Ace Hardware

Case #1: Ace Hardware

Maryclare Leonard

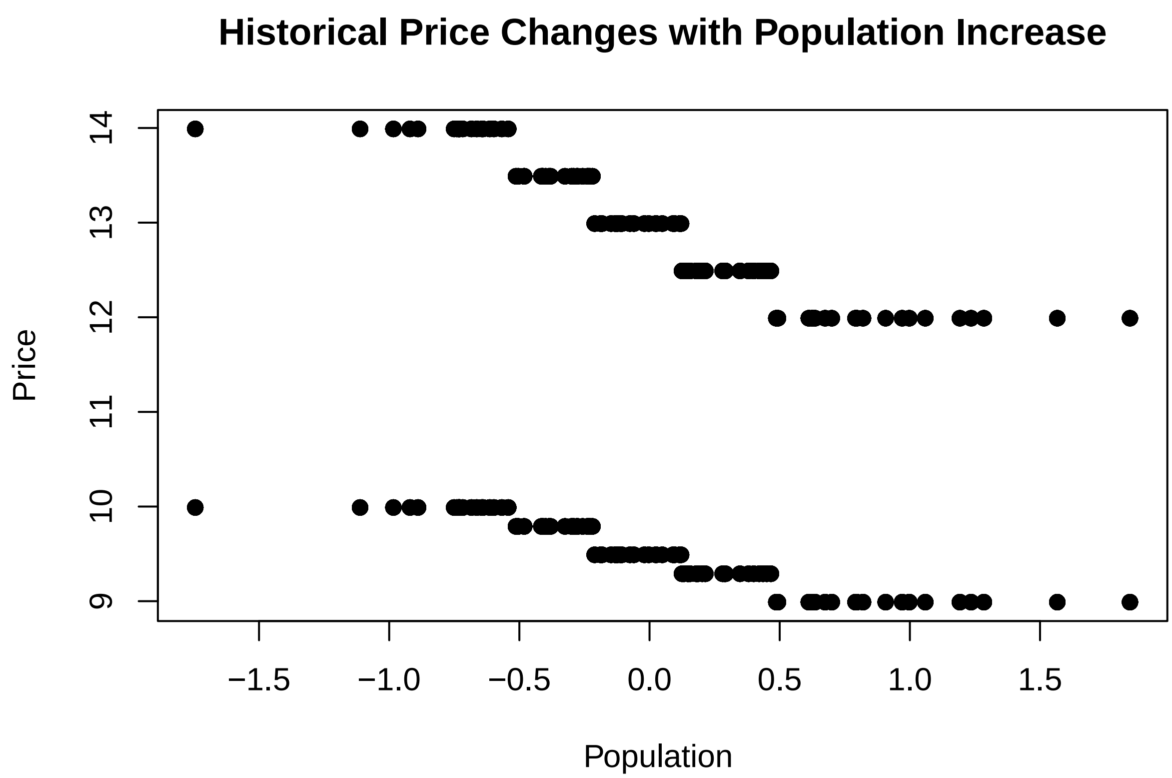
March 22 2022

1 Technical Section

```
# Read in the data
# NOTE: This assumes the current working directory contains these files
df.hist <- read_excel("ace_historical.xlsx")
df.test <- read_excel("ace_testlearn.xlsx")
```

1.1 Data Exploration

```
#plot population to historical price
plot(df.hist$mpop, df.hist$regprice, main="Historical Price Changes with Population Incr
   xlab="Population ", ylab="Price", pch=19)
```



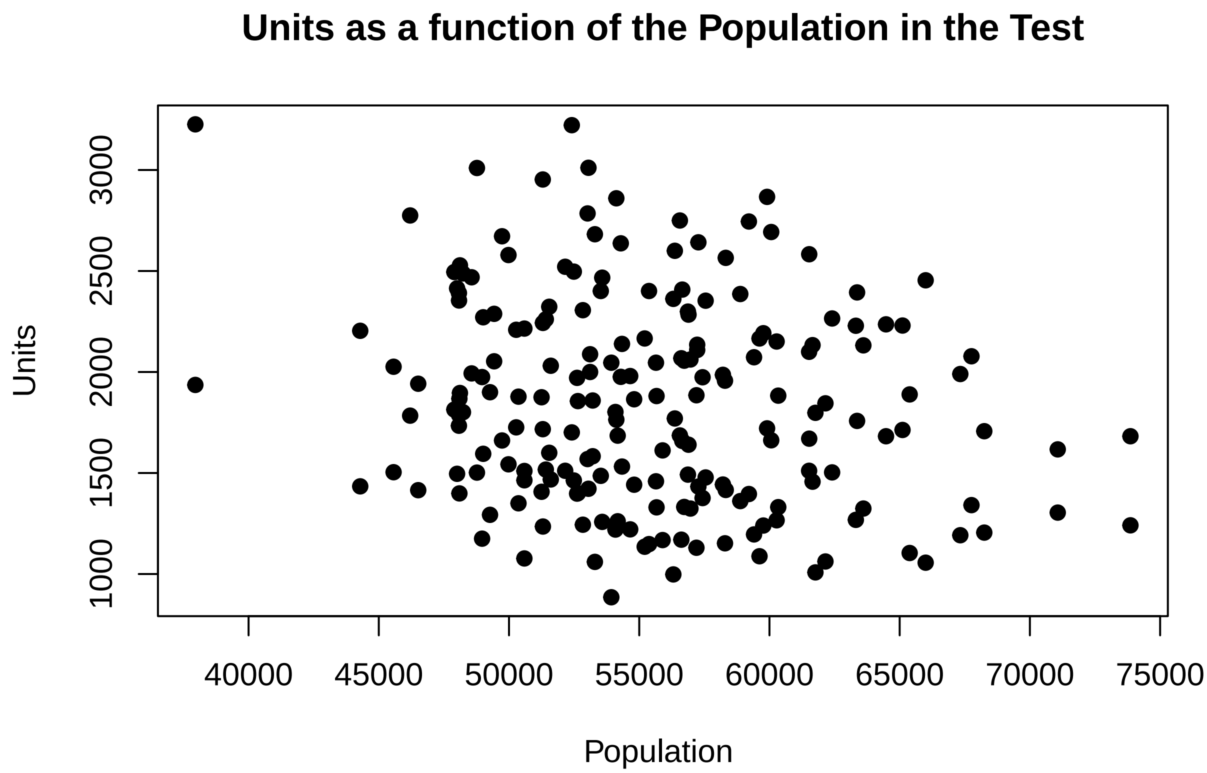

```r
#plot population to test price
plot(df.test$mpop, df.test$regprice, main="Test Price Changes with Population Increase",
   xlab="Population ", ylab="Price", pch=19)
```

1.1.1 Data Exploration: Part A

Based on the the first of the two previous graphs, historically, price decreased as the population in the area increased.

However, in the tests, which is displayed in the second of the two graphs, price is not related to the population. The price is set independent of the population of the area. Areas with the same population have prices all over the board, from $9 - $10 for the bottom product and from $12 to $14 for the top product.

```
# Part 1 Data Exploration
#plot population to units
plot(df.test$population, df.test$units, main="Units as a function of the Population in t
   xlab="Population ", ylab="Units", pch=19)
```



1.1.2 Data Exploration: Part B

Based on the above graph, the units sold reaches its peak at the lowest population level. The maximum at the highest population is almost half of this peak. The maximums at each population size trends down as population increases. Therefore, Ace does not sell more in areas with larger population.

1.2 Analytics Elasticity

```
df.hist <- df.hist %>%
  mutate(lnpHist = log(regprice),
         lnqHist = log(units))
head(df.hist)
```

```
# A tibble: 6 x 11
   week store product regprice units population  mpop distance  mdist lnpHist
  <dbl> <dbl>   <dbl>    <dbl> <dbl>      <dbl> <dbl>    <dbl>  <dbl>   <dbl>
1     1     1       1     12.5   624      59408 0.400     1.26 -0.481    2.52
```

```
2     2     1        1      12.5    557        59408 0.400      1.26 -0.481     2.52
3     3     1        1      12.5    486        59408 0.400      1.26 -0.481     2.52
4     4     1        1      12.5    670        59408 0.400      1.26 -0.481     2.52
5     5     1        1      12.5    449        59408 0.400      1.26 -0.481     2.52
6     6     1        1      12.5    533        59408 0.400      1.26 -0.481     2.52
# ... with 1 more variable: lnqHist <dbl>
```

```
reg1 <- lm(lnqHist ~ lnpHist, data=df.hist)
summary(reg1)
```

```
Call:
lm(formula = lnqHist ~ lnpHist, data = df.hist)

Residuals:
     Min       1Q   Median       3Q      Max
-0.69616 -0.19156 -0.00775  0.17694  0.73301

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.23659    0.07474   29.92   <2e-16 ***
lnpHist      1.58937    0.03098   51.30   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2465 on 2398 degrees of freedom
Multiple R-squared:  0.5233,    Adjusted R-squared:  0.5231
F-statistic:  2632 on 1 and 2398 DF,  p-value: < 2.2e-16
```

The regression indicates that the price elasticity is for the historical data is 1.589.

```
df.test <- df.test %>%
  mutate(lnpTest = log(regprice),
         lnqTest = log(units))
head(df.test)
```

```
# A tibble: 6 x 11
  store product regprice units population   mpop distance   mdist  cost lnpTest
  <dbl>   <dbl>    <dbl> <dbl>      <dbl>  <dbl>    <dbl>   <dbl> <dbl>   <dbl>
1     1       1     12.0  2073      59408  0.400    1.26  -0.481    8.8    2.48
2     2       1     14.0  1510      50591 -0.481    1.65  -0.0915   8.8    2.64
3     3       1     13.0  2243      51302 -0.410    1.98   0.239    8.8    2.56
4     4       1     13.5  1617      71070  1.57     0.290 -1.45     8.8    2.60
5     5       1     13.5  1845      62150  0.674    1.21  -0.531    8.8    2.60
6     6       1     13.5  1859      53213 -0.219    1.95   0.209    8.8    2.60
# ... with 1 more variable: lnqTest <dbl>
```

```r
reg2 <- lm(lnqTest ~ lnpTest, data=df.test)
summary(reg2)
```

```
Call:
lm(formula = lnqTest ~ lnpTest, data = df.test)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.58092 -0.15553 -0.00424  0.16900  0.50614 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  4.76403    0.22002   21.65   <2e-16 ***
lnpTest      1.13074    0.09119   12.40   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2094 on 198 degrees of freedom
Multiple R-squared:  0.4371,    Adjusted R-squared:  0.4342 
F-statistic: 153.7 on 1 and 198 DF,  p-value: < 2.2e-16
```

For the experiment, the price elasticity is 1.131,

Therefore, the price elasticity decreases in the experiment. For a increase in price by $1, the percentage quantity purchased will decrease by 1.589 - 1.131 = 0.458.

2 Managerial Discussion

2.1 Pricing Recommendation

Since the Log-Log Model demonstrated a price elasticity for the experiment of NA, the gross margin percentage can be calculated as follows:

-1 / -1.131 = 0.884

Margin: 88%

Therefore, based on this case, each product should earn a margin of 88%

The first is at cost of $8.80. The price should therefore be $16.54.

The second is at cost of $5.43. The price should therefore be $4.78.

2.2 Zone Pricing

```
#mean population calculation
mean_pop <- mean(df.test$population)
print(mean_pop)
```

```
[1] 55406.09
```

```
#create two data tables for below and above mean
df.test.low <- df.test %>% filter(population < mean_pop)
df.test.high <- df.test %>% filter(population > mean_pop)
```

2.2.1 Part A

```
#above mean
df.test.high <- df.test.high %>%
  mutate(lnpTestAbove = log(regprice),
         lnqTestAbove = log(units))
head(df.test.high)
```

```
# A tibble: 6 x 13
  store product regprice units population   mpop distance  mdist  cost lnpTest
  <dbl>   <dbl>    <dbl> <dbl>      <dbl>  <dbl>    <dbl>  <dbl> <dbl>   <dbl>
1     1       1     12.0  2073      59408 0.400     1.26  -0.481   8.8    2.48
2     4       1     13.5  1617      71070 1.57      0.290 -1.45    8.8    2.60
3     5       1     13.5  1845      62150 0.674     1.21  -0.531   8.8    2.60
4     7       1     13.5  2046      55643 0.0237    2.16   0.419   8.8    2.60
5     8       1     13.0  2284      56892 0.149     1.59  -0.151   8.8    2.56
6    10       1     12.5  2299      56870 0.146     1.52  -0.222   8.8    2.52
# ... with 3 more variables: lnqTest <dbl>, lnpTestAbove <dbl>,
#   lnqTestAbove <dbl>
```

```
reg4 <- lm(lnqTestAbove ~ lnpTestAbove, data=df.test.high)
summary(reg4)
```

```
Call:
lm(formula = lnqTestAbove ~ lnpTestAbove, data = df.test.high)

Residuals:
     Min       1Q   Median       3Q      Max
```

```
-0.41901 -0.14953 -0.02973  0.16975  0.41371

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    4.6475     0.3059  15.192  < 2e-16 ***
lnpTestAbove   1.1632     0.1267   9.179 1.22e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2013 on 92 degrees of freedom
Multiple R-squared:  0.478,	Adjusted R-squared:  0.4724
F-statistic: 84.25 on 1 and 92 DF,  p-value: 1.219e-14
```

For the stores with below average population, the price elasticity is 1.163. The product is very inelastic and consumers are price insensitive.

2.2.2 Part B

```r
#below mean
df.test.low <- df.test.low %>%
  mutate(lnpTestBelow = log(regprice),
         lnqTestBelow = log(units))
head(df.test.low)
```

```
# A tibble: 6 x 13
  store product regprice units population   mpop distance   mdist  cost lnpTest
  <dbl>   <dbl>    <dbl> <dbl>      <dbl>  <dbl>    <dbl>   <dbl> <dbl>   <dbl>
1     2       1     14.0  1510      50591 -0.481     1.65 -0.0915   8.8    2.64
2     3       1     13.0  2243      51302 -0.410     1.98  0.239    8.8    2.56
3     6       1     13.5  1859      53213 -0.219     1.95  0.209    8.8    2.60
4     9       1     12.5  2579      49980 -0.543     1.52 -0.222    8.8    2.52
5    11       1     12.0  2682      53298 -0.211     2.01  0.269    8.8    2.48
6    12       1     13.0  2323      51545 -0.386     2.64  0.899    8.8    2.56
# ... with 3 more variables: lnqTest <dbl>, lnpTestBelow <dbl>,
#   lnqTestBelow <dbl>
```

```r
reg3 <- lm(lnqTestBelow ~ lnpTestBelow, data=df.test.low)
summary(reg3)
```

```
Call:
lm(formula = lnqTestBelow ~ lnpTestBelow, data = df.test.low)
```

```
Residuals:
     Min       1Q   Median       3Q      Max 
-0.61762 -0.14734  0.01513  0.16641  0.47428 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)    4.8617     0.3088  15.742  < 2e-16 ***
lnpTestBelow   1.1043     0.1281   8.622  7.9e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2125 on 104 degrees of freedom
Multiple R-squared:  0.4168,    Adjusted R-squared:  0.4112 
F-statistic: 74.34 on 1 and 104 DF,  p-value: 7.897e-14
```

For the stores with above average population, the price elasticity is 1.104. The product is very inelastic and consumers are price insensitive.

Knowing that: For the stores with below average population, the price elasticity is 1.163. For the stores with above average population, the price elasticity is 1.104. Both products are very inelastic and consumers are price insensitive.

2.2.3 Part C

Considering what we just learned in Part B, this zone pricing policy is not dependent upon population to determine which zone or pricing strategy a store should fall in.

Note the elasticities only vary by 0.059. With this experimental model, the price elasticity remains relatively consistent across zones and is therefore better to employ versus the historical model that had even higher price elasticities.

2.3 Your Recommended Next Steps

Since there are 10,000 items across the every store I would recommend testing one item in each category and employing a markup procedure that works for this item across the rest of the items in the category. For example all single sell nails will be marked up the same, based on the results from one nail. However, a package of nails would have a different markup since you are buying in bulk, yet bulk sets of nails of the similar quantity can follow the markup procedure of this set.

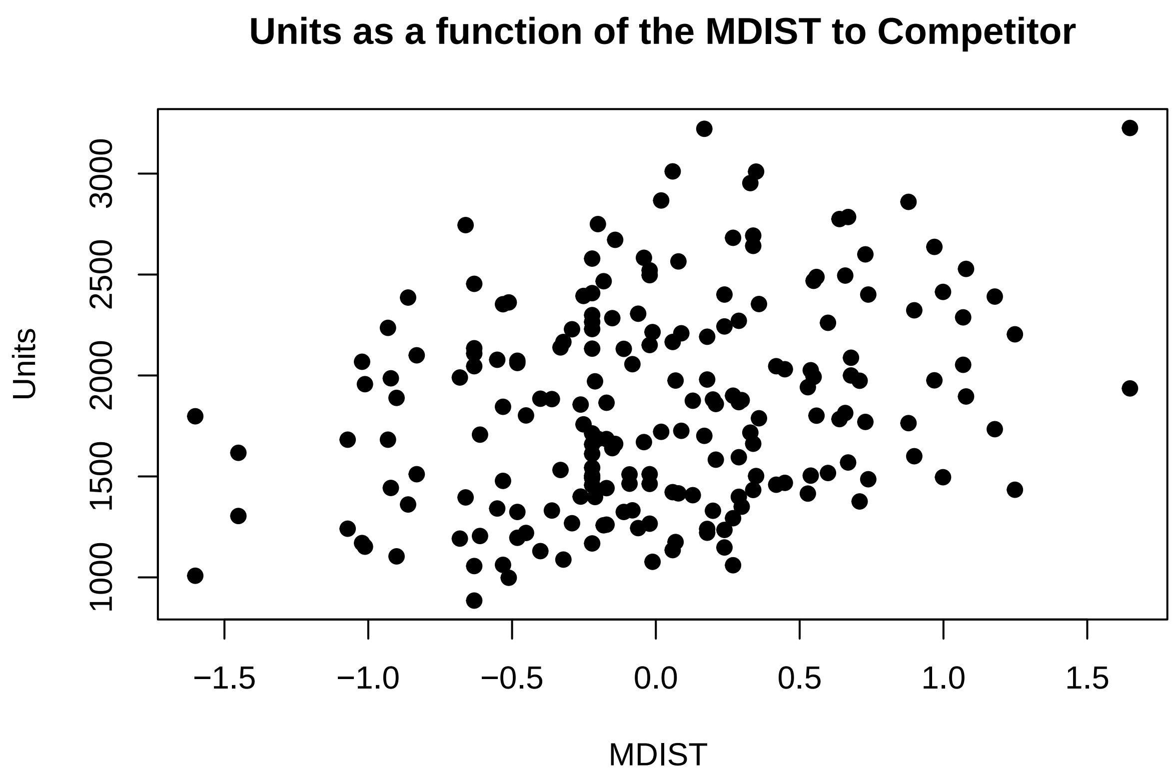
In scaling, start with the best sellers first to ensure that you are maximizing profit out of these items that are frequently being purchased.

2.4 Further Discussions

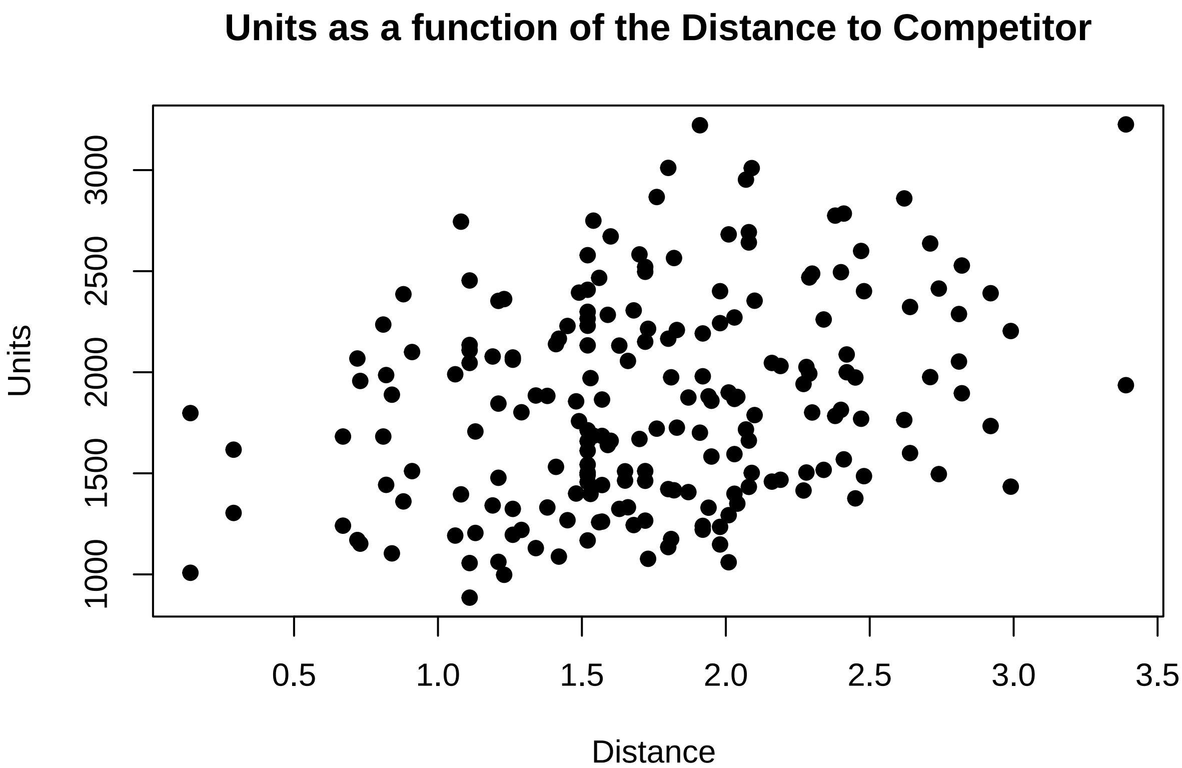
As illustrated discussing the impact of the population results in understanding that the population should not have an effect on the zone that a store falls in in the pricing model, if zone pricing is employed.

###DIST

```
plot(df.test$mdist, df.test$units, main="Units as a function of the MDIST to Competitor"
   xlab="MDIST ", ylab="Units", pch=19)
```




```
plot(df.test$distance, df.test$units, main="Units as a function of the Distance to Compe
   xlab="Distance ", ylab="Units", pch=19)
```



Mean distance (mdist) was not beneficial in my analysis in terms of being explanatory. The two above graphs display the same data points in space, however the postitive distance measures are easier to compare against one another. In the end we learn that the further the distance from the competitor, the more units sold.