National Bank of Ukraine

NBU Working Papers

01/2023

Sparse Warcasting

Mihnea Constantinescu

The NBU Working Papers

The NBU Working Papers present independent research by employees of the National Bank of Ukraine (NBU) or by outside contributors on topics relevant to central banks. The Working Papers aim is to provide a platform for critical discussion. They are reviewed internationally to ensure a high content quality. The opinions and conclusions in the papers are strictly those of the author(s) and do not necessarily reflect the views of the National Bank of Ukraine or of the members of the Board of the National Bank of Ukraine.

This publication is available on the NBU's website at **www.bank.gov.ua**

Address:

9 Instytutska Street

01601, Kyiv, Ukraine

research@bank.gov.ua

Sparse Warcasting[1]

Mihnea Constantinescu[2]

June, 2023

Abstract

Forecasting economic activity during an invasion is a nontrivial exercise. The lack of timely statistical data and the expected nonlinear effect of military action challenge the use of established nowcasting and short-term forecasting methodologies. This study explores the use of Partial Least Squares (PLS) augmented with an additional sparsity step to nowcast quarterly Ukrainian GDP using Google search data. Model outputs are benchmarked against both static and dynamic factor models. Preliminary results outline the usefulness of PLS in capturing the effects of large shocks in a setting rich in data, but poor in statistics.

JEL Classification Codes: C38, C53, C55, E32, E37.

Keywords: nowcasting, quarterly GDP, Google Trends, machine learning, partial, least squares, sparsity, Markov blanket

[1] I would like thank Ugo Panizza, Cedric Tille and participants of the BCC 10th Annual Conference for their helpful comments and suggestions.

[2] National Bank of Ukraine and University of Amsterdam.
Email: mihnea.const@gmail.com; m.constantinescu@uva.nl.

1 Introduction

The 2022 russian invasion has been a shock of varying regional and temporal intensity, with a highly heterogeneous impact on the Ukrainian economy. While the initial stages of the assault increased uncertainty and affected Ukrainian economic activity as a macro shock, subsequent internal migration from east to west and south-west gave way to a much more nuanced pattern. Deep and long lasting contractions in the regions experiencing intense military aggression and constant population outflows were contrasted by shorter and shallower cycles in the regions acting as emigration gateways to the European Union. The launch of the Black Sea Grain Initiative further revived the southern oblasts.[3]

An immediate effect of the invasion was the introduction of Martial Law on 24 February. As a result, data gathering and processing by Ukrainian state statistical agencies were virtually suspended in the first quarters of 2022. The lack of up-to-date economic data was widespread, affecting both the functioning of the national statistical agency as well as that of private surveying companies.

The current context differs from the COVID-19 period. Then, in both developing and developed economies, data gathering witnessed a robust extension in both scope and coverage, with many high-frequency alternative data sources being used to complement official hard and soft data. In contrast, the full-scale war has virtually frozen all sampling and processing of information related to economic activity of both public and private entities. For security reasons, access to novel data used during the COVID-19 period on mobility, mobile phone and internet access, electricity and fuel consumption, had also been suspended.

Under these conditions, central banking nowcasting models, which are developed around regular statistical releases, become considerably constrained. These models may still be employed, albeit with little flexibility beyond scenario analysis and comparative exercises using past conflicts as empirical counterparties. Without any readings on consumption, investment and production aggregates or their survey-based micro estimates, assessing the scale and speed of changes in the Ukrainian economy transformed from a well-tuned multistage process into creative exploration.

The first necessary step in this context is replacing hard and soft data with alternative measures expected to correlate well with the variables of interest, in our case quarterly Gross Domestic Product. This implies moving further back on the data creation value chain and finding appropriate alternative inputs and models, the output of which correlate well with GDP. In Constantinescu et al. (2022, 2023) we present the preliminary results of forecasting annual regional GDP using a

[3] https://www.un.org/en/black-sea-grain-initiative

multitude of alternative data sources. Night lights, social media activity and Google search volumes may become valuable substitutes when no official data are present.

2 Literature Review

Standard nowcasting models, as those put forward by Giannone et al. (2008), Stock and Watson (2002, 2012), require the availability of an extended list of variables. The models were developed to distill a large number of potentially useful time-series in a much smaller number of driving factors. These factors, generally described by a multivariate state-space VAR, become inputs in subsequent estimation exercises, such as bridge equations or state-space models. Bańbura et al. (2013), Bok et al. (2018) widely employed specifications. More recent literature leverages high frequency data to extract signals of lower frequency targets. Bayesian alternatives open up new modeling avenues, as in Cimadomo et al. (2022).

The current context, rich in data but poor in statistics, sets up the task of nowcasting quarterly aggregate GDP in terms similar to the above literature, yet with a notable exception. Many hard and soft variables included in the estimation exercises referenced above, are gathered in line with statistical guidelines and have, to varying degrees and via theoretical justifications, a connection to the target variable. Extensions with high-frequency social media or web-search activity build on top of an already existing skeleton of hard and soft variables.

Empirical univariate and multivariate exercises cement the usefulness of these alternative data as well as the need of a pre-selection stage when the list of possibly useful variables grows larger than the sample size. Various alternatives are present in the literature to deal with this issue, ranging from sparse principal components as in Zou et al. (2006), Pena et al. (2021) to sparse dynamic factors as the model put forward in Mosley et al. (2023).

The regularization algorithm proposed, for example, in Ferrara and Simoni (2022), dubbed by the authors *Ridge after Model Selection*, requires the availability of both Google Trends and official variables. The second stage of their procedure employs a Ridge regularization of Google Trends, preselected in the first stage, together with official variables. They show Google Trends improve GDP predictions conditional on hard or soft data being available, with forecasting gains depending on the stage of the economic cycle.

In the current study, no hard or soft data are used in the aggregate nowcasting exercise as none was available in the first quarters of 2022 when the first steps of the current exercise were undertaken. Extensions using cash-usage are nevertheless used in policy applications and briefly presented in the context of a regional Partial Least Squares model.

2.1 Partial Least Squares

At the core of most big-data nowcasting applications one may find algorithms estimating a small number of potential latent factors (with a dynamic structure) driving a large number of observed explanatory variables. In this respect, principal components (PCs) are currently the most widely employed estimation strategy of nowcasting models. Some notable exceptions are the recent applications of PLS in macroeconomic forecasting in Eickmeier and Ng (2011), Cubadda et al. (2013), Groen and Kapetanios (2016), and in finance in the study by Preda and Saporta (2005) who use PLS to forecast stock returns. By leveraging both cross-sectional and time-series information, Kelly and Pruitt (2015) propose a multi-stage algorithm that features PLS regressions as a special case.

Wold et al. (1984), Wold et al. (2006) introduce the partial least square method as an alternative projection method to PCR. Stoica and Söderström (1998) explore conditions under which PCR and PLS produce equivalent results, deriving in the process, asymptotic formulas for bias and variance of the PLS estimator. Helland (1990) further illuminates the relationship between PLS and PCR presenting conditions under which the two methodologies yield similar results. Cubadda and Hecq (2011) build a bridge to PLS using economic terminology and widely employed VAR canonical models. They indicate the improved performance of PLS in estimating sources of autocorrelation commonality in a data-rich environment.

Götz and Knetsch (2019) use well-established bridge equation models augmented with Google search time-series and outline their utility in improving the estimation of both GDP, GDP components and monthly activity indicators. The authors highlight that Google search data may become a possible alternative to surveys in the manufacturing sector. PLS is employed in this study but only as an intermediary step to extract relevant factors. More recent publications leverage new machine-learning models in conjunction with large volumes and varieties of online search data to nowcast GDP growth rates, as in Dauphin et al. (2022).

3 Data Description

Target variables are deflated quarterly changes in aggregate GDP. The initial estimation timeframe is 2012 to 2021, reduced to 2013 to 2021 once appropriate lags are accounted for. The explanatory variables are a large number of Google Trends selected on the basis of prior studies and observed local preferences.[4] The raw daily data is aggregated at a quarterly level. Quarterly changes are then computed based on the quarterly means and medians of monthly values.

Google Topics are available as time series ranging from 1 to 100, for a selected period in time and geographical area, representing the relative ranking of keywords (car) or categories of

[4] See for example the evolution of searches on work.ua, Ukraine's largest online job portal; a close competitor is the job section of olx.ua. Searches on these two sites correlate strongly with the topic "Labor".

keywords (car, automobile, etc.). Searches of the same keyword in different languages used in a given geographical areas are considered together. Low search volumes are set to 0 to preserve the anonymity of searches. A value of 100 represents the most searched for term in the selected (time frame, geography) tuple. The widespread coverage of topics and categories allows for a possible matching of Trends in many official statistical series. The empirical relevance has already been established for a wide number of Google Trends, see for the earliest studies Ettredge et al. (2005), Choi and Varian (2009) or McLaren and Shanbhogue (2011). Labor market dynamics implied by Google Trends are presented in McLaren and Shanbhogue (2011) who use the volume of online searches to track labor and housing markets in the United Kingdom. Askitas and Zimmermann (2009) validate the use of internet searches during the 2008 Great Recession to better pinpoint the contraction in the German labor market.

Figure 1. Selected Google Trends for Ukraine

Figure 1 shows quarterly changes in Trends tracking searches related to Unemployment, Inflation, Travel and Export.[5] For example, searches related to Inflation reflect the dynamics of large

[5] Data wrangling and exploratory data analysis are performed using the R packages *tidyverse* by Wickham et al. (2019), *fabletools* by O'Hara et. al and *tsibble* by Wang et al. (2020).

inflationary swings associated with the 2014 Revolution of Dignity and the subsequent profound reforms of the NBU. Note that overall, following the introduction of Inflation Targeting in 2016, changes in inflation searches have on average become negative reflecting a decrease in the relative interest in the topic. In retrospect, the post-Covid robust increase in search volumes may have foreshadowed the above target inflationary pressures flaring already in 2021. The Covid- and war-induced spikes in Unemployment and, post-Covid strong rebound in Travel are further preliminary indications of the ability of Google Trends to recompose the mosaic of economic information present in official statistics.

The nature of the available data requires a new methodological lens. Most models and applications referenced above use variables with a proven track record in nowcasting and forecasting, and are shown to load on the latent factors under a variety of macroeconomic contexts. In the current case, one should expect a much larger number of variables to be irrelevant or only weakly related to the underlying latent factors. Furthermore, the "large p (number of variables) and small n (sample size)" setting poses its own challenges. Although PLSs have been successfully used in bioinformatic applications with $p \gg n$[6], the risk of overfitting to samples and predictors has been indicated as a challenge, and various regularization techniques have been developed and employed (see details below). In the current setting, overfitting will result in non-generalizabilty (of both variable selection and factor loading estimates), thus leading to poor out-of-sample performance.

4 Methodology

Principal Components, a dimensionality reduction unsupervised algorithm, does not have a predefined target. PCR's goal is to identify a lower rank representation of a high dimensionality matrix X, where many of the vectors may be highly correlated.[7] In a second stage, PCs from the first stage are used as inputs in a linear or non-linear regression model.

The Partial Least Squares algorithm, a supervised algorithm, is designed to account for the possible presence of multi-collinearity among the vectors of X *and* also to reflect their covariance with a selected target (or targets).[8] This is a fundamental difference as the underlying vector space representation and projections of X are computed to reflect their covariance with a target variable.

[6] High dimensional genomic studies regularly employ a number of highly collinear explanatory variables which are several tens of times larger than the number of samples or observations

[7] The PCR algorithm is used for example to compress images, represented as large matrices of 1s and 0s, for transmission over the internet. Via PCR, redundant information is eliminated, reducing the size of the file. In the case of an image, nearby pixels are highly correlated (in terms of color, and so on), the reduced form representation being of much smaller size, appearing almost indistinguishable to the human eye.

[8] In ML jargon, a variable supervises the outcome of the algorithm when the algorithm optimizes over inputs in an attempt to predict as closely as possible the supervising variable

Different target variables will therefore lead to different projections and loadings, depending on the strength of the correlation between X and different targets y.

The PLS model and the associated estimation algorithm are worth exploring in detail, in particular highlighting similarities and differences to the widely used PCR estimation algorithm and its underlying assumptions. A simple simulation is presented in the Appendix. It features a small sample and relatively many explanatory variables driven by two latent factors. It is used to showcase settings in which PCR fails to properly estimate the latent factors, unlike PLS which, despite the small n, recovers their structure to a very good degree.

4.1 The Latent Factor Model

In the PLS algorithm, the structural assumption is that a few latent factors *K* drive both the p vectors in X and univariate observations in *y* (Helland, 1990; Stoica and Söderström, 1998), with $K \ll p$. The objective of the algorithm is to recover both the latent factors as well as the loadings of both X and y on the factors via repeated partial regressions of y on the vectors in X, with regressions ordered according to the strength of the covariance between *y* and x_j. The assumed latent structure is given in the equations 1 to 3 below.

$$T = X \times W^*, \quad X \in \mathbb{R}^{n \times p}, T \in \mathbb{R}^{n \times K}, K < p \tag{1}$$

$$X = T \times P' + \epsilon, \quad P \in \mathbb{R}^{p \times K} \tag{2}$$

$$y = T \times C' + \xi, \quad C \in \mathbb{R}^{1 \times K} \tag{3}$$

Most of the time, for computational purposes, both *y* and X will be scaled and centered. The first equation represents the relation between the latent factor T and X. The scores computed in T are linear combinations of the vectors in X and their respective weights. They summarize the information in the original predictor variables that is most relevant to the response variable y. As will be shown below, the weights W^* are the directions in the predictor variables' space which maximize the covariance with the response variable.

The second equation indicates that X is approximated by the latent factors in T times the loading matrix P and a remainder random noise matrix ϵ.

The third structural equation shows y to depend on the same latent factors T and a random noise component.

Given predictions from the above model for both $\hat{X} = T * P'$ and $\hat{y} = T * C'$, one can cast the problem in terms of the traditional multivariate regression expressing y as a linear model of X:

$$\hat{y} = T \times C' = X \times W^* \times C' = X \times \hat{b}^{OLS} \tag{4}$$

The second equality follows by replacing T with its inflated equivalent in terms of X. The third equality should be read as an identity of the OLS regression parameters from the model $y = X * b + \epsilon$ as a function of the latent factor parameters in equations 1 to 3. Specifically, $\hat{b}^{OLS} = W^* \times C'$.

The graph of a simple model with two latent variables is shown in Figure 2 below. L_1 drives X_1, X_2while X_3 only loads on L_2. The target *y* is driven by both L_1 and L_2.

The structure of the model should reflect, even if at first in a heuristic manner, the presumed link between X and y. In the context of traditional nowcasting and near forecasting exercises, many variables that are included in a prototypical factor model can be presumed to be driven both by the underlying latent factors, and themselves have idiosyncratic components driving y. This is sustained by both theoretical and empirical exercises if one considers labor market variables, data related to firm investment decision, surveys of imports and exports, and so on. In this case, y will be affected by a selected x via its indirect loading on the latent factors and via direct shocks in x. This case would be reflected in the graph with additional arrows pointing from X_1, X_2, X_3 to y.

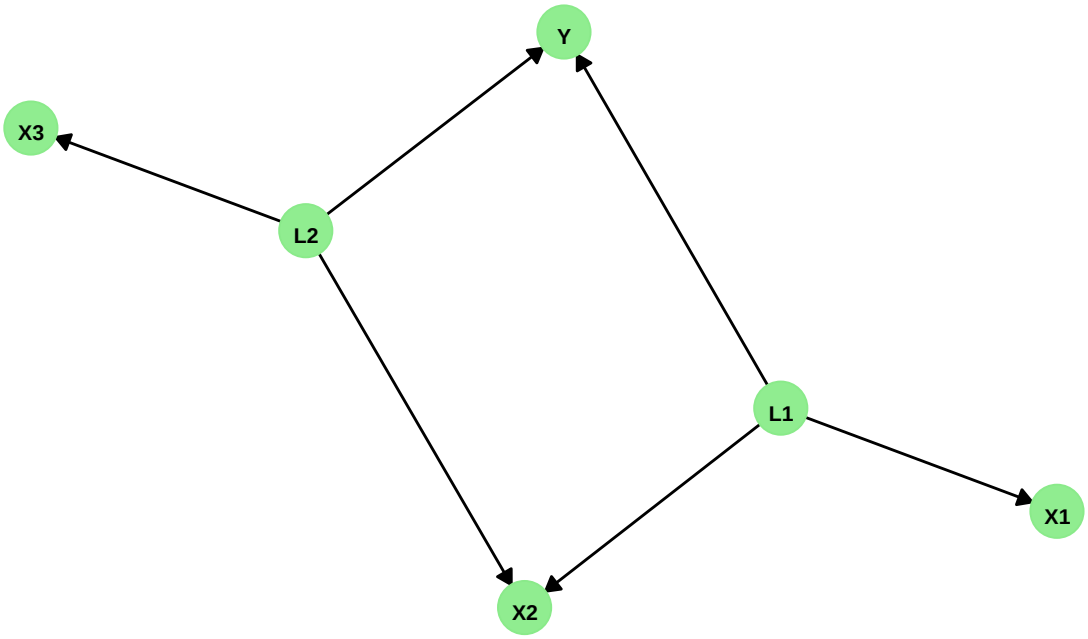

Figure 2. Network Graph of a Simple Latent Factor Model

4.2 The PLS algorithm

The PLS algorithm is described in Wold et al. (1984), Helland (1990), Bastien et al. (2005), Wold et al. (2006) as the iterated use of OLS regressions defined such that the loadings reflect the covariance between the explanatory variables and the target variable. Given a vector of

demeaned variables, the PLS regression model with K latent factors and p centered explanatory variables is given as

$$\mathrm{y} = \sum_{h=1}^{K} c_h \times t_h + \epsilon \tag{5}$$

$$t_h = \sum_{j=1}^{P} w_{h,j}^{*} \mathrm{x}_j \tag{6}$$

with t_h extracted to be independent of each other. The first PLS factor, $\mathrm{t}_1 = \mathrm{X} \times \mathrm{w}_1^*$ is computed as

$$\mathrm{t}_1 = \frac{1}{\sqrt{\sum_{j=1}^{p} \mathrm{cov}(\mathrm{y}, \mathrm{x}_j)^2}} \sum_{j=1}^{p} \mathrm{cov}(\mathrm{y}, \mathrm{x}_j)\mathrm{x}_j \tag{7}$$

The covariance between y and x is the regression coefficient in the simple OLS model of y and scaled x. Following the calculation of the first component, the algorithm proceeds to retrieve the second component with respect to the residuals from the first stage. The remaining variation in y and x_j, purged via t_1, is then used to build the second factor t_2. The algorithm is repeated either for a pre-imposed number of steps (to yield a determined number of factors), or is subject to cross-validation or bootstrap. In the current setting, the bootstrap or CV is employed as no a-priori knowledge is available on the optimal number of factors nor on the amount of penalty. This is particularly important in the quarterly PLS model with regional Trends, where, depending on the specification, well over 2,000 variables may appear in the list of possible explanatory variables (25 regions * 40 Google Trends per region + appropriate autocorrelation structures). For the yearly regional GDP model, the difference between the target sample size (9 yearly GDP changes) and the number of explanatory variables poses similar, though significantly smaller, challenges.

Note the difference in the standard PCR regression in which only the covariance between explanatory variables is considered. Given the same p, explanatory variables (centered and scaled), M Principal Components Z_m are extracted as

$$Z_m = \sum_{j=1}^{p} \lambda_{jm} X_j \tag{8}$$

The λ_{jm} parameters, the principal component loadings, are selected so that the extracted components are orthogonal. In a second stage, an OLS regression is estimated with $M \ll p$ PCs.

$$y_i = \sum_{m=1}^{M} \xi_m z_{im} + \epsilon_i, \quad i = 1, \dots, n \tag{9}$$

Another useful methodological lens through which to consider the PLS and PCR algorithms is to view them as constrained optimization problems. The objective function reflects the supervised nature of PLS as opposed to the PCR, which only looks for lower dimensional representation of X.

The latent components stored in T are computed using the recursive solutions W^* of the following maximization problem:

$$\max_{w} Corr^2(y, X \times w) Var(X \times w) \tag{10}$$

$$\text{s.t.} \, w' \times \text{w} = 1, \quad w' \times \Sigma_X \times w_j, \quad \forall_j = 1, \dots, K-1 \tag{11}$$

Contrast the above with the PCR optimization problem in eq. 12 below:

$$\max_{w} Var(X \times w) \tag{12}$$

$$\text{s.t.} \, w' \times \text{w} = 1, \quad w' \times \Sigma_X \times w_j, \quad \forall_j = 1, \dots, M-1 \tag{13}$$

Alternative algorithms modifying standard PC have been proposed for example in Bair et al. (2006), where a classification problem is tackled via supervised principal components. The proposed algorithm, similar in spirit to PLS, conducts feature selection as a preliminary step to the estimation of the latent factors. The authors highlight the superior performance of their proposed algorithm against a suite of alternatives, including PLS. Nevertheless the standard PLS, lacking a regularization step, is employed as a benchmark. As indicated above, PLS will shrink the parameters of the variables with a weak connection to the latent factor, but will not fully exclude them from the estimation exercise. In this respect, the standard PLS is more similar to performing a ridge regression rather than a Lasso regression.

4.3 Sparse PLS and PCR

Given a mix of potentially useful and less useful predictors, a sparsity step is needed as the inclusion of more data does not guarantee that the performance of the model will be improved. Regularization restrains estimation in the presence of a large number of variables, by screening potentially not relevant or little relevant inputs via a penalty term, the Lasso being one such L_1 example (Tibshirani, 1996). This step is even more important at quarterly frequency given that both target and explanatory variables may have autocorrelation structures which a priori are not necessarily homogeneous with respect to the modelled latent factor. This leads to a several fold increase in the number of possible variables used at the estimation stage. In this setting, the risk of fitting noise looms large and threatens the validity of out-of-sample predictions.

Why is sparsity needed in a PLS regression? Chun and Keleş (2010) indicate challenges to asymptotic consistency of the PLS estimator in "large p small n" contexts, with fixed p_1 relevant and increasing $p - p_1$ irrelevant variables. The intuition for the lack of asymptotic consistency comes from the ridge-like nature of the PLS algorithm. Given that PLS latent factors load on all variables available in X, a larger fraction of irrelevant variables weakens the ability of the algorithm to identify the true factor directions. This is in contrast with the menu of assumptions accompanying the PC estimation of the Dynamic Factor Model. Sparsity is achieved via variable selection in a multitude of ways, depending on the joint specificities of data sample and the machine learning model. Tsamardinos and Aliferis (2003) highlight the close connection between variable selection, the estimated model and the evaluation metric. The authors also present the necessary properties of successful variable selection algorithms, which account for the overall network of dependencies among variables. The Markov Blanket of the associated Bayesian network offers a new avenue to test dependence structures in a large set of variables with the overall goal to identify causal mechanisms in large datasets.

Variable selection, also known as feature selection, is a step in the process of building predictive machine learning models particularly useful for small sample sizes or when model performance may be negatively impacted by possibly noisy inputs. It offers a means to improve model accuracy, reduce complexity, and enhance interpretability. The selection models can be broadly classified into three categories: Filter methods, Wrapper methods, and Embedded methods.

Filter methods rely on a selected statistical measure to assign a score to each variable. This score is then compared to a selected threshold, hard or soft, and based on this, a variable is either included or excluded from a model. Typical score choices for PLS applications are the loading weights from the PLS algorithm or regression coefficients from the subsequent PLS regression. These methods are often univariate and consider features independently of the subsequent model in which they are used, that is, no further model tuning is undertaken once variables have been selected. Despite being straightforward and computationally efficient, which makes them a good choice for large datasets, they may have limited accuracy in selecting the optimal sets with possible interactions. More importantly, filter methods do not consider the overall model performance.

Wrapper methods evaluate different variable subsets based on their predictive performance in a specific machine learning model. The considered model is used to evaluate a combination of features iteratively. Different feature combinations are evaluated in terms of model accuracy, using for example cross-validated R2 or MSRE. Examples of such methods are recursive feature elimination, forward selection, and backward elimination in linear models. They are computationally expensive and model dependent. The chosen features depend on the model, limiting the ability to generalize the selection across different models. The same feature selection wrapper will produce, for example, different optimal subsets for PCR, as compared to PLS or Generalized Additive Models.

Regularization methods are the most common type of embedded methods. Examples of these methods are LASSO (Least Absolute Shrinkage and Selection Operator), Elastic Net, and Ridge Regression, as in Tibshirani (1996). Embedded solutions, such as the algorithm put forward in Chun and Keleş (2010), incorporate the variable selection step within the PLS algorithm. The identification of the optimal subset of variables is performed for each considered factor. Chun and Keleş (2010) propose an L_1 and L_2 penalty in the optimization problem recasting the algorithm in terms of the traditional PLS problem, augmented with additional sparsity, which induces penalties and associated constraints. In case of the univariate PLS regression, the problem is equivalent to imposing a Lasso penalty. In applications, both the amount of penalty and the quantity of the relevant number of factors are determined via cross-validation. Like wrapper methods, these are computationally fast because variable selection and model training are done simultaneously. They may nevertheless overfit in-sample, and by design are model dependent.

Given the current exercise, which method should one use? Mehmood et al. (2020) provide some guidelines, benchmarking several variable selection methods in mainly bioinformatics and genomics PLS applications. Their Monte Carlo analysis points out that predictive performance depends on the amount of collinearity among explanatory variables and the overall number of predictive variables. The compromise faced across the different methods is one between good and stable variable selection accuracy and overall prediction ability. Given the time series exercise, emphasis is placed on Genetic Algorithms Wrappers and Chun and Keleş (2010)'s embedded method.

A further related issue is data sampling variability. Sample variability is a potential issue for topics with low level of searches, especially when the forecasting exercise is performed at high frequencies (weekly or daily). As Google does not return the entire history of searchers, but only a sample, repeated queries for the same (topic, period, geographical area) can return quite different results. This variability depends on the underlying searches, and indirectly, on the population size in a given area. Establishing the relationship between the population size and with in-month sampling variability is the goal of Eichenauer et al. (2021). The authors focus on the necessary conditions to create a daily index, accounting for both sampling variance and the lack of consistency between daily and lower frequency searches. To tackle this issue, in line with their recommendations, repeated samples should be produced for the latest period.

4.4 Genetic Algorithms

GAs are part of a larger group of evolutionary algorithms that employ search heuristics inspired by the process of natural evolution (Holland, 1992; Goldberg and Deb, 1991). A Genetic Algorithm starts with a population of candidate solutions, representing different subsets of potential features. These solutions, often referred to as 'chromosomes,' are usually represented as binary strings, where each bit corresponds to the presence (1) or absence (0) of a feature in the subset. The quality of each initial solution is assessed by a fitness function, in our case, the performance of the PLS model trained to use the corresponding subset of features.

Following the evaluation of the first generation of feature subsets, chromosomes are chosen to form the basis of the next generation. The new generation is assembled via selection. Selection operates under the principle of "the survival of the fittest," where fitter chromosomes, subsets of variables performing well in terms of in-sample PLS fit, have a higher chance of being selected. Techniques for selection include roulette wheel selection, tournament selection, and rank selection. Selected chromosomes undergo crossover, mimicking biological reproduction. Two parent chromosomes are combined to form one or more offspring, each containing features from both parents. To avoid getting trapped in local optima, mutation introduces random modifications to the offspring, enhancing the diversity of the solution space. In feature selection, mutation might involve the random swapping of the presence or absence of a feature. This is repeated over multiple generations until a stopping criterion is satisfied. Genetic Algorithms can handle non-linearity and interactions between variables in an effective manner. They can be computationally demanding, as multiple solutions have to be evaluated over multiple generations and may converge prematurely in populations with little diversity of considered feature combinations. R implementations are available, for example in the *GA* package.[9]

Genetic Algorithms (GA) and their variants have been successfully employed in time series forecasting, for example in Hansen et al. (1999), where neural networks with a GA-determined network structure show substantial predictive improvement over traditional ARIMA models, and in Messias et al. (2016), where GAs are used to optimize load allocation in cloud computing. Hasegawa et al. (1997) and Leardi (2000) are early applications of GAs for selecting variables in PLS regressions.

5 Results

The Dynamic Factor Model is estimated and used as a benchmark alongside the considered PLS variants. The two stage DFM version ala Doz et al. (2011) as well as the quasi-MLE specification of Bańbura and Modugno (2012) are estimated and employed to produce forecasts that are one or two quarters ahead. Bai and Ng (2002)'s IC_2 test plateaus between 4 and 8 components, followed by a further decrease above 10 PCs. A similar profile is returned when considering the IC_1 and IC_3 tests. Given that cross-validation for the PLS versions indicates either 3 or 4 latent factors may be at work,to maintain a good degree of comparability across the modelsall DFM specifications are estimated using 4 principal components. The results of Monte Carlo simulations in Hastie et al. (2013) show CV will select fewer PLS latent factors, as compared to similar PCR exercises of the same data. On a preliminary basis, and considering the higher number of PCs used for example in Giannone et al. (2008), this provides initial support in favor of employing PLS estimations for small data samples.

[9] https://cran.r-project.org/web/packages/GA/vignettes/GA.html

Figure 3 shows the in-sample goodness of fit over the period of Q1 2018 through Q4 2021, comparing the performance of the Dynamic Factor Model, the sparse PLS as in Chun and Keleş (2010) and the GA PLS. To facilitate comparison, a shorter time window is selected for illustration.

In the appendix, the fit for the entire estimation period is shown in Figure 8.

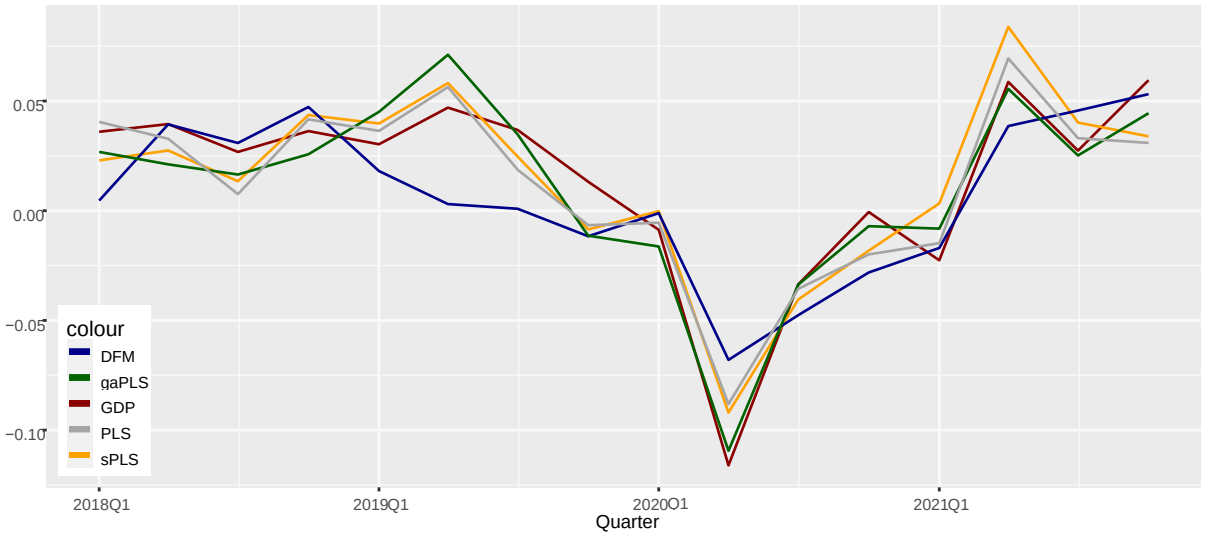

Figure 3. In-Sample Performance of Latent Factor Models

Several important observations follow. Overall, there are qualitatively and quantitatively non-negligible differences among the DFM and PLS alternatives regarding in-sample predictions. The in-sample fit is much closer for PLS and its variants, as compared to DFM estimates.

This should not be surprising as the PLS algorithms specifically target the outcome variable.

But the in-sample fit is a nowcaster's false friend. Relevant criteria are out-of-sample performance, ability in picking turning points, and properly gauging surges and contractions, as they occur.

DFMs produce a similar directional nowcast in the first quarters of 2021 as sPLS and gaPLS. However, they do not properly capture the Q3 slowdown in growth, most likely owing to the strong autocorrelation feature that is uniformly forced on all variables via the state-space AR(1) specification. The standard PLS and sPLS also indicate a slowdown in Q4 2021 when none was observed. The gaPLS points in the same direction as actual GDP, although in a more muted fashion.

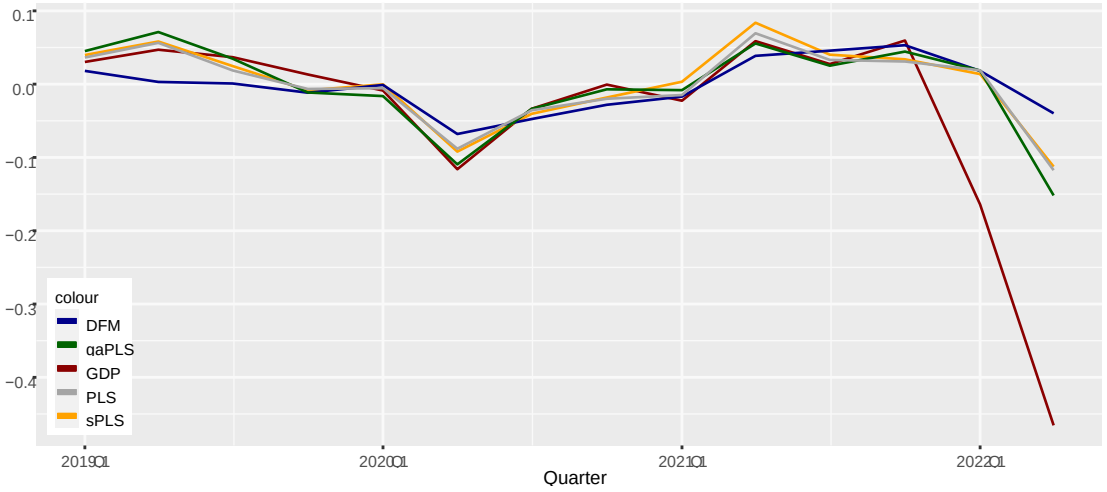

Figure 4. Out-of-Sample Forecasts

The models trained to use 2013 to 2021 data are then employed to produce nowcasts for Q1 and Q2 2022. The Q2 2022 value may be more properly considered a short-term forecast rather than a nowcast as in May 2022 Google Trends data was available, yet no other macrodata had been released since the start of the invasion. Three sets of nowcasts are shown, using a comparable number of latent factors to facilitate performance comparison. In the tradition of ML estimation, this represents a test on previously unseen data. All of the considered models identify the contraction although of a markedly different size.

The regional model that uses regional Google Trends as input variables strengthens the case in favor of shrinkage. Most variables, related to Unemployment, Economy, Inflation and various consumer goods, are selected from economically large regions, such as Kyiv city and Kharkiv, Lviv, and Odesa oblasts. Very few of the smaller oblasts' variables weigh on the evolution of aggregates, according to the Sparse PLS model.

5.1 Variable Selection

Which variables are selected and what are their regression parameters? In Figure 5 one may find the result of the GA wrapper. All estimations are performed with standardized data. Note the high positive parameter associated with "lng L1", the first lag of changes in GDP. The AR(1) component is autonomously picked by the algorithm, as it is part of the general pool of genes. A positive impact comes from contemporaneous changes in searches on "Holiday", "Travel" and "Interest Rate" and lags in "Unemployment", "Labor" and "Interest Rate". High past "Inflation" searches have a strong and negative impact on current GDP, as do changes in "Savings" and "Sony". Past searches related to particular brands may act as consumption proxies. Although preliminary, the results support the general economic intuition that there is a link between the macro variables possibly tracked by Google Trends and their effect on GDP.

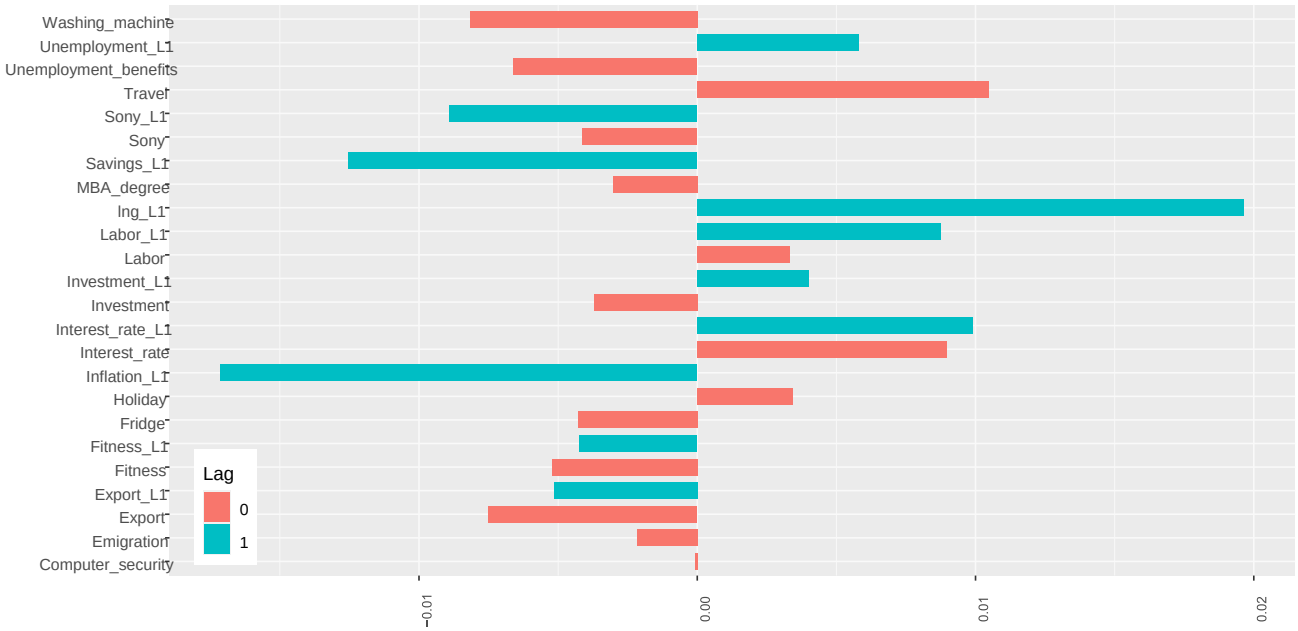

Figure 5. Variable Selection and Parameter Values

In the Appendix, the results of the sPLS algorithms are available in figure 9. Zeros are overlaid to offer a better view of the sparsity structure. Although there is some overlap in terms of selected variables and lags, as well as in the signs of the parameters, they are far from being uniform.

5.2 The Regional Factor Model

The regional factor model is estimated using annual regional GDP growth rates. PLS latent factors for each region are extracted from their respective cash-turnover growth rates. Payment data contain strong and timely signals about the direction of the economy, as shown in Galbraith and Tkacz (2018) and in Aprigliano et al. (2019). Recent studies leverage the ability of ML algorithms to handle large volumes of data to predict short-run macro dynamics, as for example in the work of Chapman and Desai (2022). In particular for Ukraine, cash turnover may be more relevant to track due to much higher cash usage.

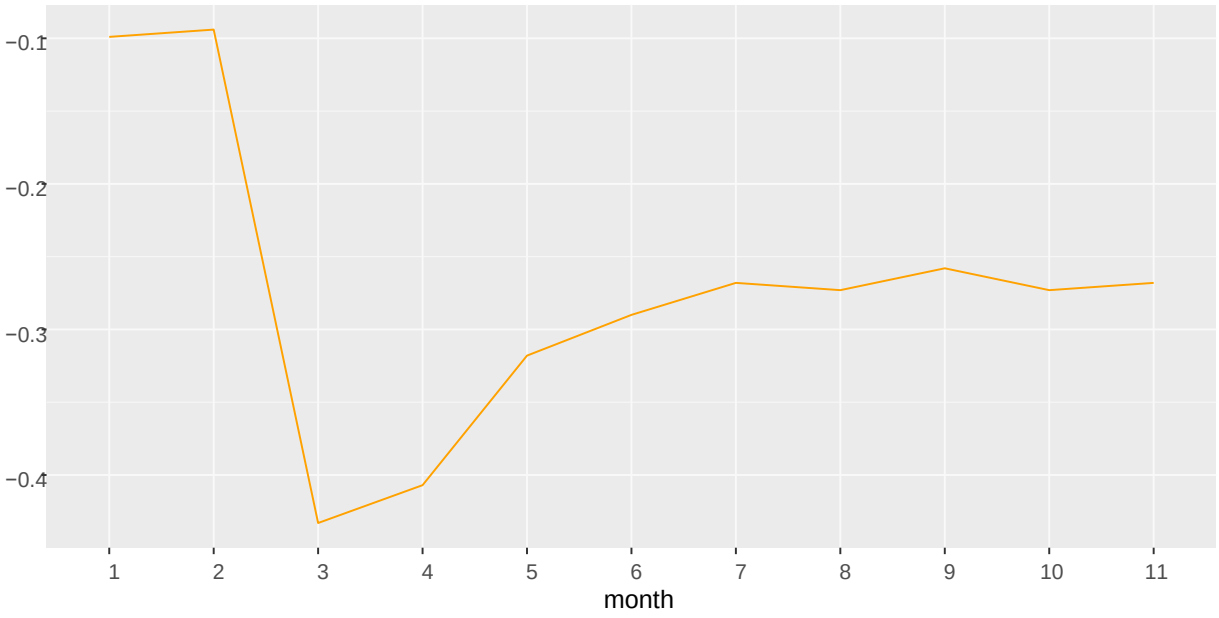

Figure 6. Aggregate Nowcasts – Regional PLS Model

The nowcast is computed using annual bridge equations estimated over the period of 2013 through 2020. Regional nowcasts are then aggregated using 2020 national GDP shares. The model captures well the depth of the March contraction, the mid-year rebound as well as the ensuing dynamics for the year-end. The official y-o-y 2022 contraction equals -29.1 percent.

5.3 Bayesian Blankets and PLS

Bayesian networks, also known as directed acyclic graphical models, are a type of probabilistic graphical model that represent the conditional dependencies among a set of variables (Pearl, 1986). These models use a directed acyclic graph (DAG), where nodes represent variables and edges symbolize direct dependencies between the variables. The absence of an edge represents a specific conditional independence assertion in the joint distribution. Figure 2 is one such example.

In a Bayesian network graph, directed edges typically have associated probabilities. For each node, we attach a conditional probability that quantifies the influence of its parents on the node itself. The entire network then defines a unique joint probability distribution over the variables. Bayesian networks are employed in a multitude of applications, such as prediction, anomaly detection, diagnostics, automated insight, reasoning under uncertainty, and providing compact representations of joint probability distributions in large datasets.

For a given node in a Bayesian network, the Markov Blanket consists of its parents, its children, and any other nodes that share a child with the node (Pearl, 1988). This set of nodes is essential, as it shields the node from the rest of the network. Specifically, given the variables in its Markov

Blanket, a node is conditionally independent of all other nodes in the network. When it comes to variable selection in a dataset, the Markov Blanket of a particular variable is the minimal subset of variables needed for optimal prediction of that variable. This can significantly simplify complex models by reducing the number of variables considered.

The PC Algorithm (Spirtes et al., 2000, 2012), is a widely used method for determining the structure of the Bayesian Network and subsequently identifying the Markov Blanket of a particular node. This constraint-based algorithm learns the structure of the network by systematically testing conditional independence among subsets of variables. The algorithm beings with a fully connected, undirected graph, and proceeds with two key steps: skeleton identification and v-structure orientation. In the first phase, the algorithm tests conditional independence for every pair of nodes, given a conditioning set. If a pair of nodes is conditionally independent, the edge connecting them is removed. This process is repeated with increasing conditioning set sizes until no further edges can be removed.

In the v-structure orientation phase, the algorithm assigns directions to the edges to form a directed acyclic graph (DAG). It identifies structures where two nodes have a common child but lack a direct edge between them, orienting the edges towards the common child. The result is a learned structure of the Bayesian network. To identify the Markov Blanket of a specific node, one needs to find the parents, children, and nodes sharing a child with that node. An extension of the standard PC, the temporal PC algorithm, which accounts for the time series structure of variables, is presented in Petersen et al. (2021).

The temporal PC Algorithm serves as a tool for discovering relationships among time-ordered variables viewed through the lens of Bayesian networks. The temporal partially directed DAG of the Google Trends and GDP data is presented in Figure 7 below. Directed and undirected edges point towards current and past values of GDP, indicating the complex nature of their joint distribution. The directed edge between GDP and its first temporal lag is identified, along with a number of temporal direct and indirect effects. For example, as also uncovered by the Genetic Algorithm, changes in searches related to lagged “Inflation” and present “Export” impact changes in GDP.

What is the PLS algorithm doing differently compared to PCR? By allowing flexibility in the identification of factors, PLS may be better positioned to capture non-linearities among variables. Factors are not constrained to load on current and past values of variables in a uniform AR fashion. In a standard DFM, the AR parameters of the state space equation impose a structure on the relationship between factor representations that may not be faithfully present in the joint probability distribution of variables. Furthermore, sparsity identifies variables with strong predictive power across considered lag structures without the need to explicitly model their autoregressive feature. In the current study, PLS factors load on current and past values of considered variables differently across the identified components, with parameters linked to the strength of their predictive ability.

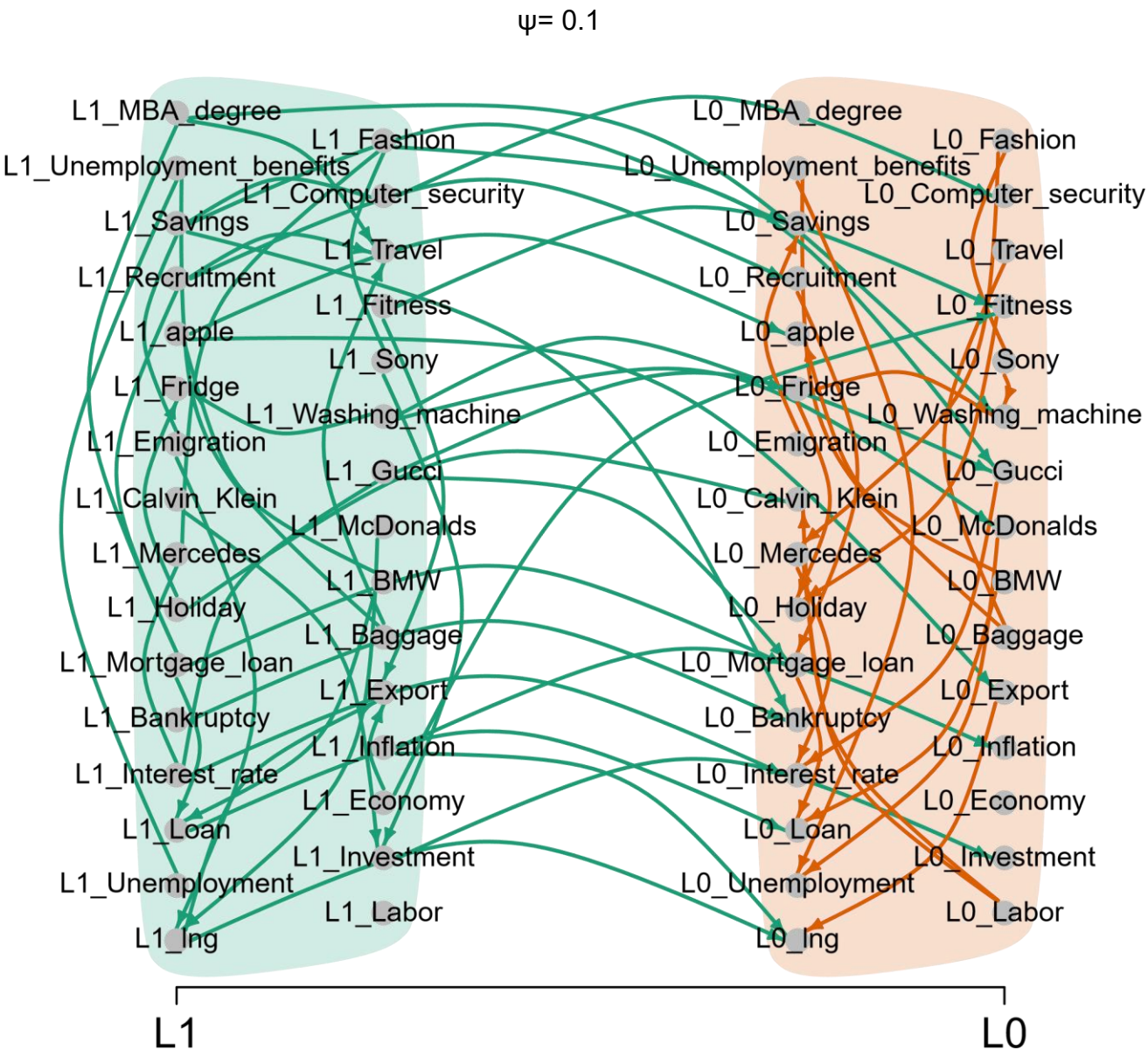

Figure 7. Temporal Bayesian Network Structure

6 Conclusions

The current study provides evidence of the ability of Partial Least Squares in conjunction with Genetic Algorithm variable selection algorithms to estimate economic activity during periods of large shocks. The results are benchmarked against standard Dynamic Factor Models. Google searches contain relevant information for nowcasting quarterly Ukrainian GDP and regional annual GDP at times when no hard or soft data is available. PLS regional latent factors extracted from cash-usage data perform well in nowcasting aggregate GDP, indicating supervised algorithms, in conjunction with data of high economic relevance to the target, produce the best predictive results among the considered methodologies. The relevance of regional data in short-term forecasting of aggregate GDP is an important future research avenue.

References

Aprigliano, V., Ardizzi, G., Monteforte, L. (2019). Using payment system data to forecast economic activityю. International Journal of Central Banking, 15(4), 55-80.

Askitas, N., Zimmermann, K. F. (2009). Google econometrics and unemployment forecasting. IZA Disscussion Paper, 4201. Bonn: Institute of Labor Economics.

Bai, J., Ng, S. (2002). Determining the number of factors in approximate factor models. Econometrica, 70(1), 191–221. https://doi.org/10.1111/1468-0262.00273

Bair, E., Hastie, T., Paul, D., Tibshirani, R. (2006). Prediction by supervised principal components. Journal of the American Statistical Association, 101(473), 119–137.

Bańbura, M., Giannone, D., Modugno, M., Reichlin, L. (2013). Now-casting and the real-time data flow. In Handbook of Economic Forecasting, Ch. 4, pp. 195–237. https://doi.org/10.1016/B978-0-444-53683-9.00004-9

Bańbura, M., Modugno, M. (2012). Maximum likelihood estimation of factor models on datasets with arbitrary pattern of missing data. Journal of Applied Econometrics, 29(1), 133–160. https://doi.org/10.1002/jae.2306

Bastien, P., Vinzi, V. E., Tenenhaus, M. (2005). PLS generalised linear regression. Computational Statistics and Data Analysis, 48(1), 17–46. https://doi.org/10.1016/j.csda.2004.02.005

Boivin, J., Ng, S. (2006). Are more data always better for factor analysis? Journal of Econometrics, 132(1), 169–194. https://doi.org/10.1016/j.jeconom.2005.01.027

Bok, B., Caratelli, D., Giannone, D., Giannone, D., Sbordone, A. M., Tambalotti, A. (2018). Macroeconomic nowcasting and forecasting with big data. Annual Review of Economics, 10, 615–643. https://doi.org/10.1146/annurev-economics-080217-053214

Chapman, J., Desai, A. (2022). Macroeconomic predictions using payments data and machine learning. Staff Working Paper, 2022-10. Bank of Canada. https://doi.org/10.34989/swp-2022-10

Choi, H., Varian, H. (2009). Predicting the present with google trends. Economic Record, 88(s1), 2–9. https://doi.org/10.1111/j.1475-4932.2012.00809.x

Chun, H., Keleş, S. (2010). Sparse partial least squares regression for simultaneous dimension reduction and variable selection. Journal of the Royal Statistical Society Series B: Statistical Methodology, 72(1), 3–25. https://doi.org/10.1111%2Fj.1467-9868.2009.00723.x

Cimadomo, J., Giannone, D., Lenza, M., Monti, F., Sokol, A. (2022). Nowcasting with large Bayesian vector autoregressions. Journal of Econometrics, 231(2), 500–519. https://doi.org/10.1016/j.jeconom.2021.04.012

Constantinescu, M., Kappner, K., Szumilo, N. (2022). Estimating the short-term impact of war on economic activity in Ukraine. Techreport, 1, CEPR. Retrieved from https://cepr.org/voxeu/columns/estimating-short-term-impact-war-economicactivity-ukraine-0

Constantinescu, M., Kappner, K., Szumilo, N. (2023). The warcast index: Nowcasting economic activity without official data. Unpublished Manuscript.

Cubadda, G., Guardabascio, B., Hecq, A. (2013). A general to specific approach for constructing composite business cycle indicators. Economic Modelling, 33, 367–374. https://doi.org/10.1016/j.econmod.2013.04.007

Cubadda, G., Hecq, A. (2011). Testing for common autocorrelation in data-rich environments. Journal of Forecasting, 30(3), 325–335. https://doi.org/10.1002/for.1186

Dauphin, J.-F., Dybczak, K., Maneely, M., Sanjani, M. T., Suphaphiphat, N., Wang, Y., Zhang, H. (2022). Nowcasting GDP – A scalable approach using DFM, machine learning and novel data, applied to European economies. IMF Working Paper, 2022/052. International Monetary Fund. Retrieved from https://www.imf.org/en/Publications/WP/Issues/2022/03/11/Nowcasting-GDP-A-Scalable-Approach-Using-DFM-Machine-Learning-and-Novel-Data-Applied-to-513703

Doz, C., Giannone, D., Reichlin, L. (2011). A two-step estimator for large approximate dynamic factor models based on Kalman filtering. Journal of Econometrics, 164(1), 188– 205. https://doi.org/10.1016/j.jeconom.2011.02.012

Eichenauer, V. Z., Indergand, R., Martinez, I. Z., Sax, C. (2021). Obtaining consistent time series from google trends. Economic Inquiry, 60(2), 694–705. https://doi.org/10.1111/ecin.13049

Eickmeier, S., Ng, T. (2011), Forecasting national activity using lots of international predictors: An application to New Zealand. International Journal of Forecasting, 27(2), 496– 511. https://doi.org/10.1016/j.ijforecast.2009.10.011

Ettredge, M., Gerdes, J., Karuga, G. (2005), Using web-based search data to predict macroeconomic statistics. Communications of the ACM, 48(11), 87–92. https://doi.org/10.1145/1096000.1096010

Ferrara, L., Simoni, A. (2022). When are Google data useful to nowcast GDP? An approach via preselection and shrinkage. Journal of Business and Economic Statistics, 1–15. https://doi.org/10.1080/07350015.2022.2116025

Galbraith, J. W., Tkacz, G. (2018). Nowcasting with payments system data. International Journal of Forecasting, 34(2), 366–376. https://doi.org/10.1016/j.ijforecast.2016.10.002

Giannone, D., Reichlin, L., Small, D. (2008). Nowcasting: The real-time informational content of macroeconomic data. Journal of Monetary Economics, 55(4), 665–676. https://doi.org/10.1016/j.jmoneco.2008.05.010

Goldberg, D. E., Deb, K. (1991). A comparative analysis of selection schemes used in genetic algorithms. Foundations of Genetic Algorithms, 1, 69–93. https://doi.org/10.1016/B978-0-08-050684-5.50008-2

Götz, T. B., Knetsch, T. A. (2019). Google data in bridge equation models for German GDP. International Journal of Forecasting, 35(1), 45–66. https://doi.org/10.1016/j.ijforecast.2018.08.001

Groen, J. J., Kapetanios, G. (2016). Revisiting useful approaches to data rich macroeconomic forecasting. Computational Statistics and Data Analysis, 100, 221–239. https://doi.org/10.1016/j.csda.2015.11.014

Hansen, J. V., McDonald, J. B., Nelson, R. D. (1999). Time series prediction with genetic algorithm designed neural networks: An empirical comparison with modern statistical models. Computational Intelligence, 15(3), 171–184. https://doi.org/10.1111/0824-7935.00090

Hasegawa, K., Miyashita, Y., Funatsu, K. (1997). GA strategy for variable selection in QSAR studies: GA-based PLS analysis of calcium channel antagonists. Journal of Chemical Information and Computer Sciences, 37(2), 306–310. https://doi.org/10.1021/ci960047x

Hastie, T., Tibshirani, R., Friedman, J. (2013). The Elements of Statistical Learning: Data Mining. Inference, and Prediction. London: Springer.

Helland, I. S. (1990). Partial least squares regression and statistical models. Scandinavian Journal of Statistics, 17(2), 97–114.

Holland, J. H. (1992). Genetic algorithms. Scientific American, 267(1), 66–73.

Kelly, B., Pruitt, S. (2015). The three-pass regression filter: A new approach to forecasting using many predictors. Journal of Econometrics, 186(2), 294–316. https://doi.org/10.1016/j.jeconom.2015.02.011

Leardi, R. (2000). Application of genetic algorithm-PLS for feature selection in spectral data sets. Journal of Chemometrics, 14(5-6), 643–655. https://doi.org/10.1002/1099-128X(200009/12)14:5/6%3C643::AID-CEM621%3E3.0.CO;2-E

McLaren, N., Shanbhogue, R. (2011). Using internet search data as economic indicators. Bank of England Quarterly Bulletin, 2, 134–140. Retrieved from https://www.bankofengland.co.uk/-/media/boe/files/quarterly-bulletin/2011/using-internet-search-data-as-economic-indicators.pdf

Mehmood, T., Sæbø, S., Liland, K. H. (2020). Comparison of variable selection methods in partial least squares regression. Journal of Chemometrics, 34(6), e3226. https://doi.org/10.1002/cem.3226

Messias, V. R., Estrella, J. C., Ehlers, R., Santana, M. J., Santana, R. C., Reiff-Marganiec, S. (2016). Combining time series prediction models using genetic algorithm to autoscaling web applications hosted in the cloud infrastructure. Neural Computing and Applications, 27(8), 2383–2406. https://doi.org/10.1007/s00521-015-2133-3

Mevik, B.-H., Wehrens, R. (2007). The pls package: Principal component and partial least squares regression in R. Journal of Statistical Software, 18(2). https://doi.org/10.18637/jss.v018.i02

Mosley, L., Chan, T.-S. T., Gibberd, A. (2023). The sparse dynamic factor model: A regularised quasi-maximum likelihood approach. arXiv, 2303.11892. https://doi.org/10.48550/arXiv.2303.11892

Pearl, J. (1986). Fusion, propagation, and structuring in belief networks. Artificial Intelligence 29(3), 241–288. https://doi.org/10.1016/0004-3702(86)90072-X

Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Elsevier. https://doi.org/10.1016/C2009-0-27609-4

Pena, D., Smucler, E., Yohai, V. J. (2021). Sparse estimation of dynamic principal components for forecasting high-dimensional time series. International Journal of Forecasting, 37(4), 1498–1508. https://doi.org/10.1016/j.ijforecast.2020.10.008

Petersen, A. H., Osler, M., Ekstrom, C. T. (2021). Data-driven model building for life-course epidemiology. American Journal of Epidemiology, 190(9), 1898–1907. https://doi.org/10.1093/aje/kwab087

Preda, C., Saporta, G. (2005). PLS regression on a stochastic process. Computational Statistics and Data Analysis, 48(1), 149–158. https://doi.org/10.1016/j.csda.2003.10.003

Spirtes, P., Glymour, C., Scheines, R. (2012). Causation, Prediction, and Search. Springer. https://doi.org/10.1007/978-1-4612-2748-9

Spirtes, P., Glymour, C., Scheines, R., Kauffman, S., Aimale, V., Wimberly, F. (2000). Constructing Bayesian network models of gene expression networks from microarray data.

Stock, J. H., Watson, M. W. (2002). Forecasting using principal components from a large number of predictors. Journal of the American Statistical Association, 97(460), 1167–1179.

Stock, J. H., Watson, M. W. (2012). Dynamic factor models, In Clements M.J., Hendry D.F. Oxford Handbook on Economic Forecasting, 35–60. Oxford: Oxford University Press.

Stoica, P., Söderström, T. (1998). Partial least squares: A first-order analysis. Scandinavian Journal of Statistics, 25(1), 17–24. https://doi.org/10.1111/1467-9469.00085

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1), 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Tsamardinos, I., Aliferis, C. F. (2003). Towards principled feature selection: Relevancy, filters and wrappers. Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, In Proceedings of Machine Learning Research, R4, 300-307. Retrieved from https://proceedings.mlr.press/r4/tsamardinos03a.html

Wang, E., Cook, D., Hyndman, R. J. (2020). A new tidy data structure to support exploration and modeling of temporal data. Journal of Computational and Graphical Statistics, 29(3), 466–478. https://doi.org/10.1080/10618600.2019.1695624

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T., Miller, E., Bache, S., Müller, K., Ooms, J., Robinson, D., Seidel, D., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., Yutani, H. (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686. https://doi.org/10.21105/joss.01686

Wold, S., Martens, H., Wold, H. (2006). The multivariate calibration problem in chemistry solved by the PLS method. In: Kågström, B., Ruhe, A. (eds) Matrix Pencils. Lecture Notes in Mathematics, vol. 973, pp. 286-293. Berlin, Heidelberg: Springer. https://doi.org/10.1007/BFb0062108.

Wold, S., Ruhe, A., Wold, H., W. J. Dunn, I. (1984). The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses. SIAM Journal on Scientific and Statistical Computing, 5(3), 735–743. https://doi.org/10.1137/0905052

Zou, H., Hastie, T., Tibshirani, R. (2006). Sparse principal component analysis. Journal of Computational and Graphical Statistics, 15(2), 265–286. https://doi.org/10.1198/106186006X113430

Appendix

Figure 8. In-sample Fit 2013-2022

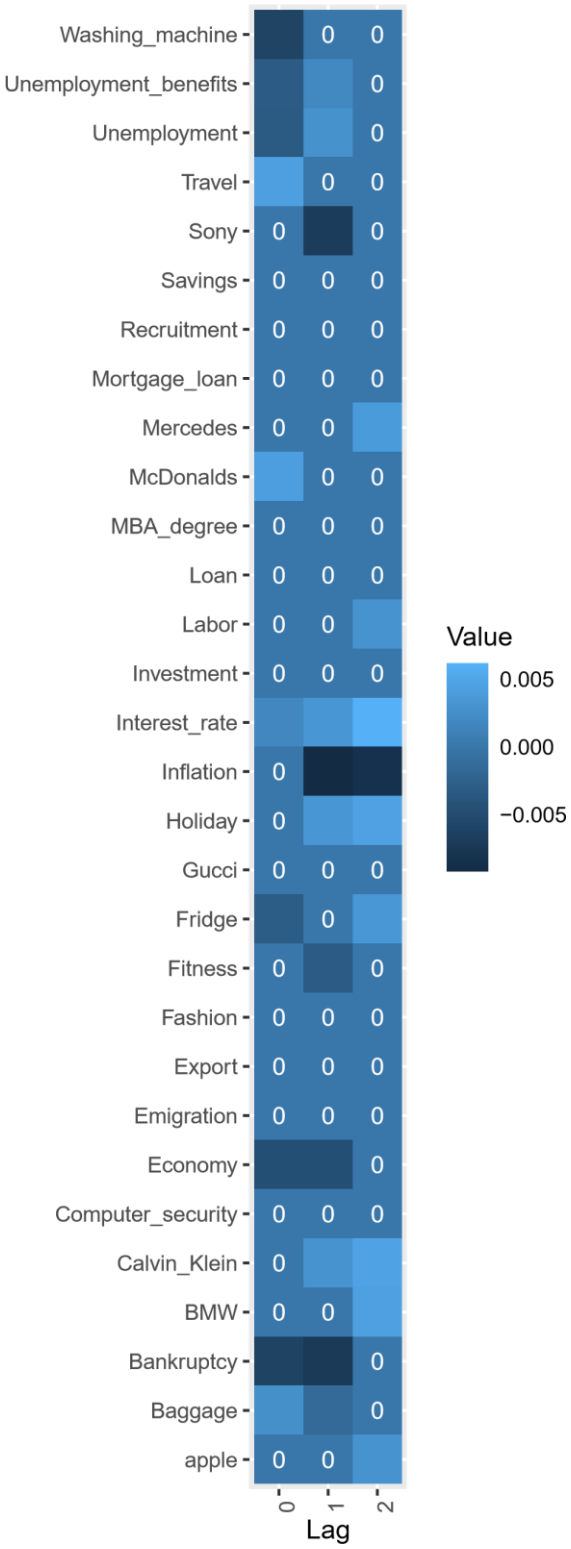

Figure 9. sPLS Variable Selection

The R code comparing performance of PLS, sPLS and PCR. Two latent factors F_1 and F_2 drive the first four variables, X_1 to X_4 and the target variable y. Six additional irrelevant variables X_5 to X_{10}, highly correlated, are included in the model. Subsequently, the PLS and PCR regression are estimated using the *spls* and Mevik and Wehrens (2007) *pls* R packages.

```
# Function to generate data for the simulation
generate_data <- function(n, seed = NULL) {
  set.seed(seed)

  # True latent factors
  L1 <- rnorm(n)
  L2 <- rnorm(n)

  # Explanatory variables driven by the latent factors
  X1 <- 0.9*L1 + rnorm(n, sd = 0.5)
  X2 <- -0.3*L1 + rnorm(n, sd = 0.8)
  X3 <- 0.5*L2 + rnorm(n, sd = 0.7)
  X4 <- L2 + rnorm(n, sd = 0.7)
  X5 <- rnorm(n)  # Independent noise variable
  X6 <- 0.7*X5 + rnorm(n, sd= 0.3)
  X7 <- -0.7*X5 + rnorm(n, sd= 0.3)
  X8 <- 0.5*X5 + rnorm(n, sd= 0.3)
  X9 <- 0.3*X5 + rnorm(n, sd= 0.3)
  X10 <- 0.1*X5 + rnorm(n, sd= 0.3)
  # Response variable
  Y <- 0.8 * L1 - 0.6 * L2 + rnorm(n, sd = 0.5)

  data.frame(L1, L2, X1, X2, X3, X4, X5, X6, X7, X8, X9, X10, Y)
}

SimData <- generate_data(100, 42)
cor(SimData[,c('L1', 'L2','X1','X2','X3','X4','X5','Y')])


plsr_model <- plsr(Y ~ X1 + X2 + X3 + X4 + X5 +
                   X6 + X7 + X8 + X9 + X10, data = SimData, ncomp = 5, validation = "LOO")
pcr_model <- pcr(Y ~ X1 + X2 + X3 + X4 + X5 +
                   X6 + X7 + X8 + X9 + X10, data = SimData, ncomp = 5, validation = "LOO")
spls_cv <- cv.spls(x = SimData[,3:(ncol(SimData)-1)], y = SimData[,"Y"],
```

```
                eta = seq(0.3, 0.9, 0.001), K = c(1:5))
spls_model <- spls(x = SimData[,3:(ncol(SimData)-1)], y = SimData[,"Y"],
                eta = spls_cv$eta.opt, K = spls_cv$K.opt)
summary(plsr_model)
plot(RMSEP(plsr_model)) # min RMSEP at 2 components
summary(pcr_model)
plot(RMSEP(pcr_model)) # min RMSEP at 3 components
spls_model

# Plotting True Latent vs. Estimated Latent
par(mfrow = c(2, 1))
plot(SimData[1:70, 'L1'], type = "l", col = "darkgrey")
#title("True Latent vs. Estimated")
lines(plsr_model$scores[1:70, 1], col = "blue")
#lines(pcr_model$scores[1:70, 1], col = "red")
plot(SimData[1:70, 'L2'], type = "l", col = "grey")
lines(plsr_model$scores[1:70, 2], col = "blue")

#
plot(SimData[1:70, 'Y'], type = "l", col = "darkgrey")
lines(plsr_model$fitted.values[1:70], col = "blue")
```