Statistical Arbitrage Pairs Trading
Preprocessing and tests for co-integration using R

Final Exam Project for Financial Econometrics
MSc in Advanced Economics and Finance

Financial Econometrics | May 20, 2020 | Copenhagen Business School
Exam Number: S133708 | Department of Economics

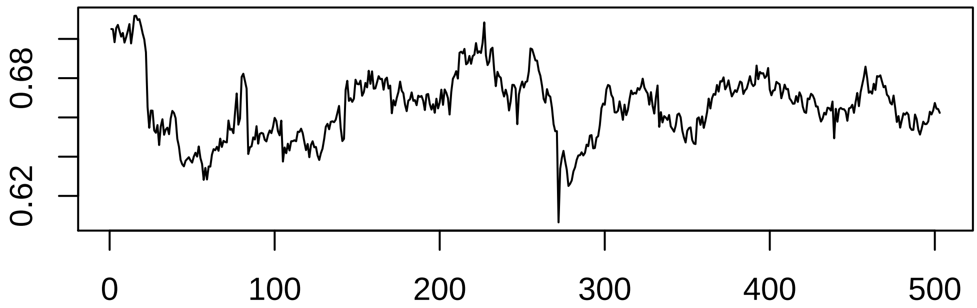


Figure 1: Engle-Granger Estimated Spread: $log(P_V) - 0.8\ log(P_M)$

1 Introduction

One of the most influential concepts in financial theory is the concept of no-arbitrage best represented as law of one price, implying that in absence of all frictions two identical goods should have the same price. Using this no-arbitrage argument Ross (1976) introduced the general Arbitrage Pricing Theory (APT), a theoretical framework for modelling expected asset returns as a linear combination of multiple factors. The main prediction is that the price of an asset should equal the sum of all future cash flows discounted at the implied APT rate. An implication of this theory is that if any one asset diverges from its predicted price an investor can achieve risk-free profits from either buying or shorting this asset, depending on the direction of the mispricing, and take an opposing position in a portfolio of assets meant to simulate a correctly priced asset with identical characteristics.

This last implication of the APT implies that temporary negligible deviations from the predicted prices can occur and at any given time two similar assets can be either equally or inversely mispriced. However, any such relative mispricing should be of a temporary nature, as any meaningful deviations would imply a risk-free arbitrage opportunity.

The trading strategy known as “pairs trading” seeks to develop statistical bounds for identifying and trading such arbitrage opportunities. Fundamentally the strategy is based on constructing a portfolio of opposing positions in two similar securities and exploit any deviations. The trading strategy is non-directional and market neutral as it involves trading the price differential, i.e. the spread, between two securities.

The fundamental concept for identifying a tradable pair of securities and defining the statistical bounds on their price-spread is the concept of cointegration and the main intention of this paper is establishing a framework for identifying and testing for cointegration between two securities and estimation meaningful models of the spread. The main methods will be the Engle-Granger two-step procedure and the Johansen approach, where each approach have different advantages and disadvantages. The final outcome of both approaches is an error-correction model that adjusts for the cointegration relationship and can be assessed using different diagnostic measures.

2 Data

The basis of the analysis is data extracted from *Yahoo! Finance* using the *quantmod* package for R. The specific data is the daily stock prices on Visa (V) and Mastercard (MA) from January 1, 2018 to January 1, 2020. The total length of the data is 2 years with $T = 503$ observations. As will be discussed in subsection 2.2 the selection of a stock pair can be based on complex algorithms or as is the case in this paper an intuitive approach based on theory. The selection of V and MA is the simple result of being two similar U.S. financial service companies listed on the same stock exchange, namely NYSE.

2.1 Price Quotes

The price quotes used is the daily closing price adjusted for dividends and stocks splits. This is at odds with the recommendation by Vidyamurthy (2004) to use the volume weighted average price but is simply a result of practicality and data accessibility and does not affect the statistical analysis itself. However, in real-world applications it would adversely affect the trading signals.

Another important choice is using the natural logarithm of the adjusted stock quotes, as this transformation have attractive statistical properties. Most notably time-additivity, making the compounding of returns computationally easier, as the compounded return in any given period is the simple difference in log-prices:

$$\sum_{i=1}^{n} log(1 + r_i) = log(P_n) - log(P_0)$$

Where $log(P_i)$ is the log-price of an asset and $log(1+r_i) = log(P_i) - log(P_{i-1})$ is the log-return of an asset.

This simple transformation is consistent with much financial research as the assumption of log-normal returns is widely used. This assumption has some drawbacks for real applications as it would entail continuous re-balancing of the portfolio, because the ratio in market value of the portfolio would have to be kept constant rather than the ratio of the stocks, as discussed by Chan (2013).

2.2 Pair Selection

As presented in the introduction, pairs trading is founded on the idea that similar characteristic stocks should exhibit similar price behavior. Thus, when selecting a stock pair for trading, the main criteria is similar characteristic companies. This is implemented either by developing parametric (i.e. APT based) or non-parametric selection algorithms or as proposed by Tsay (2010) and Vidyamurthy (2004) simply selected based on theory and intuition.

However, no selection method is sufficient to imply a meaningful and tradeable statistical relationship. Using correlation for selection might be easy to implement, but is not sufficient by itself to makes a stock pair interesting from a statistical arbitrage perspective, although it might be indicative of the necessary property of cointegration. The existence of cointegration between the price series will be examined in the methodology (3) and results (4) sections of this paper, but in the initial analysis in the next section evidence is found for the important prerequisite for cointegration, namely that the price series is integrated of the same order $I(d)$.

2.3 Preliminary Analysis

Before investigating if the two log-price processes are indeed co-integrated and thereby relevant for modelling using the framework described in the methodology section (3), this section independently examines the two time series in order to determine if the series by themselves is stationary. To answer this question the log-price series and their respective Autocorrelation Function (ACF) and Partial ACF of different orders are visually inspected (Plots are included in the R output in the appendix: figure A.3 and A.4 respectively) and next two different tests for stationarity will be conducted, namely the Dickey-Fuller (DF) and the KPSS test proposed by Kwiatkowski et al. (1992).

2.3.1 Dickey-Fuller test

The first test procedure proposed by Dickey and Fuller (1979) tests if AR-model has a unit root and involves testing if $a_1 = 1$ in the model $y_t = a_1 y_{t-1} + \epsilon_t$. To do this the model is transformed by subtracting y_{t-1} on both sides, such that $\Delta y_t = \gamma y_{t-1} + \epsilon_t$, $\gamma \equiv (a_1 - 1)$ and the null hypothesis is constructed such that $\gamma = 0$. Dickey & Fuller expands on this approach to also cover models including an intercept a_0 or both an intercept and a deterministic drift term $a_2 t$. If included more deterministic terms, an Augmented Dickey-Fuller (ADF) test using the same fundamentals can also be used. In either case the null hypothesis is presence of a unit root in the model, and for the $AR(1)$ model this implies that the model can be considered a random walk (with or without intercept and drift term depending on which model is tested). For stock-prices this would comply with the random walk hypothesis and implicitly the efficient market hypothesis. Considering how debated these hypothesis is, a random walk in any form would not be an unexpected result when conducting the Dickey-Fuller test on stock price data.

Table A: ADF Test Statistics and Critical Values

	log-prices		log-returns	
	V	MA	V	MA
τ	1.60	1.89	-24.76	-24.21
1%	-2.58	-2.58	-2.58	-2.58
5%	-1.95	-1.95	-1.95	-1.95
10%	-1.62	-1.62	-1.62	-1.62
τ_μ	-1.19	-1.38	-24.88	-24.37
1%	-3.43	-3.43	-3.43	-3.43
5%	-2.86	-2.86	-2.86	-2.86
10%	-2.57	-2.57	-2.57	-2.57
τ_τ	-3.54	-3.35	-24.86	-24.35
1%	-3.96	-3.96	-3.96	-3.96
5%	-3.41	-3.41	-3.41	-3.41
10%	-3.12	-3.12	-3.12	-3.12

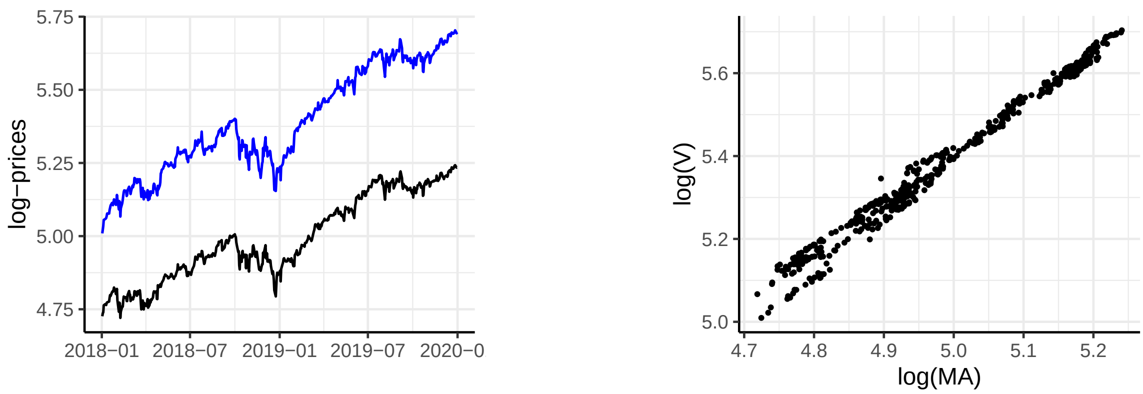
The Dickey-Fuller test is conducted using the *ur.df* function in R. Before undergoing the test, it is important to consider how the process should be modelled and specifically how many lag terms should be included in the model. The *ur.df* function have a built-in lag selection algorithm based on either the Aikaike or Bayesian Information Criteria (AIC and BIC respectively), using BIC for the basis of lag length selection indicates a 1-lag model for both log-prices and log-returns (i.e. the first difference of the log prices: $log(P_t) - log(P_{t-1})$). Using the AIC indicates lag lengths of up to 6, but with many insignificant lags in-between. The different results from AIC and BIC is expected as the BIC generally tends towards more parsimonious models and the significance of the 6th lag that AIC indicates should be included is the likely result of using daily stock quotes, i.e. with jumps every fifth day around weekends. Consequently just a single lag is included in the models at this stage.

The critical values of the DF-test are dependent on the sample size and which of the models is being tested. The critical values is estimated by Dickey and Fuller (1979) using a simulation approach and provided when running the *ur.df* routine in R. The test statistics as well as the critical values are summarized in table (A), where τ, τ_μ and τ_τ corresponds to the test statistic from the pure random walk model, the model with intercept and the model with intercept and drift respectively.

From the DF-test results in table (A) the non-stationary nature of the log-prices are confirmed as the null hypothesis of prevalence of a unit root cannot be rejected at the 5% significance level for both price-series in all DF-models. However, when running the same procedure for the log-returns (i.e. first difference of the log-prices), the null hypothesis is clearly rejected, even at the 1% level. Indicating that the log-prices are $I(1)$ processes.

2.3.2 KPSS test

I will not include an elaborate description of the KPSS test procedure, but the test is run using the *kpss.test* routine in R and code is provided in the appendix. The KPSS test is a useful complement to the DF-test, as it tests for trend-stationarity against the presence of a unit root, with the null hypothesis being stationarity. The R routine is unable to estimate p-values outside of the 1% to 10% interval, but the p-values obtained, are either very close to or less than 1% (greater than 10%) when testing the MA and V log-prices (log returns) respectively and the KPSS results is thus consistent with the findings from the DF-tests as in KPSS the null is rejected at the 1%-level for the log-prices but maintained for the log-return series of both stocks ($p > .10$).



(a) log(Adjusted Closing Price)

(b) Scatterplot of log(MA) and log(V)

Figure 2: Log-prices and scatterplot of Mastercard and Visa, Jan. 1, 2018 to Jan. 1, 2020. Prices used are log-transformed adjusted closing prices from Yahoo Finance, extracted using the Quantmod package in R. In (a) black line is Visa and blue line is Mastercard. In (b) the correlation coefficient is very high at 0.99.

3 Methodology

Having concluded the individual analysis on the two stock price series and determined that the log-price series are likely $I(1)$ processes, this section seeks to explain the concept of cointegration in further detail and introduce two different approaches to testing for cointegration, relevant in the analysis and implementation of pairs trading. The two approaches introduced is the Engle-Granger two-step procedure and the Johansen procedure and while they have different advantages and drawbacks and uses different estimators (OLS and MLE respectively) they should produce asymptotically identical result.

3.1 Cointegration

As already discussed, the basis of this paper is to identify if two stock prices are indeed cointegrated, as this property is fundamental when constructing a model for statistical arbitrage pairs trading. Cointegration is defined by a linear combination of two nonstationary processes being weakly stationary and for pairs trading, this linear combination is coined the spread between two series and is defined as:

$$x_1 - \gamma x_2 = \mu + e_t \tag{1}$$

If this spread is stationary then the two series $\{x_1\}$ and $\{x_2\}$ is $CI(1,1)$, i.e. cointegrated of order 1.

As cointegration measures the difference in means of two series it is intrinsically different from the familiar Pearson correlation coefficient ρ, that gauges how well two series move in tandem. Although no direct relation between cointegration and correlation exist and it is possible for two series to be uncorrelated $\rho = 0$ but still cointegrated $CI(1,1)$[*], two cointegrated stock prices are often highly correlated and because inference on correlation are much simpler than for cointegration a selection algorithm for pairs trading can advantageously be based on the correlation measure (Vidyamurthy, 2004).

In the following sections the Engle-Granger and Johansen approaches to testing for cointegration and estimating useful models is reviewed.

3.2 Engle-Granger approach

In their 1987 paper Robert Engle and Clive Granger proposes a two-step method for determining if two nonstationary series are cointegrated (Engle and Granger, 1987). The first step involves estimating the γ coefficient in the regression: $x_1 = \mu + \gamma x_2 + e_t$ using ordinary least squares (OLS). Note that this estimated model is identical to the spread defined in the previous section (1), with μ assumed to be a constant.

The second step involves testing if e_t is stationary, which it would be if the series is $CI(1,1)$ and is composed of an adapted Dickey-Fuller test on the estimated residual series (e_t). Enders (2014) builds on the standard Engle-Granger two-step procedure to make it suitable for practical implementations and estimation of an error-corrected model. The four steps of Enders expanded approach is explained stepwise in relation to this paper below:

Step 1) A requirement for two processes to be cointegration, is that they are individually integrated of the same order. This first step involves pretesting the individual processes for their order of integration $I(d)$. This pretesting of the variables is already explained in the data section, where the variables is tested using the Dickey-Fuller (2.3.1) and KPSS tests (2.3.2).

Step 2) The second step comprises Engle-Grangers original two-step procedure and involves estimating the long-run equilibrium relationship using OLS in the standard linear regression form:

$$p_t^V = \mu + \gamma p_t^M + e_t \tag{2}$$

Where p_t^V, p_t^M denotes the log-prices of Visa and Mastercard respectively, μ is the long-run equilibrium and the residual e_t corresponds to the deviation from the long-run equilibrium in period t.

The second part involves checking stationarity of the residual series $\{e_t\}$ by conducting a Dickey-Fuller test. If $\{e_t\}$ is stationary then the two price series is $CI(1,1)$ and the spread (2) can be considered stationary as μ is a constant by construction.

The Dickey-Fuller tests if $a_1 = 0$ in the model $\Delta \hat{e}_t = a_1 \hat{e}_{t-1} + \epsilon_t$, where $\hat{e}_t$ is the estimated residuals. As the null hypothesis is $a_1 = 0$, i.e. presence of a unit root, then if H_0 is rejected then stationarity of e_t and implicitly stationarity of model 2 can*not* be rejected and the two series is said to be cointegrated of order 1. As this DF test is conducted on the estimated residual sequence, the critical values of the test needs to be adjusted accordingly and relative to the sample size and number of variables included. These critical values is found in supplementary material in Enders (2014) and is included in the appendix of this paper as figure A.1.

As a diagnostic check, one should also test if the residuals $\{\epsilon_t\}$ from the DF regression appears as white noise, as any serial correlation in these residuals would indicate that more lags should be added to the model.

Step 3) The third step involves estimating an error-correction model (ECM), using the residuals from the equilibrium regression as the error correction term: $\text{ECT}_t \equiv e_{t-1} = p_t^V - \gamma p_t^M - \mu$, this necessi-

[*] I refer to the r-bloggers.com entry by Kammers and Smith (2017) for good visual representations of such examples.

tates that the residual sequence is stationary and the two variables is $CI(1,1)$. The error-corrected model will take the form of a Vector Autoregressive model (VAR) in first differences:

$$\begin{pmatrix} \Delta p_t^V \\ \Delta p_t^M \end{pmatrix} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} \mathrm{ECT}_t + A_1 \begin{pmatrix} \Delta p_{t-1}^V \\ \Delta p_{t-1}^M \end{pmatrix} + \boldsymbol{e_t} \quad (3)$$

Where A_1 is a 2×2 coefficient matrix and $\boldsymbol{e_t}$ is a 2-row vector of residuals. Note that this model corresponds to a 2-lag model in first differences with an ECT included.

The model is efficiently estimated using OLS as all terms are stationary and standard inference can be made using the estimated t statistics on single variables or χ^2 and F-tests can be conducted for multiple restrictions, which is a favorable feature when conducting the next step:

Step 4) The final step in Enders (2014) 4-step adaptation of the Engle-Granger procedure is conducting diagnostic tests to assess the adequacy of the ECM. Different approaches can be taken to validate if the ECM is appropriate and the results from these tests is explained and summarized in the results section (4.1) as they are dependent on the specific model used. For both general use and specifically for trading, the size of the α-coefficients have important implications as they define the speed of adjustment when the variables are not in equilibrium. In general, convergence to equilibrium requires that $\alpha_1 < 0$ and $\alpha_2 > 0$.

3.3 Johansen approach

When a single cointegration relationship is investigated the Engle-Granger two-step procedure is easy to implement. However, the procedure also have some drawbacks, some of which are relevant for this paper others less so. One of the major drawbacks is the compounding of estimation errors, as the procedure calls for estimation in two distinct steps where errors from the first part is compounded into the last step.

Another drawback is the decision of which variable should be the dependent one in the linear model of the long-run equilibrium. In the asymptotic case the relation is unaffected by this choice but in finite samples this decision might result in different estimates of the spread. Lastly, another important constraint of the Engle-Granger procedure is that it just allows for testing of a single cointegrating relationship. However, this is sufficient for the pair trading approach in this paper, as it is based on trading of just two securities.

An alternative approach is the maximum likelihood estimation (MLE) proposed by Johansen (1988). The Johansen approach is based on a multivariate approach where the two-step estimation is avoided by estimating the model using a VAR framework and the result is a single Vector Error-Corrected Model (VECM). For the two variable case the asymptotic distribution of the two different procedures is identical, but the Johansen procedure have the important advantage of allowing restriction testing directly on the cointegrating and speed-of-adjustment vectors.

In the following paragraphs Enders (2014) four-step adaptation of the Johansen procedure will be outlined and the actual estimation and testing will be described in the results section (4.2).

Step 1) As identical order of integration of the individual series is always a necessary condition for cointegration this should be also be pretested in the Johansen procedure. This can be tested using the Dickey-Fuller and KPSS tests as discussed in section 2.3.1 and 2.3.2 respectively.

Next, the very important decision about choosing the correct lag length should be considered, as the estimation of the VECM is very sensitive to misspecification. Fortunately, the correct lag length from an undifferenced VAR model can be generalized to the Johansen setting and any procedure to determine this lag length is applicable here. In this simple two variable case, the multivariate generalizations of the Aikaike and Schwarz-Bayesian information criteria (AIC and BIC) is used and the chosen lag length is then investigated using different diagnostic tests based on the assumption that the residuals of a well-specified VAR model should resemble white noise. These diagnostic tests are discussed in the results section (4.2).

Step 2) After pretesting the individual variables order of integration and determining the lag length, the next step regards the actual estimation of the Johansen model and testing for the number of cointegrating relationship (in this two series case a single cointegrating relation is tested). To get an understanding of the Johansen model the simple VAR(2) model is stated below:

$$\boldsymbol{p_t} = A_1 \boldsymbol{p}_{t-1} + A_2 \boldsymbol{p}_{t-2} + \boldsymbol{e_t} \quad (4)$$

Where $\boldsymbol{p_t} = \begin{pmatrix} p_t^V & p_t^M \end{pmatrix}'$ is the dependent variable vector and A_1 and A_2 is 2×2 coefficient matrices.

Next the Johansen VAR(2) model is transformed using an approach similar to that of the Dickey-Fuller model for univariate case, by subtracting $\boldsymbol{p_{t-1}}$ on both sides and defining the return vector as $\boldsymbol{r_t} \equiv \boldsymbol{p_t} - \boldsymbol{p_{t-1}}$:

$$\boldsymbol{r_t} = (A_1 - I)\boldsymbol{p}_{t-1} + A_2 \boldsymbol{p}_{t-2} + \boldsymbol{e_t} \quad (5)$$

Where I the 2×2 identity matrix, and in order to make all determinants stationary $A_2 \boldsymbol{p}_{t-1}$ is added and subtracted on the right-hand side:

$$\boldsymbol{r_t} = (A_1 + A_2 - I)\boldsymbol{p}_{t-1} + A_2 \boldsymbol{r}_{t-1} + \boldsymbol{e_t} \quad (6)$$

Letting $\Pi \equiv -(I - A_1 - A_2)$ and $\Gamma_1 \equiv -A_2$, the Johansen VECM is defined as †:

$$\boldsymbol{r_t} = \Pi \boldsymbol{p}_{t-1} - \Gamma_1 \boldsymbol{r}_{t-1} + \boldsymbol{e_t} \tag{7}$$

This approach can be generalized to include more lags, an unrestricted constant term and/or a deterministic trend term as will be discussed in the results section (4.2.1).

The interesting feature in this model is the Π matrix, as it is the rank of this matrix that determines the number of individual cointegration vectors. In this two variable case $rank(\Pi) = 1$ for the two stock price series to be $CI(1,1)$. If $rank(\Pi) = 0$ then the series would not be cointegrated and if $rank(\Pi) = 2$ (i.e. full rank) then the individual log-price series would be independently stationary.

Knowing that the rank of Π is equal to the number of characteristic roots different from zero, two test statistics can be formulated λ_{trace} and λ_{max}. These will not be defined any further in this paper but the definitions can be found in Enders (2014). However, it is worth noting that both statistics are using the estimated characteristic roots $\hat{\lambda_i}$ and that the null hypothesis is either $H_0 : rank(\Pi) \leq r$ against a general alternative or $H_0 : rank(\Pi) = r$ against the well-defined alternative $H_a : rank(\Pi) = r + 1$ for the λ_{trace} and λ_{max} tests respectively.

Step 3) After having determined the rank r of Π, the matrix can be disaggregated such that, for the two variable case:

$$\underset{(2\times2)}{\Pi} = \underset{(2\times r)(r\times2)}{\boldsymbol{\alpha\beta}} \tag{8}$$

If the two series is $CI(1,1)$ then $\boldsymbol{\beta}$ corresponds to the cointegration matrix of unique cointegration vectors, where $\boldsymbol{\beta p_{t-1}}$ would be stationary and the $\boldsymbol{\alpha}$ matrix would denote the speed of adjustment coefficients, like in the Engle-Granger ECM

As mentioned in the previous step the model (7) can be expanded to also include a drift term but it is also possible to adapt the cointegration vector to include a intercept, much like in the Engle-Granger case (When the first step is estimated with an unrestricted intercept). This implies that Π and $\boldsymbol{p_{t-1}}$ is transformed, such that:

$$\underset{(2\times3)}{\Pi^*} = \begin{pmatrix} \pi_{11} & \pi_{12} & \pi_{13} \\ \pi_{21} & \pi_{22} & \pi_{23} \end{pmatrix}$$

$$\underset{(3\times1)}{\boldsymbol{p_{t-1}}^*} = \begin{pmatrix} p^V_{t-1} & p^M_{t-1} & 1 \end{pmatrix}'$$

Having established this, the third step of the 4-step procedure involves investigating the cointegration vectors and the components of the $\boldsymbol{\alpha}$ vector as these represent the speed of adjustment coefficients. As for the Engle-Granger case $\alpha_1 < 0$ and $\alpha_2 > 0$ is still necessary conditions for convergence to equilibrium. Using the Johansen framework allows meaningful testing of restricted versions of the both.

Step 4) The last step involves causality testing and innovation accounting of the estimated model. Specifically for pairs trading this step serves an important purpose as these diagnostic tests are indicative of the usefulness of the estimated model from a trading perspective, e.g. how large is equilibrium-deviation and how quick will the system react. These characteristics are essential to investigate when deciding on the tradability and implementing actual trading signals using the model.

Actual analysis for trading applications such as determining trading signals and tradability of the assets will not be discussed in detail in this paper, but as a diagnostic check plots of the impulse response function and forecast error variance decomposition will be inspected.

3.4 Model Assessment

This section will briefly outline the two main methods for assessing the usefulness of the error-corrected models, that is included in more detail in the results section 4.2.2.

Impulse Response Function) VAR(p) models can similarly to AR(p) models be expressed in an infinite moving average (VMA) representation, corresponding to a linear function of past innovations:

$$\boldsymbol{p_t} = \boldsymbol{\mu} + \sum_{i=0}^{\infty} \boldsymbol{\Phi_i e_{t-i}} \tag{9}$$

This representation is useful as it allows for analysis of the coefficients in the 2×2 matrix $\boldsymbol{\Phi_i}$ that defines the impulse response function. As an additional tool, the Cholesky decomposition on the residuals 2×2 covariance matrix $\boldsymbol{\Sigma}$ can be applied, as high correlation between the residuals complicates interpretation. This lower-triangular Cholesky decomposition of the residuals covariance matrix will be denoted $\boldsymbol{\Sigma^*}$.

Variance Decomposition) Using the same VMA representation (9) in an n-step in-sample forecast can help estimating the amount of information that each of the variables contribute to the other. This forecast error variance decomposition can also assist in determining if the variables are endogenous. As for the Impulse Response Function, the Cholesky decomposition of the residual covariance matrix can be helpful when assessing the results.

†In higher-order VAR(p) models the estimated Γ_i can take two different forms depending on what effects is investigated. In the long-run specification $\Gamma_i \equiv (-I + A_1 + ... + A_i),\ i = 1, ..., p-1$, while in the transitory specification $\Gamma_i \equiv -(A_{i+1} + ... + A_p),\ i = 1, ..., p-1$, where p indicates the lag length from the VAR(p) model. In the simple VAR(2) example $\Gamma_1 \equiv -A_2$.

4 Results

In this section the methods from the preceding sections will be applied on the adjusted stock price data of Visa and Mastercard to determine if the stock prices are indeed cointegrated and whether the estimated error-correction models is useful in modelling the spread. Having already defined the spread as: $p_t^V - \gamma p_t^M - \mu$, modelling of this spread is pivotal in successful pairs trading, as the trading strategy is based on buying one unit of the first security and simultaneously shorting γ unit of the other security. The total return of this portfolio is thus equal to the spread in first differences: $p_t^V - \gamma p_t^M - \mu - (p_{t-1}^V - \gamma p_{t-1}^M - \mu) = r_t^V - \gamma r_t^M$.

On the following pages the estimated models, diagnostic tests and results from the Engle-Granger and Johansen procedure will summarized and discussed.

4.1 Engle-Granger Approach

Following Enders (2014) adapted Engle-Granger procedure, the first step is determining the order of integration of the individual price series. This initial analysis is already covered in the data section (2.3) where reasonable evidence is found that both log-price series is $I(1)$. After this initial analysis the next step is investigating if the two series is cointegrated using the Engle-Granger two step procedure:

First the linear model is estimated using the *lm* routine in R (t-statistics in parentheses).

$$\hat{p}_V = \underset{(29.8)}{0.67} + \underset{(193.5)}{0.80}\,\hat{p}_M + \hat{e} \tag{10}$$

The model has an Adj. $R^2 = .99$ and is very significant: $F(1,501) = 37450,\ p < .001$ and both parameters: intercept $\mu = 0.67$ and cointegration coefficient $\gamma = 0.8$ is also very significant $p < .001$.

Proceeding to the Dickey-Fuller test for stationarity of the errors, the τ-statistic is estimated as -3.973 using the *ur.df* routine in R. This should be compared to the 1% critical value from Enders supplementary table (figure A.1) of -3.921 $(n > 500)$ and the null hypothesis of a unit root is rejected at the 1%-level implying that cointegration of the two stock price processes can*not* be rejected.

These results are sensitive to the specification of the model and if too few lags is included, the residuals of from the DF regression $\{\epsilon_t\}$ will exhibit serial correlation or other forms of deterministic behaviour. A visual inspection of the $\{\epsilon_t\}$ series and its ACF and PACF indicates that the residuals $\{\epsilon_t\}$ can be characterized as a white noise process. These plots are included in the R output in the appendix (figure A.5).

More advanced diagnostics can be applied to the DF-test results, as is the case when the Johansen procedure is conducted (Section 4.2) but based on the residual plots it is concluded that the model is sufficiently specified to proceed with estimation of the error-corrected model:

4.1.1 Engle-Granger Error-Corrected Model

Using the *VECM* routine from the *tsDyn* R package to estimate the error-corrected model using Engle-Grangers two-step procedure and including the intercept as a constant in the ECT yields the following models (p-values in parentheses):

$$\hat{r}_t^V = \underset{(.138)}{-0.056}\ \mathrm{ECT}_t - \underset{(.008)}{0.275}\ \hat{r}_{t-1}^V + \underset{(.053)}{0.173}\ \hat{r}_{t-1}^M + \hat{\epsilon}_t^V$$

$$\hat{r}_t^M = \underset{(.808)}{0.011}\ \mathrm{ECT}_t - \underset{(.149)}{0.171}\ \hat{r}_{t-1}^V + \underset{(.589)}{0.056}\ \hat{r}_{t-1}^M + \hat{\epsilon}_t^M$$

Where: $\hat{r}_t^i = \Delta\hat{p}_t^i$ is the log-return of asset i in period t and the error-correction term ECT_t equals the lagged residuals $\{\hat{e}_{t-1}\}$ from model 10.

When assessing the Engle-Granger error-corrected model it should be noted that although the speed of adjustment coefficients (i.e. the coefficients on the ECT term) indicates the right direction for the variables to converge to equilibrium, both estimates are insignificant at any relevant significance level ($p = 0.14$ and $p = 0.81$ respectively) and this might be a result of misspecification, which would show up as autocorrelation in the residual series $\hat{\epsilon}_t^V$ $\hat{\epsilon}_t^M$. However, a visual inspection of the residual plots does not indicate that this is the case (figure A.6).

It should again be noted that these results are constrained by how the Engle-Granger two-step procedure is constructed, as any errors from misspecification might carry over from the first to the second step of the procedure. As a diagnostic test, the *VECM* routine is re-run using a lag length of 6, that was indicated by the AIC in the initial analysis of section 2.3.1. Using this lag length changes the estimate of α_1 to -0.061 with $p = .120$, i.e. still insignificant at the 10%-level. Adding more variables also entails some detrimental effects, especially in this context where all additional lags are insignificant and thus this longer 6-lag length model is not a desirable choice.

Next, a similar model will be estimated using the Johansen procedure from section 3.3 and more sophisticated diagnostic tests will be introduced for the multivariate case.

4.2 Johansen Procedure

Having already established the $I(1)$ nature of the log-price series in the previous section, the initial step of the Johansen procedure is specifying the lag length to be used. As introduced in the methodology section, this is done within a traditional VAR framework and in this case the multivariate generalizations of the AIC and BIC is used by running the *VARselect* routine from the *vars* R package. The estimated information criteria values is illustrated in figure 3 and analogous to the univariate case different lag lengths is indicated by the two criteria. BIC indicates a parsimonious 1-lag model and AIC indicates using $p = 3$.

Utilizing that the residuals in a well-specified model should be well-behaved (i.e. white noise), several diagnostic tests from the *vars* R package is run (Residual plots is also provided in appendix as figure A.7):

- *serial.test*: The first test is the Portmanteau test for serial correlation in the error term with the null hypothesis of *no* autocorrelation.
- *arch.test*: The second test is the multivariate ARCH-LM test with the null hypothesis of *no* Autoregressive conditional heteroskedasticity (ARCH).
- *normality.test*: The last tests is the multivariate Jarque-Bera test for the joint hypothesis of skewness of 0 and excess kurtosis of 0, i.e. normality. The test also includes output from the separate statistics for skewness and kurtosis respectively.

The diagnostic test results are summarized in table (B) for both $p = 1$ and $p = 3$ cases and from these results the validity of the lag lengths can be determined.

Beginning with the Portmanteau test for autocorrelation in the errors, which is of a particular interest when deciding on the lag length as any misspecification of this would imply serially correlated errors. As the null hypothesis is *no* autocorrelation the 3-lag model is preferable as H_0 cannot be rejected at any reasonable significance level. For the 1-lag model however, the null of no autocorrelation is rejected at $p = 0.03$. The null of the two additional tests are clearly rejected for both models, indicative of non-normality and volatility clustering. The latter is also an obvious conclusion from a visual inspection of the residuals (A.7).

In conclusion of the diagnostic tests on the residuals a lag length of 3 should be preferred when estimating the VECM in the next step. This is a consequence of the residuals from the VAR(3) model being better behaved based on the results from the Portmanteau test and most likely satisfies the weak stationarity assumption necessary to progress.

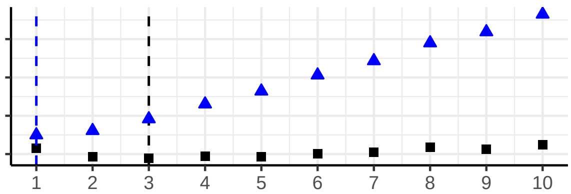


Figure 3: AIC (■) and BIC (▲)

4.2.1 Johansen Model

Running the *VECM* function from the *tsDyn* R package on the two variables yields the estimated 3-lag Johansen VECM (11). As a cross-check the model is also estimated using the *ca.jo* function from the *urca* package. Where the *VECM* function is easier to apply when representing the model in error-corrected form, the *ca.jo* function is better suited for testing restrictions on the model. In the following sections output from both functions is used interchangeably.

It is worth noting that no unrestricted mean or trend is included in the model, but the Π and $\boldsymbol{p_t}$ matrices are adapted to include a constant term in the cointegration vectors.

Table B: Multivariate tests on the VAR models

	BIC ($p = 1$)			AIC ($p = 3$)		
	χ^2	*df*	*p-value*	χ^2	*df*	*p-value*
Portmanteau Test	81.79	60	0.03	55.91	52	0.33
ARCH	130.40	45	0.00	104.80	45	0.00
JB-Test	2530.73	4	0.00	2956.63	4	0.00
Skewness only	132.85	2	0.00	190.47	2	0.00
Kurtosis only	2397.89	2	0.00	2766.16	2	0.00

Table C: λ_{trace} and $\lambda_{\max}$ test statistics

		Trace				Maximum Eigen		
	H_0	λ_{trace}	5%	1%	H_0	$\lambda_{\max}$	5%	1%
Constant	$r = 0$	19.77	19.96	24.60	$r = 0$	14.40	15.67	20.20
	$r \leq 1$	5.37	9.24	12.97	$r = 1$	5.37	9.24	12.97
None	$r = 0$	14.93	17.95	23.52	$r = 0$	13.93	14.90	19.19
	$r \leq 1$	1.00	8.18	11.65	$r = 1$	1.00	8.18	11.65

$$\begin{pmatrix} \hat{r}_t^V \\ \hat{r}_t^M \end{pmatrix} = \begin{pmatrix} -0.021 & 0.017 & 0.011 \\ 0.045 & -0.037 & -0.024 \end{pmatrix} \begin{pmatrix} \hat{p}_{t-1}^V \\ \hat{p}_{t-1}^M \\ 1 \end{pmatrix} - \begin{pmatrix} 0.297 & -0.174 & 0.191 & -0.070 \\ 0.184 & -0.048 & 0.162 & -0.046 \end{pmatrix} \begin{pmatrix} \hat{r}_{t-1}^V \\ \hat{r}_{t-1}^M \\ \hat{r}_{t-2}^V \\ \hat{r}_{t-2}^M \end{pmatrix} + \begin{pmatrix} \hat{\epsilon}_t^V \\ \hat{\epsilon}_t^M \end{pmatrix} \tag{11}$$

In addition the the representation in model 11, the estimated eigenvalue vector is estimated as: $\hat{\lambda} = \begin{pmatrix} 0.028 & 0.011 & 0.000 \end{pmatrix}$ and the estimated test statistics λ_{trace} and λ_{max} is summarized in table C. Somewhat surprisingly neither test statistic can reject the null of $rank(\Pi) = 0$ at the 5% level. This is surprising as even though an additional lag is included, this conclusion is different from the Engle-Granger results. The p-values of the two tests ($H_0: \ r = 0$) is $p = 0.057$ and 0.084 for the trace and maximum eigenvalue statistics respectively. This indicates a single cointegration vector at the 10% level but concluding this would be very incautious especially if used for actual trading.

Having raised a red flag in regards to the significance of the cointegration vector, the analysis will carry on as if $rank(\Pi) = 1$ and one cointegration vector actually exists.

This single cointegration vector $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_0)$, where the components corresponds to the coefficient of the two lagged level variables p_{t-1}^V, p_{t-1}^M and the constant term respectively, is estimated in the normalized form (such that $\beta_1 = 1$), as:

$$\boldsymbol{\beta} = \begin{pmatrix} 1.00 & -0.83 & -0.53 \end{pmatrix} \tag{12}$$

And the speed of adjustment coefficient vector $\boldsymbol{\alpha}$ is estimated as:

$$\boldsymbol{\alpha} = \begin{pmatrix} -0.0211 & 0.0451 \end{pmatrix}' \tag{13}$$

The components of the estimated cointegration vector is very close to the estimates from the Engle-Granger linear model (10), which is confirmed by a visual inspection of the Johansen long-run error (figure 4) relative to the Engle-Granger estimated spread (front page figure 1).

Also, when interpreting the speed-of-adjustment coefficients in the α vector, they have valid signs for convergence, but as in the Engle-Granger model they are still insignificant ($p = 0.56$ and $p = 0.28$ respectively).

Table D: LR-test, restrictions on $\boldsymbol{\beta}$

H_0	χ^2	*df*	*p-value*
$\beta_0 = 0$	9.42	2	.009
$\beta_2 = -1$	9.80	2	.007
$\beta_0 = 0$ and $\beta_2 = -1$	10.12	2	.006
$\alpha_1 = 0$	0.22	1	.639
$\alpha_2 = 0$	0.22	1	.384

Restriction Testing on $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$) In addition to the t-test statistics on the coefficients, the Johansen methodology allows for testing restrictions of the cointegration vector. Using the *blrtest* function from the *urca* R package, three restricted forms of the cointegration vector is tested against the original using the Likelihood Ratio test. The restricted vectors tested is:

$$\beta_1^* = \begin{pmatrix} 1.00 & -0.83 & 0 \end{pmatrix}$$
$$\beta_2^* = \begin{pmatrix} 1.00 & -1 & -0.53 \end{pmatrix}$$
$$\beta_3^* = \begin{pmatrix} 1.00 & -1 & 0 \end{pmatrix}$$

The test results is summarized in table D and interpreting these implies that the null hypothesis of $\beta_0 = 0$ and $\beta_1 = -1$ is rejected independently ($p < .01$ and $p < .01$) and jointly ($p < .01$). Also restricted forms of the $\boldsymbol{\alpha}$ vector is tested, these results indicate that neither speed-of-adjustment coefficient is significant ($p = .64$ and $p = .38$) as was also the conclusion for the Engle-Granger model (4.1.1) and a expected result as the χ^2-tests on $\boldsymbol{\alpha}$ is asymptotically equivalent to the t-statistics on the coefficients from the model estimation ($p = .56$ and $p = .28$).

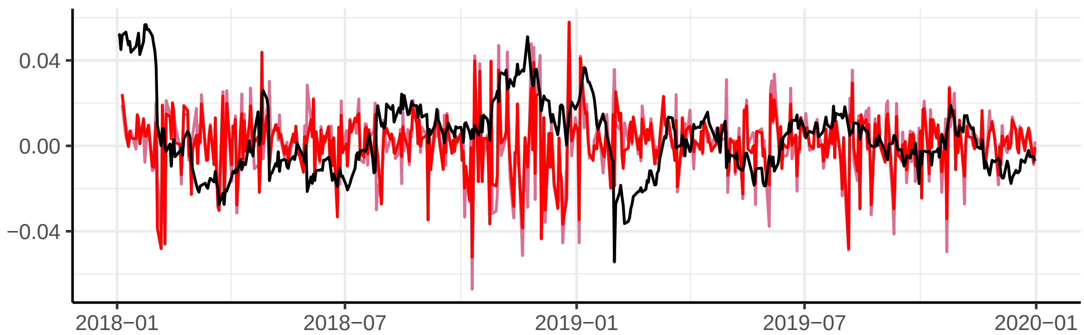


Figure 4: Plot of $\{\hat{\epsilon}_t^V\}$, $\{\hat{\epsilon}_t^M\}$ and long-run error (red, pink and black lines respectively)

$$\text{corr}(\hat{e}_t^V, \hat{e}_t^M) = \begin{pmatrix} 1.00 & 0.90 \\ 0.90 & 1.00 \end{pmatrix} \tag{14}$$

$$\hat{\boldsymbol{\Sigma}}^* = \begin{pmatrix} .0136 & .0000 \\ .0141 & .0068 \end{pmatrix} \tag{15}$$

VECM Representation Letting $\text{ECT}_t \equiv \boldsymbol{\beta' p_{t-1}}$, $\boldsymbol{\epsilon_t} \equiv \begin{pmatrix} \epsilon_t^V & \epsilon_t^M \end{pmatrix}'$ and $\boldsymbol{r_{t-i}} \equiv \begin{pmatrix} r_{t-i}^V & r_{t-i}^M \end{pmatrix}'$, $i = 0, ..., 2$ and adjusting the deterministic terms slightly, so the $\boldsymbol{r_{t-1}}$ and $\boldsymbol{r_{t-2}}$ vectors have separate coefficient matrices, the error-corrected representation of model 11 is illustrated as model 16. The error-correction term ECT_t represents the deviations from long-run equilibrium as in the Engle-Granger case and is visualized by the main line of figure 4.

Having defined the vector error-corrected model, the final step of Enders 4-step adaptation of the Johansen procedure is causality testing and innovation accounting to determine the validity of the model, especially important from a trading perspective.

Before undertaking this assessment it is useful to represent the VECM as a VAR model. Using the *vec2var* routine from *vars* R package on the VECM model (16) changes the representation to a VAR(3) model with the undifferenced $\boldsymbol{\hat{p}_t}$ vector on the left hand side (17), this is a practical representation as the analysis concerns the undifferenced variables.

4.2.2 Model Assessment

The first diagnostic check on the model is interpreting the impulse response function. As mentioned in section 3.4, it can be useful to do a Cholesky Decomposition on the residual matrix $\boldsymbol{\Sigma}$, especially when the residual correlation is high. The estimated correlation matrix is illustrated in equation 14 and because of the very high correlation in the residuals of the model, the interpretation of the impulse response functions using the standard covariance matrix $\hat{\boldsymbol{\Sigma}}$ is not very meaningful and using the $chol$ routine in R provides the Choleski Decomposition matrix (Illustrated in equation 15).

Using the *irf* R function to estimate and plot the impulse response of the VAR(3) representation provides the plots in figure A.8. It should be noted that the usual convergence to zero in a IRF plot should not be expected in any cointegrated VAR(p) model, as the included variables per definition is non-stationary. However, having noted this, the IRF plots are somewhat irregular as the correlation is of such extend and the shock to V in the model results in an almost symmetric effect on both variables.

Running the *fevd* R function, which automatically uses the orthogonalized matrix, emphasizes this conclusion. The plot of the variance decomposition is illustrated in figure A.9 and illustrates how almost all of the forecast error variance in both variables are explained by exogenous shocks to V.

This effect is however the result of the Choleski Decomposition and how the model is ordered. Had Mastercard been the first dependent variable, then this interdependent relation would have been in the opposite direction.

In conclusion of the impulse response function and forecast error variance decomposition, these diagnostic results is not very supportive of the overall usefulness of the model.

$$\boldsymbol{\hat{r}_t} = \begin{pmatrix} -0.021 \\ 0.045 \end{pmatrix} ECT_t - \begin{pmatrix} 0.3164 & -0.1949 \\ 0.2059 & -0.0719 \end{pmatrix} \boldsymbol{r_{t-1}} - \begin{pmatrix} 0.2049 & -0.0871 \\ 0.1767 & -0.0653 \end{pmatrix} \boldsymbol{r_{t-2}} + \boldsymbol{\hat{\epsilon}_t} \tag{16}$$

$$\boldsymbol{\hat{p}_t} = \begin{pmatrix} 0.011 \\ -0.024 \end{pmatrix} + \begin{pmatrix} 0.662 & 0.212 \\ -0.161 & 1.035 \end{pmatrix} \boldsymbol{p_{t-1}} + \begin{pmatrix} 0.112 & -0.108 \\ 0.029 & -0.007 \end{pmatrix} \boldsymbol{p_{t-2}} + \begin{pmatrix} 0.205 & -0.087 \\ 0.177 & -0.065 \end{pmatrix} \boldsymbol{p_{t-3}} + + \boldsymbol{\hat{e}_t} \tag{17}$$

5 Conclusion

In extension to the discoveries from the diagnostic test in the previous section, the overall tradability and usefulness of the estimated models is doubtful and considering the opposing results of the cointegration tests, it is still inconclusive if the log-price series of Visa and Mastercard is actually cointegrated or just highly correlated. That said, this project have demonstrated some important methods for identifying and testing for cointegration: namely the Engle-Granger two-step procedure, that provides a simple approach to modelling but has its inherent drawbacks as possible errors from the first-step carry over to the second part and the perhaps more severe limitation of not being able to test restrictions on the adjustment coefficients or the cointegration vector, this being one of the major advantage of the more sophisticated Johansen procedure that also allows for testing of multiple cointegrating relationships and thus provides a more general framework.

A proposal for further investigation would be to consider the relation using different time horizons and quoting intervals. The decision to use 2-year daily quotes was somewhat arbitrarily decided on and it is likely that shorter horizons would imply less variation in the underlying risk factors influencing the security prices and thereby increase the likelihood of a cointegrating relation under such horizons. The discussion of horizon is also highly relevant for real-life implementation of any trading strategy, as one would have to decide on a trading horizon to "bet on" and then construct statistical bounds for when a position should be entered and how much the position can deviate from these bounds before being closed in the given horizon.

References

Chan, E. (2013). *Algorithmic trading: winning strategies and their rationale*, volume 625. John Wiley & Sons.

Dickey, D. A. and Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American statistical association*, 74(366a):427–431.

Enders, W. (2014). *Applied Econometric Time Series, 4th Edition*. Wiley Series in Probability and Statistics. Wiley.

Engle, R. F. and Granger, C. W. (1987). Co-integration and error correction: representation, estimation, and testing. *Econometrica: journal of the Econometric Society*, pages 251–276.

Johansen, S. (1988). Statistical analysis of cointegration vectors. *Journal of economic dynamics and control*, 12(2-3):231–254.

Kammers, E. and Smith, C. (2017). Cointegration, correlation, and log returns. Accessed 2020-05-16 at: https://www.r-bloggers.com/cointegration-correlation-and-log-returns.

Kwiatkowski, D., Phillips, P. C., Schmidt, P., Shin, Y., et al. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of econometrics*, 54(1-3):159–178.

Ross, S. (1976). The arbitrage theory of capital asset pricing. *Journal of Economic Theory*, 13(3):341–360.

Tsay, R. (2010). *Analysis of Financial Time Series*. Wiley series in probability and statistics. Wiley.

Vidyamurthy, G. (2004). *Pairs Trading: Quantitative Methods and Analysis*. Wiley Finance. Wiley.

Appendix

Critical Values from Enders (2014) Supplementary Material

TABLE C: Critical Values for the Engle–Granger Cointegration Test

T	1%	5%	10%	1%	5%	10%
		Two Variables			*Three Variables*	
50	−4.123	−3.461	−3.130	−4.592	−3.915	−3.578
100	−4.008	−3.398	−3.087	−4.441	−3.828	−3.514
200	−3.954	−3.368	−3.067	−4.368	−3.785	−3.483
500	−3.921	−3.350	−3.054	−4.326	−3.760	−3.464
		Four Variables			*Five Variables*	
50	−5.017	−4.324	−3.979	−5.416	−4.700	−4.348
100	−4.827	−4.210	−3.895	−5.184	−4.557	−4.240
200	−4.737	−4.154	−3.853	−5.070	−4.487	−4.186
500	−4.684	−4.122	−3.828	−5.003	−4.446	−4.154

The critical values are for cointegrating relations (with a constant in the cointegrating vector) estimated using the Engle–Granger methodology.

Source: Critical values are interpolated using the response surface in MacKinnon (1991)

Figure A.1: Critical Values for the Engle-Granger Cointegration Test

Table F: Critical Values for $\beta_1 = 0$ in the Error–Correction Model

k		$T^a = 50$	$T^a = 100$	$T^a = 200$	$T^a = 500$
		No Intercept or Trend ($d = 0$)			
2	1%	−3.309	−3.259	−3.235	−3.220
	5%	−2.625	−2.609	−2.602	−2.597
	10%	−2.273	−2.268	−2.266	−2.265
3	1%	−3.746	−3.683	−3.652	−3.633
	5%	−3.047	−3.026	−3.016	−3.009
	10%	−2.685	−2.680	−2.677	−2.675
4	1%	−4.088	−4.015	−3.979	−3.957
	5%	−3.370	−3.348	−3.337	−3.331
	10%	−3.000	−2.997	−2.995	−2.994
		Intercept but no Trend ($d = 1$)			
2	1%	−3.954	−3.874	−3.834	−3.811
	5%	−3.279	−3.247	−3.231	−3.221
	10%	−2.939	−2.924	−2.916	−2.911
3	1%	−4.268	−4.181	−4.138	−4.112
	5%	−3.571	−3.538	−3.522	−3.512
	10%	−3.216	−3.205	−3.199	−3.195
4	1%	−4.537	−4.446	−4.401	−4.374
	5%	−3.819	−3.789	−3.774	−3.765
	10%	−3.453	−3.447	−3.444	−3.442
		Intercept and Trend ($d = 2$)			
2	1%	−4.451	−4.350	−4.299	−4.269
	5%	−3.778	−3.733	−3.710	−3.696
	10%	−3.440	−3.416	−3.405	−3.398
3	1%	−4.712	−4.605	−4.552	−4.519
	5%	−4.014	−3.971	−3.949	−3.935
	10%	−3.662	−3.643	−3.634	−3.629
4	1%	−4.940	−4.831	−4.776	−4.743
	5%	−4.221	−4.182	−4.162	−4.150
	10%	−3.857	−3.846	−3.840	−3.837

Note: T^a is the adjusted sample size equal to $T - (2k - 1) - d$ where T is the usable sample size, d is the number of deterministic regressors, and k is the number of $I(1)$ variables in the model. The critical values are calculated using equation (26) in Ericsson and MacKinnon (2002).

Figure A.2: Critical Values for $\beta_1 = 0$ in the Error-Corrected Model

R Output

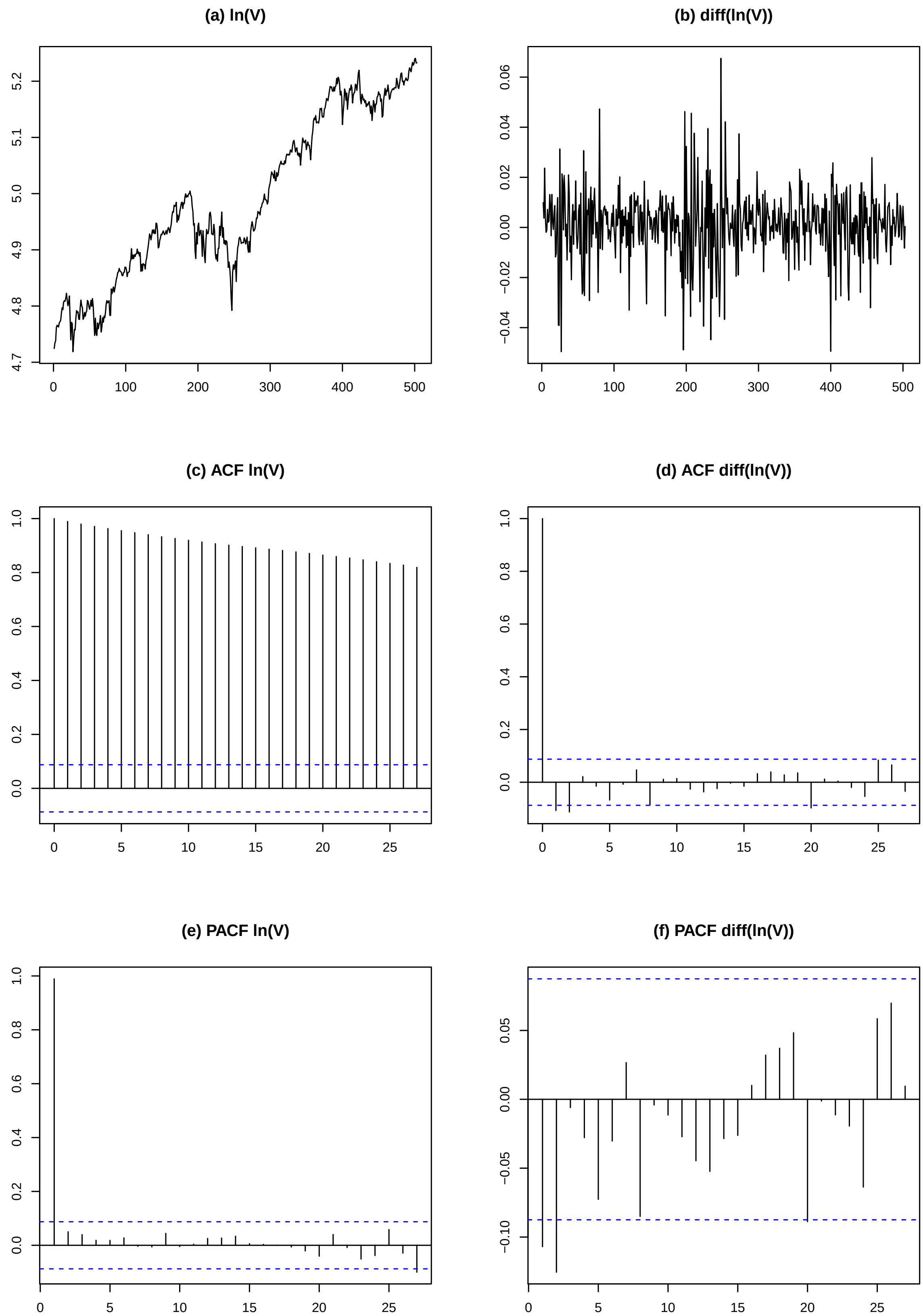


Figure A.3: Visa: log-price, ACF, PACF and differenced data.

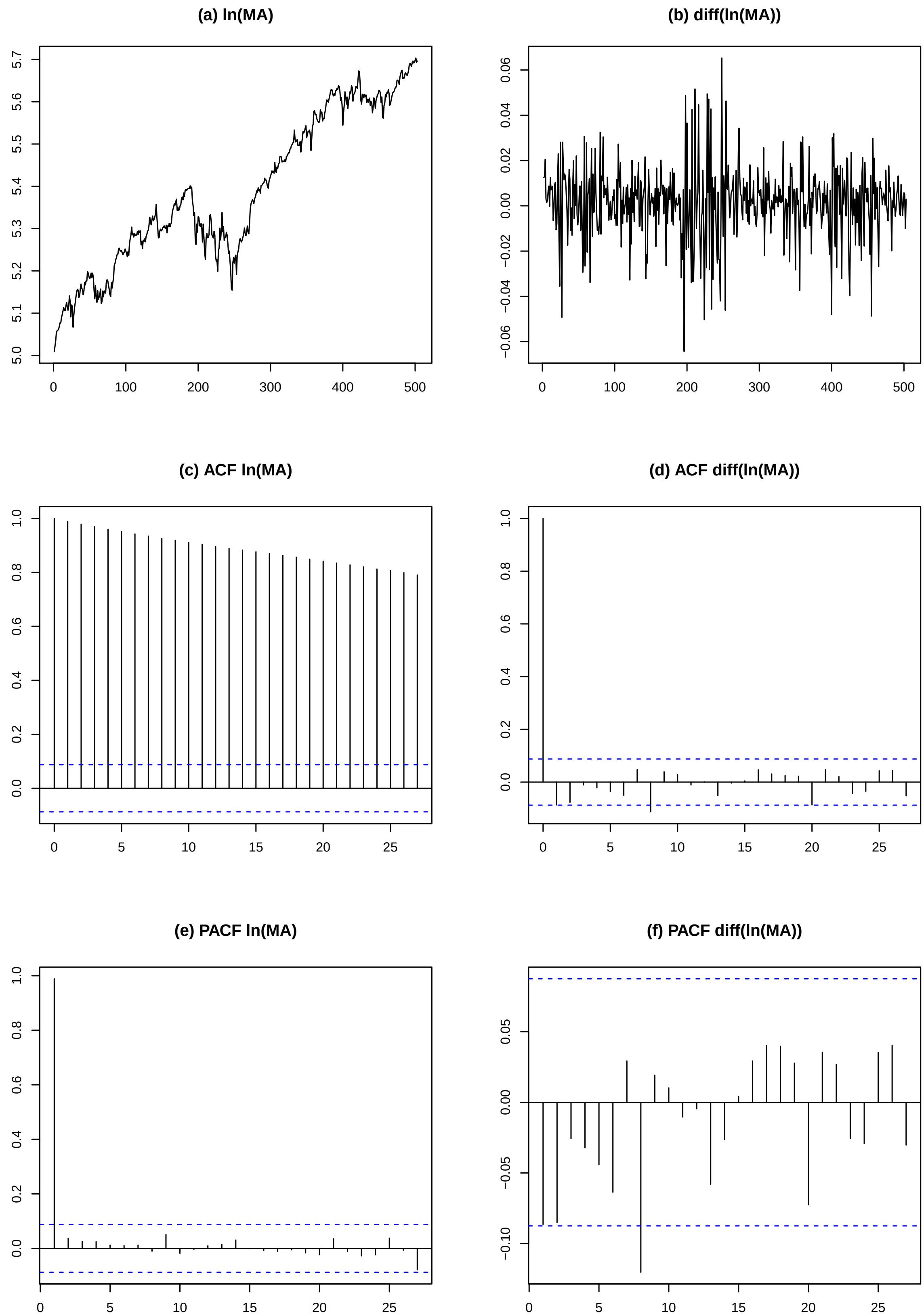


Figure A.4: Mastercard: log-price, ACF, PACF and differenced data.

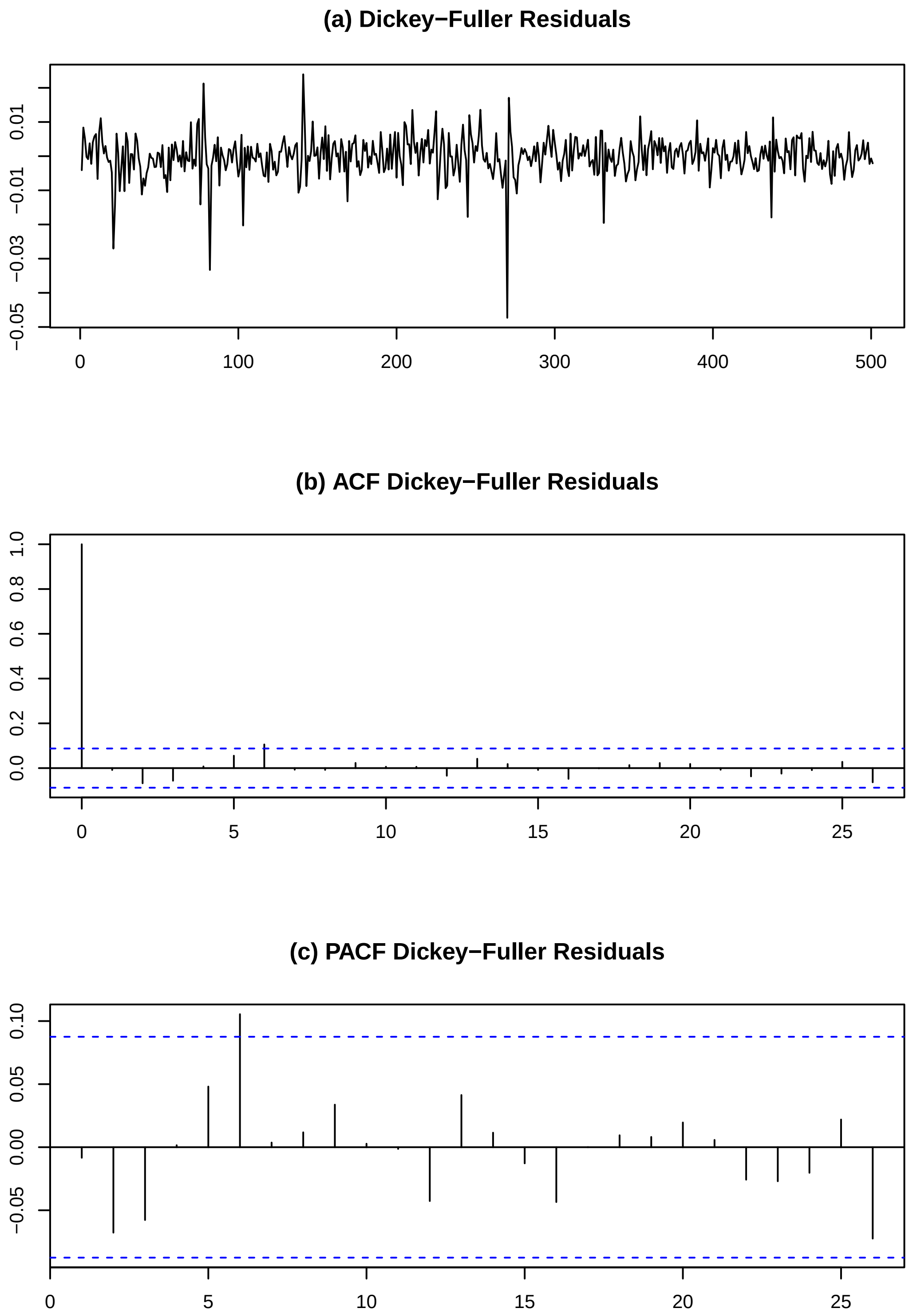


Figure A.5: Plot, ACF and PACT of Dickey-Fuller residuals $\{\epsilon_t\}$

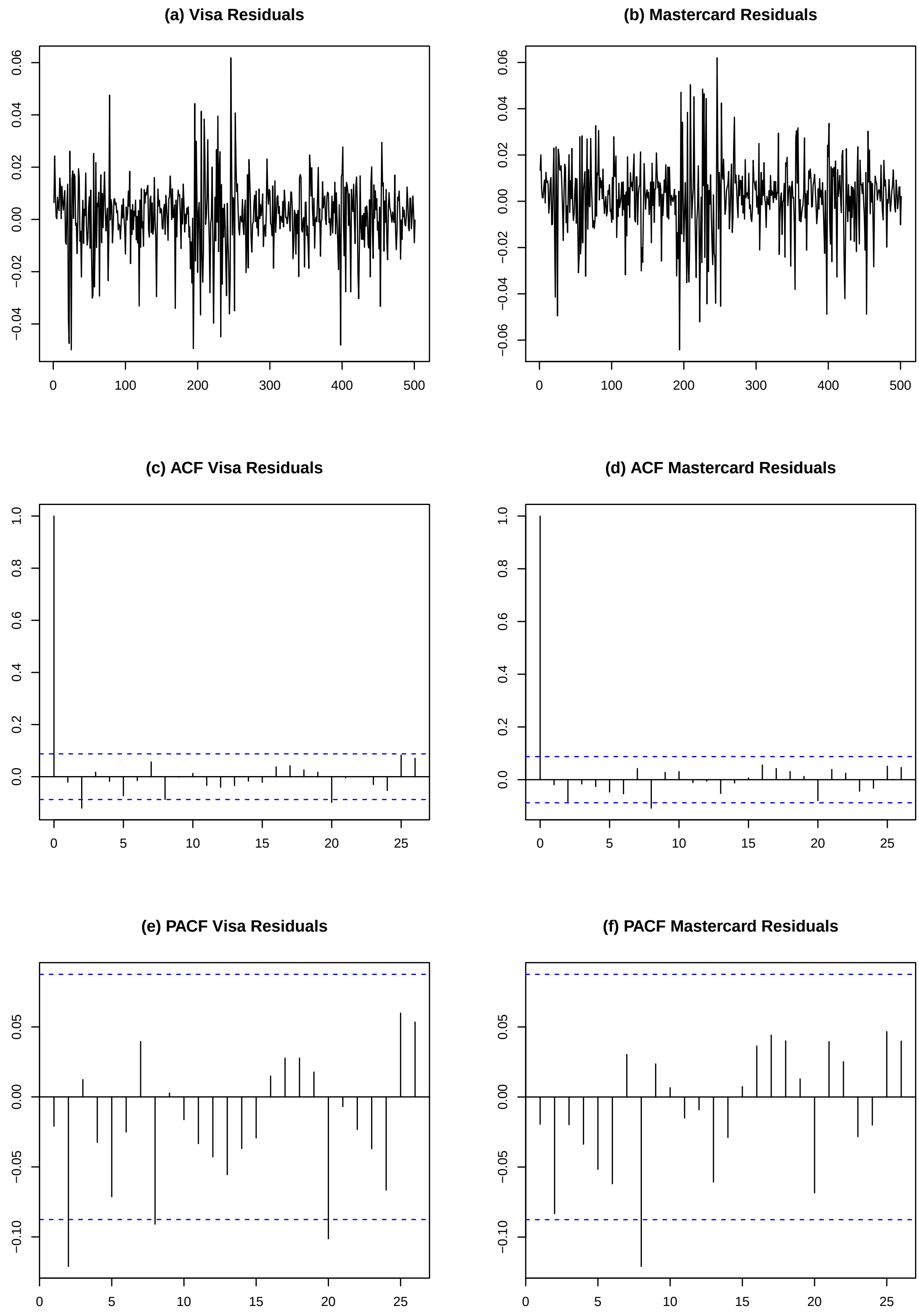


Figure A.6: Plot, ACF and PACT of residuals from the Engle-Granger ECM model

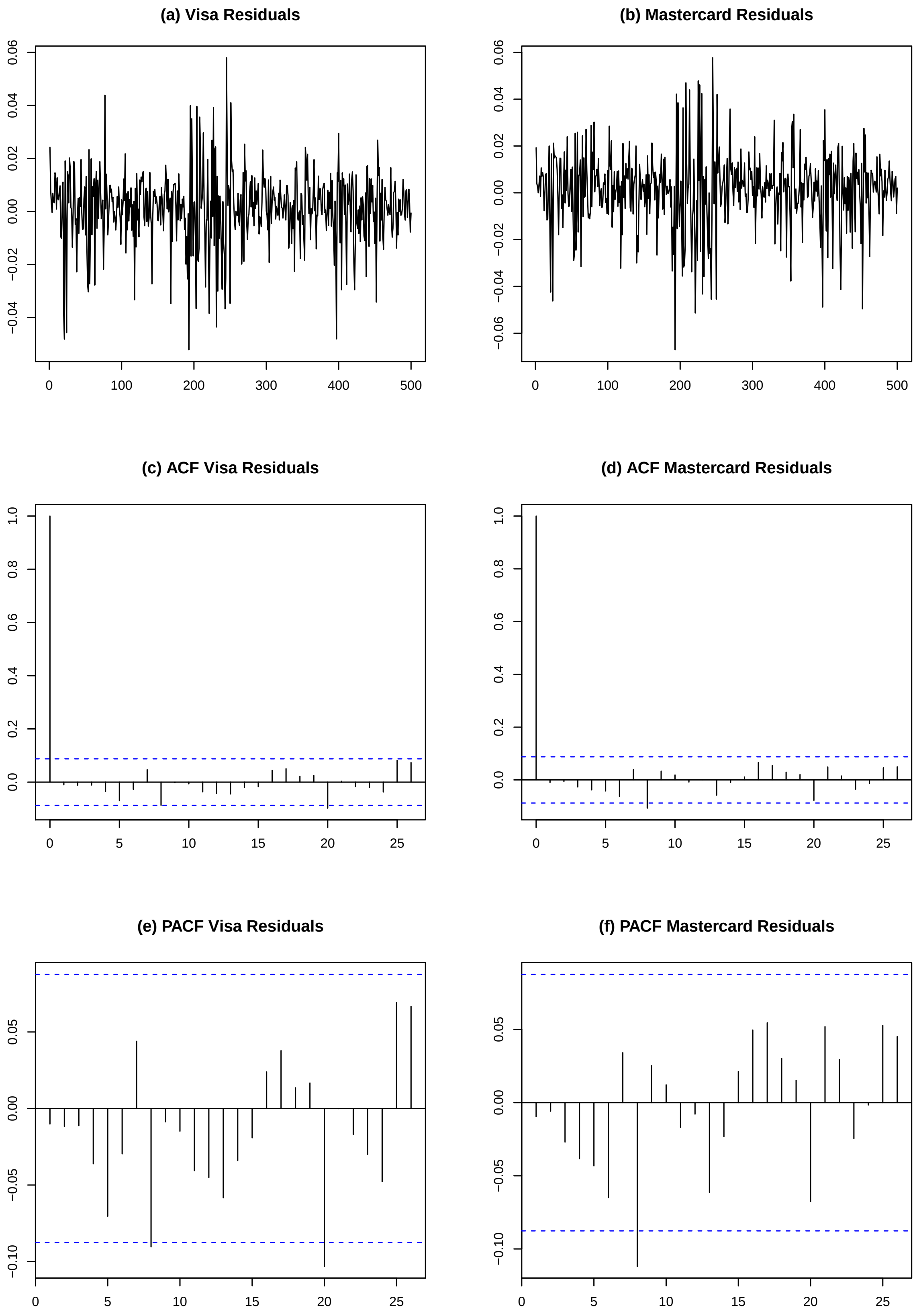
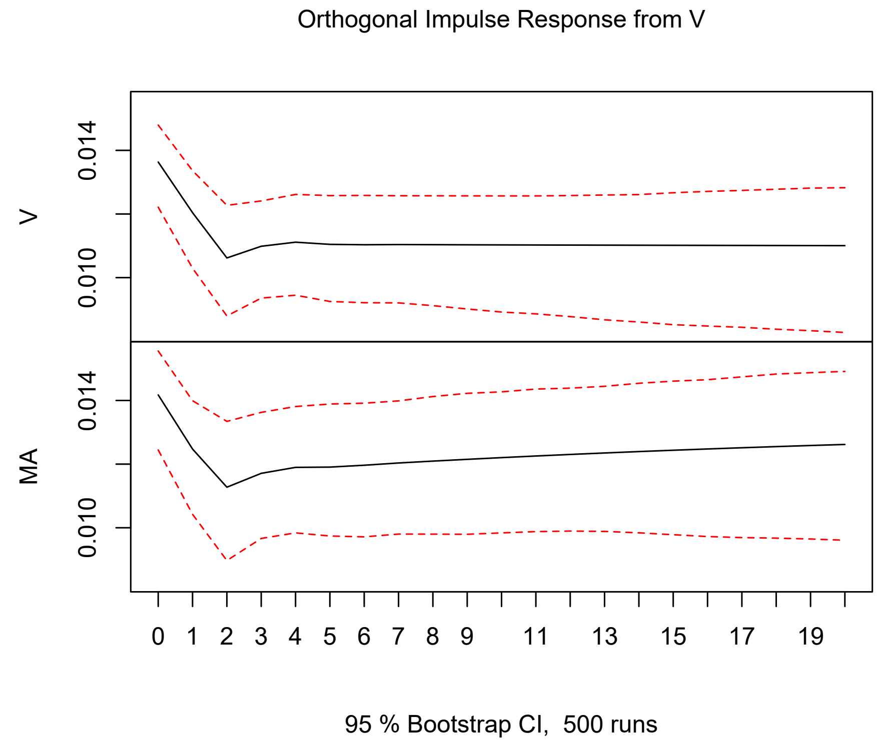


Figure A.7: Plot, ACF and PACT of residuals from the Johansen VECM model



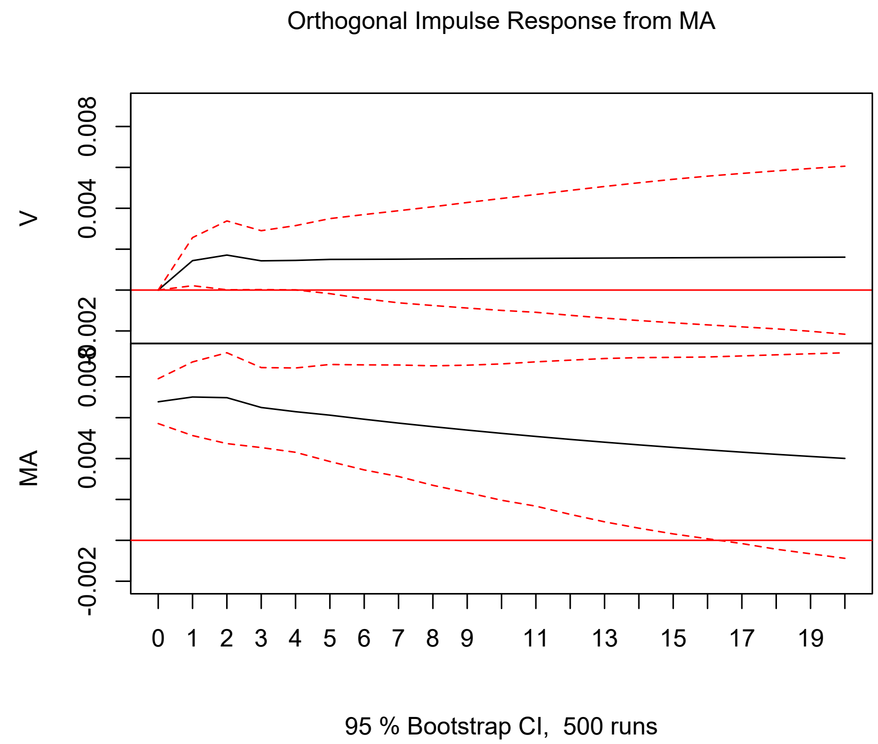


Figure A.8: Impulse Response Function using Cholesky decomposed matrices

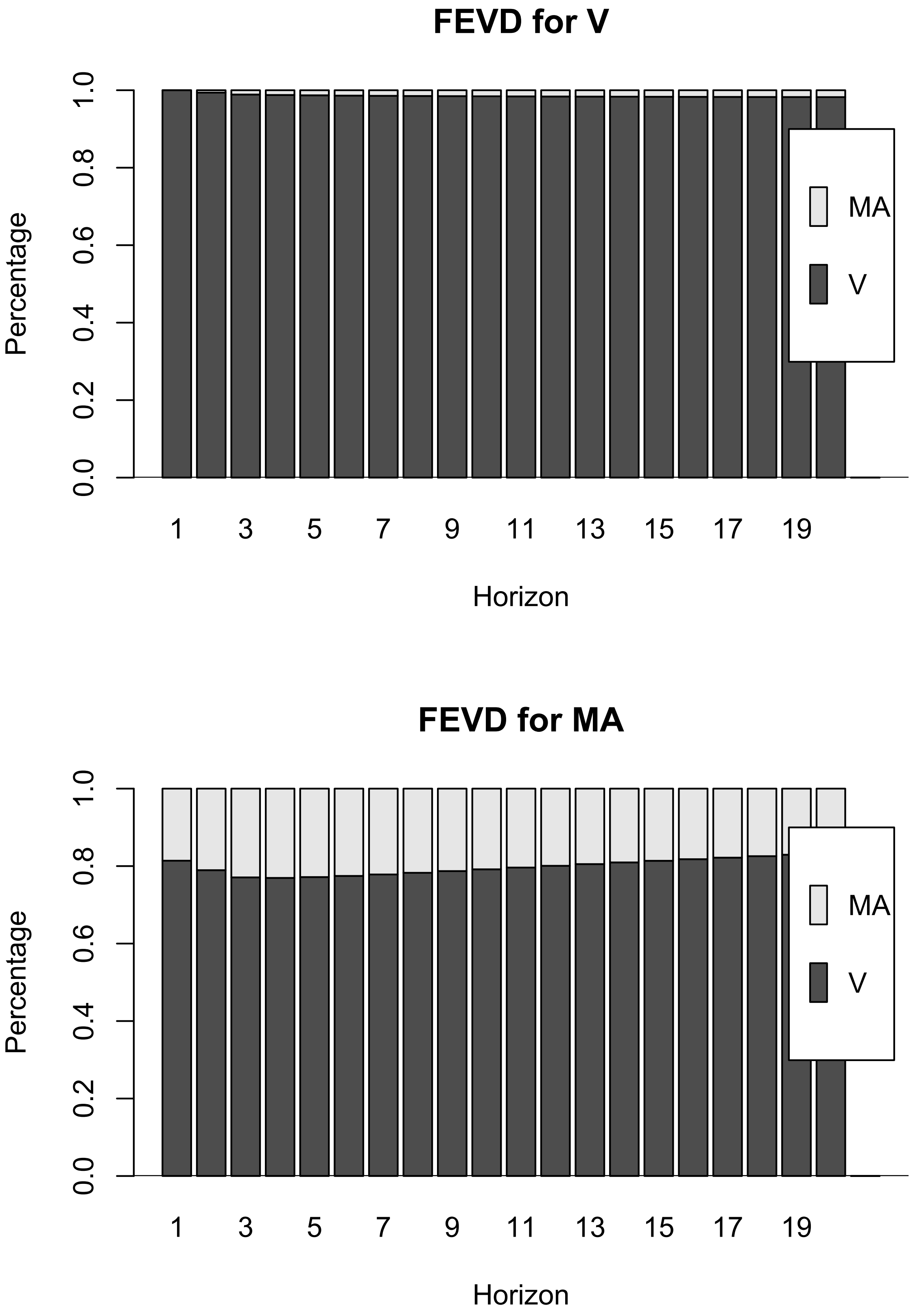


Figure A.9: Forecast Error Variance Decomposition

R Code

```r
#####################################################
## Financial Econometrics, cand.oecon, Final Exam ##
#######  Statistical Arbitrage Pairs Trading  ######
###############     May 20, 2020     ###############
#####################################################
# Clear the workspace
rm(list = ls())

# Load Packages
packages <- c("quantmod","astsa","xts","forecast","utils",
              "urca", "gets","dynlm","lubridate","ggplot2",
              "dplyr","tikzDevice", "astsa","xtable", "vars",
              "tseries", "tidyverse","tidyr","tsDyn")
lapply(packages, library, character.only = TRUE)
theme_set(theme_bw(base_size = 10)) # Set ggplot theme

# Load Data using Quantmod
stockpair <- c("V","MA")
dates <- as.Date(c("2018-01-01","2020-01-01"))
dfEnv <- new.env()
getSymbols(stockpair,
           env = dfEnv,
           verbose = TRUE,
           src = "yahoo",
           periodicity = "daily",
           from=dates[1],
           to=dates[2])
adj_df <- as.data.frame(do.call(merge, c(eapply(dfEnv,Ad))))
adj_df <- adj_df[, c(2, 1)] #Visa as first col

# Estimate new log-transformed variables and log-returns in dataframe
for (i in 0:3){
  if (i < 2){
    adj_df[,3+i] <- log(adj_df[,1+i])
    colnames(adj_df)[3+i] <- paste("log",colnames(adj_df)[1+i],sep = ".")
  } else{
    adj_df <- within(adj_df, diff <- c(NA, diff(adj_df[,1+i])))
    colnames(adj_df)[3+i] <- paste("diff",colnames(adj_df)[1+i],sep = ".")
  }
}

# Correlation of log-prices
cor(adj_df[,3], adj_df[,4]) # [1] 0.9933776
cor(na.omit(adj_df[,5]), na.omit(adj_df[,6])) # [1] 0.8982112

# Show the weekly price series
prices <- ggplot(adj_df, aes(x=as.Date(rownames(adj_df)))) +
  geom_line(aes(y=adj_df[,1]),col='black') +
  geom_line(aes(y=adj_df[,2]),col='blue') +
  ylab("") +
  xlab("") +
  theme(axis.title = element_blank(),
        panel.border = element_blank(),
        axis.line.x = element_line(color="black"),
        axis.line.y = element_line(color="black"))
prices

# Show weekly log-transformed prices
lnprices <- ggplot(adj_df, aes(x=as.Date(rownames(adj_df)))) +
```

```
  geom_line(aes(y=adj_df[,3]),col='black') +
  geom_line(aes(y=adj_df[,4]),col='blue') +
  ylab("log-prices") +
  xlab("") +
  theme(#axis.title = element_blank(),
        panel.border = element_blank(),
        axis.line.x = element_line(color="black"),
        axis.line.y = element_line(color="black"))
lnprices
#ggsave("lnprices.pdf", height = 2.3, width = 2.84)

#Show Scatterplot of log-prices
lnscatter <- ggplot(adj_df,
                    aes(x=adj_df[,3],
                        y=adj_df[,4])) +
                    geom_point(size=0.5) +
                    ylab("log(V)") +
                    xlab("log(MA)") +
                    theme(#axis.title = element_blank(),
                          panel.border = element_blank(),
                          axis.line.x = element_line(color="black"),
                          axis.line.y = element_line(color="black"))
lnscatter
#ggsave("scatterlnprices.pdf", height = 2.3, width = 2.84)


#####################################################
#  Testing Log-prices and returns for stationarity  #
#####################################################

# Visa: log-prices and first differenced log-prices (Series, ACF and PACF)
par(mfrow=c(3,2))
  plot(adj_df[,3], type='l', main ="(a) ln(V)", ylab="",xlab="")
  plot(adj_df[,5], type='l', main ="(b) diff(ln(V))", ylab="",xlab="")
  acf(na.omit(adj_df[,3]), col='black',main ="(c) ACF ln(V)", ylab="",xlab="")
  acf(na.omit(adj_df[,5]), col='black',main ="(d) ACF diff(ln(V))", ylab="",xlab="")
  pacf(na.omit(adj_df[,3]), col='black',main ="(e) PACF ln(V)", ylab="",xlab="")
  pacf(na.omit(adj_df[,5]), col='black',main ="(f) PACF diff(ln(V))", ylab="",xlab="")

# Mastercard: log-prices and first differenced log-prices (Series, ACF and PACF)
par(mfrow=c(3,2))
  plot(adj_df[,4], type='l', main ="(a) ln(MA)", ylab="",xlab="")
  plot(adj_df[,6], type='l', main ="(b) diff(ln(MA))", ylab="",xlab="")
  acf(na.omit(adj_df[,4]), col='black',main ="(c) ACF ln(MA)", ylab="",xlab="")
  acf(na.omit(adj_df[,6]), col='black',main ="(d) ACF diff(ln(MA))", ylab="",xlab="")
  pacf(na.omit(adj_df[,4]), col='black',main ="(e) PACF ln(MA)", ylab="",xlab="")
  pacf(na.omit(adj_df[,6]), col='black',main ="(f) PACF diff(ln(MA))", ylab="",xlab="")

### ADF test for non-stationarity/presence of unit roots
adf_table <- function(df, x){
  df <- NULL
  for (i in 1 : length(x)){
```

```
    out1 <- ur.df(na.omit(adj_df[,x[i]]), type = "none", lags = 0)
    out2 <- ur.df(na.omit(adj_df[,x[i]]), type = "drift", lags = 0)
    out3 <- ur.df(na.omit(adj_df[,x[i]]), type = "trend", lags = 0)
    df <- cbind(df,
                rbind(out1@teststat[1],
                      out1@cval[1,1],
                      out1@cval[1,2],
                      out1@cval[1,3],
                      out2@teststat[1],
                      out2@cval[1,1],
                      out2@cval[1,2],
                      out2@cval[1,3],
                      out3@teststat[1],
                      out3@cval[1,1],
                      out3@cval[1,2],
                      out3@cval[1,3]))
  }
  rownames(df) <- c(out1@model, "1%", "5%", "10%",
                    out2@model, "1%", "5%", "10%",
                    out3@model, "1%", "5%", "10%")
  colnames(df) <- colnames(adj_df[x])
  print(df)
  #xtable(df) # For LaTeX table
}
adf_table(adj_df, c(3,4,5,6))

### KPSS test for stationarity
kpss_table <- function(df, x){
  df <- NULL
  for (i in 1 : length(x)){
    out1 <- kpss.test(adj_df[,x[i]],null = "Level", lshort =TRUE)
    out2 <- kpss.test(adj_df[,x[i]],null = "Trend", lshort =TRUE)
    out3 <- kpss.test(adj_df[,x[i]],null = "Level", lshort =FALSE)
    out4 <- kpss.test(adj_df[,x[i]],null = "Trend", lshort =FALSE)
    df <- cbind(df,
                rbind(out1$statistic,
                      out1$p.value,
                      out1$parameter,
                      out2$statistic,
                      out2$p.value,
                      out2$parameter,
                      out3$statistic,
                      out3$p.value,
                      out3$parameter,
                      out4$statistic,
                      out4$p.value,
                      out4$parameter))
    }
  rownames(df) <- c(out1$method, "p-value","truncation parameter",
                    out2$method, "p-value","truncation parameter",
                    out3$method, "p-value","truncation parameter",
                    out4$method, "p-value","truncation parameter")
  colnames(df) <- colnames(adj_df[x])
  print(df)
  warnings()
  #xtable(df) # For LaTeX table
}
kpss_table(adj_df, c(3,4,5,6))
# Indicates that the prices are non-stationary and first-differences
# is stationary consistent with the results from the ADF test.

```

```
###################################################
#####          Tests for Co-integration         #####
###################################################

######
### Engle-Granger (1987) 4-step procedure (from Enders textbook)
######

# 1st step: Pre-test for order of integration --- Done both likely I(1)

# 2nd step: estimate long-run equilibrium
V <- adj_df$log.V.Adjusted
MA <- adj_df$log.MA.Adjusted
VAR.variables <- data.frame(cbind(V, MA),
                            row.names = rownames(adj_df))
lreg <-lm(V ~ MA, data=VAR.variables)
summary(lreg)

# Coefficient of log.V.Adjusted is ~1.23, fairly close to 1
VAR.variables$res <- lreg$residuals
adf.test <- ur.df(VAR.variables$res, type="none")
summary(adf.test)
# ADF test stat is -3.973 vs. 1% critical value of -3.92 (n>500 and 2
    variables) --> rejection of H0
adf.res <- adf.test@res
par(mfrow=c(3,1))
  plot(adf.res, type='l',main ="(a) Dickey-Fuller Residuals",
    ylab="",xlab="")
  acf(adf.res,main ="(b) ACF Dickey-Fuller Residuals", ylab="",xlab="")
  pacf(adf.res,main ="(c) PACF Dickey-Fuller Residuals", ylab="",xlab="")
# Residuals from the DF test looks like white noise.

# 3rd step: Estimate coefficients in error-correction representation
alpha <- coef(lreg)[1]
gamma <- coef(lreg)[2]
VAR.variables$spread <- (V - (gamma * MA)) # log(V_t)-gamma*log(MA_t)

# Spread is similar to calculated residuals plus the equilibrium mean.
par(mfrow=c(1,1)); plot(VAR.variables$spread, type ='l', main = "",
    ylab="", xlab="")

# Estimate model with constant in cointegration vector
EG.VECM <- VECM(VAR.variables[,1:2],
                lag = 1,
                estim="2OLS",
                LRinclude="const")
summary(EG.VECM)

# 4th step: model adequacy/specification
# Significance of coefficients
tstats <- (summary(EG.VECM)$coefficients)/(summary(EG.VECM)$StDev)
pval <- summary(EG.VECM)$Pvalues
print(rbind(tstats,pval))

# Residuals
EGres <- EG.VECM$residuals
par(mfrow=c(3,2))
  plot(EGres[,"V"], type='l', main ="(a) Visa Residuals", ylab="",xlab="")
  plot(EGres[,"MA"], type='l', main ="(b) Mastercard Residuals",
    ylab="",xlab="")
  acf(EGres[,"V"], col='black',main ="(c) ACF Visa Residuals",
```

```
  ylab="",xlab="")
  acf(EGres[,"MA"], col='black',main ="(d) ACF Mastercard Residuals",
    ylab="",xlab="")
  pacf(EGres[,"V"], col='black',main ="(e) PACF Visa Residuals",
    ylab="",xlab="")
  pacf(EGres[,"MA"], col='black',main ="(f) PACF Mastercard Residuals",
    ylab="",xlab="")
# The residuals of both models look like white-noise.

# 6 lag estimation
EG.VECM6 <- VECM(VAR.variables[,1:2],
                 lag = 5,
                 estim="2OLS",
                 LRinclude="const")
summary(EG.VECM6)
summary(EG.VECM6)$Pvalues # p-val of alpha_1 is 0.1199

######
### Johansen 4-step approach
######
# 1st step: Pre-test for order of integration and selecting lag lengths
lags <- VARselect(VAR.variables[,1:2],type="none")
print(lags)
VAR <- VAR(VAR.variables[,1:2],type="none")

# Save lag lengths in matrix
jo.p <- NULL
jo.p$AIC <- lags$selection["AIC(n)"] %>% as.integer # Save lag length
jo.p$BIC <- lags$selection["SC(n)"] %>% as.integer # Save lag length

# Build VAR to confirm that it is well-behaved using either 1 or 3 lags
lagtest <- function(df, x){
  df <- NULL
  par(mfrow=c(2,2))
  for (i in 1 : length(x)){
    VAR <- VAR(VAR.variables[,1:2],p = x[i],type="none")
    serial <- serial.test(VAR)
    arch <- arch.test(VAR)
    norm <-normality.test(VAR)
    df <- cbind(df,rbind(serial$serial$statistic,
                         arch$arch.mul$statistic,
                         norm$jb.mul$JB$statistic,
                         norm$jb.mul$Skewness$statistic,
                         norm$jb.mul$Kurtosis$statistic),
                   rbind(serial$serial$parameter,
                         arch$arch.mul$parameter,
                         norm$jb.mul$JB$parameter,
                         norm$jb.mul$Skewness$parameter,
                         norm$jb.mul$Kurtosis$parameter),
                   rbind(serial$serial$p.value,
                         arch$arch.mul$p.value,
                         norm$jb.mul$JB$p.value,
                         norm$jb.mul$Skewness$p.value,
                         norm$jb.mul$Kurtosis$p.value))
  pdf(file=paste("ResidualLag",x[i],".pdf",sep=""))
  par(mfrow=c(2,2))
    plot(VAR$varresult$V$residuals, main ="(a) Visa Residuals", type='l',
    ylab="",xlab="")
    plot(VAR$varresult$MA$residuals, main ="(b) Mastercard Residuals",
    type='l', ylab="",xlab="")
    acf(VAR$varresult$V$residuals, main ="(c) Visa Residuals (ACF)",
```

```
    ylab="",xlab="")
    acf(VAR$varresult$MA$residuals, main ="(d) Mastercard Residuals (ACF)",
    ylab="",xlab="")
  dev.off()
  }
  rownames(df) <- c(serial$serial$method,
                    arch$arch.mul$method,
                    norm$jb.mul$JB$method,
                    norm$jb.mul$Skewness$method,
                    norm$jb.mul$Kurtosis$method)
  colnames(df) <- c("Chi-squared","df","p-val","Chi-squared","df","p-val")
  print(df)
  #xtable(df) # For LaTeX table
}
lagtest(VAR.variables[,1:2], c(jo.p$BIC,jo.p$AIC))

# Plot AIC and (S)BIC
lagx <- as.data.frame(as.matrix(t(lags$criteria)))
lagx.plot <- ggplot(lagx, aes(x = as.integer(rownames(lagx)))) +
  geom_point(aes(y=lagx[,1]),col='black',shape=15) +
  geom_vline(xintercept = jo.p$AIC, linetype="dashed",col='black') +
  geom_point(aes(y=lagx[,3]),col ='blue', fill="blue",shape=24) +
  geom_vline(xintercept = jo.p$BIC, linetype="dashed",col='blue') +
  ylab("") +
  xlab("") +
  scale_x_continuous(breaks=seq(1:10)) +
  theme(axis.title=element_blank(),
        axis.text.y=element_blank(),
        panel.border = element_blank(),
        axis.line.x = element_line(color="black"),
        axis.line.y = element_line(color="black"))
lagx.plot
#ggsave("lagx.pdf", height = 1.25, width = 3.5)

# 2nd step: Model estimation and determine rank
jo.test.const <- ca.jo(VAR.variables[,1:2], type="trace",ecdet="const",
    K=jo.p$AIC, spec = "transitory")
jo.test.none <- ca.jo(VAR.variables[,1:2], type="trace",ecdet="none",
    K=jo.p$AIC, spec = "transitory")
jo.test.const.eigen <- ca.jo(VAR.variables[,1:2], type="eigen",
    ecdet="const", K=jo.p$AIC, spec = "transitory")
jo.test.none.eigen <- ca.jo(VAR.variables[,1:2], type="eigen",ecdet="none",
    K=jo.p$AIC, spec = "transitory")
summary(jo.test.const)
summary(jo.test.none)
summary(jo.test.const.eigen)
summary(jo.test.none.eigen)

# Check terms
round(jo.test.const@PI,3)       # Pi matrix
summary(jo.test.const@ZK)       # Lagged variables in Levels + 1
summary(jo.test.const@Z1)       # The regressor matrix, except for the
    lagged variables in levels.
round(jo.test.const@GAMMA,3)    # The coeffecient matrix of Z1
round(jo.test.const@lambda,3)   # The eigenvalue vector

# Johansen Test Results
JohansenTable1 <- NULL
JohansenTable1 <- rbind(cbind(jo.test.none@teststat[2],
                              jo.test.none@cval[2,2],
                              jo.test.none@cval[2,3],
```

```
                                jo.test.none.eigen@teststat[2],
                                jo.test.none.eigen@cval[2,2],
                                jo.test.none.eigen@cval[2,3]),
                          cbind(jo.test.none@teststat[1],
                                jo.test.none@cval[1,2],
                                jo.test.none@cval[1,3],
                                jo.test.none.eigen@teststat[1],
                                jo.test.none.eigen@cval[1,2],
                                jo.test.none.eigen@cval[1,3]),
                          cbind(jo.test.const@teststat[2],
                                jo.test.const@cval[2,2],
                                jo.test.const@cval[2,3],
                                jo.test.const.eigen@teststat[2],
                                jo.test.const.eigen@cval[2,2],
                                jo.test.const.eigen@cval[2,3]),
                          cbind(jo.test.const@teststat[1],
                                jo.test.const@cval[1,2],
                                jo.test.const@cval[1,3],
                                jo.test.const.eigen@teststat[1],
                                jo.test.const.eigen@cval[1,2],
                                jo.test.const.eigen@cval[1,3]))
rownames(JohansenTable1) <- c(jo.test.none@model,
                              jo.test.none@model,
                              jo.test.const@model,
                              jo.test.const@model)
colnames(JohansenTable1) <- c("Trace", "5%","1%",
                              "Eigen", "5%","1%")
xtable(JohansenTable1)

# 3rd step: Analyze the normalized cointegration vector and alpha coefficients
jo.VECM <- VECM(VAR.variables[,1:2],
                lag=(jo.p$AIC-1), # The VECM routine uses the lags in the VECM representation
                estim="ML",       # and a single lag should be subtracted.
                LRinclude="const")
summary(jo.VECM) # Summary of estimated model 1-lag with constant included in cointegration vector.
jo.VECM.rtest <- rank.test(jo.VECM)
summary(jo.VECM.rtest) # Rank test yields same results as ca.jo routine.
summary(jo.VECM)$Pvalues # P-values of VECM

# Residuals of Johansen VECM
JOres <- jo.VECM$residuals
par(mfrow=c(3,2))
  plot(JOres[,"V"], type='l', main ="(a) Visa Residuals", ylab="",xlab="")
  plot(JOres[,"MA"], type='l', main ="(b) Mastercard Residuals", ylab="",xlab="")
  acf(JOres[,"V"], col='black',main ="(c) ACF Visa Residuals", ylab="",xlab="")
  acf(JOres[,"MA"], col='black',main ="(d) ACF Mastercard Residuals", ylab="",xlab="")
  pacf(JOres[,"V"], col='black',main ="(e) PACF Visa Residuals", ylab="",xlab="")
  pacf(JOres[,"MA"], col='black',main ="(f) PACF Mastercard Residuals", ylab="",xlab="")
  # Or using urca-packages built in residual plot.
  plotres(jo.test.const)
# Add long-run error
beta <- as.matrix(jo.VECM$model.specific$beta)
x <- as.matrix(cbind(VAR.variables[,1:2],1))
```

```
error <- x%*%beta

reserror_df <- data.frame(merge(error, JOres, by="row.names", all=TRUE))
row.names(reserror_df) <- as.Date(format(reserror_df$Row.names), format =
    "%Y-%m-%d")

# Plot of Visa Residuals and long run errors from Johansen model
reserror <- ggplot(data = reserror_df,
    aes(x=as.Date(row.names(reserror_df)))) +
  geom_line(aes(y=MA), col='palevioletred') +
  geom_line(aes(y=V), col='red') +
  geom_line(aes(y=r1), col='black') +
  ylab("") +
  xlab("") +
  theme(#axis.title = element_blank(),
    panel.border = element_blank(),
    axis.line.x = element_line(color="black"),
    axis.line.y = element_line(color="black"))
reserror
#ggsave("reserror.pdf", dpi = 1000, height = 2, width = 5.9)


# Restriction testing
jo.test.const.beta <- jo.test.const@V[,1] # same as:
    jo.VECM$model.specific$beta
jo.test.const.alpha <- jo.test.const@W[,1] # same as:
    jo.VECM$coefficients[,1]
  # H0: Beta_0 = 0
    jo.const.restrict1 <- c(jo.test.const.beta[1],
                            jo.test.const.beta[2],
                            0)
    jo.const.test1 <- blrtest(z = jo.test.const, # Estimated Model
                              H = jo.const.restrict1, # Restricted Beta
                              r=1)
    summary(jo.const.test1)
    # p-value is 0 and we firmly reject that beta_0 = 0.
  # H0: Beta_2 = -1
    jo.const.restrict2 <- c(jo.test.const.beta[1],
                            -1,
                            jo.test.const.beta[3])
    jo.const.test2 <- blrtest(z = jo.test.const, # Estimated Model
                              H = jo.const.restrict2, # Restricted Beta
                              r=1)
    summary(jo.const.test2)
  # H0: Beta_0 = 0 and Beta_2 = -1 (Joint test)
    jo.const.restrict3 <- c(jo.test.const.beta[1],-1,0)
    jo.const.test3 <- blrtest(z = jo.test.const, # Estimated Model
                              H = jo.const.restrict3, # Restricted Beta
                              r=1)
    summary(jo.const.test3)
  # H0: alpha_1 = 0
    jo.const.restrict4 <- matrix(c(0,jo.test.const.alpha[2]), c(2,1))
    jo.const.test4 <- alrtest(z = jo.test.const, # Estimated Model
                              A = jo.const.restrict4, # Restricted Beta
                              r=1)
    summary(jo.const.test4)
  # H0: alpha_2 = 0
    jo.const.restrict5 <- matrix(c(jo.test.const.alpha[1],0), c(2,1))
    jo.const.test5 <- alrtest(z = jo.test.const, # Estimated Model
                              A = jo.const.restrict5, # Restricted Beta
                              r=1)
```

```
    summary(jo.const.test5)

# Johansen Restriction Test
JohansenTable2 <- NULL
JohansenTable2 <- rbind(cbind(jo.const.test1@teststat,
                              "2",jo.const.test1@pval[1]),
                        cbind(jo.const.test2@teststat,
                              "2",jo.const.test2@pval[1]),
                        cbind(jo.const.test3@teststat,
                              "2",jo.const.test3@pval[1]),
                        cbind(jo.const.test4@teststat,
                              "2",jo.const.test4@pval[1]),
                        cbind(jo.const.test4@teststat,
                              "2",jo.const.test5@pval[1]))
colnames(JohansenTable2) <- c("Chi-squared","df","pval")
rownames(JohansenTable2) <- c("Beta_0=0",
    "Beta_2=-1","Joint","alpha_1=0","alpha_2=0")
xtable(JohansenTable2)

# 4th step: Innovation accounting and causality test on error-corrected
    model
# Transform VEC to VAR with r = 1
VARx <- vec2var(jo.test.const, r = 1)

# Check matrices for inclusion of VAR model in paper
round(VARx$deterministic,3) # A_0
round(VARx$A$A1,3) # A_1
round(VARx$A$A2,3) # A_2
round(VARx$A$A3,3) # A_3

######################################################################
#  IMPULSE RESPONSE FUNCTIONS AND FORECAST ERROR VARIANCE COMPOSITION #
######################################################################

# Obtain IRF
irfx <- irf(VARx, n.ahead = 20, ortho = FALSE, runs = 250)

# Plot
plot(irfx)

# Obtain variance-covariance matrix
# Calculate summary statistics
summary(VARx)

# Obtain variance-covariance matrix
cor(VARx$resid) # Check correlation
covvar <- cov(VARx$resid) # Same as VARx$vecm@DELTA
covvar
t(chol(covvar)) # Choleski Decomposition matrix

# IRF plots using orthogonal matrices
plot(irf(VARx, impulse = "V", response=c("MA","V"), n.ahead = 20, ortho =
    TRUE, runs = 500, seed = 123))
plot(irf(VARx, impulse = "MA", response=c("MA","V"), n.ahead = 20, ortho =
    TRUE, runs = 500, seed = 123))

# Forecast Error Variance Decomposition
fevd <- fevd(VARx, n.ahead=20)
plot(fevd)
```