**Fabric Softener Project**

**Processing the data:**

d1pur\_data <- read.delim("d1pur.DAT")

d1pur\_data

d1pur\_data\_HHID <- as.integer(substr(d1pur\_data[,1], 2, 5))

d1pur\_data\_HHID

d1pur\_data\_IRIWeek <- as.integer(substr(d1pur\_data[,1], 7, 9))

d1pur\_data\_IRIWeek

d1pur\_data\_store\_id <- as.integer(substr(d1pur\_data[,1], 10, 12))

d1pur\_data\_store\_id

d1pur\_data\_sku\_id <- as.integer(substr(d1pur\_data[,1], 13, 15))

d1pur\_data\_sku\_id

d1pur <- data.frame(d1pur\_data\_HHID, d1pur\_data\_IRIWeek, d1pur\_data\_store\_id, d1pur\_data\_sku\_id)

arsp\_data <- read.delim("arsp.DAT")

arsp\_data

arsp\_data\_sku\_id <- as.integer(substr(arsp\_data[,1], 3, 4))

arsp\_data\_sku\_id

arsp\_data\_store\_id <- as.integer(substr(arsp\_data[,1], 6, 7))

arsp\_data\_store\_id

arsp\_data\_price <- as.double(substr(arsp\_data[,1], 9, 13))

arsp\_data\_price

arsp = data.frame(arsp\_data\_sku\_id, arsp\_data\_store\_id, arsp\_data\_price)

brs\_info\_data <- read.delim("brsinfo.DAT")

brs\_info\_data

brs\_info\_sku\_id <- as.integer(substr(brs\_info\_data[,1], 2, 4))

brs\_info\_sku\_id

brs\_info\_brand\_num <- as.integer(substr(brs\_info\_data[,1], 35, 36))

brs\_info\_brand\_num

brs\_info\_form\_num <- as.integer(substr(brs\_info\_data[,1], 38, 38))

brs\_info\_form\_num

brs\_info\_formula2\_num <- as.integer(substr(brs\_info\_data[,1], 40, 40))

brs\_info\_formula2\_num

brs\_info\_size\_num <- as.integer(substr(brs\_info\_data[,1], 42, 42))

brs\_info\_size\_num

brs\_info = data.frame(brs\_info\_sku\_id, brs\_info\_brand\_num, brs\_info\_form\_num, brs\_info\_formula2\_num, brs\_info\_size\_num)

**Part A: Visualizations of Temporal and Household-Level Variation in SKU choices**

**Household Variation**

I thought that it would be interesting to measure the variety in the products bought by each household. In order to accomplish this I made use of the sqldf package, a package which allows r to process sql queries.

install.packages("sqldf")

require(sqldf)

I started by measuring the number of purchases made by each household:

purchases\_per\_household <- sqldf('select count(\*) from d1pur group by d1pur\_data\_HHID')

purchases\_per\_household <- as.numeric(purchases\_per\_household[,1])

summary(purchases\_per\_household)

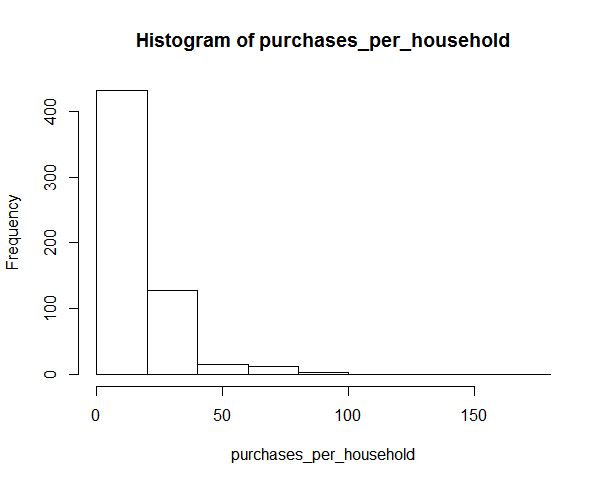
sd(purchases\_per\_household)

hist(purchases\_per\_household)

The standard deviation of purchases per household was 16.4. Here is the summary data:

Min. 1st Qu. Median Mean 3rd Qu. Max.

1.00 6.00 12.00 16.46 21.75 175.00



This data however is not particularly useful, as it only features houses that have bought at least one product. A slightly more useful vector would be the number of *different* products bought by each household (i.e. if a household bought the same product for every purchase the corresponding value would be 1.)

distinct\_purchases\_per\_household <- sqldf('select count(distinct d1pur\_data\_sku\_id) from d1pur group by d1pur\_data\_HHID')

distinct\_purchases\_per\_household <- as.numeric(distinct\_purchases\_per\_household[,1])

summary(distinct\_purchases\_per\_household)

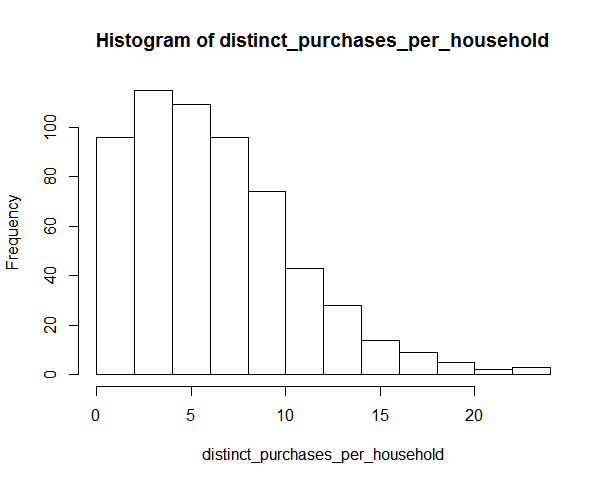
sd(distinct\_purchases\_per\_household)

hist(distinct\_purchases\_per\_household)

The standard deviation of distinct purchases per household is 4.3. Here is the summary data and the corresponding graph.

Min. 1st Qu. Median Mean 3rd Qu. Max.

1.000 3.000 6.000 6.788 9.000 24.000



The ideal measure for the household purchase variety, however, is the quotient of distinct\_purchases\_per\_household and purchases per household:

household\_purchase\_variation <- distinct\_purchases\_per\_household / purchases\_per\_household

summary(household\_purchase\_variation)

sd(household\_purchase\_variation)

hist(household\_purchase\_variation)

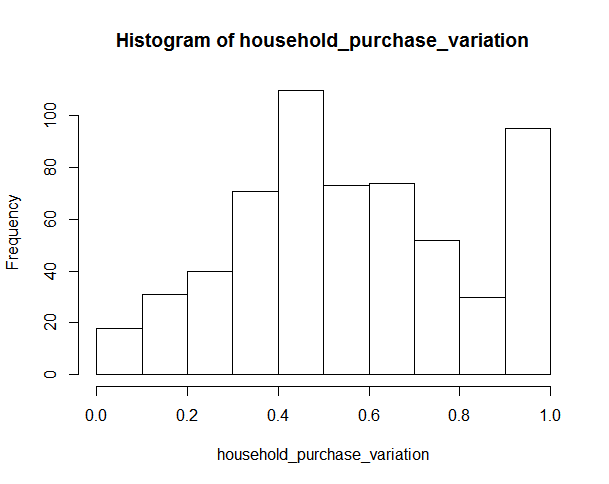
The summary of the household purchase variation is as follows:

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.03333 0.38750 0.55560 0.57630 0.75000 1.00000

Standard Deviation: 0.261

Graphical Representation:



Of all the graphs I’ve seen so far this by far the most interesting. Values of household purchase variation closer to zero entail that the household in question has a tendency to buy the same product many times. Whereas values of household purchase variation closer to one indicate a low tendency to buy the same product multiple times. It is interesting that the graph is similar in shape to a normal bell curve with the exception of the large spike I see as I move very close to 1. This means that a large proportion of households never bought the same product twice.

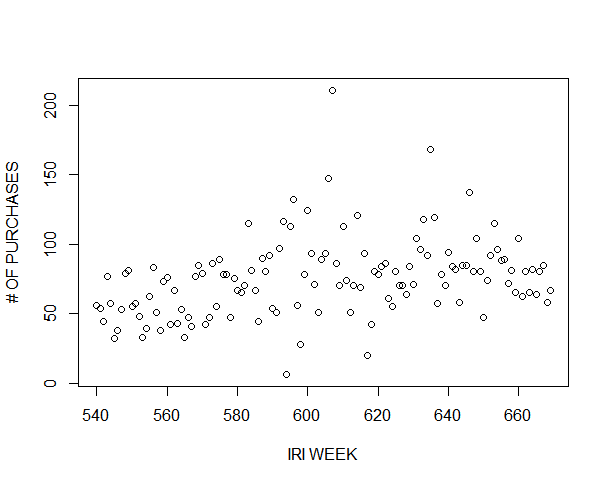
**Temporal Variation**

As for temporal variation, I opted first to determine the number of products sold each week:

weekly\_purchases <- sqldf('select d1pur\_data\_IRIWeek, count(\*) from d1pur group by d1pur\_data\_IRIWeek order by d1pur\_data\_IRIWeek asc')

Then I plotted the data:

plot(weekly\_purchases[,1], weekly\_purchases[,2], xlab = "IRI WEEK", ylab = "# OF PURCHASES")



After plotting the data I ran a linear regression on the data:

lm(weekly\_purchases[,2] ~ weekly\_purchases[,1])

Call:

lm(formula = weekly\_purchases[, 2] ~ weekly\_purchases[, 1])

Coefficients:

(Intercept) weekly\_purchases[, 1]

-76.3858 0.2508

summary(lm(weekly\_purchases[,2] ~ weekly\_purchases[,1]))

Call:

lm(formula = weekly\_purchases[, 2] ~ weekly\_purchases[, 1])

Residuals:

Min 1Q Median 3Q Max

-66.597 -17.533 -3.055 12.299 135.142

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -76.3858 38.0339 -2.008 0.046709 \*

weekly\_purchases[, 1] 0.2508 0.0628 3.994 0.000109 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 26.87 on 128 degrees of freedom

Multiple R-squared: 0.1108, Adjusted R-squared: 0.1039

F-statistic: 15.95 on 1 and 128 DF, p-value: 0.0001089

From this I can ascertain that there is a decent chance of the price of a given product increasing over time.

**Part B: Evaluation of the importance of pricing and promotions**

In this part, I try to figure out the effect of pricing and promotions on sale of a SKU i.e. I map the number of SKUs sold to their prices and the amount which was spent on their promotions.

First I determine the total number of each type SKUs present in our dataset:

sku\_purchases <- sqldf('select d1pur\_data\_sku\_id, count(\*) as purchases from d1pur group by d1pur\_data\_sku\_id order by d1pur\_data\_sku\_id asc')

Then I join this data with the prices of the SKUs:

sku\_price\_purchases <- sqldf('select d1pur\_data\_sku\_id, purchases, arsp\_data\_price from sku\_purchases inner join arsp on d1pur\_data\_sku\_id = arsp\_data\_sku\_id')

The attributes which I have taken for modelling the number of SKUs sold is the SKU price. I determine the total number of a particular SKU(id) sold depending upon the price

lm(sku\_price\_purchases$purchases~sku\_price\_purchases$arsp\_data\_price)

Call:

lm(formula = sku\_price\_purchases$purchases ~ sku\_price\_purchases$arsp\_data\_price)

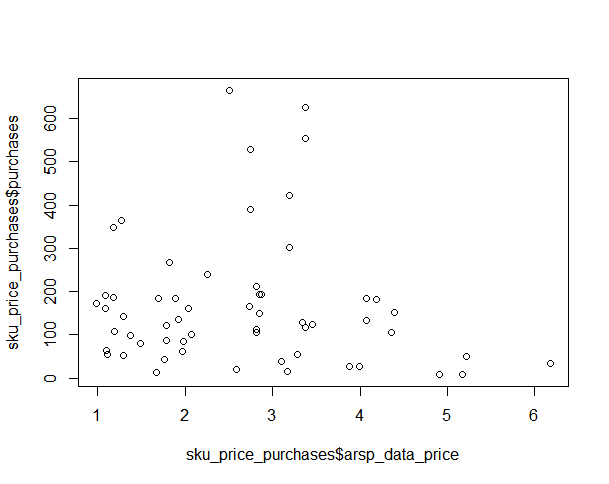
Coefficients:

(Intercept) sku\_price\_purchases$arsp\_data\_price

195.46 -10.39

The plot of the above model suggests that the SKUs within the price range of 2-3 are sold the most and the plot is, for obvious reasons, skewed towards the lower price ranges

plot(sku\_purchases$arsp\_data\_price, sku\_purchases$purchases)



The multiple R-squared of this linear regression was exceedingly low. It seems that a better way to go about this would be to create some dummy variables placing the price into groups:

dummy\_price\_L2 <- as.integer(sku\_price\_purchases$arsp\_data\_price < 2)

dummy\_price\_2\_4 <- as.integer(sku\_price\_purchases$arsp\_data\_price > 2 & sku\_price\_purchases$arsp\_data\_price < 4)

There are 3 groups, one for price < 2, one for 2 < price < 4, and one for price > 6. Now let me run the linear regression with these new dummy variables:

lm(sku\_price\_purchases$purchases ~ dummy\_price\_L2 + dummy\_price\_2\_4)

Call:

lm(formula = sku\_price\_purchases$purchases ~ dummy\_price\_L2 +

dummy\_price\_2\_4)

Coefficients:

(Intercept) dummy\_price\_L2 dummy\_price\_2\_4

95.33 43.84 123.05

The multiple r-squared for this regression is 0.1018, which is much better than our previous one which was less than 0.01.

**Part C: Evaluations of the Importance of Different Attributes for Each SKU**

In this part I attempt to ascertain the effects of different attributes of each SKU on the number of SKUs sold. Therefore I should start by determining how many purchases there were of each SKU:

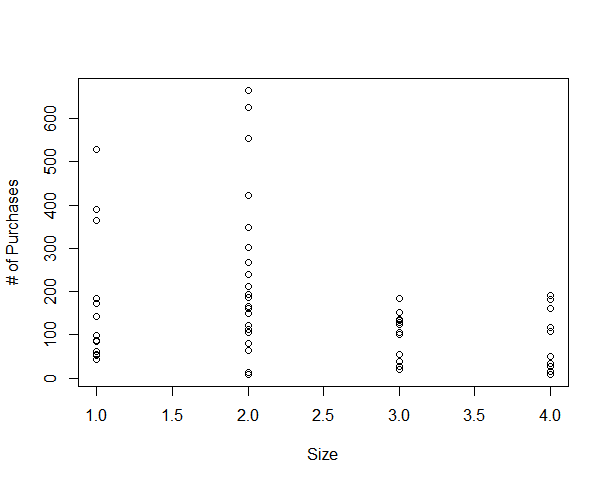
sku\_purchases <- sqldf('select d1pur\_data\_sku\_id, count(\*) as purchases from d1pur group by d1pur\_data\_sku\_id order by d1pur\_data\_sku\_id asc')

The next step is combining the sku\_purchases data with the brs\_info table:

brs\_purchases <- sqldf('select d1pur\_data\_sku\_id, purchases, brs\_info\_brand\_num, brs\_info\_form\_num, brs\_info\_formula2\_num, brs\_info\_size\_num from sku\_purchases inner join brs\_info on d1pur\_data\_sku\_id = brs\_info\_sku\_id')

The first attribute I will analyze will be the size number. This attribute is unique in that it has more of a continuity to it than the others (i.e. the larger the size number, the larger the size.) I started by visualizing the relationship between the size and number of purchases:

plot(brs\_purchases$brs\_info\_size\_num, brs\_purchases$purchases, xlab = “Size”, ylab = “# of Purchases”)



One can discern by looking at this plot that the size variable alone will not be sufficient to put into the linear regression formula. Since the distribution of values for sizes 3 and 4 are relatively similar, I will content ourselves with creating dummy variables for size 1 and 2.

dummy\_size1 <- as.integer(brs\_purchases$brs\_info\_size\_num == 1)

dummy\_size2 <- as.integer(brs\_purchases$brs\_info\_size\_num == 2)

Next I will make dummy variables for the form and formula2 attributes:

dummy\_form2 = as.integer(brs\_purchases$brs\_info\_form\_num == 2)

dummy\_form3 = as.integer(brs\_purchases$brs\_info\_form\_num == 3)

dummy\_form4 = as.integer(brs\_purchases$brs\_info\_form\_num == 4)

dummy\_formula2\_2 = as.integer(brs\_purchases$brs\_info\_formula2\_num == 2)

dummy\_formula2\_3 = as.integer(brs\_purchases$brs\_info\_formula2\_num == 3)

dummy\_formula2\_4 = as.integer(brs\_purchases$brs\_info\_formula2\_num == 4)

Now I run a linear regression on these variables:

summary(lm(brs\_purchases$purchases ~ dummy\_form2 + dummy\_form3 + dummy\_form4 + dummy\_formula2\_2 + dummy\_formula2\_3 + dummy\_formula2\_4 + dummy\_size1 + dummy\_size2))

Call:

lm(formula = brs\_purchases$purchases ~ dummy\_form2 + dummy\_form3 +

dummy\_form4 + dummy\_formula2\_2 + dummy\_formula2\_3 + dummy\_formula2\_4 +

dummy\_size1 + dummy\_size2)

Residuals:

Min 1Q Median 3Q Max

-309.86 -89.19 -13.45 45.02 384.99

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 116.86 54.15 2.158 0.03584 \*

dummy\_form2 37.85 81.09 0.467 0.64271

dummy\_form3 -111.48 77.72 -1.434 0.15783

dummy\_form4 -52.97 44.87 -1.180 0.24356

dummy\_formula2\_2 13.18 51.56 0.256 0.79931

dummy\_formula2\_3 -28.40 100.51 -0.283 0.77870

dummy\_formula2\_4 -19.88 99.84 -0.199 0.84302

dummy\_size1 60.55 52.24 1.159 0.25204

dummy\_size2 150.97 44.49 3.393 0.00138 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

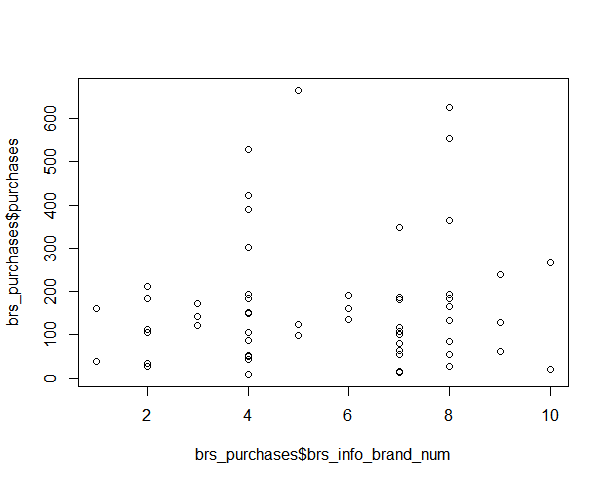
Residual standard error: 142.6 on 49 degrees of freedom

Multiple R-squared: 0.2358, Adjusted R-squared: 0.111

F-statistic: 1.889 on 8 and 49 DF, p-value: 0.08307

The Multiple R-squared for this linear regression is still far from ideal. Let’s take a look at the brand attribute.

plot(brs\_purchases$brs\_info\_brand\_num, brs\_purchases$purchases)



We will make dummy variables for brand 4 and 8 as they look the most distinctive, and adding 9 more dummy variables would wreak havoc on our adjusted r-squared.

dummy\_brand\_4 = as.integer(brs\_info\_brand\_num == 4)

dummy\_brand\_8 = as.integer(brs\_info\_brand\_num == 8)

Let’s also add the dummy price variables from part b:

Call:

lm(formula = brs\_purchases$purchases ~ dummy\_form2 + dummy\_form3 +

dummy\_form4 + dummy\_formula2\_2 + dummy\_formula2\_3 + dummy\_formula2\_4 +

dummy\_size1 + dummy\_size2 + dummy\_price\_L2 + dummy\_price\_2\_4 +

dummy\_brand\_4 + dummy\_brand\_8)

Coefficients:

(Intercept) dummy\_form2 dummy\_form3 dummy\_form4

58.769 20.473 -47.037 -63.858

dummy\_formula2\_2 dummy\_formula2\_3 dummy\_formula2\_4 dummy\_size1

28.375 -16.168 42.422 111.079

dummy\_size2 dummy\_price\_L2 dummy\_price\_2\_4 dummy\_brand\_4

142.361 -17.925 73.127 -5.007

dummy\_brand\_8

65.960

The multiple r-squared for this model was 0.3272. By far the best r-squared I’ve seen. This model will be a good predictor for the number of purchases of a given product.