Pricing Solutions - Case Study

Overview of the Data

General Metrics

```
apply_to_list(datasets, function(x, n) x %>% summarise(
  `Type`=n,
  `Number of Stores`=max(store, na.rm=T),
  `Number of Products`=length(unique(mstrprodid)),
  `Start Period`=min(year), `End Period`=max(year),
)) %>% arrange(desc(Type))
```

Type	Number of Stores	Number of Products	Start Period	End Period
non_loyalty	511	253	2012	2015
loyalty	511	248	2012	2015
all	511	255	2012	2015

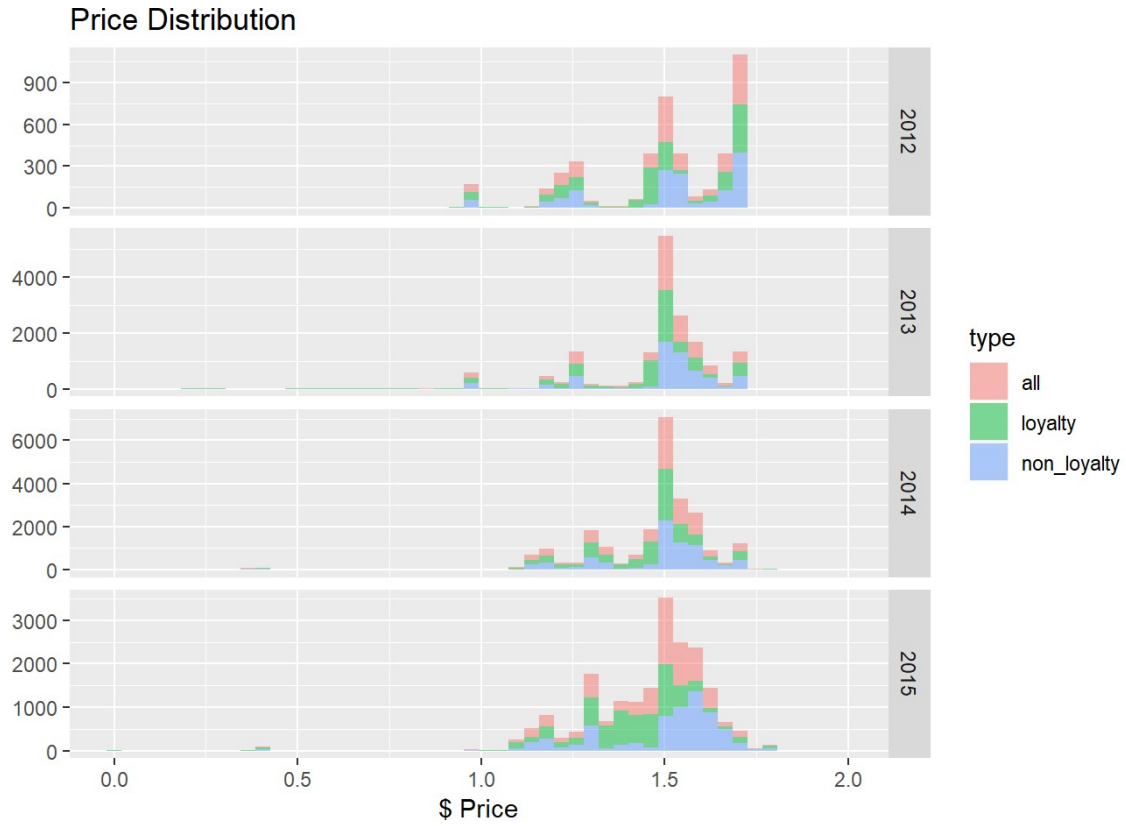
Summary of Price Data

```
psummary %>%
  group_by(year, type) %>%
  summarise(
    Min=min(value, na.rm = T),
    Mean=round(mean(value, na.rm = T), 2),
    Max=max(value, na.rm = T),
    `Na's`=sum(is.na(value)),
    N=n()
  )
```

year	type	Min	Mean	Max	Na's	N
2012	all	0.23	1.50	1.69	0	1447
2012	loyalty	0.12	1.48	1.69	0	1447
2012	non_loyalty	0.99	1.51	1.69	0	1447
2013	all	0.42	1.48	1.69	0	5570
2013	loyalty	0.19	1.45	1.69	0	5570
2013	non_loyalty	0.75	1.50	1.69	0	5570
2014	all	0.00	1.46	1.91	0	8050
2014	loyalty	0.00	1.44	1.99	0	8050
2014	non_loyalty	0.00	1.47	1.99	0	8050
2015	all	0.00	1.46	1.89	0	6604
2015	loyalty	0.00	1.41	1.89	0	6604

year	type	Min	Mean	Max	Na's	N
2015	non_loyalty	0.00	1.49	1.89	0	6604

```
psummary %>%
  ggplot(aes(
    value,
    fill=type
  )) +
  geom_histogram(alpha=0.5, bins = 50) +
  facet_grid(year~., scales = "free_y")+
  labs(x="$ Price",y="")+
  ggtitle("Price Distribution")
```

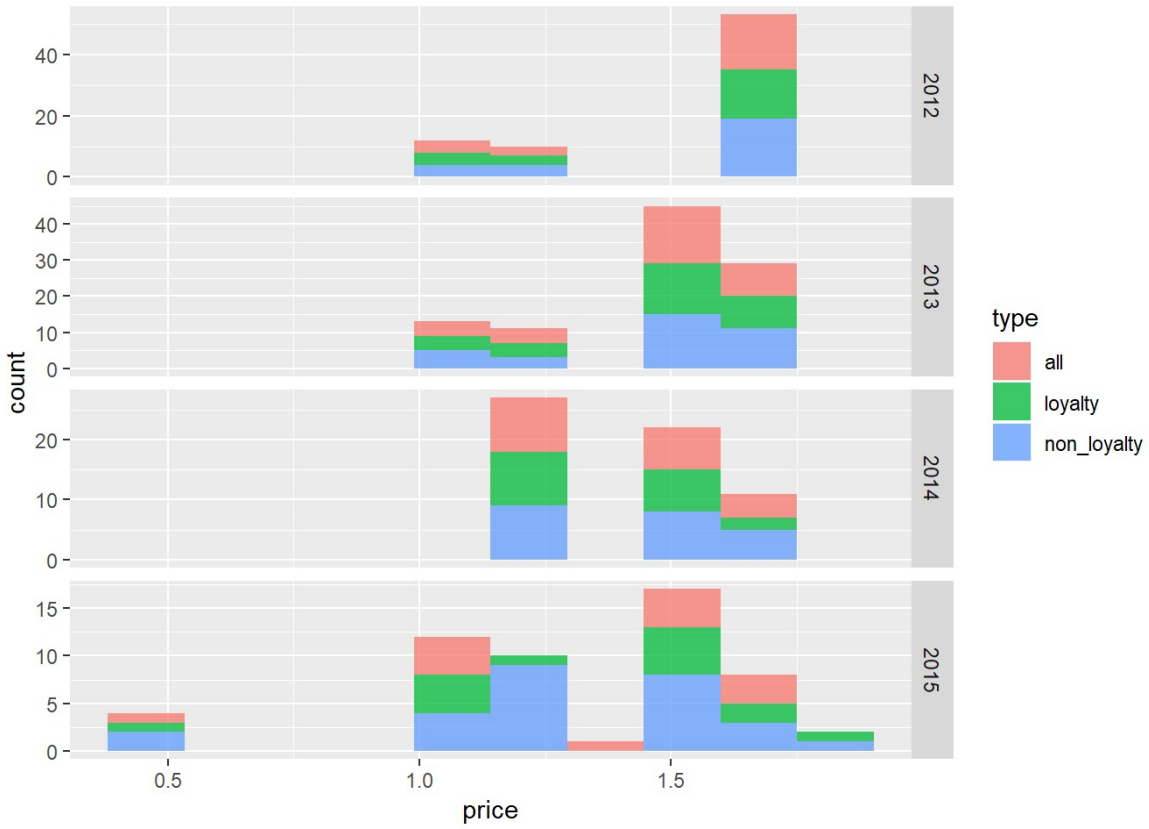



```
psummary %>%
  group_by(type, pid, year) %>%
  summarise(
    sd = replace_na(sd(value),0)
  ) %>%
  group_by(type, sd==0, year) %>%
  summarise(
    pids=length(unique(pid))
  ) %>% group_by(year, type) %>%
  spread(`sd == 0`, pids) %>%
  rename(`sd =/= 0`=`FALSE`, `sd==0`=`TRUE`)
```

type	year	sd =/= 0	sd==0
all	2012	83	25
all	2013	104	33

type	year	sd =/= 0	sd==0
all	2014	173	20
all	2015	194	13
loyalty	2012	85	23
loyalty	2013	106	31
loyalty	2014	175	18
loyalty	2015	193	14
non_loyalty	2012	81	27
non_loyalty	2013	103	34
non_loyalty	2014	171	22
non_loyalty	2015	180	27

```
psummary %>%
  group_by(type, pid, year) %>%
  summarise(
    price = round(mean(value),2),
    sd = replace_na(sd(value),0)
  ) %>%
  filter(sd==0) %>%
  ggplot() +
  geom_histogram(aes(
    price, fill=type
  ), bins=10, alpha=0.75) +
  facet_grid(year~., scales='free_y')
```



Summary of Quantity

```
qsummary %>%
  group_by(year, type) %>%
  summarise(
    Min=min(value, na.rm = T),
    Mean=round(mean(value, na.rm = T), 2),
    Max=max(value, na.rm = T),
    `Na's`=sum(is.na(value)),
    N=n()
  )
```

year	type	Min	Mean	Max	Na's	N
2012	all	2	5813.43	123485	0	1447
2012	loyalty	1	1998.03	38811	0	1447
2012	non_loyalty	1	3815.40	85128	0	1447
2013	all	2	6372.22	133418	0	5570
2013	loyalty	1	2079.61	37634	0	5570
2013	non_loyalty	1	4292.61	95784	0	5570
2014	all	0	4900.20	135707	0	8050
2014	loyalty	1	1642.58	40618	0	8050
2014	non_loyalty	-1	3257.62	95089	0	8050
2015	all	2	4707.53	134403	0	6604
2015	loyalty	1	1912.50	54670	0	6604
2015	non_loyalty	1	2795.03	80594	0	6604

The following had negative units sold

```
qsummary %>% filter(value<0)
```

year	week_num	pid	type	value
2014	45	40886	non_loyalty	-1

```
qsummary %>% filter(pid==40886) %>%
  group_by(type) %>%
  summarise(qty=sum(value,na.rm=T)) %>%
  spread(type, qty) %>%
  mutate(`CHECK=loyalty+non_loyalty`=loyalty+non_loyalty) %>%
  gather(type, qty) %>% arrange(type)
```

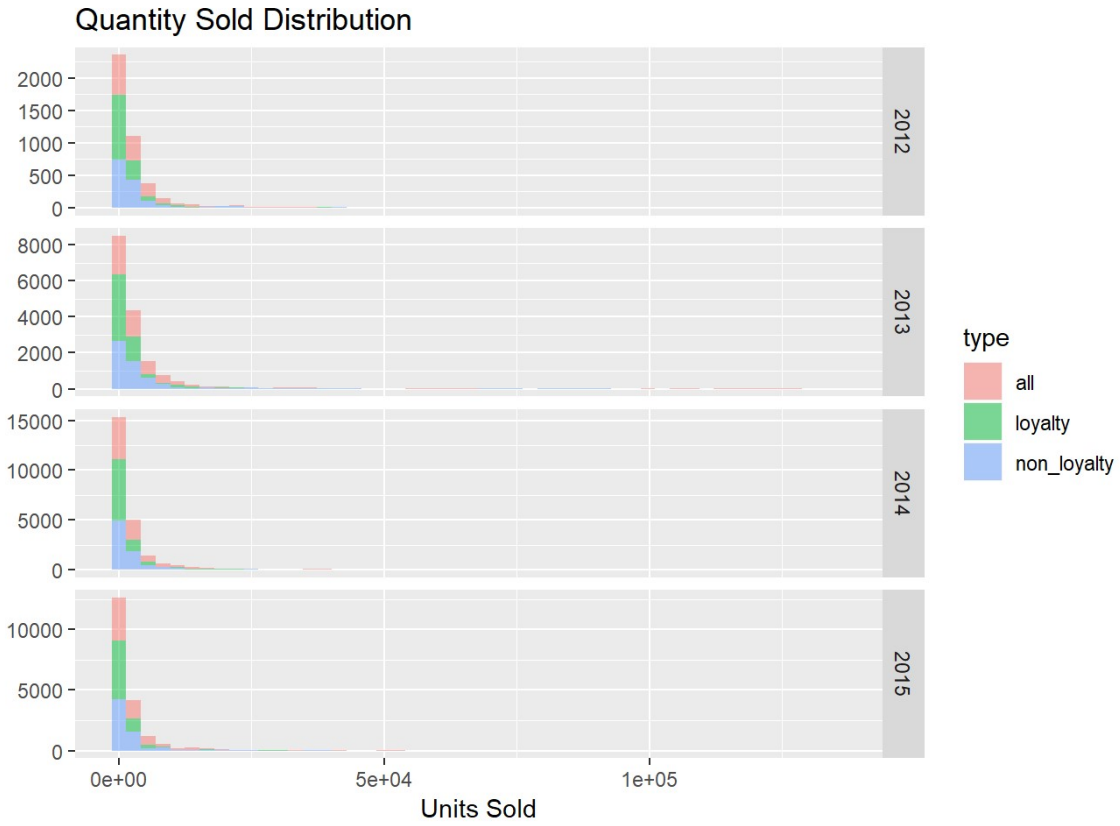
type	qty
all	15784
CHECK=loyalty+non_loyalty	15784
loyalty	4089

type	qty
non_loyalty	11695

```
datasets$all %>% filter(mstrprodid==40886) %>%
  group_by(mstrprodid, year) %>%
  summarise(
    Revenue = scales::dollar(sum(price*qty)),
    `Average Quantity Sold By Store` = round(avg(qty_bs)),
    `Average Price` = scales::dollar(avg(price)),
    `Total Quantity Sold` = scales::comma(sum(qty))
  )
```

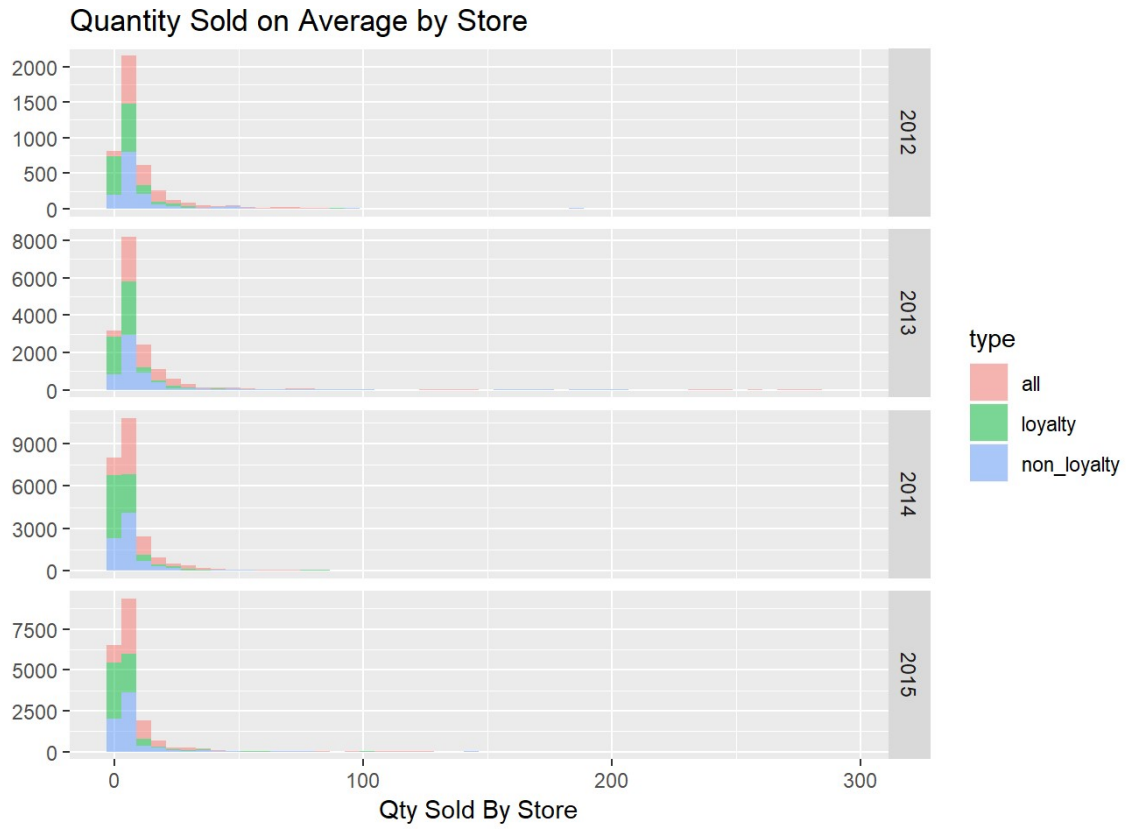
mstrprodid	year	Revenue	Average Quantity Sold By Store	Average Price	Total Quantity Sold
40886	2013	$312.65	5	$1.69	185
40886	2014	$99.71	3	$1.69	59
40886	2015	$24,329.07	4	$1.57	15,619

```
qsummary %>%
  ggplot(aes(
    value,
    fill=type
  )) +
  geom_histogram(alpha=0.5, bins = 50) +
  facet_grid(year~., scales = "free_y")+
  labs(x="Units Sold",y="")+
  ggtitle("Quantity Sold Distribution")
```



Average Quantity by Store

```
qsummary %>%
  ggplot(aes(
    value,
    fill=type
  )) +
  geom_histogram(alpha=0.5, bins = 50) +
  facet_grid(year~., scales = "free_y")+
  labs(x="Qty Sold By Store",y="")+
  ggtitle("Quantity Sold on Average by Store")
```




```
qsummary %>%
  group_by(year, type) %>%
  summarise(
    Min=min(value, na.rm = T),
    Mean=round(mean(value, na.rm = T), 2),
    Max=max(value, na.rm = T),
    `Na's`=sum(is.na(value)),
    N=n()
  )
```

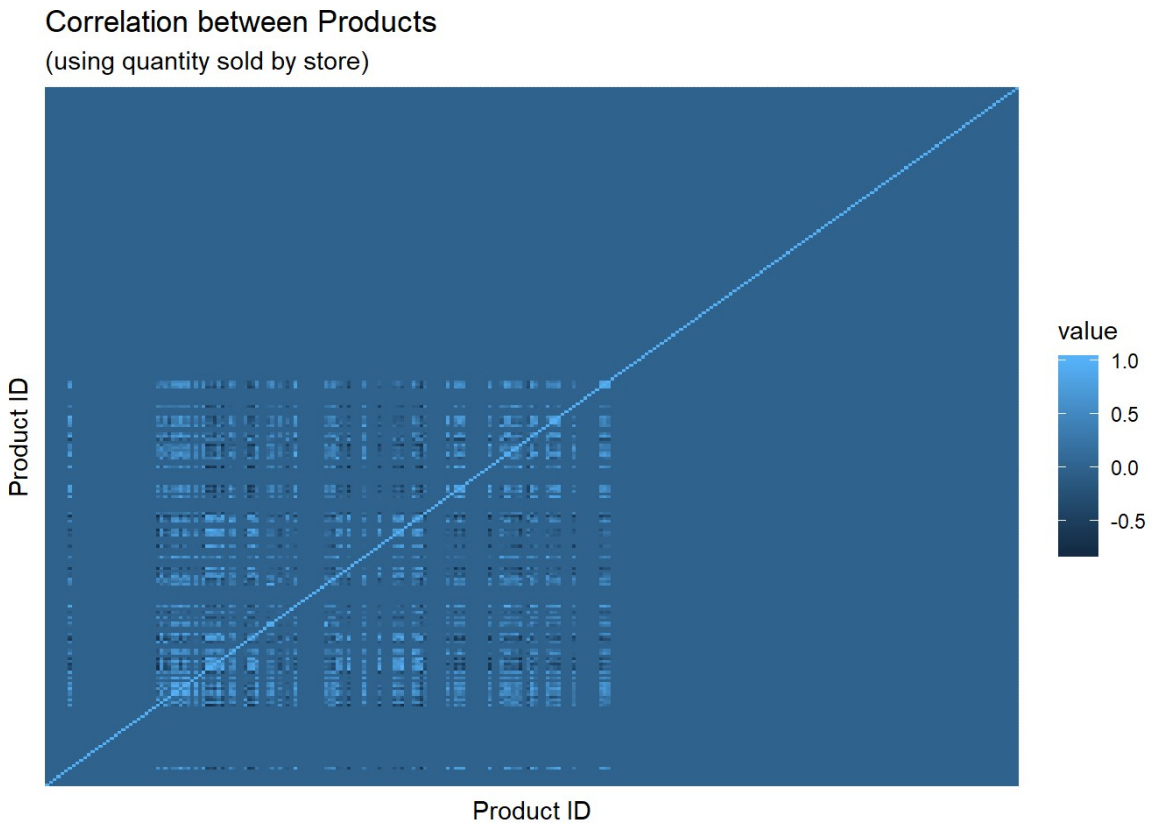
year	type	Min	Mean	Max	Na's	N
2012	all	1.0000000	17.95	287.17442	0	1447
2012	loyalty	1.0000000	6.68	89.84028	0	1447
2012	non_loyalty	0.3333333	11.90	197.97209	0	1447
2013	all	1.0000000	17.51	292.58333	0	5570

year	type	Min	Mean	Max	Na's	N
2013	loyalty	1.0000000	6.23	84.36281	0	5570
2013	non_loyalty	1.0000000	11.93	210.05263	0	5570
2014	all	0.0000000	13.31	283.90586	0	8050
2014	loyalty	1.0000000	5.09	84.97490	0	8050
2014	non_loyalty	-1.0000000	9.12	199.58403	0	8050
2015	all	1.0000000	12.35	270.48589	0	6604
2015	loyalty	1.0000000	5.55	108.61895	0	6604
2015	non_loyalty	1.0000000	7.69	161.86694	0	6604

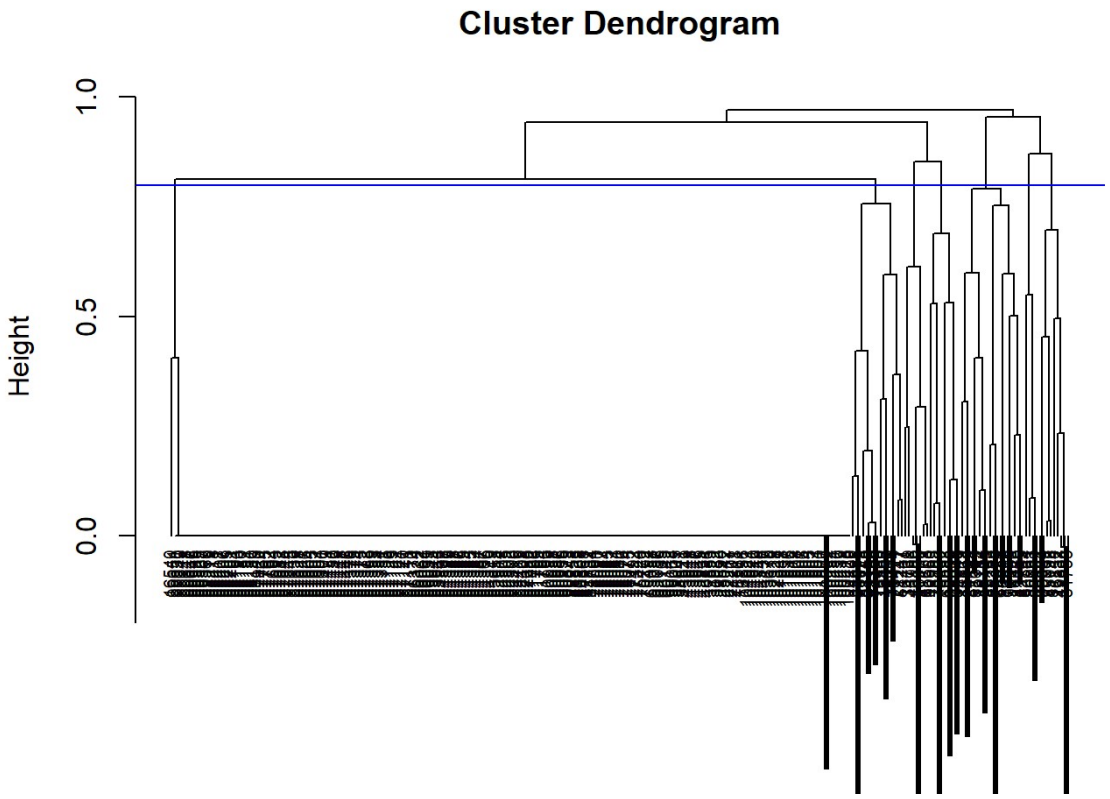
Clustering: Group Generation

Correlation Clustering

```
cor_table %>% ggplot(aes(x=var1,y=var2,fill=value)) + geom_tile() +
  theme(axis.text.x=element_blank(),axis.ticks.x=element_blank(),
        axis.text.y=element_blank(),axis.ticks.y=element_blank()) +
  ggtitle("Correlation between Products",
          "(using quantity sold by store)") +
  labs(x="Product ID",y="Product ID")
```



```
hclst = hclust(cor_dist_matrix, method = 'complete')
plot(hclst, hang = -1, cex = 0.6, xlab='', sub='')
abline(h=0.8, col=4)
```




```
cutree(hclst, 7) %>% {
  tibble(
    mstrprodid=names(.),
    cor_cluster=as.numeric(.)
  )
} %>% group_by(cor_cluster) %>%
  summarise(N=n())
```

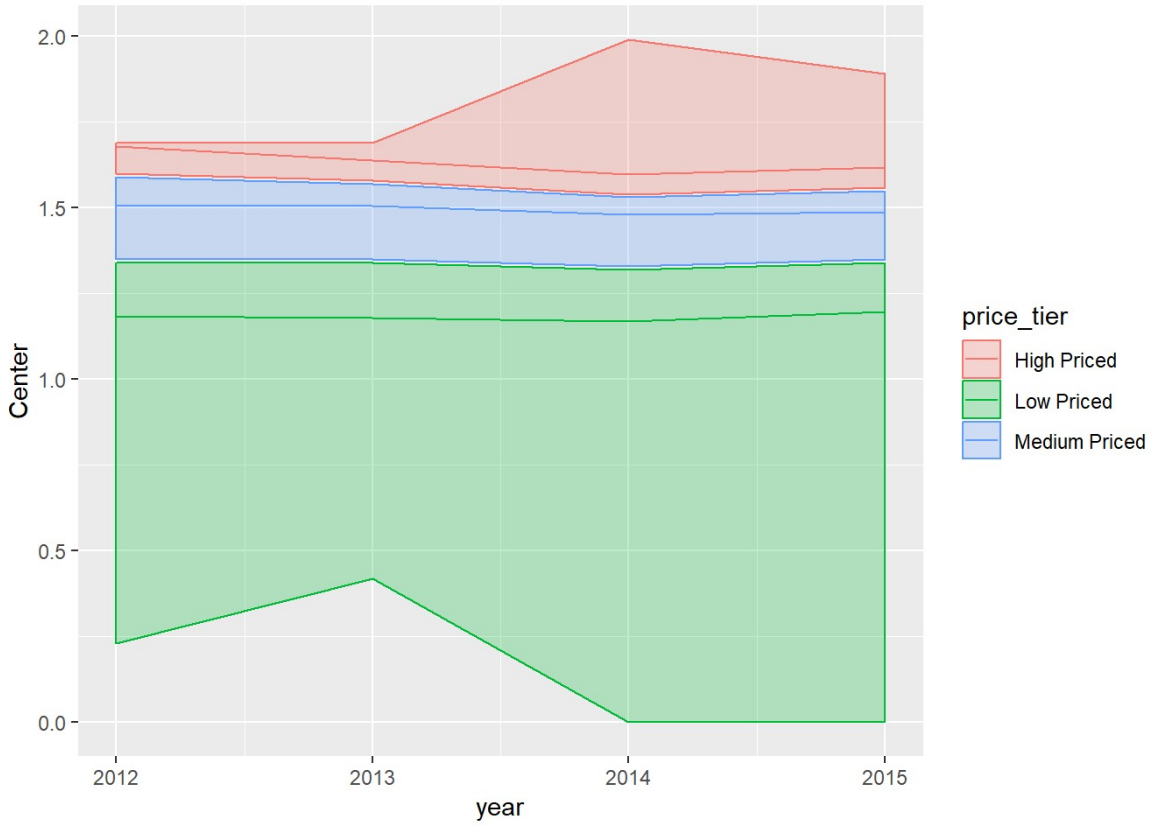
cor_cluster	N
1	193
2	7
3	15
4	18
5	9
6	9
7	4

```
datasets$all %>%
  group_by(cor_cluster_d) %>%
  summarise(
    `Revenue`=sum(price*qty),
    `Number of Products` = length(unique(mstrprodid))
  ) %>%
  mutate(
    `% of Revenue` = scales::percent(Revenue/sum(Revenue)),
    Revenue = scales::dollar(Revenue)
  )
```

cor_cluster_d	Revenue	Number of Products	% of Revenue
1	$34,812,634	193	20.3%
2	$24,447,862	7	14.3%
3	$24,796,277	15	14.5%
4	$27,346,686	18	16.0%
5	$36,126,531	9	21.1%
6	$19,616,166	9	11.5%
7	$4,074,432	4	2.4%

Kmeans Clustering

```
datasets$all %>%
  filter(!is.na(price)) %>%
  group_by(year) %>%
  mutate(
    cluster = kmeans(
      tibble(price), 3
    )$cluster
  ) %>%
  group_by(cluster, year) %>%
  summarise(
    N=n(), `Lower Limit`=min(price),
    `Center`=mean(price),
    `Upper Limit`=max(price)
  ) %>% arrange(year, `Lower Limit`) %>%
  group_by(year) %>%
  mutate(
    price_tier = c("Low Priced", "Medium Priced", "High Priced")
  ) -> price_tiers
price_tiers %>% ggplot(aes(
  x=year, color=price_tier, fill=price_tier
))+
  geom_ribbon(aes(ymax=`Upper Limit`, ymin=`Lower Limit`), alpha=0.25) +
  geom_line(aes(y=Center))
```




```
datasets$all %>%
  group_by(cluster_name) %>%
  summarise(
    `Revenue`=sum(price*qty),
    `Number of Products` = length(unique(mstrprodid))
  ) %>%
  mutate(
    `% of Revenue` = scales::percent(Revenue/sum(Revenue)),
    Revenue = scales::dollar(Revenue)
  )
```

cluster_name	Revenue	Number of Products	% of Revenue
High Priced	$33,969,825	144	19.8%
Low Priced	$10,705,333	139	6.3%
Medium Priced	$126,545,431	178	73.9%

Predictive Models

Model 1: Lagged Price and Quantity

```
joined_dataset %>% group_by(pid) %>%
  arrange(pid, year, week_num) %>%
  mutate(
    last_dqty_bs_all = lag(dqty_bs_all),
    last_price_all = lag(price_all)
  ) %>% ungroup() %>% {lm(
    dqty_bs_all~week_num*last_dqty_bs_all+last_price_all+price_all-1,
    data=.
  )} -> m1

broom::glance(m1)
```

	r.squared	adj.r.squared	sigma	statistic	p.value	df	logLik	AIC	BIC	deviance	df.residual
value	0.0240579	0.0238278	0.3351794	104.5593	0	5	-6909.739	13831.48	13879.25	2382.618	21208

```
broom::tidy(m1) %>% mutate(
  significant=(p.value<0.1)+(p.value<0.05)+(p.value<0.01)+(p.value<0.005)
)
```

term	estimate	std.error	statistic	p.value	significant
week_num	-0.0004893	0.0001538	-3.1803664	0.0014730	4
last_dqty_bs_all	-0.0007206	0.0134289	-0.0536597	0.9572068	0
last_price_all	0.9020981	0.0486432	18.5451977	0.0000000	4
price_all	-0.8761546	0.0487161	-17.9848938	0.0000000	4
week_num:last_dqty_bs_all	-0.0016337	0.0004944	-3.3045307	0.0009529	4

Model 2: Lagged Price and Quality, and clusters

```
joined_dataset %>% group_by(pid) %>%
  arrange(pid, year, week_num) %>%
  mutate(
    last_dqty_bs_all = lag(dqty_bs_all),
    last_price_all = lag(price_all),
    cor_cluster_all=as.character(cor_cluster_all),
  ) %>% ungroup() %>% {lm(
    dqty_bs_all~week_num+last_price_all+
        last_dqty_bs_all+last_price_all+price_all+
        price_tier_all+cor_cluster_all,
    data=.
  )} -> m2

broom::glance(m2)
```

	r.squared	adj.r.squared	sigma	statistic	p.value	df	logLik	AIC	BIC	deviance	df.residual
value	0.0230241	0.0224711	0.3345506	41.63459	0	13	-6865.901	13759.8	13871.27	2372.79	21200

```
broom::tidy(m2) %>% mutate(
  significant=(p.value<0.1)+(p.value<0.05)+(p.value<0.01)+(p.value<0.005)
)
```

term	estimate	std.error	statistic	p.value	significant
(Intercept)	0.1313463	0.0456009	2.880345	0.0039764	4
week_num	-0.0005927	0.0001572	-3.770646	0.0001633	4
last_price_all	0.9019427	0.0492542	18.311995	0.0000000	4
last_dqty_bs_all	-0.0436874	0.0066758	-6.544151	0.0000000	4
price_all	-0.9352148	0.0528943	-17.680814	0.0000000	4
price_tier_allLow Priced	-0.0428171	0.0136004	-3.148217	0.0016450	4
price_tier_allMedium Priced	-0.0276648	0.0063579	-4.351266	0.0000136	4
cor_cluster_all2	-0.0414315	0.0106873	-3.876688	0.0001062	4
cor_cluster_all3	-0.0408147	0.0077007	-5.300127	0.0000001	4
cor_cluster_all4	-0.0421669	0.0071743	-5.877503	0.0000000	4
cor_cluster_all5	-0.0408855	0.0096070	-4.255786	0.0000209	4
cor_cluster_all6	-0.0410463	0.0095575	-4.294679	0.0000176	4
cor_cluster_all7	-0.0415307	0.0139727	-2.972285	0.0029592	4