Analysis of Relationship Between Twitter and Bitcoin Prices

Nicholas Piccaro, Andy Donis-Paz, Mari Dudek, Kiley Krosky

Loading Libraries

```
library(ggplot2)
library(tidyr)
library(base)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

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

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

```
library(stringr)
library(MASS)
```

```
##
## Attaching package: 'MASS'

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

```
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(zoo)
```

```
##
## Attaching package: 'zoo'

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

Data PreProcessing

As data produced by Hadoop does not always appear in such a graceful form, the data requires small amounts of date-time configuring as well as some formatting. Date time format needed to be fixed, as well removing commas from numbers and changing prefixes to make it R friendly.

```r
## Data Processing
# Top Comments
topComments <- read.delim("top100comments.txt",header=FALSE)
colnames(topComments) <- c("ID","comment_count","date", "username")
topComments<-topComments[,-5]

#Top Retweets
topRetweets <- read.delim("top100retweets.txt",header=FALSE)
colnames(topRetweets) <- c("ID","retweet_count","date", "username")
topRetweets<-topRetweets[,-5]

#COunts
counts<-read.delim("monthlycount.txt", header=FALSE)
colnames(counts) <- c("month","count")
l<-as.Date(paste(counts$month,"-01",sep=""))
counts$month<-l
#Daily Price
dailyPrice<-read.delim("dailybitcoinprice.txt", header=FALSE)
dailyPrice<-separate(data = dailyPrice, col = V2, into = c("day1", "day2","day3","day4","day5","day6","d
```

```
## Warning: Expected 7 pieces. Additional pieces discarded in 4145 rows [1, 2, 3,
## 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```r
dailyPrice$day1<-as.numeric(gsub(",","",dailyPrice$day1))
dailyPrice$day2<-as.numeric(gsub(",","",dailyPrice$day2))
dailyPrice$day3<-as.numeric(gsub(",","",dailyPrice$day3))
dailyPrice$day4<-as.numeric(gsub(",","",dailyPrice$day4))
dailyPrice$day5<-as.numeric(gsub(",","",dailyPrice$day5))
dailyPrice$day6<-as.numeric(gsub(",","",dailyPrice$day6))
dailyPrice$day7<-as.numeric(gsub(",","",dailyPrice$day7))

#bitcoin
bitcoin<-read.delim("bitcoin.txt", header=TRUE,sep=",")
bitcoin$Date<-format(as.Date(bitcoin$Date, format = "%b %d, %Y"), "%Y-%m-%d")
bitSub<-bitcoin%>%
  dplyr::select(Date, Price, Open, High, Low, Vol., Change..)%>%
  filter(Date >= "2009-01-01" & Date <"2019-11-30")

bitSub$Price<-as.numeric(gsub(",","",bitSub$Price))
bitSub$High<-as.numeric(gsub(",","",bitSub$High))
bitSub$Low<-as.numeric(gsub(",","",bitSub$Low))
bitSub$Open<-as.numeric(gsub(",","",bitSub$Open))
Vol<-bitSub$Vol.
Vol1<- ifelse(str_detect(Vol, 'M'), 1e6, ifelse(str_detect(Vol, 'K'), 1e3, 1))
Vol2 <- Vol1 * as.numeric(str_remove(Vol, 'K|M'))
```

```
## Warning: NAs introduced by coercion
```

```r
bitSub$Vol.<-Vol2
##COunts
counts<-data.frame(date=counts$month,count=counts$count)

## LIkes
topLikes <- read.delim("top100likes.txt",header=FALSE)
colnames(topLikes) <- c("ID","like_count","date", "username")
```

```
topLikes<-topLikes[,-5]
```

Visualization and Exploration

In a Brief Summary of all of our datasets from HADOOP we find some important imformation like quartile information and we can usually find any unreasonable outliers but in our case the data looks orderly.

```
## Summary of Bitcoin Prices
summary(bitSub)
```

```
##      Date               Price             Open              High         
##  Length:3422        Min.   :    0.1   Min.   :    0.0   Min.   :    0.10  
##  Class :character   1st Qu.:   13.7   1st Qu.:   13.7   1st Qu.:   13.93  
##  Mode  :character   Median :  412.6   Median :  412.4   Median :  419.25  
##                     Mean   : 2116.8   Mean   : 2114.5   Mean   : 2179.18  
##                     3rd Qu.: 2809.2   3rd Qu.: 2804.8   3rd Qu.: 2879.85  
##                     Max.   :19345.5   Max.   :19346.6   Max.   :19870.60  
##                                                                           
##       Low              Vol.            Change..        
##  Min.   :    0.0   Min.   :      80   Length:3422       
##  1st Qu.:   13.4   1st Qu.:   22982   Class :character  
##  Median :  402.1   Median :   51750   Mode  :character  
##  Mean   : 2041.2   Mean   :  241682                     
##  3rd Qu.: 2699.5   3rd Qu.:  128822                     
##  Max.   :18750.9   Max.   :15430000                     
##                    NA's   :6
```

```
## Summary of Tweets of Monthly Counts of Tweets
summary(counts)
```

```
##       date                count          
##  Min.   :2009-01-01   Min.   :      4.0  
##  1st Qu.:2012-06-16   1st Qu.:    230.5  
##  Median :2014-12-01   Median :  48790.0  
##  Mean   :2014-11-15   Mean   : 174432.9  
##  3rd Qu.:2017-05-16   3rd Qu.:  99937.0  
##  Max.   :2019-11-01   Max.   :1785172.0
```

```
## Summary of Top 100 comments
summary(topComments)
```

```
##        ID             comment_count       date             username        
##  Min.   :9.388e+17   Min.   : 1715   Length:100         Length:100        
##  1st Qu.:9.885e+17   1st Qu.: 2070   Class :character   Class :character  
##  Median :1.082e+18   Median : 2828   Mode  :character   Mode  :character  
##  Mean   :1.077e+18   Mean   : 4363                                        
##  3rd Qu.:1.157e+18   3rd Qu.: 3666                                        
##  Max.   :1.195e+18   Max.   :77384
```

```
## Summary of Top 100 Retweets
summary(topRetweets)
```

```
##        ID             retweet_count        date             username        
##  Min.   :8.935e+17   Min.   :  5941   Length:100         Length:100        
##  1st Qu.:9.394e+17   1st Qu.:  7036   Class :character   Class :character  
##  Median :9.564e+17   Median :  8918   Mode  :character   Mode  :character  
##  Mean   :1.005e+18   Mean   : 17354
```

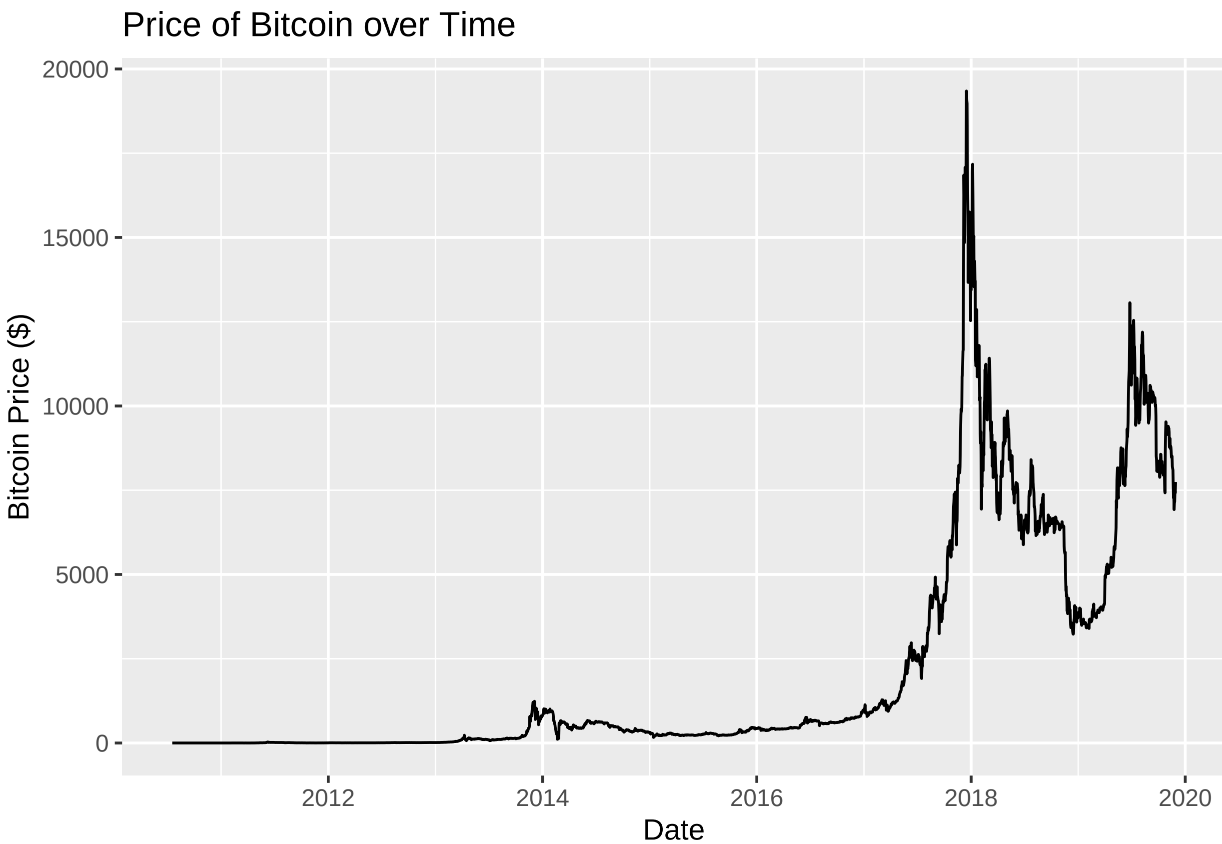
```
##  3rd Qu.:1.057e+18   3rd Qu.: 11476
##  Max.   :1.197e+18   Max.   :181240
```

```
## Summary of Daily Price Per Week
summary(dailyPrice)
```

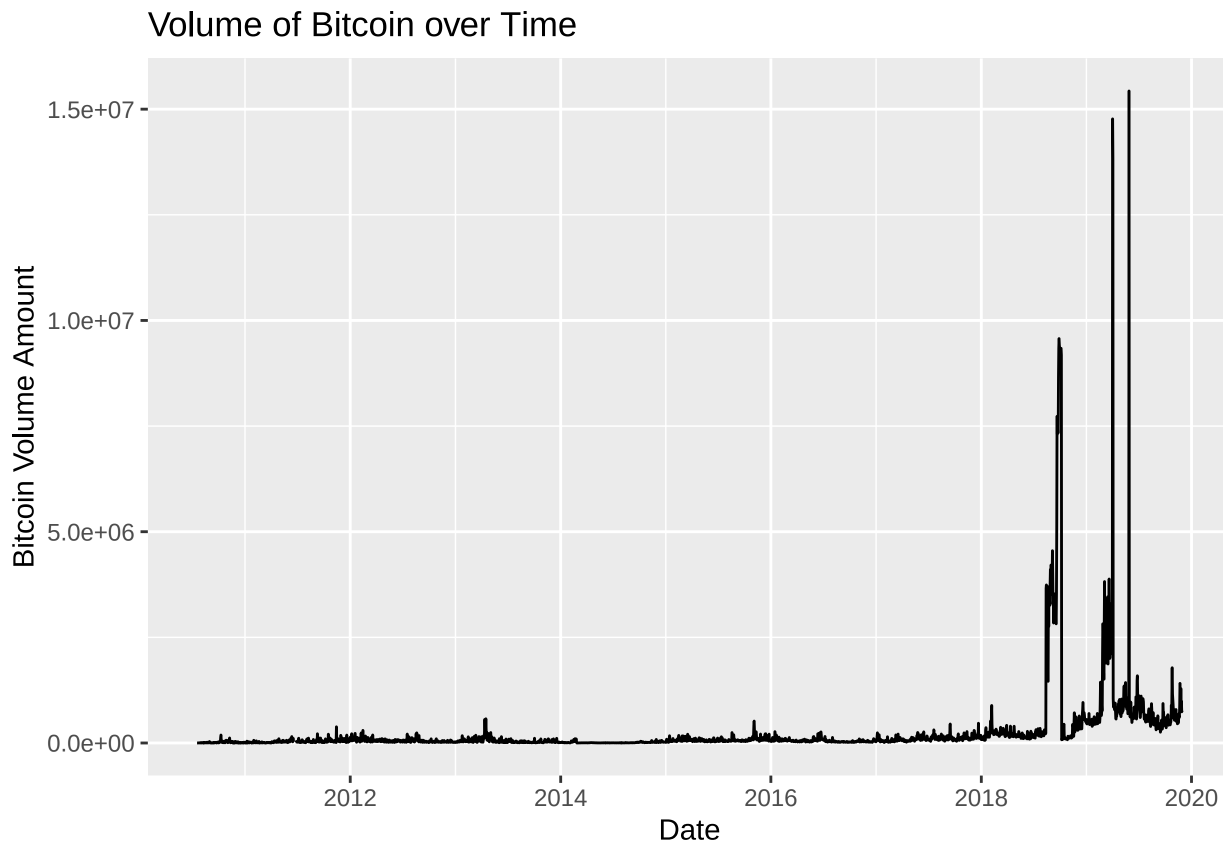
```
##       V1                 day1               day2               day3
##  Length:4145        Min.   :    0.1   Min.   :    0.1   Min.   :    0.1
##  Class :character   1st Qu.:  103.1   1st Qu.:  103.6   1st Qu.:  103.9
##  Mode  :character   Median :  608.1   Median :  608.6   Median :  608.6
##                     Mean   : 6453.4   Mean   : 6456.5   Mean   : 6458.1
##                     3rd Qu.: 7477.5   3rd Qu.: 7477.6   3rd Qu.: 7477.6
##                     Max.   :67527.9   Max.   :67527.9   Max.   :67527.9
##                                       NA's   :2         NA's   :3
##       day4               day5               day6               day7
##  Min.   :    0.1   Min.   :    0.1   Min.   :    0.1   Min.   :    0.1
##  1st Qu.:  104.0   1st Qu.:  104.0   1st Qu.:  104.0   1st Qu.:  104.1
##  Median :  608.7   Median :  608.9   Median :  609.0   Median :  609.1
##  Mean   : 6459.6   Mean   : 6461.2   Mean   : 6462.8   Mean   : 6464.3
##  3rd Qu.: 7477.7   3rd Qu.: 7480.4   3rd Qu.: 7483.1   3rd Qu.: 7485.8
##  Max.   :67527.9   Max.   :67527.9   Max.   :67527.9   Max.   :67527.9
##  NA's   :4         NA's   :5         NA's   :6         NA's   :7
```

In this we want to observe how Bitcoin has historically behaved over time. We see that big spikes for Price occcur in 2018 and 2019. Big spikes in Bitcoin volume occur in late 2018 and early 2019.

```
## Bitcoin Prices over time
bitplot<-ggplot(data=bitSub, aes(x=as.Date(Date), y=Price))+
  geom_line()+
  xlab("Date")+
  ylab("Bitcoin Price ($)")+
  ggtitle("Price of Bitcoin over Time")
bitplot
```

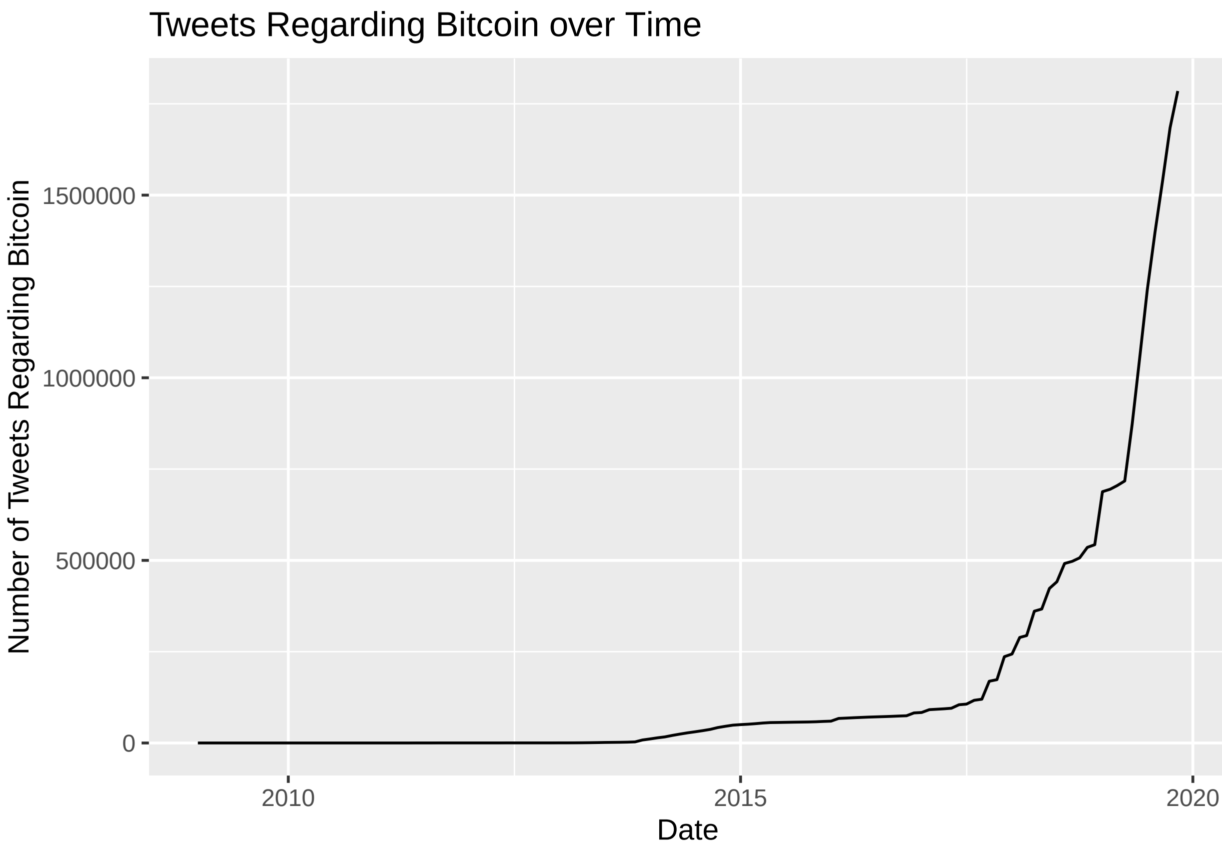



```r
## Volume of Bitcoin vs. Time
volplot<-ggplot(data=bitSub, aes(x=as.Date(Date), y=Vol.))+
  geom_line()+
  xlab("Date")+
  ylab("Bitcoin Volume Amount")+
  ggtitle("Volume of Bitcoin over Time")
volplot
```



then wanted to plot the number of Tweets Every Month regarding Bitcoin, as we see it progressively increases after 2018 as Bitcoin became more popular.

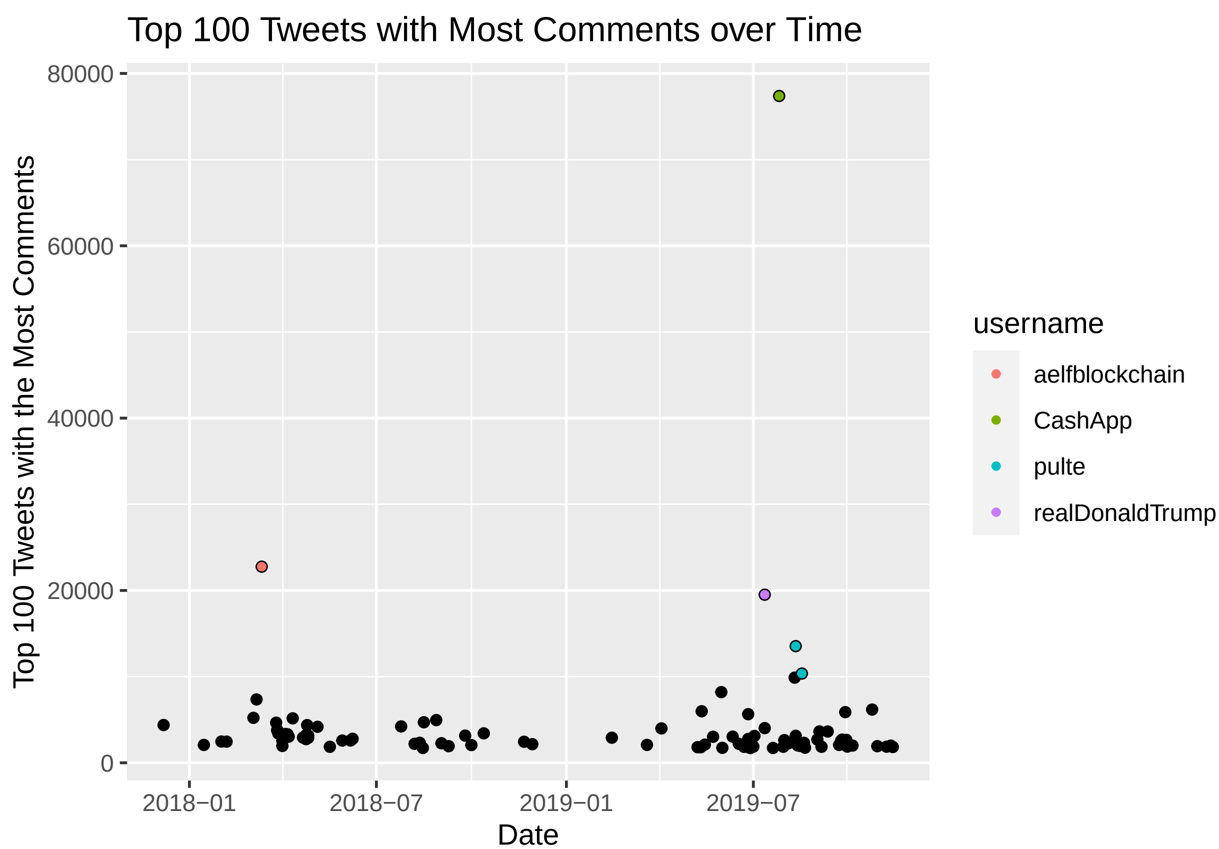
```
countplot<-ggplot(data=counts, aes(x=as.Date(date), y=count))+
  geom_line()+
  xlab("Date")+
  ylab("Number of Tweets Regarding Bitcoin")+
  ggtitle("Tweets Regarding Bitcoin over Time")
countplot
```



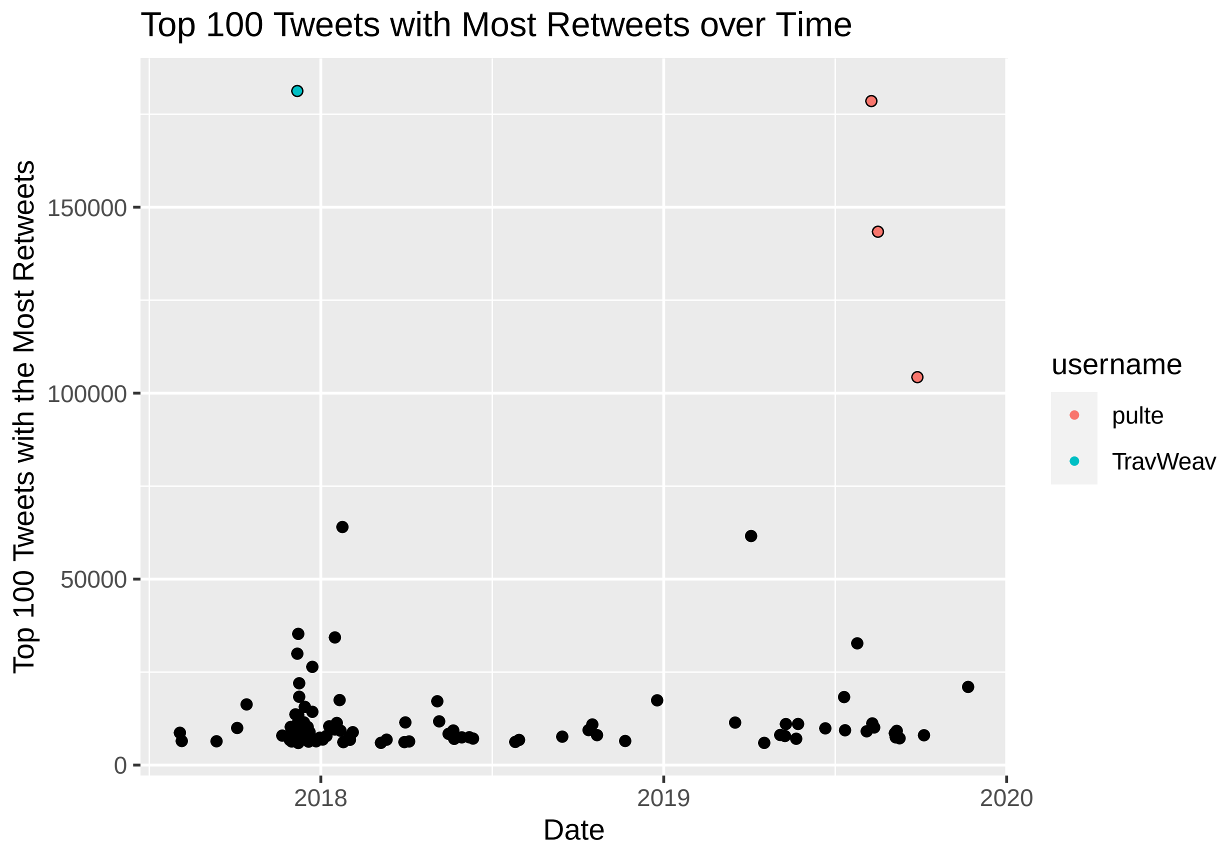
I

then plotted The Top 100 Tweets by Comment, Retweet, and Like count accordingly and additionaly highlighted the top usernames of each category.

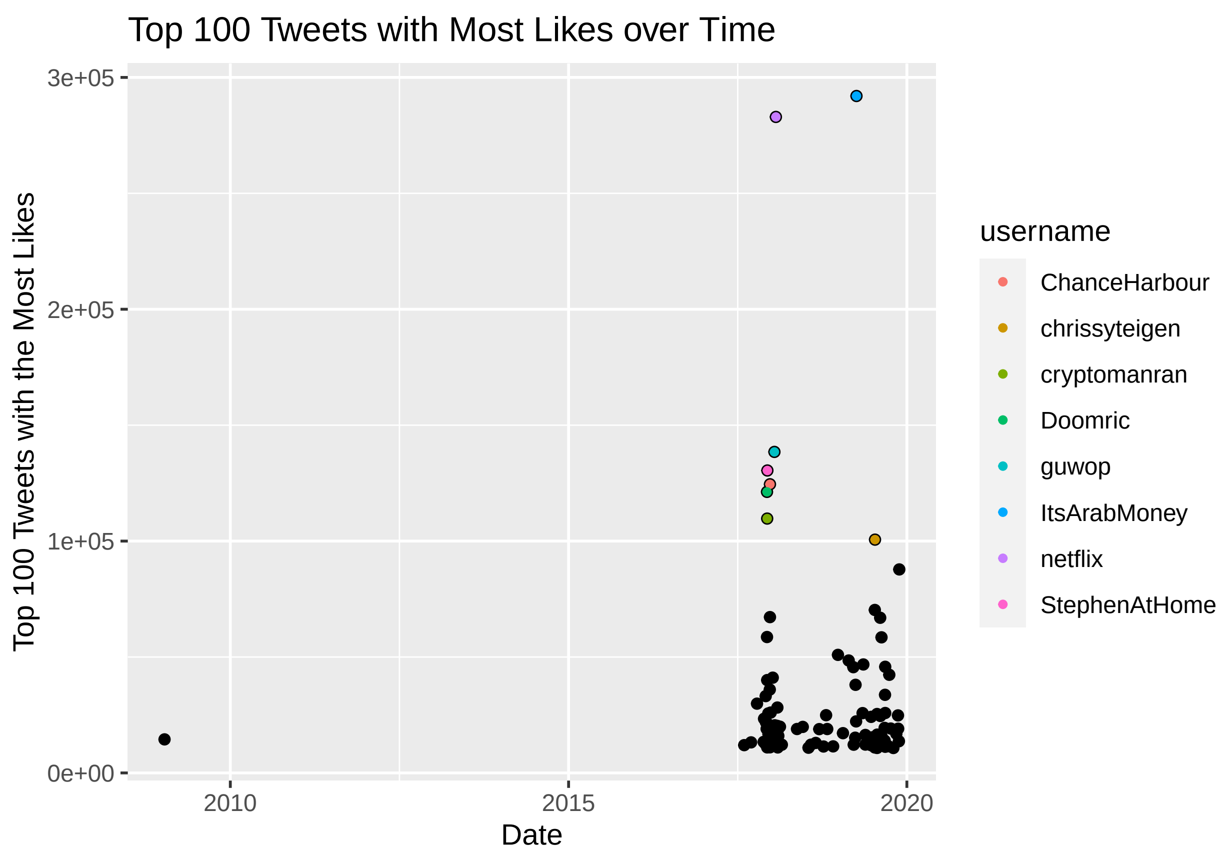
```
highlight_df <- topComments %>%
             filter(comment_count>=10000)
commentsplot<-ggplot(data=topComments, aes(x=as.Date(date), y=comment_count))+
  geom_point()+
  geom_point(data=highlight_df,
             aes(x=as.Date(date),y=comment_count,color=username),
             size=1)+
  xlab("Date")+
  ylab("Top 100 Tweets with the Most Comments")+
  ggtitle("Top 100 Tweets with Most Comments over Time")
commentsplot
```




```
highlight_df <- topRetweets %>%
              filter(retweet_count>=100000)
retweetplot<-ggplot(data=topRetweets, aes(x=as.Date(date), y=retweet_count))+
  geom_point()+
  geom_point(data=highlight_df,
              aes(x=as.Date(date),y=retweet_count,color=username),
              size=1)+
  xlab("Date")+
  ylab("Top 100 Tweets with the Most Retweets")+
  ggtitle("Top 100 Tweets with Most Retweets over Time")
retweetplot
```




```r
highlight_df <- topLikes %>%
             filter(like_count>=100000)
likeplot<-ggplot(data=topLikes, aes(x=as.Date(date), y=like_count))+
  geom_point()+
  geom_point(data=highlight_df,
             aes(x=as.Date(date),y=like_count,color=username),
             size=1)+
  xlab("Date")+
  ylab("Top 100 Tweets with the Most Likes")+
  ggtitle("Top 100 Tweets with Most Likes over Time")
likeplot
```



Analysis

Analysis Preprocessing

```
bitSub$count<-2
```

```
## Add twitter counts to bitcoin data
for(i in 1:3422) {       # for-loop over columns
  for(j in 1:119){
    count_string<-substr(counts[j,1], 1, 7)
    bit_string<-substr(bitSub[i,1], 1, 7)
    counted<-counts[j,2]
    if(all(count_string==bit_string)){

      bitSub[i,8] <- counted
    }
  }
}
```

```
bit_model<-bitSub[!(bitSub$count==2),]
```

Linear Regression

I will perform various linear regressions using different polynomials of the number of tweets variable against the Price response variable. We see in this simple linear regression model of degree 1, we have a R^2 correlation value of 0.526. Which suggests loose correlation, but still correlated.

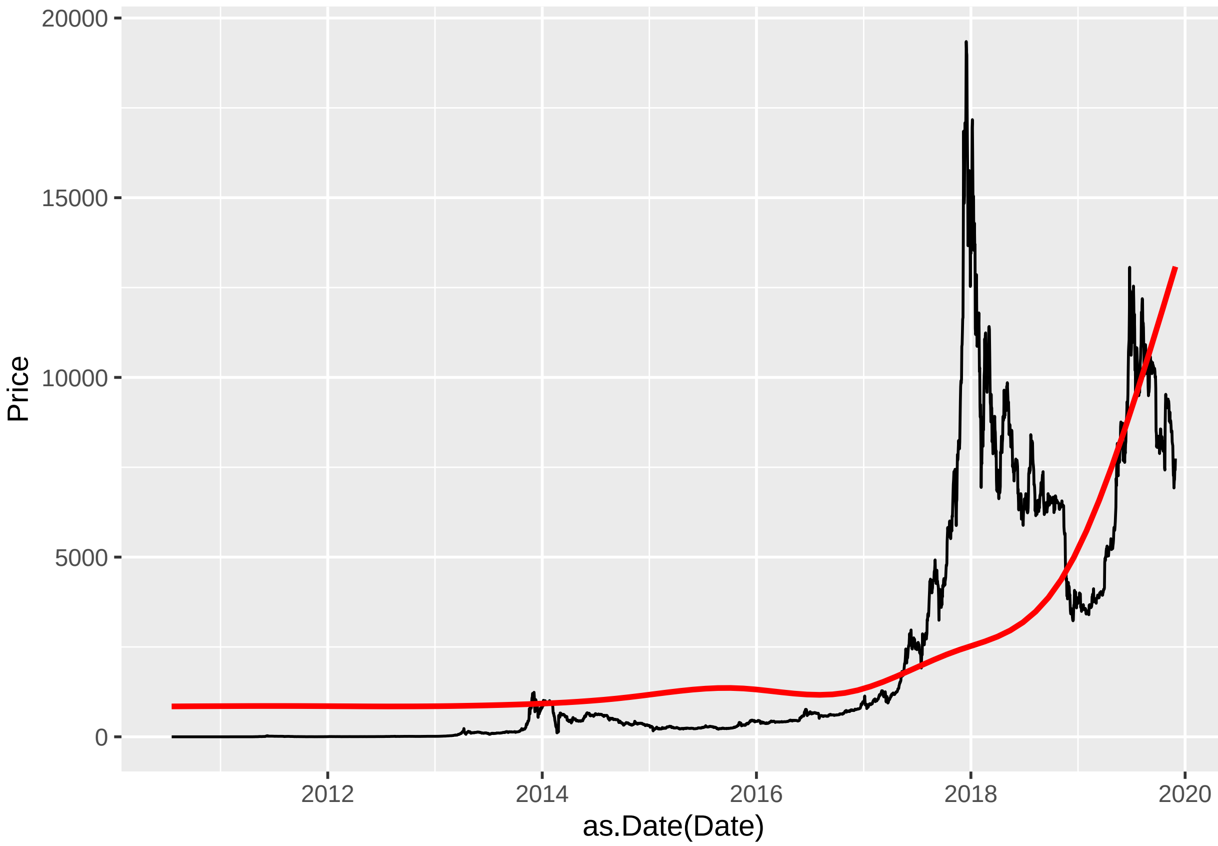
```r
bit.lm<-lm(Price~poly(count,1),data=bit_model)
summary(bit.lm)
```

```
##
## Call:
## lm(formula = Price ~ poly(count, 1), data = bit_model)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6337.9  -851.8  -810.3  -393.2 16851.4
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2155.22      40.94   52.64   <2e-16 ***
## poly(count, 1) 144922.11    2373.45   61.06   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2373 on 3359 degrees of freedom
## Multiple R-squared:  0.5261, Adjusted R-squared:  0.5259
## F-statistic:  3728 on 1 and 3359 DF,  p-value: < 2.2e-16
```

However if we plot the linear regression model over time with counts predicted Bitcoin price it actually traces the data quite well.

```r
predicted_df <- data.frame(price_pred = predict(bit.lm, bit_model), date=bit_model$Date)
bitplot<-ggplot(data=bit_model, aes(x=as.Date(Date), y=Price))+
  geom_line()+
  geom_smooth(color='red',data = predicted_df, aes(x=as.Date(date), y=price_pred))
bitplot
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Different Polynomial Regression Models

Polynomial Degree 2

We see as we add more polynomial degrees to the twitter count variable the the fit and the R^2 only become higher and more correlated, but then we beg a question of overfitting. I would suggest that, this Polynomial Degree 2 model is the best fit with R^2 of 0.651.

```
bit.lm<-lm(Price~poly(count,2),data=bit_model)
summary(bit.lm)
```

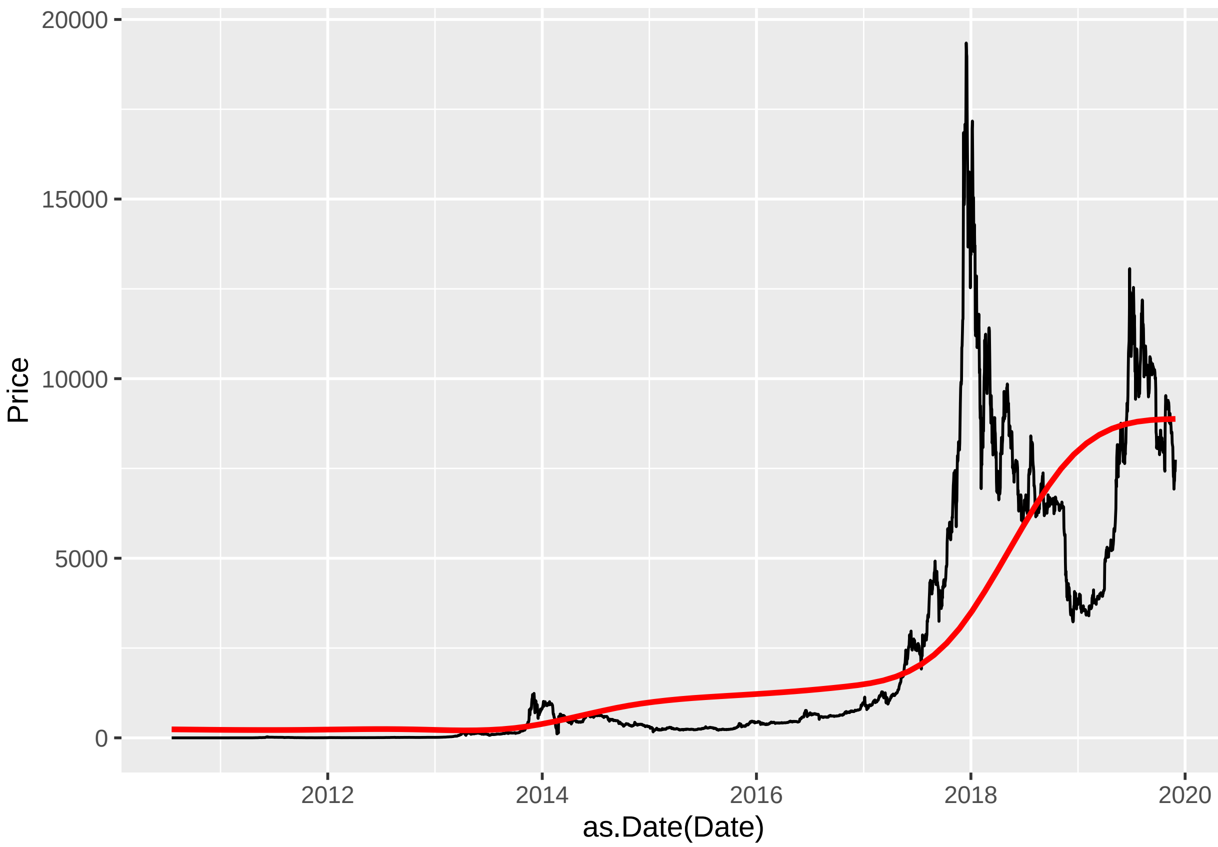
```
##
## Call:
## lm(formula = Price ~ poly(count, 2), data = bit_model)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4854.0  -727.2  -222.5   -92.3 15646.2
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2155.22      35.14   61.34   <2e-16 ***
## poly(count, 2)1 144922.11    2037.04   71.14   <2e-16 ***
## poly(count, 2)2 -70625.36    2037.04  -34.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2037 on 3358 degrees of freedom
## Multiple R-squared:  0.651,  Adjusted R-squared:  0.6508
```

```
## F-statistic:  3132 on 2 and 3358 DF,  p-value: < 2.2e-16
```

This model visually fits the data rather well considering the massive spike in 2018.

```
predicted_df <- data.frame(price_pred = predict(bit.lm, bit_model), date=bit_model$Date)
bitplot<-ggplot(data=bit_model, aes(x=as.Date(Date), y=Price))+
  geom_line()+
  geom_smooth(color='red',data = predicted_df, aes(x=as.Date(date), y=price_pred))
bitplot
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Polynomial Degree 3

As more degree terms are added R^2 only increases, but I fear the model is being overfit. As well as degree 3 and 4 I believe are rather unecessary.

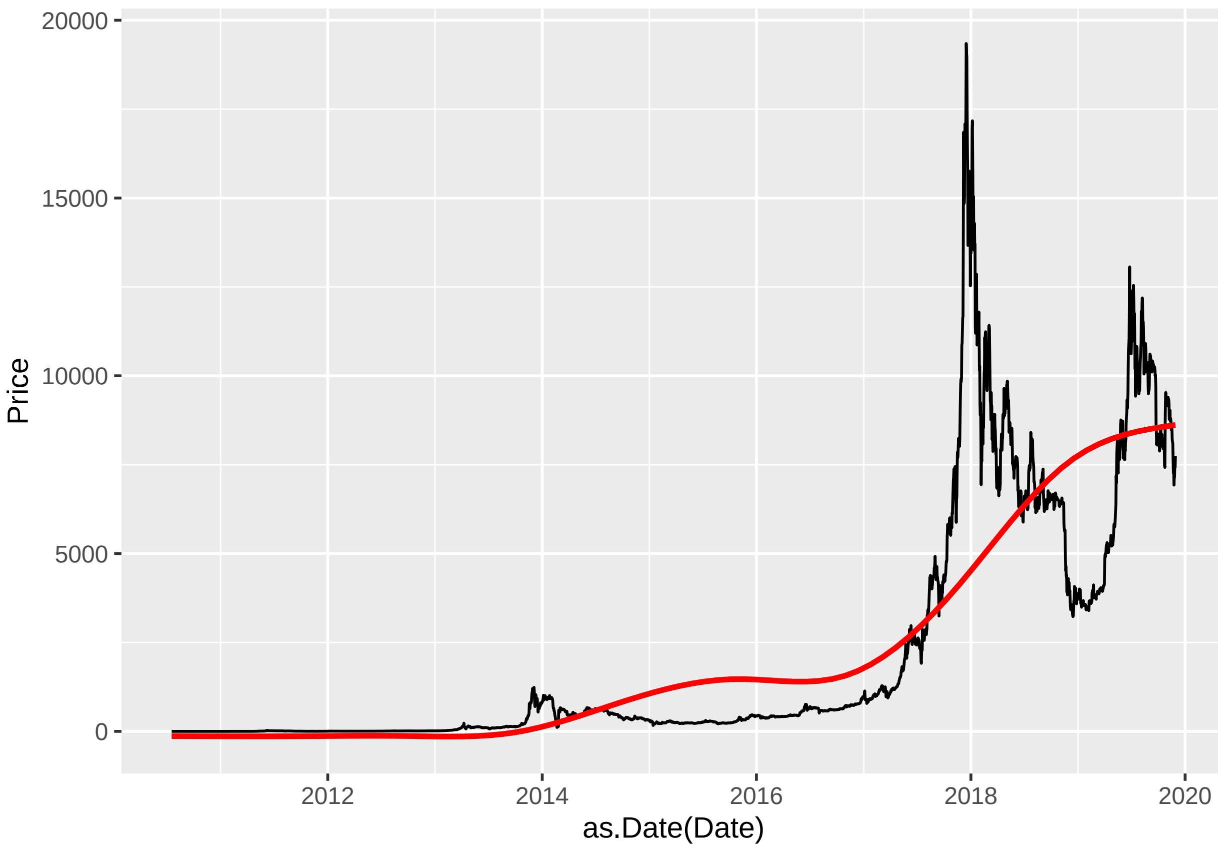
```
bit.lm<-lm(Price~poly(count,3),data=bit_model)
summary(bit.lm)
```

```
##
## Call:
## lm(formula = Price ~ poly(count, 3), data = bit_model)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4470.4  -977.8   137.4   188.7 14625.6
##
## Coefficients:
```

```
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        2155.22      33.08   65.15   <2e-16 ***
## poly(count, 3)1 144922.11    1917.94   75.56   <2e-16 ***
## poly(count, 3)2 -70625.36    1917.94  -36.82   <2e-16 ***
## poly(count, 3)3  39816.98    1917.94   20.76   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1918 on 3357 degrees of freedom
## Multiple R-squared:  0.6907, Adjusted R-squared:  0.6904
## F-statistic:  2499 on 3 and 3357 DF,  p-value: < 2.2e-16
```

```
predicted_df <- data.frame(price_pred = predict(bit.lm, bit_model), date=bit_model$Date)
bitplot<-ggplot(data=bit_model, aes(x=as.Date(Date), y=Price))+
  geom_line()+
  geom_smooth(color='red',data = predicted_df, aes(x=as.Date(date), y=price_pred))
bitplot
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Polynomial Degree 4

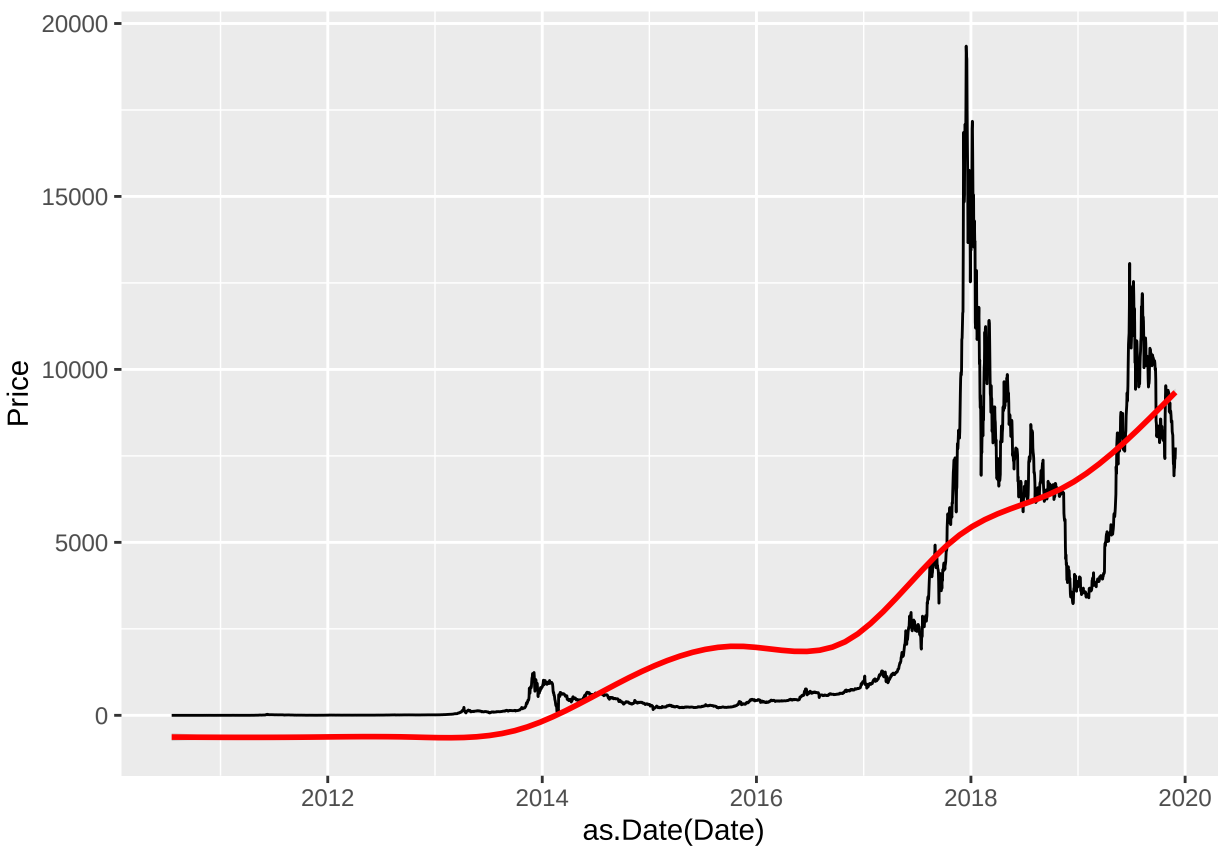
```
bit.lm<-lm(Price~poly(count,4),data=bit_model)
summary(bit.lm)
```

```
##
## Call:
## lm(formula = Price ~ poly(count, 4), data = bit_model)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3324.8 -1421.2   405.4   639.2 13502.1
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2155.22      29.69   72.59   <2e-16 ***
## poly(count, 4)1 144922.11    1721.35   84.19   <2e-16 ***
## poly(count, 4)2 -70625.36    1721.35  -41.03   <2e-16 ***
## poly(count, 4)3  39816.98    1721.35   23.13   <2e-16 ***
## poly(count, 4)4 -49038.79    1721.35  -28.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1721 on 3356 degrees of freedom
## Multiple R-squared:  0.7509, Adjusted R-squared:  0.7506
## F-statistic:  2530 on 4 and 3356 DF,  p-value: < 2.2e-16
```

```
predicted_df <- data.frame(price_pred = predict(bit.lm, bit_model), date=bit_model$Date)
bitplot<-ggplot(data=bit_model, aes(x=as.Date(Date), y=Price))+
  geom_line()+
  geom_smooth(color='red',data = predicted_df, aes(x=as.Date(date), y=price_pred))
bitplot
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Correlation

Next I wish to evaluate Correlation Between the Monthly Number of Tweets and Bitcoin price using 3 different Correlation methods. Pearson measures the bivariate correlation to detirmine the linear correlation between two variables. Kendall measures the ordinal association between two quantiles that measures strength and direction. The Spearman method measures strength between a monotonic correlation between two variables.

In our models we see that the Kendall and Spearman correlation coefficients are rather high at 0.799 and 0.941 corresponding. The Pearson correlation coefficient measures at 0.725, still fairly high suggesting that some higher level of correlation between the Twitter and Bitcoin prices.

```
cor(bit_model$count,bit_model$Price , method = "pearson")
```

```
## [1] 0.7252954
```

```
cor.test(bit_model$count,bit_model$Price, method="pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  bit_model$count and bit_model$Price
## t = 61.06, df = 3359, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7088686 0.7409359
## sample estimates:
##       cor
## 0.7252954
```

```
cor(bit_model$count,bit_model$Price , method = "kendall")
```

```
## [1] 0.7990948
```

```
cor.test(bit_model$count,bit_model$Price, method="kendall")
```

```
##
##  Kendall's rank correlation tau
##
## data:  bit_model$count and bit_model$Price
## z = 69.109, p-value < 2.2e-16
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.7990948
```

```
cor(bit_model$count,bit_model$Price , method = "spearman")
```

```
## [1] 0.9413144
```

```
cor.test(bit_model$count,bit_model$Price, method="spearman")
```

```
## Warning in cor.test.default(bit_model$count, bit_model$Price, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  bit_model$count and bit_model$Price
## S = 371351896, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
```

```
## sample estimates:
##       rho
## 0.9413144
```

How the Biggest Tweets affect the Weekly Price change of Bitcoin Prices

Adding Percentages to Daily Price

I calculate the weekly change in Bitcoin prices to show the impact of a tweet over a week.

```r
dailyPrice$pctchange<-(dailyPrice$day7-dailyPrice$day1)/dailyPrice$day1
```

Positive vs. Negative Percent Change in Bitcoin Price Means

```r
mean.posPCT<-mean(dailyPrice$pctchange[which(dailyPrice$pctchange>=0)])
mean.negPCT<-mean(dailyPrice$pctchange[which(dailyPrice$pctchange<=0)])
mean.posPCT
```

```
## [1] 0.104383
```

```r
mean.negPCT
```

```
## [1] -0.06749446
```

Top 3 Comments

```r
top3comments <- topComments[with(topComments,order(-comment_count)),]
top3comments <- top3comments[1:3,]
top3comments$percent<-1
```

```r
for(i in 1:3) {       # for-loop over columns
  for(j in 1:4145){
    comment_string<-substr(top3comments[i,3],1,10)
    price_string<-substr(dailyPrice[j,1],1,10)
    counted<-dailyPrice[j,9]
    if(all(comment_string==price_string)){
      top3comments[i,5] <- counted
    }
  }
}
```

We see that following the CashApp tweet there was a 5.584% increase in Bitcoin prices which 46% less than the mean increase of Bitcoins price in a week.

Following the aefBlockChain tweet Bitcoin's price decrease by 10.2% which was 1.84% less than the mean decrease in Bitcoin's price in a week.

Following the realdonaldtrump tweet Bitcoin's price decrease by 9.44% which was 9.53% less than the mean decrease in Bitcoin's price in a week.

```r
## CashApp
top3comments[1,5]
```

```
## [1] 0.05584376
```

```r
#percent more than mean
((abs(top3comments[1,5])-abs(mean.posPCT))/mean.posPCT)*100
```

```
## [1] -46.50108
```

```r
## aefBlockChain
top3comments[2,5]
```

```
## [1] -0.102458
```

```r
#percent less than mean
((abs(top3comments[2,5])-abs(mean.posPCT))/abs(mean.posPCT))*100
```

```
## [1] -1.844107
```

```r
## realdonaldtrump
top3comments[3,5]
```

```
## [1] -0.09443042
```

```r
#percent less than mean
((abs(top3comments[3,5])-abs(mean.posPCT))/abs(mean.posPCT))*100
```

```
## [1] -9.534647
```

Top Retweets

```r
top3retweets <- topRetweets[with(topRetweets,order(-retweet_count)),]
top3retweets <- top3retweets[1:3,]
top3retweets$percent<-1
```

```r
for(i in 1:3) {       # for-loop over columns
  for(j in 1:4145){
    retweet_string<-substr(top3retweets[i,3],1,10)
    price_string<-substr(dailyPrice[j,1],1,10)
    counted<-dailyPrice[j,9]
    if(all(retweet_string==price_string)){
      top3retweets[i,5] <- counted
    }
  }
}
```

We see that following the TravWeav tweet there was a 3.344% decrease in Bitcoin prices which 67.96% less than the mean decrease of Bitcoins price in a week.

Following the pulte tweet Bitcoin's price decrease by 8.65% which was 17.11% less than the mean decrease in Bitcoin's price in a week.

Following the pulte tweet Bitcoin's price increased by 1.669% which was 84% less than the mean increase in Bitcoin's price in a week.

```r
## TravWeav
top3retweets[1,5]
```

```
## [1] -0.03344154
```

```r
#percent less than mean
((abs(top3retweets[1,5])-abs(mean.posPCT))/abs(mean.posPCT))*100
```

```
## [1] -67.96264
```

```r
## pulte
top3retweets[2,5]
```

```
## [1] -0.0865173
```

```r
#percent less than mean
((abs(top3retweets[2,5])-abs(mean.posPCT))/abs(mean.posPCT))*100
```

```
## [1] -17.1155
```

```r
## pulte
top3retweets[3,5]
```

```
## [1] 0.01669586
```

```r
#percent more than mean
((abs(top3retweets[3,5])-abs(mean.posPCT))/mean.posPCT)*100
```

```
## [1] -84.00518
```

Top Likes

```r
top3likes <- topLikes[with(topLikes,order(-like_count)),]
top3likes <- top3likes[1:3,]
top3likes$percent<-1
```

```r
for(i in 1:3) {        # for-loop over columns
  for(j in 1:4145){
    like_string<-substr(top3likes[i,3],1,10)
    price_string<-substr(dailyPrice[j,1],1,10)
    counted<-dailyPrice[j,9]
    if(all(like_string==price_string)){
      top3likes[i,5] <- counted
    }
  }
}
```

We see that following the ItsArabMoney tweet there was a 8.269% increase in Bitcoin prices which 20.778% less than the mean increase of Bitcoins price in a week.

Following the netflix tweet Bitcoin's price decrease by 11.2% which was 7.6% more than the mean decrease in Bitcoin's price in a week.

Following the guwop tweet Bitcoin's price decrease by 4.37% which was 58.1% less than the mean decrease in Bitcoin's price in a week.

```r
## ItsArabMoney
top3likes[1,5]
```

```
## [1] 0.08269419
```

```r
#percent more than mean
((abs(top3likes[1,5])-abs(mean.posPCT))/mean.posPCT)*100
```

```
## [1] -20.77808
```

```r
## netflix
top3likes[2,5]
```

```
## [1] -0.112318
```

```r
#percent less than mean
((abs(top3likes[2,5])-abs(mean.posPCT))/abs(mean.posPCT))*100
```

```
## [1] 7.601898
```

```r
## guwop
top3likes[3,5]
```

```
## [1] -0.0437599
```

```r
#percent less than mean
((abs(top3likes[3,5])-abs(mean.posPCT))/abs(mean.posPCT))*100
```

```
## [1] -58.07754
```

Conclusion

After reviewing how the Bitcoin price corresponded to Twitter activity there is some certainty that there some relationship.

Using various forms of Linear Regression we see a considerable model produced for predicting the Price of Bitcoin given the monthly counts of Bitcoin mentioned in Tweets, and the correlation coefficients between these two values had a fairly high correlation. The Polynomial Degree 2 linear Regression we believe produced the best Model for predicting Bitcoin prices.

However, the impact of even the biggest tweets on Twitter had a negligible or uncertain effect on the price of Bitcoin. This was somewhat a positive point, suggesting that individuals are not likely to make financial decisions on whats popular.

Overall, it was fascinating to draw connections between Twitter and Bitcoin Prices through this analysis.