Problem Set3

Sachin Badole

10/29/2020

```
# Question 1.
load(file = "market_level.R")
#View(datam)
load(file = "market_airline_level.R")
#View(datama)
```

(1) Estimate a linear probability model, predicting whether American Airlines enters a market as a function of the number of competitors. Note: American Airlines' ticket carrier id is "AA".

```
datama_v1 <- datama %>%
  group_by(origin_airport_id, dest_airport_id) %>%
  mutate(carrier_market_in = ("AA"%in%ticket_carrier)*1)

#
datam_v1 <- merge(datam,datama_v1,by=c("origin_airport_id","dest_airport_id"))

# Drop some dublicates values
datam_v2 <- datam_v1[!duplicated(datam_v1[c(1,2)]),]

#
datam_v2$num_competitors <- datam_v2$num_carriers.x - datam_v2$carrier_market_in

write.csv(datam_v2,"/Users/sachin/Downloads/Econ_725/Problem Sets/Problem set 3/data_airm.csv",
          row.names = FALSE)

#
set.seed(0)

rn <- sample(seq_len(nrow(datam_v2)),size = 1000)
test <- datam_v2[rn,]
train <- datam_v2[-rn,]


# linear probability model
linear_pro_model = lm(carrier_market_in~num_competitors,data=train)
#
train$linear_pro_model_pred = predict(linear_pro_model,train)
```

2) Repeat (1) using a logit model instead of a linear probability model.

```r
# http://r-statistics.co/Logistic-Regression-With-R.html

# Logit Model
logit_model <- glm(carrier_market_in~num_competitors,data=train,
                   family=binomial(link="logit"))
#
train$logit_model_pred <- plogis(predict(logit_model, train))
```

3) Repeat (1) using a probit model instead of a linear probability model.

```r
# Probit Model
probit_model = glm(carrier_market_in~num_competitors,data=train,
                   family=binomial(link="probit"))
#
#test$predicted_p <- plogis(predict(probit_model, train))
#
train$probit_model_pred = predict.glm(probit_model,train,type="response")
```

4) Compute non-parametric estimates of the conditional probabilities of entering. (ie compute the conditional probability of entering conditional on each number of competitors directly from the data).

```r
test1 <- rep(0,times = length(unique(train$num_competitors)))

for (i in 1:length(test1)){
  p <- unique(train$num_competitors)[i]

  # calculate joint probablility of carrier in market and number of competitors i
  if_default_tranche <- ifelse(((train$num_competitors == p)
                               & (train$carrier_market_in == 1)),1,0)

  sum_default_tranche <- sum(if_default_tranche)

  # calculate probability of number of competitors
  if_tranche <- ifelse(train$num_competitors == p, 1, 0)
  sum_tranche <- sum(if_tranche)

  #
  test1[i] <- (sum_default_tranche/nrow(train)) / (sum_tranche/nrow(train))
}
probs_nums <- data.frame(cbind(test1, unique(train$num_competitors)))
names(probs_nums) <- c("probability","num_competitors")
probs_nums <- probs_nums[order(probs_nums$num_competitors),]

# merge
test_nonpara <- merge(test,probs_nums,by=c("num_competitors"))
train <- merge(train,probs_nums,by=c("num_competitors"))

kable(probs_nums)
```

	probability	num_competitors
2	0.1081633	1
5	0.5703839	2
1	0.8336331	3
3	0.9428904	4
4	0.9422604	5
6	0.9896552	6
7	1.0000000	7
9	1.0000000	8
8	1.0000000	9
10	1.0000000	10

5) Plot the fitted values of each regression in one graph (i.e. estimated probabilities on the y-axis and the number of competitors on the x-axis). In words, explain the coeifficients of the first three models. How do the estimated relationships compare? Should we interpret these relationships causally? Are the estimates for the probit and logit similar? Should we have expected this ex ante?

Ans:
Explain the coeifficients of the first three models.
(1) The coefficient of the linear model is 0.142131, it means that 14.21% of American Airlines enters a market and it is statistically significant.
The coefficient of the logit model is 1.34991, it means that 135% of American Airlines enters a market and it is statistically significant.
The coefficient of the probit model is 0.72860,it means that 72.86% of American Airlines enters a market and it is statistically significant.

How do the estimated relationships compare?
(2) The estimated linear probability model is continuously increasing and it is different from the other two models (i.e. logit model, probit model). As per below graph, we can see that these two models somewhat estimate similar results. The linear model is not a good. As number of competitors in American Airlines enters in market increases as the probability is an increase . These shows in logit, probit, and non-paprmetric models.

Should we interpret these relationships causally?
(3) It depends. The correlation does not necessary mean causation. we can not interprete these relationship causally.

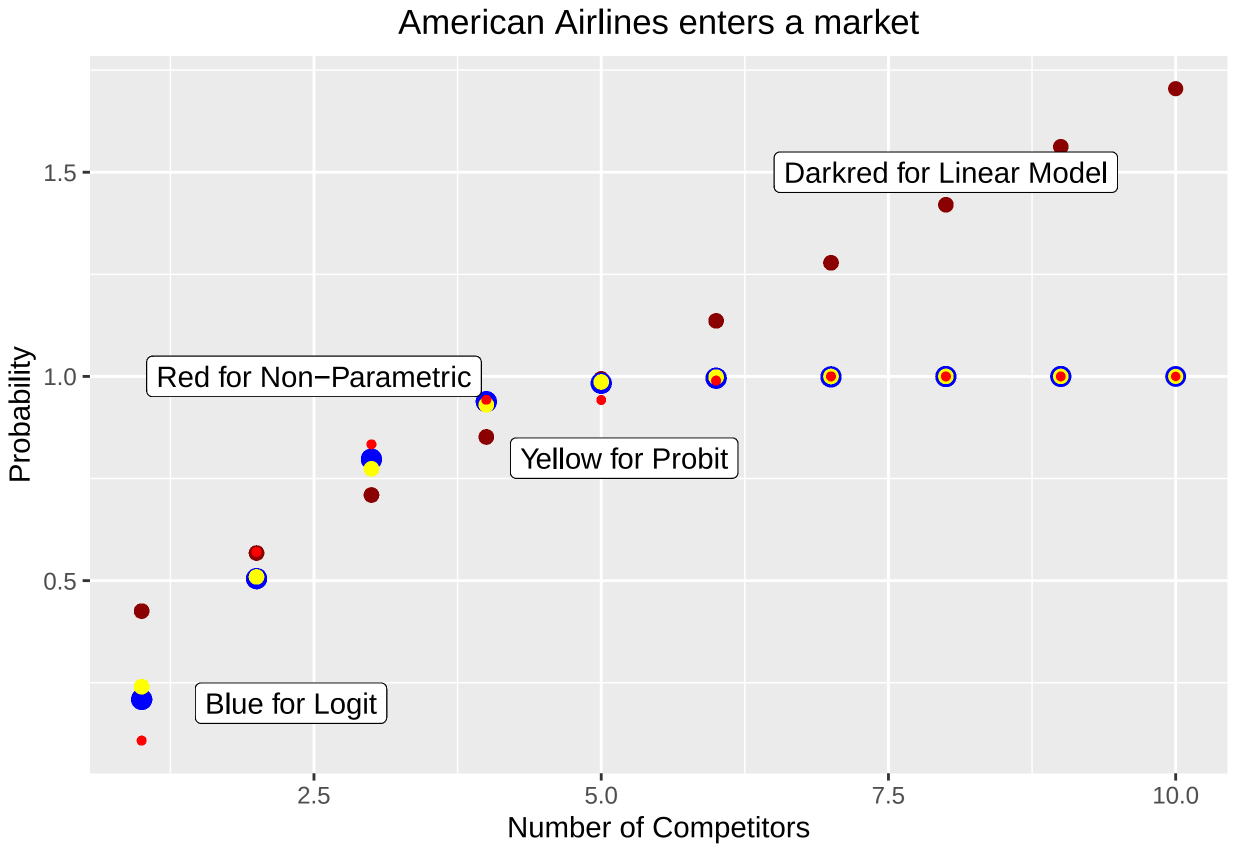
Are the estimates for the probit and logit similar?
(4) **Yes**, As per below graph, we can see that the estimates for the probit and logit are similar.

Should we have expected this ex ante?
(5) Yes

```r
datam <- subset(train, select=c(num_competitors, linear_pro_model_pred,
                                logit_model_pred, probit_model_pred,probability))

ggp <- ggplot(datam, aes(x=num_competitors)) +
  geom_point(aes(y = linear_pro_model_pred), color = "darkred", size = 2) +
  geom_point(aes(y = logit_model_pred), color="blue", size = 3) +
  geom_point(aes(y = probit_model_pred), color="yellow", size = 2) +
  geom_point(aes(y = probability), color="red", size = 1) +
  ggtitle("American Airlines enters a market") +
  xlab("Number of Competitors") +
  ylab("Probability")  +
  theme(plot.title = element_text(hjust = 0.5))

ggp +
  geom_label(label="Darkred for Linear Model", x=8, y=1.5, label.size = 0.15) +
  geom_label(label="Red for Non-Parametric", x=2.5, y=1, label.size = 0.15) +
  geom_label(label="Blue for Logit", x=2.3, y=0.2, label.size = 0.15 ) +
  geom_label(label="Yellow for Probit", x= 5.2, y=0.8, label.size = 0.15)
```



6) Obviously other covariates matter in predicting whether or not American will enter a particular route. In addition to the number of competitors, add the average market distance, market size, hub route indicator, vacation route indicator, slot controlled indicator, and market income to the set of predictors. Fit to the data L1 regularized logistic regression (ie Lasso for logit, pg 125-126 ESL) where the full model includes all squared terms and second-order cross terms. Using a 10-fold cross validation procedure, find the optimal value of lambda.

```
#lasso model

covar <- c("num_competitors", "average_distance_m.x", "market_size.x",
           "hub_route.x", "vacation_route.x", "slot_controlled.x", "market_income.x")
```

```
polyvars = data.frame(poly(as.matrix(datam_v2[,covar]),degree=2,raw=T))

traindata <- data.table(data.frame(datam_v2$carrier_market_in[-rn], polyvars[-rn,]))
names(traindata)[1] <- "carrier_market_in"

testdata <- data.table(data.frame(datam_v2$carrier_market_in[rn], polyvars[rn,]))
names(testdata)[1] <- "carrier_market_in"

train_x <- as.matrix(traindata[,!"carrier_market_in"])
train_y <- as.matrix(traindata[,"carrier_market_in"])


test_x <- as.matrix(testdata[,!"carrier_market_in"])
test_y <- as.matrix(testdata[,"carrier_market_in"])


cvg_lasso_lambda <- cv.glmnet(train_x,train_y, type.measure = "mse",
                              nfolds = 10, alpha = 1)$lambda.min

cvg_lasso <- glmnet(train_x,train_y,family="binomial",
                    alpha = 1,lambda = cvg_lasso_lambda )

lasso_mse<-mean((test_y - predict(cvg_lasso,test_x,type="response",
                                  s=cvg_lasso_lambda))^2)
```

7) Calculate the MSE on the test set for each of your 5 models and put them in a table. Explain your results.

Ans:
In my view, the linear model is not good because the MSE for this model is more as compared to the other model. The MSE for Logit and probit has a small difference and hence it depends on which one should we use. The non-parametric model is better than logit and probit because it has a small MSE as compare to the other models.

Lastly, we can see that in the below table, the MSE for lasso is samller than the logit, probit, and non-parametric models. Hence, Lasso model is the best model.

```
# Linear MSE
lm_mse <- mean((test$"carrier_market_in" - predict(linear_pro_model,test))^2)

# Logit model MSE
logit_mse <- mean((test$"carrier_market_in" - predict.glm(logit_model,test, type="response"))^2)

# Probit model MSE
probit_mse <- mean((test$"carrier_market_in" - predict.glm(probit_model,test, type="response"))^2)

# Non-parametrice MSE
nonpr_mse <- mean((test_nonpara$carrier_market_in - test_nonpara$probability)^2)

print(paste("MSE for Linear Model", lm_mse))
```

```
## [1] "MSE for Linear Model 0.113445685203697"
```

```
print(paste("MSE for Logit Model", logit_mse))
```

```
## [1] "MSE for Logit Model 0.0942584850412098"
```

```
print(paste("MSE for Probit Model", probit_mse))
```

```
## [1] "MSE for Probit Model 0.0951334556147287"
```

```
print(paste("MSE for Non-Parametrice", nonpr_mse))
```

```
## [1] "MSE for Non-Parametrice 0.0939631049165887"
```

```
print(paste("MSE for Lasso", lasso_mse))
```

```
## [1] "MSE for Lasso 0.0855571027722133"
```

```
x <- data.frame("Model" = c("Linear Model", "Logit Model", "Probit Model", "Non-Parametrice", "Lasso"),
  "MSE" = c(lm_mse, logit_mse, probit_mse, nonpr_mse, lasso_mse), stringsAsFactors = FALSE)

kable(x)
```

Model	MSE
Linear Model	0.1134457
Logit Model	0.0942585
Probit Model	0.0951335
Non-Parametrice	0.0939631
Lasso	0.0855571