Predicting Cryptocurrencies with the **eDMA** package

Leopoldo Catania

Aarhus BSS and CREATES

European R Users Meeting (eRum 2018)

Budapest, Hungary
May 13–16, 2018

Based on: Catania, Grassi, and Ravazzolo (2018) and Catania and Nonejad (2018)

Motivation I

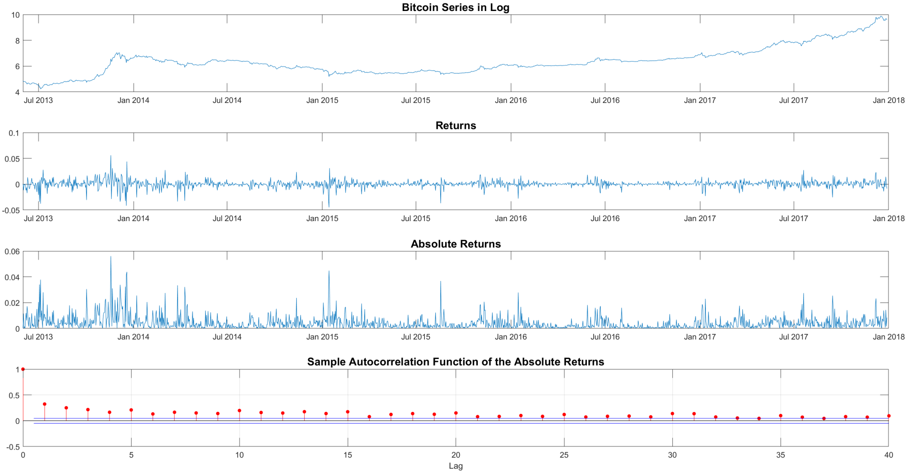
Motivation II

- Bitcoin is the first decentralized crypto–currency created in 2009 and documented in Nakamoto (2009).
- Since its inception, it gained a growing attention from the media, academics, and finance industry
- In recent months the global interest in Bitcoin and crypto–currencies has spiked dramatically:
    - Japan has recognized Bitcoin as a legal method of payment;
    - some central banks are exploring the use of the crypto–currencies;
    - a large number of companies and banks created the Enterprise Ethereum Alliance to make use of the crypto–currencies and the related technology called blockchain, Forbes (2017).
    - The Chicago Mercantile Exchange (CME) started the negotiation of Bitcoin futures on 18th of December 2017, see Exchange (2017). Nasdaq and Tokyo Financial Exchange will follow late in 2018, see Bloomberg (2017).
- All this interest has been reflected on the crypto–currencies market capitalization that exploded from around 19 billion in February 2017 to around 850 billion in January 2018.

Motivation III

- The dynamic of those series is quite complex displaying:
    - extreme observations;
    - asymmetries;
    - several nonlinear characteristics which are difficult to model, see Catania and Grassi (2017).
- New econometric model to study the characteristics of those series are needed.
- This is a new and unexplored market and these models will be important for asset allocation, risk management, and pricing of derivative securities.

Bitcoin


Bitcoin Series in Log
10
8
6
4
Jul 2013
Jan 2014
Jul 2014
Jan 2015
Jul 2015
Jan 2016
Jul 2016
Jan 2017
Jul 2017
Jan 2018
Returns
0.1
0.05
0
-0.05
Absolute Returns
0.06
0.04
0.02
0
Sample Autocorrelation Function of the Absolute Returns
1
0.5
0
-0.5
0
5
10
15
20
25
30
35
40
Lag

Our contribution

- We study the predictability of crypto–currencies time series.
- We compare several models in point and density forecasting of four of the most capitalized series: Bitcoin, Litecoin, Ripple and Ethereum.
- We apply a set of crypto–predictors and rely on Dynamic Model Averaging (DMA) to combine a large set of univariate Dynamic Linear Models with different forms of time variation.
- We find statistical significant improvements in point forecasting when using combinations of univariate models.
- The analysis is performed exploiting the **eDMA** package of Catania and Nonejad (2018).

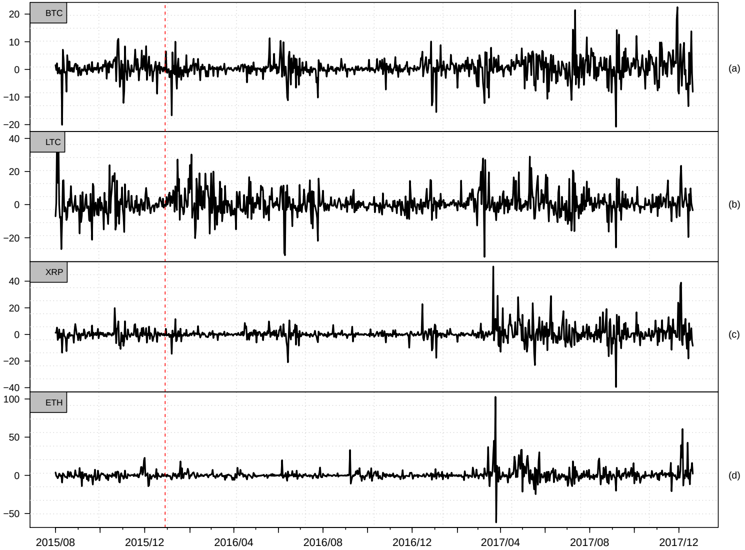


Figure: Four major crypto–currencies daily percentage log returns: (a) Bitcoin (BTC); (b) Litecoin (LTC); (c) Ripple (XRP); (d) Ethereum (ETH)

Crypto–currencies and Crypto–predictors

Data Overview		
Abbreviation	Full name	Transformation
Cryptocurrencies time series		
BTC	Bitcoin	First difference of Log
ETH	Ethereum	First difference of Log
XRP	Ripple	First difference of Log
LTC	Litecoin	First difference of Log
Additional crypto–explicative time series		
BTC_HL	Bitcoin High minus Bitcoin Low	Log
ETH_HL	Ethereum High minus Ethereum Low	Log
XRP_HL	Ripple High minus Ripple Low	Log
LTC_HL	Litecoin High minus Litecoin Low	Log
Lag	First, second and third lags of all cryptocurrencies	
Additional financial and macro time series		
CDS_5y	Europe credit default swap index 5 years	First difference of Log
ES_600	Stoxx Europe 600 - Price Index	First difference of Log
GLD	Gold Bullion LBM	First difference of Log
NK_225	Nikkei 225 Stock Average - Price Index	First difference of Log
SP_500	S&P 500 Composite - Price Index	First difference of Log
SV	Silver Handy & Harman Base Price	First difference of Log
BD_1m	1-Month US Treasury Constant Maturity Rate	First difference
BD_10y	10-Year US Treasury Constant Maturity Rate	First difference
VIX	VIX closing price	Log

Crypto–predictors

Having defined the set of crypto–predictors we would like to answer the following questions:

i) Are all these crypto–predictors relevant?

ii) Is their importance changed over time?

iii) Can be used to predict cryptocurrencies?

We answer to these questions by exploiting the Dynamic Model Averaging technique developed by Raftery et al. (2010) and implemented in the R package **eDMA** by Catania and Nonejad (2018).

What's DMA?

In order to obtain the best forecast possible, practitioners often try to take advantage of the many predictors available and seek to combine the information from these predictors in an optimal way, see Stock and Watson (1999).

DMA offers a statistical coherent way to combine predictions delivered by several models.

Which models? All possible dynamic linear regression models that can be built from subset of regressors!

How many are them? Assuming N possible crypto–predictors we have 2^N different models. In our case $N = 18$ i.e. 262'144 different models.

Some formula behind DMA

It is assumed that each model, $i \in (1, \ldots, 2^N)$, with associated vector of regressors at time t, $\boldsymbol{F}_t^{(i)}$, is a Dynamic Linear Model (DLM) á la West and Harrison (1999):

$$y_t = \boldsymbol{F}_t^{(i)\top} \boldsymbol{\theta}_t^{(i)} + \varepsilon_t^{(i)}, \quad \varepsilon_t^{(i)} \sim N\left(0, V_t^{(i)}\right) \tag{1}$$

$$\boldsymbol{\theta}_t^{(i)} = \boldsymbol{\theta}_{t-1}^{(i)} + \boldsymbol{\beta}_t^{(i)}, \quad \boldsymbol{\beta}_t^{(i)} \sim N\left(0, \boldsymbol{W}_t^{(i)}\right). \tag{2}$$

$\boldsymbol{W}_t^{(i)}$ is important since it affects the variability of the $\boldsymbol{\theta}_t^{(i)}$ coefficients. We implement the following update relying on a forgetting factor $\delta \in (0, 1)$:

$$\boldsymbol{W}_t^{(i)} = (1 - \delta)/\delta \boldsymbol{C}_{t-1}^{(i)},$$

where $\boldsymbol{C}_{t-1}^{(i)}$ is an estimate of the predicted covariance matrix of $\boldsymbol{\theta}_{t-1}^{(i)}$ delivered by the Kalman Filter.

How to select δ? We consider a grid of $d = 10$ values $\delta \in (0.9, 0.91, ..., 0.99)$ and perform DMA over these values. The total number of resulting models is now $2^N \times d = 2'621'440$

The **eDMA** package for R

- Efficiently implements a DMA procedure based on Raftery et al. (2010) and Dangl and Halling (2012).
- Routines are written in C++ using the **Armadillo** library of Sanderson (2010) exploiting the **Rcpp** and **RcppArmadillo** packages of Eddelbuettel and François (2011) and Eddelbuettel and Sanderson (2014), respectively.
- Parallel execution relays on the **OpenMP** API (ARB **OpenMP**, 2008).

The **eDMA** package is available on CRAN. It's functionalities are detailed in a JSS paper recently appeared: `10.18637/jss.v084.i11`.

DMA using **eDMA**

The SimData data set is available for illustration purposes. This is one draw from a DLM with variables x2, x3 and x4 plus a constant included. Variables x5 and x6 are fake variables.

```
R> data("SimData", package = "eDMA")
```

DMA is then performed using the function DMA() as

```
R> Fit <- DMA(y ~ x2 + x3 + x4 + x5 + x6, data = SimData,
              vDelta = seq(0.9, 1.0, 0.01))
```

DMA() automatically parallelize the code if the hardware allows for it!

Fit is an object of the class DMA and comes with several methods such as: show, plot, summary, coef and as.data.frame.

```
> plot(Fit)

Type 1-16 or 0 to exit
 1: Point forecast
 2: Predictive likelihood
 3: Posterior weighted average of delta
 4: Posterior inclusion probabilities of the predictors
 5: Posterior probabilities of the forgetting factors
 6: Filtered estimates of the regression coefficients
 7: Variance decomposition
 8: Observational variance
 9: Variance due to errors in the estimation of the coefficients, theta
10: Variance due to model uncertainty
11: Variance due to uncertainty with respect to the choice of
 the degrees of time-variation in the regression coefficients
12: Expected number of predictors (average size)
13: Number of predictors (highest posterior model probability) (DMS)
14: Highest posterior model probability (DMS)
15: Point forecasts (highest posterior model probability) (DMS)
16: Predictive likelihood (highest posterior model probability) (DMS)
Selection:
```

Posterior inclusion probabilities

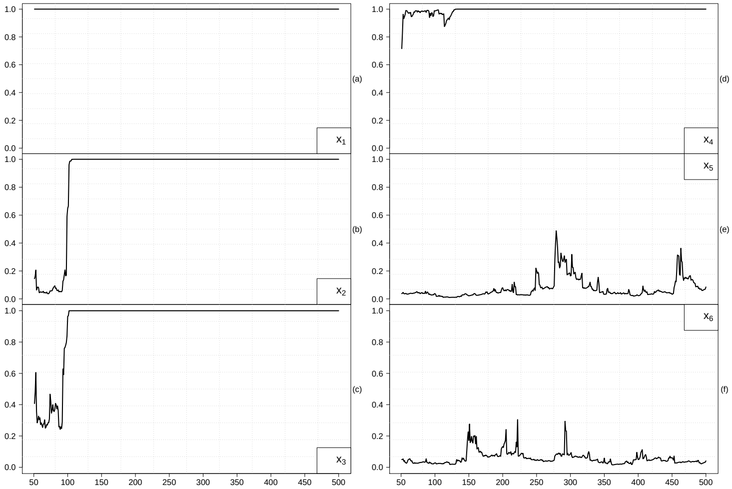


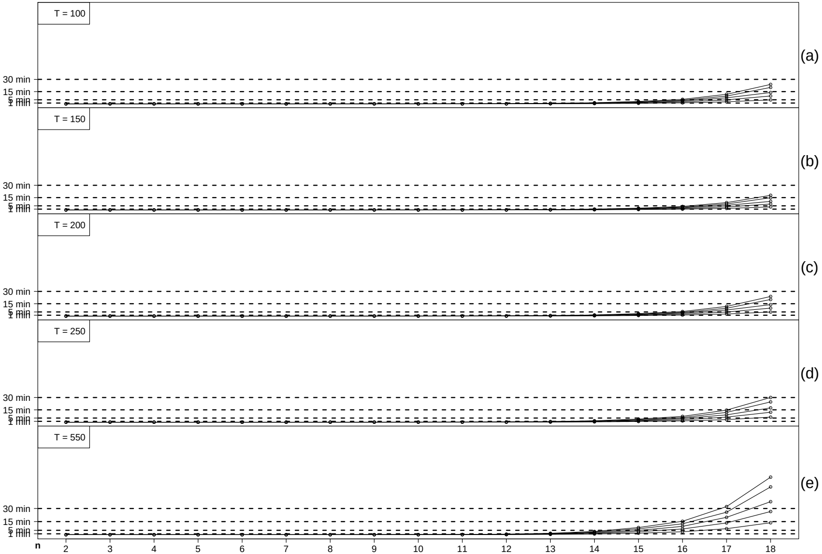
Figure: Posterior inclusion probabilities of the predictors using simulated data.

eDMA is fast!

T/N	4	6	8	10	12	14
100	34.5	41.4	60.7	81.6	69.5	54.5
500	47.3	54.3	92.9	82.4	70.5	49.0
1000	59.0	58.3	81.6	84.2	71.3	50.6

Table: Ratio of computation time between the dma() function from the **dma** package of McCormick et al. (2016) and the DMA() function of the **eDMA** package using different values of T and n. The ratio is computed using the average computational time taken after 10 code evaluations using the **microbenchmark** package of Mersmann (2018).

eDMA is fast!


T = 100
T = 150
T = 200
T = 250
T = 550
(a)
(b)
(c)
(d)
(e)
30 min
15 min
5 min
1 min
n
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18

Univariate models

All these univariate models can be estimated using **eDMA** by setting appropriate constraints, see Catania and Nonejad (2018).

Abbreviation	*Full Description*
AR(1)	Autoregressive model of order one, benchmark model.
KS	Kitchen Sink specification.
DMA	DMA across all models and forgetting factor combinations. See Dangl and Halling (2012).
DMS	Dynamic Model Selection (DMS).

Table: Univariate models considered in the forecasting exercise. The first column is the model's abbreviation. The second column provides a brief description of each individual model.

Forecasting Measures

- **Mean squared error** for each currency:

$$\text{MSE}_i = \sqrt{\frac{1}{T-R}\sum_{t=R}^{T-1}\left(\hat{y}_{t+h|t} - y_{t+h}\right)^2},$$

with T = number of observations; R = length of rolling window; $\hat{y}_{t+h|t}$ = individual crypto forecasts.

- **Average log predictive score** is broadest measure of density accuracy for each currency:

$$\text{PL}_{t+h}(y_{t+1}) = \ln \underbrace{\left(f\left(y_{t+h}|I_t\right)\right)}_{\substack{\text{predictive density for } y_{t+h} \\ \text{using infos up to t}}}$$

- Log predictive score can be computed for joint multivariate predictions Y_{t+h}.
- Tests: **Diebold-Mariano** for pairwise comparison (significance difference in bold); **model confidence set** for joint comparison (significant inclusion in grey).

Summary I

```
Call:

 DMA(formula =  vY ~ cds5y + eurostoxx600 + gold + nikkei225 + silver + sp5
 vix + bonds1m + bonds10y + bitcoin.HighLow + ethereum.HighLow +
 litecoin.HighLow + ripple.HighLow + ethereum + litecoin +
 ripple + Lag.1 + Lag.2 + Lag.3 )

Residuals:
     Min        1Q   Median       3Q      Max
-11.4253   -0.2826   0.0341   0.4746   9.9955
Coefficients:
                 E[theta_t] SD[theta_t] E[P(theta_t)] SD[P(theta_t)]
(Intercept)           -0.07        0.33          1.00           0.00
cds5y                  0.05        0.32          0.14           0.18
eurostoxx600           0.00        0.12          0.15           0.16
gold                  -0.04        0.15          0.19           0.18
nikkei225             -0.03        0.18          0.18           0.22
silver                 0.00        0.04          0.21           0.16
sp500                  0.00        0.16          0.18           0.20
vix                   -0.02        0.21          0.50           0.23
```

Summary II

```
bonds1m             0.02      0.04       0.20      0.18
bonds10y            0.00      0.10       0.17      0.21
bitcoin.HighLow     0.00      0.16       0.20      0.19
ethereum.HighLow   -0.02      0.08       0.36      0.20
litecoin.HighLow    0.01      0.12       0.19      0.13
ripple.HighLow     -0.04      0.42       0.27      0.27
ethereum           -0.03      0.13       0.54      0.29
litecoin            0.00      0.05       0.19      0.15
ripple             -0.01      0.11       0.19      0.13
Lag.1              -0.01      0.03       0.15      0.14
Lag.2              -0.01      0.06       0.16      0.20
Lag.3               0.00      0.04       0.14      0.17

Variance contribution (in percentage points):
  vobs vcoeff   vmod   vtvp
 11.61  80.86   7.27   0.26

Top 10% included regressors: (Intercept), ethereum
```

```
Forecast Performance:
                                  DMA      DMS
MSE                             1.338    1.354
MAD                             0.759    0.762
Log-predictive Likelihood -829.689 -870.350
```

Univariate forecasting results: MSE

h	1	2	3	4	5	6	7
				Bitcoin			
AR1	42.49	42.28	41.54	41.62	41.55	41.42	41.12
KS	1.52	12.25	1.84	**0.96**	1.06	1.06	1.12
DMA	**0.97**	**0.97**	1.01	1.04	1.02	1.02	1.13
DMS	1.01	1.02	1.06	1.06	1.02	1.05	1.15
				Litecoin			
AR1	134.27	132.88	133.05	133.43	133.60	133.25	131.71
KS	1.02	1.17	7.64	1.01	1.17	1.11	1.09
DMA	0.98	1.03	1.09	1.11	1.03	1.06	1.15
DMS	1.00	1.07	1.11	1.11	1.04	1.09	1.22

Table: Mean squared error (MSE), computed over the forecast horizon. Results are reported relative to the benchmark specification (AR1) for which the absolute score is reported. Models' description is reported in Table 2.

Univariate forecasting results: MSE

h	1	2	3	4	5	6	7
				Ripple			
AR1	224.02	221.31	222.02	221.13	218.93	219.62	219.45
KS	1.11	1.24	1.27	1.10	1.76	1.21	2.01
DMA	1.20	1.03	1.22	1.25	1.10	1.18	1.17
DMS	1.27	1.05	1.22	1.26	1.11	1.21	1.21
				Ethereum			
AR1	180.57	174.99	175.61	175.56	175.79	175.90	174.08
KS	1.05	12.72	1.09	1.01	1.02	1.67	1.09
DMA	0.97	1.01	1.03	1.01	1.04	1.04	1.04
DMS	1.02	1.04	1.08	1.05	1.05	1.09	1.04

Table: Values in **bold**, indicate rejection of the null hypothesis of Equal Predictive Ability between each model and the benchmark according to the Diebold–Mariano test at the 5% confidence level. Gray cells indicate those models that belong to the Superior Set of Models delivered by the Model Confidence Set procedure at confidence level 10%.

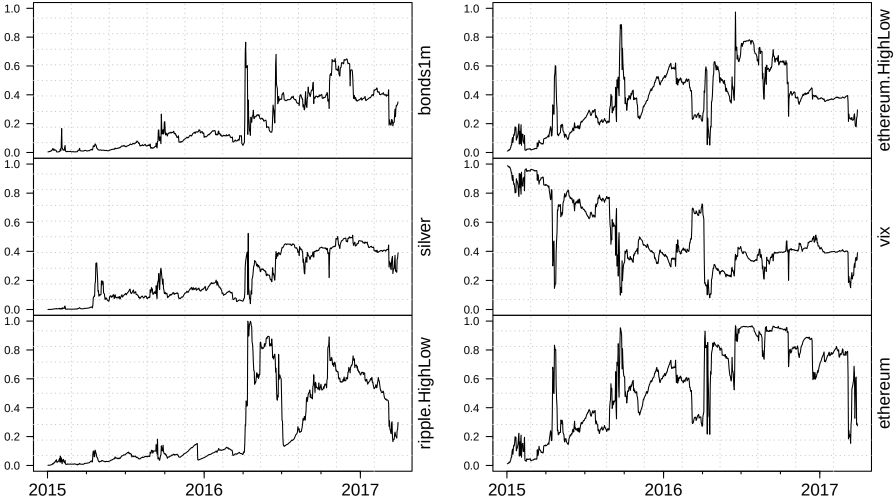


Figure: Posterior inclusion probabilities of the predictors for Bitcoin.

Posterior inclusion probabilities: Litecoin

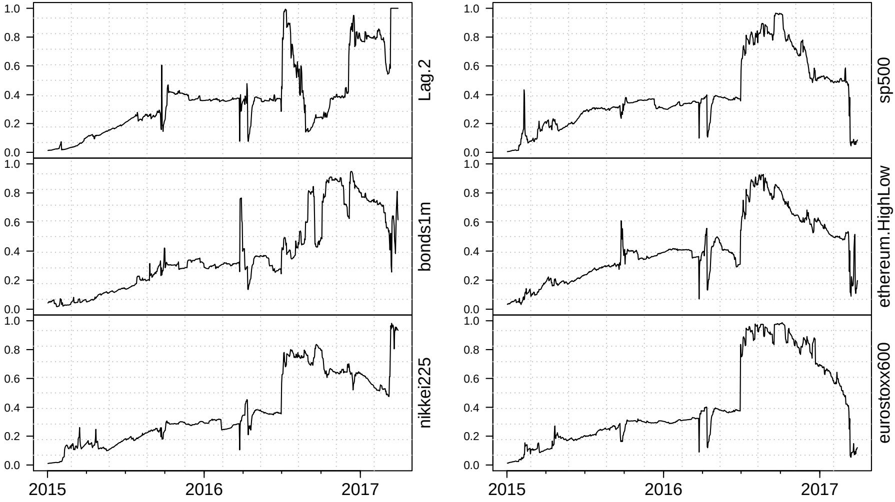


Figure: Posterior inclusion probabilities of the predictors for Litecoin.

Average number of predictors

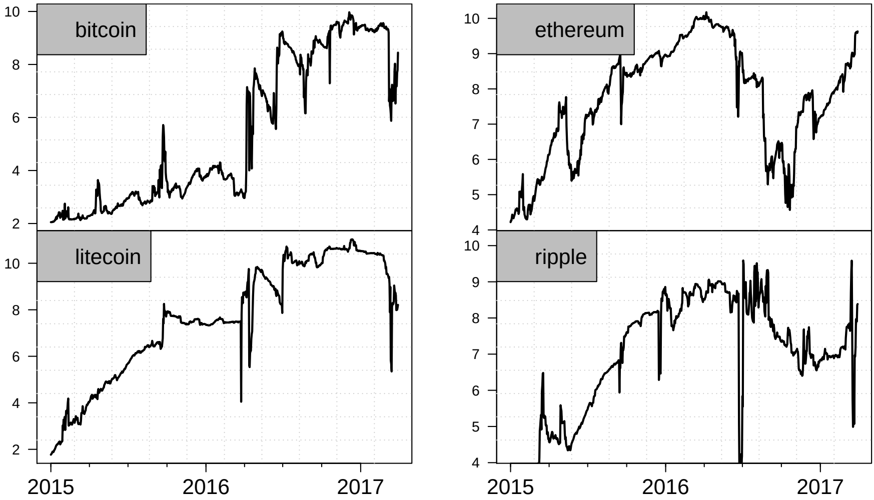


Figure: Average number of predictors for Bitcoin, Ethereum, Litecoin, and Ripple.

Conclusion

- We study the predictability of crypto–currencies time series.
- We compare several alternative models in point and density forecasting of four of the most capitalized series: Bitcoin, Litecoin, Ripple and Ethereum
- Best predictors are: i) Equity EU and USA markets, ii) US interest rates, iii) past volatility levels.
- Cryptocurrencies predictability has increased over time.
- Extension: risk management.

References I

ARB **OpenMP**. ***OpenMP** Application Program Interface*, May 2008. URL http://www.openmp.org/mp-documents/spec30.pdf. Version 3.0.

Bloomberg. Nasdaq Plans to Introduce Bitcoin Futures. 2017.

Leopoldo Catania and Stefano Grassi. Modelling crypto-currencies financial time-series. 2017.

Leopoldo Catania and Nima Nonejad. Dynamic model averaging for practitioners in economics and finance: The edma package. *Journal of Statistical Software, Articles*, 84(11):1–39, 2018. ISSN 1548-7660. doi: 10.18637/jss.v084.i11. URL https://www.jstatsoft.org/v084/i11.

Leopoldo Catania, Stefano Grassi, and Francesco Ravazzolo. Forecasting cryptocurrencies financial time series. 2018.

Thomas Dangl and Michael Halling. Predictive Regressions with Time–Varying Coefficients. *Journal of Financial Economics*, 106: 157–181, 2012.

Dirk Eddelbuettel and Romain François. **Rcpp**: Seamless R and C++ integration. *Journal of Statistical Software*, 40(8):1–18, 2011. doi: 10.18637/jss.v040.i08.

Dirk Eddelbuettel and Conrad Sanderson. **RcppArmadillo**: Accelerating R with high-performance C++ linear algebra. *Computational Statistics & Data Analysis*, 71:1054–1063, 2014. doi: 10.1016/j.csda.2013.02.005.

Chicago Mercantile Exchange. CME Group Announces Launch of Bitcoin Futures. 2017.

Forbes. Emerging Applications For Blockchain. 2017.

Tyler H. McCormick, Adrian Raftery, and David Madigan. ***dma**: Dynamic Model Averaging*, 2016. URL https://CRAN.R-project.org/package=dma. R package version 1.2-3.

Olaf Mersmann. *microbenchmark: Accurate Timing Functions*, 2018. URL https://CRAN.R-project.org/package=microbenchmark. R package version 1.4-4.

S. Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. *Available at: https://Bitcoin.org/Bitcoin.pdf*, 2009.

Adrian E Raftery, Miroslav Kárný, and Pavel Ettler. Online prediction under model uncertainty via dynamic model averaging: Application to a cold rolling mill. *Technometrics*, 52(1):52–66, 2010. doi: 10.1198/tech.2009.08104.

Conrad Sanderson. **Armadillo**: An open source C++ linear algebra library for fast prototyping and computationally intensive experiments. Technical report, NICTA, 2010. URL http://arma.sourceforge.net/.

James H Stock and Mark W Watson. Forecasting inflation. *Journal of Monetary Economics*, 44(2):293–335, 1999. doi: 10.1016/s0304-3932(99)00027-6.

Mike West and Jeff Harrison. *Bayesian Forecasting & Dynamic Models*. Springer-Verlag, 1999.