Engineering with Computers
https://doi.org/10.1007/s00366-020-01114-8

ORIGINAL ARTICLE

A new numerical learning approach to solve general Falkner–Skan model

Z. Hajimohammadi[1] · F. Baharifard[2] · K. Parand[1,3,4]

Received: 29 April 2020 / Accepted: 10 July 2020

Abstract
A new numerical learning approach namely Rational Gegenbauer Least Squares Support Vector Machines (RG_LS_SVM), is introduced in this paper. RG_LS_SVM method is a combination of collocation method based on rational Gegenbauer functions and LS_SVM method. This method converts a nonlinear high order model on a semi-infinite domain to a set of linear/nonlinear equations with equality constraints which decreases computational costs. Blasius, Falkner–Skan and MHD Falkner–Skan models and the effects of various parameters over them are investigated to satisfy accuracy, validity and efficiency of the proposed method. Both Primal and Dual forms of the problems are considered and the nonlinear models are converted to linear models by applying quasilinearization method to get the better results. Comparing the results of RG_LS_SVM method with available analytical and numerical solutions show that the present methods are efficient and have fast convergence rate and high accuracy.

Keywords Least squares support vector machines · Rational Gegenbauer functions · General Falkner–Skan model · QuasiLinearization method · Nonlinear ODE

Mathematics Subject Classification 34B40 · 65L60 · 68T05 · 76M25

1 Introduction

Many science and engineering phenomenons can be formulated by mathematical models [1–4]. Technology development leads to complexity and accuracy of high order models[5–8]. These models could be converted to nonlinear differential equations [9–11]. Complexity, non-linearity and higher order models needs more memory, computation and time. Classic methods such as numerical methods[1, 10, 12–14] are useful to solve high order models but need more memory and leads to large computations. These methods could solve models in discrete domain. In recent studies, researchers used machine learning methods in order to solve the models in continuous domain. Support Vector Machines[15], Neural Networks[16] and Deep Learning[6] are appropriate machine learning methods which can be used to solve the models. Combinations of numerical methods and machine learning methods can overcome high order complexity nonlinear models and introduce proper solutions for them. Develop and introduce a combination of collocation method as a numerical approach and LS_SVM as a machine learning approach is the aim of this paper in order to solve a nonlinear and high order model on a semi-infinite domain. The general Falkner–Skan model as a third order nonlinear model in a semi infinite interval, is applied to evaluate the performance of our proposed method namely RG_LS_SVM method. The rest of this section is ordered to introduce related basic methods and models including:

✉ Z. Hajimohammadi
Z_Hajimohammadi@sbu.ac.ir

F. Baharifard
f.baharifard@ipm.ir

K. Parand
k_parand@sbu.ac.ir

1 Department of Computer and Data Sciences, Faculty of Mathematical Sciences, Shahid Beheshti University, Tehran, Iran

2 School of Computer Science, Institute for Research in Fundamental Sciences (IPM), Tehran, Iran

3 Institute for Cognitive and Brain Sciences, Shahid Beheshti University, Tehran, Iran

4 Department of Statistics and Actuarial Science, University of Waterloo, Waterloo, Canada

LS_SVM method, Spectral methods and Falkner–Skan models.

LS_SVM method This method can achieve a global optimal solution of a model by converting a quadratic programming to a system of linear equations which leads to improve the accuracy of the solution [17, 18]. Vapnik[19] introduced SVM founded on statistical learning theory which adopt the structural risk minimization principle. Generally, using SVM method, a quadratic programming problem (QP) is solved to achieve optimal hyperplanes. Suykens and Vandewalle[5] modified SVM and introduced LS_SVM which uses a system of linear equations instead of QP and deal with equality constraints instead of inequality constraints. Lu et al. [20] introduced LS_SVM method to solve boundary value linear or nonlinear third-order and fourth-order problems with two points and multi points boundary conditions. Leake et al. [21] investigated a solution of linear/nonlinear models which have first and second orders based on Theory of Connections (ToC) and SVM method. Sharma et al.[22] applied nonlinear LS_SVM based on RBF kernel to develop an effectual automatic generation control of multi-area energy systems. Baymani et al.[23] extended ϵ-LS_SVM method to solve differential models. Also Mehrkanoon and Suykens[15] proposed a new method to obtain a solution for the delay differential equations using LS_SVM method. Mehrkanoon et al.[24] introduced an approximation approach to solve ordinary differential equations based on LS_SVM method. Yu et al.[25] investigated approximation solutions for one dimension convection diffusion model based on LS_SVM method. The selection of a kernel function is important for SVM/LS_SVM methods. Orthogonal polynomials can use for this selection to improve accuracy and performance. Moghaddam and Hamidzadeh[26] applied Hermite polynomials as kernel function to increase the speed of classifications. Padierna et al.[27] used the Gegenbauer family as kernel function and introduced a new formulation of them to improve accuracy by decreasing the number of support vectors. Legendre and Chebyshev polynomials are applied as kernel function in SVM classifiers[28–30]. Zanaty and Afifi[31] obtained an accurate classification by applying new Hermite kernel functions. This paper uses LS_SVM for introducing a new kernel function.

Spectral methods These methods have diverse applications in solving several engineering and science problems and received considerable attention[32–34]. Many of these problems are defined in semi-infinite/infinite domains and some spectral methods have been solved them. One approach is using the orthogonal polynomials to construct the approximation of the results[35, 36]. These polynomials can be defined in an infinite domain (like the Hermite and Laguerre polynomials) or a finite domain (like Jaccobi polynomials). For applying the polynomials that are orthogonal over a finite domain, the mapping rule is used to convert the primary model over an unbounded domain to a new model over a bounded interval[9, 37]. Another approach is domain truncation in which the infinite/semi-infinite domain replaced by $[-L, L]$ and $[0, L]$, respectively, where L is sufficiently large[38]. Also, rational approximations are the successful straight method to solve such problems. In the mentioned method, the orthogonal polynomials are mapped to the new spectral basis, named rational function on the unbounded domain[10, 39]. Therefore, these functions are mutually orthogonal on the semi-infinite domain and researchers used them for solving some nonlinear models over the unbounded domains[1, 10, 40]. One of the most important orthogonal polynomials, are named Gegenbauer polynomials[41] which are specific types of Jacobi polynomials, furthermore they are generalization of Legendre and Chebyshev polynomials. This paper focuses on rational Gegenbauer functions which are obtained from Gegenbauer polynomials and are used as the basic functions in the spectral methods.

Falkner–Skan models Falkner and Skan[42] introduced this model for the first time when they tried to develop the experiment of flow in a viscous fluid on a static wedge. Many researchers have attempted to solve Falkner–Skan model based on the challenging physical characteristics of it with various methods. For the condition in which this model leads to the Blasius problem, Bender et al.[43] used the new perturbation approach, He[44] applied the variational iteration method and Liao[45] obtained the solution using homotopy analysis method (HAM). Howarth[46] used homotopy perturbation method (HPM) to approximate the solution. Hashim[47] developed the numerical results for the Blasius model by employing the Adomian decomposition method (ADM) with Padé. Cortell[48] approximated the solutions of high-order initial value problems such as the Blasius equation with handling Runge–Kutta algorithm. Authors of Ref.[49, 50] checked the results of the Blasius model with collocation and Tau methods.

Furthermore, this problem was studied in the other literature[51–54].

Moreover, several researchers are interested to solving the Falkner–Skan model with different approaches. Liao[55] found the solution with HAM method, whereas this method is free of large or small physical variables. Asaithambi[56] presented numerical methods based on Taylor method. Kuo used a differential transformation method for Falkner–Skan model and presented a series solution of this model[57]. Salama[58] proposed a higher order method for solving such models.

Recently, interest in the study of magnetohydrodynamic (MHD) flows has been increased. Among these attempts, Hayat et al.[59] applied padé approximation with modified decomposition method to get the solution of MHD flow. Abbasbandy et al.[60] handled a homotopy analysis method. Authors of[61] established the uniqueness results to MHD model and showed these results are existed. The implicit finite difference scheme was applied to solve the partial differential equations governing the unsteady MHD-boundary-layer flow due to the impulsive motion of a stretching surface[62]. Takhar et al.[63] studied the MHD flow over a moving plate in a rotating fluid with the presence of a magnetic field, Hall currents and the free stream velocity. Heat generation and absorption of MHD flow of a micropolar fluid and MHD thermosolutal Marangoni convection were considered in Ref.[64, 65], respectively. Also, hydromagnetic natural convection embedded in a fluid-saturated porous medium was considered in some previous literature[66, 67]. Recently, some another numerical and analytical solutions have been applied to different types of MHD flows problems[68–70]. This paper considers the general Falkner–Skan model as high order nonlinear model on a semi-infinite domain.

Our results In this paper, we apply an important orthogonal functions, named rational Gegenbauer functions and develop the algorithm to solve the general Falkner–Skan model by a collocation method which is combined with LSSVM method. Solving different types of general Falkner–Skan model includes Blasius, Falkner–Skan and MHD Falkner–Skan models via combination of LS-SVM and QLM methods based on rational Gegenbauer functions is investigated. Our proposed method is a new and efficient numerical learning method to solve nonlinear/complex differential equations on a semi-infinite domain which is introduced in this paper for the first time. LS_SVM method has two forms includes Primal form and Dual form. We considered both mentioned forms in nonlinear and linear models and compared them together. Linearizing the model using QLM method and applying Lagrangian function, lead to solve a linear system of equations which yields to have better performance and faster convergence rate. Dual method has better run time comparing to Primal method and the run time of Dual QLM method works best by increasing the number of basic functions. The presentation of efficient numerical results leads to ensure the accuracy and performance of the proposed approach.

2 RG_LS_SVM method

In this section, Rational Gegenbauer Least Squares Support Vector Machines (RG_LS_SVM) method is introduced for finding a solution of nonlinear differential model in a semi-infinite interval. This method is a kind of LS_SVM method which uses rational Gegenbauer functions as kernel function. In the rest, first we propose LS_SVM method and then rational Gegenbauer functions are introduced.

2.1 LS_SVM method

The purpose of SVM method is to define the best discriminator hyperplanes by finding maximum margin for separating each group of similar data to classify whole data[19]. Consider TD $= \left\{(x_i, y_i) \mid i = 1 \cdots N,\ x_i \text{ and } y_i \in \mathbb{R}\right\}$ which is a set of training data. The function that approximates a behavior of training data TD is defined by y. Therefore, suppose a model has the following form:

$$y = w^T \varphi(x) = \sum_{j=1}^{N} w_j \varphi_j(x) + b. \tag{1}$$

The optimization problem is resulted from LS_SVM method is defined below as Primal form:

$$\min J(w, b, e) = \min_{w,b,e} \frac{1}{2} w^T w + \frac{\gamma}{2} e^T e, \tag{2}$$

$$s.t. \quad \left\{ y_i - \sum_{j=1}^{N} w_j \varphi_j(x_i) - b = e_i | i = 1 \dots N \right\}, \tag{3}$$

where $w = [w_1, w_2, \dots, w_N]^T$ is the normal of hyperplanes, b is a bias, $e = [e_1, e_2, \dots, e_N]^T$ is the error vector of training data, γ is a regularization parameter and $\varphi_j(x), 1 \le j \le N$ is a nonlinear function which maps x to the high dimensional feature space. The Lagrangian function for solving the optimal problem in (2)–(3), is constructed as follows:

$$\begin{aligned} L(w, b, e, \alpha) &= J(w, b, e) - \sum_{i=1}^{N} \alpha_i \left(\sum_{j=1}^{N} w_j \varphi_j(x_i) + b + e_i - y_i \right) \\ &= \min_{w,b,e} \max_{\alpha} \left\{ \tfrac{1}{2} w^T w + \tfrac{\gamma}{2} e^T e - \sum_{i=1}^{N} \alpha_i (w^T \varphi(x_i) + b + e_i - y_i) \right\}. \end{aligned} \tag{4}$$

where $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_N]^T$ is a vector of Lagrangian multipliers. The Karush–Kuhn–Tucker (KKT) optimality conditions leads to achieving the following results:

$$\begin{aligned} &\frac{\partial L}{\partial w_j} = 0 \Rightarrow w_j = \sum_{i=1}^{N} \alpha_i \varphi_j(x_i), \qquad 1 \le j \le N, \\ &\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i = 0, \\ &\frac{\partial L}{\partial e_j} = 0 \Rightarrow e_j = \frac{\alpha_j}{\gamma}, \qquad 1 \le j \le N, \\ &\frac{\partial L}{\partial \alpha_j} = 0 \Rightarrow \alpha_j = w^T \varphi(x_j) + b + e_j = y_j, \qquad 1 \le j \le N. \end{aligned} \tag{5}$$

Then an elimination of w and e will produce a system of equations which is named Dual form as the following form:

$$\begin{bmatrix} \kappa + \frac{1}{\gamma}I & J \\ J^T & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ b \end{bmatrix} = \begin{bmatrix} y \\ 0 \end{bmatrix}, \tag{6}$$

where $J = [1, 1, \dots, 1]_N^T$, $I_{N\times N}$ is an identity matrix, $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_N]^T$, $y = [y_1, y_2, \dots, y_N]^T$, $\kappa = [\kappa_{ij}] = K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$ for $i, j = 1 \cdots N$ and $\varphi(x) = [\varphi_1(x), \varphi_2(x), \dots, \varphi_N(x)]^T$.

Finally, by solving the above system of equations, y is calculated as follows:

$$y = \sum_{i=1}^{N} \alpha_i K(x, x_i) + b. \tag{7}$$

$K(x, x_i) = \varphi(x)^T \varphi(x_i)$ is named kernel function. A kernel function could have several choices which would effect on computational costs. In next subsection, the orthogonal rational Gegenbauer functions which satisfied the Mercer's theorem and could apply as a kernel function, are introduced.

2.2 Rational Gegenbauer functions

The Gegenbauer polynomials, which presented by $G_n^\lambda(\eta)$ is defined in the following form:

$$G_n^\lambda(\eta) = \sum_{j=0}^{[n/2]} (-1)^j \frac{\Gamma(n + \lambda - j)}{j!(n - 2j)!\Gamma(\lambda)} (2\eta)^{n-2j}, \tag{8}$$

where Γ is the Gamma function, λ is order of $G_n^\lambda(\eta)$ has a real number greater than $-\frac{1}{2}$ and n is degree of $G_n^\lambda(\eta)$ has an integer value[41]. These polynomials with the weight function $\rho(\eta) = (1 - \eta^2)^{\lambda - \frac{1}{2}}$ where $\lambda > -\frac{1}{2}$, are orthogonal on $[-1, 1]$ domain.

The new basis functions, induced by Gegenbauer polynomials named rational Gegenbauer functions are given by:

$$RG_n^\lambda(x) = G_n^\lambda\left(\frac{x - L}{x + L}\right) \quad n = 0, 1, 2, \dots,$$

where $\frac{x-L}{x+L} \in [-1, 1]$ and the constant parameter L is a scaling/stretching factor[71].

The recurrence relation of $RG_n^\lambda(x)$ is as follows:

$$\begin{aligned} RG_0^\lambda(x) &= 1, \\ RG_1^\lambda(x) &= 2\lambda \frac{x - L}{x + L}, \\ RG_{n+1}^\lambda(x) &= \frac{1}{n+1}\left[2\left(\frac{x - L}{x + L}\right)(n + \lambda) RG_n^\lambda(x) - (n + 2\lambda - 1) RG_{n-1}^\lambda(x)\right], n \geq 1. \end{aligned} \tag{9}$$

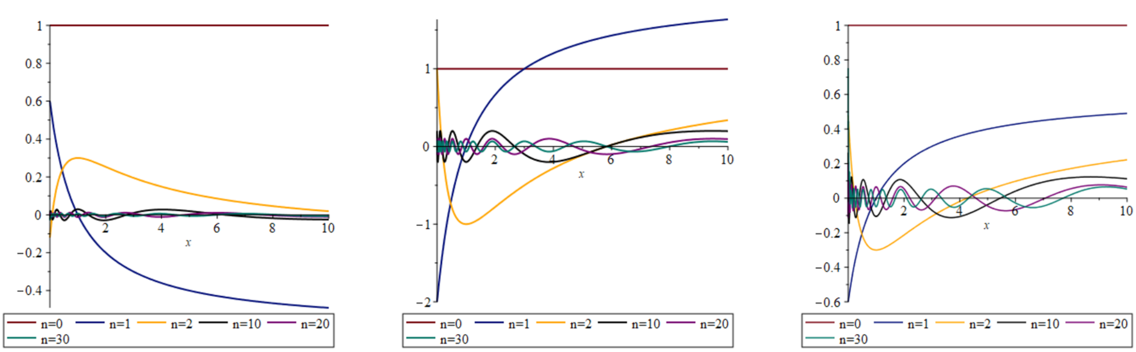
$RG_n^\lambda(x)$ is the nth eigenfunction of the singular Sturm-Liouville problem:

$$\begin{aligned} &(x + L)\frac{\sqrt{x}}{L}\frac{\mathrm{d}}{\mathrm{d}x}\left[(x + L)\sqrt{x}\frac{\mathrm{d}}{\mathrm{d}x} RG_n^\lambda(x)\right] \\ &\quad + \lambda\left(\frac{L^2 - x^2}{L}\right)\frac{\mathrm{d}}{\mathrm{d}x} RG_n^\lambda(x) + n(n + 2\lambda) RG_n^\lambda(x) = 0. \end{aligned} \tag{10}$$

The diagrams for the rational Gegenbauer functions of different degrees n for $\lambda = -0.3, 0, 0.3$ and $L = 1$ is depicted in Fig. 1.

Orthogonality: We determine $\hat{w}(x) = \frac{2L}{(x+L)^2}\left[1 - \left(\frac{x-L}{x+L}\right)^2\right]^{\lambda - \frac{1}{2}}$ as a non-negative, integrable and real-valued weight function for the rational Gegenbauer functions over the interval $I = [0, \infty)$. Therefore, $\{RG_n^\lambda(x)\}_{n\geq 0}$ becomes a system which is mutually orthogonal, i.e.

$$< RG_n^\lambda, RG_m^\lambda >_{\hat{w}} = \int_0^\infty RG_n^\lambda RG_m^\lambda \hat{w}(x)\mathrm{d}x = \frac{\pi 2^{1-2\lambda}\Gamma(n + 2\lambda)}{n!(n + \lambda)\Gamma^2(\lambda)} \delta_{nm}, \tag{11}$$



(a) $\lambda = -0.3$ **(b)** $\lambda = 0$ **(c)** $\lambda = 0.3$

Fig. 1 Samples of Rational Gegenbauer functions for various values of λ, $L = 1$ and different degrees where δ_{nm} is the Kronecker delta function.

If we define $\|v\|_{\hat{w}} = (\int_0^\infty |v(x)|^2 \hat{w}(x)\mathrm{d}x)^{\frac{1}{2}}$, then the induced system is completed in $L^2_{\hat{w}}(I) = \{v : I \to R \mid v \text{ is measurable and } \|v\|_{\hat{w}} < \infty\}$, and for any function $u \in L^2_{\hat{w}}(I)$, the following expansion holds

$$u(x) = \sum_{i=0}^{\infty} a_i RG_i^{\lambda}(x), \tag{12}$$

with

$$a_i = \frac{< u, RG_i^{\lambda} >_{\hat{w}}}{\| RG_i^{\lambda} \|^2_{\hat{w}}}. \tag{13}$$

Here, the a_i's are the expansion coefficients associated with the family $\{RG_k^{\lambda}(x)\}$.

Kernel function: Here, we introduce a LS_SVM kernel based on rational Gegenbauer function as follows:

$$K_{RG}(x, z) = \sum_{i=0}^{N} RG_i^{\lambda}(x)RG_i^{\lambda}(z), \tag{14}$$

where λ is fixed. To check the validity of the introduced LS_SVM kernel, we should show that it satisfies the Mercer theorem[72]. Mercer theorem states that for a given kernel function $K(x, z)$, the following integration should always be non-negative for any function $g(x)$:

$$\int\int K(x, z)g(x)g(z)\mathrm{d}x\mathrm{d}z \geq 0. \tag{15}$$

In the following theorem, we show that the rational Gegenbauer kernel functions has this condition.

Lemma 1 *Rational Gegenbauer function is a LS_SVM kernel function.*

Proof First, a value for λ is fixed and therefore we have

$$\begin{aligned}
&\int\int K_{RG}(x, z)g(x)g(z)\mathrm{d}x\mathrm{d}z = \int\int \left(\sum_{i=0}^{N} RG_i^{\lambda}(x)RG_i^{\lambda}(z)\right)g(x)g(z)\mathrm{d}x\mathrm{d}z \\
&= \sum_{i=0}^{N}\left(\int\int RG_i^{\lambda}(x)RG_i^{\lambda}(z)g(x)g(z)\mathrm{d}x\mathrm{d}z\right) = \sum_{i=0}^{N}\left(\int RG_i^{\lambda}(x)g(x)\mathrm{d}x \int RG_i^{\lambda}(z)g(z)\mathrm{d}z\right) \\
&= \sum_{i=0}^{N}\left(\int RG_i^{\lambda}(x)g(x)\mathrm{d}x \int RG_i^{\lambda}(x)g(x)\mathrm{d}x\right) = \sum_{i=0}^{N}\left(\int RG_i^{\lambda}(x)g(x)\mathrm{d}x\right)^2 \geq 0.
\end{aligned} \tag{16}$$

The Mercer condition is satisfied and this completes the proof. □

Lemma 2 *Weight function of rational Gegenbauer is a LS_SVM kernel function.*

Proof Weight function of rational Gegenbauer defined as follows:

$$\hat{w}(x) = \frac{2L}{(x+L)^2}\left[1 - \left(\frac{x-L}{x+L}\right)^2\right]^{\lambda - \frac{1}{2}}. \tag{17}$$

It is a LS_SVM kernel function in the following form if and only if it satisfies the Mercer condition for a fixed value of λ:

$$\begin{aligned}
&K_{\hat{w}}(x, z) = \hat{w}(x)\hat{w}(z) \\
&\int\int K_{\hat{w}}(x, z)g(x)g(z)\mathrm{d}x\mathrm{d}z = \int\int (\hat{w}(x)\hat{w}(z))g(x)g(z)\mathrm{d}x\mathrm{d}z \\
&= \int \hat{w}(x)g(x)\mathrm{d}x \int \hat{w}(z)g(z)\mathrm{d}z = \int \hat{w}(x)g(x)\mathrm{d}x \int \hat{w}(x)g(x)\mathrm{d}x \\
&= (\int \hat{w}(x)g(x)\mathrm{d}x)^2 \geq 0.
\end{aligned} \tag{18}$$

It is worth mentioning that the weight function is a non-negative and integrable, therefore the Mercer condition is satisfied. □

Theorem 1 *Multiplication of rational Gegenbauer function and weight function of rational Gegenbauer is a LS_SVM kernel function.*

Proof We introduce a novel kernel function of LS_SVM method as follows:

$$K_{RG\hat{w}}(x, z) = \sum_{i=0}^{N} RG_i^{\lambda}(x)RG_i^{\lambda}(z)\hat{w}(x)\hat{w}(z). \tag{19}$$

Rational Gegenbauer function is a kernel according to Lemma 1. Moreover, weight function of rational Gegenbauer is a kernel according to Lemma 2. So multiplication of two kernel which introduced in Eq. (19), will remain a kernel[26, 27] and the proof is done. □

The rational Gegenbauer functions can be divided in two classes: the first class has the specific values for $\lambda \in (-0.5, 0.5]$, and the other class contains $\lambda > 0.5$. The values of the weight function of rational Gegenbauer for the first class can be set to one for simplicity of the computations in the corresponding kernel.

3 Applying RG_LS_SVM method to solve nonlinear differential model

An application of RG_LS_SVM method is explained here to find a solution of a nonlinear differential model on a semi-infinite domain. This application is satisfied by two main techniques. The first technique is to apply RG_LS_SVM method in Primal and Dual forms to solve nonlinear models. In the second technique, nonlinear models must first be linearized and then the solution of the new linear forms should be found by RG_LS_SVM method in Primal and Dual forms. Note that we select Quasilinearization Method (QLM) to linearize the nonlinear models. At the rest of this section, we explain two mentioned techniques significantly.

3.1 Nonlinear method

Given the nonlinear m-order differential model as:

$$F(y^{(m)}, y^{(m-1)}, \dots, y'', y', y, x) = 0,$$
$$IB = \{y^{(i)}(x_j) = z_i, \quad | \quad 0 \le i \le m-1, \quad j \in \{0, m-1\}\}, \tag{20}$$

where $y^{(i)}$ is a i-th derivative of function y, F is a nonlinear operator of $y^{(i)}$ $(i = 0, .., m-1)$ and x, and IB is a set of initial/boundary conditions.

First, an approximation of a function $\hat{y}$ is expanded by rational Gegenbauer functions as:

$$\hat{y} = \sum_{i=1}^{N} w_i RG_i^{\lambda}(x) + b. \tag{21}$$

The derivative and residual functions are constructed in the following form:

$$\begin{aligned} D(\hat{y}) &= \hat{y}' = \sum_{i=1}^{N} w_i (RG_i^{\lambda}(x))', \\ D^2(\hat{y}) &= \hat{y}'' = \sum_{i=1}^{N} w_i (RG_i^{\lambda}(x))'', \\ &.... \\ D^{(m)}(\hat{y}) &= \hat{y}^{(m)} = \sum_{i=1}^{N} w_i (RG_i^{\lambda}(x))^{(m)}, \\ Res(x) &= F(\hat{y}^{(m)}, \hat{y}^{(m-1)}, \dots, \hat{y}'', \hat{y}', \hat{y}, x). \end{aligned} \tag{22}$$

Then, we collocate the roots of $RG_{N+1}^{\lambda}(x)$ as the training data to $Res(x)$ to build constraints for LS_SVM method and add a set of initial/boundary conditions (IB) to complete the constraints. So, we have the following set for $1 \le i \le N+m+1$:

$$\{Res(x_l) - e_i = 0 | 1 \le l \le N+1\} \cup \{y^{(k)}(x_j) - z_k - e_i = 0 | j \in \{0, m-1\} \; 0 \le k \le m-1\} \setminus A, \tag{23}$$

where $A = \emptyset$ in Primal form and in Dual form A is a subset of m members of $\{Res(x_l) - e_i = 0 | 1 \le l \le N+1\}$ set. Objective function is resulted as follows:

$$\min J(w, b, e) = \min_{w,b,e} \left\{ \frac{1}{2} w^T w + \frac{\gamma}{2} e^T e \right\}. \tag{24}$$

Finally, LS_SVM method is applied to obtain the unknown coefficients according to the method is introduced in Sect. 2.1. The whole steps of our proposed method is presented in Algorithm 1.

Algorithm 1 APPLYING RG_LS_SVM AND NONLINEAR METHOD

input: $N, \lambda, L, \gamma, F, IB, m$ and $Type$
output: $\hat{y}$

calculate $RG = [RG_1^\lambda, \cdots, RG_N^\lambda]^T$
find $X = [x_i | RG_{N+1}^\lambda(x_i) = 0]$
set $r_i = L\left(\frac{1+x_i}{1-x_i}\right), \quad 1 \le i \le N+1$
$\hat{y} \leftarrow w^T RG + b$
for $j \leftarrow 0$ to m **do**
    $\hat{y}^{(j)} \leftarrow w^T \mathrm{D}^{(j)} RG$
$\mathrm{RES}(\mathrm{x}) \leftarrow F(\hat{y}^{(m)}, \cdots, \hat{y}', \hat{y}, x)$
$\hat{\mathrm{IB}}_k \leftarrow IB_k(\hat{y}^{(m-1)}, \cdots, \hat{y}', \hat{y}, x), \quad 0 \le k \le m-1$
$\mathrm{CONS} \leftarrow \{\mathrm{RES}(r_l) - e_i = 0 | 1 \le l \le N+1\} \cup \{\hat{\mathrm{IB}}_k - e_i = 0 | 0 \le k \le m-1\} \setminus A, \quad 1 \le i \le N+m+1$
$\mathrm{OBJFUN} \leftarrow \{\frac{1}{2} w^T w + \frac{\gamma}{2} e^T e\}$
if $Type == Primal$ **then**
    $[w, b] \leftarrow$ Primal_LS_SVM(OBJFUN, CONS)
    $\hat{y} \leftarrow w^T RG + b$
if $Type == Dual$ **then**
    $[\alpha, b] \leftarrow$ Dual_LS_SVM(OBJFUN, CONS)
    $k_j \leftarrow RG_j^\lambda(x) RG_j^\lambda(r_j) \hat{w}(x) \hat{w}(r_j), \quad 1 \le j \le N$
    $K \leftarrow [k_1, \cdots, k_N]^T$
    $\hat{y} \leftarrow \alpha^T K + b$
Return $\hat{y}$

3.2 QLM method

Quasilinearization method (QLM) is an iterative method which converts the nonlinear model to a linear system of equations in each of its iteration. QLM is a generalized model of Newton-Raphson's method and has a quadratic convergence rate[10, 73]. QLM method linearizes Eq. (20) to the following equation:

$$\begin{aligned} y_{t+1}^{(m)} &= H(y_t^{(m-1)}, \dots, y_t'', y_t', y_t, x) \\ &+ \sum_{s=0}^{m-1} (y_{t+1}^{(s)} - y_t^{(s)}) \frac{\partial H(y_t^{(m-1)}, y_t^{(m-2)}, \dots, y_t', y_t, x)}{\partial y_t^{(s)}}, \\ IB_{t+1} &= \{y_{t+1}^{(i)}(x_j) = z_i, \quad | \quad 0 \le i \le m-1, \quad j \in \{0, m-1\}\}, \end{aligned} \tag{25}$$

where t is the number of iterations, $y^{(i)}$ is the i-th derivative of function y, H is the nonlinear operator of $y^{(i)}$ and x for $i = 0, .., m-1$ and IB_{t+1} is the set of initial/boundary conditions in each iteration.

An approximation of a function $\hat{y}_{t+1}$ is expanded by rational Gegenbauer functions as follows:

$$\hat{y}_{t+1} = \sum_{i=1}^{N} w_{i,t+1} RG_i^\lambda(x) + b. \tag{26}$$

The derivatives of $\hat{y}_{t+1}$ are calculated and residual function is constructed in the following form:

$$\begin{aligned} Res_{t+1}(x) &= \hat{y}_{t+1}^{(m)} - H(\hat{y}_t^{(m-1)}, \hat{y}_t^{(m-2)}, \dots, \hat{y}_t'', \hat{y}_t', \hat{y}_t, x) \\ &- \sum_{s=0}^{m-1} (y_{t+1}^{(s)} - y_t^{(s)}) \frac{\partial H(y_t^{(m-1)}, y_t^{(m-2)}, \dots, y_t', y_t, x)}{\partial y_t^{(s)}}. \end{aligned} \tag{27}$$

Then, we collocate the roots of $RG_{N+1}^\lambda(x)$ as the training data to $Res_{t+1}(x)$ to build constraints for LS_SVM method and add a set of initial/boundary conditions (IB_{t+1}) to complete the constraints. Therefore, we have the following set for $1 \le i \le N+m+1$:

$$\begin{aligned} \{Res_{t+1}(x_l) - e_i = 0 | 1 \le l \le N+1\} \cup \{\hat{y}_{t+1}^{(k)}(x_j) - z_k - e_i \\ = 0 | j \in \{0, m-1\} \ 0 \le k \le m-1\} \setminus A, \end{aligned} \tag{28}$$

where $A = \emptyset$ in Primal form and in Dual form A is a subset of m members of $\{Res_{t+1}(x_l) - e_i = 0 | 1 \le l \le N+1\}$ set. Objective function is resulted as follows:

$$\min J(w_{t+1}, b_{t+1}, e_{t+1}) = \min_{w_{t+1}, b_{t+1}, e_{t+1}} \left\{ \frac{1}{2} w_{t+1}^T w_{t+1} + \frac{\gamma}{2} e_{t+1}^T e_{t+1} \right\}, \tag{29}$$

Finally, LS_SVM method is applied to obtain the unknown coefficients according to the method introduced in Sect. 2.1 in each iteration. Iterations of QLM method stops when $\left| y_{t+1}^{(n)} - y_t^{(n)} \right| \le \epsilon$ in which ϵ is a small constant or the number of iterations of QLM method is selected constant. The whole steps of our proposed method is presented in Algorithm 2.

Algorithm 2 Applying RG_LS_SVM and QLM method

input: $N, \lambda, L, \gamma, F, IB, nqlm, m$ and $Type$

output: $\hat{y} = \hat{y}_{nqlm}$

calculate $RG = [RG_1^{\lambda}, \cdots, RG_N^{\lambda}]^T$

find $X = [x_i | RG_{N+1}^{\lambda}(x_i) = 0]$

set $r_i = L\left(\frac{1+x_i}{1-x_i}\right), \quad 1 \leq i \leq N+1$

for $t \leftarrow 1$ to $nqlm - 1$ **do**

  $\hat{y}_{t+1} \leftarrow w_{t+1}^T RG + b_{t+1}$

  for $j \leftarrow 0$ to m **do**

    $\hat{y}_{t+1}^{(j)} \leftarrow w_{t+1}^T \mathrm{D}^{(j)} RG$

  find $Res_{t+1}(x)$ according to Eq. (27)

  $\hat{\mathrm{IB}}_{k,t+1} \leftarrow IB_{k,t+1}(\hat{y}_{t+1}^{(m-1)}, \cdots, \hat{y}'_{t+1}, \hat{y}_{t+1}, x), \quad 0 \leq k \leq m-1$

  $\mathrm{Cons}_{t+1} \leftarrow \{\mathrm{Res}_{t+1}(r_l) - e_{i,t+1} = 0 | 1 \leq l \leq N+1\} \cup \{\hat{\mathrm{IB}}_{k,t+1} - e_{i,t+1} = 0 | 0 \leq k \leq m-1\} \setminus A, \quad 1 \leq i \leq N+m+1$

  $\mathrm{ObjFun}_{t+1} \leftarrow \{\frac{1}{2} w_{t+1}^T w_{t+1} + \frac{\gamma}{2} e_{t+1}^T e_{t+1}\}$

  if $Type == Primal$ **then**

    $[w_{t+1}, b_{t+1}] \leftarrow$ Primal_LS_SVM(ObjFun_{t+1}, Cons_{t+1})

    $\hat{y_{t+1}} \leftarrow w_{t+1}^T RG + b_{t+1}$

  if $Type == Dual$ **then**

    $[\alpha_{t+1}, b_{t+1}] \leftarrow$ Dual_LS_SVM(ObjFun_{t+1}, Cons_{t+1})

    $k_j \leftarrow RG_j^{\lambda}(x) RG_j^{\lambda}(r_j) \hat{w}(x) \hat{w}(r_j), \quad 1 \leq j \leq N$

    $K \leftarrow [k_1, \cdots, k_N]^T$

    $\hat{y}_{t+1} \leftarrow \alpha_{t+1}^T K + b_{t+1}$

Return $\hat{y}$

4 Application of the proposed method to solve general Falkner–Skan model

In this section, the general Falkner–Skan model is described as a nonlinear differential model on a semi-infinite domain and a solution of this model is obtained by RG_LS_SVM method.

4.1 General Falkner–Skan model

The equation describes the effects of a viscous in-compressible profile of the fluid on the steady two dimensional laminar boundary wedge flow on a semi-infinite interval can be summarized as the following form[42]:

$$f'''(\eta) + \alpha f(\eta) f''(\eta) + \beta[1 - f'(\eta)^2] = 0, \tag{30}$$

with boundary conditions

$$f(0) = \mu, \quad f'(0) = \delta, \quad f'(+\infty) = 1. \tag{31}$$

where f and η are dimensionless and f is the stream function, α and, β are parameters, μ is the wall mass transfer coefficient and δ is the wall stretching parameter. In this model, $f'(\eta)$ and $f''(\eta)$ are related to velocity distribution and the skin friction, respectively. When $\beta = 0$, we have the classical Blasius model as below:

$$f''' + \alpha f f'' = 0. \tag{32}$$

where α is a constant in the interval $\frac{1}{2} \leq \alpha \leq 1$. Therefore, the case $\alpha = \frac{1}{2}$ gives the well-known Blasius equation[2].

In Eq. (30), if $\alpha = 1$, the Falkner–Skan equation is achieved:

$$f''' + f f'' + \beta(1 - f'^2) = 0. \tag{33}$$

Here the interest is to find solution of Eq. (33) for wedge in the accelerated flow ($\beta > 0$) and decelerated flow ($\beta < 0$) with separation[60].

Moreover, the MHD Falkner–Skan equation that is related to steady two-dimensional laminar magneto-hydrodynamic flow is[74]

$$f''' + f f'' + \beta(1 - f'^2) - M^2(f' - 1) = 0. \tag{34}$$

where M is dimensionless magnetic parameter.

There are many techniques which have been applied to obtain both numerical and analytical solutions for the above equations in the boundary conditions where μ and δ are equal to zero. So, we can generalize these equations as below and investigate the solution of it for various parameters and compare with others.

$$\begin{cases} f''' + c_1 ff'' + c_2(1 - f'^2) - c_3(f' - 1) = 0, \\ f(0) = 0, \quad f'(0) = 0, \quad f'(+\infty) = 1. \end{cases} \tag{35}$$

4.2 Solution of general Falkner–Skan using by RG_LS_SVM method

General Falkner–Skan model with boundary conditions was described in Eq. (35). Two types of solution of this model are obtained by applying Algorithms 1 and 2. For using Algorithm 1, we should set the following form which satisfied the boundary conditions:

$$\begin{aligned} \hat{y} &= \frac{\eta^2}{\eta + 1} + \frac{\eta^2}{\eta^2 + 1} \sum_{i=0}^{N} w_i RG_i^{\lambda}(\eta), \\ Res(\eta) &= \hat{y}''' + c_1 \hat{y}\hat{y}'' + c_2(1 - \hat{y}'^2) - c_3(\hat{y}' - 1). \end{aligned} \tag{36}$$

For using Algorithm 2, we should set the residual function and a set of boundary conditions built in the QLM method as follows:

$$\begin{aligned} \hat{y}_0 &= \frac{\eta^2}{\eta + 1} + \frac{\eta^2}{\eta^2 + 1}, \\ \hat{y}_{t+1} &= \frac{\eta^2}{\eta + 1} + \frac{\eta^2}{\eta^2 + 1} \sum_{i=0}^{N} w_{i,t+1} RG_i^{\lambda}(\eta), \\ Res_{t+1}(\eta) &= \hat{y}'''_{t+1} + c_1(\hat{y}_{t+1}\hat{y}''_t + \hat{y}_t\hat{y}''_{t+1} - \hat{y}_t\hat{y}''_t) \\ &\quad + c_2(1 + (\hat{y}'_t)^2 - 2y'_{t+1}y'_t) - c_3(\hat{y}'_{t+1} - 1). \end{aligned} \tag{37}$$

where t is the number of QLM iterations, $t = 0 \cdots nqlm - 1$.

It should be noted that general Falkner–Skan model has not the exact solution and for evaluation of the criteria of this model, we do not have the target values. Therefore, we institute the residual function to absolute error and apply mean squared error (MSE) for training and test sets as follows:

$$\begin{aligned} \mathrm{MSE}_{\mathrm{train}} &= \frac{1}{N + 1} \sum_{i=1}^{N+1} (Res(\eta_i))^2, \\ \mathrm{MSE}_{\mathrm{test}} &= \int_0^{\infty} (Res(\eta))^2 \mathrm{d}\eta. \end{aligned} \tag{38}$$

where $\eta_i, i = 1 \cdots N + 1$ are the members of training set includes the roots of $RG_{N+1}^{\lambda}(\eta)$ and also Res is the residual function. We calculated the numerical results of general Falkner–Skan model with four approaches: Primal Nonlinear method, Dual Nonlinear method, Primal QLM method and Dual QLM method. Numerical results are reported and discussed in the next section.

5 Numerical results

Blasius, Falkner–Skan and MHD Falkner–Skan models are investigated to show the performance of RG_LS_SVM method on nonlinear differential models on a semi-infinite domain. All the results are performed in Maple software using a windows 7 system with Intel(R)-core(TM)i7 CPU with 64 bit operating system and 10.00 GB RAM. The parameters of rational Gegenbauer functions and the regularization parameter γ are the tuning parameters and affect on the performance of our experiments. The parameters of rational Gegenbauer functions, λ and L are reported in Table 1. The regularization parameter γ has a completely large value to minimize the residual functions sharply and satisfy well the boundary conditions. The amount of γ is chosen 10^8 for all tests except for Blasius model with nonlinear method which is set to 10^4 to avoid ill conditioning. Other parameters are obtained by repeating the experiments to minimize the mean squared errors in Eq. (38). To gain the convergence of the proposed method, we increase the number of basic functions (the number of training set) until the test errors and the training errors are properly aligned. The results for some samples of general Falkner–Skan model are reported in Table 2. Figure 2 presented the logarithmic scale of MSE for different models in Dual method. It is noted that the solution of models has logarithmic convergence rate. Furthermore, the graphical results of QLM method in this figure is better than the Nonlinear method in training and testing sets. The reported results in Tables 3 and 4 are based on the test data.

5.1 Blasius model's results

According to Eq. (32), the Blasius model is obtained by considering $c_2 = c_3 = 0$ and $c_1 = \alpha$ in Eq. (35). Here, we present the numerical results obtained by RG_LS_SVM method for the stream function $f(\eta)$, velocity distribution $f'(\eta)$ and the skin friction $f''(\eta)$ of the well-known Blasius model in which the parameter c_1 equals to $\frac{1}{2}$ in Table 3. These results are presented and compared with the results of Runge–Kutta method[48] to show that RG_LS_SVM is accurate. In the Blasius model, the second order derivative of f at the origin ($f''(0)$) has an important and fundamental role. Therefore we obtain this quantity of the Blasius equation for various value of parameter c_1 and present them in Table 4 by comparing with the results in Refs.[46] and[48]. It can be shown that, the RG_LS_SVM has been able to obtain this value with high precision. The figures of f, f' and f'' of the Blasius model for Dual QLM method is shown in Fig. 3. The solution of Blasius problem is increasing with respect to η. Note that for all cases of Blasius equations, the velocity profile get to one and the skin friction decays to zero, however these events are slower for the well-known Blasius model.

5.2 Falkner–Skan model's results

Falkner–Skan model is described in Eq. (33) (consider if $c_1 = 1$, $c_2 = \beta$ and $c_3 = 0$ in Eq. (35), then the model is obtained from Eq. (33) and Eq. (35) are equal). It should be noted that $f''(0)$ is important for this model with various values of the parameter c_2. A comparison of the proposed method's results with Asaithambi's[56] and Salama's results[58] are shown in Table 4. $f(\eta)$, $f'(\eta)$ and $f''(\eta)$ of Falkner–Skan model for Dual QLM method are shown in Fig. 4. Consider if $c_2 = 0$ is special type of Blasius model and numerical results for it is reported in Sect. 5.1. Graphical results of this type is reported in Fig. 4 compared with other types of Falkner–Skan model; it can be seen when c_2 increases, the values of function f have a slight change, but f' and f'' values increases.

5.3 MHD Falkner–Skan model's results

MHD Falkner–Skan model is presented in Eq. (34) (note if $c_1 = 1$, $c_2 = \beta$ and $c_3 = M^2$ in Eq. (35), then the obtained models from Eq. (34) and Eq. (35) are equal). This model has tow parameters: c_2 and c_3. $f''(0)$ is a main role for this model which evaluates for various values of parameters c_2 and c_3 and compares with two well-known other methods[51, 60]. Numerical and graphical results are investigated in two cases: the accelerated flow ($c_2 = \frac{4}{3}$) as ($c_2 > 0$) and the decelerated flow ($c_2 = -3$) as ($c_2 < 0$) for different values of a dimensionless magnetic parameter (c_3). Numerical results is represented in Table 4. Figures 5 and 6, are reported $f(\eta)$, $f'(\eta)$ and $f''(\eta)$ of MHD Falkner–Skan models for Dual

Table 1 The values of (λ, L) parameters used in our methods

Model	$1/c_1$	Nonlinear		QLM	
		Primal	Dual	Primal	Dual
Blasius ($c_2 = 0, c_3 = 0$)	1	(−0.3, 0.9)	(−0.49, 1)	(0.5, 1)	(0.5, 1)
	1.2	(−0.3, 1)	(−0.49, 1)	(0.5, 1)	(0.5, 1)
	1.5	(−0.3, 1.1)	(−0.49, 1)	(0.5, 1)	(0.5, 1)
	1.8	(−0.3, 1.2)	(−0.49, 1)	(0.5, 1)	(0.5, 1)
	2	(−0.3, 1.3)	(−0.49, 1)	(0.5, 1)	(0.5, 1)
	c_2				
Falkner–Skan ($c_1 = 0, c_3 = 0$)	1	(−0.49, 1)	(−0.49, 1)	(0.5, 1)	(0.5, 1)
	2	(−0.49, 1)	(−0.49, 1)	(0.5, 1)	(0.5, 1)
	10	(0.2, 1)	(0.2, 1)	(0.5, 1)	(0.5, 1)
	15	(0.2, 1)	(0.2, 1)	(0.5, 1)	(0.5, 1)
	20	(0.2, 1)	(0.2, 1)	(0.5, 1)	(0.5, 1)
	30	(0.2, 1)	(0.2, 1)	(0.5, 1)	(0.5, 1)
	40	(0.2, 1)	(0.2, 1)	(0.5, 1)	(0.5, 1)
	$\sqrt{c_3}$				
MHD-Falkner–Skan ($c_1 = 1, c_2 = 4/3$)	1	(0.3, 1)	(0.3, 1)	(0.5, 1)	(0.5, 1)
	2	(0.3, 1)	(0.3, 1)	(0.5, 1)	(0.5, 1)
	5	(0.3, 1)	(0.3, 1)	(0.5, 1)	(0.5, 1)
	10	(0.5, 1)	(0.5, 1)	(0.5, 1)	(0.5, 1)
	50	(0.5, 0.3)	(0.5, 0.3)	(0.5, 0.3)	(0.5, 0.3)
	100	(0.5, 0.3)	(0.5, 0.3)	(0.5, 0.3)	(0.5, 0.3)
	$\sqrt{c_3}$				
MHD-Falkner–Skan ($c_1 = 1, c_2 = -3$)	3	(−0.49, 1)	(−0.49, 1)	(0.5, 1)	(0.5, 1)
	4	(0.3, 1)	(0.3, 1)	(0.5, 1)	(0.5, 1)
	5	(0.3, 1)	(0.3, 1)	(0.5, 1)	(0.5, 1)
	10	(0.3, 1)	(0.3, 1)	(0.5, 1)	(0.5, 1)
	15	(0.5, 1)	(0.5, 1)	(0.5, 1)	(0.5, 1)
	20	(0.5, 0.5)	(0.5, 0.5)	(0.5, 0.5)	(0.5, 0.5)
	50	(0.5, 0.3)	(0.5, 0.3)	(0.5, 0.3)	(0.5, 0.3)

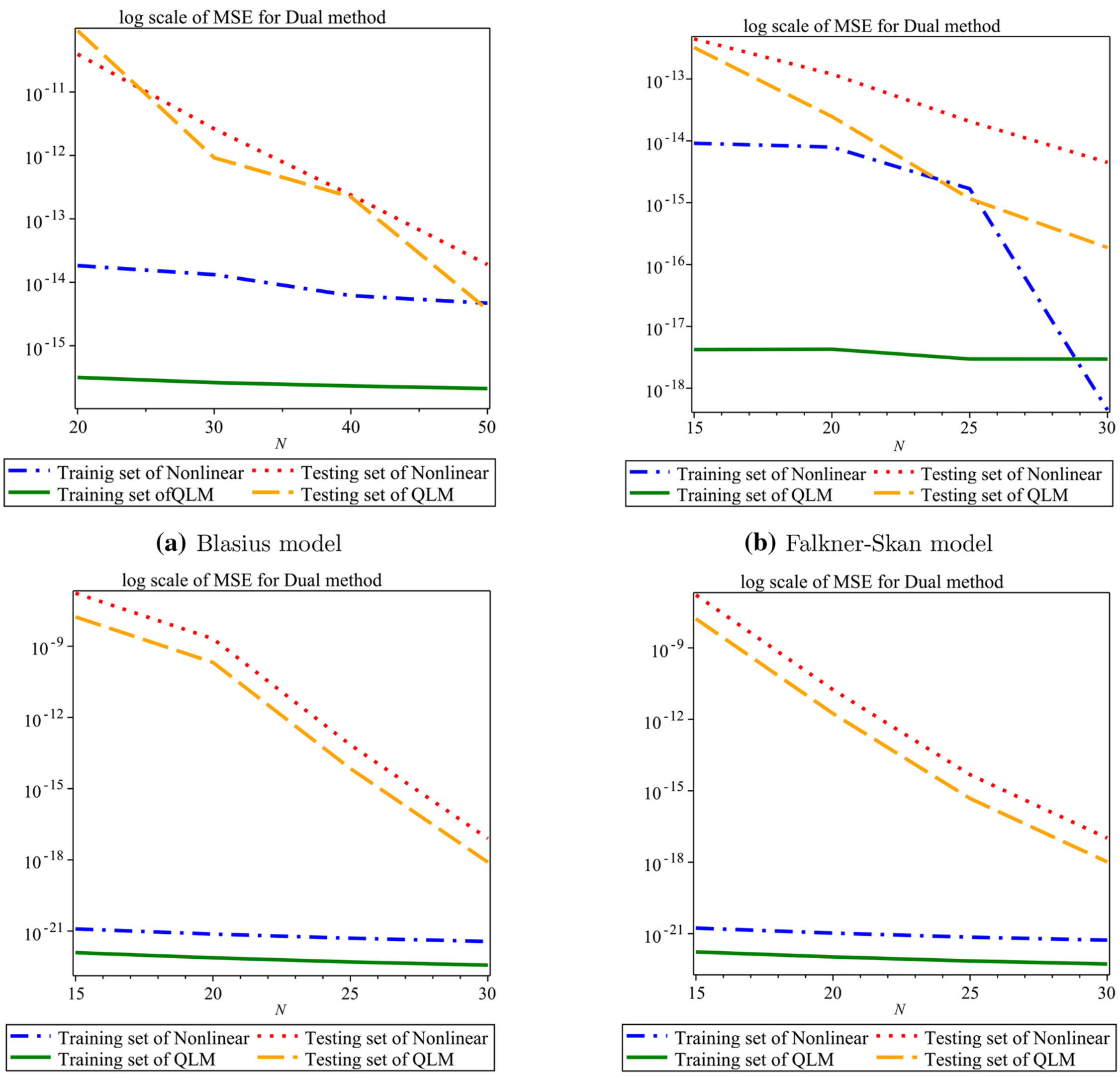
Table 2 Mean squared errors of training and testing sets and CPU time reports of some samples of general Falkner–Skan model

Model	Method	N	Primal			Dual		
			Train	Test	CPU Time(s)	Train	Test	CPU Time(s)
Blasius equation ($c_1 = 1/2, c_2 = 0, c_3 = 0$)	Nonlinear	20	3.19e−06	2.70e−04	45.49	1.83e−14	3.95e−11	7.35
		30	3.81e−08	1.01e−05	151.90	1.32e−14	2.62e−12	26.54
		40	4.74e−09	2.86e−06	398.19	6.16e−15	2.38e−13	59.26
		50	3.38e−09	2.19e−06	944.88	4.68e−15	1.90e−14	137.94
	QLM	20	2.16e−16	8.28e−11	11.93	2.16e−16	8.28e−11	8.44
		30	1.62e−16	8.18e−13	32.98	1.62e−16	8.18e−13	24.40
		40	1.32e−16	1.22e−13	71.62	1.32e−16	1.22e−13	67.03
		50	1.11e−16	3.00e−15	174.92	1.11e−16	2.70e−15	129.85
Falkner–Skan equation ($c_1 = 1, c_2 = 1, c_3 = 0$)	Nonlinear	15	1.27e−06	9.04e−04	15.19	9.15e−15	4.45e−13	5.66
		20	3.83e−09	1.03e−05	43.85	7.94e−15	1.20e−13	9.76
		25	2.60e−11	3.01e−07	73.90	1.68e−15	2.06e−14	19.29
		30	4.46e−12	4.23e−08	137.72	2.44e−16	4.48e−15	29.22
	QLM	15	4.22e−18	3.23e−13	7.47	4.22e−18	3.23e−13	5.32
		20	4.28e−18	2.47e−14	14.13	4.28e−18	2.47e−14	9.67
		25	2.97e−18	1.16e−15	24.15	2.97e−18	1.16e−15	16.77
		30	2.96e−18	1.88e−16	37.27	2.96e−18	1.88e−16	26.44
MHD-Falkner–Skan ($c_1 = 1, c_2 = 4/3, \sqrt{c_3} = 100$)	Nonlinear	15	1.10e−03	6.29e−02	22.43	1.23e−21	1.72e−07	4.01
		20	2.17e−09	2.77e−07	48.38	7.44e−22	2.08e−09	7.33
		25	1.06e−13	2.36e−11	156.15	4.98e−22	7.02e−14	14.30
		30	9.09e−19	3.42e−16	157.08	3.65e−22	7.98e−18	23.01
	QLM	15	1.23e−21	1.72e−03	6.90	1.23e−22	1.72e−08	5.69
		20	7.44e−22	2.08e−09	12.60	7.44e−23	2.08e−10	8.15
		25	4.98e−22	7.02e−14	22.59	4.98e−23	7.02e−15	13.78
		30	3.65e−22	7.98e−18	36.52	3.65e−23	7.98e−19	22.86
MHD-Falkner–Skan ($c_1 = 1, c_2 = -3, \sqrt{c_3} = 50$)	Nonlinear	15	4.54e−11	2.18e−07	24.65	1.72e−21	1.66e−07	4.06
		20	1.80e−11	2.56e−09	58.80	1.06e−21	1.75e−11	7.91
		25	1.65e−17	1.08e−14	114.68	7.15e−22	4.73e−15	14.34
		30	3.13e−19	1.31e−16	197.43	5.36e−22	1.04e−17	24.37
	QLM	15	1.72e−21	1.66e−07	6.99	1.72e−22	1.66e−08	5.68
		20	1.06e−21	1.75e−11	13.06	1.06e−22	1.75e−12	10.50
		25	7.15e−22	4.73e−15	23.37	7.15e−23	4.73e−16	18.92
		30	5.36e−22	1.04e−17	35.96	5.36e−23	1.04e−18	30.99

QLM method when c_2 is equal to $\frac{4}{3}$ and -3, respectively. The results show that RG_LS_SVM method is more efficient, accurate and has a good precision error rate.

6 Conclusion

In this paper, we studied the problem of solving a nonlinear ordinary differential equations in a semi-infinite interval by a new method named rational Gegenbauer least squares support vector machines (RG_LS_SVM). We considered the Blasius, Falkner–Skan and MHD Falkner–Skan models in fluid dynamics field which are nonlinear ordinary differential equations and applied our method for solving them. The effects of various parameters over these models were investigated and we got high-precision results in comparison to other previous well-known studies. Our method is based on combination of collocation and LS_SVM methods and this is the first time that rational Gegenbauer functions were used in LS_SVM method for solving the nonlinear ordinary differential models on a semi-infinite domain. RG functions were applied as a kernel to find the unknown coefficient of our



(a) Blasius model

(b) Falkner-Skan model

(c) MHD-Falkner-Skan model ($c_1 = 1, c_2 = 4/3, \sqrt{c_3} = 100$)

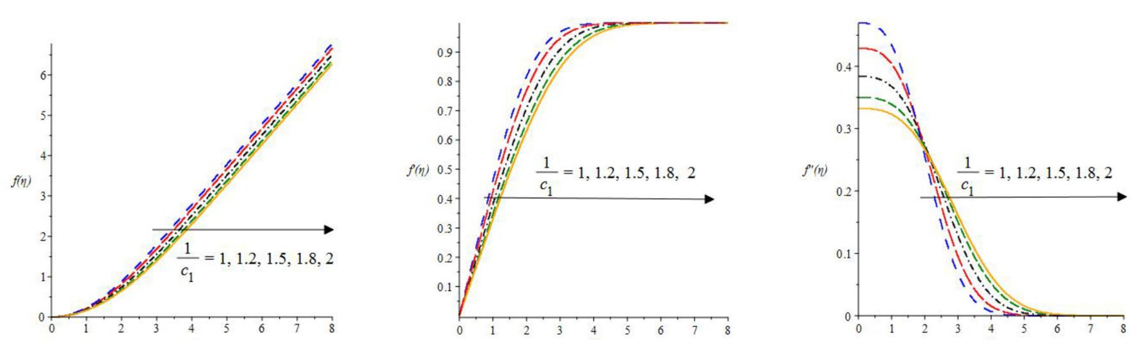
(d) MHD-Falkner-Skan model ($c_1 = 1, c_2 = -3, \sqrt{c_3} = 50$)

Fig. 2 The log scale of MSE for Dual method in different models

expansion and they satisfied the Mercer's theorem. Moreover, we converted the nonlinear model to a linear model by applying QLM method and showed that linearization of the model in Primal and Dual forms yields better performance than the nonlinear method and Dual QLM method has the best run time comparing to other methods.

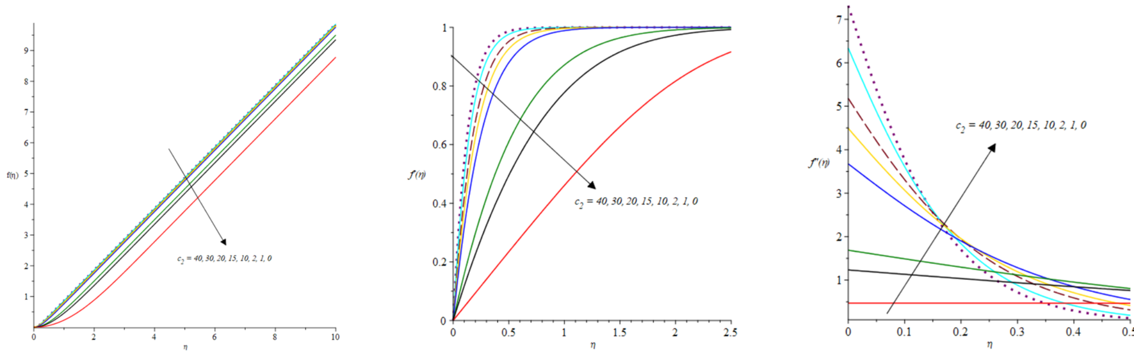
Table 3 Comparison between present methods for well-known Blasius equation with Runge–Kutta method[48]

	η	Nonlinear		QLM		Runge–Kutta
		Primal	Dual	Primal	Dual	[48]
$f(\eta)$	0	0.00021	0.00000	0.00000	0.00000	0.00000
	1	0.16622	0.16557	0.16557	0.16557	0.16557
	2	0.65103	0.65002	0.65002	0.65002	0.65003
	3	1.39799	1.39680	1.39681	1.39681	1.39682
	4	2.30688	2.30574	2.30575	2.30575	2.30576
	5	3.28420	3.28327	3.28328	3.28328	3.28330
	6	4.28027	4.27961	4.27962	4.27962	4.27965
	7	5.27959	5.27923	5.27924	5.27924	5.27927
	8	6.27925	6.27920	6.27922	6.27922	6.27925
	9	7.27893	7.27920	7.27922	7.27922	7.27925
$f'(\eta)$	0	0.00045	0.00000	0.00000	0.00000	0.00000
	1	0.33020	0.32978	0.32978	0.32978	0.32978
	2	0.63005	0.62976	0.62976	0.62976	0.62977
	3	0.84610	0.84604	0.84605	0.84605	0.84605
	4	0.95537	0.95552	0.95552	0.95552	0.95552
	5	0.99129	0.99154	0.99154	0.99154	0.99155
	6	0.99868	0.99897	0.99897	0.99897	0.99898
	7	0.99961	0.99992	0.99992	0.99992	0.99993
	8	0.99968	0.99999	1.00000	1.00000	1.00000
	9	0.99968	1.00000	1.00000	1.00000	1.00000
$f''(\eta)$	0	0.33206	0.33206	0.33206	0.33206	0.33206
	1	0.32293	0.32302	0.32305	0.32305	0.32301
	2	0.26656	0.26675	0.26675	0.26675	0.26675
	3	0.16113	0.16136	0.16136	0.16136	0.16136
	4	0.06408	0.06423	0.06423	0.06423	0.06423
	5	0.01584	0.01590	0.01591	0.01591	0.01591
	6	0.00238	0.00240	0.00240	0.00240	0.00240
	7	0.00021	0.00022	0.00022	0.00022	0.00022
	8	0.00000	0.00001	0.00001	0.00001	0.00001
	9	0.00000	0.00000	0.00000	0.00000	0.00000



(a) A graph of $f(\eta)$ **(b)** A graph of $f'(\eta)$ **(c)** A graph of $f''(\eta)$

Fig. 3 Graphical results of Blasius model for Dual QLM method

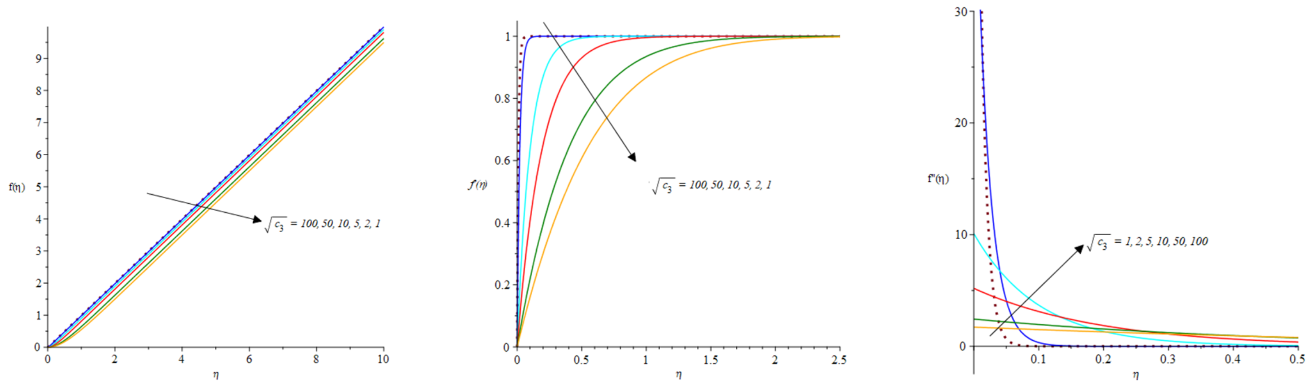


(a) A graph of $f(\eta)$ **(b)** A graph of $f'(\eta)$ **(c)** A graph of $f''(\eta)$

Fig. 4 Graphical results of Falkner–Skan model for Dual QLM method

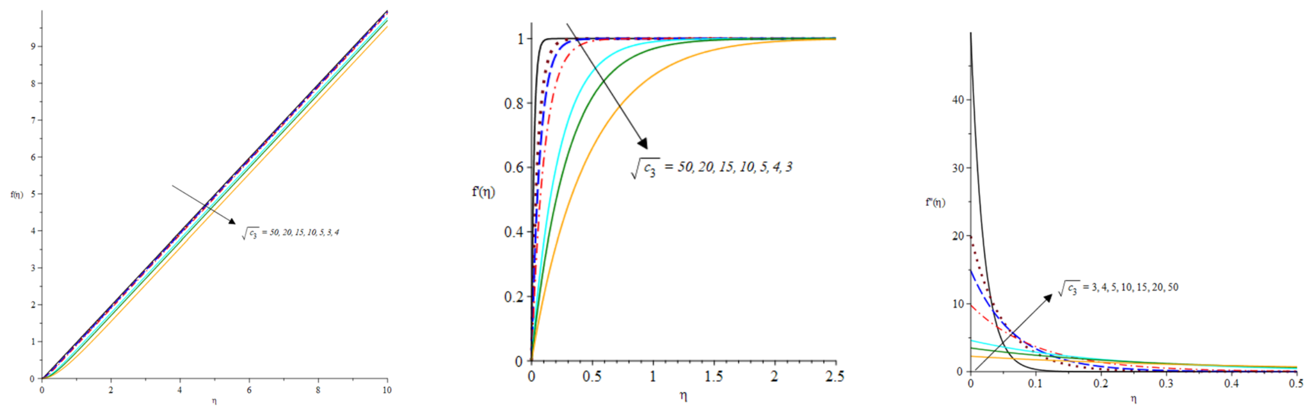
Table 4 Comparison of values of $f''(0)$ obtained by present methods for all samples of general Falkner–Skan model with previous results

Model	$1/c_1$	Nonlinear		QLM		Runge–Kutta[48]	HAM[46]
		Primal	Dual	Primal	Dual		
Blasius ($c_2 = 0, c_3 = 0$)	1	0.46960	0.46960	0.46960	0.46960	0.46960	0.46960
	1.2	0.42868	0.42868	0.42868	0.42868	0.42868	0.42868
	1.5	0.38342	0.38342	0.38342	0.38342	0.38342	0.38342
	1.8	0.35002	0.35002	0.35002	0.35002	0.35002	0.35002
	2	0.33206	0.33206	0.33206	0.33206	0.33206	0.33206
	c_2	Primal	Dual	Primal	Dual	Taylor[56]	Higher-order[58]
Falkner–Skan ($c_1 = 0, c_3 = 0$)	1	1.232588	1.232588	1.232588	1.232588	1.232589	1.232588
	2	1.687218	1.687218	1.687218	1.687218	1.687218	1.687218
	10	3.675234	3.675234	3.675234	3.675234	3.675234	3.675234
	15	4.491487	4.491487	4.491487	4.491487	4.491487	4.491487
	20	5.180718	5.180718	5.180718	5.180718	5.180718	5.180718
	30	6.338209	6.338209	6.338209	6.338209	6.338209	6.338208
	40	7.314785	7.314785	7.314785	7.314785	7.314785	7.314785
	$\sqrt{c_3}$	Primal	Dual	Primal	Dual	Shooting[51]	HAM[60]
MHD-Falkner–Skan ($c_1 = 1, c_2 = 4/3$)	1	1.71946570	1.71946503	1.71946568	1.71946568	1.71946540	1.71947219
	2	2.43949896	2.43949933	2.43949896	2.43949896	2.43949833	2.43949870
	5	5.19095980	5.19095982	5.19095980	5.19095980	5.19095945	5.19095980
	10	10.09677575	10.09677575	10.09677575	10.09677575	10.09677545	10.09677575
	50	50.01944084	50.01944084	50.01944084	50.01944084	50.01944071	50.01944084
	100	100.00972177	100.00972177	100.00972177	100.00972177	100.00972170	100.00972177
	$\sqrt{c_3}$	Primal	Dual	Primal	Dual	Shooting[51]	HAM[60]
MHD-Falkner–Skan ($c_1 = 1, c_2 = -3$)	3	2.27338482	2.27338454	2.27338480	2.27338480	2.27338836	2.27338419
	4	3.48814584	3.48814599	3.48814584	3.48814584	3.48814857	3.48814572
	5	4.60075228	4.60075232	4.60075228	4.60075228	4.60075494	4.60075228
	10	9.80646299	9.80646299	9.80646300	9.80646300	9.80646420	9.80646300
	15	14.87167401	14.87167401	14.87167401	14.87167401	14.87167484	14.87167401
	20	19.90393626	19.90393626	19.90393626	19.90393626	19.90393701	19.90393626
	50	49.96165198	49.96165198	49.96165198	49.96165198	49.96165233	49.96165198



(a) A graph of $f(\eta)$ **(b)** A graph of $f'(\eta)$ **(c)** A graph of $f''(\eta)$

Fig. 5 Graphical results of MHD Falkner–Skan model when $c_2 = \frac{4}{3}$ for Dual QLM method



(a) A graph of $f(\eta)$ **(b)** A graph of $f'(\eta)$ **(c)** A graph of $f''(\eta)$

Fig. 6 Graphical results of MHD Falkner–Skan model when $c_2 = -3$ for Dual QLM method

Acknowledgements The authors are very grateful to both reviewers for carefully reading this paper and for their comments and suggestions which have improved the paper.

References

1. Rad JA, Parand K, Ballestra LV (2015) Pricing European and American options by radial basis point interpolation. Appl Math Comput 251:363–377
2. Blasius HZ (1908) Grenzschichten in flussigkeiten mit kleiner reibung. Math Phys 1–37
3. Mohammadi V, Dehghan M, Khodadadian A, Wick T (2019) Numerical investigation on the transport equation in spherical coordinates via generalized moving least squares and moving kriging least squares approximations. Eng Comput 1–19. https://doi.org/10.1007/s00366-019-00881-3
4. Yonthanthum W, Rattana A, Razzaghi M (2018) An approximate method for solving fractional optimal control problems by the hybrid of block-pulse functions and Taylor polynomials. Optim Control Appl Methods 39(2):873–887
5. Suykens JA, Vandewalle J (1999) Least squares support vector machine classifiers. Neural Process Lett 9:293–300
6. Raissi M, Perdikaris P, Karniadakis GE (2019) Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J Comput Phys 378:686–707
7. Mashayekhi S, Razzaghi M (2018) An approximate method for solving fractional optimal control problems by hybrid functions. J Vib Control 24(9):1621–1631
8. Dehghan M, Abbaszadeh M (2019) Error analysis and numerical simulation of magnetohydrodynamics (MHD) equation based on the interpolating element free Galerkin (IEFG) method. Appl Numer Math 137:252–273
9. Guo BY (2000) Jacobi spectral approximation and its applications to differential equations on the half line. J Comput Math 18:95–112
10. Parand K, Delkhosh M (2017) Accurate solution of the Thomas–Fermi equation using the fractional order of rational Chebyshev functions. J Comput Appl Math 317:624–642

11. Abbaszadeh M, Dehghan M (2019) The interpolating element-free Galerkin method for solving Korteweg-de Vries-Rosenau-regularized long-wave equation with error analysis. Nonlinear Dyn 96(2):1345–1365
12. Mohammadi V, Dehghan M (2019) Simulation of the phase field Cahn–Hilliard and tumor growth models via a numerical scheme: element-free Galerkin method. Comput Methods Appl Mech Eng 345:919–950
13. Dehestani H, Ordokhani Y, Razzaghi M (2018) Fractional-order Legendre–Laguerre functions and their applications in fractional partial differential equations. Appl Math Comput 336:433–453
14. Keshavarz E, Ordokhani Y, Razzaghi M (2018) The Taylor wavelets method for solving the initial and boundary value problems of Bratu-type equations. Appl Numer Math 128:205–216
15. Mehrkanoon S, Suykens JA (2013) LS-SVM based solution for delay differential equations. J Phys Conf Ser 410:012–041
16. Mall S, Chakraverty S (2016) Application of Legendre neural network for solving ordinary differential equations. Appl Soft Comput 43:347–356
17. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20:273–297
18. Suykens JAK, Gestel TV, Brabanter JD, Moor BD, Vandewalle J (2002) Least squares support vector machines. World Scientific Publishing, London
19. Vapnik V (2013) The nature of statistical learning theory. Springer Science & Business Media, New York
20. Lu Y, Yin Q, Li H, Sun H, Yang Y, Hou M (2019) The LS-SVM algorithms for boundary value problems of high-order ordinary differential equations. Adv Differ Equ 1–22. https://doi.org/10.1186/s13662-019-2131-3
21. Leake C, Johnston H, Smith L, Mortari D (2019) Theory of connections applied to support vector machines to solve differential equations. arXiv:1812.05571
22. Sharma G, Panwar A, Nasiruddin I, Bansal RC (2018) Non-linear LS-SVM with RBF-kernel-based approach for AGC of multi-area energy systems. IET Gener Transm Distrib 12:3510–3517
23. Baymani M, Teymoori O, Razavi SG (2016) Method for solving differential equations. Am J Comput Sci Inf Eng 3:1–6
24. Mehrkanoon S, Falck T, Suykens JA (2012) Approximate solutions to ordinary differential equations using least squares support vector machines. IEEE Trans Neural Netw Learn Syst 23:1356–1367
25. Yu J, Li F, Yu S, Wu Z (2018) The approximate solutions to source inverse problem of 1-D convection-diffusion equation by lS-SVM. Inverse Prob Sci Eng 26:677–690
26. Moghaddam VH, Hamidzadeh J (2016) New Hermite orthogonal polynomial kernel and combined kernels in support vector machine classifier. Pattern Recognit 60:921–935
27. Padierna LC, Carpio M, Rojas-Domínguez A, Puga H, Fraire H (2018) A novel formulation of orthogonal polynomial kernel functions for SVM classifiers: the Gegenbauer family. Pattern Recognit 84:211–225
28. Pan Z-B, Chen H, You X-H (2012) Support vector machine with orthogonal Legendre kernel. Int Conf Wavelet Anal Pattern Recognit 125–130. https://doi.org/10.1109/ICWAPR.2012.6294766
29. Ozer S, Chen CH, Cirpan HA (2011) A set of new Chebyshev kernel functions for support vector machine pattern classification. Pattern Recognit 44:1435–1447
30. Ye N, Sun R, Liu Y, Cao L (2006) Support vector machine with orthogonal Chebyshev kernel. 18th Int Conf Pattern Recognit 2:752–755
31. Zanaty E, Afifi A (2018) Generalized Hermite kernel function for support vector machine classifications. Int J Comput Appl 1–9. https://doi.org/10.1080/1206212X.2018.1489571
32. Canuto C, Hussaini MY, Quarteroni A, Zang TA (1987) Spectral methods in fluid dynamic. Springer, New York
33. Delkhosh M, Parand K, Ganji DD (2019) An efficient numerical method to solve the boundary layer flow of an Eyring–Powell non-Newtonian fluid. J Appl Comput Mech 5(2):454–467
34. Parand K, Dehghan M, Rezaei AR, Ghaderi SM (2010) An approximation algorithm for the solution of the nonlinear Lane–Emden type equations arising in astrophysics using Hermite functions collocation method. Comput Phys Commun 181(6):1096–1108
35. Coulaud O, Funaro D, Kavian O (1990) Laguerre spectral approximation of elliptic problems in exterior domains. Comput Methods Appl Mech Eng 80:451–458
36. Alıcı H, Taşeli H (2015) The Laguerre pseudospectral method for the radial Schrödinger equation. Appl Numer Math 87:87–99
37. Guo BY (2001) Gegenbauer approximation and its applications to differential equations with rough asymptotic behaviors at infinity. Appl Numer Math 38:403–425
38. Boyd JP (2000) Chebyshev and Fourier spectral methods, 2nd edn. Dover, New York
39. Christov C (1982) A complete orthogonal system of functions in $l^2(-\infty, \infty)$ space. SIAM J Appl Math 42:1337–1344
40. Azarnavid B, Parand K (2018) An iterative reproducing kernel method in Hilbert space for the multi-point boundary value problems. J Comput Appl Math 328:151–163
41. Szegö G (1975) Orthogonal polynomials, 4th edn. AMS Colloquium Publications, New York
42. Falkner VM, Skan SW (1931) Some approximate solutions of the boundary-layer equations. Phiols Mag 12:865–896
43. Bender CM, Milton KA, Pinsky SS, Simmons JLM (1989) A new perturbative approach to nonlinear problems. J Math Phys 30:1447–1455
44. He JH (1998) Approximate analytical solution of Blasius equation. Commun Nonlinear Sci Numer Simul 3:260–263
45. Liao S (1998) An explicit, totally analytic solution of laminar viscous flow over a semi-infinite flat plate. Commun Nonlinear Sci Numer Simul 3:53–57
46. Howarth L (1938) On the solution of the laminar boundary layer equations. Proc R Soc Lond A 164:547–579
47. Hashim I (2006) Comments on new algorithm for solving classical Blasius equation by l. wang. Appl Math Comput 176:700–703
48. Cortell R (2005) Numerical solutions of the classical Blasius flat-plate problem. Appl Math Comput 170:706–710
49. Parand K, Nikarya M, Rad JA, Baharifard F (2012) A new reliable numerical algorithm based on the first kind of bessel functions to solve Prandtl–Blasius laminar viscous flow over a semi-infinite flat plate. Z Naturforsch A 67:665–673
50. Parand K, Dehghan M, Baharifard F (2013) Solving a laminar boundary layer equation with the rational Gegenbauer functions. Appl Math Model 37:851–863
51. Asaithambi NS (1997) A numerical method for the solution of the Falkner–Skan equation. Appl Math Comput 81:259–264
52. Temimi H, Ben-Romdhane M (2018) Numerical solution of Falkner–Skan equation by iterative transformation method. Math Model Anal 23:139–151
53. Zaraki A, Ghalambaz M, Chamkha AJ, Ghalambaz M, De Rossi D (2015) Theoretical analysis of natural convection boundary layer heat and mass transfer of nanofluids: effects of size, shape and type of nanoparticles, type of base fluid and working temperature. Adv Powder Technol 26(3):935–946
54. Magyari E, Chamkha AJ (2010) Combined effect of heat generation or absorption and first-order chemical reaction on micropolar fluid flows over a uniformly stretched permeable surface: The full analytical solution. Int J Therm Sci 49(9):1821–1828
55. Liao SJ (1999) A uniformly valid analytic solution of two-dimensional viscous flow over a semi-infinite flat plate. J. Fluid. Mech. 358:101–128

56. Asaithambi A (2005) Solution of the Falkner–Skan equation by recursive evaluation of Taylor coefficients. J Comput Appl Math 176:203–214
57. Kuo BL (2003) Application of the differential transformation method to the solutions of Falkner–Skan wedge flow. Acta Mech 164:161–174
58. Salama A (2004) Higher-order method for solving free boundary-value problems. Numer Heat Transf Part B Fundam 45:385–394
59. Hayat T, Hussain Q, Javed T (2009) The modified decomposition method and Padé approximants for the MHD flow over a non-linear stretching sheet. Nonlinear Anal Real World Appl 10:966–973
60. Abbasbandy S, Hayat T (2009) Solution of the MHD Falkner–Skan flow by Homotopy analysis method. Commun Nonlinear Sci Numer Simul 14:3591–3598
61. Gorder RAV, Vajravelu K (2010) Existence and uniqueness results for a nonlinear differential equation arising in MHD Falkner–Skan flow. Commun Nonlinear Sci Numer Simul 15:2272–2277
62. Takhar HS, Chamkha AJ, Nath G (2001) Unsteady three-dimensional MHD-boundary-layer flow due to the impulsive motion of a stretching surface. Acta Mech 146(1–2):59–71
63. Takhar HS, Chamkha AJ, Nath G (2002) MHD flow over a moving plate in a rotating fluid with magnetic field, Hall currents and free stream velocity. Int J Eng Sci 40(13):1511–1527
64. Khedr M-E, Chamkha AJ, Bayomi M (2009) MHD flow of a micropolar fluid past a stretched permeable surface with heat generation or absorption. Nonlinear Anal Model 14(1):27–40
65. Al-Mudhaf A, Chamkha AJ (2005) Similarity solutions for MHD thermosolutal Marangoni convection over a flat surface in the presence of heat generation or absorption effects. Heat Mass Transf 42(2):112–121
66. Chamkha AJ (1997) Hydromagnetic natural convection from an isothermal inclined surface adjacent to a thermally stratified porous medium. Int J Eng Sci 35(10–11):975–986
67. Chamkha AJ, Khaled A-RA (2000) Hydromagnetic combined heat and mass transfer by natural convection from a permeable surface embedded in a fluid-saturated porous medium. Int J Numer Methods Heat Fluid Flow 10(5):455–477
68. Naseri R, Malek A, Gorder RAV (2015) On existence and multiplicity of similarity solutions to a nonlinear differential equation arising in magnetohydrodynamic Falkner'Skan flow for decelerated flows. Math Methods Appl Sci 38:4272–4278
69. Reddy PS, Chamkha AJ (2016) Soret and Dufour effects on MHD convective flow of Al2O3–water and TiO2–water nanofluids past a stretching sheet in porous media with heat generation/absorption. Adv Powder Technol 27(4):1207–1218
70. Chamkha AJ, Al-Mudhaf A (2005) Unsteady heat and mass transfer from a rotating vertical cone with a magnetic field and heat generation or absorption effects. Int J Therm Sci 44(3):267–276
71. Boyd JP (1982) The optimzation of convergence for Chebyshev polynomial methods in an unbounded domain. J Comput Phys 45:43–79
72. Scholkopf B, Smola AJ (2001) Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT Press, Cambridge
73. Parand K, Hajimohammadi Z (2018) Using modified generalized Laguerre functions, QLM and collocation method for solving an Eyring–Powell problem. J Braz Soc Mech Sci Eng 40(4):182–190
74. Basto M, Semiao V, Calheiros FL (2006) A new iterative method to compute nonlinear equations. Appl Math Comput 173:468–483