.79)  0.126
##   LIKESLF (mean (SD))  2.09 (0.99) 2.52 (1.06)  0.420
##   ACCEPTED (mean (SD)) 2.14 (1.00) 2.35 (1.06)  0.207
##   FEELLOVD (mean (SD)) 1.78 (0.83) 1.93 (0.90)  0.172
```

```
# Most of the standardized mean differences (the last column) are
# larger than the rule of thumb of 0.1; this implies that for multiple
# covariate, the difference between their means of the two treatment
# groups are larger than 0.1 standard deviations. This indicates a
# considerable imbalance across the two groups with respect to these
# covariates. This means the data do not mimic an RCT very well.

# Hence, in the next exercise we are going to take some action to tackle this problem.
```

3 Controlling for Confounders Z: Model the relation of Z with Y

Schafer and Kang describe three techniques to control for confounders Z, that in some way using these confounders as predictors for the outcome variable Y:

- *ANCOVA* (which is what they call linear regression with no interactions between predictors)
- *regression* (which is what they call linear regression with interactions between predictors)
- *regression estimation*, which is based on estimating the potential outcomes Y_i^0 and Y_i^1 for each participant based on the covariates

We will include all the covariates in our analyses below, even if they had a small standardized mean difference in the previous exercise. This to align with Schafer & Kang, although it may be less important to control for these covariates. Furthermore, although there may not be a need to correct for all covariates, they may still account for variance in the outcome variable, and by accounting for this, we increase the power of our analysis. Do keep in mind that here, we assume we are sure that all of these covariates are potential confounders and hence something to control for. In practice we are often not so sure about this. Related to this, remember what we learned last week: In general we do not want to just control for all covariates we have - controlling accidentaly for mediator or collider variables could mess up our estimates of causal effects!

3.1 Method 2: *ANCOVA* (Linear Regression without Interactions)

To run an ANCOVA as described in S&K, we can simply run a regression model *without interactions* between the predictors (see Equation (6) in S&K). Our model here can be written as:

$$DISTR.2_i = \alpha + \theta DIET_i + \beta_1 DISTR.1_i + \beta_2 BLACK_i + \beta_3 NBHISP_i$$

$$+\beta_4 GRADE_i + \beta_5 SLFHLTH_i + \beta_6 SLFWGHT_i + \beta_7 WORKSHARD_i$$

$$+\beta_8 GOODQUAL_i + \beta_9 PHYSFIT_i + \beta_{10} PROUD_i$$

$$+\beta_{11} LIKESLF_i + \beta_{12} ACCEPTED_i + \beta_{13} FEELLOVED_i + e_i$$

Note: All our categorical variables are dichotomous (dummy-coded) variables, so we included them in the same way as the continuous variables. Keep in mind that for categorical predictor variables with more than 2 categories, it is prudent to make dummies. This can be done by including the categorical predictors as a factor variable, in which case dummies are made during the (g)lm analysis in R; alternatively, one would oneself create dummy variables for the various categories, and include those dummies as predictors in the regression model.

▶ Run the ANCOVA model (for instance using the function `glm()` in R), and interpret the results.

```
M2 <- glm(DISTR.2 ~ DIET    + DISTR.1 + BLACK + NBHISP
    + GRADE + SLFHLTH + SLFWGHT + WORKHARD + GOODQUAL
        + PHYSFIT + PROUD + LIKESLF + ACCEPTED + FEELLOVD,
        data=df)
summary(M2)
```

```
##
## Call:
## glm(formula = DISTR.2 ~ DIET + DISTR.1 + BLACK + NBHISP + GRADE +
##     SLFHLTH + SLFWGHT + WORKHARD + GOODQUAL + PHYSFIT + PROUD +
##     LIKESLF + ACCEPTED + FEELLOVD, data = df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.69102  -0.22781  -0.05734   0.18025   1.81213
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0052087  0.0422214  -0.123 0.901821
## DIET        -0.0136658  0.0129321  -1.057 0.290675
## DISTR.1      0.5178247  0.0125785  41.168  < 2e-16 ***
## BLACK        0.0745764  0.0120532   6.187 6.53e-10 ***
## NBHISP       0.0289905  0.0141752   2.045 0.040884 *
## GRADE        0.0019352  0.0035672   0.543 0.587490
## SLFHLTH      0.0196974  0.0059809   3.293 0.000996 ***
## SLFWGHT      0.0038613  0.0069990   0.552 0.581183
## WORKHARD    -0.0121711  0.0056656  -2.148 0.031734 *
## GOODQUAL     0.0209810  0.0098513   2.130 0.033231 *
## PHYSFIT     -0.0005642  0.0069283  -0.081 0.935099
## PROUD        0.0376379  0.0101757   3.699 0.000219 ***
## LIKESLF      0.0242944  0.0064040   3.794 0.000150 ***
## ACCEPTED     0.0167152  0.0068032   2.457 0.014040 *
## FEELLOVD     0.0389178  0.0085254   4.565 5.10e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1412421)
##
##     Null deviance: 1303.09  on 5999  degrees of freedom
## Residual deviance:  845.33  on 5985  degrees of freedom
## AIC: 5300.6
##
## Number of Fisher Scoring iterations: 2
```

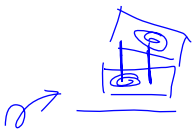
```
# The results indicate that after correcting for the effect of the covariates, the
# effect of the treatment (i.e., DIET) on the outcome (i.e. DISTR.2) is not
# significant; the parameter estimate of the ACE is -0.014 ($SE=0.013$).
```

reg. coefficient of DIET

3.2 Method 3: *Regression* (Linear Regression with extras such as interactions)

We can extend the ANCOVA model above by incorporating various product terms between the predictors:

- product between covariate and itself: *quadratic effect*
- product/interaction between two covariates: *interaction*
- product/interaction between treatment and covariates: *non-parallel planes* (see Figure 2 in Schafer and Kang)



Schafer and Kang consider the latter option in their paper.

- Note: Before including interactions, one needs to center the covariates, and the interaction term is build up out of these centered covariates. This to ensure the interpretation we want for the main effects: Centering makes the mean of a variable equal to zero. Further, remember that an interaction effect indicates that the effect of one variable on the outcome differs depending on the value of another variable. The interpretation of main effects is 'the effect of this variable on the outcome if the other variables are zero'. Hence, not centering the variables means that the main effect for treatment is the (causal) effect when the other variables are zero, which may not represent the *average* causal effect of treatment. If we center, the main effect is the (causal) effect when the other variables take on their mean value, hence we do get the *average* causal effect.

Why we need to center the covariates when running ANCOVA w/ interaction

- Note: Importantly, if there are interactions with treatment, that would imply that there are differently sized causal effects depending on the value of other variables.

▶ Run the regression model with two-way interactions between the treatment variable and the covariates, for instance using the function `glm()` in R, and interpret the results.

```
# First, center the covariates
dfc <- df
dfc$DISTR.1 <- dfc$DISTR.1 - mean(dfc$DISTR.1)
dfc$BLACK <- dfc$BLACK - mean(dfc$BLACK)
dfc$NBHISP <- dfc$NBHISP - mean(dfc$NBHISP)
dfc$GRADE <- dfc$GRADE - mean(dfc$GRADE)
dfc$SLFHLTH <- dfc$SLFHLTH - mean(dfc$SLFHLTH)
dfc$SLFWGHT <- dfc$SLFWGHT - mean(dfc$SLFWGHT)
dfc$WORKHARD <- dfc$WORKHARD - mean(dfc$WORKHARD)
dfc$GOODQUAL <- dfc$GOODQUAL - mean(dfc$GOODQUAL)
dfc$PHYSFIT <- dfc$PHYSFIT - mean(dfc$PHYSFIT)
dfc$PROUD <- dfc$PROUD - mean(dfc$PROUD)
dfc$LIKESLF <- dfc$LIKESLF - mean(dfc$LIKESLF)
dfc$ACCEPTED <- dfc$ACCEPTED - mean(dfc$ACCEPTED)
dfc$FEELLOVD <- dfc$FEELLOVD - mean(dfc$FEELLOVD)

# Next, run the model with the interactions between
# DIET and each of the centered covariates
M2b <- glm(DISTR.2 ~ DIET + DISTR.1 + BLACK + NBHISP
    + GRADE + SLFHLTH + SLFWGHT + WORKHARD + GOODQUAL
        + PHYSFIT + PROUD + LIKESLF + ACCEPTED + FEELLOVD
        + DIET*DISTR.1
        + DIET*BLACK
        + DIET*NBHISP
    + DIET*GRADE
        + DIET*SLFHLTH
        + DIET*SLFWGHT
        + DIET*WORKHARD
        + DIET*GOODQUAL
        + DIET*PHYSFIT
        + DIET*PROUD
        + DIET*LIKESLF
        + DIET*ACCEPTED
        + DIET*FEELLOVD,
        data=dfc)
summary(M2b)
```

```
##
## Call:
## glm(formula = DISTR.2 ~ DIET + DISTR.1 + BLACK + NBHISP + GRADE +
##     SLFHLTH + SLFWGHT + WORKHARD + GOODQUAL + PHYSFIT + PROUD +
##     LIKESLF + ACCEPTED + FEELLOVD + DIET * DISTR.1 + DIET * BLACK +
##     DIET * NBHISP + DIET * GRADE + DIET * SLFHLTH + DIET * SLFWGHT +
##     DIET * WORKHARD + DIET * GOODQUAL + DIET * PHYSFIT + DIET *
##     PROUD + DIET * LIKESLF + DIET * ACCEPTED + DIET * FEELLOVD,
##     data = dfc)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.72288  -0.22801  -0.05723   0.18175   1.82660
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.6587668  0.0055344 119.031  < 2e-16 ***
## DIET          -0.0058574  0.0149019  -0.393 0.694286
## DISTR.1        0.5210525  0.0143816  36.230  < 2e-16 ***
## BLACK          0.0742012  0.0131869   5.627 1.92e-08 ***
## NBHISP         0.0241334  0.0159814   1.510 0.131071
## GRADE          0.0021585  0.0039652   0.544 0.586210
## SLFHLTH        0.0257745  0.0067034   3.845 0.000122 ***
## SLFWGHT        0.0026900  0.0077575   0.347 0.728781
## WORKHARD      -0.0194426  0.0062698  -3.101 0.001938 **
## GOODQUAL       0.0154691  0.0110400   1.401 0.161212
## PHYSFIT        0.0054135  0.0077812   0.696 0.486635
## PROUD          0.0329261  0.0113575   2.899 0.003756 **
## LIKESLF        0.0225898  0.0072863   3.100 0.001942 **
## ACCEPTED       0.0159821  0.0077195   2.070 0.038462 *
## FEELLOVD       0.0412613  0.0095958   4.300 1.74e-05 ***
## DIET:DISTR.1  -0.0106838  0.0296963  -0.360 0.719033
## DIET:BLACK    -0.0003361  0.0327604  -0.010 0.991814
## DIET:NBHISP    0.0238344  0.0346707   0.687 0.491826
## DIET:GRADE     0.0023927  0.0091386   0.262 0.793465
## DIET:SLFHLTH  -0.0297672  0.0150042  -1.984 0.047310 *
## DIET:SLFWGHT  -0.0011506  0.0181310  -0.063 0.949404
## DIET:WORKHARD  0.0392602  0.0147076   2.669 0.007620 **
## DIET:GOODQUAL  0.0260125  0.0246058   1.057 0.290477
## DIET:PHYSFIT  -0.0266321  0.0171818  -1.550 0.121190
## DIET:PROUD     0.0240841  0.0255893   0.941 0.346650
## DIET:LIKESLF   0.0086665  0.0153017   0.566 0.571161
## DIET:ACCEPTED -0.0015125  0.0164325  -0.092 0.926664
```

```
## DIET:FEELLOVD -0.0125126  0.0209234  -0.598 0.549850
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1410809)
##
##     Null deviance: 1303.09  on 5999  degrees of freedom
## Residual deviance:  842.54  on 5972  degrees of freedom
## AIC: 5306.7
##
## Number of Fisher Scoring iterations: 2
```

```
# This shows the ACE is estimated to be -0.006 ($SE=0.015$),
```

3.3 Method 4: *Regression estimation* (Don't confuse with the Regression Methods 2&3...)

The method of 'Regression estimation` discussed by S&K is *not* the same as regular regression analysis with control variables (the methods discussed in the two previous exercises). In regression estimation, we make actual *predictions of the potential outcomes that were not observed*, and use these to compute the causal effect of interest.

To use regression estimation (see Equation (15) in S&K), you have to:

- divide the data set into those who were treated and those who were not treated
- estimate a regression model (with all the covariates) in each group separately
- obtain the parameter estimates from each group (see Equations (13) and (14) in S&K)
- use these and the covariates to predict the potential outcomes $\hat{Y}_i^0$ and $\hat{Y}_i^1$
- compute the average difference between these *predicted* potential outcomes

Hence, this approach is a technique to impute the missing values in the data file that only contains the potential outcomes that were observed.

*Note that it is not as straightforward as it may seem to obtain appropriate standard errors/certainty intervals/p-values for this mean difference: While you could run a paired t-test, and obtain some, those do not take into account that the potential outcomes are actually estimated rather than observed. We will ignore this here.

▶ Start with creating separate data sets for the two treatment groups, and running a regression analysis for each group separately:

① Divide groups

```
df1 <- subset(df, DIET==1)
df0 <- subset(df, DIET==0)

# Regression analysis with only people with X=1:
M3.1 <- glm(DISTR.2 ~ + DISTR.1 + as.factor(BLACK) + as.factor(NBHISP)
            + GRADE + SLFHLTH + SLFWGHT + WORKHARD + GOODQUAL
                + PHYSFIT + PROUD + LIKESLF + ACCEPTED + FEELLOVD,
                data=df1)

# Regression analysis with only people with X=0:
M3.0 <- glm(DISTR.2 ~ + DISTR.1 + as.factor(BLACK) + as.factor(NBHISP)
            + GRADE + SLFHLTH + SLFWGHT + WORKHARD + GOODQUAL
              + PHYSFIT + PROUD + LIKESLF + ACCEPTED + FEELLOVD,
                data=df0)
```

② estimate in model for each group

③

▶ Now, obtain estimates for everyone (both for those who we observed $X = 1$ and $X = 0$) for the potential outcome under treatment (i.e., $\hat{Y}_i^1$) and under no treatment (i.e., $\hat{Y}_i^1$):

```
# Obtain a prediction for the outcome using all the cases, based on
# the parameter estimates obtained above and saved in M3.1:
M3.est.Y1 <- predict(M3.1, newdata = df)

# Do the same, but now with the parameters saved in M3.0:
M3.est.Y0 <- predict(M3.0, newdata = df)
```

```
# Alternative DIY method:
# Obtain the parameters from X=0 group
M3.b0 <- as.matrix(M3.0$coeff)
M3.b0
```

```
##                            [,1]
## (Intercept)          0.003312899
## DISTR.1              0.521052495
## as.factor(BLACK)1    0.074201183
## as.factor(NBHISP)1   0.024133444
## GRADE                0.002158534
## SLFHLTH              0.025774548
## SLFWGHT              0.002690032
## WORKHARD            -0.019442600
## GOODQUAL             0.015469074
## PHYSFIT              0.005413525
## PROUD                0.032926130
## LIKESLF              0.022589843
## ACCEPTED             0.015982144
## FEELLOVD             0.041261285
```

```
# Obtain the paramaters from X=1 group
M3.b1 <- as.matrix(M3.1$coeff)
M3.b1
```

```
##                            [,1]
## (Intercept)         -0.056083033
## DISTR.1              0.510368708
## as.factor(BLACK)1    0.073865062
## as.factor(NBHISP)1   0.047967832
## GRADE                0.004551248
## SLFHLTH             -0.003992603
## SLFWGHT              0.001539476
## WORKHARD             0.019817634
## GOODQUAL             0.041481560
## PHYSFIT             -0.021218585
## PROUD                0.057010259
## LIKESLF              0.031256345
## ACCEPTED             0.014469597
## FEELLOVD             0.028748716
```

```
# Create a matrix that contains:
# a) a column with 1's for all individuals (for the intercept)
# b) the observed covariates
Z <- as.matrix(cbind(rep(1,nrow(df)),df[,1:13]))

# Multiple the matrix that contains the observed covariates with
# the vector with parameter estimates for the non-diet group and
# also for the diat group: This gives us the predicted potential
# outcomes for every person when X=0 and when X=1.

M3.est.Y0 <- Z%*%M3.b0
M3.est.Y1 <- Z%*%M3.b1
```

▶ Estimate the average causal effect based on what you have created in the previous (focus on obtaining the point estimate, you do not need to obtain a correct standard error/certainty interval/pvalue).

```
# Look at the predicted potential outcomes
cbind(M3.est.Y0, M3.est.Y1)[1:10,]
```

```
##          [,1]      [,2]
## 1   0.6798196 0.6563903
## 2   0.2418481 0.1672951
## 3   0.6594521 0.6728799
## 4   0.3555863 0.3584581
## 5   1.4298966 1.3315301
## 6   0.4128621 0.3974892
## 7   0.8623319 0.7773935
## 8   0.5531470 0.4660255
## 9   0.4186977 0.4506349
## 10  0.7299912 0.7454811
```

(+) Compute ACE using the "predicted" potential outcomes by computing mean differences.

```
# Estimate the causal effect now, using a t-test
t.test(M3.est.Y0, M3.est.Y1, paired = TRUE, alternative = "two.sided")
```

(as we want mean diff.)

```
##
##  Paired t-test
##
## data:  M3.est.Y0 and M3.est.Y1
## t = 8.2764, df = 5999, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.004470006 0.007244792
## sample estimates:
## mean of the differences
##              0.005857399
```

```
# Although in reality we can only ever observe one potential outcome
# per person, with this technique we obtained an estimate for each
# person's potential outcome under treatment (X=1) and under no
# treatment (X=0). We can now use these estimated potential outcomes
# to compute the ACE by calculating the mean difference.

# It is important to realize the standard error/intervals/p-value that is # obtained wit
h the t-test here (and the standard error) is not correct; #they are based on the assump
tion
# these are observed, rather than estimated scores.
```

▶ **(OPTIONAL)** Above we have used the predicted potential outcomes for everyone. However, one of them is actually observed, and we could use those instead of the predicted potential outcome (i.e., we use the observed fact, and predict only the counterfact). Hence, for individuals in the no treatment condition you use Y_i instead of $\hat{Y}_i^0$, and for those in the treatment condition you use Y_i instead of $\hat{Y}_i^1$ (this is also described around Equation (16) in S&K). Check whether this leads to a different result.

```
# Take the predicted potential outcome for X=0
# and only for those for whom we observed X=0
# do we overwrite the predicted potential outcome
M3b.Y0 <- M3.est.Y0
M3b.Y0[df$DIET==0] <- df$DISTR.2[df$DIET==0] put the observed value back (Y_i^0)

# Do the same for the predicted potential outcome for X=1
M3b.Y1 <- M3.est.Y1
M3b.Y1[df$DIET==1] <- df$DISTR.2[df$DIET==1] put the observed value back (Y_i^1)

# Now do the t-test with these (observed and predicted) potential outcomes:
t.test(M3b.Y0, M3b.Y1, paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  M3b.Y0 and M3b.Y1
## t = 1.1979, df = 5999, p-value = 0.231
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.003728044  0.015442842
## sample estimates:
## mean of the differences
##              0.005857399
```

```
# Although the predicted potential outcomes and the actual observed
# potential outcomes are not exactly the same, using only the predicted
# potential outcomes (like we did first), or using a combination of
# predicated and observed potential outcomes (as we did here), makes
# virtually no difference for the estimated causal effect.
```

3.4 Conclusion

In the methods discussed so far, we rely on extrapolation, which could be a problem. For example, it could happen that we have to extrapolate (e.g., predict potential outcomes) for the treated group based on the untreated group, while these groups have quite different values for their covariates. If so, we can have a lot of uncertainty about whether the parameters based on the untreated group are suitable for our treated group. The fact that we are assuming linear relations in all of the above may make this especially problematic.

To get a first impression of whether this extrapolation issue applies here, use *standardized mean differences* again. In this case, use cut off of 0.3 to indicate a potential problem.

```
# As before:
print(table1, smd=TRUE)
```

```
##                        Stratified by DIET
##                         0           1           SMD
##   n                     4780        1220
##   DISTR.1 (mean (SD))   0.62 (0.42) 0.71 (0.45)  0.214
##   BLACK (mean (SD))     0.26 (0.44) 0.17 (0.38)  0.197
##   NBHISP (mean (SD))    0.15 (0.35) 0.15 (0.36)  0.021
##   GRADE (mean (SD))     9.16 (1.39) 9.37 (1.34)  0.152
##   SLFHLTH (mean (SD))   2.20 (0.93) 2.35 (0.91)  0.171
##   SLFWGHT (mean (SD))   3.19 (0.76) 3.84 (0.70)  0.895
##   WORKHARD (mean (SD)) 2.14 (0.91) 2.05 (0.85)  0.093
##   GOODQUAL (mean (SD)) 1.80 (0.67) 1.84 (0.71)  0.055
##   PHYSFIT (mean (SD))   2.24 (0.93) 2.53 (0.93)  0.320
##   PROUD (mean (SD))     1.76 (0.77) 1.86 (0.79)  0.126
##   LIKESLF (mean (SD))   2.09 (0.99) 2.52 (1.06)  0.420
##   ACCEPTED (mean (SD)) 2.14 (1.00) 2.35 (1.06)  0.207
##   FEELLOVD (mean (SD)) 1.78 (0.83) 1.93 (0.90)  0.172
```

Is the linear extrapolation appropriate? (check SMD > 0.3) rule of thumb when checking linear approximation is problematic

```
# Large standardize mean differences imply that the linear approximation may be
# problematic, as one has to make predictions for the treated using the
# parameters of the untreated far away from the mean of
# the covariates of the untreated, and vice versa (i.e., extrapolation).

# Here several covariates have (absolute) SMDs larger than 0.3, which implies
# linear extrapolation could be a problem. Hence, it may be more appropriate to
# use a techniques based on the propensity score (which all methods below do).
```

4 Controlling for Confounders Z: Model the relation of Z with X

Above, we included the covariates as predictors of the outcome in various ways. As discussed above, these approaches can be problematic, for example due to problems with the linear extrapolations we are making.

There are various alternative techniques that are all based on using *propensity scores*: This is an individual's probability of treatment based on their scores on the covariates (i.e., $P(X = 1|Z = Z_i)$. If we know how likely a person was to receive treatment, we can use this information to mimic a randomized controlled trial (in which everyone has the same probability of receiving treatment). Schafer and Kang consider three common techniques for this:

- *matching*, in which we try to create pairs of a treated and an untreated individual that have the same propensity score

- *inverse probability weighting*, in which we create a pseudo-population that is balanced on the covariates
- *subclassification* or *stratification*, in which we create strata in which there are no (meaningful) differences in the covariates left

First,

4.1 Estimate the propensity scores

We will start with estimating a propensity score for each person in the data set. This score will then be used in the diverse techniques that follow.

▶ To compute a propensity score, run a logistic regression model in which the treatment variable X (which has values 0 and 1) is the outcome variable, and the covariates are the predictors. Make sure to save the probability for each person for scoring 1 on X (here: DIET). You can use the `glm` function from the `stats` package for this.

```
# Run the logistic regression analysis
logreg <- glm(DIET ~ DISTR.1 + as.factor(BLACK) + as.factor(NBHISP)
                     + GRADE + SLFHLTH + SLFWGHT + WORKHARD + GOODQUAL
                         + PHYSFIT + PROUD + LIKESLF + ACCEPTED + FEELLOVD,
                     family = binomial(), data = df)

# Obtain a prediction of the probability of treatment (i.e., DIET=1)
ps <- predict(logreg, type = "response")

# Add this predicted probability to the datafile
df$ps <- ps

# Look at the datafile
round(df[1:10,], 2)
```

DIET (DV) ~ all covariates (IV)

~ prob. of getting DIET=1 ~ 'propensity score'

```
##    DISTR.1 BLACK NBHISP GRADE SLFHLTH SLFWGHT WORKHARD GOODQUAL PHYSFIT PROUD
## 1     0.47     0       0    10       4       4        2        2       2     2
## 2     0.05     0       0     8       2       3        1        1       2     1
## 3     0.63     0       0     8       2       3        2        2       2     2
## 4     0.16     1       0     9       1       4        2        1       2     1
## 5     1.79     1       0     9       4       4        1        2       2     2
## 6     0.32     0       0    10       1       3        1        1       2     1
## 7     1.11     0       0    10       2       3        1        1       2     1
## 8     0.32     0       0     9       3       4        2        1       3     2
## 9     0.26     0       0    10       1       4        2        2       2     2
## 10    0.89     1       0    10       2       2        2        2       1     1
##    LIKESLF ACCEPTED FEELLOVD DIET DISTR.2   ps
## 1        5        2        2    1    0.40 0.50
## 2        1        2        1    1    0.09 0.10
## 3        3        2        2    0    0.69 0.15
## 4        2        2        1    1    0.08 0.23
## 5        3        3        2    0    1.41 0.32
## 6        4        1        1    0    0.52 0.28
## 7        2        2        2    0    1.32 0.21
## 8        2        3        3    0    0.29 0.32
## 9        1        3        2    0    0.43 0.24
## 10       1        1        1    0    1.76 0.02
```

```
# The last column in the datafile now contains the predicted
# probability of being treated, based on the covariates.
```

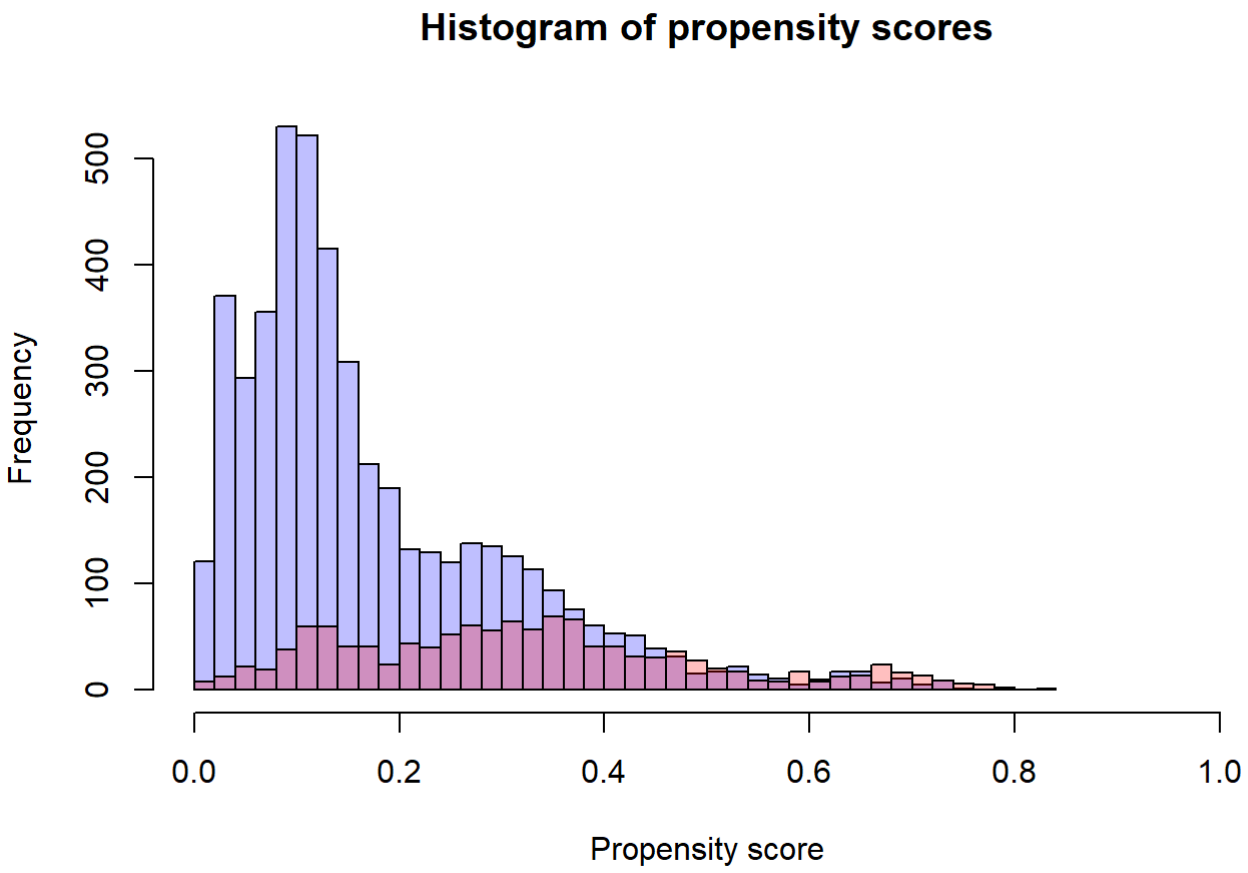
Once got the propensity score, we need to check the distribution of propensity scores for each group → & check the overlap!

★ Now that we have the propensity scores, we should first consider the distribution of propensity scores in each of the treatment groups separately, to determine whether there is overlap between the propensity scores of the two groups.

▶ Make a plot that includes a histogram for the propensities of the treated, and a histogram for the propensity scores of the untreated to evaluate this. Discuss what you see in the plot, and what this may imply for our causal inferences.

```
# Create separate data files (which now include the propensity scores)
# for those treated and those not treated:
df1 <- df[ which(df$DIET == 1), ]
df0 <- df[ which(df$DIET == 0), ]

# Create histograms, then plot one and add the other:
hist0 <- hist(df0$ps, breaks=30, plot=FALSE)
hist1 <- hist(df1$ps, breaks=30, plot=FALSE)
plot( hist0, col=rgb(0,0,1,1/4), xlim=c(0,1),
        xlab="Propensity score",
      main="Histogram of propensity scores")
plot( hist1, col=rgb(1,0,0,1/4), xlim=c(0,1), add=T)
```




```
# What we see is that the distributions of propensity scores of the
# two groups seem to overlap well, even in the tails.
# If this would not be the case, that would be an indication that the
# "assumption of positivity" is violated.
# That is, at each possible combination of the covariates, which is
# now summarized with the propensity score, there should be both
# treated and non-treated individuals in our sample.
```

positivity assumption definition

Now that we have determined the propensity scores and their distributions for the the treated and the non-treated overlap well, we can make use of these scores within different techniques.

4.2 Method 4: *Matching*

Matching based on the propensity scores is the first technique in this category that we consider. It is based on finding people in the both treatment groups that have similar propensity scores: The idea is that these individuals are comparable on the entire set of covariates, and can thus be considered randomly assigned to the two condition.

In practice, this is done by taking a person from the smallest of the two groups (here the group for which DIET=1, the treated), and finding a person in the other group that is most like this person in terms of their propensity score. We can do this using the function `matchit()` from the package `MatchIt` in R. Please note that this will give us slightly different results than those obtained by S&K.

To run the matching function, we plug in the same expression as we used when we obtained the propensity scores ourselves, and use the method "nearest" (Check out the helpfil of function `matchit` to get an impression of what it does):

```
library(MatchIt)
matchdat <- matchit(DIET ~ DISTR.1 + as.factor(BLACK) + as.factor(NBHISP)
            + GRADE + SLFHLTH + SLFWGHT + WORKHARD + GOODQUAL
            + PHYSFIT + PROUD + LIKESLF + ACCEPTED + FEELLOVD,
        method = "nearest",  data = df)
```

▶ Describe/Interpret the information that is reported in the output (matchdat).

```
matchdat
```

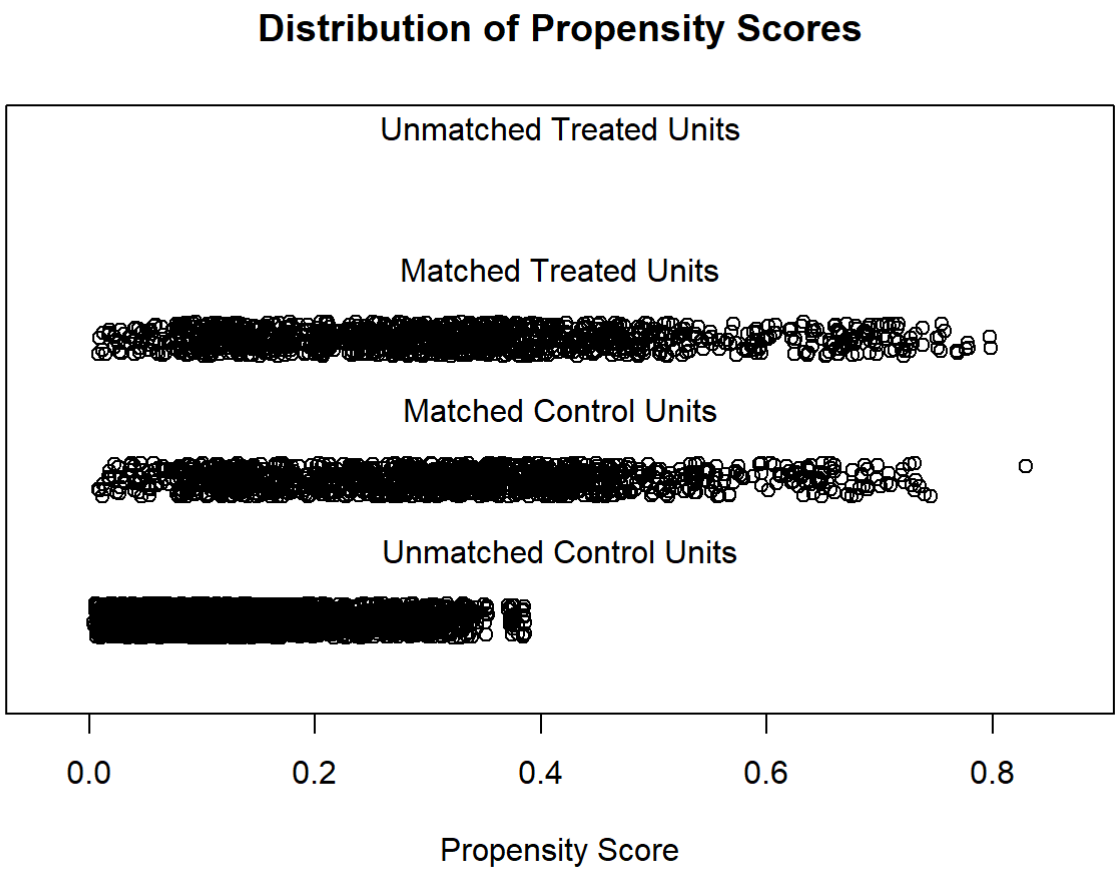
```
## A matchit object
##  - method: 1:1 nearest neighbor matching without replacement
##  - distance: Propensity score
##              - estimated with logistic regression
##  - number of obs.: 6000 (original), 2440 (matched)
##  - target estimand: ATT
##  - covariates: DISTR.1, as.factor(BLACK), as.factor(NBHISP), GRADE, SLFHLTH, SLFWGHT,
WORKHARD, GOODQUAL, PHYSFIT, PROUD, LIKESLF, ACCEPTED, FEELLOVD
```

```
# The results indicate that the original sample consisted of
# 6000 individuals, and that the matched sample consists of
# 2440 individuals. This is because the number of treated individuals
# is 1220, and these were all matched with a non-treated person.

# Note that a different model for the propensity scores (e.g., including
# interaction terms between two covariates, or non-linear relations by
# squaring covariates), would lead to different propensity scores, and
# these may subsequently lead to different matches. Hence, it is really
# model dependent!
```

▶ There are two useful plotting options regarding the propensity scores of our matched pairs: `plot(matchdat,type="jitter")` and `plot(matchdat,type="hist")`. Get both plots, and describe what they represent.

```
plot(matchdat,type="jitter")
```

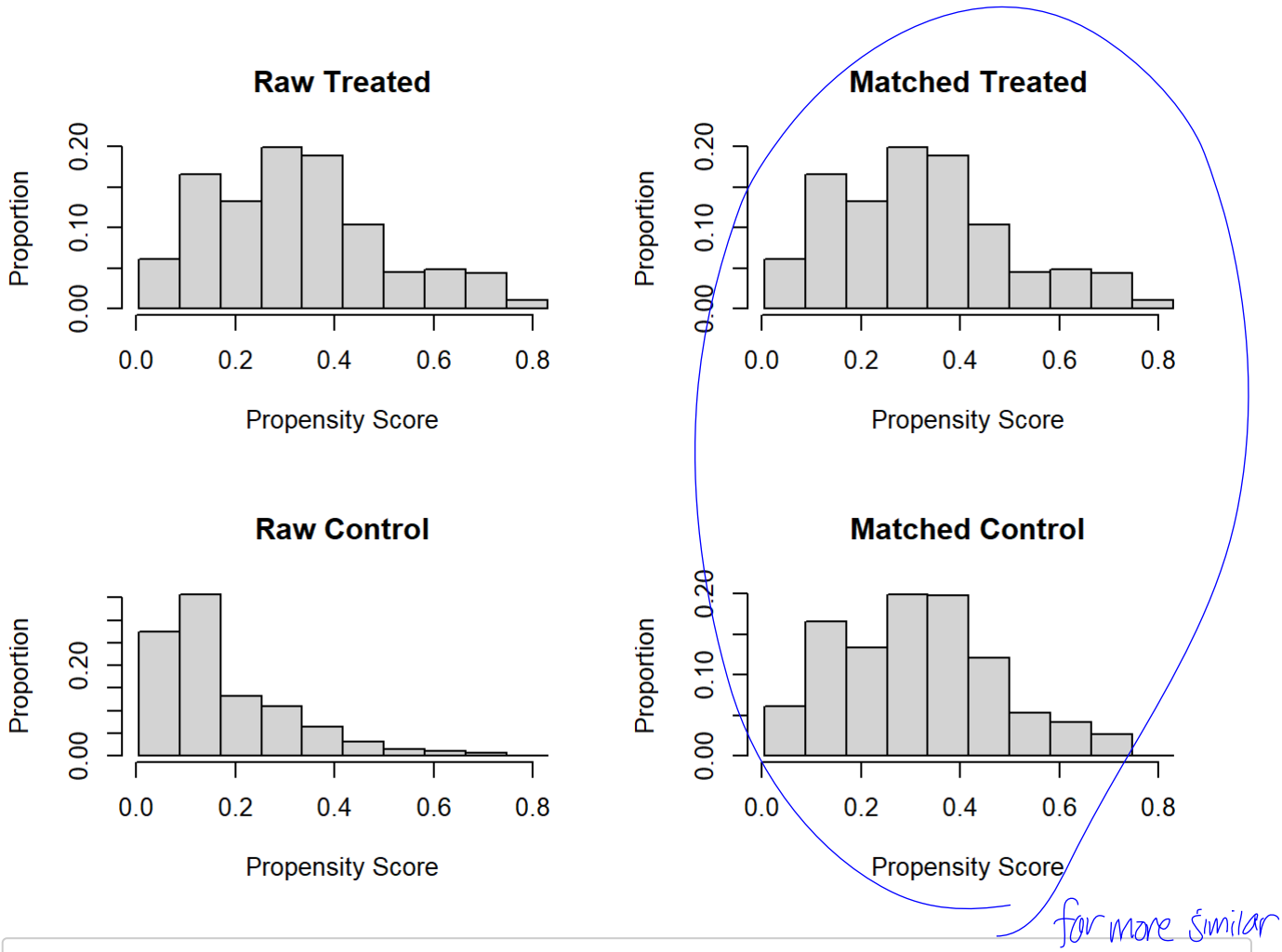



```
## [1] "To identify the units, use first mouse button; to stop, use second."
```

```
## integer(0)
```

```
# This is a plot of the propensity scores of four different subgroups
# from our original data file:
# 1) Those from the treatment group for whom there was no match
# 2) Those from the treatment group for whom there was a match
# 3) Those from the control group for whom there was a match
# 4) Those from the control group for whom there was no match
# It shows the first group is empty; it shows the last group
# has relatively low propensity scores; the middle two groups
# seem pretty similar in terms of their propensity score
# distribution (as expected, as these are the matched cases).
```

```
plot(matchdat,type="hist")
```




```
# This plot shows the histograms of the propensity scores
# of the two treatment groups in the original dataset on the left
# and for the matched groups on the right; it shows that the latter
# are far more similar than the former (as one would expect).
```

▶ To do the analysis on the matched cases only, we need to create a new data file with only the matched cases, using: `df.match <- match.data(matchdat)` . Create the Table 1 for this matched data set. What can you conclude?

```
# To extract the new datafile from the original one:
df.match <- match.data(matchdat)

# Create Table 1:
table1 <- CreateTableOne(vars=c("DISTR.1","BLACK", "NBHISP", "GRADE",
                      "SLFHLTH", "SLFWGHT", "WORKHARD", "GOODQUAL",
                          "PHYSFIT", "PROUD", "LIKESLF", "ACCEPTED",
                      "FEELLOVD"), strata="DIET", data=df.match,
                        test=FALSE)
print(table1, smd=TRUE)
```

```
##                        Stratified by DIET
##                         0            1            SMD
##   n                     1220         1220
##   DISTR.1 (mean (SD))  0.71 (0.45) 0.71 (0.45)  0.007
##   BLACK (mean (SD))    0.18 (0.38) 0.17 (0.38)  0.004
##   NBHISP (mean (SD))   0.16 (0.36) 0.15 (0.36)  0.007
##   GRADE (mean (SD))    9.37 (1.35) 9.37 (1.34)  0.002
##   SLFHLTH (mean (SD))  2.36 (0.95) 2.35 (0.91)  0.011
##   SLFWGHT (mean (SD))  3.82 (0.69) 3.84 (0.70)  0.033
##   WORKHARD (mean (SD)) 2.07 (0.86) 2.05 (0.85)  0.022
##   GOODQUAL (mean (SD)) 1.81 (0.65) 1.84 (0.71)  0.049
##   PHYSFIT (mean (SD))  2.53 (0.97) 2.53 (0.93)  0.007
##   PROUD (mean (SD))    1.85 (0.77) 1.86 (0.79)  0.011
##   LIKESLF (mean (SD))  2.46 (1.05) 2.52 (1.06)  0.057
##   ACCEPTED (mean (SD)) 2.33 (1.03) 2.35 (1.06)  0.023
##   FEELLOVD (mean (SD)) 1.92 (0.87) 1.93 (0.90)  0.010
```

```
# Note that the two groups now each have 1220 cases (compared to 4789 and
# 1220 respectively before). This is because we are now working with only
# matched cases, and there was a match in the non-treated group for every
# treated person.

# The table shows that the standardized mean differences are all
# quite small in this matched data set; this means the two groups
# are now very similar on the covariates, just as one would
# expect in an RCT. This is good news!

# Hence, matching seems to mimic an RCT here (at least with respect
# to observed covariates; note there may still be unobserved confounding).
```

▶ Subsequently, investigate with a t-test whether the means on the outcome variable DISTR.2 differ among the matched cases.

```
# Do a t-test on the matched data file:
t.test2 <- t.test(DISTR.2 ~ DIET, df.match)
t.test2
```

```
##
##  Welch Two Sample t-test
##
## data:  DISTR.2 by DIET
## t = 1.1383, df = 2438, p-value = 0.2551
## alternative hypothesis: true difference in means between group 0 and group 1 is not e
qual to 0
## 95 percent confidence interval:
##  -0.01603559  0.06041264
## sample estimates:
## mean in group 0 mean in group 1
##       0.7253279       0.7031393
```

```
# Note the means per group are included at the bottom.
# We can also compute the mean difference, using:
t.test2$estimate[2] - t.test2$estimate[1]
```

```
## mean in group 1
##     -0.02218852
```

```
# The ACE is now estimated to be: -0.0222
# It is not significantly different from zero, according to the t-test.
```

▶ Compare this result to the mean comparison you did at the start; explain why the mean differences that you have just determined is an estimate of the ACE1 rather than of the ACE.

```
# Initially, the difference was 0.0596, meaning that those who diet (X=1)
# experience MORE distress than those who do not diet (X=0).

# Here the mean difference between the matched cases is 0.703-0.725=-0.022,
# meaning that the distress for those who did diet (X=1) is actually
# LOWER than that of those who did not diet (X=0).

# Note that the matched cases are based on all the girls in our initial sample
# with X=1; hence, we now have the ACE for the treated. This implies that among
# all those girls who are LIKELY to diet (X=1), actually dieting (X=1) seems
# to result in LESS distress than not dieting (X=0).

# However, the difference is not a significant difference.
```

4.3 Method 5: *IPW*

We can also use inverse probability weighting (IPW). In this case, the estimated propensity scores $\hat{\pi}_i$ are used to determine the probability that an individual would have received the treatment that they received:

- for the treated ($X_i = 1$) this is simply $\hat{\pi}_i$
- for the non-treated ($X_i = 0$) this is $1 - \hat{\pi}_i$.

Next, we use these probabilities as weights, by taking their inverse: That way, a case that received a treatment that she was very likely to receive, will get a small weight, while a case that received a treatment that she was very unlikely to receive, will get a large weight. Thus, the inverse probability weight indicates the number of persons from the population that this person represents. For the treated, this weight is $1/\hat{\pi}_i$; for the untreated, it is $1/(1 - \hat{\pi}_i)$.

▶ Compute the ACE using this IPW (see Equation (20) in S&K).

```
Y <- df$DISTR.2
X <- df$DIET
mu1hat <- sum( X*Y/ps ) / sum(X/ps)
mu0hat <- sum( (1-X)*Y/(1-ps) ) / sum((1-X)/(1-ps))
mu1hat - mu0hat
```

```
## [1] -0.005064163
```

```
# The latter difference is the estimate of the ACE.
# Obtaining a p-value for this, is tricky. In the appendix of S&K, ways of computing
# standard errors for the estimate are provided.
```

```
# For those who are interested, a more sophisticated
# way of getting the ACE using IPW is given below; it is based on
# using the package survey in R.
library(survey)
library(tableone)
weight<-ifelse(df$DIET==1,1/(df$ps),1/(1-df$ps))
weighteddata<-svydesign(ids = ~ 1, data =df, weights = ~ weight)
weightedtable <-svyCreateTableOne(vars=c("DISTR.1","BLACK", "NBHISP", "GRADE",
                      "SLFHLTH", "SLFWGHT", "WORKHARD", "GOODQUAL",
                          "PHYSFIT", "PROUD", "LIKESLF", "ACCEPTED",
                      "FEELLOVD"), strata = "DIET",
                      data = weighteddata, test = FALSE)
print(weightedtable, smd = TRUE)
```

```
##                         Stratified by DIET
##                          0              1              SMD
##   n                      6014.23        6368.45
##   DISTR.1 (mean (SD))       0.64 (0.43)    0.63 (0.43)  0.028
##   BLACK (mean (SD))         0.24 (0.43)    0.25 (0.44)  0.035
##   NBHISP (mean (SD))        0.15 (0.35)    0.13 (0.33)  0.061
##   GRADE (mean (SD))         9.20 (1.38)    9.26 (1.41)  0.037
##   SLFHLTH (mean (SD))       2.23 (0.94)    2.25 (0.91)  0.020
##   SLFWGHT (mean (SD))       3.33 (0.80)    3.15 (1.00)  0.196
##   WORKHARD (mean (SD))      2.12 (0.90)    2.14 (0.86)  0.017
##   GOODQUAL (mean (SD))      1.81 (0.67)    1.79 (0.68)  0.036
##   PHYSFIT (mean (SD))       2.30 (0.95)    2.31 (0.90)  0.014
##   PROUD (mean (SD))         1.78 (0.77)    1.77 (0.76)  0.018
##   LIKESLF (mean (SD))       2.18 (1.02)    2.14 (1.00)  0.045
##   ACCEPTED (mean (SD))      2.18 (1.01)    2.16 (1.02)  0.024
##   FEELLOVD (mean (SD))      1.81 (0.84)    1.79 (0.83)  0.028
```

except for SLFWGHT, all are under < 0.1

```
# This shows that in the pseudo-population that was created
# using IPW, the two groups do not differ much any more on
# most of the covariates; however, there is one covariate
# SLFWGHT, whose standardized mean difference exceeds the
# rule of thumb value of 0.1. This implies that IPW is not
# entirely successful here in mimicing an RCT.
# We nevertheless proceed, but it would be better to check
# whether there are for instance very large weights, such
# that there are cases that have a disproportional effect
# on the results, and to fix this first in some way (for instance by
# replacing their weight by a large
# yet not excessive value).
# For now we proceed, using a function from the package survey.
msm <- svyglm(DISTR.2 ~ DIET, design = weighteddata)
summary(msm)
```

```
##
## Call:
## svyglm(formula = DISTR.2 ~ DIET, design = weighteddata)
##
## Survey design:
## svydesign(ids = ~1, data = df, weights = ~weight)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.660949   0.007029  94.032   <2e-16 ***
## DIET        -0.005064   0.030394  -0.167    0.868
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.2397251)
##
## Number of Fisher Scoring iterations: 2
```

```
confint(msm)
```

```
##                   2.5 %     97.5 %
## (Intercept)  0.64716947 0.67472808
## DIET        -0.06464777 0.05451944
```

```
# This shows that the effect of dieting is now estimated to be
# -0.005 (SE=0.030), which is not significantly different
# from zero.
```

4.4 Method 6: *Subclassificiation*

Subclassification, also known as stratification, is a method that consists of creating classes (strata) based on the propensity scores. The idea is that the individuals within each stratum are rather similar with respect to their propensity score, and thus with respect to the entire set of covariates on which the propensity score is based; if the covariates are well balanced within each stratum, this is a way to mimic an RCT within each stratum. By subsequently estimating the ACE in each stratum (using a mean comparison such as Method 1, or an ANCOVA or regression analysis such as Method 2), we can determine the causal effect for individuals who are similar with regard to the entire set of covariates (as these are used to determine the propensity scores).

▶ Begin with creating five strata based on the propensity scores (for instance, use the function cut() in R); each stratum should contain 20% of the (total number of) observations.

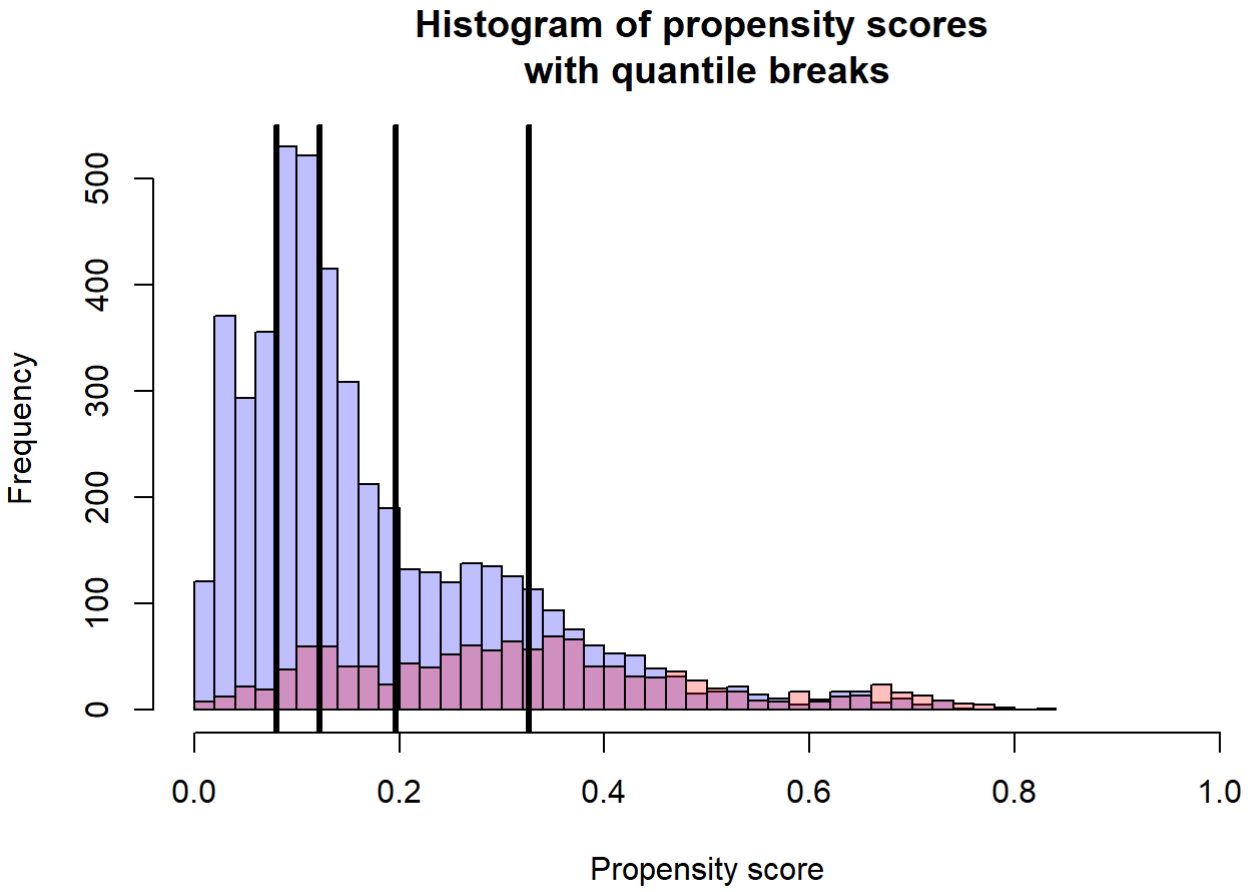
```
df$stratum <- cut(df$ps,
                  breaks=c(quantile(df$ps, probs=seq(0,1,0.2))),
                  labels=seq(1:5),
                  include.lowest=TRUE)

# We can also make a plot of these quantiles; this is based on
# using the same histrogram we had before, now adding vertical
# lines for where the breaks of the strat are.

plot( hist0, col=rgb(0,0,1,1/4), xlim=c(0,1),
    xlab="Propensity score", main="Histogram of propensity scores \nwith quantile break
s")
plot( hist1, col=rgb(1,0,0,1/4), xlim=c(0,1), add=T)

br <- c(quantile(df$ps, probs=seq(0,1,0.2)))

abline(v=br[2],col="black",lwd=3)
abline(v=br[3],col="black",lwd=3)
abline(v=br[4],col="black",lwd=3)
abline(v=br[5],col="black",lwd=3)
```




```
# This shows that especially the fifth stratum is
# very wide (in the paper by Schafer and Kang they
# decide to further split the fourth stratum in two
# groups, and the fifth in four groups, because these
# are rather wide).

# We could also further investigate whether we need more strata
# by looking at the standardized mean differences in each stratum;
# these should be small! as the idea is that each stratum can be
# thought of as an RCT in which the assignment to treatment is
# random, and thus does not depend on any of the covariates.
```

Whether we need more strata.. → look at SMD again, these should be small!

(2)

▶ Next, compute the ACE in each stratum, taking the mean difference.

```r
# We perform a t-test in each stratum.

results <- matrix(,5,1)

for (quintiles in c(1:5)) {
  t.test3 <- t.test(DISTR.2 ~ DIET, data = df[which(df$stratum==quintiles),])
  print(t.test3)
  # Difference in means:
  results[quintiles,1] <- t.test3$estimate[2] - t.test3$estimate[1]
}
```

```
##
##  Welch Two Sample t-test
##
## data:  DISTR.2 by DIET
## t = -0.64615, df = 64.606, p-value = 0.5205
## alternative hypothesis: true difference in means between group 0 and group 1 is not e
qual to 0
## 95 percent confidence interval:
##  -0.18089062  0.09246131
## sample estimates:
## mean in group 0 mean in group 1
##       0.5672608       0.6114754
##
##
##  Welch Two Sample t-test
##
## data:  DISTR.2 by DIET
## t = -0.066853, df = 117.53, p-value = 0.9468
## alternative hypothesis: true difference in means between group 0 and group 1 is not e
qual to 0
## 95 percent confidence interval:
##  -0.10149097  0.09486247
## sample estimates:
## mean in group 0 mean in group 1
##       0.5812489       0.5845631
##
##
##  Welch Two Sample t-test
##
## data:  DISTR.2 by DIET
## t = -1.7668, df = 197.78, p-value = 0.0788
## alternative hypothesis: true difference in means between group 0 and group 1 is not e
qual to 0
## 95 percent confidence interval:
##  -0.15963403  0.00876165
## sample estimates:
## mean in group 0 mean in group 1
##       0.6600766       0.7355128
##
##
##  Welch Two Sample t-test
##
## data:  DISTR.2 by DIET
```

```
## t = 2.0353, df = 665.97, p-value = 0.04222
## alternative hypothesis: true difference in means between group 0 and group 1 is not e
qual to 0
## 95 percent confidence interval:
##  0.001992495 0.111069883
## sample estimates:
## mean in group 0 mean in group 1
##       0.6792056       0.6226744
##
##
##  Welch Two Sample t-test
##
## data:  DISTR.2 by DIET
## t = 1.2039, df = 1174.8, p-value = 0.2289
## alternative hypothesis: true difference in means between group 0 and group 1 is not e
qual to 0
## 95 percent confidence interval:
##  -0.02164432  0.09038353
## sample estimates:
## mean in group 0 mean in group 1
##       0.8102329       0.7758633
```

```
# This shows that the results differ per stratum; only in stratum 4 do we find a signifi
cant difference.
```

3

▶ Subsequently, you can compute the overall ACE by taking the average of the stratum-specific ACE's (weighted by the stratum size).

```
# Note that since our five strata are based on quantiles, the sample
# size of each stratum will be the same (ie 1/5th of the total sample size)
# such that each stratum-specific ACE adds equally to the total.
# Note that this also means that our ACE estimate will differ somewhat
# from the ACE estimate reported in Table 6 by Schafer and Kang, as they
# had further divided the fifth stratum.
# To get the ACE, we simply take the mean of the stratum-specific ACEs:
mean(results[,1])
```

```
## [1] 0.006412859
```

```
# Note further that we do not have an SE for this estimate
# (this is a bit more complicated to obtain). The aim is however you have # a main idea
 of how people are trying to control for confounders, and #estimate actual causal effect
s!
```

Conclusion

The three methods that we considered in this category are all based on using the propensity score, that is, the probability of being treated given the covariates. The goal of these techniques is to somehow mimic the situation we get in an RCT, where the probability of treatment is independent of the covariates. In matching, this is done by creating pairs of a treated and an untreated person who have (almost) identical propensity scores, resulting in a smaller but balanced dataset; in inverse probability weighting, this is done by weighing each person's observation by their inverse probability of received treatment, thereby creating a balanced pseudo-population; and in subclassification this is done by creating strata based on the propensity scores such that within each stratum the covariates are balanced. In each approach, we should check whether it balances the covariates, for instance, by considering the standardized mean differences (for other options, see for instance: Austin, P. c. (2009). Balance diagnostics for comparing the distribution of baseline covariates between treatment groups in propensity-score matched samples. *Statistics in Medicine, 28*, 3093-3107. https://doi.org/10.1002/sim.3697 (https://doi.org/10.1002/sim.3697)).

5 Overall Conclusion

By now you probably see that the number of ways that we can account for covariate imbalance is about infinite. Ignoring the covariates (Method 1), is obviously not a good idea; hence, deciding which covariates to inlcude is a first important step, which we have discussed last week.

But even if we have decided which baseline covariates we need to account for, the options are almost limitless. In the ANCOVA and regression estimation procedures (Methods 2 and 3), we can add interactions and non-linear effects, for instance through using spline fitting. In all the other techniques that are based on propensity scores, we can also consider interactions and non-linear effects of covariates when predicting the log odds. Moreover, when using matching techniques (Method 4), we can choose to have more than one match per person, and we can choose between matching with or without replacement. In inverse probability weigthing (Method 5), we can decide to standardize the weights or not, and there are different ways to handle outliers. For subclassification (Method 6), we have to decide how many strata to create, and whether to further subdivide wide strata or not. The final category of techniques discussed by S&K that we did not cover (Methods 7, 8, and 9), is based on combinations of the other techniques, and therefor allow for even more researchers degrees of freedom.

To make matters worse, different methods and different choices may lead to different results and even different (substantive) conclusions, as we have also seen here. Unfortunately, there is no one method that always performs best under all circumstances. Hence, it is advisable to apply multiple methods to see whether the results are

robust, or that they contradict each other. Ideally, one would combine different studies, and methods that rely on different assumptions to eventually get a clearer picture of what assumptions are problematic, and eventually of the true causal effect. This can never succeed if we are not open about our goals (e.g., do causal inference), and our assumptions and choices.

To make informed decisions on what to do—and what to avoid—under particular circumstances, it can be very helpful to check out simulation studies, such as the one by Schafer and Kang, or the more recent study by Goetghebeur, le Cessie, Stavola, Moodie and Waernbaum (2020. Formulating causal questions and principled statistical answers. *Statistics in Medicine, 39*, 4922-4948. https://doi.org/10.1002/sim.8741 (https://doi.org/10.1002/sim.8741)), or to perform a simulation study yourself to evaluate particular situations.

Finally, it is important to always keep in mind that all the techniques discussed here were developed under the assumption of no unobserved confounding. This means that all relevant confounders were assumed to be included as covariates. We also assumed only confounders were included, an no colliders/mediators, for example. Sensitivity analysis is a valuable approach to investigate how the effect of unobserved confounding can be further investigated (not covered in this course; for more information see for instance: Blackwell, M. (2013). A selection bias approach to sensitivity analysis for causal effects. *Political Analysis, 22*, 169-182. https://doi.org/10.1093/pan/mpt006 (https://doi.org/10.1093/pan/mpt006)).
