DAR F21 Project Status Notebook Template

DeFi

Chris Cammilleri

11/19/21

Contents

Weekly Work Summary

NOTE: Follow an outline format; use bullets to express individual points.

- RCS ID: cammmic
- Project Name: Blockchain Research
- Summary of work since last week
  - I have been working on an algorithm to calculate the daily balances for user accounts
  - So far, I have a basic version of this algorithm working, but more testing is needed
  - I started to work on a way to visualize this data for a particular user
- NEW: Summary of github issues added and worked
  - Submittes issue to create plots for user account balances
- Summary of github commits
  - Added python script for algorithm (UserSummary.ipynb)
  - Uploaded csv file with a sample of the user data (usersdaily.csv)

Personal Contribution

- All work was done on my own

Discussion of Primary Findings

- I wrote an algorithm to calculate daily account balances for each user. The algorithm still needs to be tested more and has some bugs to be worked out. Here is the data for a sample user that the algorithm calculated.

```
#import libraries
library(ggplot2)
library(ggbiplot)
```

```
## Loading required package: plyr
```

```
## Loading required package: scales
```

```
## Loading required package: grid
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v tibble  3.0.6     v dplyr   1.0.7
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
## v purrr   0.3.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::arrange()    masks plyr::arrange()
## x readr::col_factor() masks scales::col_factor()
## x purrr::compact()    masks plyr::compact()
## x dplyr::count()      masks plyr::count()
## x purrr::discard()    masks scales::discard()
## x dplyr::failwith()   masks plyr::failwith()
## x dplyr::filter()     masks stats::filter()
## x dplyr::id()         masks plyr::id()
## x dplyr::lag()        masks stats::lag()
## x dplyr::mutate()     masks plyr::mutate()
## x dplyr::rename()     masks plyr::rename()
## x dplyr::summarise()  masks plyr::summarise()
## x dplyr::summarize()  masks plyr::summarize()
```

```
df<-read_csv('ViolaRodriguez.csv')
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   date = col_date(format = ""),
##   LINK_deposited = col_double(),
##   WBTC_borrowed = col_double(),
##   WETH_deposited = col_double(),
##   WBTC_deposited = col_double(),
##   USDC_borrowed = col_double(),
##   UNI_deposited = col_double(),
##   YFI_deposited = col_double(),
##   ZRX_deposited = col_double(),
##   DAI_borrowed = col_double(),
##   USDT_borrowed = col_double(),
##   user = col_character()
## )
```

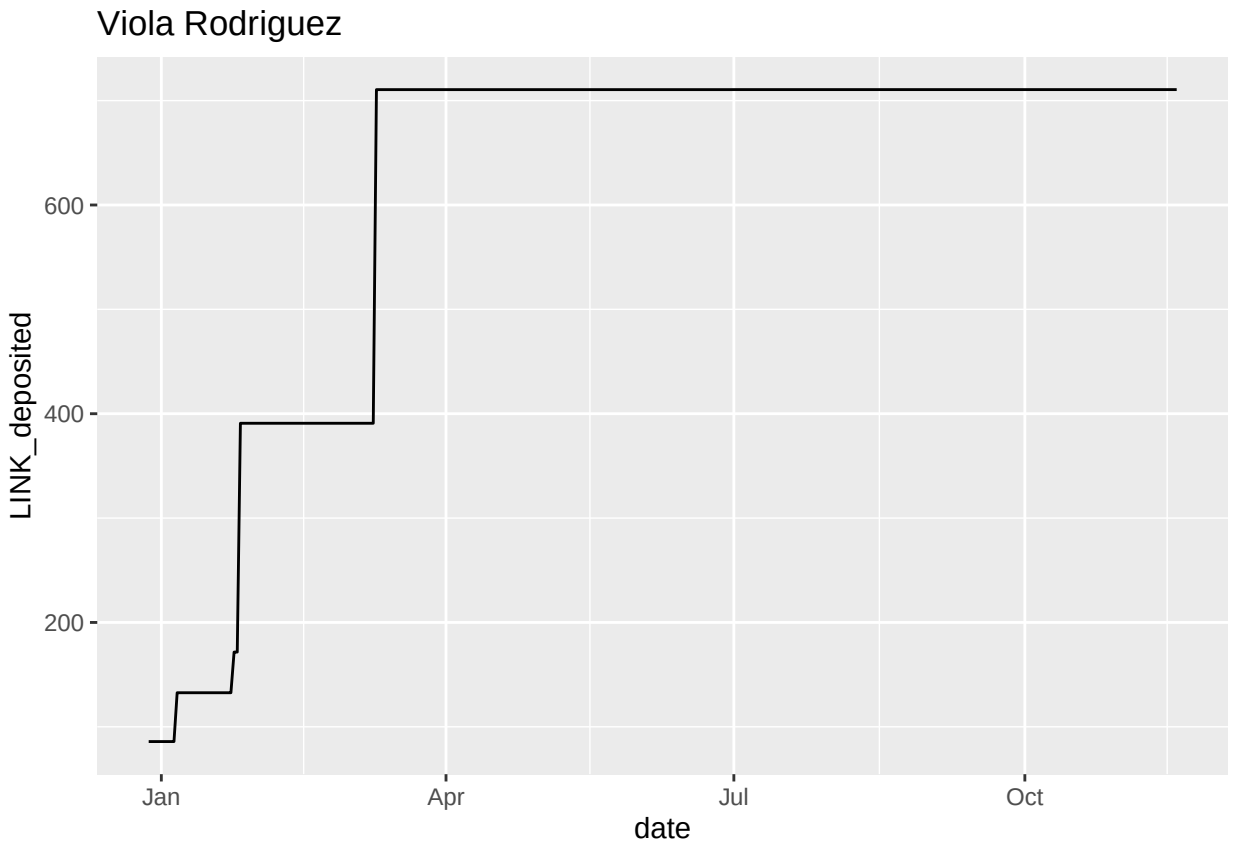
```
head(df)
```

```
## # A tibble: 6 x 12
##   date       LINK_deposited WBTC_borrowed WETH_deposited WBTC_deposited
##   <date>              <dbl>         <dbl>          <dbl>          <dbl>
## 1 2020-12-28           85.8   0.0200                0              0
## 2 2020-12-29           85.8   0.0200                0              0
## 3 2020-12-30           85.8   0.0200                0              0
## 4 2020-12-31           85.8   0.0200                0              0
## 5 2021-01-01           85.8   0.0200                0              0
## 6 2021-01-02           85.8   0.000000388           7.00           0.227
## # ... with 7 more variables: USDC_borrowed <dbl>, UNI_deposited <dbl>,
```

```
## #   YFI_deposited <dbl>, ZRX_deposited <dbl>, DAI_borrowed <dbl>,
## #   USDT_borrowed <dbl>, user <chr>
```

- We can see the amounts of tokens borrowed and deposited for each day, with interest being calculated. Next, we make a plot showing the amount of LINK this user had deposited over time.

```
ggplot(data=df, aes(x=date, y=LINK_deposited)) +
  ggtitle('Viola Rodriguez')+
  geom_line()
```



From this chart, we can clearly see the times that the user made a LINK deposit. In the future, I wish to further test the algorithm, and work on finding a way to visualize this data.