



The presence of a latent factor in gasoline and diesel prices co-movements

Cosimo Magazzino¹ · Marco Mele² · Claudiu Tiberiu Albuлесcu³ · Nicholas Apergis⁴ · Mihai Ioan Mutascu^{5,6,7}

Received: 1 May 2023 / Accepted: 19 October 2023
© The Author(s) 2023

Abstract

This paper proposes a novel approach to identify the presence of a latent factor in the co-movements of gasoline and diesel prices in the three major European Union economies, (France, Germany, and Italy) using daily data from January 3, 2005, to June 28, 2021. More precisely, we advance an artificial neural networks algorithm estimated through a machine learning experiment through the backpropagation system to show that the neural signal is altered by an element that could coincide with a latent factor in the fuel price co-movements. We consider the role of the fuel tax systems and the connection between gasoline and diesel prices in these countries. The estimations indicate the presence of an unobservable component (the latent factor) in the fuel price co-movements, capable of influencing NN. This result validates the previous findings

✉ Cosimo Magazzino
cosimo.magazzino@uniroma3.it

Marco Mele
marco.mele@unicusano.it

Claudiu Tiberiu Albuлесcu
claudiu.albuлесcu@upt.ro

Nicholas Apergis
napergis@unipi.gr

Mihai Ioan Mutascu
mihai.mutascu@e-uvt.ro

- 1 Roma Tre University, Rome, Italy
- 2 “Niccolò Cusano” University, Rome, Italy
- 3 Politehnica University of Timisoara, Timisoara, Romania
- 4 University of Piraeus, Piraeus, Greece
- 5 West University of Timisoara, Timisoara, Romania
- 6 Zeppelin University, Friedrichshafen, Germany
- 7 LEO, University of Orleans, Orleans, France

reported in the literature, indicating an excess co-movement in fuel prices. It also has implications in terms of fuel price forecasts in the short run.

Keywords Fuel prices co-movement · Artificial neural networks · Latent factor

JEL Classification C45 · Q41

1 Introduction

International oil prices recorded large fluctuations during the last decades, in the context of the global financial crisis, sanitary crisis, geopolitical risks, and high uncertainty. Starting with the 2000s, the West Texas Intermediate (WTI) oil price index recorded a historical pick in May 2008, with 176.89 US dollars/barrel, and a severe decline in April 2020 in the context of Corona Virus Disease-2019 (COVID-19), when the average WTI price was 22.10 US dollars/barrel.¹ These fluctuations have noteworthy implications for the business cycle, financial markets, and other energy prices. However, for the real economy, it is equally important to understand the fluctuations and the co-movements of fuel prices, which are not driven only by crude oil prices, but also by other elements, such as the taxation systems, environmental regulations, profit margins, and the productivity level of oil firms. This is because fuel prices represent an important cost component for all goods and services.

Gasoline and diesel prices are two critical components of the global energy landscape, and their co-movements have far-reaching implications for economies, industries, and consumers around the world. These fuels power the vast majority of vehicles, from personal cars to commercial trucks and even airplanes, making them essential commodities that directly affect people's daily lives and the cost of doing business.

The co-movements of gasoline and diesel prices are closely intertwined due to their shared dependence on crude oil as the primary feedstock. Crude oil serves as the raw material from which both gasoline and diesel are refined, so fluctuations in crude oil prices have a significant impact on the prices of these finished products. As such, understanding the dynamics of gasoline and diesel price co-movements requires examining the factors that influence crude oil prices.

Geopolitical tensions, supply disruptions, and changes in global oil production all contribute to fluctuations in crude oil prices, which, in turn, influence gasoline and diesel prices. For example, conflicts in oil-producing regions, such as the Middle East, can disrupt the supply of crude oil and lead to spikes in prices for both gasoline and diesel. Similarly, decisions by major oil-producing countries, such as OPEC (Organization of the Petroleum Exporting Countries), to increase or decrease oil production quotas might exert a profound impact on the global supply and, consequently, prices (Kpodar and Liu 2021).

Market forces also play a significant role in the co-movements of gasoline and diesel prices. Supply and demand dynamics, as well as economic conditions, affect

¹ <https://www.macrotrends.net/1369/crude-oil-price-history-chart>

consumer behavior and the choices they make regarding the type of fuel they use. For instance, during periods of economic growth, increased industrial activity often leads to greater demand for diesel fuel to power heavy machinery and commercial vehicles. Simultaneously, rising consumer confidence can result in increased gasoline consumption as more people take to the road for leisure travel and commuting.

Environmental policies and regulations further influence the co-movements of gasoline and diesel prices. Governments worldwide have been pushing for cleaner and more efficient transportation options, which has led to changes in fuel formulations and the adoption of emissions-reduction technologies. These changes can have differing impacts on the prices of gasoline and diesel. For instance, the introduction of biofuels or low-sulfur diesel can affect production costs and supply, leading to variations in pricing for both fuels.

Additionally, the exchange rate between currencies can play a role in the co-movements of gasoline and diesel prices, especially in countries that rely heavily on imported crude oil. When a country's currency weakens against the US dollar, which is often used as the standard for oil pricing, it can result in higher prices for both gasoline and diesel as the cost of imported oil rises.

In this context, several recent studies show that fuel price co-movements are not only driven by international oil price shocks (Galeotti et al. 2003; Frondel et al. 2016; Apergis and Vouzavalis 2018; Kisswani 2019; Ogbuabor et al. 2019; Schweikert 2019), but also being influenced by fuel tax systems and financial distress episodes (Albulescu and Mutascu 2021). Moreover, gasoline and diesel prices co-move at different frequencies and time horizons (Mutascu et al. 2022). Elements like increased market transparency (Dewenter et al. 2017), competition and market concentration (Kihm et al. 2016), or profit margins of oil and gas companies (Albulescu and Mutascu 2021), might also influence the fuel price co-movements, which remain very strong, even after the isolation of the international oil price effect (Mutascu et al. 2022). With a different perspective, Matar et al. (2023) investigated the nexus among CO₂ emissions, electricity consumption, economic growth, urbanization, and trade openness for six Gulf Cooperation Council (GCC) countries through the wavelet analysis (WA) approach, while Brady and Magazzino (2018) explored the sustainability and co-movement of government debt in the European Monetary Union (EMU) member countries through a panel data analysis.

Thus, in addition to the role the international oil prices and tax systems play in the fuel price dynamics, the excess co-movements might be explained by a series of unobserved factors. Indeed, Pindyck and Rotemberg (1990) underline the role of latent factors in explaining the persistent co-movement of commodity prices. Similarly, Bai and Ng (2006) raise the question of the presence of latent factors in the co-movements of an important number of economic series. If the business cycle explains most of the price changes for a non-fuel commodity basket, in the case of fuel price co-movements, the supply shocks and contract-specific events present a higher importance.

Starting from this evidence, the goal of this paper is to propose a novel approach to identify the presence of a latent factor in the co-movements of gasoline and diesel prices. On the one hand, the presence of a latent factor explains the excess

co-movements of fuel prices reported by previous papers. On the other hand, the identification of a latent factor helps in predicting fuel prices, especially at shorter time horizons.

Therefore, our first contribution to the existing literature is represented by the use of an artificial neural network (ANN) algorithm estimated through three activation functions in a machine learning (ML) experiment. We resort to a backpropagation system linearizing the activation function using a Hebbian learning procedure, in order to show that the neural signal is altered by an element that could coincide with a latent factor. In the second step, we generate an algorithm able to isolate the neural network (NN) signal at a precise point and use a mirror test to show the presence of a latent factor, capable of influencing the NN. Following Magazzino and Mele (2021), this paper innovates the literature in several ways. First, previous estimates of a latent factor—even though related to a different topic—used inferential methods; while, to the best of our knowledge, this is the first attempt to implement a completely different and more robust methodology able to show this latent factor with these variables. To achieve this objective, a new ANN algorithm has been designed, written, and tested. This mathematical setting is able to analyze the signal proceeding from the inputs to the latent factor and, as a final step, to the target. Furthermore, at the level of computer programming, the results achieved represent the first empirical assessment of the presence of a latent factor between the variables of interest.

Second, we apply this new complex algorithm to identify an unobservable driver of fuel price co-movements in three European countries, namely France, Germany, and Italy. In line with Albulescu and Mutascu (2021), we consider both the gasoline and diesel price co-movements, with and without taxes. To this end, we use weekly data, spanning the period from January 3, 2005, to June 28, 2021. Data were extracted from the Weekly Oil Bulletin of the European Commission. The reasons for analyzing fuel price co-movements in these countries are explained in detail by Mutascu et al. (2022) and refer to the fact that these countries are the largest EU countries, with different fuel tax systems in place, a different structure of energy production and a different approach taken to reduce carbon emissions. The focus on the three EU countries contributes to the validation of previous results reported in the literature, pointing to the favor of excess co-movements in fuel prices, even after the isolation of the effect of crude oil prices. More precisely, our NN estimations indicate the presence of co-movements across all categories of prices, and each variable behaves as a rescuer to correlated cyclic variations. The excess co-movements are validated by the presence of a latent factor.

The rest of the paper describes the ANN empirical model (Sect. 2), the activation of NN (Sect. 3), the identification of a latent factor through fast convolution processes (Sect. 4), and the conclusions together with relevant policy implications (Sect. 5).

2 An ANN empirical model

In this paper, we use an ML approach to estimate the presence of co-movements and a hypothetical latent factor. According to Saqur and Narasimhan (2020), we use a complex process of ANNs estimated through three activation functions in an ML

experiment. In particular, we have programmed four algorithms composed of 7962 command strings. They have been developed in Java and Python. The tests presented below represent a Python implementation of the Mplo3d package.

Our model uses the logical process of ANNs that simulates the brain at the theoretical level. In principle, the process defines the central body as a mathematical model, called “node,” characterized by an activation function (instead, we generated multiples of the activation function), a threshold value, and (possibly) a bias. Each node receives as input a set of signals from the previous units. These signals reach the neuron after being weighed and combined. After having added it through a matrix algebra process, the bias becomes the variable of the activation function, determining the activation (or the lack of activation) of the neuron. Therefore, an NN is a set of nodes arranged in layers, connected by weights. The first layer is called the “input layer,” while the last one is the “target.” The intermediate ones are called “hidden layers.” The latter are usually not accessible by the algorithm, as all the characteristics of the complete network are stored in the matrices that define the weights. The type of network determines the type of connections between the nodes of different layers and between those of the same layer. This model uses a triphasic structure of NNs. In particular, we initially choose a backpropagation system and subsequently linearize the activation function through Hebbian learning. From this process, we generate the search for a latent factor in the binary function of the algorithm. The backpropagation system is based on two phases: i) the forward phase, in which the network is crossed and the error at the output is evaluated as the difference between the correct output and the one to be obtained, and ii) the backward phase, which is the backpropagation proper, where the signal propagates in the opposite direction and the weights are adjusted to reduce the output error. The backpropagation then adjusts the parameters of the NN in the direction of the least error and is usually based on the application of the gradient descent method (GDM), which guarantees to find the local minimum of the cost function, indicating the direction of variation of the error to follow. It is also possible to find a global minimum by repeatedly searching for local minima and comparing them.

The Hebbian method of learning uses the Hebb (1949) postulate related to the self-learning of biological neurons. In other words, if two neurons, one in and one out, are simultaneously activated for a specific time, there is an increase in the ease of transmission of the signal between these two neurons. In this way, the value of the connection weight is increased as the synaptic force increases proportionally to the correlation. Mathematically, the NN activation functions to manage a set of two approaches.

We can analyze the propagation process relative to the Hebbian method in the following way. Starting from the identification in LeCun et al. (1998) and Melchior et al. (2016), we activate the Q-Hebb conscript approach.²

$$h_j = \phi(a_j) := \phi\left(\sum_i^N w_{ij}(x_i - \mu_i) + b_j\right) \quad (1)$$

² An alternative way to generate the gradient training process is provided in the Appendix.

$$\begin{aligned}
 h &= \phi(a) := \phi\left(W^T(x - \mu) + b\right) \\
 &= \phi\left(W^T x + b - \underset{b'}{W^T \mu}\right) \\
 l &= \phi\left(W^T\left(x - \mu + \underset{\mu'}{(W^T)^{-1} b}\right)\right)
 \end{aligned} \tag{2}$$

$$\mu(t+1) = (1 - \nu)\mu(t) + \nu\mu_{\text{batch}}(t) \tag{3}$$

From Eq. 4, we apply the system of Mnih et al. (2015). It represents a propagation process of the Hebbian system through a different perspective. This approach is widely used to simplify mathematical representations of the waves in an NN.

$$Q(s, a)^* = \max_{\pi} \mathbb{E}[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots \mid s_t = s, a_t = a, \pi] \tag{4}$$

$$v_j = \Gamma(v_{mo}) \tag{5}$$

$$\dot{E}(t) = -E(t)/\tau_E + \Theta(t) \tag{6}$$

$$\Delta w(t) = (r(t) + b) \cdot E(t) \tag{7}$$

$$v_o = v_{mo} + v_{do} = v_{mo} + Q(s, A; \Phi_i^-) \tag{8}$$

$$a_b = \arg \max_a (v_o) \tag{9}$$

$$L(\Phi) = \mathbb{E}_{(s,a,r,s')U(D)} \left(r + \gamma \max_{a_b} v_o(s', a_b, \Phi_i^-) - Q(s, a; \Phi_i) \right)^2 \tag{10}$$

3 Activation of the neural network

To evaluate the ideal activation function for the NN to be designed, we test the one related to Eq. (22). We have chosen a process with the sigmoid, the ReLU (Rectified Linear Unit), and the hyperbolic tangent, using three different learning rate values (0.7, 0.09, 0.001) and iterations [ITE] (10^7 , 10^{10} , 10^{12}) to be able to cover most of the possible combinations. After carrying out these analyses, we observe that using both the hyperbolic tangent and the rectified linear functions, it is impossible to describe the structure's behavior, obtaining very high errors (higher than 10%). Such a situation would have rendered the NN useless for the estimation since it would be unfeasible to test both iterations and learning rates. In contrast, the sigmoid function allows us to obtain a high accuracy of the results. Therefore, we have chosen this approach

as the activation of neurons. It would also be possible to use different functions for the different layers of the network; however, given the high levels of errors detected, we decide to simplify the system and rely on the sigmoid for all layers (including the output layer), whose values will be denormalized at the output. Regarding the learning rate, we have built an algorithm capable of heavily affecting both the convergence and computational time. In fact, since Eq. (18) can assume very high values, this situation may cause non-convergence with pronounced oscillations, generating huge errors. By contrast, Eq. (22) could generate shallow values requiring a longer time for training.

Although, in theory, the learning rate must be a number between 0 and 1, we force the algorithm by presetting the minimum value to be 0.01 and the maximum value to be 0.9. The results obtained would be more efficient than the standard ones. After 89 simulations obtained in 19 h of calculation, we observe that values close to or greater than 0.5 lead to an error beyond 1%. To this end, the value of 0.05 is assumed for subsequent simulations, which does not involve significant increases in computational time compared to values close to 0.1, ensuring slightly higher accuracy. The choice of even lower learning rates (such as 0.005), on the other hand, penalizes the NN, making it very difficult to train due to the time it takes and inconvenient to use in case there are many knots.

Figure 1 shows our NN relative to the dataset in Table A1 (Appendix), with the latent factor. As we can see, the NN has three levels of complexity. The first level coincides with a high number of activation neurons concerning the inputs and targets (11, 9, 8, 7, 6, 5–1-6, 7, 9, 10, 12); between the inputs and outputs there is a dispersed neuron (it represents the latent factor); the number of targets is more significant than three and coincides with 50% of the inputs. A network generated in this way requires

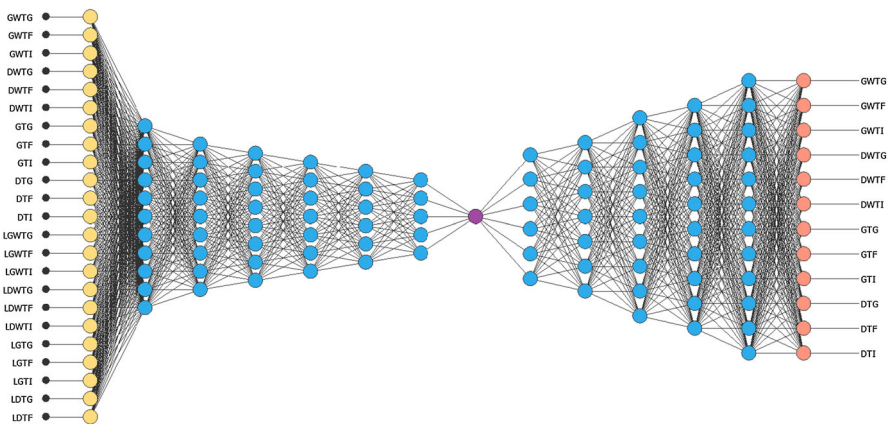


Fig. 1 The NN process. Notes: i) GWTG-Gasoline prices without tax in Germany, GWTF-Gasoline prices without tax in France, GWTI-Gasoline prices without tax in Italy, DWTG-Diesel prices without tax in Germany, DWTF-Diesel prices without tax in France, DWTI-Diesel prices without tax in Italy, GTG-Gasoline prices with tax in Germany, GTF-Gasoline prices with tax in France, GTI-Gasoline prices with tax in Italy, DTG-Diesel prices with tax in Germany, DTF-Diesel prices with tax in France, DTI-Diesel prices with tax in Italy. Source: authors' elaborations in AdNN and Oryx

Table 1 A NN model predicted target expression

NN	Variable	Results
y_1_1 = sigma	GWTG	4.8
y_2_2 = sigma	GWTF	5.6
y_3_3 = sigma	GWTI	4.7
y_4_4 = sigma	DWTG	8.3
y_5_5 = sigma	DWTF	8
y_6_6 = sigma	DWTI	8.3
y_7_7 = sigma	GTG	8.4
y_8_8 = sigma	GTF	9.3
y_9_9 = sigma	GTI	11.2
y_10_10 = sigma	DTG	7.9
y_11_11 = sigma	DTF	6.7
y_12_12 = sigma	DTI	6.3

Source: authors' elaborations in Java

an algorithm capable of separating the process into two distinct phases, while at the same time, processing them jointly to the latent factor.

As shown in Fig. 1, to see how a NN changes behavior by moving nodes from one layer to another while keeping the total number fixed, among the various possible combinations, the one best suited our estimate is the combination that generates 13,485,391,920³ total nodes. This process is achieved by combining both the neurons' mass and the network's stiffness. To obtain what is shown in Fig. 1, we have considered 27 networks with increasing node numbers. Next, we generate an algorithm that estimates the respective mirror NNs and a reference network with $n-1$ nodes per intermediate layer. The training of the networks is performed using the same parameters, in terms of iterations, learning rate, and the two sets, for all the NNs tested, to have directly comparable data. The execution time for each input-target combination averages 620 s. Since we have to carry out a dynamic analysis, we have processed the results obtained in Table 1, through a neural propagation in Java. In other words, we have generated pairs of the same input targets.

Table 1 shows the NN results in a dynamic process. Each variable seems to behave as a rescuer to correlated cyclic variations. The results are the value of the beta coefficient of the NN, which, across all levels, show positive values. The designed network has proved to be very efficient for predicting targets concerning this complex structure that we program. Table 1 confirms the presence of co-movement between prices. Moreover, as we can see from the Sunburst test in Fig. 2, the errors in stiffness and mass of the NN are almost completely absent.

Since there are $n + 1$ possibilities for the design of the network different from ours, we verify the hypothetical use of an $n + n$ input and target nodes to be inserted as parameters within the distance between the constraints of the NN. This process is tested through the expected variance process in ML.

³ Result = DRn,k . In this case, k , a positive integer, can also be greater than or equal to n .

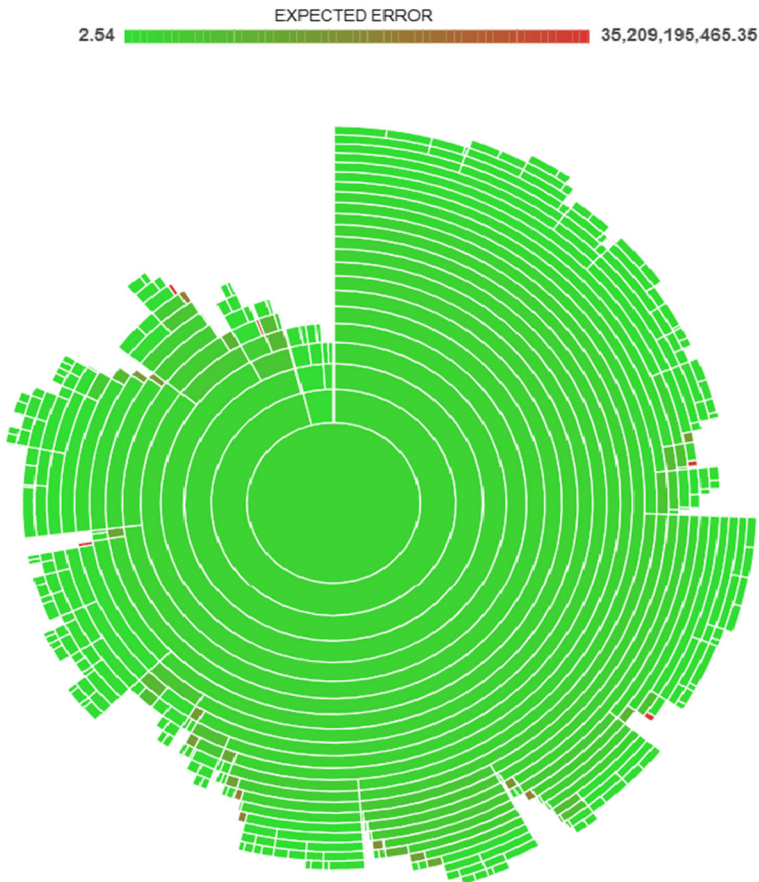


Fig. 2 Sunburst ML test results. Source: authors' elaborations in AdNN and BigML

As we can see from Fig. 3, having fixed the binding parameters of neuronal mass, upper and lower limits of the nodes, and stiffness, we obtain PC7 configurations that satisfy these requirements. The script created in Fig. 3 allows estimating either parameter ranges if their variation is possible, or a constant value. In this way, it is possible to use the hypothetical NN suitable for all the requirements of the basic structure in Fig. 1. This process is possible only if the minimum and maximum values preset in the algorithm (0 and 1) are respected. They are also essential for the standardization process parameters. However, none of them show a value close to 90%. Therefore, there is no different model than the one pre-depicted in Fig. 1. Subsequently, we carry out several tests in order to evaluate the forecast reliability of the constructed NN even in the application phase using models whose geometric characteristics are very different from those of the training set and the validation/testing set tests. In particular, we select 300 different random configurations whose parameters are very close to the values at the extremes of the applicability range. As a result, no errors have been found higher than those obtained previously (see the results of the Sunburst test).

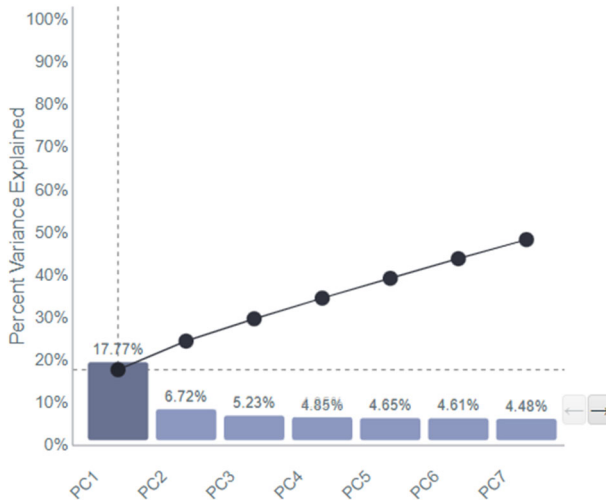


Fig. 3 ML explained variance. Source: authors' elaborations in Apache Mahout

Therefore, we test our NN process following the theoretical postulate that the best training strategy is the one that allows the least possible loss of signals within a NN. To this end, we decide to use two specific tests: the Quasi-Newton method and the Levenberg–Marquardt algorithm.

The Quasi-Newton method calculates an approximation of the inverse Hessian through the gradient present in Eq. (10). The blue line represents the training error and the orange line the selection error. As we can see, at a normalized value of 2.2 concerning 301 epochs, both the training and selection errors are practically null. In fact, the first registers a value of 6.13×10^{-5} and the second of 0.00056. Therefore, since the lines descend until they rapidly vanish as the epochs increase, the first test is correct. The Levenberg–Marquardt algorithm, on the other hand, shows that the training and selection errors in each epoch are reduced to the minimum value of 3.18. Again, the training and selection errors are almost zero: the final training error is 0.0026 and the final selection error is 0.0025.

Finally, in order to isolate the COVID-19 pandemic period, we rerun our model on the sub-period from January 3, 2005, to December 31, 2019. The applied findings do not show any remarkable difference.

4 A latent factor through fast convolution processes

In this section, we demonstrate the existence of a latent factor in the co-movement of gasoline and diesel prices for the three selected countries (Italy, France, and Germany). The presence of co-movements in prices has been reported in previous literature. Now, we ask ourselves if there is a latent factor within our NN. We will use a complex mechanism based on residual NNs to verify this. It is an architecture that generates types of blocks (called residual blocks) that act within an NN through the use of a

new and innovative algorithm. The basic concept of the residual block model is to subject an input (x) to the sequence of operations convolution-ReLU-convolution. In this way, it is possible to have a function $F(x)$, and add the same x to the result. Thus, at the output of this block, we have $H(x) = F(x) + x$. In a traditional feed-forward NN, on the other hand, we would have, in practice, that $H(x) = F(x)$. By concatenating several such blocks, the unsupervised algorithm predicts a specific output. This process represents the result of not learning a direct transformation from the input data to the output, but learning a particular term $F(x)$ to be added to the input data to get to the target by minimizing the residual error. This approach represents what has already been mentioned with the name of residual learning. Another reason for the importance of this particularly effective analysis tool is represented by the fact that in the propagation step of the backpropagation gradient, it is distributed among the signal levels of the NN. This situation makes it possible to significantly counter the problem of gradient degradation, which occurs markedly in networks with many levels and hinders the gradient's flow toward the initial levels of the network, thus slowing down training. The ability to tackle this problem is one of the main features of building intense networks. The proposed model can be viewed in Fig. 5. The presence of waves in the NN and the same size correspond to those of a trained network, which suffers from convolution problems.

As shown in Fig. 5, the NN presents numerous waves of significant magnitude. The neural signal, in other words, despite the presence of training and selection errors close to zero, is altered by an element that could coincide with a latent factor. To obtain the result in Fig. 4, it is necessary to import the additional package Mplo3d into Python. Next, it is necessary to set up the 3d visual analysis and import the results of a residual NN. The process is correct when the computational blocks are present in 2*2 strings.

Once we have verified that the NN presents a convolution trend, we can decide to make visible the latent factor present in our NN. Thus, we generated an algorithm capable of isolating the NN signal at a precise point; this point represents the latent factor. Before proceeding with the illustration of the latent factor, we divided the nodes of the NN into three different blocks.

Through a procedure of mirror parts with opposite signs, we tested an algorithm inside the Mplo3d package capable of dividing the NN in Fig. 1 into two parts: one to the right of the latent factor and one to the left. Subsequently, we isolated the junction point between the neural nodes of the inputs and those of the outputs, later reproduced in the last graph showing the isolated part's effect on the NN. What has been described so far is graphically represented in Fig. 6.

Figure 6 presents three stages of the mirror test process. The first stage (on the left) shows our NN (in Fig. 1) in the complex of signals between the left and right side nodes concerning inputs and outputs. The nodal signal of the inputs (left) records a value that starts from 0 and reaches (by a specular effect) the maximum at the value -1 . This value corresponds to the maximum signal process of the NN that produces the right side and, therefore, the target at value 1.

The central part of the three graphs in Fig. 6, on the other hand, represents the node between the inputs and outputs. At this point, the signal seems to disappear. Finally, combining the two graphs on the left and in the center gives us the last graph (on the right) in Fig. 6. It shows the NN broken at the meeting point between the

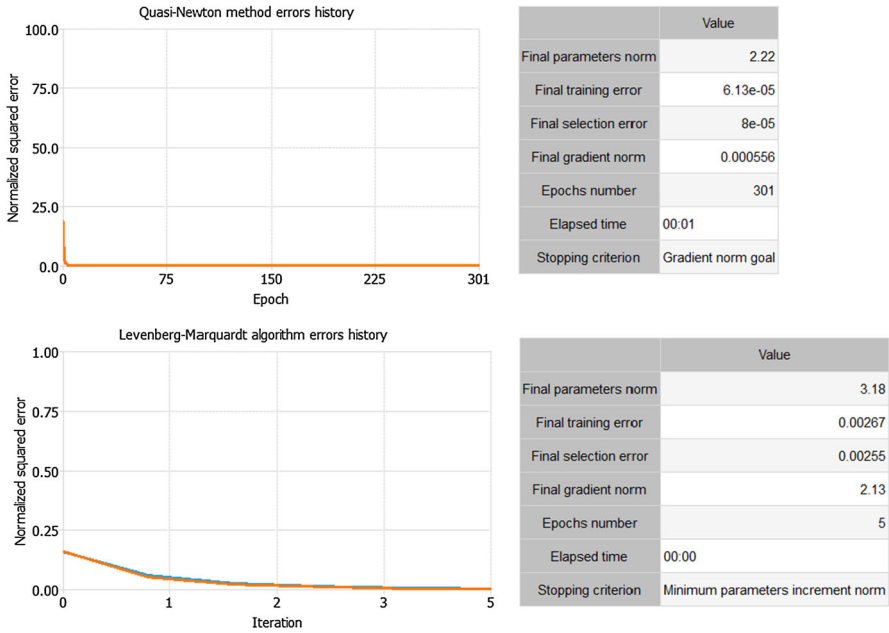


Fig. 4 Training Process tests. Source: authors' elaborations in Oryx and NeuralAd

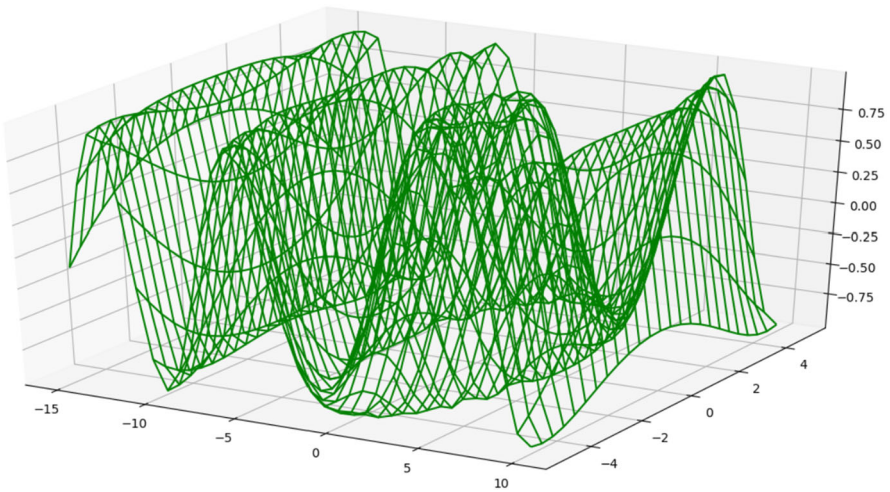


Fig. 5 NN convolution-ReLU-convolution. Source: authors' elaborations in Python (Mpl3d package) and Java

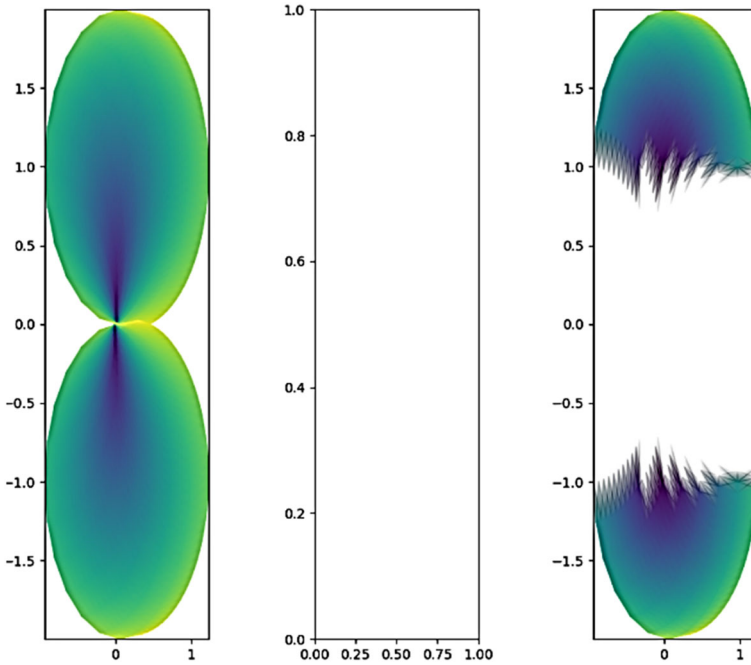


Fig. 6 The NN Mirror test. Source: authors' elaborations in Python (Mpl3d package)

signals of the input nodes, capable of generating the targets. This result most likely highlights the presence of a latent factor capable of influencing the NN. Therefore, we decide to enrich the algorithm with a computational procedure, capable of showing the presence of this factor. According to Magazzino and Mele (2021), we use the concept of non-negativity, which can be written as:

$$K = JY \rightarrow \text{in a multiple level is : } K = J_1 J_n Y \tag{11}$$

Now we apply non-negativity in a binary process:

$$\rho = B \cdot K \Rightarrow B \cdot K = B \cdot (J_1 J_n Y) \rightarrow \min_{J_1 J_n Y, K} \frac{1}{2} \|\rho - B \cdot (J_1 J_n Y)\|_q^2 \tag{12}$$

In (24), we use the projected descending gradient (PDG):

$$K^0, J_1^0, J_n^0, Y^0 \rightarrow \overline{K}^k = K^k - \delta \left(B^t \bullet \left(B \bullet K^k - \rho \right) \right) = \tag{13}$$

$$= (K^{k+1}, J_1^{k+1}, J_n^{k+1}, Y^{k+1}) \rightarrow \min_{K, J_1 J_n, Y} \|\overline{K}^k - K\|_q^2 \tag{14}$$

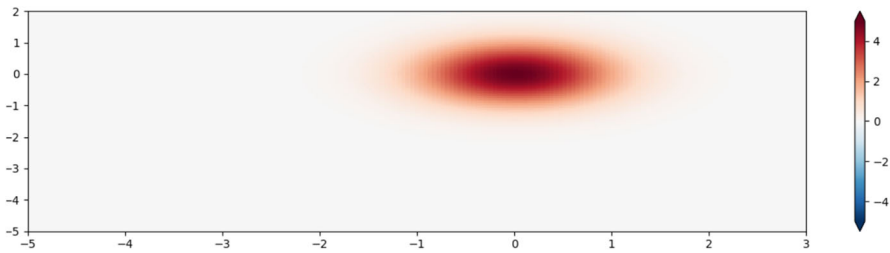


Fig. 7 The latent factor. Source: authors' elaborations in Python and Java

We apply the mirror process in (14):

$$J_1^{k+1} = L_+(J_1^k - (J_1^k J_n^k Y^k - K^{k+1})(J_1^k J_n^k Y^k K^k)! \quad (15)$$

$$J_n^{k+1} = L_+(J_n^k - (J_1^k)! (J_1^k J_n^k Y^k - K^{k+1})(J_{n+1}^k K^k)! \quad (16)$$

Finally, we have:

$$Y^{k+1} = L_+(Y^k)((J_1^k J_n^k)! (J_1^k J_n^k Y^k - K^k)) \quad (17)$$

The result obtained from the non-negativity process can be observed in Fig. 6. Figure 7 shows the visibility of the latent factor, as seen from the tests carried out in Figs. 5 and 6. It represents a disturbance process within the NN. We have been able to amplify this process by isolating the autonomous components of the neural nodes. In other words, the presence of a latent factor also plays a role in the co-movements of gasoline and diesel prices in the countries under consideration.

However, although this result represents a novelty in the literature, we are not able to explain why the latent factor is positioned in the upper right part of the neural signal box. We are supposed to observe the latent factor set in the center. In particular, we think of obtaining a latent factor with coordinates $(-1; -1)$. Instead, as we can see from Fig. 7, it has $(0; 0)$ coordinates. Such a result would have justified the following hypothesis: the latent factor is activated at the point where the neural signals of the inputs begin to activate those capable of generating the targets. Furthermore, since the algorithm we have developed uses a NN with memory, each effect represents the result of what “the NN remembers.” Therefore, the expected value would have coincided with $(-1; -1)$.

5 Conclusions and policy implications

The goal of this paper was to analyze fuel price co-movements resorting to an ANNs algorithm estimated through a ML experiment. Starting from the previous results reported in the literature, we included in our NN gasoline and diesel prices, with and without taxes, in order to identify those co-movements. The additional goal was to

identify an unobservable component that explains excess co-movements documented in previous papers. Resorting to a backpropagation system and a mirror test, we documented that the neural signal was altered by an element that could coincide with a latent factor in the fuel price co-movements. This latent factor was not only represented by crude oil prices, but also by cost competition and the profit margins of oil firms.

We designed, programmed, and tested a new ANNs algorithm able to capture and graphically visualize the latent factor in the co-movements of gasoline and diesel prices in the three bigger EU countries. Moreover, this is the first paper in the related literature that uses an AI approach, and it provides a graphical 3D representation of the latent factor.

In terms of policy implications, policymakers should carefully monitor the oil firms' competition and profit margins as their business responses can be also the result of speculative behavior. On the one hand, in such a context, an adaptive price policy is more than required by limiting either the level of retail oil prices or the oil firms' mark-up margins. In contrast, the movements of international oil prices should be a notable signal, indicating future-induced reactions of the internal market, often being accompanied by a speculative behavior. Therefore, such expectations should be strongly considered in order to especially counteract the imported rise of oil international prices by offering price subsidies for retail chains or monitoring those related prices through regulations.

However, from a technical point of view, the coordinates of the latent factor were unusually placed in the upper right part of the neural signal box. We could only explain this result by the fact that the latent factor was activated at the point where the neural signals of the inputs begun to activate those capable of generating the targets. This represents a limitation of our research. Therefore, future research, through filtering methodology in ML (which should be programmed ad hoc), could isolate these components and better explain the positioning of the NN. Notwithstanding, despite this programming dilemma, it remains proven that the latent factor is visible and affects the signal of the NN.

Funding Open access funding provided by Università degli Studi Roma Tre within the CRUI-CARE Agreement.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The datasets used during the current study are available from the website and are available on request.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Appendix

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m h_{\theta}(x^{(i)} - y^{(i)})^2 \rightarrow \min_{(\theta_0, \theta_1)} J(\theta_0, \theta_1) \quad (\text{A1})$$

In (1), we add the GD as:

$$\theta_j \leftarrow \theta_j - \alpha \frac{\delta}{\delta \theta_j} J(\theta) \quad (\text{A2})$$

In (2), we use the Cauchy approach, so that:

$$\theta_j = \theta_j - \alpha \frac{\delta}{\delta \theta_j} J(\theta_0 - \theta_1) \quad (\text{A3})$$

In (A3), according to Chen et al. (2013), we apply the six principles of an NN, and we have:

$$f(x, \theta) = o^L = o^1 = \sigma(a^1) = \sigma(W^1 o^1 + b^1) \rightarrow 1 = [1, 2 \dots L], a^1 = W^1 o^1 + b^1, o^0 = x \quad (\text{A4})$$

With n layer network: $L = n \rightarrow b^1 = 0$ our functions are:

$$f(x, \theta) = o^n = \sigma(W^n o^1 \sigma(W^{n-1} x)) \quad (\text{A5})$$

If σ is not linear:

$$\sigma = (x) = \frac{1}{1 + \exp(\exp(-x))}; \sigma'(x) = \sigma(x)(1 - \sigma(x)) \rightarrow \vartheta(-1; 1) \text{ is a tanh function} \quad (\text{A6})$$

$$\tanh(x) = \frac{e^{\uparrow} x - e^{\uparrow} -x}{e^{\uparrow} x + e^{\uparrow} -x}; \tan h'(x) \rightarrow \text{Relu}(x) = \begin{cases} x; & x > 0 \\ 0; & x \leq 0 \end{cases} \quad (\text{A7})$$

In (7)

$$MSE(\theta) = \frac{1}{2N} \sum_{n=1}^N II f(x_n, \theta) - t_n II_2^2 \quad (\text{A8})$$

where \rightarrow

$$f(x_n, \theta) - t_n II_2^2 \quad (\text{A9})$$

is the squared in the normal Gaussian. The GDM in (A8) can be written as:

$$\theta_{\vartheta} = \theta - \eta \nabla_{\theta} \rightarrow \nabla_{\theta} \text{ is the gradient respect to } \theta \quad (\text{A10})$$

In (A10), the backpropagation is equal to

$$\frac{\partial E}{\partial W_{ij}^l} = \frac{1}{N} \sum_{n=1}^N f(x_n, \theta) - t_i \frac{\partial}{\partial W_{ij}^l} f(x_i, \theta) \tag{A11}$$

Now, we develop the process as

$$\frac{\partial}{\partial W_{ij}^l}(x_i, \theta) = \tag{A12}$$

$$\frac{\partial}{\partial W_{ij}^l}(x_i, \theta) = \frac{\partial}{\partial W_{ij}^l} o^L = \frac{\partial}{\partial W_{ij}^l} \sigma(a^L) = \sigma'(a^L) \frac{\partial}{\partial W_{ij}^l}(a^L) = \sigma'(a^L) \frac{\partial}{\partial W_{ij}^l}(W^L o^L b^L) \tag{A13}$$

$$= \sigma'(a^L) W^L \frac{\partial E}{\partial W_{ij}^l} o^{L-1} = \sigma'(a^L) W^L \frac{\partial E}{\partial W_{ij}^l} \sigma(a^{L-1}) = \sigma'(a^L) W^L \sigma'(a^{L-1}) W^{L-1} \frac{\partial E}{\partial W_{ij}^l} o^{L-n} \tag{A14}$$

$$a'(a^L) W^L \sigma'(a^{L-1}) W^{L-1} \frac{\partial E}{\partial W_{ij}^l} o^1 = \prod_{K=l+1}^L (\sigma'(a^K) W^K) \frac{\partial}{\partial W_{ij}^l} o^1 \frac{\partial}{\partial W_{ij}^l} o^n \tag{A15}$$

$$= \frac{\partial}{\partial W_{ij}^l} o^1 [\sigma'(W^1 o^{1-n} + b^1)] = \sigma'(a^1) \frac{\partial}{\partial W_{ij}^l} (W^1 o^{1-n} + b^1) = \sigma'(a^1) o_{j,n}^{l-1} \tag{A16}$$

In (A16), we can calculate the error term

$$\frac{\partial E}{\partial W_{ij}^l} = \left[\frac{2}{N} \left((f(x^n), \theta) - t^i \right) \prod_{K=l+1}^L (\sigma'''(a^K) W^K \sigma'(a^l) o_j^{l-1}) \right] D_{ij}^L \tag{A17}$$

If in (A17) we generate an NN propagation effect, we have:

$$D_{ij}^L = \frac{2}{N} f(x_i, \theta) - t_i \rightarrow D_{ij}^l \sigma'(a^l) W^1 D_{ij}^{l+1} = \frac{\partial E}{\partial W_{ij}^l} D_{ij}^l \sigma'(a^l) o_j^{l-1} \tag{A18}$$

Now, we use in (A18) the Martens and Sutskever (2011) approach:

$$\frac{\partial E}{\partial \theta_i^l} = D_{ij}^l \rightarrow \begin{matrix} D^L & o^L & t \\ D^1 & W^1 & o^1 \\ o^{1-n} & b^1 & D^{1-n} \end{matrix} \tag{A19}$$

In (A19) we implement a gradient training process, so:

$$\frac{min}{\theta} = E(f([x^{T-Nt}, \dots, x^{T-Nt}], \theta), t) \rightarrow E(\theta) \frac{1}{2} II f(x[x^{T-Nt}, \dots, x^{T-Nt}]), \theta) - t) II_{\theta}^2 \tag{A20}$$

Then, we activate the Hebbian learning in (A18) and (A20), therefore:

$$\begin{cases} \frac{\partial E}{\partial W_{ij}^l} D_{ij}^l \sigma'(a^l) o_j^{l-1} \\ w_{ij} = \partial x_i x_j \end{cases} = \max_{0 \leq n \leq 1} \frac{\partial E}{\partial W_{ij}^l} D_{ij}^l \sigma'(a^l) o_j^{l-1} \cong w - \varepsilon w_{ij} \quad (\text{A21})$$

Equation (A21) represents our automatic learning ratio in an NN context where a latent factor is estimated.

See Table 2.

Table 2 Summary statistics (weekly data, January 3, 2005 to June 28, 2021)

Statistics	Gasoline Price (tax included)			Diesel Price (tax included)		
	Germany	France	Italy	Germany	France	Italy
Variable	GTG	GTF	GTI	DTG	DTF	DTI
Mean	1404.884	1383.883	1484.325	1231.095	1230.086	1367.962
Median	1387.625	1375.325	1489.735	1210.25	1229.59	1368.96
Maximum	1773	1666.4	1889.81	1569	1533.09	1777.8
Minimum	1077	1015.1	1082.64	940.8	914.48	1004.31
Std. Dev	135.4637	133.3436	180.1089	142.3637	150.027	192.546
Skewness	0.181457	-0.26927	0.024085	0.323572	-0.05752	0.131905
Kurtosis	2.498013	2.237187	2.186691	2.115873	1.848613	2.045397
Jarque-Bera	13.01378	29.572	22.5136	40.71617	45.41185	33.26758
Probability	0.001493	0	0.000013	0	0	0
Sum	1143,576	1126481	1208241	1002111	1001290	1113521
Sum Sq. Dev	14918890	14455554	26373079	16477415	18299079	30141138
Observations	814	814	814	814	814	814

Statistics	Gasoline Price (tax excluded)			Diesel Price (tax excluded)		
	Germany	France	Italy	Germany	France	Italy
Variable	GWTG	GWTF	GWTI	DWTG	DWTF	DWTI
Mean	530.2044	525.1906	565.4286	567.6933	544.3081	594.3023
Median	514.725	518.12	557.585	546.62	529.255	579.055
Maximum	835.42	780.81	833.43	848.09	787.94	851.86
Minimum	250.54	259.55	338.2	325.4	322.47	354.85
Std. Dev	109.8287	104.9496	105.1521	116.2709	111.7376	116.6087
Skewness	0.23073	0.100634	0.310605	0.354955	0.255774	0.381996
Kurtosis	2.55767	2.296229	2.342128	2.166815	2.049928	2.214476
Jarque-Bera	13.85839	18.17264	27.76752	40.6379	39.48973	40.72477
Probability	0.000979	0.000113	0.000001	0	0	0
Sum	431586.4	427505.2	460258.9	462102.3	443066.8	483762.1
Sum Sq. Dev	9806685	8954724	8989304	10990889	10150547	11054847
Observations	814	814	814	814	814	814

Data are extracted from the Weekly Oil Bulletin of the European Commission

References

- Albulescu CT, Mutascu MI (2021) Fuel price co-movements among France, Germany and Italy: a time-frequency investigation. *Energy* 225:120236
- Apergis N, Vouzavalis G (2018) Asymmetric pass through of oil prices to gasoline prices: evidence from a new country sample. *Energy Policy* 114:519–528
- Bai J, Ng S (2006) Evaluating latent and observed factors in macroeconomics and finance. *J Econ* 131:507–537
- Brady GL, Magazzino C (2018) Sustainability and co-movement of Government Debt in EMU Countries: a panel data analysis. *South Econ J* 85(1):189–202
- Chen CLP, Liu YJ, Wen GX (2013) Fuzzy neural network-based adaptive control for a class of uncertain nonlinear stochastic systems. *IEEE Trans on Cybernetics* 44(5):583–593
- Dewenter D, Heimeshoff U, Lüth H (2017) The impact of the market transparency unit for fuels on gasoline prices in Germany. *Appl Econ Lett* 24:302–305
- Frondel M, Vance C, Kihm A (2016) Time lags in the pass-through of crude oil prices: big data evidence from the German gasoline market. *Appl Econ Lett* 23:713–717
- Galeotti M, Lanza A, Manera M (2003) Rockets and feathers revisited: an international comparison on European gasoline markets. *Energy Econ* 25:175–190
- Hebb DO (1949) *The organization of behavior; a neuropsychological theory*. Wiley
- Kihm A, Ritter N, Vance C (2016) Is the German retail gasoline market competitive? a spatial-temporal analysis using quantile regression. *Land Econ* 92:718–736
- Kisswani KM (2019) Asymmetric gasoline-oil price nexus: recent evidence from non-linear cointegration investigation. *Appl Econ Lett* 26:1802–1806
- Kpodar, K., Liu, B., 2021. The Distributional Implications of the Impact of Fuel Price Increases on Inflation. *IMF Working Paper*, WP/21/271.
- LeCun YA, Bottou L, Orr GB, Müller KR (1998) Efficient BackProp. In: Montavon G, Orr GB, Müller KR (eds) *Neural Networks: Tricks of the Trade*. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp 9–48
- Magazzino C, Mele M (2021) A dynamic factor and neural networks analysis of the co-movement of public revenues in the EMU. *Italian Econ J* 8:289–338
- Martens J., Sutskever, I., 2011. Learning Recurrent Neural Networks with Hessian-Free Optimization. In: *Proceedings of the 28th international conference on machine learning*, Bellevue, WA, USA.
- Matar A, Fareed Z, Magazzino C, Al-Rdaydeh M, Schneider N (2023) Assessing the co-movements between electricity use and carbon emissions in the GCC area: evidence from a wavelet coherence method. *Environ Model Assess* 28:407–428
- Melchior J, Fischer A, Wiskott L (2016) How to center deep Boltzmann machines. *J Mach Learn Res* 17(99):1–61
- Mnih V, Kavukcuoglu K, Silver D, Hassabis D (2015) Human-level control through deep reinforcement learning. *Nature* 518:529–533
- Mutascu MI, Albulescu CT, Apergis N, Magazzino C (2022) Do gasoline and diesel prices co-move? Evidence from the time-frequency domain. *Environ Sci Pollut Res* 29:68776–68795
- Ogbuabor JE, Orji A, Aneke GC, Charles MO (2019) Did the global financial crisis alter the oil-gasoline price relationship? *Empir Econ* 57:1171–1200
- Pindyck RS, Rotemberg JJ (1990) The excess co-movement of commodity prices. *Econ J* 100:1173–1189
- Saqr, R., Narasimhan, K., 2020. Multimodal Graph Networks for Compositional Generalization in Visual Question Answering. 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.
- Schweikert K (2019) Asymmetric price transmission in the US and German fuel markets: a quantile autoregression approach. *Empirical Econ* 56:1071–1095