Token Analysis

Matt Shaffer

10/30/2017

```r
library(Hmisc)
```

```
## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

## Loading required package: ggplot2

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':
##
##     format.pval, round.POSIXt, trunc.POSIXt, units
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.1

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:Hmisc':
##
##     combine, src, summarize

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(forecast)
library(astsa)
```

```
##
## Attaching package: 'astsa'

## The following object is masked from 'package:forecast':
##
##     gas
```

```r
library(vars)
```

```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
```

```
##     select

## Loading required package: strucchange

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich

## Loading required package: urca

## Loading required package: lmtest
```

```
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 3.4.1
```

```
library(effects)
library(ggplot2)
line_color1 <- rgb(165/255, 203/255, 223/255)
line_color2 <- rgb(199/255, 37/255, 4/255, 0.5)
line_color3 <- rgb(200/255, 200/255, 200/255, 0.5)
line_color4 <- rgb(133/255, 161/255, 198/255, 0.25)
```

Data source: https://www.kaggle.com/jessevent/all-crypto-currencies

```
cc <- read.csv("./data/crypto-markets.csv")
cc <- cc[order(cc$date),]
```

```
Hmisc::describe(cc)
```

```
## Warning in w * sort(x - mean(x)): longer object length is not a multiple of
## shorter object length

## Warning in w * sort(x - mean(x)): longer object length is not a multiple of
## shorter object length

## cc
##
##  11  Variables      582020  Observations
## ---------------------------------------------------------------------------
## date
##        n  missing distinct
##   582020        0     1638
##
## lowest : 0          2013-04-28 2013-04-29 2013-04-30 2013-05-01
## highest: 2017-10-16 2017-10-17 2017-10-18 2017-10-19 2017-10-20
## ---------------------------------------------------------------------------
## open
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   582020        0   142286        1    21.79    43.33 0.000005 0.000025
##      .25      .50      .75      .90      .95
## 0.000278 0.004201 0.053669 0.696147 2.680000
##
## lowest : 0.00000e+00 1.00000e-10 2.00000e-10 3.00000e-10 5.00000e-10
## highest: 9.78454e+04 9.78826e+04 9.80508e+04 9.87964e+04 9.99178e+04
## ---------------------------------------------------------------------------
## high
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   582020        0   145268        1    22.72    45.17 0.000006 0.000029
##      .25      .50      .75      .90      .95
## 0.000324 0.004875 0.061875 0.786102 2.960000
##
## lowest : 0.00000e+00 1.30000e-10 3.10000e-10 6.10000e-10 1.10000e-09
## highest: 9.50199e+04 9.53648e+04 9.66164e+04 9.84227e+04 9.94790e+04
## ---------------------------------------------------------------------------
## low
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   582020        0   140117        1    22.89    45.56 0.000004 0.000021
##      .25      .50      .75      .90      .95
## 0.000243 0.003653 0.047379 0.614354 2.440000
##
## lowest : 0.00000e+00 9.20000e-14 1.10000e-10 1.20000e-10 2.50000e-10
## highest: 9.68347e+04 9.70293e+04 9.77273e+04 9.80576e+04 9.93792e+04
## ---------------------------------------------------------------------------
## close
##         n   missing  distinct      Info      Mean       Gmd       .05
##    582020         0    142477         1  10384231  20588291 1.990e-04
##       .10       .25       .50       .75       .90       .95
## 3.500e-04 1.837e-03 2.156e-02 1.070e+00 5.600e+05 4.000e+06
##
## lowest : 0.00e+00 5.00e-06 2.40e-05 1.01e-04 1.02e-04
## highest: 9.80e+09 9.90e+09 1.00e+10 2.00e+10 5.00e+10
## ---------------------------------------------------------------------------
## volume
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   582020        0    96352    0.999  1421088  2825613      0.0      1.0
##      .25      .50      .75      .90      .95
##     14.0    212.5   3764.0  69238.1 371967.5
##
## lowest :          0          1          2          3          4
## highest: 3196230000 3258050000 3615480000 3764240000 4148070000
## ---------------------------------------------------------------------------
## market
##         n   missing  distinct      Info      Mean       Gmd       .05
##    582020         0    303704     0.998  53580193 106197160         0
##       .10       .25       .50       .75       .90       .95
##         0      6167     56572    528362   5927880  20647210
##
## lowest :           0           1           2           3           4
## highest: 93803000000 94559000000 94947900000 95469300000 97011900000
## ---------------------------------------------------------------------------
## symbol
##        n  missing distinct
##   582020        0     1170
##
## lowest : $$$   020   1337  1ST   2GIVE, highest: ZRX   ZSC   ZSE   ZUR   ZYD
## ---------------------------------------------------------------------------
## coin
##        n  missing distinct
##   582020        0     1136
##
## lowest :                 020LondonCoin 0x            42-coin       808Coin
## highest: Zoin          Zonecoin      ZrCoin        ZSEcoin       Zurcoin
## ---------------------------------------------------------------------------
## variance
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##   582020        0   390054        1     -Inf      NaN -0.29982 -0.16364
##      .25      .50      .75      .90      .95
## -0.04566  0.01517  0.17966  1.00000  1.00000
##
## Value           -Inf -9700000        0
## Frequency         1        1   582018
## Proportion        0        0        1
## ---------------------------------------------------------------------------
## volatility
##         n   missing  distinct      Info      Mean       Gmd       .05
##    582020         0    398194         1       Inf       NaN 4.286e-15
##       .10       .25       .50       .75       .90       .95
## 1.000e-11 3.977e-02 1.327e-01 2.825e-01 5.250e-01 7.648e-01
##
## Value             0  300000 9800000     Inf
## Frequency    582017       1       1       1
## Proportion        1       0       0       0
## ---------------------------------------------------------------------------
```

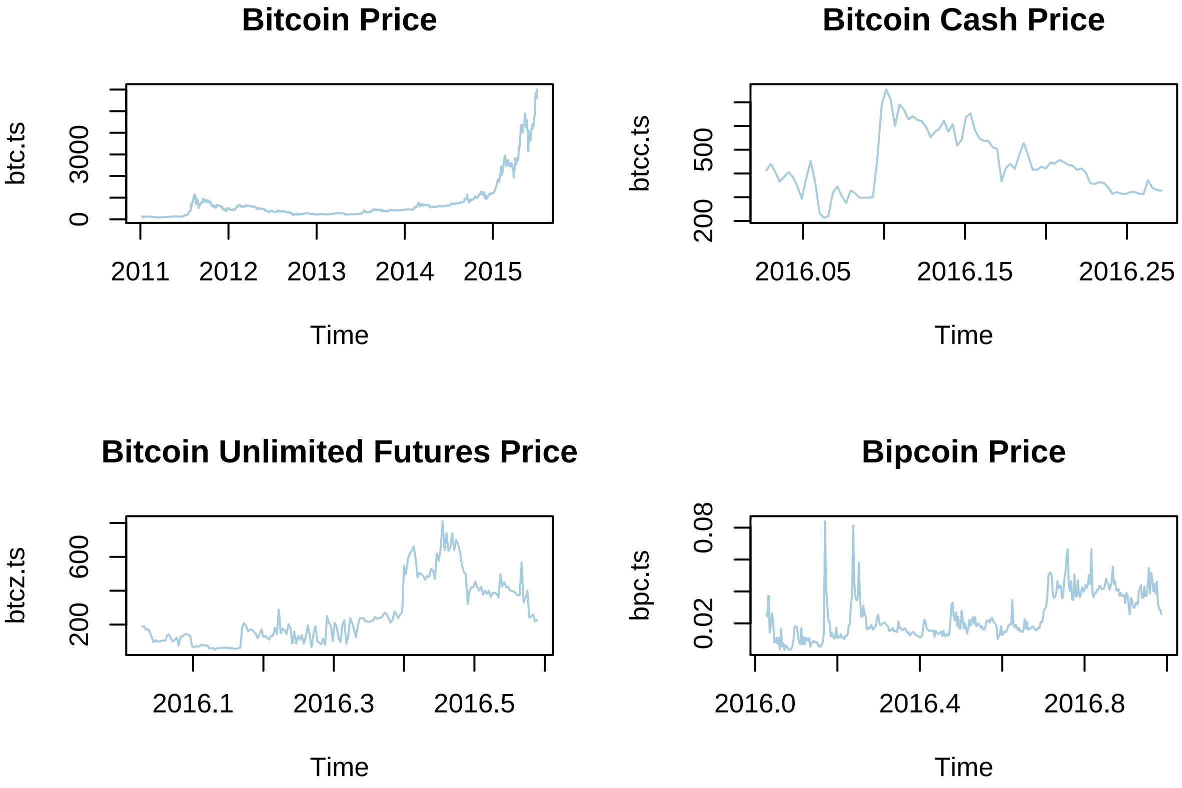
```r
names(cc)
```

```
##  [1] "date"       "open"       "high"       "low"        "close"
##  [6] "volume"     "market"     "symbol"     "coin"       "variance"
## [11] "volatility"
```

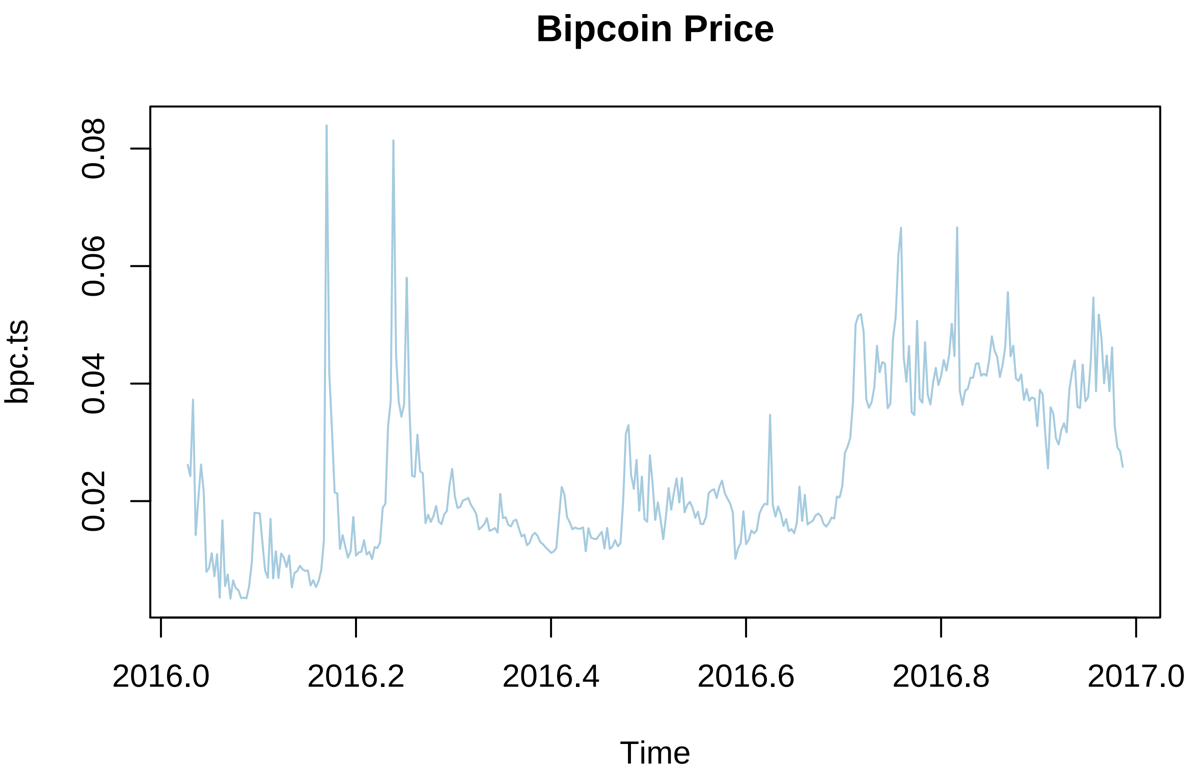
```r
#levels(factor(cc$coin))
```

```r
par(mfrow=c(2,2))
btc <- cc[which(cc$coin == "Bitcoin"),]
btc$date.fmt <- strptime(as.character(btc$date), "%Y-%m-%d")

btc.ts <- ts(btc$close, frequency=365, start=c(2011,08,07))
plot(btc.ts, type='l', col=line_color1, main="Bitcoin Price")

btcc <- cc[which(cc$coin == "Bitcoin Cash"),]
btcc$date.fmt <- strptime(as.character(btcc$date), "%Y-%m-%d")

btcc.ts <- ts(btcc$close, frequency=365, start=c(2016,11,03))
plot(btcc.ts, type='l', col=line_color1, main="Bitcoin Cash Price")

btcz <- cc[which(cc$coin == "Bitcoin Unlimited Futures"),]
btcz$date.fmt <- strptime(as.character(btcz$date), "%Y-%m-%d")

btcz.ts <- ts(btcz$close, frequency=365, start=c(2016,11,03))
plot(btcz.ts, type='l', col=line_color1, main="Bitcoin Unlimited Futures Price")

bpc <- cc[which(cc$coin == "BipCoin"),]
```

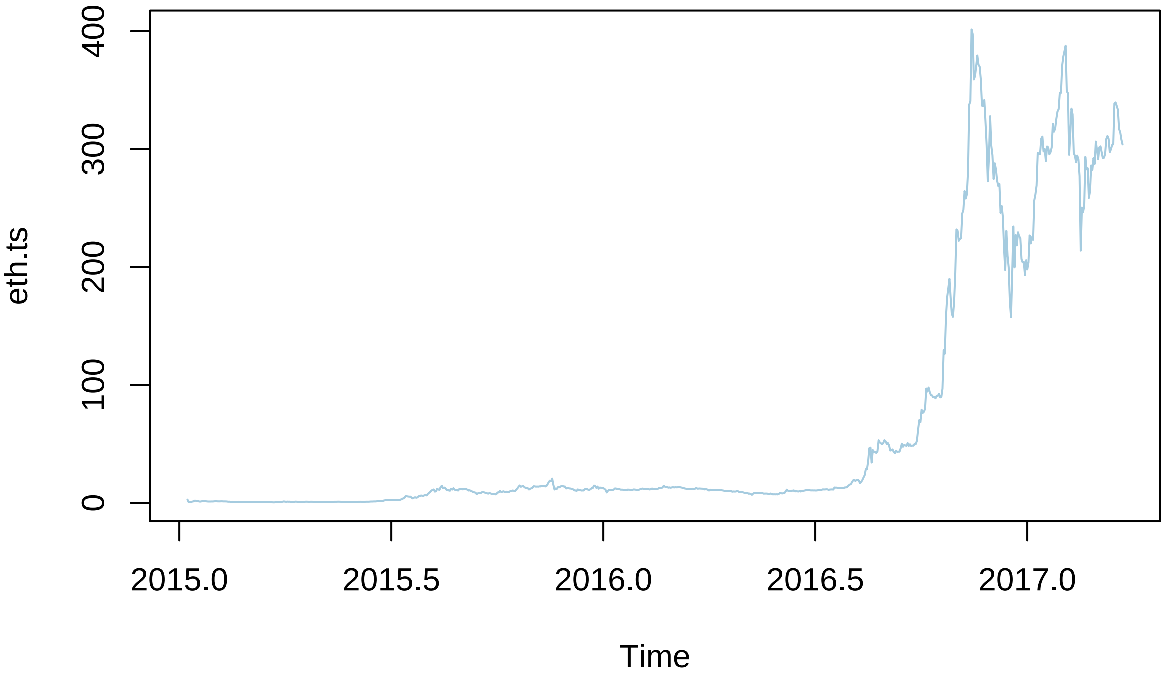
```
bpc$date.fmt <- strptime(as.character(bpc$date), "%Y-%m-%d")

bpc.ts <- ts(bpc$close, frequency=365, start=c(2016,11,03))
plot(bpc.ts, type='l', col=line_color1, main="Bipcoin Price")
```



```
bpc <- cc[which(cc$coin == "BipCoin"),]
bpc$date.fmt <- strptime(as.character(bpc$date), "%Y-%m-%d")

bpc.ts <- ts(bpc$close, frequency=365, start=c(2016,11,03))
plot(bpc.ts, type='l', col=line_color1, main="Bipcoin Price")
```

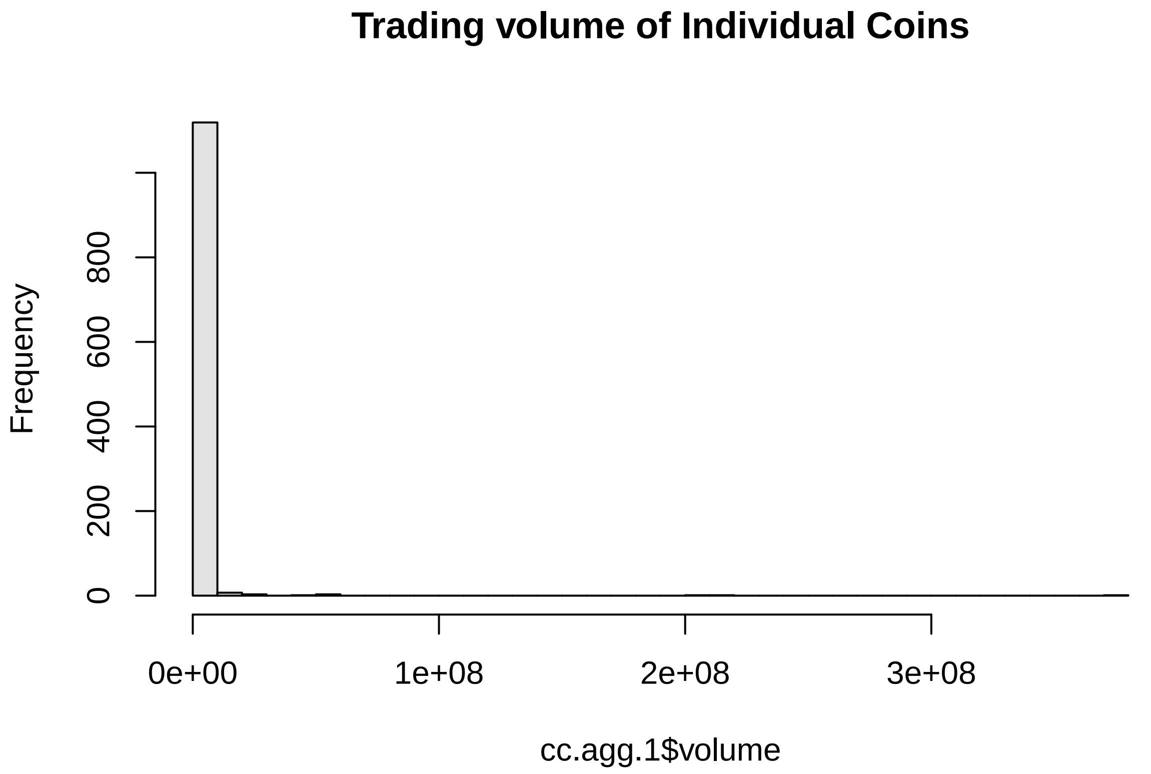


```r
eth <- cc[which(cc$coin == "Ethereum"),]
eth$date.fmt <- strptime(as.character(eth$date), "%Y-%m-%d")

eth.ts <- ts(eth$close, frequency=365, start=c(2015,08,07))
plot(eth.ts, type='l', col=line_color1)
```

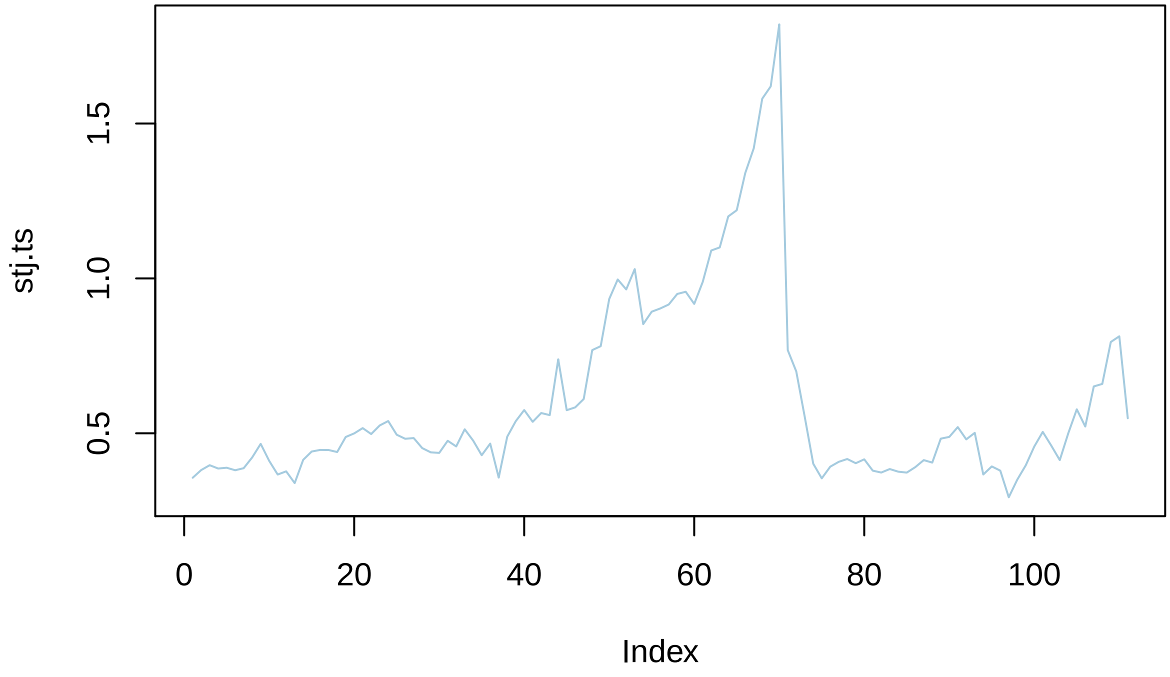


```r
cc.vol <- cc[c('volume', 'coin')]
cc.agg.1 <- aggregate(volume ~ coin, cc.vol, mean)
```

```r
hist(cc.agg.1$volume, breaks = 30, col=line_color3, main="Trading volume of Individual Coins")
```



```r
stj <- cc[which(cc$coin == "Storj"),]
stj$data.fmt <- strptime(as.character(stj$date), "%Y-%m-%d")

stj.ts <- rev(ts(stj$close, frequency=365, start=c(2017,7,2)))
plot(stj.ts, type='l', col=line_color1)
```


stj.ts
Index