Investing through Economic Cycles with Ensemble Machine Learning Algorithms

Thomas Raffinot *

May 2016

Abstract

Ensemble machine learning algorithms, referred to as random forest (Breiman (2001)) and as boosting (Schapire (1990)), are applied to quickly and accurately detect economic turning points in the United States and in the euro area. The two key features of those algorithms are their abilities to entertain a large number of predictors and to perform estimation and variable selection simultaneously. The real-time ability to nowcast economic turning points is gauged. To assess the value of the models, profit maximization measures are employed in addition to more standard criteria. When comparing predictive accuracy and profit measures, the model confidence set procedure proposed by Hansen et al. (2011) is applied to avoid data snooping. The investment strategies based on the models achieve impressive risk-adjusted returns: macroeconomists can get rich nowcasting economic turning points.

JEL classifications: C53, E32, E37, G11
Keywords: Random Forest; Boosting; Economic cycles; Profit maximization measures; Model Confidence Set

*Millesime-IS/ PSL Research University, Université Paris Dauphine, LEDa-SDFi (email: traffinot@gmail.com)
The paper has immensely benefited from discussions with Marie Bessec and Anne Epaulard. The views expressed in this paper are the sole responsibility of the authors. Any remaining errors or shortcomings are those of the authors.

Introduction

Economic turning points detection in real time is a notorious difficult task. Economists often fail to detect if a new economic phase has already begun. For instance, the Survey of Professional Forecasters conducted in May 2008, by the American Statistical Association and the National Bureau of Economic Research, said there would not be a recession in 2008, even though one had already started.

Researchers and investors focus mainly on the business cycle detection, which is meant to reproduce the cycle of the global level of activity of a country. However, Raffinot (2014) emphasizes that the growth cycle, defined as the deviation of the real GDP to its long-term trend, is much more interesting for euro and dollar-based investors. Indeed, in theory, investment strategies based on growth cycle turning points achieve better risk-adjusted returns than those based on business cycle turning points.

If the long-term trend is considered as the estimated potential level[1], then the growth cycle equals the output gap. A turning point of the output gap occurs when the current growth rate of the activity is above or below the potential growth rate, thereby signalling increasing or decreasing inflation pressures. Quickly detecting growth cycle turning points provides thus extremely reliable pieces of information for the conduct of monetary policy. For instance, if a central bank wants to loosen monetary policy, because inflation is running under the target, a through of the output gap would indicate that its strategy starts to bear fruit.

One stylised fact of economic cycles is the non-linearity: the effect of a shock depends on the rest of the economic environment. For instance, small shock, such as a decrease in housing prices, can sometimes, but not always, have large effects, such as a recession. Real-time regime classification and turning points detection require thus methods capable of taking into account the non-linearity of the cycles. In this respect, many parametric models have been proposed, especially Markov switching models (see Piger (2011)) and probit models (see Liu and Moench (2014)). Parametric models are effective if the true data generating process (DGP) linking the observed data to the economic regime is known. In practice, however, one might lack such strong prior knowledge. It leads to practical issues in estimating parametric models, especially the presence of frequent local maxima in the likelihood. Therefore, in the absence of knowledge of the true DGP, non-parametric methods are advocated, such as machine-learning algorithms, as they do not rely on a

[1]The potential output is the maximum amount of goods and services an economy can turn out at full capacity

specification of the DGP (Giusto and Piger (ming)).

The machine-learning approach assumes that the DGP is complex and unknown and attempts to learn the response by observing inputs and responses and finding dominant patterns. This places the emphasis on a model's ability to predict well and focuses on what is being predicted and how prediction success should be measured. Machine learning is used in spam filters, ad placement, credit scoring, fraud detection, stock trading, drug design, and many other applications. Giusto and Piger (ming) introduce in economics a very simple machine-learning algorithm known as Learning Vector Quantization (LVQ), which appears very competitive with commonly used alternatives. Raffinot (2015) also provides evidence that LVQ is very effective, despite its simplicity, and that some economic and financial indicators can be exploited to quickly identify turning points in real time in the United States and in the euro area. The main drawback with the latter approach is that the model selection is very complex, time and data consuming. For instance, the first step consists in narrowing down the predictors to only those that are relevant. All possible combinations of four variables from the selected predictors are then computed and the best model is selected.

Over the last couple of decades, researchers in the computational intelligence and machine learning community have developed more complex methods, also called ensemble learning, which improve prediction performances. Ensemble methods are learning models that achieve performance by combining the opinions of multiple learners. The two most popular techniques for constructing ensembles are random forest (Breiman (2001)) and boosting (Schapire (1990)). The two features of those algorithms are their abilities to entertain a large number of predictors and to perform estimation and variable selection simultaneously. Paradoxically, both methods work by adding randomness to the data (Varian (2014)), although they have substantial differences. Random forest relies on simple averaging of models in the ensemble and boosting is an iterative process where the errors are kept being modelled.

While the random forest algorithm is usually applied in medical research and biological studies, it is largely unknown in economics and to the best of my knowledge has not been applied to economic turning point detection. Boosting is increasingly applied to empirical problems in economics. Ng (2014) and Berge (2015) apply the algorithm to the problem of identifying business cycle turning points in the United States.

This paper aims at applying random forest and boosting algorithms to create several models aiming at quickly and accurately detecting growth cycle turning points in real time, not only in the United States but also in the euro area. Those models are then combined, as averaging forecasts from different models often improves upon forecasts based on a single model (Bates and

Granger (1969)).

The real-time ability to nowcast economic turning points is assessed. Since, for investors, the usefulness of a forecast depends on the rewards associated with the actions taken by the agent as a result of the forecast, profit maximization measures based on trading strategies are employed in addition to more standard criteria. To gauge the economic magnitude of the models, simple hypothetical trading strategies are created. To avoid data snooping, which occurs when a given set of data is used more than once for purposes of inference or model selection, the comparison of predictive accuracy and profit measures is assessed using the model confidence set procedure proposed by Hansen et al. (2011) .

The findings of the paper can be summarized as follows: ensemble machine learning algorithms are very effective to detect economic turning points. Investment strategies achieve thus excellent risk-adjusted returns in the United States and in the euro area. However, the selection of the best model is a challenging task. For instance, economists and investors would not always choose the same model. Moreover, depending on the data and the objective, random forest sometimes performs better than boosting, sometimes not. In the end, combining forecasts seems to be the best option.

The rest of the paper proceeds as follows. Section 1 introduces ensemble machine learning algorithms, referred to as random forest and as boosting. Section 2 describes the empirical set up: the turning point chronology, the data-set, the alternative models and the evaluation of the forecasts. Section 3 analyses the empirical results.

1 Ensemble Machine Learning Algorithms

Making decisions based on the input of multiple people or experts has been a common practice in human civilization and serves as the foundation of a democratic society. Over the last couple of decades, researchers in the computational intelligence and machine learning community have studied schemes that share such a joint decision procedure. Ensemble methods are learning models that achieve performance by combining the opinions of multiple learners.

Two of the most popular techniques for constructing ensembles are random forest (Breiman (2001)) and boosting (Schapire (1990)). The two key features of those algorithms are their abilities to entertain a large number of predictors and to perform estimation and variable selection simultaneously.

Paradoxically, both methods work by adding randomness to the data, but adding randomness turns out to be a helpful way of dealing with the

overfitting problem (Varian (2014)). Overfitting denotes the situation when a model targets particular observations rather than a general structure: the model explains the training data instead of finding patterns that generalize it. In general, overfitting the model takes the form of making an overly complex model. Attempting to make the model conform too closely to slightly inaccurate data can infect the model with substantial errors and reduce its predictive power. In other words, your model learn the training data by heart instead of learning the patterns which prevent it from being able to generalized to the test data.

Nevertheless, those methods have substantial differences. Random forest relies on simple averaging of models in the ensemble. They derive their strength from two aspects: randomizing subsamples of the training data and randomizing the selection of features. Boosting methods are based on a different strategy of ensemble formation: boosting combines models that do not perform particularly well individually into one with much improved properties. The main idea is to add new models to the ensemble sequentially. At each particular iteration, a new weak, base-learner model is trained with respect to the error of the whole ensemble learned so far.

1.1 Random forest

Random forest (RF henceforth) is a non-parametric statistical method for both high-dimensional classification and regression problems, which requires no distributional assumptions on covariate relation to the response.

RF is a way of averaging multiple deep decision trees, trained on different parts of the same training set, with the goal of overcoming overfitting problem of individual decision tree. In other words, RF builds a large collection of de-correlated trees and then averages their predictions. The method is fast, robust to noise and produces surprisingly good out-of-sample fits, particularly with highly nonlinear data (Caruana and Niculescu-Mizil (2005)).

1.1.1 Classification and Regression Trees algorithm

Classification and regression trees (CART henceforth), introduced by Breiman et al. (1984), are machine-learning methods for constructing prediction models from data that can be used for classification or regression. The models are obtained by recursively partitioning the data space and fitting a simple prediction model within each partition. As a result, the partitioning can be represented graphically as a decision tree.

The tree is generated in a recursive binary way, resulting in nodes connected by branches. A node, which is partitioned into two new nodes, is

called a parent node. The new nodes are called child nodes. A terminal node is a node that has no child nodes.

A CART procedure is generally made up of two steps. In the first step, the full tree is built using a binary split procedure. The full tree is an overgrown model, which closely describes the training set. In the second step, the model is pruned to avoid overfitting.

Given a dataset with explanatory inputs x, the CART algorithm can be summarized as follows:

Step 1 Find each predictor's best split:

Sort each predictor's entries by increasing value. Iterate over all values of the sorted predictor and find the candidate for the best split. That is the value that maximizes the splitting criterion.

Step 2 Find the node's best split:

To actually perform the split, compare all evaluated predictors from step 1 and choose the split, that maximizes the splitting criterion.

Step 3 Let s be this best split of the winning predictor. All $x \leq s$ are sent to the left node and all $x > s$ to the right node.

So constructing a CART is accomplished by finding the best split, which is just trying every possibility, calculating the "goodness"' of every possible split and choose the best one. For every split at node t a splitting criterion $\Delta i(s|t)$ is calculated. The best split s, at node t maximizes this splitting criterion $\Delta i(s|t)$, based on the Gini criterion in classification problems and measured by mean squared error in regression trees. For classification, given a node t with estimated class probabilities $p(j|t)$ with $j = 1, \ldots, J$ being the class label, a measure of node impurity given t is:

$$i(s|t) = 1 - \sum_j p(j|t)^2 = \sum_{j \neq k} p(j|t)p(k|t)$$

A search is then made for the split that most reduces node, or equivalently tree, impurity.

1.1.2 Construction of a random forest

RF is an ensemble of tree predictors. Each decision tree is built from a bootstrapped sample of the full dataset (Efron and Tibshirani (1994)) and then, at each node, only a random sample of the available variables is used as candidate variables for split point selection. Thus, instead of determining

the optimal split on a given node by evaluating all possible splits on all variables, a subset of the input variables are randomly chosen, and the best split is calculated only within this subset. Once an ensemble of K trees is built, the predicted outcome (final decision) is obtained as the average value over the K trees.

Averaging over trees, in combination with the randomisation used in growing a tree, enables random forests to approximate a rich class of functions while maintaining a low generalisation error. This enables random forests to adapt to the data, automatically fitting higher-order interactions and non-linear effects, while at the same time keeping overfitting in check (Ishwaran (2007)). As the number of trees increases, the generalization error converges to a limit (Breiman (2001)).

A RF is constructed by the following steps:

Step 1 Given that a training set consists of N observations and M features, choose a number $m \leq M$ of features to randomly select for each tree and a number K that represents the number of trees to grow.

Step 2 Take a bootstrap sample Z of the N observations. So about two third of the cases are chosen. Then select randomly m features.

Step 3 Grow a CART using the bootstrap sample Z and the m randomly selected features.

Step 4 Repeat the steps 2 and 3, K times.

Step 5 Output the ensemble of trees T_1^K

For regression, to make a prediction at a new point x:

$$\hat{y}_{RF}(x) = \frac{1}{K} \sum_{i=1}^{K} T_i(x)$$

For classification, each tree gives a classification for x. The forest chooses the class that has the most out of n votes. Calculating the associated probability is easily done.

Since Breiman (2001) uses unpruned decision trees as base classifiers, RF has basically only one parameter to set: the number of features to randomly select at each node. Typically, for a classification problem with M features, $\sqrt{M}$ features (rounded down) are used in each split and $M/3$ features (rounded down) with a minimum node size of 5 as the default are recommended for regression problems (Friedman et al. (2000)).

1.2 Boosting

Boosting is based on the idea of creating an accurate learner by combining many so-called "weak learners" (Schapire (1990)), i.e., with high bias and small variance. The main concept of boosting is to add new models to the ensemble sequentially. At each particular iteration, a new weak, base-learner model is trained with respect to the error of the whole ensemble learned so far. The final model hopefully yields greater predictive performance than the individual models. The heuristics is thus simple: an iterative process where the errors are kept being modelled.

The original boosting algorithms such as AdaBoost (Freund and Schapire (1997)), were purely algorithm-driven, which made the detailed analysis of their properties and performance rather difficult (Schapire (2003)). The gradient descent view of boosting (Friedman (2001),Friedman et al. (2000)) has connected boosting to the more common optimisation view of statistical inference. This formulation of boosting methods and the corresponding models are called the gradient boosting machines (GBM henceforth).

Using a learning sample $(y_i; \mathbf{x}_i)_{(i=1,...,n)}$, where the response y is continuous (regression problem) or discrete (classification problem) and $\mathbf{x} = (\mathbf{x}_1, ..., \mathbf{x}_d)$ denotes a d-dimensional explanatory input variables, the objective is to obtain an estimate $\hat{f}(\mathbf{x})$ of the function $f(\mathbf{x})$, which maps $\mathbf{x}$ to y. The task is thus to estimate the function $\hat{f}(\mathbf{x})$, that minimizes the expectation of some loss function, $\Psi(y, f)$, i.e.,

$$\hat{f}(\mathbf{x}) = \arg\min_{f(\mathbf{x})} \mathbf{E}(\Psi(y, f(\mathbf{x}))$$

The loss function $\Psi(y, f)$ is assumed to be smooth and convex in the second argument to ensure that the gradient method works well.

An approximate solution to the minimization problem is obtained via forward stagewise additive modeling, which approximates the solution by sequentially adding new basis functions to the expansion without adjusting the parameters and coefficients of those that have already been added.

GBM take on various forms with different programs using different loss functions, different base models, and different optimization schemes. This high flexibility makes GBM highly customizable to any particular data-driven task and introduces a lot of freedom into the model design thus making the choice of the most appropriate loss function a matter of trial and error. As a matter of fact, Friedman et al. (2000) warned that given a dataset, it is rarely known in advance which procedures and base learners should work the best, or if any of them would even provide decent results.

Loss-functions can be classified according to the type of response vari-

able y. In the case of categorical response, the response variable y typically takes on binary values $y \in 0, 1$. To simplify the notation, let us assume the transformed labels $\bar{y} = 2y - 1$ making $\bar{y} \in -1, 1$.

The most frequently used loss-functions for classification are the following:

-Adaboost loss function: $\Psi(y, f(\mathbf{x})) = \exp(-\bar{y}f(\mathbf{x}))$

-Binomial loss function: $\Psi(y, f(\mathbf{x})) = -\log(1 + \exp(-2\bar{y}f(\mathbf{x})))$

The Binomial loss function is far more robust than the Adaboost loss function in noisy settings (mislabels, overlapping classes).

The most frequently used loss-functions for regression are the following:

-Squared error loss: $\Psi(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$

-Absolute loss: $\Psi(y, f(\mathbf{x})) = |y - f(\mathbf{x}|$

Several types of weak learners have been considered in the boosting literature, including decision trees (e.g., stumps, trees with two terminal nodes) (Friedman (2001)), smoothing splines (Bühlmann and Yu (2003)), wavelets (Wu et al. (2004)) and many more.

To design a particular GBM for a given task, one has to provide the choices of functional parameters $\Psi(y, f)$ and the weak learner $h(\mathbf{x}, \theta)$, characterized by a set of parameters θ. For instance, for decision trees, θ describes the axis to be split, the split points and the location parameter in terminal nodes.

The principle difference between boosting methods and conventional machine-learning techniques is that optimization is held out in the function space (Friedman (2001)). That is, the function estimate $\hat{f}(\mathbf{x})$ is parameterized in the additive functional form:

$$\hat{f}(\mathbf{x}) = \sum_{i=0}^{M_{stop}} \hat{f}_i(\mathbf{x})$$

Moreover, a common procedure is to restrict $\hat{f}(\mathbf{x})$ to take the form:

$$\hat{f}(\mathbf{x}) = \sum_{m=1}^{M_{stop}} \beta_m h(\mathbf{x}, \theta_m)$$

The original function optimization problem has thus been changed to a parameter optimization problem.

The GBM algorithm can be summarized as follows:

Step 1 Initialize $\hat{f}_0(\mathbf{x}) = \arg\min_\rho \sum_{i=1}^N \Psi(y_i, \rho), m = 0.$

Step 2 $m = m + 1$

Step 3 Compute the negative gradient

$$z_i = -\frac{\partial}{\partial f(\mathbf{x}_i)}\Psi(y_i, f(\mathbf{x}_i))\Big|_{f(\mathbf{x}_i)=\hat{f}_{m-1}(\mathbf{x}_i)}, i = 1, \ldots, n$$

Step 4 Fit the base-learner function, $h(\mathbf{x}, \theta)$ to be the most correlated with the gradient vector.

$$\theta_m = \arg\min_{\beta,\theta} \sum_{i=1}^{n} z_i - \beta h(\mathbf{x}_i, \theta_m)$$

Step 5 Find the best gradient descent step-size ρ_m

$$\rho_m = \arg\min_{\rho} \sum_{i=1}^{N} \Psi(y_i, \hat{f}(\mathbf{x}_i)_{m-1} + \rho h(\mathbf{x}, \theta_m))$$

Step 6 Update the estimate of $f_m(\mathbf{x})$ as

$$\hat{f}_m(\mathbf{x}) \leftarrow \hat{f}(\mathbf{x})_{m-1} + \rho_m h(\mathbf{x}, \theta_m))$$

Step 7 Iterate 2-6 until $m = M_{stop}$.

The classic approach to controlling the model complexity is the introduction of the regularization through shrinkage. In the context of GBM, shrinkage is used for reducing, or shrinking, the impact of each additional fitted base-learner. It reduces the size of incremental steps and thus penalizes the importance of each consecutive iteration. A better improvement is done by taking many small steps than by taking fewer large steps. Indeed, if one of the boosting iterations turns out to be erroneous, its negative impact can be easily corrected in subsequent steps.

The simplest form of regularization through shrinkage is the direct proportional shrinkage (Friedman (2001)). In this case the effect of shrinkage is directly defined as the parameter $\lambda \in [0, 1]$. The regularization is applied to the step 6 in the gradient boosting algorithm:

$$\hat{f}_m(\mathbf{x}) \leftarrow \hat{f}(\mathbf{x})_{m-1} + \lambda\rho_m h(\mathbf{x}, \theta_m))$$

A crucial issue is the choice of the stopping iteration M_{stop}. Boosting algorithms should generally not be run until convergence. Otherwise, overfits resulting in a suboptimal prediction accuracy would be likely to occur (Friedman et al. (2000)).

One possible approach to choosing the number of iterations M_{stop} would be to use an information criterion like Akaike's AIC or some sort of minimum description length criteria. However, they have been shown to overshoot the true number of iterations (Hastie (2007)) and thus are not recommended for practical usage. Cross-validation techniques should be used to estimate the optimal M_{stop} (Hastie (2007)).

Briefly, cross-validation uses part of the available data to fit the model, and a different part to test it. K-fold cross-validation works by dividing the training data randomly into K roughly equal-sized parts. For the k^{th} part, the learning method is fit to the other $K-1$ parts of the data, and calculate the prediction error of the fitted model when predicting the k^{th} part of the data. This is done for $k = 1, 2, ..., K$ and the K prediction error estimates are averaged. An estimated prediction error curve as a function of the complexity parameter is obtained (Hastie et al. (2009)). Typical choices of K are 5 and 10. When it comes to time series forecasting, Bergmeir et al. (2015) demonstrate that K-fold cross-validation performs favourably compared to both out-of-sample evaluation and other time-series-specific techniques.

In contrast to the choice of the stopping iteration, the choice of λ has been shown to be of minor importance for the predictive performance of a boosting algorithm. The only requirement is that the value of λ is small, e.g. $\lambda = 0.1$ (Friedman (2001)).

In this paper, two different approaches are tested: a combination of a binomial loss function with decision trees ("BTB") as in Ng (2014) and a combination of a squared error loss function with P-splines ("SPB") as in Berge (2015) or Taieb et al. (2015). P-splines ((Eilers and Marx (1996))) can be seen as a versatile modeling tool for non-linear effects. Examples include smooth effects, bivariate smooth effects (e.g., spatial effects), varying coefficient terms, cyclic effects and many more.

2 Empirical setup

2.1 Turning point chronology in real time

Researchers and investors focus mainly on the business cycle detection, which is meant to reproduce the cycle of the global level of activity of a country. The turning points of that cycle separate periods of recession from periods of expansion. Since absolute prolonged declines in the level of economic activity tend to be rare events, Mintz (1974) introduces the growth cycle, defined as the deviation of the real GDP to its long-term trend, to produce information on economic fluctuations. The turning points of that cycle separate periods

of slowdowns and accelerations. A slowdown signals thus a decline in the rate of growth of the economy though not necessarily an absolute decline in economic activity.

Raffinot (2014) emphasizes the importance of the growth cycle for euro and dollar-based investors. Indeed, in theory, investment strategies based on growth cycle turning points achieve better risk-adjusted returns than those based on business cycle turning points.

Two economic phases are thus considered: slowdown and acceleration. Applied to the context of nowcasting, the variable of interest y can be summarized as follows:

$$y_t = \begin{cases} 1, & \text{if in acceleration} \\ 0, & \text{otherwise} \end{cases}$$

It should be noted that y can be seen as continuous and as discrete. In this paper, y is considered as discrete for the "RF" model and "BTB" model. In these cases, $\hat{y}$ is the probability of being in the regime referred to as acceleration. For the "SPB" model, y is considered as continuous. Even in this case, $\hat{y}_t$ is thought as estimating the probability of being in the regime referred to as acceleration.

To implement the ensemble machine learning algorithms, a chronology of economic regimes is needed. This paper employs the turning point chronology established in Raffinot (2014) (see Appendix B).

The training sample runs over the period from January 1988 to December 2001. The performance of the models are then evaluated over the period from January 2002 to December 2013. In the euro area, 54 % of the data are classified as slowdown. In the United States, 71 % of the data are classified as acceleration. Over the period from January 2002 to December 2013, there were 7 turning points in the growth cycle in the euro area and 5 in the United States.

In real time, the complete chronology is not available, but the monthly GDP introduced by Raffinot (2007)[2] allows to quickly refine the turning point chronology.

In the empirical analysis, a recursive estimation of the models is done: each month the model is estimated with the data and the chronology that would have been available at the time the nowcasting is done. The models are thus trained each month on a sample that extends from the beginning of the sample through month $T - 12$, over which the turning point chronology is assumed known. For instance, in January 2012, the chronology that

[2]A temporal disaggregation based on business surveys of the non revised values of gross domestic product GDP is used to develop a monthly indicator of GDP.

would have been available to implement the models runs over the period from January 1988 to January 2011.

Re-estimating the model at each point in time also allows the relationship between covariates and the dependent variable to change (see Ng (2014)). Since the aim of this paper is to emphasize that ensemble machine learning algorithms can provide useful signals for policymakers and for investors in real time, analysing the most frequently selected predictors is out of the scope of this study.

2.2 Data set

The real-time detection of turning points faces the difficult issues of late release dates and data revision. As a matter of fact, key statistics are published with a long delay, are subsequently revised and are available at different frequencies. For example, gross domestic product (GDP) is only available on a quarterly basis with a time span of one to three months, and sometimes with significant revisions.

However, a range of monthly economic series are released giving indications of short-term movements. Among them, business surveys provide economists for timely and reliable pieces of information on business activity. They are subject to very weak revisions and are usually less volatile than other monthly series. They are published before the end of the month they relate to or just a few days after. In the euro area, surveys published by the European Commission have been proven to be very effective (see Raffinot (2007)). In the United States, the surveys published by the Institute for Supply Management(ISM), the Conference Board and the National Association of Home Builders (NAHB) are often tested in the literature (see Liu and Moench (2014)).

Moreover, financial series, which are not revised and often available on a daily basis, have also been considered: the yield curve, defined as the difference between the ten-year and the three-month yield, the level and curvature of the yield curve (see Chauvet and Senyuz (2012) or Berge (2015)). Other financial series are the investment-grade and high-yield corporate spreads, stock markets (S&P500, Eurostoxx), stock markets volatility (see Chauvet et al. (2015)), the VIX index and the VSTOXX index, which is the VIX equivalent for the euro area. It should be added, that this paper uses end of month values to match stock index futures and options contracts settlement prices.[3]

Finally, some real economic data have been tested, such as the four-week

[3]http://www.cmegroup.com/trading/equity-index/fairvaluefaq.html

moving average of initial claims for unemployment insurance, which is a weekly measure of the number of jobless claims filed by individuals seeking to receive state jobless benefits.[4]

To detect the turning points in real-time, not only original series are screened, but also differentiated series (to underline the phases of low and high pace of growth). Because of the classical trade-off between reliability and advance, different lags of differentiation were considered: 1 to 18 months. The large dataset of predictors consists of more than 1000 monthly variables in the euro area and in the United States.

2.3 Alternative classifiers

2.3.1 Random guessing

To prove that the models are significantly better than random guessing, several alternative classifiers, which assign classes arbitrarily, are computed. The selected models should have a better accuracy than the latter.

The first one (*Acc*) classifies all data as "acceleration", the second one (*Slow*) classifies all data as "slowdown". The last one (*Random*) randomly assigns classes based on the proportions found in the training data. Thousand different simulations are computed and average criteria are provided.

2.3.2 Parametric models

The term spread has been proved to be an excellent leading indicator of recession in the United States (Liu and Moench (2014)) and in the euro area (Duarte et al. (2005)). Nowcasts from probit models based on the term spread are thus computed[5].

For a given covariate x_n, based on the learning sample $(R_1, x_1), ..., (R_{T-12}, x_{T-12})$, the model is characterized by the simple equation:

$$P(R_t^{probit} = 1) = \Phi(\alpha_0 + \alpha_1 x_t)$$

where Φ denotes a standard Gaussian cumulative distribution function, i. e.

[4]All series are provided by Datastream.

[5]Markov-switching dynamic factor models are effective to identify economic turning points (Camacho et al. (2015)). However, variable selection in factor analysis is a challenging task: forecasts often improve by focusing on a limited set of highly informative series. For instance, Boivin and Ng (2006) demonstrate that factor-based forecasts extracted from 40 variables perform better than those extracted from 147 variables. A proper comparison is thus left for future research.

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-\frac{1}{2}t^2} dt$$

The probit model maximizes the following log likelihood function:

$$\ln L(\alpha_0, \alpha_1) = \sum_{t=1}^{T-12} (1 - R_t) \ln[1 - \Phi(\alpha_0 + \alpha_1 x_t)] + R_t \ln(\Phi(\alpha_0 + \alpha_1 x_t))$$

2.3.3 Combining forecasts

A longstanding finding in the forecasting literature is that averaging forecasts from different models often improves upon forecasts based on a single model (see Bates and Granger (1969)).

Consider a situation where there are K different models to forecast a variable of interest y. Each model i implies a forecast $\hat{y}_i$. Forecasts are then combined into a single aggregated forecast:

$$\hat{y}_{comb} = \sum_{i=1}^{K} w_i \hat{y}_i$$

As regards w_i, various ways of combining forecast have been proposed. Empirical evidence show that a simple combination methods often work reasonably well relative to more complex combinations (see Clemen (1989) and Timmermann (2006)). Claeskens et al. (2016) offer a theoretical explanation for this stylized fact. In this paper, an equally weighted forecast is used:

$$\hat{y}_{comb} = \frac{1}{K} \sum_{i=1}^{K} \hat{y}_i$$

2.4 Model evaluation

2.4.1 Classical criteria

Two metrics are computed to evaluate the quality of classification of a model.

The first one is the Brier's Quadratic Probability Score (QPS), defined as follows:

$$QPS = \frac{1}{F} \sum_{t=1}^{F} (\hat{y}_t - y_t)^2$$

where $t = 1, ..., F$ is the number of forecasts. The best model should strive to minimize the QPS.

3 Confusion matrix

The *confusion matrix* (or *error matrix*) is one way to summarize the performance of a classifier for binary classification tasks. This square matrix consists of columns and rows that list the number of instances as absolute or relative "actual class" vs. "predicted class" ratios.
Let P be the label of class 1 and N be the label of a second class or the label of all classes that are *not class 1* in a multi-class setting.

		Predicted class	
		P	N
Actual Class	P	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

The following equations are based on *An introduction to ROC analysis*

The second one is the area under the Receiver Operating Characteristics (ROC) curve. The ROC curve describes all possible combinations of true positive and false positive rates that arise as one varies the threshold c used to make binomial forecasts from a real-valued classifier. Given a classifier and an instance, there are four possible outcomes. If the instance is positive and it is classified as positive, it is counted as a true positive ($T_p(c)$). If the instance is negative and classified as negative, it is counted as a true negative. If a negative instance is misclassified as positive, it is counted as a false positive ($F_p(c)$). The same goes for positive instances being misclassified as negative. Those are then counted as false negative. As c varies from 0 to 1, the ROC curve is traced out in $T_p(c), F_p(c)$ space that describes the classification ability of the model. Accuracy is measured by the Area Under the ROC curve (AUROC). An area of 1 represents a perfect test, an area of 0.5 represents a worthless test. A general rule of thumb is that an AUROC value exceeding 0.85 indicates a useful prediction performance. The ROC curve is a useful measure, because it precisely captures the ability of each model to accurately categorize economic phases. In particular, by using the area under the ROC curve (AUROC), one can evaluate the categorization ability of the model over an entire spectrum of different cutoffs for determining a chosen economic

phase, instead of evaluating predictive power at any one arbitrary threshold.

Hanley and McNeil (1982) propose a test to compare the AUROC predictive accuracy. The aim is to test the best models in the selection with another criteria, thereby further reducing the set. The t-statistic for the test of $H_0 : AUROC_1 = AUROC_2$ is given by:

$$t = \frac{AUROC_1 - AUROC_2}{\sqrt{(\sigma_1^2 + \sigma_2^2 - 2r\sigma_1 * \sigma_2}}$$

where, AUROC1 and AUROC2 are the areas under the curve for models 1 and 2 which are being compared. Similarly, σ_1 and σ_2 refer to the variances of the AUROCs for model 1 and model 2, respectively. Finally, r is the correlation between the two AUROCs (see Hanley and McNeil (1982) or Liu and Moench (2014) for more details on the test statistic and its implementation).

In this paper, a two-step model selection is computed. The first step is to select the best set of models according to Brier's Quadratic Probability Score (QPS) and then the selection is refined based on the the area under the ROC curve (AUROC) and the test proposed by Hanley and McNeil (1982).

3.0.1 Profit maximization measures

For investors, the usefulness of a forecast depends on the rewards associated with the actions taken by the agent as a result of the forecast. In addition to more standard criteria, profit maximization measures are thus employed.

Since asset classes behave differently during different phases of the economic cycles (Raffinot (2014)), investment strategies based on economic regimes induced by the models should generate significant profits.

In order to frame the concept of active portfolio management, a specified investment strategy is required. The investment strategies are as stripped-down and simple as possible without raising concerns that the key results will not carry over to more general and intricate methods or asset classes.

We first consider an equity portfolio manager investing 100 € or 100$ on January 1, 2002. Each month, the investor decides upon the fraction of wealth to be invested based on the current state of the economy induced by the model based on the data that would have been available at the time the decision was made. If the model classifies the period as acceleration, then the investor can leverage his portfolio (120% of his wealth is invested on the asset and 20% of cash is borrowed), otherwise he only invests 80% of his wealth and 20% is kept in cash.

Moreover, since asset classes perform differently during different stages of the growth cycle, it might be reasonable to rebalance the portfolio (shifting allocation weights) based on the stage of the growth cycle (Raffinot (2014)).

The second strategy aims at beating the classic asset allocation for an institutional portfolio, *i.e.* 60% of the portfolio allocated to equities and 40% to fixed income securities (bonds). The investor decides each month to rebalance his portfolio. If the model indicates acceleration, then 80% of the portfolio is allocated to equities and 20% to bonds, otherwise 40% of the portfolio is allocated to equities and 60% to bonds.

Pesaran and Timmermann (1994) and Han et al. (2013) demonstrate that the total cost of transactions appears to be low, less than 1% (around 50 basis points when trading in stocks while the cost for bonds is 10 basis points). To simplify, no transaction costs are considered.

To avoid look-ahead bias, the reallocation takes place at the beginning of the month following the turning point. As a matter of fact, an investor could not know at the beginning of any month whether a turning point would occur in that month.

For conventional comparison of the portfolio performances, annualized average returns, annualized standard deviation (volatility), performance to volatility ratio (PVR), max drawdown (MDD) are computed. The performance to volatility ratio compares the expected returns of an investment to the amount of risk undertaken to capture these returns. The Max drawdown (MDD) is the largest drop from the maximum cumulative return. In brief, the MDD offers investors a worst case scenario.

3.0.2 Data snooping

Data snooping, which occurs when a given set of data is used more than once for purposes of inference or model selection, leads to the possibility that any results obtained in a statistical study may simply be due to chance rather than to any merit inherent in the method yielding the results (see White (2000)).

To avoid data snooping, the model confidence set (MCS) procedure (Hansen et al. (2011)) is computed.

The MCS procedure is a model selection algorithm, which filters a set of models from a given entirety of models. The resulting set contains the best models with a probability that is no less than $1 - \alpha$ with α being the size of the test (see Hansen et al. (2011)).

An advantage of the test is that it not necessarily selects a single model, instead it acknowledges possible limitations in the data since the number of models in the set containing the best model will depend on how informative the data are.

More formally, define a set M_0 that contains the set of models under evaluation indexed by: $i = 0, ..., m_0$. Let $d_{i,j,t}$ denote the loss differential

between two models by

$$d_{i,j,t} = L_{i,t} - L_{j,t}, \forall i, j \in M_0$$

L is the loss calculated from some loss function for each evaluation point $t = 1, ..., T$. The set of superior models is defined as:

$$M^* = \{i \in M_0 : E[d_{i,j,t}] \leq 0 \, \forall j \in M_0\}$$

The MCS uses a sequential testing procedure to determine M^*. The null hypothesis being tested is:

$$\begin{cases} H_{0,M} : E[d_{i,j,t}] = 0 \, \forall i, j \in M \text{ where } M \text{ is a subset of } M_0 \\ H_{A,M} : E[d_{i,j,t}] \neq 0 \text{ for some } i, j \in M \end{cases}$$

When the equivalence test rejects the null hypothesis, at least one model in the set M is considered inferior and the model that contributes the most to the rejection of the null is eliminated from the set M. This procedure is repeated until the null is accepted and the remaining models in M now equal $\widehat{M}^*_{1-\alpha}$.

According to Hansen et al. (2011), the following two statistics can be used for the sequential testing of the null hypothesis:

$$t_{i,j} = \frac{\overline{d}_{i,j}}{\sqrt{\widehat{var}(\overline{d}_{i,j})}} \text{ and } t_i = \frac{\overline{d}_i}{\sqrt{\widehat{var}(\overline{d}_i)}}$$

where m is the number of models in M, $\overline{d}_i = (m-1)^{-1} \sum_{j \in M} \overline{d}_{i,j}$, is the simple loss of the i^{th} model relative to the averages losses across models in the set M, and $\overline{d}_{i,j} = (m)^{-1} \sum_{t=1}^{m} d_{i,j,t}$ measures the relative sample loss between the i^{th} and i^{th} models. Since the distribution of the test statistic depends on unknown parameters a bootstrap procedure is used to estimate the distribution.

In this paper, the MCS is applied with classical criteria loss function (Brier's Quadratic Probability Score) and with profit maximization loss function (risk-adjusted returns). As regards investment strategies, it should be noted that the MCS aims at finding the best model and all models which are indistinguishable from the best, not those better than the benchmark. To determined if models are better than the benchmark, the stepwise test of multiple reality check by Romano and Wolf (2005) and the stepwise multiple superior predictive ability test by Hsu et al. (2013) should be considered. However, if the benchmark is not selected in the best models set, investors can conclude that their strategies "beat" the benchmark.

4 Empirical results

4.1 United States

The two-step model selection is computed as described previously. The first step of the model selection is to find the best set of models according to the MCS procedure based on Brier's Quadratic Probability Score (QPS) and then the model selection is refined based on the the area under the ROC curve (AUROC). The AUROC metric is thus only computed for models included in $\widehat{M}^*_{70\%}$. Table 1 highlights classical metrics for the models in the United States.

Table 1: **Classical evaluation criteria in the United States**

	QPS	**AUROC**
SPB	0.13	
RF	0.07**	0.94
BTB	0.05**	0.94
$CF^{BTB-RF-SPB}$	0.07**	0.94
CF^{RF-SPB}	0.09	0.93
CF^{RF-BTB}	0.06**	0.94
$CF^{SPB-BTB}$	0.08**	0.94
$Prob$	0.22	
Acc	0.21	
$Slow$	0.79	
$Random$	0.25	

Note: This table reports classical metrics used to evaluate the quality of the models: the area under the ROC curve (AUROC) and the Brier's Quadratic Probability Score (QPS). * and ** indicate the model is in the set of best models $\widehat{M}^*_{90\%}$ and $\widehat{M}^*_{70\%}$, respectively. SPB refers to a boosting model based on squared error loss with P-splines, RF refers to a random forest model, BTB refers to a boosting model based on binomial loss function with decision trees, $CF^{BTB-RF-SPB}$ combines SPB,RF and BTB models, CF^{RF-SPB} combines SPB and RF models, CF^{RF-BTB} combines RF and BTB models, $CF^{SPB-BTB}$ combines SPB and BTB models,$Prob$ refers to the probit model based on the term spread, Acc classifies all data as "acceleration", $Slow$ classifies all data as "slowdown" and $Random$ randomly assigns classes based on the proportions found in the training data.

The performance of the models are impressive and are consistent with the results found in Berge (2015). Ensemble machine learning models built are significantly and statistically better than random guessing. "RF" and "BTB" belong to $\widehat{M}^*_{70\%}$. "SPB" is the only "basic" model not selected

in any best models set. Moreover, combining forecasts is confirmed to be effective. Comparisons made with the test proposed by Hanley and McNeil (1982) between models in $\widehat{M}^*_{70\%}$ conclude that no model is better than the others.

The ability to produce profits it now tested. Tables 2 and 3 emphasize that active investment strategies based on the growth cycle achieve superb risk-adjusted returns and outperform the passive buy-and-hold benchmark.

Table 2: **Summary of return and risk measures in the United States: 120/80 equity strategy**

	Average returns	**Volatily**	**PVR**	**MDD**
SPB	0.110*	0.149	0.74	-0.43
RF	0.107	0.147	0.72	-0.43
BTB	0.109**	0.146	0.75	-0.44
$CF^{BTB-RF-SPB}$	0.109**	0.145	0.75	-0.43
CF^{RF-SPB}	0.105	0.147	0.72	-0.43
CF^{RF-BTB}	0.111**	0.146	0.76	-0.43
$CF^{SPB-BTB}$	0.109**	0.146	0.74	-0.44
$Prob$	0.094	0.173	0.54	-0.57
Acc	0.099	0.177	0.56	-0.58
$Slow$	0.066	0.118	0.56	-0.43
$Random$	0.092	0.155	0.59	-0.51
$Benchmark$	0.083	0.147	0.56	-0.51

Note: This table reports profit maximization measures for 120/80 equity strategy based on the state of the growth cycle induced by the models. Returns are monthly and annualized. The volatility corresponds to the annualized standard deviation. The performance to volatility ratio (PVR) compares the expected returns of an investment to the amount of risk undertaken to capture these returns. The Max drawdown (MDD) measures the largest single drop from peak to bottom in the value of a portfolio. * and ** indicate the model is in the set of best models $\widehat{M}^*_{90\%}$ and $\widehat{M}^*_{70\%}$, respectively. SPB refers to a boosting model based on squared error loss with P-splines, RF refers to a random forest model, BTB refers to a boosting model based on binomial loss function with decision trees, $CF^{BTB-RF-SPB}$ combines SPB,RF and BTB models, CF^{RF-SPB} combines SPB and RF models, CF^{RF-BTB} combines RF and BTB models, $CF^{SPB-BTB}$ combines SPB and BTB models, $Prob$ refers to the probit model based on the term spread, Acc classifies all data as "acceleration", $Slow$ classifies all data as "slowdown", $Random$ randomly assigns classes based on the proportions found in the training data and $Benchmark$ refers to the passive buy-and-hold investment strategy.

Table 3: **Summary of return and risk measures in the United States: dynamic asset allocation**

	Average returns	Volatily	PVR	MDD
SPB	0.091**	0.090	1	-0.18
RF	0.088*	0.088	0.99	-0.18
BTB	0.091**	0.087	1	-0.20
$CF^{BTB-RF-SPB}$	0.091**	0.086	1.1	-0.18
CF^{RF-SPB}	0.086*	0.088	0.98	-0.18
CF^{RF-BTB}	0.093**	0.086	1.1	-0.18
$CF^{SPB-BTB}$	0.090**	0.087	1	-0.20
$Prob$	0.074	0.113	0.66	-0.39
Acc	0.075	0.116	0.65	-0.42
$Slow$	0.060*	0.058	1	-0.18
$Random$	0.076	0.095	0.79	-0.30
$Benchmark$	0.068	0.085	0.79	-0.31

Note: This table reports profit maximization measures for a dynamic asset allocation between bonds and equities based on the state of the growth cycle induced by the models. Returns are monthly and annualized. The volatility corresponds to the annualized standard deviation. The performance to volatility ratio (PVR) compares the expected returns of an investment to the amount of risk undertaken to capture these returns. The Max drawdown (MDD) measures the largest single drop from peak to bottom in the value of a portfolio. * and ** indicate the model is in the set of best models $\widehat{M}^*_{90\%}$ and $\widehat{M}^*_{70\%}$, respectively. SPB refers to a boosting model based on squared error loss with P-splines, RF refers to a random forest model, BTB refers to a boosting model based on binomial loss function with decision trees, $CF^{BTB-RF-SPB}$ combines SPB,RF and BTB models, CF^{RF-SPB} combines SPB and RF models, CF^{RF-BTB} combines RF and BTB models, $CF^{SPB-BTB}$ combines SPB and BTB models, $Prob$ refers to the probit model based on the term spread, Acc classifies all data as "acceleration", $Slow$ classifies all data as "slowdown", $Random$ randomly assigns classes based on the proportions found in the training data and $Benchmark$ refers to the passive buy-and-hold investment strategy.

Table 2 points out that several strategies based on the growth cycle detection outperform the benchmark: it is thus possible to time the stock market based on economic cycles in real time. These results have naturally implications for the risk management and hedging. Especially, in the options market one can utilize the current state of the economy to hedge the portfolio against the possible price declines. For example, writing an out-of-money covered call or buy a put option when the stock market is expected to decrease (slowdown) would limit the losses.

Selecting the best model is still complicated. In comparison with the "classical" case, $\widehat{M}^*_{70\%}$ turns out to be quite different.

Surprisingly, "RF" is never selected. "SPB" is not included in any

best models set for economists but can be useful for investors, since it is chosen in $\widehat{M}^*_{90\%}$. Importantly, "BTB", "$CF^{BTB-RF-SPB}$", "CF^{RF-BTB}", "$CF^{SPB-BTB}$" are selected in the best models set for economists and for equity investors.

Moreover, dynamic asset allocation delivers a substantial improvement in risk-adjusted performance as compared to static asset allocation, especially for investors who seek to avoid large losses. The reduction of the MDD, which focuses on the danger of permanent loss of capital as a sensible measure of risk, is what risk-averse investors value the most. Portfolio rebalancing based on the stage of the growth cycle in real time in the United States is thus realisable. Again, the best models sets differ from the previous cases. At last, SPB is attached to $\widehat{M}^*_{70\%}$. It should be noted that the strategy based on $Slow$, which is overweighted in bonds, belongs to $\widehat{M}^*_{90\%}$, in line with the thrilling and non-reproducible performance of the bond market. Importantly, "BTB","$CF^{BTB-RF-SPB}$","CF^{RF-BTB}","$CF^{SPB-BTB}$" are still selected in the best models set.

To sum up, depending on the data and the objective, the model selection is quite different. That said, ensemble machine learning algorithms perform very well and combining forecast is proved to be useful. For instance, "$CF^{BTB-RF-SPB}$", which is an equal weight combination of the three "basic" models, is always selected in the best models sets.

4.2 Euro area

The same model selection methodology is applied in the euro area. The first step is to find the best set of models according to the MCS procedure based on Brier's Quadratic Probability Score (QPS) and then this selection is refined based on the the area under the ROC curve (AUROC). Table 4 highlights classical metrics for the models in the euro area.

Table 4: **Classical evaluation criteria in the euro area**

	QPS	AUROC
SPB	0.12	
RF	0.11	
BTB	0.12	
$CF^{BTB-RF-SPB}$	0.10**	0.93
CF^{RF-SPB}	0.10**	0.93
CF^{RF-BTB}	0.11	
$CF^{SPB-BTB}$	0.11	
$Prob$	0.25	
Acc	0.45	
$Slow$	0.54	
$Random$	0.48	

Note: This table reports classical metrics used to evaluate the quality of the models: the area under the ROC curve (AUROC) and the Brier's Quadratic Probability Score (QPS). * and ** indicate the model is in the set of best models $\widehat{M}^*_{90\%}$ and $\widehat{M}^*_{70\%}$, respectively. SPB refers to a boosting model based on squared error loss with P-splines, RF refers to a random forest model, BTB refers to a boosting model based on binomial loss function with decision trees, $CF^{BTB-RF-SPB}$ combines SPB,RF and BTB models, CF^{RF-SPB} combines SPB and RF models, CF^{RF-BTB} combines RF and BTB models, $CF^{SPB-BTB}$ combines SPB and BTB models, $Prob$ refers to the probit model based on the term spread, Acc classifies all data as "acceleration", $Slow$ classifies all data as "slowdown" and $Random$ randomly assigns classes based on the proportions found in the training data.

The performance of ensemble machine learning models are notable and are significantly better than random guessing. Metrics in the euro area are less remarkable than in the United States. Indeed, the persistence of the regimes is smaller in the euro area growth cycle, the real-time classification is thus harder. Surprisingly, only combining forecasts models are included in the best models sets. The test proposed by Hanley and McNeil (1982) between "$CF^{BTB-RF-SPB}$" and "CF^{RF-SPB}" concludes that no model is better than the other.

The ability to generate profits it now analysed. Tables 5 and 6 highlight that active investment strategies based on the growth cycle also achieve excellent risk-adjusted returns and outperform the passive buy-and-hold benchmark. Naturally, the risk management and hedging implications described for the United States also apply in the euro area.

Table 5: **Summary of return and risk measures in the euro area: 120/80 equity strategy**

	Average returns	**Volatily**	**PVR**	**MDD**
SPB	0.085**	0.161	0.53	-0.46
RF	0.083**	0.160	0.52	-0.46
BTB	0.079*	0.158	0.50	-0.46
$CF^{BTB-RF-SPB}$	0.083**	0.161	0.53	-0.46
CF^{RF-SPB}	0.071*	0.159	0.50	-0.46
CF^{RF-BTB}	0.082*	0.163	0.52	-0.46
$CF^{SPB-BTB}$	0.080*	0.161	0.51	-0.46
$Prob$	0.075	0.182	0.41	-0.48
Acc	0.077	0.207	0.37	-0.61
$Slow$	0.051	0.138	0.37	-0.43
$Random$	0.076	0.182	0.42	-0.53
$Benchmark$	0.064	0.173	0.37	-0.54

Note: This table reports profit maximization measures for 120/80 equity strategy based on the state of the growth cycle induced by the models. Returns are monthly and annualized. The volatility corresponds to the annualized standard deviation. The performance to volatility ratio (PVR) compares the expected returns of an investment to the amount of risk undertaken to capture these returns. The Max drawdown (MDD) measures the largest single drop from peak to bottom in the value of a portfolio. * and ** indicate the model is in the set of best models $\widehat{M}^*_{90\%}$ and $\widehat{M}^*_{70\%}$, respectively. SPB refers to a boosting model based on squared error loss with P-splines, RF refers to a random forest model, BTB refers to a boosting model based on binomial loss function with decision trees, $CF^{BTB-RF-SPB}$ combines SPB,RF and BTB models, CF^{RF-SPB} combines SPB and RF models, CF^{RF-BTB} combines RF and BTB models, $CF^{SPB-BTB}$ combines SPB and BTB models, $Prob$ refers to the probit model based on the term spread, Acc classifies all data as "acceleration", $Slow$ classifies all data as "slowdown", $Random$ randomly assigns classes based on the proportions found in the training data and $Benchmark$ refers to the passive buy-and-hold investment strategy.

Table 6: **Summary of return and risk measures in the euro area: dynamic asset allocation**

	Average returns	Volatily	PVR	MDD
SPB	0.081**	0.094	0.86	-0.21
RF	0.080**	0.093	0.86	-0.22
BTB	0.075*	0.091	0.83	-0.22
$CF^{BTB-RF-SPB}$	0.079**	0.092	0.84	-0.22
CF^{RF-SPB}	0.078*	0.094	0.83	-0.22
CF^{RF-BTB}	0.080**	0.093	0.86	-0.22
$CF^{SPB-BTB}$	0.079**	0.093	0.85	-0.21
$Prob$	0.064	0.114	0.56	-0.25
Acc	0.060	0.137	0.44	-0.44
$Slow$	0.052*	0.070	0.75	-0.21
$Random$	0.064	0.115	0.55	-0.32
$Benchmark$	0.06	0.10	0.55	-0.34

Note: This table reports profit maximization measures for a dynamic asset allocation between bonds and equities based on the state of the growth cycle induced by the models. Returns are monthly and annualized. The volatility corresponds to the annualized standard deviation. The performance to volatility ratio (PVR) compares the expected returns of an investment to the amount of risk undertaken to capture these returns. The Max drawdown (MDD) measures the largest single drop from peak to bottom in the value of a portfolio. * and ** indicate the model is in the set of best models $\widehat{M}^*_{90\%}$ and $\widehat{M}^*_{70\%}$, respectively. SPB refers to a boosting model based on squared error loss with P-splines, RF refers to a random forest model, BTB refers to a boosting model based on binomial loss function with decision trees, $CF^{BTB-RF-SPB}$ combines SPB,RF and BTB models, CF^{RF-SPB} combines SPB and RF models, CF^{RF-BTB} combines RF and BTB models, $CF^{SPB-BTB}$ combines SPB and BTB models, $Prob$ refers to the probit model based on the term spread, Acc classifies all data as "acceleration", $Slow$ classifies all data as "slowdown", $Random$ randomly assigns classes based on the proportions found in the training data and $Benchmark$ refers to the passive buy-and-hold investment strategy.

As regards equity strategies, basic strategies based on the growth cycle induced by several models outperform the benchmark and no alternative classifiers are included in any best models sets. Selecting the best model is still complicated. In comparison with the "classical" case, $\widehat{M}^*_{70\%}$ turns out to be quite different and basic models such as SPB and RF are included and CF^{RF-SPB} is only attached to $\widehat{M}^*_{90\%}$. In this case, "BTB" is less effective than the others two basic models as "BTB" is only chosen in $\widehat{M}^*_{90\%}$. Importantly, "$CF^{SPB-BTB}$" is selected in the best models set for economists and for equity investors.

Dynamic asset allocation delivers a substantial improvement in risk-adjusted performance as compared to static asset allocation, especially for

investors who seek to avoid large losses. It is thus possible to rebalance the portfolio based on the stage of the growth cycle in real time in the euro area. SPB and RF perform well as those models belongs to $\widehat{M}^*_{70\%}$. "BTB" is again less effective than the others two basic models as "BTB" is only selected in $\widehat{M}^*_{90\%}$. As in the American case, the strategy based on $Slow$, which is overweighted in bonds, is selected in $\widehat{M}^*_{90\%}$, in line with the thrilling and non-reproducible performance of the bond market. $CF^{BTB-RF-SPB}$ is still selected in the best models set.

All in all, all results found for the United States also apply for the euro area. Depending on the data and the objective the best models set can be quite different. Ensemble machine learning algorithms perform well and combining forecast is proved to be useful. Again, "$CF^{BTB-RF-SPB}$", which is an equal weight combination of the three "basic" models, is selected in all best models sets.

Conclusion

Raffinot (2014) emphasizes that investment strategies based on the turning points of the growth cycle, better known as the output gap, achieve impressive risk-adjusted returns... in theory. But, in real time, economists often fail to detect if a new economic phase has already begun.

Over the last couple of decades, researchers in the machine learning community have developed more complex methods, also called ensemble learning, which improve prediction performances. Ensemble methods are learning models that achieve performance by combining the opinions of multiple learners. The two most popular techniques for constructing ensembles are random forests (Breiman (2001)) and boosting (Schapire (1990)). The two features of those algorithms are their abilities to entertain a large number of predictors and to perform estimation and variable selection simultaneously. Paradoxically, both methods work by adding randomness to the data (Varian (2014)), although they have substantial differences. Random forests rely on simple averaging of models in the ensemble and derive their strength from two aspects: randomizing subsamples of the training data and randomizing the selection of features. Boosting combines models that do not perform particularly well individually into one with much improved properties. It is an iterative process where the errors are kept being modelled.

Three models based on random forest and boosting algorithms are created to quickly and accurately detect growth cycle turning points in real time, in the United States and in the euro area. Those models are then combined, as averaging forecasts from different models often improves upon forecasts

based on a single model (Bates and Granger (1969)).

To assess the value of the models, profit maximization measures are employed in addition to more standard criteria, since, for investors, the usefulness of a signal depends on the rewards associated with the actions taken by the agent as a result of the forecast. When comparing predictive accuracy and profit measures, the model confidence set procedure (Hansen et al. (2011)) is applied to avoid data snooping, which occurs when a given set of data is used more than once for purposes of inference or model selection.

Ensemble machine learning algorithms are very effective to detect economic turning points. Strategies based on the turning points of the growth cycle achieve thus excellent risk-adjusted returns: macroeconomists can get rich nowcasting economic turning points. However, the selection of the best model is difficult. For instance, economists and investors would not always choose the same model. Moreover, depending on the data and the objective, random forest sometimes performs better than boosting, sometimes not. In the end, economists and investors should consider an equal weight combination of the three "basic" models, since this mix is selected in all best models sets.

Last but not least, this article opens the door for further research. An attempt to forecast growth cycle and business cycle turning points three to twelve months ahead could be very interesting.

Appendix A: Economic cycles

The classical business cycle definition is due to Burns and Mitchell (1946): *"Business cycles are a type of fluctuation found in the aggregate economic activity of nations that organize their work mainly in business enterprises: a cycle consists of expansions occurring at about the same time in many economic activities, followed by similarly general recessions, contractions, and revivals which merge into the expansion phase of the next cycle"*. The business cycle is meant to reproduce the cycle of the global level of activity. A cycle consists of expansions occurring at about the same time in many economic activities, followed by similarly general recessions, contractions, and revivals which merge into the expansion phase of the next cycle The turning points of that cycle (named B for peaks and C for troughs) separate periods of recessions from periods of expansions. Burns and Mitchell (1946) point out two main stylised facts of the economic cycle. The first is the co-movement among individual economic variables: most of macroeconomic time series evolve together along the cycle. The second is non-linearity: the effect of a shock depends on the rest of the economic environment. For instance, small shock, such as a decrease in housing prices, can sometimes have large effects.

The growth cycle, introduced by Mintz (1974), seeks to represent the fluctuations of the GDP around its long-term trend, which can be seen as the potential growth rate. The growth cycle is thus better known as the output gap. Mintz (1974) indicates that the rationale for investigating the growth cycle is that absolute prolonged declines in the level of economic activity tend to be rare events when the economy grows at a sustained and stable rate, so that in practice many economies do not very often exhibit recessions in classical terms, so other approaches to produce information on economic fluctuations have to be proposed. Growth cycle turning points (named A for peaks and D for troughs) have a clear meaning: peak A is reached when the growth rate decreases below the trend growth rate and the trough D is reached when the growth rate overpasses it again. Those downward and upward phases are respectively named slowdown and acceleration.

The description of different economic phases is refined by jointly considering the classical business cycle and the growth cycle: the ABCD approach (Anas and Ferrara (2004)). This framework improves the classical analysis of economic cycles by allowing two distinct phases or four distinct phases. During an acceleration phase, the current growth rate of the activity is above the long-term trend growth rate. The downward movement will first materialize when the growth rate will decrease below the long-term trend growth rate (point A). If the slowdown is not severe, the point A will be followed

by point D, when the growth rate overpasses its long-term trend. Otherwise, if the slowdown gains in intensity, the growth rate becomes negative enough to provoke a recession (point B). Eventually, the economy should start to recover and exits from the recession (point C). If the recovery strengthens, the growth rate should overpass its trend (point D), otherwise a double-dip will materialize (point B).

Hence, all recessions involve slowdowns, but not all slowdowns involve recessions.

Appendix B: Turning point chronology

The complete chronology is contained in the table 7:

Table 7: **Turning point chronology**

Euro area (Jan 1989-December 2013)		**United States (Jan 1985-Dec 2013)**	
Trough D	March 1991	Peak A	November 1985
Peak A	August 1993	Trough D	April 1987
Trough D	March 1991	Peak A	December 1989
Peak A	March 1995	Trough D	August 1991
Trough D	December 1996	Peak A	January 1993
Peak A	March 1998	Trough D	July 1993
Trough D	February 1999	Peak A	September 94
Peak A	December 2000	Trough D	March 1996
Trough D	September 2003	Peak A	June 2000
Peak A	May 2004	Trough D	February 2003
Trough D	May 2005	Peak A	October 2007
Peak A	October 2007	Trough D	September 2009
Peak B	March 2008	Peak A	June 2011
Trough C	April 2009	Trough D	December 2011
Trough D	August 2009		
Peak A	June 2011		
Peak B	August 2011		
Trough C	November 2012		
Trough D	March 2013		

Source: Raffinot (2014)

References

Anas, J. and Ferrara, L. (2004). Detecting Cyclical Turning Points: The ABCD Approach and Two Probabilistic Indicators. *Journal of Business Cycle Measurement and Analysis*, 2004(2):193–225.

Bates, J. and Granger, C. (1969). The combination of forecasts. *Operations Research Quarterly*.

Berge, T. (2015). Predicting Recessions with Leading Indicators: Model Averaging and Selection over the Business Cycle. *Journal of Forecasting*, 34(6):455–471.

Bergmeir, C., Hyndman, R. J., and Koo, B. (2015). A Note on the Validity of Cross-Validation for Evaluating Time Series Prediction. Technical report.

Boivin, J. and Ng, S. (2006). Are more data always better for factor analysis? *Journal of Econometrics*, 132(1):169–194.

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA.

Bühlmann, P. and Yu, B. (2003). Boosting with the L_2 loss: Regression and classification. *Journal of the American Statistical Association*, 98:324–338.

Burns, A. and Mitchell, W. (1946). *Measuring Business Cycles*. Number burn46-1 in NBER Books. National Bureau of Economic Research, Inc.

Camacho, M., Perez-Quiros, G., and Poncela, P. (2015). Extracting nonlinear signals from several economic indicators. *Journal of Applied Econometrics*, 30(7):1073–1089.

Caruana, R. and Niculescu-Mizil, A. (2005). An empirical comparison of supervised learning algorithms using different performance metrics. In *In Proc. 23 rd Intl. Conf. Machine learning*, pages 161–168.

Chauvet, M. and Senyuz, Z. (2012). A dynamic factor model of the yield curve as a predictor of the economy. Technical report.

Chauvet, M., Senyuz, Z., and Yoldas, E. (2015). What does financial volatility tell us about macroeconomic fluctuations? *Journal of Economic Dynamics and Control*, 52(C):340–360.

Claeskens, G., Magnus, J. R., Vasnev, A. L., and Wang, W. (2016). The forecast combination puzzle: A simple theoretical explanation. *International Journal of Forecasting*, 32(3):754 – 762.

Clemen, R. (1989). Combining forecasts: A review and annotated bibliography. *International Journal of forecasting*.

Duarte, A., Venetis, I. A., and Paya, I. (2005). Predicting real growth and the probability of recession in the euro area using the yield spread. *International Journal of Forecasting*, 21(2):261–277.

Efron, B. and Tibshirani, R. (1994). *An Introduction to the Bootstrap*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis.

Eilers, P. H. C. and Marx, B. D. (1996). Flexible smoothing with b-splines and penalties. *Statist. Sci.*, 11(2):89–121.

Freund, Y. and Schapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55:119–139.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29:1189–1232.

Friedman, J. H., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting (with discussion). *The Annals of Statistics*, 28:337–407.

Giusto, A. and Piger, J. (forthcoming). Identifying business cycle turning points in real time with vector quantization. *International Journal of Forecasting*.

Han, Y., Yang, K., and Zhou, G. (2013). A new anomaly: The cross-sectional profitability of technical analysis. *Journal of Financial and Quantitative Analysis*, 48:1433–1461.

Hanley, J. A. and McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36.

Hansen, P., Lunde, A., and Nason, J. (2011). The model confidence set. *Econometrica*, 79(2):453–497.

Hastie, T. (2007). Comment: Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22:513–515.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, 2 edition.

Hsu, Y.-C., Kuan, C.-M., and Yen, M.-F. (2013). A generalized stepwise procedure with improved power for multiple inequalities testing. IEAS Working Paper : academic research 13-A001, Institute of Economics, Academia Sinica, Taipei, Taiwan.

Ishwaran, H. (2007). Variable importance in binary regression trees and forests. *Electron. J. Statist.*, 1:519–537.

Liu, W. and Moench, E. (2014). What predicts U.S. recessions? Staff Reports 691, Federal Reserve Bank of New York.

Mintz, I. (1974). Dating united states growth cycles. In *Explorations in Economic Research, Volume 1, Number 1*, pages 1–113. National Bureau of Economic Research, Inc.

Ng, S. (2014). Viewpoint: Boosting recessions. *Canadian Journal of Economics*, 47(1):1–34.

Pesaran, H. and Timmermann, A. (1994). Forecasting stock returns: an examination of stock market trading in the presence of transaction costs. *Journal of forecasting*, 13(4):335–367.

Piger, J. (2011). *Econometrics: Models of Regime Changes*, pages 190–202. Springer New York, New York, NY.

Raffinot, T. (2007). A monthly indicator of GDP for Euro-Area based on business surveys. *Applied Economics Letters*, 14(4):267–270.

Raffinot, T. (2014). A clear, precise and efficient framework for macro-based investment decisions. Technical report.

Raffinot, T. (2015). Can macroeconomists get rich nowcasting output gap turning points with machine-learning? Technical report.

Romano, J. P. and Wolf, M. (2005). Stepwise multiple testing as formalized data snooping. *Econometrica*, 73(4):1237–1282.

Schapire, R. E. (1990). The strength of weak learnability. In *Machine Learning*, pages 197–227.

Schapire, R. E. (2003). *Nonlinear Estimation and Classification*, chapter The Boosting Approach to Machine Learning: An Overview, pages 149–171. Springer New York, New York, NY.

Taieb, S. B., Huser, R., Hyndman, R. J., and Genton, M. G. (2015). Probabilistic time series forecasting with boosted additive models: an application to smart meter data. Technical report.

Timmermann, A. (2006). *Forecast Combinations*, volume 1 of *Handbook of Economic Forecasting*, chapter 4, pages 135–196. Elsevier.

Varian, H. (2014). Big data: New tricks for econometrics. *Journal of Economic Perspectives*, 28(2):3–28.

White, H. (2000). A Reality Check for Data Snooping. *Econometrica*, 68(5):1097–1126.

Wu, B., Ai, H., and Liu, R. (2004). Glasses detection by boosting simple wavelet features. In *Pattern Recognition, 2004*, volume 1, pages 292–295 Vol.1.