Transfer-entropy-based feature selection for The Impact of El Niño and La Niña on the Realized Volatility of Agricultural Prices

Alireza Nouri
Sharif University of Technology
alireza.nouri77@sharif.edu

Hirbod Assa
University of Essex
H.assa@essex.ac.uk

Abstract

In this paper, we examine the impact of El Niño and La Niña weather episodes on the realized variance of three agricultural commodity prices. First, we employ three methods for feature selection: Transfer Entropy for capturing non-linear relationships, Backward Stepwise Regression, and Lasso for identifying linear relationships between El Niño and La Niña events and realized variance (RV). Then, we use both linear and non-linear predictive models to detect relationships between RV and other features, especially at longer forecast horizons, for several agricultural commodities studied in this research.

Keywords: El Niño, La Niña, realized volatility, agricultural prices, transfer entropy, feature selection, linear and nonlinear models

1 Introduction

Many tropical and subtropical regions have well-established climates that are greatly impacted by the El Niño-Southern Oscillation (ENSO), a periodic change in winds and sea surface temperatures in the tropical eastern Pacific Ocean [1] . El Niño denotes the warming phase and La Niña denotes the cooling phase. These periods, which vary in intensity, usually happen every few years and might persist for many months. In many parts of the world, ENSO is a significant cause of yearly climatic fluctuation [2].

Recent studies, building on the work of Bonato et al, Çepni, Gupta, Pierdzioch. (2022) [3], have documented the significant effects of ENSO on the realized variance of agricultural commodity prices . However, this study uses only the non-linear model

Random Forest to identify the impact of ENSO on the realized variance of agricultural commodity prices. Most analyses are based on in-sample and out-of-sample structural models.

Our research aims to investigate ENSO's ability to forecast the realized variance of agricultural commodity returns by using feature selection methods, including non-linear (TE) and linear (BSR, Lasso) approaches. We then employ several linear and non-linear predictive models to achieve better and higher-quality results by selecting features that provide the most relevant information. Forecasting realized variance is crucial due to the significant price fluctuations in agricultural commodities since 2008, leading to periods of high and low volatility [4]. According to Johnson (2011) [5], extreme weather events are a major cause of increased volatility, but other variables including the production of biofuels, market speculation, rising demand, and depleting food reserves all play a part. Since volatility plays a major role in investment decisions, accurate volatility projections are essential for traders and market players as well as policymakers seeking to ensure food security.

Traditional feature selection techniques, such as autocorrelation or partial autocorrelation [6], have limitations in forecasting since they are unable to find nonlinear correlations with the ideal latency for realizing volatility prediction. In order to get around this, we find features with the maximum information flow using transfer entropy [7]. These features are then used in both linear and nonlinear models to improve prediction accuracy.

We forecast weekly realized variance (RV) for our econometric model using an extended heterogeneous autoregressive (HAR)-RV model [8]. We account for several control variables, including cross-market RVs, interest rate, inflation rate, and El Nino and La Nina events. RV is calculated by summing squared 5-minute intraday returns over a 7-day period. [13].

To reflect the effect of ENSO on the realized volatility of agricultural returns, we employ both linear and nonlinear models. We use Vector Autoregression (VAR) for linear models, as explained by Lütkepohl (2005) [9]. We use Random Forests as described by Breiman, L. (2001) [10], XGBoost introduced by Chen and Guestrin (2016) [11]., and Support Vector Regression (SVR) based on the work of Smola and Schölkopf (2004) [12] for nonlinear models.

2 DATA

2.1 Agricultural Commodities and Realized Variance

Prices are sampled at 5-minute intervals. These commodities are generally considered to be highly traded, as identified by the Food and Agriculture Organization (FAO) of the United Nations (UN). Table 1 lists the individual commodities, their ticker symbols, and the trading exchanges.

We use the classical estimator of realized variance (RV), defined as the sum of squared intraday returns. [13], [3]:

$$RV_t^d = \sum_{i=1}^{M} r_{t,i}^2,$$

where $r_{t,i}$ represents the intraday returns vector and i ranges from 1 to M, the number of intraday returns. Weekly RV is calculated by aggregating the daily RV over a trading week (Monday to Friday):

$$RV_t = \sum_{i=\text{Monday}}^{\text{Friday}} RV_i.$$

Table 1 shows the summary statistics of the weekly realized variance (RV) for the three agricultural commodities, Sea Surface Temperature Anomalies (SSTA), interest rate, and inflation rate. Based on the ADF and KPSS tests, the data are stationary. The dataset covers the period from December 7, 2009, to October 23, 2022, comprising 725 observations. Figure 1 illustrates the realized variances, which exhibit occasional large peaks, a common characteristic of financial-market return volatilities.

Variable	**Mean**	**Median**	**SD**	**Min**	**Max**	**ADF (p-value)**	**KPSS (p-value)**
SSTA	0.003380	-0.100000	0.861551	-1.800000	3.000000	0.005324	0.100000
RV-wheat	0.001743	0.001178	0.001864	0.000029	0.019987	0.012094	0.119027
RV-corn	0.001204	0.000724	0.001359	0.000026	0.112291	0.000440	0.010000
RV-soybean	0.000843	0.000564	0.000990	0.000007	0.009762	0.000000	0.124204
Inflation rate	0.021699	0.017500	0.020652	-0.021000	0.090600	0.043294	0.116842
Interest rate	0.023121	0.023000	0.007383	0.006200	0.039800	0.035890	0.124624

Table 1: Descriptive statistics and test p-values for different variables.

2.2 ENSO Data

For our weekly measurements of ENSO intensity, we utilize sea surface temperature anomalies (SSTA) for the "NiŰno 3.4" region, which is located between 5°N and 5°S and

120°W and 170°W. Compared to other measures, which are also sometimes used, such the Southern Oscillation Index (SOI) or the Equatorial SOI (EQSOI) anomalies, SSTA is more often used to assess ENSO intensity [14],[15]. This approach aligns with the statistic that [16] employed in his study on predicting. High-frequency analysis is feasible since SSTA data is available on a weekly basis, but SOI and EQSOI data are only available on a monthly basis over the course of our empirical inquiry.

The National Weather Service Climate Prediction Center, a division of the National Oceanic and Atmospheric Administration (NOAA), is the source of the weekly SSTA data. Three consecutive months with an SSTA of 0.5°C (0.9°F) or above are considered an El Niño event. In a similar vein, three consecutive months with an SSTA of -0.5°C (-0.9°F) or less are considered to be part of a La Niña episode.

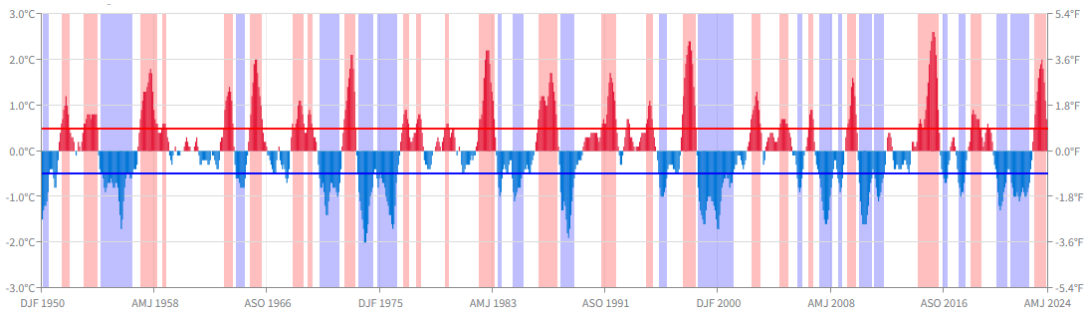

Figure 1: Oceanic Nino Index (ONI) from 1950 to 2024, showing the 3-Month Running Mean of Nino 3.4 SST Anomalies. The blue shaded areas indicate periods when the ONI index was below -0.5 degrees Celsius for five consecutive 3-month running mean SST anomalies, corresponding to La Niña phases. The red shaded areas indicate periods when the ONI index was above 0.5 degrees Celsius for five consecutive 3-month running mean SST anomalies, corresponding to El Niño phases. Source: https://www.ncei.noaa.gov/access/monitoring/enso/sst

Figure 2 illustrates the resulting El Niño and La Niña events on Interest rate, Inflation rate and Realized variances.

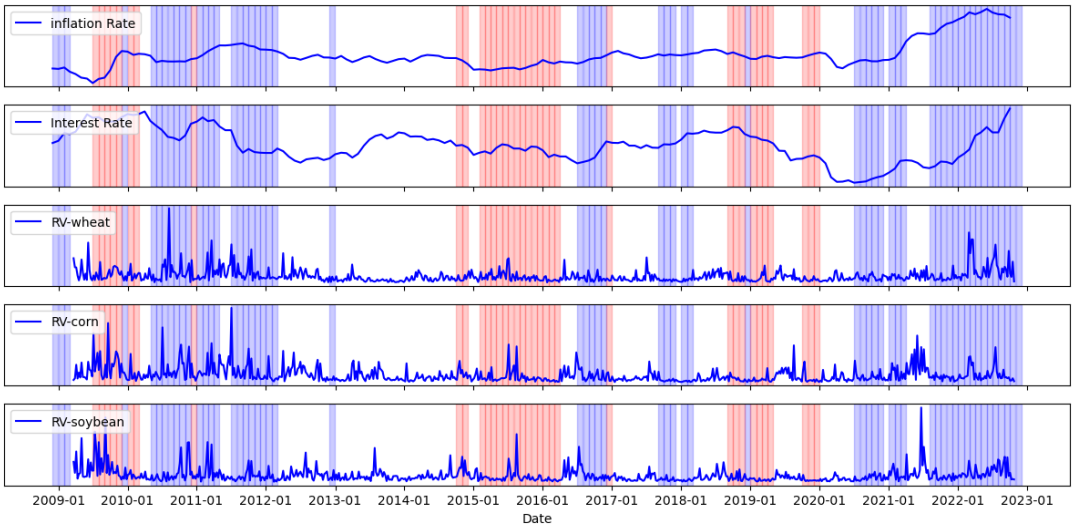

Figure 2: Figure 2: The figure displays time series data for various economic indicators and their volatility, with specific climate periods highlighted. (a) The first graph shows the monthly interest rate, illustrating fluctuations in borrowing costs over time. (b) The second graph depicts the monthly inflation rate, reflecting changes in the general price level of goods and services. (c) The third graph presents the realized variance of wheat prices, providing insights into the volatility of wheat market fluctuations. (d) The fourth graph displays the realized variance of corn prices, capturing the variability in corn price movements. (e) The final graph illustrates the realized variance of soybean prices, indicating the volatility of soybean prices.The shaded regions in the graphs are as follows:- The blue shaded areas represent periods of La Ni˜na, characterized by cooler than average sea surface temperatures in the central and eastern Pacific Ocean.- The red shaded areas correspond to periods of El Ni˜no, marked by warmer than average sea surface temperatures in the same regions.These climate events are known to influence economic indicators and commodity prices, which are highlighted in the respective graphs.

2.3 Data Modeling Methodology

we want to examine realized volatility on a weekly basis provides a clearer view of the impact of ENSO events on agricultural commodity prices

To achieve this, we use a Rolling Window (RW) approach to compute the realized volatility (RV) of agricultural commodity prices and Sea Surface Temperature Anomaly (SSTA). Specifically:

- **Monthly Realized Volatility:** We calculate the monthly realized volatility by averaging the realized volatility from the previous three weeks. This is computed as:

$$RV_{\text{monthly},t} = \frac{1}{3}\left(RV_{\text{weekly},t-1} + RV_{\text{weekly},t-2} + RV_{\text{weekly},t-3}\right)$$

- **Quarterly Realized Volatility:** For quarterly realized volatility, we average the realized volatility from the previous eleven weeks:

$$RV_{\text{quarterly},t} = \frac{1}{11}\left(RV_{\text{weekly},t-1} + \cdots + RV_{\text{weekly},t-11}\right)$$

- **Interpolating Interest Rates and Inflation:** We use linear interpolation to bring inflation and interest rates into line with the volatility data because they are recorded at a lower frequency than realized volatility and SSTA. In order to correspond with the greater frequency of RV and SSTA data, this method raises the data resolution of these variables. [17].

  In summary, the RW approach facilitates calculation of monthly and quarterly realized volatility, and linear interpolation effectively aligns the frequency of inflation and interest rate data, ensuring a consistent dataset for analysis of the ENSO effects on commodity price volatility.

2.4 Transfer Entropy

Transfer entropy is a non-parametric method used to quantify the amount of directed, time-asymmetric information flow between two stochastic processes [7, 18, 19]. It evaluates how much uncertainty about the future states of one process Y can be reduced by knowing the past values of another process X, given the past values of Y itself. Specifically, let X_t and Y_t represent two random processes at time $t \in \mathbb{N}$, and if the information is measured using Shannon entropy, transfer entropy is defined as:

$$T_{X \to Y} = H\left(Y_t \mid Y_{t-1:t-L}\right) - H\left(Y_t \mid Y_{t-1:t-L}, X_{t-1:t-L}\right),$$

where $H(\cdot)$ denotes the Shannon entropy. Variations of this concept using alternative entropy measures, such as Rényi entropy, have also been introduced. .[19, 20].

Transfer entropy is closely related to conditional mutual information [21, 22], with the conditioning based on the history of the influenced variable $Y_{t-1:t-L}$:

$$T_{X \to Y} = I\left(Y_t; X_{t-1:t-L} \mid Y_{t-1:t-L}\right).$$

In the context of vector auto-regressive processes, transfer entropy simplifies to Granger causality [23]. However, it is particularly useful when the assumptions of Granger causality are not valid, such as when dealing with non-linear data [24, 25],

although it generally requires larger datasets for accurate computation [26]. Several methods, such as binning and nearest neighbors, are used to estimate the probabilities involved, or alternatively, non-uniform embedding can be applied to reduce computational complexity [27]. Although initially conceived for bivariate systems, transfer entropy has been extended to multivariate settings by conditioning on additional potential influencing variables [28], or by assessing transfer from multiple sources simultaneously [29], although these approaches typically require even more data.

Applications of transfer entropy include measuring functional connectivity in neural systems [29, 30, 31], evaluating social influence within networks [24], and examining causal relationships in the context of armed conflicts [32]. Transfer entropy can also be seen as a finite version of directed information, first introduced by James Massey in 1990 [33]. The directed information is expressed as:

$$I(X^n \to Y^n) = \sum_{i=1}^{n} I(X^i; Y_i \mid Y^{i-1}),$$

where $X^n = (X_1, X_2, ..., X_n)$ and $Y^n = (Y_1, Y_2, ..., Y_n)$. Directed information is vital for understanding the limits of communication channels, whether feedback is present or not [34, 38], and has applications in decision-making with causal side information [35].

2.4.1 TE Estimation

Transfer entropy is a powerful statistical tool used to measure the direction and strength of information transfer between two time series. The Kraskov estimator, introduced by Kraskov, Stögbauer, and Grassberger (2004) [36], is a non-parametric method for estimating mutual information, which is a critical component in computing transfer entropy.

The concept of transfer entropy was introduced by Schreiber (2000) [7] as a measure of the information transferred from one time series to another. Unlike mutual information, which is symmetric, transfer entropy is asymmetric and can capture directional influences. The Kraskov estimator is particularly useful in this context

due to its ability to estimate mutual information from finite sample sizes robustly.

The Kraskov estimator leverages k-nearest neighbors (k-NN) to estimate mutual information. This method avoids the need for binning the data, which can be problematic for high-dimensional datasets. The core idea is to estimate the entropy based on the distances to the k-th nearest neighbor in the joint space and marginal spaces.

The mutual information $I(X;Y)$ between two continuous random variables X and Y is given by:

$$I(X;Y) = \int \int p_{XY}(x,y) \log \left(\frac{p_{XY}(x,y)}{p_X(x)p_Y(y)} \right) dx\, dy,$$

where $p_{XY}(x,y)$ is the joint probability density function, and $p_X(x)$ and $p_Y(y)$ are the marginal probability density functions of X and Y, respectively.

The Kraskov estimator uses the k-NN approach to approximate these densities. The key steps are as follows:

1. **Compute Distances**: For each point (x_i, y_i) in the joint space, compute the distances to its k-th nearest neighbor. Let ϵ_i be the distance to the k-th nearest neighbor in the joint space. Similarly, compute the distances $\epsilon_{i,x}$ and $\epsilon_{i,y}$ to the k-th nearest neighbors in the marginal spaces.

2. **Count Neighbors**: For each point, count the number of neighbors within ϵ_i in the marginal spaces. Let $n_{x,i}$ be the number of points within ϵ_i in the X-space, and $n_{y,i}$ be the number of points within ϵ_i in the Y-space.

3. **Estimate Entropy**: Use the following formula to estimate the mutual information:

$$\hat{I}(X;Y) = \psi(k) - \frac{1}{k} \sum_{i=1}^{k} \left[\psi(n_{x,i} + 1) + \psi(n_{y,i} + 1) \right] + \psi(N),$$

where ψ is the digamma function, and N is the total number of points in the dataset.

To compute transfer entropy $T_{X \to Y}$ using the Kraskov estimator, consider the time series X_t and Y_t. The transfer entropy is defined as:

$$T_{X \to Y} = I(Y_{t+1}; X_t | Y_t),$$

which measures the information contained in X_t about Y_{t+1} that is not already contained in Y_t. This can be estimated using the conditional mutual information:

$$\hat{T}_{X \to Y} = \hat{I}(Y_{t+1}; X_t | Y_t) = \hat{I}(Y_{t+1}, X_t, Y_t) - \hat{I}(Y_{t+1}, Y_t).$$

Here, $\hat{I}(Y_{t+1}, X_t, Y_t)$ and $\hat{I}(Y_{t+1}, Y_t)$ are estimated using the Kraskov method as described above.

The Kraskov estimator is advantageous due to its non-parametric nature and ability to handle small sample sizes effectively. It has been widely applied in various fields, including neuroscience, finance, and climate science, to uncover directional dependencies in complex systems.

The Kraskov estimator provides a robust method for estimating mutual information and, by extension, transfer entropy. Its application to causal discovery in heterogeneous and nonstationary data holds significant promise, as it can reveal intricate dependencies and information flows that traditional methods might miss.

2.5 Forecasting Methods

2.5.1 Random Forest

Random Forest Regressor is built on some decision trees, and for each decision tree we sample data with the bootstrap method and train every one of them separately. For prediction, we sum every output of each tree and get the mean to reduce variance and improve prediction accuracy. [10].

Let

$$\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$$

represent the training data, where $x_i \in \mathbb{R}^p$ are input feature vectors and $y_i \in \mathbb{R}$ are target values.

We sample from $\mathcal{D}$ with size n using the bootstrap approach. The sample data $\mathcal{D}_b$ is used to train the tree T_b, which minimizes the Mean Squared Error (MSE):

$$\hat{T}_b(x) = \sum_{i \in \mathcal{D}_b} (y_i - f(x_i))^2$$

[37].

To increase diversification and reduce correlation between trees, Random Forest selects a subset of features instead of using all features. In each tree, Random Forest considers m features (where $m < p$). [10].

Let

$$\mathcal{F} = \{x_1, x_2, \ldots, x_p\}$$

be the set of all features, and at each split, a random subset $\mathcal{F}_m \subseteq \mathcal{F}$ of size m is chosen for evaluating the best split.

The best split is determined by minimizing the following impurity measure, typically variance for regression:

$$\text{Var}(S) = \frac{1}{|S|} \sum_{i \in S} (y_i - \bar{y})^2$$

where S is the subset of samples at a node, and $\bar{y}$ is the mean target value at that node.

By averaging the forecasts from each individual tree, the Random Forest Regressor makes predictions about the output. The average of the predictions made by each tree, if there are B trees, is the final prediction $\hat{f}(x)$ for a given input x:

$$\hat{f}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{T}_b(x)$$

When compared to individual decision trees, Random Forests are less prone to overfitting because this averaging process reduces the variance of the predictions. [10].

2.5.2 Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is an improved version of Gradient Boosting designed to enhance the accuracy and performance of the original Gradient Boosting method. XGBoost applies several important optimization techniques to achieve this goal[11].

Gradient boosting builds an ensemble of weak learners (typically decision trees), where each learner corrects the errors of the previous ones [40]. in Gradient Boosting we minimize loss function $\mathcal{L}$ for all predictions. Given a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$, where $x_i \in \mathbb{R}^p$ are input features and $y_i \in \mathbb{R}$ are target values, we minimize the following objective function:

$$\mathcal{L}(\hat{y}) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

where $l(y_i, \hat{y}_i)$ is the loss function (e.g., squared error for regression), and $\Omega(f_k)$ is a regularization term. The regularization term penalizes the complexity of the trees f_k, which helps prevent overfitting.

For each weak learner decision tree f_k, the prediction after t iterations is:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i)$$

At each iteration t, we add a new tree f_t to minimize the objective function:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t)$$

XGBoost improves efficiency by using a second-order Taylor expansion of the loss function around the current prediction $\hat{y}_i^{(t-1)}$. The approximate objective function is:

$$\mathcal{L}^{(t)} \approx \sum_{i=1}^{n} \left[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \Omega(f_t)$$

where $g_i = \frac{\partial l(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}}$ is the gradient and $h_i = \frac{\partial^2 l(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)^2}}$ is the Hessian. .

we calculated The gain of a split to optimize the tree structure., let I be the set of data points in the node, and let I_L and I_R represent the data points after splitting

into left and right children. The gain from the split is:

$$\text{Gain} = \frac{1}{2}\left[\frac{(\sum_{i\in I_L} g_i)^2}{\sum_{i\in I_L} h_i + \lambda} + \frac{(\sum_{i\in I_R} g_i)^2}{\sum_{i\in I_R} h_i + \lambda} - \frac{(\sum_{i\in I} g_i)^2}{\sum_{i\in I} h_i + \lambda}\right] - \gamma$$

λ is the regularization parameter used to limit tree complexity. and γ is a penalty for adding a new leaf node [11].

The final prediction for an input x after T iterations is the sum of the predictions from all trees:

$$\hat{y}(x) = \sum_{t=1}^{T} f_t(x)$$

This averaging process helps reduce the variance and leads to better generalization.

2.5.3 Support Vector Regression

The Support Vector Machine (SVM), a type of supervised machine learning algorithm intended for classification tasks, is the foundation of Support Vector Regression (SVR) . Using a nonlinear kernel function ($K(x_i, x_j)$), SVR models data nonlinearity ($\psi(x)$) by mapping it into a higher-dimensional feature space, where it can be approximated linearly as $\phi(x)$. By predicting a function that minimizes an upper bound on the generalized error, the SVR algorithm reduces structural risk. The definition of a linear function $\phi(x)$ in this high-dimensional space is[**?**]:

$$\phi(x) = w \cdot \psi(x) + b,$$

where w and b are the vectors of weights and bias, respectively, and x is the input data. The goal is to minimize $\|w\|^2$ to ensure the flattest possible $\phi(x)$ that maximizes the deviation ϵ from the true observation y_i. The objective function is:

$$\min_{w,\xi,\xi^*} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\hat{n}}(\xi_i + \xi_i^*),$$

subject to:

$$y_i - (w \cdot \psi(x_i) + b) \leq \epsilon + \xi_i,$$
$$(w \cdot \psi(x_i) + b) - y_i \leq \epsilon + \xi_i^*,$$
$$\xi_i, \xi_i^* \geq 0,$$

where ξ_i and ξ_i^* are slack variables, C is the penalization parameter, and $\hat{n}$ is the training data size. Solving the Lagrange dual problem and using the RBF kernel $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$ results in:

$$\phi(x) = \sum_{i=1}^{\hat{n}} (\alpha_i - \alpha_i^*) \exp(-\gamma \|x_i - x_j\|^2),$$

where α_i and α_i^* are the Lagrange multipliers[42].

2.5.4 LASSO

Lasso Regression, or Least Absolute Shrinkage and Selection Operator (Lasso), is a type of linear regression that controls overfitting and the complexity of a model by incorporating L1 regularization. Due to the nature of the L1 norm, this method can set some coefficients to zero, effectively performing feature selection. [43]. The objective function for Lasso is:

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1 \right\}$$

where:

- $\frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$ is the Mean Squared Error (MSE) term, representing the fit of the model to the data.
- $\lambda \|\boldsymbol{\beta}\|_1$ is the L1 regularization term, with $\|\boldsymbol{\beta}\|_1 = \sum_{j=1}^{p} |\beta_j|$ being the L1 norm of the coefficient vector $\boldsymbol{\beta}$. The regularization parameter λ controls the penalty strength.

The L1 regularization term in Lasso has two key effects:

- **Shrinkage**: Reduces the variance of the model by shrinking some coefficients to zero.

– **Feature Selection**: Performs implicit feature selection by setting some coefficients to zero, thereby focusing on the most significant predictors.

2.5.5 Back Stepwise Regression (BSR)

Back Stepwise Regression (BSR) is a method used in regression analysis to select a subset of features from all available predictors in order to improve the accuracy and performance of the model. The goal of this method is to enhance the efficiency and performance of the model by systematically removing less significant features. [37]:

How Back Stepwise Regression Works

1. **Start with All Predictors:** Creating a model with all features.
2. **Evaluate Predictors:** Assess the importance of each feature variable in the model by statistical criteria such as p-values or information criteria (e.g., AIC or BIC).
3. **Remove the Least Significant Predictor:** Remove features that have the least significant contribution to the model, i.e., those with the highest p-value or lowest importance score.
4. **Reassess the Model:** After removing a predictor, refit the model to see how well it performs with the remaining predictors.
5. **Repeat the Process:** Continue removing predictors one by one based on their significance. Each time a predictor is removed, the model is reassessed and updated.
6. **Stop When Criteria Are Met:** The process continues until all remaining predictors meet a predefined criterion for significance, or until removing additional predictors no longer improves the model.

This method is useful for simplifying complex models and improving interpretability.

2.5.6 Clark-West Test

To evaluate the predictive value of El Niño and La Niña events on realized variance, we employ the Clark-West test [44]. This test assesses whether the inclusion of ENSO events improves **significantly** the forecasting performance of our model compared to a model that excludes these events.

The Clark-West test statistic is calculated as follows:

$$f_{t+h} = (RV_{t+h} - \hat{RV}_{A,t+h})^2 - \left[(RV_{t+h} - \hat{RV}_{B,t+h})^2 - (\hat{RV}_{A,t+h} - \hat{RV}_{B,t+h})^2\right]$$

where $\hat{RV}_{A,t+h}$ and $\hat{RV}_{B,t+h}$ denote the forecasted realized variances from Models A and B, respectively, at forecast horizon h. Model A does not include ENSO events, while Model B incorporates these events[3].

We then regress f_{t+h} on a constant term. The Clark-West test rejects the null hypothesis of no significant improvement if the t-statistic of the constant term is significantly positive (one-sided test). To account for potential heteroskedasticity and autocorrelation, we use Newey-West robust standard errors for our significance tests.

Additionally, to understand the impact of El Niño and La Niña events on realized variance, we perform sensitivity analysis. This analysis helps us determine the extent to which these climatic events influence the realized variance beyond the model improvements captured by the Clark-West test.

Results are presented for various forecast horizons: short-term ($h = 1, 2$), medium-term ($h = 4, 8$), and long-term ($h = 16$).

3 Experimental Design

3.1 Transfer Entropy

We use the Information Dynamics Toolkit XL (IDTxl) for Python to calculate transfer entropy and employ the OpenCL Kraskov CMI estimator, which is based

on a GPU estimator. The maximum lags for both sources and the target are fixed at five, which is equivalent to one trading week [45].

3.2 Forecasting

We design different forecasting models by combining various input data. We use the notations TE-RF, TE-SVR, TE-Reg, and TE-XGB to denote forecasting models that utilize TE-based feature selection. Similarly, LS-RF, LS-SVR, LS-Reg, LS-XGB, BSR-RF, BSR-SVR, BSR-Reg, and BSR-XGB refer to models that use the LASSO and BSR feature-selection methods. For models that forecast using the full set of features, we use the notations All-RF, All-SVR, All-Reg, and All-XGB.

For the TE-f, LS-f, and BSR-f models, where $f \in \{\text{RF}, \text{SVR}, \text{Reg}, \text{XGB}\}$, we first apply the respective feature selection method to the data to select lag features with the highest information flow, and then apply the forecasting model f to those features. To study the p-value of the Clark-West test, we consider the smallest average MSE for large models (including ENSO features) to examine the impact of El Niño and La Niña on crops. [42].

4 EMPIRICAL Results

In order to examine the impact of ENSO on corn, wheat, and soybeans, we take samples from our data using a window size of 5. We forecast over different horizons, specifically h = 1, 2, 4, 8, and 16, to assess both short- and long-term effects. To evaluate the significance of these effects, we use the Clark-West test to determine the impact of models that include ENSO data. Additionally, we apply both linear and non-linear models in our analysis.

In Table 2, we present the Mean Squared Error (MSE) for the benchmark model, with results for wheat shown in Panel A, and results for corn and soybeans in Panels B and C, respectively. Table 3 displays the MSE for an alternative model that incorporates ENSO data, comparing it with the benchmark model. Finally, the results of the Clark-West test for each crop are provided in Table 4.

	Horizon (h)					
	h=1	**h=2**	**h=4**	**h=8**	**h=16**	**Average**
Panel A - Wheat						
TE-RF	-	-	-	-	-	-
LS-RF	1.397661	1.296434	1.531960	**1.464017**	1.474047	1.432824
BSR-RF	1.358378	1.451666	1.647996	1.480196	1.615357	1.510319
All-Rf	1.304591	1.284612	1.458044	1.501045	1.640569	1.437772
TE-XGB	-	-	-	-	-	-
LS-XGB	1.553295	1.416256	1.578454	1.613599	1.948475	1.622816
BSR-XGB	1.649869	1.893961	2.105236	3.139294	2.034905	2.164653
All-XGB	1.987390	1.407599	1.635553	1.562295	1.577418	1.634051
TE-SVR	-	-	-	-	-	-
LS-SVR	1.504899	1.445546	1.633786	1.663120	**1.402963**	1.530063
BSR-SVR	1.405975	1.464962	1.698342	1.464500	1.717284	1.550613
All-SVR	1.410027	1.346435	**1.369307**	1.437029	1.437758	1.400111
TE-Reg	-	-	-	-	-	-
LS-Reg	**1.158883**	1.234760	1.476309	1.618154	1.432738	1.384569
BSR-Reg	1.199675	**1.218452**	1.493360	1.604740	1.589841	1.421614
All-Reg	1.169156	1.230245	1.506467	1.571451	1.440270	**1.383518**
Panel B - Corn						
TE-RF	-	-	-	-	-	-
LS-RF	1.49502	1.398256	1.469645	1.73833	-	1.5253
BSR-RF	1.341284	1.179563	1.875059	1.177785	1.182213	1.3512
All-Rf	1.207531	1.237225	1.400227	1.355739	1.279734	1.296091
TE-XGB	-	-	-	-	-	-
LS-XGB	1.183404	1.483751	1.940601	1.482141	-	1.5225
BSR-XGB	1.318922	1.236479	1.728101	1.020236	1.716352	1.4048
All-XGB	1.476981	1.099066	1.336475	1.336926	1.372594	1.324408
TE-SVR	-	-	-	-	-	-
LS-SVR	1.142653	1.046377	1.392611	1.202289	-	1.1950
BSR-SVR	**0.972274**	1.133577	1.191761	1.131068	**1.000708**	1.0855
All-SVR	1.001924	0.982159	**1.030392**	1.136888	1.044110	1.039095

Table 2

	Horizon (h)					
	h=1	**h=2**	**h=4**	**h=8**	**h=16**	**Average**
TE-Reg	-	-	-	-	-	-
LS-Reg	1.070842	**0.954481**	1.080679	**1.011465**	-	**1.0299**
BSR-Reg	1.06533	1.060755	1.331289	1.372515	1.231511	**1.2127**
All-Reg	1.081343	1.091102	1.343796	1.433318	1.293983	1.248709
Panel C - Soybean						
TE-RF	-	-	-	-	-	-
LS-RF	1.450791	1.484654	1.665858	1.497056	1.674368	1.5549
BSR-RF	1.506023	1.499194	1.590931	1.520832	1.755195	1.5744
All-Rf	1.538218	1.468918	1.646294	1.602624	1.814577	1.614126
TE-XGB	-	-	-	-	-	-
LS-XGB	1.704123	1.694247	1.629321	3.190763	2.222754	2.0886
BSR-XGB	1.635409	1.611503	1.641942	1.522021	3.349851	1.9529
All-XGB	1.538203	1.470388	1.481228	1.513335	1.636224	1.527876
TE-SVR	-	-	-	-	-	-
LS-SVR	1.442381	1.434035	1.505507	1.466607	1.596360	**1.4894**
BSR-SVR	1.511817	1.458455	1.510339	**1.429923**	1.681854	1.5189
All-SVR	1.452124	1.440028	1.467344	1.515878	**1.599067**	1.494888
TE-Reg	-	-	-	-	-	-
LS-Reg	**1.260253**	**1.307183**	**1.408789**	1.802901	1.881571	1.5321
BSR-Reg	1.412053	1.545746	1.604531	1.867611	1.822508	1.6509
All-Reg	1.550338	1.644103	1.839553	1.946573	1.862489	1.768611

Table 2: Mean Squared Error (MSE) of the forecasting models for RV Wheat, Corn, and Soybeans across different horizons (h). The table compares several algorithms: TE (Transfer Ensemble), LS (Lasso), BSR (Back Stepwise Regression) ,All features, RF (Random Forest), XGB (Extreme Gradient Boosting), SVR (Support Vector Regression), and Reg (Regression). Each model is evaluated at multiple forecasting horizons (h = 1, 2, 4, 8, 16) and their average performance is provided. Missing values indicate that the algorithm could not select any significant features.

In our empirical model, we observe that the benchmark model has some missing values. This indicates that, for those horizons, the selection algorithm could not identify any significant features with meaningful information for the crops. For all algorithms, the Transfer Ensemble (TE) failed to find significant information that

affects the crops. Among the models, we find that the linear regression model with All features performs better for wheat in average and for corn (LS-Reg) performs better on average . However, for soybeans, the non-linear model Support Vector Regression (SVR) with Lasso regularization proves to be more effective (LS-SVR) on average .

	Horizon (h)					
	h=1	**h=2**	**h=4**	**h=8**	**h=16**	**Average**
Panel A - Wheat						
TE-RF	-	-	-	-	-	-
LS-RF	1.370661	1.306827	1.497114	**1.361528**	1.430663	1.393558
BSR-RF	1.365066	1.451666	1.622796	1.490234	1.671526	1.522258
All-RF	1.284910	1.289488	1.437629	1.384567	1.430693	1.365457
TE-XGB	-	-	-	-	-	-
LS-XGB	1.484713	1.369630	1.586942	1.799649	1.629760	1.572739
BSR-XGB	1.645537	1.893961	1.802449	2.139082	1.889153	1.889836
All-XGB	1.896353	1.397480	1.547814	1.580252	1.522821	1.588944
TE-SVR	-	-	-	-	-	-
LS-SVR	1.486163	1.423945	1.610255	1.598354	1.443688	1.412081
BSR-SVR	1.364613	1.464962	1.683673	1.439122	1.687812	1.527636
All-SVR	1.471430	1.417690	1.479910	1.582592	1.515802	1.493485
TE-Reg	-	-	-	-	-	-
LS-Reg	**1.147098**	**1.215616**	**1.426483**	1.537746	**1.401317**	**1.345652**
BSR-Reg	1.194562	1.218452	1.441473	1.619398	1.514064	1.397590
All-Reg	1.205039	1.249817	1.518563	1.619642	1.447500	1.408113
Panel B - Corn						
TE-RF	-	-	-	-	-	-
LS-RF	1.49502	1.398256	1.469645	1.73833	-	1.5253
BSR-RF	1.264035	1.176266	1.789941	1.180801	1.078611	1.29791
All-RF	1.150435	1.198884	1.365864	1.239554	1.106662	1.212280
TE-XGB	-	-	-	-	-	-
LS-XGB	1.183404	1.483751	1.940601	1.482141	-	1.5225
BSR-XGB	1.588331	1.372525	1.984618	1.046426	1.177109	1.433198

Table 3

	Horizon (h)					
	h=1	**h=2**	**h=4**	**h=8**	**h=16**	**Average**
All-XGB	1.291988	1.071027	1.230424	1.146932	1.508890	1.249852
TE-SVR	-	-	-	-	-	-
LS-SVR	1.142653	1.046377	1.392611	1.202289	-	1.1950
BSR-SVR	**1.04797**	1.221797	1.068982	1.122739	0.983589	1.088415
All-SVR	1.057006	1.023913	**1.033727**	1.046683	**0.944218**	**1.021109**
TE-Reg	-	-	-	-	-	-
LS-Reg	1.070842	**0.954481**	1.080679	**1.011465**	-	1.0299
BSR-Reg	1.082422	1.059227	1.346889	1.303806	1.183302	1.195729
All-Reg	1.086029	1.075403	1.317036	1.339107	1.291490	1.221813
Panel C - Soybean						
TE-RF	-	-	-	-	-	-
LS-RF	1.450791	1.420876	1.651588	1.497888	1.675338	1.539696
BSR-RF	1.506023	1.495303	1.584009	1.569459	1.694420	1.569443
All-RF	1.378614	1.343864	1.578126	1.574364	**1.485880**	**1.472169**
TE-XGB	-	-	-	-	-	-
LS-XGB	1.704123	1.778374	1.608221	1.791879	1.897089	1.755737
BSR-XGB	1.635409	1.895928	1.512679	**1.289810**	3.341492	1.733463
All-XGB	1.371548	1.413978	1.432174	1.587171	1.566784	1.474331
TE-SVR	-	-	-	-	-	-
LS-SVR	1.442381	1.452500	1.492890	1.558177	1.540944	1.485578
BSR-SVR	1.511817	1.548149	1.537195	1.524201	1.664113	1.557895
All-SVR	1.506017	1.504030	1.542600	1.630994	1.513897	1.539508
TE-Reg	-	-	-	-	-	-
LS-Reg	**1.260253**	**1.308087**	**1.397817**	1.921946	1.998565	1.577734
BSR-Reg	1.412053	1.633460	1.677343	1.984034	1.912229	1.723424
All-Reg	1.610278	1.723240	1.911461	2.062221	1.976816	1.856803

Table 3: Mean Squared Error (MSE) of the forecasting models incorporating ENSO data for Wheat, Corn, and Soybeans across different horizons (h). The table compares several algorithms: Lasso (LS) and Back Stepwise Regression (BSR) models with various techniques (RF, XGB, Reg, SVR). Each model is evaluated at multiple forecasting horizons (h = 1, 2, 4, 8, 16) and their average performance is provided. Missing values indicate that the algorithm could not select any significant features.

In the larger model that includes ENSO data, we observe an improvement in the MSE values, especially over longer periods. the linear regression model with Lasso feature selection (LS-Reg) performs better overall for wheat on average. For corn, the non-linear SVR model using all features (All-SVR) shows superior performance, while for soybeans, the non-linear RF model with all features (All-RF) demonstrates the best performance. .

We now examine the p-value of the Clark-West test for wheat (LS-Reg), corn (All-SVR), and soybeans (All-RF).

	Clark-West Test Results				
	h=1	**h=2**	**h=4**	**h=8**	**h=16**
Panel A - Wheat					
LS-RF	0.015566	0.484892	0.036326	0.001569	0.015791
LS-XG	0.098465	0.033538	0.157835	0.002470	0.000163
LS-Reg	**0.066919**	**0.016475**	**0.000844**	**0.000211**	**0.005668**
LS-SVR	0.091593	0.060446	0.087920	0.000550	0.328731
BSR-RF	0.720797	-	0.069676	0.399948	0.002067
BSR-XG	0.149733	-	0.019223	0.000001	0.000095
BSR-Reg	0.210922	-	0.000606	0.759044	0.003038
BSR-SVR	0.046399	-	0.136096	0.003030	0.018528
All-RF	0.06863	0.853263	0.059802	0.000319	0.000902
All-XG	0.048944	0.395771	0.007173	0.071532	0.001805
All-Reg	0.472415	0.902649	0.726769	0.943623	0.402223
All-SVR	0.236031	0.346064	0.089784	0.058031	0.343596
Panel B - Corn					
LS-RF	-	-	-	-	-
LS-XG	-	-	-	-	-
LS-Reg	-	-	-	-	-
LS-SVR	-	-	-	-	-
BSR-RF	0.001132	0.396181	0.003302	0.281908	0.000020
BSR-XG	0.990144	0.519282	0.711777	0.606298	0.000013
BSR-Reg	0.926831	0.235348	0.824492	0.002101	0.024751

Table 4

	Clark-West Test Results				
	h=1	**h=2**	**h=4**	**h=8**	**h=16**
BSR-SVR	0.035113	0.007254	0.000174	0.270018	0.083336
All-RF	0.000499	0.005202	0.005474	0.000676	0.001836
All-XG	0.00103	0.041987	0.001238	0.000499	0.033729
All-Reg	0.493662	0.094813	0.063266	0.001692	0.138479
All-SVR	**0.682726**	**0.964022**	**0.172867**	**0.000495**	**0.000141**
Panel C - Soybean					
LS-RF	-	0.009119	0.039955	0.309709	0.202655
LS-XG	-	0.003358	0.096363	0.000006	0.000071
LS-Reg	-	0.897311	0.044074	0.000253	0.002156
LS-SVR	-	0.592833	0.082554	0.026077	0.005315
BSR-RF	-	0.441056	0.045286	0.447614	0.000010
BSR-XG	-	0.587593	0.001445	0.004895	0.416005
BSR-Reg	-	0.001814	0.012185	0.000405	0.028639
BSR-SVR	-	0.075430	0.543402	0.238404	0.141252
All-RF	**0.001664**	**0.000382**	**0.007727**	**0.031969**	**0.000001**
All-XG	0.002399	0.008091	0.022272	0.200679	0.019592
All-Reg	0.159572	0.030033	0.037659	0.060571	0.147091
All-SVR	0.019267	0.029163	0.127529	0.137976	0.002456

Table 4: Clark-West Test p-values across different horizons (h = 1, 2, 4, 8, 16) for Wheat, Corn, and Soybean. The table compares several models using Lasso (LS) and Back Stepwise Regression (BSR) with Random Forest (RF), XGBoost (XGB), Regression (Reg), and Support Vector Regression (SVR). Missing values indicate that the features for benchmark model and ENSO model are the same.

Based on the Clark-West test results, we can observe that ENSO data has a significant impact on the realized volatility (RV) of crops over longer periods. If we focus on the best prediction algorithm for wheat, as determined by the minimum MSE, we see that the effect of ENSO data is most pronounced in the LS-Reg model at h=8, where the p-value is 0.000211, which is well below the critical threshold of 0.05.

For corn, we consider All-SVR. In this case, ENSO also demonstrates a long-term effect on RV, with the most significant result at

h=16, where the p-value is 0.000141, again below the 0.05 threshold.

For soybean, considering the best-performing algorithm (All-RF), we find that ENSO features also have a more significant effect over longer periods. The most notable result is at

h=16, where the p-value is 0.000001, which is similarly below the critical value of 0.05. This confirms that ENSO data exerts a notable influence on crop volatility, particularly in the long term.

5 Conclusion

We can draw two key conclusions from these results. First, linear feature selection models, such as Lasso (LS) for wheat , tend to outperform non-linear models like Transfer Entropy (TE) in predicting realized volatility (RV) for wheat, corn, and soybean. Second, ENSO data has a clear impact on the RV of these crops, particularly over longer time horizons.

We find that for wheat, Lasso feature selection with linear forecasting regression performs better than the non-linear method. For corn and soybeans, we observe that using all features with a non-linear forecasting function performs better than both linear feature selection and the linear forecaster.

One limitation of Transfer Entropy is that it struggles with linear relationships and indirect effects. ENSO phenomena, such as El Niño and La Niña, may affect RV indirectly through confounders like crop production and supply chain disruptions. In such cases, TE may not effectively capture these indirect relationships, which could explain why it underperforms with this dataset.

6 References

References

[1] Trenberth, K. E., Jones, P. D., Ambenje, P., Bojariu, R., Easterling, D., Tank, K. A., Parker, D., Rahimzadeh, F., Renwick, J. A., Rusticucci, M., Soden, B.,

Zhai, P. (2007). Observations: *Surface and atmospheric climate change, Climate change 2007: The physical science basis. Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change* (pp. 235–336). Cambridge, UK: Cambridge University Press.

[2] Shabbar, A., Khandekar, M. (1996). *The impact of El NiñoSouthern Oscillation on the temperature field over Canada.* Atmosphere-Ocean, 34(2), 401–416

[3] Bonato, M., Çepni, O., Gupta, R., Pierdzioch, C. (2023). Niño, La Niña, and forecastability of the realized variance of agricultural commodity prices: Evidence from a machine learning approach. Journal of Forecasting, 42(4), 785–801. https://doi.org/10.1002/for.2914

[4] Greb, F., Prakash, A. (2015). *Has price volatility changed? Food Outlook, Rome, Italy*: Food and Agriculture Organization of the United Nations

[5] Johnson, T. (2011). *Food price volatility and insecurity.* Council on Foreign Relations (CFR), Available for download from: https://www.cfr.org/backgrounder/foodprice-volatility-and-insecurity

[6] Brockwell, P. J., Davis, R. A. (2016). *Introduction to Time Series and Forecasting* (3rd ed.). Springer

[7] Schreiber, T. (2000). Measuring information transfer. *Physical Review Letters*, 85(2), 461-464.

[8] Corsi, F. (2009). *A simple approximate long-memory model of realized volatility.* Journal of Financial Econometrics, 7, 174–196

[9] Lütkepohl, H. (2005). *New Introduction to Multiple Time Series Analysis.* Springer.

[10] Breiman, L. (2001). *Random forests.* Machine Learning, 45(1), 5–32. https://doi.org/10.1023/a:1010933404324

[11] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785-794).

[12] Smola,A.J. Schölkopf, B. (2004). *A tutorial on support vector regression. Statistics and Computing*, 14(3), 199-222. doi:10.1023/B:STCO.0000035301.49549.88

[13] Andersen, T. G., Bollerslev, T. (1998). Answering the skeptics: *Yes, standard volatility models do provide accurate forecasts.* International Economic Review, 39(4), 885–905.

[14] Atems, B., McGraw, E., Maresca, M., Ma, B. (2020). *The impact of El Niño-Southern Oscillation on U.S. Agricultural Stock Returns.* Water Resources and Economics, 32, 100157

[15] Atems, B., Sardar, N. (2021). *Exploring asymmetries in the effects of El Niño-Southern Oscillationon U.S. food and agricultural stock prices.* Quarterly Review of Economics and Finance, 81, 1–14

[16] Ubilava, D. (2018). *The Role of El Niño Southern oscillation in commodity price movement and predictability.* American Journal of Agricultural Economics, 100(1), 239–263

[17] Hamilton, J. D. (1994). *Time Series Analysis. Princeton University Press.* (Chapter 6 discusses interpolation and frequency conversion in time series analysis)

[18] Seth, Anil (2007). *Granger causality". Scholarpedia.* 2 (7): 1667. Bibcode:2007SchpJ...2.1667S. doi:10.4249/scholarpedia.1667

[19] Hlaváčková-Schindler, Katerina; Palus, M; Vejmelka, M; Bhattacharya, J (1 March 2007). "Causality detection based on information-theoretic approaches in time series analysis". Physics Reports. 441 (1): 1–46. Bibcode:2007PhR...441....1H. CiteSeerX 10.1.1.183.1617. doi:10.1016/j.physrep.2006.12.004.

[20] Jizba, Petr; Kleinert, Hagen; Shefaat, Mohammad (2012-05-15). "Rényi's information transfer between financial time series". Physica A: Statistical Mechanics and Its Applications. 391 (10): 2971–2989. arXiv:1106.5913. Bibcode:2012PhyA..391.2971J. doi:10.1016/j.physa.2011.12.064. ISSN 0378-4371. S2CID 51789622.

[21] Wyner, A. D. (1978). "A definition of conditional mutual information for arbitrary ensembles". Information and Control. 38 (1): 51–59. doi:10.1016/s0019-9958(78)90026-8.

[22] Dobrushin, R. L. (1959). "General formulation of Shannon's main theorem in information theory". Uspekhi Mat. Nauk. 14: 3–104.

[23] Barnett, Lionel (1 December 2009). "Granger Causality and Transfer Entropy Are Equivalent for Gaussian Variables". Physical Review Letters. 103 (23): 238701. arXiv:0910.4514. Bibcode:2009PhRvL.103w8701B. doi:10.1103/PhysRevLett.103.238701. PMID 20366183. S2CID 1266025.

[24] Ver Steeg, Greg; Galstyan, Aram (2012). "Information transfer in social media". Proceedings of the 21st international conference on World Wide Web (WWW '12). ACM. pp. 509–518. arXiv:1110.2724. Bibcode:2011arXiv1110.2724V.

[25] Lungarella, M.; Ishiguro, K.; Kuniyoshi, Y.; Otsu, N. (1 March 2007). "Methods for quantifying the causal structure of bivariate time series". International Journal of Bifurcation and Chaos. 17 (3): 903–921. Bibcode:2007IJBC...17..903L. CiteSeerX 10.1.1.67.3585. doi:10.1142/S0218127407017628.

[26] Pereda, E; Quiroga, RQ; Bhattacharya, J (Sep–Oct 2005). "Nonlinear multivariate analysis of neurophysiological signals". Progress in Neurobiology. 77 (1–2): 1–37. arXiv:nlin/0510077. Bibcode:2005nlin.....10077P. doi:10.1016/j.pneurobio.2005.10.003. PMID 16289760. S2CID 9529656

[27] Montalto, A; Faes, L; Marinazzo, D (Oct 2014). "MuTE: A MATLAB Toolbox to Compare Established and Novel Estimators of the Multivariate Transfer Entropy". PLOS ONE. 9 (10): e109462. Bibcode:2014PLoSO...9j9462M. doi:10.1371/journal.pone.0109462. PMC 4196918. PMID 25314003.

[28] Lizier, Joseph; Prokopenko, Mikhail; Zomaya, Albert (2008). "Local information transfer as a spatiotemporal filter for complex systems". Physical Review E. 77 (2): 026110. arXiv:0809.3275. Bibcode:2008PhRvE..77b6110L. doi:10.1103/PhysRevE.77.026110. PMID 18352093. S2CID 15634881

[29] Lizier, Joseph; Heinzle, Jakob; Horstmann, Annette; Haynes, John-Dylan; Prokopenko, Mikhail (2011). "Multivariate information-theoretic measures reveal directed information structure and task relevant changes in fMRI

connectivity". Journal of Computational Neuroscience. 30 (1): 85–107. doi:10.1007/s10827-010-0271-2. PMID 20799057. S2CID 3012713.

[30] Vicente, Raul; Wibral, Michael; Lindner, Michael; Pipa, Gordon (February 2011). "Transfer entropy—a model-free measure of effective connectivity for the neurosciences". Journal of Computational Neuroscience. 30 (1): 45–67. doi:10.1007/s10827-010-0262-3. PMC 3040354. PMID 20706781

[31] Shimono, Masanori; Beggs, John (October 2014). "Functional clusters, hubs, and communities in the cortical microconnectome". Cerebral Cortex. 25 (10): 3743–57. doi:10.1093/cercor/bhu252. PMC 4585513. PMID 25336598

[32] Kushwaha, Niraj; Lee, Edward D (July 2023). "Discovering the mesoscale for chains of conflict". PNAS Nexus. 2 (7): pgad228. doi:10.1093/pnasnexus/pgad228. ISSN 2752-6542. PMC 10392960. PMID 37533894

[33] Massey, James (1990). "Causality, Feedback And Directed Information" (ISITA).

[34] Permuter, Haim Henry; Weissman, Tsachy; Goldsmith, Andrea J. (February 2009). "Finite State Channels With Time-Invariant Deterministic Feedback". IEEE Transactions on Information Theory. 55 (2): 644–662. arXiv:cs/0608070. doi:10.1109/TIT.2008.2009849. S2CID 13178.

[35] Permuter, Haim H.; Kim, Young-Han; Weissman, Tsachy (June 2011). "Interpretations of Directed Information in Portfolio Theory, Data Compression, and Hypothesis Testing". IEEE Transactions on Information Theory. 57 (6): 3248–3259. arXiv:0912.4872. doi:10.1109/TIT.2011.2136270. S2CID 1172259

[36] Kraskov, A., Stögbauer, H., & Grassberger, P. (2004). Estimating mutual information. *Physical Review E*, 69(6), 066138.

[37] Hastie, T., Tibshirani, R., Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

[38] Kramer, G. (January 2003). "Capacity results for the discrete memoryless network". IEEE Transactions on Information Theory. 49 (1): 4–21. doi:10.1109/TIT.2002.806135

[39] Schapire, R. E. (1999). A brief introduction to boosting. Paper presented at the Ijcai.

[40] J. H. Friedman. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189-1232.

[41] Vapnik, V. (2013). The nature of statistical learning theory. Springer Science Business Media.

[42] Barak, S., Parvini, N. (2023). *Transfer-entropy-based dynamic feature selection for evaluating Bitcoin price drivers*. The Journal of Futures Markets, 1–32. https://doi.org/10.1002/fut.22453

[43] Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1), 267-288.

[44] Clark, T. D., West, K. D. (2007). Approximately normal tests for equal predictive accuracy in nested models. Journal of Econometrics, 138, 291–311

[45] https://github.com/pwollstadt/IDTxl

[46] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(3), 379-423.

[47] Cover, T. M., & Thomas, J. A. (2006). *Elements of Information Theory*. John Wiley & Sons.

[48] Lizier, J. T., & Prokopenko, M. (2013). Local information dynamics in complex systems: A unified approach to capturing information storage, transfer, and modification. *Entropy*, 15(1), 374-398.

[49] Bossomaier, T., Barnett, L., Harré, M., & Lizier, J. T. (2016). *An introduction to transfer entropy: Information flow in complex systems*. Springer.