Neural Computing and Applications (2019) 31:7227–7243
https://doi.org/10.1007/s00521-018-3559-1

ORIGINAL ARTICLE

A web-oriented expert system for planning hurdles race training programmes

Krzysztof Przednowek[1] · Krzysztof Wiktorowicz[2] · Tomasz Krzeszowski[2] · Janusz Iskra[3]

Received: 22 March 2017 / Accepted: 21 May 2018 / Published online: 30 May 2018

Abstract

This paper presents a web-oriented expert system, named iHurdling, to predict results and generate training loads for 110 and 400 m hurdles races. The database contains 40 annual training programmes for the 110 m hurdles and 48 programmes for the 400 m hurdles. The predictive models include linear regressions in the form of ordinary least squares, ridge, LASSO, elastic net and nonlinear models in the form of a radial basis function neural network and fuzzy rule-based system. The leave-one-out cross-validation method is used to compare, and choose the best model. It shows that the proposed fuzzy-based model has the lowest validation error. The developed web application can support a coach in planning training programmes for hurdles races. It allows the athlete's results to be predicted and can generate training loads for an athlete, selected from database. The application can be run on a computer or a mobile device. The system was implemented using the R programming language with the Shiny framework and additional packages. The limitations of the presented approach are related to the lack of consideration of an athlete's physiological and psychological parameters, but the generated training programs might be used as a suggestion for the coach.

Keywords Predictive models · Expert system · Hurdles race · R programming language

1 Introduction

Expert systems have been widely implemented and examined by researchers. They mimic the decision-making abilities of a human expert, and they are designed to solve complex problems by reasoning. Expert system applications include, among others, medicine [29, 53, 60], diagnosis and control of power systems [26, 27], evaluation of journal grades [61], information systems investment evaluation [19], transport management [31, 51], industry [14, 38] and sport [12, 15, 32, 33, 39].

Nowadays, in sports science various types of computer tools and methods play an important role. Competitors and coaches are looking for new solutions that can support their work. One aspect of such support can be the application of machine learning methods, which can be used to calculate performance results [13, 15, 43], identify sporting talent [35, 39, 48, 49] or support the training process [30, 40, 41, 45, 50, 52].

For example, in the paper [13], the authors use artificial neural networks to predict competitive performance in swimming. The neural models were cross-validated, and the results show that the modelling was very precise. The paper [43] describes the use of linear and nonlinear multivariable models as tools to predict the results of 400 m hurdles races. All the models were constructed using the training data of 21 athletes from the Polish National Team.

✉ Tomasz Krzeszowski
tkrzeszo@prz.edu.pl

Krzysztof Przednowek
krzprz@ur.edu.pl

Krzysztof Wiktorowicz
kwiktor@prz.edu.pl

Janusz Iskra
j.iskra@awf.katowice.pl

[1] Faculty of Physical Education, University of Rzeszow, ul. Towarnickiego 3, 35-959 Rzeszow, Poland

[2] Faculty of Electrical and Computer Engineering, Rzeszow University of Technology, al. Powstanców Warszawy 12, 35-959 Rzeszow, Poland

[3] Faculty of Physical Education and Physiotherapy, Opole University of Technology, ul. Prószkowska 76, 45-758 Opole, Poland

The best prediction results were obtained by the LASSO regression method. Gu et al. [15] proposed an expert system to predict National Hockey League (NHL) game outcome. The prediction accuracy of the system was 77.5%. Another paper [17] presents a review of data mining techniques that are used for prediction in various sports disciplines.

Roczniok et al. [48] proposed using Kohonen's neural networks for the recruitment process in competitive swimming. Experiments were conducted on a group of 140 young swimming contestants aged about 10. Another approach to identifying sporting talent was proposed by Rogulj et al. [49]. The authors have developed two methodological approaches to recognize an athlete's morphological compatibility for various sports. In the paper [35], Maszczyk et al. determined the usefulness of neural models in optimizing recruitment processes. Statistical analyses were carried out on the measured results of javelin throwers using full take off. For the investigated group, the perceptron network with the 4–3–2–1 structure achieved the best predictive results.

In the paper [50], Ryguła et al. proposed using an artificial neural network (ANN) to model swimming performance in the 200 m individual medley and the 400 m front crawl events. The ANNs were also used to analyze tactics in team sports [41]. Another study was devoted to the use of ANNs to classify kick techniques [30]. The aim of that paper was to find out whether it is possible to distinguish two different kick techniques from a kick impact force profile. The paper [52] presents the application of a neural network to model swimming performance. The authors created highly realistic models of swimming performance prediction based on previously selected criteria that were related to the dependent variable. Experiments were conducted on 138 swimmers (65 males and 73 females) at national level.

Despite the existing methods to predict and support training, there is lack of tools that could be used by coaches during the training process. Papić et al. [39] developed a fuzzy expert system for scouting and evaluating young sporting talent. A similar system is presented in [33], where the authors perform talent identification in soccer using a web-oriented expert system.

From the review of literature, it can be seen that there is a need to create tools for supporting sports training. The main contribution of this paper is, therefore, to develop a web-oriented expert system, named iHurdling, to predict results and generate training loads in the 110 and 400 m hurdles. The system we have developed can support a coach in planning training programmes in hurdles races. The system uses linear regression models (OLS, ridge, LASSO, elastic net) and nonlinear models (RBF, fuzzy model, OLS with fuzzy correction). The main advantages of this system are an easy-to-use interface and compatibility with different platforms which means that it can be run from a computer or a mobile device.

2 Training data

The training data contain training plans carried out by hurdlers in the Polish National Team. One record contains the parameters of an athlete and the training programme carried out by this athlete during their annual training cycle. The models for result prediction (PR) and for generating training loads (GT) were build using 21 variables (Table 1). For the PR models, the input variables $x_1 - x_5$ represent the parameters of the athlete, the input variables $x_6 - x_{20}$ represent the training loads and the output variable y represents the predicted result. For the GT models, $x_1 - x_6$ represent the parameters of the athlete and $y_1 - y_{15}$ represent the training loads. The training programs were recorded according to the classification proposed in [22]. The classification consists of two areas of influence: energy (exercise) and information (related to the formation of technique). In the analyzed training loads, there are speed, endurance and strength as well as exercises that develop the technique of hurdles clearance. A similar classification of exercises can be found in another papers devoted to sprinters and hurdlers [2, 37]. The values of these loads are the sum of all loads of the same type realized during the annual training cycle. The results for the hurdles races were registered before and after the cycle. Both runs were carried out under simulated starting conditions of the 110 and 400 m hurdle race. In this study, the current result at the training distance was assumed as the indicator of performance level. As concluded in the paper [25], this result is strongly correlated with performance parameters and other motor skills tests used in hurdles races. For the 110 m hurdles, the training data contain 40 records. These records were collected from 18 highly trained athletes (mean result in 110 m hurdles: 14.02 s) aged 18–28. In 400 m hurdles, the 48 records from 21 athletes aged 19–27 were used. The hurdlers practising the 400 m had also a high sport level. (Mean result on 400 m hurdles was equal to 51.26 s.)

3 Mathematical models

In this paper, we use the regression methods for building multi-input, single-output (MISO) and multi-input, multi-output (MIMO) models. The MISO models are used for the prediction of result, while the MIMO models are used in the generation of training loads. In the simplified description that follows we assume that we have one output, since

Table 1 Description of variables used to construct the PR and GT modules for 110 and 400 m hurdles

PR	GT	Description	110 m			400 m		
			$\bar{x}$	$x_{\min}$	$x_{\max}$	$\bar{x}$	$x_{\min}$	$x_{\max}$
y	x_1	Result after training (s)	14.02	13.26	15.13	51.27	48.19	53.60
x_1	x_2	Age (years)	21.9	18.0	28.0	22.3	19.0	27.0
x_2	x_3	Body height (cm)	187.3	181.0	195.0	185.0	177.0	192.0
x_3	x_4	Body mass (kg)	77.8	71.0	83.0	74.3	69.0	82.0
x_4	x_5	Body mass index	22.1	20.3	23.5	21.2	19.7	24.4
x_5	x_6	Result before training (s)	14.33	13.34	15.40	51.91	48.70	54.70
x_6	y_1	Maximal and technical speed exercises (m)	12,513	5800	17,970	9428	2910	18,920
x_7	y_2	Technical and speed exercises (m)	5925	2470	10,200	4253	240	9450
x_8	y_3	Speed and specific hurdle endurance exercises (m)	11,961	3150	20,400	25,342	6400	101,450
x_9	y_4	Pace runs exercises (m)	64,087	25,780	100,300	163,796	88,000	393,800
x_{10}	y_5	Aerobic endurance exercises (m)	328,631	80,600	550,000	363,257	151,000	692,500
x_{11}	y_6	Strength endurance exercises (m)	20,638	1850	46,595	41,069	1750	169,265
x_{12}	y_7	Strength of lower limbs exercises (kg)	291,119	96,400	658,600	224,099	96,900	504,540
x_{13}	y_8	Trunk strength exercises (amount)	38,442	5240	145,000	46,438	6100	233,680
x_{14}	y_9	Upper body strength exercises (kg)	3352	1630	4850	3305	760.0	29,610
x_{15}	y_{10}	Explosive strength of lower limbs exercises (amount)	1244	0	2214	823	282	2138
x_{16}	y_{11}	Explosive strength of upper limbs exercises (amount)	656	213	1850	443	60	1360
x_{17}	y_{12}	Technical exercises – walking pace (min)	456	130	1110	424	45	816
x_{18}	y_{13}	Technical exercises – running pace (min)	574	195	1450	518	150	1500
x_{19}	y_{14}	Runs over hurdles exercises (amount)	778	362	1317	416	121	775
x_{20}	y_{15}	Hurdle runs in varied rhythm exercises (amount)	1077	320	1850	857	36	1680

a MIMO model will be represented as a set of MISO models.

In our expert system, we use:

- linear models in the form of *ordinary least squares* (OLS) [7], *ridge regression* (RIDGE) [18], *least absolute shrinkage and selection operator* (LASSO) [54] and *elastic net* (ENET) [62],
- nonlinear models in the form of *radial basis function network* (RBF) [6] and *fuzzy rule-based system* (FRBS) [47].

3.1 Linear models

Consider a MISO model with p inputs (predictors) creating the vector $\mathbf{x} = [x_1, x_2, \ldots, x_p]$ and one output (response) y. The goal is to build the regression function

$$y = f(\mathbf{x}) = \sum_{j=1}^{p} x_j w_j \tag{1}$$

based on a data set containing n observations in the form of pairs $(\mathbf{x}_i, y_i)$, where $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{ip}]$, $i = 1, \ldots, n$. The element x_{ij} denotes the jth predictor in the ith observation, and y_i is the response in the ith observation.

The linear regression problem can be written as a matrix equation of the form

$$\mathbf{y} = \mathbf{X}\mathbf{w} \tag{2}$$

where

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix} \tag{3}$$

and $\mathbf{w} = [w_1, w_2, \ldots, w_p]^T$, $\mathbf{y} = [y_1, y_2, \ldots, y_n]^T$. Denoting by $J(\mathbf{w}, \cdot)$ a *cost function*, the problem of finding a linear model involves minimizing the function $J(\mathbf{w}, \cdot)$, that is

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} J(\mathbf{w}, \cdot) \tag{4}$$

where $\hat{\mathbf{w}}$ is the vector of the optimal parameter values. For the linear models, the cost functions have the form of

$$J_{\mathrm{OLS}}(\mathbf{w}) = \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 \tag{5}$$

$$J_{\mathrm{RIDGE}}(\mathbf{w},\lambda) = \|\mathbf{y}-\mathbf{X}\mathbf{w}\|_2^2 + \lambda\|\mathbf{w}\|_2^2 \tag{6}$$

$$J_{\mathrm{LASSO}}(\mathbf{w},\lambda) = \|\mathbf{y}-\mathbf{X}\mathbf{w}\|_2^2 + \lambda\|\mathbf{w}\|_1 \tag{7}$$

$$J_{\mathrm{ENET}}(\mathbf{w},\lambda_1,\lambda_2) = \|\mathbf{y}-\mathbf{X}\mathbf{w}\|_2^2 + \lambda_1\|\mathbf{w}\|_1 + \lambda_2\|\mathbf{w}\|_2^2 \tag{8}$$

where λ, λ_1 and λ_2 are non-negative regularization parameters. The norms $||\cdot||_2$ and $||\cdot||_1$ denote the Euclidean and the Manhattan norms, respectively. The RIDGE, LASSO and ENET regressions are regularized which means that they can be used when the problem is ill-conditioned. The detailed description of the linear models can be found, for example, in [58].

3.2 Choosing the best model

All models were tested using cross-validation method. This is a method of evaluating the generalization ability (prediction for new data, not involved in modelling) of the model being created. In cross-validation, data are divided into two subsets: a training set and a testing (validation) set. In this study, due to the small amount of data ($n = 40$ for 110 m and $n = 48$ for 400 m), LOOCV (*leave-one-out cross-validation*) was used [3]. The idea of this method is to extract from the set of data n learning subsets. Each subset is created by removing only one pair from the data set, which becomes a test pair. Then, for each resulting subset, the model is constructed that is evaluated by determining the error for the remaining test pair. The predictive ability of a model is expressed by the *root of the mean square error of cross-validation* (RMSE$_{\mathrm{CV}}$) calculated as

$$\mathrm{RMSE}_{\mathrm{CV}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_{-i})^2} \tag{9}$$

where $\hat{y}_{-i}$ is the output of a model obtained after removing the pair $(\mathbf{x}_i, y_i)$ from the data set.

3.3 Nonlinear models

3.3.1 RBF models

An RBF network is a feed-forward network that typically consists of three layers: an input layer, a hidden layer and an output layer.

The input layer is composed of nodes that receive input signal $\mathbf{x}$, and there is one node for each predictor variable. The hidden layer is composed of nodes with radially symmetric activation functions. The hidden node measures the distance between the input vector $\mathbf{x}$ and the centre $\mathbf{c}_k$ of its radial function:

$$\varphi_k(\mathbf{x}) = \varphi_k(\|\mathbf{x}-\mathbf{c}_k\|) \tag{10}$$

The norm $||\cdot||$ is usually taken as the Euclidean distance, and $\varphi(\mathbf{x})$ is typically taken to be the Gaussian function. The output layer is composed of a node that receives the outputs of nodes in the hidden layer. This node calculates the output of the network as a linear combination of nonlinear functions of the form

$$y = \sum_{k=1}^{m}\varphi_k(\mathbf{x})w_k \tag{11}$$

where m is the number of nodes in the hidden layer, $\varphi_k(\mathbf{x})$ is a basis function and w_k is the weight of the kth neuron in the output node.

The training of the RBF network involves: the number of hidden neurons, the parameters of radial functions in the hidden layer and the weights in the output layer.

3.3.2 Fuzzy models

In this paper, we propose two approaches to use the FRBS [47] in regression problems. In the first approach, the fuzzy model is build similarly to the RBF model, that is, it is learned from the original data. (This model is called FUZZY.) In the second approach, the FRBS is used for the nonlinear correction of the OLS model. (This model is called F-OLS.) The idea is to change the output of a linear model by adding a nonlinear correction term, in such a way that the predictive error is reduced (Fig. 1). First, we build the OLS model and remember its cross-validation errors, and next we build a fuzzy model that "learns" these errors.

The design procedure for building F-OLS models is listed below.

Step 1. Cross-validation of the OLS model $y = f_{\mathrm{OLS}}(\mathbf{x})$ for the data $(\mathbf{x}_i, y_i)$. In the ith step of cross-validation, the error has the form

$$e_i = y_i - y_{-i} \tag{12}$$

where $y_{-i} = f_{\mathrm{OLS}}(\mathbf{x}_{-i})$.

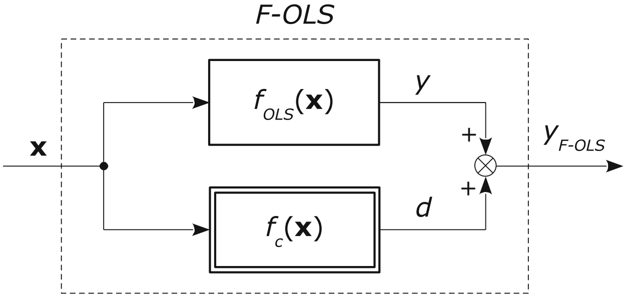


Fig. 1 The idea of calculating the output of the F-OLS model. The variable $y = f_{\mathrm{OLS}}(\mathbf{x})$ is the output of the ordinary least squares estimator, and $d = f_c(\mathbf{x})$ is the output of the fuzzy nonlinear corrector

Step 2. Constructing the fuzzy (nonlinear) corrector

$$d = f_c(\mathbf{x}) \qquad (13)$$

for the data $(\mathbf{x}_i, e_i)$. This corrector predicts the errors obtained in Step 1. The best fuzzy model can be chosen on the basis of cross-validation conducted for different number of fuzzy sets.

Step 3. Cross-validation of the OLS model with the corrected error in the form

$$e_i^{new} = y_i - (y_{-i} + d_i) \qquad (14)$$

where $y_{-i} = f_{\mathrm{OLS}}(\mathbf{x}_{-i})$ and $d_i = f_c(\mathbf{x}_i)$.

Step 4. The predicted output of the F-OLS model is determined as

$$y_{\mathrm{F-OLS}} = f_{\mathrm{OLS}}(\mathbf{x}) + f_c(\mathbf{x}) \qquad (15)$$

where $f_c(\mathbf{x})$ is the function of the fuzzy corrector chosen in Step 3 (Fig. 1).

4 Expert system modules

The expert system consists of two modules, the prediction of result (PR) and the generation of training loads (GT) for the 110 and 400 m distances. The regression models for both modules were calculated in R language [46]. The functions with arguments used to generate the models are shown in Table 2, and they are described below.

The function `lm` was used to calculate the OLS, and the ridge regressions were calculated using the function `lm.ridge` from the "MASS" package [55] (with $\lambda > 0$ in 6). The LASSO and the elastic net regressions were obtained with the function `enet` included in the "elasticnet" package [63]. This function has two parameters (λ, s), where $\lambda \geq 0$ denotes λ_2 in the formula (8) and $s \in [0, 1]$ is a fraction of the norm L_1. The pair (λ, s) is used instead of the pair (λ_1, λ_2) in the formula (8) because the elastic net regression can be treated as the LASSO regression for an augmented data set [62]. Taking $\lambda = 0$ we get the LASSO regression with one parameter s for the original data. The ENET models were selected by searching the parameters λ and s.

This study uses artificial neural networks in the form of the radial basis function (RBF). The training data were scaled before the RBF training, and the results of the predictions were unscaled. All the analyzed networks have one hidden layer. For the implementation of neural networks, the function `RSNNS::rbf` was used [6]. The optimal neural model was determined by searching a number of hidden neurons in the range from 2 to 10.

Table 2 R functions for models training

Model	Function	Arguments
OLS	`lm`	`formula` = y ∼ · `data.train`
RIDGE	`lm.ridge`	`formula` = y ∼ · `data.train` `lambda` $\in [0, \infty)$

LASSO	`enet`	`data.x, data.y`
		`lambda = 0,`
		`normalize = TRUE`
		`intercept = TRUE`

	`predict.enet`	`model`
		`newx`
		`s` $\in [0, 1]$

ENET	`enet`	`data.x, data.y`
		`lambda` $\in [0, \infty)$,
		`normalize = TRUE`
		`intercept = TRUE`

	`predict.enet`	`model`
		`newx`
		`s` $\in [0, 1]$

RBF	`rbf`	`data.x, data.y`
		`size` $\in [1, 10]$
		`maxit = 1000`
		`linOut = TRUE`

Fuzzy	`frbs.learn`	`data.train`
		`range.data`
		`method.type = ''WM''`
		`control = list(`
		`num.labels = 1,`
		`type.mf = ''GAUSSIAN'',`
		`type.defuz = ''WAM'',`
		`type.tnorm = ''MIN'',`
		`type.implication.func`
		`= ''MIN'')`

The function `predict.enet` is used for selecting the parameter s for LASSO and ENET models

The fuzzy models were calculated using the function `frbs.learn` from the "frbs" package [47]. The learning method was the Wang–Mendel (W–M) algorithm [56]. This algorithm generates fuzzy rules from input–output data pairs. The input space is divided into fuzzy subspaces, and fuzzy rules are extracted for each subspace. The W–M method is a one-pass procedure and does not need time-consuming training. In the fuzzy model, the Gaussian membership functions are used, the t-norm is “minimum”, the defuzzification is “weighted average method”, and the implication is “minimum”. The number of fuzzy sets l was determined by calculating cross-validation errors as described in Sect. 3.3.2.

4.1 Models for result prediction

The cross-validation errors $\mathrm{RMSE}_{\mathrm{CV}}$ and parameters of the models for the PR module are presented in Table 3. The parameters were chosen on the basis of the plots shown in Figs. 2 and 3. In the case of the ridge regression, the regularization parameter $\lambda \in [0, 40]$ was considered with the step 0.1 for both distances. In the case of the LASSO regression, the parameter $s \in [0, 1]$ was considered with the step 0.01. For the ENET regression, the following parameters were chosen: $\lambda \in [0.1, 0.25]$ with the step 0.008 and $s \in [0.4, 0.8]$ with the step 0.021 for the distance of 110 m, and $\lambda \in [0, 0.06]$ with the step 0.0032 and $s \in [0.3, 0.5]$ with the step 0.01 for the distance of 400 m. The RBF model was analyzed for the number of neurons in the hidden layer $m \in \{2, 3, \ldots, 10\}$, and the fuzzy models were analyzed for the number of fuzzy sets $l \in \{2, 3, \ldots, 13\}$. Based on the conducted analysis, the best models (models with the smallest cross-validation error) were selected (Table 3). It can be seen that for both the 110 and the 400 m distances, the lowest error was obtained by the F-OLS regression. The best F-OLS models have eight fuzzy sets for 110 m and nine sets for 400 m. The largest error for the 110 m was obtained by the OLS regression and by the FUZZY model for the 400 m.

4.2 Models for generation of training loads

For the GT module, each output of the model (y_1-y_{15}) was considered and analyzed in a similar way as for the result prediction module. The errors $\mathrm{RMSE}_{\mathrm{CV}}$ for the GT module are presented in Table 4, while the parameters of the models are presented in Table 5. The models in the GT module were cross-validated similarly to the PR module. For example, for the output y_{14} the FUZZY model has the largest errors (200.1 for 110 m and 132.9 for 400 m), and the F-OLS model has the smallest errors (33.18 for 110 m and 105.1 for 400 m). From Table 4, it can be observed that the smallest $\mathrm{RMSE}_{\mathrm{CV}}$ for all outputs has the F-OLS model.

5 Graphical user interface

The graphical user interface was implemented in R language using the `shiny` [11], `shinyjs` [4], `shinythemes` [10], `shinydashboard` [9] and `rmarkdown` [1] libraries. This interface is a web-oriented application and therefore requires only a web browser and an Internet connection to be used. The current version of the developed system is available on https://hurdles.shinyapps.io/ihurdling. The application shown in Fig. 4 consists of three panels labelled “Result prediction”, “Generation of training loads” and “Athletes’ database”.

On the left side of window is a sidebar menu with links to each panel. The radio button in this sidebar is used to select the PR or GT module. Moreover, the user can choose one of the developed regression models and generate reports. The footer contains the information about the application and the authors.

Table 3 Errors and parameters for the PR module for 110 and 400 m hurdles

	OLS	RIDGE	LASSO	ENET	RBF	FUZZY	F-OLS
110 m							
$\mathrm{RMSE}_{\mathrm{CV}}$	0.3807	0.2276	0.2397	0.1996	0.1985	0.2572	0.0851
Param.	—	$\lambda = 16.1$	$\lambda = 0$ $s = 0.04$	$\lambda = 0.16$ $s = 0.56$	$m = 8$	$l = 4$	$l = 8$
400 m							
$\mathrm{RMSE}_{\mathrm{CV}}$	0.6959	0.5743	0.4463	0.4308	0.6953	0.9288	0.1533
Param.	—	$\lambda = 11.6$	$\lambda = 0$ $s = 0.28$	$\lambda = 0.01$ $s = 0.36$	$m = 6$	$l = 3$	$l = 9$

The meaning of the parameters is as follows: λ and s are tuning parameters for regularized models, m is the number of hidden neurons in the RBF network, l is the number of fuzzy sets in fuzzy models

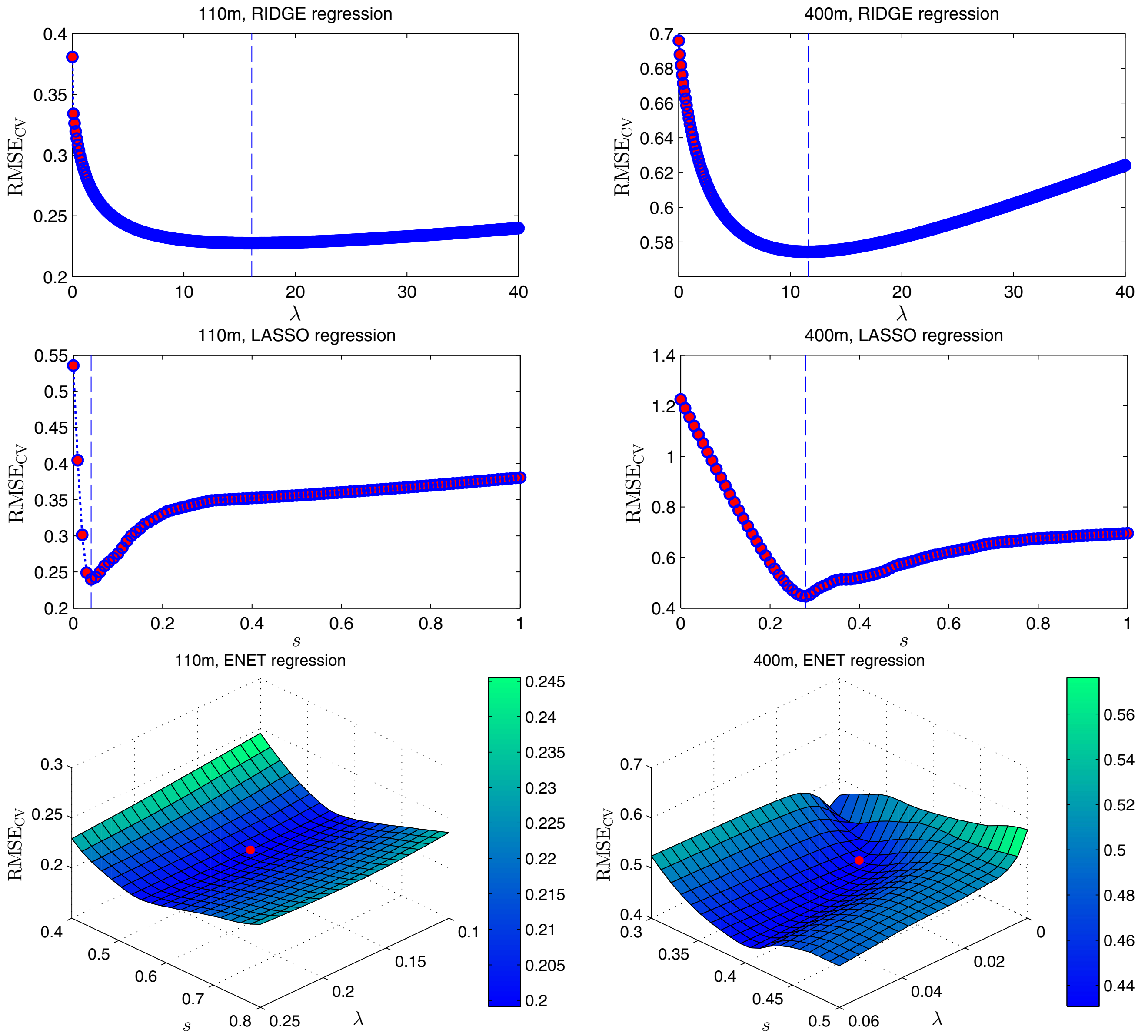


Fig. 2 Cross-validation errors for linear models (RIDGE, LASSO, ENET) in result prediction

5.1 Panel for result prediction

The "Result prediction" panel is used for entering data and for result prediction (Fig. 4). The input variables are grouped into five boxes: "Athlete's parameters", "Training loads—endurance", "Training loads—technique and rhythm", "Training loads—strength" and "Training loads—speed". The value of each input can be modified using appropriately scaled sliders. For example, the box "Training loads—endurance" presented in Fig. 5 has four sliders for changing endurance training loads. Each slider has a range determined on the basis of the minimum and maximum values in the database (Table 1) and depends on the distance of the hurdles race. For instance, the slider "Pace runs" for the 110 m hurdles has a range from 25,000 to 101,000 m with the step equal to one metre.

In the last box labelled "Results" two `textOutput` fields display the current and predicted results. Prediction of the result is performed automatically after changing the position of any slider. Moreover, the result depends on the radio button that selects the method in the sidebar menu. In this way, the user can modify the training loads and

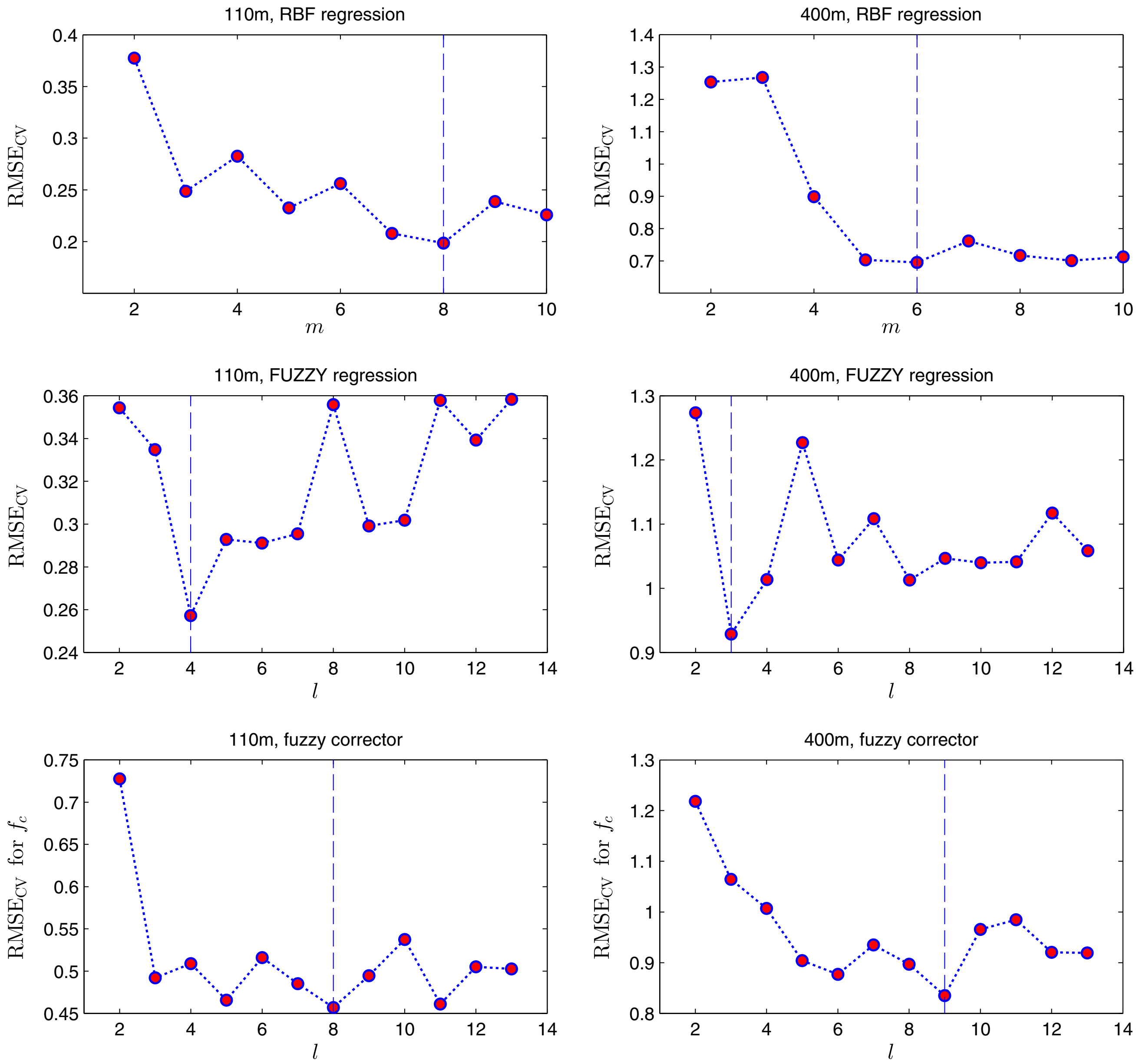


Fig. 3 Cross-validation errors for nonlinear models (RBF, FUZZY, f_c) in result prediction

observe the changes that occur in the expected result. Generating a report from result prediction creates a .pdf file, which contains the values from all sliders and the predicted result.

5.2 Panel for generation of training loads

Another system panel is the generation of training loads for both hurdles distances (Figs. 6, 7). This module consists of two boxes: “Athletes’ parameters” and “Generated training—annual cycle”. The first box is used to enter the athlete’s data, i.e. age, body height, body weight and his current result. This box also includes an option to choose the training generation mode. The user can choose the option of one training generation or the option to generate the training loads for a longer period of his career. The selection of the first option will cause a slider with the expected result to appear under the slider with the current result. If the “career” option is selected, these sliders are not available. The “career” option makes it possible to

Table 4 Errors for the GT module for 110 and 400 m hurdles

Output	OLS	RIDGE	LASSO	ENET	RBF	FUZZY	F-OLS
110 m							
y_1	2550	2429	2304	2304	2219	2572	1829
y_2	1700	1632	1617	1606	1528	1675	244.0
y_3	4351	3985	3933	3814	3843	3629	2630
y_4	24,040	21,170	21,270	21,120	20,310	20,210	6254
y_5	112,000	110,900	106,600	106,600	80,681	88,870	70,780
y_6	12,570	11,240	11,170	11,170	10,693	9644	2546
y_7	152,100	129,600	140,200	129,600	131,123	112,200	21,270
y_8	30,190	26,730	27,970	26,360	25,366	22,570	3410
y_9	654.1	610.9	606.1	606.1	632.8	622.3	519.2
y_{10}	579.7	484.0	482.4	482.4	398.3	327.9	155.2
y_{11}	267.4	244.7	247.5	239.7	191.5	236.7	143.1
y_{12}	302.1	263.1	263.4	259.4	248.9	266.8	98.17
y_{13}	385.5	322.8	320.5	320.5	310.7	256.1	39.93
y_{14}	190.9	179.7	188.8	186.3	167.7	200.1	33.18
y_{15}	436.0	401.5	400.7	391.2	399.1	434.0	205.6
400 m							
y_1	4477	4189	4110	4109	3038	2511	597.7
y_2	1864	1778	1755	1754	1422	1460	855.9
y_3	15,120	14,270	14,190	14,190	14,308	16,670	3473
y_4	62,580	60,770	60,040	60,040	63,378	52,860	11,110
y_5	107,100	98,490	98,120	98,120	98,906	110,600	39,340
y_6	24,777	23,440	23,380	23,270	25,972	26,070	3261
y_7	92,100	88,030	84,830	84,830	81,357	73,900	21,150
y_8	46,360	44,810	44,510	44,510	44,399	47,740	6754
y_9	4581	4190	4188	4185	4274	4139	1481
y_{10}	369.4	350.0	350.2	350.2	365.8	428.1	61.88
y_{11}	287.5	275.8	276.6	275.6	290.9	223.3	37.23
y_{12}	265.8	237.6	235.8	235.8	239.3	187.3	108.3
y_{13}	285.2	273.6	271.2	271.2	282.1	284.7	175.0
y_{14}	127.7	124.6	122.8	122.8	115.6	132.9	105.1
y_{15}	369.0	362.4	356.3	356.3	314.0	301.2	111.4

generate six training programmes which are consecutive and improve the result by 0.25 s each year (from 15.00 to 13.50 s) for 110 m hurdles and by 1 s each year (from 53.00 to 48.00 s) for 400 m hurdles.

The contents of the box "Generated training—annual cycle" change dynamically, depending on the mode. The "one training" option will generate a list of training loads with suggested values (Fig. 6). In addition, a graph is generated, in which the values of training loads, expressed as a percentage of the maximum value of the given output, are presented. The second option is "career"; its selection generates a table containing six annual training plans and 15 graphs showing the loads over the athlete's entire career (Fig. 7). The "career" is an additional option that allows us to generate training loads for six consecutive years. In this option, the starting result is always constant and is 14.75 s for 100 m and 53 s for 400 m, respectively. Results are generated in the form of a table where each row represents the annual training and in the form of graphs where the *x*-axis is the expected result and the *y*-axis is the value of the training load. Career graphs allow observations of changes in individual loads in terms of a 6-year career. The coach can observe which load needs to be increased, which decreased and which should stay at the same level.

Generating a report from the "Generated training" panel creates a .pdf file, which contains values from the "Ath-

Table 5 Parameters for the GT module for 110 and 400 m hurdles. The meaning of the parameters is as follows: λ and s are tuning parameters for regularized models, m is the number of hidden neurons in the RBF network, l is the number of fuzzy sets in fuzzy models

Output	RIDGE λ	LASSO s	ENET (λ, s)	RBF m	FUZZY l	F-OLS l
110 m						
y_1	15.9	0.06	(0, 0.06)	4	5	3
y_2	15.3	0.03	(6.03, 0.40)	2	7	10
y_3	32.5	0.01	(1.80, 0.41)	5	4	3
y_4	47.5	0.01	(0.87, 0.59)	7	7	7
y_5	4.38	0.01	(0, 0.01)	9	5	3
y_6	500	0	(0, 0)	9	8	7
y_7	39.6	0.04	(0.65, 0.56)	2	6	10
y_8	16.6	0.05	(1.30, 0.54)	10	12	12
y_9	500	0	(0, 0)	2	13	3
y_{10}	500	0	(0, 0)	9	5	5
y_{11}	15.0	0.06	(1.20, 0.47)	10	4	4
y_{12}	256	0	(49.0, 0.17)	2	3	6
y_{13}	294	0.01	(0, 0.01)	7	10	13
y_{14}	10.7	0.48	(19.3, 0.38)	5	4	10
y_{15}	500	0	(0.01, 0.16)	5	3	4
400 m						
y_1	500	0.03	(0.06, 0.09)	9	13	9
y_2	500	0.02	(0.02, 0.03)	2	3	5
y_3	288	0.05	(0, 0.05)	2	10	10
y_4	500	0.04	(0, 0.04)	2	9	9
y_5	500	0.01	(0.01, 0.01)	2	10	6
y_6	30.7	0.51	(0.04, 0.72)	2	13	13
y_7	19.5	0.33	(0, 0.33)	9	13	8
y_8	156	0.15	(0, 0.15)	2	11	11
y_9	322	0.02	(0.02, 0.05)	2	11	7
y_{10}	88.8	0.08	(0, 0.08)	3	10	11
y_{11}	72.6	0.06	(0.01, 0.12)	4	6	13
y_{12}	500	0	(0, 0)	2	11	5
y_{13}	339	0.02	(0, 0.02)	3	2	5
y_{14}	10.5	0.21	(0, 0.21)	7	2	2
y_{15}	140	0.04	(0, 0.04)	7	5	6

lete's parameters" box and a table with one or six annual training cycles depending on the types of generating training loads.

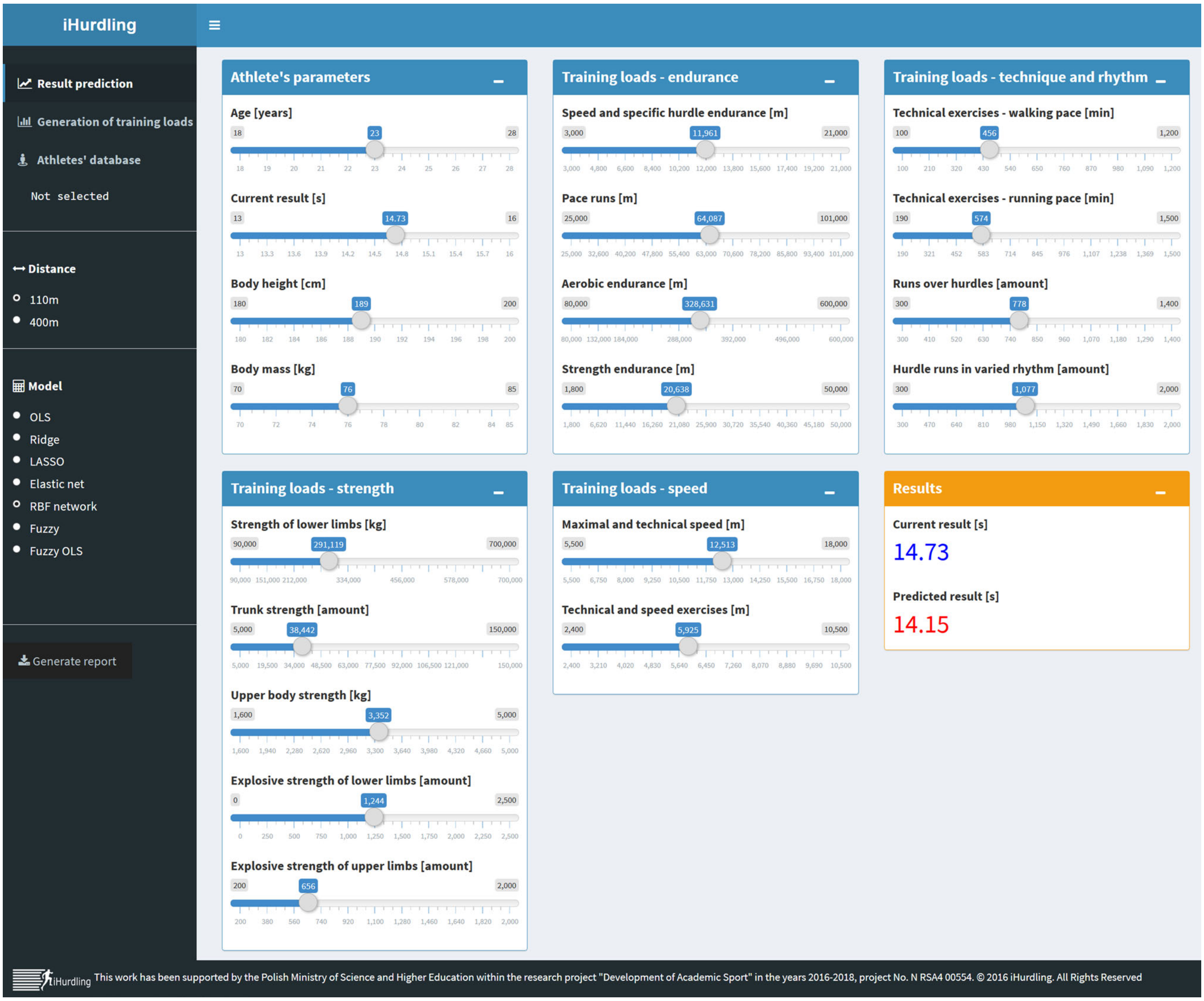
5.3 Panel for athletes' database

The third system panel is used to create and change the database containing athletes' details (Fig. 8). This panel consists of two boxes: "Athlete's database" and "Edit". In the first box, the records of the database loaded from the file are displayed. The system supports files saved in the .csv format with field separator ";" and "." as the decimal point. The database file contains the following columns: "Name", "Surname", "Age", "Body Mass" and "Body Height". This box displays all athletes in a table; the choice of athlete is done by marking the appropriate line in this table. Furthermore, the name of the selected athlete is displayed in the sidebar. When an athlete is selected, his data can be edited via the "Edit" box. The saving of the edition is approved with the "Save" button. The "Delete" button removes the athlete from the database. The deselection of the athlete is done by re-selecting him/her in the database. If no athlete is selected, a new athlete can be entered into the database using the "Edit" window. After a new athlete is entered, you should click "Save as new". After each operation performed on the database, the user should save the database using the button "Save database" on the first panel. The second button on the panel ("Clear database") performs cleaning the database from the application memory. In the "Athletes' database" panel, it is not possible to generate reports.

6 Discussion

In this paper, mathematical models for generation of training loads and prediction of results expected from athletes training the 110 and 400 m hurdle races were presented. The best model verified by LOOCV in each of the considered tasks and for each distance turned out to be the model F-OLS proposed by the authors. The application of fuzzy models in sport was also presented by Mezyk and Unold [36]. The goal was to find the rules that can express swimmers feelings the day after in-water training. Their method was characterized by better predictive ability than the traditional methods of classification, and the effectiveness was at the level of 68.66%. In Papić et al. [39], the fuzzy expert system was also presented. This system was based on knowledge of experts in the field of sport, as well as the data obtained as a result of motor tests. The model suggested the most suitable sport, and it was designed to search for prospective sports talents. Evaluation of the system showed high reliability and high correlation with top experts in the field.

While analyzing the literature, it can be also noticed that mathematical models frequently used in sports are artificial neural networks [34, 35, 40, 44, 48, 50, 52, 58]. Numerous studies have shown that the ANN is a means of predicting sports results which has a good predictive ability [13, 59]. Thus, the ANN enables a coach to model the future level of athletes performance and supports the process of sports



Not selected

Distance
○ 110m
● 400m

Model
● OLS
● Ridge
● LASSO
● Elastic net
○ RBF network
● Fuzzy
● Fuzzy OLS

Athlete's parameters

Age [years] 18 23 28

Current result [s] 13 14.73 16

Body height [cm] 180 189 200

Body mass [kg] 70 76 85

Training loads - endurance

Speed and specific hurdle endurance [m] 3,000 11,961 21,000

Pace runs [m] 25,000 64,087 101,000

Aerobic endurance [m] 80,000 328,631 600,000

Strength endurance [m] 1,800 20,638 50,000

Training loads - technique and rhythm

Technical exercises - walking pace [min] 100 456 1,200

Technical exercises - running pace [min] 190 574 1,500

Runs over hurdles [amount] 300 778 1,400

Hurdle runs in varied rhythm [amount] 300 1,077 2,000

Training loads - strength

Strength of lower limbs [kg] 90,000 291,119 700,000

Trunk strength [amount] 5,000 38,442 150,000

Upper body strength [kg] 1,600 3,352 5,000

Explosive strength of lower limbs [amount] 0 1,244 2,500

Explosive strength of upper limbs [amount] 200 656 2,000

Training loads - speed

Maximal and technical speed [m] 5,500 12,513 18,000

Technical and speed exercises [m] 2,400 5,925 10,500

Results

Current result [s]
14.73

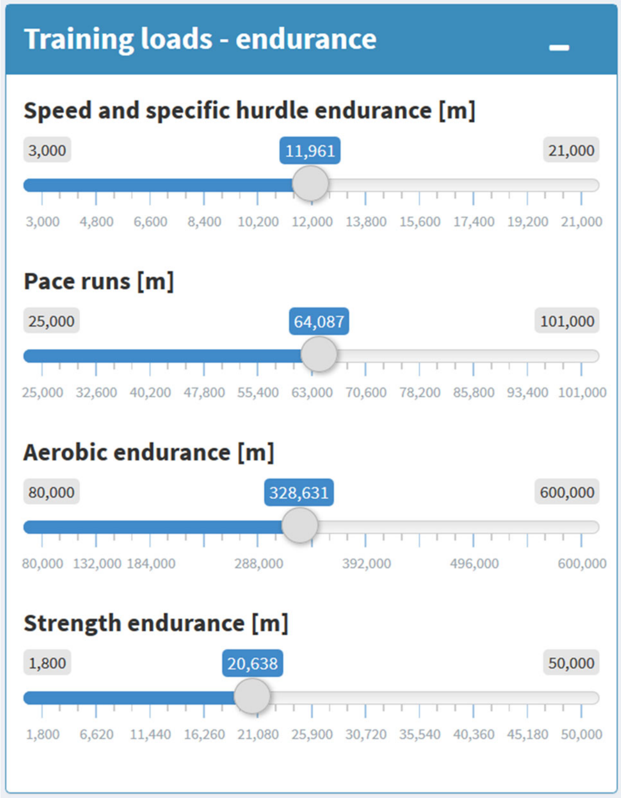
Predicted result [s]
14.15

iHurdling This work has been supported by the Polish Ministry of Science and Higher Education within the research project "Development of Academic Sport" in the years 2016-2018, project No. N RSA4 00554. © 2016 iHurdling. All Rights Reserved

Fig. 4 Screenshot of the iHurdling application with PR panel

selection [34, 40, 48, 52]. For example, Silva et al. [52] presented high realistic models of swimming performance prediction based on multilayer perceptron. To establish a profile of the young swimmer, nonlinear combinations between preponderant variables for each gender and swim performance in the 200 m medley and 400 m front crawl events were developed. Artificial neural networks are also widely used in the process of planning training loads [44, 50]. In [50], Ryguła presents a new approach for determining training loads in a group of 16- and 17-year-old girls practising 100 m run.

Sports training is the matter of making decisions about the quality (type of exercise) and the quantity (volume). This is a classical principle of sports training, emphasized in all textbooks on the theory of sports [8, 42]. Selection of training means and their distribution at subsequent stages of sports training is the main element of hurdlers' training optimization on both distances, i.e. 110 and 400 m [50]. The selection of exercises (training means) in hurdling is supported by research in the field of motor preparation (strength, speed, endurance) as well as in relation to the technical structure of the event (kinematic analyses) [21].


Training loads - endurance

Speed and specific hurdle endurance [m]

3,000 11,961 21,000

Pace runs [m]

25,000 64,087 101,000

Aerobic endurance [m]

80,000 328,631 600,000

Strength endurance [m]

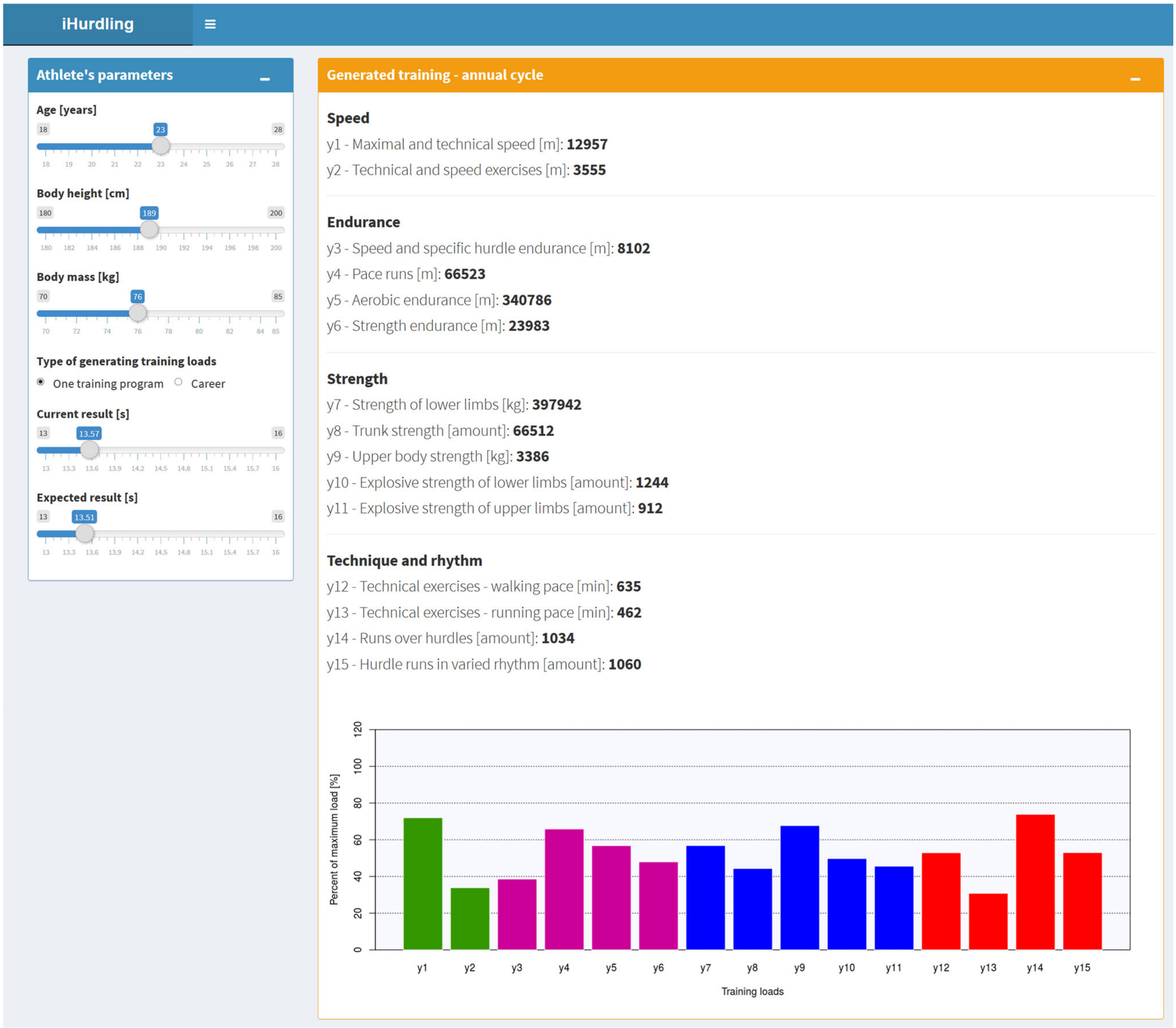
1,800 20,638 50,000

Fig. 5 Screenshot of the box for entering endurance training loads

The observation of training programs of the best athletes [24], supported by the analysis of correlation between the results of hurdle run and the tests results including the physiological [16, 64] and biochemical basis [28], allows for selection of groups of the most valuable basic exercises. The performance tests carried out during the ergometric effort [5] are of great importance in assessing the specifics of hurdlers' effort [5]. It should be emphasized that individualization of hurdlers' training programs also requires an individual approach to the type of capacity, oscillating between aerobic and anaerobic capacity. Sprinting distances in hurdling are considered to be typical running efforts of anaerobic nature. In the case of 110 m hurdle race, anaerobic non-lactic acidic changes with the final accent on anaerobic lactic acidic changes are predominant. The 400 m hurdle run requires first of all an effort of anaerobic non-lactic acidic nature [57]. Data concerning the specificity of the effort at a distance of 400 m indicate that the proportions of aerobic and anaerobic efforts can be significantly varied, taking into account the material (sports performance level of runners), method and period of training. In the review study by Arcelli et al., [2] those parameters adopted values within the range of 28–70% (aerobic) and 30–72% (anaerobic). The authors suggested that the higher the sports performance level, the higher the share of anaerobic element.

The determination of the type of runner due to the aerobic and anaerobic processes would certainly make it possible to introduce some additional information in order to develop individual training. However, this problem has a logistic disadvantage, as monitoring of physiological reactions in hurdling is limited to the months when the athletes take part in competitions. Winter conditions are not conducive to specific running tests, and the choice of substitute distances may negatively affect the individual abilities of the hurdler.

Taking into account the extensive scope of hurdlers' exercises, the basic problem of a coach is the choice of exercises, their volume and proportions during particular training periods. The researchers pay their attention to that specificity of sports [8, 37]. Apart from the representative collection of training means, the body physique and age, often identified with the sports performance level, were also used. The impact of those elements on the organization of training has been already emphasized on several occasions [20, 23]. It seems appropriate to determine the initial value (record from the given year) and the estimated scale of progress (plans for the next season), because it makes it possible to control training loads depending on the athletes age, their current performance level and the main objective. Each athlete has different predispositions, also to perform specific training tasks. The selection of exercises is necessary, because it is impossible to perform the same volume of all exercises at the same time. Such a procedure would also be pointless, since the "rhythmic" type of a hurdler prefers running exercises with hurdles, and the "speedy" type of the hurdle runner prefers shorter distances of the interval nature [20]. The database used is based on the period of 20 years of training Polish hurdlers, members of the national team. Those hurdlers represented various types (somatic, efficient and technical); therefore the scope of generalization (approximation) possibilities of the proposed computer system is significant and partly representative.



Athlete's parameters

Age [years]: 18 – 23 – 28

Body height [cm]: 180 – 189 – 200

Body mass [kg]: 70 – 76 – 85

Type of generating training loads

● One training program ○ Career

Current result [s]: 13 – 13.57 – 16

Expected result [s]: 13 – 13.51 – 16

Generated training - annual cycle

Speed

y1 - Maximal and technical speed [m]: **12957**

y2 - Technical and speed exercises [m]: **3555**

Endurance

y3 - Speed and specific hurdle endurance [m]: **8102**

y4 - Pace runs [m]: **66523**

y5 - Aerobic endurance [m]: **340786**

y6 - Strength endurance [m]: **23983**

Strength

y7 - Strength of lower limbs [kg]: **397942**

y8 - Trunk strength [amount]: **66512**

y9 - Upper body strength [kg]: **3386**

y10 - Explosive strength of lower limbs [amount]: **1244**

y11 - Explosive strength of upper limbs [amount]: **912**

Technique and rhythm

y12 - Technical exercises - walking pace [min]: **635**

y13 - Technical exercises - running pace [min]: **462**

y14 - Runs over hurdles [amount]: **1034**

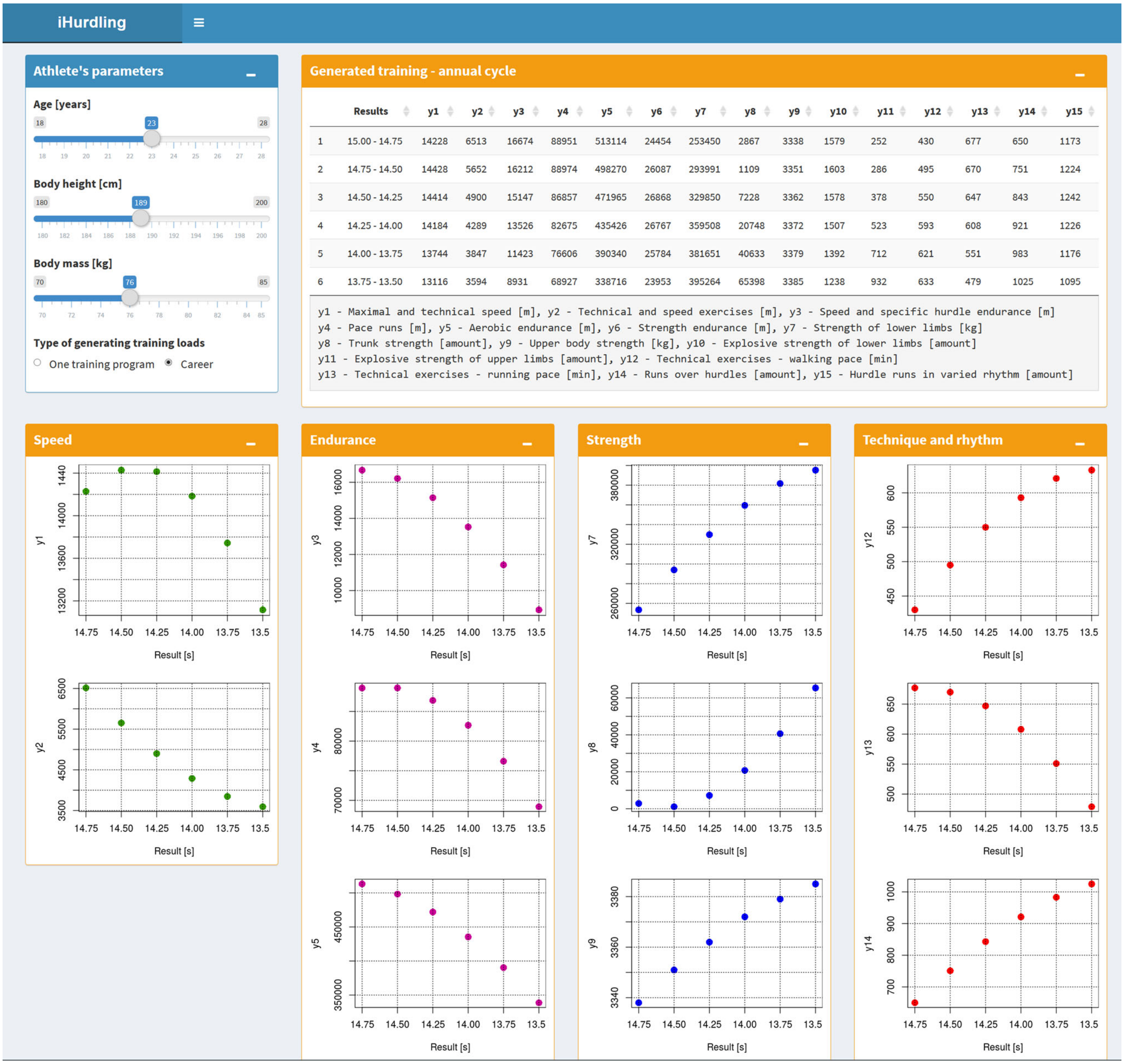
y15 - Hurdle runs in varied rhythm [amount]: **1060**

Fig. 6 Screenshot of the panel for generation of training loads for one training programme

Summarizing the discussion, it should be noted that there are severe limitations of the presented approach connected with using the results in practice. The training programs do not consider the individual physiological and psychological parameters of an athlete. However, the generated training programs might be used as a suggestion for the coach who can perform necessary adjustments in order to adapt them for a particular athlete.

7 Conclusions

In this paper, a web-oriented expert system to predict results and generate training loads for 110 and 400 m hurdles races was presented. The system uses the linear regression models (OLS, ridge, LASSO, elastic net) and nonlinear regression models (RBF, fuzzy model, OLS with fuzzy correction). The lowest errors were obtained by the



iHurdling

Athlete's parameters

Age [years]: 23

Body height [cm]: 189

Body mass [kg]: 76

Type of generating training loads

○ One training program ● Career

Generated training - annual cycle

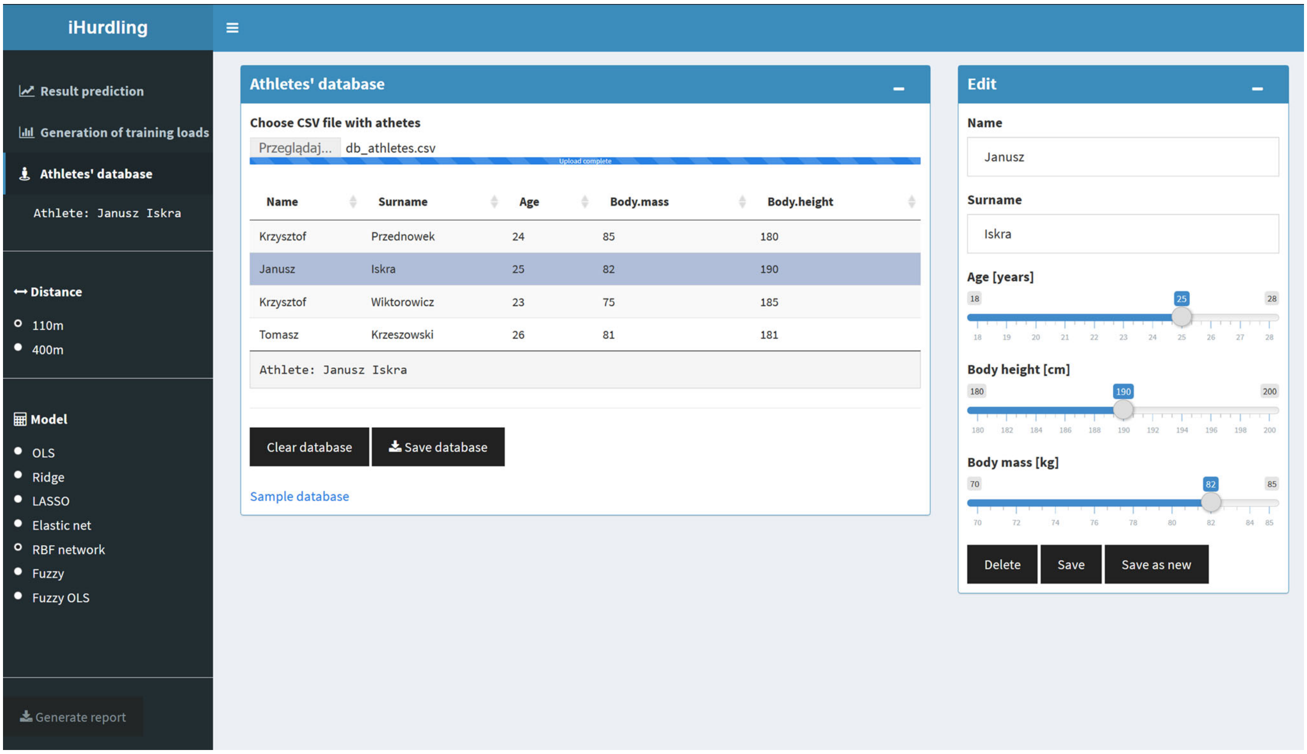
	Results	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
1	15.00 - 14.75	14228	6513	16674	88951	513114	24454	253450	2867	3338	1579	252	430	677	650	1173
2	14.75 - 14.50	14428	5652	16212	88974	498270	26087	293991	1109	3351	1603	286	495	670	751	1224
3	14.50 - 14.25	14414	4900	15147	86857	471965	26868	329850	7228	3362	1578	378	550	647	843	1242
4	14.25 - 14.00	14184	4289	13526	82675	435426	26767	359508	20748	3372	1507	523	593	608	921	1226
5	14.00 - 13.75	13744	3847	11423	76606	390340	25784	381651	40633	3379	1392	712	621	551	983	1176
6	13.75 - 13.50	13116	3594	8931	68927	338716	23953	395264	65398	3385	1238	932	633	479	1025	1095

```
y1 - Maximal and technical speed [m], y2 - Technical and speed exercises [m], y3 - Speed and specific hurdle endurance [m]
y4 - Pace runs [m], y5 - Aerobic endurance [m], y6 - Strength endurance [m], y7 - Strength of lower limbs [kg]
y8 - Trunk strength [amount], y9 - Upper body strength [kg], y10 - Explosive strength of lower limbs [amount]
y11 - Explosive strength of upper limbs [amount], y12 - Technical exercises - walking pace [min]
y13 - Technical exercises - running pace [min], y14 - Runs over hurdles [amount], y15 - Hurdle runs in varied rhythm [amount]
```

Fig. 7 Screenshot of the panel for generation of training loads over the athlete's entire career

proposed F-OLS model, but creating this model is more complicated.

The application was implemented using R programming language with Shiny framework. The advantage of this application is that it can be run on multiple platforms such as personal computers and mobile devices. The easy-to-use interface allows the parameters of an athlete and the training loads to be changed. In this way, the coach can predict the expected result and select individual training components for a given athlete.



```
Athlete: Janusz Iskra
```

Distance
○ 110m
● 400m

Model
● OLS
● Ridge
● LASSO
● Elastic net
○ RBF network
● Fuzzy
● Fuzzy OLS

Choose CSV file with athetes: db_athletes.csv

Name	Surname	Age	Body.mass	Body.height
Krzysztof	Przednowek	24	85	180
Janusz	Iskra	25	82	190
Krzysztof	Wiktorowicz	23	75	185
Tomasz	Krzeszowski	26	81	181

```
Athlete: Janusz Iskra
```

Name: Janusz

Surname: Iskra

Age [years]: 25

Body height [cm]: 190

Body mass [kg]: 82

Fig. 8 Screenshot of the panel for athletes’ database

Further work will focus on migrating the developed expert system to mobile application.

Acknowledgements This work has been supported by the Polish Ministry of Science and Higher Education within the research project “Development of Academic Sport” in the years 2016–2019, Project No. N RSA4 00554.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest

References

1. Allaire J, Cheng J, Xie Y, McPherson J, Chang W, Allen J, Wickham H, Atkins A, Hyndman R (2016) rmarkdown: dynamic Documents for R. https://CRAN.R-project.org/package=rmarkdown. R package version 1.2. Accessed Feb 2017
2. Arcelli E, Mambretti M, Cimadoro G, Alberti G (2008) The aerobic mechanism in the 400 metres. New Stud Athl 23(2):15–23
3. Arlot S, Celisse A (2010) A survey of cross-validation procedures for model selection. Stat Surv 4:40–79
4. Attali D (2016) shinyjs: easily Improve the User Experience of Your Shiny Apps in Seconds. https://CRAN.R-project.org/package=shinyjs. R package version 0.8. Accessed Feb 2017
5. Balsalobre-Fernández C, Tejero-González CM, del Campo-Vecino J, Alonso-Curiel D (2013) The effects of a maximal power training cycle on the strength, maximum power, vertical jump height and acceleration of high-level 400-meter hurdlers. J Hum Kinet 36(1):119–126
6. Bergmeir C, Benítez JM (2012) Neural networks in R using the Stuttgart neural network simulator: RSNNS. J Stat Softw 46(7):1–26. http://www.jstatsoft.org/v46/i07/. Accessed Feb 2017
7. Bishop CM (2006) Pattern recognition and machine learning. Information science and statistics. Springer, New York
8. Bompa TO, Haff G (1999) Periodization: theory and methodology of training, vol 199. Human Kinetics, Champaign
9. Chang W (2016) Shinydashboard: create dashboards with ’Shiny’. https://CRAN.R-project.org/package=shinydashboard. R package version 0.5.3. Accessed Feb 2017
10. Chang W (2016) Shinythemes: themes for Shiny. https://CRAN.R-project.org/package=shinythemes. R package version 1.1.1. Accessed Feb 2017
11. Chang W, Cheng J, Allaire J, Xie Y, McPherson J (2016) Shiny: web application framework for R. https://CRAN.R-project.org/package=shiny. R package version 0.14.2. Accessed Feb 2017
12. Curtis KM (2010) Cricket batting technique analyser/trainer: a proposed solution using fuzzy set theory to aid West Indies Cricket. In: Proceedings of the 9th WSEAS international conference on artificial intelligence, knowledge engineering and data bases, AIKED’10. World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, pp 71–76

13. Edelmann-Nusser J, Hohmann A, Henneberg B (2002) Modeling and prediction of competitive performance in swimming upon neural networks. Eur J Sport Sci 2(2):1–10
14. Er A, Dias R (2000) A rule-based expert system approach to process selection for cast components. Knowl Based Syst 13(4):225–234. https://doi.org/10.1016/S0950-7051(00)00075-7
15. Gu W, Saaty TL, Whitaker R (2016) Expert system for ice hockey game prediction: data mining with human judgment. Int J Inf Technol Decis Mak 15(04):763–789. https://doi.org/10.1142/S0219622016400022
16. Gupta S, Goswami A, Mukhopadhyay S (1999) Heart rate and blood lactate in 400 m flat and 400 m hurdle running: a comparative study. Indian J Physiol Pharmacol 43:361–366
17. Haghighat M, Rastegari H, Nourafza N (2013) A review of data mining techniques for result prediction in sports. Adv Comput Sci Int J 2(5):7–12
18. Hoerl AE, Kennard RW (1970) Ridge regression: biased estimation for nonorthogonal problems. Technometrics 12(1):55–67
19. Irani Z, Sharif A, Kamal MM, Love PE (2014) Visualising a knowledge mapping of information systems investment evaluation. Expert Syst Appl 41(1):105–125. https://doi.org/10.1016/j.eswa.2013.07.015 (**21st Century Logistics and Supply Chain Management**)
20. Iskra J (2012) Athlete typology and training strategy in the 400 m hurdles. New Stud Athl 27(1–2):27–37
21. Iskra J, Čoh M (2011) Biomechanical studies on running the 400 m hurdles. Hum Mov 12(4):315–323
22. Iskra J, Ryguła I (2001) The optimization of training loads in high class hurdlers. J Hum Kinet 6:59–72
23. Iskra J, Walaszczyk A (2003) Anthropometric characteristics and performance of 110m and 400m male hurdlers. Kinesiology 35(1):36–47
24. Iskra J, Widera J (2001) The training preparation of the world junior 400 m hurdles champion. Track Coach 156:4980–4984
25. Iskra J, Zajac A, Waskiewicz Z (2006) Laboratory and field tests in evaluation of anaerobic fitness in elite hurdlers. J Hum Kinet 16:25
26. Jain MB, Jain A, Srinivas MB (2008) A web based expert system shell for fault diagnosis and control of power system equipment. In: International conference on condition monitoring and diagnosis, CMD 2008, pp 1310–1313 (2008). https://doi.org/10.1109/CMD.2008.4580217
27. Kiartzis SJ, Bakirtzis AG, Theocharis JB, Tsagas G (2000) A fuzzy expert system for peak load forecasting. Application to the Greek power system. In: MELECON 2000: 10th Mediterranean Electrotechnical Conference, 2000, vol 3, pp 1097–1100. https://doi.org/10.1109/MELCON.2000.879726
28. Kłapcińska B, Iskra J, Poprzecki S, Grzesiok K (2001) The effects of sprint (300 m) running on plasma lactate, uric acid, creatine kinase and lactate dehydrogenase in competitive hurdlers and untrained men. J Sports Med Phys Fit 41(3):306–311
29. Kusy M, Obrzut B, Kluska J (2013) Application of gene expression programming and neural networks to predict adverse events of radical hysterectomy in cervical cancer patients. Med Biol Eng Comput 51(12):1357–1365. https://doi.org/10.1007/s11517-013-1108-8
30. Lapková D, Pluháček M, Komínková Oplatková Z, Adámek M (2014) Using artificial neural network for the kick techniques classification—an initial study. In: Proceedings 28th European conference on modelling and simulation ECMS, pp 382–387
31. Lee HJ, Rhee KP (2001) Development of collision avoidance system by using expert system and search algorithm. Int Shipbuild Prog 48(3):197–212
32. Lo CY, Chang HI, Chang YT (2009) Research on recreational sports instruction using an expert system. Springer, Berlin, pp 250–262. https://doi.org/10.1007/978-3-642-04875-3_28
33. Louzada F, Maiorano AC, Ara A (2016) iSports: a web-oriented expert system for talent identification in soccer. Expert Syst Appl 44:400–412. https://doi.org/10.1016/j.eswa.2015.09.007
34. Maszczyk A, Roczniok R, Waśkiewicz Z, Czuba M, Mikołajec K, Zajac A, Stanula A (2012) Application of regression and neural models to predict competitive swimming performance. Percept Motor Skills 114(2):610–626
35. Maszczyk A, Zajac A, Ryguła I (2011) A neural network model approach to athlete selection. Sports Eng 13(2):83–93
36. Mezyk E, Unold O (2011) Machine learning approach to model sport training. Comput Hum Behav 27(5):1499–1506
37. Mujika I (2009) Tapering and peaking for optimal performance. Human Kinetics, Champaign
38. Najjaran H, Sadiq R, Rajani B (2006) Fuzzy expert system to assess corrosion of cast/ductile iron pipes from backfill properties. Comput Aided Civ Infrastruct Eng 21(1):67–77. https://doi.org/10.1111/j.1467-8667.2005.00417.x
39. Papić V, Rogulj N, Pleština V (2009) Identification of sport talents using a web-oriented expert system with a fuzzy module. Expert Syst with Appl 36(5):8830–8838. https://doi.org/10.1016/j.eswa.2008.11.031
40. Pfeiffer M, Hohmann A (2012) Applications of neural networks in training science. Hum Mov Sci 31(2):344–359
41. Pfeiffer M, Perl J (2006) Analysis of tactical structures in team handball by means of artificial neural networks. Int J Comput Sci Sport 5(1):4–14
42. Platonow N (2015) Sistema a podgotowki sportsmienow o olimpijskom sportie. Olimpijskaja literatura, Kiev (**in Russian**)
43. Przednowek K, Iskra J, Przednowek KH (2014) Predictive modeling in 400-metres hurdles races. In: 2nd international congress on sport sciences research and technology support-icSPORTS 2014. SCITEPRESS, Rome, pp 137–144
44. Przednowek K, Iskra J, Wiktorowicz K, Krzeszowski T, Maszczyk A (2017) Planning training loads for the 400 m hurdles in three-month mesocycles using artificial neural networks. J Hum Kinet 60(1):175–189
45. Przednowek K, Wiktorowicz K, Krzeszowski T, Iskra J (2016) A fuzzy-based software tool used to predict 110m hurdles results during the annual training cycle. In: Proceedings of the 4th international congress on sport sciences research and technology support (icSPORTS-2016). SCITEPRESS, pp 176–181
46. R Core Team (2016) R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna. http://www.R-project.org/. Accessed Feb 2017
47. Riza LS, Bergmeir C, Herrera F, Benítez JM (2015) frbs: Fuzzy rule-based systems for classification and regression in R. J Stat Softw 65(6):1–30
48. Roczniok R, Ryguła I, Kwaśniewska A (2007) The use of Kohonen's neural networks in the recruitment process for sport swimming. J Hum Kinet 17:75–88
49. Rogulj N, Papić V, Cavala M (2009) Evaluation models of some morphological characteristics for talent scouting in sport. Coll Antropol 33(1):105–110
50. Ryguła I (2005) Artificial neural networks as a tool of modeling of training loads. In: 27th annual international conference of the engineering in medicine and biology society, IEEE-EMBS, pp 2985–2988
51. Schröder S, Dabidian P, Liedtke G (2015) A conceptual proposal for an expert system to analyze smart policy options for urban cep transports. In: Smart cities symposium Prague (SCSP), pp 1–6. https://doi.org/10.1109/SCSP.2015.7181555
52. Silva AJ, Costa AM, Oliveira PM, Reis VM, Saavedra J, Perl J, Rouboa A, Marinho DA (2007) The use of neural network technology to model swimming performance. J Sports Sci Med 6(1):117–125

53. Singh PK, Sarkar R (2015) A simple and effective expert system for schizophrenia detection. Int J Intell Syst Technol Appl 14(1):27–49. https://doi.org/10.1504/IJISTA.2015.072218
54. Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc Ser B 58(1):267–288
55. Venables WN, Ripley BD (2002) Modern applied statistics with S. Springer, New York. http://www.stats.ox.ac.uk/pub/MASS4
56. Wang LX, Mendel JM (1992) Generating fuzzy rules by learning from examples. IEEE Trans Syst Man Cybern 22(6):1414–1427. https://doi.org/10.1109/21.199466
57. Ward-Smith A (1997) A mathematical analysis of the bioenergetics of hurdling. J Sport Sci 15(5):517–526
58. Wiktorowicz K, Przednowek K, Lassota L, Krzeszowski T (2015) Predictive modeling in race walking. Comput Intell Neurosci 2015:9. https://doi.org/10.1155/2015/735060 Article ID 735060
59. Wilk R, Fidos-Czuba O, Rutkowski Ł, Kozłowski K, Wiśniewski P, Maszczyk A, Stanula A, Roczniok R (2015) Predicting competitive swimming performance. Cent Eur J Sport Sci Med 9(1):105–112
60. Zarinbal M, Fazel Zarandi MH, Turksen IB, Izadi M (2015) A type-2 fuzzy image processing expert system for diagnosing brain tumors. J Med Syst 39(10):1–20. https://doi.org/10.1007/s10916-015-0311-6
61. Zhou D, Ma J, Turban E, Bolloju N (2002) Soft decision analysis a fuzzy set approach to the evaluation of journal grades. Fuzzy Sets Syst 131(1):63–74. https://doi.org/10.1016/S0165-0114(01)00255-X
62. Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. J R Stat Soc Seri B (Stat Methodol) 67(2):301–320
63. Zou H, Hastie T (2012) Elasticnet: elastic-net for sparse estimation and sparse PCA. https://CRAN.R-project.org/package=elasticnet. R package version 1.1. Accessed Feb 2017
64. Zouhal H, Jabbour G, Jacob C, Duvigneau D, Botcazou M, Abderrahaman AB, Prioux J, Moussa E (2010) Anaerobic and aerobic energy system contribution to 400-m flat and 400-m hurdles track running. J Strength Cond Res 24(9):2309–2315