Homogenous demand, conduct exercise

Import data

First, import the data:

```
coffee_raw <- read.csv("dutch_coffee.csv")
```

Deflate prices using price index

Next, deflate the prices:

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Explore variations in price over time

Convert year/month variable into R time format:

```
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

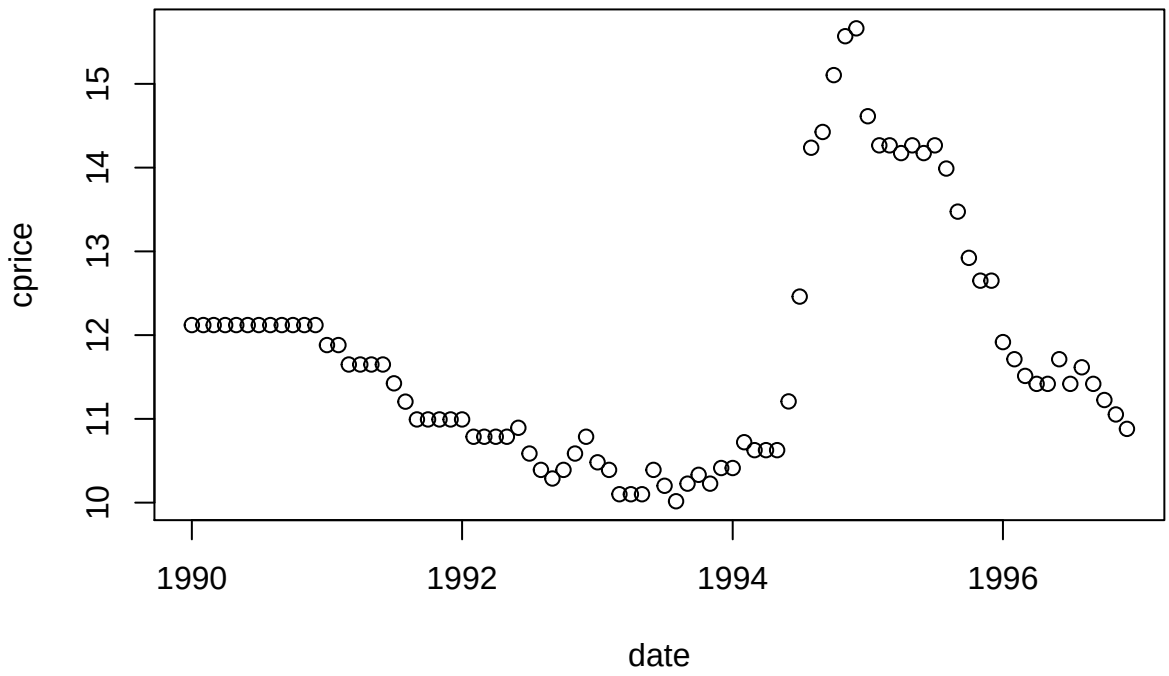
```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

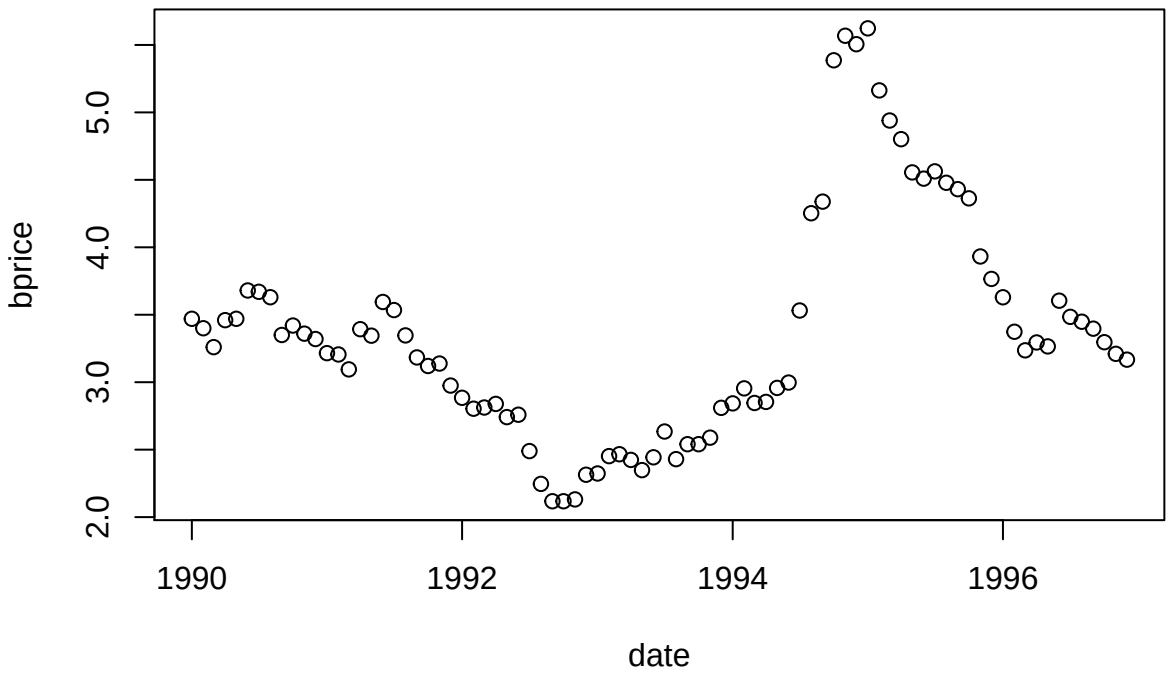
```
coffee <- coffee %>%
  mutate(date = as.Date(as.yearmon(month)))
```

Make some nice line graphs

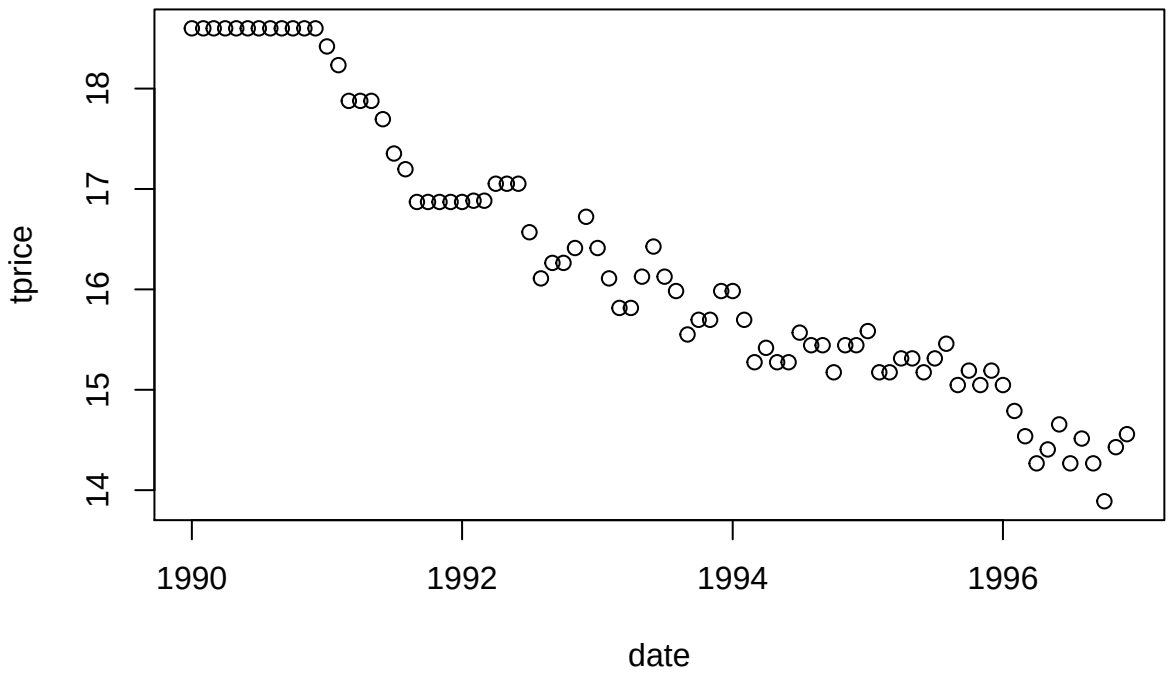
Create simple line graphs of key price and income variables over time.

Basic plot of coffee price over time.


cprice
10
11
12
13
14
15
1990
1992
1994
1996
date

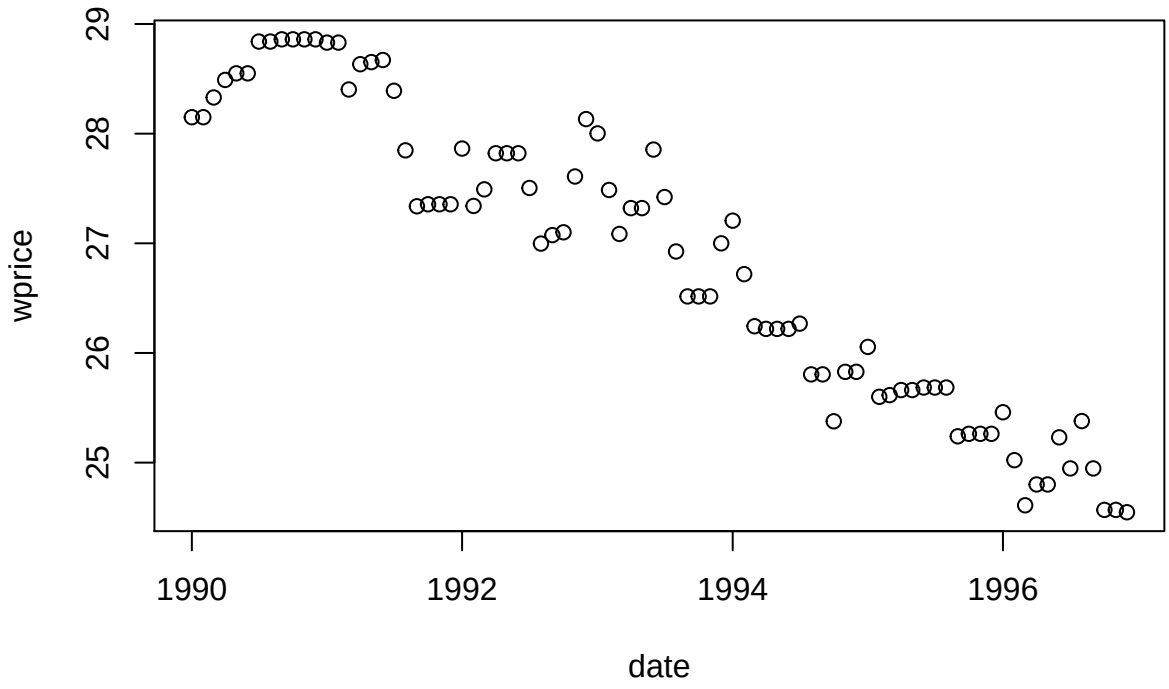


Basic plot of tea price against time:



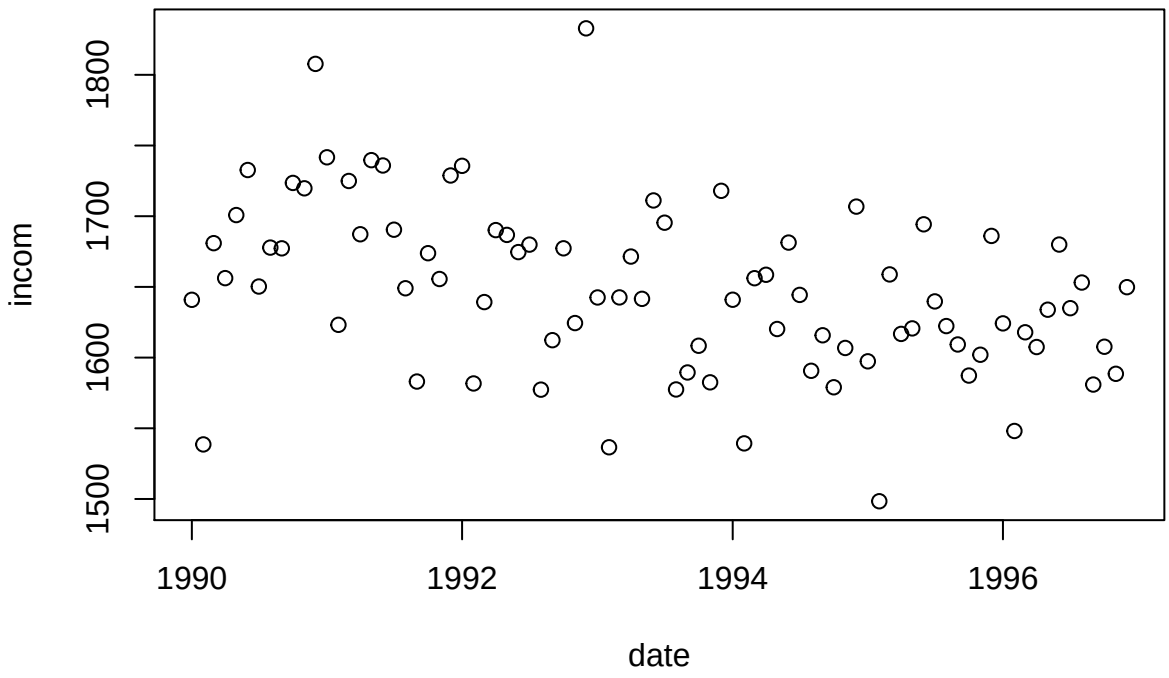
Plot of wage prices over time:

```
plot(wprice ~ date, coffee)
```

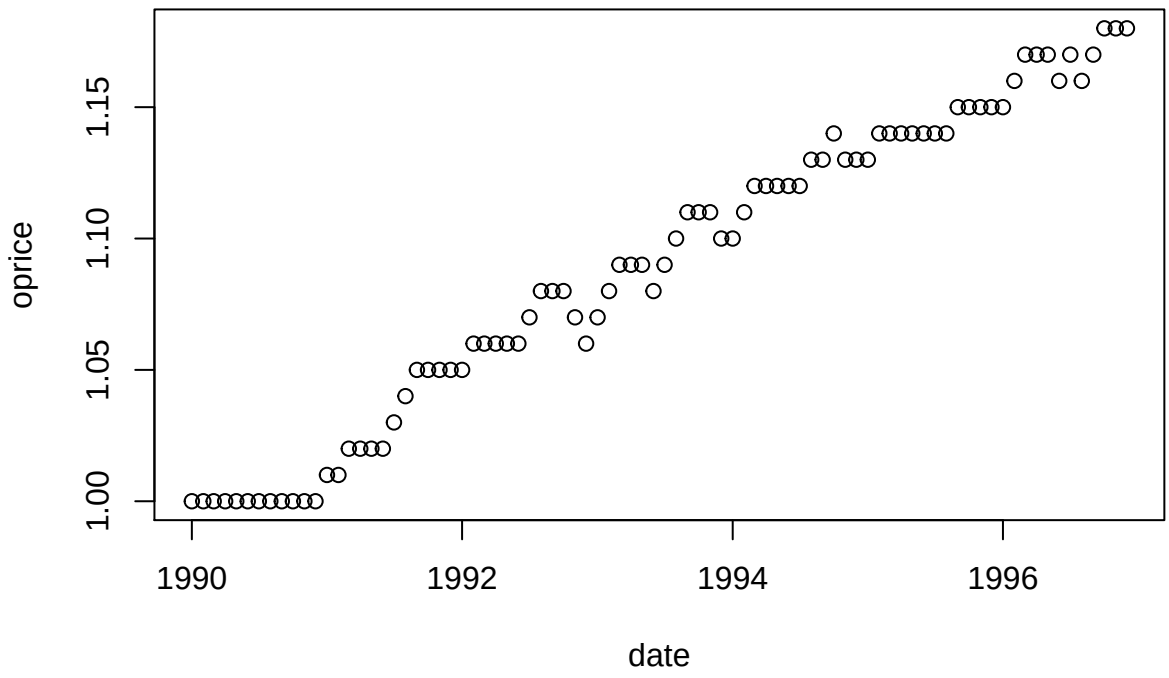


Basic plot of income against time:

```
plot(incom ~ date, coffee)
```

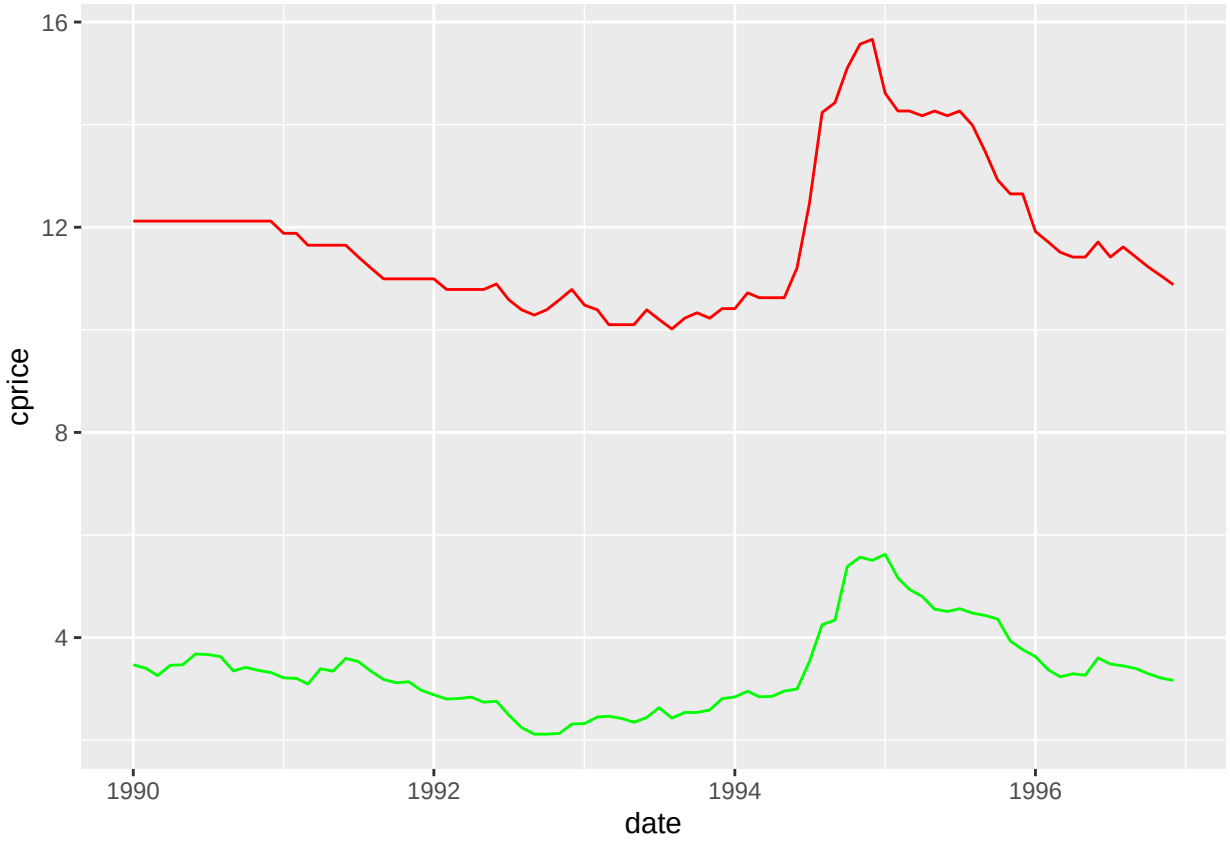



```
plot(oprice ~ date, coffee)
```



Now plot coffee retail price agains the bean price using ggplot2:

```
library(ggplot2)
ggplot(coffee, aes(date)) +
       geom_line(aes(y = cprice), colour="red") +
         geom_line(aes(y = bprice), colour="green")
```



Summary statistics

Produce a table of summary statistics:

```
coffee %>%
  select(qu, cprice, tprice, wprice, bprice, incom) %>%
  summary()
```

```
##        qu             cprice          tprice          wprice
##  Min.   :0.5200   Min.   :10.02   Min.   :13.89   Min.   :24.55
##  1st Qu.:0.6300   1st Qu.:10.70   1st Qu.:15.27   1st Qu.:25.68
##  Median :0.6600   Median :11.42   Median :15.98   Median :27.08
##  Mean   :0.6815   Mean   :11.76   Mean   :16.28   Mean   :26.87
##  3rd Qu.:0.7400   3rd Qu.:12.12   3rd Qu.:17.05   3rd Qu.:27.90
##  Max.   :1.0400   Max.   :15.66   Max.   :18.60   Max.   :28.86
##      bprice          incom
##  Min.   :2.118   Min.   :1498
##  1st Qu.:2.812   1st Qu.:1608
##  Median :3.280   Median :1643
##  Mean   :3.363   Mean   :1648
##  3rd Qu.:3.611   3rd Qu.:1686
##  Max.   :5.623   Max.   :1833
```

Now consider summary statistics for quarter 1 alone:

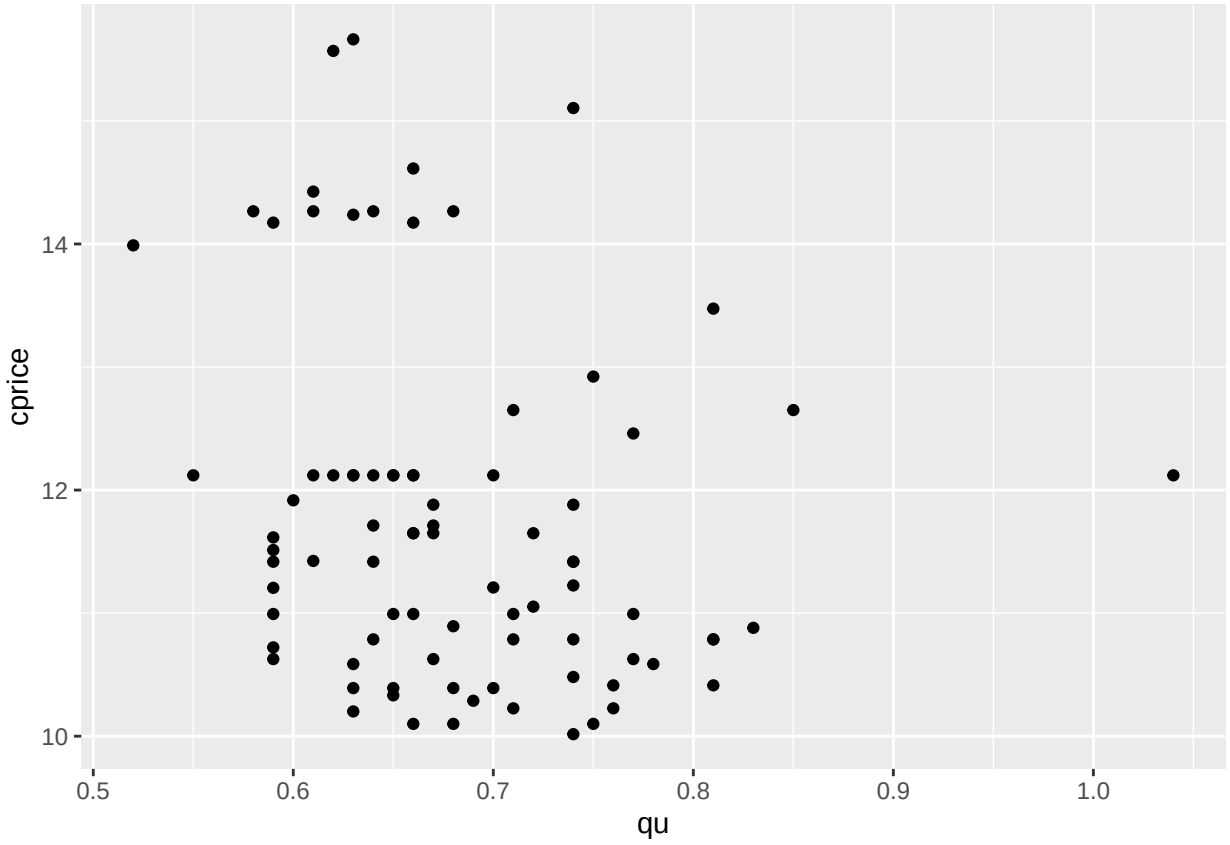
```
coffee %>%
  filter(q1 == 1) %>%
  select(qu, cprice, tprice, wprice, bprice, incom) %>%
  summary()
```

```
##        qu             cprice          tprice          wprice
##  Min.   :0.5500   Min.   :10.10   Min.   :14.54   Min.   :24.61
##  1st Qu.:0.5900   1st Qu.:10.72   1st Qu.:15.27   1st Qu.:26.06
##  Median :0.6500   Median :11.65   Median :16.11   Median :27.34
##  Mean   :0.6529   Mean   :11.68   Mean   :16.50   Mean   :27.07
##  3rd Qu.:0.7100   3rd Qu.:12.12   3rd Qu.:17.88   3rd Qu.:28.15
##  Max.   :0.7600   Max.   :14.61   Max.   :18.60   Max.   :28.83
##      bprice          incom
##  Min.   :2.323   Min.   :1498
##  1st Qu.:2.843   1st Qu.:1582
##  Median :3.206   Median :1639
##  Mean   :3.333   Mean   :1624
##  3rd Qu.:3.400   3rd Qu.:1656
##  Max.   :5.623   Max.   :1742
```

Now do the same for q2, q3, and q4.

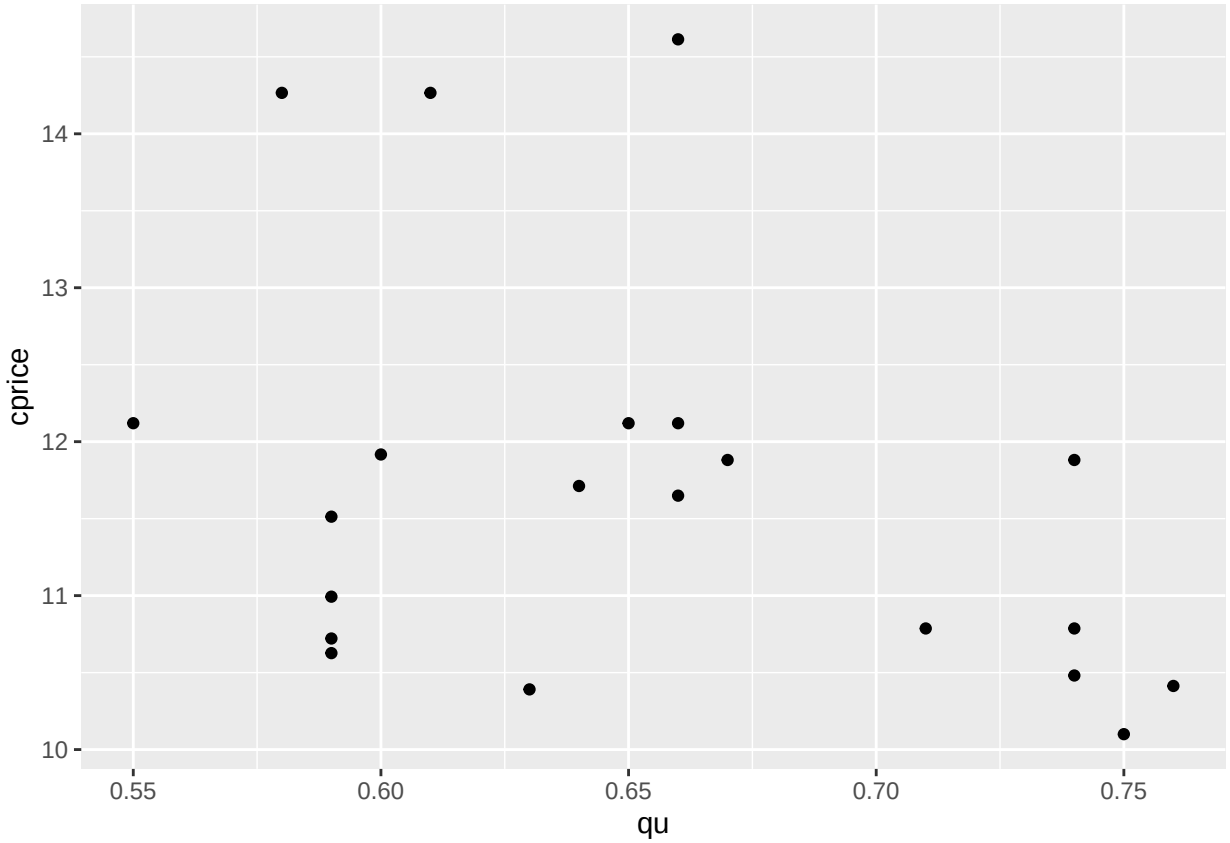
Considering prices and quantities

First, use ggplot to produce a basic graph of prices and quantities:



Now do the same thing but only for the first quarter:

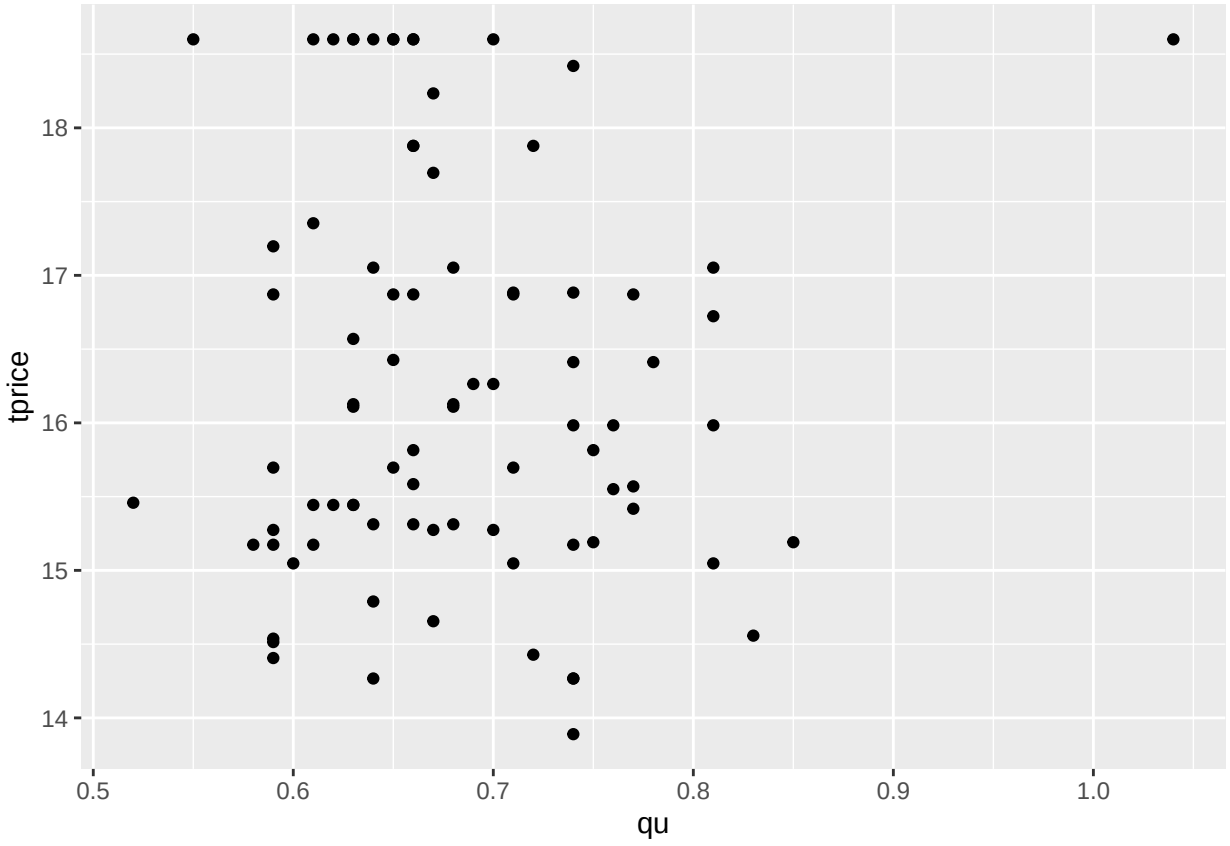
```
coffee %>%
  filter(q1 == 1) %>%
  ggplot(aes(x = qu)) +
    geom_point(aes(y = cprice))
```



Now consider for q2, q3 and q4.

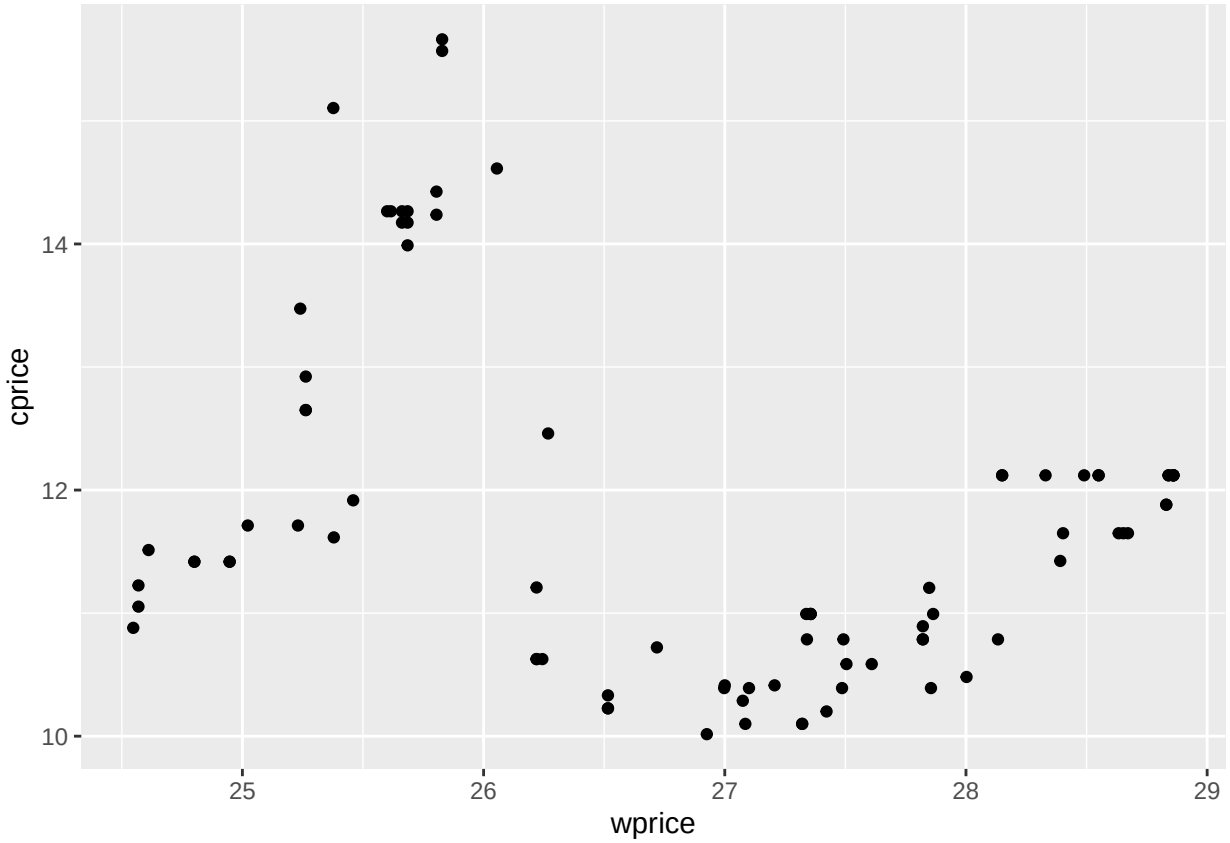
Assess relationship between tea price and quantity demanded of coffee

```
coffee %>%
  ggplot(aes(x = qu)) +
    geom_point(aes(y = tprice))
```



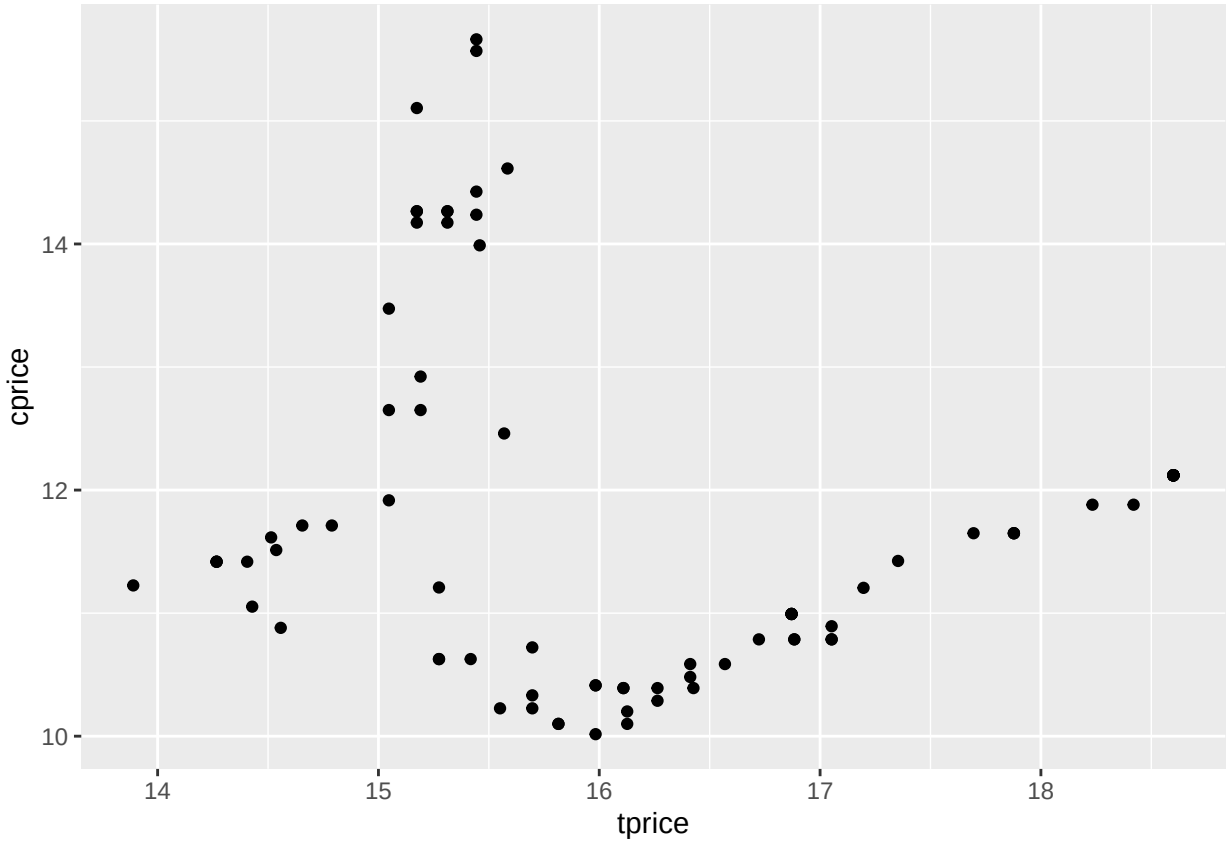
Price of coffee and labor

```
coffee %>%
  ggplot(aes(x = wprice)) +
    geom_point(aes(y = cprice))
```



Price of coffee and price of tea

```
coffee %>%
  ggplot(aes(x = tprice)) +
    geom_point(aes(y = cprice))
```



Overall correlations

```
coffee %>%
  select(qu, cprice, tprice, wprice, bprice, incom) %>%
  cor()
```

```
##                qu     cprice      tprice      wprice     bprice      incom
## qu      1.00000000 -0.2243179 -0.01592424  0.01687637 -0.2367654  0.2352513
## cprice -0.22431789  1.0000000 -0.11878101 -0.30051704  0.9595010 -0.1068931
## tprice -0.01592424 -0.1187810  1.00000000  0.94795657 -0.1978217  0.4703265
## wprice  0.01687637 -0.3005170  0.94795657  1.00000000 -0.3774048  0.5145200
## bprice -0.23676540  0.9595010 -0.19782168 -0.37740483  1.0000000 -0.1657889
## incom   0.23525135 -0.1068931  0.47032651  0.51452003 -0.1657889  1.0000000
```

create log variables

Basic demand regressions

```
q_p1 <- lm(ln_qu ~ ln_cprice + q1 + q2 + q3, data = coffee)
summary(q_p1)
```

```
##
## Call:
## lm(formula = ln_qu ~ ln_cprice + q1 + q2 + q3, data = coffee)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -0.18478 -0.06511 -0.00992  0.06617  0.35404
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.31947    0.23357   1.368 0.175270
## ln_cprice   -0.25424    0.09427  -2.697 0.008552 **
## q1          -0.12698    0.03032  -4.188 7.26e-05 ***
## q2          -0.09192    0.03035  -3.029 0.003315 **
## q3          -0.11786    0.03029  -3.891 0.000207 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09815 on 79 degrees of freedom
## Multiple R-squared:  0.2648, Adjusted R-squared:  0.2275
## F-statistic: 7.112 on 4 and 79 DF,  p-value: 6.07e-05
```

```
q_p2 <- lm(ln_qu ~ ln_cprice + tprice + q1 + q2 + q3, data = coffee)
summary(q_p2)
```

```
##
## Call:
## lm(formula = ln_qu ~ ln_cprice + tprice + q1 + q2 + q3, data = coffee)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.18526 -0.06476 -0.00999  0.06625  0.35644
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.3370139  0.2793459   1.206 0.231295
## ln_cprice   -0.2552341  0.0952513  -2.680 0.008989 **
## tprice      -0.0009386  0.0080759  -0.116 0.907774
## q1          -0.1265904  0.0306931  -4.124 9.21e-05 ***
## q2          -0.0916536  0.0306206  -2.993 0.003696 **
## q3          -0.1177935  0.0304855  -3.864 0.000229 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09876 on 78 degrees of freedom
## Multiple R-squared:  0.2649, Adjusted R-squared:  0.2178
## F-statistic: 5.621 on 5 and 78 DF,  p-value: 0.000178
```

```
q_p3 <- lm(ln_qu ~ ln_cprice + tprice + incom + q1 + q2 + q3, data = coffee)
summary(q_p3)
```

```
##
## Call:
## lm(formula = ln_qu ~ ln_cprice + tprice + incom + q1 + q2 + q3,
##     data = coffee)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.18792 -0.06895 -0.00947  0.06014  0.33323
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0218835  0.4201731  -0.052 0.958598
## ln_cprice   -0.2490158  0.0952224  -2.615 0.010728 *
## tprice      -0.0065034  0.0094193  -0.690 0.492000
## incom        0.0002601  0.0002278   1.142 0.257106
## q1          -0.1134993  0.0327091  -3.470 0.000856 ***
## q2          -0.0920199  0.0305629  -3.011 0.003522 **
## q3          -0.1085006  0.0314962  -3.445 0.000928 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09857 on 77 degrees of freedom
## Multiple R-squared:  0.2771, Adjusted R-squared:  0.2208
## F-statistic:  4.92 on 6 and 77 DF,  p-value: 0.0002628
```

Instrumental variable regressions

```
library("AER")
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## Loading required package: lmtest
```

```
## Loading required package: sandwich
```

```
## Loading required package: survival
```

```
iv1 <- ivreg(ln_qu ~ q1 + q2 + q3 + ln_cprice  | ln_bprice + q1 + q2 + q3, data = coffee)
summary(iv1)
```

```
##
## Call:
## ivreg(formula = ln_qu ~ q1 + q2 + q3 + ln_cprice | ln_bprice +
##     q1 + q2 + q3, data = coffee)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.181011 -0.067238 -0.008444  0.067356  0.354644
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.37349    0.24693   1.513 0.134386
## q1          -0.12730    0.03033  -4.197 7.04e-05 ***
## q2          -0.09235    0.03036  -3.042 0.003194 **
## q3          -0.11789    0.03030  -3.891 0.000207 ***
## ln_cprice   -0.27613    0.09970  -2.770 0.006994 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.09818 on 79 degrees of freedom
## Multiple R-Squared: 0.2643,  Adjusted R-squared: 0.227
## Wald test: 7.207 on 4 and 79 DF,  p-value: 5.322e-05
```

```
iv2 <- ivreg(ln_qu ~ q1 + q2 + q3 + ln_tprice + ln_incom + ln_cprice  | ln_bprice + q1 + q2 + q3 + ln_t
summary(iv2)
```

```
##
## Call:
## ivreg(formula = ln_qu ~ q1 + q2 + q3 + ln_tprice + ln_incom +
##     ln_cprice | ln_bprice + q1 + q2 + q3 + ln_tprice + ln_incom,
##     data = coffee)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.184037 -0.070170 -0.008924  0.061933  0.336405
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.35533    2.62891  -0.896 0.373081
## q1          -0.11446    0.03276  -3.494 0.000791 ***
## q2          -0.09263    0.03060  -3.027 0.003358 **
## q3          -0.10925    0.03148  -3.470 0.000855 ***
## ln_tprice   -0.10444    0.15533  -0.672 0.503360
## ln_incom     0.40537    0.37685   1.076 0.285436
## ln_cprice   -0.27125    0.10068  -2.694 0.008656 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09868 on 77 degrees of freedom
## Multiple R-Squared: 0.2755,  Adjusted R-squared: 0.2191
## Wald test: 4.956 on 6 and 77 DF,  p-value: 0.0002455
```

Now lets show the adjusted learner index

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.07434 0.08372 0.08748 0.08855 0.09292 0.10597
```

```
##      Ln_adj
##  Min.   :0.07434
##  1st Qu.:0.08445
##  Median :0.08834
##  Mean   :0.08837
##  3rd Qu.:0.09311
##  Max.   :0.09822
```

Now, consider for q2, q3, and q4.

Estimating conduct parameter

```
conduct1 <- lm(cprice ~ cost + q1 + q2 + q3 -1, data = coffee) # omits intercept
summary(conduct1)
```

```
##
## Call:
```

```
## lm(formula = cprice ~ cost + q1 + q2 + q3 - 1, data = coffee)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.06569 -0.28829 -0.03361  0.28889  1.00238
##
## Coefficients:
##       Estimate Std. Error t value Pr(>|t|)
## cost  1.466267   0.010910 134.398   <2e-16 ***
## q1    0.002807   0.125644   0.022    0.982
## q2   -0.007612   0.125362  -0.061    0.952
## q3    0.084503   0.126116   0.670    0.505
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4158 on 80 degrees of freedom
## Multiple R-squared:  0.9988, Adjusted R-squared:  0.9988
## F-statistic: 1.703e+04 on 4 and 80 DF,  p-value: < 2.2e-16
```

```
conduct2 <- lm(cprice ~ cost -1, data = coffee) # omits intercept
summary(conduct2)
```

```
##
## Call:
## lm(formula = cprice ~ cost - 1, data = coffee)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.08922 -0.28022 -0.05497  0.29080  0.98628
##
## Coefficients:
##      Estimate Std. Error t value Pr(>|t|)
## cost 1.468731   0.005548   264.7   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4099 on 83 degrees of freedom
## Multiple R-squared:  0.9988, Adjusted R-squared:  0.9988
## F-statistic: 7.007e+04 on 1 and 83 DF,  p-value: < 2.2e-16
```

From the FOC we have: $p = \frac{\gamma}{(\gamma+\theta)}c$ This is the estimated coefficient for cost.

```
b=1.47
theta=gamma*(1-b)/b
summary(theta)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.08856 0.08856 0.08856 0.08856 0.08856 0.08856
```

```
n=1/theta
summary(n)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   11.29   11.29   11.29   11.29   11.29   11.29
```

Linear demand

$Q = \beta(\alpha - P)$

```
linear1 <- ivreg(qu ~ cprice + incom + q1 + q2 + q3 | bprice + incom + q1 + q2 + q3, data = coffee)
summary(linear1)
```

```
##
## Call:
## ivreg(formula = qu ~ cprice + incom + q1 + q2 + q3 | bprice +
##     incom + q1 + q2 + q3, data = coffee)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.116823 -0.049428 -0.009419  0.041008  0.279560
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.6353799  0.2493831   2.548 0.012806 *
## cprice      -0.0138031  0.0057188  -2.414 0.018139 *
## incom        0.0001617  0.0001392   1.162 0.248701
## q1          -0.0839335  0.0224717  -3.735 0.000355 ***
## q2          -0.0699218  0.0217508  -3.215 0.001901 **
## q3          -0.0783453  0.0221914  -3.530 0.000700 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07023 on 78 degrees of freedom
## Multiple R-Squared: 0.2691,  Adjusted R-squared: 0.2222
## Wald test: 5.781 on 5 and 78 DF,  p-value: 0.0001366
```

```
linear2 <- ivreg(qu ~ cprice + q1 + q2 + q3 | bprice + q1 + q2 + q3, data = coffee)
summary(linear2)
```

```
##
## Call:
## ivreg(formula = qu ~ cprice + q1 + q2 + q3 | bprice + q1 + q2 +
##     q3, data = coffee)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.113895 -0.046497 -0.009371  0.044152  0.302896
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.918705   0.069596  13.201  < 2e-16 ***
## cprice      -0.014984   0.005707  -2.626  0.01038 *
## q1          -0.090782   0.021763  -4.171 7.71e-05 ***
## q2          -0.068903   0.021786  -3.163  0.00222 **
## q3          -0.083902   0.021730  -3.861  0.00023 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07041 on 79 degrees of freedom
## Multiple R-Squared: 0.2558,  Adjusted R-squared: 0.2181
## Wald test: 6.939 on 4 and 79 DF,  p-value: 7.7e-05
```

Compute adjusted learner index:

```
coffee$gamma_l=-0.015*coffee$cprice/coffee$qu
coffee$Ln_adj_l=-coffee$gamma_l*coffee$Ln
```

Now summarise the lerner indices:

```
coffee %>%
  select(Ln_adj_l) %>%
  summary()
```

```
##     Ln_adj_l
##  Min.   :0.05623
##  1st Qu.:0.07298
##  Median :0.08197
##  Mean   :0.08407
##  3rd Qu.:0.09099
##  Max.   :0.13442
```

Now summarise by quarters.

Demand rotaters

Create seasonal demand rotators:

```
coffee <- coffee %>%
  mutate(cprice_q1=cprice*q1,
         bprice_q1=bprice*q1,
         cprice_q2=cprice*q2,
         bprice_q2=bprice*q2,
         cprice_q3=cprice*q3,
         bprice_q3=bprice*q3,
         cprice_q4=cprice*q4,
         bprice_q4=bprice*q4
  )
```

Now, use rotators in regression:

```
rotator <- ivreg(qu ~  q1 + q2 + q3 + (cprice + cprice_q1) |
                   (bprice + bprice_q1) + q1 + q2 + q3 , data = coffee)
summary(rotator)
```

```
##
## Call:
## ivreg(formula = qu ~ q1 + q2 + q3 + (cprice + cprice_q1) | (bprice +
##     bprice_q1) + q1 + q2 + q3, data = coffee)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.111909 -0.047215 -0.009324  0.041327  0.302610
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.903591   0.078116  11.567  < 2e-16 ***
## q1          -0.028064   0.166265  -0.169  0.86640
## q2          -0.068552   0.021913  -3.128  0.00247 **
## q3          -0.083854   0.021842  -3.839  0.00025 ***
## cprice      -0.013713   0.006438  -2.130  0.03631 *
```

```
## cprice_q1   -0.005345   0.014053  -0.380  0.70473
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07077 on 78 degrees of freedom
## Multiple R-Squared: 0.2577,  Adjusted R-squared: 0.2101
## Wald test: 5.503 on 5 and 78 DF,  p-value: 0.0002165
```

```
coffee$alpha=0.90/0.015
alpha_q1 = 0.90/0.019
coffee <- coffee %>%
  mutate(alpha = ifelse(q1 == 1, alpha_q1, alpha))
```

pricing equation

```
lm(cprice ~ alpha + bprice, data = coffee )
```

```
##
## Call:
## lm(formula = cprice ~ alpha + bprice, data = coffee)
##
## Coefficients:
## (Intercept)        alpha       bprice
##    6.077713     0.003337     1.634484
```