

# Proportion Treated

Waveley Qiu

2022-02-11

$$h(x) = \log\left(\frac{x}{1-x}\right).$$

Fix  $n \in \mathbb{N}$ , and consider a population of individuals  $i \in \{1, \dots, n\}$ . Let  $X_{1i}, X_{2i}, X_{3i} \sim N(0, 1)$  be covariates associated with individual  $i$ . Let  $T_i$  be a binary variable indicating some treatment, and let the data-generating process for  $T_i$  satisfy

$$h\left(P(T_i = 1 \mid \mathbf{X}_i, \boldsymbol{\alpha})\right) = \alpha_0 + \alpha_1 X_{1i} + \alpha_2 X_{2i} + \varepsilon_i$$

for  $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$ . That is, the probability of receiving treatment  $T_i$  for individual  $i$  is determined by  $X_{1i}, X_{2i}$ , and random noise.

If we want to set a certain proportion of the population to be treated, we want to fix  $E(T_i) = \pi_0$ , for a given  $\pi_0$ .

$$\begin{aligned} E(T_i) &= E[E(T_i \mid \mathbf{X}_i, \boldsymbol{\alpha})] \\ &= E\left[\frac{\exp(\alpha_0 + \alpha_1 X_{1i} + \alpha_2 X_{2i} + \varepsilon_i)}{1 + \exp(\alpha_0 + \alpha_1 X_{1i} + \alpha_2 X_{2i} + \varepsilon_i)}\right] \\ &= \frac{\exp(\alpha_0 + \alpha_1 E(X_{1i}) + \alpha_2 E(X_{2i}) + E(\varepsilon_i))}{1 + \exp(\alpha_0 + \alpha_1 E(X_{1i}) + \alpha_2 E(X_{2i}) + E(\varepsilon_i))} \\ &= \frac{\exp(\alpha_0)}{1 + \exp(\alpha_0)} \\ \implies \pi_0 &= \frac{\exp(\alpha_0)}{1 + \exp(\alpha_0)} \\ \implies \alpha_0 &= \log\left(\frac{\pi_0}{1 - \pi_0}\right) \end{aligned}$$

## Treatment Assignment

```
# assign_treatments() produces a vector of treatments.
# inputs:
# - df: input dataframe
# - p: desired proportion of treated individuals
# - alpha_vec: vector of alpha coefficients excluding alpha_0
# - var_mat: selected variables (also need to specify error term)
# - cov_expval_vec: vector of covariate expected values (if  $X_{-i} \sim N(0, 1)$ , all 0)

assign_treatments <- function(df, p, alpha_vec = c(0.4, 0.5), var_mat = c("L1", "L2"), cov_expval_vec =

  # turn alpha coefficients into vector
```

```

alpha_vec_loc <- alpha_vec %>% as.vector()

# turn covariates from input df into matrix
var_mat_loc <- df %>% select(var_mat) %>% as.matrix()

# turn covariate expected values into vector
cov_expval_vec_loc <- cov_expval_vec %>% as.vector()

# calculating logit parameter
# a linear combination of:
# - alpha coefficients
# - covariate expected values
# - desired proportion of treated individuals

alpha_0_logit_param <- sum(alpha_vec_loc %*% t(cov_expval_vec_loc)) + p

# sanity check
# print(alpha_0_logit_param)

# calculate alpha_0 (if E(X_i) all 0, alpha_0 is just simple logit(p))
alpha_0 <- log(alpha_0_logit_param/(1 - alpha_0_logit_param))

# set treatments
A_logit <- alpha_0 + var_mat_loc %*% alpha_vec_loc # adding alpha_0 to product of var_mat and alpha_v
p_A <- exp(A_logit)/(1 + exp(A_logit)) # probability is inverse logit of A_logit
A <- rbinom(N, 1, p_A) # assigning treatments

# return treatment column
return(A)
}

```