UNIVERSITÄT PASSAU
Wirtschaftswissenschaftliche Fakultät

Seminar Paper

im Studiengang Business Administration, M.Sc.

Universität Passau
Wirtschaftswissenschaftliche Fakultät

Financial Data Analytics
Prof. Dr. Ralf Kellner

Thema: Identify relevant variables with Granger causality methods

submitted by: Fabian Dick
Matr. Nr. : 104936
E-Mail: dick16@ads.uni-passau.de

submitted on: 11th February 2021

Supervisor: Jan König

Contents

List of Tables

List of Figures

Abstract

Thinking about a large set of functions which influence each other in various sizes and relationships the Granger causality test can help to detect these. This paper examines different approaches implemented in different libraries, in R and Python, to measure different kind of relationships and interactions linear and nonlinear and the sensitivities regarding e.g. the error term, approached significance levels or sample size. In order to gain full control about the relationship the data is simulated manually over a variation of different settings.

1 Introduction

Deerwester et al., 1990 Kanakaraj and Guddeti, 2015

Modeling multivariate time series in various fields such as finance, macroeconomic environments, or neuroscience Granger´s sight on causality (Granger, 1980) and especially it´s operationalization (Granger, 1969) became a popular tool in detecting causal links within a network. The main idea of Granger´s view on causality is that if a lagged version of timeseries_1, e.g. X, is significant in the sense of predicting timeseries_2, e.g. Y, then X is said to "Granger cause" Y (c. p.Granger, 1969). Since the term *causality* is under discussion in the literature it is important to distinguish between "Granger causing" and an actual casual relationship, as the idea of Granger causality is based on additional marginal prediction performance by including the lags of another variable, which is called more precise (Granger) temporal directed relation (c. p. Shang et al., 2020). Various authors showed this important difference by applying the Granger causality framework to the popular example of predicting the Christmas date, which is a fixed date, based on Christmas card sales, e.g. Studenmund, 2017. However, the Granger causality approach survives this criticism and is used with various adjustments throughout the literature in (multivariate) time series analysis in order to detect causal inference within networks, (c.p. Atukeren, 2007).

The literature on Granger causality can be divided into work which tries to optimize the inference in general, e.g., with machine learning, while other papers try to bring the idea of causality to different frequencies and distribution parts, such as quantiles or tails (c.p. Mazzarisi et al., 2020). However other literature tries to gain insights to various networks in different areas with the method. One recent topic in the literature of the improvement of the method is about the aim to connect the idea of Granger with the use of neural networks in order to infer more complex Granger causal links, such as non-linearity. Neural networks are in theory allowed at detecting Granger causality and making estimates about the significance of the found interrelationship, which is due to the broad definition of the method (c. p. Tank et al., 2021). Therefore, no parametric relations between the timeseries is assumed and the method is dedicated to focus on the conditional dependence of their distributions without violating theoretical assumptions of the underlying method. While some proposed models infer the Granger causality with a significance level, Tank et. al already use the process of neural networks in order to examine whether a variable is Granger caused by another according to the assigned weights within the network,(c. p. Tank et al., 2021). Tank et. al solve in their approach "Neural Granger causality" two common challenges in the use of neural networks at Granger causal link detection. One problem is the black box acting structure of neural networks

which make it difficult to interpret the results of the assumed inherent timeseries structural relationship in the data, according to the authors. Another complexity is the sharing of hidden layers across all observations *i*. Since in reality it is possible that not all timeseries depend always on the same lags of the other time series for every range of observations *i*. Basically, this is achieved by estimating component-wise models which turn the prediction model from the target x_t into each component of the matrix X. Therefore it is possible to determine the lag of each granger causal interaction which means, that for each i_{th} observation of x_{t_i} it is possible to detect the certain lag for the i_{th} observation of Y[1]. Applying neural network architecture leads to a model for each component of the tested effect variable. The advantage of this approach, so Tank et. al lies in the ability of detecting Granger causation in every i_{th} observation in the dataset. Therefore, the authors turn the specifically hierarchical lag selection problem from the linear the nonlinear approach. The lag selection problem is therefore solved by introducing a structured hierarchical lasso penalty which selects the lag of each interaction of the k equations. Applying this to the spectrum of neural networks, the penalty is applied to the component-wise (since every interaction is calculated specifically) artificial neural network. This is achieved by decomposing the weightmatrix W across all time lags p, which accounts for the lag dependency. In other words if the j_{th} column of W contains zeros for all p lags, then Y is invariant to X. Since the series does not influence the hidden unit. While in the linear setting the lasso penalty is put over the coefficient matrix, analogously to the VAR model the penalty in the neural network model is set to the columns of the weightmatrix W. This way the penalty shrinks the weights to zero for all observations. Further the authors describe types of different lasso penalties to detect hierarchical Granger causation and show how this method can also be applied to more complex architectures such as recurrent neural networks (RNN). The history of the literature in the field of Granger temporal directed relations shows that the possibility of recognizing non-linear relations did not only emerge with the successful application of machine learning in the last years, but that already other non-parametric models were successful without the use of neural networks. A famous model is developed by Baek and Brock in 1998(c. p. Baek and Brock, 1989). The model uses correlation integrals which can be seen as an estimator of spatial probabilities over time. [2]

Another method in detecting non-linear Granger causal links is applied by Zhu et al., 2021 by demonstrating via a vector autoregressive process and especially due to the inherent Granger causal relationship that investor attention, measured by the Google Search Volume Index, that shows the percentage of search volumes on certain keywords and is often used in the Granger causation literature in the finance area, see also S. Li et al., 2019 or Y. Li et al., 2021. [3]. The model for which the Granger causality relationship shall be examined consists then of the reduced vector autoregressive model with equations for the investor attention (Att_t), the Return (R_t) and the (realized) volatility V_t on a weekly basis by applying a chi2 test statistic, while the

[1] Please note that the authors use a p-dimensional matrix x_t and do not distinguish between X and Y. For the sake of consistency, the formulas or expressions are adjusted to be align with equations (5) and (6)

[2] The interested reader is advised to read the original paper "A Nonparametric test for independence of a multivariate time series" (1989) or the discussion of the approach in "Testing for Linear and Nonlinear Granger causality in the Stock Price-Volume Relation" (1994) from Hiemastr C. and Jones J.

[3] This can be simply downloaded by typing the keyword into: https://trends.google.com/trends

bivariate alternative hypothesis is constructed in the sense whether Att_t Granger causes R_t or V_t and vice versa. The results show that the variable Att_t had (positive) impacts on the crypto currency in return and also in volatility up to the 4th lag measured by the Akaike information criterion (AIC). However, a larger amount of lags seems to be insignificant for Bitcoin return and the timeseries inherent relationship of R_t and Att_t seems to be short term while Att_t has a longer lasting impact on V_t, according to the applied AIC. Therefore, the authors show also the importance of the right criterion of Granger causality detection and thus set out to determine the right inference criterion before the test is conducted, since the criterion is not only important for the detection in general but more specific when it comes to the correct lag size. Additional current investor attention is positively influenced by Bitcoin returns one week prior (R_{t-1}) while V_t shows invariant behavior towards Att_t. Further Zhu et. al represent a solution of the limitations of the classical Granger causality approach when it comes to the detection of the causation at higher moments by considering the squared and cubic terms of Att_t, which shows that Att_t^3 has a higher explanatory power as Att_t^2 and Att_t. The authors conclude that the attention of investors, approximated by google searches, is the Granger cause of Bitcoin market movements. More precise the shock from investors´ attention can last for several weeks in the market.

The remaining paper is structured as follows. Chapter 2 describes the method Granger causality in its classical understanding in a mathematical manner and demonstrates extensions from the literature. In chapter 3 selected packages for Granger causality in Python and R are tested for various scenarios, while the data is artificially created and manipulated for sensitivity analysis, linear and nonlinear.

2 Methods

2.1 Granger causality

As already mentioned, the Granger causality test is performed in order to examine whether a lagged version of X, is able to significantly increase the prediction performance of another variable Y, which would, if Granger causation is inherent, yield a network with uni- or multidirectional links. Knowing these relationships significantly increases the performance of various applications. For the bivariate case the Granger causality test can be performed by building two models, one which uses solely previous values of Y and one that uses precedent values of Y and X, as demonstrated in equations (1) and (2) (c. p. Hmamouche, 2020):

$$Y_t = \alpha_0 + \sum_{i=-p}^{p} \alpha_i Y_{t-i} + \varepsilon_t, \tag{1}$$

$$Y_t = \alpha_0 + \sum_{i=-p}^{p} \alpha_i Y_{t-i} + \sum_{i=-p}^{p} \beta_i X_{t-i} + \varepsilon_t \tag{2}$$

While ε_t is the residual term of each equation which is constrained to the underlying assumptions of the model used to detect these relationships and p is the lag parameter while α_p and β_p represent the coefficients. Typically, the residuals capture the noise of the variables included in the model, but it could be also caused by variables which affect, in Granger sense or on the same lag level, the target variable Y. Therefore, the Granger causality test is often based on the residuals or its moments, as explained in the following.

Hmamouche describes a common measure for (Granger) causal detection, the Granger causality index (GCI), which is the logarithmised relation between the variances of the error term of $Model_1$ and $Model_2$, (c. p. Hmamouche, 2020):

$$GCI = log(\frac{\sigma_1^2}{\sigma_2^2}) \tag{3}$$

As demonstrated by Shang et al. it is also possible to derive the additional prediction performance based on a variance decomposition formula which represents the explained variance X by Y, (c. p. Shang et al., 2020) and results in the generalized measure of correlation (GMC). The advantage of the GMC is its ability to measure also nonlinear interactions due to the more generalized procedure since the inference of Granger causal links is model free (c. p. Tank et al., 2021) while the GCI needs two models to derive the variance of ε_1 and ε_2.

The GMC can be expressed as follows:

$$GMC(Y|X) = 1 - \frac{E[var(Y|X)]}{var(Y)} \tag{4}$$

In the case of detecting not only if one time series is significant to forecast another but more precise in which lag order, as the Granger causality test allows, the GMC can be turned into the measure of a (number of) lag(s) dependent measure for a pair of bivariate scalar time series called the generalized measure of correlation (AGMC), such that:

$$AGMC_k(Y_t|X_t) = GMC(Y_t|X_t - k), \tag{5}$$

Where k is the lagparameter with $\in (0, n]$ and n denotes the number of observations. From there on different adjustments are made within the literature, such as the Granger causality generalized measures of correlation (GcGMC) by Zheng et al., 2012, which also considers the cross- and autocorrelation.

2.2 Construction of linked networks via vector autoregressive models

Since the Granger causality test is also able to detect relationships in various directions such as X granger causing Y as shown in equation (2) or the opposite direction or both, a multidirectional linked network can be used for simulating data for test inference purposes. In the case that both variables or, more general, all k functions influence each other, vector autoregressive models (VAR) are a popular tool in order to model complex interacting processes (c. p. Stock and Watson, 2001):

$$Y_t = \alpha_0 + \sum_{i=-p}^{p} \alpha_i Y_{t-i} + \sum_{i=-p}^{p} \beta_i X_{t-i} + \varepsilon_t \tag{6}$$

$$X_t = \alpha_0 + \sum_{i=-p}^{p} \alpha_i Y_{t-i} + \sum_{i=-p}^{p} \beta_i X_{t-i} + \varepsilon_t \tag{7}$$

Within the model the time series effect each other, while the Granger causal effect is quantified by the coefficients, (c. p. Tank et al., 2021). A VAR(k) process of the lagorder p creates an environment of differently - different to if and the size of the relationship - interacting functions. While the bidirectional case X granger causing Y and Y granger causing X, shown in equations (5) and (6), is an example of a bivariate vector autoregressive process. The number of functions can be represented by k and can be, in theory, infinitely large with individually causal links to the remaining equations. This kind of network construction serves as the basis for the data simulation and is adjusted for the coefficients, the interactions, and the general directions.

3 Test inference of Granger causality detection implemenations

3.1 Scope

While there is a brought source of different packages, libraries tests and methods online, the tests are selected for the sake of comparability. Further, except the classical implementations, the libraries include functions which make are in theory able to detect nonlinear causal links between simulated time series. The tests are departed in the ability of inferring bi- and multidirectional (granger) causation and are listed in table 3.1. Deriving from the fact that the packages use to a certain extent different tests to infer causal links it must be made an important consideration of the null hypothesis for the sake of comparability. The underlying hypothesis within this paper can be expressed as follows:

$$H_0 = \beta_{lag_1}, \vee \beta_{lag_2}, ..., \vee \beta_{lag_p} = 0$$

$$H_1 = \beta_{lag_1}, \wedge \beta_{lag_2}, ..., \wedge \beta_{lag_p} \neq 0$$

While p is the maximum amount of lags tested, often refered to as "maxLag" or "maxlag" in the packages and β_{lag_p} is the coefficient of maxlag. As can be seen from the Hypothesis the underlying methods test for a joint number of lags rather then every lag on its own.

3.2 Packages

The analyzed packages are taken literature backed, regarding that they have the same output regarding a p value and the given value of the test statistic. Also, the tests are analyzed in the sense of including the right number of lags / the right lag into a artifical model of whatever kind. Therefore, the relationship simulated in the data is also lag dependent and the tests are examined

Library	Function	Documentation	Programming Language
VLTimeSeries	GrangerFunc	cran.r-project.org	R
lmtest	grangertest	rdocumentation.org	R
NlinTS	causality.test	cran.r-project.org	R
NlinTS	nlin_ causality.test	cran.r-project.org	R
bruceR	granger_causality	cran.r-project.org	R
statsmodels	grangercausalitytests	statsmodels.org	Python
statsmodels	test_causality	statsmodels.org	Python
Nonlincausality	nonlincausality_NN	github.com	Python
Nonlincausality	nonlincausality_GRU	github.com	Python
Nonlincausality	nonlincausality_LSTM	github.com	Python

Table 3.1: Library Scope

whether they are able to detect the certain lag(s) or not. The selection of the packages [1] can be found in table 3.1 while granger_causality and test_causality are for multivariate analysis and the remaining functions can be used for bivariate Granger causal detection inference. Since the focus of this paper is on bivariate interdependencies between timeseries, these packages are explained in more detail in the following.

3.2.1 statsmodels, grangertest and grangercausalitytest

The functions grangertest, grangercausalitytests and causality.test perform an OLS approach in order to examine whether the sum of squared residuals decreases by adding precedent values of e.g. X in a VAR(2) model for e.g. Y. To derive whether the increase in the prediction performance is significant a simple Wald test is applied in the form of equation 8.

$$F = \frac{(RSS_1 - RSS_2)/p}{RSS_2/(n - 2p - 1)} \tag{8}$$

While RSS_1 and RSS_2 are the residual sum of square (RSS) of $Model_1$ and $Model_2$ in the bivariate setting explained in equation (1) and (2). This implementations can be seen as the *classical* Granger causality approach and represent the benchmark models within this seminar paper.

[1] It has to be mentioned that the function "grangertest", "causality.test" and "grangercausalitytests" perform the same approach and can therefore be seen as testing the implementations in R and Python against each other rather than comparing the tests themself. When combined the tests are referenced together as "OLS approach".

3.2.2 VLTimeCausality

The library Variable-Lag Time Series Causality Inference Framework (VLTimeCausality), introduced in May 2021, translates the component wise modelling approach similar to the neural network approach from Tank et. al into a more parametric setting. The framework offers functions which are intended to detect variable lag dependence, variable in the sense of changing causal links over time, called *VLGrangerFunc*. While this type of test would require a different data simulation as it is drawn for the other tests, another test from the framework is used. The function used in the first empirical analysis from the package is *GrangerFunc* which uses, different to the classical approaches, a GLM model in order to compare the marginal additional prediction accuracy [2].

3.2.3 Nonlincausality

The package nonlincausality is a extensive library which includes eight different functions in total. While three of them represent a neural network approach of the classical granger implementation with different types of neural architectures, the package also includes functions which measure causality over time, similar to the so-called variable lag Granger causality from VLTimeSeries. The three functions which perform a neural network approach to detect Granger causality differ, as already mentioned within their network architecture in the sense of a Forward Neural Network (FNN) and two Recurrent Neural Networks (RNN) a gated recurrent unit (GRU)and a long short-term memory (LSTM) approach. For each tested lag the functions create two neural networks. The first one is forecasting the present value of Y based on past values of X defined by the variable *maxlag*, while the second model is forecasting the same value based on n=current lag past values of X and Y time series. This yields basically a neural network framework for the equations (1) and (2). If the prediction error of $Model_2$ is statistically significantly smaller than the error of $Model_1$ than it means that Y is Granger caused by X[3]. The method used in order to judge the significance about the detected relationship is made via a *Wilcoxon Test*, which has the benefit that the significance can also be derived from non-parametric models. The Wilcoxon test can therefore be seen as the nonparametric counterpart of a t-test. In particular, it tests whether the distribution of the differences is symmetric around zero (c. p. Rey and Neuhäuser, 2011). The creators of the packages use the absolute errors of the two different equations as can be seen in the source code [4]. Using this test the authors may allow to solve the tradeoff between performance and interpretability of neural networks to a certain extend, as addressed by Tank et al., 2021.

[2] Please note that the default settings of the function set the parameter "autoLagflag" to true which activates the automatic lag inference function. However the function behind determines the tested lag according to cross-correlation, regardless of what is used as "maxLag" by the user. The function then returns a negative value for the suggested lag which is not incorrect but the function GrangerFunc takes the maximum of 1 and the suggested lag. Since the suggested lag is negative, GrangerFunc always uses lag 1 if "autoLagflag" is set to true. The source code can be viewed on Github: https://github.com/DarkEyes/VLTimeSeriesCausality/blob/master/R/granger.R

[3] The functions in the source code are called "adj" since the version in the package uses an outdated keras version

[4] github.com/mrosol/Nonlincausality/blob/master/nonlincausality/nonlincausality.py

3.2.4 NlinTS

The NlinTS package performs a similar approach as nonlincausality in the sense that it also uses a neural network for Granger causality detection. The authors suggest to use a VARNN model. The VARNN (p) model is a multi-layer perceptron neural network model that considers the p previous values of the predictor variables and the target variable (Y) in order to predict future values of Y. The model reorganizes the data in a form of a supervised learning form with respect to the lag parameter. While the functions in nonlincausality use the Adam optimizer from tensorflow/keras the optimization algorithm in NlinTS is based on the Stochastic Gradient Descent (SGD) algorithm. Compared to the classical test the difference is that the Fisher test, used in statsmodels etc., is changed due to the higher amount of parameters in the VARNN model as in the VAR model. The adjustment is made by including d_1 and d_2 which represent the number of parameters θ of both models and are dependent on the number of layers and neurons.

$$F = \frac{(RSS_1 - RSS_2)/a}{RSS_2/b}, \tag{9}$$

with a = $\theta_{Model_{NN_2}}$ - $\theta_{Model_{NN_1}}$ and b = n - $\theta_{Model_{NN_2}}$.

While $Model_{NN_1}$ and $Model_{NN_1}$ are again the neural network form of equation (1) and (2) and n being the number of observations.

3.3 Causal network and testing environment construction

The following section presents the outlined strategy and data construction of the empirical analysis in chapter 4. The data is constructed such that the lags of one series have a effect on the future of each other series to a controlled extend and the magnitude of the coefficients quantifies the Granger causal effect. For the sake of comparability, it is important to have full control over the structure of the interacting time series, but also the extent of the interactions in the sense of the lag(s), as compromised timing of interactions may disrupt network functions and distorts test results, especially the classical approaches, that use OLS.

While the neural networks, such as the NlinTS package, implemented test methods allow for a way weaker assumption set, the classical Granger causality test uses OLS regression in order to examine causal inference and is therefore bound to its assumptions, such as stationary data, and the distribution of the error term in the sense of E[ε] = $E(\varepsilon|X)$ and V(ε) = $V(\varepsilon|X) = {\sigma^2}_\varepsilon$ * I for all k equations. To ensure comparability these assumptions are strictly retained at the beginning but loosened to some extent for sensitivity analysis. Regarding that from the assumptions arises that ε_i and x_i are independent, this might be violated if there is e.g. a third variable Z that affects the variables X and Y, which could be in theory inferred by the Granger causality test.

The data generating process (DGP) is based on a vector autoregressive model as explained in chapter 1 with k different functions (VAR(k)) to be able to measure the underlying Granger causality in various directions bi- and multivariate. The DGP will be explained in detail in the following. A fixed amount of samples n is drawn from the process of a multivariate normal distribution representing the error term. The VAR Process is then created with coefficients for the lag parameters p for every simulation step. The different coefficients for one simulation step are captured in the matrix L. All coefficient matrices are then combined within a tensor T for each lag. During the simulation it is iterated over the different coefficient matrices within **T** by multiplying L with the *kth* observation of time series S. Then an error term is added which is drawn from a multivariate normal distribution with a variance-covariance matrix Σ, while the cross-correlation between each ε_i can be controlled over a exogenous variable. The outcome of this process are k timeseries which are dependent on each other according to the selected coefficient development (e.g., steady, decreasing or nonlinear). In the bivariate case, which constitutes a large share over all tests performed, such as the first time series is tested against the second and vice versa. This way it is possible to derive the test behavior within one VAR Process to analyze where the different results are originated for the same test. For sensitivity analysis adjustments were made to the coefficient tensor T and to the error term dynamics within Σ. The construction for an exemplary bivariate tensor for one lag can be seen in figure 3.1. Each matrix consists of the dimension (k x k) and each (lag-) tensor consists of n matrices.

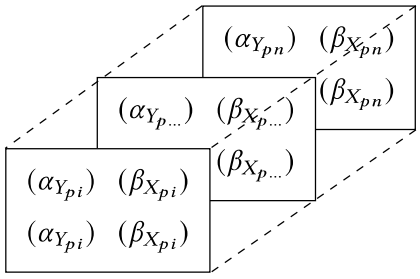

Figure 3.1: Tensor construction for bivariate timeseries simulation

In order to be able to compare the different tests implemented in the packages different already existing instruments are used or constructed by modifying them for the needed purpose. The instruments are mainly constructed out of tables which capture the p-values, and the test statistic values per tensor and per lag. The tables can be reconstructed within the source code. Also, every instrument can be reproduced for every case in the source code but are only shown in the paper when it is appropriate.

Confusion Matrices

Confusion matrices are popular in the case of classification problems and aggregate the results in a table such as in table 3.2. However, in the area of test inference confusion matrices can be used by transforming the idea of classification to the underlying significance level. Then the True Positive rate can be described as the number of TPs corresponding to the p value belonging to the respective lag which is in indeed significant. The remaining table contents can

be described accordingly. Then the confusion matrix at a certain significance level can be seen as:

		Inferred causal links: Positive	Inferred causal links: Negative	Total
True causal links	Positive	a	b	$a+b$
	Negative	c	d	$c+d$
	Total	$a+c$	$b+d$	N

Table 3.2: Confusion matrix

Receiver Operating Characteristic (ROC)

Typically, the ROC measure is applied whenever a certain probability threshold P can be selected ex ante. In the case of test inference the ROC-approach is adjusted in the way such that P is changed to the significance level α which varies between 0 and 1[5]. This way it is possible to see the true positive (TP) versus the false positive (FP) rates at different significance levels rather then only look at the total of the truly recognized lags for one α. Furthermore, it is possible to see what marginal increase or decrease in TPR and FPR one must take into account when increasing or decreasing the significance level.

3.4 Test Inference

Based on the same DGP the testing strategy is further departed in two different approaches. The first approach *Overall Inference Power* tests the different packages for a general setting but with metaparameters varying over the simulations. This setting tests the overall coefficient, sample size and significance level sensitivity of the different implementations to detect the dependence of one given lag, which is lag 3 in the first simulation. This way it is possible to derive sensitivities of metaparameters of the simulation such as the sample size, the inherent coefficient range in the data and the significance level α. The Meta Simulation is performed with a VAR(2) process while only one time direction is presented in the results in the sense of X Granger causes Y or in other words that Y is not invariant to X.

The second type of tests extend the underlying assumptions of the meta simulation by varying the lag dependence, coefficient development and error term. The testing approach of the ability of detecting the certain lags, rather than only the general result if the variable is caused follows directly from the assumption that the used packages are used for modeling with targeted lag variables.

[5] Significance levels beyond 0.1 are of course inappropriate but it allows the ROC curves to have the same scale for all tests

3.4.1 Linear relationships

3.4.1.1 Granger causality inference I - Meta simulation

The following results are obtained by a simulation based on a VAR(2) process with assumed dependencies on the own third lag and on the third lag of the causing time series. The granger causal link β was thereby increased per iteration by 0.1 from -0.5 to 0.5, while the remaining coefficients were held steady over the simulation. Then the coefficient-matrix for the third lag can be seen as:

$$L_3 = \begin{bmatrix} -0.3 & \beta \\ 0.03 & 0.4 \end{bmatrix}$$

Then the equation system is defined by:

$$Y_t = -0.3 * Y_{t-3} + \beta X_{t-3} + \varepsilon_t \tag{10}$$

$$X_t = 0.4 * X_{t-3} + 0.03 * Y_{t-3} + \varepsilon_t \tag{11}$$

While the error terms for each iteration are drawn form a multivariate normal distribution with a mean of 0, a standard deviation of 0.5 and without cross correlation.

Each coefficient in the range of [-0.5, 0.5] was then processed in the equation system individually and the respective p value for the third lag from the test was obtained. If the p value was below the significance level α then a 1 was assigned for that iteration and 0 otherwise. This process was repeated ten times for every coefficient β, each sample size (750, 1,000, 10,000) and for the significance levels (0.01, 0.05 , 0.1). The binary results were then aggregated in a result matrix of the dimension (10 x 11) per sample size and significance level combination, while the number of rows represent the simulation iterations, and the columns represent the coefficient range. Then for each column/each coefficient the average was calculated to derive the probability of rejecting the null hypothesis (P(reject H_0)). The results for the significance level of 0.05 are presented by the power curves in figure 3.2 [6].

[6]The meta simulation for α = 0.01 and α = 0.1 can be reconstructed within the source code

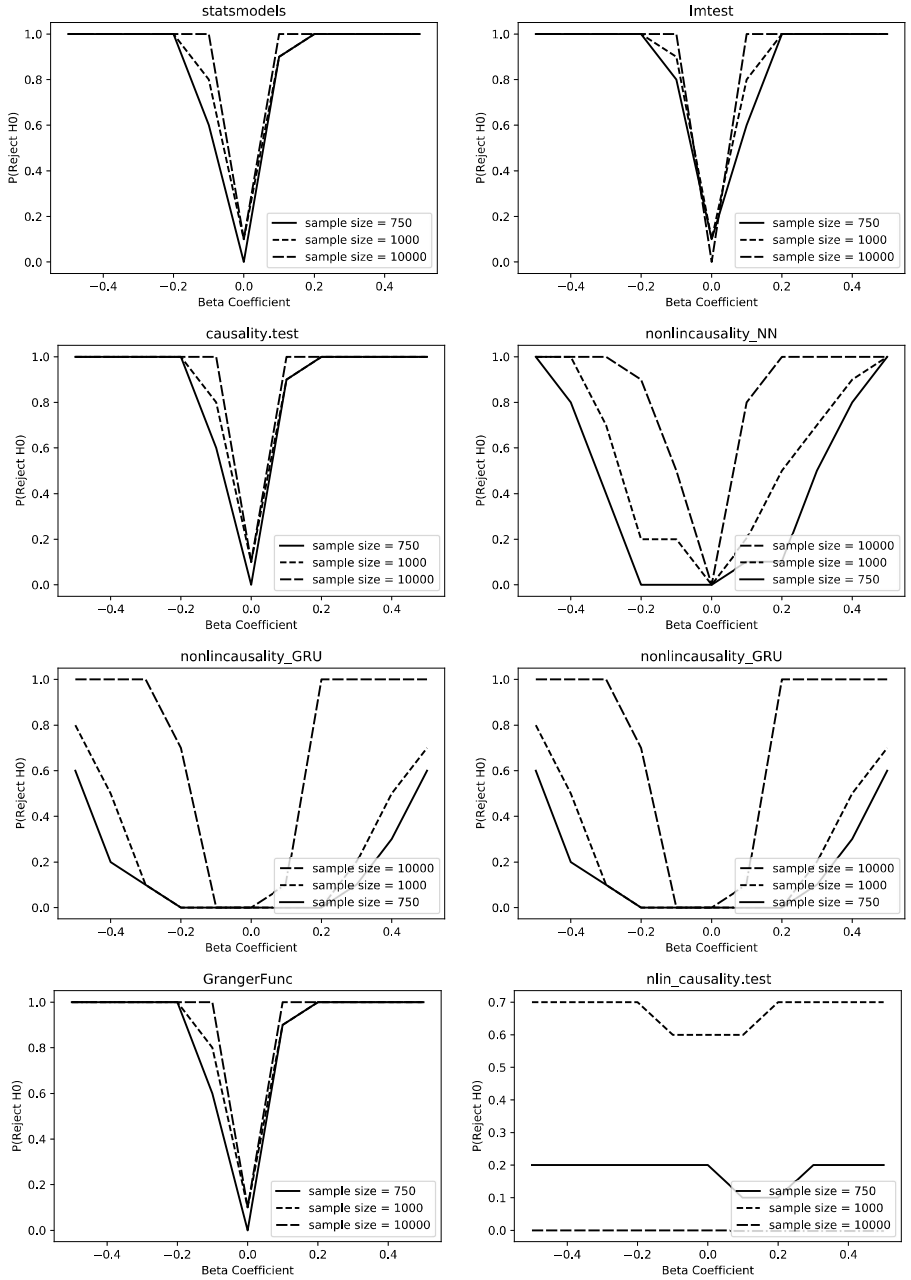

Figure 3.2: Meta Simulation at a significance level of 0.05

The meta simulation shows that all implementations, except *nlin.test*, have a higher probability of correctly rejecting the null hypothesis with a increasing sample size. On the other hand, some tests exhibit also a higher probability of falsely rejecting the null hypothesis at higher sample sizes, e.g. *grangercausalitytests* or *nonlincausality_NN*. The RNN approaches of nonlincausality, *nonlincausality_GRU* and *nonlincausality_LSTM*, perform poorly and the results

reveal that a large sample size is needed in order to infer granger causal links efficiently, the latter also applies to *nonlincausality_NN*. This might also indicate that the implementations may perform better under rather more complex scenarios. The classical approach exhibits the same shape for all sample sizes, such as *lmtest* and *grangercausalitytests*. Additionaly the classical approach and *nonlincausality_NN* converge with increasing sample size. Regarding the coefficient range most tests perform well with values higher then | 0.2 | and a given sample size of 10,000 but show different results for coefficient ranges around zero. While most curves are symmetric, *nonlincausality_LSTM* and *nonlincausality_GRU* seem to have to some extent more problems in detecting small negative coefficients, as indicated by the break in the negative area. Similar observations can be made at *nonlincausality_NN* for smaller positive coefficients. *nlin_causality.test* from the package NlinTS seems to have problems in detecting the causal links. For one lag case it shows either a high rejection probability for a small sample size or no rejection probability at higher sample sizes. This indicates that a rejection from *nlin_causality.test* could only occure because only one lag is significant or because the sample size is not appropriate and therefore other tests should also be applied rather then relying on the results alone.

The result of the meta simulation demonstrates that all tests are in general able to detect causal links in the sense of Granger but show different performances. In order to analyze the sensitivity for various scenarios the test strategy will be from now on extended and discussed in more detail. This way it is possible not only to examine the inference for one lag and with only one significance level but for multiple lags with individual coefficients and different results for different significance levels.

3.4.1.2 Granger causality inference II - error term sensitivity

Granger causality Inference II (GC II) describes a test scenario in which the coefficients are fixed per lag over time, but the error term is constructed differently regarding variance and correlation. This way it is possible to elaborate the changing behavior of the tests solely on the adjustments of the error term. To do this a basic setup is simulated at the beginning with an error term drawn from the normal distribution, e.g. $\epsilon_t \sim \mathcal{N}(\mu, \sigma^2)$, for each equation in the VAR(2) process, which then will be adjusted for cross correlation and variance.

The noise for both equations within GC II was constructed via the covariance matrix $\Sigma_{GC_{II}}$, with

$$\Sigma_{GC_{II}} = \begin{bmatrix} 0.25 & 0.00 \\ 0.00 & 0.25 \end{bmatrix}.$$

Further the mean of both error terms was set to zero, to stay in line with the OLS assumptions. Since the tests showed for a sample size of 10,000 similar behavior for relatively larger coefficients as can be seen in the meta simulation, the (granger) causal links were drawn from a range in which the results differed significantly, which is approximately between -0.2 and 0.2. Therefore the quantified Granger causation was drawn from a uniformed distribution within that certain

range, such as $L_{t_{i,j}} \sim \mathcal{U}(-0.2, 0.2)$ for i and j = 1,...,k, while t is the simulation step, i represents the row and j the column of the coefficient matrix L within the tensor T. In addition, L_t is only allowed to vary per lag and not over the simulation steps in order to be able to draw a connection between the underlying coefficients and the error term. Further the assumed granger links were superimposed over the lags 2,3,4 and 5. The tensor count within the DGP was set to 30 and therefore 30 different datasets with different coefficients per lag and 10,000 samples each were simulated. This way the methods get rewarded the more lags they detect from the range 2-5 but also get penalized if the methods falsely detect a diminishing causal dependency structure, for example, from the third lag onwards or if the tests imply significance of the first lag. The results of this setting are compromised within the ROC Curves in figure 3.3.

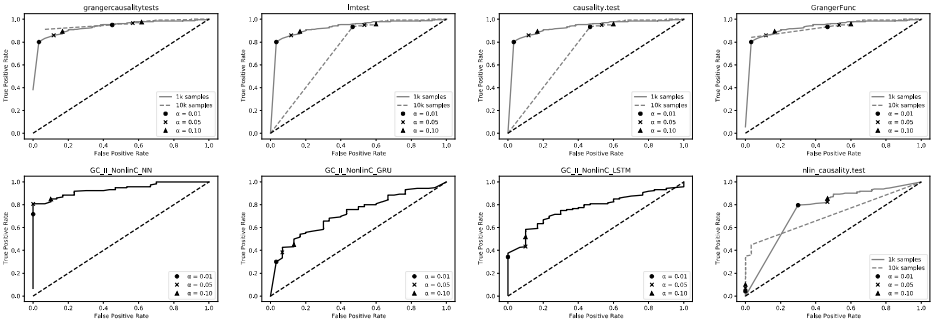

Figure 3.3: ROC Curves - Setting II - Y Granger caused by X and vice versa

Looking at the TPR FPR Distribution of the different methods it seems to be the case that the RNN implementations of nonlincausality seem again to suffer from the inherent linear coefficient development while *nonlincausality_LSTM* shows a better performance as *nonlincausality_GRU*. However this seems not to be true for more simpler neural network architectures such as *nonlincausality_NN* or *nlin_causality.test*, which use a forward neural network (FNN). This indicates the suspicion already indicated in the meta simulation that rather less complex neural network architectures perform better at less complex Granger causality networks. Overall, the OLS approaches, the GLM approach and *nonlincausality_NN* show the strongest performance in this scenario. In addition, the nonlincausality_NN seem to be very robust regarding the significance level. The implementations are able to increase the TPR by only a small increase of the FPR, compared to the results of *nlin_causality.test* which already shows a high FPR for a conservative significance level of $\alpha = 0.01$. However, *grangercausalitytests* seems to perform best for a significance level of 0 which can be explained by the fast convergence to 0 of the p value. What is also surprising to see is the different performance regarding the classical approaches and the glm approach in combination with a high sample size and multi-lag Granger causal links. While the results for a uni-granger-causal-link showed robust results, as shown in section 3.4.11., the results differ for a multi-lag setting as GC II. The implementations show a higher TPR, e.g., at an alpha of 0.01, but also have more false positives, with a underlying sample size of 10,000. The slightly different shapes of the results, regarding a sample size of 10,000 is due to the different convergence speed of the p value to zero from each test. The results indicate that the classical implementations exhibit a higher probability of rejecting the null hypothesis

with increasing sample size if more than one Granger causal link is inherent. To ensure that the test exhibit the best possible performance under the given scenario the sample size for the classical implementations will be reduced to 1,000 from now on. Since the *nlin_causality.test* shows similar behavior and performs better at smaller sample sizes, n is adjusted accordingly. Further, since the results are identical the function grangercausalitytests will represent all other classical implementations and is reffered to as "OLS approach".

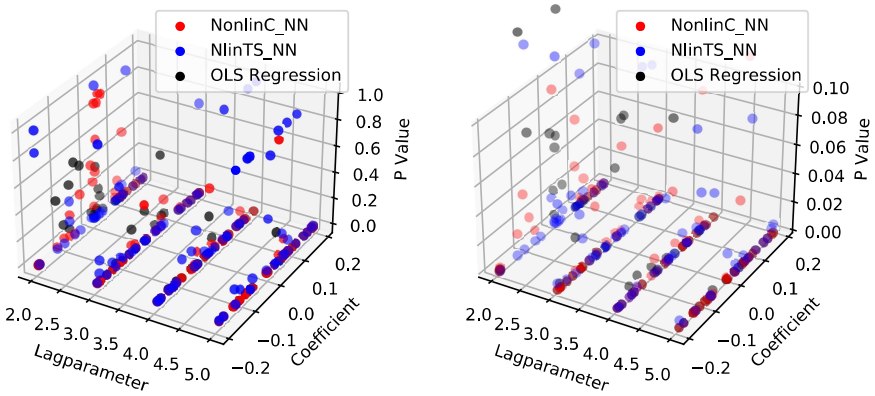

Figure 3.4: P Value development across lags and coefficient range

Similar as in the simulation above the FNN function from nonlincausality, the classical approach and to some extent from statsmodels *nlin_causality.test* show convergence. This convergence can be analysed in more detail at a different perspective by looking at the p value distribution for the various coefficients over the number of lags. The plots demonstrate that the OLS approaches and the to some extend simpler built neural network models exhibit a convergence with increasing tested lags. This can be seen due to a denser distribution of the p values around zero, especially at the fourth lag. However, this convergence seems to vanish with greater amount of lags tested. In addition, the OLS approach seems to handle smaller coefficients better also in a multi-lag case which can be observed by looking at the p values for the corresponding coefficients around zero. This can especially be seen by looking at the p value range of 0 to 0.1, which shows the right plot of figure 3.4. While for the tails of the coefficient distribution the tests are relatively robust, as already analyzed in the meta simulation, the significance levels of the corresponding coefficients left and right to zero seem to be mainly exceeded by the neural network implementations.

Additionally, the methods without the use of neural networks seem to be less fluctuating regarding the p value and therefore may more robust, at least in this scenario. This can be seen by the p value development of *nonlincausality_NN* and *nlin_causality.test* over the lags: It seems to be the case that if the lags 2 and 3 are found to be significant it tends to some extend to regard the fourth lag as not significant. However this behavior might be only an advantage if the lags which Granger cause the timeseries are consecutive as in the underlying scenario[7]).

[7]For non consecutive Granger causal links it may be advisable to select implementations which use criterions such

3.4.1.2.1 Increased correlation

The following simulations examine the behavior of the packages due to different error term constructions as explained in the section above. The error term in the first adjustment includes a correlation of 0.75 while no adjustments are made to the individual mean, variance or coefficients in order to be able to analyze the impact solely on the basis of the increased cross correlation. The data generating process yields the error term relationship and the time series (last 50 datapoints) in figure 3.5.

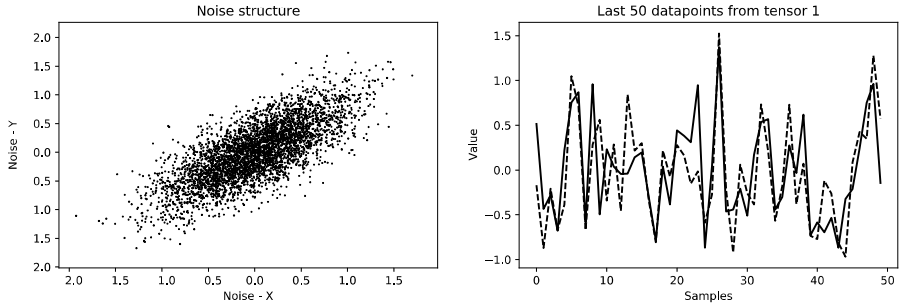

Figure 3.5: Correlation Structure and First 50 Samples (Tensor 1)

The results of the different tests in figure 3.6 show that an increased cross correlation in the error terms lead to a change in the performance across all functions/packages, while the straight lines represent the results for the error term construction in GC II and the dotted line the results including the error term with inherent cross correlation in GC III. It seems to be the case that some tests tend to be more robust towards cross correlated error terms, such as *nlin_causality.test*, then others, such as *nonlincausality_NN*. However, for most of the tests the TPR drops while the FPR increases, especially at the nonlincausality RNN implementations *nonlincausality_GRU* and *nonlincausality_LSTM*.

The decrease of the TPR suggests that an increased correlation in the error term leads to a higher probability of not rejecting the null hypothesis for significant lags. The reason lies probably in the similar construction of the implementations. Since $Model_1$ and $Model_2$, as explained in Chapter 2, are compared by building each model and compare the sum of squared residuals, it could be the case that the increase in the error term leads to multicollinearity and distorts the results. While the increasing correlation does not increase the correlation for different lags between X and Y it leads to a correlation of X and Y for the same lag. This multicollinearity could occur, due to an unknown set of predictors which affects both the X and the Y variable. The result is that the increased correlation leads to correlated predictors within the regression in order to examine Granger causality. Therefore, it could be possible that the marginal increase in prediction performance due to the coefficients is not sufficient to outperform the inherent spurious Granger causality caused by the cross correlation. This increases the likelihood that the additional inclusion of an additional lag in the vector autoregressive model does not seem to provide any additional information in predicting e.g. Y, since the variables X and Y are similar for the same lag, due to the correlation in the error term. This phenomenon occurs

as AIC with higher penalties for additional variables, which are in the sense of Granger causality tests additional lags, such as in S. Li et al., 2019

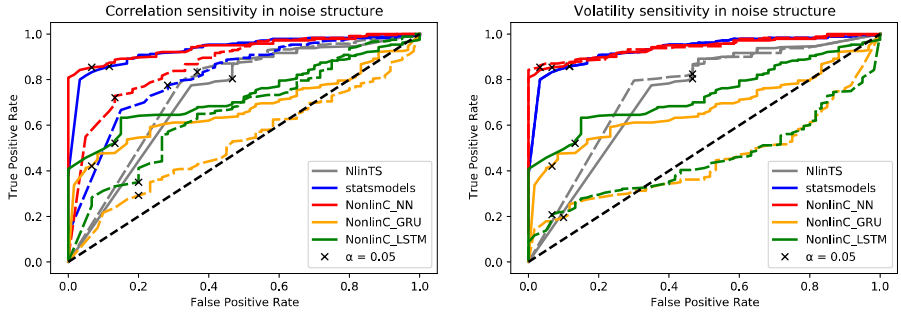

Figure 3.6: Correlation Structure and First 50 Samples (Tensor 1)

more frequently with smaller coefficients, as can be seen for instance for the tensor 24 for the direction of X Granger causing Y, whose results are shown in table 3.3 for *nonlincausality_NN* and the OLS approaches. While the implementations are able to detect the smaller coefficients without cross correlation in the error term it seems that the tests struggle if cross correlation is inherent. This might indicates some kind of spurious *non Granger causality* and supports the discussion that Granger causality is more of a marginal prediction performance increase and less of a causality. For some cases this has also the consequence that lags that do not have any granger causal links, in this simulation the first one, are seen to be significant therefore the false positive rate increases.

	Coefficients	nonlincausality_NN		OLS	
	βX_p	GC II	GC III	GC II	GC III
Lag 1	0.00	0.05	0.16	0.83	0.13
Lag 2	0.08	0.03	0.21	0.00	0.05
Lag 3	-0.05	0.03	0.48	0.00	0.02
Lag 4	-0.05	0.01	0.32	0.00	0.00
Lag 5	0.15	0.00	0.01	0.00	0.00

Table 3.3: Impact of increased cross correlation using the example of tensor 24

3.4.1.2.2 Increased volatility The next analysis derives the impact of increased volatility in the error term in order to examine the tests´ ability in order to detect Granger causation in fluctuating scenarios, the ROC curves can also be seen in figure 3.6. The standard deviation was therefore increased from $\sigma = [0.5, 0.5]$ to $\sigma = [0.9, 0.9]$.

In contrast to an increased change in the cross-correlation within the error term most tests tend to be robust against an increase in the volatility. However, similar to GC III the Granger causality tests which use RNN architectures react with a high decrease in the true positive rate. Overall, it can be concluded that the NlinTS neural network implementation seems to be the most robust in the sense of changes in the error term, at least for the regarded scenarios.

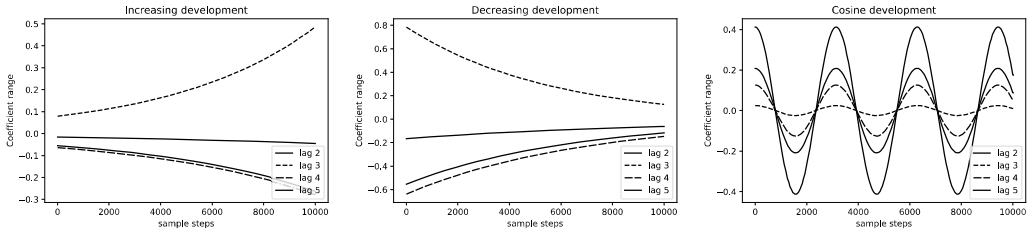

Figure 3.7: Nonlinear coefficient development

3.4.2 Nonlinearity and Granger causality

While the previous chapter examined the packages´ ability to infer causality in the sense of Granger in a linear process with fixed or linear coefficient development and a linear dependency between the equations, this chapter aims to analyze the tests power in nonlinear settings. Therefore nonlinearity is introduced in two ways. The first approach assumes a nonlinear development of the coefficient, e.g. the cosine curve or exponential de- and increasing relationships, as can be seen in figure 3.7. A second approach draws from the autoregressive model from the meta simulation and extends the number of lag dependencies in a nonlinear manner. [8] Nonlinear grangercausality detection is not something new in the timeseries literature, Hiemestra et. al already described the limitations of the classical approach in 1994 in causation test between Stock Price and Volume of the Dow Jones and use the nonlinear approach from Baek and Brock from 1989 with the use of correlation integrals. Papagiannopoulou et. al also suggest replacing linear models such as $Model_1$ and $Model_2$ in equation (1) and (2) by a random forest model. To adopt for the Granger cause in the original sense the authors still infer the causality if the $R^2 improves when the random forest uses more lags, such as x_{t-1}, x_{t-2}, ..., x_{t-3}$. However, there are only sparse representations of these methods applicable in python which output also allows for comparability.

3.4.2.1 Cosine Granger causal link development

In order to increase the data inherent nonlinearity, the flow of the causation between cause and effect is drawn from a cosine curve which is then multiplied with a sample from a random uniform distribution in order to vary across 20 different tensors for each lag and therefore 20 different datasets in which again the equations have a bidirectional relationship and lag dependence is set to the lags reaching from 2 to 5.

In order to increase the performance of the neural network methods, hyperparameter tuning was conducted and the following changes were made. Regarding *nonlincausality_GRU* the number

[8] Please note in this paper only the cosine scenario is presented while the other scenarios can be reproduced in the source code.

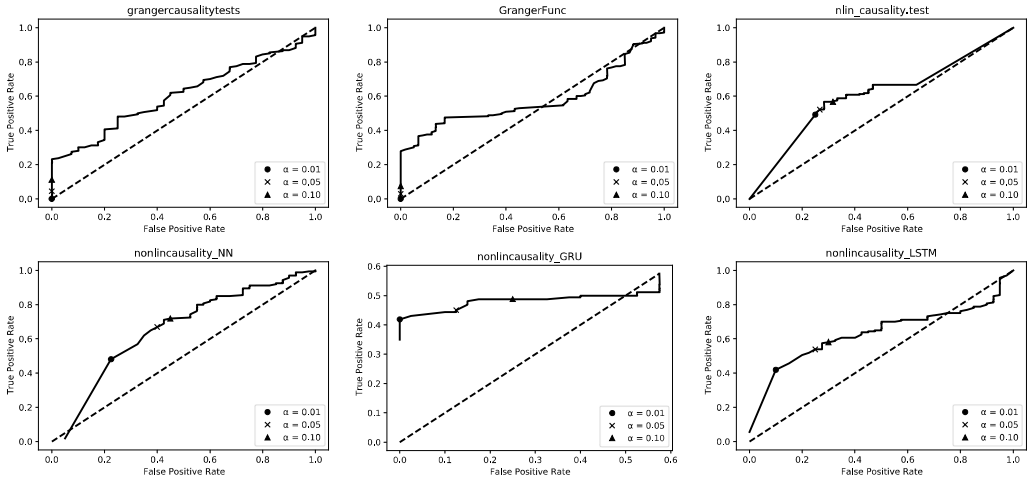

Figure 3.8: ROC Results - cosine granger causal links

of neurons for two layers was changed from [15,15] to [30,25], while for *nonlincausality_NN* the number of neurons were changed from [25,10] to [250,100]. As can be seen *nonlincausality_NN* required the highest amount of additional neurons in order to perform well in a nonlinear scenario. The results for *nonlincausality_LSTM* and *nlin_causality.test* showed only marginal variation at different settings and were therefore set to the same architecture as in the previous scenarios.

While the differences in the linear increasing and decreasing scenarios were acceptable especially comparing the computational time of the neural network implementations, the results in figure 3.8 clearly differ and the OLS and GLM regression approaches suffer under a nonlinear setting, e.g. provoked by a cosine run of the causal effects between the time series represented by the coefficient. The implementations for linear (lag) dependencies can only increase the TPR by using an unacceptable signficance level beyond 0.1. Methods using neural networks are able to outperform the OLS and GLM approaches, which what speaks in favor of using neural networks in a nonlinear coefficient setting. However, regarding the TPR at a significance level of 0.05 *nonlincausality_NN* is able to outperform all other implementations. The suspicion in the previous chapter that the RNN implementations perform better in nonlinear or more complex contexts has partly been proven. At a significance level of 0.01, *nonlincausality_LSTM* and *nonlincausality_GRU* show a higher TPR with lower FPR compared to scenario GC II. However, the distances of the results for the considered significance levels are further apart. Even if the neural network implementations show a good performance, it should be mentioned that a conservative significance level should be used if non-linear coefficient curves are suspected. Overall, no implementation shows very good results as in the previous scenarios, especially GC II. Regarding the neural network approaches hyperparameter tuning could be extended.

3.4.2.2 Autoregressive exogenous model (ARX)

While in the previous chapter the nonlinearity was achieved by a cosine signal attached to the granger causal link, this setting examines the test inference when the causing variable has a direct nonlinear shape. This was obtained by generating the nonlinear timeseries through the interaction of a cosine term and a sine term. The values for the nonlinear function were constructed by an evenly spaced, regarding the sample size, intervals between 0 and 20 for the cosine and 0 and 3 for the sine signal. To be able to vary the data across the tensor a fraction of 0.2 from a random number was subtracted from the timeseries. Then the function for creating the causing variable X_t is as follows:

$$f(x,z) = cos(x) + sin(z) - 0.2 * \lambda \tag{12}$$

λ is a random number while x and z are obtained by:

x $\in$D$\subset$Z, $with$D= $\frac{i}{a} * 20 \forall i \in [0;a]; a$ =D

z $\in$D$\subset$Z, $with$D= $\frac{i}{a} * 3 \forall i \in [0;a]; a$ =D

Then Y was simulated as a autoregressive exogenous process which is dependent on its own third and fourth lag and on the third and fourth lag of X which is created by f(x,z). In addition, a multivariate normal distributed noise was added to both processes in order to ensure that the variance covariance matrix has full rank, while the first column was attached to the effected and the second column was added to the causing timeseries. This way it is possible to create, if wished by the user, a higher correlation in the error term. The effected variable, was then created through an autoregressive process by also including lags of X. The data sample of the exogenous nonlinear created variables for the first tensor can be seen in figure 3.9.

$$Y_t = \alpha_0 + \sum_{i=-p}^{p} \alpha_i Y_{t-i} + \sum_{i=-p}^{p} \beta_i X_{t-i} + \varepsilon_t \tag{13}$$

While the selected lags are lag 3 and lag 4, as already mentioned.

The results are again represented in the ROC curves in figure 3.10. The classical approaches exhibit strange behavior and seem to have p values which are always very close to zero or zero, leading to very high TPR and FPR rates, which can be seen from the confusion matrix of *grangercausalitytests* in table 3.4.

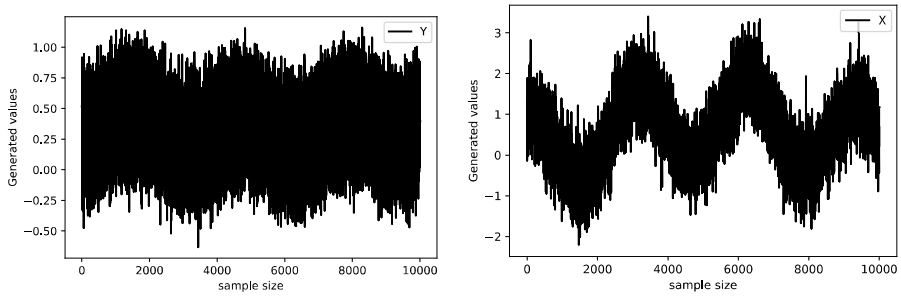

Figure 3.9: ARX model generated data example

		Inferred causal links		
		Positive	Negative	Total
True causal links	Positive	22	38	60
	Negative	5	55	60
	Total	60	60	120

Table 3.4: grangercauslality tests - confusion matrix ARX model with $\alpha = 0.05$

This is probably due to the fact that the generated X data is not stationary anymore, exemplary for the first data set analysed via an ADFuller test with a p value of 0.34. To account for that fact the data was differentiated in order to make the time series stationary, the results for X Granger causing Y can be found in appendix A and the improved results in table 3.5. Differentiating the data normalizes the results and the classical approaches are able to perform well. The results show probably mistakenly a high TPR. As already explained for the figure 3.4 the p values of statsmodels tend to stay zero for a rather high amount of lags as soon as one lag is seen significant. By looking at the high FPR rate, which results from false positives regarding the first and the second lag, the p values stay at zero and therefore the TPR rises since the underlying granger causal links are set to lag three and four. If the data is differentiated the results seem to be more robust for all implementations. Therefore, it can be concluded that it is advisable to test for stationarity for all p values in order to account for the problem that the p values have a higher probability of staying zero when the data is not stationary even for lags which are not granger causing.

What also can be observed is the higher performance of the RNN approaches. It seems to be the case that the LSTM and GRU implementations perform better than a nonlinear dataset

		Inferred causal links		
		Positive	Negative	Total
True causal links	Positive	37	23	60
	Negative	15	45	60
	Total	60	60	120

Table 3.5: grangercauslality tests - confusion matrix ARX model differentiated with $\alpha = 0.05$

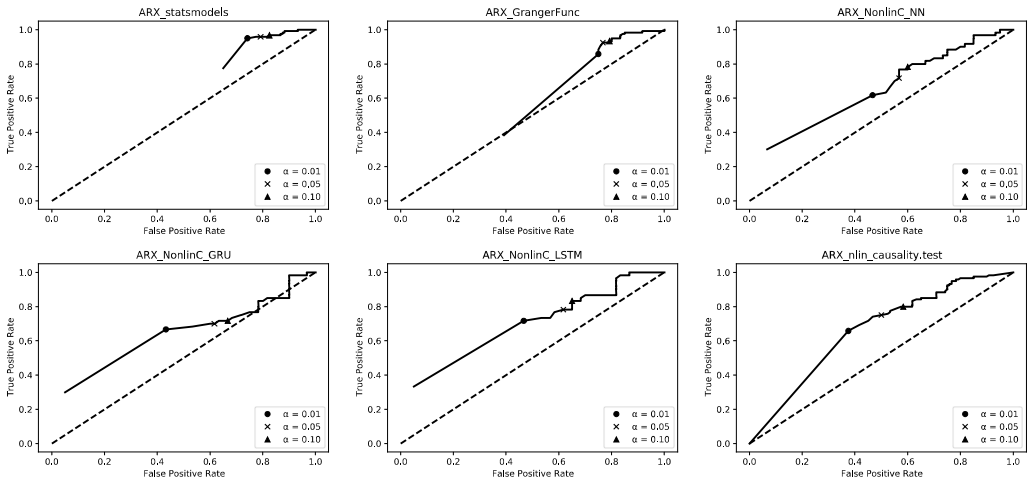

Figure 3.10: ROC Results - cosine granger causal links

is created, rather than created nonlinear (granger) causal link development within the data creation. *nonlincausality_LSTM* even shows the highest performance in this scenario.

3.5 Multivariate Granger causality detection

Moving forward on methodological grounds of Granger causality it is necessary to talk about multivariate Granger causality detection to show the full power of the method. Since only then it is possible to account for all available information and therefore all possible causal effects (c. p.Atukeren, 2007). It was shown that with the already proposed packages it is possible to detect Granger causality efficiently in both directions within a bivariate VAR Process, linear and at higher dimensions. However the Granger causality is frequently criticized in the literature due to the phenomenon of spurious Granger causality which describes the case that the bivariate Granger causality test suggests that one time series, e.g. X is granger causing Y which is in reality not true. This is possible due to a indirect Granger-cause in the sense that X is invariant to Y in reality but another Variable, e.g. Z, is granger causing both variables.

An empirical simulation of the Granger causality frameworks tested in the previous sections shows that some tests are very susceptible to this kind of causality which would in reality lead to wrong results. In order to trigger this kind of scenario an activation matrix is used of the dimension kxk. Within this matrix the existence of granger causal links are determined by adding the integer 1 to the off diagonal elements of a identity matrix for which a granger causal link would occur due to matrix multiplication. Then the activation function is processed within the DGP and gets coefficients assigned for which the value is 1 and 0 otherwise while the coefficients are similar drawn as in GC II:

$$C_{ij}(A_{i,j}) = \begin{cases} U(-0.2, 0.2), & \text{if } A_{i,j} = 1 \\ 0, & \text{otherwise} \end{cases}$$

The following example demonstrates the process for one matrix transformation, from the activation matrix A to the coefficient matrix C which will be then inserted to the tensor T and varied over the Tensor from T1 to T30.

Assuming that the first column is Y, the second X and the third Z the matrix A represents the setting that Y is invariant to X and Z, X is Granger caused by Y and Z is invariant to X but Granger caused by Y.

$$\mathrm{A} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \quad \mathrm{C} = \begin{bmatrix} 0.19 & 0.00 & 0.00 \\ 0.02 & 0.18 & 0.00 \\ 0.08 & 0.00 & -0.11 \end{bmatrix}$$

While A represents the activation matrix and C the corresponding coefficient matrix which is again varied over the tensors to create different datasets. The results of *grangercausalitytests* in table 3.6 show that for a significance level of $\alpha = 0.10$ the test suffers from sporious Granger causality for almost every third time series simulation. Hence for a full detection of all relevant and actual, in reality, existing granger causal links in a equation system it is necessary to also test for multivariate Granger causality. The remaining tests show similar behavior and can be reconstructed in the source code.

Lags	T_5	T_6	T_{10}	T_{11}	T_{14}	T_{15}	T_{21}	T_{25}
1	0.52	0.56	**0.05**	0.33	**0.04**	0.17	**0.00**	0.31
2	**0.06**	0.35	**0.03**	0.29	**0.07**	0.39	**0.00**	0.18
3	0.16	0.12	**0.07**	0.45	**0.10**	0.52	**0.10**	**0.07**
4	0.29	**0.09**	**0.00**	**0.05**	0.14	**0.10**	**0.09**	**0.07**
5	0.59	0.14	**0.04**	**0.09**	0.30	0.18	0.53	0.34

Table 3.6: Results of multivariate simulation - grangercausalitytests

Libraries which provide these kind of analysis are for example bruceR (R) with the function *granger_causality* and also statsmodels (Python) which offers a multivariate grangercausality analysis for the VAR results. While statsmodels is able to correctly infer that no Granger causal link is inherent from $timeseries_2$ to $timeseries_3$, bruceR creates a perfect replication of the links and also shows which timeseries are independent from each other. The results can be seen in appendix B.

4 Conclusion

The results showed different packages or libraries are suited more or less to different scenarios. While the classical implementations like *grangercausalitytests* or *lmtest* perform well under linear scenarios, the tests suffer when nonlinearity or nonstationary is inherent. Differentiating partially solved this problem but also showed that in case of nonlinearity the tests still tend to have a high FPR rate which leads to a high TPR due to the test construction. Furthermore, it could be shown that a higher accuracy by using neural networks depends on the architecture as well as on the data construction. While the RNN implementations *nonlincausality_LSTM* and *nonlincausality_GRU* performed well when the causing variable itself exhibits a nonlinear development, the methods were outperformed in less complex scenarios from *nonlincausality_NN* and *nlin_causality.test*. Also the amount of samples plays an important role if neural networks are used for Granger causality detection. While the implementations from *nonlincausality* work better at rather high sample sizes, *nlin_causality.test* shows a higher performance at smaller samples. Overall, the inclusion of neural networks in general are useful in detecting granger causal links between bivariate time series. However, it could also be shown that some of the methods also tend to suffer, e.g. from cross correlation in the error term, which might by introduced by a third variable which Granger causes both timeseries. Therefore an extension to the bivariate packages was given and it was demonstrated how the spurious Granger causality could be avoided at least in linear scenarios. For the future it may be advisible to create nonlinear multivariate Granger causality tests, e.g. with the use of neural networks.

Bibliography

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, *41*(6), 391–407.

Kanakaraj, M., & Guddeti, R. M. R. (2015). NLP based sentiment analysis on twitter data using ensemble classifiers.

Granger, C. W. (1980). Testing for causality: A personal viewpoint. *Essays in Econometrics vol II: Collected Papers of Clive W. J. Granger*, 48–70.

Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, *37*(3), 424–428.

Shang, H. L., Ji, K., & Beyaztas, U. (2020). Granger causality of bivariate stationary curve time series. *Journal of Forecasting*, *40*(4), 626–635.

Studenmund, A. H. (2017). *Using econometrics: A practical guide*. Pearson.

Atukeren, E. (2007). Christmas cards, easter bunnies, and granger-causality. *Quality amp; Quantity*, *42*(6), 835–844.

Mazzarisi, P., Zaoli, S., Campajola, C., & Lillo, F. (2020). Tail granger causalities and where to find them: Extreme risk spillovers vs spurious linkages. *Journal of Economic Dynamics and Control*, *121*, 104022.

Tank, A., Covert, I., Foti, N., Shojaie, A., & Fox, E. B. (2021). Neural granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1.

Baek, K. G., & Brock, W. A. (1989). *A Nonparametric Test For Independence Of A Multivariate Time Series* (ISU General Staff Papers No. 198912010800001205). Iowa State University, Department of Economics.

Zhu, P., Zhang, X., Wu, Y., Zheng, H., & Zhang, Y. (2021). Investor attention and cryptocurrency: Evidence from the bitcoin market. *PLOS ONE*, *16*(2).

Li, S., Zhang, H., & Yuan, D. (2019). Investor attention and crude oil prices: Evidence from nonlinear granger causality tests. *Energy Economics*, *84*, 104494.

Li, Y., Goodell, J. W., & Shen, D. (2021). Comparing search-engine and social-media attentions in finance research: Evidence from cryptocurrencies. *International Review of Economics amp; Finance*, *75*, 723–746.

Hmamouche, Y. (2020). Nlints: An r package for causality detection in time series. *The R Journal*, *12*(1), 21.

Zheng, S., Shi, N.-Z., & Zhang, Z. (2012). Generalized measures of correlation for asymmetry, nonlinearity, and beyond. *Journal of the American Statistical Association*, *107*(499), 1239–1252.

Stock, J. H., & Watson, M. W. (2001). Vector autoregressions. *Journal of Economic Perspectives*, *15*(4), 101–115.

Rey, D., & Neuhäuser, M. (2011). Wilcoxon-signed-rank test. *International Encyclopedia of Statistical Science*, 1658–1659.

A Stationary ARX results

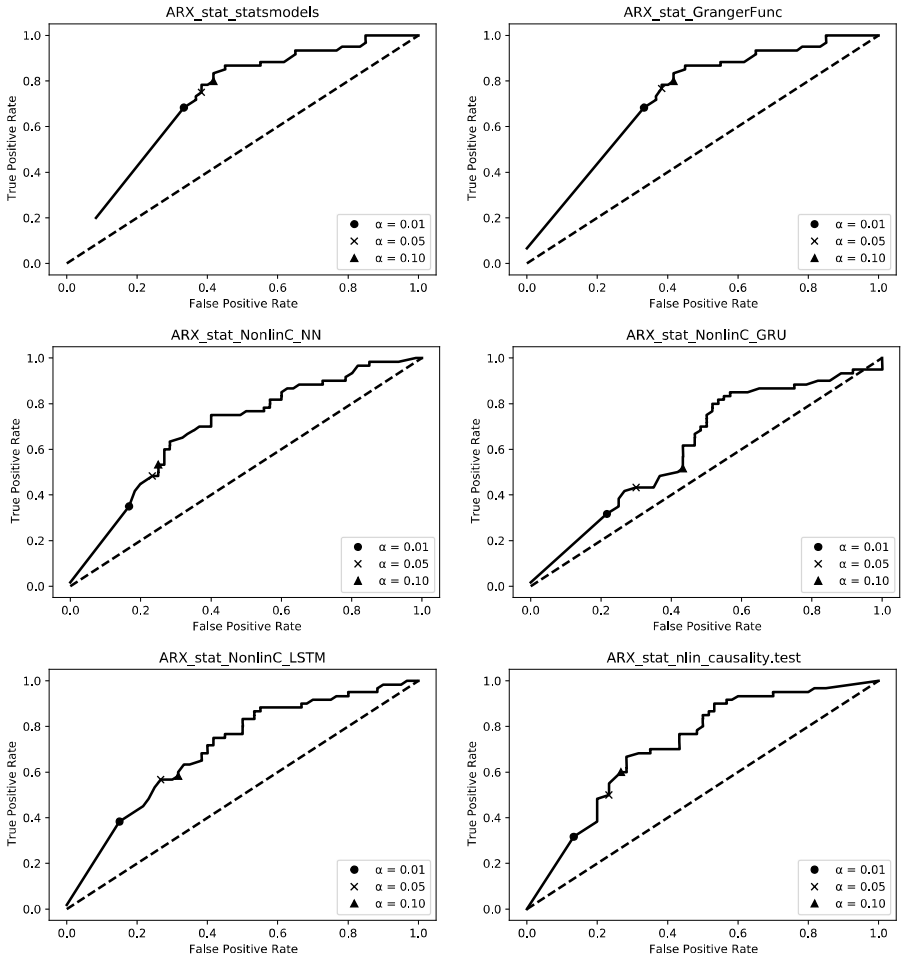

Figure A.1: ROC Results - stationary ARX data

B Multivariate results

```
Granger causality F-test. H_0: y2 does not Granger-cause y3. Conclusion: fail to reject H_0 at 5% significance level.
================================================
Test statistic Critical value p-value     df
------------------------------------------------
       0.05516          2.214   0.998 (5, 29937)
------------------------------------------------
```

Figure B.1: Python multivariate Granger causality implementation via statsmodels

```
F test and Wald χ² test based on VAR(5) model:
-------------------------------------------------------------------------------
                        F  df1   df2      p          Chisq  df      p
-------------------------------------------------------------------------------
  ---------
  X0 <= X1          0.47    5  9979   .798           2.35   5   .798
  X0 <= X2          0.66    5  9979   .656           3.29   5   .656
  X0 <= ALL         0.51   10  9979   .883           5.12  10   .883
  ---------
  X1 <= X0        147.99    5  9979 <.001  ***    739.97   5 <.001  ***
  X1 <= X2        187.95    5  9979 <.001  ***    939.76   5 <.001  ***
  X1 <= ALL       170.54   10  9979 <.001  ***   1705.38  10 <.001  ***
  ---------
  X2 <= X0        156.07    5  9979 <.001  ***    780.36   5 <.001  ***
  X2 <= X1          0.31    5  9979   .907            1.55   5   .907
  X2 <= ALL        79.48   10  9979 <.001  ***    794.85  10 <.001  ***
-------------------------------------------------------------------------------
```

Figure B.2: R multivariate Granger causality implementation via bruceR

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.
Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Passau, den 11th February 2021

.....
(Fabian Dick)