The Relationship Between Arsenic Concentrations and Well Usage

Nagaprasad Rudrapatna and Martin Olarte

Introduction

Araihazar, Bangladesh

Many of the wells used for drinking water in Bangladesh and other South Asian countries are contaminated with natural arsenic, affecting an estimated 100 million people. Arsenic is a cumulative poison, and exposure increases the risk of cancer and other diseases, with risks estimated to be proportional to exposure. Any locality can include wells with a range of arsenic levels, as can be seen from the map in Figure 5.7 of all the wells in a collection of villages in a small area of Bangladesh. The bad news is that even if your neighbor's well is safe, it does not mean that yours is safe. However, the corresponding good news is that, if your well has a high arsenic level, you can probably find a safe well nearby to get your water from—if you are willing to walk the distance and your neighbor is willing to share. (The amount of water needed for drinking is low enough that adding users to a well would not exhaust its capacity, and the surface water in this area is subject to contamination by microbes, hence the desire to use water from deep wells.) In the area shown in Figure 5.7, a research team from the United States and Bangladesh measured all the wells and labeled them with their arsenic level as well as a characterization as "safe" (below 0.5 in units of hundreds of micrograms per liter, the Bangladesh standard for arsenic in drinking water) or "unsafe" (above 0.5). People with unsafe wells were encouraged to switch to nearby private or community wells or to new wells of their own construction. A few years later, the researchers returned to find out who had switched wells. We shall perform a logistic regression analysis to understand the factors predictive of well switching among the users of unsafe wells. In the notation of the previous section, our outcome variable is

yi = 1 if household i switched to a new well yi = 0 if household i continued using its own well

We consider the following inputs:

We shall first fit the model just using distance to nearest well and then put in arsenic concentration, organizational membership, and education.

```
# Set up cleaned dataset for Bangladesh well-switching

# Read in the data

all <- read.dta ("all.dta", convert.factors=F)
```

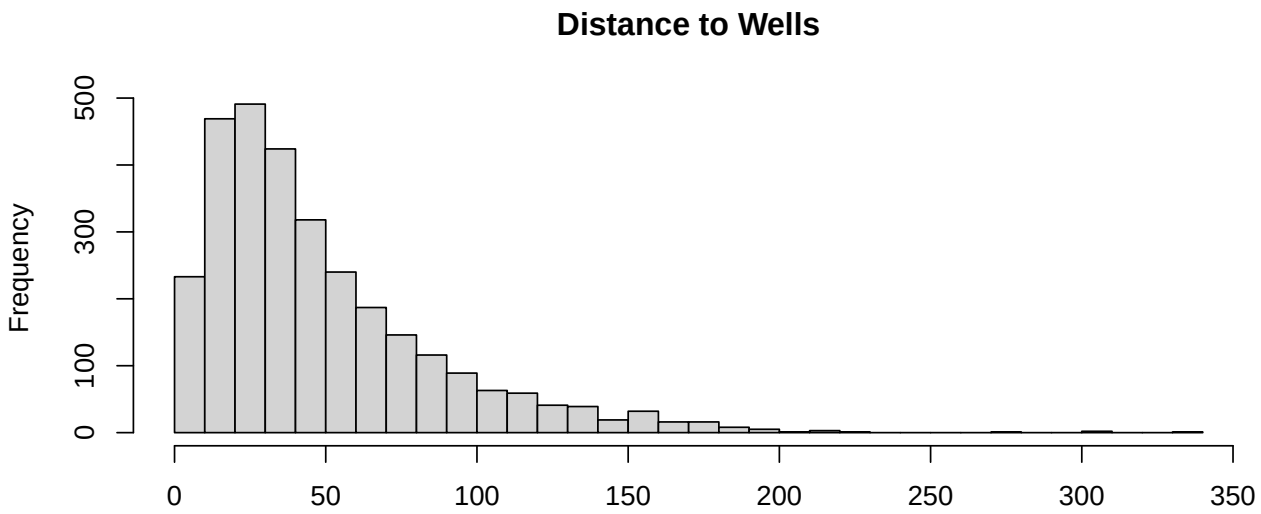
```r
# For simplicity, pull out all wells with missing data in the variables that
# we will be using in our analysis

# according to well owner's wife/sister-in-law
# survey of laypersons, not scientific experts

missing <- is.na (all[,"func"] + all[,"as"] + all[,"distnearest"] + all[,"assn"]
                  + all[,"ed"] + all[,"ed4"] + all[, "drink"] + all[, "status"]
                  + all[, "change"] + all[, "shifted"])
table(missing)
```

```
## missing
## FALSE  TRUE
##  6498    12
```

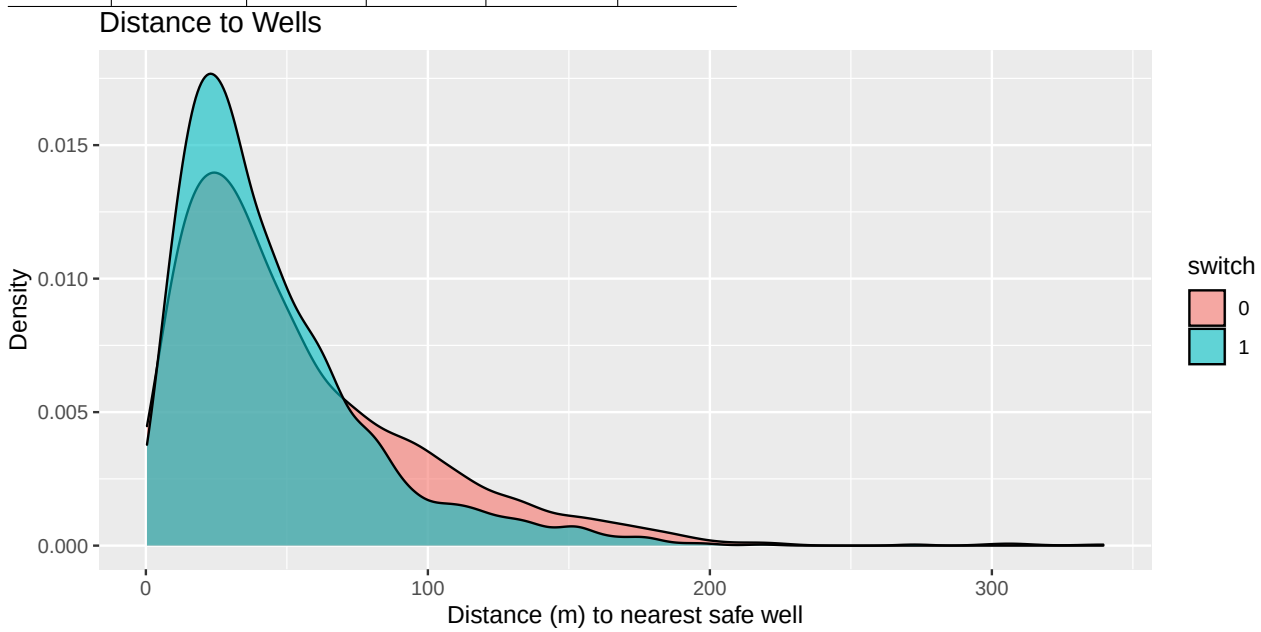
```r
# with the added predictors, still only 12 missing out of 6510

# recode change and shifted to 0/1

# https://www.ldeo.columbia.edu/~avangeen/publications/documents/vanGeen_JESH_07.pdf

# PREDICTIVE ACCURACY

# try recoding: the levels of education among the 3020 respondents varied from
# 0 to 18 years, with nearly a third having zero. We repeated our analysis with
# a discrete recoding of the education variable
# (0 = 0 years, 1 = 1-8 years, 2 = 9-12 years, 3 = 12+ years),
# and our results were essentially unchanged.

# ed4: ed/4 - if ed is not a multiple of 4, ed4 truncates the quotient,
# e.g. ed = 10, ed4 = 10/4 = 2.5 -> 2

# perception of well safety; not based on arsenic concentrations
# (already know that every well in consideration is unsafe) -> see question 6
# to explain potential biases, source of misinformation

# Include only the wells that are functioning (func==1) - analysis does not
# consider preferences (switching for no apparent reason related to safety)
# and "unsafe" (as>50) - arsenic concentration micrograms per liter > 50
keep <- all[,"func"]==1 & all[,"as"]>50
attach.all (all[!missing & keep,])
```

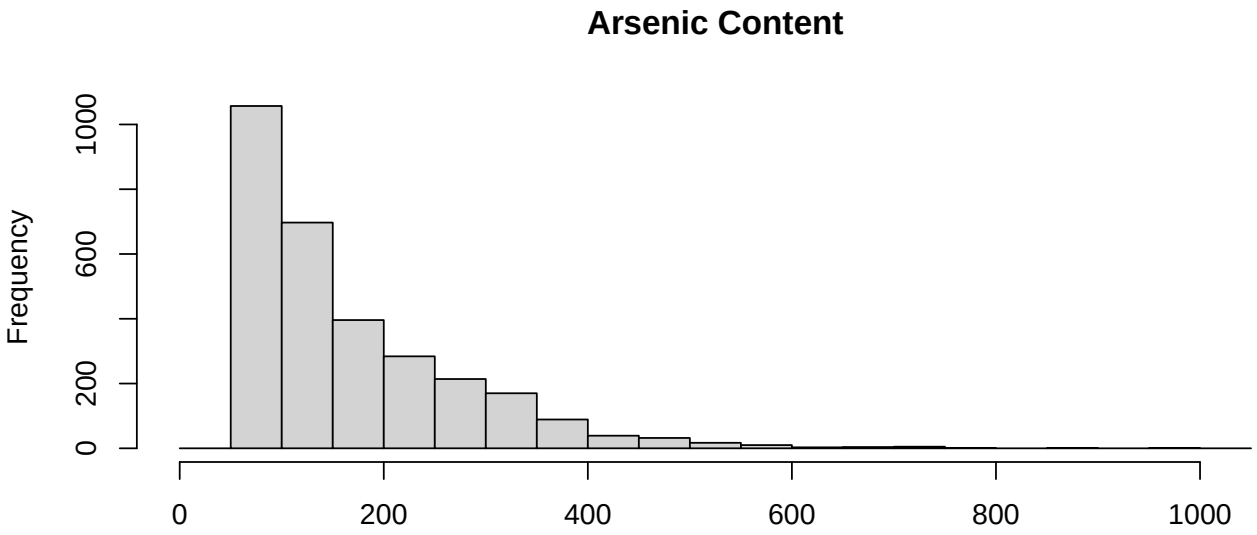
```
## The following object is masked from package:MASS:
##
##     survey
```

```r
# Give convenient names to the variables

switch <- switch
arsenic <- as
dist <- distnearest
assoc <- ifelse (assn > 0, "Yes", "No")
educ <- ed
educ4 <- case_when(ed4 == 0 ~ "0",
                   ed4 == 1 ~ "1-8",
                   ed4 == 2 ~ "9-12",
                   ed4 == 3 ~ "12+")
use <- case_when(drink == 0 ~ "Neither",
                 drink == 1 ~ "Unfrequently",
                 drink == 2 ~ "Drinking",
                 drink == 3 ~ "Cooking",
                 drink == 13 ~ "Cooking",
                 drink == 23 ~ "Both")
status_perceived <- case_when(status == 0 ~ "Unsafe",
                              status == 1 ~ "Safe",
                              status == 2 ~ "Don't Know")
well_shift <- ifelse (shifted == 0, "No", "Yes")

wells.data <- cbind (switch, arsenic, dist, assoc, educ, educ4, use,
                     status_perceived, well_shift)
write.table (wells.data, "wells.dat")

wells <- read.table("wells.dat", header = TRUE)
df <- data.frame(wells) %>%
  mutate(switch = as.factor(switch),
         assoc = as.factor(assoc),
         educ4 = factor(educ4, levels = c("0", "1-8", "9-12", "12+")),
         use = factor(use, levels = c("Unfrequently", "Drinking", "Cooking", "Both", "Ne
         status_perceived = factor(status_perceived, levels = c("Safe", "Unsafe", "Don't
         well_shift = as.factor(well_shift))

drinkorcook_wells <- wells[wells$use!=0,]
## Stopping point
```

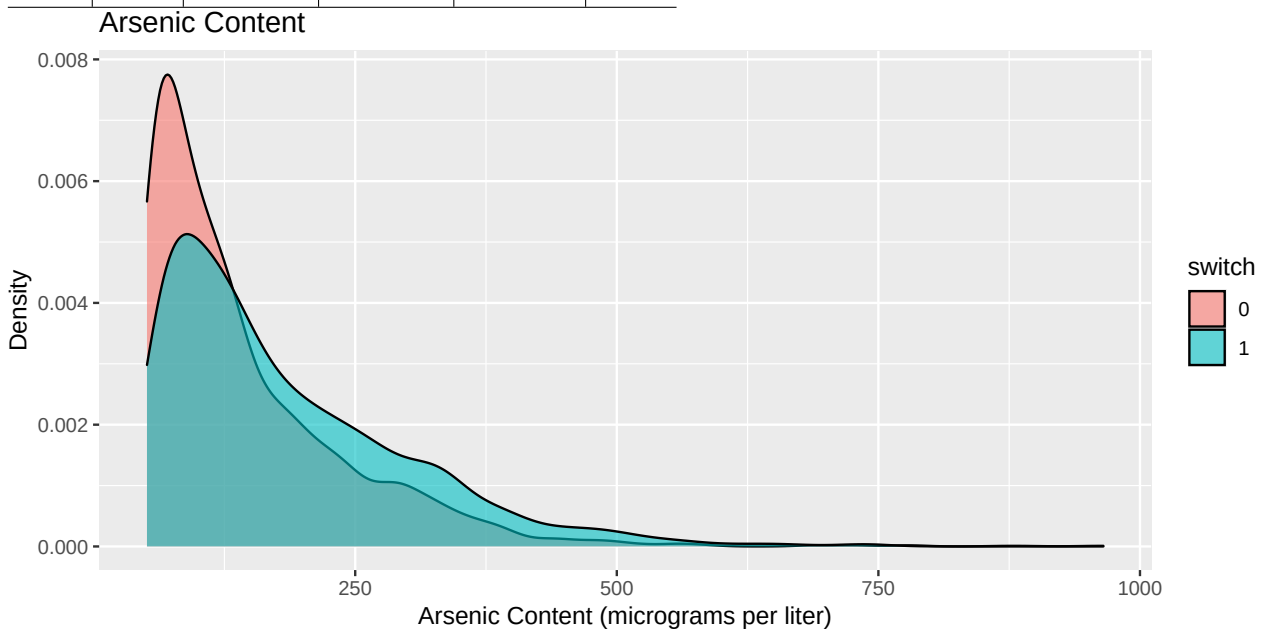


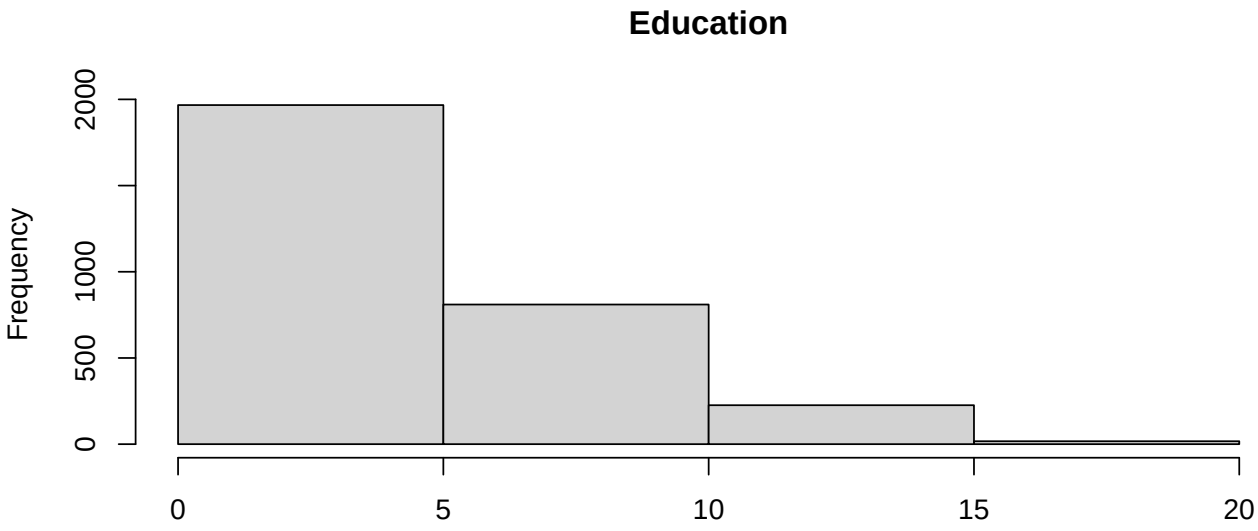
min	max	mean	sd	median	IQR
0.387	339.531	48.332	38.479	36.761	42.924



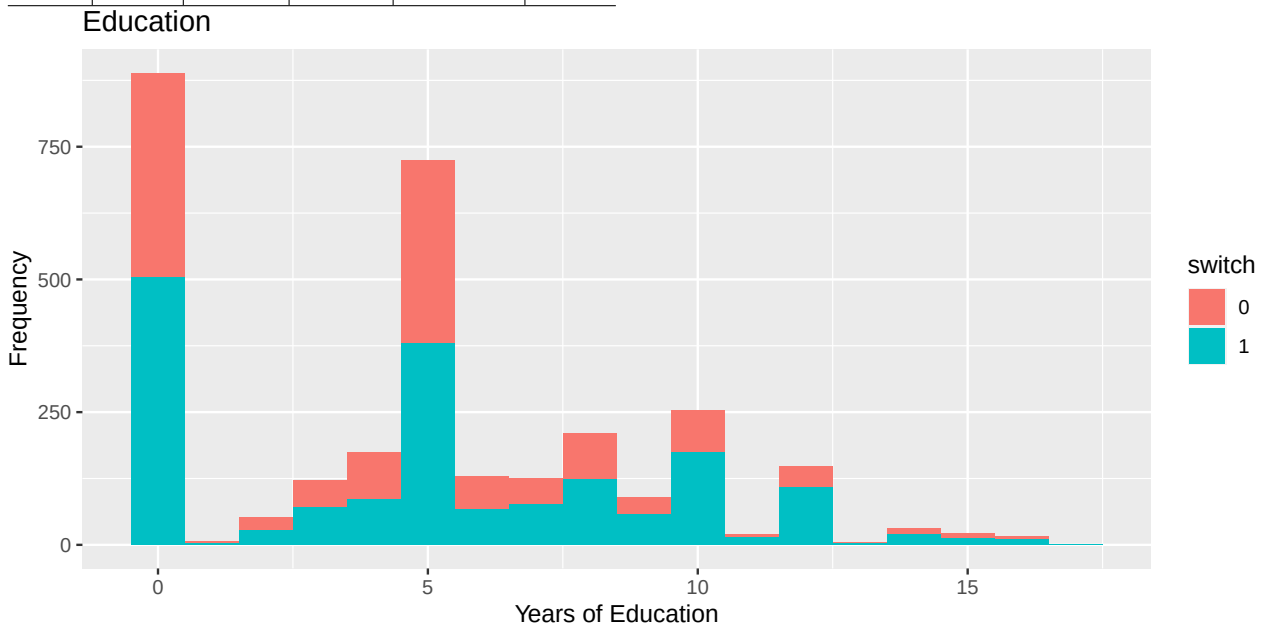


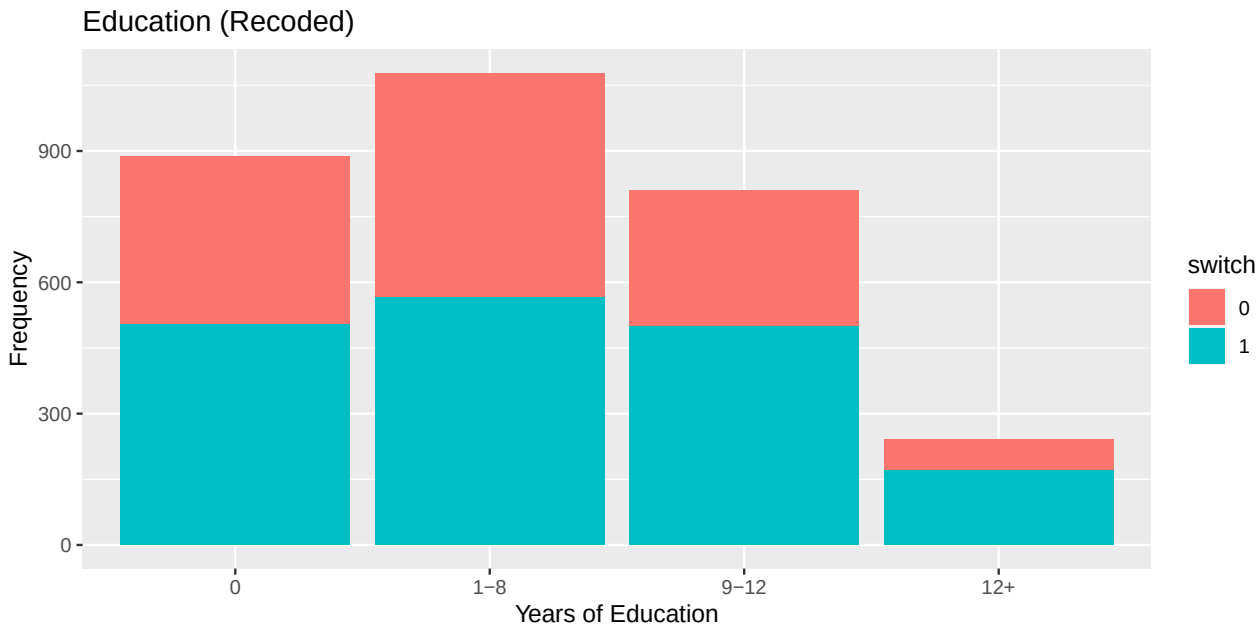
min	max	mean	sd	median	IQR
51	965	165.693	110.739	130	138

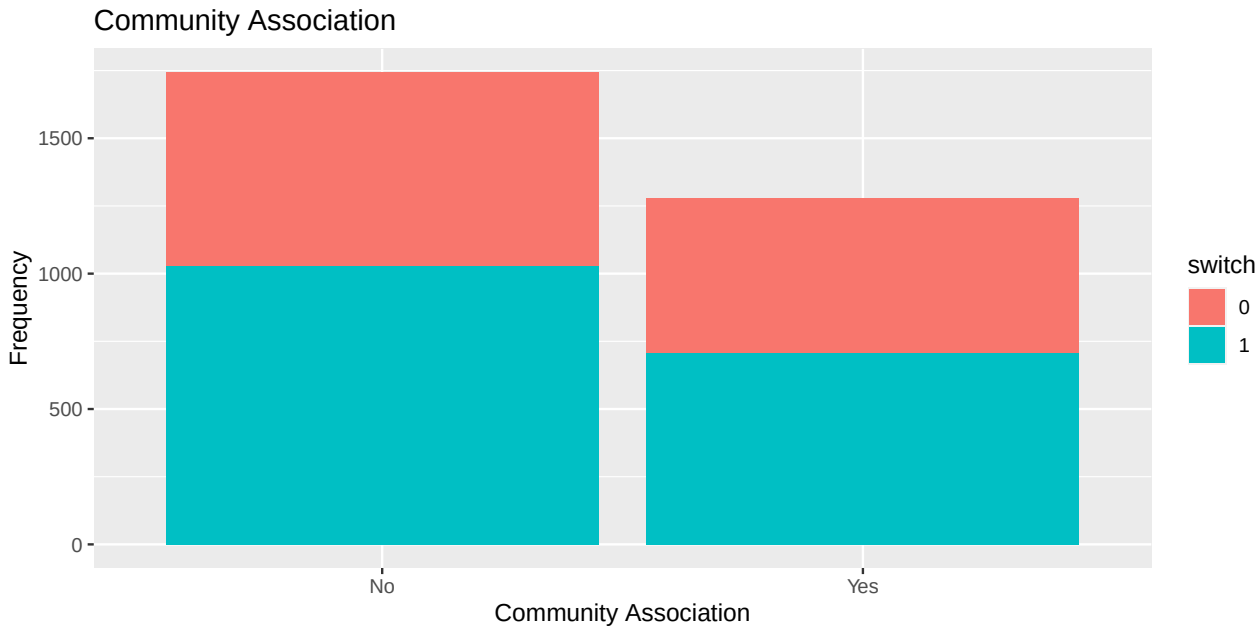


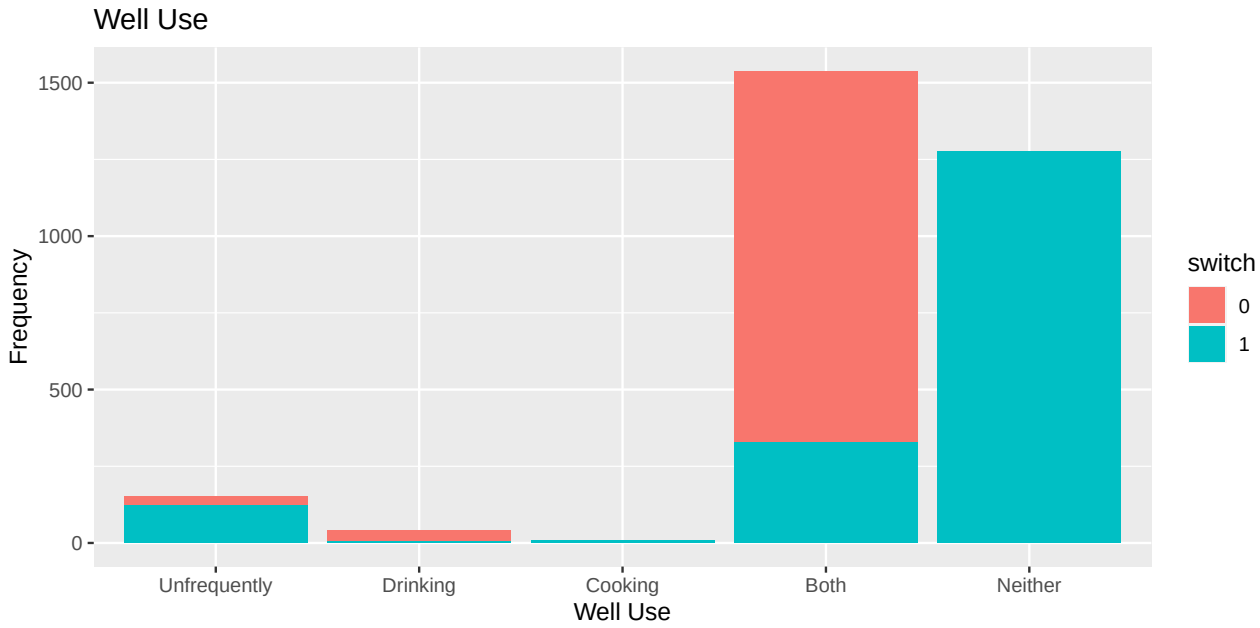


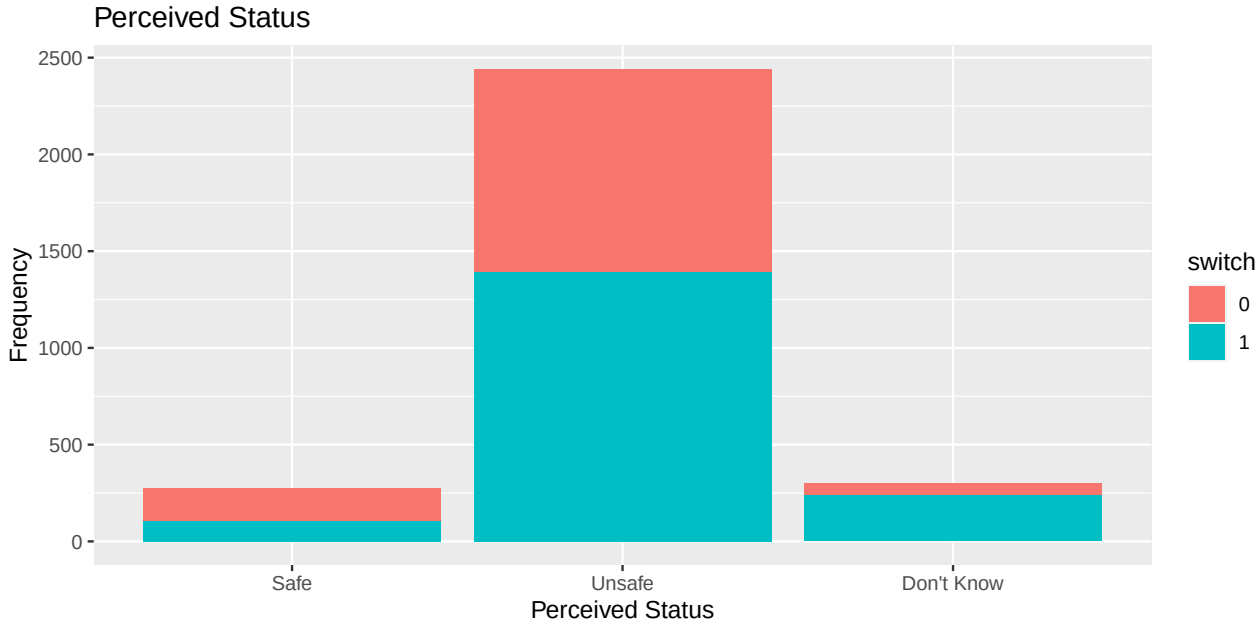
min	max	mean	sd	median	IQR
0	17	4.828	4.017	5	8

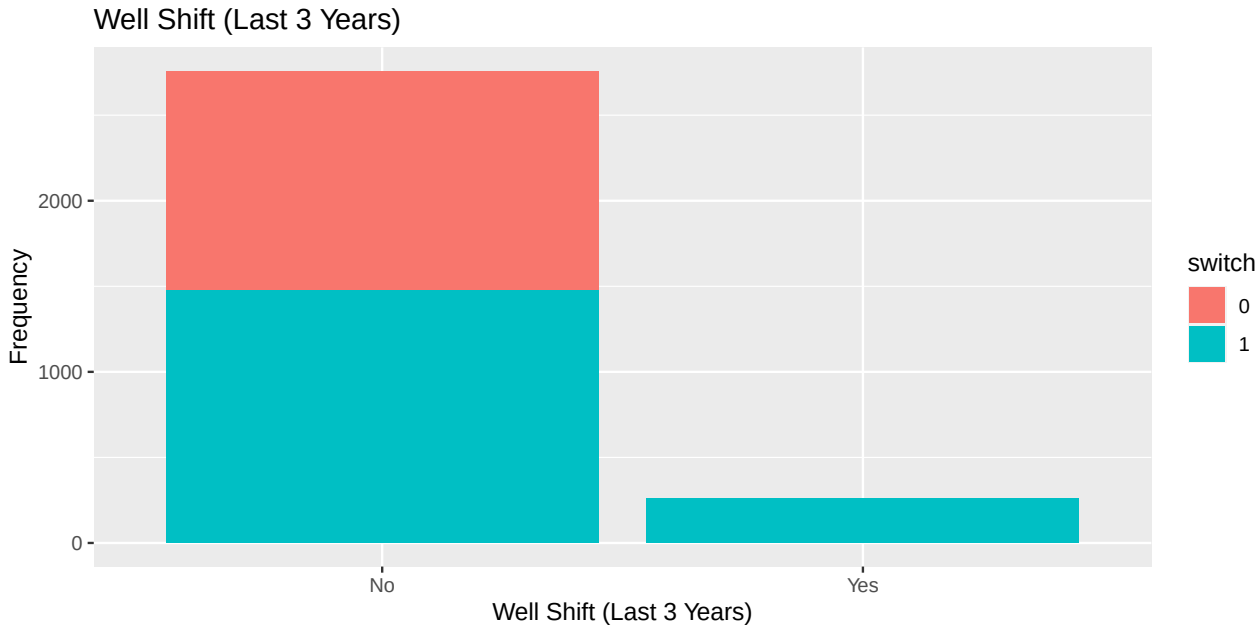


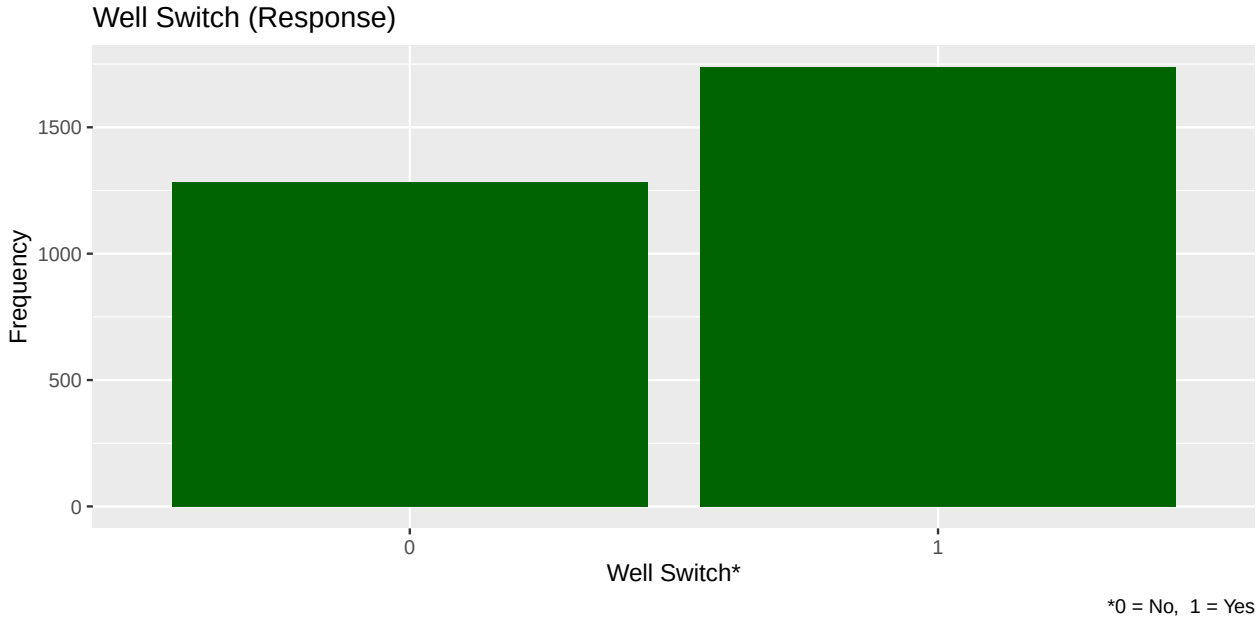

Education (Recoded)
900
600
300
0
Frequency
0
1–8
9–12
12+
Years of Education
switch
0
1


Community Association
1500
1000
500
0
Frequency
No
Yes
Community Association
switch
0
1


Well Use
Frequency
1500
1000
500
0
Unfrequently
Drinking
Cooking
Both
Neither
Well Use
switch
0
1


Perceived Status
Frequency
2500
2000
1500
1000
500
0
Safe
Unsafe
Don't Know
Perceived Status
switch
0
1





- arsenic concentration: all > 50 based on structure of problem; max: 965 micrograms per liter which is almost 20x the safe concentration threshold (justifies why this is even an issue that needs to be urgently addressed); seems that arsenic content is less of a significant predictor than perceived well status and distance to nearest safe well (interaction with perceived well status could be relevant as perhaps households decide whether to switch based on a combination of perceptions and science; interaction with distance: wells that are far from the nearest safe well are also likely to be particularly high in arsenic)

- community association vs. switch: the relative proportions of switch/no switch seem consistent regardless of community association; evidence in the data suggests community association is insignificant (still include in initial model before model selection to confirm

our EDA observations are supported)

- continuous education histogram: clear trend, low levels of education most prevalent among population
- years of education vary greatly among households; majority either have no formal education or five years; difficult to determine if there is a significant effect since some of the levels have little observations -> motivates recoding
- recoded education: as expected, 0 years is still most frequent; the overall population of interest has between 0-12 years of education (up to high school); 12+ (tertiary education) seems to be a luxury; among those who have 12+ years in education, most did not choose to switch - is this because of well use? (interaction with well use justified)
- well use with switch: the vast majority of households use wells for both drinking and cooking, or neither; those who do not use wells for drinking and cooking, as expected, responded they would switch most frequently; those who used wells for both mostly did not choose to switch -> well use significant predictor
- the univariate distribution of distances confirms our initial suspicions about the proximity of safe and unsafe wells. It is clear that safe and unsafe wells are intermingled in most of the area, which suggests that users of unsafe wells can switch to nearby safe wells; summary stats tell us that the maximum distance to the nearest safe (define earlier) well is 339.531 meters (which is reasonable for daily round trips (double distance); assume speed 1 m/s, then roughly 340 s $\sim$ 5.7 min) and the average distance is 48.332 m (interaction with well use, well purpose determines frequency of use and daily travel time; interaction with perceived status, maybe households did not switch even with small distances due to misassessment of risk)
- since the densities do not exactly align, there is evidence in the data that distance to the nearest safe well is not the only significant covariate; there is a noticeable difference in the propensity of households to switch when the distance to the nearest safe well is between 0-200 m; at the tails (i.e. when the distance exceeds 230 or so, the densities overlap); switching occurs with greatest probability when distance to nearest safe well is between 0-50 m
- based on the plot of well shift in the last 3 years and the response, there is evidence in the data suggesting that this covariate should be significant in the logistic regression model; note that all of the households which shifted to a different well in the past three years, expected to switch once more when informed about the unsafe well (interaction with distance to safest well; maybe some sections of the region have more unsafe wells than others, so frequent shifting is best practice)
- 1283 - no switch
- 1737 - switch although there are a lot of 0's, the data is not unbalanced (no cloglog regression)

- based on the plot of perceived well status and the response, there is evidence in the data suggesting that this covariate should be significant in the logistic regression model (seemingly significant differences, at least visually, among levels of predictor); note that a significant proportion of households who believed the well to be unsafe, still chose not to switch (interaction with well use; if not using for cooking and/or drinking, the standard for cleanliness may be lower in people's minds; interaction with education: people's beliefs and perceptions are shaped by their experiences; lack of formal education may indicate a lack of appreciation for science and lead to misassessment of risk)

Modeling

- Covariates:

```
# Logistic regression of switching on distance to nearest safe well
# Centering


seed <- 200

fit.1 <- stan_glm(switch ~ dist, data = wells,
                  family = binomial(link = "logit"),
                  prior = normal(0,1), prior_intercept = normal(0,1),
                  seed = seed,
                  refresh = 0)

print(fit.1$stanfit)
```

```
## Inference for Stan model: bernoulli.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##                   mean se_mean   sd     2.5%      25%      50%      75%
## (Intercept)       0.61    0.00 0.06     0.49     0.57     0.61     0.65
## dist             -0.01    0.00 0.00    -0.01    -0.01    -0.01    -0.01
## mean_PPD          0.58    0.00 0.01     0.55     0.57     0.57     0.58
## log-posterior -2041.03    0.03 1.04 -2043.84 -2041.43 -2040.70 -2040.29
##                  97.5% n_eff Rhat
## (Intercept)       0.73  1409    1
## dist              0.00  3112    1
## mean_PPD          0.60  1373    1
## log-posterior -2040.03   904    1
##
## Samples were drawn using NUTS(diag_e) at Sat May  1 12:44:33 2021.
## For each parameter, n_eff is a crude measure of effective sample size,
```

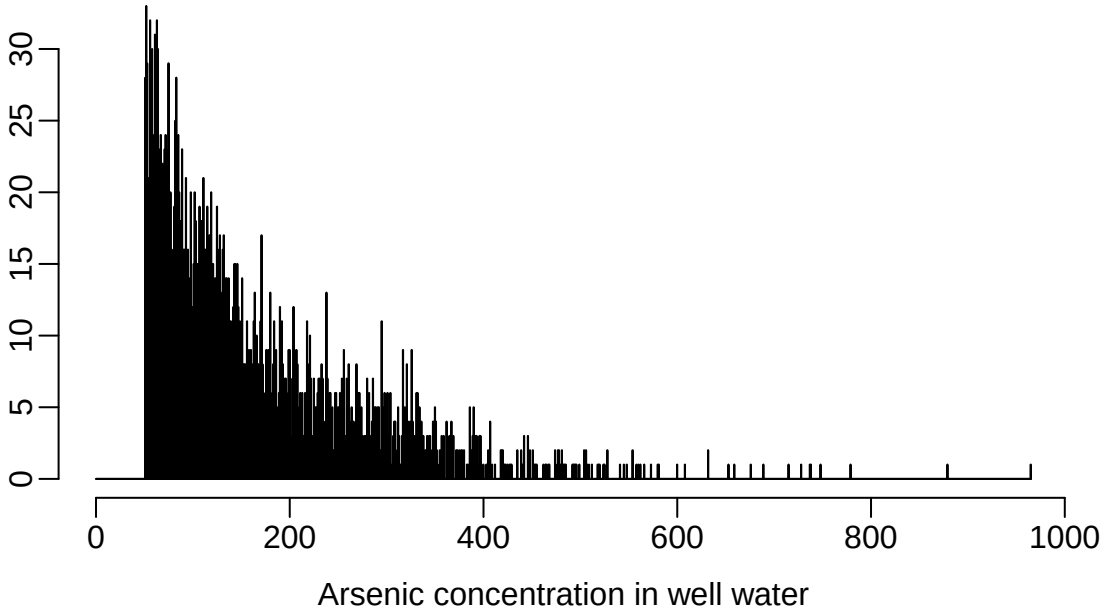
```
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
# Redefine distance in 100-meter units and fit the model again

dist100 <- dist/100
fit.2 <- stan_glm(switch ~ dist100, data = wells,
                  family = binomial(link = "logit"),
                  prior = normal(0,1), prior_intercept = normal(0,1),
                  seed = seed,
                  refresh = 0)

print(fit.2$stanfit)
```

```
## Inference for Stan model: bernoulli.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##                   mean se_mean   sd     2.5%      25%      50%      75%
## (Intercept)       0.60    0.00 0.06     0.49     0.56     0.60     0.65
## dist100          -0.62    0.00 0.10    -0.79    -0.68    -0.62    -0.55
## mean_PPD          0.57    0.00 0.01     0.55     0.57     0.57     0.58
## log-posterior -2041.20    0.02 0.98 -2043.80 -2041.61 -2040.92 -2040.49
##                  97.5% n_eff Rhat
## (Intercept)       0.72  2248    1
## dist100          -0.43  2358    1
## mean_PPD          0.60  3007    1
## log-posterior -2040.22  1756    1
##
## Samples were drawn using NUTS(diag_e) at Sat May  1 12:44:42 2021.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
### END OF EDITED CODE

# plots of model fit

jitter.binary <- function(a, jitt=.05){
  a + (1-2*a)*runif(length(a),0,jitt)
}

#postscript ("c:/books/multilevel/arsenic.logitfit.1new.a.ps", height=3.5, width=4, ho
#plot(c(0,max(dist, na.rm=TRUE)*1.02), c(0,1), xlab="Distance (in meters) to nearest s
```

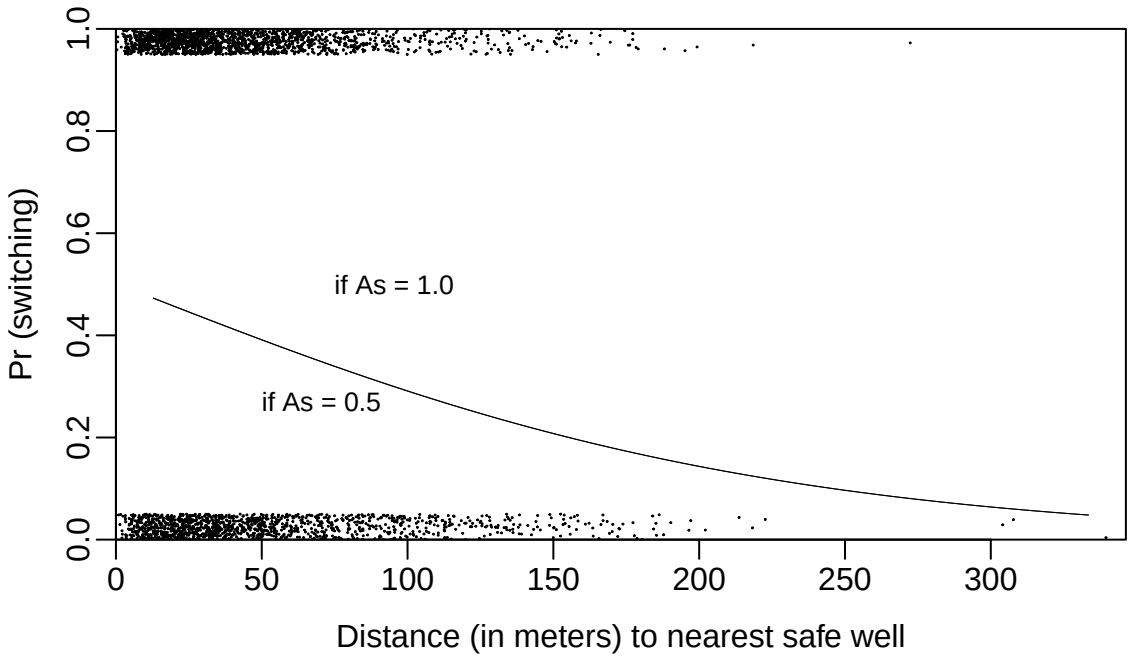
```
#curve (invlogit(coef(fit.1)[1]+coef(fit.1)[2]*x), lwd=1, add=TRUE)
#points (dist, jitter.binary(switch), pch=20, cex=.1)
#dev.off ()

# histogram of As levels

#postscript ("c:/books/multilevel/arsenic.levels.a.ps", height=3, width=4, horizontal=
hist (arsenic, breaks=seq(0,.25+max(arsenic[!is.na(arsenic)]),.25), freq=TRUE, xlab="Ars
```



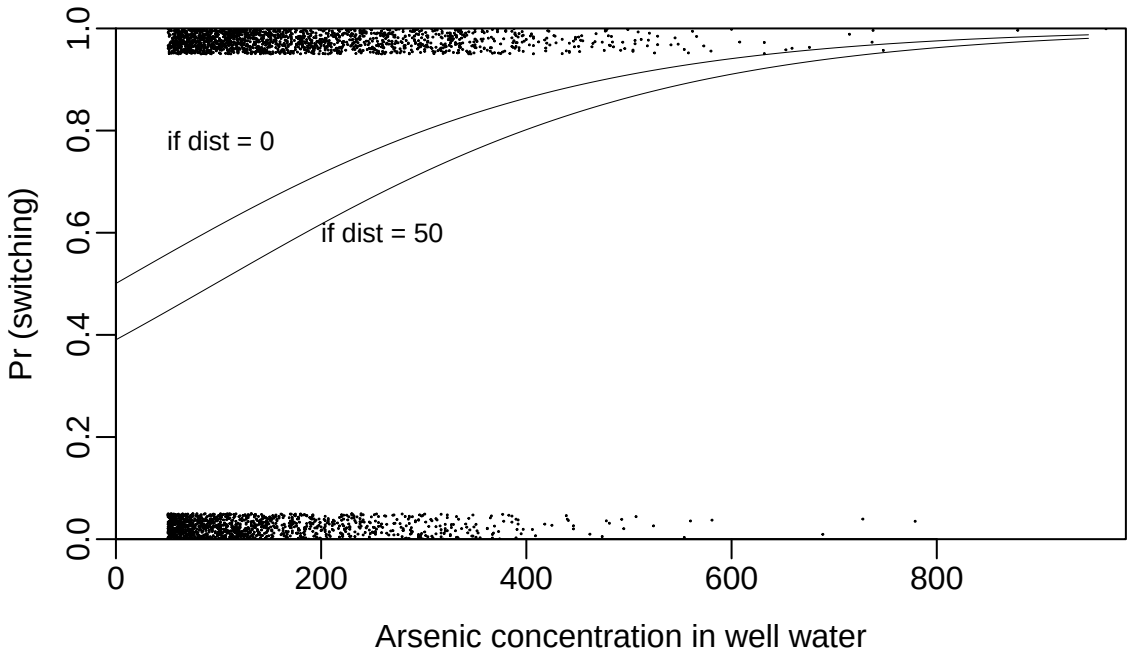

```
#dev.off ()

# model with 2 predictors

fit.3 <- glm (switch ~ dist100 + arsenic, family=binomial(link="logit"))
display (fit.3)
```

```
## glm(formula = switch ~ dist100 + arsenic, family = binomial(link = "logit"))
##             coef.est coef.se
## (Intercept)  0.00     0.08
## dist100     -0.90     0.10
## arsenic      0.00     0.00
## ---
##   n = 3020, k = 3
##   residual deviance = 3930.7, null deviance = 4118.1 (difference = 187.4)
```

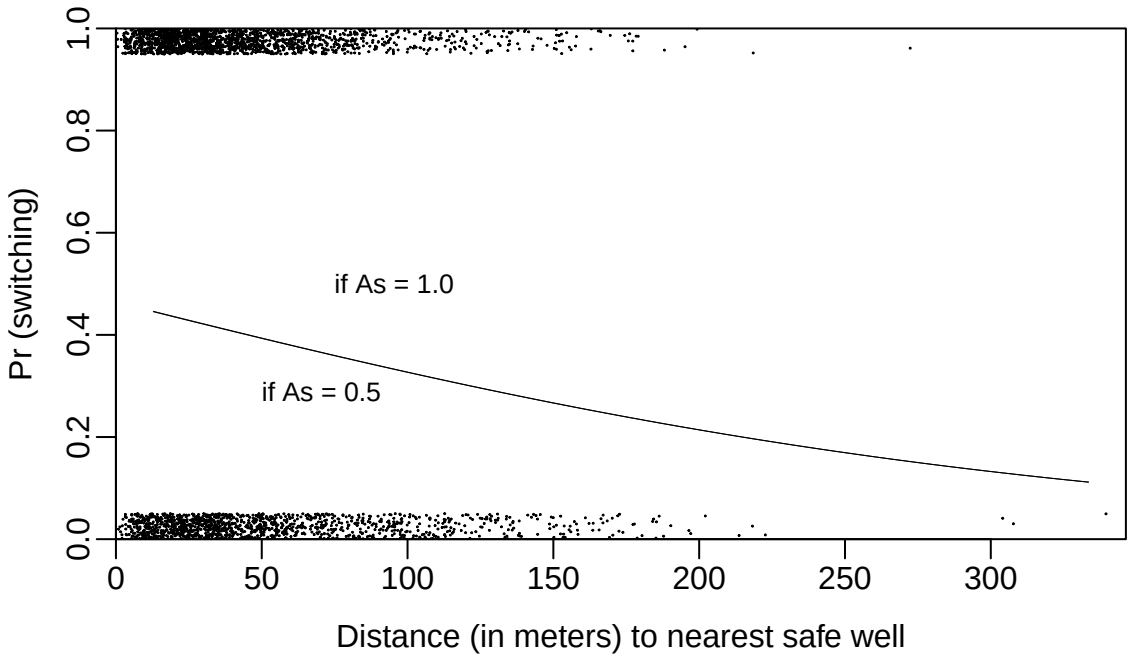
```
#postscript ("c:/books/multilevel/arsenic.2variables.a.ps", height=3.5, width=4, horiz
plot(c(0,max(dist,na.rm=TRUE)*1.02), c(0,1), xlab="Distance (in meters) to nearest safe
points (dist, jitter.binary(switch), pch=20, cex=.1)
curve (invlogit(coef(fit.3)[1]+coef(fit.3)[2]*x/100+coef(fit.3)[3]*.50), lwd=.5, add=TRU
curve (invlogit(coef(fit.3)[1]+coef(fit.3)[2]*x/100+coef(fit.3)[3]*1.00), lwd=.5, add=TR
text (50, .27, "if As = 0.5", adj=0, cex=.8)
text (75, .50, "if As = 1.0", adj=0, cex=.8)
```



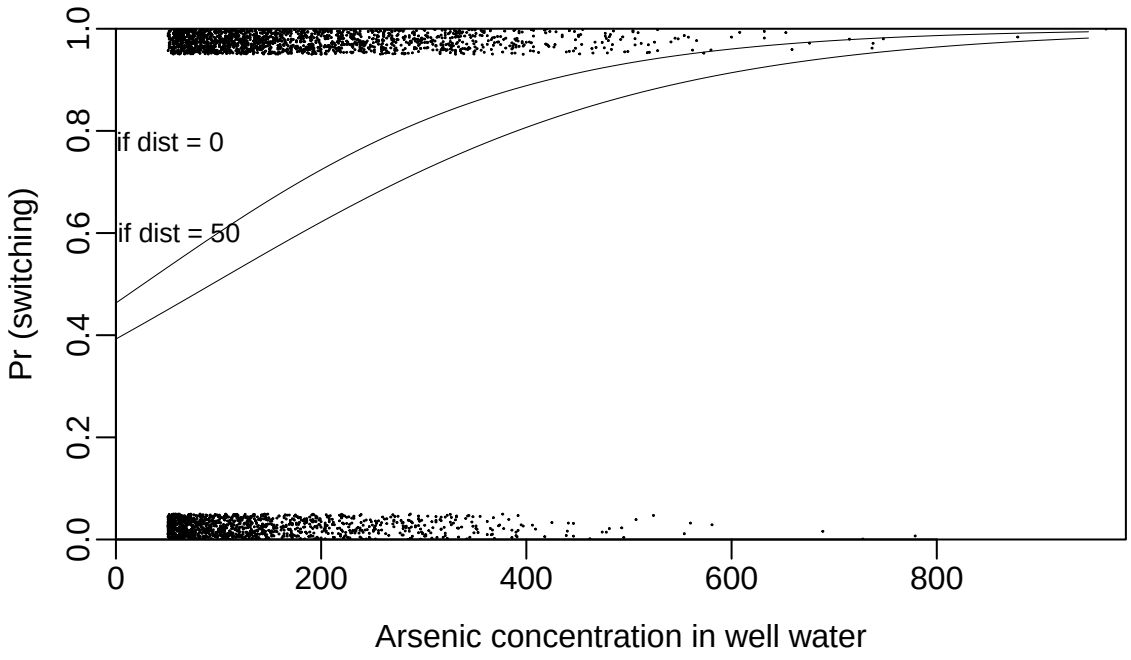

```
#dev.off ()

#postscript ("c:/books/multilevel/arsenic.2variables.b.ps", height=3.5, width=4, horiz
plot(c(0,max(arsenic,na.rm=TRUE)*1.02), c(0,1), xlab="Arsenic concentration in well wate
points (arsenic, jitter.binary(switch), pch=20, cex=.1)
curve (invlogit(coef(fit.3)[1]+coef(fit.3)[2]*0+coef(fit.3)[3]*x), from=0.5, lwd=.5, ad
curve (invlogit(coef(fit.3)[1]+coef(fit.3)[2]*0.5+coef(fit.3)[3]*x), from=0.5, lwd=.5,
text (50, .78, "if dist = 0", adj=0, cex=.8)
text (200, .6, "if dist = 50", adj=0, cex=.8)
```




```
#dev.off ()

# including an interaction

fit.4 <- glm (switch ~ dist100 + arsenic + dist100:arsenic,
  family=binomial(link="logit"))

# centering the input variables

c.dist100 <- dist100 - mean (dist100)
c.arsenic <- arsenic - mean (arsenic)

fit.5 <- glm (switch ~ c.dist100 + c.arsenic + c.dist100:c.arsenic,
  family=binomial(link="logit"))

#postscript ("c:/books/multilevel/arsenic.interact.a.ps", height=3.5, width=4, horizon
plot(c(0,max(dist,na.rm=TRUE)*1.02), c(0,1), xlab="Distance (in meters) to nearest safe
points (dist, jitter.binary(switch), pch=20, cex=.1)
curve (invlogit(coef(fit.4)[1]+coef(fit.4)[2]*x/100+coef(fit.4)[3]*.50+coef(fit.4)[4]*(
curve (invlogit(coef(fit.4)[1]+coef(fit.4)[2]*x/100+coef(fit.4)[3]*1.00+coef(fit.4)[4]*
text (50, .29, "if As = 0.5", adj=0, cex=.8)
text (75, .50, "if As = 1.0", adj=0, cex=.8)
```




```
#dev.off ()

#postscript ("c:/books/multilevel/arsenic.interact.b.ps", height=3.5, width=4, horizon
plot(c(0,max(arsenic,na.rm=TRUE)*1.02), c(0,1), xlab="Arsenic concentration in well wate
points (arsenic, jitter.binary(switch), pch=20, cex=.1)
curve (invlogit(coef(fit.4)[1]+coef(fit.4)[2]*0+coef(fit.4)[3]*x+coef(fit.4)[4]*0*x), f
curve (invlogit(coef(fit.4)[1]+coef(fit.4)[2]*0.5+coef(fit.4)[3]*x+coef(fit.4)[4]*0.5*x
text (.50, .78, "if dist = 0", adj=0, cex=.8)
text (2.00, .6, "if dist = 50", adj=0, cex=.8)
```



```
#dev.off ()

# adding social predictors

educ4 <- educ/4

fit.6 <- glm (switch ~ c.dist100 + c.arsenic + c.dist100:c.arsenic +
  assoc + educ4, family=binomial(link="logit"))
display (fit.6)
```

```
## glm(formula = switch ~ c.dist100 + c.arsenic + c.dist100:c.arsenic +
##     assoc + educ4, family = binomial(link = "logit"))
##                     coef.est coef.se
## (Intercept)          0.20     0.07
## c.dist100           -0.88     0.11
## c.arsenic            0.00     0.00
## assocYes            -0.12     0.08
## educ4                0.17     0.04
## c.dist100:c.arsenic  0.00     0.00
## ---
##   n = 3020, k = 6
##   residual deviance = 3905.4, null deviance = 4118.1 (difference = 212.7)
```

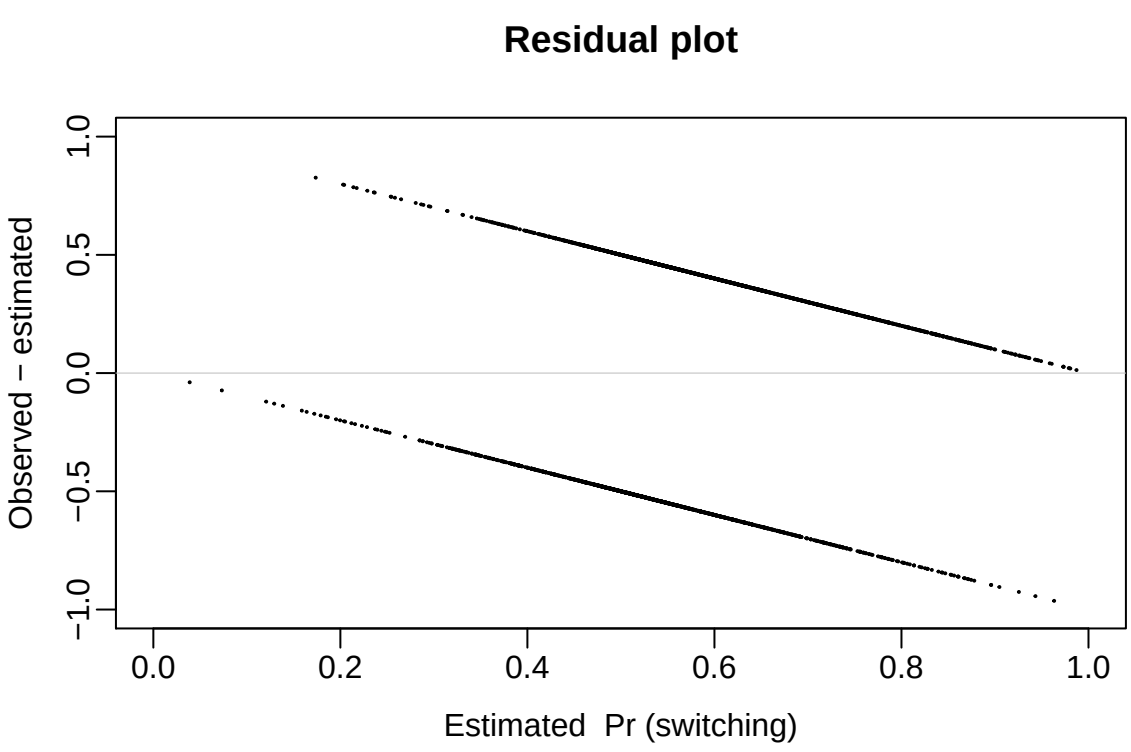
```
fit.7 <- glm (switch ~ c.dist100 + c.arsenic + c.dist100:c.arsenic +
  educ4, family=binomial(link="logit"))
display (fit.7)
```

```
## glm(formula = switch ~ c.dist100 + c.arsenic + c.dist100:c.arsenic +
##     educ4, family = binomial(link = "logit"))
##                     coef.est coef.se
## (Intercept)          0.15     0.06
## c.dist100           -0.87     0.11
## c.arsenic            0.00     0.00
## educ4                0.17     0.04
## c.dist100:c.arsenic  0.00     0.00
## ---
##   n = 3020, k = 5
##   residual deviance = 3907.9, null deviance = 4118.1 (difference = 210.2)
```

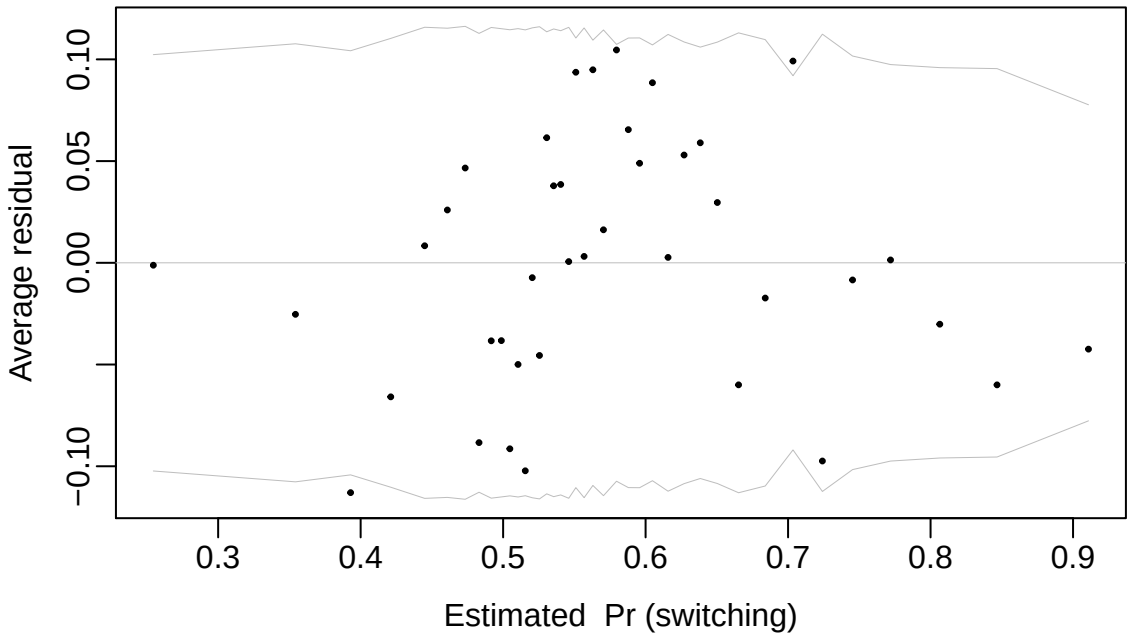
```
c.educ4 <- educ4 - mean(educ4)

fit.8 <- glm (switch ~ c.dist100 + c.arsenic + c.educ4 + c.dist100:c.arsenic +
  c.dist100:c.educ4 + c.arsenic:c.educ4, family=binomial(link="logit"))
display (fit.8)
```

```
## glm(formula = switch ~ c.dist100 + c.arsenic + c.educ4 + c.dist100:c.arsenic +
##     c.dist100:c.educ4 + c.arsenic:c.educ4, family = binomial(link = "logit"))
##                       coef.est coef.se
## (Intercept)            0.36     0.04
## c.dist100             -0.90     0.11
## c.arsenic              0.00     0.00
## c.educ4                0.18     0.04
## c.dist100:c.arsenic    0.00     0.00
## c.dist100:c.educ4      0.32     0.11
## c.arsenic:c.educ4      0.00     0.00
## ---
##   n = 3020, k = 7
##   residual deviance = 3891.7, null deviance = 4118.1 (difference = 226.4)
```

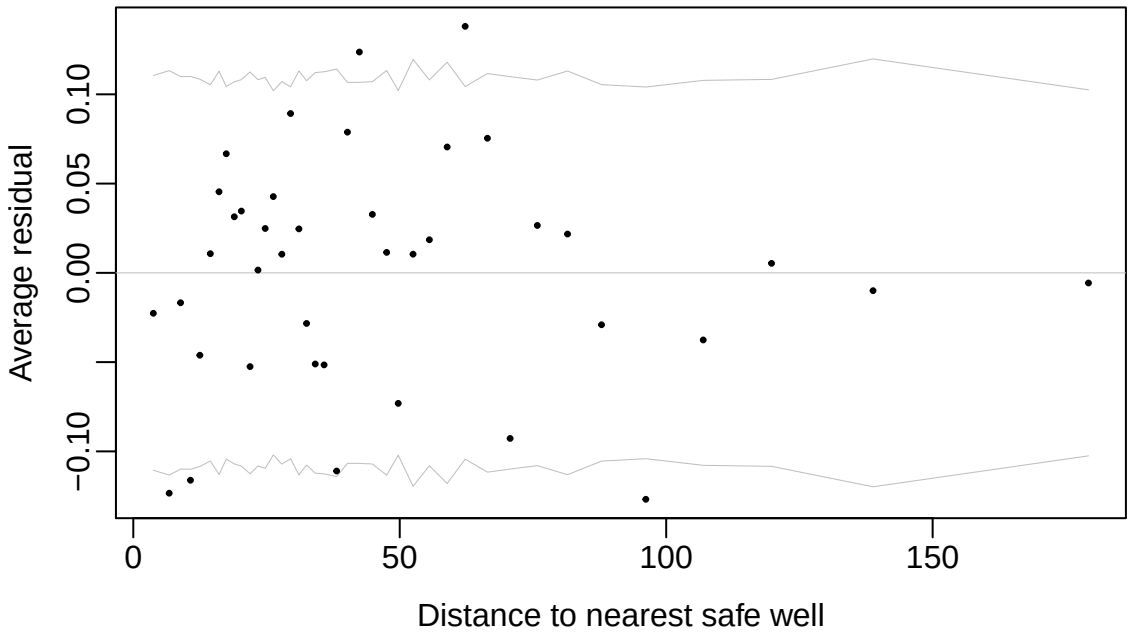
```
# plots of residuals

pred.8 <- fit.8$fitted.values

#postscript ("c:/books/multilevel/arsenic.logitresidsa.ps", height=3.5, width=4, horiz
plot(c(0,1), c(-1,1), xlab="Estimated  Pr (switching)", ylab="Observed - estimated", typ
abline (0,0, col="gray", lwd=.5)
points (pred.8, switch-pred.8, pch=20, cex=.2)
```



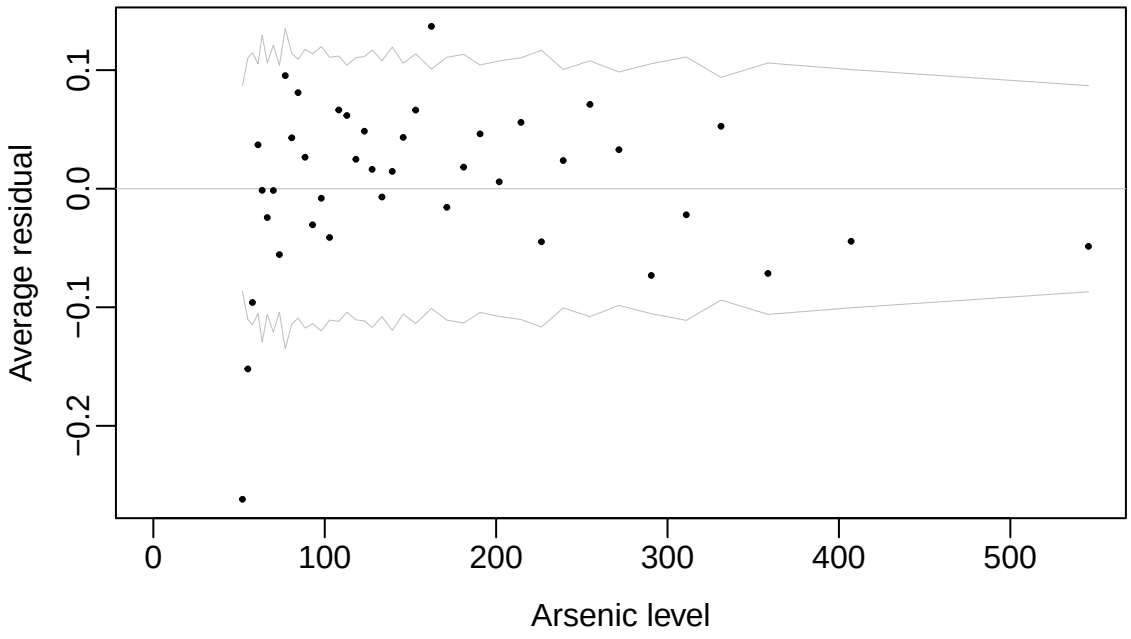
```
#dev.off ()

binned.resids <- function (x, y, nclass=sqrt(length(x))){
  breaks.index <- floor(length(x)*(1:(nclass-1))/nclass)
  breaks <- c (-Inf, sort(x)[breaks.index], Inf)
  output <- NULL
  xbreaks <- NULL
  x.binned <- as.numeric (cut (x, breaks))
  for (i in 1:nclass){
    items <- (1:length(x))[x.binned==i]
    x.range <- range(x[items])
    xbar <- mean(x[items])
    ybar <- mean(y[items])
    n <- length(items)
    sdev <- sd(y[items])
    output <- rbind (output, c(xbar, ybar, n, x.range, 2*sdev/sqrt(n)))
  }
  colnames (output) <- c ("xbar", "ybar", "n", "x.lo", "x.hi", "2se")
  return (list (binned=output, xbreaks=xbreaks))
}

#postscript ("c:/books/multilevel/arsenic.logitresidsb.ps", height=3.5, width=4, horiz
br.8 <- binned.resids (pred.8, switch-pred.8, nclass=40)$binned
plot(range(br.8[,1]), range(br.8[,2],br.8[,6],-br.8[,6]), xlab="Estimated  Pr (switching
abline (0,0, col="gray", lwd=.5)
lines (br.8[,1], br.8[,6], col="gray", lwd=.5)
lines (br.8[,1], -br.8[,6], col="gray", lwd=.5)
points (br.8[,1], br.8[,2], pch=20, cex=.5)
```




```
#dev.off ()

# compute error rates

error.rate <- mean(round(abs(switch-pred.8)))
error.rate.null <- mean(round(abs(switch-mean(pred.8))))

# more residual plots

#postscript ("c:/books/multilevel/arsenic.logitresids.2a.ps", height=3.5, width=4, hor
br <- binned.resids (dist, switch-pred.8, nclass=40)$binned
plot(range(br[,1]), range(br[,2],br[,6],-br[,6]), xlab="Distance to nearest safe well",
abline (0,0, col="gray", lwd=.5)
n.within.bin <- length(y)/nrow(br)
lines (br[,1], br[,6], col="gray", lwd=.5)
lines (br[,1], -br[,6], col="gray", lwd=.5)
points (br[,1], br[,2], pch=20, cex=.5)
```




```
#dev.off ()

#postscript ("c:/books/multilevel/arsenic.logitresids.2b.ps", height=3.5, width=4, hor
br <- binned.resids (arsenic, switch-pred.8, nclass=40)$binned
plot(range(0,br[,1]), range(br[,2],br[,6],-br[,6]), xlab="Arsenic level", ylab="Average
abline (0,0, col="gray", lwd=.5)
lines (br[,1], br[,6], col="gray", lwd=.5)
lines (br[,1], -br[,6], col="gray", lwd=.5)
points (br[,1], br[,2], pch=20, cex=.5)
```

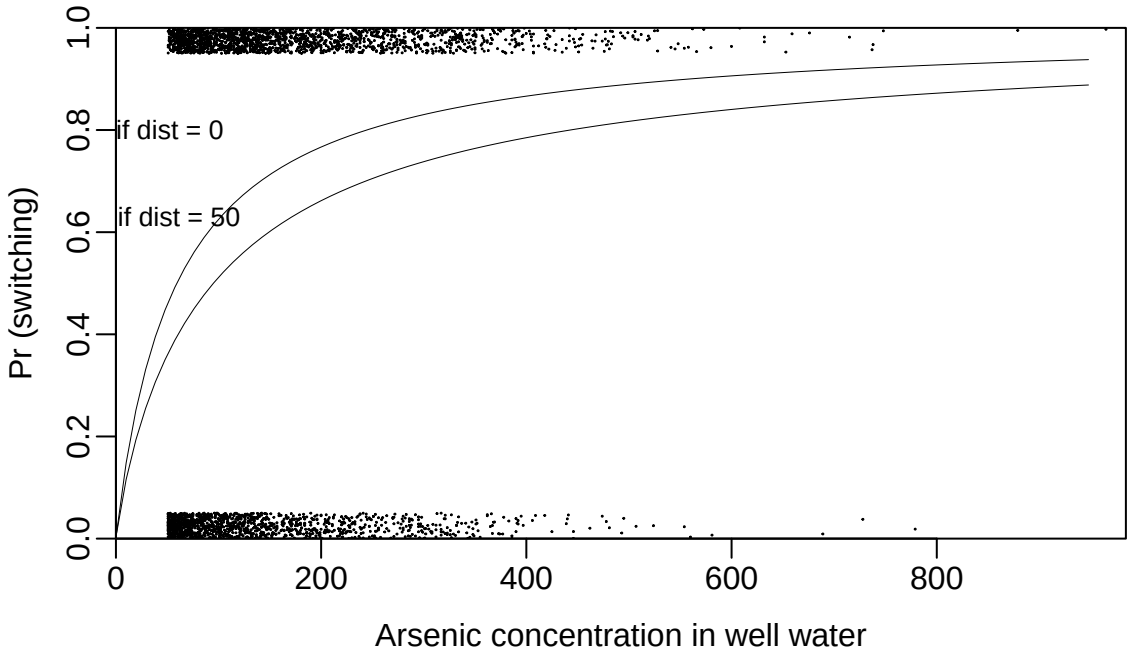



```
#dev.off ()

# new model on log scale

log.arsenic <- log (arsenic)
c.log.arsenic <- log.arsenic - mean (log.arsenic)

fit.9 <- glm (switch ~ c.dist100 + c.log.arsenic + c.educ4 +
  c.dist100:c.log.arsenic + c.dist100:c.educ4 + c.log.arsenic:c.educ4,
  family=binomial(link="logit"))
display (fit.9)
```

```
## glm(formula = switch ~ c.dist100 + c.log.arsenic + c.educ4 +
##     c.dist100:c.log.arsenic + c.dist100:c.educ4 + c.log.arsenic:c.educ4,
##     family = binomial(link = "logit"))
##                          coef.est coef.se
## (Intercept)               0.35     0.04
## c.dist100                -0.98     0.11
## c.log.arsenic             0.90     0.07
## c.educ4                   0.18     0.04
## c.dist100:c.log.arsenic -0.16     0.19
## c.dist100:c.educ4         0.34     0.11
## c.log.arsenic:c.educ4     0.06     0.07
## ---
##   n = 3020, k = 7
```

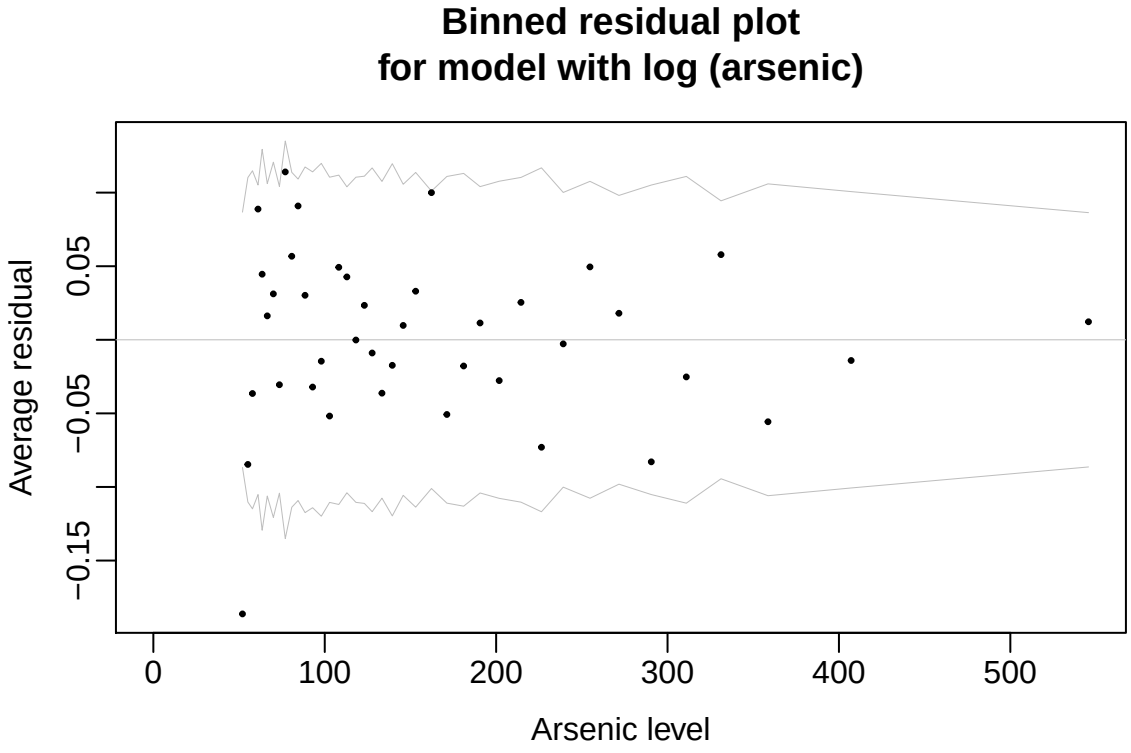
```
##   residual deviance = 3863.1, null deviance = 4118.1 (difference = 255.0)
```

```
fit.9a <- glm (switch ~ dist100 + log.arsenic + educ4 +
  dist100:log.arsenic + dist100:educ4 + log.arsenic:educ4,
  family=binomial(link="logit"))

# graphs for log model

#postscript ("c:/multilevel/arsenic.logmodel.ps", height=3.5, width=4, horizontal=TRUE
plot(c(0,max(arsenic,na.rm=TRUE)*1.02), c(0,1), xlab="Arsenic concentration in well wate
points (arsenic, jitter.binary(switch), pch=20, cex=.1)
curve (invlogit(coef(fit.9a)[1]+coef(fit.9a)[2]*0+coef(fit.9a)[3]*log(x)+coef(fit.9a)[4
curve (invlogit(coef(fit.9a)[1]+coef(fit.9a)[2]*.5+coef(fit.9a)[3]*log(x)+coef(fit.9a)[
text (.25, .80, "if dist = 0", adj=0, cex=.8)
text (2.00, .63, "if dist = 50", adj=0, cex=.8)
```




```
#dev.off ()

pred.9 <- fit.9$fitted.values

#postscript ("c:/books/multilevel/arsenic.logitresids.3b.ps", height=3.5, width=4, hor
br <- binned.resids (arsenic, switch-pred.9, nclass=40)$binned
plot(range(0,br[,1]), range(br[,2],br[,6],-br[,6]), xlab="Arsenic level", ylab="Average
abline (0,0, col="gray", lwd=.5)
n.within.bin <- length(y)/nrow(br)
lines (br[,1], br[,6], col="gray", lwd=.5)
```

```
lines (br[,1], -br[,6], col="gray", lwd=.5)
points (br[,1], br[,2], pch=20, cex=.5)
```




```
#dev.off ()

# calculations for average predictive differences

# simple model
fit.10 <- glm (switch ~ dist100 + arsenic + educ4,
  family=binomial(link="logit"))
display (fit.10)
```

```
## glm(formula = switch ~ dist100 + arsenic + educ4, family = binomial(link = "logit"))
##             coef.est coef.se
## (Intercept) -0.21     0.09
## dist100     -0.90     0.10
## arsenic      0.00     0.00
## educ4        0.17     0.04
## ---
##   n = 3020, k = 4
##   residual deviance = 3910.4, null deviance = 4118.1 (difference = 207.7)
```

```
# avg pred diffs for distance to nearest safe well
b <- coef (fit.10)
hi <- 1
lo <- 0
delta <- invlogit (b[1] + b[2]*hi + b[3]*arsenic + b[4]*educ4) -
         invlogit (b[1] + b[2]*lo + b[3]*arsenic + b[4]*educ4)
print (mean(delta))
```

```
## [1] -0.2044681
```

```
# avg pred diffs for arsenic level
hi <- 1.0
lo <- 0.5
delta <- invlogit (b[1] + b[2]*dist100 + b[3]*hi + b[4]*educ4) -
         invlogit (b[1] + b[2]*dist100 + b[3]*lo + b[4]*educ4)
print (mean(delta))
```

```
## [1] 0.0005434365
```

```
# avg pred diffs for education
hi <- 3
lo <- 0
delta <- invlogit (b[1]+b[2]*dist100+b[3]*arsenic+b[4]*hi) -
         invlogit (b[1]+b[2]*dist100+b[3]*arsenic+b[4]*lo)
print (mean(delta))
```

```
## [1] 0.1167189
```

```
# example model with interaction
fit.11 <- glm (switch ~ dist100 + arsenic + educ4 + dist100:arsenic,
  family=binomial(link="logit"))
display (fit.11)
```

```
## glm(formula = switch ~ dist100 + arsenic + educ4 + dist100:arsenic,
##     family = binomial(link = "logit"))
##                 coef.est coef.se
## (Intercept)     -0.35     0.13
## dist100         -0.60     0.21
## arsenic          0.01     0.00
## educ4            0.17     0.04
## dist100:arsenic  0.00     0.00
## ---
##   n = 3020, k = 5
##   residual deviance = 3907.9, null deviance = 4118.1 (difference = 210.2)
```

```
b <- coef (fit.11)
hi <- 1
lo <- 0
delta <- invlogit (b[1] + b[2]*hi + b[3]*arsenic + b[4]*educ4 +
                   b[5]*hi*arsenic) -
         invlogit (b[1] + b[2]*lo + b[3]*arsenic + b[4]*educ4 +
                   b[5]*lo*arsenic)
print (mean(delta))
```

```
## [1] -0.1944495
```