DSC5101 ANALYTICS IN MANAGERIAL ECONOMICS
Group Project 1
Coffee Demand and Supply Function Prediction

Cho Zin Tun (Student ID:A0098996W, Email:e0230036@u.nus.edu)

Derek Li (Student ID:A0176652X, Email:e0232256@u.nus.edu)

Sophia Yue (Student ID:A0176615Y, Email:e0232219@u.nus.edu)

Jingjing Song (Student ID:A0077954M, Email:e0229901@u.nus.edu)

1. Introduction

Coffee is a necessity to many people around the world. It is also an important commodity in the international trading markets. Most traders have huge interests to predict the market demand and price function of coffee, based on information from supply (bean price, labor cost and etc), consumer (income and etc) and market (substitute tea price, other goods price index and etc). There are also many research papers discussing about such prediction. Basically different markets show different characteristics, so particular mathematical models are needed to be derived from individual market's data for demand prediction.

The purpose of this group project is to estimate the demand and supply functions by using coffee consumption production data from Dutch market. A simplified small dataset of 84 observations and 14 variables were used. Basic analytical tools including Ordinary Least Square (OLS) regression and Two Stage Least Square (TSLS) regression were applied. Section 2 of this report will illustrate the methodology of prediction, including how variables are chosen and a description of the model. Section 3 will give interpretation on the results, and discuss about significance and endogeneity of the predictors. Section 4 shows robustness test of the model using root mean square error. Section 5, which is the last section, briefly discuss about the limitations of the models.

2. Methodology

2.1 Choice of Models

Linear regression was applied in this project in order to estimate coffee demand and supply functions with the coffee consumption and production data. By assuming constant price elasticity η and linear relationship between price and the control variables for both demand and supply curve, we first derive the following OLS regression models:

Demand Function: $q^D = \alpha_0 - \eta^D \times p + \text{control variables} + \varepsilon^D$ *(where* $q^D = lnQ, p = lnP$*)*
Supply Function: $q^S = \beta_0 + \eta^S \times p + \text{control variables} + \varepsilon^S$ *(where* $q^S = lnQ, p = lnP$*)*

By observing price variable P being correlated with the error term ε, we further applied the TSLS regression to eliminate the endogeneity problem. The formula for predicted price $\hat{p}$ was constructed by taking the control variables from both demand and supply functions. We subsequently substituted the predicted price back to the OLS models to get the final demand and supply functions. Hausman test was conducted on the TSLS models to examine the validity of the instrumental variables.

2.2 Choice of Variables

Table 1 below shows a summary of the variables used in estimating the demand and supply functions. Variables in terms of dollar values were first adjusted by being divided by the price of other goods oprice, to eliminate the effect of inflation. Income and season dummy q4 were used as control variables for demand functions, while price of coffee bean and price of labour were used in supply function. Price of tea was not taken as a variable as it turned out to be insignificant after an initial experiment on the models. And the reason for using only season dummy 4 was because we observed that only q4 is affecting the outcome significantly.

Variable Name	Calculation	Use in Models	
		Demand Function	**Supply Function**
ln_qu	ln(qu)	Dependent Variable	
ln_cprice	ln(cprice/oprice)	Independent Variable	
ln_incom	ln(incom/oprice)	Control Variable	Instrument Variable
ln_bprice	ln(bprice/oprice)	Instrument Variable	Control Variable
ln_wprice	ln(wprice/oprice)	Instrument Variable	Control Variable
q4	q4	Control Variable	Instrument Variable

Table 1: Variable Summary

Therefore, the final demand and supply functions were as follow.

OLS Regression Models

- Demand Function: $ln\_qu = \alpha_1\, ln\_cprice + \alpha_2\, ln\_incom + \alpha_3 q4 + \varepsilon_{D1}$
- Supply Function: $ln\_qu = \beta_1 ln\_cprice + \beta_2 ln\_bprice + \beta_3 ln\_wprice + \varepsilon_{S1}$

TSLS Regression Models

- Demand Function: $ln\_qu = \alpha_3\, \hat{p} + \alpha_4\, ln\_incom + \alpha q4 + \varepsilon_{D2}$
- Supply Function: $ln\_qu = \beta_4\, \hat{p} + \beta_5 ln\_bprice + \beta_6 ln\_wprice + \varepsilon_{S2}$
  Where $\hat{p} = \vartheta_1 ln\_incom + \vartheta_2 q4 + \vartheta_3 ln\_bprice + \vartheta_4 ln\_wprice$

3. Result Interpretation

3.1 Demand Function

Applying OLS regression directly to our demand function, the coefficient of cprice was found to be -0.31219, with standard error being 0.14544. As for the TSLS regression, the function was proved to be valid as it passed the Hausman Test with R Square being close to zero (0.001819393), and the coefficient was -0.32593, with a standard error of 0.15350.

OLS Regression Result

```
Residuals:
     Min       1Q   Median       3Q      Max
-0.19648 -0.07281 -0.01323  0.06598  0.31804

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.09637    2.66196  -1.915  0.06291 .
ln_cprice   -0.31219    0.14544  -2.147  0.03811 *
ln_incom     0.73159    0.37400   1.956  0.05764 .
q4           0.10963    0.04045   2.710  0.00994 **
---

Residual standard error: 0.1116 on 39 degrees of freedom
Multiple R-squared:  0.3089, Adjusted R-squared:  0.2558
F-statistic: 5.812 on 3 and 39 DF,  p-value: 0.002204
```

TSLS Regression Result

```
Residuals:
     Min       1Q   Median       3Q      Max
-0.20481 -0.05888 -0.00581  0.06939  0.32190

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.17264    2.67860  -1.931   0.0608 .
P_hat       -0.32593    0.15350  -2.123   0.0401 *
ln_incom     0.74646    0.37810   1.974   0.0555 .
q4           0.10911    0.04054   2.692   0.0104 *
---

Residual standard error: 0.1117 on 39 degrees of freedom
Multiple R-squared:  0.3074, Adjusted R-squared:  0.2541
F-statistic: 5.769 on 3 and 39 DF,  p-value: 0.002298
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Table 2: Regression Results For Demand Function

Comparing the results from the two regression models, their coefficient for price, R square and significance level for variables are quite close, thus the original OLS model with smaller standard error was chosen as the final demand function:

ln_qu = -5.10 - 0.312 x ln_cprice + 0.731 x ln_incom + 0.110 x q4 + Error

where consumption of roasted coffee (quantity demanded) is relatively price inelastic i.e. not as responsive to a change in the price, while being positively impacted by an increase in income most likely due to higher purchasing power. It also seems that more coffee is consumed during the fourth quarter in a year during the colder months.

3.2 Supply Function

Applying OLS regression directly to our supply function, the price coefficient was calculated to be -0.04384. Using the TSLS regression, the supply price coefficient was found to be 10.008. The huge difference between results of OLS and TSLS models indicated that the price in supply function is very much affected by the endogeneity problem. As the coefficient of supply function should not be negative, the TSLS model should be applied instead, though we suffer from a greater standard error.

OLS Regression Result

```
Residuals:
     Min       1Q   Median       3Q      Max
-0.23210 -0.07465 -0.02277  0.06856  0.42852

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.60866    4.89729  -0.737    0.466
ln_cprice   -0.04384    0.53193  -0.082    0.935
ln_bprice   -0.08191    0.28658  -0.286    0.777
ln_wprice    1.01920    1.49901   0.680    0.501

Residual standard error: 0.1301 on 39 degrees of freedom
Multiple R-squared:  0.06063,    Adjusted R-squared:  -
0.01162
F-statistic: 0.8391 on 3 and 39 DF,  p-value: 0.4807
```

TSLS Regression Result

```
Residuals:
     Min       1Q   Median       3Q      Max
-0.22526 -0.05263 -0.00050  0.06922  0.38445

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    3.709      5.431   0.683   0.4987
Phat_2        10.008      4.084   2.450   0.0189 *
ln_wprice     -6.794      3.446  -1.971   0.0558 .
ln_bprice     -5.263      2.107  -2.498   0.0168 *

## Residual standard error: 0.1211 on 39 degrees of
freedom
## Multiple R-squared:  0.1858, Adjusted R-
squared:  0.1232
## F-statistic: 2.967 on 3 and 39 DF,  p-value: 0.04365
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Table 3: Regression Results For Supply Function

The R-squared results (3.430953e-29) from the Hausman Test shows that instrument variables and residual errors are not correlated. Thus we conclude that the supply function estimation based on $\hat{p}$ is valid. The supply function

ln_qu = 3.709 + 10.008 x ln_cprice - 6.794 x ln_bprice - 5.263 x ln_wprice+ Error

indicates that the supply of roasted coffee is highly price sensitive. Quantity of roasted coffee supplied is negatively correlated to the cost of production of coffee - the price of coffee beans as well as the wages of the coffee suppliers.

Please refer to Appendix Step 4 – 6 and 8 – 10 for how the demand and supply functions were derived respectively using R.

4. Model Validation

Overfitting or underfitting might be introduced while building a model. Thus, it is necessary to check robustness of the model. In order to test robustness, we have randomly split the data into two datasets: train and test. We trained the model using train dataset and checked its robustness with test dataset. Root Mean Square Error (RMSE) is used as a measurement for robustness testing as it indicates the absolute fit of the model to the data-how close the observed data points are to the model's predicted values. Please refer to Appendix Step 12 for R code used to get RMSE for demand and supply functions.

We got RMSE of 0.02399645 and 0.2860978 for demand and supply function respectively. From the result, we can conclude that the model is robust.

5. Limitations

Although cross validation has been done to verify the robustness of the model, the original dataset is too small, so the model derived is not conclusive to represent the actual function. However, if larger dataset is available, exactly same approach can be applied to get a more accurate prediction.

The project data itself does not reveal market structure, so the market type was not discussed and investigated. Different market structures, such as oligopolistic and monopolistic competition will have different impact on the demand and supply function. In a real world prediction, it is necessary to understand the market structure prior to modelling so the results could be explained in a rational way.

APPENDIX

Demand Function

Step 1:

Identify Variables for Demand and Supply functions respective. The following variables are given in the data:

maand year and month of observation
year year of observation
month month of observation
qu per capita consumption of roasted coffee in kg
cprice price of roasted coffee per kg in current guilders
tprice price of per kg tea in current guilders
oprice price index for other goods
income income per capita in current guilders
q1 season dummy 1
q2 season dummy 2
q3 season dummy 3
q4 season dummy 4
bprice price of coffee beans per kg in current guilders
wprice price of labor per man hours (work 160 hours per month)

The following variables were believed to directly affect the supply function and was tested in the initial models: **cprice**, **bprice**, **wprice**, **q1**, **q2**, **q3**, **q4**
The following variables were believed to directly affect the demand function and was tested in the initial models: **cprice**, **tprice**, **oprice**, **income**, **q1**, **q2**, **q3**, **q4**
However, some variables (.e.g tprice, q1,q2, q3) were found not significant from the result of regression. So they were omitted in the final model.

Step 2:

To test the robustness of the model, we have randomly split the data into train data and test data equally.

```
#library(readxl)
#RawData <- read_excel("~/Documents/MSBA/DAO5101/project/Project1Data.xlsx")

library(readr)
Project1Data <- read_csv("C:/Users/sophi/Google Drive/MSBA/DSC5101 Analytics in Managerial Economics/Group work/Case 1/Project1Data.csv")

## Parsed with column specification:
## cols(
##   maand = col_character(),
##   year = col_integer(),
##   month = col_integer(),
##   qu = col_double(),
```

```
##   cprice = col_double(),
##   tprice = col_double(),
##   oprice = col_double(),
##   incom = col_double(),
##   q1 = col_integer(),
##   q2 = col_integer(),
##   q3 = col_integer(),
##   q4 = col_integer(),
##   bprice = col_double(),
##   wprice = col_double()
## )
```

```
View(Project1Data)

RawData <- Project1Data

data <- RawData[1:14]

library(caTools)
set.seed(4352)
split = sample.split(data$qu, SplitRatio = 0.435)
train_data = subset(data, split == TRUE)
test_data = subset(data, split == FALSE)
train_data <- rbind(train_data, test_data[1:2,])
```

Step 3:

The tea price, coffee price, bean price, wage price, and income will all be divided by price index of other goods. This will normalize the price and eliminate the inflation factor.

```
#setwd("C:/Users/SGDELI/Desktop/MSBA Bootcamp/DSC5101 ANALYTICS IN MANAGERIAL ECONOMICS/Homework and Group Project/Group Project 1")

ln_qu <- log(train_data$qu)
ln_cprice <- log(train_data$cprice/train_data$oprice)
ln_bprice <- log(train_data$bprice/train_data$oprice)
ln_wprice <- log(train_data$wprice/train_data$oprice)
q1 <- train_data$q1
q2 <- train_data$q2
q3 <- train_data$q3
q4 <- train_data$q4
ln_tprice <- log(train_data$tprice/train_data$oprice)
oprice <- train_data$oprice
ln_incom <- log(train_data$incom/train_data$oprice)
```

Step 4:

Run Simple Linear Regression Directly for Demand Function. Tea price, q1, q2, q3 were found not significant. So they were omitted in the demand function.

```
Demand_Model_OLS <- lm(ln_qu ~ ln_cprice + ln_incom + q4)
summary(Demand_Model_OLS)
```

```
##
## Call:
## lm(formula = ln_qu ~ ln_cprice + ln_incom + q4)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.19648 -0.07281 -0.01323  0.06598  0.31804
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.09637    2.66196  -1.915  0.06291 .
## ln_cprice   -0.31219    0.14544  -2.147  0.03811 *
## ln_incom     0.73159    0.37400   1.956  0.05764 .
## q4           0.10963    0.04045   2.710  0.00994 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1116 on 39 degrees of freedom
## Multiple R-squared:  0.3089, Adjusted R-squared:  0.2558
## F-statistic: 5.812 on 3 and 39 DF,  p-value: 0.002204
```

As the cprice will be endogenous from the residual errors, the result can not be trusted. Two stage least squares regression needs to be carried out to make sure the coefficients are good.

Step 5:

Based on Qs = Qd, Ps = Pd at equilibrium, rewrite the demand and supply function, which will give us an equation for Coffee price derived from extrogenous variables. Run Linear Regression for the coffee price prediction (P_hat). q1, q2 and q3 were found not significant. So they were omitted in the Phat prediction.

```
Price_Predict_Model <- lm(ln_cprice ~ ln_bprice + ln_incom + ln_wprice +  q4)
summary(Price_Predict_Model)
```

```
##
## Call:
## lm(formula = ln_cprice ~ ln_bprice + ln_incom + ln_wprice + q4)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.092917 -0.017728 -0.008306  0.015677  0.099811
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.8279837  1.4864011  -0.557    0.581
## ln_bprice    0.5043524  0.0295977  17.040   <2e-16 ***
## ln_incom     0.1130961  0.1552992   0.728    0.471
## ln_wprice    0.5600487  0.5261970   1.064    0.294
## q4          -0.0003689  0.0143988  -0.026    0.980
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.03938 on 38 degrees of freedom
## Multiple R-squared:  0.9177, Adjusted R-squared:  0.909
## F-statistic: 105.9 on 4 and 38 DF,  p-value: < 2.2e-16
```

```
P_hat = fitted(Price_Predict_Model)
```

Step 6:

Use the predicted price P_hat to do linear regression again for demand function.

```
demand_with_Phat_model <- lm(ln_qu ~ P_hat +ln_incom + q4)
summary(demand_with_Phat_model)
```

```
##
## Call:
## lm(formula = ln_qu ~ P_hat + ln_incom + q4)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.20481 -0.05888 -0.00581  0.06939  0.32190
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.17264    2.67860  -1.931   0.0608 .
## P_hat       -0.32593    0.15350  -2.123   0.0401 *
## ln_incom     0.74646    0.37810   1.974   0.0555 .
## q4           0.10911    0.04054   2.692   0.0104 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1117 on 39 degrees of freedom
## Multiple R-squared:  0.3074, Adjusted R-squared:  0.2541
## F-statistic: 5.769 on 3 and 39 DF,  p-value: 0.002298
```

the coefficient of Phat is significant. Therefore the output is valid if the instrument variables pass Hausman test.

Step 7:

Hausman Test is run to check the correlation of instrument variable ln_bprice and ln_wprice with the residual errors in demand_with_Phat_model.

```
ResidualError = lm(demand_with_Phat_model$residuals~ln_bprice + ln_wprice)
print(summary(ResidualError)$r.squared)
```

```
## [1] 0.001819393
```

The result shows that R-squared is almost zero.Thus, instrument variables and residual errors are not correlated. So the demand function estimation based on Phat is valid. The direct linear regression in step 4 got coffee price coefficient -0.3121937 and the two stage least squares regression got coffee price coefficient -0.3259258. The results are quite close and the original linear regression result can be used for smaller standard error.

Supply Function

Step 8:

The supply function estimation is the same as demand function. 1st Run simple linear regression first for supply function.

```
Supply_Model_1 <- lm(ln_qu~ln_cprice + ln_bprice + ln_wprice )
summary(Supply_Model_1)

##
## Call:
## lm(formula = ln_qu ~ ln_cprice + ln_bprice + ln_wprice)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.23210 -0.07465 -0.02277  0.06856  0.42852
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.60866    4.89729  -0.737    0.466
## ln_cprice   -0.04384    0.53193  -0.082    0.935
## ln_bprice   -0.08191    0.28658  -0.286    0.777
## ln_wprice    1.01920    1.49901   0.680    0.501
##
## Residual standard error: 0.1301 on 39 degrees of freedom
## Multiple R-squared:  0.06063,	Adjusted R-squared:  -0.01162
## F-statistic: 0.8391 on 3 and 39 DF,  p-value: 0.4807
```

Step 9:

Based on Qs = Qd, Ps = Pd at equilibrium, rewrite the demand and supply function. Run Linear Regression for the coffee price prediction (Phat_2).

```
Price_Predict_Model2 <- lm(ln_cprice ~ ln_bprice + ln_incom + ln_wprice )
summary(Price_Predict_Model2)

##
## Call:
## lm(formula = ln_cprice ~ ln_bprice + ln_incom + ln_wprice)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.092859 -0.017642 -0.008331  0.015731  0.099543
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.82561    1.46438  -0.564    0.576
```

```
## ln_bprice    0.50449    0.02875  17.545   <2e-16 ***
## ln_incom     0.11195    0.14676   0.763    0.450
## ln_wprice    0.56182    0.51491   1.091    0.282
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03887 on 39 degrees of freedom
## Multiple R-squared:  0.9177, Adjusted R-squared:  0.9114
## F-statistic: 144.9 on 3 and 39 DF,  p-value: < 2.2e-16
```

```
Phat_2 = fitted(Price_Predict_Model2)
```

Step 10:

Use the predicted price Phat_2 to do linear regression again for supply function.

```
supply_with_Phat_model <- lm(ln_qu ~ Phat_2 +ln_wprice + ln_bprice)
summary(supply_with_Phat_model)
```

```
##
## Call:
## lm(formula = ln_qu ~ Phat_2 + ln_wprice + ln_bprice)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.22526 -0.05263 -0.00050  0.06922  0.38445
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.709      5.431   0.683   0.4987
## Phat_2        10.008      4.084   2.450   0.0189 *
## ln_wprice     -6.794      3.446  -1.971   0.0558 .
## ln_bprice     -5.263      2.107  -2.498   0.0168 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1211 on 39 degrees of freedom
## Multiple R-squared:  0.1858, Adjusted R-squared:  0.1232
## F-statistic: 2.967 on 3 and 39 DF,  p-value: 0.04365
```

the coefficient of P_hat2 is significant. Therefore the output is valid if the instrument variables pass Hausman test

Step 11:

Hausman Test. Check the correlation of instrument variable ln_incom with the residual errors in demand_with_Phat_model.

```
ResidualError = lm(supply_with_Phat_model$residuals~ln_incom)
print(summary(ResidualError)$r.squared)
```

```
## [1] 3.430953e-29
```

The result shows that instrument variables and residual errors are not correlated. So the supply function estimation based on Phat2 is valid. The direct linear regression in step 8 got coffee price coefficient -0.0438429 and the two stage least squares regression got coffee price coefficient 10.0083147 The results are very different. So the original simple linear regression output is not good due to endogeneity, and result from TSLS regression should be used.

Step 12:

Robustness

```
test_data$cprice <- log(test_data$cprice/test_data$oprice)
test_data$qu <- log(test_data$qu)
test_data$bprice <- log(test_data$bprice/test_data$oprice)
test_data$wprice <- log(test_data$wprice/test_data$oprice)
test_data$incom <- log(test_data$incom/test_data$oprice)

colnames(test_data)[which(names(test_data) == "qu")] <- "ln_qu"
colnames(test_data)[which(names(test_data) == "cprice")] <- "ln_cprice"
colnames(test_data)[which(names(test_data) == "bprice")] <- "ln_bprice"
colnames(test_data)[which(names(test_data) == "wprice")] <- "ln_wprice"
colnames(test_data)[which(names(test_data) == "incom")] <- "ln_incom"


predicted_test <- predict(demand_with_Phat_model,test_data)
rmse <- sqrt(mean((predicted_test-test_data$ln_qu)^2)/length(test_data))
rmse

## [1] 0.02399645

predicted_supply_test <- predict(supply_with_Phat_model, test_data)
rmse_supply <- sqrt(mean((predicted_supply_test-test_data$ln_qu)^2)/length(test_data))
rmse_supply

## [1] 0.2860978
```

Model is checked against with test data set to check for robustness and got the root-mean-squared value of 0.0239965 and 0.2860978 for demand and supply function respectively.