Open your mind. LUT.
Lappeenranta University of Technology

Financial Econometrics course Term paper

Fall 2018

Daily and hourly seasonality among cryptocurrencies

Henri ******

Joni *******

Table of Contents

Introduction

This paper is done for the Financial Econometrics course as a practical assignment. Because our background is B.Sc. in Industrial Engineering and Management, we have very limited knowledge in finance and economics. As this course was mostly about the methods and tools on how to make financial data-analysis, the choice of topic for this term paper was hard due to lack of basic knowledge from previous studies.

After going through several articles, we encountered the day-of-the-week anomaly, which can be assessed using the methods taught in this course. We decided to pick this anomaly as a topic and find practical data to use it on. Efficient market hypothesis says that asset prices fully reflect all available information (Kenneth, 1980). To make doing this term paper more interesting, we thought which markets could be inefficient in terms of day-of-the-week or hour-of-the-day anomalies. We ended up choosing low market cap and low volatility cryptocurrencies, as the data of these is easily available.

During doing the MATLAB code we thought why not take this one step further. We could not find many previous studies of hour-of-the-day effects, which makes sense because usually financial markets are open during limited timeframes. However, cryptocurrency markets are open 24/7. Using the same methodology as in day-of-the-week anomaly hourly differences can be modelled, the difference is that the number of regressors are increased to match every hour of the day. Our method to study hour-of-the-day anomaly is experimental because previous studies on the matter was not found. Due to lack of accurate data, hour-of-the-day analysis was only done to commonly known high frequency traded cryptocurrencies instead of low market cap ones.

The paper consists literature review of previous studies on day-of-the-week anomalies on several different markets and analysis how these studies have formed the regression equation. After the study method is described we look at the data which was available. At the end of this paper we discuss about the results and possible further studies on this matter.

Literature review

Day-of-the-week anomaly is very controversial topic and results in studies are varying depending on the market. Hla et. al (2015) states that many studies confirm negative Monday returns. Evidence to support this statement can be found on multiple markets, including US, UK, Canada, Australia, Japan, BRIC countries, Colombo, Malaysia, Singapore, Malaysia, Hong Kong and Thailand. Day-of-the-week effect is highly documented among the seasonal effects and has been investigated in different markets by various authors. (Hla et. al, 2015)

However, many of the findings are not universal. For example, negative Monday returns in Romania in contrast to positive Monday returns in Egypt. No day-of-the-week effect is found in Mauritius, but in Indian capital market positive Monday returns and negative Tuesday returns are found. The effects of the anomaly vary also depending on the chosen instrument. For instance, Hla et. al (2015) points out that no meaningful difference was found in mean daily returns for fixed income securities or in government securities, the so-called risk-free securities.

The impact of the chosen instrument has also impact on the theoretical and empirical framework of the study. Kalat (2007) assumes in the hypothesis in his study that day-of-the-week will be strongly present in examined cryptocurrency's returns. The reason for this is that cryptocurrencies' markets are highly volatile and there is strong suspicion for presence of noise in trading. Kalat also suggests that risk might be more profound with cryptocurrency of higher market capitalization. This is due to fact that higher market capitalization can attract more risk-averse investors. This factor might also affect the day-of-the-week and hour-of-the-day anomalies making the cryptocurrencies with high market capitalization more reflecting to anomalies.

Methodologies used in studies

Kalat (2007) describes weekday anomaly as a dependency of returns on a particular weekday. To test the anomaly, following linear equation was run:

$$r_t = a + B_1 D_{1t} + B_2 D_{2t} + B_3 D_{3t} + B_4 D_{4t} + B_5 D_{5t} + B_6 D_{6t} + \varepsilon_t$$

Independent variable is the return on cryptocurrency in specific time and dependent variables are dummy variables for weekdays from Monday to Saturday. One day of the week is omitted to avoid problem with perfect collinearity. If there is no day-of-the-week anomaly present, selected dummy variables will be estimated statistically insignificant. The same methodology was used also by Plastum & Caprole (2017).

Hla et. al (2015) studied the day-of-the-week anomaly in a very similar manner with dummy variables. The main difference to previous examples is that model is estimated without using a common intercept to represent the average return of the observed day. Additionally, the model assumes that the error terms and variances are constant across the time, the error terms are mean zero and regressors are unbiased. Alternative estimating methodologies used by Hla et. al (2015) in the same study are GARCH and ARCH models. These models were used to study whether the volatility of market returns was persistent.

There is close to zero previous studies related to hour-of-the-day anomaly, although for instance Caporale et al. (2014) have studied time-of-the-day anomaly in their study “Intraday Anomalies and Market Efficiency”. In the study, they investigated the first 45 minutes and the last 15 minutes of the trading sessions, as prices tend to up during those timeframes. One of the reasons for lacking studies of hour-of-the-day anomaly might be that traditional financial markets are open for limited time, but cryptocurrencies’ markets are open at all times. Additionally, blockchain technology and cryptocurrencies are relatively new phenomena, which might explain why there is not so many studies about the subject.

Conclusions of studies versus expectations

Hla et. al (2015) assumed that there should be evidence of day-of-the-week effect on returns of indices in Bursa Malaysia. Results of the study concludes that average returns of each day of the week are not equal. Monday returns are negative and Friday returns are significantly higher than other days. Kalat (2007) on the other came into conclusion that day-of-the-week anomaly was not presence for either cryptocurrencies he investigated in his study. He also had a hypothesis that higher market capitalization cryptocurrency might be more reflected to day-of-the-week anomaly. This could not be proven, since there was no evidence of anomaly in the first place.

Plastum & Caprole (2017) studied day-of-the-week effect in the cryptocurrency market using serveral testing methods, including regression analysis with dummy variables. They used a trading simulation approach to validate the results and found out that most cryptocurrencies do not exhibit day-of-the-week anomaly. The only exception was BitCoin, which had Monday returns significantly higher than those on the other days of the week.

Caporale et al. (2014) in their study about the time-of-the-day anomaly also rejected the hypothesis of the first 45 minutes and the last 15 minutes of the trading sessions being effective to market prices. This was concluded on both mature and less developed stock markets. The final conclusion was that it was not possible to exploit intraday patterns to make abnormal profit.

Empirical framework

The following equation is used to get logarithmic returns from the daily/hourly closing prices.

Logarithmic returns (Brooks, 2014. S.7-8):

$R_t = ln(P_t \ / \ P_{t-1})$

Where P_t = Closing price , P_{t-1} = previous closing price, R_t = logarithmic return

Based on literature review on the topic logarithmic returns were used on all the studies of day-of-the-week anomaly.

Different studies used slightly different regression equations for the day-of-the-week anomaly. The main difference on the studies is whether to include the intercept or not. On some of the studies the intercept term contained predefined values such as average weekly return or Mondays returns. If the intercept is included one of the regressor values must be removed from the equation due to dummy variable trap (Brooks, 2014, s.495). Dummy variable trap comes from multicollinearity issue in the dummy variable matrix.

When the intercept is not included, the coefficient estimates for each timeframe are the average values of the logarithmic returns for the timeframe. The results can be interpreted by checking significant values in the logarithmic returns. In the case where intercept is included, the dependent variables would represent the deviations of the dependent variables from their average values for the excluded timeframe. (Brooks, 2014, s.496) Since such “reference” category timeframe is not useful in this study, a model withouth intercept was chosen.

In this study we used primarily the method without constant to get regressor coefficients for each day or hour. In order to get R^2 values, the last regressor of the equation was removed and a constant was added. In this context the R^2 value is expected to be very low, since daily or hourly differences and constant are unlikely to model the returns effectively. Also, F-test can be used to test if the daily/hourly differences in returns provide any explanatory value to the model, and this cannot be done without including intercept in the model.

Day-of-the-week anomaly:

$R_t = \beta_1 Mon + \beta_2 Tue + \beta_3 Wed + \beta_4 Thurs + \beta_5 Fri + \beta_6 Sat + \beta_7 Sat + \varepsilon_i$

Where β_1 is dummy for regressor for Monday and so on. The dummy variable gets 1 if the returns day is the specific weekday and 0 if it is other day of the week. ε_i is the error term.

Hypotheses:

H_0: Day-of-the-week anomaly is NOT present in cryptocurrency markets.
H_1: Day-of-the-week anomaly is present in cryptocurrency markets.

Hour-of-the-day anomaly:

$R_t = \beta_1(0-1) + \beta_2(1-2) + \beta_3(2-3) + \dots + \beta_{23}(22-23) + \beta_{24}(23-24) + \varepsilon_i$

Where β_1 is the regressor for timeframe 00:00 to 01:00 UTC 0 (British standard time). Each hour timeframe has own regressor.

Hypotheses:

H_0: Hour-of-the-day anomaly is NOT present in cryptocurrency markets.
H_1: Hour-of-the-day anomaly is present in cryptocurrency markets.

Data

Historical price data of cryptocurrencies is available form many free sources. Because the chosen cryptocurrencies are traded on several brokers/markets the price data from the biggest average daily trading volume provider for each cryptocurrency is used. The data for daily closing prices (last hour of the day) is downloaded from CoinGecko (GoinGecko, 2018). And the hourly data is downloaded from website called cryptodatadownload (cryptodatadownload, 2018).

The cryptocurrencies for testing are chosen on three criteria:

1. Enough continuous historical price data
2. At least some trading volume on each day/hour
3. Traded on same timeframe as the other chosen cryptocurrencies for comparison

Legitimacy or quality of the projects behind the cryptocurrencies are not assessed in this study, as the goal is to only seek for hour-of-the-day or day-of-the-week anomalies. The cryptocurrencies to study are chosen randomly based on these three factors. Bitcoin was chosen as “baseline” as it is the most commonly traded and known cryptocurrency.

Daily price data

Bitcoin (BTC), Dogecoin (DOGE) and GridCoin (GRC) were chosen to be included in the study of day-of-the-week analyses. These cryptocurrencies were selected, because they had accurate data to support the study. The data samples of daily closing prices were checked to make sure that every day was included in the data samples and there were no duplicates. This was done with MATLAB script, which can be found in appendices. Duplicate or missing data rows would have led into inaccurate model and as a result regressor values would have been invalid. There is no direct USD to Dogecoin or USD to GridCoin trading pair on the market, so the historical price data for Dogecoin and GridCoin was calculated through trading pairs. USD-Bitcoin and Bitcoin-Dogecoin trading pairs were used to get daily returns for Dogecoin, and same was done to GridCoin.

Figure 1. contains logarithmic returns against time for each chosen cryptocurrency. The timeframe of this data is from 16-Mar-2015 to 24-Nov-2018 and includes daily returns for approximately 44 months. Figure shows that logarithmic returns seems to very highly volatile till the end of 2016 for BTC and DOGE, but GRC seems to be more settle in the beginning of the timeframe. From analytical point of view data seems to be roughly stationary, although this is not needed in terms of creating the model. Additionally, Jarque-Bera tests were done to each logarithmic return, and results concludes that any of the data series were not normally distributed at the 5% significance level.

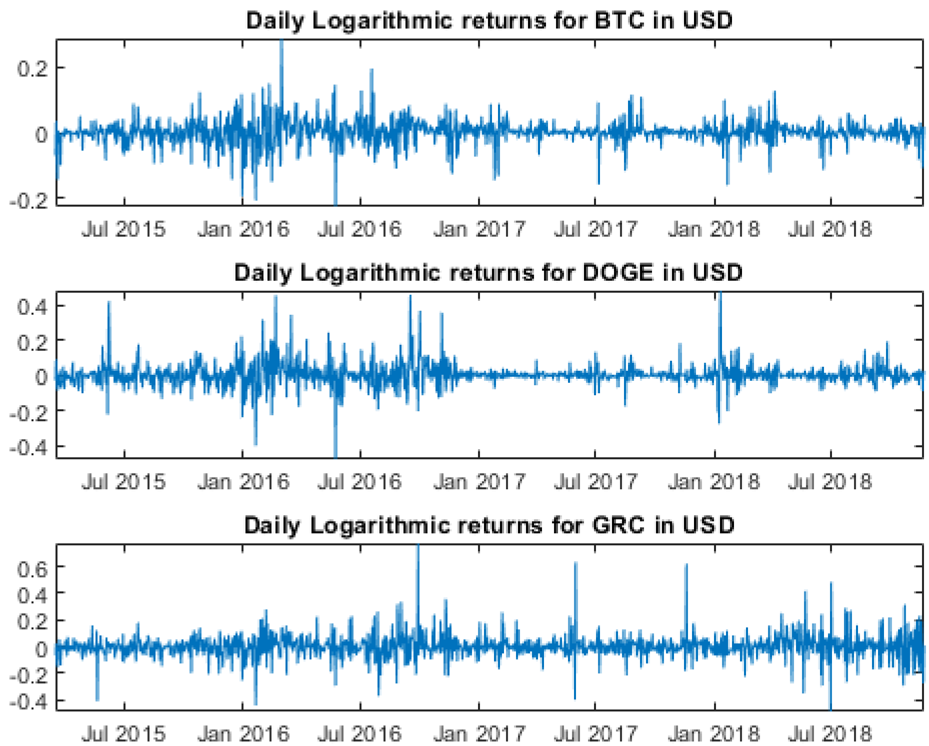


Figure 1. Daily logarithmic returns for selected cryptocurrencies

Descriptive statistics of the daily logarithmic returns are included in table 1**.** Mean, minimum and maximum values are calculated of each day for each cryptocurrency. In addition, mean, minimum, maximum and standard deviation values are presented for whole week for each cryptocurrency. Mean logarithmic values for each day for each cryptocurrency are also presented in bar chart, which are included in appendices.

According to statistics, Thursday is the only day of the week that gets positive mean return value for every cryptocurrency. For other days, mean logarithmic returns values varies depending on the chosen cryptocurrency. Largest and smallest daily logarithmic returns are drawn from GRC returns. Thursday for GRC is the lowest with mean value of -0.0105 and Monday is in the same time the highest with mean value of 0.0140.

In the 7 days statistics BTC and DOGE are really close to each other in terms of mean value, BTC having 7 days mean value of 0.0020 and DOGE having mean value of 0.0021. GRC mean value of 7 days is a bit lower, 0.0012. This is interesting, since the highest and the lowest value of daily logarithmic returns was from GRC data series. Additionally, on 7-day perspective maximum value goes to GRC with value 0.7731 with quite a wide margin. Maximum 7-day value for DOGE is 0.4802 and for BTC only 0.2871. In minimum values GRC got the lowest value with -0.4853, but DOGE being close with -0.4782. It is quite logical that the standard deviation values increase from BTC, to DOGE and to GRC, since latter cryptocurrencies also got higher maximum values and lower minimum values in mean comparison. These results suggest that smaller and more unknown cryptocurrencies are more volatile, which means that extreme daily price changes are more common. This same phenomenon is also noticeable in hour-of-the-day anomaly analysis for cryptocurrencies.

Table 1. Descriptive statistics of the daily logarithmic returns

		BTC	DOGE	GRC
Mean	Monday	0.0014	-0.0011	0.0140
	Tuesday	0.0026	-0.0013	-0.0105
	Wednesday	-0.0006	0.0037	0.0063
	Thursday	0.0025	0.0055	0.0073
	Friday	0.0039	-0.0028	0.0011
	Saturday	0.0011	0.0035	-0.0013
	Sunday	0.0031	0.0072	-0.0085
Mean	7 Days	0.0020	0.0021	0.0012
SD	7 Days	0.0385	0.0672	0.0920
Maximum	7 Days	0.2871	0.4802	0.7731
Minimum	7 days	-0.2252	-0.4782	-0.4853
Note: Total number of observations 1,349.				

Hourly price data

The historical price data for hourly prices had to be parsed and repaired, since the historical price data had missing hours and some multiple same data entries. MATLAB script was used to check if all the hours within the timeframe are included in the historical price data, and it can be found in appendices. Due to lack of accurate enough hourly data and the fact that the data repairing process took a long time only bigger market capitalization cryptocurrencies were used, since the small capitalization ones had more missing data entries. Due to chosen empirical method missing data or duplicate data for any hour would cause the regression equation to “skip” an hour and the regressor values provided by multivariate regression would be invalid. A timeframe from 1.11.2017 to 20.3.2018 was chosen for all selected cryptocurrencies. This timeframe consists the most recent “bubble” since the price of bitcoin goes from 6370 to a all time high of 19850 to 8500 USD.

The picked cryptocurrencies for hour-of-the-day analysis are Bitcoin, Ethereum and Dogecoin (BTC, ETH, DOGE). Since there is no direct USD to Dogecoin trading pair on the markets, the historical price data for Dogecoin was calculated through USD-Bitcoin and Bitcoin-Dogecoin trading pairs.

The figure 2 contains logarithmic return graphs for hourly data. The volatility peaks seem to occur at the same moments, so returns and volatility of each cryptocurrency should be correlated with each other. It is interesting to notice that ETH graph looks more similar to BTC graph, even though DOGE has to be traded through exchanging first USD to BTC and then to DOGE. Also, from graphical analysis the data seems stationary, though stationarity is not required in the regression equation.

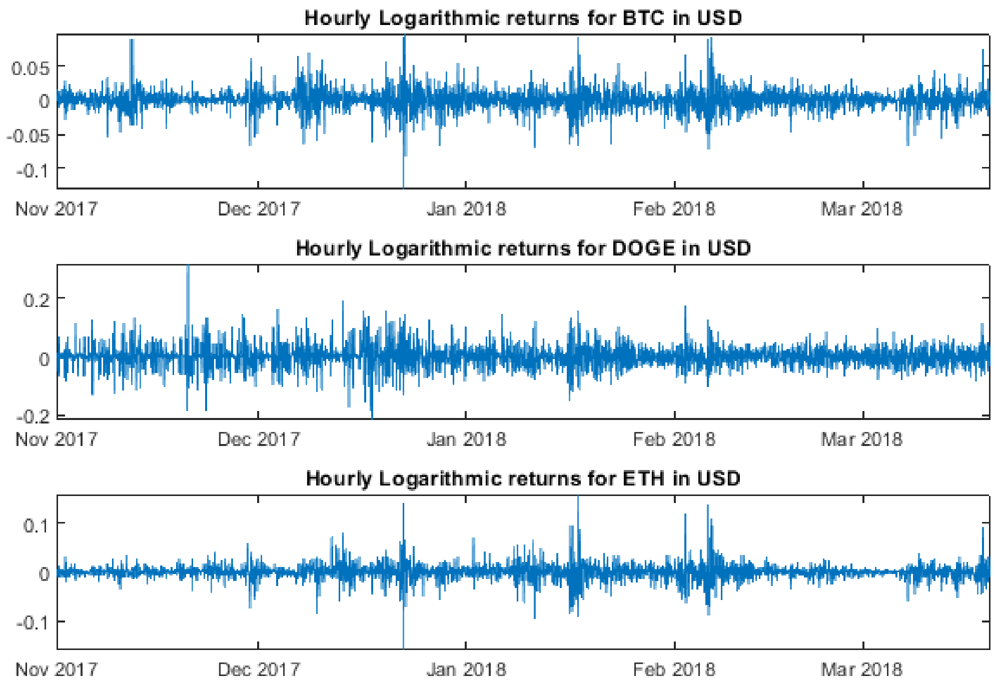


Figure 2, Hourly logarithmic returns for selected cryptocurrencies

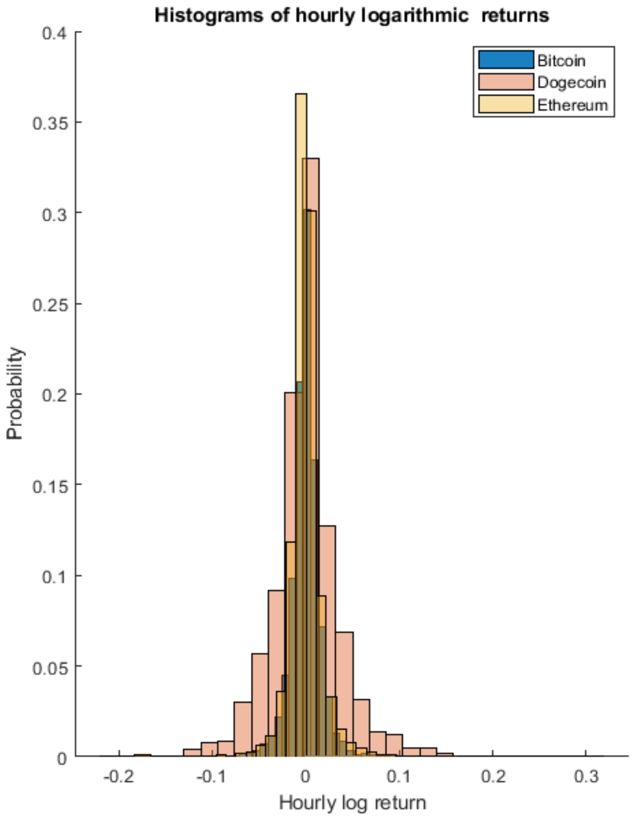


Figure 3, Histogram of hourly returns

By plotting histograms of each logarithmic returns volatility can be analysed (Figure 3). As expected, the histogram for dogecoin seems wider and probabilities for more extreme hourly price changes are higher. As the hourly returns of +-5% are not rare, a trader who could perform better than the market could make higher profits by day trading when compared to traditional financial markets. Since these markets hold no regulation at all, it could be seen as a playground for someone who develops trading algorithms for fun.

Jarque-Bera test indicates that none of the series are normally distributed.

Table 2 contains basic statistics of the selected three cryptocurrencies. As it can be seen from the histogram above, the maximum and standard deviation values are higher for DOGE, which indicates that it is riskier asset.

Table 2. Descriptive statistics of the hourly logarithmic returns

	BTC hourly log returns	DOGE hourly log returns	ETH hourly log returns
Mean	9.468e-05	3.022e-04	-1.759e-04
SD	0.0157	0.0373	0.0169
Max	0.0980	0.3178	0.1596

Results

In this chapter results of the day-of-the-week and the hour-of-the-day anomaly are represented. Results are compiled into table in the day-of-the-week anomaly test and into a visual figure in the results of hour-of-the-day anomaly test. In both cases results are analysed and implications are made out of the results.

Day-of-the-week results

Table 3. contains the results of the model estimated for dummy variables. Coefficients are represented, as well as the corresponding p-values in brackets. Asterisks are used to point significant coefficient values, ***, ** and * indicating significance at 1%, 5% and 10% levels, respectively. Additionally, R squared -values and results of the Chow test for the models are included in the table.

The first noticeable thing is that the estimated coefficient values for the cryptocurrencies are exactly the same as mean values for each day shown in table 1. This verifies that the model is at least modelling the right variables, but also concludes that this model should not be used for predicting returns. Mean values of over two years might not be the best way to predict the future return of the cryptocurrency tomorrow.

According to p-values it seems that there are not many significant coefficient values for weekdays at chosen significance levels. Only positive returns on Monday for GRC seems to be significant at 5% significance level. The remaining day returns are really mixed, and it is really hard to get universal conclusion out of them.

R squared values of estimation models are really low, which is logical since this model should not be used for prediction. This model is not good at explaining movements of logarithmic returns when we compare them to logarithmic returns from the data sample. This is not a problem, as point of this study was to find out if the weekdays or particular weekday got any explanatory power related to returns.

Chow test results indicates that coefficients are stable during the timeframe. In Chow test, data sample was split into half and coefficients for the both sides were compared. This means that coefficient values or corresponding p-values should not change during the timeframe. If a very short timeframe was chosen for Chow test, results might be different since logarithmic returns for cryptocurrencies seems to be highly volatile.

Table 3. Regression results

Day of the Week	BTC Coef. (Prob.)	DOGE Coef. (Prob.)	GRC Coef. (Prob.)
Monday	0.0014 (0.6056)	-0.0011 (0.8254)	0.0140 (0.03443)**
Tuesday	0.0026 (0.3516)	-0.0013 (0.7860)	-0.0105 (0.1118)
Wednesday	-0.0006 (0.8331)	0.0037 (0.4473)	0.0063 (0.3421)
Thursday	0.0025 (0.3619)	0.0055 (0.2553)	0.0073 (0.2672)
Friday	0.0039 (0.1635)	-0.0028 (0.5631)	0.0011 (0.8666)
Saturday	0.0011 (0.6937)	0.0035 (0.4757)	-0.0013 (0.8487)
Sunday	0.0031 (0.2584)	0.0072 (0.1357)	-0.0085 (0.1993)
R^2	0.0386	0.0672	0.0918
Chow test 0 = Coefficients are stable. 1 = Coefficients are not stable.	0	0	0
Note: ***, ** and * indicate significance at 1%, 5% and 10% levels, respectively.			

Hour-of-the-day results

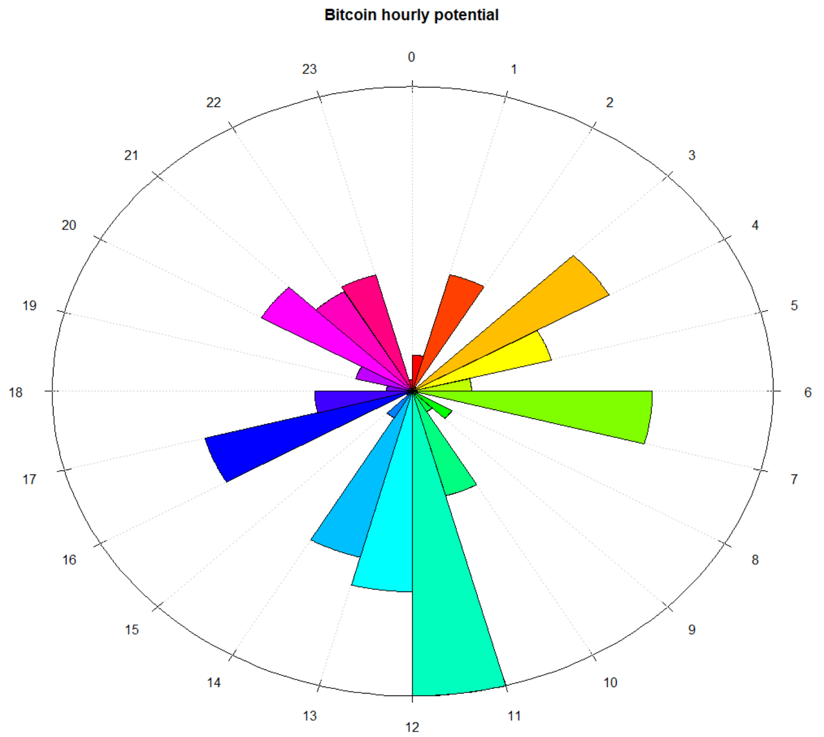
To examine the results of the hour-of-the-day anomaly, following equation is used to illustrate the data:

$$HourlyPotential = abs(HourlyCoefficient) * (1 - HourlyPvalue) * 1000$$

The sign of the hourly coefficient is not important, as trader could benefit from bearish of bullish hourly trends. For this reason, absolute values of the coefficients are taken. (1- HourlyPvalue) indicates that the smaller the Pvalue is the more likely the coefficient estimate is significant. The thousand multiplier is in the equation in order to illustrate the data. This illustration method is experimental and was done because reading coefficient estimates and p-values from a 24-row table for different variables is a slow process. As described in the empirical part of this study higher values of coefficient estimates should indicate higher average values of returns for each timeframe. As the absolute value of coefficient (average for the hour) is higher for certain timeframe, the p-value should indicate more significant values.

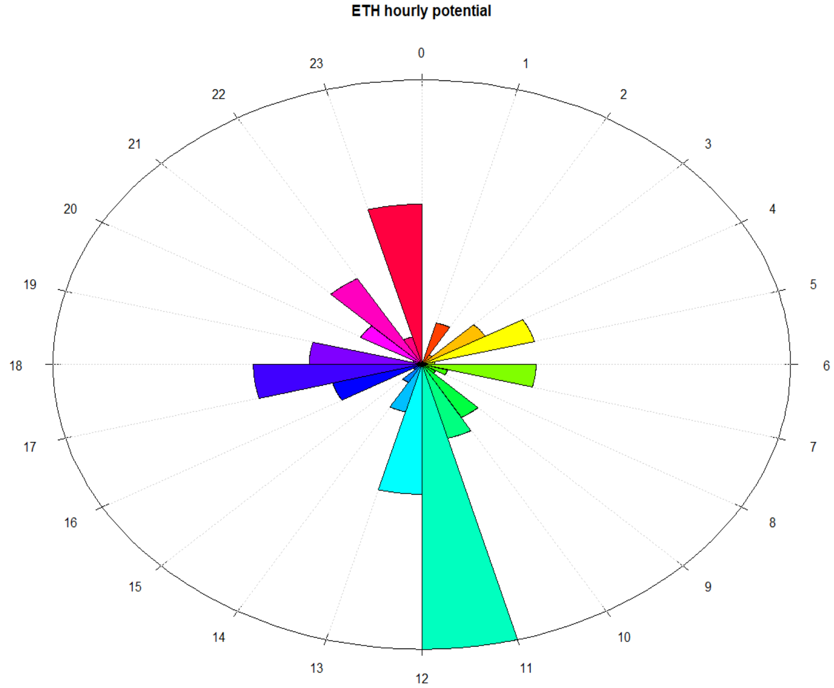
So, in a way this illustration over exaggerates the differences over each timeframe, but from a perspective of gaining profit from the anomaly, the profit over several iterations would be higher if price difference is higher and more probable for a certain timeframe.

The following figures 4, 5 & 6 are made in R using the script developed by Zoonekynd (Zoonekynd, 2018).



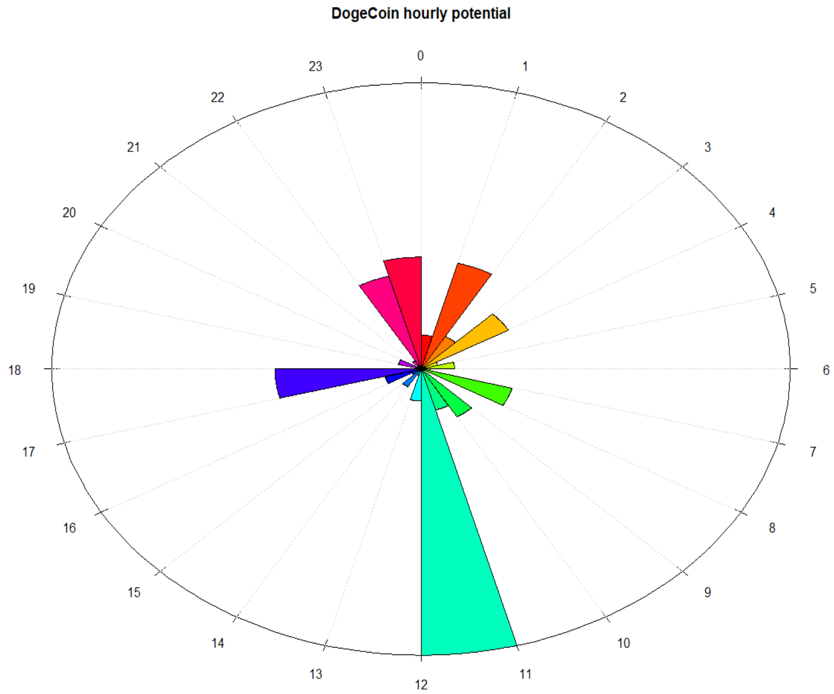
P-values BTC	
Hours 0-12	Hours 12-24
0.5867	**0.1474***
0.2925	0.191
0.8446	0.62
0.1591	0.8361
0.2911	0.1749
0.521	0.3972
0.1442*	0.6753
0.8517	0.5221
0.5758	0.2348
0.6651	0.3059
0.3246	0.2935
0.0485**	0.7636

Figure 4. Bitcoin hourly potential



P-values DOGE	
Hours 0-12	Hours 12-24
0.5585	0.5719
0.2605	0.8035
0.5374	0.6541
0.3568	0.7803
0.7177	0.5902
0.6117	0.2508
0.7971	0.985
0.3748	0.6727
0.9094	0.8627
0.448	0.7626
0.5112	0.2981
0.0257**	0.2543

Figure 5. Dogecoin Hourly potential



P-values ETH	
Hours 0-12	Hours 12-24
0.9533	0.2205
0.5137	0.4844
0.746	0.6574
0.4491	0.7917
0.3255	0.3895
0.7589	0.2194
0.3313	0.3349
0.6574	0.7944
0.7331	0.4572
0.4301	0.2972
0.3715	0.5989
0.0304**	0.1594

Figure 6. Ethereum hourly potential

There are very few significant values in the coefficients for each cryptocurrency. Bitcoin has 3 estimates which are significant with a significance level of 15% Dogecoin and Ethereum have only one. But the spike in the hourly potential of 11-12 is very interesting and it seems to happen on all of the three cryptocurrencies. The estimated coefficient estimates for this hour timeframe are 0.0026 for Bitcoin, 0.0071 for Dogecoin and 0.0031 for Ethereum. This means that during 11-12 timeframe the returns are expected to be likely positive.

There are several factors that could have gone wrong in this hour-of-the-day part of this study, and the low p-values are more likely due to issues in methodologies or data samples rather than actual anomaly. As the data repairing process took a long time only a time period of ~5 months was used which means only 139 closing prices for each hour of the day. In order to confirm this anomaly, the test methodology should be reviewed, and the timeframe should be larger. Chow test was done to check for parameter stability, and the test indicated failure to reject coefficient stability for each cryptocurrency.

A regression equation with constant and without 23-24 timeframe was also done, and the coefficient tables can be found in appendices. As expected, the R^2 values are very low, indicating that differences in hourly returns do not explain that much about the price trends. F-statistics p-values are high, meaning that that the overall the model contains almost none significant information. However, the overall accuracy of the overall model is not important in this study, as the focus was on finding hourly seasonality. This could have been done without using regression equation, but this way p-values for t-test for each timeframe was relatively easy to get. The regression equation could also be updated to include auto regressive and moving average lagged values, which could provide higher R^2 values, which could lead to probable market winning trading strategy. The anomaly found on timeframe 11-12 could be analysed with using software that can benchmark trading strategies. If a trading strategy could perform better than the market overall with the hour-of-the day anomaly information presented in this study, the anomaly could be confirmed.

Conclusion

The focus of this study was to analyse the presence of day-of-the-week and hour-of-the-day anomalies among cryptocurrencies. Literature review showed that results varied depending on chosen financial instrument and market and there was no universal truth according to researchers. Test for day-of-the-week concluded that there was no clear presence of day-of-the-week anomaly in the logarithmic returns. The average returns of each day of the week was more or less equal and the zero hypothesis “Day-of-the-week anomaly is present in cryptocurrency markets” is rejected.

The results of the hour-of-the-day anomaly test are more interesting. It seems that there is clear persistence of the hour-of-the-day anomaly and we accept the zero hypothesis: “Hour-of-the-day anomaly is present in cryptocurrency markets.”. During the time period from 11 AM to 12 PM it seems that returns are on average significantly higher than on other hours on all of the three cryptocurrencies. Data sample for the hour-of-the-day anomaly was significantly smaller, because data sample needed to be repaired and as a result only a time period of ~5 months was used in the model. To confirm these results, timeframe of the data sample should be larger, and methodologies used in this study should be reviewed.

P-values give probabilities and it is hard to define, how this information should be utilized to gain benefits on investment market. Another study should be done to benchmark value of this data by investing based on the findings of this study and then comparing the returns to the market average returns. Additionally, alternative methodologies, Such as Kruskal-Wallis and ANOVA test should be used to confirm test results. It would also be interesting to see if these test results vary during different time periods. This could be done based on various factors, like time of the year or market decline. This would give more valuable information, since these anomalies might be more present during smaller timeframes. Another interesting topic to study would be how the location of trader affects the return prices during different hours. The stock prices might change, when traders of specific continent are awake and trading.

During the work process of this term paper, we have learned quite a lot. The usage of dummy variables in regression equation was not thoroughly gone through during lectures. In the future we are able to use dummy variables as a part of regression equation, whereas in this study we included only dummy variables in the equation since we came to conclusion that more advanced models would be out of our skill level. Neither of us had done a larger study on statistical analysis before, and this was one of the few larger papers we have written in English.

References

Brooks, C. (2014). Introductory Econometrics for Finance. *Cambridge University Press*.

Brooks, C., & Persand, G. (2001). Seasonality in Southeast Asian stock markets: some new evidence on day-of-the-week effects. *Applied Economics Letters*, 8(3), p. 155-158.

Caporale, G. M., Gil-Alana, L., Plastun, A., & Makarenko, I. (2014). Intraday Anomalies and Market Efficiency: A Trading Robot Analysis. *DIW Berlin Discussion Paper No. 1377*.

Cryptodatadownload. (2018). Free Cryptocurrency Time Series Data. Referenced 23.11.2018 Available: http://www.cryptodatadownload.com/

Gajdošová, K., Heryán, T., & Tufan, E. (2011). Day of the Week Effect in the European Emerging Stock Markets: Recent Evidence from the Financial Crisis Period. *Scientific Papers of the University of Pardubice*.

GoinGecko. (2018). Homepage of GoinGecko. Available: https://www.coingecko.com

Hla, D., Gunathilaka, C., & Isa, A. (2015). Day-of-the-Week Effect: An Enduring Anomaly in Bursa Malaysia. *IUP Journal of Applied Finance*, 21(4), p. 38-50.

Kalát, P. (2017). Are Markets of Cryptocurrencies Efficient? Base of Ethereum, Bitcoin and Ripple. Diploma thesis. *Vysoká škola ekonomická v Praze*.

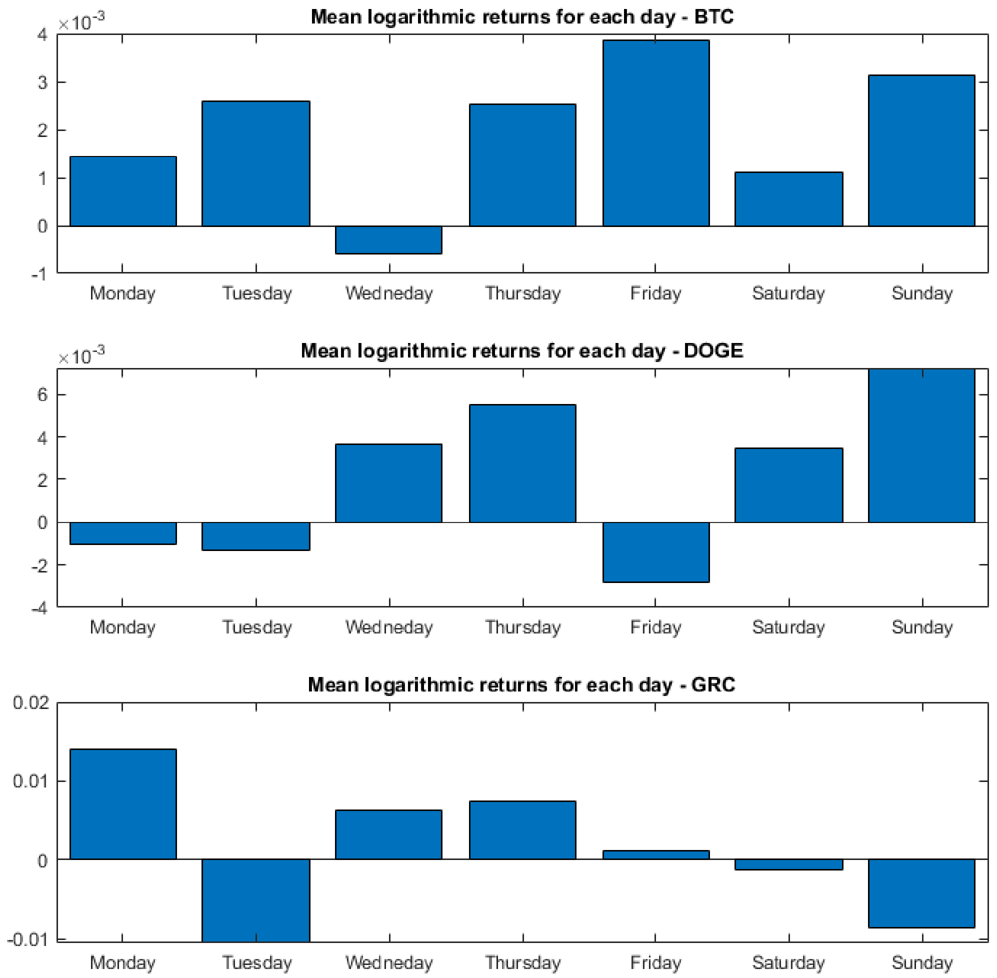
Kenneth, R. (1980). Stock returns and the weekend effect. *Journal of financial economics*, 8(1), p. 55-69.

Plastum, A., & Caprole, G. M. (2017). The Day of The Week Effect In The Crypto Currency Market. *Economics and Finance Working Paper Series.*

Zoonekynd, V. (2018). Clock plot, The R Graph Gallery. Available: https://www.r-graph-gallery.com/49-clock-plot/

Appendices

Empirical results for the day of the week effect



Matlab code

Hour-of-the-day

```
clc
clear all
```

importing the data

```
% importing BTC data
[numbers1,date1,useless1] = xlsread('Coinbase_BTCUSD_1h.csv');
dates_BTC = datetime(date1(2:end,1),'InputFormat','yyyy-MM-dd hh-a');
logReturnsBTC = numbers1(1:end,3);

% importing DOGEcoin data
[numbers2,date2,useless2] = xlsread('MuokattuDOGETOBTCANDUSD1h.csv');
dates_DOGE = datetime(date2(2:end-1,1),'InputFormat','yyyy-MM-dd hh-a');
logReturnsDOGE = numbers2(1:end-1,6);

% importing ETH data

[numbers3,date3,useless3] = xlsread('MuokattuETHUSD1h.csv');
dates_ETH = datetime(date3(2:end,1),'InputFormat','yyyy-MM-dd hh-a');
logReturnsETH = numbers3(1:end,7);
```

checking that the data has all the hours during that time period

```
d1BTC = datetime(dates_BTC(1),'InputFormat','yyyy-MM-dd hh-a');
d2BTC = datetime(dates_BTC(end),'InputFormat','yyyy-MM-dd hh-a');
periodBTC = d1BTC:hours(1):d2BTC;

d1DOGE = datetime(dates_DOGE(1),'InputFormat','yyyy-MM-dd hh-a');
d2DOGE = datetime(dates_DOGE(end),'InputFormat','yyyy-MM-dd hh-a');
periodDOGE = d1DOGE:hours(1):d2DOGE;

d1ETH = datetime(dates_ETH(1),'InputFormat','yyyy-MM-dd hh-a');
d2ETH = datetime(dates_ETH(end),'InputFormat','yyyy-MM-dd hh-a');
periodETH = d1ETH:hours(1):d2ETH;

if length(periodBTC) == length(dates_BTC) && length(periodBTC) == length(logReturnsBTC)
    fprintf('All hours/days are included in source data for the trading pair BTCUSD \n');
    fprintf('First datetime %s  \n',d1BTC);
    fprintf('Last datetime %s  \n',d2BTC);
end

if length(periodDOGE) == length(dates_DOGE)&& length(periodDOGE) == length(logReturnsDOGE)
    fprintf('All hours/days are included in source data for the trading pair DOGEUSD \n');
    fprintf('First datetime %s  \n',d1DOGE);
    fprintf('Last datetime %s  \n',d2DOGE);
end
```

```
if length(periodETH) == length(dates_ETH)&& length(periodETH) == length(logReturnsETH)
    fprintf('All hours/days are included in source data for the trading pair ETHUSD \n');
    fprintf('First datetime %s  \n',d1ETH);
    fprintf('Last datetime %s  \n',d2ETH);
end

% checking for nan values

if sum(isnan(logReturnsBTC)) == 0 && sum(isnan(logReturnsDOGE))== 0 &&
sum(isnan(logReturnsETH)) == 0
    fprintf('no NaN values')
end
```

```
All hours/days are included in source data for the trading pair BTCUSD
First datetime 01-Nov-2017 00:00:00
Last datetime 20-Mar-2018 00:00:00
All hours/days are included in source data for the trading pair DOGEUSD
First datetime 01-Nov-2017 00:00:00
Last datetime 20-Mar-2018 00:00:00
All hours/days are included in source data for the trading pair ETHUSD
First datetime 01-Nov-2017 00:00:00
Last datetime 20-Mar-2018 00:00:00
no NaN values
```

normality test for the logaritmic returns

```
[h,p,jbstat,critval] = jbtest(logReturnsBTC,0.05) %% not normally distributed
[h,p,jbstat,critval] = jbtest(logReturnsDOGE,0.05)
[h,p,jbstat,critval] = jbtest(logReturnsETH,0.05)
```

```
Warning: P is less than the smallest tabulated value, returning 0.001.
h =  1
p = 1.0000e-03
jbstat = 5.2430e+03
critval = 5.9753

Warning: P is less than the smallest tabulated value, returning 0.001.

h = 1
p = 1.0000e-03
jbstat = 2.9677e+03
critval = 5.9753

Warning: P is less than the smallest tabulated value, returning 0.001.
h = 1
p = 1.0000e-03
jbstat = 2.2097e+04
critval =  5.9753
```

making a matrix of the dummy variables for every hour

```
matrix = zeros(length(logReturnsBTC),24); matrix2 = zeros(length(logReturnsDOGE),24); matrix3
= zeros(length(logReturnsETH),24);


for i = 1:24:length(logReturnsBTC)
    for j = 0:1:23
        if i+j <= length(logReturnsBTC)
            matrix(i+j,j+1) = 1;
        end
    end
end

for i = 1:24:length(logReturnsDOGE)
    for j = 0:1:23
        if i+j <= length(logReturnsDOGE)
            matrix2(i+j,j+1) = 1;
        end
    end
end

for i = 1:24:length(logReturnsETH)
    for j = 0:1:23
        if i+j <= length(logReturnsETH)
            matrix3(i+j,j+1) = 1;
        end
    end
end

parameterNames =
{'DailyLogReturns','From_0_to_1','From_1_to_2','From_2_to_3','From_3_to_4','From_4_to_5','Fro
m_5_to_6','From_6_to_7','From_7_to_8','From_8_to_9','From_9_to_10','From_10_to_11','From_11_t
o_12','From_12_to_13','From_13_to_14','From_14_to_15','From_15_to_16','From_16_to_17','From_1
7_to_18','From_18_to_19','From_19_to_20','From_20_to_21','From_21_to_22','From_22_to_23','Fro
m_23_to_24'};
tbl1 =
table(logReturnsBTC,matrix(:,1),matrix(:,2),matrix(:,3),matrix(:,4),matrix(:,5),matrix(:,6),m
atrix(:,7),matrix(:,8),matrix(:,9),matrix(:,10),matrix(:,11),matrix(:,12),matrix(:,13),matrix
(:,14),matrix(:,15),matrix(:,16),matrix(:,17),matrix(:,18),matrix(:,19),matrix(:,20),matrix(:
,21),matrix(:,22),matrix(:,23),matrix(:,24),'VariableNames',parameterNames);

tbl2 =
table(logReturnsDOGE,matrix2(:,1),matrix2(:,2),matrix2(:,3),matrix2(:,4),matrix2(:,5),matrix2
(:,6),matrix2(:,7),matrix2(:,8),matrix2(:,9),matrix2(:,10),matrix2(:,11),matrix2(:,12),matrix
2(:,13),matrix2(:,14),matrix2(:,15),matrix2(:,16),matrix2(:,17),matrix2(:,18),matrix2(:,19),m
atrix2(:,20),matrix2(:,21),matrix2(:,22),matrix2(:,23),matrix2(:,24),'VariableNames',paramete
rNames);

tbl3 =
table(logReturnsETH,matrix3(:,1),matrix3(:,2),matrix3(:,3),matrix3(:,4),matrix3(:,5),matrix3(
:,6),matrix3(:,7),matrix3(:,8),matrix3(:,9),matrix3(:,10),matrix3(:,11),matrix3(:,12),matrix3
(:,13),matrix3(:,14),matrix3(:,15),matrix3(:,16),matrix3(:,17),matrix3(:,18),matrix3(:,19),ma
trix3(:,20),matrix3(:,21),matrix3(:,22),matrix3(:,23),matrix3(:,24),'VariableNames',parameter
Names);
```

making the models WITHOUT CONSTANT

```
Model_BTC_hourly = fitlm(tbl1,'DailyLogReturns ~ From_0_to_1 + From_1_to_2 + 
From_2_to_3+From_3_to_4+From_4_to_5+From_5_to_6+From_6_to_7+From_7_to_8+From_8_to_9+From_9_to
_10+From_10_to_11+From_11_to_12+From_12_to_13+From_13_to_14+From_14_to_15+From_15_to_16+From_
16_to_17+From_17_to_18+From_18_to_19+From_19_to_20+From_20_to_21+From_21_to_22+From_22_to_23+
From_23_to_24 -1')
Model_DOGE_hourly = fitlm(tbl2,'DailyLogReturns ~ From_0_to_1 + From_1_to_2 + 
From_2_to_3+From_3_to_4+From_4_to_5+From_5_to_6+From_6_to_7+From_7_to_8+From_8_to_9+From_9_to
_10+From_10_to_11+From_11_to_12+From_12_to_13+From_13_to_14+From_14_to_15+From_15_to_16+From_
16_to_17+From_17_to_18+From_18_to_19+From_19_to_20+From_20_to_21+From_21_to_22+From_22_to_23+
From_23_to_24 -1')
Model_ETH_hourly = fitlm(tbl3,'DailyLogReturns ~ From_0_to_1 + From_1_to_2 + 
From_2_to_3+From_3_to_4+From_4_to_5+From_5_to_6+From_6_to_7+From_7_to_8+From_8_to_9+From_9_to
_10+From_10_to_11+From_11_to_12+From_12_to_13+From_13_to_14+From_14_to_15+From_15_to_16+From_
16_to_17+From_17_to_18+From_18_to_19+From_19_to_20+From_20_to_21+From_21_to_22+From_22_to_23+
From_23_to_24 -1')
```

```
Model_BTC_hourly = 


Linear regression model:
    DailyLogReturns ~ [Linear formula with 24 terms in 24 predictors]

Estimated Coefficients:
                      Estimate         SE         tStat       pValue  
                    ___________    _________    ________    ________

    From_0_to_1     -0.00072245     0.001329    -0.54362     0.58674
    From_1_to_2      -0.0014043    0.0013337     -1.0529     0.29245
    From_2_to_3     -0.00026147    0.0013337    -0.19604     0.84459
    From_3_to_4      -0.0018786    0.0013337     -1.4085     0.15908
    From_4_to_5       0.0014084    0.0013337       1.056     0.29106
    From_5_to_6     -0.00085606    0.0013337    -0.64185     0.52102
    From_6_to_7        0.001948    0.0013337      1.4605     0.14424
    From_7_to_8      0.00024929    0.0013337     0.18691     0.85174
    From_8_to_9     -0.00074626    0.0013337    -0.55952     0.57584
    From_9_to_10    -0.00057736    0.0013337    -0.43289     0.66512
    From_10_to_11    -0.0013139    0.0013337    -0.98512     0.32464
    From_11_to_12     0.0026325    0.0013337      1.9738     0.04849
    From_12_to_13     0.0019327    0.0013337      1.4491      0.1474
    From_13_to_14    -0.0017445    0.0013337      -1.308     0.19097
    From_14_to_15    0.00066143    0.0013337     0.49592     0.61998
    From_15_to_16   -0.00027587    0.0013337    -0.20684     0.83615
    From_16_to_17    -0.0018098    0.0013337     -1.3569     0.17489
    From_17_to_18    -0.0011294    0.0013337    -0.84677     0.39718
    From_18_to_19    0.00055868    0.0013337     0.41888     0.67533
    From_19_to_20    0.00085384    0.0013337     0.64018      0.5221
    From_20_to_21     0.0015849    0.0013337      1.1883     0.23479
    From_21_to_22     0.0013658    0.0013337       1.024     0.30589
    From_22_to_23     0.0014014    0.0013337      1.0507     0.29346
    From_23_to_24    0.00040118    0.0013337     0.30079     0.76359


Number of observations: 3337, Error degrees of freedom: 3313
Root Mean Squared Error: 0.0157

Model_DOGE_hourly = 
```

```
Linear regression model:
    DailyLogReturns ~ [Linear formula with 24 terms in 24 predictors]

Estimated Coefficients:
                      Estimate         SE          tStat        pValue
                     ___________    _________    _________    ________

    From_0_to_1        -0.001846    0.0031551     -0.58509     0.55853
    From_1_to_2       -0.0035633    0.0031664      -1.1253     0.26052
    From_2_to_3       -0.0019532    0.0031664     -0.61685     0.53738
    From_3_to_4         0.002918    0.0031664      0.92156     0.35683
    From_4_to_5        0.0011449    0.0031664      0.36158     0.71769
    From_5_to_6        0.0016074    0.0031664      0.50764     0.61174
    From_6_to_7      -0.00081403    0.0031664     -0.25708     0.79713
    From_7_to_8        0.0028108    0.0031664      0.88769     0.37477
    From_8_to_9       0.00036021    0.0031664      0.11376     0.90944
    From_9_to_10      -0.0024028    0.0031664     -0.75882     0.44801
    From_10_to_11     -0.0020804    0.0031664     -0.65701     0.51122
    From_11_to_12      0.0070643    0.0031664        2.231    0.025748
    From_12_to_13      0.0017899    0.0031664      0.56526     0.57193
    From_13_to_14    -0.00078791    0.0031664     -0.24883     0.80351
    From_14_to_15      0.0014191    0.0031664      0.44817     0.65406
    From_15_to_16    -0.00088342    0.0031664       -0.279     0.78027
    From_16_to_17     -0.0017055    0.0031664     -0.53861     0.59019
    From_17_to_18      0.0036368    0.0031664       1.1485     0.25083
    From_18_to_19    -5.9503e-05    0.0031664    -0.018792     0.98501
    From_19_to_20      0.0013379    0.0031664      0.42251     0.67268
    From_20_to_21     0.00054767    0.0031664      0.17296     0.86269
    From_21_to_22    -0.00095673    0.0031664     -0.30215     0.76256
    From_22_to_23      0.0032953    0.0031664       1.0407     0.29809
    From_23_to_24     -0.0036106    0.0031664      -1.1403     0.25426


Number of observations: 3337, Error degrees of freedom: 3313
Root Mean Squared Error: 0.0373

Model_ETH_hourly = 


Linear regression model:
    DailyLogReturns ~ [Linear formula with 24 terms in 24 predictors]

Estimated Coefficients:
                      Estimate         SE          tStat        pValue
                     ___________    _________    _________    ________

    From_0_to_1      -8.3906e-05     0.001432    -0.058596     0.95328
    From_1_to_2      -0.00093854    0.0014371     -0.65308     0.51375
    From_2_to_3       0.00046558    0.0014371      0.32397     0.74598
    From_3_to_4       -0.0010878    0.0014371     -0.75695     0.44913
    From_4_to_5        0.0014131    0.0014371      0.98327     0.32555
    From_5_to_6       0.00044119    0.0014371        0.307     0.75886
    From_6_to_7        0.0013964    0.0014371      0.97171     0.33126
    From_7_to_8      -0.00063733    0.0014371     -0.44348     0.65745
    From_8_to_9      -0.00049018    0.0014371     -0.34109     0.73305
    From_9_to_10       -0.001134    0.0014371      -0.7891     0.43011
```

```
    From_10_to_11      -0.0012844    0.0014371     -0.89372      0.37154
    From_11_to_12       0.0031117    0.0014371       2.1653     0.030439
    From_12_to_13       0.0017612    0.0014371       1.2255      0.22046
    From_13_to_14      -0.0010049    0.0014371     -0.69927      0.48443
    From_14_to_15      0.00063745    0.0014371      0.44357      0.65738
    From_15_to_16      0.00037966    0.0014371      0.26418      0.79165
    From_16_to_17      -0.0012367    0.0014371     -0.86056      0.38954
    From_17_to_18      -0.0017654    0.0014371      -1.2285      0.21936
    From_18_to_19       -0.001386    0.0014371     -0.96441      0.33491
    From_19_to_20      0.00037458    0.0014371      0.26065      0.79438
    From_20_to_21       0.0010685    0.0014371      0.74351      0.45722
    From_21_to_22       0.0014984    0.0014371       1.0427      0.29718
    From_22_to_23      0.00075587    0.0014371      0.52597      0.59894
    From_23_to_24       0.0020227    0.0014371       1.4075      0.15937


Number of observations: 3337, Error degrees of freedom: 3313
Root Mean Squared Error: 0.0169
```

making the models WITH CONSTANT

```
Model_BTC_hourly2 = fitlm(tbl1,'DailyLogReturns ~ From_0_to_1 + From_1_to_2 +
From_2_to_3+From_3_to_4+From_4_to_5+From_5_to_6+From_6_to_7+From_7_to_8+From_8_to_9+From_9_to
_10+From_10_to_11+From_11_to_12+From_12_to_13+From_13_to_14+From_14_to_15+From_15_to_16+From_
16_to_17+From_17_to_18+From_18_to_19+From_19_to_20+From_20_to_21+From_21_to_22+From_22_to_23'
)
Model_DOGE_hourly2 = fitlm(tbl2,'DailyLogReturns ~ From_0_to_1 + From_1_to_2 +
From_2_to_3+From_3_to_4+From_4_to_5+From_5_to_6+From_6_to_7+From_7_to_8+From_8_to_9+From_9_to
_10+From_10_to_11+From_11_to_12+From_12_to_13+From_13_to_14+From_14_to_15+From_15_to_16+From_
16_to_17+From_17_to_18+From_18_to_19+From_19_to_20+From_20_to_21+From_21_to_22+From_22_to_23'
)
Model_ETH_hourly2 = fitlm(tbl3,'DailyLogReturns ~ From_0_to_1 + From_1_to_2 +
From_2_to_3+From_3_to_4+From_4_to_5+From_5_to_6+From_6_to_7+From_7_to_8+From_8_to_9+From_9_to
_10+From_10_to_11+From_11_to_12+From_12_to_13+From_13_to_14+From_14_to_15+From_15_to_16+From_
16_to_17+From_17_to_18+From_18_to_19+From_19_to_20+From_20_to_21+From_21_to_22+From_22_to_23'
)
```

```
Model_BTC_hourly2 = 


Linear regression model:
    DailyLogReturns ~ [Linear formula with 24 terms in 23 predictors]

Estimated Coefficients:
                      Estimate         SE          tStat       pValue  
                     ___________    _________    _________    ________

    (Intercept)       0.00040118    0.0013337      0.30079     0.76359
    From_0_to_1       -0.0011236    0.0018828     -0.59678      0.5507
    From_1_to_2       -0.0018055    0.0018862     -0.95723     0.33852
    From_2_to_3      -0.00066265    0.0018862     -0.35132     0.72537
    From_3_to_4       -0.0022798    0.0018862      -1.2087     0.22688
    From_4_to_5        0.0010072    0.0018862      0.53399     0.59338
    From_5_to_6       -0.0012572    0.0018862     -0.66655     0.50511
    From_6_to_7        0.0015468    0.0018862      0.82005     0.41225
    From_7_to_8      -0.00015189    0.0018862    -0.080526     0.93582
```

```
    From_8_to_9        -0.0011474     0.0018862      -0.60834     0.54301
    From_9_to_10      -0.00097854     0.0018862      -0.51879     0.60394
    From_10_to_11      -0.0017151     0.0018862      -0.90928     0.36327
    From_11_to_12       0.0022313     0.0018862         1.183      0.2369
    From_12_to_13       0.0015316     0.0018862       0.81198     0.41686
    From_13_to_14      -0.0021457     0.0018862       -1.1376     0.25538
    From_14_to_15      0.00026025     0.0018862       0.13797     0.89027
    From_15_to_16     -0.00067705     0.0018862      -0.35895     0.71965
    From_16_to_17       -0.002211     0.0018862       -1.1722      0.2412
    From_17_to_18      -0.0015306     0.0018862      -0.81145     0.41716
    From_18_to_19      0.00015749     0.0018862      0.083499     0.93346
    From_19_to_20      0.00045266     0.0018862       0.23998     0.81036
    From_20_to_21       0.0011837     0.0018862       0.62758     0.53032
    From_21_to_22      0.00096462     0.0018862       0.51141      0.6091
    From_22_to_23       0.0010002     0.0018862       0.53029     0.59595


Number of observations: 3337, Error degrees of freedom: 3313
Root Mean Squared Error: 0.0157
R-squared: 0.00697,  Adjusted R-Squared 7.96e-05
F-statistic vs. constant model: 1.01, p-value = 0.446

Model_DOGE_hourly2 = 


Linear regression model:
    DailyLogReturns ~ [Linear formula with 24 terms in 23 predictors]

Estimated Coefficients:
                      Estimate        SE         tStat       pValue  
                     __________    _________    ________    ________

    (Intercept)      -0.0036106    0.0031664     -1.1403     0.25426
    From_0_to_1       0.0017646      0.00447     0.39476     0.69305
    From_1_to_2      4.7261e-05     0.004478    0.010554     0.99158
    From_2_to_3       0.0016574     0.004478     0.37011     0.71132
    From_3_to_4       0.0065286     0.004478      1.4579     0.14495
    From_4_to_5       0.0047555     0.004478       1.062     0.28833
    From_5_to_6        0.005218     0.004478      1.1653       0.244
    From_6_to_7       0.0027966     0.004478     0.62451     0.53234
    From_7_to_8       0.0064214     0.004478       1.434     0.15167
    From_8_to_9       0.0039708     0.004478     0.88673     0.37529
    From_9_to_10      0.0012078     0.004478     0.26972     0.78739
    From_10_to_11     0.0015302     0.004478     0.34172     0.73258
    From_11_to_12      0.010675     0.004478      2.3839    0.017189
    From_12_to_13     0.0054004     0.004478       1.206     0.22791
    From_13_to_14     0.0028227     0.004478     0.63034     0.52851
    From_14_to_15     0.0050297     0.004478      1.1232     0.26144
    From_15_to_16     0.0027272     0.004478     0.60901     0.54256
    From_16_to_17     0.0019051     0.004478     0.42544     0.67055
    From_17_to_18     0.0072474     0.004478      1.6184     0.10566
    From_18_to_19     0.0035511     0.004478     0.79301     0.42783
    From_19_to_20     0.0049485     0.004478      1.1051     0.26922
    From_20_to_21     0.0041583     0.004478      0.9286     0.35317
    From_21_to_22     0.0026539     0.004478     0.59264     0.55346
    From_22_to_23     0.0069059     0.004478      1.5422     0.12313
```

```
Number of observations: 3337, Error degrees of freedom: 3313
Root Mean Squared Error: 0.0373
R-squared: 0.00445,  Adjusted R-Squared -0.00246
F-statistic vs. constant model: 0.644, p-value = 0.901

Model_ETH_hourly2 =


Linear regression model:
    DailyLogReturns ~ [Linear formula with 24 terms in 23 predictors]

Estimated Coefficients:
                      Estimate        SE         tStat       pValue
                     ___________    _________    ________    ________

    (Intercept)        0.0020227    0.0014371      1.4075     0.15937
    From_0_to_1       -0.0021066    0.0020287     -1.0384     0.29916
    From_1_to_2       -0.0029613    0.0020324     -1.4571     0.14519
    From_2_to_3       -0.0015571    0.0020324    -0.76618     0.44363
    From_3_to_4       -0.0031105    0.0020324     -1.5305     0.12599
    From_4_to_5      -0.00060967    0.0020324    -0.29998     0.76421
    From_5_to_6       -0.0015815    0.0020324    -0.77817     0.43652
    From_6_to_7      -0.00062628    0.0020324    -0.30816     0.75798
    From_7_to_8       -0.0026601    0.0020324     -1.3088     0.19068
    From_8_to_9       -0.0025129    0.0020324     -1.2364     0.21638
    From_9_to_10      -0.0031567    0.0020324     -1.5532     0.12046
    From_10_to_11     -0.0033071    0.0020324     -1.6272     0.10379
    From_11_to_12       0.001089    0.0020324     0.53581     0.59213
    From_12_to_13     -0.0002615    0.0020324    -0.12867     0.89763
    From_13_to_14     -0.0030276    0.0020324     -1.4897     0.13639
    From_14_to_15     -0.0013853    0.0020324    -0.68161     0.49553
    From_15_to_16     -0.0016431    0.0020324    -0.80845     0.41889
    From_16_to_17     -0.0032594    0.0020324     -1.6038     0.10886
    From_17_to_18     -0.0037881    0.0020324     -1.8639    0.062422
    From_18_to_19     -0.0034087    0.0020324     -1.6772    0.093597
    From_19_to_20     -0.0016481    0.0020324    -0.81095     0.41745
    From_20_to_21    -0.00095423    0.0020324    -0.46952     0.63873
    From_21_to_22    -0.00052432    0.0020324    -0.25799     0.79643
    From_22_to_23     -0.0012669    0.0020324    -0.62334      0.5331


Number of observations: 3337, Error degrees of freedom: 3313
Root Mean Squared Error: 0.0169
R-squared: 0.00557,  Adjusted R-Squared -0.00133
F-statistic vs. constant model: 0.807, p-value = 0.725
```

Parameter stability testing

```
[h,pValue,stat,cValue] =
chowtest(matrix,logReturnsBTC,[1500],'Intercept',false,'Display','summary');
[h,pValue,stat,cValue] =
chowtest(matrix,logReturnsDOGE,[1500],'Intercept',false,'Display','summary');
[h,pValue,stat,cValue] =
chowtest(matrix,logReturnsETH,[1500],'Intercept',false,'Display','summary');
```

```
RESULTS SUMMARY
```

```
***************
Test 1

Sample size: 3337
Breakpoint: 1500

Test type: breakpoint
Coefficients tested: All

Statistic: 0.7654
Critical value: 1.5206

P value: 0.7842
Significance level: 0.0500

Decision: Fail to reject coefficient stability


RESULTS SUMMARY

***************
Test 1

Sample size: 3337
Breakpoint: 1500

Test type: breakpoint
Coefficients tested: All

Statistic: 1.4200
Critical value: 1.5206

P value: 0.0843
Significance level: 0.0500

Decision: Fail to reject coefficient stability


RESULTS SUMMARY

***************
Test 1

Sample size: 3337
Breakpoint: 1500

Test type: breakpoint
Coefficients tested: All

Statistic: 0.8356
Critical value: 1.5206

P value: 0.6933
Significance level: 0.0500

Decision: Fail to reject coefficient stability
```

f-test

illustrating all of the 24h models in one pic (Creating the data for R)

```
figure
hold on
plot(1:1:24,Model_BTC_hourly.Coefficients(:,1).Estimate)
plot(1:1:24,Model_DOGE_hourly.Coefficients(:,1).Estimate)
plot(1:1:24,Model_ETH_hourly.Coefficients(:,1).Estimate)

toPlotBTC = abs(Model_BTC_hourly.Coefficients.Estimate).*(1-
Model_BTC_hourly.Coefficients.pValue).*10000;

toPlotDOGE = abs(Model_DOGE_hourly.Coefficients.Estimate).*(1-
Model_DOGE_hourly.Coefficients.pValue).*10000;

toPlotETH = abs(Model_ETH_hourly.Coefficients.Estimate).*(1-
Model_ETH_hourly.Coefficients.pValue).*10000;

% EOF
```

Day-of-the-week

importing the data

```
[numbers_BTC,date1,~]=xlsread('BTC_USD_DAILY.csv');
[numbers_DOGE,date2,~]=xlsread('DOGE_USD_DAILY.csv');
[numbers_GRC,date3,~]=xlsread('GRC_USD_DAILY.csv');

logChanges_BTC = numbers_BTC([2:end-1],4); % 2 because first parameter = Monday.
logChanges_DOGE = numbers_DOGE([2:end-1],4);
logChanges_GRC = numbers_GRC([2:end-1],4);
```

Checking that the data has all the days during that time period

BTC

```
dates_BTC = datetime(date1(2:end,1),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC');

d2_BTC = datetime(dates_BTC(1),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC');
d1_BTC = datetime(dates_BTC(end),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC')+1; %
moving to Monday.
NumDays_BTC = daysact(d1_BTC,d2_BTC);

% comparing the number of days to number of columns in returns.
if (size(logChanges_BTC,1) == NumDays_BTC)
    fprintf('All days are included in source data for BTC. \n');
    fprintf('First datetime %s.\n',d1_BTC);
    fprintf('Last datetime %s.\n\n',d2_BTC);
end
```

```
% DOGE
dates_DOGE = datetime(date2(2:end,1),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC');

d2_DOGE = datetime(dates_DOGE(1),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC');
d1_DOGE = datetime(dates_DOGE(end),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC')+1;
NumDays_DOGE = daysact(d1_DOGE,d2_DOGE);

% comparing the number of days to number of columns in returns.
if (size(logChanges_DOGE,1) == NumDays_DOGE)
    fprintf('All days are included in source data for DOGE. \n');
    fprintf('First datetime %s.\n',d1_DOGE);
    fprintf('Last datetime %s.\n\n',d2_DOGE);
end

% GRC

dates_GRC = datetime(date2(2:end,1),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC');

d2_GRC = datetime(dates_GRC(1),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC');
d1_GRC = datetime(dates_GRC(end),'InputFormat','yyyy-MM-dd HH:mm:ss Z','TimeZone','UTC')+1;
NumDays_GRC = daysact(d1_GRC,d2_GRC);

% comparing the number of days to number of columns in returns.
if (size(logChanges_GRC,1) == NumDays_GRC)
    fprintf('All days are included in source data for GRC. \n');
    fprintf('First datetime %s.\n',d1_GRC);
    fprintf('Last datetime %s.\n\n',d2_GRC);
end

% checking for NaN -elements
if sum(isnan(logChanges_BTC)) == 0 && sum(isnan(logChanges_DOGE))== 0 &&
sum(isnan(logChanges_GRC)) == 0
    fprintf('No NaN -values.\n')
end
```

```
All days are included in source data for BTC.
First datetime 16-Mar-2015 00:00:00.
Last datetime 24-Nov-2018 00:00:00.

All days are included in source data for DOGE.
First datetime 16-Mar-2015 00:00:00.
Last datetime 24-Nov-2018 00:00:00.

All days are included in source data for GRC.
First datetime 16-Mar-2015 00:00:00.
Last datetime 24-Nov-2018 00:00:00.

No NaN -values.
```

making a matrix of the dummy variables for every day

```
matrix = zeros(length(logChanges_BTC),7);

for i = 1:7:length(logChanges_BTC)
    for j = 0:1:6
        if i+j <= length(logChanges_BTC)
            matrix(i+j,j+1) = 1;
```

```
        end
    end
end

parameterNames =
{'DailyLogReturns','Monday','Tuesday','Wedneday','Thursday','Friday','Saturday','Sunday'};

table1 =
table(logChanges_BTC,matrix(:,1),matrix(:,2),matrix(:,3),matrix(:,4),matrix(:,5),matrix(:,6),
matrix(:,7),'VariableNames',parameterNames);
table2 =
table(logChanges_DOGE,matrix(:,1),matrix(:,2),matrix(:,3),matrix(:,4),matrix(:,5),matrix(:,6)
,matrix(:,7),'VariableNames',parameterNames);
table3 =
table(logChanges_GRC,matrix(:,1),matrix(:,2),matrix(:,3),matrix(:,4),matrix(:,5),matrix(:,6),
matrix(:,7),'VariableNames',parameterNames);
```

making the models

```
mdl_BTC = fitlm(table1,'DailyLogReturns ~
Monday+Tuesday+Wedneday+Thursday+Friday+Saturday+Sunday -1')

mdl_DOGE = fitlm(table2,'DailyLogReturns ~
Monday+Tuesday+Wedneday+Thursday+Friday+Saturday+Sunday -1')

mdl_GRC = fitlm(table3,'DailyLogReturns ~
Monday+Tuesday+Wedneday+Thursday+Friday+Saturday+Sunday -1')
```

```
mdl_BTC =


Linear regression model:
    DailyLogReturns ~ [Linear formula with 7 terms in 7 predictors]

Estimated Coefficients:
                   Estimate          SE         tStat       pValue
                 ___________    _________    ________    _______

    Monday         0.0014347    0.0027774     0.51655    0.60555
    Tuesday         0.002588    0.0027774     0.93181     0.3516
    Wedneday     -0.00058542    0.0027774    -0.21078    0.83309
    Thursday        0.002533    0.0027774     0.91202    0.36192
    Friday         0.0038722    0.0027774      1.3942    0.16349
    Saturday       0.0010971    0.0027846     0.39399    0.69365
    Sunday         0.0031482    0.0027846      1.1306    0.25844


Number of observations: 1349, Error degrees of freedom: 1342
Root Mean Squared Error: 0.0386

mdl_DOGE =


Linear regression model:
    DailyLogReturns ~ [Linear formula with 7 terms in 7 predictors]
```

```
Estimated Coefficients:
                  Estimate        SE         tStat       pValue
                 __________    _________    ________    _______

    Monday       -0.0010682    0.0048407    -0.22068    0.82537
    Tuesday      -0.0013144    0.0048407    -0.27153    0.78602
    Wedneday      0.0036799    0.0048407     0.76021    0.44727
    Thursday       0.005509    0.0048407      1.1381     0.2553
    Friday       -0.0027995    0.0048407    -0.57832    0.56314
    Saturday      0.0034624    0.0048533     0.71342    0.47571
    Sunday         0.007245    0.0048533      1.4928    0.13572


Number of observations: 1349, Error degrees of freedom: 1342
Root Mean Squared Error: 0.0672

mdl_GRC =


Linear regression model:
    DailyLogReturns ~ [Linear formula with 7 terms in 7 predictors]

Estimated Coefficients:
                  Estimate        SE         tStat        pValue
                 __________    _________    ________    ________

    Monday         0.013996    0.0066108      2.1172    0.034425
    Tuesday       -0.010518    0.0066108      -1.591     0.11184
    Wedneday      0.0062824    0.0066108     0.95032     0.34212
    Thursday      0.0073379    0.0066108        1.11      0.2672
    Friday        0.0011106    0.0066108       0.168     0.86661
    Saturday      -0.001265     0.006628    -0.19086     0.84866
    Sunday       -0.0085112     0.006628     -1.2841     0.19931


Number of observations: 1349, Error degrees of freedom: 1342
Root Mean Squared Error: 0.0918
```

Parameter stability testing

```
[h,pValue,stat,cValue] =
chowtest(matrix,logChanges_BTC,[675],'Intercept',false,'Display','summary');
[h,pValue,stat,cValue] =
chowtest(matrix,logChanges_DOGE,[675],'Intercept',false,'Display','summary');
[h,pValue,stat,cValue] =
chowtest(matrix,logChanges_GRC,[675],'Intercept',false,'Display','summary');
```

```
RESULTS SUMMARY

***************
Test 1

Sample size: 1349
Breakpoint: 675
```

```
Test type: breakpoint
Coefficients tested: All

Statistic: 0.3466
Critical value: 2.0164

P value: 0.9324
Significance level: 0.0500

Decision: Fail to reject coefficient stability


RESULTS SUMMARY

***************
Test 1

Sample size: 1349
Breakpoint: 675

Test type: breakpoint
Coefficients tested: All

Statistic: 1.7597
Critical value: 2.0164

P value: 0.0916
Significance level: 0.0500

Decision: Fail to reject coefficient stability


RESULTS SUMMARY

***************
Test 1

Sample size: 1349
Breakpoint: 675

Test type: breakpoint
Coefficients tested: All

Statistic: 1.1695
Critical value: 2.0164

P value: 0.3173
Significance level: 0.0500

Decision: Fail to reject coefficient stability
```

multicollinnearity test

```
inv(matrix'*matrix);
```

normal distribution of the residuals

```
% figure(4)
% btcResiduals = mdl_BTC.Residuals.Raw;
% plot(btcResiduals)
% [h,p,jbstat,critval] = jbtest(btcResiduals,0.05)
%
% figure(5)
% dogeResiduals = mdl_DOGE.Residuals.Raw;
% plot(dogeResiduals)
% [h,p,jbstat,critval] = jbtest(dogeResiduals,0.05)
%
% figure(6)
% grcResiduals = mdl_GRC.Residuals.Raw;
% plot(grcResiduals)
% [h,p,jbstat,critval] = jbtest(grcResiduals,0.05)
```

Statistics

```
% STDEV

% Sdev_BTC = std(logChanges_BTC)
% Sdev_DOGE = std(logChanges_DOGE)
% Sdev_GRC = std(logChanges_GRC)
%
% % MEAN, MAX, MIN
%
% max_BTC = max(logChanges_BTC)
% max_DOGE = max(logChanges_DOGE)
% max_GRC = max(logChanges_GRC)
%
% min_BTC = min(logChanges_BTC)
% min_DOGE = min(logChanges_DOGE)
% min_GRC = min(logChanges_GRC)
%
% mean_BTC = mean(logChanges_BTC)
% mean_DOGE = mean(logChanges_DOGE)
% mean_GRC = mean(logChanges_GRC)
```

Daily log returns

```
% BTC
logChanges_Monday_BTC = logChanges_BTC(1:7:end);
logChanges_Tuesday_BTC = logChanges_BTC(2:7:end);
logChanges_Wednesday_BTC = logChanges_BTC(3:7:end);
logChanges_Thursday_BTC = logChanges_BTC(4:7:end);
logChanges_Friday_BTC = logChanges_BTC(5:7:end);
logChanges_Saturday_BTC = logChanges_BTC(6:7:end);
logChanges_Sunday_BTC = logChanges_BTC(7:7:end);

% means of daily log returns
Means_BTC = [mean(logChanges_Monday_BTC)
mean(logChanges_Tuesday_BTC)
mean(logChanges_Wednesday_BTC)
mean(logChanges_Thursday_BTC)
mean(logChanges_Friday_BTC)
```

```
mean(logChanges_Saturday_BTC)
mean(logChanges_Sunday_BTC)];



% DOGE
logChanges_Monday_DOGE = logChanges_DOGE(1:7:end);
logChanges_Tuesday_DOGE = logChanges_DOGE(2:7:end);
logChanges_Wednesday_DOGE = logChanges_DOGE(3:7:end);
logChanges_Thursday_DOGE = logChanges_DOGE(4:7:end);
logChanges_Friday_DOGE = logChanges_DOGE(5:7:end);
logChanges_Saturday_DOGE = logChanges_DOGE(6:7:end);
logChanges_Sunday_DOGE = logChanges_DOGE(7:7:end);

% means of daily log returns
Means_DOGE = [mean(logChanges_Monday_DOGE)
mean(logChanges_Tuesday_DOGE)
mean(logChanges_Wednesday_DOGE)
mean(logChanges_Thursday_DOGE)
mean(logChanges_Friday_DOGE)
mean(logChanges_Saturday_DOGE)
mean(logChanges_Sunday_DOGE)];


% GRC
logChanges_Monday_GRC = logChanges_GRC(1:7:end);
logChanges_Tuesday_GRC = logChanges_GRC(2:7:end);
logChanges_Wednesday_GRC = logChanges_GRC(3:7:end);
logChanges_Thursday_GRC = logChanges_GRC(4:7:end);
logChanges_Friday_GRC = logChanges_GRC(5:7:end);
logChanges_Saturday_GRC = logChanges_GRC(6:7:end);
logChanges_Sunday_GRC = logChanges_GRC(7:7:end);

% means of daily log returns
Means_GRC = [mean(logChanges_Monday_GRC)
mean(logChanges_Tuesday_GRC)
mean(logChanges_Wednesday_GRC)
mean(logChanges_Thursday_GRC)
mean(logChanges_Friday_GRC)
mean(logChanges_Saturday_GRC)
mean(logChanges_Sunday_GRC)];
```

Published with MATLAB® R2018a