**Using machine learning algorithms
to find patterns in stock prices^{*}**
by
Pedro N. Rodríguez^{}**
Simón Sosvilla-Rivero^{*}**
DOCUMENTO DE TRABAJO 2006-12

March 2006

* Pedro N. Rodríguez thanks CONACYT (Mexico) for financial support (Fellowship: 170328).

** Universidad Complutense de Madrid.

*** FEDEA and Universidad Complutense de Madrid.

Los Documentos de Trabajo se distribuyen gratuitamente a las Universidades e Instituciones de Investigación que lo solicitan. No obstante están disponibles en texto completo a través de Internet: <http://www.fedea.es>.

These Working Paper are distributed free of charge to University Department and other Research Centres. They are also available through Internet: <http://www.fedea.es>.

Abstract

We use a machine learning algorithm called *Adaboost* to find direction-of-change patterns for the S&P 500 index using daily prices from 1962 to 2004. The patterns are able to identify periods to take long and short positions in the index. This result, however, can largely be explained by first-order serial correlation in stock index returns.

JEL Classification Numbers: C45, G11, G14

Keywords: Direction-of-change predictability, Machine learning algorithms, Adaboost

1. – Introduction

Is a move upward or downward in stock prices predictable? A considerable amount of work has been devoted to examining whether or not this is feasible. Even though the presence of linear predictable components in stock returns is nowadays widely accepted [see, e.g., Fama (1965), Lo and MacKinlay (1988), Conrad and Kaul (1988), Jegadeesh (1990), and Kaul (1996)], the existence of a function (or formula) which expresses the likelihood of a market fluctuation is not.

However, recent advances in both analytic and computational methods have helped empirical investigation on the behavior of security prices. Particularly, direction-of-change (or sign) predictability is currently evaluated via either supervised learning techniques or machine learning algorithms or classifier induction techniques [see, e.g., Apte and Hong (1995), Tsaih, Hsu, and Lai (1998), Zemke (1999), Chen, Leung, and Daouk (2003), Kim (2003), and Rodriguez and Rodriguez (2004)]. Although this branch of research provides evidence in support of the existence of a function that discriminates up from down movements, it is not clear whether or not machine learning algorithms are extracting information beyond that contained in autocorrelation patterns.

In this paper, we reexamine the sample evidence of direction-of-change predictability in weak-form tests. In particular, we use an algorithm that is among the most popular and most successful algorithms for classification tasks called *Adaboost*. One of the main properties that make the application of Adaboost to financial data bases interesting is its relative (although not complete) robustness to over-fitting.

When we apply Adaboost to S&P 500 daily data, one main conclusion emerges about stock return predictability. We show that periods characterized by high first-order serial correlation in stock returns allow both in-sample and out-of-sample direction-of-change predictability. In essence, the lack of autocorrelation in stock returns does not permit Adaboost to discover a function that discriminate future upwards from downwards movements better than random. Indeed, simple random classifiers (i.e., coin-toss classifiers) are able to explain the apparent predictability in such periods.

In Section 2, we provide a brief review of machine learning algorithms and describe in detail the specific machine learning algorithm we use in our analysis: Adaboost. We apply this algorithm to the daily returns of the S&P 500 stock index from 1962 to 2004 and report the results in Section 3. To check the accuracy of our

predictions, we estimate several random classifiers and autoregressive models and the results are also given in Section 3. Finally, in Section 4 we offer some concluding remarks.

2. – Machine Learning Algorithms and Adaboost

The starting point for any study of stock return predictability is the recognition that prices, or more specifically, returns develop in either linear or nonlinear fashion over time and that the behavior contain certain stable patterns. In order to obtain those patterns, we start by declaring that stock price movements $\{y\}$ satisfy an expression like the following:

$$y = f(\mathbf{x}) \quad (1)$$

where $\mathbf{x}$ is a set of (lagged) “inputs” or “explanatory” variables $\mathbf{x} = \{x_1, \dots, x_p\}$ and y is the “output” or “response” variable $y \in C$, where C is the set of class labels.

When stock price movements are expressed as in Equation (1), it is evident that quantitative patterns may emerge from the application of machine learning algorithms, in which the goal is to find a function $f^*(\mathbf{x})$ that maps $\mathbf{x}$ to y , such that over the joint distribution of all pairs $(\mathbf{x}, y)$, the expected value of some specified loss function is minimized. But how useful is this function to discriminate financial movements?

To answer this question empirically, we must test the in-sample and out-of-sample discriminatory accuracy of machine learning algorithms trained to understand specific movements. Moreover, the performance must be compared against coin-toss classifiers to assess the statistical significance of such functions. In Section 2.A, we provide a brief review of the Adaboost algorithm, while Section 2 examines tree-based models, which are the cornerstone of such algorithm.

A. Adaboost

Boosting was created from the desire to transform a collection of weak classifiers into a strong ensemble or weighted committee. It is a general method for improving the performance of any learning algorithm. Boosting was proposed in the computational learning theory literature by Schapire (1990) and Freund (1995).

Freund and Schapire (1997) solved many practical difficulties of earlier boosting algorithms with the creation of Adaboost.

Much has been written about the success of Adaboost in producing accurate classifiers. In fact, one of the main characteristic of this procedure is that the test error seems to consistently decrease and then level off as more classifiers are added, without having an ultimately increase. The main steps of the Adaboost's algorithm are:

1. Start with weights $w_i = 1/n, i = 1, \dots, n$
2. For $m = 1, \dots, M$ do:
 - (a) Fit the machine learning algorithm $f_m^*(\mathbf{x})$ using weights w_i on the training data.
 - (b) Compute $\text{err}_m = E_w[1_{(y \neq f_m(x))}]$, $c_m = \log((1 - \text{err}_m) / \text{err}_m)$.
 - (c) Update $w_i \leftarrow w_i \exp[c_m \cdot 1_{(y_i \neq f_m^*(\mathbf{x}_i))}]$, $i = 1, \dots, n$ and renormalize so that $\sum w_i = 1$.
3. Output $f^*(\mathbf{x}) = \text{sign}[\sum_{m=1}^M c_m f_m(\mathbf{x})]$.

In this paper, we use tree-based models as base learners: the machine learning algorithm used to obtain $f_m^*(\mathbf{x})$ at each iteration. Such models are reviewed below. Note that at step m , those observations that were misclassified by the classifier $f_{m-1}^*(\mathbf{x})$ induced at the previous step have their weights increased; therefore, as Hastie *et al.* (2001, p. 300-301) clearly state “[e]ach successive classifier is thereby forced to concentrate on those training observations that are missed by previous one in the sequence.”

B. Tree-based models

The origins of classification trees or hierarchical classification come from two areas of investigation. In the field of statistical pattern recognition, Breiman, Friedman, Olshen, and Stone (1984) developed a technique named CART (Classification and Regression Trees). The Machine Learning community provided a computer program called ID3, which evolved into a new system named C4.5 [see Quinlan (1986, 1993)].

Tree-based techniques involve partitioning the explanatory variables space into a set of rectangles and then fit a simple model to each one. A tree-based model

tries to find the split that maximizes the decrement in a loss function in order to make a tree grow. This is done iteratively until a certain amount of observations is reached or no further decrements in the loss function are found. More formally, a tree may be expressed as,

$$T(\mathbf{x}; \Theta) = \sum_{j=1}^J \gamma_j I(\mathbf{x} \in R_j), \quad (2)$$

with parameters $\Theta = \{R_j, \gamma_j\}_1^J$. Where γ_j (a constant) is assigned to a region (R_j). The constant can be a value, a probability or a class label assigned to an element in the region R_j . J is usually treated as a meta-parameter and can be interpreted as the maximum amount of admissible interactions among explanatory variables less one, and $I(\bullet)$ is an indicator function. It is worth mentioning that J also represents the stopping criteria of the top-down algorithm of the tree-based models (briefly described below) and that we fixed to two. In other words, we use the so-called stumps. Here the parameters $\Theta = \{R_j, \gamma_j\}_1^J$ are found by minimizing the empirical risk, like in the following equation:

$$\hat{\Theta} = \arg \min_{\Theta} \sum_{j=1}^J \sum_{\mathbf{x}_i \in R_j} L(y_i, \gamma_j) \quad (3)$$

where $L(\bullet)$ denotes a loss function. This is an extraordinary combinatorial optimization problem, so we must relay on sub-optimal solutions. The aforementioned optimization problem can be divided into two parts. The first one, finding γ_j given R_j , is typically trivial, where $\hat{\gamma}_j$ is the modal class of observations falling in region R_j . The difficulty of this combinatorial optimization problem is based on finding R_j . A helpful solution is to employ heuristic methods.

Safavian and Landgrebe (1991) provide a survey on heuristic methods proposed for designing decision trees. The most popular heuristic method in tree-based models is the top-down recursive partitioning, which starts with a single region covering the entire space of all joint input values. This is partitioned into two regions by choosing an optimal splitting input variable x_j and a corresponding optimal split point s . Values in $\mathbf{x}$ for which $x_j \leq s$ are defined to be the left daughter region, and those for which $x_j > s$ denote the right daughter region. Then each of this two daughter regions is optimally partitioned with the same strategy, and so forth.

In this paper, we replaced the loss function with the Gini index, given by

$$\text{Gini index} : \sum_{k=1}^k \hat{\rho}_{mk} (1 - \hat{\rho}_{mk}) \quad (4)$$

where in a node m , representing a region R_j , let $\hat{\rho}_{mk}$ be the proportion of class k observations in the node m , and K represents the total number of classes or populations under study. The implementation was carried out in R: Environment for Statistical Computing and Graphics with the *ada* add-on package [developed by Culp, Johnson, and Michailidis (2005)].

3. – Empirical Results

The empirical application uses S&P 500 daily closing prices from August 7, 1962 to December 31, 2004. The data set was divided into four non-overlapping sets. Since it is well-known that the ultimate measure of quality of a learner is its generalization performance, we divided each set into two sub-samples. The first sub-sample is used for training, whereas the second sub-sample is used for testing. We assume that future stock price movements $\{y\}$ may be related to past returns, as in the following equation:

$$y_i = f(r_{i-1}, \dots, r_{i-24}), i = 1, \dots, n \quad (5)$$

where y_i equals to 1 if the return observed at time i $\{r_i\}$ is greater than zero, 0 otherwise. We applied the algorithm described in Section 2.A to the each training set. The results are shown in Table 1.

Table 1 displays several accuracy measures, as explained below. We use these accuracy measures to analyze the properties of Adaboost's in-sample and out-of-sample performance. We study six specifications corresponding to iterations $\{M\}$ equal to 1, 2, 25, 50, 200, and 1000.

Table 1. Direction-of-change predictability of Adaboost.

A. First sub-sample. Training (19620807-19681112) and Testing (19681113-19721229)								
Specification	In-sample evidence				Out-of-sample evidence			
	Error	Bias	Vu	Vb	Error	Bias	Vu	Vb
M = 1	0.4136	0.4405	0.2427	0.2697	0.3895	0.4715	0.2246	0.3066
M = 2	0.4136	0.4405	0.2427	0.2697	0.3895	0.4715	0.2246	0.3066
M = 25	0.3911	0.4405	0.177	0.2260	0.4136	0.4715	0.1793	0.2372
M = 50	0.3898	0.4405	0.175	0.2261	0.4127	0.4715	0.1765	0.2353
M = 200	0.3866	0.4405	0.117	0.1708	0.4147	0.4715	0.1369	0.1938
M = 1000	0.3230	0.4405	0.112	0.2293	0.4272	0.4715	0.1745	0.2189
B. Second sub-sample. Training (19730102-19790808) and Testing (19780809-19831230)								
Specification	In-sample evidence				Out-of-sample evidence			
	Error	Bias	Vu	Vb	Error	Bias	Vu	Vb
M = 1	0.4353	0.4994	0.0803	0.1445	0.4658	0.4766	0.1061	0.1169
M = 2	0.4353	0.4994	0.0803	0.1445	0.4658	0.4766	0.1061	0.1169
M = 25	0.4107	0.4994	0.21	0.2992	0.4604	0.4766	0.2437	0.2599
M = 50	0.4053	0.4994	0.174	0.2686	0.4667	0.4766	0.2221	0.2320
M = 200	0.3783	0.4994	0.181	0.3022	0.4739	0.4766	0.2338	0.2365
M = 1000	0.3237	0.4994	0.153	0.3291	0.4937	0.4766	0.2572	0.2401
C. Third sub-sample. Training (19840103-19900808) and Testing (19900809-19941230)								
Specification	In-sample evidence				Out-of-sample evidence			
	Error	Bias	Vu	Vb	Error	Bias	Vu	Vb
M = 1	0.5494	0.4506	0.5494	0.4506	0.5198	0.4802	0.5198	0.4802
M = 2	0.5494	0.4506	0.5494	0.4506	0.5198	0.4802	0.5198	0.4802
M = 25	0.4446	0.4506	0.029	0.0348	0.4847	0.4802	0.0207	0.0162
M = 50	0.4230	0.4506	0.081	0.1084	0.4838	0.4802	0.0791	0.0755
M = 200	0.3913	0.4506	0.116	0.1756	0.4784	0.4802	0.1214	0.1232
M = 1000	0.3355	0.4506	0.104	0.2187	0.4703	0.4802	0.1466	0.1565
D. Forth sub-sample. Training (19950103-20001226) and Testing (20011227-20041231)								
Specification	In-sample evidence				Out-of-sample evidence			
	Error	Bias	Vu	Vb	Error	Bias	Vu	Vb
M = 1	0.5410	0.4590	0.5410	0.4590	0.5084	0.4916	0.5084	0.4916
M = 2	0.5410	0.4590	0.5410	0.4590	0.5084	0.4916	0.5084	0.4916
M = 25	0.4187	0.4590	0.13	0.1700	0.4945	0.4916	0.1440	0.1410
M = 50	0.3981	0.4590	0.083	0.1442	0.5055	0.4916	0.1470	0.1331
M = 200	0.3724	0.4590	0.101	0.1872	0.4985	0.4916	0.1698	0.1629
M = 1000	0.3247	0.4590	0.1210	0.2553	0.4826	0.4916	0.1917	0.2006

The in-sample and out-of-sample accuracy measures are: (a) the error rate, which corresponds to the total number of misclassified observations divided by the total number of observations; (b) the bias, defined as systematic loss incurred by the function; (c) the unbiased variance (denoted by Vu) evaluates the extent to which the estimated function deviates from the correct predictions; and (d) the biased variance (denoted as Vb) assesses the extent to which the estimated function deviates from the incorrect predictions. These accuracy measures are described in detail in Domingos (2000) and Valentini and Dietterich (2004). Rodríguez and Sosvilla-Rivero (2006) provide an application of the bias-variance decomposition to stock return predictability.

In the first two data sets, the bias plays a significant role in its contribution to the error rate. In other words, the systematic loss incurred by the functions is higher than the total error rate. Moreover, Adaboost's error has a positive relationship with the total number of iterations $\{M\}$. Evidently, this later result indicates that Adaboost rapidly over-fits the data.

In contrast, in the last two data sets, the bias is lower than the error rate. This only occurs when the loss incurred by function's fluctuations around the central tendency in response to different samples has a direct effect on error. Furthermore, the error decrease has the number of iterations increases. But to what extent randomness can explain the results provided by Adaboost?.

To answer this question, we simulated 1000 coin-toss classifiers for each data set. To obtain each random classifier, we generate random values from a discrete distribution in which two values were possible: 1's and 0's. Each value was assigned 50 per cent of probabilities. The results for each data set are shown in Table 2.

Table 2. Random predictability

A. First sub-sample. Training (1962/08/07-1968/11/12) and Testing (1965/11/13-1972/12/29)							
Percentil	0.01	0.05	0.25	0.50	0.75	0.95	0.99
In-sample error	0.4701	0.4804	0.4907	0.4997	0.5080	0.5202	0.5292
Out-of-sample error	0.4648	0.4754	0.4899	0.4995	0.5092	0.5236	0.5362
B. Secon sub-sample. Training (1973/01/02-1979/08/08) and Testing (1978/08/09-1994/12/30)							
Percentil	0.01	0.05	0.25	0.50	0.75	0.95	0.99
In-sample error	0.4724	0.4802	0.4922	0.5000	0.5078	0.5204	0.5288
Out-of-sample error	0.4685	0.4757	0.4901	0.5009	0.5108	0.5234	0.5324
C. Third sub-sample. Training (1984/01/03-1990/08/08) and Testing (1990/08/09-1994/12/30)							
Percentil	0.01	0.05	0.25	0.50	0.75	0.95	0.99
In-sample error	0.4727	0.4805	0.4924	0.4997	0.5075	0.5195	0.5261
Out-of-sample error	0.4649	0.4748	0.4901	0.5009	0.5108	0.5252	0.5342
D. Forth sub-sample. Training (1995/01/03-2000/12/26) and Testing (2001/12/27-2004/12/31)							
Percentil	0.01	0.05	0.25	0.50	0.75	0.95	0.99
In-sample error	0.4702	0.4795	0.4914	0.4993	0.5099	0.5212	0.5284
Out-of-sample error	0.4667	0.4747	0.4886	0.4995	0.5104	0.5263	0.5353

Table 2 shows the distribution of the error rate of the random classifiers. As can be seen, randomness can explain up to 46 percent, approximately, of out-of-sample errors. Thus, classifiers achieving higher out-of-sample error rates can be considered as random. In fact, only in the first two data sets, Adaboost was able to obtain lower out-of-sample error rates. But what are the factors that affect Adaboost's ability to discriminate stock price movements?

One possible way to answer this question is to gauge traditional benchmarks. In doing so, we can evaluate whether or not simple linear models are able to explain Adaboost's predictability. To that end, we estimated a simple first-order autoregressive model for each period, and the results are shown in Table 3.

Table 3. Direction-of-change predictability of an AR(1) model.

A. First sub-sample. Training (1962/08/07-1968/11/12) and Testing (1968/11/13-1972/12/29) Model: $R_t = 0.0005313117 + 0.1417891R_{t-1}$							
In-sample evidence				Out-of-sample evidence			
Error	Bias	Vu	Vb	Error	Bias	Vu	Vb
0.4248	0.4402	0.0790	0.0944	0.4011	0.4715	0.0848	0.1552
B. Second sub-sample. Training (1973/01/02-1979/08/08) and Testing (1978/08/09-1983/12/30) Model: $R_t = 0.0001423972 + 0.22248261R_{t-1}$							
In-sample evidence				Out-of-sample evidence			
Error	Bias	Vu	Vb	Error	Bias	Vu	Vb
0.4259	0.4997	0.1968	0.2705	0.4676	0.4766	0.2185	0.2275
C. Third sub-sample. Training (1984/01/03-1990/08/08) and Testing (1990/08/09-1994/12/30) Model: $R_t = 0.0006423667 + 0.05325351R_{t-1}$							
In-sample evidence				Out-of-sample evidence			
Error	Bias	Vu	Vb	Error	Bias	Vu	Vb
0.4514	0.4502	0.0330	0.0318	0.4838	0.4802	0.0234	0.0198
D. Forth sub-sample. Training (1995/01/03-2000/12/26) and Testing (2001/12/27-2004/12/31) Model: $R_t = 0.003193694 + 0.0008291922R_{t-1}$							
In-sample evidence				Out-of-sample evidence			
Error	Bias	Vu	Vb	Error	Bias	Vu	Vb
0.4586	0.4586	0.0000	0.0000	0.4916	0.4916	0.0000	0.0000

Table 3 displays the same accuracy measures as Table 1. In addition, Table 3 shows the AR(1) model estimated in the training sample of each data set. Not surprisingly, a simple autoregressive model is able to obtain very similar direction-of-change predictability as Adaboost. Similar to Table 1, the autoregressive models are able to obtain in-sample predictability but fail to detect out-of-sample predictability in the last two data sets.

The disappearance of the predictability documented here is consistent with Allen and Karjalainen's (1999) finding of usefulness of daily prices, at the presence of positive low-order serial correlation in S&P 500 returns, to identify periods to be in the S&P 500 index when returns are positive and volatility is low and out when the reverse is true.

4. – Concluding remarks

In this paper, we have implemented a classifier induction approach to analyze the sample evidence on return predictability. We obtain the following general results. First, periods characterized by high first-order serial correlation in stock returns allow both in-sample and out-of-sample direction-of-change predictability. Specifically, a powerful machine learning algorithm called *Adaboost* is able to find a stable function which discriminate, better than random, upward from downward movements.

Second, *Adaboost* does over-fit. Functions induced in periods characterized by the lack of autocorrelation in stock returns are able to obtain in-sample predictability but fail to detect out-of-sample predictability. In fact, in many cases, *Adaboost* out-of-sample performance decreases as more iterations are run. We have also examined different *Adaboost*'s specifications, such as using 4- and 8-node tree-based models instead of stumps, and achieve faster over-fitting.

There are several natural extensions to our analysis. First, machine learning algorithms can be used to examine large price change predictability. They can also be modified to study predictability of large absolute price movements, which are useful for option trading strategies. Second, machine learning algorithms are sufficient flexible to examine the performance of nested models. For example, one can induce classifiers for small-cap indices using small-cap's or large-cap's lags, and evaluate the lead-lag effect in terms of movement predictability. Finally, machine learning algorithms can be used to identify risk exposures. For instance, we can codify costly lower-tail outcomes and search for "inputs" or "explanatory" variables that help a machine learning algorithm discriminate the costly lower-tail outcomes from the remainder of outcomes. We hope to explore these issues more fully in future research.

References

- Allen, Franklin, and Risto Karjalainen, 1999, Using genetic algorithms to find technical trading rules, *Journal of Financial Economics* 51, 245-271.
- Apte, Chidanand, and Se June Hong, 1995, Predicting equities returns from securities data with minimal rule generation, Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy Uthurusamy, eds.: *Advances in Knowledge Discovery and Data Mining* (AAAI Press).
- Breiman, Leo, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone, 1984, *Classification and Regression Trees* (Wadsworth, Belmont, California).
- Chen, An-Sing, Mark T. Leung, and Hazem Daouk, 2003, Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan stock index, *Computers and Operations Research* 30, 901-923.
- Conrad, Jennifer, and Gautam Kaul, 1988, Time varying expected returns, *Journal of Business* 61, 409-425.
- Culp, Mark, Kjell Johnson, and George Michailidis, 2005, ada: an R package for boosting, *Journal of Statistical Software*, forthcoming.
- Domingos, Pedro, 2000, A unified bias-variance decomposition for zero-one and squared loss. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence* (Austin, TX, AAAI Press) 564-569.
- Fama, Eugene F., 1965, The behavior of stock market prices, *Journal of Business* 30, 34-105.
- Freund, Yoav, 1995, Boosting a weak learning algorithm by majority, *Information and Computation* 121, 256-285.
- Freund, Yoav, and Robert E. Schapire, 1997, A decision-theoretic generalization of on-line learning and an application to boosting, *Journal of Computer and Systems Sciences* 55, 119-139.
- Hastie, Trevor, Robert Tibshirani, and Jerome H. Friedman, 2001. *The Elements of Statistical Learning: Data Mining, Inference and Prediction* (Springer-Verlag, New York).
- Jegadeesh, Narasimhan, 1990, Evidence of predictable behavior of security returns, *Journal of Finance* 45, 881-898.
- Kaul, Gautam, 1996, Predictable components in stock returns, in G.S. Maddala and C.R. Rao, eds.: *Handbook of Statistics, Vol. 14, Statistical Methods in Finance* (Elsevier Science B.V., The Netherlands).
- Kim, Kyoung-jae, 2003, Financial time series forecasting using support vector machines, *Neurocomputing* 55, 307-319.
- Lo, Andrew W., and A. Craig MacKinlay, 1988, Stocks market prices do not follow random walks: Evidence from a simple specification test, *Review of Financial Studies* 1, 41-66.

- Quinlan, J. Ross, 1986, Induction of decision trees, *Machine Learning* 1, 81-106.
- Quinlan, J. Ross, 1993, *C4.5: Programs for Machine Learning* (Morgan Kaufmann, San Mateo, California).
- Rodriguez, Pedro N., and Arnulfo Rodriguez, 2004, Predicting stock market indices movements, in Marco Costantino and Carlos Brebbia, eds.: *Computational Finance and its Applications* (Wessex Institute of Technology, Southampton).
- Rodriguez, Pedro N., and Simon Sosvilla-Rivero, 2006, Understanding and forecasting stock price changes, FEDEA Working Paper 2006-03. Available at http://papers.ssrn.com/sol3/papers.cfm?abstract_id=882028.
- Safavian, S. Rasoul, and David Landgrebe, 1991, A survey of decision tree classifier methodology, *IEEE Transactions on Systems, Man, and Cybernetics* 21, 660-674.
- Schapire, Robert E., 1990, The strength of weak learnability, *Machine Learning* 5, 197-227.
- Tsaih, Ray, Yenshan Hsu, and Charles C. Lai, 1998, Forecasting S&P 500 stock index futures with a hybrid AI system, *Decision Support System* 23, 161-174.
- Valentini, Giorgio, and Thomas G. Dietterich, 2004, "Bias-variance analysis of support vector machines for the development of SVM-based ensemble methods," *Journal of Machine Learning Research*, 5, 725-775.
- Zemke, Stefan, 1999, Nonlinear index prediction, *Physica A* 269, 177-183.

ÚLTIMOS DOCUMENTOS DE TRABAJO DE FEDEA

- 2006-12: “Using machine learning algorithms to find patterns in stock prices”, **Pedro N. Rodríguez y Simón Sosvilla-Rivero**.
- 2006-11: “Regulation of Legal and Medical Professions in the US and Europe: A Comparative Analysis”, **Nuno Garoupa**.
- 2006-10: “Secondhand Market and the Lifetime of Durable Goods”, **Omar Licandro, Luis A. Puch y Antonio R. Sampayo**.
- 2006-09: “A Law and Economics Perspective on Terrorism”, **Nuno Garoupa, Jonathan Klick y Francesco Paris**.
- 2006-08: “El diferencial entre las tasas de desempleo de hombres y mujeres en Argentina”, **Ana Carolina Ortega Masagué**.
- 2006-07: “Efectos de las ayudas europeas sobre la economía Madrileña 2007-2013: Un análisis basado en el Modelo Hermin”, **Simón Sosvilla-Rivero y Emma García**.
- 2006-06: “Further evidence about alcohol consumption and the business cycle”, **Sergi Jiménez-Martín, José M. Labeaga, y Cristina Vilaplana Prieto**.
- 2006-05: “Las dimensiones del bienestar en las Comunidades Autónomas Españolas. Un análisis de Sigma y Gamma – Convergencia”, **Andrés J. Marchante Mera y Bienvenido Ortega Aguaza y José Sánchez Maldonado**.
- 2006-04: “Efectos de las ayudas europeas sobre la economía Española, 2000-2006: Un análisis basado en el Modelo Hermin”, **Simón Sosvilla-Rivero y Emma García**.
- 2006-03: “Understanding and Forecasting Stock Price Changes”, **Pedro N. Rodríguez y Simón Sosvilla-Rivero**.
- 2006-02: “A Macro and Microeconomic Integrated Approach to Assessing the Effects of Public Policies”, **Xavier Labandeira, José M. Labeaga y Miguel Rodríguez**.
- 2006-01: “The Dynamics of Regional Inequalities”, **Salvador Barrios y Eric Strobl**.
- 2005-28: “New European Member States and the Dependent Elderly”, **Corinne Mette**.
- 2005-27: “Efectos del Programa Operativo Integrado de Castilla-La Mancha, 2000-2006: Un análisis basado en el modelo Hermin”, **Simón Sosvilla-Rivero y Emma García**.
- 2005-26: “It's a Small Small Welfare Cost of Fluctuations”, **Franck Portier y Luis A. Puch**.
- 2005-25: “Obsolescence and Productivity”, **Fernando del Rio y Antonio R. Sampayo**.
- 2005-24: “EU Structural Funds and Spain's Objective 1 Regions: An Analysis Based on the Hermin Model”, **Simón Sosvilla-Rivero**.
- 2005-23: “A sequential model for older workers' labor transitions after a health shock”, **Sergi Jiménez-Martín, José M. Labeaga y Cristina Vilaplana Prieto**.
- 2005-22: “Price Convergence in the European Car Market”, **Salvador Gil-Pareja y Simón Sosvilla-Rivero**.
- 2005-21: “Implicit Regimes for the Spanish Peseta/Deutschmark Exchange Rate”, **Francisco Ledesma-Rodríguez, Manuel Navarro-Ibáñez, Jorge Pérez-Rodríguez y Simón Sosvilla-Rivero**.
- 2005-20: “A Projection of Spanish Pension System under Demographic Uncertainty”, **Namkee Ahn, Javier Alonso-Meseguer y Juan Ramón García**.
- 2005-19: “The Dynamic of temporary jobs: Theory and Some Evidence for Spain (The Role of Skill)”, **Elena Casquel y Antoni Cunyat**.
- 2005-18: “The Welfare Cost of Business Cycles in an Economy with Nonclearing Markets”, **Franck Portier y Luis A. Puch**.
- 2005-17: “Life Satisfaction among Spanish Workers: Importance of Intangible Job Characteristics”, **Namkee Ahn**.
- 2005-16: “Persistence and ability in the innovation decisions”, **José M. Labeaga y Ester Martínez-Ros**.
- 2004-15: “Measuring Changes in Health Capital”, **Néboa Zozaya, Juan Oliva y Rubén Osuna**.