

Market Share and Pricing Determination in the Ready-to-Eat Cereal Industry*

Christin Bivens[†]

May 4, 2018

Abstract

The ready-to-eat cereal market has been found to be an interesting case study as it has some of the highest profit margins in the grocery industry and the variety of products and advertising employed by major brands gives researchers a plethora of effects to study (Nevo (2001)). Generally, grocery is thought of as retail and therefore imperfect competition by economists. However, RTE cereal displays attributes of an oligopoly structure. By studying price determination and market share in the cereal industry, economists can have insights into the market structure and elasticity of retail trades. This article aims to provide a predictive linear model for market share in the ready-to-eat cereal industry that accounts for market share as a function of pricing, product characteristics of nutritional content, and product size.

1 Introduction

The market structure of the ready-to-eat (RTE) breakfast cereal industry provides insights into both imperfect competition and competitive oligopolies. Stemming from grocery retail (which is usually categorized as imperfect competition), RTE cereal companies have managed to create a

*Special thanks to Dr. Myongjin Kim, University of Oklahoma

[†]Department of Economics, University of Oklahoma. E-mail address: christin.e.bivens-1@ou.edu

market where only a few firms are dominant and cost-profit ratios are so high that collusion has been suspected (Nevo (2001)). In addition to making for an interesting case study, the RTE cereal industry also lends itself to economic research through relatively obtainable data on both product characteristics and firm costs (Stanley and Tschirhart 1991).

This paper looks to prior research and data cleaning/visualization methods to create a linear regression capturing market share in the industry. As market share and price have a relationship rooted in reverse causation, a regression to first predict price is necessary. The new predicted variable for price is labeled as \widehat{P}_{it} for use in the market share model. The response variable MS_{it} was specified as a function of the relationship between the inside good and outside goods at time t using a logit model. The explicit equations can be found in the empirical methods section.

The estimated coefficients from the findings were all statistically significant with p -values below .05 for the market share model. However, only 7 out of 12 coefficients for the pricing regression were significant. Given the time restraints of this research, the ratio of significant results is considered a triumph by the author. Results of the final model for market share found that calorie content, protein content, box size, and sugar content all were positively correlated with market share. Carbohydrate count and price had inverse relationships with market share. Given economic theory on the relationship between pricing and sales, the inverse relationship between price and market share for the RTE cereal industry was to be expected. The strong influence of product characteristics

2 Literature Review

"Measuring Market Power in the Ready-to-Eat Cereal Industry" by Aviv Nevo in 2001 is used as a general reference and starting off point for this study. Nevo creates multiple models to capture RTE demand then creates a nested mode with the goal of having a predictive model that is not hindered by unobservables. Time constraints and the scope of this project do not allow for Nevo's thorough approach to be replicated. However, the insight found by Nevo that the high cost-price margins are not a result of collusion in the RTE industry, but rather it is achieved through product differentiation is a benchmark for developing both the pricing and market share models.

Research prior to Nevo showed that despite the large profit margins for cereal, that the demand was not inelastic. "Using a Natural Experiment to Estimate price Elasticity: The 1974 Sugar Shortage and the Ready-to-Eat Cereal Market" by Neslin and Shoemaker 1983 finds that price elasticity for the cereal industry is relatively high. As firms gain market power the demand elasticity for their product decreases. This allows them to increase prices while reducing sales and consequently enjoy higher price-cost margins. Neslin and Shoemaker help to refute the idea that the RTE cereal industry is a price collusive oligopoly by using results from a natural experiment. The sugar shortage of 1974 created an increase in the price of sugar and therefore in an increase in the price of ready-to-eat breakfast cereals. This shock to the market provided data to compare consumer reactions. Consumers bought far less cereal at the heightened prices across multiple brands showing a strong responsiveness to price. Which Neslin and Shoemaker mathematically proved.

For constructing a market share variable the article "Implications of Market Structure for Elasticity Structure" by Russell and Bolton from 1988 was referenced. The authors "develop a parsimonious description of elasticity structure that relates market share and marketing mix variables

(e.g., price) to brand elasticities” (Russell and Bolton (1988)). Their research also concludes that the presence of a market dominating brand affects both pricing and market share for other products in the market. Both their findings with regards to relationships of products in the market and their use of a logit model for market share are included in this research.

The 1991 article ”Hedonic Prices for a Nondurable Good: The Case of Breakfast Cereals” by Stanley and Tschirart was used as reference for determining which variables to include in the pricing and market share regressions. Stanley and Tschirhart exclude calories from their study as they make the observation that factors such as sugar and fiber content greatly affect the caloric content and therefore it is not necessary to include. This research uses that logic to go in the opposite direction. When predictive models become overly laden with variables of interest they lose predictive power. Therefore, this research will include calories in the regression as a means of including the effects of product nutrition characteristics, such as fiber, that can be captured within the caloric content. Stanley and Tschirhart also find that fat content varies very little between cereal products and therefore are not a strong enough contributing factor to include in research. Their insight about fat is likewise used in this paper to further slim down the pricing estimate model.

To decide whether to include certain parent companies or all products in the data for the final model ”The Welfare Economics of Product Variety: An Application to the Ready-to-Eat Cereals Industry” by Scherer in 1979 was considered. Scherer lays the ground work for which brands control the market share in the RTE industry (General Mills, Kellogg Co, PepsiCo, and a few private brands like Quaker that can be umbrella-ed together). The paper asserts that the market share of brands highly affect the success of new products. the author considers whether there is a cannibalistic effect or if firms are purposefully generating high numbers of new products in order

to edge out smaller firms. Because Scherer finds a significant relationship between product variety and market power, this research will focus on the four largest parent companies. This allows for the results to reflect the core behavior of the market as well as which market strategies are effective.

3 Data

Data has been graciously provided by Dr. Kim of the University of Oklahoma from a previous joint research endeavor. The data contains records for 95 weeks of observations from 2001 to 2003 on 18 parent companies within the market. Each record is for a product description (i) for the given week (t). Product description in this case refers to a brand under a parent company that is specific to product size and type (e.g. Captain Crunch Berries 32oz.). The focus will be on products by the dominating companies in the market: General Mills, Kellogg Co., Private Labels, and PepsiCo (Scherer (1979)).

Looking at time series data allows the research to have insights on real pricing and product characteristics without possible issues arising from collecting data from an experiment(Neslin and Shoemaker (1983)). While there are disadvantages to working with observed data like not having an available control group, differences in selling environments, and possible changes in brand quality, the scope of this research is not heavily affected by those shortcomings. The focus of this paper is to model market share based on observable attributes, and therefore there is no need for counterfactual analysis.

Observed variables include time in terms of weeks, months, quarters, and years, product characteristics of nutritional content, box size, and character size on the box, cost inputs for the firm, unit sales per product description, total sales per product description, as well as others that were

not pertinent to this study. For the regressions the data was aggregated by product description per week.

4 Empirical Methods

All empirical exercises were done using R and RStudio. In addition to the literature review a process of data cleaning and visualization was used to determine correlation between variables of interest and the response variable. This process included using only one response variable and one variable of interest at a time to create scatter plots, summary statistics, correlation tests, and simple linear regressions. Examples of the plots between price and quantity sold for the four main parent companies can be found in the figures and tables section. This was done on the final focused data set as well as broken down to the four parent companies. The specification by parent company in the visualization stage was to compare effects to ensure that there was some sense of uniformity across the market. That is to say, that there was not an observed effect of a variable on price for one company that did not exist for another therefore throwing off the mean effects for the market.

In line with research and findings through data exploration, the pricing linear regression uses input variables of the observable product characteristics (calorie content, carbohydrate content, protein content, box size, and sugar content) to capture expectations of consumer preferences, cost inputs (corn, wheat, gas, sugar, electricity, and labor costs), as well as dummy variables for both time (weekly) and other products available in the market at weekly time t .

$$\widehat{P}_{it} = \beta_1 X_{Calories} + \beta_2 X_{Carbs} + \beta_3 X_{Protein} + \beta_4 X_{Size} + \beta_5 X_{Sugar} + \alpha_1 C_{Corn} + \alpha_2 C_{Gas} + \alpha_3 C_{Wheat} +$$

$$\alpha_4 C_{Sugar} + \alpha_5 C_{Electricity} + \alpha_6 C_{Labor} + \alpha_7 D_{Week} + \alpha_8 D_{ProductDescription}$$

(1)

Where \hat{P}_{it} is a continuous outcome variable for product description i in week t , and X_{it} are characteristics of product i for which the parameter of interest is β_j , C_{it} are input costs for product i and D_{it} are dummy variables for time (weeks) and available products in the market for which the parameter of interest is α_k .

The model for market share (MS_{it}) was constructed using the new price variable - as an effort to account for the reverse causality of price and MS, product characteristics (calorie content, carbohydrate content, protein content, box size, and sugar content), and dummy variables for time (weekly) and other products available in the market at weekly time t .

$$MS_{it} = \log(UnitSales_{it}/Population_t) - \log((Population_t - UnitSales_{it})/Population_t)$$

(2)

Where MS_{it} is a continuous outcome variable for product description i in week t as a function of unit sales for product description i in week t in relation to unit sales for the entire market population

at time t .

$$MS_{it} = \beta_1 X_{Calories} + \beta_2 X_{Carbs} + \beta_3 X_{Protein} + \beta_4 X_{Size} + \beta_5 X_{Sugar} + \alpha_1 \widehat{P}_{it} + \alpha_7 D_{Week} + \alpha_8 D_{ProductDescription}$$

(3)

Where MS_{it} is a continuous outcome variable for product description i in week t , and X_{it} are characteristics of product i for which the parameter of interest is β_j , and D_{it} are dummy variables for time (weeks) and available products in the market for which the parameter of interest is α_k .

5 Research Findings

The results for the pricing prediction regression can be found in Table 1. For \widehat{P}_{it} the coefficient for caloric content was .144 with a statistically significant p -value under .01. Indicating that as caloric content increases, firms are more likely to increase prices. For carbohydrates there was an inverse relationship with pricing with a coefficient of -1.847, and a statistically significant p -value under .01. The reasoning behind this could be as simple as inputs for carbohydrates are cheaper and therefore lower pricing. Indeed when sugar cost is considered there is a statistically significant coefficient of -.031. Sugar inputs would affect carbohydrate content and these are in line with each other. However, sugar content has a coefficient of .061 contradicting this theory - although the p -value was not significant. Protein and box size both have positive and significant relationships with price (4.824 and .091 respectively). This seems in line with basic economic theory that as inputs

increase so does pricing. Box size not only affects the input of the cereal itself, but the inputs for the box. Protein is often one of the more expensive food costs, therefore, these numbers seem logical. Labor cost (.077), electricity cost (-.042), wheat cost (-.018), and corn cost (.056) all were found to be statistically insignificant. Finally, gas cost was found to have an inverse relationship with price with a significant p -value of less than .01. This seems counter intuitive as gas will always add to the bottom line for a company. This and the lack of significant results on many of the variables of interest in the regression could be due to the fact that cereal companies enjoy economies of scope - therefore there could be a veiled effect on proper measurements of their product inputs on pricing. The inverse relationship with gas could be due to a decrease in average variable cost for gas as production increases.

For the market share model all coefficients were statistically significant with p -values below .01, with the exception of sugar content that had a p -value smaller than .05. Variables of interest that had a positive relationship with market share were caloric content (.577), protein (20.144), box size (.311) and sugar content (.699). The constructed pricing variable had an inverse relationship with market share, captured by a value of -2.873. This is very much in line with economic theory for imperfect or monopolistic competition - that the demand is more elastic than inelastic and consumers are sensitive to price. The only nutritional product characteristic that had an inverse relationship with MS was the amount of carbohydrates at a value of -7.642. these results can be reviewed in Table 2 under the Figures and Tables section.

6 Conclusion

The findings in this study reflect different aspects of previous research on the RTE cereal industry and economic theory. While few firms dominate the market and there is a high cost-price margin suggesting an oligopoly structure (Nevo (2001)), the coefficients of the regression reflect that customers are highly responsive to price and product characteristics as is in line with imperfect or monopolistic competition.

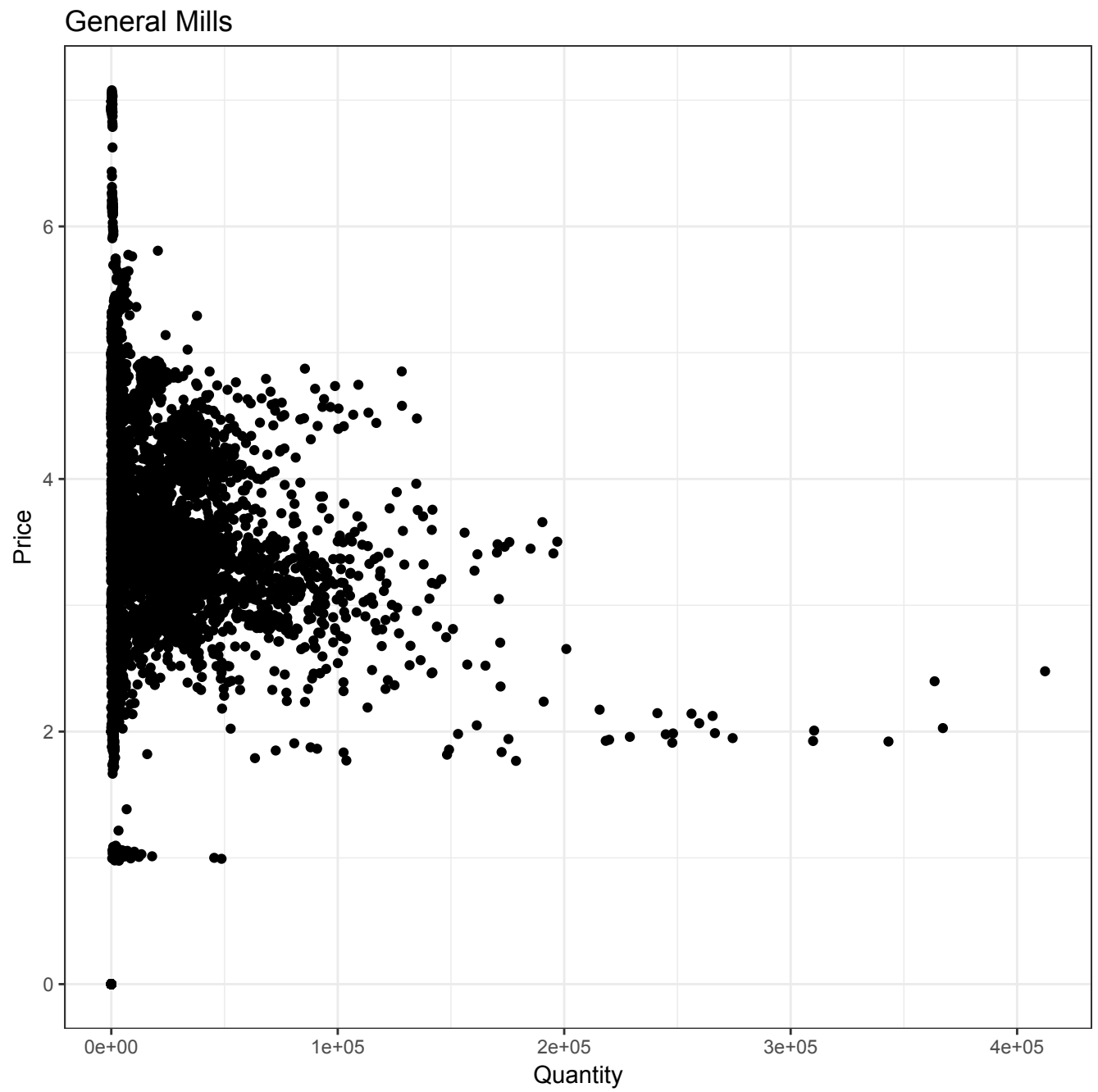
Many firms who enter an imperfectly competitive market, like grocery retail, do so with the hope of increasing their market share in order to escape long term zero profit theories. By studying the effects of pricing and product differentiation in the RTE industry, businesses can gain insight in how to greatly increase their market share and therefore increase profits and longevity. Specifically this linear regression model can be used to help predict market share of a new product by entry firms or an existing firm. This is achieved by using a logit model to predict market share taking into account market share of other products, as well as creating a new price variable that adjusts for the reverse-causal relationship of market share and price.

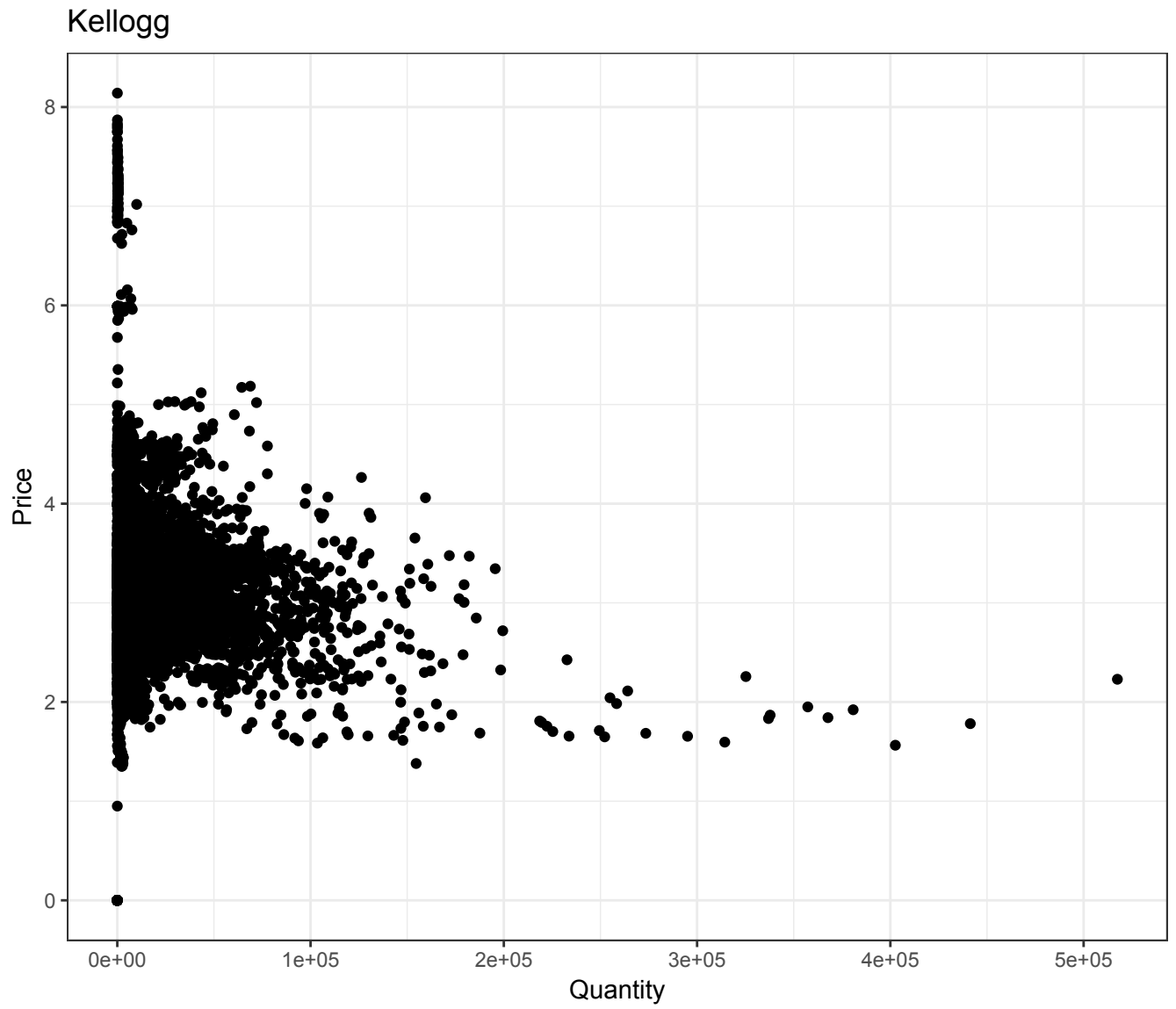
In addition to commercial use, the findings in this research can help economists to further study the boundaries and characteristics of market structure as industries/firms evolve from one structure to the next. The significant positive correlation between product characteristics and consumer preference show that differentiation is a large part of gaining market power. The less predictable effects of inputs on pricing lend to the theory that RTE cereal companies are able to create a higher price-cost margin by enjoying economies of scope, although this needs further research. Finally, the sensitivity of consumers to price reinforces the theory that demand in grocery is more elastic and therefore reducing price while increasing sales leads to a larger market share.

References

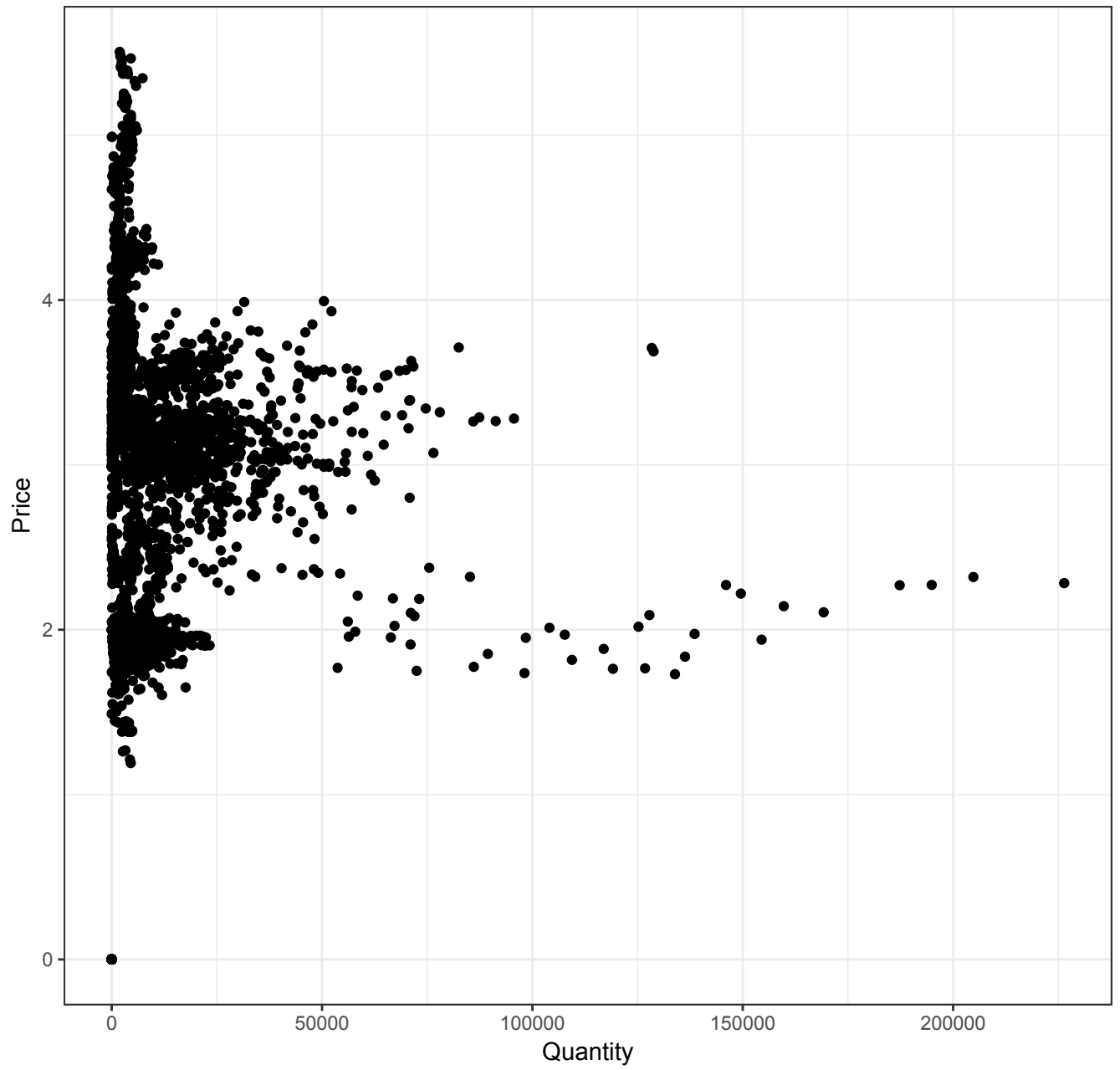
- Neslin, Scott A. and Robert W. Shoemaker. 1983. "Using a Natural Experiment to Estimate Price Elasticity: The 1974 Sugar Shortage and the Ready-to-Eat Cereal Market." *Journal of Marketing* 47 (1):44–57.
- Nevo, Aviv. 2001. "Measuring Market Power in the Ready-to-Eat Cereal Industry." *Econometrica* 69 (2):307–342.
- Russell, Gary J. and Ruth N. Bolton. 1988. "Implications of Market Structure for Elasticity Structure." *Journal of Marketing Research* 25 (3):229–241.
- Scherer, F. M. 1979. "The Welfare Economics of Product Variety: An Application to the Ready-to-Eat Cereals Industry." *The Journal of Industrial Economics* 28 (2):113–134.
- Stanley, Linda R. and John Tschirhart. 1991. "Hedonic Prices for a Nondurable Good: The Case of Breakfast Cereals." *The Review of Economics and Statistics* 73 (3):537–541.

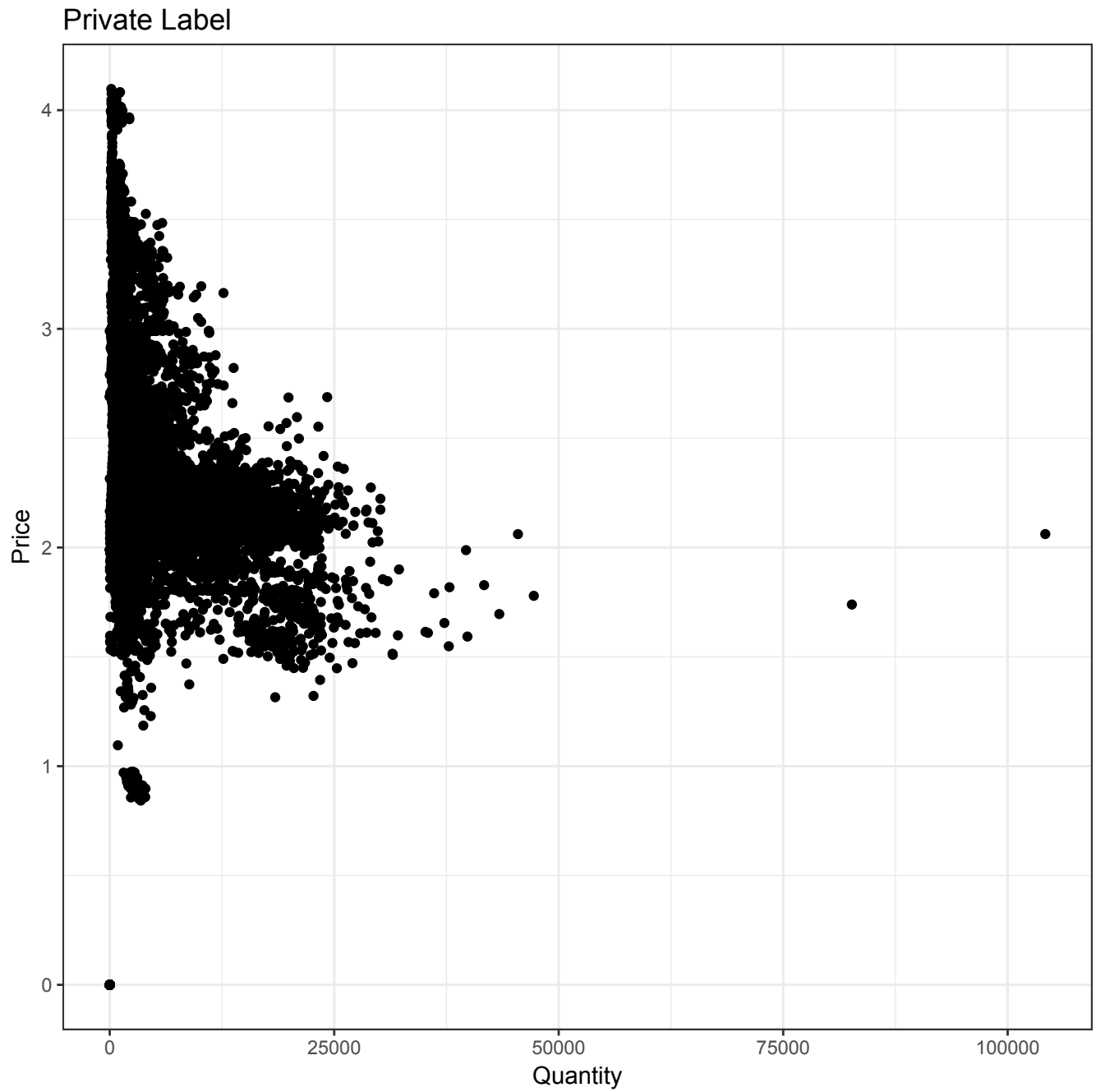
Figures and Tables





PepsiCo





7 Program

RTE CEREAL DATA ANALYSIS: Market Share and Pricing Determinants Christin Bivens Spring

2018

Table 1: Pricing Regression Results

	<i>Dependent variable:</i>
	Pricing
Calories	0.144*** (0.021)
Carbohydrates	−1.847*** (0.353)
Protein	4.824*** (1.131)
Box Size	0.091*** (0.005)
Sugar Content	0.061 (0.100)
Corn Cost	0.056 (0.071)
Gas Cost	−0.173*** (0.046)
Wheat Cost	−0.018 (0.018)
Sugar Cost	−0.031*** (0.004)
Electricity Cost	−0.042 (0.036)
Labor Cost	0.077 (0.172)
Week	0.003* (0.002)
Observations	37,047
R ²	0.671
Adjusted R ²	0.666
Residual Std. Error	0.692 (df = 36555)
F Statistic	151.564*** (df = 491; 36555)
<i>Note:</i>	*p<0.1; **p<0.05; ***p<0.01

Table 2: Market Share Regression Results

	<i>Dependent variable:</i>
	Y _{it}
Calories	0.577*** (0.069)
Carbohydrates	−7.642*** (1.086)
Protein	20.144*** (3.397)
Box Size	0.311*** (0.024)
Sugar Content	0.699** (0.288)
\widehat{P}_{it}	−2.873*** (0.216)
Week	0.002*** (0.001)
Observations	37,047
R ²	0.465
Adjusted R ²	0.457
Residual Std. Error	1.980 (df = 36560)
F Statistic	65.280*** (df = 486; 36560)
<i>Note:</i>	*p<0.1; **p<0.05; ***p<0.01

TABLE OF CONTENTS

1. Packages 2. Data 3. Creating New Variables 4. Making Tibbles of the Parent Companies
5. Exploring the Data Using the 4 Biggest Parent Companies 5.1 Data Frames and GG Plots 5.2
Summary Statistics 5.3 Correlation Tests 5.4 Simple Linear Regressions 6. Making a Tibble of 4
Biggest Parent Companies Combined 7. Creating the Predictive Models

1. PACKAGES

```
library(stringi) library(dplyr) library(tidyverse) library(ggthemes) library(knitr) library(haven)
library(covr) library(Rcpp) library(forcats) library(hms) library(testthat) library(tibble) library(readr)
library(ggplot2) library(stargazer)
```

2. LOADING/VIEWING DATA

```
cereal_i-read_dta("~/Users/christin/Downloads/CerealPD03052016.dta", encoding = 'latin1') converting STA
2) viewing header labels
```

```
head(cerealAttrParentLevel_ID) head(cerealAttrParentLevel) viewing parent company info gen-
eral mills = 1
```

```
tail(cerealYear) head(cerealYear) spans 2001 - 2003
```

```
Variables for price and parent company AttrParentLevel_ID(1 – 12) AttrParentLevel(company names)
```

```
Price variables AvgPriceperUnit
```

```
Cereal_i-as_tibble(cereal) converting.csv to tibble format
```

```
Cereal[is.na(Cereal)]_i-0 making nan values equal to 0
```

3. MAKING NEW VARIABLES

CerealPricing < $-(\text{CerealDollarSales}/\text{CerealUnitSales})$ making a solo column of prices

CerealPricePerOZ < $-(\text{CerealPricing}/\text{Cerealchar4boxsize})$ making a column of price per ounce

MAKING POPULATION COLUMN
WITH PER WEEK POPULATION

Cerealpop < $-NAweek1 < -filter(\text{Cereal}, \text{CerealWeek}=="1")$ *Cerealpop*[*CerealWeek*==1];
-
sum(*week1UnitSales*)*week2* < $-filter(\text{Cereal}, \text{CerealWeek}=="2")$ *Cerealpop*[*CerealWeek*==2];
-
sum(*week2UnitSales*)*week3* < $-filter(\text{Cereal}, \text{CerealWeek}=="3")$ *Cerealpop*[*CerealWeek*==3];
-
sum(*week3UnitSales*)*week4* < $-filter(\text{Cereal}, \text{CerealWeek}=="4")$ *Cerealpop*[*CerealWeek*==4];
-
sum(*week4UnitSales*)*week5* < $-filter(\text{Cereal}, \text{CerealWeek}=="5")$ *Cerealpop*[*CerealWeek*==5];
-
sum(*week5UnitSales*)*week6* < $-filter(\text{Cereal}, \text{CerealWeek}=="6")$ *Cerealpop*[*CerealWeek*==6];
-
sum(*week6UnitSales*)*week7* < $-filter(\text{Cereal}, \text{CerealWeek}=="7")$ *Cerealpop*[*CerealWeek*==7];
-
sum(*week7UnitSales*)*week8* < $-filter(\text{Cereal}, \text{CerealWeek}=="8")$ *Cerealpop*[*CerealWeek*==8];
-
sum(*week8UnitSales*)*week9* < $-filter(\text{Cereal}, \text{CerealWeek}=="9")$ *Cerealpop*[*CerealWeek*==9];
-
sum(*week9UnitSales*)*week10* < $-filter(\text{Cereal}, \text{CerealWeek}=="10")$ *Cerealpop*[*CerealWeek*==10];
-
sum(*week10UnitSales*)*week11* < $-filter(\text{Cereal}, \text{CerealWeek}=="11")$ *Cerealpop*[*CerealWeek*==11];
-
sum(*week11UnitSales*)*week12* < $-filter(\text{Cereal}, \text{CerealWeek}=="12")$ *Cerealpop*[*CerealWeek*==12];
-
sum(*week12UnitSales*)*week13* < $-filter(\text{Cereal}, \text{CerealWeek}=="13")$ *Cerealpop*[*CerealWeek*==13];
-
sum(*week13UnitSales*)*week14* < $-filter(\text{Cereal}, \text{CerealWeek}=="14")$ *Cerealpop*[*CerealWeek*==14];
-
sum(*week14UnitSales*)*week15* < $-filter(\text{Cereal}, \text{CerealWeek}=="15")$ *Cerealpop*[*CerealWeek*==15];
-
sum(*week15UnitSales*)*week16* < $-filter(\text{Cereal}, \text{CerealWeek}=="16")$ *Cerealpop*[*CerealWeek*==16];
-
sum(*week16UnitSales*)*week17* < $-filter(\text{Cereal}, \text{CerealWeek}=="17")$ *Cerealpop*[*CerealWeek*==17];
-
sum(*week17UnitSales*)*week18* < $-filter(\text{Cereal}, \text{CerealWeek}=="18")$ *Cerealpop*[*CerealWeek*==18];

```
sum(week18UnitSales)week19 <- filter(Cereal, CerealWeek=="19") Cerealpop[CerealWeek==19];
sum(week19UnitSales)week20 <- filter(Cereal, CerealWeek=="20") Cerealpop[CerealWeek==20];
sum(week20UnitSales)week21 <- filter(Cereal, CerealWeek=="21") Cerealpop[CerealWeek==21];
sum(week21UnitSales)week22 <- filter(Cereal, CerealWeek=="22") Cerealpop[CerealWeek==22];
sum(week22UnitSales)week23 <- filter(Cereal, CerealWeek=="23") Cerealpop[CerealWeek==23];
sum(week23UnitSales)week24 <- filter(Cereal, CerealWeek=="24") Cerealpop[CerealWeek==24];
sum(week24UnitSales)week25 <- filter(Cereal, CerealWeek=="25") Cerealpop[CerealWeek==25];
sum(week25UnitSales)week26 <- filter(Cereal, CerealWeek=="26") Cerealpop[CerealWeek==26];
sum(week26UnitSales)week27 <- filter(Cereal, CerealWeek=="27") Cerealpop[CerealWeek==27];
sum(week27UnitSales)week28 <- filter(Cereal, CerealWeek=="28") Cerealpop[CerealWeek==28];
sum(week28UnitSales)week29 <- filter(Cereal, CerealWeek=="29") Cerealpop[CerealWeek==29];
sum(week29UnitSales)week30 <- filter(Cereal, CerealWeek=="30") Cerealpop[CerealWeek==30];
sum(week30UnitSales)week31 <- filter(Cereal, CerealWeek=="31") Cerealpop[CerealWeek==31];
sum(week31UnitSales)week32 <- filter(Cereal, CerealWeek=="32") Cerealpop[CerealWeek==32];
sum(week32UnitSales)week33 <- filter(Cereal, CerealWeek=="33") Cerealpop[CerealWeek==33];
sum(week33UnitSales)week34 <- filter(Cereal, CerealWeek=="34") Cerealpop[CerealWeek==34];
sum(week34UnitSales)week35 <- filter(Cereal, CerealWeek=="35") Cerealpop[CerealWeek==35];
sum(week35UnitSales)week36 <- filter(Cereal, CerealWeek=="36") Cerealpop[CerealWeek==36];
sum(week36UnitSales)week37 <- filter(Cereal, CerealWeek=="37") Cerealpop[CerealWeek==37];
sum(week37UnitSales)week38 <- filter(Cereal, CerealWeek=="38") Cerealpop[CerealWeek==38];
sum(week38UnitSales)week39 <- filter(Cereal, CerealWeek=="39") Cerealpop[CerealWeek==39];
sum(week39UnitSales)week40 <- filter(Cereal, CerealWeek=="40") Cerealpop[CerealWeek==40];
sum(week40UnitSales)week41 <- filter(Cereal, CerealWeek=="41") Cerealpop[CerealWeek==41];
```

```
sum(week41UnitSales)week42 <- filter(Cereal, CerealWeek=="42") Cerealpop[CerealWeek==42];
sum(week42UnitSales)week43 <- filter(Cereal, CerealWeek=="43") Cerealpop[CerealWeek==43];
sum(week43UnitSales)week44 <- filter(Cereal, CerealWeek=="44") Cerealpop[CerealWeek==44];
sum(week44UnitSales)week45 <- filter(Cereal, CerealWeek=="45") Cerealpop[CerealWeek==45];
sum(week45UnitSales)week46 <- filter(Cereal, CerealWeek=="46") Cerealpop[CerealWeek==46];
sum(week46UnitSales)week47 <- filter(Cereal, CerealWeek=="47") Cerealpop[CerealWeek==47];
sum(week47UnitSales)week48 <- filter(Cereal, CerealWeek=="48") Cerealpop[CerealWeek==48];
sum(week48UnitSales)week49 <- filter(Cereal, CerealWeek=="49") Cerealpop[CerealWeek==49];
sum(week49UnitSales)week50 <- filter(Cereal, CerealWeek=="50") Cerealpop[CerealWeek==50];
sum(week50UnitSales)week51 <- filter(Cereal, CerealWeek=="51") Cerealpop[CerealWeek==51];
sum(week51UnitSales)week52 <- filter(Cereal, CerealWeek=="52") Cerealpop[CerealWeek==52];
sum(week52UnitSales)week53 <- filter(Cereal, CerealWeek=="53") Cerealpop[CerealWeek==53];
sum(week53UnitSales)week54 <- filter(Cereal, CerealWeek=="54") Cerealpop[CerealWeek==54];
sum(week54UnitSales)week55 <- filter(Cereal, CerealWeek=="55") Cerealpop[CerealWeek==55];
sum(week55UnitSales)week56 <- filter(Cereal, CerealWeek=="56") Cerealpop[CerealWeek==56];
sum(week56UnitSales)week57 <- filter(Cereal, CerealWeek=="57") Cerealpop[CerealWeek==57];
sum(week57UnitSales)week58 <- filter(Cereal, CerealWeek=="58") Cerealpop[CerealWeek==58];
sum(week58UnitSales)week59 <- filter(Cereal, CerealWeek=="59") Cerealpop[CerealWeek==59];
sum(week59UnitSales)week60 <- filter(Cereal, CerealWeek=="60") Cerealpop[CerealWeek==60];
sum(week60UnitSales)week61 <- filter(Cereal, CerealWeek=="61") Cerealpop[CerealWeek==61];
sum(week61UnitSales)week62 <- filter(Cereal, CerealWeek=="62") Cerealpop[CerealWeek==62];
sum(week62UnitSales)week63 <- filter(Cereal, CerealWeek=="63") Cerealpop[CerealWeek==63];
sum(week63UnitSales)week64 <- filter(Cereal, CerealWeek=="64") Cerealpop[CerealWeek==64];
```

```
sum(week64UnitSales)week65 <- filter(Cereal, CerealWeek=="65") Cerealpop[CerealWeek==65];
sum(week65UnitSales)week66 <- filter(Cereal, CerealWeek=="66") Cerealpop[CerealWeek==66];
sum(week66UnitSales)week67 <- filter(Cereal, CerealWeek=="67") Cerealpop[CerealWeek==67];
sum(week67UnitSales)week68 <- filter(Cereal, CerealWeek=="68") Cerealpop[CerealWeek==68];
sum(week68UnitSales)week69 <- filter(Cereal, CerealWeek=="69") Cerealpop[CerealWeek==69];
sum(week69UnitSales)week70 <- filter(Cereal, CerealWeek=="70") Cerealpop[CerealWeek==70];
sum(week70UnitSales)week71 <- filter(Cereal, CerealWeek=="71") Cerealpop[CerealWeek==71];
sum(week71UnitSales)week72 <- filter(Cereal, CerealWeek=="72") Cerealpop[CerealWeek==72];
sum(week72UnitSales)week73 <- filter(Cereal, CerealWeek=="73") Cerealpop[CerealWeek==73];
sum(week73UnitSales)week74 <- filter(Cereal, CerealWeek=="74") Cerealpop[CerealWeek==74];
sum(week74UnitSales)week75 <- filter(Cereal, CerealWeek=="75") Cerealpop[CerealWeek==75];
sum(week75UnitSales)week76 <- filter(Cereal, CerealWeek=="76") Cerealpop[CerealWeek==76];
sum(week76UnitSales)week77 <- filter(Cereal, CerealWeek=="77") Cerealpop[CerealWeek==77];
sum(week77UnitSales)week78 <- filter(Cereal, CerealWeek=="78") Cerealpop[CerealWeek==78];
sum(week78UnitSales)week79 <- filter(Cereal, CerealWeek=="79") Cerealpop[CerealWeek==79];
sum(week79UnitSales)week80 <- filter(Cereal, CerealWeek=="80") Cerealpop[CerealWeek==80];
sum(week80UnitSales)week81 <- filter(Cereal, CerealWeek=="81") Cerealpop[CerealWeek==81];
sum(week81UnitSales)week82 <- filter(Cereal, CerealWeek=="82") Cerealpop[CerealWeek==82];
sum(week82UnitSales)week83 <- filter(Cereal, CerealWeek=="83") Cerealpop[CerealWeek==83];
sum(week83UnitSales)week84 <- filter(Cereal, CerealWeek=="84") Cerealpop[CerealWeek==84];
sum(week84UnitSales)week85 <- filter(Cereal, CerealWeek=="85") Cerealpop[CerealWeek==85];
sum(week85UnitSales)week86 <- filter(Cereal, CerealWeek=="86") Cerealpop[CerealWeek==86];
sum(week86UnitSales)week87 <- filter(Cereal, CerealWeek=="87") Cerealpop[CerealWeek==87];
```

```

sum(week87UnitSales)week88 <- filter(Cereal, CerealWeek=="88") Cerealpop[CerealWeek==88];-
sum(week88UnitSales)week89 <- filter(Cereal, CerealWeek=="89") Cerealpop[CerealWeek==89];-
sum(week89UnitSales)week90 <- filter(Cereal, CerealWeek=="90") Cerealpop[CerealWeek==90];-
sum(week90UnitSales)week91 <- filter(Cereal, CerealWeek=="91") Cerealpop[CerealWeek==91];-
sum(week91UnitSales)week92 <- filter(Cereal, CerealWeek=="92") Cerealpop[CerealWeek==92];-
sum(week92UnitSales)week93 <- filter(Cereal, CerealWeek=="93") Cerealpop[CerealWeek==93];-
sum(week93UnitSales)week94 <- filter(Cereal, CerealWeek=="94") Cerealpop[CerealWeek==94];-
sum(week94UnitSales)week95 <- filter(Cereal, CerealWeek=="95") Cerealpop[CerealWeek==95];-
sum(week95UnitSales)

```

CONSTRUCTING A Y VARIABLE

$Y_{it} = \text{MARKET SHARE} = \text{QUANTITY SOLD PER PRODUCT DESCRIPTION PER WEEK} /$
 $\text{POPULATION PER WEEK} = \log(\text{MS}_{it}) - \log(\text{MS}_{ot}) = (\log(\text{ProductDescriptionUnitSales} / \text{Cerealpop}))$
 $- \log((\text{pop} - \text{ProductDescriptionUnitSales}) / \text{pop}))$

$\text{MS}_{ot} = \text{outside goods at time } t / \text{population at time } t = (\text{population} - \text{total quantity}) / \text{population}$

$\text{Cereal}Y_{it} <- ((\log(\text{CerealUnitSales} / \text{Cerealpop})) - \log((\text{Cerealpop} - \text{CerealUnitSales}) / \text{Cerealpop}))$

$\text{Cereal}[\text{Cereal}=="Inf"]_i \leftarrow 0$ taking care of infinity values $\text{Cereal}[\text{is.na}(\text{Cereal})]_i \leftarrow 0$ making nan values equal to 0 again (had nan in pricing column)

4. MAKING PARENT LEVEL DATA

SETS: AS TIBBLES

```

GeneralMills <- filter(Cereal, AttrParentLevel == "GENERALMILLS") KelloggCo <- filter(Cereal, AttrParentLevel ==
"KELLOGGCO") PepsiCo <- filter(Cereal, AttrParentLevel == "PEPSICOINC") USMills <- filter(Cereal, AttrParentLevel ==
"USMILLSINC") HealthValley <- filter(Cereal, AttrParentLevel == "HEALTHVALLEY - 100000000000TUM")

```

```

-filter(Cereal,AttrParentLevel=="MCKEEFOODSCORPORATION")MaltOMeal < -filter(Cereal,AttrParentLevel=="MALT-O-MEALCO")PrivateLabel < -filter(Cereal,AttrParentLevel=="PRIVATELABEL")Weetabix < -filter(Cereal,AttrParentLevel=="WEETABIXLTD")OrganicMilling < -filter(Cereal,AttrParentLevel=="ORGANICMILLINGCOMPANY")WorldFinerFoods < -filter(Cereal,AttrParentLevel=="WORLDFINERFOODS")TheBaker < -filter(Cereal,AttrParentLevel=="THEBAKER")PhilipMorris < -filter(Cereal,AttrParentLevel=="PHILIPMORRISCOINC")HainCelestial < -filter(Cereal,AttrParentLevel=="THEHAINCELESTIALGROUPOFAMERICA")Weetibakes < -filter(Cereal,AttrParentLevel=="WEETIBAKESLTD")OutrageousFruitandGrains < -filter(Cereal,AttrParentLevel=="OUTRAGEOUSFRUITANDGRAINS")
-filter(Cereal,AttrParentLevel=="-10000000000TURE'SPATH")

```

5. Exploring the Data

four biggest we are most concerned with are: GENERAL MILLS KELLOGG PEPSICO PRIVATE LABEL

FINDING HOW MANY PRODUCT DESCRIPTIONS THERE WILL BE

```

unique(sort(GeneralMillsProductDescription)) 86 unique(sort(KelloggCoProductDescription))
108 unique(sort(PepsiCoProductDescription))149unique(sort(PrivateLabelProductDescription))
142 TOTAL = 485 *95 WEEKS = 46075 RECORDS

```

MAKING DATA FRAMES OF THE 4 PARENT GROUPS FOR PRICE/QUANTITY

```

GeneralMillsQuantPrice;-cbind(GeneralMillsPricing,GeneralMillsUnitSales) GeneralMillsQuant;-
data.frame(GeneralMillsQuantPrice) colnames(GeneralMillsQuant)[1]='Price' colnames(GeneralMillsQuant)[2]

```


= 'Quantity'

```
KelloggQuantPrice;-cbind(Kellogg $_{Co}$ Pricing,Kellogg $_{Co}$ UnitSales) KelloggQuant;-data.frame(KelloggQuantPrice,
colnames(KelloggQuant)[1] = 'Price' colnames(KelloggQuant)[2] = 'Quantity'
```

```
PrivateQuantPrice;-cbind(Private $_{Label}$ Pricing,Private $_{Label}$ UnitSales) PrivateQuant;-data.frame(PrivateQuantPrice,
colnames(PrivateQuant)[1] = 'Price' colnames(PrivateQuant)[2] = 'Quantity'
```

```
PepsiCoQuantPrice;-cbind(PepsiCoPricing,PepsiCoUnitSales) PepsiCoQuant;-data.frame(PepsiCoQuantPrice,
colnames(PepsiCoQuant)[1] = 'Price' colnames(PepsiCoQuant)[2] = 'Quantity'
```

MAKING GGLOTS OF THE 4 PARENT GROUPS FOR PRICE/QUANTITY

```
ggplot(data=GeneralMillsQuant,aes(x=Quantity, y =Price))+ geom $_{point}$ () + coord $_{cartesian}$ () +
scale $_{color_g}$ gradient() + theme $_{bw}$ () + ggtitle("GeneralMills")
```

```
ggplot(data = KelloggQuant, aes(x= Quantity, y = Price))+ geom $_{point}$ () + coord $_{cartesian}$ () +
scale $_{color_g}$ gradient() + theme $_{bw}$ () + ggtitle("Kellogg")
```

```
ggplot(data = PrivateQuant, aes(x= Quantity, y = Price))+ geom $_{point}$ () + coord $_{cartesian}$ () +
scale $_{color_g}$ gradient() + theme $_{bw}$ () + ggtitle("PrivateLabel")
```

```
ggplot(data = PepsiCoQuant, aes(x= Quantity, y = Price))+ geom $_{point}$ () + coord $_{cartesian}$ () +
scale $_{color_g}$ gradient() + theme $_{bw}$ () + ggtitle("PepsiCo")
```

MAKING DATA FRAMES OF THE 4 PARENT GROUPS FOR PRICE/WEEK

-

```
GeneralMillsWeekPrice;-cbind(General $_{Mills}$ Pricing,General $_{Mills}$ Week) GeneralMillsWeek;-data.frame(GeneralMillsWeekPrice,
colnames(GeneralMillsWeek)[1] = 'Price' colnames(GeneralMillsWeek)[2] = 'Week'
```

```
KelloggWeekPrice;-cbind(Kellogg $_{Co}$ Pricing,Kellogg $_{Co}$ Week) KelloggWeek;-data.frame(KelloggWeekPrice)
```

```
colnames(KelloggWeek)[1] = 'Price' colnames(KelloggWeek)[2] = 'Week'
```

```
PrivateWeekPrice;-cbind(PrivateLabelPricing,PrivateLabelWeek) PrivateWeekPrice;-data.frame(PrivateWeekPrice)
colnames(PrivateWeek)[1] = 'Price' colnames(PrivateWeek)[2] = 'Week'
```

```
PepsiCoWeekPrice;-cbind(PepsiCoPricing,PepsiCoWeek) PepsiCoWeekPrice;-data.frame(PepsiCoWeekPrice)
colnames(PepsiCoWeek)[1] = 'Price' colnames(PepsiCoWeek)[2] = 'Week'
```

MAKING GGPLOTS OF THE 4 PARENT GROUPS FOR PRICE/WEEK

```
ggplot(data=GeneralMillsWeek,aes(x=Week, y =Price))+ geompoint() + coordcartesian() +
scalecolorgradient() + themebw() + ggtitle("GeneralMills")
```

```
ggplot(data=KelloggWeek, aes(x=Week, y=Price))+ geompoint() + coordcartesian() + scalecolorgradient() +
themebw() + ggtitle("Kellogg")
```

```
ggplot(data=PrivateWeek, aes(x=Week, y=Price))+ geompoint() + coordcartesian() + scalecolorgradient() +
themebw() + ggtitle("PrivateLabel")
```

```
ggplot(data = PepsiCoWeek, aes(x=Week, y =Price))+ geompoint() + coordcartesian() + scalecolorgradient() +
themebw() + ggtitle("PepsiCo")
```

MAKING DATA FRAMES OF THE 4 PARENT GROUPS FOR PRICE PER OZ/WEEK

```
GeneralMillsWeekPPO;-cbind(GeneralMillsPricePerOZ,GeneralMillsWeek) GeneralMillsWeekPPO;-
data.frame(GeneralMillsWeekPPO) colnames(GeneralMillsWeekPPO)[1] = 'Price' colnames(GeneralMillsWeekPPO)[2] = 'Week'
```

```
KelloggWeekPPO;-cbind(KelloggCoPricePerOZ,KelloggCoWeek) KelloggWeekPPO;-data.frame(KelloggWeekPPO)
colnames(KelloggWeekPPO)[1] = 'Price' colnames(KelloggWeekPPO)[2] = 'Week'
```

```
PrivateWeekPPO_i-cbind(Private_LabelPricePerOZ,Private_LabelWeek) PrivateWeekPPO_i-data.frame(PrivateW
colnames(PrivatePPO)[1] = 'Price' colnames(PrivatePPO)[2] = 'Week'
```

```
PepsiCoWeekPPO_i-cbind(PepsiCoPricePerOZ,PepsiCoWeek) PepsiCoWeekPPO_i-data.frame(PepsiCoWeek
colnames(PepsiCoWeekPPO)[1] = 'Price' colnames(PepsiCoWeekPPO)[2] = 'Week'
```

MAKING GGPLOTS OF THE 4 PAR-

ENT GROUPS FOR PRICE PER OZ/WEEK

```
ggplot(data=GeneralMillsWeekPPO,aes(x=Week, y=Price))+ geom_point() + coord_cartesian() +
scale_color_gradient() + theme_bw() + ggtitle("GeneralMills")
```

```
ggplot(data = KelloggWeekPPO, aes(x=Week, y=Price))+ geom_point() + coord_cartesian() +
scale_color_gradient() + theme_bw() + ggtitle("Kellogg")
```

```
ggplot(data = PrivateWeekPPO, aes(x=Week, y=Price))+ geom_point() + coord_cartesian() +
scale_color_gradient() + theme_bw() + ggtitle("PrivateLabel")
```

```
ggplot(data = PepsiCoWeekPPO, aes(x=Week, y=Price))+ geom_point() + coord_cartesian() +
scale_color_gradient() + theme_bw() + ggtitle("PepsiCo")
```

MAKING DATA FRAMES OF THE

4 PARENT GROUPS FOR Quantity/Week

```
GeneralMillsWeekQ_i-cbind(General_MillsUnitSales,General_MillsWeek) GeneralMillsWeekQ_i-
data.frame(GeneralMillsWeekQ) colnames(GeneralMillsWeekQ)[1] = 'Quantity' colnames(GeneralMillsWeekQ)
= 'Week'
```

```
KelloggWeekQ_i-cbind(KelloggCoUnitSales,KelloggCoWeek) KelloggWeekQ_i-data.frame(KelloggWeekQ)
colnames(KelloggWeekQ)[1] = 'Quantity' colnames(KelloggWeekQ)[2] = 'Week'
```

```
PrivateWeekQ;-cbind(PrivateLabelUnitSales,PrivateLabelWeek) PrivateWeekQ;-data.frame(PrivateWeekQ)
colnames(PrivateQ)[1] = 'Quantity' colnames(PrivateQ)[2] = 'Week'
```

```
PepsiCoWeekQ;-cbind(PepsiCoUnitSales,PepsiCoWeek) PepsiCoWeekQ;-data.frame(PepsiCoWeekQ)
colnames(PepsiCoWeekQ)[1] = 'Quantity' colnames(PepsiCoWeekQ)[2] = 'Week'
```

MAKING GGPLOTS OF THE 4 PARENT GROUPS FOR QUANTITY/WEEK

```
ggplot(data=GeneralMillsWeekQ,aes(x=Week, y=Quantity))+ geompoint()+coordcartesian()+
scalecolorgradient()+themebw()+ggtitle("GeneralMills")
```

```
ggplot(data = KelloggWeekQ, aes(x=Week, y=Quantity))+ geompoint()+coordcartesian()+
scalecolorgradient()+themebw()+ggtitle("Kellogg")
```

```
ggplot(data = PrivateWeekQ, aes(x=Week, y=Quantity))+ geompoint()+coordcartesian()+
scalecolorgradient()+themebw()+ggtitle("PrivateLabel")
```

```
ggplot(data = PepsiCoWeekQ, aes(x=Week, y=Quantity))+ geompoint()+coordcartesian()+
scalecolorgradient()+themebw()+ggtitle("PepsiCo")
```

MAKING DATA FRAMES OF THE 4 PARENT GROUPS FOR PRICE/GAS

```
GeneralMillsGas;-cbind(GeneralMillsPricing,GeneralMillsgas) GeneralMillsGas;-data.frame(GeneralMillsGas)
colnames(GeneralMillsGas)[1] = 'Price' colnames(GeneralMillsGas)[2] = 'Gas'
```

```
KelloggGas;-cbind(KelloggCOPricing,KelloggCOgas) KelloggGas;-data.frame(KelloggGas) col-
names(KelloggGas)[1] = 'Price' colnames(KelloggGas)[2] = 'Gas'
```

```
PrivateGas;-cbind(PrivateLabelPricing,PrivateLabelgas) PrivateGas;-data.frame(PrivateGas) col-
names(PrivateGas)[1] = 'Price' colnames(PrivateGas)[2] = 'Gas'
```

```
PepsiCoGas;-cbind(PepsiCoPricing,PepsiCoGas) PepsiCoGas;-data.frame(PepsiCoGas) colnames(PepsiGas)
= 'Price' colnames(PepsiGas)[2] = 'Gas'
```

MAKING GGPLOTS OF THE 4 PAR-

ENT GROUPS FOR PRICE/GAS

```
ggplot(data=GeneralMillsGas,aes(x=Gas, y =Price))+ geom_point() + coord_cartesian() + scale_color_gradient() +
theme_bw() + ggtitle("GeneralMills")
```

```
ggplot(data=KelloggGas, aes(x=Gas, y=Price))+ geom_point() + coord_cartesian() + scale_color_gradient() +
theme_bw() + ggtitle("Kellogg")
```

```
ggplot(data=PrivateGas, aes(x=Gas, y=Price))+ geom_point() + coord_cartesian() + scale_color_gradient() +
theme_bw() + ggtitle("PrivateLabel")
```

```
ggplot(data = PepsiCoGas, aes(x=Gas, y =Price))+ geom_point() + coord_cartesian() + scale_color_gradient() +
theme_bw() + ggtitle("PepsiCo")
```

MAKING DATA FRAMES OF THE

4 PARENT GROUPS FOR PRICE/LABOR COSTS

```
GeneralMillsLaborCost;-cbind(GeneralMillsPricing,GeneralMillsLaborCost) GeneralMillsLaborCost;-
data.frame(GeneralMillsLaborCost) colnames(GeneralMillsLaborCost)[1] = 'Price' colnames(GeneralMillsLabor
= 'LaborCost'
```

```
KelloggLaborCost;-cbind(KelloggCoPricing,KelloggCoLaborCost) KelloggLaborCost;-data.frame(KelloggLa
colnames(KelloggLaborCost)[1] = 'Price' colnames(KelloggLaborCost)[2] = 'LaborCost'
```

```
PrivateLaborCost;-cbind(PrivateLabelPricing,PrivateLabelLaborCost) PrivateLaborCost;-data.frame(PrivateL
colnames(PrivateLaborCost)[1] = 'Price' colnames(PrivateLaborCost)[2] = 'LaborCost'
```

```
PepsiCoLaborCost;-cbind(PepsiCoPricing,PepsiCoLaborCost) PepsiCoLaborCost;-data.frame(PepsiCoLabo
```

```
colnames(PepsiCoLaborCost)[1] = 'Price' colnames(PepsiCoLaborCost)[2] = 'LaborCost'
```

MAKING DATA FRAMES OF THE 4 PARENT GROUPS FOR PRICE/CALORIES

```
GeneralMillsCalories<-cbind(GeneralMillsPricing,GeneralMillschar1_calories)GeneralMillsCalories <-  
-data.frame(GeneralMillsCalories)colnames(GeneralMillsCalories)[1]='Price'colnames(GeneralMillsCalories)[2]='Calories'
```

```
KelloggCalories<-cbind(KelloggCoPricing,KelloggCochar1_calories)KelloggCalories <- -data.frame(KelloggCalories)  
colnames(KelloggCalories)[2]='Calories'
```

```
PrivateCalories<-cbind(PrivateLabelPricing,PrivateLabelchar1_calories)PrivateCalories <- -data.frame(PrivateCalories)  
colnames(PrivateCalories)[2]='Calories'
```

```
PepsiCoCalories<-cbind(PepsiCoPricing,PepsiCochar1_calories)PepsiCoCalories <- -data.frame(PepsiCoCalories)  
colnames(PepsiCoCalories)[2]='Calories'
```

MAKING DATA FRAMES OF THE 4 PARENT GROUPS FOR PRICE/CARBS

```
GeneralMillsCarbs<-cbind(GeneralMillsPricing,GeneralMillschar2_carb)GeneralMillsCarbs <-  
-data.frame(GeneralMillsCarbs)colnames(GeneralMillsCarbs)[1]='Price'colnames(GeneralMillsCarbs)[2]='Carbs'
```

```
KelloggCarbs<-cbind(KelloggCoPricing,KelloggCochar2_carb)KelloggCarbs <- -data.frame(KelloggCarbs)  
colnames(KelloggCarbs)[2]='Carbs'
```

```
PrivateCarbs<-cbind(PrivateLabelPricing,PrivateLabelchar2_carb)PrivateCarbs <- -data.frame(PrivateCarbs)  
colnames(PrivateCarbs)[2]='Carbs'
```

```
PepsiCoCarbs_i-cbind(PepsiCoPricing,PepsiCochar2_carb)PepsiCoCarbs <-data.frame(PepsiCoCarbs)colnames(PepsiCoCarbs)[2]='Carbs'
```

MAKING DATA FRAMES OF THE
4 PARENT GROUPS FOR PRICE/PROTEIN

```
GeneralMillsProtein_i-cbind(GeneralMillsPricing,GeneralMillschar3_protein)GeneralMillsProtein <-data.frame(GeneralMillsProtein)colnames(GeneralMillsProtein)[1]='Price'colnames(GeneralMillsProtein)[2]='Protein'
```

```
KelloggProtein_i-cbind(KelloggCoPricing,KelloggCochar3_protein)KelloggProtein <-data.frame(KelloggProtein)colnames(KelloggProtein)[2]='Protein'
```

```
PrivateProtein_i-cbind(PrivateLabelPricing,PrivateLabelchar3_protein)PrivateProtein <-data.frame(PrivateProtein)colnames(PrivateProtein)[2]='Protein'
```

```
PepsiCoProtein_i-cbind(PepsiCoPricing,PepsiCochar3_protein)PepsiCoProtein <-data.frame(PepsiCoProtein)colnames(PepsiCoProtein)[2]='Protein'
```

MAKING DATA FRAMES OF THE
4 PARENT GROUPS FOR PRICE/BOXSIZE

```
GeneralMillsBox_i-cbind(GeneralMillsPricing,GeneralMillschar4_boxsize)GeneralMillsBox <-data.frame(GeneralMillsBox)colnames(GeneralMillsBox)[2]='Box'
```

```
KelloggBox_i-cbind(KelloggCoPricing,KelloggCochar4_boxsize)KelloggBox <-data.frame(KelloggBox)colnames(KelloggBox)[2]='Box'
```

```
PrivateBox_i-cbind(PrivateLabelPricing,PrivateLabelchar4_boxsize)PrivateBox <-data.frame(PrivateBox)colnames(PrivateBox)[2]='Box'
```

```
PepsiCoBox<-cbind(PepsiCoPricing,PepsiCochar4_boxsize)PepsiCoBox<-data.frame(PepsiCoBox)colnames(PepsiCoBox)[1]='Price' colnames(PepsiCoBox)[2]='Box'
```

MAKING DATA FRAMES OF THE
4 PARENT GROUPS FOR PRICE/SUGAR

```
GeneralMillsSugar<-cbind(GeneralMillsPricing,GeneralMillschar5_sugar)GeneralMillsSugar<-data.frame(GeneralMillsSugar)colnames(GeneralMillsSugar)[1]='Price' colnames(GeneralMillsSugar)[2]='Sugar'
```

```
KelloggSugar<-cbind(KelloggCoPricing,KelloggCochar5_sugar)KelloggSugar<-data.frame(KelloggSugar)colnames(KelloggSugar)[1]='Price' colnames(KelloggSugar)[2]='Sugar'
```

```
PrivateSugar<-cbind(PrivateLabelPricing,PrivateLabelchar5_sugar)PrivateSugar<-data.frame(PrivateSugar)colnames(PrivateSugar)[1]='Price' colnames(PrivateSugar)[2]='Sugar'
```

```
PepsiCoSugar<-cbind(PepsiCoPricing,PepsiCochar5_sugar)PepsiCoSugar<-data.frame(PepsiCoSugar)colnames(PepsiCoSugar)[1]='Price' colnames(PepsiCoSugar)[2]='Sugar'
```

MAKING DATA FRAMES OF THE
4 PARENT GROUPS FOR PRICE/CORN

```
GeneralMillsCorn<-cbind(GeneralMillsPricing,GeneralMillsCORN) GeneralMillsCorn<-data.frame(GeneralMillsCorn)colnames(GeneralMillsCorn)[1]='Price' colnames(GeneralMillsCorn)[2]='Corn'
```

```
KelloggCorn<-cbind(KelloggCoPricing,KelloggCoCORN) KelloggCorn<-data.frame(KelloggCorn)colnames(KelloggCorn)[1]='Price' colnames(KelloggCorn)[2]='Corn'
```

```
PrivateCorn<-cbind(PrivateLabelPricing,PrivateLabelCORN) PrivateCorn<-data.frame(PrivateCorn)colnames(PrivateCorn)[1]='Price' colnames(PrivateCorn)[2]='Corn'
```



```
PepsiCoCorn;-cbind(PepsiCoPricing,PepsiCocorn) PepsiCoCorn;-data.frame(PepsiCoCorn) col-  
names(PepsiCoCorn)[1] = 'Price' colnames(PepsiCoCorn)[2] = 'Corn'
```

MAKING DATA FRAMES OF THE
4 PARENT GROUPS FOR PRICE/WHEAT

```
GeneralMillsWheat;-cbind(GeneralMillsPricing,GeneralMillsWheat) GeneralMillsWheat;-data.frame(General  
colnames(GeneralMillsWheat)[1] = 'Price' colnames(GeneralMillsWheat)[2] = 'Wheat'
```

```
KelloggWheat;-cbind(KelloggCoPricing,KelloggCowheat) KelloggWheat;-data.frame(KelloggWheat)  
colnames(KelloggWheat)[1] = 'Price' colnames(KelloggWheat)[2] = 'Wheat'
```

```
PrivateWheat;-cbind(PrivateLabelPricing,PrivateLabelWheat) PrivateWheat;-data.frame(PrivateWheat)  
colnames(PrivateWheat)[1] = 'Price' colnames(PrivateWheat)[2] = 'Wheat'
```

```
PepsiCoWheat;-cbind(PepsiCoPricing,PepsiCowheat) PepsiCoWheat;-data.frame(PepsiCoWheat)  
colnames(PepsiCoWheat)[1] = 'Price' colnames(PepsiCoWheat)[2] = 'Wheat'
```

MAKING DATA FRAMES OF THE
4 PARENT GROUPS FOR PRICE/SUGAR COST

```
GeneralMillsSugCost;-cbind(GeneralMillsPricing,GeneralMillsSugCost) GeneralMillsSugCost;-data.frame(Gen  
colnames(GeneralMillsSugCost)[1] = 'Price' colnames(GeneralMillsSugCost)[2] = 'SugCost'
```

```
KelloggSugCost;-cbind(KelloggCoPricing,KelloggCosugar) KelloggSugCost;-data.frame(KelloggSugCost)  
colnames(KelloggSugCost)[1] = 'Price' colnames(KelloggSugCost)[2] = 'SugCost'
```

```
PrivateSugCost;-cbind(PrivateLabelPricing,PrivateLabelsugar) PrivateSugCost;-data.frame(PrivateSugCost)  
colnames(PrivateSugCost)[1] = 'Price' colnames(PrivateSugCost)[2] = 'SugCost'
```

```
PepsiCoSugCost;-cbind(PepsiCoPricing,PepsiCosugar) PepsiCoSugCost;-data.frame(PepsiCoSugCost)
```

colnames(PepsiCoSugCost)[1] = 'Price' colnames(PepsiCoSugCost)[2] = 'SugCost'

----- GENERAL MILLS SUMMARY TABLES -----

stargazer(summary(GeneralMillsQuant)) stargazer(summary(GeneralMillsWeek)) stargazer(summary(GeneralMillsWeekQ)) stargazer(summary(GeneralMillsCalories)) stargazer(summary(GeneralMillsProtein)) stargazer(summary(GeneralMillsBox)) stargazer(summary(GeneralMillsGas)) stargazer(summary(GeneralMillsCorn)) stargazer(summary(GeneralMillsSugCost)) stargazer(summary(GeneralMillsLaborCost))

----- KELLOGG SUMMARY TABLES -----

stargazer(summary(KelloggQuant)) stargazer(summary(KelloggWeek)) stargazer(summary(KelloggWeekPPO)) stargazer(summary(KelloggWeekQ)) stargazer(summary(KelloggCalories)) stargazer(summary(KelloggCarbs)) stargazer(summary(KelloggProtein)) stargazer(summary(KelloggBox)) stargazer(summary(KelloggSugar)) stargazer(summary(KelloggGas)) stargazer(summary(KelloggCorn)) stargazer(summary(KelloggWheat)) stargazer(summary(KelloggSugCost)) stargazer(summary(KelloggLaborCost))

----- PRIVATE LABEL SUMMARY TABLES -----

stargazer(summary(PrivateQuant)) stargazer(summary(PrivateWeek)) stargazer(summary(PrivateWeekPPO)) stargazer(summary(PrivateWeekQ)) stargazer(summary(PrivateCalories)) stargazer(summary(PrivateCarbs)) stargazer(summary(PrivateProtein)) stargazer(summary(PrivateBox)) stargazer(summary(PrivateSugar)) stargazer(summary(PrivateGas)) stargazer(summary(PrivateCorn)) stargazer(summary(PrivateWheat)) stargazer(summary(PrivateSugCost)) stargazer(summary(PrivateLaborCost))

----- PEPSI CO SUMMARY TABLES -----

stargazer(summary(PepsiCoQuant)) stargazer(summary(PepsiCoWeek)) stargazer(summary(PepsiCoWeekPP
stargazer(summary(PepsiCoWeekQ)) stargazer(summary(PepsiCoCalories)) stargazer(summary(PepsiCoCarbs))
stargazer(summary(PepsiCoProtein)) stargazer(summary(PepsiCoBox)) stargazer(summary(PepsiCoSugar))
stargazer(summary(PepsiCoGas)) stargazer(summary(PepsiCoCorn)) stargazer(summary(PepsiCoWheat))
stargazer(summary(PepsiCoSugCost)) stargazer(summary(PepsiCoLaborCost))

GENERAL MILLS SIMPLE LINEAR REGRESSIONS

GenralMillsPQj- $\ln(\text{GeneralMillsPricing} \text{ GeneralMillsUnitSales} - 1)$ stargazer(GenralMillsPQ)
GenralMillsPWeekj- $\ln(\text{GeneralMillsPricing} \text{ GeneralMillsWeek} - 1)$ stargazer(GenralMillsPWeek)
GenralMillsPPOWeekj- $\ln(\text{GeneralMillsPricePerOZ} \text{ GeneralMillsWeek} - 1)$ stargazer(GenralMillsPPOWeek)
GenralMillsQuantWeekj- $\ln(\text{GeneralMillsUnitSales} \text{ GeneralMillsWeek} - 1)$ stargazer(GenralMillsQuantWeek)
GenralMillsCalj- $\ln(\text{GeneralMillsPricing} \text{ GeneralMillschar1}_{calories} - 1)$ stargazer(GenralMillsCal) GenralMills
- $\ln(\text{GeneralMillsPricing} \text{ GeneralMillschar2}_{carb} - 1)$ stargazer(GenralMillsCarb) GenralMillsProt <
- $\ln(\text{GeneralMillsPricing} \text{ GeneralMillschar3}_{protein} - 1)$ stargazer(GenralMillsProt) GenralMillsBsize <
- $\ln(\text{GeneralMillsPricing} \text{ GeneralMillschar4}_{boxsize} - 1)$ stargazer(GenralMillsBsize) GenralMillsSugarChar <
- $\ln(\text{GeneralMillsPricing} \text{ GeneralMillschar5}_{sugar} - 1)$ stargazer(GenralMillsSugarChar) GenralMillsGasChar
- $\ln(\text{GeneralMillsPricing} \text{ GeneralMillsgas} - 1)$ stargazer(GenralMillsGasChar) GenralMillsCornCharj-
 $\ln(\text{GeneralMillsPricing} \text{ GeneralMillscorn} - 1)$ stargazer(GenralMillsCornChar) GenralMillsWhtCharj-
 $\ln(\text{GeneralMillsPricing} \text{ GeneralMillswheat} - 1)$ stargazer(GenralMillsWhtChar) GenralMillsSUGcostj-
 $\ln(\text{GeneralMillsPricing} \text{ GeneralMills sugar} - 1)$ stargazer(GenralMillsSUGcost) GenralMillsLABcostj-
 $\ln(\text{GeneralMillsPricing} \text{ GeneralMillsLaborCost} - 1)$ stargazer(GenralMillsLABcost)

KELLOGG SIMPLE LINEAR RE-

GRESSIONS ---

$Kellogg_{CoPQ} < -lm(Kellogg_{CoPricing} \ Kellogg_{CoUnitSales} - 1) \text{ stargazer}(Kellogg_{CoPQ})Kellogg_{CoPWeek} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{CoWeek} - 1) \text{ stargazer}(Kellogg_{CoPWeek})Kellogg_{CoPPOWeek} <$
 $-lm(Kellogg_{CoPricePerOZ} \ Kellogg_{CoWeek} - 1) \text{ stargazer}(Kellogg_{CoPPOWeek})Kellogg_{CoQuantWeek} <$
 $-lm(Kellogg_{CoUnitSales} \ Kellogg_{CoWeek} - 1) \text{ stargazer}(Kellogg_{CoQuantWeek})Kellogg_{CoCal} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Cochar1_{calories}} - 1) \text{ stargazer}(Kellogg_{CoCal})Kellogg_{CoCarb} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Cochar2_{carb}} - 1) \text{ stargazer}(Kellogg_{CoCarb})Kellogg_{CoProt} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Cochar3_{protein}} - 1) \text{ stargazer}(Kellogg_{CoProt})Kellogg_{CoBsize} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Cochar4_{boxsize}} - 1) \text{ stargazer}(Kellogg_{CoBsize})Kellogg_{CoSugarChar} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Cochar5_{sugar}} - 1) \text{ stargazer}(Kellogg_{CoSugarChar})Kellogg_{CoGasChar} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Cogas} - 1) \text{ stargazer}(Kellogg_{CoGasChar})Kellogg_{CoCornChar} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Cocorn} - 1) \text{ stargazer}(Kellogg_{CoCornChar})Kellogg_{CoWhtChar} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Coweat} - 1) \text{ stargazer}(Kellogg_{CoWhtChar})Kellogg_{CoSUGcost} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{Cosugar} - 1) \text{ stargazer}(Kellogg_{CoSUGcost})Kellogg_{CoLABcost} <$
 $-lm(Kellogg_{CoPricing} \ Kellogg_{CoLaborCost} - 1) \text{ stargazer}(Kellogg_{CoLABcost})$

 PRIVATE LABEL SIMPLE LINEAR

REGRESSIONS ---

$Private_{LabelPQ} < -lm(Private_{LabelPricing} \ Private_{LabelUnitSales} - 1) \text{ stargazer}(Private_{LabelPQ})Private_{LabelPQ} <$
 $-lm(Private_{LabelPricing} \ Private_{LabelWeek} - 1) \text{ stargazer}(Private_{LabelPWeek})Private_{LabelPPOWeek} <$
 $-lm(Private_{LabelPricePerOZ} \ Private_{LabelWeek} - 1) \text{ stargazer}(Private_{LabelPPOWeek})Private_{LabelQuantWeek} <$
 $-lm(Private_{LabelUnitSales} \ Private_{LabelWeek} - 1) \text{ stargazer}(Private_{LabelQuantWeek})Private_{LabelCal} <$
 $-lm(Private_{LabelPricing} \ Private_{Labelchar1_{calories}} - 1) \text{ stargazer}(Private_{LabelCal})Private_{LabelCarb} <$
 $-lm(Private_{LabelPricing} \ Private_{Labelchar2_{carb}} - 1) \text{ stargazer}(Private_{LabelCarb})Private_{LabelProt} <$

$-\ln(\text{PrivateLabelPricing } \text{PrivateLabelchar3}_{protein} - 1) \text{stargazer}(\text{PrivateLabelProt}) \text{PrivateLabelBsize} <$
 $-\ln(\text{PrivateLabelPricing } \text{PrivateLabelchar4}_{boxsize} - 1) \text{stargazer}(\text{PrivateLabelBsize}) \text{PrivateLabelSugarChar} <$
 $-\ln(\text{PrivateLabelPricing } \text{PrivateLabelchar5}_{sugar} - 1) \text{stargazer}(\text{PrivateLabelSugarChar}) \text{PrivateLabelGasChar} <$
 $-\ln(\text{PrivateLabelPricing } \text{PrivateLabelgas} - 1) \text{stargazer}(\text{PrivateLabelGasChar}) \text{PrivateLabelCornChar} <$
 $-\ln(\text{PrivateLabelPricing } \text{PrivateLabelcorn} - 1) \text{stargazer}(\text{PrivateLabelCornChar}) \text{PrivateLabelWhtChar} <$
 $-\ln(\text{PrivateLabelPricing } \text{PrivateLabelwheat} - 1) \text{stargazer}(\text{PrivateLabelWhtChar}) \text{PrivateLabelSUGcost} <$
 $-\ln(\text{PrivateLabelPricing } \text{PrivateLabelsugar} - 1) \text{stargazer}(\text{PrivateLabelSUGcost}) \text{PrivateLabelLABcost} <$
 $-\ln(\text{PrivateLabelPricing } \text{PrivateLabelLaborCost} - 1) \text{stargazer}(\text{PrivateLabelLABcost})$

_____ PEPSICO SIMPLE LINEAR REGRES-
 SIONS _____

$\text{PepsiCoPQ}_i - \ln(\text{PepsiCoPricing } \text{PepsiCoUnitSales} - 1) \text{stargazer}(\text{PepsiCoPQ}) \text{PepsiCoPWeek}_i -$
 $\ln(\text{PepsiCoPricing } \text{PepsiCoWeek} - 1) \text{stargazer}(\text{PepsiCoPWeek}) \text{PepsiCoPPOWeek}_i - \ln(\text{PepsiCoPricePerOZ } \text{PepsiCoWeek} - 1) \text{stargazer}(\text{PepsiCoPPOWeek}) \text{PepsiCoQuantWeek}_i - \ln(\text{PepsiCoUnitSales } \text{PepsiCoWeek} - 1) \text{stargazer}(\text{PepsiCoWeek})$
 $\text{PepsiCoCal}_i - \ln(\text{PepsiCoPricing } \text{PepsiCochar1}_{calories} - 1) \text{stargazer}(\text{PepsiCoCal}) \text{PepsiCoCarb} <$
 $-\ln(\text{PepsiCoPricing } \text{PepsiCochar2}_{carb} - 1) \text{stargazer}(\text{PepsiCoCarb}) \text{PepsiCoProt} < -\ln(\text{PepsiCoPricing } \text{PepsiCochar3}_{protein} - 1) \text{stargazer}(\text{PepsiCoProt}) \text{PepsiCoBsize} <$
 $-\ln(\text{PepsiCoPricing } \text{PepsiCochar4}_{boxsize} - 1) \text{stargazer}(\text{PepsiCoBsize}) \text{PepsiCoSugarChar} < -\ln(\text{PepsiCoPricing } \text{PepsiCochar5}_{sugar} - 1) \text{stargazer}(\text{PepsiCoSugarChar})$
 $-\ln(\text{PepsiCoPricing } \text{PepsiCogas} - 1) \text{stargazer}(\text{PepsiCoGasChar}) \text{PepsiCoCornChar} < -\ln(\text{PepsiCoPricing } \text{PepsiCocorn} - 1) \text{stargazer}(\text{PepsiCoCornChar}) \text{PepsiCoWhtChar} <$
 $-\ln(\text{PepsiCoPricing } \text{PepsiCowheat} - 1) \text{stargazer}(\text{PepsiCoWhtChar}) \text{PepsiCoSUGcost} < -\ln(\text{PepsiCoPricing } \text{PepsiCosugar} - 1) \text{stargazer}(\text{PepsiCoSUGcost})$
 $-\ln(\text{PepsiCoPricing } \text{PepsiCoLaborCost} - 1) \text{stargazer}(\text{PepsiCoLABcost})$

_____ GENERAL MILLS CORRELATION
 TESTS _____

cor.test(General_{Mills}Pricing, General_{Mills}Week, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}UnitSales, alternative = "two.sided") cor.test(General_{Mills}PricePerOZ, General_{Mills}Week, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}schar1_calories, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}schar2_carb, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}schar3_protein, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}schar4_boxsize, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}schar5_sugar, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}sgas, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}scorn, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}wheat, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}sugar, alternative = "two.sided") cor.test(General_{Mills}Pricing, General_{Mills}LaborCost, alternative = "two.sided") cor.test(General_{Mills}UnitSales, General_{Mills}Week, alternative = "two.sided")

KELLOGG CORRELATION TESTS

cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}Week, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}UnitSales, alternative = "two.sided") cor.test(Kellogg_{Co}PricePerOZ, Kellogg_{Co}Week, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}char1_calories, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}char2_carb, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}char3_protein, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}char4_boxsize, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}char5_sugar, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}gas, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}corn, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}wheat, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}sugar, alternative = "two.sided") cor.test(Kellogg_{Co}Pricing, Kellogg_{Co}LaborCost, alternative = "two.sided") cor.test(Kellogg_{Co}UnitSales, Kellogg_{Co}Week, alternative = "two.sided")

PRIVATE LABEL CORRELATION TESTS

cor.test(*PrivateLabelPricing*, *PrivateLabelWeek*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelUnitSales*, alternative = "two.sided") cor.test(*PrivateLabelPricePerOZ*, *PrivateLabelWeek*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelchar1_calories*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelchar2_carb*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelchar3_protein*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelchar4_boxsize*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelchar5_sugar*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelgas*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelcorn*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelwheat*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelsugar*, alternative = "two.sided") cor.test(*PrivateLabelPricing*, *PrivateLabelLaborCost*, alternative = "two.sided") cor.test(*PrivateLabelUnitSales*, *PrivateLabelWeek*, alternative = "two.sided")

PEPSI CO CORRELATION TESTS

cor.test(*PepsiCoPricing*, *PepsiCoWeek*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCoUnitSales*, alternative = "two.sided") cor.test(*PepsiCoPricePerOZ*, *PepsiCoWeek*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCochar1_calories*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCochar2_carb*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCochar3_protein*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCochar4_boxsize*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCochar5_sugar*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCogas*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCocorn*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCowheat*, alternative = "two.sided") cor.test(*PepsiCoPricing*, *PepsiCosugar*, alternative =

```
"two.sided")cor.test(PepsiCoPricing, PepsiCoLaborCost, alternative = "two.sided")cor.test(PepsiCoUnitSales,
PepsiCoWeek, alternative = "two.sided")
```

6. MAKING ONE CEREAL DATA
FRAME OF JUST THE 4 PARENT COMPANIES

```
CerealPARENT <- NACerealPARENT[CerealAttrParentLevel == "PRIVATELABEL"] <- 1CerealPARENT[
"CerealAttrParentLevel" == "KELLOGGCO"] <- 1CerealPARENT[CerealAttrParentLevel == "PEPSICOINC"] <- 1CerealPARENT[Cer
"CerealAttrParentLevel" == "GENERALMILLS"] <- 1
```

```
CEREAL_i-filter(Cereal, PARENT == "1")
```

```
dim(CEREAL)
```

DATA FRAMES OF THE PRICING
DETERMINANTS AND CEREAL MAIN TIBBLE

```
CEREALcal_i-cbind(CEREALPricing, CEREALchar1_calories)CEREALcal <- data.frame(CEREALcal)colnames(CEREALcal)
Price'colnames(CEREALcal)[2] = 'Calories'
```

```
CEREALcarbs_i-cbind(CEREALPricing, CEREALchar2_carbs)CEREALcarbs <- data.frame(CEREALcarbs)colnames(CEREALcarbs)
Price'colnames(CEREALcarbs)[2] = 'Carbohydrates'
```

```
CEREALprotein_i-cbind(CEREALPricing, CEREALchar3_protein)CEREALprotein <- data.frame(CEREALprotein)colnames(CEREALprotein)
Price'colnames(CEREALprotein)[2] = 'Protein'
```

```
CEREALsize_i-cbind(CEREALPricing, CEREALchar4_boxsize)CEREALsize <- data.frame(CEREALsize)colnames(CEREALsize)
Price'colnames(CEREALsize)[2] = 'BoxSize'
```

```
CEREALsugar_i-cbind(CEREALPricing, CEREALchar5_sugar)CEREALsugar <- data.frame(CEREALsugar)colnames(CEREALsugar)
Price'colnames(CEREALsugar)[2] = 'SugarContent'
```



```
CEREALcorn;-cbind(CEREALPricing,CEREALcorn) CEREALcorn;-data.frame(CEREALcorn)
colnames(CEREALcorn)[1] = 'Price' colnames(CEREALcorn)[2] = 'Corn Cost'
```

```
CEREALGas;-cbind(CEREALPricing,CEREALgas) CEREALGas;-data.frame(CEREALGas)
colnames(CEREALGas)[1] = 'Price' colnames(CEREALGas)[2] = 'Gas Cost'
```

```
CEREALwheat;-cbind(CEREALPricing,CEREALwheat) CEREALwheat;-data.frame(CEREALwheat)
colnames(CEREALwheat)[1] = 'Price' colnames(CEREALwheat)[2] = 'Wheat Cost'
```

```
CEREALSC;-cbind(CEREALPricing,CEREALSugar) CEREALSC;-data.frame(CEREALSC)
colnames(CEREALSC)[1] = 'Price' colnames(CEREALSC)[2] = 'Sugar Cost'
```

```
CEREALele;-cbind(CEREALPricing,CEREALelectricity) CEREALele;-data.frame(CEREALele)
colnames(CEREALele)[1] = 'Price' colnames(CEREALele)[2] = 'Electricity Cost'
```

```
CEREALlabor;-cbind(CEREALPricing,CEREALLaborCost) CEREALlabor;-data.frame(CEREALlabor)
colnames(CEREALlabor)[1] = 'Price' colnames(CEREALlabor)[2] = 'Labor Cost'
```

```
CEREALWeek;-cbind(CEREALPricing,CEREALWeek) CEREALWeek;-data.frame(CEREALWeek)
colnames(CEREALWeek)[1] = 'Price' colnames(CEREALWeek)[2] = 'Week'
```

```
CEREALPD;-cbind(CEREALPricing,CEREALProductDescription) CEREALPD;-data.frame(CEREALPD)
colnames(CEREALPD)[1] = 'Price' colnames(CEREALPD)[2] = 'Product Description'
```

_____ 7. MAKING PREDICTIVE MODELS _____

_____ MAKING P HAT WITH DUMMIES
FOR PRODUCT DESCRIPTION AND TIME _____

_____ P hat = price determined by costs and product characteristics

$$\text{Phat}_i = \ln(\text{CEREALPricing}_i \text{CEREALchar1}_{calories} + \text{CEREALchar2}_{carb} + \text{CEREALchar3}_{protein} + \text{CEREALchar4}_{boysize} + \text{CEREALchar5}_{ugar} + \text{CEREALcorn} + \text{CEREALgas} + \text{CEREALwheat} +$$

$CEREAL_{sugar} + CEREAL_{electricity} + CEREAL_{LaborCost} + CEREAL_{Week} + CEREAL_{ProductDescription}$)su

$CEREAL_{Phat} < -NACEREAL_{Phat}$; predict(Phat, data = CEREALPricing)

_____ MAKING THE MARKET SHARE

MODEL WITH DUMMIES FOR P.D. AND TIME _____

MODEL_j-lm(CEREAL_{Yit} CEREALchar1_calories + CEREALchar2_carb + CEREALchar3_pprotein +
CEREALchar4_boxsize + CEREALchar5_sugar + CEREALPhat + CEREALWeek + CEREALProductDescription)
stargazer(MODEL)

print(MODEL) Coefficients: (Intercept) Cerealchar1_caloriesCerealchar2_carbCerealchar3_protein –
4.13624 – 0.002920.02692 – 0.07762Cerealchar4_boxsizeCerealchar5_sugarCerealPricing 0.03211
0.04530 -1.30963