Erasmus University Rotterdam
Erasmus School of Economics
Bachelor Thesis Econometrics and Operations Research

Machine learning methods for combining forecasts of the European GDP growth

Nicky Chen (575827)

Supervisor:	Sam van Meer
Second assessor:	-
Date final version:	2nd July 2023

Abstract

In this research, different machine learning models are compared with each other. This research specifically considers the (egalitarian) LASSO and Ridge, partially egalitarian LASSO, and the octagonal shrinkage and clustering algorithm for regression (OSCAR). The goal is to find the model that forms the forecast combination with the highest forecast accuracy for predicting the GDP growth rate of the Eurozone. This is relevant to look into because the European Central Bank (ECB) chooses its monetary policy based on these forecasts. To find the best model, data from the European Central Bank is used to form the forecast combination. The real GDP growth is the dependent variable for which forecasts are made. By means of the root mean squared error and the mean absolute error, we conclude that the peLASSO is the best-performing machine learning model for making forecast combinations to predict the real GDP growth of the Eurozone.

1 Introduction

Forecast combinations have proven to be successful in economic environments. By combining forecasts, one could increase the robustness and accuracy. Having multiple forecasts could decrease the effect of outliers (forecasts that predict way too high or low). Additionally, forecast combinations could help in reducing various errors and uncertainties of an individual model, which results in better predictions that are less susceptible to the idiosyncrasies of a model. Thus, the risk of experiencing negative effects caused by errors in one individual model is mitigated when combining multiple forecasts. For major decision-makers like the European Central Bank (ECB), it is of great importance to have good forecasts of for example the gross domestic product (GDP), inflation, and unemployment rate. The goal of the ECB is to keep prices stable in the Eurozone, so they have to make decisions on what monetary policy to maintain and as these decisions depend on forecasts, accurate forecasts are essential.

Despite all the benefits, there remain unresolved issues in the world of forecast combinations. The first issue is that it is difficult to choose the set of forecasters that should be combined. Secondly, even after selecting the forecasters, a method for combining the selected forecasters needs to be chosen, which is a challenging task. The partially-egalitarian least absolute shrinkage and selection operator (peLASSO) method devised by Diebold and Shin (2019) considers both issues. In addition, the peLASSO applies necessary shrinkage to the forecasters.

Selecting variables is found to be difficult as mentioned before, especially in situations in which there are a large number of predictors that are highly correlated with each other. Bondell and Reich (2008) tried to solve this problem with their machine learning method, the octagonal shrinkage and clustering algorithm for regression (OSCAR). OSCAR does not only select variables but also groups them into predictive clusters, whereas the standard LASSO does not do this. As survey forecasts are highly correlated, the OSCAR model will be used in this paper.

For the first part of our research, the research of Diebold and Shin (2019) is replicated. In the second part, which is the extension, we try to answer the following research question: 'How effective is the machine learning method OSCAR compared to the peLASSO for combining survey forecasts of the European GDP?'. The main goal of this research is to find the best possible way to combine forecasts. This is achieved by means of comparing different models that have proven to have good performance. To answer the research question, the OSCAR and peLASSO models are built and tuned to get the best forecast combinations. Afterwards, the models are compared using performance measures root mean squared error (RMSE) and mean absolute error (MAE), and to check whether one model outperforms the other, the test by Diebold and Mariano (2002) is used. The data on which we build the models come from the European Central Bank's quarterly Survey of Professional Forecasters (ECB SPF). The GDP growth rate is the variable for which we try to make predictions with the best forecast accuracy, namely, the predictions with the lowest RMSE and MAE. Moreover, to pre-process the data, the programming language Python is used. Lastly, R and Matlab are used to build and compare the models.

By reading this paper, you will get to know whether OSCAR gives better results than the peLASSO and methods like the simple average, by means of the RMSE and MAE. In this research, we find that the OSCAR performs similarly to standard LASSO/Ridge and simple average, but it fails to beat the peLASSO.

The paper is structured as follows: First, some literature is introduced in Section 2. Furthermore, in Section 3, we will introduce the dataset used in this research and provide interesting statistics of the data. Section 4 is dedicated to the models used and the performance measures/tests. Moreover, in Section 5 the results of the replication as well as the extension part will be presented. Lastly, a conclusion is made, the answer to the research question is presented and suggestions for further research are provided in Section 6.

2 Literature Review

Over the years, the literature about forecast combinations has grown significantly. This is due to the success of forecast combinations, as combining forecasts could improve forecast accuracy by a large amount. Moreover, the literature shows that simple methods for combining forecasts work relatively well compared to more complex methods. For example, Smith and Wallis (2009) showed with their analysis that if the weights of the forecast combinations are close or even equal to equality, a simple average is on average more accurate, in terms of the root mean squared error (RMSE), than weights that are estimated. Furthermore, for biased forecasts, it holds that the larger the number of forecasters in a combination, the more efficient the simple average gets. Despite the great success of simple averages, it does have a few issues. The problem with simple averages is that individual forecasters who do not perform well, are also taken into the combination, which could result in forecast combinations with high RMSE and MAE. Therefore, some kind of selection is needed to remove these forecasters.

Penalized regressions have proven to be a successful method for selecting variables. The least absolute shrinkage and selection operator (LASSO) introduced by Tibshirani (1996) puts a bound on the L_1-norm of the coefficients. The constraint on the L_1-norm leads to shrinkage and variable selection. As a consequence, some of the coefficients are being set equal to zero. In this case, some forecasts are neglected in the forecast combination. Although LASSO does the selection, which is desired, it does not shrink the coefficients into the right direction, which is the simple average of the coefficients.

Diebold and Shin (2019) introduced a new variant of LASSO, egalitarian LASSO (eLASSO). This method shrinks the coefficients towards equality. Note that this also has a disadvantage, as it selects the coefficients to be equal to each other, which is not desired. So even though one problem is solved, another problem arises. Therefore Diebold and Shin (2019) proposed the peLASSO, which can select to zero and shrink the remaining coefficients to equality. The peLASSO consists of two penalties, one is the standard LASSO penalty, and the other penalty ensures selection and shrinkage towards equality, which is the eLASSO penalty. In brief, the peLASSO first selects coefficients that should be zero and afterwards, it shrinks the remaining coefficients towards the simple average. The penalties and the formulation of (p)eLASSO will be further explained in Section 4.

To test how the newly introduced model would perform, Diebold and Shin (2019) performed the model and compared it with the simple average, (e)LASSO, and (e)Ridge. The models were performed on the GDP forecast data from the European Central Bank's Survey of Professional Forecasters. Using the 1-year-ahead out-of-sample forecasts and a 20-quarter moving window to estimate the coefficients, Diebold and Shin (2019) concludes that peLASSO is accurate (relatively low RMSE) for out-of-sample forecasting as this method performed better than the other methods in forecasting the GDP growth of the Eurozone. Moreover, Diebold and Shin (2019) found that most of the time only a few forecasts needed to be combined and the selected forecasts should be regularized via shrinkage. Furthermore, the shrinkage direction should be towards equality (the simple average) and the shrinkage itself should be strong (selected coefficients should be averaged).

Another problem of the LASSO is that if there is a group of highly correlated variables (multicollinearity), it is challenging for LASSO to estimate the regression coefficients accurately. Small changes in the data could cause LASSO to change the estimated coefficients significantly. Thus, making the coefficients obtained by LASSO less reliable. This leads to problems with interpretation as some highly correlated predictors are not included in the model. The coefficient estimates could have an increased bias, which makes it more difficult to identify the true relationships between the regressors and the dependent variables. The OSCAR model proposed by Bondell and Reich (2008) handles this problem by adding an extra constraint to the regression.

Bondell and Reich (2008) built the OSCAR model based on soil data to find the relation between the composition of soil and forest diversity. While the LASSO only sets a bound on the L_1-norm of the coefficients, OSCAR also puts a bound on the L_∞-norm of the coefficients. Supervised clustering is thus directly implemented into the estimation process via a penalization method. Consequently, OSCAR not only performs variable selection and shrinkage toward zero, but it at the same time performs supervised clustering on the relevant variables. With supervised clustering, variables are selected and shrunk towards equal coefficients. Results of the OSCAR model, performed on soil data in Bondell and Reich (2008) showed that OSCAR performs better than LASSO in an environment with highly correlated predictors. As forecasts of the GDP from the survey are likely highly correlated, we choose to use OSCAR in this research Note that OSCAR shrinks coefficients towards equality, but this does not necessarily mean that the weights of the coefficients are equal to simple averages. The peLASSO method on the other hand does shrink the coefficients toward simple averages, which could have an impact on the forecast accuracy. OSCAR is not the only penalized regression method that focuses on grouped predictors. Tibshirani, Saunders, Rosset, Zhu and Knight (2005) introduced a method called fused LASSO, however, due to the computational speed of fused LASSO, it is not feasible to include this method in our research. Furthermore, Zou and Yuan (2008) proposed the F_∞-norm support vector machine, however, this method needs the group size and the number of groups as input beforehand, while the OSCAR chooses them automatically.

As mentioned earlier, the peLASSO first selects and afterwards shrinks coefficients to the simple average of the remaining variables. The OSCAR selects and shrinks coefficients to equality, which does not necessarily mean that the weights are equal to the inverse of the number of remaining variables. Diebold and Shin (2019) stated that the simple average is sub-optimal, thus it could be possible that OSCAR does not even reach sub-optimality. For this reason, we arrive at the following hypothesis: 'Compared to the peLASSO, the OSCAR method is not effective for combining forecasts to predict the real GDP growth.'

3 Data

To make forecast combinations for predicting real GDP, we make use of the forecast data coming from the ECBs SPF. The ECB SPF is a survey that gathers information on the expected inflation, real GDP growth, and unemployment rate in the euro area. The goal of ECB with this survey, is to provide a good assessment of the risk and uncertainty of the economy. Forecasts are made at multiple time horizons, ranging from one month to two years ahead. For this research, we will be focusing on the one-year-ahead forecasts of the real GDP growth of the euro area. For example, from the survey of 1999Q1, the GDP growth rate forecast for 1999Q3 is taken. Thus, we have that the 'one-year-ahead' forecasts are actually only six to eight months ahead. The survey results of the GDP growth from 1999Q1 until 2016Q2 are taken to make forecast combinations and to do evaluations of our models. As for each quarter, there is a data file with the inflation, GDP growth, and unemployment rate expectations, we had to make a program that collects all the necessary information. The final data set only consists of the quarters, forecasters, and the forecasts themselves.

As forecasters do not make a forecast of the real GDP growth rate each quarter, there are forecasters with many missing observations. To get rid of these forecasters and simplify the analysis, we filtered out the forecasters with five or more consecutive missing forecasts. Furthermore, we selected the 23 forecasts who had made a forecast of the GDP growth most frequently between 1999Q1 and 2016Q2. Even after selecting the 23 forecasters with the most replies, there are still missing observations in the data set. To solve this problem, we used an interpolation method proposed by Genre, Kenny, Meyler and Timmermann (2013). This method involves the following panel regression for each forecaster i , where $i = 1, ..., 23$:

$$f_{i,t} - \bar{f}_t = \beta_i(f_{i,t-1} - \bar{f}_{t-1}) + \epsilon_{i,t}, \tag{1}$$

where $f_{i,t}$ is the forecast of forecaster i for quarter t and $\bar{f}_t = \sum_{i=1}^{23} f_{i,t}$ is the simple average of the forecasts for quarter t.

Equation 1 shows an AR(1) model, where the relative deviation of a forecast from the simple average at quarter t is being regressed on the relative difference at quarter $t - 1$. We fill the missing forecasts of forecaster i at quarter t with the simple average at quarter t plus a fraction of the deviation from the average forecast at quarter $t-1$. The calculation of the missing values is done recursively using a Python program. As some forecasters have a missing value in the first few quarters, and therefore the panel regression cannot be initialized, we replace these with the forecast average of the corresponding quarters. Note that the missing values are calculated recursively. The summary statistics of the real GDP growth rate can be found in Table 1. It shows that on average the year-on-year GDP growth rate is 1.34%. The real GDP data comes from the Federal Reserve Economic Data (FRED) database. After collecting the data, the real GDP growth rate is calculated for 1999Q3-2016Q4. As there are no missing observations, there is no need for an interpolation method.

Variables	**Obs.**	**Mean**	**Std. Dev.**	**Min.**	**Max.**
GDP growth rate	70	1.34	1.98	-5.68	4.52

Table 1: Summary statistics of real GDP growth rate (dependent variable in the models).

Table 7 in the Appendix shows the summary statistics of the 23 forecasters of the GDP growth rate. Each row corresponds to a forecaster i. The first column shows the forecaster. The second column shows the total number of forecasts of a forecaster. The third column shows the mean of the forecasts of each forecaster. The last two columns show the range of each forecast, this is done by means of their minimum and maximum values. Note that the summary statistics in this table were computed after running the panel regression. Some interesting observations can be made from Table 7:

- Forecasters 2, 4, 15, 29, 36, and 38 make on average higher forecasts than the rest of the forecasters, which could imply that these forecasters are relatively more optimistic. The forecasters could perform relatively better during economic expansions.

Intuitively, we could say that these forecasters will be selected by the models when the real GDP growth is increasing.

- Forecasters 7, 24, 26, 42, 52, 85, and 94 make on average lower forecasts than the rest of the forecasters. This could mean that these forecasters are relatively more pessimist. Therefore, these forecasters could have (relatively) better performance during recessions. Intuitively, pessimistic forecasters have a higher chance to be selected by the models when the real GDP growth rate is declining strongly.

- Forecasters 7, 26, and 94 make relatively more volatile forecasts, as the standard deviation and the difference between their maximum and minimum are relatively high.

Moreover, comparing Table 1 and 7, we can observe that the standard deviation of all forecasters is low compared to the Real GDP. This could mean that the GDP growth rates are more volatile than the GDP predictions of the forecasters. Lastly, the difference between the minimum and maximum values is lower for all forecasters than that of the real GDP. This could imply that during recessions or a booming economy, the forecasters do not predict accurately. In conclusion, even with forecast combinations, predictions of the real GDP will not be accurate during volatile periods.

Figure 1 shows a graph of the forecasters (average) together with the real GDP growth rate.

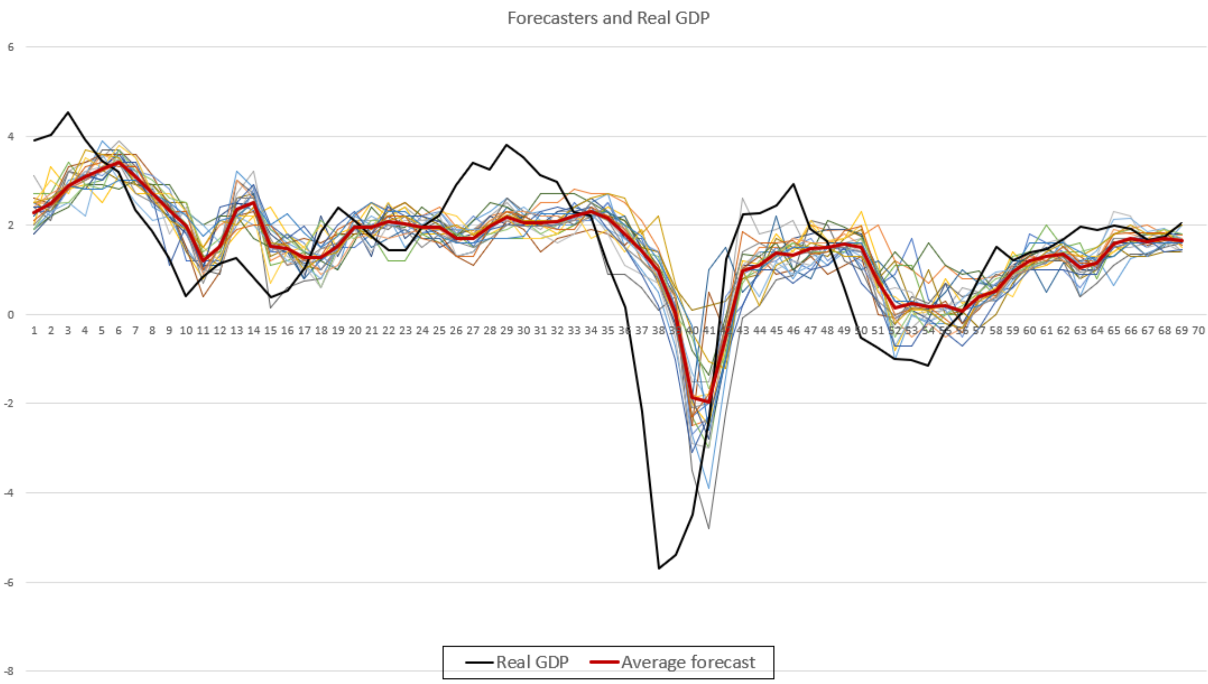

Figure 1: Graph of the forecasts of every forecaster (and the mean), and the real GDP growth. The values on the horizontal axis are the quarters, 1 corresponds to quarter 1999Q3 and 70 corresponds to quarter 2016Q4.

The following can be observed from the graph. The forecasters (and average) are all lagging, namely, forecasters adjust their forecasts upwards/downwards after the real GDP growth has moved upwards/downwards. For example, looking at observations 34 until 41, we can see that the forecasters predict a lower GDP growth only after the real GDP growth has been declining.

4 Methodology

The aim of this research is to find the best-penalized regression model for combining forecasts. Several machine learning models are estimated and one-year ahead forecasts are done over a moving window. The models are (e)Ridge, (e)LASSO, peLASSO, and OSCAR. The survey runs from 1999Q1-2016Q2, but our sample period is from 1999Q3 until 2016Q4, since we have one-year ahead forecasts. The models from both the replication part, as well as the extension part of our research, are introduced in this section. For the replication part of this research, we compare simple average, (e)Ridge, (e)LASSO, and peLASSO with each other. The extension consists of a comparison between peLASSO and OSCAR.

4.1 Moving window

The forecast combinations are made using a moving window of 20 quarters. For the first twenty observations, we will not be using the full 20-quarters estimation window, since it would mean that 15 observations in our evaluation sample would be lost in the coefficient estimation process. The evaluation period starts from period $t = 6$. For the periods $t = 6$ until $t = 20$, the coefficients will be estimated using all available data from time 1. Thus, five forecasts are burned in the estimation process. For $t > 20$, the full 20-period estimation window is used. So for each forecast combination, the coefficients need to be re-estimated and the one step ahead forecast combinations need to be collected. At last, a robustness check is done to look into the importance of the moving window size.

4.2 Notation

First, denote the following notation:

- y_t: The dependent variable (real GDP growth rate) at quarter $t = 1, ..., T$.
- f_{it}: The prediction of forecaster i at quarter $t = 1, ..., T$.
- β_i: The weight of the prediction of forecaster i in the forecast combination.
- K and k: The total number of forecasts and remaining forecasts after selection, respectively.

4.3 Penalized regressions

Consider the penalized regression with parameter λ:

$$\min_{\beta} \left\{ \sum_{t=1}^{T} (y_t - \sum_{i=1}^{K} f_{it}\beta_i)^2 + \lambda \sum_{t=1}^{K} |\beta_i|^q + \lambda \right\}. \tag{2}$$

The optimal β is denoted as follows:

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{t=1}^{T} (y_t - \sum_{i=1}^{K} \beta_i f_{it})^2 + \lambda \sum_{t=1}^{K} |\beta_i|^q \right\}, \tag{3}$$

where λ is the parameter to set the strength of the penalty.

4.3.1 Ridge regression

To get shrinkage, one can set the parameter q equal to 2, which results in the Ridge regression (Hoerl & Kennard, 2000). The Ridge regression can shrink the coefficients towards zero, but it cannot set it equal to zero:

$$\hat{\beta}_{Ridge} = \arg\min_{\beta} \left\{ \sum_{t=1}^{T} (y_t - \sum_{i=1}^{K} \beta_i f_{it})^2 + \lambda \sum_{t=1}^{K} \beta_i^2 \right\}. \tag{4}$$

The Ridge regression is solved using the glmnet package in R. Note that in this regression, we do not have an intercept as we are working with forecast combinations. Moreover, the alpha parameter in glmnet is set equal to 0 (to get the Ridge regression).

4.3.2 LASSO regression

For $q = 1$, the penalized regression can do both selection and shrinkage, which is known as the LASSO regression by Tibshirani (1996).

$$\hat{\beta}_{LASSO} = \arg\min_{\beta} \left\{ \sum_{t=1}^{T} (y_t - \sum_{i=1}^{K} \beta_i f_{it})^2 + \lambda \sum_{t=1}^{K} |\beta_i| \right\}. \tag{5}$$

The R package glmnet is used to perform the LASSO regression with the same settings as the Ridge regression, except for alpha being 1 (to get the LASSO).

4.3.3 eRidge and eLASSO regression

Diebold and Shin (2019) also introduced eRidge and eLASSO (which will be later used in the two-step procedure in peLASSO), which can be achieved by replacing the β_i in the second term of both regressions with $\beta_i - \frac{1}{K}$:

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{t=1}^{T} (y_t - \sum_{i=1}^{K} \beta_i f_{it})^2 + \lambda \sum_{t=1}^{K} |\beta_i - \frac{1}{K}|^q \right\}, \tag{6}$$

with q=1 for eLASSO and q=2 for eRidge.
The problem can be solved easily after rewriting Equation 6 to:

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{t=1}^{T} ((y_t - \bar{f}_t) - \sum_{i=1}^{K} \delta_i f_{it})^2 + \lambda \sum_{t=1}^{K} |\delta_i|^q \right\}, \tag{7}$$

where $\delta_i = \beta_i - \frac{1}{K}$ and $\bar{f}_t = \frac{1}{K} \sum_{i=1}^{K} f_{it}$.

So to get the results of eRidge and eLASSO, we simply do standard Ridge and LASSO with $(y_t - \bar{f}_t)$ being the dependent variable and $f_{1t}, .., f_{Kt}$ being the regressors. Once again, the R package glmnet is used to build the regressions.

4.3.4 peLASSO regression

The peLASSO proposed by Diebold and Shin (2019) consists of two penalties that need to be solved. The first one is the LASSO, which shrinks and selects coefficients to zero. The second penalty shrinks the remaining variables to equality. As this optimization problem is not continuous because of the k, this problem is solved in two steps.

$$\hat{\beta}_{peLASSO} = \arg\min_{\beta} \left\{ \sum_{t=1}^{T} (y_t - \sum_{i=1}^{K} \beta_i f_{it})^2 + \lambda_1 \sum_{t=1}^{K} |\beta_i|^q \right\} + \lambda_2 \sum_{t=1}^{K} |\beta_i - \frac{1}{k}|, \tag{8}$$

where k is the number of non-zero coefficients after selection.
The two-step procedure works as follows:

1. The LASSO is applied on the forecasts, which results in variables being set equal to zero and shrunk towards zero

2. Using eRidge, eLASSO, or simple average the remaining non-zero coefficients are selected or shrunk towards $\frac{1}{k}$, depending on which method is used.

4.3.5 OSCAR regression

The OSCAR regression model performs selection and supervised clustering on the selected variables simultaneously. OSCAR is given by setting q equal to 1 and adding another penalty term, which is the L_∞-norm. This results in the following problem:

$$\hat{\beta}_{OSCAR} = \arg\min_{\beta} \left\{ \sum_{t=1}^{T} (y_t - \sum_{i=1}^{K} \beta_i f_{it})^2 + \lambda_1 \Big(\sum_{j=1}^{K} |\beta_j| + \lambda_2 \sum_{m<n} \max\{|\beta_m|, |\beta_n|\} \Big) \right\}, \tag{9}$$

where $\lambda_1 \geq 0$ and $\lambda_2 > 0$ the tuning parameters with λ_1 the parameter to control the relative weight of the two norms and λ_2 to parameter to control the strength of the constraint. The penalty consists of a combination of the L_1-norm and the pairwise $L_\infty-$norm.

Bondell and Reich (2008) solved this problem using the following formulation:

$$\begin{aligned} \hat{\beta}_{OSCAR} &= \arg\min_{\beta} \left\{ \sum_{t=1}^{T} (y_t - \sum_{i=1}^{K} \beta_i f_{it}) \right\}^2 \\ \text{s.t.} \quad &(1-c)(\sum_{j=1}^{K} |\beta_j|) + c \sum_{m<n} \max\{|\beta_m|, |\beta_n|\}) \leq pT_0 \end{aligned} \tag{10}$$

where $0 \leq c \leq 1$ and $0 < p < 1$ are the tuning parameters with c the parameter to control the relative weight of the two norms, and t the parameter to control the strength of both norms. The T_0 here is the value of the constraint at an initial solution, this is in our case computed using the Ridge as recommended by Bondell and Reich (2008). OSCAR is solved with the 'Statistics and Machine Learning Toolbox' and 'Deep Learning Toolbox' in MATLAB. The code from Bondell and Reich (2008) is used after adjusting it.

The optimization problem of OSCAR is motivated geometrically. Thus, the interpretation of OSCAR will be further explained geometrically. The constraint region of the OSCAR can be found in Figure 2. It shows the constraint region in the (β_1, β_2) plane. The constraint region is shown for four different values of the parameter c, and the parameter p is kept constant. Data is generated to plot these constraint regions. The left upper graph shows the constraint region for $c = 0$, this is equal to the LASSO regression. The right lower graph shows the constraint region for $c = 1$, which implies that only the L_∞-norm is used. For the upper right and lower left graph, c is set equal to 0.3 and 0.6, respectively. Looking at these graphs, we could conclude that the octagonal term in the name OSCAR is now obvious. The constraint region in two dimensions has the shape of an octagon.

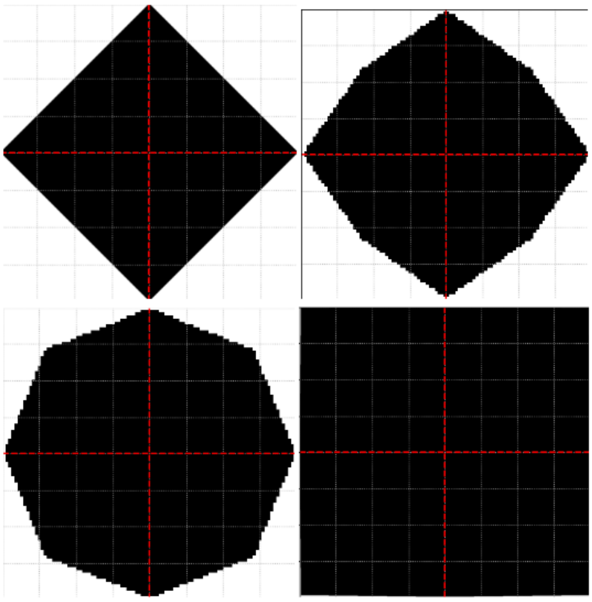

Figure 2: Plot of the constraint region of the (β_1, β_2) plane for OSCAR. The graphs are made for four different values of c.

Figure 3 shows the constraint region and the optimal solution of the OSCAR. We generate independent variables with high and low correlation. The optimal solution for independent variables with low correlation is $\beta_1 = 0$ and $\beta_2 = 1$. For highly correlated independent variables, we find that $\beta_1 = \beta_2$ is the optimal solution. From this, we can conclude that the OSCAR indeed groups highly correlated predictors.

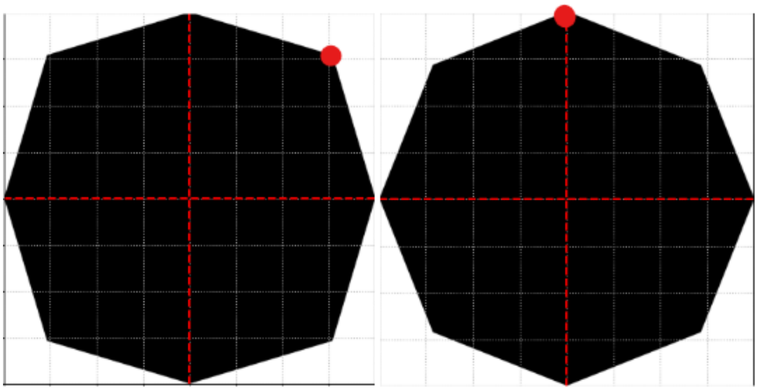

Figure 3: Constraint region of the (β_1, β_2) plane for the OSCAR regression for $c = 0.6$. The OSCAR solution for high and low-correlated independent variables is marked with a red dot.

4.4 Model comparison

The following two forecast performance measures are used:

- $RMSE = \sqrt{\frac{1}{T-R}\sum_{t=R}^{T} e_{t+1|R}^2}$
- $MAE = \frac{1}{T-R}\sum_{t=R}^{T} |e_{t+1|R}|$

where R is the number of in-sample observations, e is defined as the actual GDP growth minus the forecast combination and T is the total number of observations. The RMSE is used as this was also seen in Diebold and Shin (2019). Furthermore, the MAE is used because it takes into account that there could be potential outliers in forecasting. These 'forecast outliers' could cause the RMSE to be high. The Diebold-Mariano test proposed by Diebold and Mariano (2002) is used to test whether a model, on average, outperforms another model significantly.

4.5 Parameter tuning

To choose the best parameter values, a set of parameters will be used and compared. The out-of-sample root mean squared error (RMSE) tied to each parameter (combination) is compared to make a choice. The parameter combination with the lowest RMSE is then chosen. For (e)Ridge, (e)LASSO, and the three variants of peLASSO, the forecast accuracy is then examined for many λs. A forecast is computed for 200 λs. Starting with an equally-spaced grid on $[-15, 15]$, which then gets exponentiated, resulting in a grid on $(0, 3269017]$.

As the OSCAR penalization problem is solved differently, the grid of the tuning parameters is different. The forecasts for 200 c's and 200 p's are computed. They are both in an equally-spaced grid, which results in 40000 different tuning parameter combinations. The grids of c and p are $[0, 1]$ and $(0, 1)$, respectively. As this caues the computing time to be very high for building OSCAR, parallel processing is used.

5 Results

We first build the models introduced in Section 4. This section can be divided into two sub-sections. In Section 5.1, the standard penalized regression models and the peLASSO are compared with the simple average, which is the replication part of our research. Thereafter, in Section 5.2, OSCAR is compared with peLASSO as an extension. Lastly, a robustness check is done to examine whether a different moving window size could give different results.

5.1 Comparison peLASSO with simple average - Replication

Table 2 shows the results of the replication part of our research. The first column shows the models built for the replication part of our research. The second column shows the RMSE value. The optimal parameters can be found in the third column. The fourth column shows the average number of forecasts selected over all the estimation windows. The one-sided Diebold and Mariano (2002)-statistic can be found in column five, this statistic is computed as in Harvey, Leybourne and Newbold (1997).

Method	**RMSE**	**Optimal parameter**	**#**	**DM**	**p-value**
Ridge	1.51	1.97	23.00	0.08	0.47
LASSO	1.52	0.44	2.71	-0.13	0.55
eRidge	1.51	3269017 (maximum)	23.00	-2.15	0.98
eLASSO	1.51	3.60	23.00	1.00	0.16
peLASSO (Average)	1.41	0.21	3.40	1.63	0.05
peLASSO (eRidge)	1.38	(0.44, 3269017)	2.71	0.92	0.18
peLASSO (eLASSO)	1.38	(0.44, 3.6)	2.71	1.41	0.08
Forecaster	**RMSE**	**Optimal parameter**	**#**	**DM**	**p-value**
Best	1.42	-	1.00	1.11	0.14
90%	1.46	-	1.00	0.83	0.21
Median	1.53	-	1.00	-0.44	0.67
10%	1.69	-	1.00	-2.84	1.00
Worst	1.74	-	1.00	-2.69	1.00
Simple average	1.51	-	23.00	-	-

Table 2: Results of (e)Ridge, (e)LASSO and the three variants of peLASSO

Some interesting observations can be made from Table 2:

- eLASSO and eRidge have about the same RMSE. This result could be explained by the strong shrinkage of both models towards the simple average $\frac{1}{K}$.
- The worst individual forecaster does not perform better than all of the forecasting combination methods. In addition, the best individual forecaster performs similarly to the best machine learning methods
- From the p-value of Ridge, LASSO, and eRidge, we can conclude that they perform similarly to the simple average, even though Ridge and LASSO shrink coefficients towards zero. eRidge shrinks all coefficients towards the simple average, which could explain the high p-value.
- The optimal parameter of the eRidge and the eRidge in peLASSO are both the maximum value of our exponentiated grid. Therefore, we could conclude that a strong eRidge penalty gives the best predictions.
- The shrinkage of eRidge and eLASSO toward zero is strong, which is the reason that it performs similarly to the simple average.
- The peLASSO methods have almost equal performance, this is due to the first step of this method, because the LASSO is used there.

- The peLASSO with eRidge and eLASSO have an RMSE that is close to ten percent lower than the other methods, including the simple average.
- The peLASSO with eLASSO on average performs significantly better than the simple average with a ten percent significance level.

5.2 Comparison between OSCAR and peLASSO - Extension

In this subsection, the peLASSO is compared against the OSCAR, and the RMSE values are compared here. The three variants of peLASSO, and OSCAR are built. OSCAR is built two times, one with both parameters being tuned, and another time for which c is set equal to 1. Thus, for the second run of OSCAR, we only tune the parameter p.

Method	**RMSE**	**Optimal parameter**	**#**	**DM**	**p-value**
peLASSO (Average)	1.41	0.21	3.40	-	-
peLASSO (eRidge)	1.38	(0.44, 3269017)	2.71	-	-
peLASSO (eLASSO	1.38	(0.44, 3.6)	2.71	-	-
OSCAR	1.52	(0, 0.2915)	3.55	-1.40	0.92
OSCAR (c =1)	1.53	0.66	13.09	-1.36	0.91

Table 3: Results of the three peLASSO methods and the two OSCAR methods (RMSE).

Table 3 shows the results of the peLASSO and the OSCAR. The first four columns are the same as that of Table 2. The fifth column is the Diebold-Mariano statistic against peLASSO (eLASSO), which is the best-performing one. The last column contains the p-values of the Diebold-Mariano tests. A few important observations can be made from this table:

- The RMSE of all three variants of the peLASSO is lower than that of the two of OSCAR. The peLASSO models have an RMSE that is at least 7% lower than the best-performing OSCAR model.
- The average number of variables selected by the standard OSCAR is higher than the best-performing peLASSO method. This could imply that the peLASSO is more strict with selecting variables.
- Leaving the L_1-penalty (which is the penalty that the LASSO regression uses) out of the OSCAR regression, results in similar performance as the standard OSCAR, based on the RMSE.
- Optimal parameter c of the standard OSCAR is equal to 0, which means that standard OSCAR removes the L_∞-penalty completely from its constraint. The only penalty left then, is the L_1-penalty, which is just the LASSO.

- We can conclude that the two OSCAR methods do not perform significantly better than the peLASSO (eLASSO) model because the p-values are high.

- Comparing Table 2 with Table 3, we could see that the RMSE of the LASSO is approximately equal to the OSCAR (c=1). This is because setting c equal to 1 in OSCAR results in the LASSO. However, we would expect that the RMSEs are the same, but it is not in this case. This could be the result of using different packages from different programming languages, for LASSO we used a package in R and for OSCAR a machine learning package in MATLAB was used.

Table 4 shows the results of the peLASSO and OSCAR with the values of the MAE. The first columns show the models built. The second column shows the MAE values. The last two columns show the DM-statistic and p-value, respectively, here OSCAR is compared to the best peLASSO method, peLASSO (eLASSO).

Method	**MAE**	**DM**	**p-value**
peLASSO (Average)	1.02	-	-
peLASSO (eRidge)	1.04	-	-
peLASSO (eLASSO)	0.97	-	-
OSCAR	1.05	-0.98	0.84
OSCAR (c =1)	1.07	-1.12	0.87

Table 4: Results of the peLASSO and OSCAR (MAE).

From Table 4, the following conclusions can be made:

- The best performing peLASSO method has an MAE that is almost ten percent lower than the OSCAR methods. However, the peLASSO with eRidge and simple average have an MAE close to the OSCAR methods.

- From the DM-statistic and the p-value, we can conclude that the OSCAR does not perform significantly better than the peLASSO. Which is the same result as when we compared the RMSEs of both models.

From Table 3 and 4 we conclude that OSCAR does not make significantly better predictions than peLASSO. The MAE and RMSE of the best peLASSO are lower than that of OSCAR and the p-value is large (hypothesis of equal performance not rejected).

5.3 Robustness check

As a robustness check, we check whether the results still hold for different moving window sizes. The peLASSO and OSCAR models are built using half and double the moving window size of the current one (20), 10 and 40 quarters respectively. The RMSE of the models is then compared.

Table 5 shows the results of the peLASSO and the OSCAR with a moving window of 10 quarters. The first four columns are the same as that of Table 3. The fifth column is the Diebold-Mariano statistic against peLASSO (average), which is the best-performing one. The p-value corresponding to the DM statistic is shown in the last column.

Method	**RMSE**	**Optimal parameter**	**#**	**DM**	**p-value**
peLASSO (Average)	1.46	0.32	2.48	-	-
peLASSO (eRidge)	1.50	(0.00, 0.02)	19.06	-	-
peLASSO (eLASSO)	1.51	(0.03, 3.60)	6.42	-	-
OSCAR	1.59	(0.08, 0.63)	10.71	-1.09	0.86
OSCAR (c =1)	1.59	0.92	13.80	-1.05	0.85

Table 5: Results of the three peLASSO methods and the two OSCAR methods (RMSE) with a moving window of 10 quarters.

Table 6 shows the results of the peLASSO and the OSCAR with a moving window of 40 quarters. The first four columns are the same as that of Table 3. The fifth column is the Diebold-Mariano statistic against peLASSO (average), the best-performing peLASSO. The last column shows the p-value corresponding to the DM statistic.

Method	**RMSE**	**Optimal parameter**	**#**	**DM**	**p-value**
peLASSO (Average)	1.47	0.80	2.862	-	-
peLASSO (eRidge)	1.47	(0.02, 0.59)	14.28	-	-
peLASSO (eLASSO)	1.51	(0.00, 0.11)	22.71	-	-
OSCAR	1.49	(1.00, 0.44)	14.85	-0.48	0.68
OSCAR (c =1)	1.49	0.44	14.85	-0.48	0.68

Table 6: Results of the three peLASSO methods and the two OSCAR methods (RMSE) with a moving window of 40 quarters.

An interesting observation that could be made is that the RMSE of OSCAR gets closer to the RMSE of the peLASSO, the larger the moving window gets. It could imply that the larger the estimation sample is, the better OSCAR performs. However, further research is needed to confirm this. Both Table 5 and 6 show that OSCAR does not significantly outperform the peLASSO. Moreover, the RMSE of the best peLASSO models are for both moving windows lower than that of OSCAR. In conclusion, this confirms our earlier findings.

6 Conclusion

In this paper, we looked at two different penalized regression models for combining forecasts of the year-on-year quarterly GDP growth rate. The two models are the octagonal shrinkage and clustering algorithm for regression (OSCAR) and the peLASSO (partially-egalitarian LASSO). We then compare the forecast combinations against the real GDP growth by means of the mean squared error and the mean absolute error. The results show that the peLASSO with eRidge and eLASSO has the lowest RMSE and that the peLASSO with eLASSO performs significantly better than the OSCAR. A robustness check confirmed for other sizes of the moving window, peLASSO still has a lower RMSE than OSCAR.

In Section 1, we introduced the following research question: 'How effective is the machine learning method OSCAR compared to the peLASSO for combining survey forecasts of the European GDP?'. The answer to this question is that the OSCAR is not effective compared to the peLASSO. All three versions (simple average, eRidge, and eLASSO) of the peLASSO performed better than OSCAR, by means of the RMSE and MAE. Therefore, we do not reject our hypothesis: 'Compared to the peLASSO, the OSCAR method is not effective for combining forecasts to predict the real GDP growth'. Furthermore, the Diebold-Mariano test showed that the best peLASSO method performs significantly better than the OSCAR. The OSCAR had almost equal RMSE compared to standard Ridge, LASSO, eRidge, eLASSO, and simple average.

The results of this research could be used to make accurate forecast combinations for predicting GDP growth. The models considered in this model could be used for other macroeconomic variables like inflation and the unemployment rate. Moreover, these models could be applied in more volatile environments like the stock market to combine forecasts of analysts.

For further research, one could consider adding more traditional econometric models to the comparison. For example, ordinary least squares with forward or backward feature selection. Moreover, if there is no time constraint, one could consider the fused LASSO or other time-consuming machine learning models that are more complex. Furthermore, further research could examine the effects using an expanding window for forecast combinations. Lastly, the machine learning models in this research could be used for forecasting instead of combining forecasters. One could for example try to predict GDP growth using other macroeconomic variables.

References

Bondell, H. D. & Reich, B. J. (2008). Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar. *Biometrics*, *64*(1), 115-123. doi:

https://doi.org/10.1111/j.1541-0420.2007.00843.x

Diebold, F. X. & Mariano, R. S. (2002). Comparing predictive accuracy. *Journal of Business & economic statistics*, *20*(1), 134–144.

Diebold, F. X. & Shin, M. (2019). Machine learning for regularized survey forecast combination: Partially-egalitarian lasso and its derivatives. *International Journal of Forecasting*, *35*(4), 1679-1691. doi: https://doi.org/10.1016/j.ijforecast.2018.09.006

Genre, V., Kenny, G., Meyler, A. & Timmermann, A. (2013). Combining expert forecasts: Can anything beat the simple average? *International Journal of Forecasting*, *29*(1), 108-121. doi: https://doi.org/10.1016/j.ijforecast.2012.06.004

Harvey, D., Leybourne, S. & Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, *13*(2), 281-291. doi: https://doi.org/10.1016/S0169-2070(96)00719-4

Hoerl, A. E. & Kennard, R. W. (2000). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, *42*(1), 80–86. Retrieved 2023-06-18, from `http://www.jstor.org/stable/1271436`

Smith, J. & Wallis, K. F. (2009). A simple explanation of the forecast combination puzzle*. *Oxford Bulletin of Economics and Statistics*, *71*(3), 331-355. Retrieved from `https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1468-0084.2008.00541.x` doi: https://doi.org/10.1111/j.1468-0084.2008.00541.x

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, *58*(1), 267–288. Retrieved 2023-06-11, from `http://www.jstor.org/stable/2346178`

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J. & Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(1), 91–108.

Zou, H. & Yuan, M. (2008). The f-norm support vector machine. *Statistica Sinica*, 379–398.

7 Appendix

Forecaster i	Obs.	Mean	Std. Dev.	Min.	Max.
1	70	1.50	1.00	-2.30	3.70
2	70	1.73	1.02	-2.50	3.38
4	70	1.63	1.08	-1.50	3.90
5	70	1.51	1.02	-2.50	3.30
7	70	1.36	1.30	-3.90	3.90
15	70	1.58	0.96	-2.60	3.00
16	70	1.45	1.07	-3.00	3.49
20	70	1.50	1.08	-2.80	3.60
24	70	1.44	1.03	-2.50	3.60
26	70	1.26	1.40	-4.80	3.70
29	70	1.58	0.84	-3.23	-0.30
31	70	1.55	0.85	-1.88	3.40
36	70	1.68	0.82	-1.36	3.00
37	70	1.48	1.09	-2.70	3.40
39	70	1.47	1.09	-2.00	3.60
42	70	1.43	1.03	-2.10	3.38
38	70	1.59	1.00	-1.20	3.70
52	70	1.41	1.13	-3.00	3.50
54	70	1.54	1.04	-1.86	3.80
85	70	1.45	0.97	-2.00	3.60
89	70	1.55	0.96	-1.67	3.70
94	70	1.33	1.26	-3.10	3.40
95	70	1.52	1.03	-2.30	3.40

Table 7: Summary statistics of the forecasts of the most interesting forecasters (predictors in the models).