607 Project

Munkhnaran Gankhuyag

December 06, 2017

Motivation

Cryptocurrency is a digital currency that is created and managed through the use of cryptography, which is an advanced type of encryption technique. Bitcoin was one of the earliest forms of cryptocurrency and since then, many other types of cryptocurrencies have been created. I first heard about Bitcoin back in 2013, when I was a financially struggling college student, and I thought 80 Dollars per coin was a bit too much. Today, one bitcoin is about 100 times that amount.

Below is the index for November 19th 2017 from https://www.coindesk.com

Today's Open: $8,033.94 Today's High: $8,049.12 Today's Low: $8,021.33 Today's Closed: $8,034.42

*Update: Since the time this proposal was written, Bitcoin has increased dramatically. Below is the index for December 9th 2017 from https://www.coindesk.com

Today's Open: $16,057.15 Today's High: $16,291.68 Today's Low: $15,538.25

Even though Bitcoin seems like a great investment, many investors remain skeptical. The main reason is bitcoin and other cryptocurrencies are virtual currencies that are not backed by anything. Since there are a limited amount of bitcoin(20 million), the market purely fluctuates on demand, making it extremely volatile. This gives spectators reason to believe it is a bubble. Currently, many new investors, attracted by its potential, are entering the market, increasing its value significantly, but this can change if the demand drops. Currently there are many theories out there, but for now, I wanted to see how volatile bitcoin is compared to other currencies.

Obtain the Data

Bitcoin API from Quandl

Since bitcoin is the most widely known cryptocurrency, there is a lot of data available for it. I was able to find an API for Bitcoin's historical data.

```r
#Daily Bitcoin exchange rate (BTC vs. USD) on Bitstamp from 09/13/2011 to 12/09/2017
Quandl.api_key('mxcqtzYcf5Co4fbG3WAX')
Bitcoin <- Quandl('BCHARTS/BITSTAMPUSD', start_date='2011-01-01', end_date ='2017-12-07')
head(Bitcoin )
```

```
##         Date     Open     High      Low    Close Volume (BTC)
## 1 2017-12-07 13623.00 16615.62 13085.90 16599.99    25787.677
## 2 2017-12-06 11676.99 13700.00 11659.80 13623.50    19784.873
## 3 2017-12-05 11613.07 11850.00 11384.25 11677.00    11875.034
## 4 2017-12-04 11250.00 11613.07 10850.00 11613.07    13621.482
## 5 2017-12-03 10875.68 11800.01 10513.16 11250.00    14238.526
## 6 2017-12-02 10840.45 11200.00 10637.69 10872.00     9267.161
##   Volume (Currency) Weighted Price
## 1         382694044       14840.19
## 2         250560790       12664.26
## 3         138370076       11652.18
```

```
## 4         154122918       11314.70
## 5         160176290       11249.50
## 6         101270084       10927.84
```

```r
tail(Bitcoin)
```

```
##            Date Open High  Low Close Volume (BTC) Volume (Currency)
## 2273 2011-09-18 4.87 4.92 4.81  4.92    119.81280          579.8431
## 2274 2011-09-17 4.87 4.87 4.87  4.87      0.30000            1.4610
## 2275 2011-09-16 4.82 4.87 4.80  4.85     39.91401          193.7631
## 2276 2011-09-15 5.12 5.24 5.00  5.13     80.14080          408.2590
## 2277 2011-09-14 5.58 5.72 5.52  5.53     61.14598          341.8548
## 2278 2011-09-13 5.80 6.00 5.65  5.97     58.37138          346.0974
##      Weighted Price
## 2273       4.839576
## 2274       4.870000
## 2275       4.854515
## 2276       5.094272
## 2277       5.590798
## 2278       5.929231
```

Main Competitors

The second tier of cryptocurrencies in terms of popularity include Ethereum, Litecoin. I was interested in those two, plus a couple more, Dash and Bitcoin cash. I was able to find a csv file on the historical data from the website: https://coinmetrics.io/data-downloads/

I saved the csv files in Github.

```r
Bc_cash <- read.csv("https://raw.githubusercontent.com/mikegankhuyag/607-Projects/master/Final/bch.csv"
Dash <- read.csv("https://raw.githubusercontent.com/mikegankhuyag/607-Projects/master/Final/dash.csv", 
Ethereum <- read.csv("https://raw.githubusercontent.com/mikegankhuyag/607-Projects/master/Final/eth.csv
Litecoin <- read.csv("https://raw.githubusercontent.com/mikegankhuyag/607-Projects/master/Final/ltc.csv
head(Bc_cash)
```

```
##        date txVolume.USD. txCount marketcap.USD. price.USD.
## 1 7/31/2017    2406864986  183998              0     294.46
## 2  8/1/2017     906913244  230867              0     380.01
## 3  8/2/2017     603435293   76537     6302360000     452.66
## 4  8/3/2017      83677447    7416     7392030000     364.05
## 5  8/4/2017     218502200   20909     5969720000     233.05
## 6  8/5/2017     263414318   26517     3809330000     213.15
##   exchangeVolume.USD. generatedCoins       fees
## 1             1075960         1837.5 138.116822
## 2            65988800         1812.5 194.868866
## 3           416207000         1112.5  51.706019
## 4           161518000           87.5   6.288903
## 5           185038000          437.5  20.234731
## 6           144043000          237.5  21.324498
```

```r
head(Dash)
```

```
##        date txVolume.USD. txCount marketcap.USD. price.USD.
## 1 2/14/2014     150641.42    3421         702537   0.374024
## 2 2/15/2014      78256.18    3663        1092120   0.314865
## 3 2/16/2014      96549.99    3236        1085280   0.406976
```

```
## 4 2/17/2014     367922.18    2766        1360260   1.450000
## 5 2/18/2014     838488.30    2631        3960320   1.040000
## 6 2/19/2014     675559.38    2551        3497850   0.941647
##   exchangeVolume.USD. generatedCoins     fees
## 1               15422          33849 3.355010
## 2               21119          33235 3.739035
## 3               28017          30944 5.238558
## 4              178618          20713 2.473010
## 5              160779          15940 2.827082
## 6               60551          13548 2.363510
```

```r
head(Ethereum)
```

```
##        date txVolume.USD. txCount marketcap.USD. price.USD.
## 1  8/7/2015             0       0              0   2.770000
## 2  8/8/2015       1513209    2016      167911000   0.753325
## 3  8/9/2015       1180418    2807       42637600   0.701897
## 4 8/10/2015        825663    1298       43130000   0.708448
## 5 8/11/2015       1787874    1999       42796500   1.070000
## 6 8/12/2015       1812412    4945       64018400   1.220000
##   exchangeVolume.USD. generatedCoins     fees
## 1              164329       27075.47  0.00000
## 2              674188       27437.66 37.31841
## 3              532170       27943.44 68.09997
## 4              405283       27178.28 14.09895
## 5             1463100       27817.34 31.16514
## 6             2150620       28027.81 11.31145
```

```r
head(Litecoin)
```

```
##        date txVolume.USD. txCount marketcap.USD. price.USD.
## 1 4/28/2013      39038951    8847       73773400       4.35
## 2 4/29/2013      48283929    9408       74952700       4.38
## 3 4/30/2013      38686090    9092       75726800       4.30
## 4  5/1/2013      33849471    9205       73901200       3.80
## 5  5/2/2013      58715299    8927       65242700       3.37
## 6  5/3/2013      13752345    8290       58607400       3.04
##   exchangeVolume.USD. generatedCoins     fees
## 1                   0          32800 511.0816
## 2                   0          31500 634.1212
## 3                   0          32450 597.0982
## 4                   0          31600 755.4951
## 5                   0          31450 689.1598
## 6                   0          28300 551.3278
```

New Comers

I was interested in two coins that are currently valued really low, Ripple and Tether. I was able to find the historical data on https://coinmarketcap.com. Since I couldn't download it, I decided to scrape the data.

```r
Rip <- read_html("https://coinmarketcap.com/currencies/ripple/historical-data/?start=20130428&end=20171
Tet <- read_html("https://coinmarketcap.com/currencies/tether/historical-data/?start=20130428&end=20171
```

```r
Ripp <- html_text(html_nodes(Rip, "td"))
Rippl <- matrix(Ripp, ncol = 7, byrow = TRUE)
```

```r
Ripple <- data.frame(Rippl[2:1588,], stringsAsFactors = TRUE)
colnames(Ripple) <- c("Date","Open","High","Low","Close","Volume","Market Cap")
head(Ripple)
```

```
##           Date     Open     High      Low    Close      Volume
## 1 Dec 08, 2017 0.223636 0.278673 0.222168 0.252125 660,172,000
## 2 Dec 07, 2017 0.232623 0.233760 0.221340 0.222823 275,205,000
## 3 Dec 06, 2017 0.245416 0.245705 0.227742 0.232544 274,526,000
## 4 Dec 05, 2017 0.253598 0.253988 0.245234 0.246101 174,591,000
## 5 Dec 04, 2017 0.252919 0.255362 0.247160 0.253571 104,650,000
## 6 Dec 03, 2017 0.255530 0.263072 0.247391 0.252558 134,710,000
##      Market Cap
## 1 8,663,460,000
## 2 9,011,630,000
## 3 9,507,190,000
## 4 9,815,990,000
## 5 9,768,480,000
## 6 9,869,310,000
```

```r
Teth <- html_text(html_nodes(Tet, "td"))
Tethe <- matrix(Teth, ncol = 7, byrow = TRUE)
```

```r
Tether <- data.frame(Tethe[2:1013,], stringsAsFactors = TRUE)
colnames(Tether) <- c("Date","Open","High","Low","Close","Volume","Market Cap")
head(Tether)
```

```
##           Date     Open High      Low Close        Volume  Market Cap
## 1 Dec 08, 2017     1.04 1.06 0.986563  1.02 1,993,030,000 843,587,000
## 2 Dec 07, 2017     1.01 1.08     1.00  1.03 1,671,610,000 819,775,000
## 3 Dec 06, 2017 0.999760 1.02 0.995840  1.01 1,281,490,000 813,822,000
## 4 Dec 05, 2017     1.00 1.01 0.996458  1.00   814,146,000 816,872,000
## 5 Dec 04, 2017     1.00 1.01 0.992132  1.00   668,510,000 816,012,000
## 6 Dec 03, 2017     1.00 1.03 0.985320  1.00   946,749,000 814,847,000
```

Cleaning the data

I want all the to be in the same format as the Bitcoin API. For the CSV files, I first wanted to reorder the data since it starts from the earliest point.

```r
Bc_cash <-Bc_cash[130:1,]
Dash <-Dash[1393:1,]
Ethereum <-Ethereum[854:1,]
Litecoin <-Litecoin[1685:1,]
```

The Bitcoin API is in a YYYY-MM-DD format. So I needed to change that all the dates to that format.

```r
Bc_cash$date <- as.Date(Bc_cash$date, format = "%m/%d/%Y")
Ethereum$date <- as.Date(Ethereum$date, format = "%m/%d/%Y")
Dash$date <- as.Date(Dash$date, format = "%m/%d/%Y")
Litecoin$date <- as.Date(Litecoin$date, format = "%m/%d/%Y")
```

Change column name 'date' to 'Date'

```r
colnames(Bc_cash)[1] <- "Date"
colnames(Ethereum)[1] <- "Date"
```

```r
colnames(Dash)[1] <- "Date"
colnames(Litecoin)[1] <- "Date"
```

Since the two scraped data is from a html website, all of it needs to be reformatted.

```r
Tether[,2:5] %<>%
      mutate_each(funs(if(is.factor(.)) as.character(.) else .)) %<>%
       mutate_each(funs(if(is.character(.)) as.numeric(.) else .))
Ripple[,2:5] %<>%
      mutate_each(funs(if(is.factor(.)) as.character(.) else .)) %<>%
       mutate_each(funs(if(is.character(.)) as.numeric(.) else .))
```

Also, change the date format to YYYY-MM-DD format.

```r
Tether$Date <- as.Date(Tether$Date, format = "%b %d, %Y")
Ripple$Date <- as.Date(Ripple$Date, format = "%b %d, %Y")
```

Combining the data

Now that all of our data is in consistent format, lets get all the variables needed for analysis.

I want to average out the highs and lows to get one price for the day.

```r
require(magrittr)
Bitcoin %<>% mutate(bitcoin_price = (High + Low)/2,
                bitcoin_gain = Close - Open)
Ripple %<>% mutate(ripple_price = (High + Low)/2,
                ripple_gain  = Close - Open)
Tether %<>% mutate(tether_price = (High + Low)/2,
                tether_gain =  Close - Open)
```

Selecting the data we need.

```r
Bitcoin_data <- select(Bitcoin, Date,bitcoin_price)
Ripple_data <- select(Ripple, Date, ripple_price)
Tether_data <- select(Tether, Date, tether_price)
Ethereum_data <- select(Ethereum, Date, price.USD.)
Dash_data <- select(Dash, Date, price.USD.)
Bc_cash_data <- select(Bc_cash, Date, price.USD.)
Litecoin_data <- select(Litecoin, Date, price.USD.)
```

Since, all of our data is in the same format, we join them using the 'Date' field as the identifier. Bitcoin data has the most rows, so I used it as the left part of left join.

```r
cryptocurrencies <-
left_join(
left_join(
left_join(
left_join(
  left_join(
    left_join(Bitcoin_data, Ripple_data, "Date"),
    Tether_data, "Date"),
    Ethereum_data, "Date"),
    Dash_data, "Date"),
    Bc_cash_data, "Date"),
    Litecoin_data, "Date")
```

```r
colnames(cryptocurrencies) <- c("Date","bitcoin_price","ripple_price","tether_price",
                                "ethereum_price", "dash_price","bc_cash_price","litecoin_price")
head(cryptocurrencies)
```

```
##         Date bitcoin_price ripple_price tether_price ethereum_price
## 1 2017-12-07      14850.76    0.2275500     1.040000         434.41
## 2 2017-12-06      12679.90    0.2367235     1.007920         428.59
## 3 2017-12-05      11617.12    0.2496110     1.003229         463.28
## 4 2017-12-04      11231.53    0.2512610     1.001066         470.20
## 5 2017-12-03      11156.58    0.2552315     1.007660         465.85
## 6 2017-12-02      10918.85    0.2549870     1.007805         463.45
##   dash_price bc_cash_price litecoin_price
## 1     697.90       1330.93          98.29
## 2     700.07       1430.10         100.35
## 3     756.36       1501.85         102.40
## 4     774.01       1576.92         104.24
## 5     768.88       1559.93         101.26
## 6     778.43       1434.98         100.28
```

Since the prices of the currencies between each other have huge differences, it would be hard to compare them. I decided to calculate the relative change between them.

I created another table with the previous day's price, added 1 to the day and did a left join to match the day.

```r
Prior_day <- cryptocurrencies[2:2278,]
Prior_day$Date <- as.Date(Prior_day$Date)+1
colnames(Prior_day) <- c("Date","yes_bitcoin_price","yes_ripple_price","yes_tether_price",
                                "yes_ethereum_price", "yes_dash_price","yes_bc_cash_price","yes_liteco

r_cyptocurrencies <- left_join(cryptocurrencies, Prior_day, by ="Date")
r_cyptocurrencies <- select(r_cyptocurrencies, Date, bitcoin_price, yes_bitcoin_price, ripple_price, ye
                            ethereum_price, yes_ethereum_price, dash_price, yes_dash_price, bc_cash_pri
head(r_cyptocurrencies)
```

```
##         Date bitcoin_price yes_bitcoin_price ripple_price yes_ripple_price
## 1 2017-12-07      14850.76          12679.90    0.2275500        0.2367235
## 2 2017-12-06      12679.90          11617.12    0.2367235        0.2496110
## 3 2017-12-05      11617.12          11231.53    0.2496110        0.2512610
## 4 2017-12-04      11231.53          11156.58    0.2512610        0.2552315
## 5 2017-12-03      11156.58          10918.85    0.2552315        0.2549870
## 6 2017-12-02      10918.85          10160.00    0.2549870        0.2491330
##   tether_price yes_tether_price ethereum_price yes_ethereum_price
## 1     1.040000         1.007920         434.41             428.59
## 2     1.007920         1.003229         428.59             463.28
## 3     1.003229         1.001066         463.28             470.20
## 4     1.001066         1.007660         470.20             465.85
## 5     1.007660         1.007805         465.85             463.45
## 6     1.007805         1.006522         463.45             466.54
##   dash_price yes_dash_price bc_cash_price yes_bc_cash_price litecoin_price
## 1     697.90         700.07       1330.93           1430.10          98.29
## 2     700.07         756.36       1430.10           1501.85         100.35
## 3     756.36         774.01       1501.85           1576.92         102.40
## 4     774.01         768.88       1576.92           1559.93         104.24
## 5     768.88         778.43       1559.93           1434.98         101.26
## 6     778.43         797.53       1434.98           1462.68         100.28
```

```
##   yes_litecoin_price
## 1             100.35
## 2             102.40
## 3             104.24
## 4             101.26
## 5             100.28
## 6              99.00
```

Calculate relative change

```
r_cyptocurrencies %<>% mutate(bitcoin_change = ((bitcoin_price - yes_bitcoin_price)/yes_bitcoin_price)*
                              ripple_change =  ((ripple_price - yes_ripple_price)/yes_ripple_price)*100
                              tether_change =  ((tether_price - yes_tether_price)/yes_tether_price)*100
                              ethereum_change =  ((ethereum_price - yes_ethereum_price)/yes_ethereum_pr
                              dash_change =  ((dash_price - yes_dash_price)/yes_dash_price)*100,
                              bc_cash_change =  ((bc_cash_price - yes_bc_cash_price)/yes_bc_cash_price)
                              litecoin_change =  ((litecoin_price - yes_litecoin_price)/yes_litecoin_pr
```

Create a new table with just the relative changes.

```
relative_change <- select(r_cyptocurrencies, "Date", ends_with("change"))
relative_change_ <- relative_change

relative_change_[is.na(relative_change_ <- relative_change)] <- 0
head(relative_change_)
```

```
##         Date bitcoin_change ripple_change tether_change ethereum_change
## 1 2017-12-07     17.1204820   -3.87519617    3.18279229       1.3579412
## 2 2017-12-06      9.1483478   -5.16303368    0.46759015      -7.4879123
## 3 2017-12-05      3.4331015   -0.65668767    0.21606967      -1.4717142
## 4 2017-12-04      0.6718006   -1.55564654   -0.65438739       0.9337770
## 5 2017-12-03      2.1773365    0.09588724   -0.01443731       0.5178552
## 6 2017-12-02      7.4689469    2.34974893    0.12756806      -0.6623226
##   dash_change bc_cash_change litecoin_change
## 1  -0.3099690      -6.934480      -2.0528151
## 2  -7.4422233      -4.777441      -2.0019531
## 3  -2.2803323      -4.760546      -1.7651573
## 4   0.6672042       1.089151       2.9429192
## 5  -1.2268284       8.707438       0.9772637
## 6  -2.3948942      -1.893784       1.2929293
```

Round the percentages

```
relative_change_$bitcoin_change <- round(relative_change_$bitcoin_change, 4)
relative_change_$ripple_change <- round(relative_change_$ripple_change ,4)
relative_change_$tether_change <- round(relative_change_$tether_change ,4)
relative_change_$ethereum_change <- round(relative_change_$ethereum_change ,4)
relative_change_$dash_change <- round(relative_change_$dash_change ,4)
relative_change_$bc_cash_change <- round(relative_change_$bc_cash_change ,4)
relative_change_$litecoin_change <- round(relative_change_$litecoin_change ,4)
```

Analysis

Relative Change

Tidy the data for analysis. The mean relative change shows the average changes.

```
tidy_change <- (gather(relative_change_, key= Currency, "relative_change", desc("Date")))
tidy_change %>% filter(relative_change !=0, is.finite(relative_change) == TRUE) %>%
  group_by(Currency) %>%  summarise(mean = mean(relative_change), count = n())
```

```
## # A tibble: 7 x 3
##          Currency        mean count
##             <chr>       <dbl> <int>
## 1  bc_cash_change  1.97931085   129
## 2  bitcoin_change -0.05745514  2249
## 3     dash_change  0.98085324  1360
## 4 ethereum_change  0.93719399   849
## 5 litecoin_change  0.44710412  1602
## 6   ripple_change  0.44249058  1581
## 7   tether_change  0.02202607   399
```

From the data, we can see that Bitcoin prices has a negative relative change. This might be due to some skews in our data. Lets limit high percentage changes.

```
tidy_change <- (gather(relative_change_, key= Currency, "relative_change", desc("Date")))
tidy_change %>% filter(relative_change !=0, is.finite(relative_change) == TRUE, relative_change < 50, r
  group_by(Currency) %>%  summarise(mean = mean(relative_change), count = n())
```

```
## # A tibble: 7 x 3
##          Currency       mean count
##             <chr>      <dbl> <int>
## 1  bc_cash_change 1.16503465   127
## 2  bitcoin_change 0.41088594  2233
## 3     dash_change 0.57381055  1355
## 4 ethereum_change 0.90714835   846
## 5 litecoin_change 0.22708623  1598
## 6   ripple_change 0.29571635  1578
## 7   tether_change 0.02202607   399
```

We see that when we limit the relative changes between -50 and 50, there is significant changes to the mean, but not the count.

```
tidy_change <- (gather(relative_change_, key= Currency, "relative_change", desc("Date")))
tidy_change %>% filter(relative_change !=0, is.finite(relative_change) == TRUE, relative_change < 10, r
  group_by(Currency) %>%  summarise(mean = mean(relative_change), count = n())
```

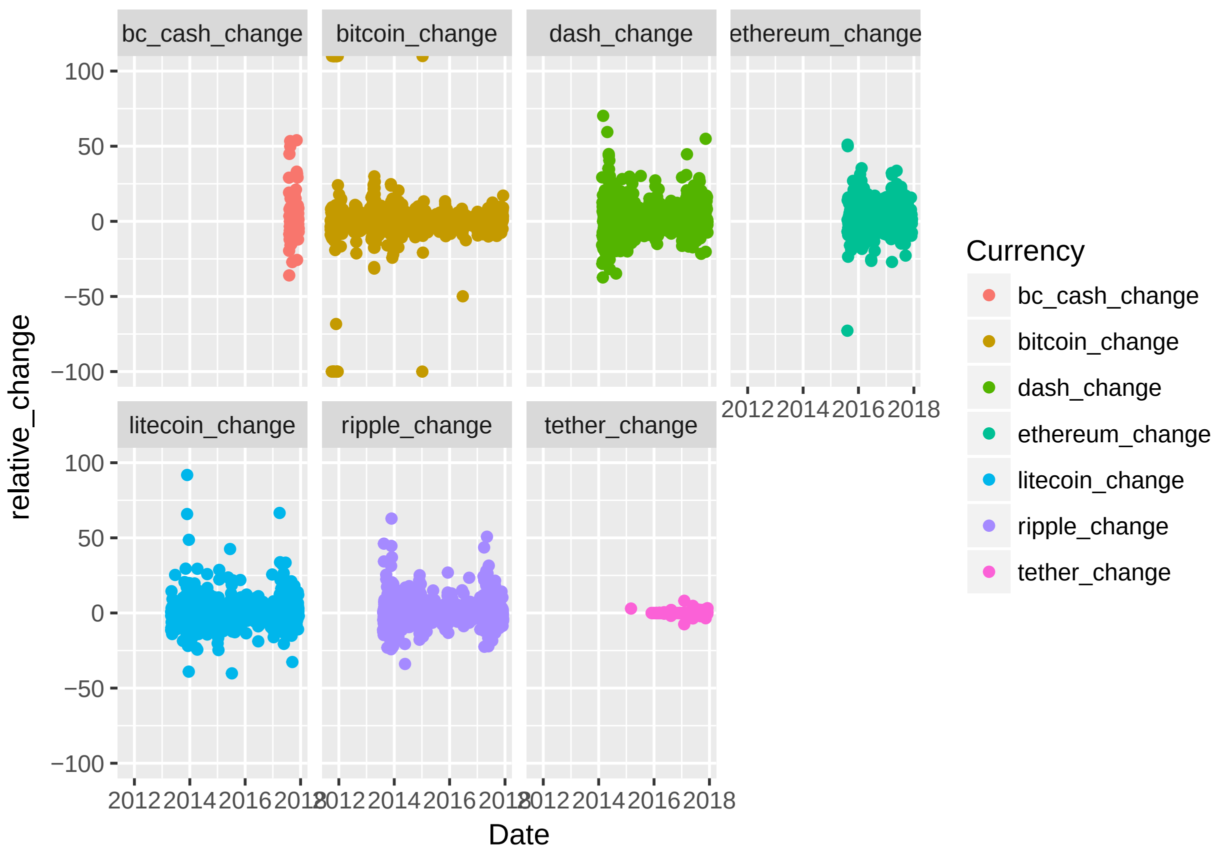
```
## # A tibble: 7 x 3
##          Currency        mean count
##             <chr>       <dbl> <int>
## 1  bc_cash_change -0.71430515    97
## 2  bitcoin_change  0.36466306  2152
## 3     dash_change -0.12132624  1189
## 4 ethereum_change -0.12964241   738
## 5 litecoin_change -0.08903114  1461
## 6   ripple_change -0.14734908  1465
## 7   tether_change  0.02202607   399
```

When we limit the relative change to between -10 and 10, most of the mean is negative. This means for 5/7 observations, the increases are due to high shifts in the market in the day, but most days, they are losing value. This is opposite for Bitcoin, which is increasing on most days and its losses come from large drops in the market. For Tether, the market looks surprisingly stable. This may be due to it being a small player in the market.

Let's visualize the relative changes.

```
ggplot(data =tidy_change[which(tidy_change$relative_change != 0),]) +
  geom_point(mapping = aes(x= Date, y= relative_change, color= Currency,fill = Currency), position = "j
  facet_wrap(~ Currency, nrow = 2)+
  ylim(-100,100)
```

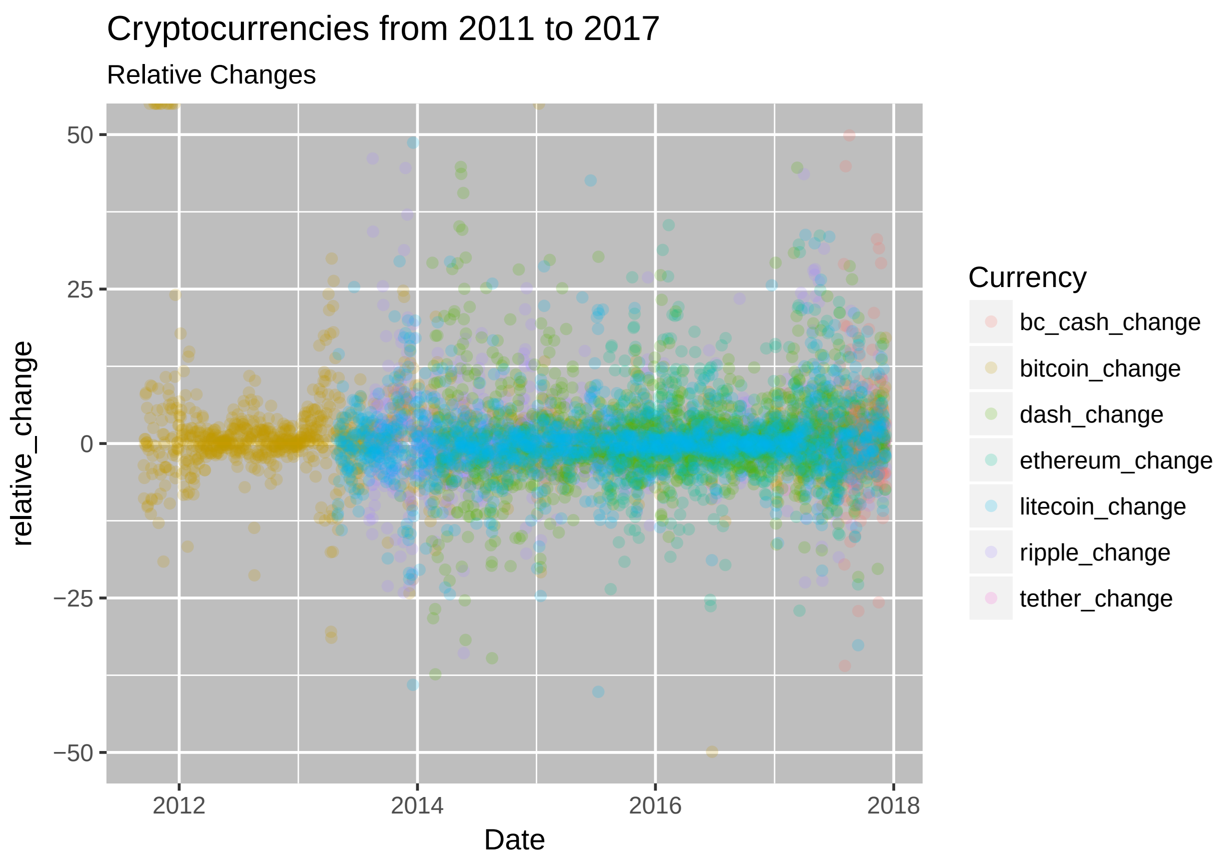
```
## Warning: Removed 11 rows containing missing values (geom_point).
```



From the visualization, we can see that the spread is very large for most of the currencies with exception to tether. Visualizing the relative changes together.

```
ggplot(data =tidy_change[which(tidy_change$relative_change != 0),]) +
  geom_point(mapping = aes(x= Date, y= relative_change, color= Currency,fill = Currency),alpha = 1/5, p
  ylim(-50,50)+
  theme(panel.background = element_rect(fill = 'gray'))+
  ggtitle("Cryptocurrencies from 2011 to 2017",subtitle = "Relative Changes")
```

```
## Warning: Removed 33 rows containing missing values (geom_point).
```



Gain to Loss Ratio

Let's dive deeper into how the currencies fluctuate and look at gain to loss ratio. Categorize for each relative change percentage to 'gain', 'loss' or 'no change'.

```
tidy_change$lost_gain <- ifelse(tidy_change$relative_change > 0, "gain",
                                ifelse(tidy_change$relative_change ==0,"no,change","loss"))
tidy_change[sample(nrow(iris), 10), ]
```

```
##           Date       Currency relative_change lost_gain
## 81  2017-09-18 bitcoin_change          7.5544      gain
## 69  2017-09-30 bitcoin_change          3.0746      gain
## 102 2017-08-28 bitcoin_change         -1.3231      loss
## 27  2017-11-11 bitcoin_change         -5.3339      loss
## 60  2017-10-09 bitcoin_change          4.3199      gain
## 82  2017-09-17 bitcoin_change         -1.8633      loss
## 73  2017-09-26 bitcoin_change          2.5388      gain
## 51  2017-10-18 bitcoin_change         -4.9985      loss
## 43  2017-10-26 bitcoin_change          5.0117      gain
## 78  2017-09-21 bitcoin_change         -4.8734      loss
```

Untidy the data for analysis

```
#All
ratio<- tidy_change %>% group_by(Currency) %>% count(lost_gain)
gain_loss_ratio <- ratio %<>% spread(lost_gain, n) %<>%   mutate(ratio = gain/loss)
gain_loss_ratio <- select(gain_loss_ratio,Currency,ratio)
```

```r
#Limit -100 to 100
ratio100<- tidy_change %>% filter(relative_change < 100, relative_change > -100) %>% group_by(Currency)
gain_loss_ratio100 <- ratio100 %<>% spread(lost_gain, n) %<>%   mutate(ratio100 = gain/loss)
gain_loss_ratio100 <- select(gain_loss_ratio100,Currency,ratio100)

#Limit -10 to 10
ratio1<- tidy_change %>% filter(relative_change < 10, relative_change > -10) %>% group_by(Currency) %>%
gain_loss_ratio1 <- ratio1 %<>% spread(lost_gain, n) %<>%   mutate(ratio10 = gain/loss)
gain_loss_ratio1 <- select(gain_loss_ratio1,Currency,ratio10)
```

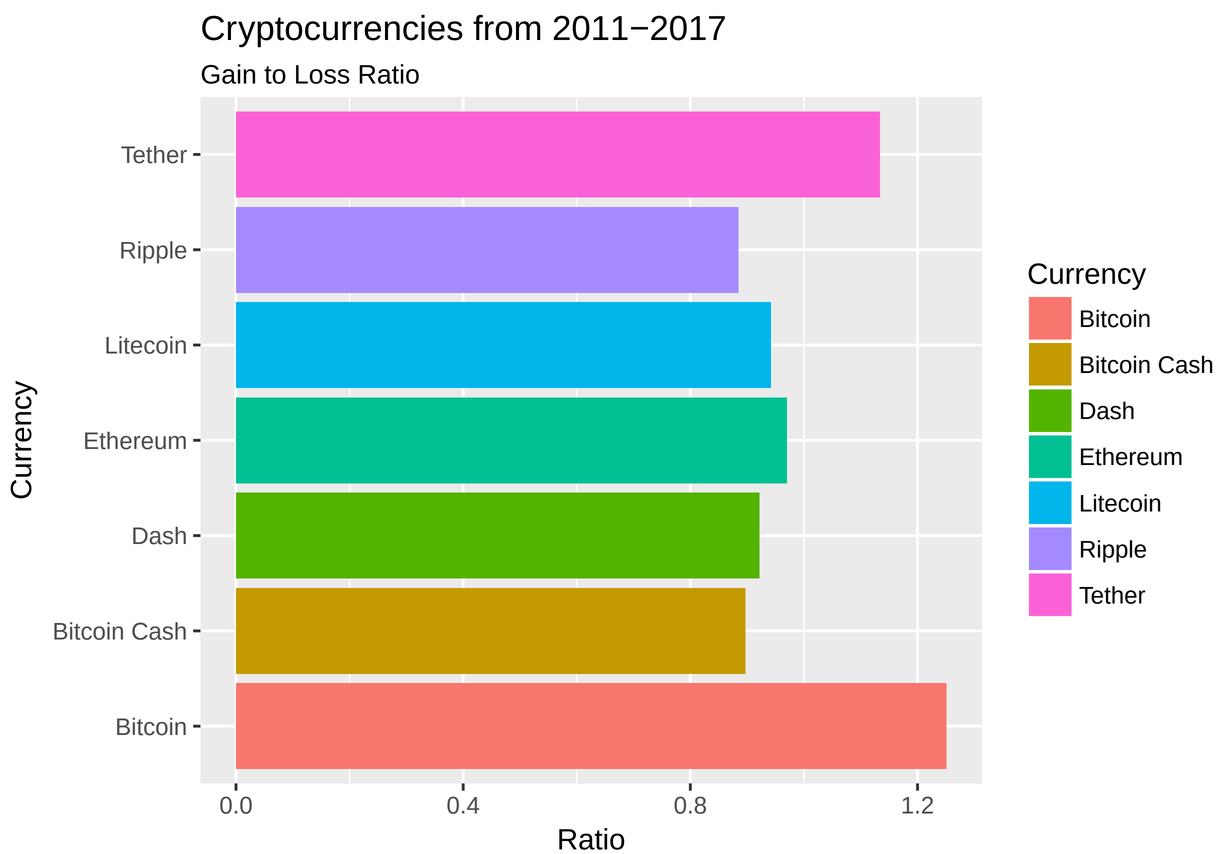
```r
All_ratios <- inner_join(gain_loss_ratio,inner_join(gain_loss_ratio100,gain_loss_ratio1,by = "Currency"
All_ratios$Currency <- c("Bitcoin Cash", "Bitcoin", "Dash", "Ethereum","Litecoin","Ripple","Tether")
All_ratios
```

```
## # A tibble: 7 x 4
## # Groups:   Currency [?]
##       Currency     ratio  ratio100   ratio10
##          <chr>     <dbl>     <dbl>     <dbl>
## 1 Bitcoin Cash 0.8970588 0.8970588 0.7017544
## 2      Bitcoin 1.2507463 1.2520161 1.2510460
## 3         Dash 0.9209040 0.9180791 0.8405573
## 4     Ethereum 0.9698376 0.9698376 0.8403990
## 5     Litecoin 0.9418182 0.9406061 0.9023438
## 6       Ripple 0.8843862 0.8831943 0.8381430
## 7       Tether 1.1336898 1.1336898 1.1336898
```

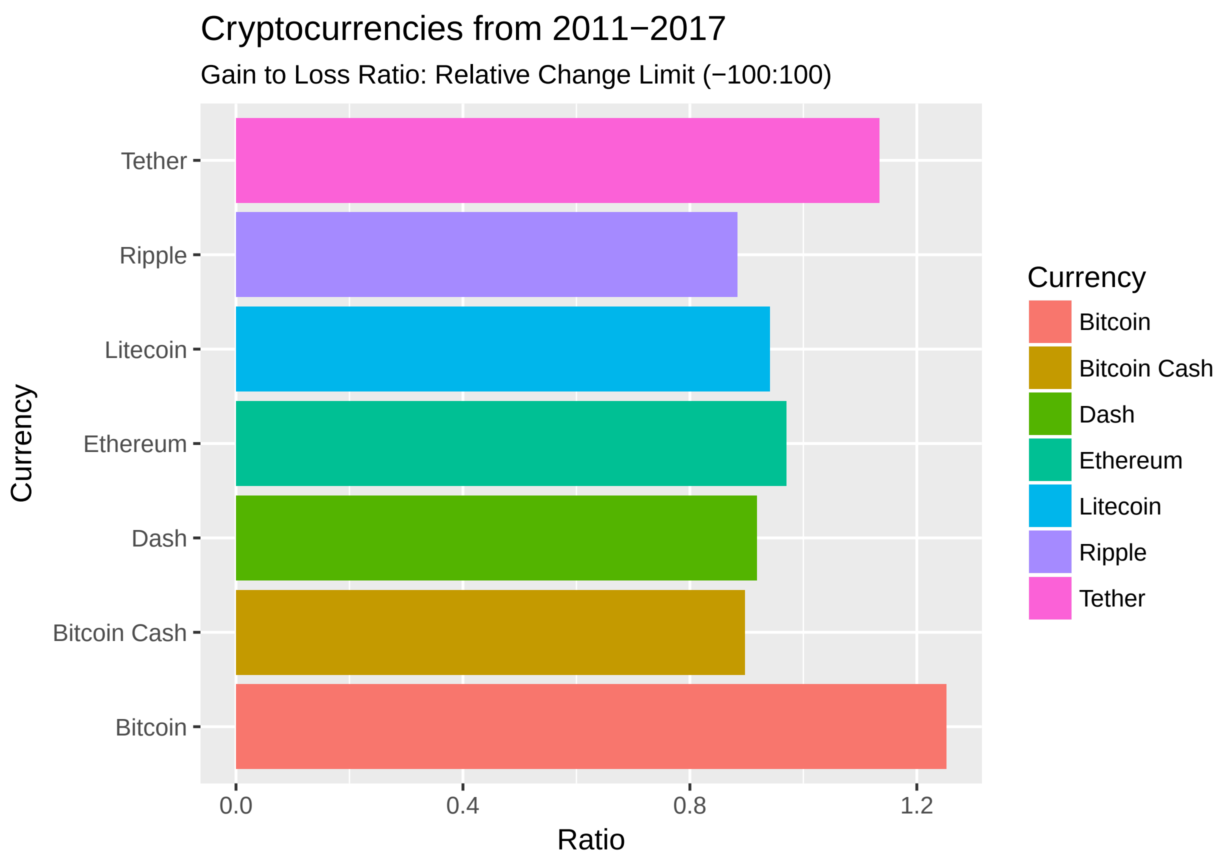
The gain to loss ratio table shows that Bitcoin has the highest ratio with over +25% margin over total losses. 2ND is Tether with a +12% margin over total losses. The rest have more losses than gains in their history.

Lets visualize the results.

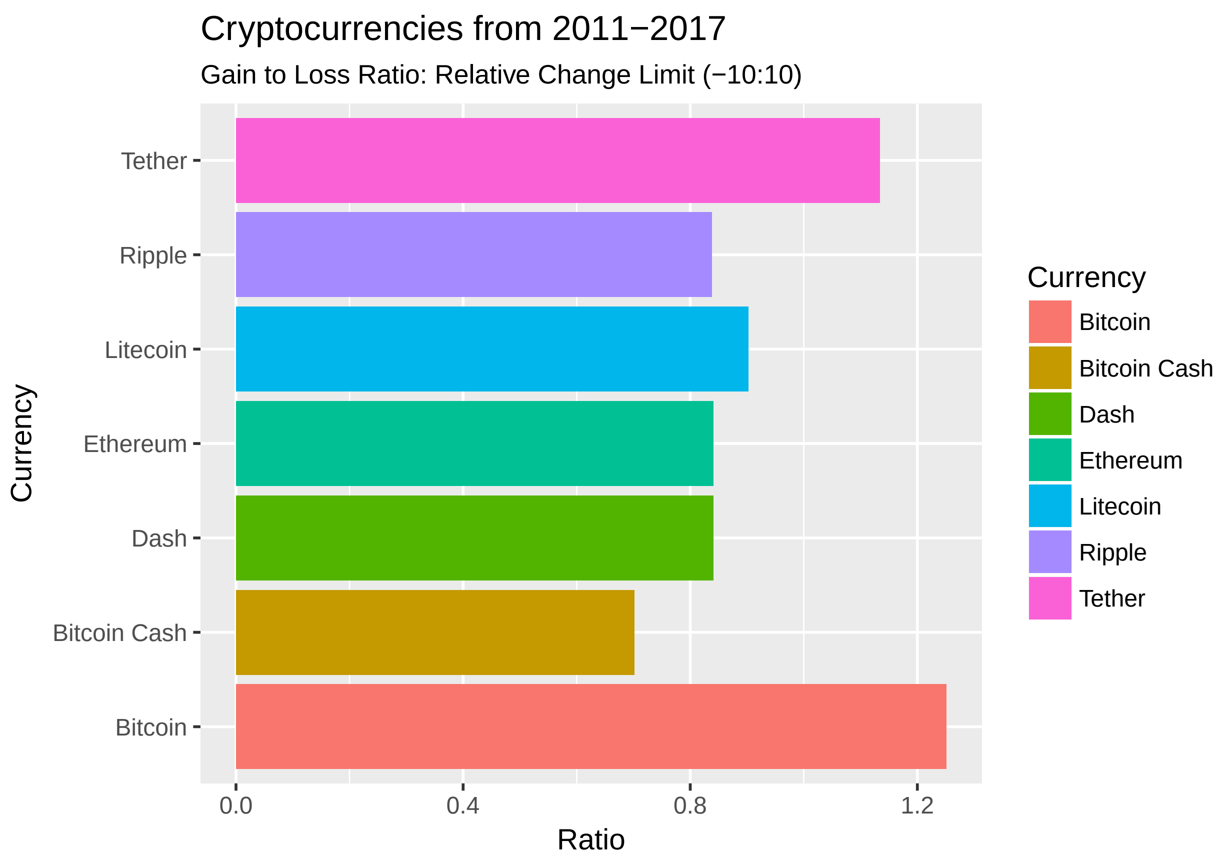
```r
ratio_bar <- ggplot(All_ratios, mapping = aes(y= ratio,x= Currency, fill=Currency))  +
  geom_bar(stat = "identity") +
  ggtitle("Cryptocurrencies from 2011-2017", subtitle = "Gain to Loss Ratio")+
  labs(y= "Ratio")
ratio_bar + coord_flip()
```




```
ratio_bar100 <- ggplot(All_ratios, mapping = aes(y= ratio100,x= Currency, fill=Currency))  +
  geom_bar(stat = "identity") +
  ggtitle("Cryptocurrencies from 2011-2017", subtitle = "Gain to Loss Ratio: Relative Change Limit (-10
  labs(y= "Ratio")
ratio_bar100 + coord_flip()
```




```r
ratio_bar1 <- ggplot(All_ratios, mapping = aes(y= ratio10,x= Currency, fill=Currency))+
  geom_bar(stat = "identity") +
  ggtitle("Cryptocurrencies from 2011-2017", subtitle = "Gain to Loss Ratio: Relative Change Limit (-10
  labs(y= "Ratio")
ratio_bar1 + coord_flip()
```



When we compare the overall gains to losses, we can see that there is a significant drop for Ripple, Litecoin, Ethereum, Dash and Bitcoin cash.

Probability

Lets look at the probability of currencies increasing on a given day, based on the count of gains and losses.

```
gain_loss_counts <- tidy_change %>% group_by(Currency, lost_gain) %>% summarise(n=n())
gain_loss_counts <- spread(gain_loss_counts, lost_gain, n)
gain_loss_counts <- mutate(gain_loss_counts, probability = (gain/(gain+loss)))
probability <- select(gain_loss_counts, Currency, probability)

gain_loss_counts100 <- tidy_change %>% filter(relative_change < 100, relative_change > -100) %>%
                      group_by(Currency, lost_gain) %>% summarise(n=n())
gain_loss_counts100 <- spread(gain_loss_counts100, lost_gain, n)
gain_loss_counts100 <- mutate(gain_loss_counts100, probability100 = (gain/(gain+loss)))
probability100 <- select(gain_loss_counts100, Currency, probability100)

gain_loss_counts10 <- tidy_change %>% filter(relative_change < 10, relative_change > -10) %>%
             group_by(Currency, lost_gain) %>% summarise(n=n())
gain_loss_counts10 <- spread(gain_loss_counts10, lost_gain, n)
gain_loss_counts10 <- mutate(gain_loss_counts10, probability10 = (gain/(gain+loss)))
probability10 <- select(gain_loss_counts10, Currency, probability10)

All_probability <- inner_join(probability,inner_join(probability100,probability10,by = "Currency"),by=
All_probability$Currency <- c("Bitcoin Cash", "Bitcoin", "Dash", "Ethereum","Litecoin","Ripple","Tether
```

```r
All_probability
```

```
## # A tibble: 7 x 4
## # Groups:   Currency [?]
##       Currency probability probability100 probability10
##          <chr>       <dbl>          <dbl>         <dbl>
## 1 Bitcoin Cash   0.4728682      0.4728682     0.4123711
## 2      Bitcoin   0.5557029      0.5559534     0.5557621
## 3         Dash   0.4794118      0.4786451     0.4566863
## 4     Ethereum   0.4923439      0.4923439     0.4566396
## 5     Litecoin   0.4850187      0.4846971     0.4743326
## 6       Ripple   0.4693232      0.4689873     0.4559727
## 7       Tether   0.5313283      0.5313283     0.5313283
```

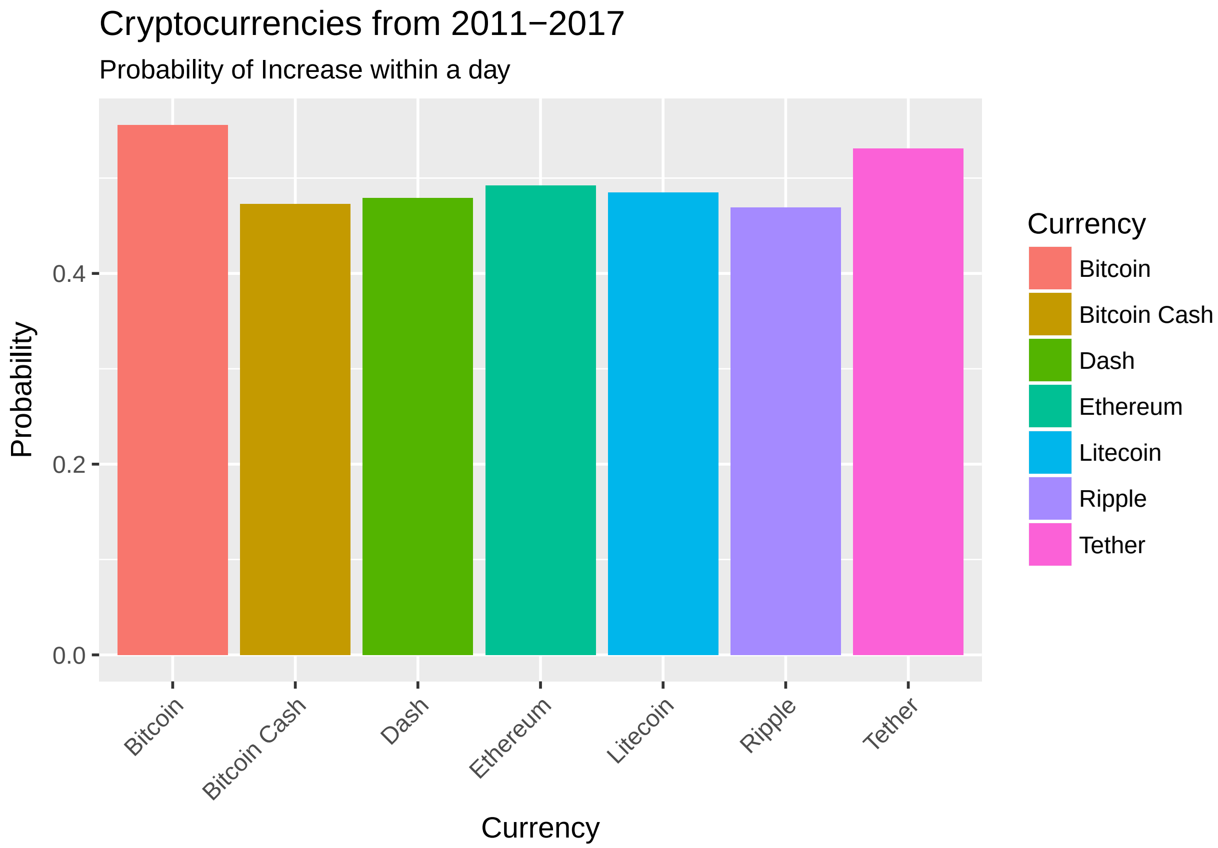
The probability table shows that Bitcoin has the highest chance of increase on a given day with 55% and that stays consistent when outliers are removed. Second is Tether with 53%. The rest is in the 47%-49% range, but decreases when outliers are removed. out of the remaining 5, Litecoin is most consistent, dropping from 48.5% to 47.4%.

Lets visualize the results.

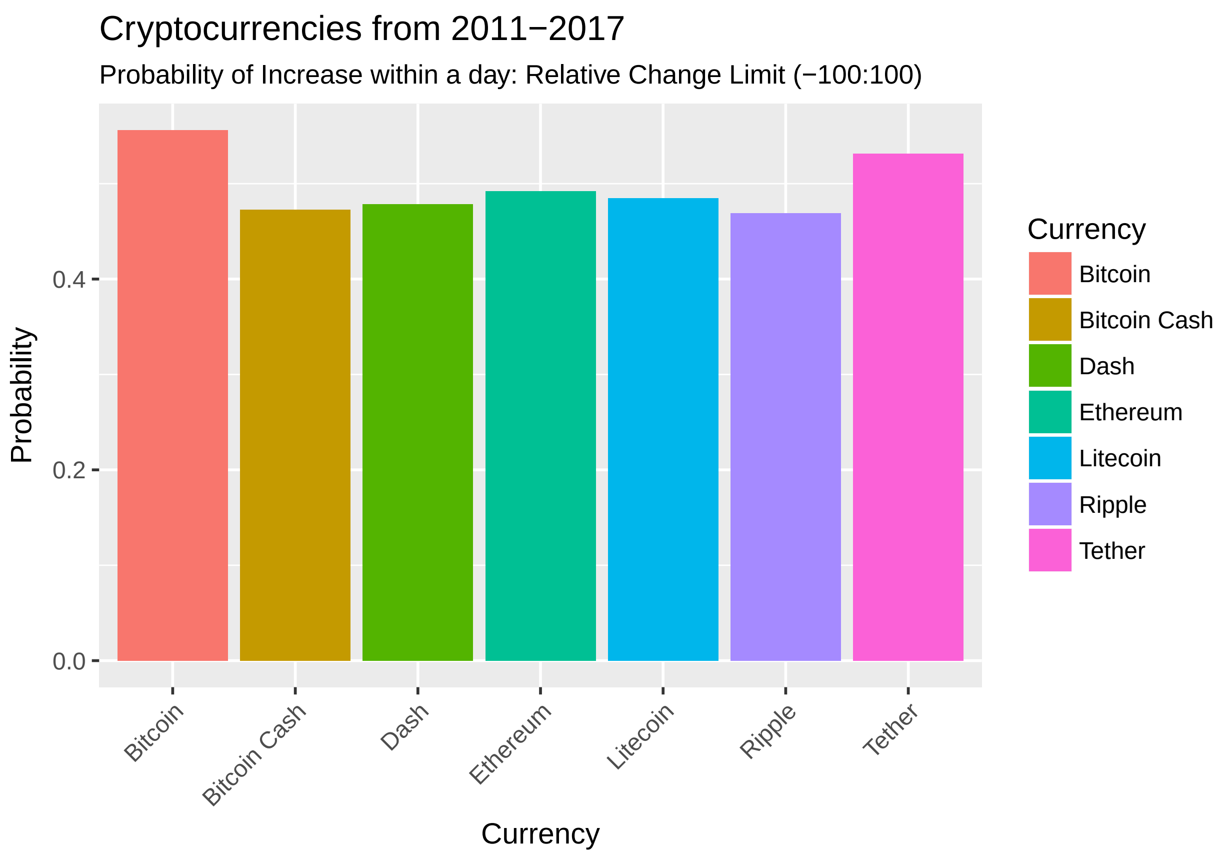
```r
facet_wrap(~All_ratios,nrow = 3)
```

```
## <ggproto object: Class FacetWrap, Facet>
##     compute_layout: function
##     draw_back: function
##     draw_front: function
##     draw_labels: function
##     draw_panels: function
##     finish_data: function
##     init_scales: function
##     map: function
##     map_data: function
##     params: list
##     render_back: function
##     render_front: function
##     render_panels: function
##     setup_data: function
##     setup_params: function
##     shrink: TRUE
##     train: function
##     train_positions: function
##     train_scales: function
##     super:  <ggproto object: Class FacetWrap, Facet>
```

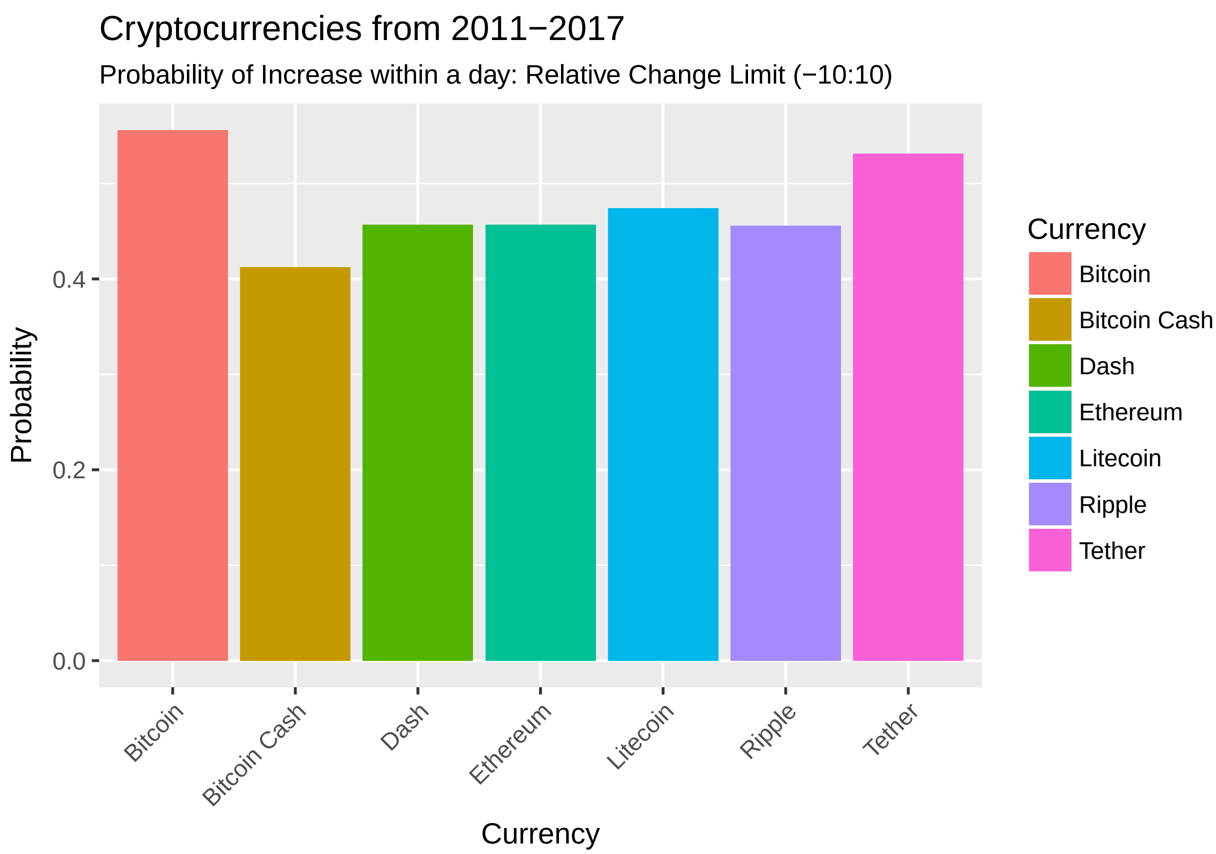
```r
probability_bar <- ggplot(All_probability, mapping = aes(y=probability,x= Currency, fill=Currency))  +
  geom_bar(stat = "identity") +
  ggtitle("Cryptocurrencies from 2011-2017", subtitle = "Probability of Increase within a day")+
  labs(y= "Probability") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

probability_bar
```




```
probability_bar100 <- ggplot(All_probability, mapping = aes(y= probability100,x= Currency, fill=Currency
  geom_bar(stat = "identity") +
  ggtitle("Cryptocurrencies from 2011-2017", subtitle = "Probability of Increase within a day: Relative
  labs(y= "Probability") +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
probability_bar100
```




```r
probability_bar1 <- ggplot(All_probability, mapping = aes(y=probability10,x= Currency, fill=Currency))+
  geom_bar(stat = "identity") +
  ggtitle("Cryptocurrencies from 2011-2017", subtitle = "Probability of Increase within a day: Relative
  labs(y= "Probability")+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
probability_bar1
```



Conclusion

Based on the day to day relative change of the currencies, we have some insight on how the market fluctuates. We saw that the relative change showed that there was an inverse relationship between Bitcoin and most of the currencies. Bitcoin Cash, Dash, Ethereum, Litecoin, and Ripple's decrease most days with few large increases making up for the losses. While Bitcoin and Tether increase most days, with large drops evening it out.

When we looked at the gain to loss ratio, Bitcoin and Tether had a surplus in increases over losses, while the other 5 had a ratio of below 1. This supports the previous conclusion that there are more gains than losses and when we look at probabilities of this. We see that Bitcoin and Tether have a chance of increase more than 50% in a given day.

For someone that is interested in investing in Cryptocurrency, based on the analysis above, their best bet would be to invest in Bitcoin since. Since there isn't much more information on Tether, more in depth research needs to be done for a conclusion. If someone is looking to invest in another currency, the data tells us that Litecoin has the highest upside.

Overall, since the market moves so fast, a day to day analysis might not be enough to come up with an accurate assumption.