# Financial stability indicator predictability by support vector machines

Kristýna Ivanková[1]

**Abstract.** Support Vector Machines are a successful machine-learning algorithm used for classification, regression and prediction of time series. We optimize learning parameters and select the feature set with the smallest predictive error. We also explore the development of errors for predictions with larger time skips.

We've applied the method for the prediction of CISS (Composite Indicator of Systemic Stress), a stability indicator created by the European Central Bank. We've chosen this indicator among other state-of-the-art indicators because of its high frequency, which indicates a quick response to distinctive changes on the market.

The results show that CISS can be partially explained just by its past values up to six to eight weeks ahead, but large behaviour shifts are still surprising for the model. We've also discovered that including data over three months of age into the prediction context won't improve the results.

**Keywords:** SVM, prediction, financial stability, systemic stress, CISS.

**JEL classification:** C13, C45, G32.
**AMS classification:** 68T10, 62H30, 62P20.

## 1 Introduction

The ongoing financial crisis has motivated lots of financial instability research in the recent years. Many authors began to study the notions of systemic risk and systemic stress. Referencing the definitions from Holló [4]:

> Systemic risk can be defined as the risk that financial instability becomes so widespread that it impairs the functioning of a financial system to the point where economic growth and welfare suffer materially. Systemic stress is interpreted as that amount of systemic risk which has materialised.

Systemic stress is usually accompanied by increases in uncertainty, disagreement among investors, asymmetry of information between borrowers and lenders, risk aversion, flight to quality or flight to liquidity.

One of the recurring research themes is the introduction of novel indicators of financial stability (or financial stress). These indicators capture the current or past financial situation. Some examples of actively used indicators follow.

- The Laeven-Valencia index of systemic banking crises introduced in [7] is a broad, coincident indicator for systemic financial crises. It covers 169 countries and specifies the presence or absence of a crisis with yearly frequency since 1970.
- The Financial Stress Index (FSI) for 17 advanced economies [3] and its modified version for 26 emerging economies [1] introduced by the International Monetary Fund. Both are monthly indicators of national financial system strain and rely on price movements relative to past levels or trends. The FSI-AE index is defined for most of its countries since 1980, the FSI-EM since the end of 1996.

[1] Charles University in Prague, Institute of Economic Studies, Opletalova 26, 110 00, Praha 1, kristyna.ivankova@gmail.com

- A Composite Indicator of Systemic Stress (CISS) in the financial system [4] introduced in the European Central Bank. This is a short-term (weekly) composite systemic stress measure for the EU sector aggregating five market-segment subindices: money market, bond market, equity market, financial intermediaries and foreign exchange market. The combined index is based on cross-correlation of subindices after a rank transformation. The original CISS is defined only after the introduction of the Euro (1999), but the authors present a version extended up to 1987.

It would prove useful to be able to predict systemic stress and handle an emerging crisis before it actually breaks out. We will focus on CISS since the other indicators are too sparse for our purposes.

In section 2 we present a theoretical overview of our prediction methodology and in Section 3 we discuss its application to the prediction of CISS for various delays and present the obtained results.

## 2 Support vector machine regression

Support Vector Machines (SVMs) are a machine learning method introduced by Vapnik [10]. Good introductions into the SVMs are [2], [9]; for an application of SVMs to prediction of time series see [8].

Suppose we are given examples $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, where $\mathbf{x}_i \in \mathbb{R}^d$ are the training contexts and $y_i \in \mathbb{R}$ are training targets. We seek a function $f(\mathbf{x})$ that maps contexts to targets and will predict new, previously unseen data well. Predictions $\tilde{y}$ for an unseen context $\tilde{\mathbf{x}}$ will be done by evaluating $f(\tilde{\mathbf{x}})$.

For the special problem of predicting a time series, the contexts $\mathbf{x}_i$ may consist of past values, their transformations, and external data.

### 2.1 The linear case

Let us first consider the linear case $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b$ where $\langle \cdot, \cdot \rangle$ denotes the dot product. Following Smola and Schölkopf [9], we will penalize training errors by the standard $\varepsilon$-insensitive loss function ([9], Equation 4)

$$|z|_\varepsilon = \begin{cases} 0 & \text{if } |z| \le \varepsilon, \\ |z| - \varepsilon & \text{otherwise.} \end{cases}$$

We prevent overfitting by adding the regularization term $\|w\|^2$. The trade-off between accuracy on the training set and model complexity will be controlled by the regularization constant $C > 0$.

We can now formulate the optimization problem ([9], Equation 3):

$$\begin{array}{ll} \underset{\mathbf{w},\, b,\, \xi_i,\, \xi_i^*}{\text{minimize}} & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ \text{subject to} & -\varepsilon - \xi_i \le \langle \mathbf{w}, \mathbf{x}_i \rangle + b - y_i \le \varepsilon + \xi_i^*, \quad i \in \{1, \ldots, n\} \\ & \xi_i,\ \xi_i^* \ge 0, \quad i \in \{1, \ldots, n\}. \end{array}$$

The Lagrange dual problem has the form

$$\begin{array}{lll} \underset{\alpha_i,\, \alpha_i^*,\, \eta_i,\, \eta_i^*}{\text{maximize}} & \underset{\mathbf{w},\, b,\, \xi_i,\, \xi_i^*}{\inf} & \Big[ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ & & - \sum_{i=1}^{n} \alpha_i \left( \langle \mathbf{w}, \mathbf{x}_i \rangle + b - y_i + \varepsilon + \xi_i \right) \\ & & + \sum_{i=1}^{n} \alpha_i^* \left( \langle \mathbf{w}, \mathbf{x}_i \rangle + b - y_i - \varepsilon - \xi_i^* \right) \\ & & - \sum_{i=1}^{n} (\eta_i \xi_i + \eta_i^* \xi_i^*) \Big] \\ \text{subject to} & & \alpha_i, \alpha_i^*, \eta_i, \eta_i^* \ge 0, \quad i \in \{1, \ldots, n\}. \end{array}$$

The gradient of the Lagrangian function with respect to the primary variables must vanish at the infimum. This gives the following set of constraints:

$$\begin{aligned}\sum_{i=1}^{n}(\alpha_i^* - \alpha_i) &= 0,\\ \mathbf{w} &= \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\mathbf{x}_i,\\ \eta_i &= C - \alpha_i,\\ \eta_i^* &= C - \alpha_i^*.\end{aligned}$$

After substitution we finally arrive at ([9], Equation 10):

$$\begin{aligned}&\underset{\alpha_i,\,\alpha_i^*}{\text{maximize}} && -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\,\langle \mathbf{x}_i, \mathbf{x}_j\rangle \;-\; \varepsilon\sum_{i=1}^{n}(\alpha_i + \alpha_i^*) \;+\; \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*)\\ &\text{subject to} && \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0,\\ &&& 0 \le \alpha_i \le C, \quad 0 \le \alpha_i^* \le C, \quad i \in \{1, \ldots, n\}.\end{aligned}$$

This is a quadratic convex programming problem with linear constraints and a dense dot product matrix and as such it can be solved using well-known quadratic programming algorithms. For notes about computing $b$ see [9].

Let us note that $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x}\rangle + b$ can be rewritten as $\sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\langle \mathbf{x}_i, \mathbf{x}\rangle + b$. Furthermore, it can be shown [9] that Lagrange multipliers $\alpha_i$, $\alpha_i^*$ that correspond to input vectors lying inside the $\varepsilon$-insensitive region $|f(\mathbf{x}_i) - y_i| < \varepsilon$ vanish in the solution. This means that only a subset of the input examples is needed to describe $f$; examples from this subset are called the *Support Vectors*.

### 2.2 The non-linear case

We will now describe the *kernel trick* which makes SVMs the algorithm of choice for many machine-learning settings.

We want to create a (generally non-linear) mapping $\phi$ that maps inputs $\mathbf{x}_i$ into a high-dimensional *feature space* $F$, and perform the whole computation in $F$.

We've established earlier that both the optimization problem and function evaluation require only dot products between input examples. The trick is to choose a *kernel* function that corresponds to the dot product in $F$,

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle_F,$$

and can be evaluated directly from the contexts without computing the map $\phi$. This allows us to perform computations in the larger space $F$ with no extra costs.

The most popular kernel choices for SVM regression are

- polynomial kernels $\quad k(\mathbf{x}_i, \mathbf{x}_j) = \big(\langle \mathbf{x}_i, \mathbf{x}_j\rangle + 1\big)^p$,
- Gaussian radial basis function (RBF) kernels $\quad k(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{e}^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2/\,2\sigma^2}$, and
- sigmoidal kernels $\quad k(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\kappa\langle \mathbf{x}_i, \mathbf{x}_j\rangle - c)$.

For problems with a large number of training examples, the basic linear kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j\rangle$ also enjoys some popularity because its special structure enables significant speed improvements (see e.g. [6]).

## 3 Application

We've divided the CISS dataset $\boldsymbol{a}$ into the training set (from Jan 1991 to Sep 2007) and the test set (Oct 2007 to Dec 2010). We've tried to learn a model for the CISS value $\delta$ weeks ahead:

$$y_i = a_{i+\delta} \quad \text{with } \delta \in \{1, 2, 4, 6, 8\}.$$

To find the SVM model with the least predictive error, we need to optimize the learning parameters. These consist of

- the margin width $\varepsilon$,
- the regularization parameter $C$, and
- the kernel parameters.

The objective function was chosen to be the average absolute prediction error on the test set.

After several trial runs, we've fixed the kernel class to Gaussian RBFs (it consistently gave the best results).

The training contexts $\mathbf{x}_i$ originally consisted only of the history $\{a_{i-H}, \ldots, a_i\}$ (where $H$ is the age of the oldest sample in the context), but we've found out that adding simple transformations improves the results. The transformation set with the best results turned out to be

$$\mathbf{x}_i = \bigcup_{h=0}^{H} \{a_{i-h}, a_{i-h}^2, \log a_{i-h}, m_{i,h}, \log m_{i,h}, M_{i,h}, \log M_{i,h}\}$$

where $m_{i,h} = \min\{a_{i-h}, \ldots, a_i\}$ and $M_{i,h} = \max\{a_{i-h}, \ldots, a_i\}$ are moving minima and maxima.

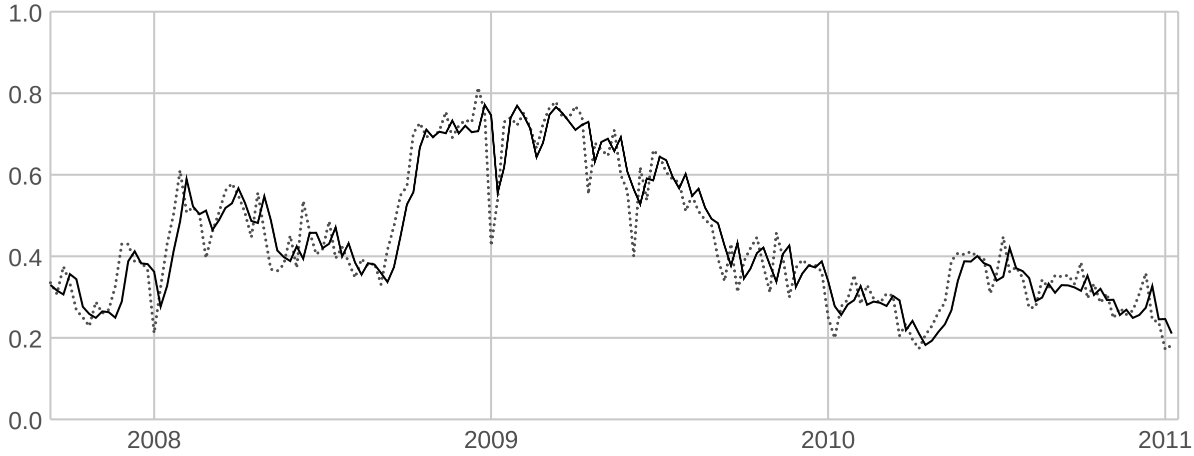


**Figure 1** CISS predictions one week into the future, average absolute error = 0.0487. The dotted line denotes the actual value of the index, the solid line denotes the predicted value.

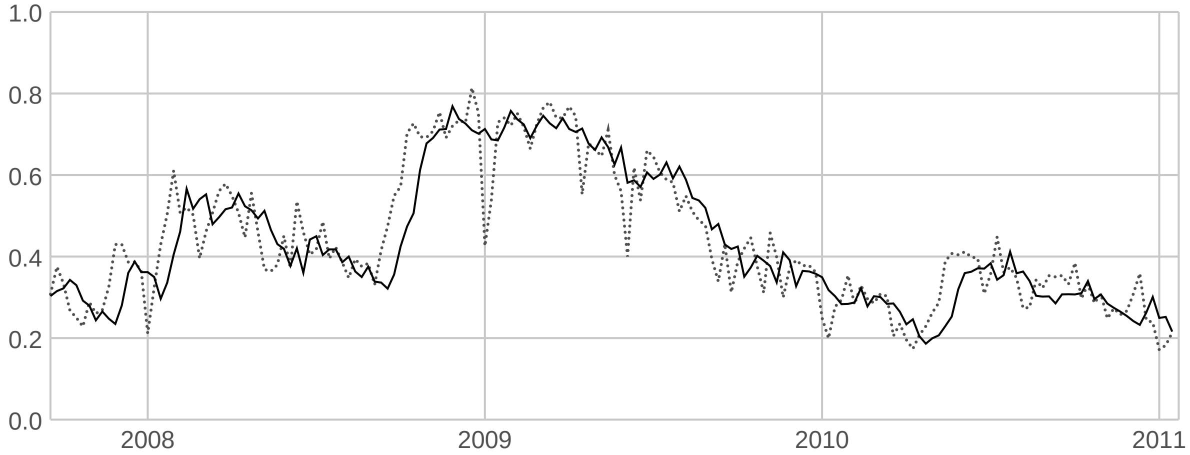


**Figure 2** CISS predictions two weeks into the future, average absolute error = 0.0585.

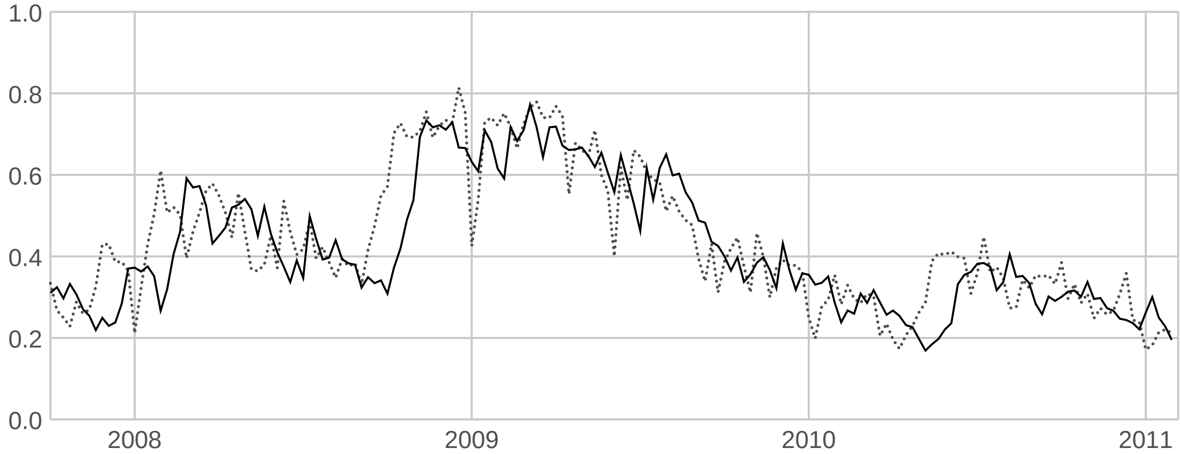


**Figure 3** CISS predictions four weeks into the future, average absolute error = 0.0748.

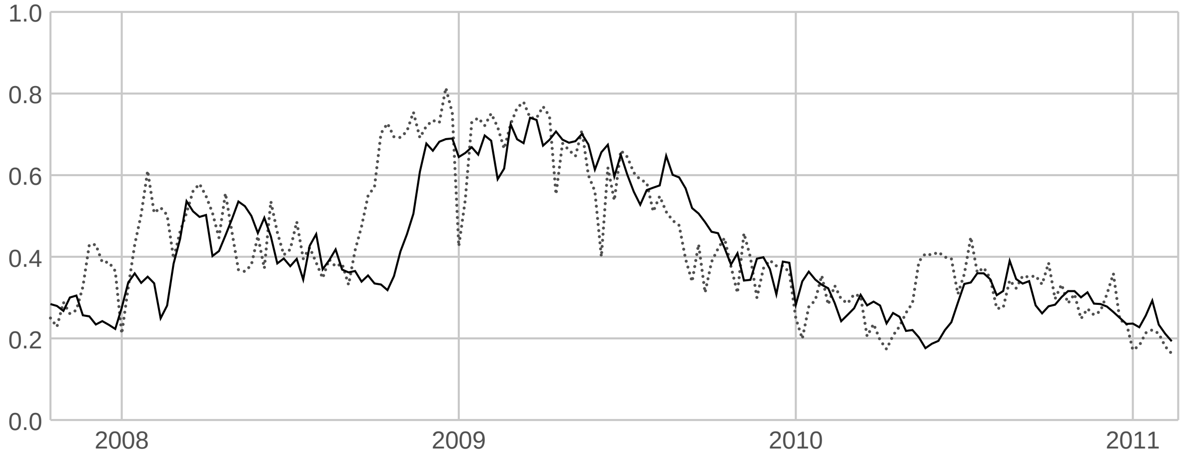


**Figure 4** CISS predictions six weeks into the future, average absolute error = 0.0836.

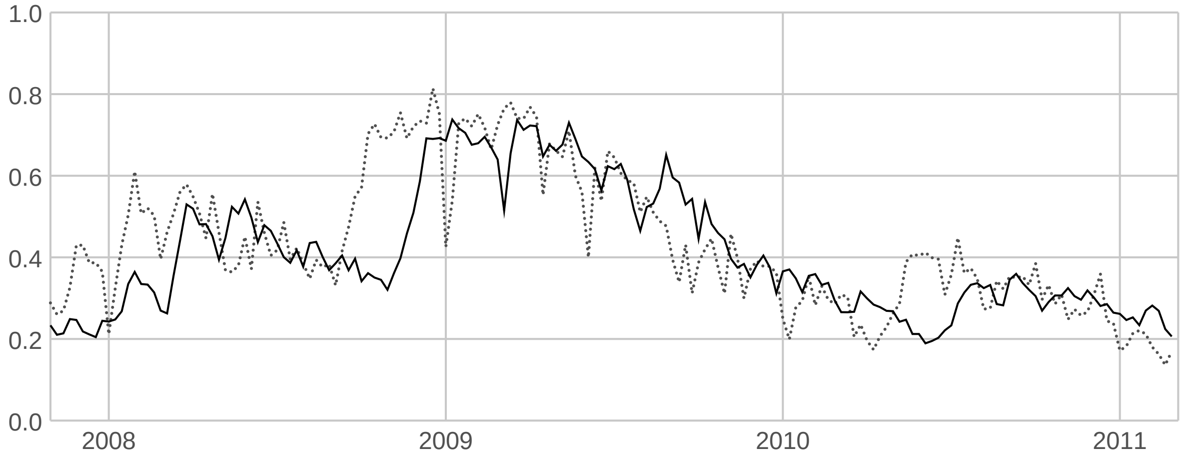


**Figure 5** CISS predictions eight weeks into the future, average absolute error = 0.0934.

All components of the context were finally normalized to zero mean and unit variance. Furthermore, it proved beneficial to create contexts only from a subset $\mathcal{S}$ of the full layout. Thus the final parameter set to optimize was $\{\varepsilon, C, \sigma, H, \mathcal{S}\}$.

SVM training and classification was performed with the use of the SVM$^{light}$ library [5].

Figures 1–5 depict the predictions of the best models for each time skip $\delta$. It can be seen that the CISS index can be partially predicted just from its past values. Large behaviour shifts are, however, surprising for the learned model. Incorporating external variables into the model might improve the predictions: this will be a subject of further study.

The most surprising result is that the history length cap $H$ fluctuated between only two and three months. This suggests that data beyond three months ago only confuses the SVM algorithm and that the increased model complexity isn't justified.

## 4 Conclusion

We've shown the predictability of the CISS index from only its past values for up to eight weeks ahead by the means of a non-linear SVM model. We've optimized the SVM learning parameters and the context layout to minimize prediction error.

In future research we'll try to further improve the prediction accuracy by including additional micro- and macroeconomic time series into the context layout. Based on context lengths of the best obtained models, we conjecture that adding samples beyond three months in the past won't bring any improvements.


## Acknowledgements

The author would like to express her thanks to her supervisor Prof. Ing. Miloslav Vošvrda, CSc. and to her colleagues in the ÚTIA AV ČR postgraduate workshop.


## References


[1] Balakrishnan, R. et al.: The Transmission of Financial Stress from Advanced to Emerging Economies, *IMF working papers* **09/133** (2009), 1–52.

[2] Burges, C. J. C.: A tutorial on support vector machines for pattern recognition, *Data mining and knowledge discovery* **2** (1998), 121–167.

[3] Cardarelli, R. et al.: Financial stress and economic contractions, *J. Financial Stability* **7** (2011), 78–97.

[4] Holló, D. et al.: CISS – A Composite Indicator of Systemic Stress in the Financial System (2011).

[5] Joachims, T.: Making Large-Scale SVM Learning Practical. In: *Advances in Kernel Methods – Support Vector Learning* (Schölkopf, B., Burges, C., and Smola, A. J., ed.), MIT Press, 1999, 41–56.

[6] Kowalczyk, A.: Maximal margin perceptron. In: *Advances in Large Margin Classifiers* (Smola, A. J., Bartlett, P. L., Schölkopf, B., and Schuurmans, D., eds.), MIT Press, Cambridge, MA, 2000, 75–113.

[7] Laeven, L., and Valencia, F.: Resolution of banking crises: the good, the bad, and the ugly, *IMF working paper* **10/146** (2010).

[8] Mukherjee, S. et al.: Nonlinear prediction of chaotic time series using support vector machines, *Proc. of IEEE NNSP'97*, 1997, 511–520.

[9] Smola, A. J., and Schölkopf, B.: A tutorial on support vector regression, *Statistics and Computing* **14** (2004), 199—222.

[10] Vapnik, V. N.: *The Nature of Statistical Learning Theory*. Springer, 1995.