36-401 HW 8

36-401 Section B

Question 1:

First, we want to show that:

$$\mathbb{E}[Y(a)] = \mathbb{E}\left[\frac{YI(A=a)}{\pi(A \mid X)}\right]$$

We know that $\mathbb{E}[Y(a)] = \int \mu(x,a)p(x)dx$, so it suffices to show that $\mathbb{E}\left[\frac{YI(A=a)}{\pi(A|X)}\right] = \int \mu(x,a)p(x)dx$ Let's do some algebra:

$$\begin{aligned}
\mathbb{E}\left[\frac{YI(A=a)}{\pi(A \mid X)}\right] &= \sum_{a'} \int \mathbb{E}\left[\frac{YI(A=a)}{\pi(A \mid X)} \mid X=x, A=a'\right] p(A=a', X=x)dx \\
&\hspace{20em} \text{(law of total probability)} \\
&= \int \mathbb{E}\left[\frac{Y}{\pi(A \mid X)} \mid X=x, A=a\right] p(A=a, X=x)dx \\
&\hspace{14em} (I(a'=a) = 1 \text{ iff } a'=a, 0 \text{ otherwise}) \\
&= \int \mathbb{E}\left[\frac{Y}{\pi(A \mid X)} \mid X=x, A=a\right] p(A=a \mid X=x)p(x)dx \qquad \text{(chain rule)} \\
&= \int \mathbb{E}\left[\frac{Y}{\pi(A \mid X)} \mid X=x, A=a\right] \pi(a \mid x)p(x)dx \\
&= \int \mathbb{E}\left[\frac{Y}{\pi(a \mid x)} \cdot \pi(a \mid x) \mid X=x, A=a\right] p(x)dx \qquad \text{(linearity of expectation)} \\
&= \int \mathbb{E}\left[Y \mid X=x, A=a\right] p(x)dx \\
&= \int \mu(x,a)p(x)dx \qquad \text{(definition of } \mu(x,a))
\end{aligned}$$

completing the proof.

Now, we'd like to estimate $\mathbb{E}[Y(a)]$ by using the IPW formula. We can estimate by sampling. More concretely, we'd like to average all the points from our dataset:

$$\mathbb{E}[Y(a)] = \mathbb{E}\left[\frac{YI(A=a)}{\pi(A \mid X)}\right] \approx \frac{1}{n}\sum_{i=1}^{n} \frac{Y_i I(A_i=a)}{\pi(A_i \mid X_i)}$$

This leads us to the two step process:

1. Estimate $\pi(a \mid x)$ by some plugin classifier (ex. logistic regression)
2. Compute $\frac{1}{n}\sum_{i=1}^{n} \frac{Y_i I(A_i=a)}{\pi(A_i|X_i)}$. As a simplification, we can also only count points with $A_i = a$: $\frac{1}{n}\sum_{i,A_i=a} \frac{Y_i}{\pi(a|X_i)}$

HW8 - 36401

[redacted]

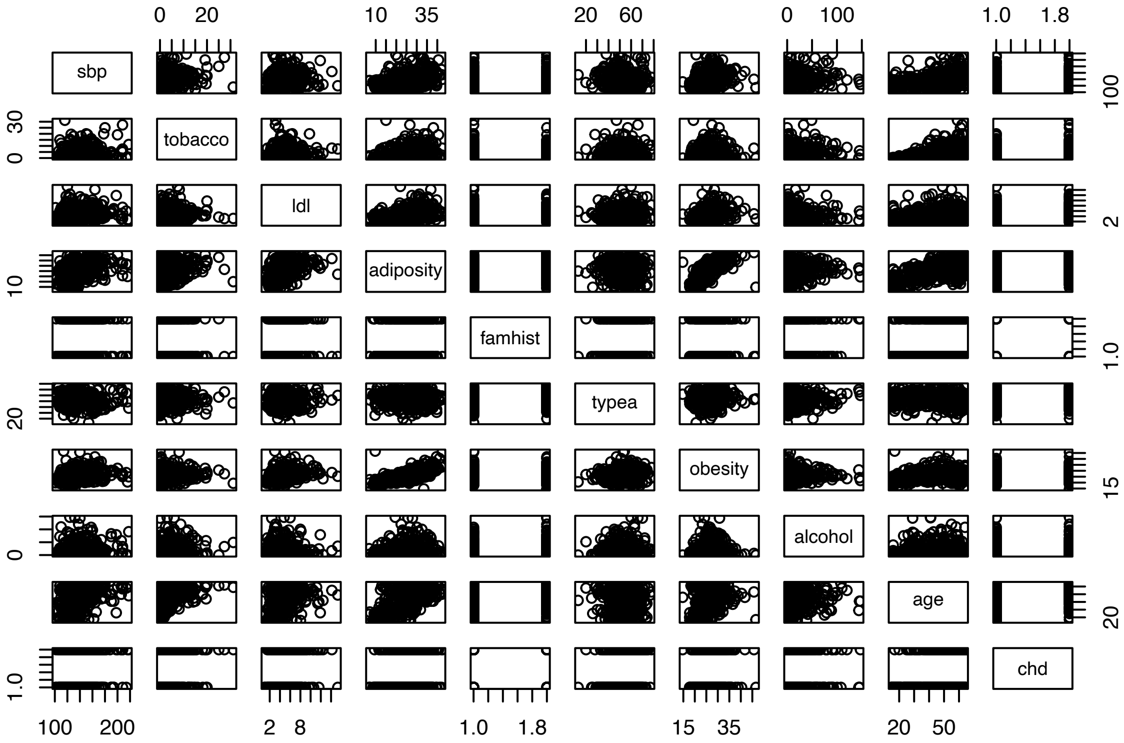
Problem 2

Let's load the data:

```
## 'data.frame':    462 obs. of  10 variables:
##  $ sbp      : int  160 144 118 170 134 132 142 114 114 132 ...
##  $ tobacco  : num  12 0.01 0.08 7.5 13.6 6.2 4.05 4.08 0 0 ...
##  $ ldl      : num  5.73 4.41 3.48 6.41 3.5 6.47 3.38 4.59 3.83 5.8 ...
##  $ adiposity: num  23.1 28.6 32.3 38 27.8 ...
##  $ famhist  : Factor w/ 2 levels "Absent","Present": 2 1 2 2 2 2 1 2 2 2 ...
##  $ typea    : int  49 55 52 51 60 62 59 62 49 69 ...
##  $ obesity  : num  25.3 28.9 29.1 32 26 ...
##  $ alcohol  : num  97.2 2.06 3.81 24.26 57.34 ...
##  $ age      : int  52 63 46 58 49 45 38 58 29 53 ...
##  $ chd      : Factor w/ 2 levels "0","1": 2 2 1 2 2 1 1 2 1 2 ...
```

Part A

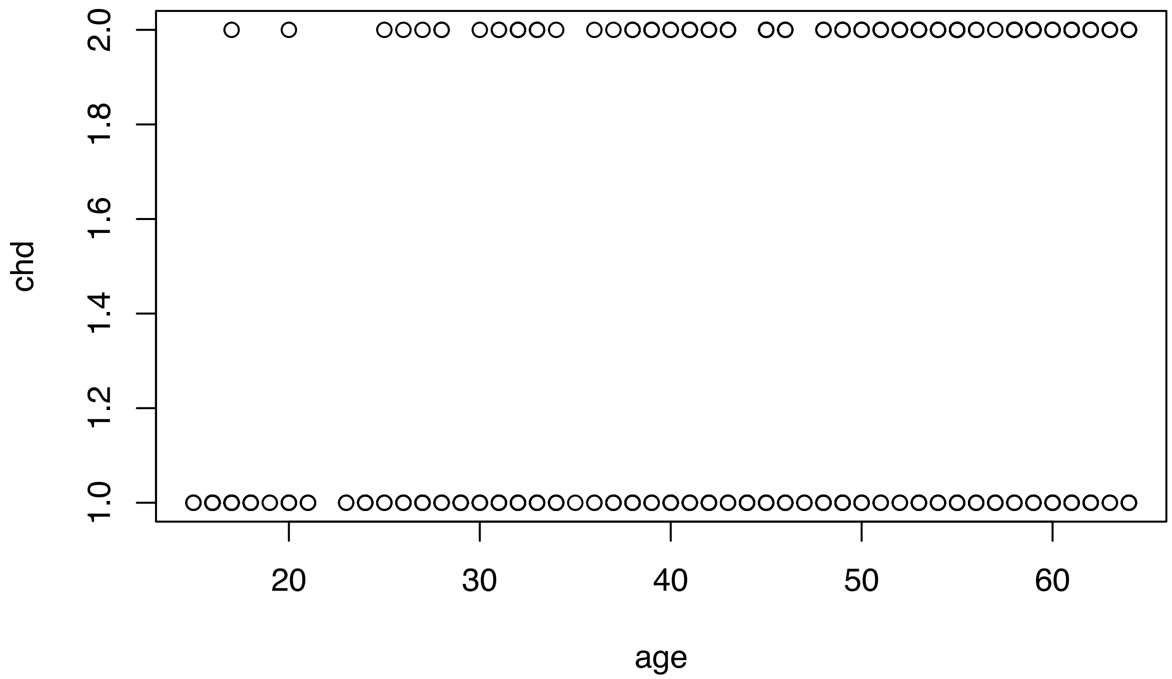
Let's have a look first at our pairs plot:



Most of my conclusions will be based around age, as this is what we want to eventually find the causal effect of on our response (chd). In general, we first find that age does covary with some of our confounders. Notably, adiposity covaries clearly, while alcohol and systolic blood pressure also covary with age.

Many of our other variables don't seem to covary strongly with each other, implying that our set of covariates is quite robust. This supports our assumption that we might have measured all confounding variables.

Let's finish off by looking at a plot of chd vs age, before we do some causal analysis:



It seems that higher age tends to lead to more coronary heart disease. Let's see if this is still the case after our causal analysis:

Part B

Let's do our fit:

```
##
## Call:
## glm(formula = chd ~ ., family = "binomial", data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7781  -0.8213  -0.4387   0.8889   2.5435
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -6.1507209  1.3082600  -4.701 2.58e-06 ***
## sbp             0.0065040  0.0057304   1.135 0.256374
## tobacco         0.0793764  0.0266028   2.984 0.002847 **
## ldl             0.1739239  0.0596617   2.915 0.003555 **
## adiposity       0.0185866  0.0292894   0.635 0.525700
## famhistPresent  0.9253704  0.2278940   4.061 4.90e-05 ***
## typea           0.0395950  0.0123202   3.214 0.001310 **
## obesity        -0.0629099  0.0442477  -1.422 0.155095
## alcohol         0.0001217  0.0044832   0.027 0.978350
## age             0.0452253  0.0121298   3.728 0.000193 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 596.11  on 461  degrees of freedom
## Residual deviance: 472.14  on 452  degrees of freedom
## AIC: 492.14
##
## Number of Fisher Scoring iterations: 5
```

First, we see that family history and age are the most predictive covariates by far. Our coefficient for age is also positive, implying that older people tend to correlate with higher levels of chd. Because we've already conditioned on our confounders, we can possibly interpret this coefficient causally.

Some other confounders seem to also be decently predictive, such as tobacco, ldl, and type-a behavior.

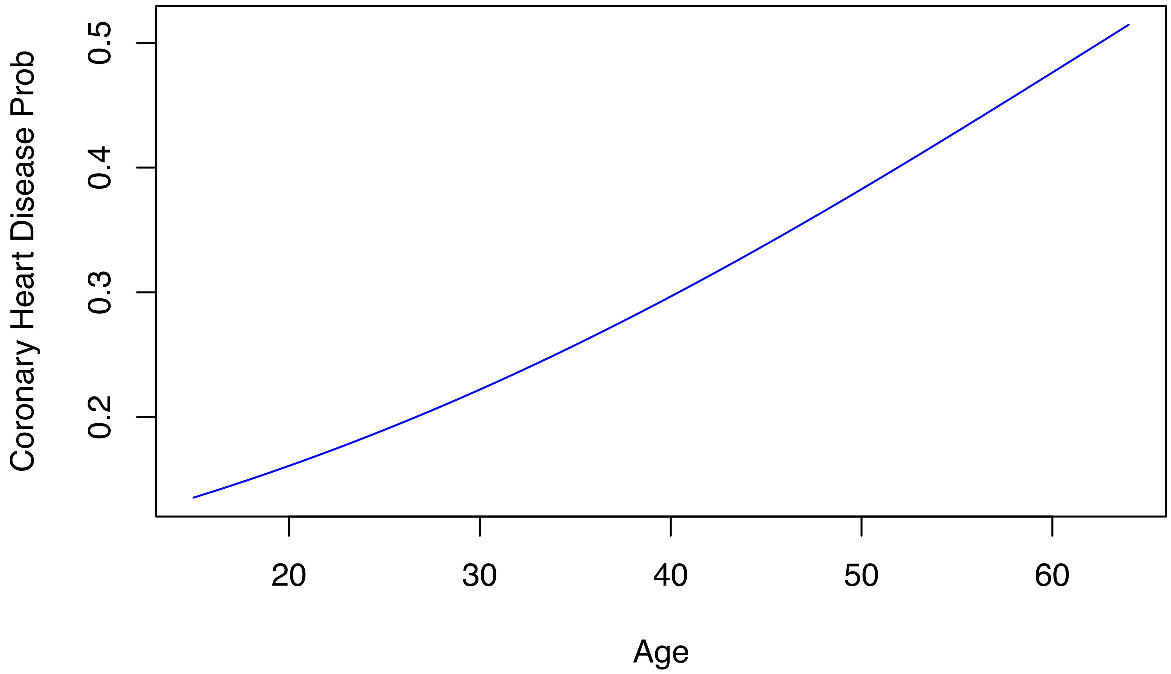
Let's have a look at our accuracy. This isn't on a held out test set, so it's not really prediction error, or even reported with confidence intervals. It's just for us to get an idea of if logistic regression even has close to the representational power to predict chd at all:

```
## [1] 0.7337662
```

We see that our model performs decently, although not exceptionally. Regardless, this should probably be good enough to draw some causal conclusions later when we use it for part C.

Part C

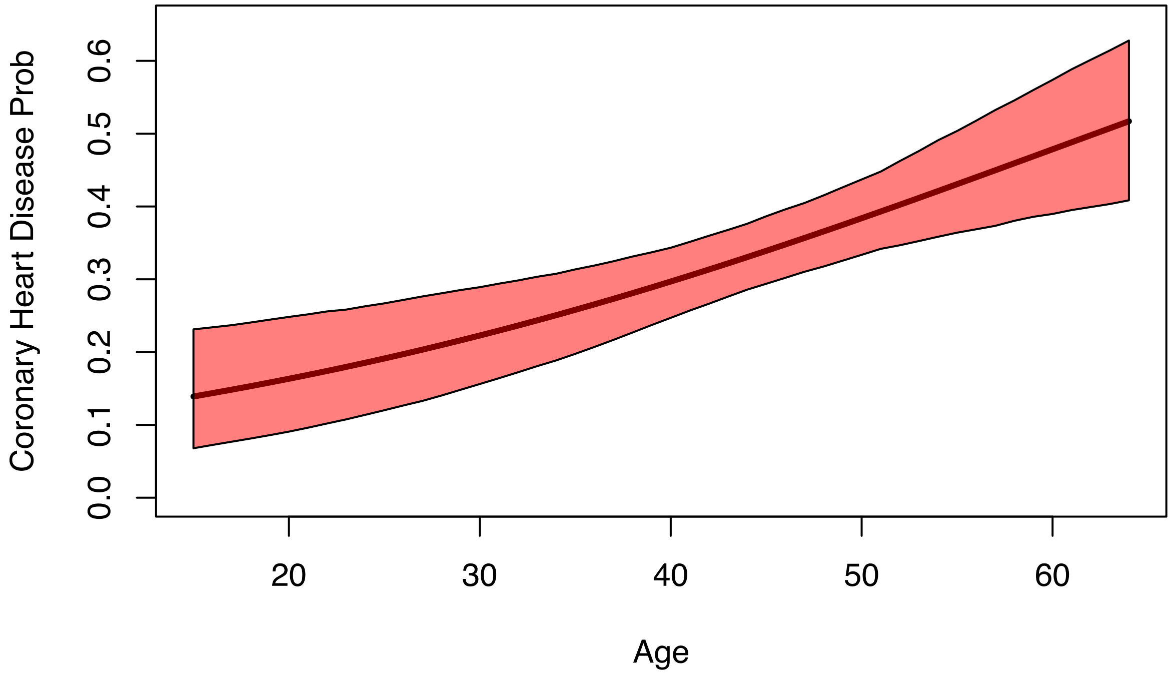
We've already estimated the plugin estimator $\hat{\mu}(X_i, a)$ in Part B. Let's now construct our graph of $\hat{\psi}(a)$.



It seems that there is indeed a causal effect of age on coronary heart disease, which was suggested by our logistic regression coefficient being positive for age.

Part D

Let's bootstrap 1000 times, and make our plot for $\hat{\psi}(a)$:



Code Appendix

```
## ----setup, echo=FALSE--------------------------------------------------------
knitr::opts_chunk$set(cache = TRUE)
options(knitr.duplicate.label = "allow")
set.seed(42)
options(warn=-1)


## ----p2_data, echo=FALSE------------------------------------------------------
df = read.csv("data/SAheart.csv", head=TRUE)

# Drop first column
df = df[,-1]

# Make appropriate things factors
df$famhist = as.factor(df$famhist)
df$chd = as.factor(df$chd)

str(df)
attach(df)


## ----p2a_explore, echo=FALSE--------------------------------------------------
pairs(df)


## ----p2a_plot, echo=FALSE-----------------------------------------------------
plot(age, chd)


## ----p2b_fit, echo=FALSE------------------------------------------------------
logistic_fit = glm(chd ~ ., data=df, family="binomial")
summary(logistic_fit)


## ----p2b_acc, echo=FALSE------------------------------------------------------
logistic_preds = predict(logistic_fit, type="response")
n = length(logistic_preds)
logistic_yhat = rep(0, n)
logistic_yhat[logistic_preds > 0.5] = 1
logistic_T = table(chd, logistic_yhat)
logistic_accuracy = (logistic_T[1,1] + logistic_T[2,2]) / sum(logistic_T)

print(logistic_accuracy)


## ----p2c_plot, echo=FALSE-----------------------------------------------------
# We first make a function to compute Y(a) for some a
compute_y = function(data, a, fit) {
  newdata = data
  n = length(chd)
  new_a = rep(a, n)
  newdata$age = new_a
  preds = predict(fit, newdata=newdata, type="response")
  return(mean(preds))
}

plot_as = c(15:64)
num_as = length(plot_as)
plot_ys = rep(0, num_as)

for (i in 1:num_as) {
  plot_ys[i] = compute_y(df, plot_as[i], logistic_fit)
}

plot(plot_as, plot_ys, type="l", col="blue", xlab="Age", ylab="Coronary Heart Disease Prob")


## ----p2d_plot, echo=FALSE-----------------------------------------------------
bootstrap_ys = matrix(0, 1000, num_as)

for (i in 1:1000) {
  # Sample
  n = length(chd)
  I = sample(1:n, size=n, replace=TRUE)
  dfdf = df[I,]

  # Fit new logistic regression
  fit = glm(dfdf$chd ~ ., data=dfdf, family="binomial")

  for (j in 1:num_as) {
    bootstrap_ys[i, j] = compute_y(dfdf, plot_as[j], fit)
  }
}

# Find quantiles
mean_ys = colMeans(bootstrap_ys)
low_ys = rep(0, num_as)
high_ys = rep(0, num_as)

for (i in 1:num_as) {
  quant = quantile(bootstrap_ys[,i], probs=c(0.025, 0.975))
  low_ys[i] = quant[1]
  high_ys[i] = quant[2]
}

# Plot our data
plot(plot_as, mean_ys, type="l", lwd=3, xlab="Age", ylab="Coronary Heart Disease Prob", ylim=c(0.0, 0.65
polygon(c(plot_as, rev(plot_as)), c(low_ys, rev(high_ys)), col=rgb(1, 0, 0, 0.5))


## ----appendix, code=readLines(knitr::purl(knitr::current_input(), documentation = 1)), echo = T, eval
## NA
```