Behavioral Analysis : Cohort Repayment Curve

Oppong-Agyare Anokye Nkansah

2024-01-19

Introduction

Asset financing is a so-called structured financing solution. It allows companies and individuals to finance the purchase of assets such as aircraft, ships, trains and, in some cases, real estate. These are medium- to long-term financing projects. Top Asset financing include **Nordic Aviation Capital**, **DLL Group**, **Lease Corporation International**, **Praetura Asset Finance**, **Capitalflow** and **M-KOPA**.

M-KOPA is an African connected asset financing platform that provides underbanked customers in Africa to essential products including solar lighting, televisions, fridges, smartphones & financial services. M-KOPA currently operates in 4 markets, namely Ghana, Kenya, Uganda and Nigeria with about 4 Million customers in these markets.

Problem Statement

Credit companies heavily rely on customer repayment to make profits. Crediting underbanked people with very little credit history and poor data poses extra threat and risk to such companies like M-KOPA. It is therefore very important to track repayment of loans and debts. M-KOPA credit team has been implementing different behavioural nudges to improve the repayment of debts by customers over the years. These nudges include;

- Daily repayment of very small amount.
- A recent implementation of a remote lock access to these products.
- All essential products that are fully solar powered.

To track the repayment over time, Customers are segmented into cohorts based on the month registration. A cohort is a subset of customers who were all registered in the same month.

A Cohort Repayment Curve is the cumulative percentage paid of the total cohort value at each month since registration (months on books).

The credit team think that newer cohorts have a higher repayment percent.

I am interested in;

1. Understanding and Breaking Down Operations using the Data.
2. Building Cohort Repayment Curve to track different customer segments.
3. Testing the Statistical Significance of differences between cohorts.

Methodology

This data was provided by company representative during an online assessment. This is Dummy data and does not contain any identifiable information (Get data here). There are 4 csv files in total namely:

`Payment` : Payment data from all customers with accountid.

`PaymentPlan` : Payment plans for the different products with data on initial deposit, loan terms and Daily amount and Total value.

`Account` : Customer Registration Data and Account Id

`Customer` : Customer information and Demographics

Load Libraries

```
library(dplyr)
library(tidyverse)
library(gridExtra)
library(ggplot2)
```

Import Data

```
Account_df <- read.csv("E:/Documents/BI/Data/Account.csv") %>%
  mutate(RegistrationDate = as.Date(RegistrationDate))

Customer_df <- read.csv("E:/Documents/BI/Data/Customer.csv")

Payment_df <- read.csv("E:/Documents/BI/Data/Payment.csv")%>%
  mutate(ReceivedWhen = as.Date(ReceivedWhen))

PaymentPlan_df <- read.csv("E:/Documents/BI/Data/PaymentPlan.csv")
```

Preview Data

```r
head(Account_df,5)
```

	AccountId	RegistrationDate	CustomerId	PaymentPlanId
1	5000	2020-03-30	4720	63
2	5002	2020-06-06	2674	63
3	5003	2020-02-28	2495	69
4	5007	2020-02-20	1749	37
5	5010	2020-09-03	2905	20

Table 1: Preview of Account Dataframe

```r
head(Customer_df,5)
```

	CustomerId	FirstName	LastName	Region
1	1000	Obinna	Mbori	mombasa
2	1003	Frank	Nyakwea	kisumu
3	1004	Victor	Nyakwea	nairobi
4	1005	Brian	Mbori	kisumu
5	1006	Mercy	Muguku	kisumu

Table 2: Preview of Customer Dataframe

```r
head(Payment_df,5)
```

	PaymentId	Amount	ReceivedWhen	AccountId	PaymentType
1	1000	125.96078	2020-09-13	6717	DailyPayment
2	1001	87.66168	2020-02-01	8804	DailyPayment
3	1009	61.50524	2021-01-15	5735	DailyPayment
4	1013	131.91756	2021-07-12	6837	DailyPayment
5	1017	100.77060	2020-09-24	5457	DailyPayment

Table 3: Preview of Payment Dataframe

```r
head(PaymentPlan_df,5)
```

	PaymentPlanId	Product	DailyValue	LoanTerm	Deposit	TotalValue
1	10	tv	75	200	1125	16125
2	11	phone	35	150	175	5425
3	12	phone	25	300	500	8000
4	13	solar	45	200	1125	10125
5	15	solar	50	200	1250	11250

Table 4: Preview of Account Dataframe

Part 1: Understanding and Breaking Down Operations using the Data.

Customer Demographics: Where are the most customers found?

```
Customer_df %>%
  group_by(Region)%>%
  summarise(customers = n())%>%
  ggplot(aes(x=customers, y=Region, fill=Region))+
  geom_bar(stat = 'identity') + theme_minimal()+
  theme(legend.position="none")+
  labs(title = "Customer Segmentation by Region",
       x= "Total Customers",y= "Region")+
  geom_text(aes(label = customers), nudge_x = 15)
```

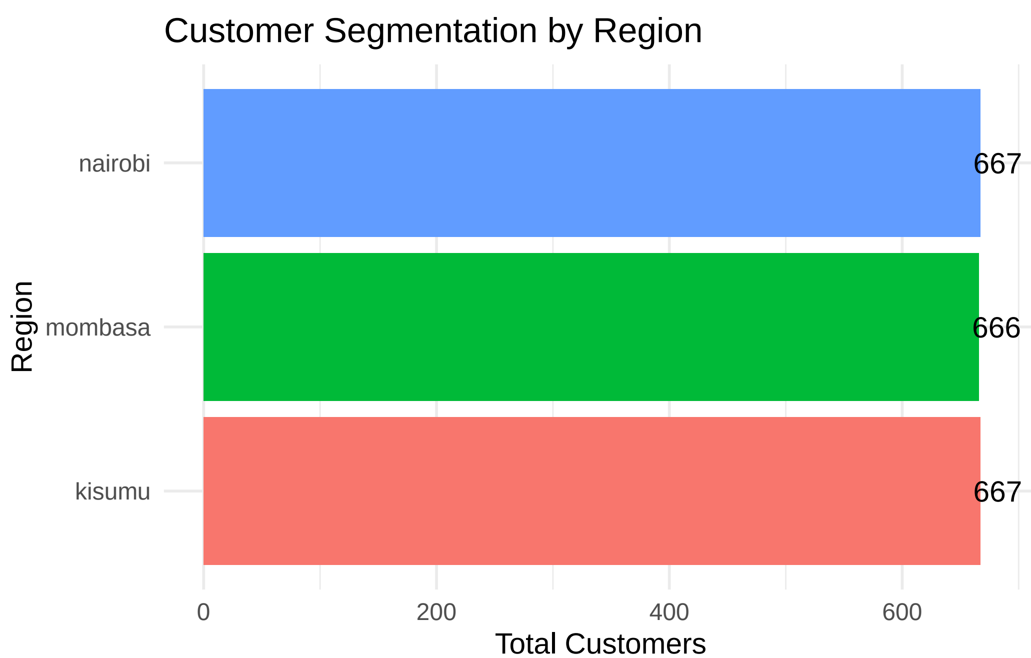

Figure 1: Customer Segmentation by Region

Insight 1: Balanced approach to different markets.

As shown in Figure 1 above, M-KOPA has an almost equal customer distribution in all three (3) markets available in this dataset. This indicates a balanced approach to marketing and customer onboarding across all three (3) regions.

Customer Demographics: Evolution of customer registration across Region

```
Account_df %>%
  left_join(Customer_df, by=join_by(CustomerId))%>%
  group_by(Region,
           month = lubridate::floor_date(RegistrationDate, 'month'))%>%
  summarise(CustomerRegistered=n(),
            .groups = 'keep')%>%
  ggplot( aes(x=month,
              y=CustomerRegistered,
              colour=Region)) +
  geom_line()+theme_minimal()+
  labs(title="Customer Registration over Time",y='# of Customer Registrations')
```

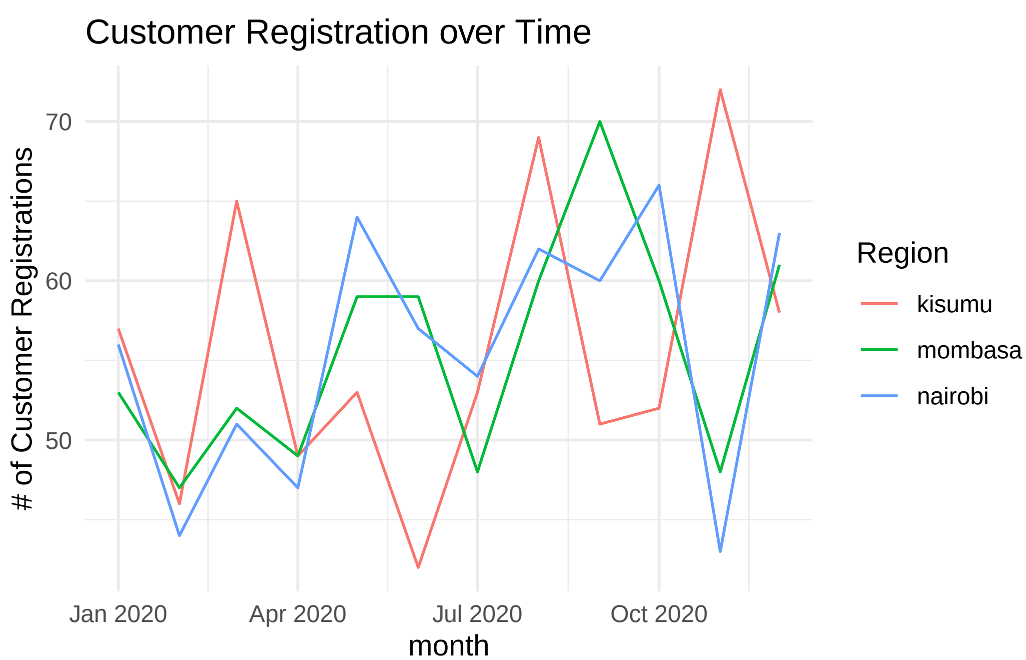

Figure 2: Evolution of customer registration across Region over time

The number of registration of customers in different regions vary over the time. There seem to be a very random trend per each region. However, across the months, there seem to be mostly an increase in the maximum registered region i.e. the region with maximum registration for a particular month is mostly higher than the maximum registration of the previous month.

```r
Account_df %>%
  left_join(Customer_df, by=join_by(CustomerId))%>%
  group_by(month = lubridate::floor_date(RegistrationDate, 'month'), Region)%>%
  summarise(CustomerRegistered=n(), .groups = 'keep')%>%
  mutate('max_month'= max(CustomerRegistered))%>%
  ungroup()%>%
  filter(CustomerRegistered==max_month)%>%
  select(month,Region, max_month)%>%
  arrange(desc(max_month))%>%
  head(10)
```

```
# A tibble: 10 x 3
   month      Region  max_month
   <date>     <chr>       <int>
 1 2020-11-01 kisumu         72
 2 2020-09-01 mombasa        70
 3 2020-08-01 kisumu         69
 4 2020-10-01 nairobi        66
 5 2020-03-01 kisumu         65
 6 2020-05-01 nairobi        64
 7 2020-12-01 nairobi        63
 8 2020-08-01 nairobi        62
 9 2020-12-01 mombasa        61
10 2020-08-01 mombasa        60
```

Table 4: Maximum Customer Registration per each month and the associated Region

Insight 2: Randomness in Customer Registrations in Different Regions

This could be due to lack of employees as M-KOPA is a start-up and might possibly not have independent recruiters for different regions. Onboarding the underbanked is a very face-to-face process, online ads, google ads and youtube ads are not possible means of reaching the target. However, more information about the company beyond the data is need to fully explain the randomness.

Product Analysis: What is the average cost of the products ?

```
PaymentPlan_df %>%
  group_by(Product)%>%
  summarise(avg_dailyValue = mean(DailyValue), avg_deposit = mean(Deposit),
            avg_loanterm = mean(LoanTerm), avg_Totalvalue = mean(TotalValue))
```

```
# A tibble: 3 x 5
  Product avg_dailyValue avg_deposit avg_loanterm avg_Totalvalue
  <chr>            <dbl>       <dbl>        <dbl>          <dbl>
1 phone               29         315          210           6340
2 solar               54        1218.         260          15518.
3 tv                  68        1350          330          23700
```

Table 5: Average Daily Value, Avg Deposit, Avg Loan Term, Avg Total Value per Product.

On average, phone cost the lowest compared to the other two products. This trend is the same for Daily Value, Deposit, Loan Term and Total Value.

Product Analysis: What is the most popular payment plans for each product ?

```
PaymentPlan_df %>%
  inner_join(Account_df, by=join_by(PaymentPlanId))%>%
  filter(Product=='phone')%>%
  group_by(PaymentPlanId)%>%
  summarise(CustomerRegistered = n())%>%
  arrange(desc(CustomerRegistered))%>%
  mutate(PaymentPlanId=as.character(PaymentPlanId))%>%
  ggplot(aes(x=reorder(PaymentPlanId,-CustomerRegistered),
             y=CustomerRegistered))+
  geom_bar(stat='identity', fill='forestgreen')+theme_minimal()+
  labs(title = "Customer Registration by Payment Plan (Phone Only)",
       x = "Payment Plan",
       y = "# of Customer Registration") +
  geom_text(aes(label = CustomerRegistered), nudge_y = -5, colour='white')
```

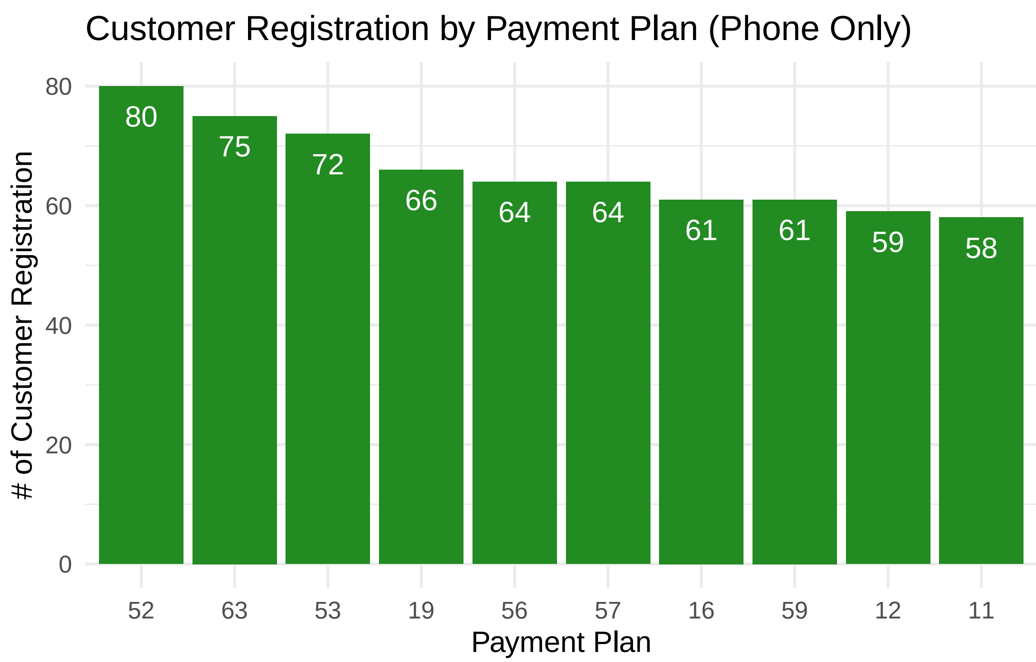

Figure 3: Customer Registration by Payment Plan for Phone.

As seen in Figure 3, Payment Plan 52, 63, 53 are the most popular payment plans for the customers who registered for a phone product. The least popular are Payment Plan 12 and 11. To find out if there is any connection between the customer registration and other variables such as Total Value and Deposit of the Payment Plan, we have displayed these information in Table 6 below.

```
PaymentPlan_df %>%
  inner_join(Account_df, by=join_by(PaymentPlanId))%>%
  filter(Product=='phone')%>%
  group_by(PaymentPlanId)%>%
  summarise(CustomerRegistered = n())%>%
  arrange(desc(CustomerRegistered))%>%

  # joining to Payment Plan to get information about payment Plan
  left_join(PaymentPlan_df, by=join_by(PaymentPlanId))%>%

  # removing Product and Customer Registered from result
  select(-Product, -CustomerRegistered)
```

```
# A tibble: 10 x 5
```

```
   PaymentPlanId DailyValue LoanTerm Deposit TotalValue
           <int>      <int>    <int>   <int>      <int>
 1            52         25      150     125       3875
 2            63         25      300     250       7750
 3            53         25      200     500       5500
 4            19         35      300     175      10675
 5            56         35      150     700       5950
 6            57         25      200     125       5125
 7            16         30      150     300       4800
 8            59         30      200     300       6300
 9            12         25      300     500       8000
10            11         35      150     175       5425
```

Table 6: Payment Plans for Phones arranged in Descending order of Customer Registration

There seem to be no customer preference related to Total Value or Deposit or Daily Value. This might be due to limitation in the data which does no provide the exact phone (product). In other words, we are unsure where these Payment Plans are for X number of phones. However, for phones it seem that the payment plan with the lowest Total value has the most customer registration.

```
tv_plot <- PaymentPlan_df %>%
  inner_join(Account_df, by=join_by(PaymentPlanId))%>%
  filter(Product=='tv')%>%
  group_by(PaymentPlanId)%>%
  summarise(CustomerRegistered = n())%>%
  arrange(desc(CustomerRegistered))%>%
  mutate(PaymentPlanId=as.character(PaymentPlanId))%>%
  ggplot(aes(x=reorder(PaymentPlanId,-CustomerRegistered),
             y=CustomerRegistered))+
  geom_bar(stat='identity', fill='darkorange2')+theme_minimal()+
  labs(title = "Customer Registration by Payment Plan (TV Only)",
       x = "Payment Plan",
       y = "Customer Registration")+
  geom_text(aes(label = CustomerRegistered), nudge_y = -5, colour='white')


solar_plot <- PaymentPlan_df %>%
  inner_join(Account_df, by=join_by(PaymentPlanId))%>%
  filter(Product=='solar')%>%
  group_by(PaymentPlanId)%>%
  summarise(CustomerRegistered = n())%>%
```

```
  arrange(desc(CustomerRegistered))%>%
  mutate(PaymentPlanId=as.character(PaymentPlanId))%>%
  ggplot(aes(x=reorder(PaymentPlanId,-CustomerRegistered),
             y=CustomerRegistered))+
  geom_bar(stat='identity', fill='darkslategrey')+theme_minimal()+
  labs(title = "Customer Registration by Payment Plan (Solar Only)",
       x = "Payment Plan",
       y = "Customer Registration")+
  geom_text(aes(label = CustomerRegistered), nudge_y = -5, colour='white')


par(mfrow=c(10,1))
grid.arrange(solar_plot, tv_plot)
```

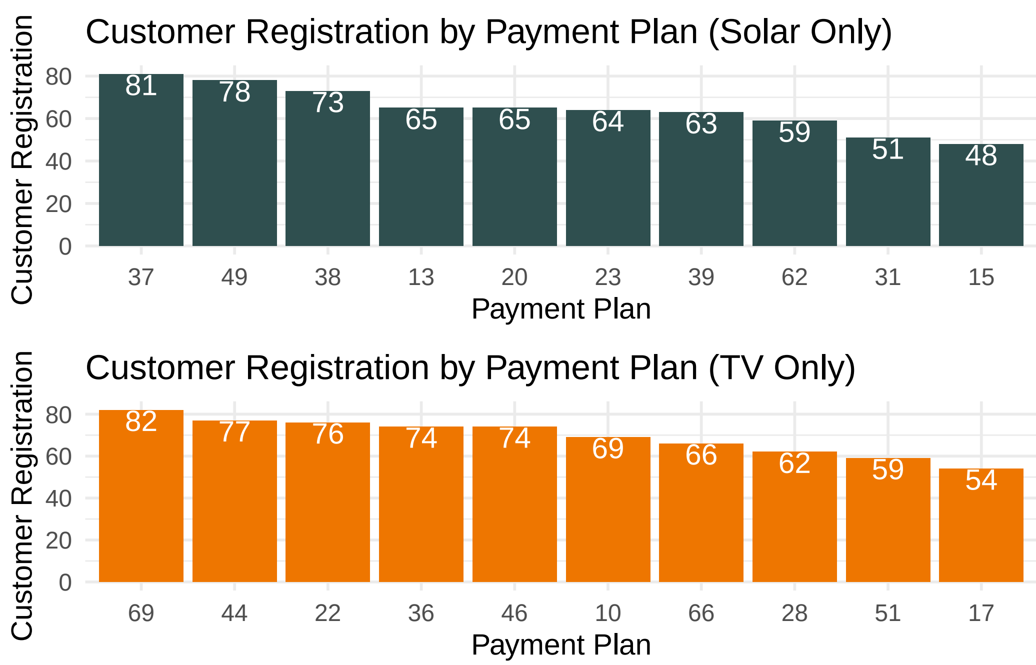

Figure 4: Customer Registration by Payment Plan for Solar & TV.

```
PaymentPlan_df %>%
  inner_join(Account_df, by=join_by(PaymentPlanId))%>%
  filter(Product=='tv')%>%
  group_by(PaymentPlanId)%>%
  summarise(CustomerRegistered = n())%>%
```

```r
  arrange(desc(CustomerRegistered))%>%

  # joining to Payment Plan to get information about payment Plan
  left_join(PaymentPlan_df, by=join_by(PaymentPlanId))%>%

  # removing Product and Customer Registered from result
  select(-Product, -CustomerRegistered)
```

```
# A tibble: 10 x 5
   PaymentPlanId DailyValue LoanTerm Deposit TotalValue
           <int>      <int>    <int>   <int>      <int>
 1            69         55      400    1375      23375
 2            44         55      300    1100      17600
 3            22         75      400    1125      31125
 4            36         75      300    1875      24375
 5            46         75      300    1125      23625
 6            10         75      200    1125      16125
 7            66         75      400    1875      31875
 8            28         65      400    1625      27625
 9            51         65      200     975      13975
10            17         65      400    1300      27300
```

Table 7: Payment Plans for TV arranged in Descending order of Customer Registration

There seem to be no clear customer preference based on Total Value, Deposit, Loan Value in Table 7 above. However it looks like the top 2 most registered Payment Plans for TVs have the lowest Daily Value.

```r
PaymentPlan_df %>%
  inner_join(Account_df, by=join_by(PaymentPlanId))%>%
  filter(Product=='solar')%>%
  group_by(PaymentPlanId)%>%
  summarise(CustomerRegistered = n())%>%
  arrange(desc(CustomerRegistered))%>%

  # joining to Payment Plan to get information about payment Plan
  left_join(PaymentPlan_df, by=join_by(PaymentPlanId))%>%

  # removing Product and Customer Registered from result
  select(-Product, -CustomerRegistered)
```

```
# A tibble: 10 x 5
   PaymentPlanId DailyValue LoanTerm Deposit TotalValue
           <int>      <int>    <int>   <int>      <int>
 1            37         45      200     675       9675
 2            49         45      400     900      18900
 3            38         65      200    1625      14625
 4            13         45      200    1125      10125
 5            20         65      400    1625      27625
 6            23         65      400    1300      27300
 7            39         50      200    1250      11250
 8            62         45      200    1125      10125
 9            31         65      200    1300      14300
10            15         50      200    1250      11250
```

Table 8: Payment Plans for TV arranged in Descending order of Customer Registration

Just like the other Payment Plans, there seem to be no clear customer preference. It should be noted that the Total Value for the most registered has the least Total Value, Deposit, Loan Term & Daily Value. However, there are no noticeable pattern in the popularity of the other payment plans.

Insight 3: No Clear Customer Preference for Choosing Payment Plans.

Generally, customers register to payment plans with no clear preference for Total Value, Deposit, Loan Term & Daily Value. To uncover more patterns we will need to create a correlation matrix.

Product Analysis: Customer registration by different products

```
Account_df%>%
  #merge account to PaymentPlan to get Products attached to each AccountId(every customer
  left_join(PaymentPlan_df, by= join_by(PaymentPlanId))%>%
  select(Product,
         RegistrationDate) %>%
  mutate(RegistrationDate= as.Date(RegistrationDate)) %>%


  group_by(Product, month = lubridate::floor_date(RegistrationDate, 'month'))%>%
  summarise(CustomersReg = n(), .groups = "keep" )%>%
  ggplot( aes(x=month, y=CustomersReg,colour = Product)) +
  geom_line(size=1) + theme_minimal() +
  geom_text(aes(label = CustomersReg), nudge_y = -4, check_overlap = TRUE)+
  labs(y='Customer Registrations')
```

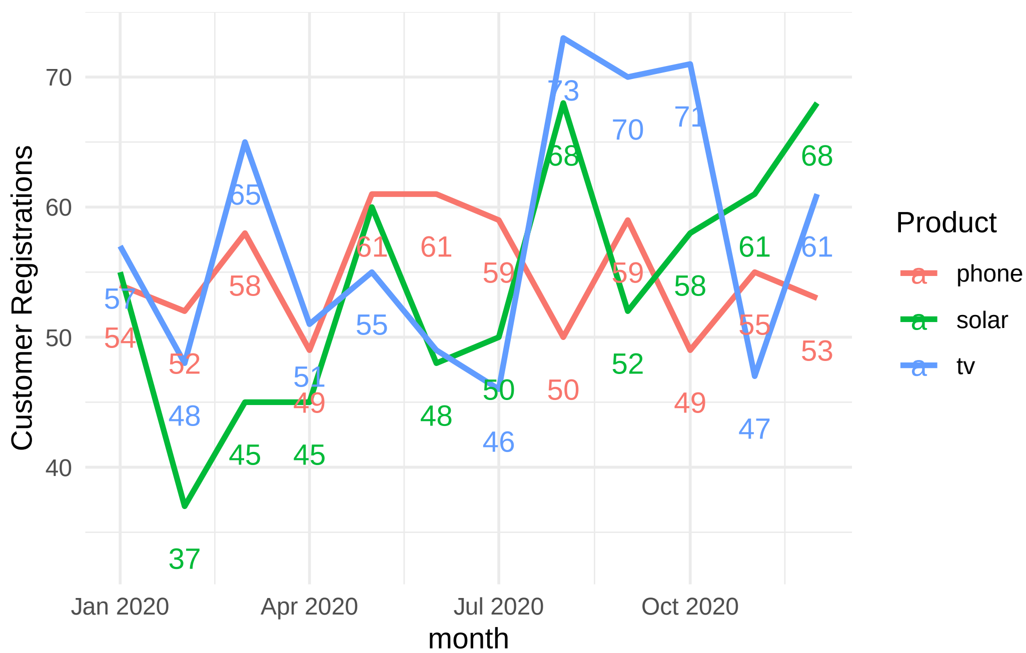

Figure 5: Customer Registration by Products.

Insight 4: Customer Registrations has risen since July 2020.

TV and Solar have risen in popularity with Phone staying relatively the same. Both TV and Solar peaked at 73 & 68 respectively in the month of August 2020. phone registration peaked in March 2020 at 65.

Product Analysis: Total Repayment per Product

Table 9: Repayment Percentage for Products

```
# a nested code
#first part is to get the Total Credit (Total Value of all Products) grouped by Products

PaymentPlan_df %>%
  inner_join(Account_df, by="PaymentPlanId")%>%
  group_by(Product)%>%
  summarise(TotalCredit= sum(TotalValue))%>%

# second part is Total Amount repaid by customers also grouped by Products

  left_join(
    Payment_df%>%
      inner_join(Account_df, by='AccountId')%>%
      left_join(PaymentPlan_df, by='PaymentPlanId')%>%
      group_by(Product)%>%
      summarise(TotalPaid = sum(Amount)),
    by = "Product")%>%

# finally we calculated % repayment

mutate("Repayment %" = TotalPaid/TotalCredit)
```

```
# A tibble: 3 x 4
  Product TotalCredit TotalPaid `Repayment %`
  <chr>         <int>     <dbl>         <dbl>
1 phone       4164350  3231093.         0.776
2 solar      10101875  7821618.         0.774
3 tv         16417300 12540683.         0.764
```

Insight 5: Total Repayment for each product is around 77%

For all three products, Total Repayment is around 77% of the Total value. On average, customers have paid 70% of the Total Value of the product taken on credit.

Part 2: Building Cohort Repayment Curve to track different customer segments.

Customer cohort analysis is the act of segmenting customers into groups based on their shared characteristics, and then analyzing those groups to gather targeted insights on their behaviors and actions. This technique provides a way of understanding customer trends, which aids an organization to better target its audience, and make better business decisions.

Creating Cohort Table

Merging all tables into a single table. The idea is to have all products and regions associated with each payment.

Columns needed are:
Payment_df: **PaymentId** , **Amount** , **ReceivedWhen** , **AccountId**
PaymentPlan_df: **TotalValue**, **Product**
Customer_df: **Region**
Account_df: **RegistrationDate**

```r
Cohort_df <-Account_df%>%
  inner_join(Customer_df, by="CustomerId")%>%
  inner_join(PaymentPlan_df, by="PaymentPlanId")%>%
  inner_join(Payment_df, by="AccountId")%>%
  select(PaymentId, Amount, ReceivedWhen, AccountId,
         TotalValue, Product, Region, RegistrationDate)%>%

  #add Cohort column with month registered for each customer
  #add column for month interval between Registration Date and Payment Date
  mutate(Cohort = format(RegistrationDate, "%Y-%m"),
         MonthsAfter = interval(floor_date(ymd(RegistrationDate), 'month'),
                              floor_date(ymd(ReceivedWhen), 'month'))
         %/%months(1))%>%
  arrange(MonthsAfter)

#preview top 10 rows
head(Cohort_df,5)
```

```
  PaymentId    Amount ReceivedWhen AccountId TotalValue Product  Region
1    284771  28.47538   2020-03-31      5000       7750   phone nairobi
2    384960 284.75379   2020-03-30      5000       7750   phone nairobi
3     61965  42.29656   2020-06-17      5002       7750   phone  kisumu
4     88357  42.29656   2020-06-26      5002       7750   phone  kisumu
```

```
5     92084  42.29656   2020-06-07      5002       7750    phone   kisumu
  RegistrationDate  Cohort MonthsAfter
1       2020-03-30 2020-03           0
2       2020-03-30 2020-03           0
3       2020-06-06 2020-06           0
4       2020-06-06 2020-06           0
5       2020-06-06 2020-06           0
```

Table 10: All payment data with products, regions and cohorts.

We have all the data in a single Table and we have also created the various Cohorts for futher analysis.

As shown in Table 10, multiple payment received for one account duplicates Total Value, Product and Region. Now we have to group data by Cohort, Months After, AccountId, Region & Product.

```
Cohort_df <-Cohort_df %>%
  group_by(Cohort, MonthsAfter, AccountId, Region, Product)%>%
  summarise(TotalAmount = sum(Amount), TotalValue= max(TotalValue), .groups = "keep")

head(Cohort_df,5)
```

```
# A tibble: 5 x 7
# Groups:   Cohort, MonthsAfter, AccountId, Region, Product [5]
  Cohort  MonthsAfter AccountId Region  Product TotalAmount TotalValue
  <chr>         <dbl>     <int> <chr>   <chr>         <dbl>      <int>
1 2020-01           0      5077 nairobi tv            2324.      27300
2 2020-01           0      5099 kisumu  solar         1861.      18900
3 2020-01           0      5112 mombasa phone          282.      10675
4 2020-01           0      5121 nairobi solar         2112.      10125
5 2020-01           0      5134 nairobi tv            2893.      31125
```

Table 11: Cohort Table grouped by Cohorts, MonthsAfter, AccountId, Product

We are only taking the first unique value of the Total Value per each AccountId since each Account is linked to 1 Product & PaymentPlan i.e. There is a single TotalValue for any AccountId.

There are only 2000 unique accounts, and each account is linked with a total value for the product purchase. However our current table has 18364 rows. Therefore, summing the Total

Value will lead to inaccurate results.

```
Cohort_df$TotalValue[duplicated(Cohort_df$AccountId)]<- 0
Overall_Cohort <- Cohort_df%>%
  group_by(Cohort, MonthsAfter)%>%
  summarise(TotalAmount = sum(TotalAmount),
            TotalValue = sum(TotalValue),
            .groups = "keep")

head(Overall_Cohort,5)
```

```
# A tibble: 5 x 4
# Groups:   Cohort, MonthsAfter [5]
  Cohort  MonthsAfter TotalAmount TotalValue
  <chr>         <dbl>       <dbl>      <dbl>
1 2020-01           0     316689.    2498425
2 2020-01           1     225089.          0
3 2020-01           2     195116.          0
4 2020-01           3     156071.          0
5 2020-01           4     134534.          0
```

Table 12: Overall Cohort Table grouped by Cohorts & MonthsAfter.

The data is now grouped by Cohort & Month Afterand summed by Amount & TotalValue. The dataset has now been reduced to 156 rows. To accurately calculate the Percentage Paid we will need a Running Total of the Amount paid by the entire cohort & the entire Total Value for the Cohort.

```
Overall_Cohort<- Overall_Cohort %>%
  group_by(Cohort)%>%
  mutate(RunningAmount = cumsum(TotalAmount),
         RunningTotal = cumsum(TotalValue),
         PercentPaid = (RunningAmount/RunningTotal)*100)

head(Overall_Cohort,5)
```

```
# A tibble: 5 x 7
# Groups:   Cohort [1]
```

```
  Cohort  MonthsAfter TotalAmount TotalValue RunningAmount RunningTotal
  <chr>         <dbl>       <dbl>      <dbl>         <dbl>        <dbl>
1 2020-01           0     316689.    2498425       316689.      2498425
2 2020-01           1     225089.          0       541778.      2498425
3 2020-01           2     195116.          0       736895.      2498425
4 2020-01           3     156071.          0       892966.      2498425
5 2020-01           4     134534.          0      1027500.      2498425
# i 1 more variable: PercentPaid <dbl>
```

Table 13: Overall Cohort Table with Running Total of Amount Paid & Percent Paid

Creating Cohort Pivot Chart

```
cohorts.wide <- Overall_Cohort %>%
  select(Cohort,MonthsAfter, PercentPaid )%>%
  pivot_wider(
    names_from = "MonthsAfter",
    values_from = "PercentPaid"
  )

head(cohorts.wide,10)
```

```
# A tibble: 10 x 16
# Groups:   Cohort [10]
   Cohort    ‘0‘   ‘1‘   ‘2‘   ‘3‘   ‘4‘   ‘5‘   ‘6‘   ‘7‘   ‘8‘   ‘9‘  ‘10‘
   <chr>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 2020-01  12.7  21.7  29.5  35.7  41.1  45.4  49.1  51.6  53.6  55.3  56.5
 2 2020-02  12.0  22.1  30.0  36.8  42.2  46.7  50.3  52.8  55.0  56.9  58.3
 3 2020-03  13.8  23.8  32.3  39.1  44.7  49.4  53.0  55.6  57.8  59.7  61.2
 4 2020-04  13.2  24.0  32.4  39.6  45.5  50.3  54.5  57.4  59.8  62.0  63.4
 5 2020-05  14.9  25.4  34.4  41.6  47.3  52.2  56.4  59.3  61.7  63.6  65.3
 6 2020-06  20.8  35.9  47.9  56.8  64.0  69.6  73.9  77.1  79.3  81.4  82.8
 7 2020-07  22.5  38.8  50.6  60.1  67.2  72.7  76.9  79.3  81.4  83.1  84.3
 8 2020-08  21.1  36.1  48.1  57.2  64.4  70.3  74.5  77.8  80.2  82.5  84.2
 9 2020-09  20.4  35.9  47.6  57.1  64.7  70.1  75.0  78.4  81.2  83.5  85.1
10 2020-10  21.2  36.5  48.7  58.6  65.5  71.8  76.5  79.8  82.4  84.6  85.9
# i 4 more variables: ‘11‘ <dbl>, ‘12‘ <dbl>, ‘13‘ <dbl>, ‘14‘ <dbl>
```

Table 14: Pivot Chart showing Cohorts as Rows and Number of Months after Registration as Columns.

Displaying Cohort Chart

```
Overall_Cohort %>%
   ggplot(aes(MonthsAfter, reorder(Cohort,desc(Cohort)))) +
   geom_raster(aes(fill = log(PercentPaid))) +
   coord_equal(ratio = 1) +
   geom_text(aes(label = glue::glue("{round(PercentPaid,0)}%")),
                 size  = 3,
                 colour = "snow") +
  scale_fill_gradient(low = "brown2", high="darkgreen") +
  theme_minimal(base_size = 13) +theme(legend.position="none")+
  theme(panel.grid   = element_blank(),
        panel.border = element_blank())+
  labs(y = "cohort")
```

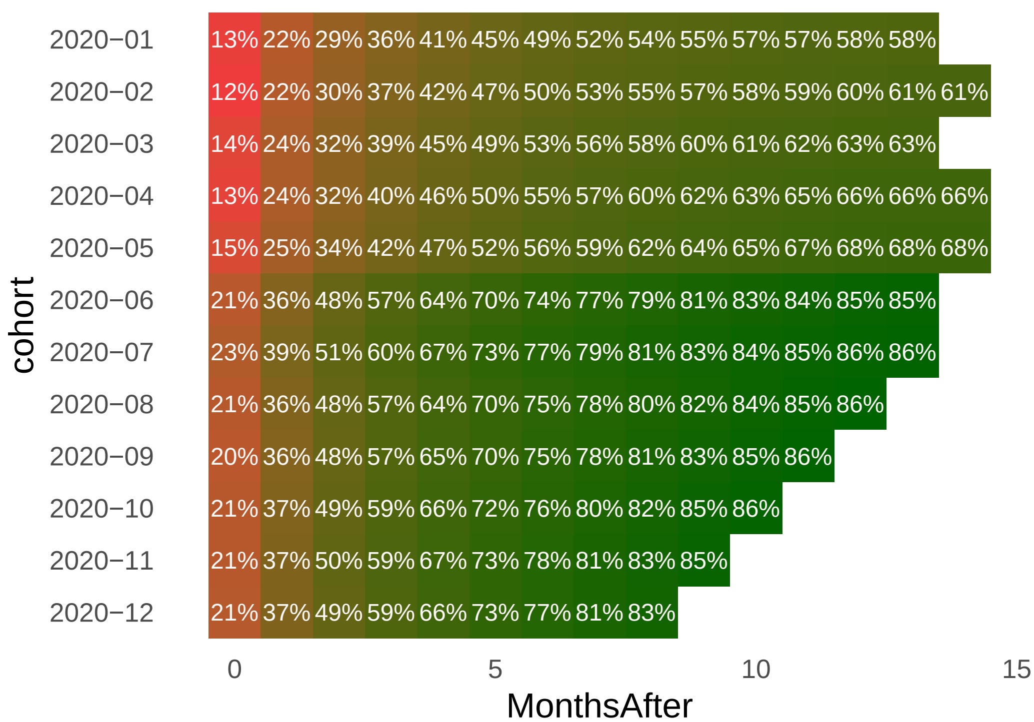

Figure 6: Cohort Repayment Chart

In Figure 6 presented above, a discernible enhancement in the percentage of repayment is evident among recent cohorts. As one progresses along the cohort axis, there is a notable decrease in the timeframe required to achieve an average repayment of 80% or more. For instance, in the initial cohort, only 58% of the Total Value was repaid after 13 months, while the subsequent cohort (2020-07) achieved an 86% repayment within the same period.

Insight 5: There are considerable differences in the repayment % between Cohorts.

There seem to be a clear improvement in repayment behaviour recent cohorts. The biggest change happened between Cohort 2020-05 and 2020-06. For the purpose of decision making, we can look further at Cohorts per Region or Cohorts per Product for more in-depth insights. However, for this report, we are interested in the overall difference.

Part 3: Statistical Significance of Differences between Cohorts

Is the Difference between 2020-01 cohort and 2020-06 cohort significant?

```
Cohort_df%>%
  filter(Cohort == '2020-01'|Cohort=='2020-06')%>%
  group_by(Cohort, Product,)%>%
  summarise(n(), .groups = 'keep')%>%
  ggplot(aes(fill=Product,x=Cohort ,y= `n()`, label=`n()`))+
  geom_bar(position='dodge',stat='identity')+
  theme_minimal()+labs(y='Number of Registrations')
```

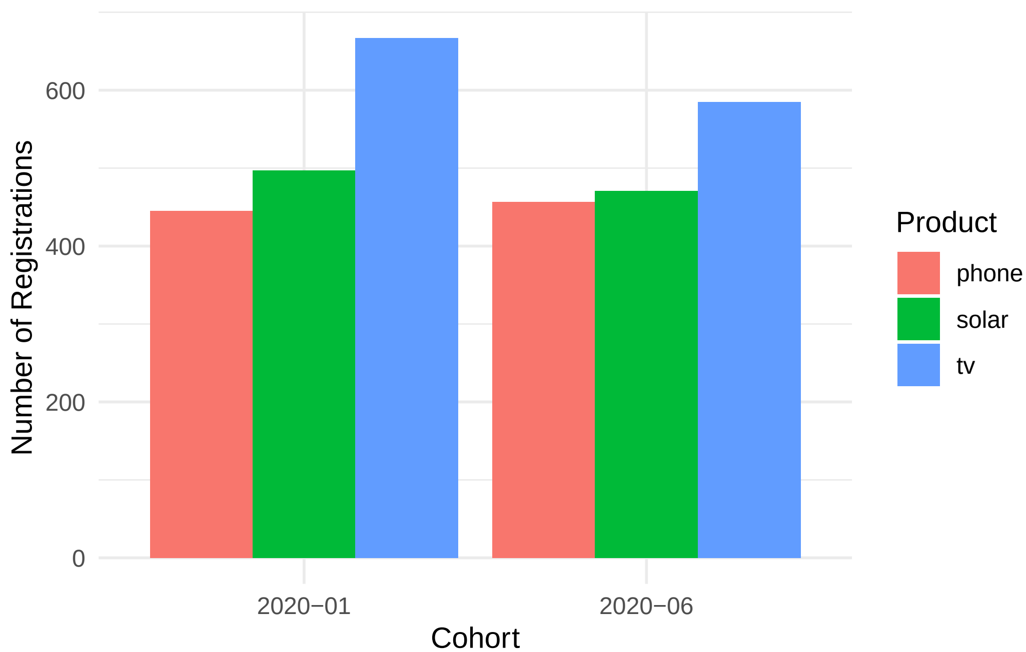

Figure 7: Stack Bar Chart of Customer Registration for both Cohorts subdivided into Products.

The distribution patterns appear comparable across both cohorts. However, the 2020-06 cohort exhibits a lower overall registration volume compared to the 2020-01 cohort. Furthermore, there is a noticeable decrease in registrations specifically for TVs, which represent the highest-value product. It seems counterintuitive that a cohort with fewer registrations would exhibit a higher likelihood of making more monthly payments.

Hypothesis Testing

- **Null Hypothesis: There is no difference in Repayment Received from Customers in Cohort 2020-01 & Cohort 2020-06**
- **Alternative Hypothesis: There is a difference in Repayment Received from Customers in Cohort 2020-01 & Cohort 2020-06**

Compute summary statistics by group

```
Cohort_test_df <- Cohort_df%>%
  filter(Cohort == '2020-01'|Cohort=='2020-06')%>%
  ungroup()%>%
  select(Cohort, TotalAmount)

Cohort_test_df%>%
  group_by(Cohort)%>%
  summarise( count = n(),mean = mean(TotalAmount, na.rm = TRUE),
             sd = sd(TotalAmount, na.rm = TRUE))
```

```
# A tibble: 2 x 4
  Cohort  count  mean    sd
  <chr>   <int> <dbl> <dbl>
1 2020-01  1609  908.  695.
2 2020-06  1513 1345. 1212.
```

Table 15: Summary Statistics of Cohorts

```
#uncomment to install packages
#install.package('hrbrthemes')
#install.package('viridis')

library(hrbrthemes)
library(viridis)



Cohort_test_df%>%
  ggplot( aes(x=Cohort, y=TotalAmount, fill=Cohort)) +
    geom_boxplot() +
    scale_fill_viridis(discrete = TRUE, alpha=0.6) +
    geom_jitter(color="black", size=0.4, alpha=0.9) +
    theme_ipsum() +
```

```r
    theme(
      legend.position="none",
      plot.title = element_text(size=11)
    )+theme_minimal()+labs(y='Amount')
```

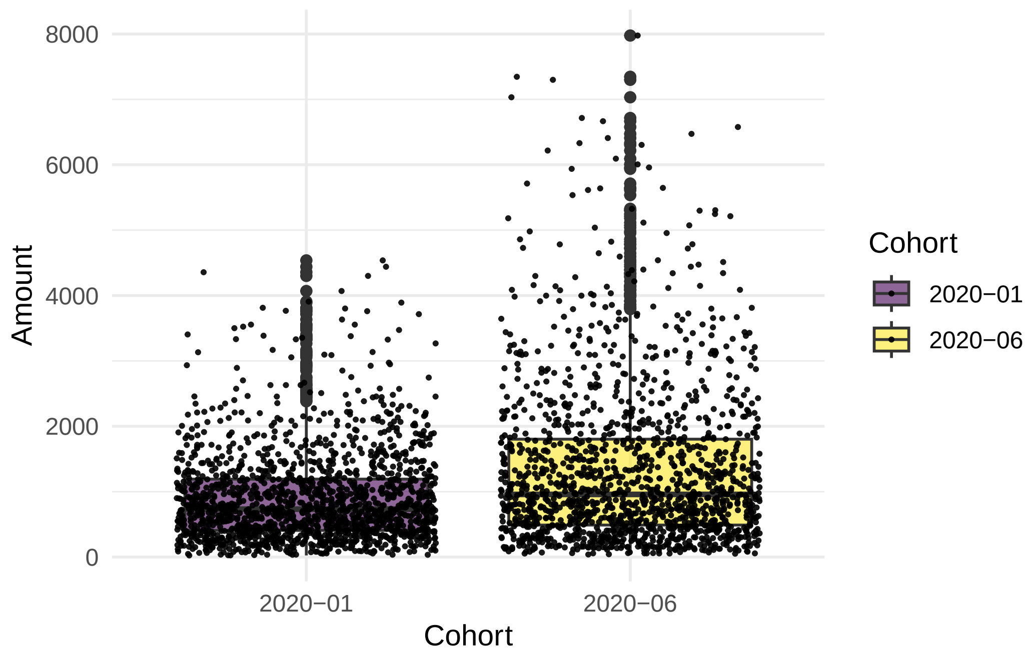

Figure 8: Box Plot of Amount Repaid by Cohort

Sampling into 500 observations for each Cohort

```r
Cohort01 <-Cohort_test_df%>%
  filter(Cohort=='2020-01')

Cohort06 <-Cohort_test_df%>%
  filter(Cohort=='2020-06')

# Appending Sampled Dataframes into A new DataFrame
Cohort_sample <- rbind(Cohort01[sample(nrow(Cohort01), size=500), ],
                       Cohort06[sample(nrow(Cohort06), size=500), ])
```

Performing T-Test with difference in variance

```r
res <- t.test(TotalAmount ~ Cohort, data = Cohort_sample, var.equal = FALSE)
res
```

```
    Welch Two Sample t-test

data:  TotalAmount by Cohort
t = -6.4726, df = 765.39, p-value = 1.72e-10
alternative hypothesis: true difference in means between group 2020-01 and group 2020-06 is n
95 percent confidence interval:
 -547.2351 -292.5407
sample estimates:
mean in group 2020-01 mean in group 2020-06
             923.6463             1343.5342
```

There is a statistically significant difference in the two groups leading to the rejection the null hypothesis.

Insight 6: There is statistical difference between Cohorts 2020-01 & Cohort 2020-06

The choice of statistical sampling test between Cohorts 2020-10 and Cohort 2020-06 was influenced by the same number of Months on Book of both Cohorts (13 months). Both Cohorts were randomly sampled to 500 observations each. We are 95% confident that the mean of Cohort 2020-01 is 308 - 555 less than the mean of Cohort 2020-06.

Conclusion & Potential Improvements

We conducted a thorough examination of M-KOPA's business processes by delving into the intricacies of the provided dummy data. While the data's simplification facilitated a straightforward analysis, it presented limitations, notably the absence of scenarios where customers purchased multiple items. This deviation from realistic consumer behavior, wherein customers commonly acquire more than one product, warrants consideration.

Our primary focus centered on investigating disparities among cohorts, defined as customers registering for a product in the same month. Notably, we identified statistically significant distinctions in credit repayment within recent cohorts, which could be classified into two distinct clusters: January 2020 to May 2020 and June 2020 to December 2020.

The discernible variations between these clusters appear linked to the introduction of a remote lock feature for products in case of default. Unfortunately, the information available on the M-KOPA website lacks clarity regarding the precise implementation timeline of this feature. It is crucial to acknowledge that the dummy data provided may not necessarily reflect real-world correlations.

To further explore relationships, we propose conducting correlation analyses utilizing data from the Payment and PaymentPlan tables. Specifically, we aim to assess whether customer registration exhibits correlations with TotalValue, Deposit, and LoanTerm, thereby offering insights into potential patterns within M-KOPA's operational dynamics.