Additional information paper

Forecasting and Analyzing Insurance Companies' Ratings

by Tony Van Gestel, David Martens, Bart Baesens, Daniel Feremans, Johan Huysmans, Jan Vanthienen

to be published in International Journal of Forecasting

1 Difference in Model Fit

The quality of the model fit is assessed by the negative log likelihood. In information criteria, one often uses the model deviance, which is twice the negative log likelihood, as a main indicator. In the case of ordinary least squares regression, the deviance is closely related to the sum squared error. The deviance[1]) between the full model $\mathcal{M}_1$ (with inputs $1, \ldots, i-1, i, i+1, \ldots, m$) and the reduced model $\mathcal{M}_0$ without the corresponding input (inputs $1, \ldots, i-1, i+1, \ldots, m$) are compared. The Bayes factor $\mathcal{B}_{10}$ is approximated via

$$2\log(\mathcal{B}_{10}) \approx \mathrm{dev}(\mathcal{M}_0) - \mathrm{dev}(\mathcal{M}_1) = \Delta\mathrm{dev} \tag{1}$$

and indicates the model improvement. This has to be sufficiently large as indicated by Table 1 (Jeffreys (1961)).

Table 1: Evidence against the H_0 hypothesis of no improvement of model $\mathcal{M}_1$ over model $\mathcal{M}_0$ for different values of the Bayes factor B_{10} (Jeffreys (1961)).

2 $log(B_{10})$	B_{10}	Evidence against H_0
0 to 2	1 to 3	Not worth more than a bare mention
2 to 5	3 to 12	Positive
5 to 10	12 to 150	Strong
> 10	> 150	Decisive

[1] It is preferred to report the deviance as it is straightforward to compute the appropriate complexity criteria from the deviance.

2 Support Vector Machine

The Support Vector Machine (SVM) is a state-of-the art data mining technique (Suykens et al. (2002); Van Gestel et al. (2004); Vapnik (1998)) that is able to capture non-linearities, resulting in complex mathematical models. This advantage is also its main weakness: the model may provide a high accuracy compared to other data mining techniques but the comprehensibility of this 'black-box' model is much worse. Note that several techniques have been proposed to extract comprehensible rules from SVM models (Martens et al. (2005)), but still involve a degradation in accuracy.

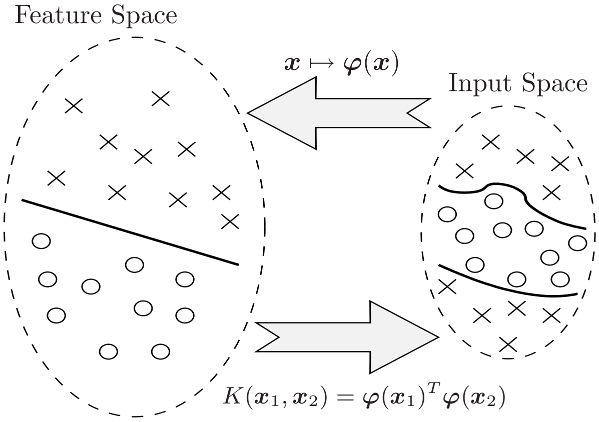
Feature Space
$\boldsymbol{x} \mapsto \boldsymbol{\varphi}(\boldsymbol{x})$
Input Space
$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \boldsymbol{\varphi}(\boldsymbol{x}_1)^T \boldsymbol{\varphi}(\boldsymbol{x}_2)$

Figure 1: Illustration of SVM based classification. The inputs are first mapped in a non-linear way to a high-dimensional feature space ($\boldsymbol{x} \mapsto \boldsymbol{\varphi}(\boldsymbol{x})$), in which a linear separating hyperplane is constructed. By applying the Mercer theorem ($K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x}_j)$), a non-linear classifier in the input space is obtained.

SVM classifier

The SVM classifier is of the form

$$y(\mathbf{x}) = \operatorname{sign}[\mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}) + b], \tag{2}$$

with weight vector $\mathbf{w}$ and bias term b derived from the data. The latent variable $z = \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}) + b$ provides a scoring function that ranks the data instances, e.g. the insurance companies, from high to low score/default risk. The scoring function is obtained by the non-linear mapping $\boldsymbol{\varphi}(\mathbf{x})$ of the input space to a high (possibly infinite) dimensional feature space, in which a linear separating hyperplane is constructed (see Fig. 2). A key element of SVMs is that the $\boldsymbol{\varphi}$ mapping is implicit, defined in terms of the positive definite kernel function $K(\mathbf{x}_i, \mathbf{x}) = \boldsymbol{\varphi}(\mathbf{x}_i)^T \boldsymbol{\varphi}(\mathbf{x})$ (Mercer's theorem). Typical kernel functions are:

- $K(\boldsymbol{x}_i, \boldsymbol{x}) = \boldsymbol{x}_i^T \boldsymbol{x}$ (linear kernel)
- $K(\boldsymbol{x}_i, \boldsymbol{x}) = (\boldsymbol{x}_i^T \boldsymbol{x} + \eta)^d$ (polynomial kernel of degree d with η a positive real constant)
- $K(\boldsymbol{x}_i, \boldsymbol{x}) = \exp(-||\boldsymbol{x} - \boldsymbol{x}_i||_2^2/\sigma^2)$ (Radial Basis Function (RBF) kernel with bandwidth parameter σ)

The resulting classifier is given by

$$y(\boldsymbol{x}) = \text{sign}[\textstyle\sum_{i=1}^N \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b], \tag{3}$$

with latent variable $z = \sum_{i=1}^N \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b$.

The support vector machine (SVM) classifier, according to Vapnik's original formulation satisfies the following conditions, Vapnik (1998):

$$\begin{cases} \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_i) + b \geq +1, & \text{if } \ y_i = +1 \\ \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_i) + b \leq -1, & \text{if } \ y_i = -1 \end{cases} \tag{4}$$

which is equivalent to

$$y_i[\mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_i) + b] \geq 1, \qquad i = 1, ..., N. \tag{5}$$

One defines the convex optimization problem:

$$\min_{\mathbf{w}, b, \boldsymbol{\xi}} \mathcal{J}(\mathbf{w}, b, \boldsymbol{\xi}) = \frac{1}{2}\mathbf{w}^T \mathbf{w} + C \sum_{i=1}^N \xi_i \tag{6}$$

subject to

$$\begin{cases} y_i[\mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_i) + b] \geq 1 - \xi_i, & i = 1, ..., N \\ \xi_i \geq 0, & i = 1, ..., N. \end{cases} \tag{7}$$

The variables ξ_i are slack variables which are needed in order to allow misclassifications in the set of inequalities (e.g. due to overlapping distributions). The first part of the objective function tries to maximize the margin between both classes in the feature space, whereas the second part minimizes the misclassification error. The positive real constant C should be considered as a tuning parameter in the algorithm, together with the choice of the kernel function and its parameters.

The Lagrangian to the constraint optimization problem (6) and (7) is given by

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}; \boldsymbol{\alpha}, \boldsymbol{\nu}) = \mathcal{J}(\mathbf{w}, b, \boldsymbol{\xi}) - \sum_{i=1}^N \alpha_i \{y_i[\mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_i) + b] - 1 + \xi_i\} - \sum_{i=1}^N \nu_i \xi_i. \tag{8}$$

The solution to the above optimization problem is given by the saddle point of the Lagrangian, i.e. by minimizing $\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}; \boldsymbol{\alpha}, \boldsymbol{\nu})$ with respect to $\mathbf{w}$, b, $\boldsymbol{\xi}$ and maximizing it with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\nu}$:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\nu}} \min_{\mathbf{w},b,\boldsymbol{\xi}} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}; \boldsymbol{\alpha}, \boldsymbol{\nu}). \tag{9}$$

One obtains

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0 \quad \rightarrow \quad \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{\varphi}(\mathbf{x}_i) \\ \\ \frac{\partial \mathcal{L}}{\partial b} = 0 \quad \rightarrow \quad \sum_{i=1}^{N} \alpha_i y_i = 0 \\ \\ \frac{\partial \mathcal{L}}{\partial \xi_i} = 0 \quad \rightarrow \quad 0 \leq \alpha_i \leq C \quad , \quad i = 1, ..., N. \end{cases} \tag{10}$$

By substituting the first expression into (2), the resulting classifier becomes (3).

The Lagrange multipliers α_i are then determined by means of the following optimization problem (dual problem):

$$\max_{\alpha_i} -\frac{1}{2} \sum_{i,j=1}^{N} y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \alpha_i \alpha_j + \sum_{i=1}^{N} \alpha_i \tag{11}$$

subject to

$$\begin{cases} \sum_{i=1}^{N} \alpha_i y_i = 0 \\ 0 \leq \alpha_i \leq C, \quad i = 1, ..., N. \end{cases} \tag{12}$$

The entire classifier construction problem now simplifies to a convex quadratic programming (QP) problem in α_i. Typically, many of the α_i will be equal to zero (sparseness property). The training observations corresponding to non-zero α_i are called support vectors and are located close to the decision boundary.

Nyström sampling

A typical disadvantage of the SVM formulations is that the computational and memory requirements grow as a power of N, the number of training data points. To counter this issue, we will use a method called Nyström sampling, which essentially makes estimations on a subsample of size $M < N$ and is described next.

Given the data points $x_1, \ldots, x_N$ and the kernel function K, one can estimate the non-linear mapping $\boldsymbol{\varphi}(x)$ based on the eigenvalue decomposition of the kernel matrix Ω:

$$\Omega = \boldsymbol{U} \boldsymbol{\Upsilon} \boldsymbol{U}^T \tag{13}$$

with $\boldsymbol{U} = [u_1, u_2, \ldots, u_N] \in \mathcal{R}^{N \times N}$ and $\boldsymbol{\Upsilon} = diag([v_1, v_2 \ldots, v_N]) \in \mathcal{R}^{N \times N}$. The elements $\varphi_i(x)$ of the mapping $\boldsymbol{\varphi} = [\varphi_1, \varphi_2, \ldots, \varphi_{n_f}]^T$ are estimated as follows (Suykens et al. (2002)).

$$\varphi_i(x) = \frac{\sqrt{N}}{\sqrt{v_i}} \sum_{k=1}^{N} v_{ki} K(x_k, x), \quad i = 1, 2, \ldots, N \tag{14}$$

and $\varphi(x) = 0$ for $v_i = 0$ or $i \geq N+1$. Using this estimate, it is easy to see that $\varphi(x_i)^T\varphi(x_j) = K(x_i, x_j)$ for $i, j = 1, 2, \ldots, N$. For large data sets, the computational requirements may become too high. The idea of Nyström sampling is to estimate $\boldsymbol{\varphi}$ on a (carefully) selected sub-sample of size $M \leq N$ from the data $\{x_i\}_{i=1}^N$. The computational complexity for Equation 14 reduces from $O(N^3)$ to $O(M^3)$, while the memory requirements drop from $O(N^2)$ to $O(M^2)$. Here, M is set at 50, as it is shown that datasets with $N = 20000$ can already be approximated well with a subset of size 20 (Suykens et al. (2002)). Out of 20 random subsamples of size 50, the one for which the histogram best reflects the full training sample is selected.

Addition of SVM terms

An RBF kernel is chosen as it is found to provide good generalization behavior (Van Gestel et al. (2004)). The tuning of the SVM part involves selection of the set of relevant SVM inputs $\boldsymbol{x} = [x_{\text{SVM},1}, \ldots, x_{\text{SVM},n}]^T$, the selection of the kernel parameter σ and selection of the relevant components $\varphi_i(\boldsymbol{x})$ of the feature vector $\boldsymbol{\varphi}(\boldsymbol{x})$.

The design is done in a hierarchical way: for a selection of candidate input parameters, the optimal kernel parameter is selected and then the significant components $\varphi_i(\boldsymbol{x})$ are selected. The input selection is done in a backward way, starting from a large set of candidate inputs and removing in turn that variable that, when removed, yields the best performance improvement.

For a given candidate set of input variables $\boldsymbol{x}$, the kernel parameter σ is selected from a grid $\Sigma = \sqrt{n} \times [0.8 \quad 1 \quad 1.2 \quad 1.5 \quad 2 \quad 5]$ using cross-validation (Van Gestel et al. (2004)). For each candidate σ-value, elements of the feature vector $\boldsymbol{\varphi}(\boldsymbol{x})$ are calculated from (14). Next, backward input selection is performed on the candidate set consisting of z_{IL} and the first 20 principal components[2]:

$$[z_{\text{IL}} \quad \phi_1(x) \quad \phi_2(x) \quad \phi_3(x) \quad \ldots \quad \phi_{20}(x)] \tag{15}$$

The linear term z_{IL} is included, so as to check its validity in the non-linear model. As expected, the linear term is found to be very significant with an estimated coefficient close to 1.

[2] Principal components with lower eigenvalues typically contain less information and it would involve more computations. The number of principal components is selected using the 90% scree graph criterion, with a maximum of 20.

3 Encoding of Rating Variable

Table 2: Encoding of the target variable.

Moody's Rating	S&P/Fitch Rating	Target Value
Aaa	AAA	1
Aa1	AA+	2
Aa2	AA	3
Aa3	AA-	4
A1	A+	5
A2	A	6
A3	A-	7
Baa1	BBB+	8
Baa2	BBB	9
Baa3	BBB-	10
Ba1	BB+	11
Ba2	BB	12
Ba3	BB-	13
B1	B+	14
B2	B	15
B3	B-	16
CCC	CCC	17

4 Considered Financial Variables

Table 3: Considered financial variables.

Type	Variable	Type	Variable
Capital Adequacy	Capital Adequacy	Debt & Leverage	Debt/Capital (%)
	Solvency ratio		Debt/Equity (All) (%)
	Free reserve ratio		Debt/Gross premium written (%)
	(Surplus + Net technical reserves)/Gross premium written (%)		Debt/Gross premium written (%)
	(Surplus + Net technical reserves)/Net premium written (%)		Debt/Gross premium written (%)
	Net technical reserves/Gross premium written (%)		External borrowings/Surplus (%)
	Net technical reserves/Net premium written (%)		Total liabilities/Surplus (%)
	Net technical reserves/Surplus (%)	Cash Flow	Total cash flow ratio (%)
	Safety margin		U/W cash flow ratio (%)
	Net premium written/(Surplus + Net technical reserves) (%)	Profitability	ROA (AT) (%)
	Net premium written/Net technical reserves (%)		ROA (BT) (%)
	Net unpaid losses/Net technical reserves (%)		ROAA (AT) (%)
	Net unpaid losses/Qualified statutory capital (%)		ROAA (BT) (%)
	Net unpaid losses/Surplus (%)		ROAE (AT) (%)
	U/W expenses/Net technical reserves (%)		ROE (AT) (%)
	Gross premium written/Surplus		ROE (BT) (%)
	Net premium written/Surplus		Profit after tax/Hard capital Avg (%)
	Net premium written/Net technical reserves		Profit after tax/Net premium written (%)
	Net premium written/(Surplus+Net technical reserves)		Profit before tax/Gross premium written (%)
	Expenses/Net technical reserves		Profit before tax/Net premium earned (%)
Performances	Combined ratio		Profit after tax/Gross premium written (%)
	Expense ratio		Profit before tax/Net premium written (%)
	Loss ratio		Profit margin (%)
	Net claims/Gross claims		Return on capital (Qualified statutory capital Avg) (%)
	Net investment income/Net premium written	Liquidity	Liquid A/Illiquid A (%)
	Net investment income/Profit before tax (%)		Liquid A/Total A (%)
	Net premium earned/Gross premium written (%)		Cash & Deposits/Total A (%)
	Net premium earned/Net premium written (%)	Size	Gross Premium Written
	Net unpaid losses/Net claims (%)		Net Premium Written
	U/W result/Profit before tax (%)		Net Technical Reserves
	Underwriting profitability ratio (S&P's definition) (%)	Other	percentage reinsurance
	Underwriting result/Profit before tax (%)		Retained profit for the year/Profit after tax (%)
	Investment return (%)		Retained profit/Profit after tax
	Investment yield (%)		listed

References

Jeffreys, H., 1961. Theory of Probability. Oxford University Press.

Martens, D., Baesens, B., Gestel, T. V., Vanthienen, J., 2005. Comprehensible credit scoring models using rule extraction from support vector machines. Research Report 0581, Dept. of Decision Sciences and Information Management, Katholieke Universiteit Leuven.

Suykens, J., Van Gestel, T., De Brabanter, J., De Moor, B., Vandewalle, J., 2002. Least Squares Support Vector Machines. World Scientific, Singapore.

Van Gestel, T., Suykens, J., Baesens, B., Viaene, S., Vanthienen, J., Dedene, G., De Moor, B., Vandewalle, J., 2004. Benchmarking least squares support vector machine classifiers. Machine Learning 54, 5–32.

Vapnik, V., 1998. Statistical learning theory. John Wiley, New-York.