Assignment 5: Detailing Effects and ROI Estimation

Günter J. Hitsch

May 17, 2020

Contents

1 Overview

In this assignment we use a linear probability model and logistic regression to estimate and predict the effect of detailing on the prescription behavior of physicians. We will use two key concepts in data driven marketing: Short-run and long-run effects of promotions, and the return on marketing investment, ROI.

The data are in the data frame `detailing_df`, file `Detailing.RData`. The data consist of a series of prescription decisions and marketing activities of a pharmaceutical company for a panel of 831 physicians. The prescription decisions are for Eli Lilly's erectile dysfunction (ED) drug, Cialis, which was launched in November 2003. The marketing activity that is observed is detailing (sales rep visits to physicians). The data are available for the months of January-May, 2004. Each row in the data set contains an occasion when the physician made a decision to prescribe Cialis or not.

The variables in the data file are:

- `choice` – this variable takes a value = 1 if Cialis was prescribed at the prescription occasion, and = 0 otherwise.
- `det` – this variable takes positive integer values and indicates the number of detailing visits to a physician in the 30-day period immediately preceding the prescription occasion.
- `lag_det` – this variable indicates the number of detailing visits to the physician in the lagged 30-day period (i.e. between 31 and 60 days before the prescription occasion).
- `type` – this is a categorical variable that takes an integer value between 1 and 3. `type = 1` indicates that the physician is a light prescriber in the category, i.e. gets few patients with ED, `type = 2` indicates a moderate prescriber, and `type = 3` indicates a heavy prescriber.

```r
library(ggplot2)
library(tidyverse)
library(dplyr)
library(tidyr)
library(lfe)
load('Detailing.RData')
```

2 Data description

Show a histogram of detailing to document how frequently doctors are exposed to detailing for Cialis. Then examine if there are any systematic differences in the amount of detailing to different types of physicians (remember how to summarize data separately for different groups).

```
hist(detailing_df$choice)
detailing_type = detailing_df %>%
   group_by(type) %>%
   summarize(detailing = sum(choice*(det+lag_det)))
plot(detailing_type$type, detailing_type$detailing)
```

3 Linear probability model of prescription choice

3.1 Short-run effect of detailing

Estimate a linear probability model to predict the prescription choice of Cialis using the amount of current detailing. The dependent variable is $y =$ `choice`. For now do not use lags. Interpret the detailing coefficient estimate.

```
lm_sr = glm(choice ~ det, data = detailing_df)
```

3.2 Long-run effect of detailing

Now allow for long-run effects of detailing. Re-estimate the linear probability model, now also including lagged detailing as an independent variable.

To interpret the results, think of current and past detailing creating a detailing stock, similar to the adstock (goodwill) model discussed in class. In the case with one lag:

$$\texttt{detstock} = \texttt{det} + \delta \cdot \texttt{lag\_det}$$

The detailing stock affects prescription choices:

$$\begin{aligned}
\texttt{choice} &= \beta_0 + \gamma \cdot \texttt{detstock} + \epsilon \\
&= \beta_0 + \gamma \cdot (\texttt{det} + \delta \cdot \texttt{lag_det}) + \epsilon \\
&= \beta_0 + \beta_1 \cdot \texttt{det} + \beta_2 \cdot \texttt{lag_det} + \epsilon
\end{aligned}$$

Using the estimated regression coefficients, β_1 and β_2, infer the carry-over factor δ.

Interpret the estimate of γ and the estimated carry-over.

```
lm_lr = lm(choice ~ det + lag_det, data = detailing_df)
lm_lr_beta0 = lm_lr$coefficients[1]
lm_lr_beta1 = lm_lr$coefficients[2]
lm_lr_beta2 = lm_lr$coefficients[3]
gamma = lm_lr_beta1
theta = lm_lr_beta2/gamma
data.frame(gamma = gamma, carry_over_factor = theta)
# The gamma is estimiated to be,
gamma
# The carry over factor is estimated to be,
theta
```

The gamma is estimated to be ‘r gamma’

3.3 Optional: Estimate δ using a grid search

We can alternatively estimate the carry-over parameter δ using a grid search. Indeed, **if the detailing stock model included more than one lag, such an approach would be necessary**.

We use the range $\delta = 0.0, 0.01, 0.02, \ldots, 0.99$ and employ the following algorithm:

(i) Given one of the δ values in the grid, calculate the `detstock` variable
(ii) Estimate the linear probability model
(iii) Record the AIC (Akaike information criterion)

The AIC is used for model selection. Among all candidate models, based on different δ values, we choose the model with the lowest AIC.

Note that the AIC can be directly calculated from a linear probability or logistic regression (more generally: GLM) model output:

```
fit = lm(...)
AIC(fit)

fit_logistic = glm(...)
AIC(fit_logistic)
```

The repetitive grid search process above can be easily automated using a loop. In R, the syntax for a basic loop is as follows:

```
for (k in a:b) {
   ...
}
```

Here, `a` and `b` are two numbers, and `b` will be larger than `a`. Suppose that `a = 2` and `b = 5`. Then R will loop over the values `k = 2, 3, 4, 5` and execute the code inside the brackets `{...}` for each separate value of `k`.

Below, the loop is used to estimate the linear probability model separately for each value of δ. The script stores the AIC value for each δ in the data frame `aic_df`.

```
# Data frame with delta and AIC value columns
aic_df = data.frame(
   delta = seq(from = 0.0, to = 0.99, by = 0.01),
   AIC   = 0
)

# Loop over all delta values
n_delta = nrow(aic_df)
for (i in 1:n_delta) {

   # Pick delta corresponding to index i
   delta = aic_df[i, "delta"]

   # Construct detailing stock for the delta value
   detailing_df = detailing_df %>%
      mutate(detstock = det + delta*lag_det)

   # Estimate the linear probability model
   fit_i = lm(choice ~ detstock, data = detailing_df)

   # Store the delta and AIC values
   aic_df[i, "delta"] = delta
   aic_df[i, "AIC"]   = AIC(fit_i)
}

# delta value corresponding to lowest AIC value
best_delta = aic_df[which.min(aic_df$AIC), "delta"]
```

4 Logistic regression model of prescription choice

Estimate a logistic regression model that predicts the prescription choice using current and past detailing.

Calculate the carry-over factor implied by the logistic regression estimates.

For conceptual clarity when performing the ROI calculations below, estimate a logistic regression model that directly includes the detailing stock as independent variable:

(i) Use `mutate` to add a `detstock` variable to the `detailing_df` data frame. When calculating `detstock`, use the carry-over factor, δ, that you estimated above.
(ii) Estimate a logistic regression with `detstock` as the independent variable.

Save the logistic regression output, which we need below to predict prescription probabilities.

5 ROI calculations

Data:

- Each doctor sees 15.7 ED patients (on average) per month
- The revenue from one prescription written for Cialis is $100
- The cost of each detailing call is $60
- Note that the marginal cost of a prescription drug is essentially 0

Focus on the ROI from one additional detailing call. To calculate the ROI, you first need to predict the incremental volume (prescriptions) from additional detailing. The expected change in the total number of prescriptions written per month depends on the change in the prescription probability times the number of ED patients that a doctor sees per month.

5.1 Change (difference) in prescription probabilities

To calculate the change in the prescription probability for a one unit increase in the detailing stock predict the prescription probabilities using the estimated logistic regression model of prescription choice.

To predict the prescription probabilities at the values of the independent variables observed in the data, use the `predict` function as follows:

```
Pr_0 = predict(fit_logistic, type = "response")
```

Alternatively, you could use `mutate` to add the predicted probabilities as a new column to `detailing_df`:

```
detailing_df = detailing_df %>%
   mutate(Pr_0 = predict(fit_logistic, type = "response"))
```

Here, `fit_logistic` is the logistic regression estimation result. R uses the estimated regression coefficients, $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_K$, and then predicts the probabilities using the formula

$$\Pr\{y = 1 | x_1, \ldots, x_K\} = \frac{\exp(\hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_K x_K)}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_K x_K)}.$$

Predict the prescription probabilities at the sample values of detailing. Summarize and plot a histogram of the predicted prescription probability variable.

Note: `Pr_0`, as created in the first code chunk above, is a vector, not a data frame. If you use ggplot2 to plot a histogram you either need to convert `Pr_0` to a data frame or, alternatively, add `Pr_0` to an existing data frame such as `detailing_df`.

```
ggplot(as.data.frame(Pr_0), aes(x = Pr_0)) + ...
```

Now predict the prescription probability for a one unit increase in the detailing stock. First, create a copy of the original data (this allows you to preserve the original data):

```
new_detailing_df = detailing_df
```

Then increase the detailing stock in the new data by 1 (use `mutate`) and predict the corresponding prescription probabilities:

```
Pr_1 = predict(fit_logistic, newdata = new_detailing_df, type = "response")
```

Or you can use the pipe, which avoids having to copy the original data.

```r
Pr_1 = detailing_df %>%
   mutate(<add 1 to detstock>) %>%
   predict(fit_logistic, newdata = ., type = "response")
```

Calculate the average increase in the predicted prescription probabilities based on the mean difference between `Pr_1` and `Pr_0`. Is the prediction similar to the estimate of the effect of detailing from the linear probability model?

5.2 Short-run ROI

Now predict the incremental prescription volume and the corresponding incremental profit from one additional detailing call. Finally, calculate the corresponding ROI. Is the ROI positive? What recommendation would you make based on this ROI?

You may use Excel to predict the ROIs.

5.3 Long-run ROI

The ROI calculated above is the short-run ROI—it only captures the effect of detailing on revenues in the same 30-day period. However, since we know that current detailing also affects future prescriptions, we need to account for the fact that detailing also affects profits in the next period (between 31 and 60 days after the detailing call).

It is important to understand exactly what we would like to calculate: The total incremental prescription volume if we increase detailing by one unit only in this period *but not in the next period*. We already predicted the current period incremental volume. To calculate the incremental volume in the next period, remember how the detailing stock is defined:

$$\texttt{detstock} = \texttt{det} + \delta \cdot \texttt{lag\_det}$$

The detailing stock in the next 31 to 60 day period will increase due to the one-unit increase in `lag_det`, but not due to `det`, because we only increase detailing in the current period.

To calculate the change in the detailing stock for a one-unit increase in `lag_det` use the carry-over factor that you estimated using the logistic regression model before. Then predict the prescription probabilities for the increase in `lag_det`, and finally calculate the average increase in the prescription probabilities due to the increase in lagged detailing.

Now calculate the long-run ROI based on the total incremental prescription volume, i.e. the sum of the total incremental volume in this period and in the next period. Do the short-run and long-run ROIs have the same implications for how the currently used detailing schedule should be adjusted?

6 Physician type-specific ROIs

Recall that the data contain information on physician type: `type = 1` indicates that the physician is a light prescriber in the category, `type = 2` indicates a moderate prescriber, and `type = 3` indicates a heavy prescriber.

Estimate logistic regression models for each physician type separately, using only the data for the specific type. For simplicity, however, I recommend to use the *same carry-over factor* δ that you found using the estimated logistic regression model before.

Calculate the short-run and long-run ROIs for each type. You will need to use these data for the ROI calculations:

Type	No. of patients per month	Revenue per patient	Cost of detailing call
1	1.9	$100	$60
2	7.2	$100	$60
3	30.2	$100	$60

Interpret the ROI results. Do you recommend to reallocate the detailing expenditures based on the ROI estimates? If so, how?

Once again, if you repeatedly perform calculations that have many steps you may, **optionally**, automate the task. Below is a script that loops over all physician types `k`. The script subsets the physician type-specific data, estimates the logistic regression model, and then predicts the incremental probabilities.

```
# Summary data frame: Incremental prescription probabilities by physician type
type_df = data.frame(
   type        = 1:3,
   diff_Pr     = 0,          # Effect in period of detailing increase
   diff_Pr_next = 0          # Effect one period after the detailing increase
)

# Loop over all physician types
for (k in 1:3) {

   # Choose data for type k
   detailing_type_k = <choose type k data from detailing_df>

   # Predict prescription probabilities for type k
   fit_type_k = <logistic regression for type k>

   # Effect on prescription probability from a one unit increase in detailing
   Pr_0 = predict(fit_type_k, type = "response")
   Pr_1 = <predicted probability for type k from 1 unit increase in detailing>

   type_df[k, "diff_Pr"] = mean(Pr_1 - Pr_0)

   # Effect on the prescription probability in the next period
   Pr_1_next = <predicted probability in next period for type k
               from 1 unit increase in detailing in this period>

   type_df[k, "diff_Pr_next"] = mean(Pr_1_next - Pr_0)
}
```