Computers and Concrete, *Vol. 12, No. 2 (2013) 187-209*
DOI: http://dx.doi.org/10.12989/cac.2013.12.2.187

Prediction of the transfer length of prestressing strands with neural networks

José R. Martí-Vargas[1a], Francesc J. Ferri[2b] and Víctor Yepes[*1]

[1]*Department of Construction Engineering, Institute of Concrete Science and Technology (ICITECH), Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain*
[2]*Department of Computer Science, Universitat de València, Av. Universitats s/n, 46100 Burjassot, Spain*

(Received November 30, 2011, Revised January 21, 2013, Accepted February 17, 2013)

Abstract. This paper presents a study on the prediction of transfer length of 13 mm seven-wire prestressing steel strand in pretensioned prestressed concrete members with rectangular cross-section including several material properties and design and manufacture parameters. To this end, a carefully selected database consisting of 207 different cases coming from 18 different sources spanning a variety of practical transfer length prediction situations was compiled. 16 single input features and 5 combined input features are analyzed. A widely used feedforward neural regression model was considered as a representative of several machine learning methods that have already been used in the engineering field. Classical multiple linear regression was also considered in order to comparatively assess performance and robustness in this context. The results show that the implemented model has good prediction and generalization capacity when it is used on large input data sets of practical interest from the engineering point of view. In particular, a neural model is proposed -using only 4 hidden units and 10 input variables- which significantly reduces in 30% and 60% the errors in transfer length prediction when using standard linear regression or fixed formulas, respectively.

Keywords: transfer length; prestressing strand; prestressed concrete; neural networks; machine learning

1. Introduction

According to the ACI Code 318-11 (2011), the transfer length (TL) is defined as the distance over which the strand should be bonded to the concrete to develop the effective prestress in the prestressing strand (see Fig. 1). The TL is an important parameter for pretensioned, prestressed concrete (PC) structural design (Russell and Burns 1996, Barnes *et al*. 2003). The accuracy of any attempt to predict the specific location of the critical sections for checking stresses in the end region of pretensioned, PC members depends upon the accuracy of the estimation of the TL (Barnes *et al*. 2003). There are many expressions from authors and Codes available in the literature, which can be used to predict TL in the design exercise (Martí-Vargas *et al*. 2012b).

∗Corresponding author, Associate Professor, E-mail: vyepesp@cst.upv.es
[a]Associate Professor, E-mail: jrmarti@cst.upv.es
[b]Professor, E-mail: francesc.ferri@uv.es

http://www.techno-press.org/?journal=cac&subpage=8 ISSN: 1598-8198(Print), 1598-818X (Online)

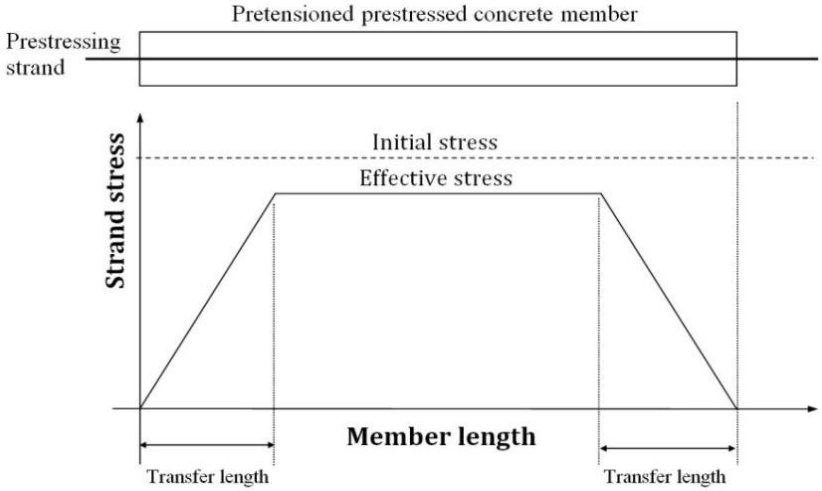

Fig. 1 Strand stress variation along a pretensioned concrete member after release

Several expressions are derived from theoretical studies based on the relationship between the strand end slip and the TL (Guyon 1953, Balázs 1993, Martí-Vargas *et al*. 2007b), and others are based on the equilibrium of the transfer bond force over the TL of prestressing strands and the developed force in the prestressing strand assuming the hypothesis of uniform bond stress (ACI 2011, Martí-Vargas *et al*. 2007a).

Bond strength as well as TL depend on the properties of both the prestressing strand and the concrete and also on several design and manufacture parameters (CEB 1987, FIB 2000). These properties and parameters include: concrete strength at the time of detensioning, level of prestress at detensioning, concrete cover, strand spacing, size of the cross-section, strand size and type, strand surface condition, detensioning method (sudden, gradual), etc. However, there is no consensus on the main parameters to be considered in the equations to predict the TL (Martí-Vargas *et al*. 2012a). An example of such is the ACI Code (2011) provisions for TL that are not a function of concrete strength. On the other hand, the Eurocode 2 (CEN 2004) and the CEB-FIP Model Code (FIB 2010) provisions for TL consider the concrete properties.

The TL may be determined experimentally (Martí-Vargas *et al*. 2006a, 2012c), and several experimental research works on bond of prestressing strands have been conducted over the years. Three experimental methods to determine TL are frequently used: the longitudinal concrete surface strain profile (Thorsen 1956), the prestressing strand end slip (Guyon 1953), and the prestressing strand force at several cross-sections (ECADA test method (Martí-Vargas *et al*. 2006a). There are several inherent difficulties to obtain reliable measures for TL: Slope-Intercept method (Deatherage *et al*. 1994) and the 95% Average Maximum Strain method (Russell and Burns 1996) are attempts for best-fit methods from concrete strain profiles; there is a larger scatter of experimental results from strand end slip (Martí-Vargas *et al*. 2007b), and excessive free end slip is registered in prestressed members with poor concrete consolidation around the strand, (Balogh 1992) and the application of the ECADA test method requires an adequate simulation of the sectional rigidity of the specimens (Caro *et al*. 2013).

The effect on TL of the aforementioned properties and parameters are usually described by using regression analysis and statistical models (Martí-Vargas *et al*. 2006b, Kose 2007, García-

Taengua *et al*. 2011). Based on Martí-Vargas *et al*. (2007a) and Caro *et al*. (2012), Eq. (1) presents a parametric form for most of the proposed equations from authors and Codes in the literature

$$L_t = \left[\frac{f_{px}^{\,n} A_p}{(k_1 \pi d_b) U_t} + k_2 \right] \beta \tag{1}$$

where
L_t = transfer length
f_{px} = stress in prestressing strand (possible values: initial effective stress f_{si}; long-term effective stress f_{se}; stress in prestressing strand at the time of tensioning f_{pt}; stress in prestressing strand just before detensioning f_{p0}; ultimate tensile strength of prestressing strand f_{pu})
n = exponent
A_p = cross-sectional area of prestressing strand
k_1 = perimeter factor (k_1=1 for circular cross-section; k_1= 4/3 for seven-wire strand)
d_b = nominal diameter of prestressing strand
U_t = average bond stress along the transfer length
k_2 = adjustment constant
β = factor account for release method

As observed in Eq. (1), a few properties and parameters are included. Also, agreeing with Kose and Kayadelen (2010), generally these equations give accurate results only for the experimental data used in the corresponding development and cannot be generalized for results and data from other studies. The prediction of TL can also be tackled as a learning problem in the context of statistical machine learning and intelligent data mining (Bishop 1995). In this case, none or very little a priori knowledge about the physical problem being studied is explicitly considered and only particular cases or examples including all input and output measurements are used.

Several data mining methods, including knowledge-based systems, heuristic algorithms, fuzzy logic and neural networks, are available to optimizing the prediction of a certain output parameter. Though many of these methods on concrete structures design and prediction have been developed in recent years (Yeh 2008, Parichatprecha and Nimityongskul 2009, Bilgehan and Turgut 2010, Huang *et al*. 2010, Martínez *et al*. 2010, Yepes *et al*. 2012), the earliest studies date back to the 1990s: as an example, Adeli and Yeh (1989) present the first journal article on neural network application in structural engineering. A review of neural networks in civil engineering can be found in Adeli (2001), and a complete example of application of these methods on concrete properties is presented in Chou (2011). However, a few studies have proposed approaches for modelling the steel-concrete bond phenomenon: artificial neural networks (ANNs) (Sancak 2009, Dahou *et al*. 2009) and fuzzy logic (Tanyildizi 2009) are used in reinforced concrete, and only the works presented in Kose (2007) and Kose and Kayadelen (2010) investigate and demonstrate the usability and the efficiency of several techniques (artificial neural network, neuro-fuzzy inference system, and genetic expression programming) in PC.

Therefore, as the application of data-mining methods is relatively scarce in PC, the main purpose of this work is to analyze the accuracy and robustness in ANN based prediction of TL of 13 mm seven-wire prestressing steel strand in pretensioned, PC members with rectangular cross-section including several material properties and design and manufacture parameters. To this end, a carefully selected database spanning a variety of practical TL prediction situations was compiled.

The widely used ANN model has been considered as a representative of Machine Learning methodologies along with several analytical models and the classical multiple linear regression in order to assess performance and robustness in this engineering context.

2. Selection of data collection

The experimental data set has been collected from an extensive literature review on experimental research works, which report measured values of TL. Several requirements were established on materials (concrete and prestressing strand), prestressed members features, and design and manufacture parameters. These requirements were focused on pretensioned, PC members with rectangular cross-section using 13 mm seven-wire prestressing steel strand.

Data sets described in the literature are often incomplete and contain unexpected inaccuracies. Consequently, making use of all available information from data constitutes a challenging problem in itself. For example, usually the concrete compressive strength at prestress transfer is reported. However, parameters as the age of concrete at prestress transfer, or the conservation conditions, are seldom reported. The experimental database used in Kose (2007) to predict the TL of prestressing strands in PC beams with different shaped cross-sections using neural networks includes 84 cases from 9 different sources, and 7 input variables (strand diameter, geometric ratio, strand surface, % debonded, number of plateau in strain profile, effective prestress, concrete strength) are considered. In the present work, an important effort was made to have a homogeneous data collection with a significantly larger number of input variables (properties and parameters). A total of 16 single or primary input variables and 5 combined (including some relations between variables and a normalized variable to analyze their effects), higher order input variables were set up for the present study (Table 1). The explicit use of these combined input variables that are similar to the ones in the standard formulae is aimed at discovering similarities and differences between the way classical and adaptive regression models give importance to inputs (Section 5.3).

Only sources reporting all the aforementioned input variables are used. Also, only data concerning the focused requirements within a source are included. As a result, a total of 18 different sources spanning a variety of practical TL prediction situations were selected, and a final set of 207 samples of TL determination was evaluated. Table 2 summarizes this data collection. The top value in a box corresponds to the maximum reported value and the bottom value in a box corresponds to the minimum reported value.

3. TL prediction methods

3.1 Classical prediction methods

Given a convenient set of input variables and/or parameters, one can obtain predictions of TL or any other output variable basically in two opposite and sometimes complementary ways:

a) By exploiting prior knowledge about the problem. It is mainly used in the domain specific literature that led to many prediction formulas currently used as standards. A convenient account on these is given in Martí-Vargas *et al.* (2007a).

b) By using a method to learn a model from training data consisting of both input and output

Table 1 Input features included in the experimental data collection

Input nº	Designation	Description	Remarks
1	d_b	nominal diameter of prestressing strand	in mm; selected d_b = 12.4 to 12.9 mm
2	f_{pu}	nominal tensile strength of prestressing strand	in MPa; no limitation
3	A_p	cross-sectional area of prestressing strand	in mm^2; no limitation
4	*ss*	strand surface	uncoated; classified in bright or rusty
5	*ns*	number of prestressing strands in the cross-section of the member	no limitation
6	*b*	width cross-section of the member	in mm; no limitation
7	*h*	depth cross-section of the member	in mm; no limitation
8	*cv*	vertical concrete cover	in mm; no limitation
9	*ch*	horizontal concrete cover	in mm; no limitation
10	*sep*	strand spacing	in mm; no limitation; only if *ns* >1; *sep* =0 if *ns* =1
11	*transv*	transverse reinforcement	no limitation
12	f_{ci}	concrete compressive strength at prestress transfer	in MPa; no limitation; always normal weight concrete
13	*dest*	release method	classified in sudden (S) or gradual (G)
14	*extr*	type of member end region	classified in active-live (A) or passive-dead (P)
15	f_{p0}	stress in prestressing strand just before release	in MPa; no limitation
16	A_p / A_c	geometric ratio (A_c = concrete cross section)	combined input
17	f_{si}	initial effective stress of prestressing strand	in MPa; reported value or $0.9f_{p0}$
18	$f_{si} \cdot d_b$	numerator in ACI 318 expression for transfer length	combined input
19	$sqrt(f_{ci})$	square root of f_{ci} (related with concrete modulus)	combined input
20	$f_{si}/sqrt(f_{ci})$	relation present in expressions from several authors	combined input
21	*modsep*	normalized strand spacing	combined input; *modsep* =2·max(*sep*) if *ns* =1 otherwise *modsep* = *sep*

3.2 Prediction using Artificial Neural Networks

One of the most widely accepted artificial neuron model, whose origin can be traced back to MacCulloch and Pitts (1950), consists of a summation element followed by a nonlinear, sigmoid-like activation function *f*, as shown in Fig. 2. Given an input vector $\boldsymbol{x} = (x_1,\ldots,x_d)$ the neuron computes $f(\boldsymbol{w} \cdot \boldsymbol{x} + w_0)$, where the weight vector $\boldsymbol{w} = (w_1,\ldots,w_d)$ and bias w_0 are the parameters of the neuron unit.

Table 2 Experimental data collection summary

Authors		Kaar *et al.* (1963)	Hanson (1969)	Lindgren and Hoonmeg (1970)	Dorsten *et al.* (1984)	Cousins *et al.* (1990)	Nanni *et al.* (1992)	Lane (1992)	Mitchell *et al.* (1993)	Issa *et al.* (1993)	Uijl (1995)	Russell and Burns (1996)	Logan (1997)	Burns and Russell (1997)	Rose and Russell (1997)	Mahmoud *et al.* (1999)	Oh and Kim (2000)	Kovács and Balazs (2004)	Martí-Vargas *et al.* (2006b)
Source #		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Size (# of tests)		10	2	4	8	20	5	2	14	2	2	26	12	12	30	8	36	2	12
d_b		12.7	12.7	12.7	12.7	12.7	12.4	12.7	12.7	12.4	12.5	12.7	12.7	12.7	12.7	12.7	12.7	12.7	12.9
f_{pu}		1750	1725	1860	1860	1860	1720	1860	1860	1860	1860	1860	1860	1860	1860	1860	1860	1770	1860
A_p	max	92.88	96.68	91.50	98.68	98.68	93.00	98.68	98.68	108.00	91.2	98.68	98.71	98.68	98.68	99.00	98.68	100	99.69
	min			89.50															
ss	max	bright	bright	bright	bright	bright	bright	bright	rusty	bright	bright	bright	bright	bright	bright	bright	bright	bright	bright
	min			rusty									rusty		rusty				
ns	max	6	4	2	1	1	1	4	1	1	1	5	1	1	2	1	2	1	1
	min	2						1				1					1		
b	max	212.7	304.8	200.0	88.9	127.0	120.0	229.0	150.0	102.0	87.5	127.0	165.1	102.0	152.4	100.0	176.2	80.0	100.0
	min					101.6		102.0	100.0			101.6				95.0	112.7		
h	max	152.4	203.2	400.0	114.3	203.2	210.0	229.0	225.0	162.0	100.0	330.2	304.8	127.0	304.8	250.0	200.0	120.0	100.0
	min					101.6		102.0	175.0			127.0				95.0			
cv	max	69.9	67.3	33.7	50.8	63.5	63.8	44.7	43.7	69.8	37.5	57.2	44.5	57.2	44.5	50.8	50.0	33.7	43.6
	min					44.5										41.2	30.0		
ch	max	38.1	60.0	43.7	38.1	57.2	53.8	44.7	68.7	44.8	37.5	57.2	76.2	44.7	40.0	43.7	50.0	33.7	43.6
	min					44.5			43.7			44.5				41.2			
sep	max	85.7	98.0	87.3	--	--	--	114.3	--	--	--	38.1	--	--	47.0	--	50.8	--	--
	min	36.5															25.4		
transv		no	no	yes	no	no	no / yes	no	no	no	no	no / yes	no	no	yes	no / yes	no	no	no
f_{ci}	max	34.9	34.2	26.8	27.6	46.3	34.6	29.8	50.0	40.0	55.2	32.0	29.3	28.9	32.3	46.0	46.7	37.6	54.8
	min	10.7		24.0		28.0	29.1		21.0			26.6		19.2	27.5	35.0	32.5		24.2
dest		S	S	S	S	S	G	S	G	S	G	S	S	S	S	G	S	G	G
extr		A-P	A-P	A-P	A-P	A-P	A	A	A-P	A-P	A-P	A-P	A-P	A-P	A-P	A	A-P	A-P	A
f_{p0}	max	1256.9	1090.8	1325.2	1385.0	1469.4	1340.8	1395.0	1442.0	754.0	1575.0	1365.2	1302.0	1406.0	1395.0	1302.0	1395.0	1360.0	1402.0
	min	1179.7		1233.4			670.4		1298.0			1344.5		1299.0					1357.5
f_{si}	max	1186.6	981.7	1253.8	1287.0	1383.8	1206.7	1255.8	1374.0	678.6	1472.0	1228.7	1276.0	1265.4	1255.5	1227.6	1255.5	1177.0	1337.0
	min	1008.1		1162.1		1229.4	603.4		1217.0			1210.1		1169.1		1116.0			1290.0
L_t	max	1104.9	660.4	673.1	863.6	1879.6	1450.0	1016.0	710.0	482.6	308.0	1117.6	1346.2	1524.0	713.7	600.0	898.0	700.0	650.0
	min	812.8	558.8	317.5	533.4	812.8	550.0	838.2	330.0	444.5	297.0	406.4	355.6	584.0	213.4	350.0	434.0	550.0	400.0

These simple elements can be organized into layers so that the neurons in one layer feed the neurons in the next layer in such a way that the information flows along this network of neurons in a feed-forward way from the inputs to the outputs. It is clear that at least one hidden layer (following the input and preceding the output layers) is needed in order to have an effective nonlinear model. This is basically the well-known and widely used Multi Layer Perceptron (MLP) model. Even with only one hidden layer and with a wide range of activation functions, the MLP is a universal approximator (Hornik *et al*. 1989). Also, there exists a basic back propagation (BP) learning algorithm (Rumelhart *et al*. 1986) with many variants and enhancements that allow an effective use of this ANN model for many practical classification and regression problems starting from some training data for which its desired or target value is known.

The complexity of a given MLP depends on the number of neurons and their organization into different layers i.e., its architecture. Mathematically, the model can be described (for each neuron) as a linear combination of nonlinear functions from the previous layer. That is, if $\boldsymbol{w}^j = (w^j_1,\ldots w^j_d)$ is the weight vector corresponding to the j^{th} neuron in the first hidden layer, w^j_0 is the corresponding bias and d is the number of inputs (neurons in the previous layer), then the output of the j^{th} neuron is given by Eq. (2)

$$y_j = f\,(\boldsymbol{w}^j \cdot \boldsymbol{x} + w^j_0) \tag{2}$$

where $\boldsymbol{x} = (x_1,\ldots,x_d)$ is the input vector.

If there are k neurons in the first hidden layer, a neuron in the next layer has a weight vector $\boldsymbol{w}^o = (w^o_1,\ldots,w^o_k)$ and a bias w^o_0. The output corresponding to this neuron is given by Eq. (3)

$$F\,(\boldsymbol{w}^o \cdot \boldsymbol{y} + w^o_0) \tag{3}$$

where $\boldsymbol{y} = (y_1,\ldots,y_k)$ are the outputs of all neurons in the previous layer. Successive layers can be added to the model in a similar way. These relations are schematized in Fig. 3.

To compute its output, the network needs to compute first the outputs of each layer in turn from the previously computed outputs of the previous one. For a given network with a particular architecture, its behavior depends only of the weights and biases of all neurons. The learning

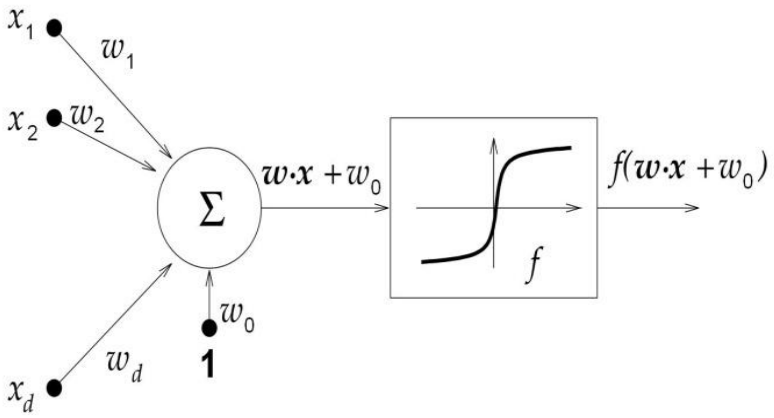

Fig. 2 Artificial neuron model consisting of a summation element followed by an activation sigmoid-like function

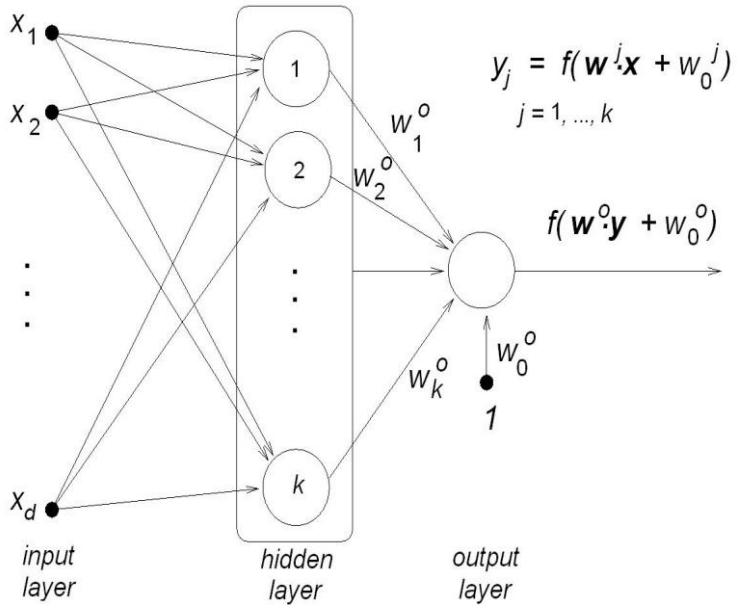

Fig. 3 Description of a Multi-Layer Perceptron with one hidden layer with k units and their corresponding interconnection weights

algorithm consists of a correction rule that can be applied recursively starting from the last (output) layer. The correction applied to each weight in the network depends on the output error (difference between target value and the current output) and modifies the weights of the network iteratively. The optimization criterion in ANN learning is an error measure usually concerning summation of (squared) errors over the whole training data. Available learning algorithms for the

4. Assessing the prediction performance

As in several preceding and related works using regression (Kose 2007, Bilgehan and Turgut 2010b), two classic measures as the coefficient of determination, R^2, and the Root MSE (RMSE) will be considered in the present work to account for the goodness of fit of the different predictors (Bishop 1995, Draper and Smith 1998). To evaluate model bias, the average ratio of predicted to experimental results (RPE) will be also considered in some experiments. If n is the number of available experimental (x_i) and predicted (y_i) values, then the corresponding definitions are

$$R^2 = 1 - \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\left(x_i - \sum_{j=1}^{n} \frac{x_j}{n}\right)} \tag{4}$$

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{n}} \tag{5}$$

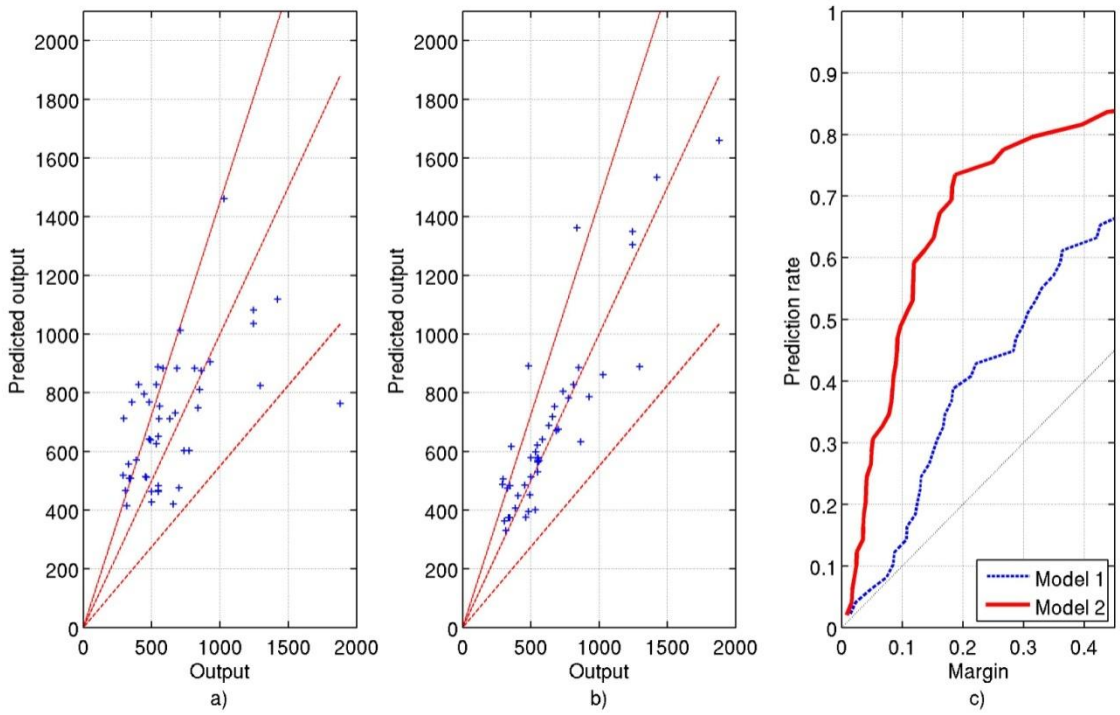

Fig. 4 (a) and (b) Predicted outputs and measured outputs (targets) of two different prediction models. (c) Plot showing correct prediction rate with regard to margin or maximum relative error permitted

$$RPE = \sum_{i=1}^{n} \frac{y_i / x_i}{n} \tag{6}$$

It is widely acknowledged that criteria based on squared residuals as RMSE and R^2, can give rise to misleading results when the initial assumptions about normality or data being independent and identically distributed (i.i.d.) do not hold. In particular, they are very sensitive to outliers both in the explanatory and output variables. In order to circumvent these problems and inspired in the way engineering solutions are adopted; one can measure the goodness of a particular prediction as the proportion of predicted values that are at a certain distance from the corresponding target value. This can be seen as a maximum relative error on the measure being predicted and this value could be considered as a prespecified design parameter. For example, in Figs. 4a and 4b, the results of two different prediction models are displayed. The horizontal axis represents measured target (e.g., TL) and vertical axis corresponds to the values predicted by each model. The three discontinuous lines represent the perfect prediction (the $y = x$ curve) and a 0.45 deviation (that is, $y = \pm 0.45x$). The proportion of predictions within this 0.45 maximum relative error is 0.67 in the first model (Fig. 4a) and 0.86 in the second model (Fig. 4b), while the corresponding R^2 values are 0.33 for both models. This illustrates the fact that squared and averaged error measures do not account for qualitative differences among different regression models.

More interestingly, it is possible to evaluate how good a model is by evaluating the proportion

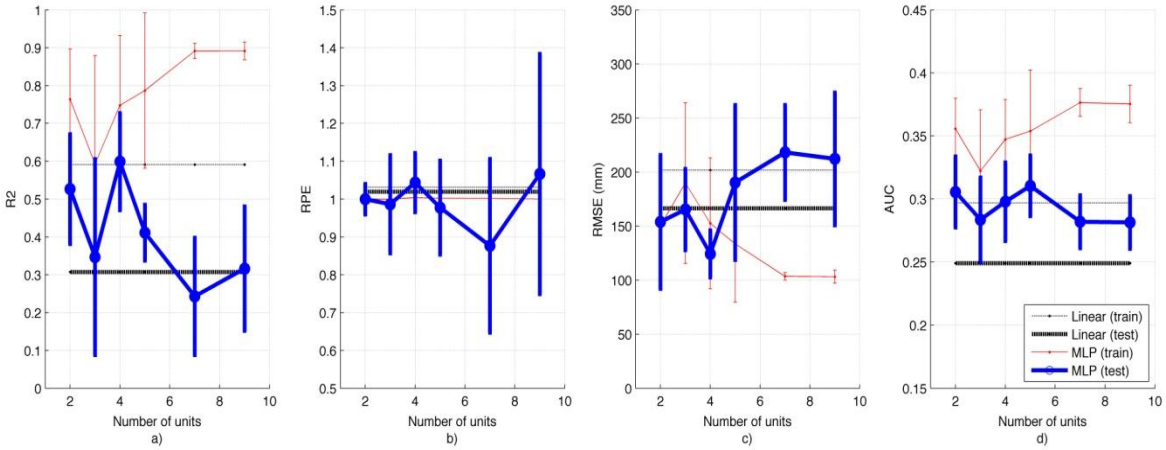

Fig. 5 Averaged prediction results obtained using linear regression and MLP for values of k up to 9. R^2, RPE, RMSE, and AUC values are shown

of correctly predicted cases as the maximum relative error increases within a prespecified range. This can be represented as a Relative Operating Characteristic (ROC) curve (Fawcett 2006, Paclík *et al*. 2010). Following the aforementioned example, in Fig. 4(c), the two curves corresponding to the prediction results by models in the previous Figs. 4(a) and 4(b) are shown.

As observed in Fig. 4(c), these are monotonically non-decreasing curves going (potentially) from 0 to 1 in both coordinates. The more the curve deviates from the $y = x$ behavior, the better the prediction is, exactly as in ROC analysis (Fawcett 2006). Consequently, a convenient global measure of the goodness of prediction is the Area Under the Curve (AUC). This measure is commonly used in many different machine learning applications in which similar curves can be defined. In our particular case, as too large relative errors in the prediction do not make sense, the curve (and its AUC) can be conveniently restricted for a particular range. In the present work the range of relative errors was set at [0, 0.45]. The AUC measures of the two curves in the previous figure are 0.16 and 0.31 for the first model and the second model, respectively. Note that these AUC values are in the range [0, 0.45] and that a uniform random predictor (corresponding to the $y = x$ curve, i.e., the worst predictor in practice) will give rise to an AUC value of $(0.45)^2/2 = 0.1013$ (Fawcett 2006).

5. Experiments and results

5.1 Experimental setup

The only output variable considered in the present work is the TL. Empirical measures of this output variable together with corresponding values for the 16 basic input features and 5 combined ones constitute the database used. Consequently, regression is done in a 21-dimensional space with an important number of redundant variables.

Even though experiments normalizing input and output data to zero mean and unit variance were carried out, all results in this paper involve unnormalized data in order to compare the

prediction results with the corresponding standards that use raw measures. The main difference is that adjusting the parameters is slightly harder but in all cases prediction results both with and without normalization gave statistically indistinguishable results.

All computer code for regression, ANN simulation and data processing was written and run in the Matlab framework (Mathworks 2011).

5.2 Generalization, complexity and model selection

In a first set of experiments, several parameter settings of the ANN model were tried on random partitions of the available data in order to evaluate and select the appropriate complexity of the ANN model for the problem at hand. In these first experiments, stepwise multiple linear regression was also considered as a baseline. Using well separated and representative training and test data sets is very important to obtain reliable results and well founded conclusions about the generalization abilities of different approaches.

Cross validation (CV) and *m*-fold CV in particular, is regarded as the best practical way of making use of available data while keeping bias and variance of the error estimates at moderate values. Low values of *m* are closer to optimality in the asymptotic case and lead to a behavior similar to using only one random bipartition (as in Kose 2007) while higher values lead to dependent and biased estimates but are well behaved in practice (Jain *et al*. 1987). Even though some authors recommend settings as m=10 when using standard databases of moderate to large sizes (Kohavi 1995), in our case and possibly due to the high variability in the available data, we observed slightly more stable results with lower values of *m*. In particular, 4-fold cross validation experiments taking 75% of the cases (arranged in three folds of 25% each) for train and the remaining 25% (the fourth fold) for test were considered. The whole procedure was rotated over the 4 folds and repeated 4 times (4 different random partitions) so that each single case was taken into account for testing exactly 4 times. Note that, in this whole experimentation, a total of 16 prediction models were produced from the database.

As data came from 18 different sources in a very unbalanced way, a special care was taken to ensure that these were equally spread over all folds. In particular, if folds of size αS are to be generated (where $\alpha = 0.25$, $S = s_1 + \cdots + s_{18}$ is the total size and s_i is the amount of data from source i), exactly $[\alpha s_i]$ cases are put in each fold by performing random selection and replicating data if necessary. Final performance measures are the averages over all the results per fold along with their corresponding standard deviations.

Stepwise multiple linear regression was used in a standard way by including variables whose associate *p*-values are at most 0.005. In particular, the method selected the ordered set of variables {8, 20, 11, 5} or {8, 15, 11, 12, 18, 5, 9, 3} (see input features of Table 1) depending on partitions and folds.

With regard to experiments involving ANN models, an extensive experimentation using a wide range of learning algorithms and parameters was tried in our database. Nevertheless, the setting finally used for TL prediction using MLP was fixed following the one used in Kose (2007) because none of the other options was clearly superior in terms of generalization ability and also to facilitate comparison with previous works.

As in our database, variability is high and more important than the expected intrinsic complexity of the problem, a relatively simple ANN model with a reduced range of parameters has been considered to carry out exhaustive experimentation. In particular, only one hidden layer was considered and the number of neurons, *k*, ranged from 2 to 9. Note that very similar settings are

used in other related works dealing with this and other engineering problems (Kose and Kayadelen 2010, Bilgehan and Turgut 2010b). The activation function at each neuron is the hyperbolic tangent function. Weights were learnt using the Levenberg-Marquardt (LM) algorithm that constitutes a kind of tradeoff between gradient descent and second order (Gauss-Newton like) methods. Moreover, this learning algorithm showed faster convergence rates than other options while keeping the same performance figures (Hagan and Mehaj 1994). To control learning rate, the LM algorithm uses a dumping parameter (μ = 0.001) which can be increased or decreased (increasing and decreasing factors are 10 and 0.1) to a maximum value (10^{10}). Each network architecture (i.e., number of neurons) was tried 5 times on the same data and run for a maximum of 25 epochs to prevent overtraining. Performance goal and minimum gradient parameters have been set to 10^{-6} and 10^{-10}, respectively. The best network on this data was kept to be used for predicting test data. This gave relatively good results with regard to preventing overfitting in our particular database.

The averaged prediction results obtained both with multiple linear regression and ANN models are shown in the Fig. 5. In particular, R^2, RPE, RMSE and AUC measures are shown both for training and test data. It is important to remark that these measures were averaged over a large effective number of models and data. In particular, 16 models trained with 75% of data and 4 times 100% of test data (16 times 25% of available data). Note also that standard deviation of these averaged measures is relatively large in the corresponding figures.

As expected, the prediction accuracy measured on the training data either by the R^2 coefficient or the RMSE (Figs. 5(a) and 5(b)) improves clearly when the number of neurons is increased. With a small number of neurons (i.e., less than 4) the results are very similar to the ones obtained with multiple linear regression (especially with regard to RMSE values). On the other hand, the RPE measure remains approximately constant around its best value but its variability increases with the number of neurons.

On the other hand, when predicting outputs corresponding to test data (that was not taken into account to construct the prediction models); the performance gets worse as the number of neurons increases (and in particular for 5 neurons and more). This puts forward the strong tradeoff between complexity and generalization ability. Even though the variability is high, it can be observed a significant difference (in R^2) between the pure linear predictor and the MLP model, specifically when using 4 neurons. The same best result is observed in the RMSE results.

Finally, when considering the proportion of correctly predicted values within a margin (Fig. 5(c)), several important differences between prediction models can be observed. First, the difference between ANN and linear models for most values of k was increased and the behavior of the model with regard to this parameter is more stable. It can be said that the curves in Fig. 5(c) are more representative of the expected behavior of the model with real (noisy) data. From this result it can be seen that even with 5 neurons it is possible to obtain a good model with regard to generalization ability. Nevertheless, for subsequent experiments using a unique MPL predictor, the value of k = 4 was adopted as a convenient and conservative tradeoff.

5.3 Sensibility analysis and selection of variables

From the point of view of the engineering problem, it is interesting to know the relative importance of the input variables with regard to the goodness of the prediction model. In its full generality, this is a very hard problem due to complex dependence relations among variables (Somol and Pudil 2002) but we are only interested in obtaining an approximate ranking together

with a measure of importance associated to each variable. To this end, an empirical study was carried out using all available data and two prediction models: the linear one and the MLP model with 4 units in the hidden layer.

The ranking of variables for the linear model was directly obtained from the stepwise regression method, which was forced to use all input variables. The corresponding measure in this case is the RMSE associated to the corresponding predictions using each variable (accumulated to the previously selected ones). The results obtained are shown in Fig. 6(a) where features are shown in decreasing order of importance. The last (best) value of RMSE corresponds to the whole set of variables. Note that performance gets very close to the best with roughly half the available set of variables. It is important to note that the ranking of variables obtained when using the whole database is very similar to the sets of variables selected in the experiments with random partitions in the previous section for linear models. In particular, variables 8-*cv*, 15-f_{p0}, and 11-*transv* are present in all sets. The fourth variable in the ranking is 12-f_{ci}. The remaining variables to complete the 10 best input variables are: 1 - d_b, 7 - *h*, 9 - *ch*, 17 - f_{si}, 14-*extr*, and 3 - A_p. Worth remarking is also the fact that all combined input variables (16 and 18 to 21) significantly decreased its importance in this experiment with regard to the one in Section 5.2 possibly due to the inherent variability in the data and its impact depending on training sizes.

As it can be observed, the variables 15 - f_{p0}, 12 - f_{ci}, 1 - d_b, 17 - f_{si}, 14 - *extr* (related to the release method) and 3 - A_p are included in Eq. (1). The concrete compressive strength at prestress transfer, 12 - f_{ci}, appears with an important significance in Fig. 6a, and several authors (Cousins *et*

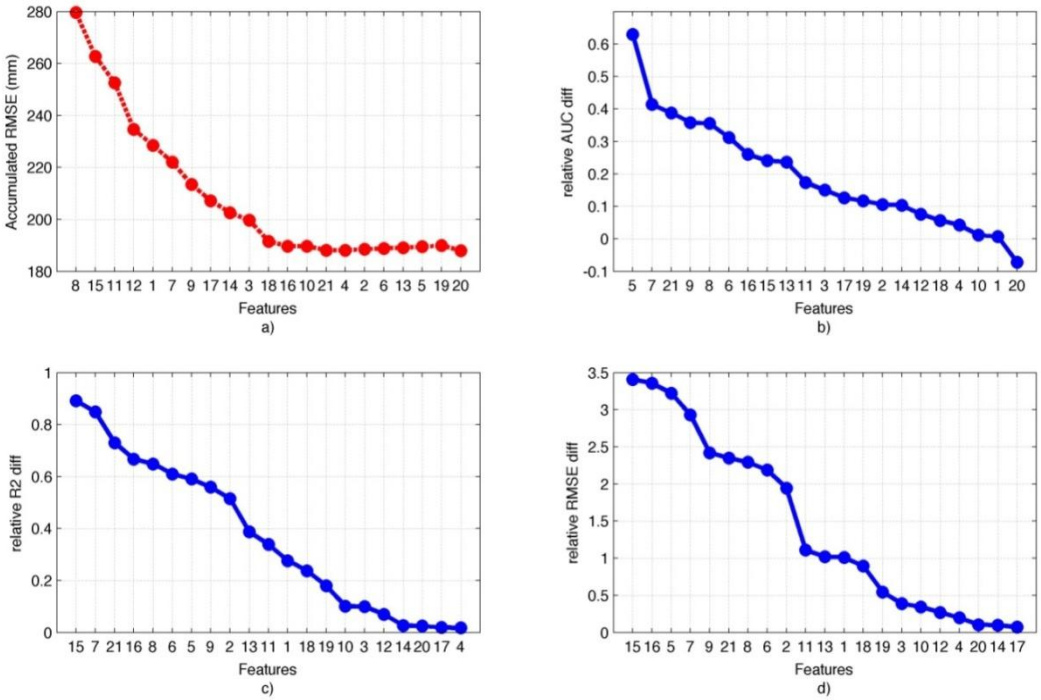

Fig. 6 (a) Ranking of variables for the linear prediction model along with the RMSE value associated to the accumulated subsets of variables. (b), (c) and (d) Importance values associated for each input variable from the sensitivity analysis for the ANN model using the three considered criteria

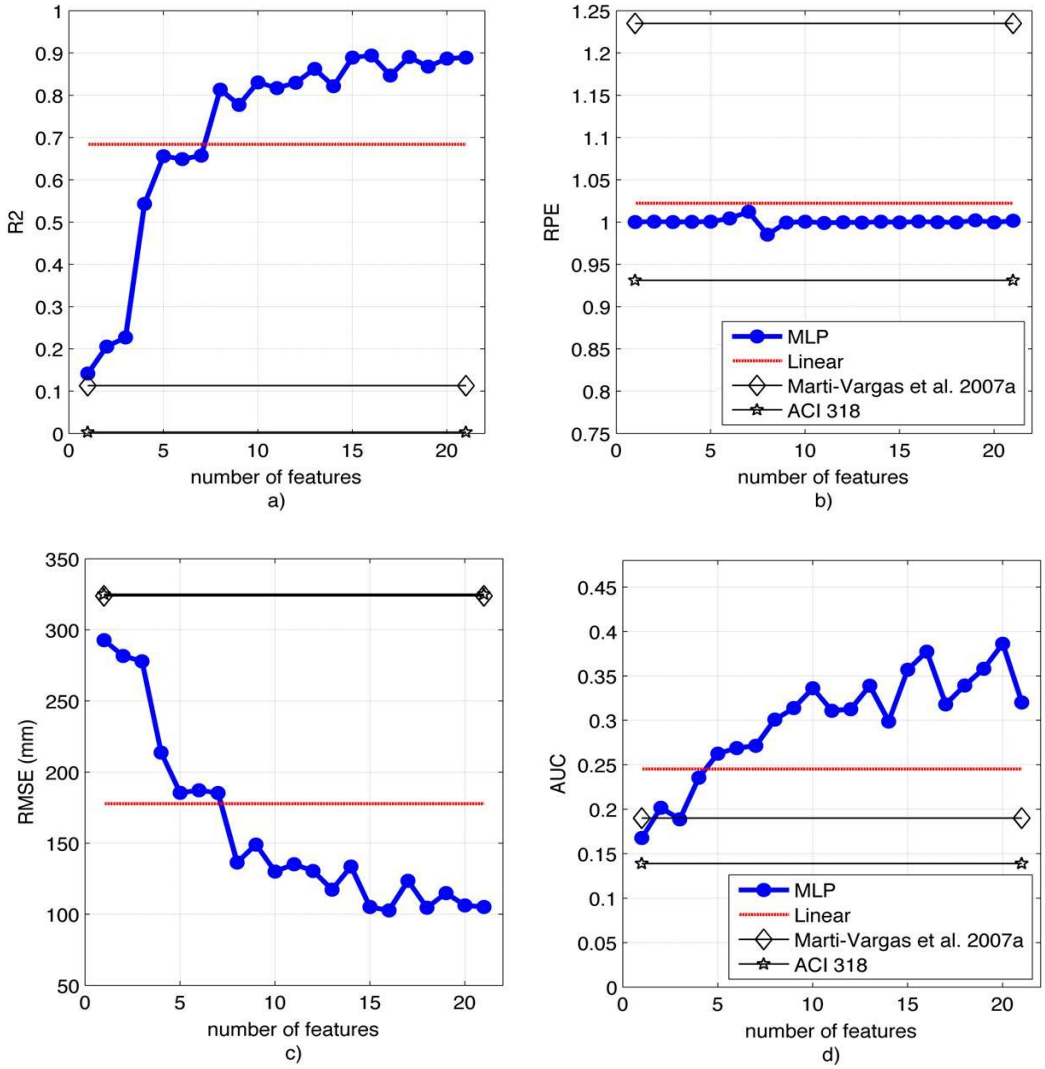

Fig. 7 Prediction results when increasing number of variables (ordered by importance) are considered

al. 1990, Mitchell *et al*. 1993, Mahmoud *et al*. 1999, Martí-Vargas *et al*. 2012a) proposed U_t values in Eq. (1) as a function of f_{ci} in contrast to the constant value U_t = 2.76 MPa used by ACI Code 318 provisions (ACI 2011).

It is also important to note that the variables 8-*cv* and 9-*ch*, related to the concrete cover, and the variables 11-*transv* and 7-*h* representatives of the confinement are not included in Eq. (1). According to FIB (2000), a minimum concrete cover is required to prevent bond splitting, and a reduction of the TL can be expected when the concrete cover increases. On the other hand, confining effects benefit the bond performance (CEB 1987), and therefore the placement of transverse reinforcement was recommended (Russell and Burns 1997, Mahmoud *et al*. 1999) even though its influence was not quantified.

With the ANN model, an analysis of sensitivity was carried out using the sensitivity based input pruning method (Moody and Utans 1992, Diaz-Villanueva *et al*. 2010). Each input variable in turn was substituted in all input data by its corresponding average value. The (negative) impact on the predictions was measured (using R^2, RMSE and AUC). The more impact on the prediction, the more important the feature is. Relative differences on the three criteria are used to assign importance values to each of the input variables. The results obtained for the whole set of variables are shown in Figs. 6(b), 6(c) and 6(d).

The three criteria used for the ANN model lead to very similar rankings. In particular, the 8 best input variables are the same according to all criteria and there is only one and two disagreements in the first 9 and 10 respectively (involving variables 2 - f_{pu}, 11 - *transv* and 13 - *dest*).

In order to evaluate the real impact of the subsets of variables used in the ANN model, an increasing number of input variables according to the importance ranking was considered in the training and testing of the model in a new experiment. The best settings including number of units and learning parameters were taken from the previous experimentation about model selection. The particular ranking used in this experiment was fixed to the one obtained using the AUC criterion (Fig. 6(b)).

After retraining the neural model with all available data with an increasing number of input variables ordered according to the previous fixed ranking, the prediction results (R^2, RMSE and AUC) followed a slight increasing (decreasing in the RMSE case) behavior as shown in Fig. 7. The RPE values shown in Fig. 7(b) are very close to the optimal value regardless of the number of features kept.

Fig. 7 also shows the prediction results obtained through multiple linear regression and standard formulae as a reference. It can be seen that none of the performance measures gets significantly worse (with regard to the model with all input variables and within a moderate variability) when at least 10 variables are used. Consequently, the ANN model with (the best) 10 input variables (d = 10) can be considered as a good and convenient tradeoff (the stepwise linear regression method selects 8 input variables with p = 0.005 when all database is considered). The particular ANN model with 10 input variables (d = 10) and 4 hidden units (k = 4) with all its weights and biases is shown in Table 3.

As it can be observed, the ANN model with 10 input variables (Table 3 and Fig. 6(b)) includes 5 of the 10 most important variables identified by the linear regression model (Fig. 6(a)): 8 - *cv*, 15 - f_{p0}, 11 - *transv*, 7 - *h*, and 9 - *ch*. The most worth remarking differences between variables identified as important by ANN comprise variables 5 - *ns*, 21 - *modsep*, 6 - *b*, 16 - A_p/A_c, and 13-*dest*, instead of variables 12 - f_{ci}, 1 - d_b, 17 - f_{si}, 14 - *extr*, and 3-A_p identified by linear regression (Fig. 6(a)). In the case of the ANN model, only variables15 - f_{p0} and 13-*dest* are included in Eq. (1). Important variables according to ANN model not included in Eq. (1) are geometric parameters such as 8-*cv*, 7 - *h*, 9-*ch*, 5 - *ns*, 21 - *modsep*, 6-*b*, and 16 - A_p/A_c influencing concrete cover and strand spacing (Cousins *et al*. 1994, Deatherage *et al*. 1994, FIB 2000), and also influencing confinement joint the variable 11-*transv* as it was aforementioned.

In conclusion, the linear regression model has identified more variables included in Eq. (1) than the ANN model. However, the ANN model shows a better prediction of TL than the linear regression model (Fig. 7). Note that the effect on TL of the properties and parameters are usually described by using regression analysis and statistical models as it was explained above. Therefore, future works with other techniques and data may help to obtain new expressions that include the most representative variables of the bond phenomenon to get a best prediction of TL.

Table 3 Parameters of the finally proposed ANN model with $d = 10$ and $k = 4$

First hidden layer											
Inputs (features)	5	7	21	9	8	6	16	15	13	11	bias
Unit 1	0.4022	0.9850	-0.8248	0.8881	1.5718	2.1317	-0.4759	4.6902	-0.3137	-0.0154	-3.5620
Unit 2	-1.0781	-0.8634	1.3991	-0.9036	-0.7540	-0.1825	1.2243	-0.9332	-0.8194	0.0282	0.7969
Unit 3	-1.1415	1.4824	-0.4791	-0.5206	0.1641	-1.4070	1.1626	-2.8920	1.7839	-0.1336	3.2641
Unit 4	0.3641	1.2234	-0.2753	1.2037	-1.7048	0.6937	-0.0418	1.3243	-1.3003	0.4930	1.2475

Second output layer					
Inputs	Unit 1	Unit 2	Unit 3	Unit 4	bias
Output unit	-4.8821	-3.4598	-2.6726	-0.9191	-0.303

5.4 Sensibility analysis and selection of variables

Final experiments using both full ($d = 21$) and reduced ($d = 10$) ANN models using the settings arrived at in the previous section have been also performed. Comparative experimentation was carried out with these neural models along with the linear model and their results were compared to the prediction given by two of the parametric formulae available in the specific literature. In particular, the one in the standard ACI 318 (2008) and the one proposed in Martí-Vargas *et al*. (2007a).

The ACI 318 equation for TL is

$$L_t = \frac{f_{se} d_b}{21} \tag{7}$$

which can be obtained from Eq. (1) considering $f_{px} = f_{se}$, $A_p = 0.725\pi d_b^2/4$, $k_1 = 4/3$, $U_t = 2.76$ MPa, and $k_2 = 0$.

The equation proposed by Martí-Vargas *et al*. (2007a) is

$$L_t = \frac{f_{si} A_p}{(4/3)\pi d_b 0.4 f_{ci}^{0.67}} \tag{8}$$

which can be obtained from Eq. (1) considering $f_{px} = f_{si}$, $k_1 = 4/3$, $U_t = 0.4 f_{ci}$, and $k_2 = 0$.

The scatter plots corresponding to the four prediction models are shown in Fig. 8 (the plot for the reduced ANN model is not shown as it is visually very similar to the one with the full model). From these scatter plots, it can be seen that linear models produce results with an equally spread prediction error in the output range. On the other hand, neural models (with few neurons) tend to produce results visually more accurate than the linear ones but with few outliers that have a high impact in the R^2 and RMSE measures. This behavior was observed also in almost all partitions and folds in the experiments in Section 5.2. The superiority of both adaptive prediction models (both linear and neural) with regard to the use of fixed formulas is clear from these plots.

The curves showing the prediction rate with regard to the maximum relative error permitted (margin) are plotted in Fig. 9. And finally, the corresponding AUC values for each curve along

with the R^2, RPE and RMSE values for all models are shown in Table 4. From the results obtained one can conclude that methods that adapt to training data using several input variables are clearly superior to any of the proposed formulae. This is not in contradiction with the fact that the formulae use the correct prior assumption about the underlying phenomenon leading to TL. This only implies that variabilities, noise and errors in the procedures lead to a too complex prediction problem that cannot be modeled in practice with such a (simple) formula. The best results of all criteria correspond to ANN. These models lead to expected errors (as measured by RMSE) in the range 105-130 mm which is roughly 30% and 60% smaller than the results obtained using linear regression and fixed formulae, respectively. Similar differences are observed in the other criteria. The other key fact is that prediction curves and AUC measures account in a more robust way for the goodness of the prediction and nicely correlate with the visual and practical results obtained with each model.

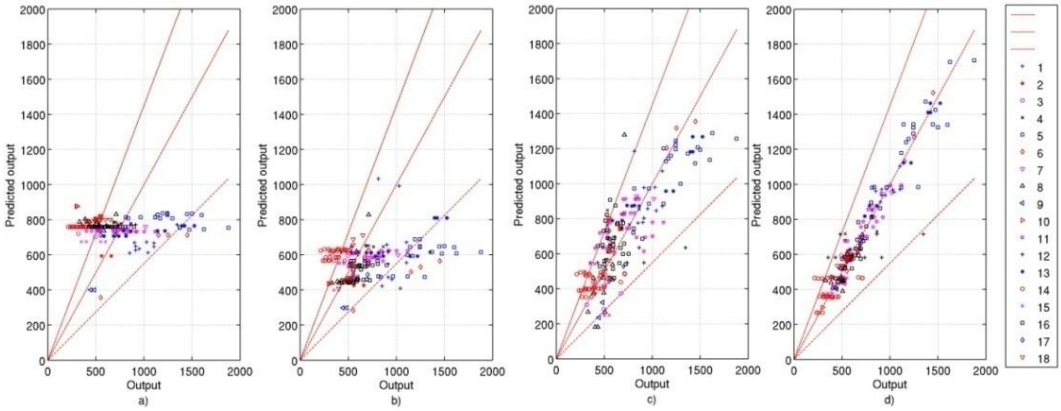

Fig. 8 Scatter plots showing predicted TL in terms of the measured value for ACI (2011), Martí-Vargas *et al*. (2007a), linear and ANN

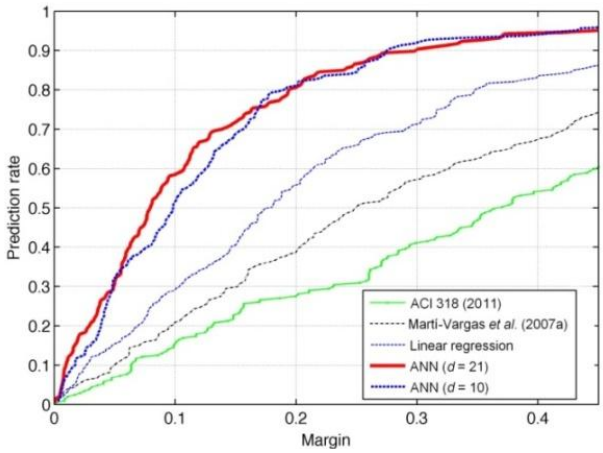

Fig. 9 Prediction rate with regard to maximum relative error for all considered prediction models

Table 4 Prediction accuracy measures for all models on the whole database

	ACI (2011)	Martí-Vargas *et al*. (2007a)	Linear	ANN (d=21)	ANN (d=10)
R^2	0.003	0.113	0.684	0.889	0.831
RPE	0.931	1.235	1.022	0.999	1.000
RMSE	325	324	178	105	130
AUC	0.139	0.190	0.245	0.320	0.336

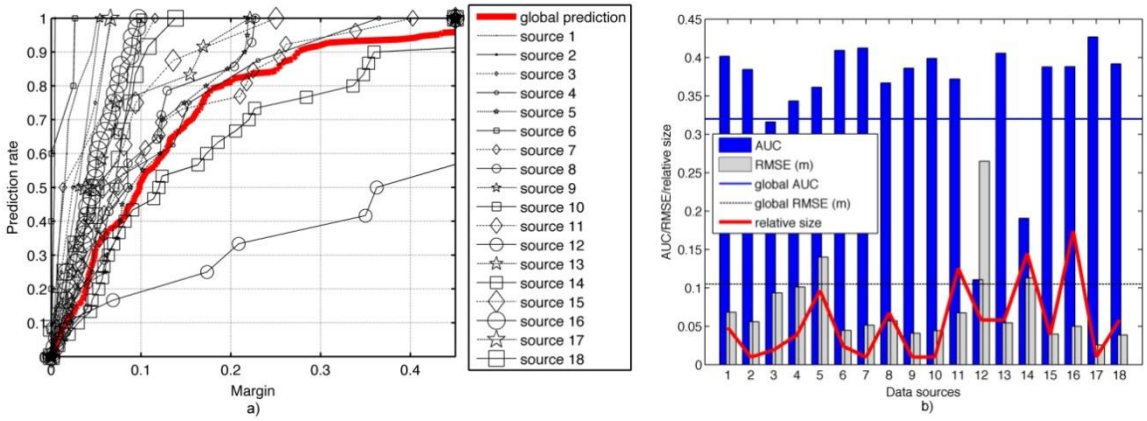

Fig. 10 Prediction curves and performance measures for data coming from each of the 18 different sources along with global measures. The RMSE is shown in meters (m) and the relative size of each source is shown as a continuous line

6. Comparing different sources of data

As a final experiment to illustrate how differently the data from different sources behave, the same performance measures taken in the previous section were computed separately for data coming from each one of the 18 different sources considered in this work (see Table 2). For this experiment, the proposed ANN model with k = 4 and 10 input variables has been used to predict TL.

Figs. 10(a) and 10(b) illustrate the corresponding prediction curves and prediction performance measures, respectively (R^2 measures are not shown in the figure as they are highly correlated with RMSE). Fig. 10(b) also shows the relative size of each data source with regard to the total number of cases in the whole database. What can be immediately seen is that the variability among different sources is very high regardless of the prediction measure considered. For example, the RMSE ranges from 25 to 265 mm (sources 17 and 12, respectively). It is worth remarking that in most cases, good AUC values imply also good RMSE values. But the global behavior is slightly different. It is also noticeable that sources 12 and 14 give extremely low values of AUC (which is also apparent in Fig. 10a) that are about 25% and 50% the average AUC value, respectively. These two sources along with source 5 lead also to the worst RMSE results. The best behaved source for both measures is 17 and other sources with good values of AUC and RMSE are 6,7,9,10,13,15 and 18. Note that the most populated sources are not necessarily among the best behaved ones.

In our opinion, the worse results on TL obtained by sources 5, 12 and 14 may be caused by additional unreported factors related to the strand surface condition as residues from production process or lab-stored incidents. On the other hand, almost half of the sources show good values for

the analyzed parameters. Moreover, the TL results from source 18 -obtained by means the recent methodology ECADA based on measuring the effective prestressing force (Martí-Vargas *et al*. 2006a) - showed good values in the prediction curves.

7. Conclusions

The bond performance of prestressed reinforcement is a complex phenomenon, and the TL prediction is a difficult task to achieve giving rise to the fact that several parameters influencing TL do not appear in the proposed equations from authors and Codes in the literature.

TL value of prestressing strand is an important design parameter for pretensioned prestressed and therefore it is reasonable the application of data-mining methods as they are currently being introduced in this field. More than using a specific method, an appropriate assessment protocol and methodology is needed in order to be able to draw useful and practical outcomes from the different prediction processes. This work constitutes a first attempt in this direction using a widely used neural based predictor. The use of ROC-like measures to assess prediction performance gives rise to an innovative analysis for TL prediction in which other classical measures as R^2 and RMSE were considered. Prediction curves and AUC measures account in a more robust way for the goodness of the prediction and nicely correlate with the visual and practical results obtained with different models.

A carefully selected database spanning a variety of practical TL of prestressing strands from real prediction situations coming from different sources was compiled. The amount of information and the specific trade off between homogeneity (in the range of parameters), specificity (in problem definition) and variability (coming from the different experimental methods) makes this database and its study one of the strong points of the present work.

The straightforward sensitivity analysis carried out has lead to an approximate ranking of variables according to several importance measures related to TL prediction. Important variables as 8, 15, 11, 7, and 9 were identified by all considered methods and criteria. The linear regression model has identified more variables included in the proposed equations from authors and Codes in the literature. However, several important variables related to geometric parameters influencing concrete cover, strand spacing and confinement do not appear in these equations and their influence on TL was detected by means ANN. Moreover, the ANN model shows a better prediction of TL than the linear regression model.

Methods that adapt to training data using several input variables are clearly superior to any of the proposed formulae for TL in the literature. This is not in contradiction with the fact that the formulae use the correct prior assumption about the underlying phenomenon leading to transfer length. This only implies that variabilities, noise and errors in the procedures lead to a too complex prediction problem that cannot be modeled in practice with such a simple formula.

Powerful prediction models as ANN need to conservatively control the amount of nonlinearity (or degrees of freedom) in order to arrive at convenient predictions given the huge amount of noise and variability which is present in the data when it comes from different sources.

A final ANN model trained with all available data is proposed using only 4 hidden units and 10 selected variables. Expected errors in TL are about 30% and 60% smaller than the ones corresponding to using linear regression and fixed formulas, respectively.

The quality of the data coming from different sources was also studied with regard to its behavior when input to the learned ANN models. As a result of this study, it can be said that almost

half of the sources showed a good predicting behavior. When worse results on TL are obtained systematically for a given source, they may be caused by additional unreported factors related to the strand surface condition as residues from production process or lab-stored incidents among others.

Acknowledgments

Funding for this study were received from the Spanish Ministry of Science and Innovation and ERDF (Research Project BIA2006-05521, BIA2009-12722, and BIA2011-23602) and from the Spanish Ministry of Education (TIN2009-14205-C04-03 and Consolider Ingenio 2010 CSD2007-00018), as well as the European Community with the FEDER funds.

References

ACI 318-11 (2011), *Building Code Requirements for Reinforced Concrete*, (ACI 318-11), ACI, USA.

Adeli, H. (2001), "Neural networks in civil engineering: 1989-2000", *Comput.-Aided Civil Infrastruct. Eng.*, **16**(2), 126-142.

Adeli, H. and Yeh, C. (1989), "Perceptron learning in engineering design", *Microcomput. Civil Eng.*, **4**(4), 247-256.

Balázs, G. (1993), "Transfer length of prestressing strand as a function of draw-in and inicial prestress", *PCI J.*, **38**(2), 86-93.

Balogh, T. (1992), "Statistical distribution of draw-in of seven-wire strands", *Proceedings of the International Symposium Bond in Concrete: From Research to Practice*, Riga Technical University and CEB, 2.10-2.19, Riga.

Barnes, R.W., Grove, J.W. and Burns, N.H. (2003), "Experimental assessment of factors affecting transfer length", *ACI Struct. J.*, **100**(6), 740-788.

Bishop, C.M. (1995), *Neural Networks for Pattern Recognition*, University Press, Oxford, UK.

Bilgehan, M. and Turgut, P. (2010a), "The use of neural networks in concrete compressive strength estimation", *Comput. Concrete*, **7**(3), 271-283.

Bilgehan, M. and Turgut, P. (2010b). "Artificial neural network approach to predict compressive strength of concrete through ultrasonic pulse velocity", *Res. Nondestruct. Eval.*, **21**(1), 1-17.

Bilgehan, M. (2011). "A comparative study for the concrete compressive strength estimation using neural network and neuro-fuzzy modelling approaches", *Res. Nondestruct. Eval.*, **26**(1), 35-55.

Caro, L.A., Martí-Vargas, J.R. and Serna, P. (2012), "Time-dependent evolution of strand transfer length in pretensioned prestressed concrete members", *Mech. Time-Depend. Mater.*, doi:10.1007/s11043-012-9200-2.

Caro, L.A., Martí-Vargas, J.R. and Serna, P. (2013), "Prestress losses evaluation in prestressed concrete prismatic specimens", *Eng. Struct.*, **48**, 704-715.

CEB (1987), *Bulletin d'Information n°181: Anchorage Zones of Prestressed Concrete Members*, Comité Euro-International du Béton, Lausanne, Switzerland.

CEN (2004), *EN 1992-1-1:2004. Eurocode 2. Design of Concrete Structures - Part 1-1: General Rules and Rules for Building*, CEN, Brussels, Belgium.

Chou, J.S., Chiu, Ch.K., Farfoura, M. and Al-Taharwa, I. (2011), "Optimizing the prediction accuracy of concrete compressive strength based on a comparison of data-mining techniques", *J. Comput. Civil Eng. - ASCE*, **25**(3), 242-253.

Cousins, Th.E., Johnston, D.W. and Zia, P. (1990), "Transfer and development length of epoxy-coated and uncoated prestressing strand", *PCI J.*, **35**(4), 92-103.

Cousins, Th.E., Stallings, J.M. and Simmons, M.B. (1994), "Reduced strand spacing in pretensioned, prestressed members", *ACI Struct. J.*, **91**(3), 277-286.
Dahou, Z., Sbartaï, Z.M., Castel, A. and Ghomari, F. (2009), "Artificial neural network model for steel-concrete bond prediction", *Eng. Struct.*, **31**(8), 1724-1733.
Deatherage, J.H., Burdette, E. and Chew, Ch.K. (1994), "Development length and lateral spacing requirements of prestressing strand for prestressed concrete bridge girders", *PCI J.*, **39**(1), 70-83.
Diaz-Villanueva, W., Ferri, F.J. and Cerverón, V. (2010), "Learning improved feature rankings through decremental input pruning for support vector based drug activity prediction", *Proceedings of IEA/AIE'10 (2), Lecture Notes in Artificial Intelligence*, vol. 6097, 653-661, Springer.
Dorsten, V., Hunt, F.F. and Preston, H.K. (1984), "Epoxy coated seven-wire strand for prestressed concrete", *PCI J.*, **29**(4), 100-109.
Draper, N.R. and Smith, H. (1998), *Applied Regression Analysis*, 3rd Ed., John Wiley & Sons, NY, USA.
Fawcett, T. (2006), "Introduction to ROC analysis", *Pattern Recogn. Lett.*, **27**(8), 861-874.
FIB (2000), *Bulletin nº10: Bond of Reinforcement in Concrete. State of the Art Report*, International Federation for Structural Concrete, Lausanne, Switzerland.
FIB (2010), *Bulletin nº55: Model Code 2010. First Complete Draft - Volume 1*, International Federation for Structural Concrete, Lausanne, Switzerland.
García-Taengua, E., Martí-Vargas, J.R. and Serna-Ros, P. (2011), "Statistical approach to effect of factors involved in bond performance of steel fiber reinforced concrete", *ACI Struct. J.*, **108**(4), 461-468.
Hagan, M.T. and Menhaj, M.B. (1994), "Training feedforward networks with the Marquardt algorithm", *IEEE T. Neural Netw.*, **5**(6), 989-993.
Hanson, N.W. (1969), "Influence of surface roughness of prestressing strand on bond performance", *PCI J.*, **14**(1), 32-45.
Holmberg, A. and Lindgren, S. (1970), *Document D1/1970: Anchorage and Prestress Transmission*, National Swedish Building Research, Stockholm, Sweden.
Hornik, K., Stinchcombe, M. and White, H. (1989), "Multilayer feedforward networks are universal approximator", *Neural Netw.*, **2**(5), 359-366.
Issa, M.A., Sen, R. and Amer, A. (1993), "Comparative study of transfer length in fiberglass and steel pretensioned concrete members", *PCI J.*, **38**(6), 52-63.
Jain, A.K., Dubes, R.C. and Chen, C.C. (1987), "Bootstrap Techniques for Error Estimation", *IEEE T. Pattern Anal.*, PAMI, **9**(5), 628-633.
Kaar, P.H., LaFraugh, R.W. and Mass, M.A. (1963), "Influence of concrete strength on strand transfer length", *PCI J.*, **8**(5), 47-67.
Kohavi, R. (1995). "A study of cross-validation and bootstrap for accuracy estimation and model selection", *Procs of the 14th Intl. Joint Conf. on Artificial Intelligence*, **2**, 1137-1143.
Kose, M.M. (2007), "Prediction of transfer length of prestressing strands using neural networks", *ACI Struct. J.*, **104**(2), 162-169.
Kose, M.M. and Kayadelen, C. (2010), "Modeling of transfer length of prestressing strands using genetic programming and neuro-fuzzy", *Adv. Eng. Softw.*, **41**(2), 315-322.
Kovács, I. and Balázs, G.L. (2004), "Behaviour of steel fibre reinforced prestressed pretensioned concrete beams", *Structural Performance of Steel Fibre Reinforced Concrete*, Ed. Z. Wintermantel, Publishing Company of Budapest University of Technology and Economics, Budapest, 131-163.
Lane, S. (1992), "Transfer lengths in rectangular prestressed concrete concentric specimens", *Public Roads*, **56**(2), 67-71.
Logan, D. (1997), "Acceptance criteria for bond quality of strand for pretensioned prestressed concrete applications", *PCI J.*, **42**(2), 52-90.
Mahmoud, Z.I., Rizkalla, S.H. and Zaghloul, E.R. (1999), "Transfer and development lengths of carbon fiber reinforcement polymers prestressing reinforcing", *ACI Struct. J.*, **96**(4), 594-602.
Martínez, F.J., Gonzalez-Vidosa, F., Hospitaler, A. and Yepes, V. (2010), "Heuristic optimization of RC bridge piers with rectangular hollow sections", *Comput. Struct.*, **88**(5-6), 375-386.
Martí-Vargas, J.R., Serna-Ros, P., Fernández-Prada, M.A., Miguel-Sosa, P.F. and Arbeláez, C.A. (2006a),

"Test method for determination of the transmission and anchorage lengths in prestressed reinforcement", *Mag. Concr. Res.*, **58**(1), 21-29.
Martí-Vargas, J.R., Arbeláez, C.A., Serna-Ros, P., Fernández-Prada, M.A. and Miguel-Sosa, P.F. (2006b), "Transfer and development lengths of concentrically prestressed concrete", *PCI J.*, **51**(5), 74-85.
Martí-Vargas, J.R., Arbeláez, C.A., Serna-Ros, P., Navarro-Gregori, J. and Pallarés-Rubio, L. (2007a), "Analytical model for transfer length prediction of 13 mm prestressing strand", *Struct. Eng. Mech.*, **26**(2), 211-229.
Martí-Vargas, J.R., Arbeláez, C.A., Serna-Ros, P. and Castro-Bugallo, C. (2007b), "Reliability of transfer length estimation from strand end slip", *ACI Struct. J.*, **104**(4), 487-494.
Martí-Vargas, J.R., Serna, P., Navarro-Gregori, J. and Bonet, J.L. (2012a), "Effects of concrete composition on transmission length of prestressing strands", *Const. Build. Mater.*, **27**, 350-356.
Martí-Vargas, J.R., Serna, P., Navarro-Gregori, J. and Pallarés, L. (2012b), "Bond of 13 mm prestressing steel strands in pretensioned concrete members", *Eng. Struct.*, **41**, 403-412.
Martí-Vargas, J.R., Caro, L.A. and Serna, P. (2012c), "Experimental technique for measuring the long-term transfer length in prestressed concrete", *Strain*, **49**, 125-134.
MathWorks (2011) Matlab [Computer programme] [accessed 2011 Oct 12]. Available from: http://www.mathworks.com
McCulloch, W. and Pitts, W. (1943), "A logical calculus of ideas immanent in nervous activity", *B. Math. Biophy.*, **5**, 115-133.
Mitchell, D., Cook, W.D., Khan, A.A. and Tham, Th. (1993), "Influence of high strength concrete on transfer and development length of pretensioning strand", *PCI J.*, **38**(3), 52-66.
Moody, J. and Utans, J. (1992), "Principled architecture selection for neural networks: Application to corporate bond rating prediction", *Advances in Neural Information Processing Systems*, Morgan Kauffmann Publishers, Ed. J. Moody, S. Hanson and R. Lippmann, **4**, 683-690.
Nanni, A. and Tanigaki, M. (1992), "Pretensioned prestressed concrete members with bonded fiber reinforced plastic tendons: development and flexural bond lengths (static)", *ACI Struct. J.*, **89**(4), 433-441.
Oh, B.H. and Kim, E.S. (2000), "Realistic evaluation of transfer lengths in pretensioned, prestressed concrete members", *ACI Struct. J.*, **97**(6), 821-830.
Paclík, P., Lai, T., Landgrebe, T. and Duin, R.P.W. (2010), "ROC analysis and cost-sensitive optimization for hierarchical classifiers", *Proceedings of ICPR'2010*, IEEE, 2977-2980.
Parichatprecha, R. and Nimityongskul, P. (2009), "An integrated approach for optimum design of HPC mix proportion using genetic algorithm and artificial neural networks", *Comput. Concrete*, **6**(3), 253-268.
Rose, D.R. and Russell, B.W. (1997), "Investigation of standardized tests to measure the bond performance of prestressing strand", *PCI J.*, **42**(4), 56-80.
Rumelhart, D.E., McClelland, J.L. and the PDP Research Group (1986), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Volume 1: Foundations, MIT Press, Cambridge, UK.
Russell, B.W. and Burns, N.H. (1996), "Measured transfer lengths of 0.5 and 0.6 in. strands in pretensioned concrete", *PCI J.*, **41**(5), 44-65.
Russell, B.W. and Burns, N.H. (1997), "Measurement of transfer length of pretensioned concrete elements", *ASCE J. Struct. Eng.*, **123**(5), 541-549.
Sancak, E. (2009), "Prediction of bond strength of lightweight concretes by using artificial neural networks", *Sci. Res. Essays*, **4**(4), 256-266.
Somol, P. and Pudil, P. (2002), Feature selection toolbox, *Pattern Recognit.*, **35**(12), 2749-2759.
Tanyildizi, H. (2009), "Fuzzy logic model for the prediction of bond strength of high-strength lightweight concrete", *Adv. Eng. Softw.*, **40**(3), 161-169.
Thorsen, N. (1956), "Use of large tendons in pretensioned concrete", *ACI J.*, **53**(6), 649-659.
Uijl, J.A. (1995), "Transfer length of prestressing strands in HPC", *Prog. Concr. Res.*, **4**, 75-90.
Yeh. I.C. (2008), "Modeling slump of concrete with fly ash and superplasticizer", *Comput. Concrete*, **5**(6), 559-572.
Yepes, V., González-Vidosa, F., Alcalá, J. and Villalba, P. (2012), "CO_2-Optimization design of reinforced concrete retaining walls based on a VNS-threshold acceptance strategy", *J. Comp. Civil Eng. - ASCE*,

26(3), 378-386.
Yepes, V., Díaz, J., Gonáalez-Vidosa, F. and Alcalá, J. (2009), "Statistical characterization of prestressed concrete road bridge decks", *Rev. Constr.*, **8**(2), 95-108.

CC