Class 5: Predicting Response with RFM analysis

Professor Song Yao
Olin Business School

Customer Analytics

What is RFM Analysis?

Predictive models come in non-statistical and statistical varieties

TYPES OF PREDICTIVE MODELS

- Heuristics (rule of thumb)
  - Recency, frequency, Monetary (RFM) analysis
- Statistical models
  - Regression models
  - Discrete choice models
  - Neural Networks
  - Decision Trees (Boosted and Random Forests)
  - Recommender systems

RFM is easy to use and requires little statistical knowledge

RFM FACTS

- A useful and easy segmentation framework
- We know normally two things about customers
  - Who they are (demographic data)
  - What they do (behavioral data)
  - Behavior beats demographics most of the time!
- RFM is purely based on behavior
- Applies only to existing customers, not to prospects
- RFM is about response rate (binary DV), not revenue/profitability (continous DV)

The premise of RFM is that past behavior predicts future behavior

ELEMENTS OF AN RFM ANALYSIS

- **Recency**
  - How long ago did the customer make the purchase?
- **Frequency**
  - How many purchases has the customer made (in given time period)
  - Not as a good predictor of response as recency
- **Monetary**
  - How much has the consumers spent in total (in given time period)
  - Least predictive

We begin by coding RFM into N-tiles (quantiles)

START OF RFM ANALYSIS: Segmentation

- Pick variable of interest (recency, frequency, monetary)
- Sort database from best to worst on the variable
- Decide into how many groups to classify consumers
  - Normally pick 5 groups --> split consumers into "quintiles"
  - If pick 10 groups --> "deciles"
- Assign consumers to groups
  - Top group is quintile 1, second is quintile 2, etc.
  - *Make sure most desirable group for each variable is in the first group (BUT WHY?)*
    - Hint: It is a segmentation framework

Next we combine the N-tiles into an "RFM Index"

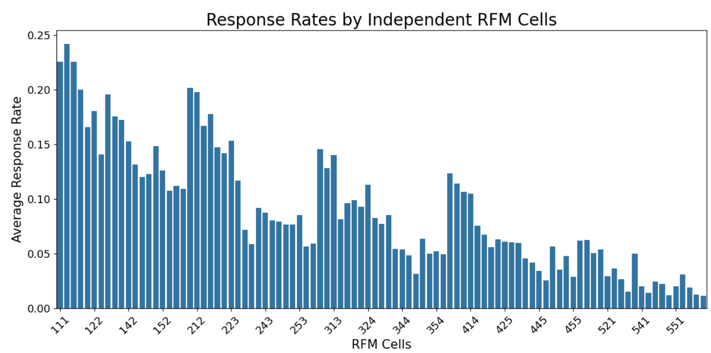
NEXT STEPS IN RFM ANALYSIS: Test round on a small set of customers

- Assign every customer a 3-digit code, e.g. 125, 555, etc.
  (quintile for R, quintile for F, quintile for M)
  -> will have 125 cells for quintiles

- Take a random sample of customers

- Approach them with an offer

- Calculate response rate per RFM cell



Finally, we select profitable cells and target them with an offer

FINAL STEPS IN RFM ANALYSIS: Targeting remaining customers

- An offer is profitable if

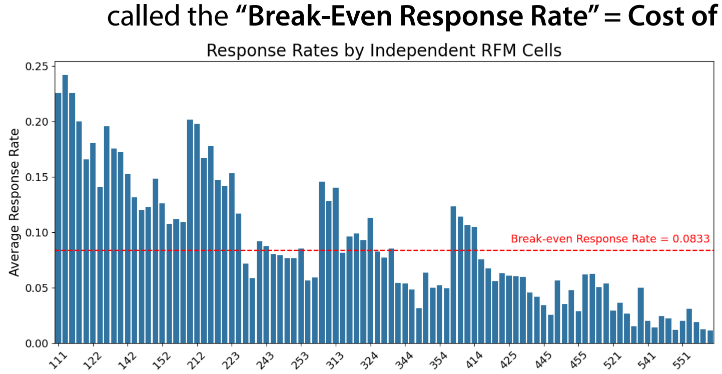
$$\overbrace{\text{Probability of Response} * \text{Profit if Sold}}^{\text{Expected Profits}} - \text{Cost of Targeting} \geq 0$$

$$\text{Probability of Response} \geq \boxed{\text{Cost of Targeting}} / \boxed{\text{Profit if Sold}}$$

Happen for sure if targeted

May not happen even if targeted

- The lowest probability of response for which the offer is still profitable is called the **"Break-Even Response Rate" = Cost of Targeting/Profit if Sold**



- Select cells whose test resp. rate higher than break-even response rate

- Target remaining consumers in those cells

Check if the RFM Premises Hold

We use the Bookbinders Book Club as an example of how to target an offer with RFM analysis

RFM TEST AT BOOKBINDERS

- Stan Lawton (marketing director) pulls a random sample of 50,000 customers from the Bookbinders database
- Stan mails "The Art History of Florence" to the entire sample
- 4522 customers buy the book
- Stan has information on the
  - recency of the last purchase,
  - the purchase frequency, and
  - total expenditure of each customers
- Plans to use test to determine which customers to target from the entire database (500,000 remaining customers, excluding test group)

We will use the Bookbinders example to, first, explore the assumptions of RFM and, second, perform an RFM analysis

BOOKBINDERS ANALYSIS

1. **Explore the assumptions of RFM:**
   - Do buyers and non-buyers differ on R, F, and M variables?
   - Are R, F, and M independent measures?
2. **Perform RFM:**
   - Construct the RFM index in R
     - How predictive are R, F, and M variables of the likelihood of purchasing?
   - Introduce 3 approaches to creating RFM index
   - Calculate bread-even response rate
   - Calculate profitability of RFM

Does the premise of RFM Analysis seem to hold for Bookbinders?

SUMMARY STATISTICS BY BUYERS

buyer	count	percentage
0.00	45478.00	90.96
1.00	4522.00	9.04

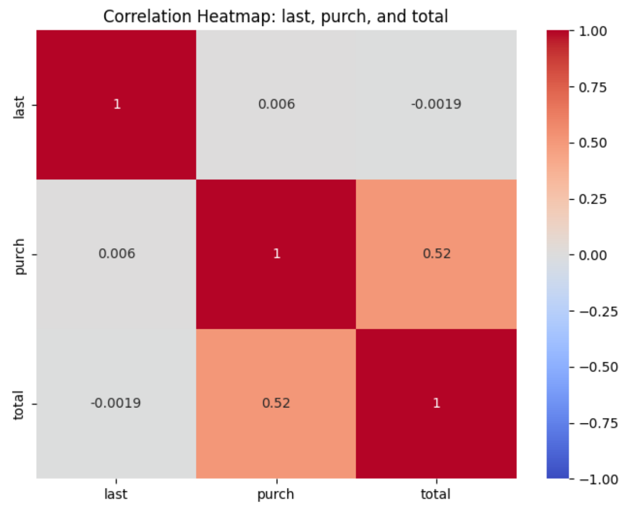
Comparison of average last, purch, and total between buyers and non-buyers:

buyer	last_average	purch_average	total_average
Non-buyer	12.73	3.76	205.73
Buyer	8.61	5.22	234.30

- What can we conclude?

Do RFM all capture the same underlying behavioral characteristic?

CORRELATION BETWEEN RFM VARIABLES



- What does this say and mean?

Create Customer RFM Cells

We begin by forming R, F, M quintiles

```
# Create quintiles for R, F, M (last, purch, and total)
BBB['rec_quin'] = xtile(BBB['last'], 5)
BBB['freq_quin'] = xtile(BBB['purch'], 5)
BBB['mon_quin'] = xtile(BBB['total'], 5)

# Display the first few rows to verify the new columns
print(BBB[['last', 'rec_quin', 'purch', 'freq_quin', 'total', 'mon_quin']].head())
```

	last	rec_quin	purch	freq_quin	total	mon_quin
0	29	5	10	5	357	5
1	27	5	3	4	138	2
2	15	4	2	2	172	2
3	7	2	1	1	272	4
4	15	4	1	1	149	2

The xtile function is self-defined and segments customers based on a specific variable (e.g., Recency)

```
# define a function called xtile to create tiles
# the function is used to create tiles for RFM analysis
def xtile(data, num_tiles):
    # Calculate quantiles
    quantiles = np.percentile(data, np.linspace(0, 100, num_tiles + 1))

    # Use numpy.digitize to assign labels
    tile_labels = np.digitize(data, quantiles[1:-1], right=True) + 1

    # Ensure the maximum label is num_tiles
    tile_labels[tile_labels > num_tiles] = num_tiles

    return tile_labels
```

After forming recency quintiles, let's check average recency and response rate of each quintile

Average Recency BY RECENCY QUINTILE

```
# Check the average recency (last) across rec_quin
average_recency = BBB.groupby('rec_quin')['last'].mean().reset_index()
average_recency = average_recency.sort_values('rec_quin')

print("Average recency across rec_quin:")
print(average_recency)
```

```
Average recency across rec_quin:
   rec_quin       last
0         1   2.984962
1         2   8.241612
2         3  12.005732
3         4  15.297825
4         5  26.553675
```

Response Rate of people in the same recency quintile

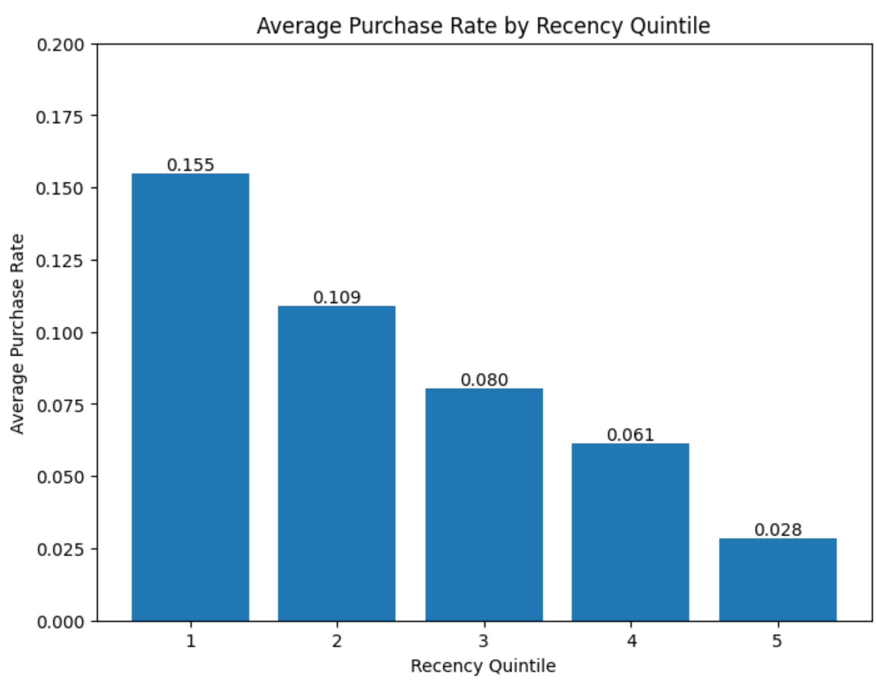
% Buyers BY RECENCY QUINTILE (response rate)

```
# Calculate average purchase rate for each rec_quin
average_purchase_rate = BBB.groupby('rec_quin')['buyer'].mean().reset_index()
print("Average purchase rate across rec_quin:")
print(average_purchase_rate)
```

```
Average purchase rate across rec_quin:
   rec_quin     buyer
0         1  0.154672
1         2  0.108894
2         3  0.080326
3         4  0.061244
4         5  0.028239
```


Using a bar chart to visualize purchase rates by recency quintiles

% BUYERS BY RECENCY QUINTILE



Next, we do the same for frequency quintiles

Freuquency BY FREQUENCY QUINTILE

Are the most frequent customers in quintile 1?

```
# Check the average frequency (purch) across freq_quin
average_frequency = BBB.groupby('freq_quin')['purch'].mean().reset_index()
average_frequency = average_frequency.sort_values('freq_quin')

print("Average frequency across freq_quin:")
print(average_frequency)
```

```
Average frequency across freq_quin:
   freq_quin      purch
0          1   1.000000
1          2   2.000000
2          4   5.009859
3          5  10.018378
```

NO! Flip it!

```
# Flip the labels of freq_quin
BBB['freq_quin'] = 6 - BBB['freq_quin']
```

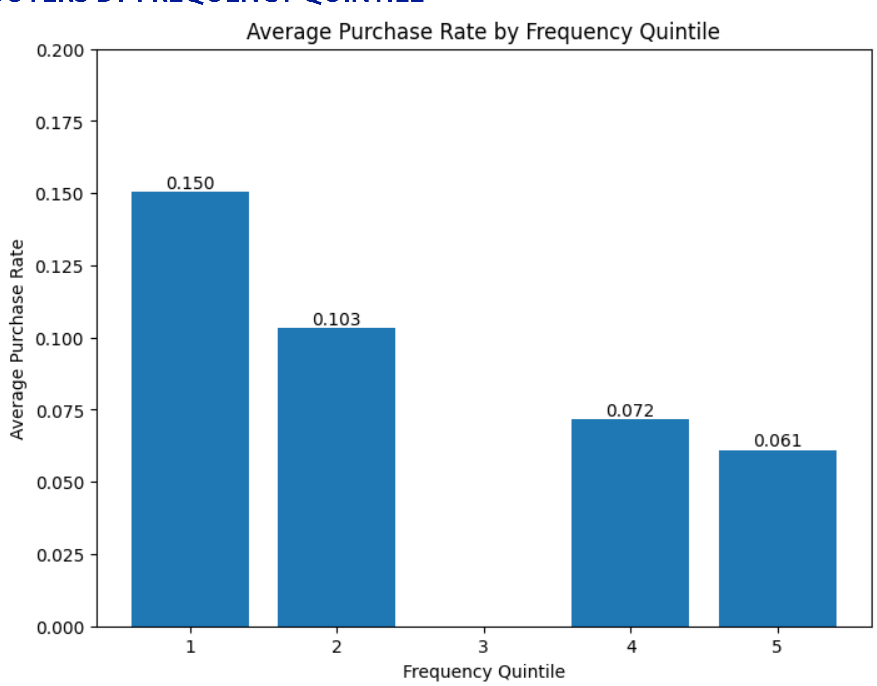
```
# Check the average frequency (purch) across freq_quin
average_frequency = BBB.groupby('freq_quin')['purch'].mean().reset_index()
average_frequency = average_frequency.sort_values('freq_quin')

print("Average frequency across freq_quin:")
print(average_frequency)
```

```
Average frequency across freq_quin:
   freq_quin      purch
0          1  10.018378
1          2   5.009859
2          4   2.000000
3          5   1.000000
```


Using a bar chart to visualize purchase rates by Frequency quintiles

% BUYERS BY FREQUENCY QUINTILE



We asked for 5 Frequency cells. Why did we only see 4 cells?
TABULATION OF NUMBER OF PURCHASES

```
Crosstab for BBB['purch']:
 purch  count  percentage
     1  15120      30.240
     2  14935      29.870
     3   2019       4.038
     4   1963       3.926
     5   2018       4.036
     6   1984       3.968
     7   2058       4.116
     8   1955       3.910
     9   1945       3.890
    10   1968       3.936
    11   2033       4.066
    12   2002       4.004
```


Finally, let's look at monetary quintiles

Are the largest spenders in quintile 1?

```python
# Check the average money (total) across mon_quin
average_money = BBB.groupby('mon_quin')['total'].mean().reset_index()
average_money = average_money.sort_values('mon_quin')

print("Average money across mon_quin:")
print(average_money)
```

```
Average money across mon_quin:
   mon_quin       total
0         1   68.140231
1         2  145.625627
2         3  209.180557
3         4  269.048896
4         5  351.279562
```

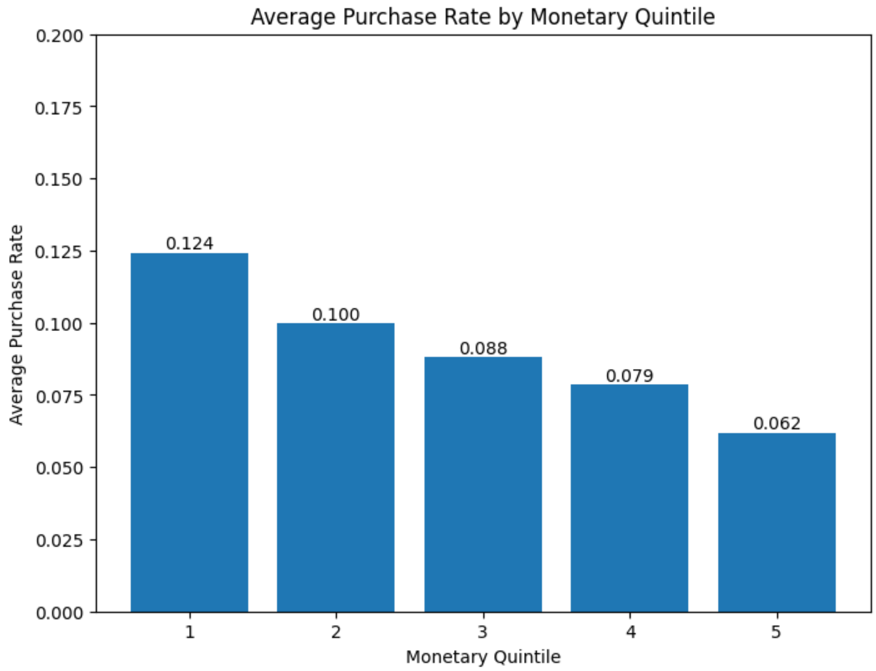
NO! Flip it!

```python
# Flip the labels of mon_quin
BBB['mon_quin'] = 6 - BBB['mon_quin']
```

```
Average money across mon_quin:
   mon_quin       total
0         1  351.279562
1         2  269.048896
2         3  209.180557
3         4  145.625627
4         5   68.140231
```

Using a bar chart to visualize purchase rates by monetary quintiles

% BUYERS BY MONETARY QUINTILE



Now, let's combine the R, F, M into a single index

```
# Create a new variable 'RFM_label' by concatenating R, F, M quintile labels
BBB['RFM_index'] = BBB['rec_quin'].astype(str) + BBB['freq_quin'].astype(str) \
    + BBB['mon_quin'].astype(str)
BBB['RFM_index'] = BBB['RFM_index'].astype(int)
```

```
First few rows of BBB with the new RFM_label:
   rec_quin  freq_quin  mon_quin  RFM_index
0         5          1         1        511
1         5          2         4        524
2         4          4         4        444
3         2          5         2        252
4         4          5         4        454
```

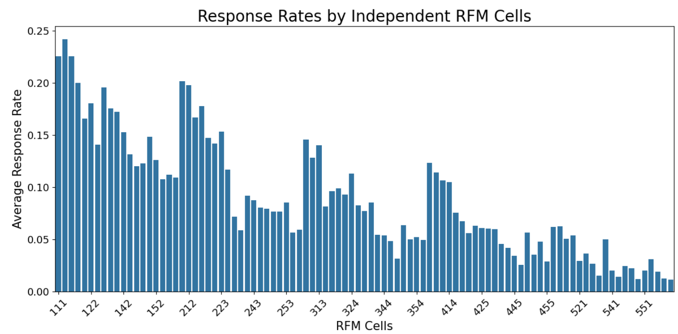
The RFM-index is easy to calculate

RESPONSE RATE BY INDEPENDENT N-TILE RFM INDEX

```python
# Calculate average purchase rate for each RFM_index
average_purchase_rate_rfm = \
    BBB.groupby('RFM_index')['buyer'].mean().reset_index(name='resp_rate_rfm')

# Sort the dataframe by RFM_index for better readability
average_purchase_rate_rfm = average_purchase_rate_rfm.sort_values('RFM_index')
```

```
Average purchase rate by RFM_index:
    RFM_index  resp_rate_rfm
0         111       0.225448
1         112       0.242009
2         113       0.225734
3         114       0.200000
4         121       0.165455
```

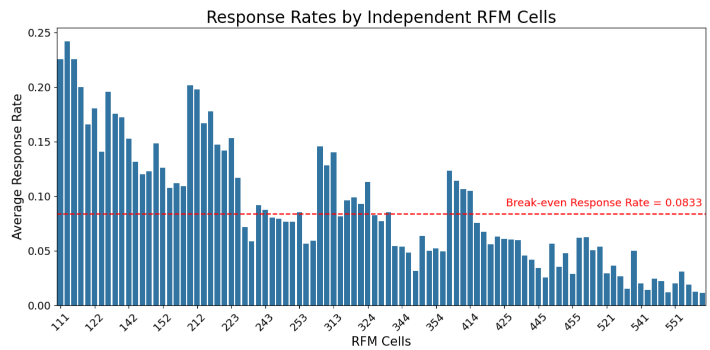


Whom shall we target using RFM analysis and how does RFM improve the ROI?

Bookbinder RFM: The break-even response rate tells us to which cells to extend the offer

BREAK EVEN RESPONSE RATE

- Cost of mailing an offer = $0.50
- Selling price (includes shipping) = $18
- Wholesale price paid by Bookbinders = $9
- Shipping costs = $3
  - Profit if sold = $18-9-3 =$6
- Break-even = Cost to mail / Profit if sold = 0.5 / 6 = **8.3%**



As a benchmark we calculate the profitability of mailing the full 500,000 consumers

PROFITABILITY (FULL SAMPLE)

- Mail to 500,000 (10 times of the sample size)
- Average response rate for these 500,000 customers?

buyer	count	percentage
0.00	45478.00	90.96
1.00	4522.00	9.04

Aver. response rate: 9.04%
Expected number of buyers:
9% * 500,000 = 45,200

Total expected profit:
Profit of one book sold *
Expected number of buyers

Total mailing cost (cost of offer) is $250,000

- Gross profit = $\boxed{(\$18 - 9 - 3)* 45{,}200}$ - $\boxed{0.5*500{,}000}$ = **$21,200**
- Return on marketing expenditure = $21,200/$250,000 = **8.5%**

What is the response rate of each RFM cell?

```
# Calculate average purchase rate for each RFM_index
average_purchase_rate_rfm = \
    BBB.groupby('RFM_index')['buyer'].mean().reset_index(name='resp_rate_rfm')

# Display the results
print("Average purchase rate by RFM_index:")
print(average_purchase_rate_rfm)
```

```
Average purchase rate by RFM_index:
    RFM_index  resp_rate_rfm
0         111       0.225448
1         112       0.242009
2         113       0.225734
3         114       0.200000
4         121       0.165455
..        ...            ...
90        551       0.020134
91        552       0.030534
```


Whom do we target: If the person belongs to a group with the average response rate > 8.3%, the break-even rate

```
# Define the break-even point
unit_profit = 18-9-3
unit_mkt_cost = 0.5
break_even = unit_mkt_cost/unit_profit

# Create a new column 'targeted' based on the break-even point
BBB['targeted'] = (BBB['avg_response_rate_rfm'] > break_even).astype(int)
```

8.3%

```
First few rows of BBB with targeted column:
   RFM_index  avg_response_rate_rfm  targeted
0        511               0.061937         0
1        524               0.014815         0
2        444               0.041420         0
3        252               0.076220         0
4        454               0.047431         0

Summary of targeted customers:
targeted
0    0.53464
1    0.46536
```

=1, If the customer belongs to an RFM group with average reps. rate >8.3%
=0, otherwise

Using the RFM index we target fewer customers who have higher response rates than the break-even rate

```
Summary of targeted customers:
targeted
0    0.53464
1    0.46536
```

Mail to 46.54% of customers: 500,000 * 46.54% **=232,700 (i.e., targeted_customers)**

```
targeted_buyers = BBB[(BBB['targeted'] == 1) & (BBB['buyer'] == 1)].shape[0]
targeted_buyers_percentage = targeted_buyers / targeted_customers
```

```
Targeted customers who bought: 3270 (14.05% of targeted customers)
```

Expected response rate = 14.05%
Expected number of buyers:
14.05% * 232,700 = **32,694**

- Profit = ($18-9-3)* 32,694 - 0.5*232,700 = **$79,814**
- Marketing expenditure = 0.5*232,700 = **$116,350**
- Return on marketing expenditure = $79,814/$116,350 = **68.6%**

Three Approaches for Creating RFM Cells

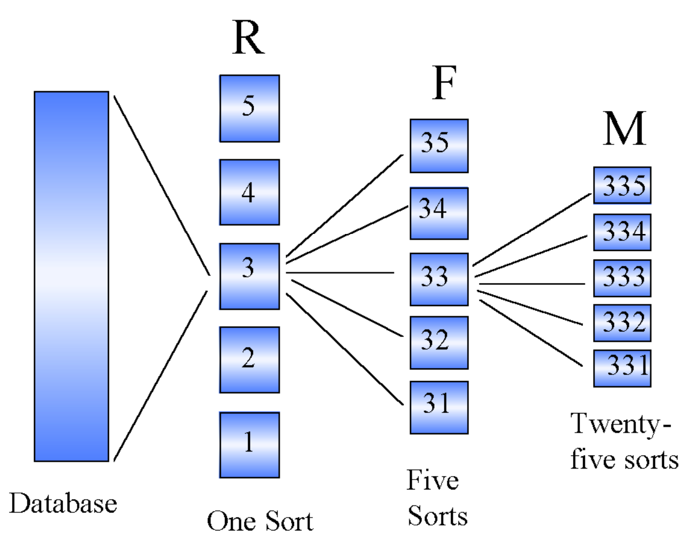
There are several ways of constructing an RFM index

TYPES OF RFM INDICES

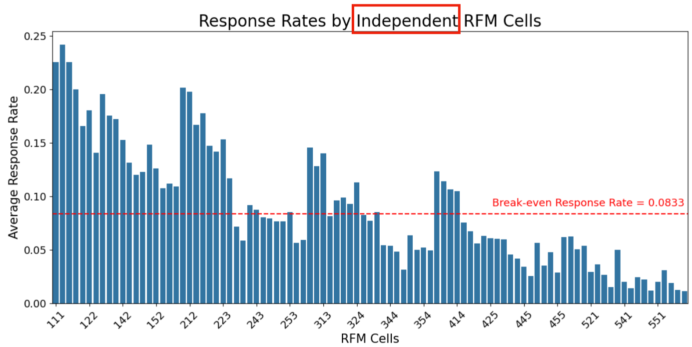
- **Independent N-tile approach**
  - Create quintile for recency
  - Independently create quintile for frequency
  - Independently create quintile for monetary
- **Sequential N-tile approach**
  - Create quintile for recency
  - Within each recency quintile, create quintiles for frequency
  - Within each of 25 recency-frequency groups, create quintiles for monetary
- **Intuitive groupings approach**
  - Pick intuitive cutoff points for R, F, and M
  - e.g. one-time buyers vs. repeat-buyers, less than 6 months, 6mo-1year, more than 1 year

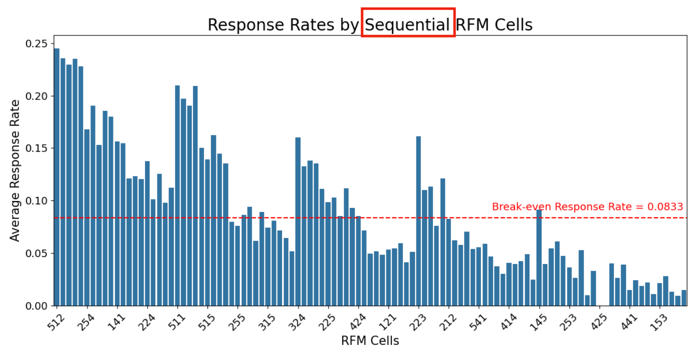
The RFM index based on sequential N-tiles is substantially harder to calculate but is considered a better approach

CONSTRUCTION OF SEQUENTIAL N-TILES



RESPONSE RATES BY INDEPENDENT AND SEQUENTIAL RFM INDEX





There are several ways of constructing an RFM index

TYPES OF RFM INDICES

- Independent N-tile approach
  - Create quintile for recency
  - Independently create quintile for frequency
  - Independently create quintile for monetary
- Sequential N-tile approach
  - Create quintile for recency
  - Within each recency quintile, create quintiles for frequency
  - Within each of 25 recency-frequency groups, create quintiles for monetary
- **Intuitive groupings approach**
  - Pick intuitive cutoff points for R, F, and M
  - e.g. one-time buyers vs. repeat-buyers, less than 6 months, 6mo-1year, more than 1 year

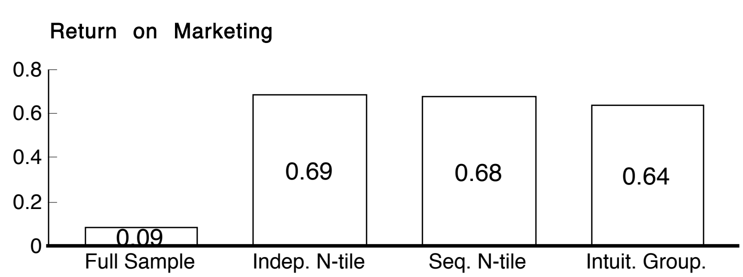
The RFM index based in intuitive groupings uses intuitive breakpoints instead of strict N-tiles

EXAMPLE OF INTUITIVE GROUPINGS

	One-Time buyers		Repeat Buyers	
	Low $	High $	Low $	High $
0-6 months				
7-12 months				
1-2 years				
2-3 years				
3+ years				

Using behavioral information with RFM analysis can dramatically improve the return on a marketing campaign

RFM IMPROVEMENTS



Independent and Sequential RFM do have a key difference: Group size variation