R Fit Prices Given Quantities Logistic Choice with Aggregate Data

Fan Wang

2021-06-26

Contents

1 Prices from Aggregate Shares in Logistic Choice Model

Go to the **RMD**, **R**, **PDF**, or **HTML** version of this file. Go back to fan's REconTools Package, R Code Examples Repository (bookdown site), or Intro Stats with R Repository (bookdown site).

1.1 Observed Shares and Wages

In Estimate Logistic Choice Model with Aggregate Shares, we described and developed the multinomial logistic model with choice over alternatives.

The scenario here is that we have estimated the logistic choice model using data from some prior years. We know that for another set of years, model parameters, and in particular, the effect of alternative-specific prices are the same. We have market share information, as well as all other observables from other years, however, no price for alternatives. The goal is to use existing model parameters and the aggregate shares to back out the alternative-specific prices that would explain the data.

We know that values for choice-specific alternatives, with p_m as the alternative-specific price/wage, are:

$$V_{im} = \alpha_m + \beta \cdot w_m + \epsilon_{im}$$

Choice probabilities are functions of wages. The probability that individual i chooses alternative m is:

$$P(o=m) = P_m = \frac{\exp\left(\alpha_m + \beta \cdot w_m\right)}{1 + \sum_{\widehat{m}=1}^{M} \exp\left(\alpha_{\widehat{m}} + \beta \cdot w_{\widehat{m}}\right)}$$

We observe $P(o=m)$, we know α_m across alternatives, and we know already β. We do not know w_m across alternatives. Fitting means adjusting $\{w_m\}_{m=1}^{M}$ to fit $\{P(o=m)\}_{m=1}^{M}$ observed.

Moving terms around and cross multiplying, we have:

$$
\exp\left(\alpha_m + \beta \cdot w_m\right) = P_m + P_m \sum_{\widehat{m}=1}^{M} \exp\left(\alpha_{\widehat{m}} + \beta \cdot w_{\widehat{m}}\right)
$$
$$
e^{\alpha_m} \exp\left(\beta \cdot w_m\right) = P_m + P_m \sum_{\widehat{m}=1}^{M} e^{\alpha_{\widehat{m}}} \exp\left(\beta \cdot w_{\widehat{m}}\right)
$$

This can be viewed as a linear equation, let $\exp(\alpha_m) = A_m$, which is known, and let $\exp(\beta \cdot w_m) = \theta_m$, which is a function of the known parameter β and unknown price w_m. We have:

$$
P_m = A_m \cdot \theta_m - P_m \sum_{\widehat{m}=1}^{M} A_{\widehat{m}} \cdot \theta_{\widehat{m}}
$$

Suppose $M = 3$, and we label the categories as m, r, a. Note that implicitly we have an outside option category that we are normalizing against. We have then:

$$
\begin{aligned}
P_m &= +A_m \left(1 - P_m\right) \cdot \theta_m - A_r P_m \cdot \theta_r - A_a P_m \cdot \theta_a \\
P_r &= -A_m P_r \cdot \theta_m + A_r \left(1 - P_r\right) \cdot \theta_r - A_a P_r \cdot \theta_a \\
P_a &= -A_m P_a \cdot \theta_m - A_r P_a \cdot \theta_r + A_a \left(1 - P_a\right) \cdot \theta_a
\end{aligned}
$$

Above, we have a system of equations, with three unknown parameters. Regressing the left-hand-side aggregate share vectors against the matrix of right-hand values composed of A and P values generates θ values, which then map one to one to the wages.

An important issue to note is that the "backing-out" procedure does not work with any arbitrary probabilities. Note that $\exp\left(\beta \cdot w_m\right) > 0$. The estimated unknown θ_m will indeed be positive if the probabilities for example sum up to less than 1, however if the probabilities on the left hand side sum to greater than 1, then $\theta_m < 0$ is possible, which leads to no solutions.

Additionally, note that while the procedure here is correct, we can also obtain the wages that can explaine observed probabilities simply by using the log-odds ratio equations. Doing that requires first computing the appropriate log-odds, which requires positive probabilities on the outside option category.

1.2 Simulate Market Share

In this section, we now simulate the above model, with $M = 3$ and data over three periods, and estimate α and β via OLS. Note that $m = 0$ is leisure. This is identical what is what the simulate market share section from Estimate Logistic Choice Model with Aggregate Shares.

First, wages across alternatives.

```
# set seed
set.seed(123)
# T periods, and M occupations (+1 leisure)
it_M <- 3
# define alpha and beta parameters
ar_alpha <- runif(it_M) + 0
fl_beta <- runif(1) + 0
# wage matrix, no wage for leisure
mt_wages <- matrix(runif(1 * it_M) + 1, nrow = 1, ncol = it_M)
colnames(mt_wages) <- paste0("m", seq(1, it_M))
rownames(mt_wages) <- paste0("t", seq(1, 1))
# Show wages
st_caption <- "Wages across occupations"
kable(mt_wages, caption = st_caption) %>% kable_styling_fc()
```

Wages across occupations

	m1	m2	m3
t1	1.940467	1.045556	1.528105

Second, shares across alternatives (and outside option).

```
# Define a probability assignment function
ffi_logit_prob_alpha_beta_1t <- function(ar_alpha, fl_beta, mt_wages) {
  # Dimensions
  it_M <- dim(mt_wages)[2]
  # Aggregate probabilities
  mt_prob_o <- matrix(data = NA, nrow = 1, ncol = it_M + 1)
  colnames(mt_prob_o) <- paste0("m", seq(0, it_M))
  rownames(mt_prob_o) <- paste0("t", seq(1, 1))
  # Generate Probabilities
  # Value without shocks/errors, M+1
  ar_V_hat <- c(0, ar_alpha + fl_beta * mt_wages[1, ])
  # Probabilities across M+1
  mt_prob_o[1, ] <- exp(ar_V_hat) / sum(exp(ar_V_hat))
  return(mt_prob_o)
}
# Show probabilities
ar_prob_o <- ffi_logit_prob_alpha_beta_1t(ar_alpha, fl_beta, mt_wages)
st_caption <- "Participation probabilities across categories"
kable(ar_prob_o, caption = st_caption) %>% kable_styling_fc()
```

Participation probabilities across categories

	m0	m1	m2	m3
t1	0.0506663	0.374767	0.2805663	0.2940004

1.3 Create Inputs for Wages Fit/Estimation

See the linearized structure above, where the LHS is a vector of non-outside-option alternative probabilities. And the RHS is M by M, where each row is multiplying by a different occupation-specific probability, and each column is a different non-wage component of the category specific value (without the error term).

Create the right-hand-side matrix.

```
# A Matrix with share from 1:M columns
mt_rhs_input <- matrix(data = NA, nrow = it_M, ncol = it_M)
colnames(mt_rhs_input) <- paste0("mValNoWage", seq(1, it_M))
rownames(mt_rhs_input) <- paste0("mProb", seq(1, it_M))

# Loop over rows
for (it_m_r in seq(1, it_M)) {
  # +1 to skip the outside-option category
  P_m <- ar_prob_o[it_m_r + 1]
  # Loop over columns
  for (it_m_c in seq(1, it_M)) {
    # Column value for non-wage component of the category-specific value
    A_m <- exp(ar_alpha[it_m_c])
    # Diagonal or not
    if (it_m_r == it_m_c) {
```

```
      fl_rhs_val <- A_m * (1 - P_m)
    } else {
      fl_rhs_val <- -1 * A_m * P_m
    }
    # Fill value
    mt_rhs_input[it_m_r, it_m_c] <- fl_rhs_val
  }
}

# Show rhs matrix
st_caption <- "RHS fit wages matrix"
kable(mt_rhs_input, caption = st_caption) %>% kable_styling_fc()
```

RHS fit wages matrix

	mValNoWage1	mValNoWage2	mValNoWage3
mProb1	0.8335569	-0.8243618	-0.5641281
mProb2	-0.3740494	1.5825131	-0.4223301
mProb3	-0.3919596	-0.6467025	1.0627249

Add in the LHS probability column.

```
# Construct data structure, LHS and RHS, LHS first oclumn
mt_all_inputs <- cbind(ar_prob_o[2:length(ar_prob_o)], mt_rhs_input)
colnames(mt_all_inputs)[1] <- "prob_o"
tb_all_inputs <- as_tibble(mt_all_inputs)
# Show data
st_caption <- "col1=prob in non-outside options; other columns, RHS matrix"
kable(tb_all_inputs, caption=st_caption) %>% kable_styling_fc()
```

col1=prob in non-outside options; other columns, RHS matrix

prob_o	mValNoWage1	mValNoWage2	mValNoWage3
0.3747670	0.8335569	-0.8243618	-0.5641281
0.2805663	-0.3740494	1.5825131	-0.4223301
0.2940004	-0.3919596	-0.6467025	1.0627249

1.4 Solve/Estimate for Wages that Explain Shares

Given the RHS matrix just generated estimate the wages, and check that the match the wages used to simulate the probabilities.

```
# Regression
fit_ols_agg_prob_to_wages <- lm(prob_o ~ . - 1, data = tb_all_inputs)
summary(fit_ols_agg_prob_to_wages)
```

```
##
## Call:
## lm(formula = prob_o ~ . - 1, data = tb_all_inputs)
##
## Residuals:
## ALL 3 residuals are 0: no residual degrees of freedom!
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## mValNoWage1    5.548        NaN     NaN      NaN
## mValNoWage2    2.517        NaN     NaN      NaN
## mValNoWage3    3.855        NaN     NaN      NaN
##
## Residual standard error: NaN on 0 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:    NaN
## F-statistic:   NaN on 3 and 0 DF,  p-value: NA
```

```r
# alpha estimates
ar_coefficients <- fit_ols_agg_prob_to_wages$coefficients
ar_wages_esti <- log(ar_coefficients)/fl_beta
# Compare estimates and true
print(paste0("ar_coefficients=", ar_coefficients))
```

```
## [1] "ar_coefficients=5.54816023677087" "ar_coefficients=2.51744522035287" "ar_coefficients=3.85489489
```

```r
print(paste0("ar_wages_esti=", ar_wages_esti))
```

```
## [1] "ar_wages_esti=1.94046728429384" "ar_wages_esti=1.04555649938993" "ar_wages_esti=1.528105488047"
```

```r
print(paste0("mt_wages=", mt_wages))
```

```
## [1] "mt_wages=1.94046728429385" "mt_wages=1.04555649938993" "mt_wages=1.528105488047"
```