

MATP-4910 Final Project Notebook

DAR Project DeFi

Soumya Mishra

15 November 2021

Contents

Final Project: Github Info	1
Overview & Problems Tackled	1
Data Description	2
Results	2
Problem 1	4
Problem 2	7
Summary and Recommendations	11
References	11
Appendix	11

Final Project: Github Info

- github repository:
- Your github ID: mishrs4
- Final notebook: *mishrs4_final_assignment.Rmd*
- Summary of github contributions including github issues addressed.

I committed the generateSurvival.R file that includes the function to perform survival analysis given 2 data sets such as borrow to repay or any two transactions. This was added to the shiny app with the option to analyze various transaction data sets. The issues I resolved include:

- create a notebook with survival analysis at the coin level #81
- Survival Analysis function #87

Overview & Problems Tackled

The Defi project involves transactions on various coins. The data set includes transactions such as borrows, repays, deposits, redeems, swaps, and liquidations. A borrow transaction is when a user is taking a form of a loan from the Defi pool for a specific coin. Each borrow is classified with a borrowRate and the mode of borrowRate as either variable or stable. The reserve, i.e. the coin, is classified as AMM, Stable, or Non-stable. These are classifications of the coins. The repay transaction is when a user is repaying the loan or the borrowed amount. The deposit transaction is when a user deposits an amount into a specific pool while a

redeem is when the user redeems the amount in their account. A user can deposit to earn an APY or interest on their account. Depositing to earn an APY is a more passive investment strategy than borrowing and investing in Defi coins that seem to be increasing in value which is active investing. This is one of the main aspects of the data I chose to analyze and compare to traditional finance with active and passive investment strategies.

I performed survival analysis based on the timestamps of these transactions. Survival analysis is a method that uses the Kaplan-Meier estimator which helps measure the percentage for the probability of an event occurring. In the case of my analysis I used one transaction and measured the probability of another transaction not occurring. For example, the analysis of borrows to repays takes into account when a loan is taken out by a user and then the chance of that loan “surviving” or not being repaid is measured in various survival graphs. I also further ran this analysis on different sets of coins such as stable and non-stable coins.

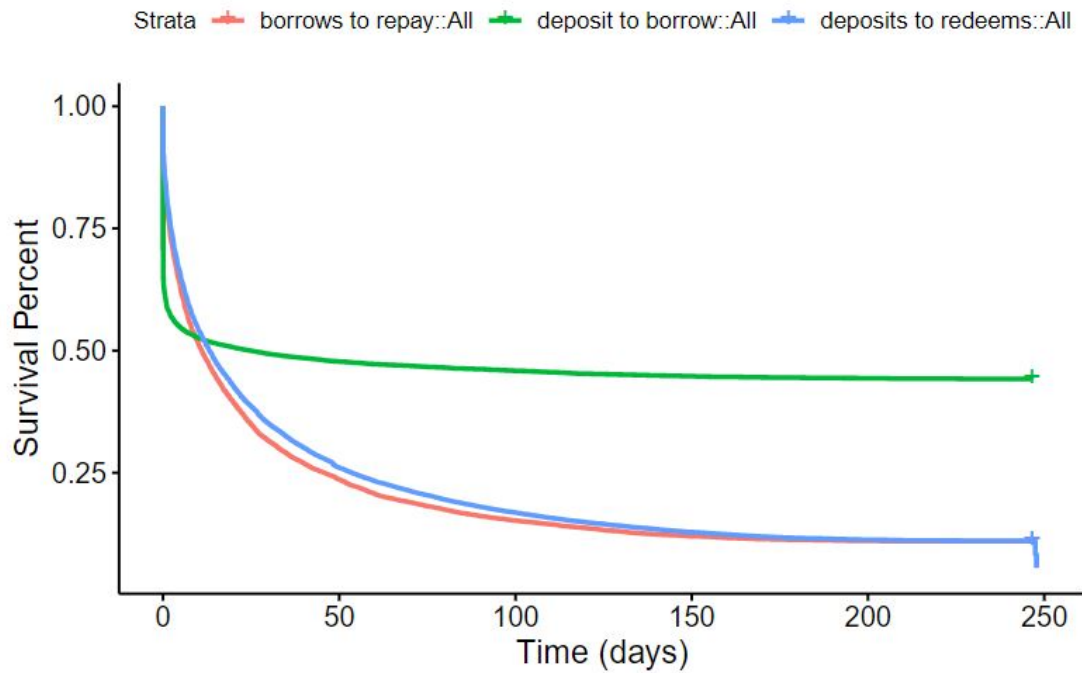
Data Description

The data set included all the transactions by users. Each transaction would have the following relevant information the amount, onBehalfOf, reserve, timestamp, user, type. I used data sets for each of the different transaction types where there are 94977 borrow transactions, 192006 deopsit transactions, 126705 redeem transactions, and 60542 repay transactions. Each of these transaction types has various coins and users that made the transaction. I used these data sets of different transactions in analyzing the timestamps of each transaction for a given user and the time it took for the user to complete another corresponding transaction. I further broke down this analysis as I separated the dataset of a given transaction, borrows and repays, to analyze the survival of a given coin type. In this case I analyzed the stable and non-stable coins. So instead of the entire dataset of borrows, I had a dataset of borrows for stable coins and one for non-stable coins. Likewise with repays, the survival analysis was conducted on the transaction types like before, however this time on specific types of coins. A transaction not included in my survival analysis was liquidation for which I used in a density plot of time to liquidation of the small percentage of the Defi coins that undergo this process. .

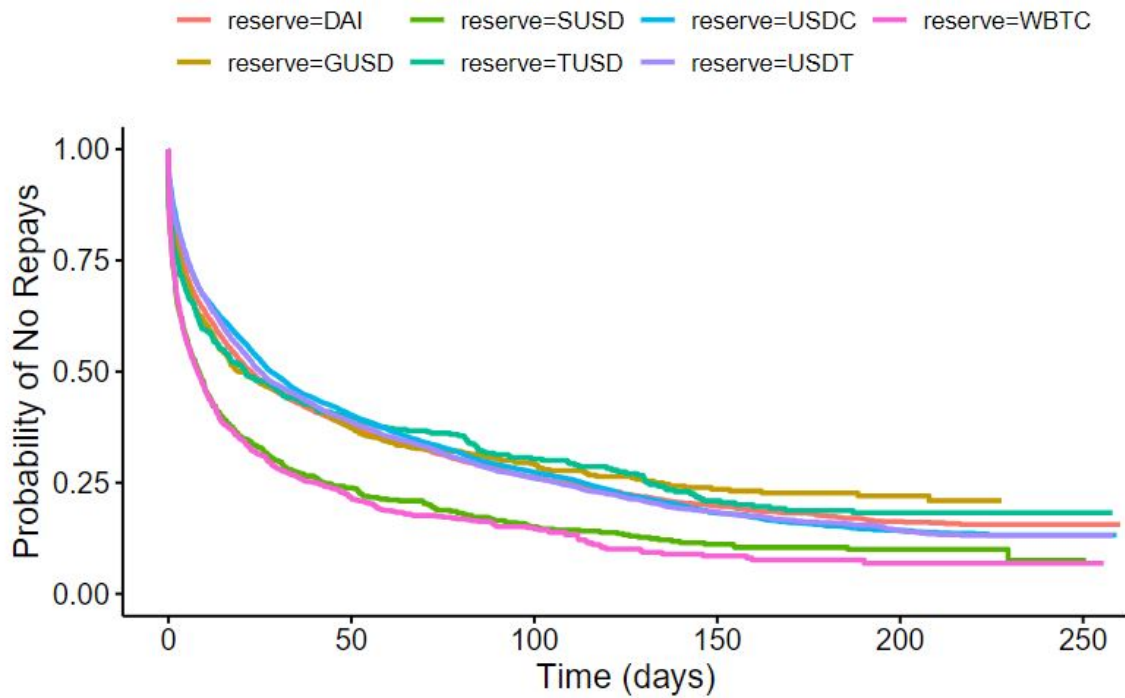
Results

The following are my survival graphs of the different data sets I analyzed. They utilized the generateSurvival function I wrote which resolved GitHub Issue #87. The results are further discussed in detail below.

Survival Analysis of Transaction Types



Borrow to Repay of Top Stable Coins



Time until Loans are Repaid in Stable and Non-stable Coins?

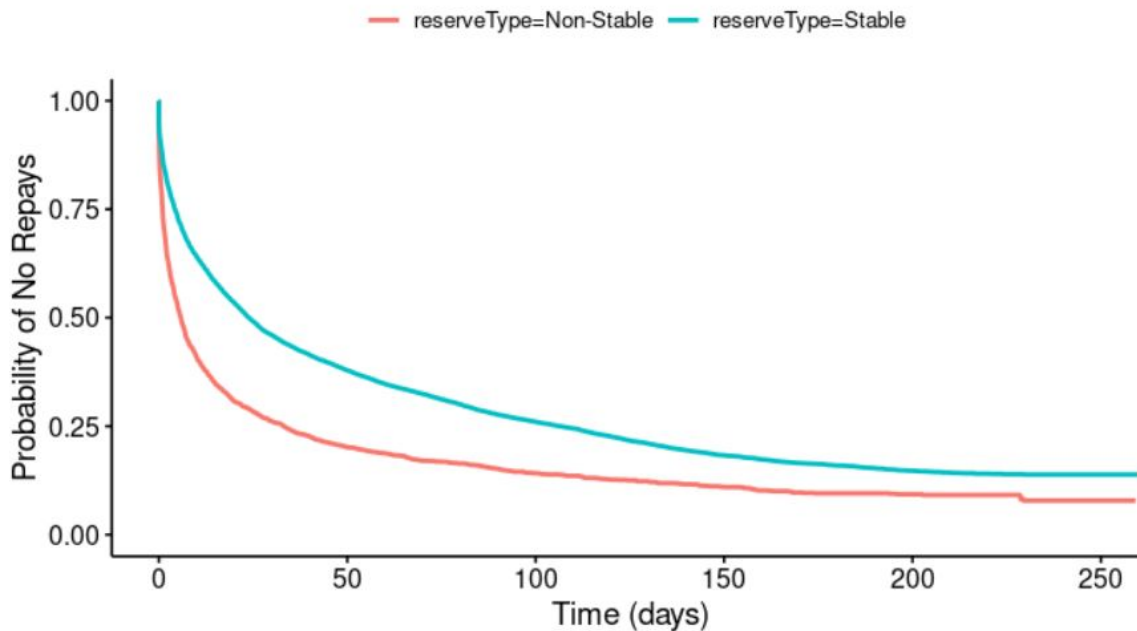


Figure 1: Specific Coin Survival.

Problem 1

What sort of investors are the users in the Defi pool? I am looking at the transaction types and their survivals to determine whether the users generally follow a more active investment strategy or a more passive investment strategy. The survival curve of borrows to repays would indicate the survival or likeliness of a user repaying their borrowed amount. Borrows to repays generally indicate a more active investment style as the user is actively choosing to take out a loan in order to try to make a profit and use future funds to repay the original loan. In contrast, a passive investment strategy would be the survival curve of deposits to redeems. This transaction type indicates a user would like to deposit funds in an account to accumulate a small APY or interest and then eventually redeem from the account. It is indicative of a passive investment strategy. This is an overall analysis on the investment strategies within the Defi transactions, but furthermore it shows how borrows to repays are the most illuminative for analysis of individual coins. An active investment strategy depends on the coin type and thus analyzing the transaction time of borrows to repays show user hesitancy in repaying a specific coin pool.

Methods

I created a `generateSurvival` function that takes in 2 data sets. I used this function for the various transaction types so in the code below for the function what is occurring is I am censoring the data by taking the minimum `startTime` which is the timestamp of the starting transaction for a given user and then the minimum `endTime` of the transaction I am calculating survival until. This data is then passed into the `survfit` method that takes in the kaplan meier statistic to calculate and return a model that fits the data

```
generateSurvival <- function(start, end)
{
  funcTable <- left_join(end,start,by="user") %>%
  dplyr::rename(endTime=timestamp.x) %>%
```

```

dplyr::rename(startTime=timestamp.y) %>%
group_by(user) %>%
dplyr::summarise(timeDiff=case_when(min(endTime)-min(startTime)>0 ~ min(endTime)-min(startTime), TRUE
mutate(status=case_when(timeDiff==as.integer(21294796) ~ 0, timeDiff<=0 ~ 0, timeDiff>0 ~ 1)) %>%
select(user,timeDiff,status)

km <- with(funcTable, Surv(timeDiff/86400, status))
km_fit <- survfit(Surv(timeDiff/86400, status) ~ 1, data=funcTable)
summary(km_fit, times = c(1,30,60,90*(1:10)))
#p1 <- ggplot(km_fit)
return (km_fit)

}

```

I used the survminer package which includes the function Surv() and survfit(). The Surv function calculates the Kaplan Meier statistic given by: $\hat{S}(t) = \prod_{i: t_i \leq t} \left(1 - \frac{d_i}{n_i}\right)$,

The graph generated has the time in days for a given probability of the end action not occurring or the original starting action to survive. In the case of a borrow to repay the survival percent represents the probability of no repays for a given user. I used my generateSurvival function which included the survminer package with the Surv function that calculates the probabilities along with dplyr to separate out the data set by users. I originally started with code for each of the transaction data sets but then consolidated the process of censoring the data and running the survival function on the data into a single generateSurvival function. This function was then included in the shiny app where others can run the survival analysis on different transaction types based on the most recent data set or even clusters of the users, coins, or other transformations of the data.

Results

The results of this problem on which transaction type to use were Borrows to Repays as the most sensical transaction to analyze. The percentages of Deposits to Borrows were lower along with a smaller dataset. Out of the different transaction types when further analyzing survival we used the time to repay as a good metric.

```

df<-read_rds('~/.IDEA-Blockchain/DefiResearch/Data/transactions.Rds')

borrows <- df %>%
  filter(type=="borrow" )

repays <- df %>%
  filter(type=="repay")

deposits <- df %>%
  filter(type=="deposit")

redeems <- df %>%
  filter(type=="redeem")%>%
  select(user, type, reserve, timestamp)

p1 <- generateSurvival(borrows, repays)

p2 <- generateSurvival(deposits, borrows)

```

```

p3 <- generateSurvival(deposits, redeems)

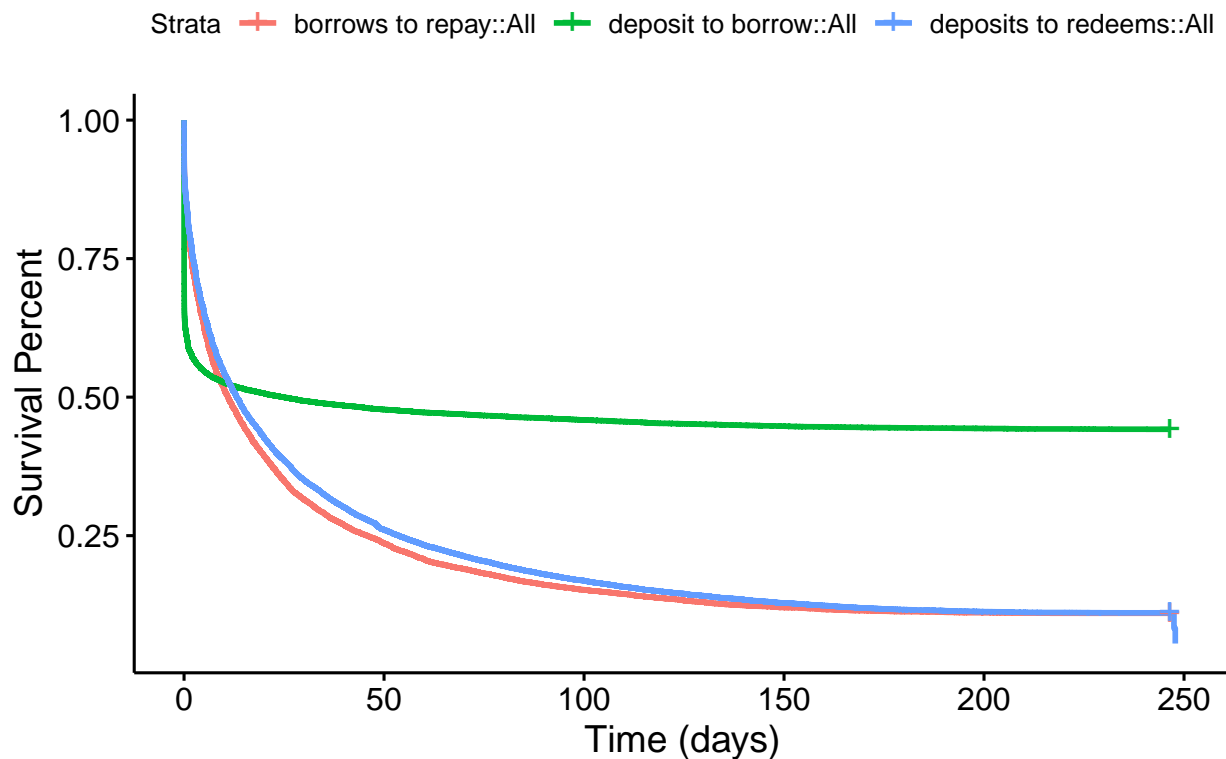
surv_fit_list <- list("borrows to repay" =p1,
                      "deposit to borrow" =p2,
                      "deposits to redeems" =p3)

ggsurvplot(surv_fit_list, combine = TRUE, xlab="Time (days)", ylab="Survival Percent", ylim = c(0.05,1))

## Warning: `select_()` was deprecated in dplyr 0.7.0.
## Please use `select()` instead.
## Warning: Removed 2 row(s) containing missing values (geom_path).
## Warning: Removed 2 row(s) containing missing values (geom_path).

```

Survival Analysis of Transaction Types



```

activeMedian <- surv_median(p1)
passiveMedian <- surv_median(p3)

```

The active investment strategy survival median:

```
activeMedian
```

```

##   strata  median   lower   upper
## 1    All 10.99316 10.36134 11.55295

```

The passive investment strategy survival median:

```
passiveMedian
```

```
##   strata  median   lower   upper
## 1     All 13.13476 12.75633 13.79034
```

Discussion

Based on the survival percent there is a 50% chance a user would use their deposit to borrow from a Defi pool. In fact it seems those who do deposit to borrow do so close to immediately. This phenomenon is shown in the sudden drop of the green line close to 0 days. This is fitting as it would be logical to use a deposit almost as collateral for a higher borrow amount. The plateau around 50% shows users who would borrow from the pool they deposited into, do so immediately or not at all after some time. It is increasingly unlikely for a user to deposit, wait several days and then borrow from that pool. This is a reasonable thought process as those borrowing or taking out a loan after depositing would like to do so nearly immediately in the interest of time given the coins rapidly changing nature.

As stated earlier, borrows represent an active investment strategy which almost always has time as a factor. Focusing on the red line on the survival graph of the transaction types indicates the most relevant transaction types for the active investment strategy. Users seem ever so slightly more inclined to repaying the borrows earlier in comparison to redeeming deposited funds. However, the two investment strategies are very close with the active investment strategy garnering only a slightly more fast response. By 200 days the two investment strategies are equal in that the percentage chance of a borrow being not repaid is equivalent to a deposit not being redeemed. Thus the active and passive investment strategies seem to have a roughly equivalent outcome as far as analysis of the associated transaction types.

There is a slight leaning towards the active investment strategy in terms of time, as the borrow is repaid slightly earlier than the redeem of a deposit. This is further evident in the median survival values that offer a more concrete value for each of the investment strategies. The active investment strategy has a median survival value of 10.993 while the passive investment strategy transactions has a median of 13.135. The range of the confidence intervals for both with the lower and upper bound seem comparable though the disparity in the median value leans to active investments as slightly superior. These results are logical and they correspond to investment strategies in traditional finance.

In traditional finance, though there is a lot more potential to make a significant return with an active investment strategy, very few do and on average an active investment managed portfolio only beats a passive counterpart 10-25% of the time. Hence, it is not surprising to see the survival graph have the two different transaction approaches fall close to one another. However, it still stands the active investment strategy requires thought of the user in selecting specific coins and an opinion on the investment more than a passive investment which simply grows at a small rate. Thus when further analyzing survival of specific coins, I use borrows to repays as only active investors would have an opinion and choose to actively invest into specific Defi coins and pools.

Problem 2

The second problem to analyze were the coin types and their specific survival. I grouped the coins based on stability to see how the individual stable coins compared to each other and how stable coins in general compare to non-stable coins. For this analysis I used the borrow and repay transactions as determined from the previous Problem. The survival analysis of the individual coins show the confidence and repayment of loans by users towards specific coin types. The stability of the coin is labeled in the data set, however analyzing user's actions towards the pool shows their opinion and their stake in repaying and borrowing from a certain coin.

Methods

Below is a comprehensive breakdown of the stable coins within the Defi coinbase. I further separated the borrows data set into the transactions for each individual coin. I did the same for the repays of those coins. Now using the generateSurvivalCensoredData function which is simply a shortened version from earlier I generated the survival data for each coin's borrows and repays transactions by returning the censored data and then separating the stable coins from the non-stable coins. The plots are then created using ggplot to plot the different reserve coins.

```
generateSurvivalCensoredData <- function(start, end, timeFinal)
{
  data_end <- left_join(start,end,by=c("user", "reserve")) %>%
  dplyr::rename(startTime=timestamp.x) %>%
  dplyr::rename(endTime=timestamp.y) %>%
  group_by(user, reserve) %>%
  dplyr::summarise(timeDiff=case_when(min(endTime)-min(startTime)>0 ~ min(endTime)-min(startTime),
  TRUE ~ min(startTime) - timeFinal)) %>%
  mutate(status=case_when(timeDiff<=0 ~ 0, timeDiff>0 ~ 1)) %>%
  mutate(timeDiff = abs(timeDiff))%>%
  select(user,reserve,timeDiff,status)%>%
  ungroup()
  return (data_end)
}

#setting up the transaction data for borrows and repays
loanData <- df %>%
filter(type == 'borrow' | type == 'repay') %>%
select(user, type, reserve, timestamp) %>%
arrange(timestamp)
loanDataKM <- data.frame(time=numeric(0), eventType = numeric(0), reserve = factor(levels = unique(df$reserve)))
timeFinal <- loanData[nrow(loanData), "timestamp"]
#after calculating the timestamp for the end of survival filter the two transactions
borrows <- df %>%
filter(type=="borrow")
repays <- df %>%
filter(type=="repay")
#run the function for the censoring of the data
censoredData <- generateSurvivalCensoredData(borrows, repays, timeFinal)

## `summarise()` has grouped output by 'user'. You can override using the `.groups` argument.

#mark the stable and non-stable coins
loanCountReserve <- censoredData %>%
count(reserve) %>%
arrange(-n)
stableCoins <- df %>%
filter(reserve == 'USDT' | reserve == 'USDC' | reserve == 'GUSD' | reserve == 'DAI' | reserve == 'WBTC')
group_by(reserve) %>%
count(reserve) %>%
arrange(-n)
df <- df %>%
mutate(reserveType = if_else(reserve %in% stableCoins$reserve, "Stable", "Non-Stable"))
reserveTypes <- df %>%
select(reserve, reserveType) %>%
distinct()
loanDataKMStable <- censoredData %>%
```



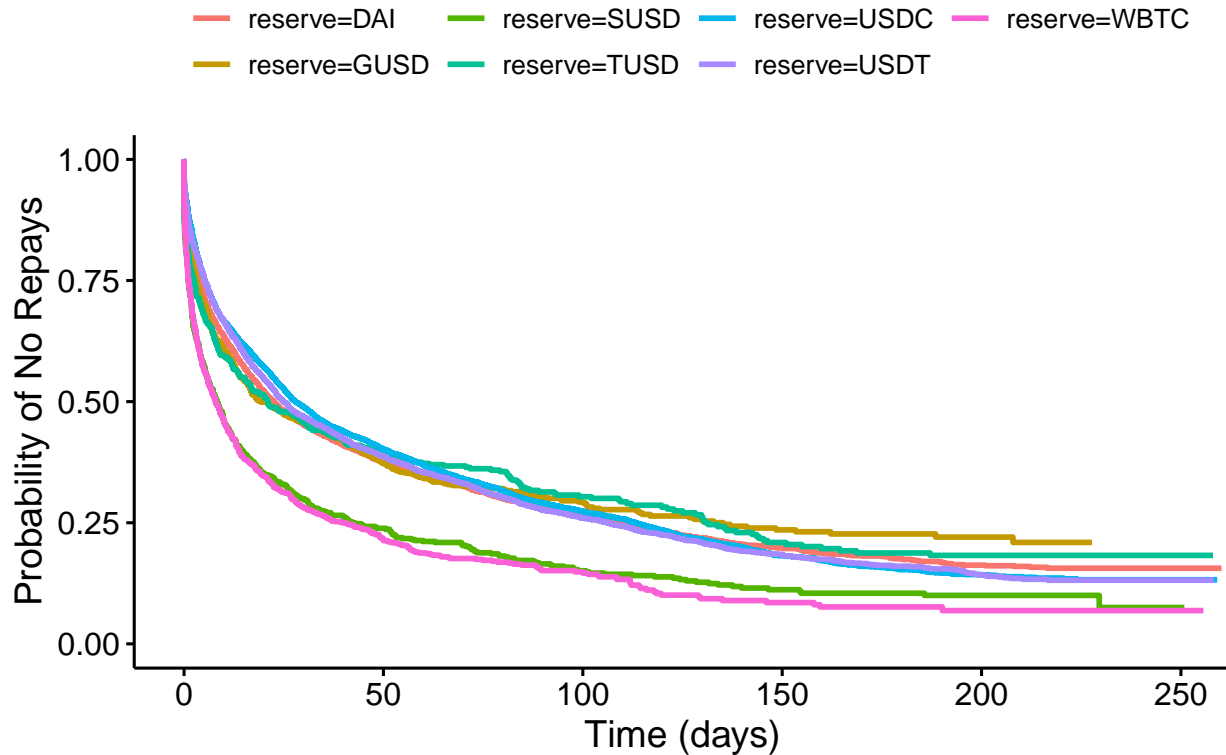
```

left_join(reserveTypes) %>%
filter(reserve %in% head(loanCountReserve, 10)$reserve, reserveType=="Stable")

## Joining, by = "reserve"
#plot the top stable coins
km_fitStable <- survfit(Surv(as.numeric(timeDiff/86400), as.numeric(status)) ~ reserve, data=loanDataKM)
ggsurvplot(km_fitStable, xlab="Time (days)", ylab="Probability of No Repays", title="Borrow to Repay of

```

Borrow to Repay of Top Stable Coins



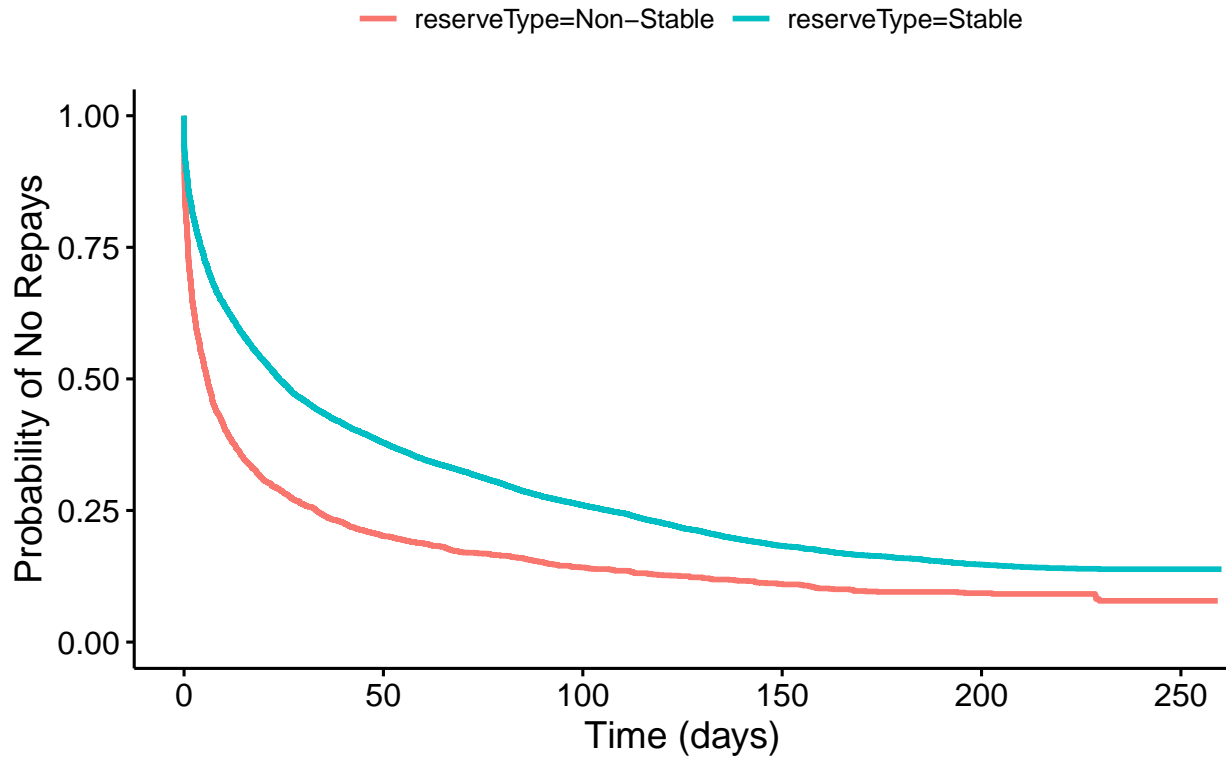
```

#plot the non-stable and stable coins
loanDataKMNonStable <- censoredData %>%
left_join(reserveTypes)

## Joining, by = "reserve"
km_fitNonStable <- survfit(Surv(as.numeric(timeDiff/86400), as.numeric(status)) ~ reserveType, data=loanDataKMNonStable)
coinTypesMedianSurv <- surv_median(km_fitNonStable)
ggsurvplot(km_fitNonStable, xlab="Time (days)", ylab="Probability of No Repays", title="Time until Loan

```

Time until Loans are Repaid in Stable and Non-stable Coin



Results

The first graph compares all the stable coins to get a definitive ranking of which coin is most likely to have its loans repaid. The ranking of the stable coins in order of least likely to have a loan repaid to most likely is as follows: GUSD, USDC, DAI, USDT, SUSD, WBTC. Thus if one were to deposit in a pool for a coin and would like to make sure their deposit is not affected by loans that do not get repaid in a timely fashion, WBTC would be the best stable coin as the borrows are most probable to get repaid and get repaid the fastest as per the first graph in which the magenta line of WBTC is the steepest drop, hence the fastest to have repays, and the lowest percentage over time, thus the most probable to be repaid among the stable coins. However among the stable coins does not include all the coins. Plotting all non-stable coins results in a plot with too many lines to decipher so instead I grouped the stable coins and non-stable coins to see on average which would have a higher change of have borrows repaid. Stable coins have a higher probability of no repays, thus the non-stable coins are more favorable in the second graph.

Discussion

The results of stable and non-stable coins seem counterintuitive as the stability of the coin would theoretically act as a factor in repaying a loan from the coin. A theory I concluded is that users feel stable coins are less volatile and therefore less likely to lose value unlike non-stable coins. Perhaps users see the unstable coin pool as more risky and would like to keep the loan term short as they are unsure of the future value of the non-stable coin amount borrowed. Thus a loan from a non-stable coin pool could lose value faster than that of a stable coin so users may feel rushed to repay the borrowed amount. On the other hand users may be hesitant to repay loans from the stable coins as a lack of confidence in the stable coin. Though the exact reasoning of the users is unknown and simply hypothesized about, it remains true that loans or borrows from the unstable coin pools are repaid faster and more often than those of stable coins.

Summary and Recommendations

In conclusion, I analyzed two investment strategies of users within DeFi transactions by using survival analysis to comprehend the pattern of various transaction types. I compared active investment by using the borrows to repays transactions and passive investment by using the deposit to redeems transactions. Overall I found the two investment strategies in DeFi reflect that of traditional finance since though active investing holds an edge over passive investing, the difference is not significantly larger for active investments. In theory active investment has potential of much higher returns than that of passive investment in traditional finance, and the survival analysis of the transactions of borrow to repays gives a better survival curve as a repaid loan matches this concept. However, as displayed on the graph between the various transaction types, the more passive deposit to redeem strategy is not far off. This is also confirmed to match traditional finance investments with various portfolio managers who struggle to outperform the passive benchmark in traditional finance.

An interesting result is that between stable and non-stable coins, non-stable coins have a higher likeliness in having a borrow repaid and repaid faster than those of stable coins. Though the exact reasoning is unknown, future analysis of user confidence in stable coins vs unstable coins could help determine whether this is due to hesitancy and lack of confidence in stable coins, or rather the opposite as I believe where users have less doubt the amount they borrowed will depreciate over time due to volatility of the DeFi coin.

Overall I found the process of survival analysis helps put perspective to the transactions by users and how they correlate to the perception of different coins. Analysis of the various coin types highlights how users may view that coin, and their loans from specific pools. I would recommend the survival analysis be included in the shiny app, especially as the function is versatile and can be applied to further cluster analysis that others within the team worked on throughout the semester.

For the future it would be interesting to plot the “price” of a specific coin in a specific set currency across time and see how the transactions affect the price of different coins over time. It would solidify the understanding of whether active investment through borrows and repays yield a conclusive profit over time. Another aspect to analyze would include how unpaid borrows result in liquidation or affect the price of a coin. Are interest rates affected, or liquidations? Do unpaid borrows affect the rate at which a future user can borrow from a coin?

The difference of investment strategies between active and passive transactions and investments concerning DeFi coins held more similarity to the workings of traditional finance. I found unexpected similarities despite the difference in investment resources.

References

Packages used: - survminer - survplot -autoplot -deplyr -ggplot

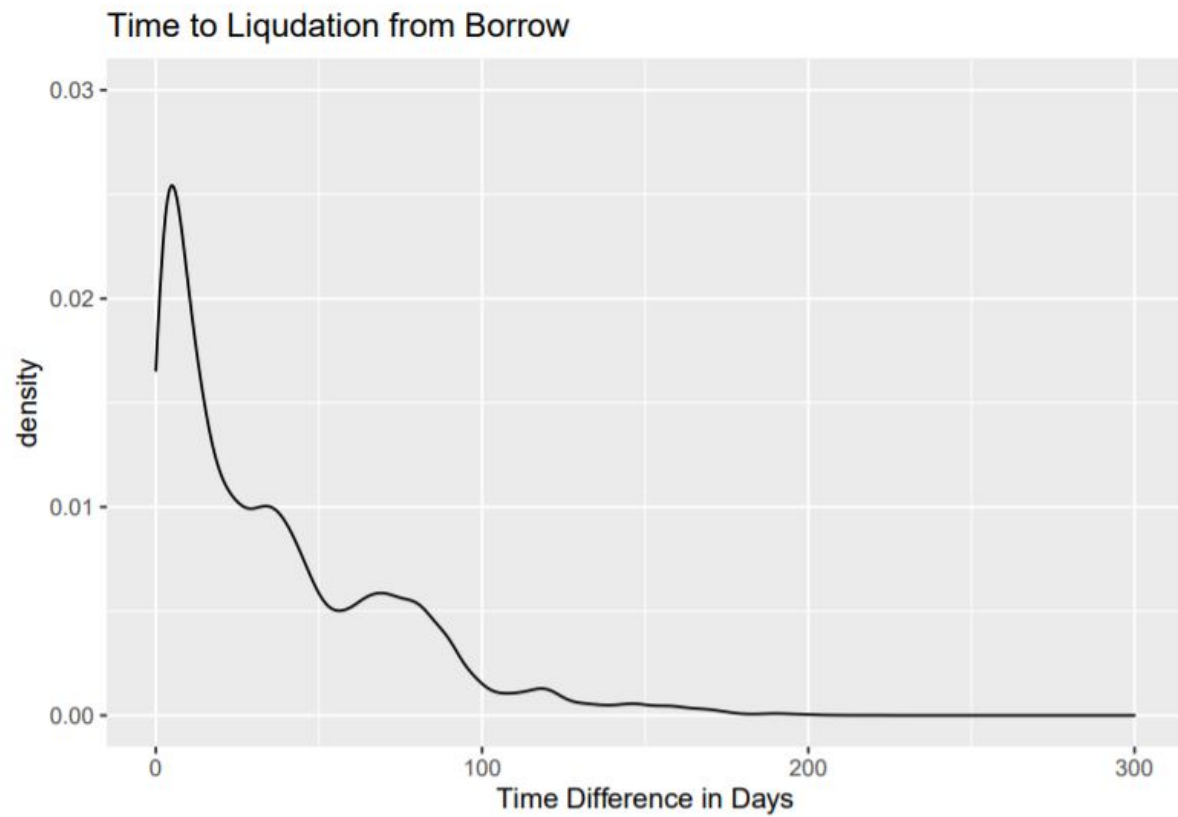
Sources on research for traditional finance investment strategies: (active vs passive) <https://www.cnbc.com/2021/11/01/in-one-of-the-most-volatile-markets-in-decades-active-fund-managers-underperformed-again.html>

<https://www.marketwatch.com/story/more-evidence-that-passive-fund-management-beats-active-2019-09-12>

<https://www.investopedia.com/ask/answers/040315/what-difference-between-passive-and-active-portfolio-management.asp>

Appendix

Some extra graphs for points not discussed:



Github issue #81 example:

Borrow to Repay of Top Stable Coins

