Felix Analysis 1

Justin Skycak

June 3, 2016

The Dataset

About 10,000,000 transactions from 9999 accounts, via 71 codes

accountID	date	code	amount
1	9/9/2013	81	-$6289.69
1	9/10/2013	997	+$4.25
...	...	...	...
1	6/3/2015	18	+$1831.17
...	...	...	...
9999	6/20/2015	81	-$3051.23

(activity_feature_histograms.R)

Activity Features

n_{ic} = number of code-*c* transactions for account *i*

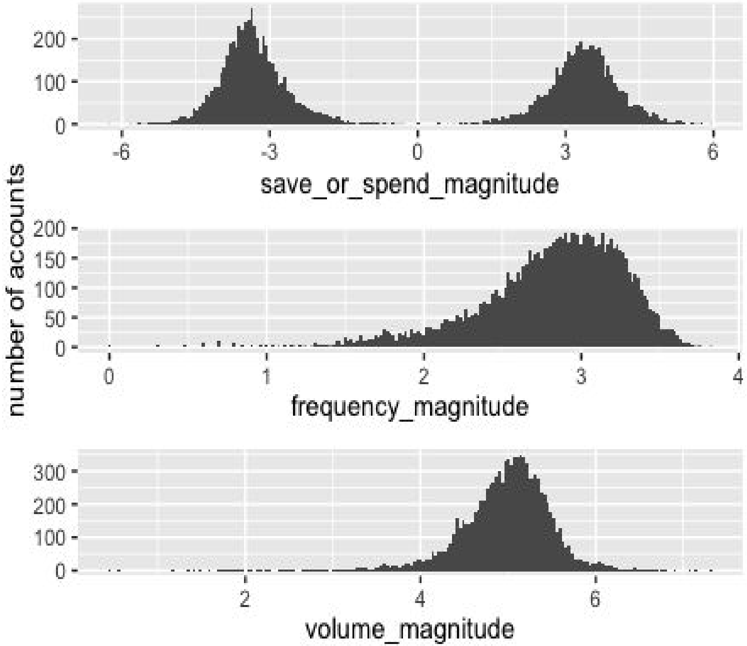
s_{ic} = sum of code-*c* transactions for account *i*

*The **order of magnitude** of …*

… an account's total savings or spendings $$sign\left(\sum_c s_{ic}\right)\log_{10}\left|\sum_c s_{ic}\right|$$

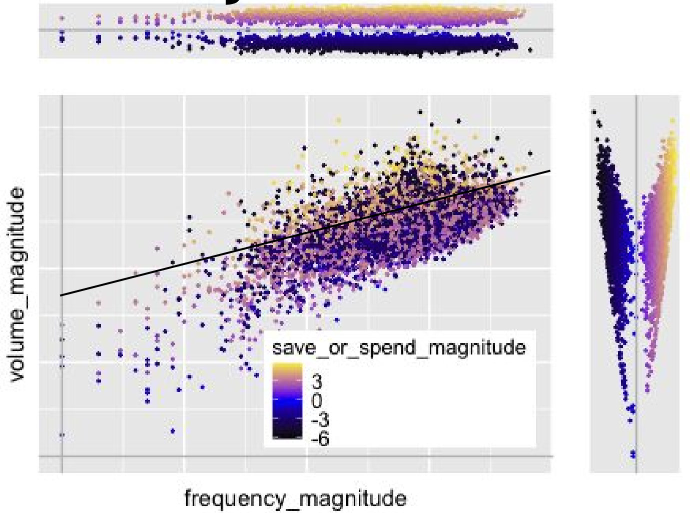
… how often an account was used $$\log_{10}\left(\sum_c n_{ic}\right)$$

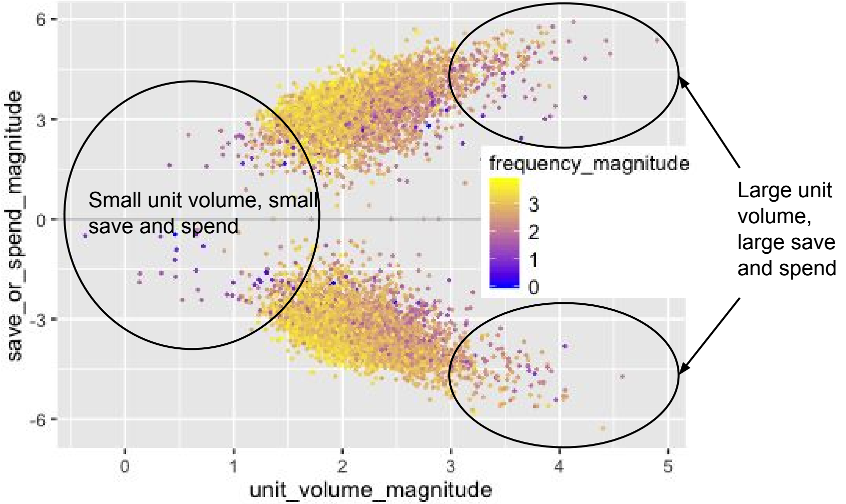
… the volume of money processed by an account $$\log_{10}\left(\sum_c |s_{ic}|\right)$$



(activity_visualization.R)
(activity_insights.R)

Activity Visualization and Insights





(unit volume = volume per transaction)

Observation: large save (yellow) and large spend (black) tend to be above the line, while small save (pink) and small spend (blue) tend to fall below it

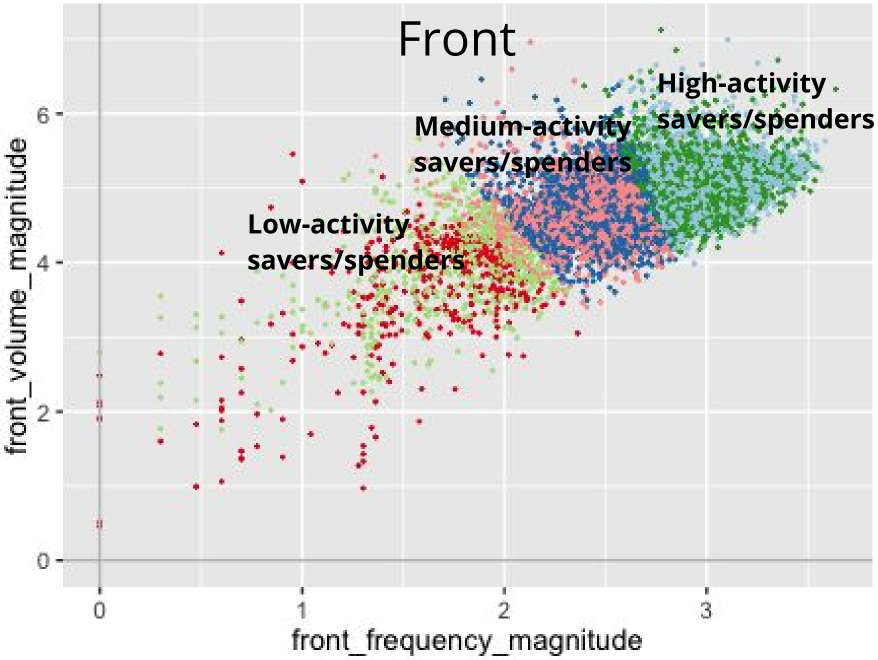
Regardless of frequency:

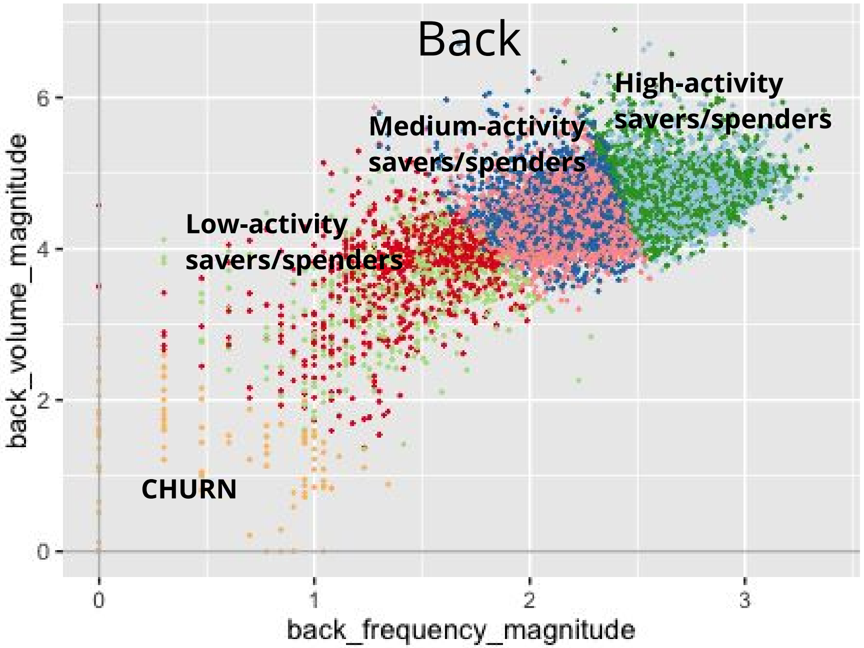
- small unit volume ⟵⟶ small save or spend
- large unit volume ⟵⟶ large save or spend

(plot_activity_clusters.R)

Activity Transitions

1. **Break dataset into front and back and cluster each**
2. Compute cluster transition probabilities



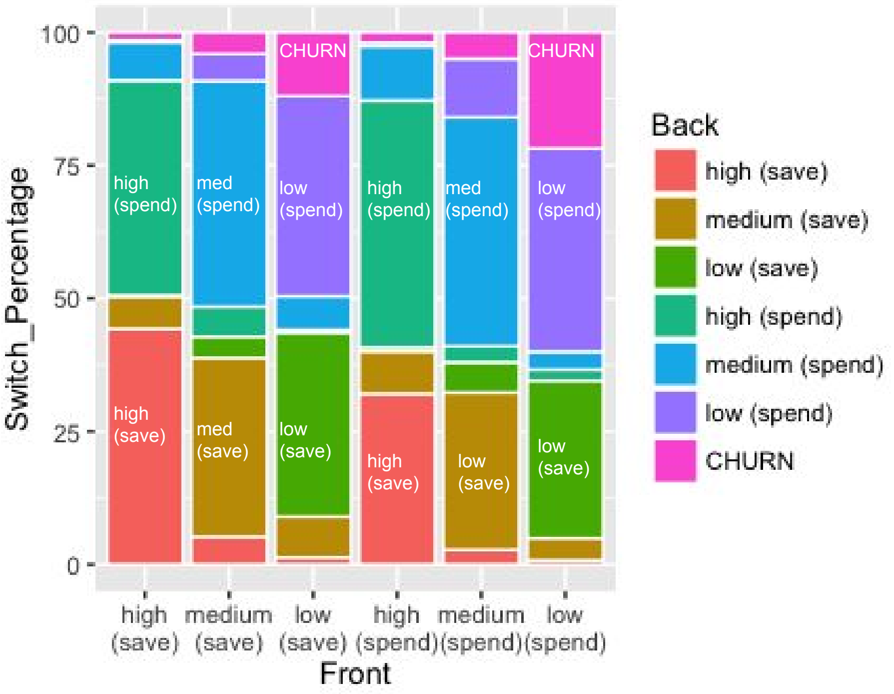


(activity_transitions_bar_plot.R)

Activity Transitions

1. Break dataset into front and back and cluster each
2. **Compute cluster transition probabilities**

- *Accounts like to maintain their activity level, but don't particularly care about maintaining spending or saving*
- *Churns most often come from low-activity customers*



Code Profiles

Account code usage represented by a binary *code_use* vector

$$code\_use_{ic} = \begin{cases} 1, & n_{ic} > 0 \\ 0, & n_{ic} = 0 \end{cases}$$

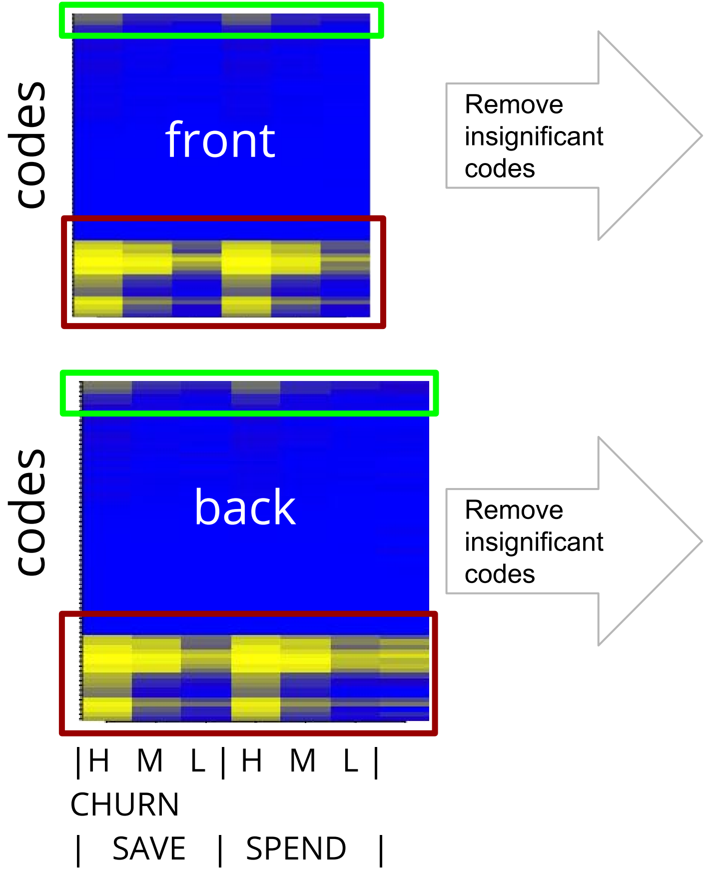
Why binary instead of graded?

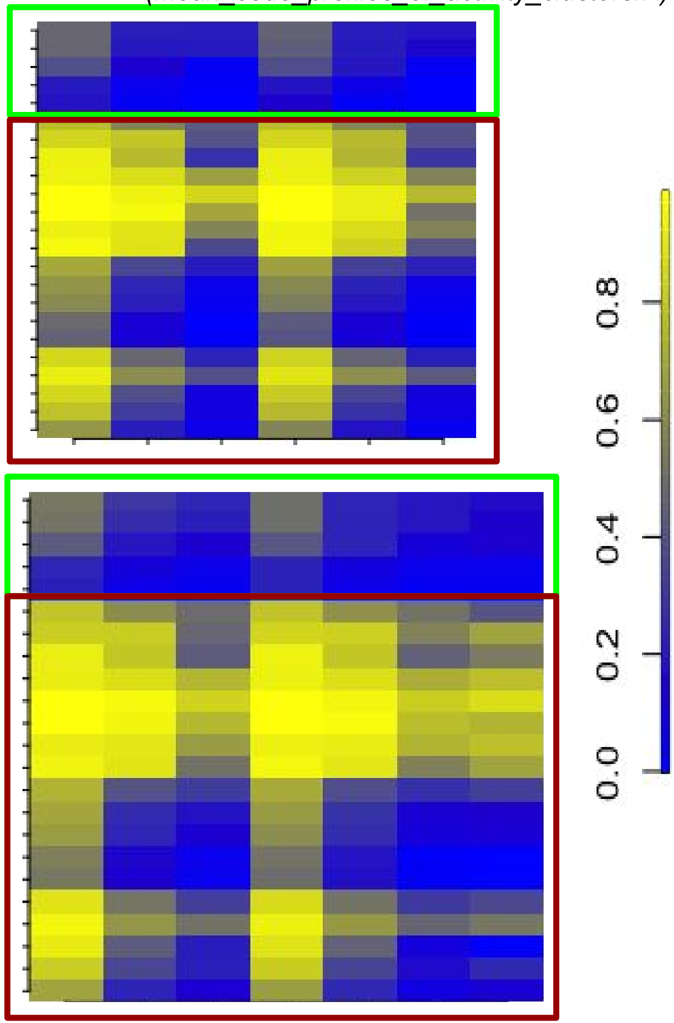
- Accounts use codes sparsely (1.4% of *code_use* entries were nonzero)
- Averages could be interpreted as probabilities of using codes

(mean_code_profiles_of_activity_clusters.R)

Code Profiles of Activity Clusters

- Time-invariant
- High (save) matches high (spend)
- Medium (save) matches medium (spend)
- Low (save) matches low (spend)
- Churn matches low (save) and low (spend)





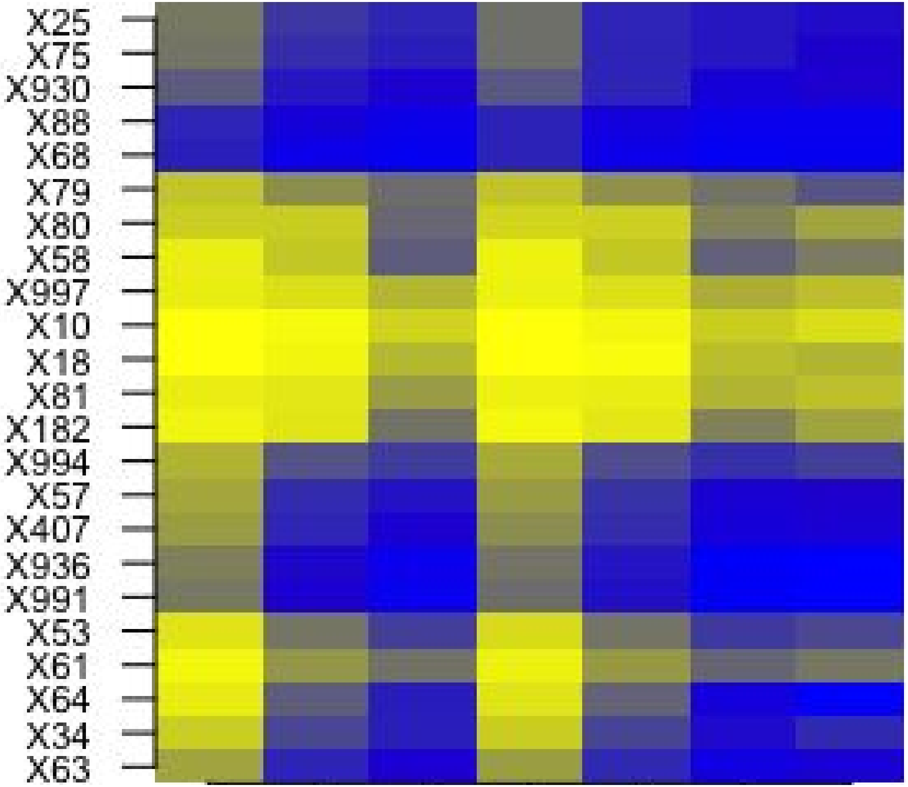
Code Profiles of Activity Clusters

(mean_code_profiles_of_activity_clusters.R)

Anything significant about the code descriptions?

(I am unfamiliar)

TRANSFER
CREDIT..................
TRANSFER
DEBIT....................
OVERDRAFT
CHARGE.............
LOAN
PAYMENT......................
SERVICE CHARGE
DEBIT........
CHECKING
WITHDRAWAL......
CHECK.....................................
ACH
DEBIT..............................
INTEREST
PAYMENT..............
DEPOSIT..................................
ACH
CREDIT...........................
CHECK....................................
ACH CONVERTED
CHECKS....
SERVICE
FEE...........................
NON-RESOURCE ATM
W/D



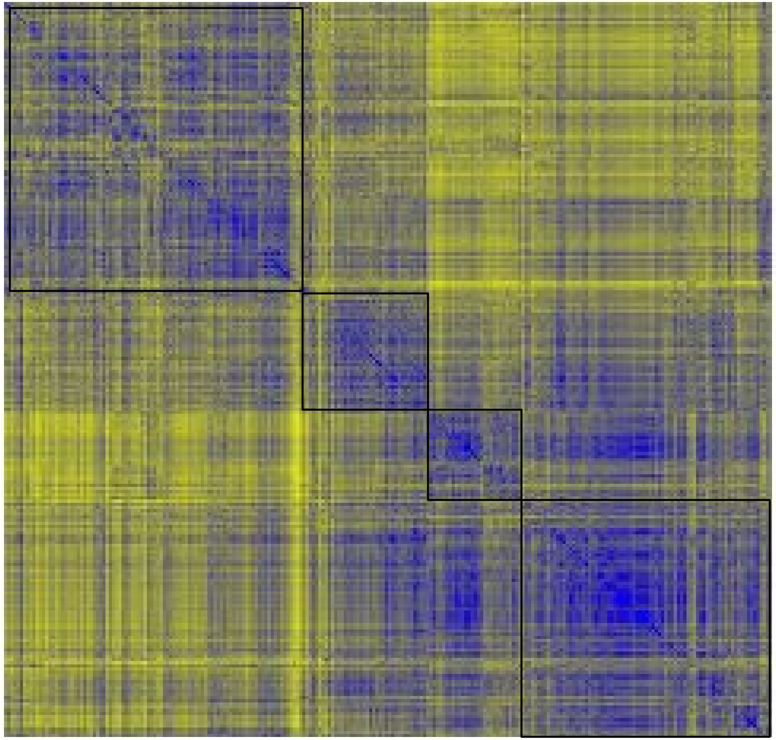
| H | M | L | H | M | L | CHURN
| SAVE | SPEND |

(account_distances_by_code_profile.R)

Code Profile Clustering

- Row i, column j contains distance between code profiles of accounts i and j
  (full dataset used)

- At least 4 major groups

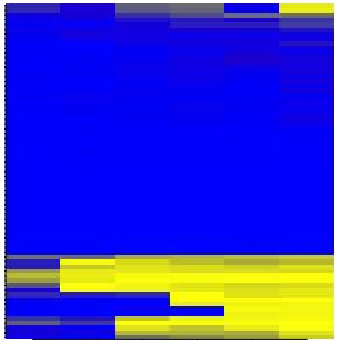


yellow = far | blue = close

Code Profile Clustering (k=6)

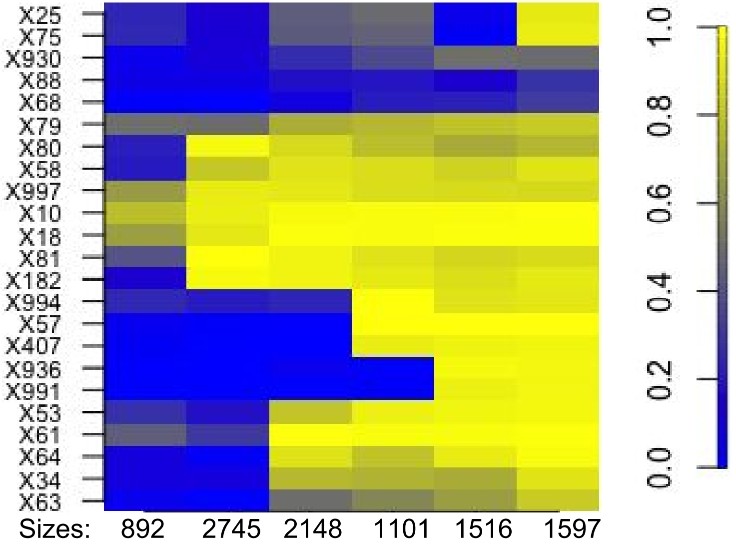
(full dataset used)

(code_profile_clusters.R)



Remove insignificant codes

TRANSFER CREDIT.................
TRANSFER DEBIT...................
OVERDRAFT CHARGE............
LOAN PAYMENT.....................
SERVICE CHARGE DEBIT........
CHECKING WITHDRAWAL......
CHECK.....................................
ACH DEBIT..............................
INTEREST PAYMENT...............
DEPOSIT.................................
ACH CREDIT..........................
CHECK...................................
ACH CONVERTED CHECKS....
SERVICE FEE..........................
NON-RESOURCE ATM W/D.....
CIRRUS FEE REND.............
ATM FEE.................................
ATM TRANSACTION FEE........
RESOURCE CENTER W/D......
POINT OF SALE DEBIT...........
DDA PURCHASE......................



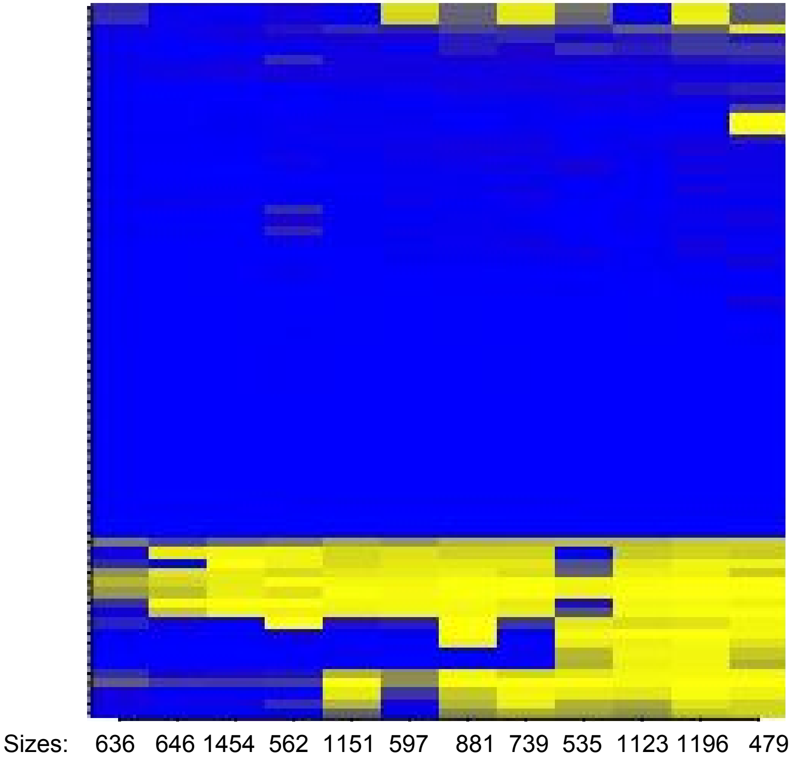
- Interesting result (noticed by Dave): clustering induced an *ordering* on the codes. (The ordering corresponds to the spectral color-coding red, orange, green, blue, purple, pink)
- Natural follow-up question: is ordering preserved with more clusters?

(code_profile_clusters.R)

Code Profile Clustering (k=12)

Using twice as many clusters as before

Ordering isn't strict, but seems to occur to some extent!



Recap of Key Findings

Activity visualization:

- small unit volume ←→ small save or spend
- large unit volume ←→ large save or spend

Front-back activity cluster transitions:

- accounts like to maintain their activity level, but don’t particularly care about maintaining spending or saving
- churns most often come from low-activity customers

Code profiles:

- high (save) matches high (spend)
- medium (save) matches medium (spend)
- low (save) matches low (spend)
- churn matches low (save) and low (spend)

Code clustering induces an order on the codes