Foundations of Marketing Analytics: Module 1: Statistical Segmentation

Stefan Avey

2017-01-16

Contents

1 Load the Data

```
################
## Load data ##
################
purchases <- read.delim("data/purchases.txt", header = FALSE)

###################
## Preprocessing ##
###################
## Add column names
colnames(purchases) <- c("customer_id", "purchase_amount", "date_of_purchase")

## Convert date and add column for purchase year
purchases <- purchases %>%
    mutate(date_of_purchase = ymd(date_of_purchase)) %>%
    mutate(year_of_purchase = year(date_of_purchase))

## Look at the data
head(purchases)
```

customer_id	purchase_amount	date_of_purchase	year_of_purchase
860	50	2012-09-28	2012
1200	100	2005-10-25	2005
1420	50	2009-07-09	2009
1940	70	2013-01-25	2013
1960	40	2013-10-29	2013

customer_id	purchase_amount	date_of_purchase	year_of_purchase
2620	30	2006-03-09	2006

```
summary(purchases)
```

```
##   customer_id     purchase_amount   date_of_purchase     year_of_purchase
##  Min.   :    10   Min.   :   5.00   Min.   :2005-01-02   Min.   :2005
##  1st Qu.: 57722   1st Qu.:  25.00   1st Qu.:2009-01-17   1st Qu.:2009
##  Median :102440   Median :  30.00   Median :2011-11-23   Median :2011
##  Mean   :108937   Mean   :  62.34   Mean   :2011-07-14   Mean   :2011
##  3rd Qu.:160528   3rd Qu.:  60.00   3rd Qu.:2013-12-29   3rd Qu.:2013
##  Max.   :264200   Max.   :4500.00   Max.   :2015-12-31   Max.   :2015
```

2 Compute Recency, Frequency, and Monetary Value

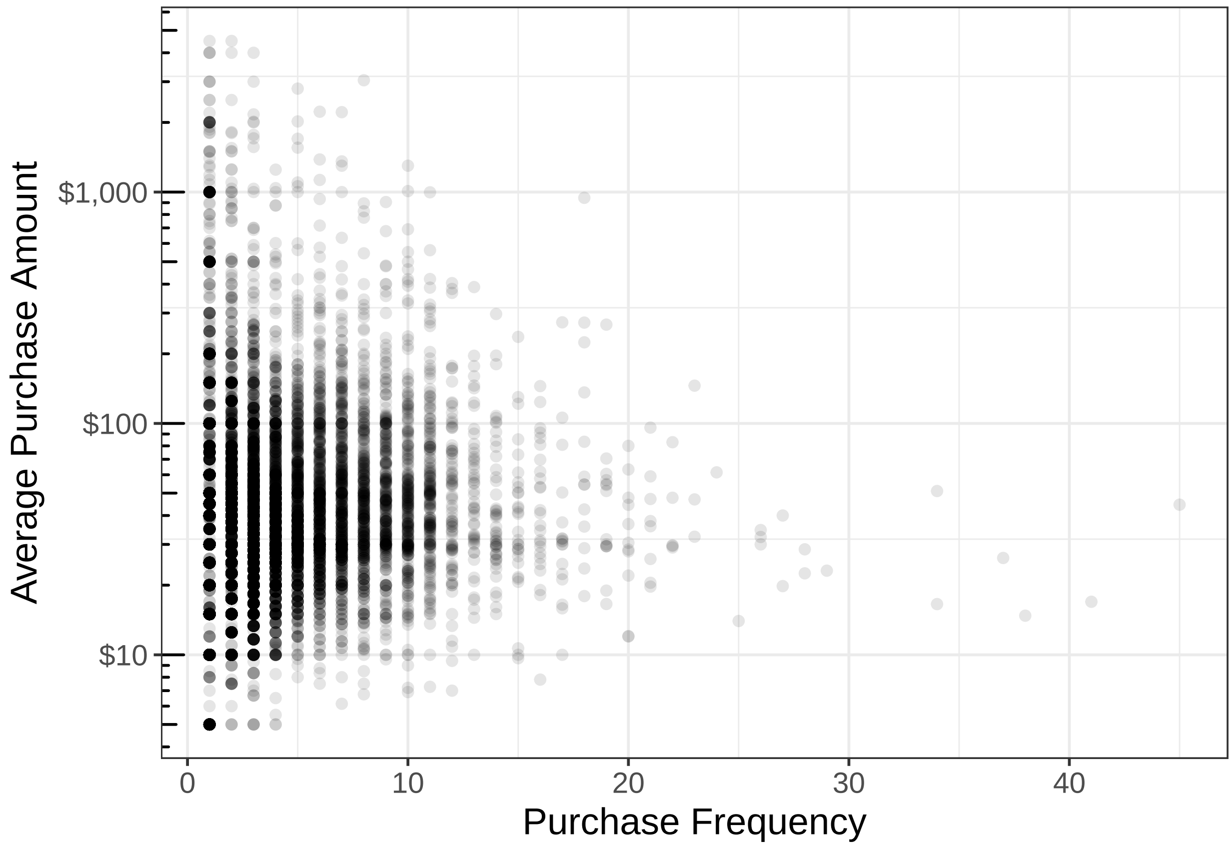
Three common characteristics of customers readily available from a transactional database are recency, frequency, and monetary value.

1. *Recency*: Time since last purchase.
2. *Frequency*: Number of purchases made in the past.
3. *Monetary value*: Amount spent at each purchase occasion.

Here we use these 3 simple characteristics to divide customers into actionable segments.

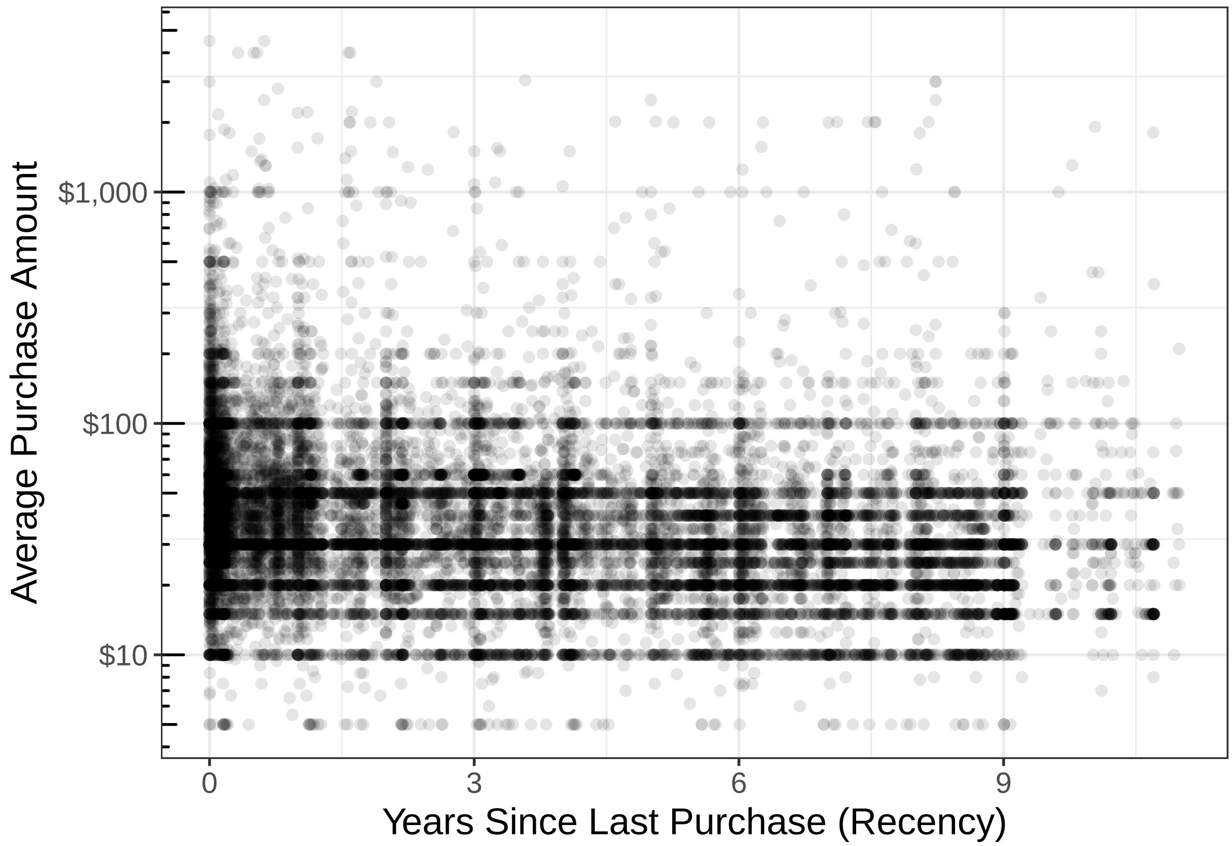
```
## Add columns for recency, frequency, and monetary value
recentDate <- max(purchases$date_of_purchase)
plotDat <- purchases %>%
    group_by(customer_id) %>%
    summarize(recency = min(recentDate - date_of_purchase),
              frequency = n(),
              monetary_value = mean(purchase_amount))

######################################
## Visually summarize these metrics ##
######################################
ggplot(data = plotDat, aes(x = frequency, y = monetary_value)) +
    geom_point(alpha = 0.1) +
    scale_y_log10(labels = scales::dollar) +
    annotation_logticks(sides = "l") +
    xlab("Purchase Frequency") +
    ylab("Average Purchase Amount") +
    getBaseTheme()
```




```
ggplot(data = plotDat, aes(x = recency / 365, y = monetary_value)) +
    geom_point(alpha = 0.1) +
    ## geom_hex() +
    scale_y_log10(labels = scales::dollar) +
    annotation_logticks(sides = "l") +
    xlab("Years Since Last Purchase (Recency)") +
    ylab("Average Purchase Amount") +
    getBaseTheme()
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.
```



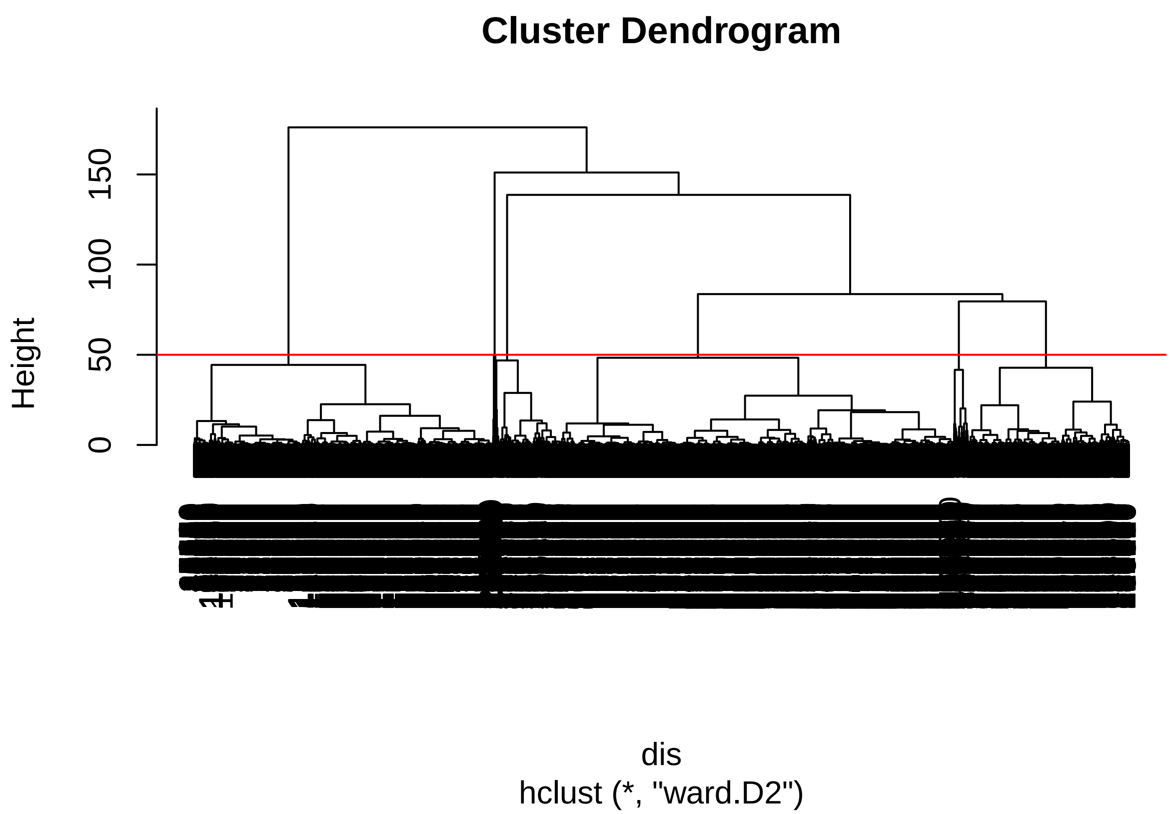
2.1 Customer Segmentation

```
## Note: The example code uses log of purchase amount because of the skewed distribution. This makes it
dat <- purchases %>%
    group_by(customer_id) %>%
    summarize(recency = as.numeric(min(recentDate - date_of_purchase)),
              monetary_value = mean(purchase_amount),
              ## monetary_value = mean(log10(purchase_amount)),
              frequency = n()) %>%
    as.data.frame()

## Put customer_id in the rownames of dat
rownames(dat) <- plotDat$customer_id
dat <- select(dat, -customer_id)

## Calculate distance matrix and hierarchical clustering
dis <- dist(scale(dat))
hc <- hclust(dis, method = "ward.D2")


## Plot the dendrogram
plot(hc)
abline(h = 50, col = "red")
```



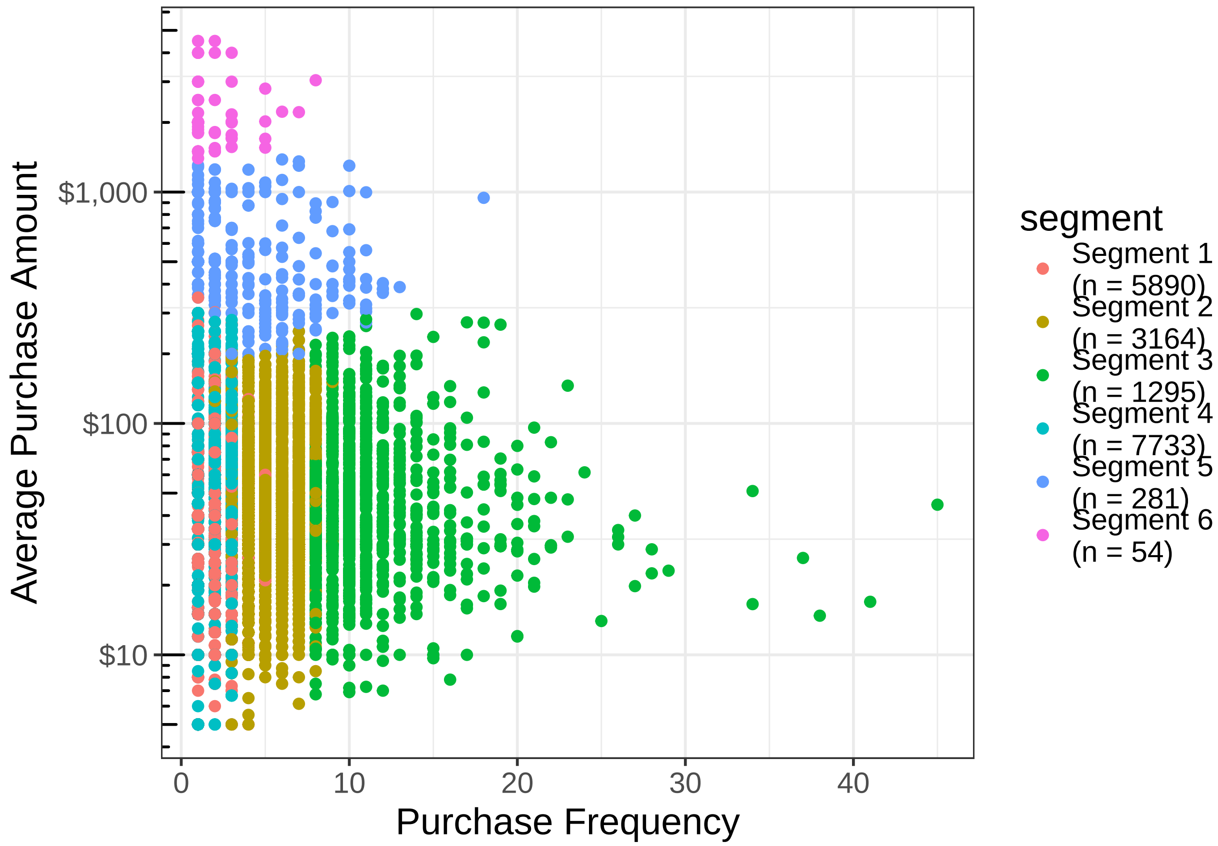

```
cuts <- cutree(hc, h = 50)
segments <- paste0("Segment ", cuts, "\n(n = ", table(cuts)[cuts], ")")
```

2.2 What does each segment look like?

We can make the same plot of Average Purchase Amount as a function of purchase frequency but color by segment.

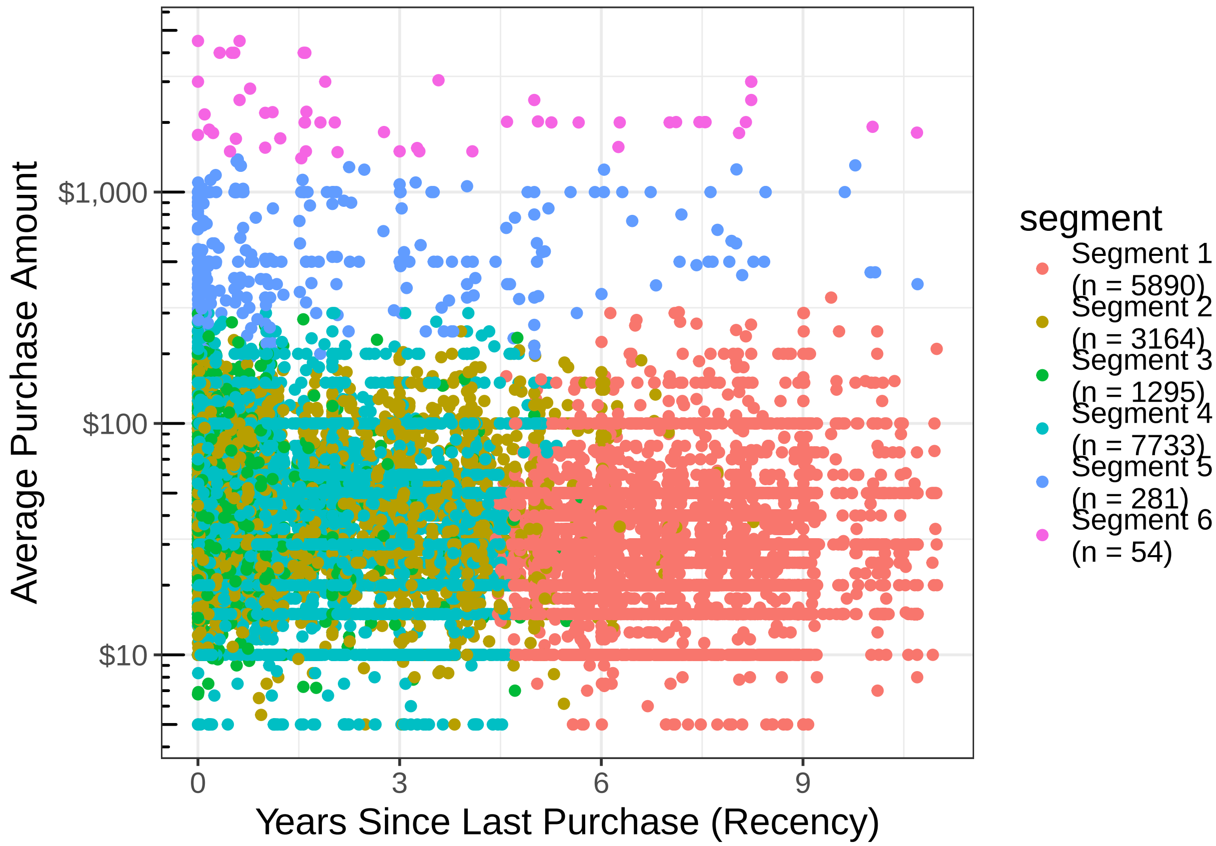
```
## Plot 2 variables at a time using cuts
## Shuffle row order to see overlapping points more clearly on plot
plotDat2 <- plotDat %>%
    mutate(segment = factor(segments)) %>%
    arrange(sample(nrow(plotDat)))

## Same plot as before but colored by cluster
ggplot(data = plotDat2,
       aes(x = frequency, y = monetary_value, color = segment)) +
    geom_point() +
    scale_y_log10(labels = scales::dollar) +
    annotation_logticks(sides = "l") +
    xlab("Purchase Frequency") +
    ylab("Average Purchase Amount") +
    getBaseTheme()
```




```
ggplot(data = plotDat2, aes(x = recency / 365, y = monetary_value, color = segment)) +
    geom_point() +
    scale_y_log10(labels = scales::dollar) +
    annotation_logticks(sides = "l") +
    xlab("Years Since Last Purchase (Recency)") +
    ylab("Average Purchase Amount") +
    getBaseTheme()
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.
```

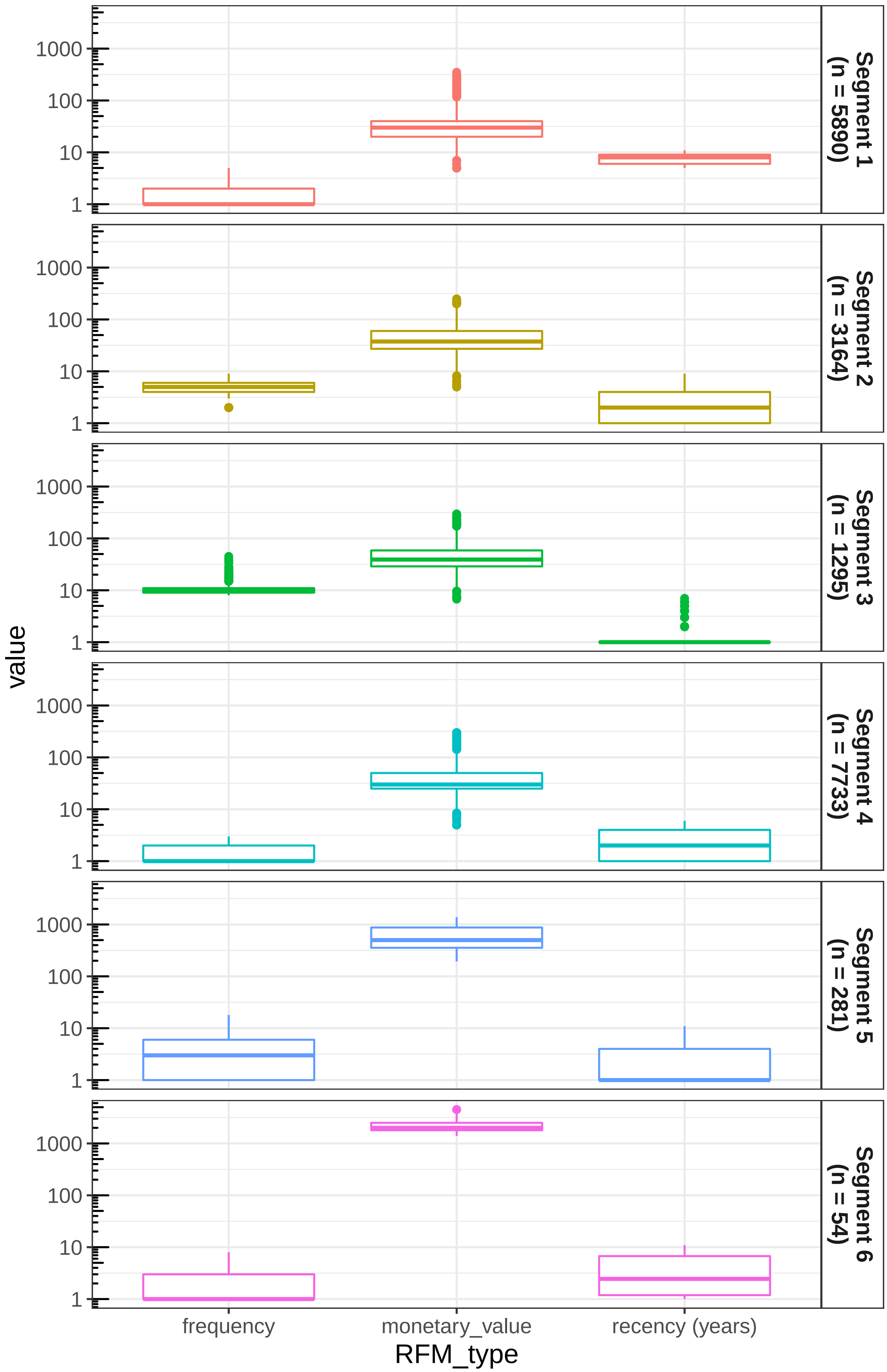


From the first plot we can see that all segments are clearly separated by these 2 variables except for 1 and 4. Segments 1 and 4 are separated by recency in the second plot.

We can visualize all 3 variables simultaneously in boxplots. This is a bit confusing because of the common scale when we are really showing 3 different things (frequency, dollars, years).

```
## Plot all 3 variables in boxplots
plotDat3 <- plotDat2 %>%
    mutate(recency = (as.numeric(recency) %/% 365) + 1) %>% # integer division
    rename(`recency (years)` = recency) %>%
    gather(key = "RFM_type", value = "value",
           `recency (years)`, frequency, monetary_value)

ggplot(data = plotDat3, aes(x = RFM_type, y = value, color = segment)) +
    geom_boxplot(position = "dodge") +
    scale_y_log10(breaks = c(1, 10, 100, 1000)) +
    annotation_logticks(sides = "l") +
    facet_grid(segment ~ .) +
    getBaseTheme() +
    scale_color_discrete(guide = FALSE) +
    theme(strip.text = element_text(size = 12, face = "bold"))
```


value
1000
100
10
1
Segment 1 (n = 5890)
Segment 2 (n = 3164)
Segment 3 (n = 1295)
Segment 4 (n = 7733)
Segment 5 (n = 281)
Segment 6 (n = 54)
frequency
monetary_value
recency (years)
RFM_type

```
## Summarize each segment by the median of each variable
plotDat %>%
    mutate(segment = cuts) %>%
    group_by(segment) %>%
    summarize(number = n(), median(frequency), median(recency), median(monetary_value)) %>%
    as.data.frame()
```

segment	number	median(frequency)	median(recency)	median(monetary_value)
1	5890	1	2568 days	30.00000
2	3164	5	394 days	37.50000
3	1295	10	86 days	39.16667
4	7733	1	617 days	30.00000
5	281	3	300 days	500.00000
6	54	1	717 days	2003.00000

3 Conclusions

Using hierarchical clustering we were able to define 6 clear customer segments with definining characterstics.

1. **Cold**

- These customers shopped at our store 1-5 times over 4 years ago and haven't been back.

2. **Almost Regulars**

- These customers spend a similar amount as the **Regulars** but don't buy as frequently (4 - 8 transactions)

3. **Regulars**

- These customers are regulars. They have shopped at least 8 times (some as many as 40 times). They have all made a purchase in the past year though their monetary value varies widely.

4. **Average Joes**

- Customers who have shopped 1 or 2 times in the past 4 years and spend about $30 each time.

5. **High End**

- A small segment of 281 customers who have shopped 1-10 times (the majority have shopped in the past year) and spend around $500.

6. **Money Items**

- The smallest segment of 54 customers who don't shop frequently, but when they shop, they spend close to $2,000.