ỨNG DỤNG PHƯƠNG PHÁP HỌC MÁY TÍNH TOÁN CHIỀU DÀI NƯỚC NHẢY TRONG KÊNH LĂNG TRỤ MẶT CẮT HÌNH CHỮ NHẬT

Hồ Việt Hùng
Trường Đại học Thủy lợi

Tóm tắt: *Chiều dài nước nhảy là một đặc trưng quan trọng cần được tính toán chính xác vì nó ảnh hưởng trực tiếp đến chiều dài bể tiêu năng. Vì vậy, mục đích của nghiên cứu này là phát triển và đánh giá sáu mô hình học máy, gồm có: Cây quyết định (Decision Tree – DT), Rừng cây ngẫu nhiên (Random Forest - RT), Tăng cường thích ứng (Adaptive Boosting – Ada), Tăng cường độ dốc (Gradient Boosting - GB), Cây bổ sung (Extra Trees - ET) và Máy Vector hỗ trợ (Support Vector Machine – SVM). Nghiên cứu này đã sử dụng Định lý π-Buckingham để tìm năm tham số không thứ nguyên phục vụ cho các mô hình học máy và ứng dụng các mô hình này để đánh giá mức độ ảnh hưởng của các biến độc lập đến biến mục tiêu. Phương pháp học máy cho thấy hiệu quả vượt trội so với phương pháp công thức kinh nghiệm. Các mô hình học máy có xét đến ảnh hưởng của độ nhám và chiều rộng lòng dẫn, tính nhớt của chất lỏng, có sai số dự báo nhỏ hơn so với các công thức kinh nghiệm. Mô hình ET cho kết quả tốt nhất với hệ số Nash đạt 0.99, sau đó là Ada, RF, GB, DT, SVR, theo thứ tự giảm dần. Kết quả nghiên cứu cho thấy mô hình ET có thể thay thế các công thức kinh nghiệm trong việc tính toán chiều dài nước nhảy trong kênh lăng trụ đáy bằng có mặt cắt chữ nhật.*
Từ khóa: *Nước nhảy, Buckingham, học máy, mô hình, Froude.*

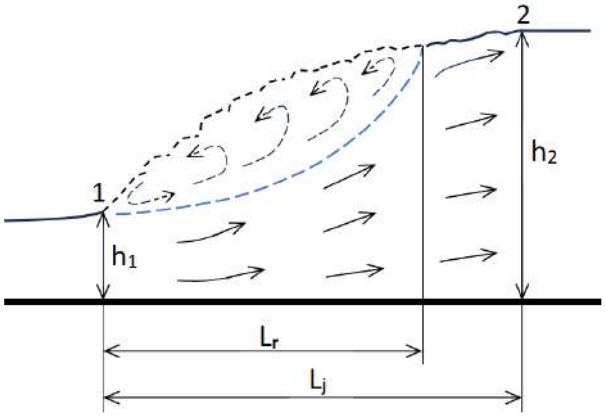
Summary: *The length of the hydraulic jump is an important characteristic that needs to be calculated accurately because it directly affects the length of the energy dissipator. Therefore, the purpose of this study is to develop and evaluate six machine learning models, including Decision Tree (DT), Random Forest (RT), Adaptive Boosting (Ada), Gradient Boosting (GB), Extra Trees (ET), and Support Vector Machine (SVM). This study used the Buckingham Theorem to identify five dimensionless parameters for machine learning models, which were then utilized to assess the influence of independent variables on the target variable. The machine learning method shows superior performance compared to the empirical formula method. Machine learning models that consider the effects of channel surface roughness, channel width, and fluid viscosity produce lower prediction errors than empirical equations. The model ET performs best, with a Nash coefficient of 0.99, followed by Ada, RF, GB, DT, and SVR in descending order. According to the research findings, instead of using empirical equations, the model ET can be used to calculate the hydraulic jump length in a horizontal prismatic channel with a rectangular cross-section.*
Keywords: *Hydraulic jump, Buckingham, machine learning, model, Froude.*

1. GIỚI THIỆU CHUNG

Nước nhảy thường xảy ra sau đập tràn hoặc cửa cống lộ thiên, khi dòng chảy chuyển từ trạng thái chảy xiết sang chảy êm. Vận tốc dòng chảy và số Froude giảm đột ngột từ trước nước nhảy đến sau nước nhảy. Một đặc trưng hình học quan trọng của nước nhảy là chiều dài nước nhảy, cần được tính toán chính xác vì nó ảnh hưởng trực tiếp đến chiều dài bể tiêu năng và kích thước công trình. Cho đến nay, chiều dài nước nhảy được tính toán bằng các công thức kinh nghiệm, không có phương trình thuần túy lý thuyết cho việc này. Các công thức kinh nghiệm có ưu điểm là đơn giản, dễ sử dụng. Chỉ cần biết độ sâu và vận tốc trước nước nhảy hoặc hai độ sâu nước nhảy là tính được chiều dài của nó. Các nhà khoa học như Chertausov (1935), Pikalov (1954), Silvester (1964), Hager (1992) đã đề xuất các công thức tính tỷ số chiều dài với độ sâu trước nước nhảy, gọi là chiều dài tương đối của nước nhảy, theo số Froude trước nước nhảy trong kênh chữ nhật nằm ngang (Hager, 1992;

Ngày nhận bài: 22/02/2024
Ngày thông qua phản biện: 10/4/2024
Ngày duyệt đăng: 30/5/2024

Mammadov, 2017; Silvester R., 1964) (Brakeni et al., 2021) [5; 12; 16; 3]. Các công thức này không cần độ sâu sau nước nhảy, giúp cho việc tính toán đơn giản mà vẫn đảm bảo độ chính xác, vì độ sâu sau nước nhảy có thể được tính từ độ sâu và số Froude trước nước nhảy. Tuy nhiên, các công thức kinh nghiệm có hạn chế là: không đồng nhất nên dẫn đến các kết quả khác nhau; một số trường hợp có sai số lớn với sai số trung bình lên đến 27% (xem Bảng 5); không xét đến ảnh hưởng của chiều rộng và độ nhám lòng dẫn, tính nhớt của chất lỏng. Vì vậy, cần có một phương pháp khác để khắc phục những hạn chế trên và tính toán chính xác hơn chiều dài nước nhảy trong kênh chữ nhật nằm ngang. Hình 1 minh họa các đặc trưng hình học của nước nhảy, trong đó: L_r là chiều dài khu xoáy; L_j là chiều dài nước nhảy; h_1 là độ sâu trước nước nhảy; h_2 là độ sâu sau nước nhảy.



Hình 1: Các đặc trưng hình học của nước nhảy

Hiện nay, các thuật toán học máy (Machine Learning – ML) đã và đang được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau, bao gồm tài nguyên nước nói chung và thủy lực nói riêng (Ho et al., 2022; Truong et al., 2021) [7; 17]. Các mô hình ML thuộc nhóm các mô hình dựa trên cơ sở dữ liệu, đã được áp dụng để nghiên cứu các thông số của nước nhảy từ năm 2012 (Abbaspour et al., 2013; Naseri & Othman, 2012) [1; 13]. Những mô hình này sử dụng mối quan hệ thống kê giữa dữ liệu đầu vào và đầu ra để đưa ra dự báo. Việc ứng dụng các mô hình ML cho hiệu quả tốt trong nghiên cứu các vấn đề của cơ học chất lỏng và thủy lực, hỗ trợ các mô hình vật lý để giải quyết các bài toán thực tế (Brunton et al., 2020) [4]. Các mô hình toán dựa trên ML đã cho kết quả tương đối tốt khi tính toán các đặc trưng hình học của nước nhảy (Baharvand et al., 2021; Houichi et al., 2013; Khosravinia et al., 2018) [2; 8; 10]. Các thuật toán ML như ANFIS (adaptive neuro-fuzzy inference system), ANFIS-PSO (ANFIS-particle swarm optimization), LASSO (least absolute shrinkage and selection operator) đã được sử dụng để tính toán độ sâu liên hiệp của nước nhảy (Baharvand et al., 2021) [2]. Bên cạnh đó, các mô hình: mạng nơ-ron nhân tạo (ANN), GEP (gene expression programming), MARS (multivariate adaptive regression spline), DENFIS (dynamic evolving neural-fuzzy inference system), SVM (support vector machine) cũng được ứng dụng để giải quyết các bài toán thủy lực và kinh tế (Kisi et al., 2019) [11]. Hơn thế nữa, các mô hình ML được sử dụng nhiều trong lĩnh vực quản lý nguồn nước nhằm dự báo mực nước mặt và nước ngầm, gồm có: RF (random forest – rừng cây ngẫu nhiên), GB (gradient boosting - tăng cường độ dốc) và ET (extra trees - cây bổ sung). Phần lớn các thuật toán ML này đều phục vụ cho bài toán hồi quy, thuộc nhóm học máy có giám sát (Kenda et al., 2020; Rezaee et al., 2023) [9; 15].

Vì những nguyên nhân kể trên, mục đích của nghiên cứu này là phát triển và đánh giá khả năng dự báo của 6 mô hình ML, gồm Cây quyết định (Decision Tree – DT), Rừng cây ngẫu nhiên (Random Forest - RT), Tăng cường thích ứng (Adaptive Boosting – Ada), Tăng cường độ dốc (Gradient Boosting - GB), Cây bổ sung (Extra Trees - ET) và Máy Vector hỗ trợ (Support Vector Machine – SVM). Kết quả dự báo của sáu mô hình này sẽ được so sánh với bốn công thức kinh nghiệm nhằm tìm ra mô hình hiệu quả nhất cho việc tính toán chiều dài nước nhảy trong kênh lăng trụ đáy bằng có mặt cắt chữ nhật.

2. CÁC DỮ LIỆU VÀ PHƯƠNG PHÁP NGHIÊN CỨU

2.1. Các dữ liệu cho mô hình toán

Nghiên cứu này đã thu thập dữ liệu từ thí

nghiệm của (Peterka, 1984) [14] được công bố trong các báo cáo kỹ thuật của Bộ Nội vụ Hoa Kỳ, Cục Khai hoang (U.S. Department of the Interior, Bureau of Reclamation - USBR). Tất cả các thí nghiệm đã được thực hiện trên sáu máng hình chữ nhật có kích thước khác nhau, là các máng A, B, C, D, E và F, với lưu lượng dòng chảy dao động từ 1 đến 28 cfs. Các máng A, B, C, D, E tạo ra nước nhảy sau chân dốc của đập tràn. Trong khi đó, máng F tạo nước nhảy sau cửa cống phẳng, đáy cống nằm ngang. Các kích cỡ và cách sắp xếp máng khác nhau giúp xác định ảnh hưởng của chiều rộng máng (b) và góc dòng chảy đi vào nước nhảy. Các thí nghiệm có nhiều thông số được liệt kê trong Bảng 1, cho phép quan sát nước nhảy với các kích cỡ khác nhau. Các máng có tường bên làm bằng kính để tiện theo dõi thí nghiệm. Do đó theo Hager, độ nhám tuyệt đối của lòng dẫn mô hình là e = 0.005 mm (Hager & Bremen, 1989) [6]. Các thí nghiệm ở nhiệt độ khoảng 18 °C, hệ số nhớt động học của nước là $\upsilon = 1.1*10^{-6}$ m^2/s. Số Reynolds và số Froude tại mặt cắt (1) trước nước nhảy được tính theo các công thức (1) và (2).

$$\mathrm{Re}_1^* = \frac{V_1 h_1}{\upsilon} \quad (1)$$

$$Fr_1 = \frac{V_1}{\sqrt{g h_1}} \quad (2)$$

Trong đó: h_1 – độ sâu trước nước nhảy (xem Hình 1); V_1 – vận tốc trung bình tại mặt cắt trước nước nhảy; υ - hệ số nhớt động học; g – gia tốc trọng trường.

Bảng 1: Các thông số của thí nghiệm và các máng kính

Máng thí nghiệm	Trị số	Q (cfs)	Fr_1	Re_1*	h_1/b	e/h_1
A	max	5.00	5.58	85920	0.0228	0.00023
b = 4.92 ft	min	3.00	4.80	51552	0.0147	0.00015
B	max	8.00	12.65	337838	0.1145	0.00028
b = 2.0 ft	min	2.00	6.45	84459	0.0290	0.00007
C	max	4.44	19.67	250000	0.0894	0.00050
b = 1.5 ft	min	1.00	10.21	56306	0.0220	0.00012
D	max	26.16	18.04	603555	0.0733	0.00043
b = 3.97 ft	min	3.00	8.05	63823	0.0096	0.00006
E	max	11.00	5.80	234019	0.0856	0.00017
b = 3.97 ft	min	2.44	1.73	51910	0.0239	0.00005
F	max	2.23	7.64	188345	0.2774	0.00021
b = 1.0 ft	min	0.68	2.24	57432	0.0790	0.00006

Tổng cộng 120 mẫu kết quả thí nghiệm đã được sử dụng cho nghiên cứu này. Bộ dữ liệu này được chia làm hai phần để phục vụ các mô hình ML, phần thứ nhất gồm 96 mẫu (80% số liệu) nhằm mục đích huấn luyện mô hình (training), phần thứ hai gồm 24 mẫu (20% số liệu) để kiểm định mô hình (testing). Thuật toán ML sẽ chọn ngẫu nhiên 24 số liệu kiểm định dùng chung cho tất cả các mô hình nhằm đảm bảo tính khách quan, không phụ thuộc vào ý muốn của người sử dụng mô hình.

2.2. Áp dụng Định lý π-Buckingham

Chiều dài nước nhảy L_j trong Hình 1 phụ thuộc vào các yếu tố sau: độ sâu và vận tốc trung bình tại mặt cắt trước nước nhảy; chiều rộng và độ nhám lòng dẫn; khối lượng riêng và tính nhớt của chất lỏng; gia tốc trọng trường. Mối quan hệ này được thể hiện trong phương trình (3).

$$L_j = f(h_1, V_1, b, \rho, \mu, e, g) \quad (3)$$

Trong đó: b - chiều rộng kênh; ρ - khối lượng riêng của nước; μ – hệ số nhớt của nước; e – độ nhám bề mặt kênh. Hệ số nhớt động học được tính theo công thức: υ = μ / ρ.

Để biểu thị đơn vị đo của tám đại lượng trong phương trình (3) cần đủ ba thứ nguyên cơ bản M, L, T. Theo Định lý π-Buckingham sẽ có năm hàm π thay thế cho tám đại lượng trong

phương trình (3). Để tìm năm hàm π này, ba biến lặp lại sẽ là h_1, V_1, ρ; năm biến không lặp lại sẽ là L_j, e, b, μ, g. Kết quả tính toán, giải một hệ năm phương trình thu được năm hàm π như sau:

$\boldsymbol{\Pi_1 = L_j/h_1 ; \Pi_2 = Fr_1 ; \Pi_3 = Re_1^* ; \Pi_4 = e/h_1 ; \Pi_5 = h_1/b.}$

Như vậy, tỷ số chiều dài với độ sâu trước nước nhảy, gọi là chiều dài nước nhảy tương đối, được biểu thị qua bốn hàm π như phương trình (4).

$$\frac{L_j}{h_1} = \Phi\left(Fr_1, \mathrm{Re}_1^*, \frac{e}{h_1}, \frac{h_1}{b}\right) \quad (4)$$

2.3. Các công thức kinh nghiệm

Chiều dài nước nhảy L_j phụ thuộc vào nhiều yếu tố như đã trình bày trong phương trình (4), do đó có nhiều dạng công thức kinh nghiệm khác nhau để tính toán nó. Có thể tính L_j theo hai độ sâu của nước nhảy, hoặc chỉ tính gần đúng theo độ sâu sau nước nhảy, hoặc theo hai độ sâu và số Fr_1, hay theo độ sâu h_1, số Fr_1 và số Re_1^*. Bài báo này trình bày các công thức tính L_j theo độ sâu h_1 và số Fr_1. Đó là các công thức của Chertausov (1935), Pikalov (1954), Silvester (1964) và Hager (1992), được thể hiện qua các phương trình dưới đây.

Công thức Chertausov (1935): $$\frac{L_j}{h_1} = 10.3\left(Fr_1 - 1\right)^{0.81} \quad (5)$$

Công thức Pikalov (1954): $$\frac{L_j}{h_1} = 4\sqrt{1 + 2Fr_1^2} \quad (6)$$

Công thức Silvester (1964): $$\frac{L_j}{h_1} = 9.75\left(Fr_1 - 1\right)^{1.01} \quad (7)$$

Công thức Hager (1992): $$\frac{L_j}{h_1} = 220\tanh\left(\frac{Fr_1 - 1}{22}\right) \quad (8)$$

Các công thức trên sẽ được sử dụng để tính toán chiều dài nước nhảy tương đối và so sánh với kết quả dự báo của sáu mô hình ML.

2.4. Các thuật toán ML

Mục này trình bày tổng quát về sáu mô hình ML được sử dụng để tính toán chiều dài nước nhảy tương đối trong nghiên cứu này.

2.4.1. Mô hình cây quyết định (Decision Tree - DT)

Mô hình cây quyết định (DT) là một mô hình được sử dụng khá phổ biến và hiệu quả trong bài toán dự báo của học máy có giám sát. Khác với những thuật toán khác trong học có giám sát, mô hình cây quyết định không tồn tại phương trình dự báo. Chúng ta cần tìm ra một cây quyết định dự báo tốt trên tập huấn luyện và sử dụng cây quyết định này dự báo trên tập kiểm tra. Các tiêu chí để lựa chọn biến phù hợp là các độ đo như entropy, Gini đo lường mức độ tinh khiến (purity) và vẩn đục (impurity) của một biến nào đó. Chỉ số gini được sử dụng trong thuật toán CART (Classification And Regression Tree) của sklearn. Đây là thuật toán được sử dụng phổ biến nhất trong học máy. Ưu điểm của thuật toán này là có thể sử dụng cho cả bài toán phân loại và hồi qui.

Ký hiệu x_i là quan sát thứ i của tập S, bao gồm m chiều tương ứng với số lượng biến đầu vào; k là số lượng tập con của tập S; S_j là phương sai của biến mục tiêu y_i tại node S. Thuật toán sẽ tìm cách lựa chọn x_i và ngưỡng phân chia sao cho độ suy giảm phương sai là lớn nhất. Khi đó, các quan sát được phân về cùng một node lá sẽ có giá trị dự báo gần nhau và một ước lượng chung cho node lá bằng trung bình cộng của biến mục tiêu. Như vậy giá trị ước lượng của một quan sát (x_i, y_i) thuộc về node S_j sẽ bằng trung bình cộng biến mục tiêu của node theo phương trình (9) dưới đây:

$$\hat{y}_i = \frac{1}{|S_j|}\sum_{k=1}^{|S_j|} y_k \quad (9)$$

2.4.2. Mô hình rừng cây ngẫu nhiên (Random Forest - RT)

Dù có độ chính xác khá cao nhưng thuật toán cây quyết định (DT) tồn tại những hạn chế lớn. Sức mạnh của một cây quyết định là không cao thì hợp sức của nhiều cây sẽ trở nên mạnh mẽ hơn. Đó chính là mô hình rừng cây ngẫu nhiên (RT). Vì có độ chính xác cao, giảm thiểu hiện tượng quá khớp (overfitting) nên mô hình RT được sử dụng rộng rãi trong cả hai bài toán phân loại và dự báo của học có giám sát. Mô hình RT được huấn luyện dựa trên sự phối hợp giữa quá trình kết hợp (ensembling) và lấy mẫu tái lặp (boostrapping). Mô hình này tạo ra nhiều DT mà mỗi DT được huấn luyện dựa trên nhiều mẫu con khác nhau và kết quả dự báo là giá trị trung bình thu được từ toàn bộ những DT. Do đó, một kết quả dự báo được tổng hợp từ nhiều mô hình sẽ không bị sai lệch do các DT đều sử dụng bộ dữ liệu huấn luyện chung. Ngoài ra, tập hợp kết quả dự báo từ nhiều mô hình sẽ có phương sai nhỏ hơn và ít bị ảnh hưởng bởi nhiễu so với chỉ từ một mô hình. Trong mô hình RT, những DT là hoàn toàn độc lập với nhau.

Dữ liệu huấn luyện mô hình là một tập D bao gồm N quan sát. Thuật toán RF sẽ sử dụng phương pháp lấy mẫu tái lặp để tạo thành k tập dữ liệu con. Mô hình dự báo có kết quả là giá trị trung bình của các dự báo từ những mô hình con như phương trình (10).

$$\hat{y}_j = \frac{1}{K}\sum_{i=1}^{K}\hat{y}_j^{(i)} \qquad (10)$$

Trong đó: $\hat{y}_j^{(i)}$ là dự báo của quan sát thứ j từ mô hình thứ i, $\hat{y}_j^{(i)} = f_i(x_j)$; x_j là giá trị véc tơ đầu vào; f_i là hàm dự báo của mô hình thứ i; K là số lượng các DT.

2.4.3. Mô hình Ada (Adaptive Boosting)

Thuật toán Ada, viết tắt của "Adaptive Boosting - Tăng cường thích ứng", là một phương pháp tổng hợp lặp đi lặp lại, chủ yếu được sử dụng để tăng hiệu suất của các mô hình phân loại yếu (weak classifiers). Một mô hình phân loại yếu có tỷ lệ dự báo sai lớn và giả định nó chỉ tốt hơn so với phân loại ngẫu nhiên một chút. Nguyên tắc cốt lõi của mô hình Ada là cân nhắc từng mẫu trong tập dữ liệu đầu vào dựa trên các lỗi của lần lặp trước đó. Mô hình Ada áp dụng liên tiếp các mô hình phân loại yếu để điều chỉnh lại trọng số cho các quan sát. Việc điều chỉnh trọng số của mỗi lần lặp nhằm đảm bảo rằng bộ học yếu (weak learner) tiếp theo tập trung nhiều hơn vào các mẫu bị phân loại sai trước đó. Việc điều chỉnh này tiếp tục lặp lại cho đến khi sai số hội tụ về một giá trị nhỏ nhất hoặc đạt được một số cây (DT) nhất định. Như vậy, Ada là một mô hình dự báo được kết hợp từ các mô hình phân loại yếu trong chuỗi. Do tính chất thích ứng của mình, mô hình Ada có hiệu quả tốt trong các dự báo có ranh giới phức tạp giữa các lớp hoặc các bài toán hồi quy phi tuyến. Tiềm năng của mô hình Ada trong việc xác định các mối tương quan phi tuyến phức tạp giữa các yếu tố đầu vào và đầu ra có thể đóng vai trò then chốt trong việc dự báo chính xác. Phương trình hồi quy của Ada có thể được biểu diễn dưới dạng (11).

$$\hat{y}(x) = \sum_{i=1}^{K}\alpha_i \cdot f_i(x) \qquad (11)$$

Trong đó: α_i biểu thị trọng số của cây thứ i, được tính dựa trên sai số của cây đó; x là giá trị véc tơ đầu vào; f_i là hàm dự báo của cây thứ i; K là số lượng các cây.

2.4.4. Mô hình GB (Gradient Boosting)

Thuật toán GB là một thuật toán hiện đại được xây dựng dựa trên Ada. Cũng tương tự như Ada, nó huấn luyện liên tiếp các mô hình yếu. Thuật toán GB kết hợp các DT nhưng các cây không hoàn toàn độc lập mà chúng có sự phụ thuộc theo chuỗi. Tức là một DT được phát triển từ việc sử dụng thông tin được dự báo từ những DT được huấn luyện trước đó. Mô hình GB không sử dụng mẫu tái lặp để tạo dữ liệu huấn luyện mà mô hình được huấn luyện ngay trên dữ liệu gốc. Điểm đặc biệt của mô hình này là thay vì cố gắng khớp giá trị biến mục tiêu, nó sẽ tìm cách khớp giá trị sai số của mô hình trước đó. Sau đó mô hình huấn luyện sẽ được đưa thêm vào hàm dự báo để cập nhật dần phần dư. Thuật toán sẽ dừng cập nhật khi số lượng DT đạt ngưỡng tối đa K, hoặc toàn bộ các quan sát trên tập huấn luyện được dự

báo đúng. Bằng cách khớp trên những DT có kích thước rất nhỏ trên những phần dư, hàm dự báo sẽ từ từ được cải thiện trong vùng mà nó không dự báo tốt. Giống như phương pháp kết hợp, kết quả dự báo từ chuỗi mô hình sẽ là kết hợp của các mô hình con, theo phương trình (12).

$$\hat{y}(x) = \sum_{b=1}^{K} \lambda \hat{f}_b(x) \quad (12)$$

Trong đó: $\hat{y}(x)$ là hàm dự báo từ thuật toán GB; x là ma trận đầu vào; $\hat{f}_b(x)$ là hàm dự báo của mô hình thứ b trong chuỗi mô hình dự báo; λ là hệ số co (shrinkage parameter); K là số lượng cây.

2.4.5. Mô hình ET (Extra Trees)

Thuật toán ET xây dựng một tập hợp các DT hoặc cây hồi quy chưa được cắt tỉa theo quy trình từ trên xuống một cách cổ điển. Nó có hai điểm khác biệt chính so với các phương pháp tổng hợp khác dựa trên DT, đó là nó phân chia các nút bằng cách chọn các điểm cắt hoàn toàn ngẫu nhiên và sử dụng toàn bộ mẫu huấn luyện (chứ không phải bản sao mẫu tái lặp) để tạo cây. Đối với mô hình ET, thủ tục tách (splitting procedure) các thuộc tính số gồm có hai tham số: K - số lượng thuộc tính được chọn ngẫu nhiên tại mỗi nút và n_{min} - cỡ mẫu tối thiểu để tách một nút. Thủ tục này được sử dụng nhiều lần với toàn bộ mẫu huấn luyện ban đầu để tạo ra một mô hình tổng hợp (với M là số cây của tập hợp này). Các dự báo của các DT được tổng hợp để đưa ra dự báo cuối cùng, bằng cách lấy theo đa số trong các bài toán phân loại hoặc trung bình cộng trong các bài toán hồi quy như phương trình (13).

$$\hat{y}_j = \frac{1}{M} \sum_{i=1}^{M} \hat{y}_j^{(i)} \quad (13)$$

Trong đó: $\hat{y}_j^{(i)}$ là kết quả dự báo của quan sát thứ j từ mô hình thứ i, $\hat{y}_j^{(i)} = f_i(x_j)$; x_j là giá trị véc tơ đầu vào; f_i là hàm dự báo của mô hình thứ i; M là số lượng cây.

Từ quan điểm phương sai, yếu tố căn bản của phương pháp ET là sự ngẫu nhiên rõ ràng của điểm giới hạn và thuộc tính kết hợp với tính trung bình tổng thể có thể làm giảm phương sai mạnh hơn so với các sơ đồ ngẫu nhiên yếu hơn được sử dụng trong các phương pháp khác. Việc sử dụng mẫu huấn luyện ban đầu đầy đủ thay vì bản sao mẫu tái lặp được thúc đẩy để làm giảm thiểu sai lệch. Tuy nhiên, do tính đơn giản của quy trình tách nút, hy vọng rằng, hệ số không đổi sẽ nhỏ hơn nhiều so với các phương pháp tổng hợp khác nhằm tối ưu hóa cục bộ các điểm giới hạn.

Các thông số K, n_{min} và M có tác dụng khác nhau: K quyết định cường độ của quá trình lựa chọn thuộc tính, n_{min} quyết định cường độ của nhiễu đầu ra trung bình và M quyết định cường độ giảm phương sai của tập hợp mô hình tổng hợp. Các tham số này có thể được điều chỉnh cho phù hợp với từng bài toán cụ thể bằng cách thủ công hoặc tự động, ví dụ như bằng cách xác thực chéo (cross-validation).

2.4.6. Mô hình SVM (Support Vector Machine)

SVM là một thuật toán khá hiệu quả trong việc phân loại nhị phân và dự báo của học máy có giám sát. Thuật toán này có ưu điểm là hoạt động tốt đối với những mẫu dữ liệu có kích thước lớn và thường mang lại kết quả vượt trội so với các thuật toán khác trong học có giám sát. Nó tiêu tốn ít bộ nhớ vì chỉ sử dụng các điểm trong tập hỗ trợ để dự báo trong hàm quyết định được tạo ra từ những hàm kernel khác nhau. Việc sử dụng đúng hàm kernel có thể giúp cải thiện đáng kể kết quả dự báo của thuật toán. Những hàm kernel phổ biến đã được tích hợp bên trong sklearn gồm có: Kernel RBF dựa trên hàm Gaussian RBF biến đổi phi tuyến; Kernel tuyến tính (linear): đây là tích vô hướng giữa hai véc tơ; Kernel đa thức (poly) tạo ra một đa thức bậc cao kết hợp giữa hai véc tơ; Kernel Sigmoid dựa trên kernel về đa thức, chuyển tiếp qua hàm tanh và có thể biểu diễn theo hàm sigmoid. Trong bài toán dự báo, thuật toán có tên là SVR (Support Vector Regression), kết quả dự báo được thể hiện trong phương trình (14).

$$\hat{y}(x) = \omega \cdot \phi(x) + b \quad (14)$$

Trong đó: ω là véc tơ trọng số, b là độ sai lệch;

Φ(x) là phép biến đổi của véc tơ đầu vào x thông qua hàm kernel.

2.5. Phương pháp đánh giá sai số

Các giá trị của các chỉ số thống kê như: Hệ số Nash (NSE), Sai số tuyệt đối trung bình (MAE), Sai số căn quân phương (RMSE), Sai số tương đối trung bình (MAPE), được sử dụng để đánh giá sai số của kết quả dự báo từ mô hình ML. Các trị số trên được tính toán theo các công thức từ (15) đến (18). Mô hình ML cho kết quả có độ chính xác cao khi giá trị của NSE lớn, gần bằng 1 và các sai số nhỏ, gần bằng 0.

$$NSE = 1 - \frac{\sum_{i=1}^{n}(O_i - P_i)^2}{\sum_{i=1}^{n}(O_i - \overline{O}_i)^2} \quad (15)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}(|P_i - O_i|) \quad (16)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - O_i)^2} \quad (17)$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|O_i - P_i|}{O_i} \quad (18)$$

Trong đó: O_i, $\overline{O}_i$ và P_i lần lượt là trị số thực đo, trị số thực đo trung bình và trị số dự báo tương ứng thứ i; n là số lần tính.

3. LỰA CHỌN THAM SỐ MÔ HÌNH VÀ ĐÁNH GIÁ ẢNH HƯỞNG CỦA CÁC BIẾN

3.1. Lựa chọn các siêu tham số của mô hình ML

Trong nghiên cứu này, sáu mô hình ML đã được lập trình nhờ ứng dụng thư viện phần mềm mã nguồn mở Keras và Scikit-learn cũng như API cấp cao của TensorFlow 2. Hơn nữa, ngôn ngữ lập trình Python 3.7 và một số thư viện nhằm minh họa và quản lý dữ liệu như Numpy, Pandas và Matplotlib, đã được sử dụng.

Chiến lược tìm kiếm bằng lưới (grid search) kết hợp với phương pháp thử dần đã được sử dụng trong nghiên cứu này để điều chỉnh các siêu tham số, đây là một phương pháp được sử dụng rộng rãi nhằm cải thiện độ chính xác và độ tin cậy của kết quả dự báo. Phương pháp này huấn luyện và đánh giá thuật toán ML với từng bộ siêu tham số được xác định trong một lưới do người sử dụng mô hình chỉ định trước. Sáu mô hình ML đã được thử nghiệm để thu được các siêu tham số phù hợp và từ đó tạo ra các mô hình tốt nhất. Kết quả lựa chọn các tham số chính của từng mô hình được thống kê trong Bảng 2.

Bảng 2: Siêu tham số của các mô hình ML

No	Mô hình	Tham số chính	Khoảng giá trị	Giá trị chọn
1	DT	max_depth min_samples_split max_features criterion	1, 2, …, 7, None 2, 3, 4, …, 14 1, 2, 3, 4 “squared_error”, “friedman_mse”, “absolute_error”, “poisson”	None 2 None “squared_error”
2	RF	n_estimators max_depth max_samples max_features criterion	100, 500, 1000, 3000 1, 2, …, 7, None 0.1, 0.2, 0.3 ..., 1.0 1, 2, 3, 4 'squared_error', 'friedman_mse'	1000 None 1.0 4 'squared_error'
3	ET	n_estimators max_depth max_samples max_features min_samples_split criterion	100, 300, 500, 1000 1, 2, …, 7, None 0.1, 0.2, 0.3, …, 1.0 1, 2, 3, 4 2, 3, 4, …, 14 'squared_error', 'friedman_mse'	300 None 0.9 4 2 'squared_error'
4	Ada	n_estimators learning_rate loss	500, 1000, 3000, 5000 0.1, 0.2, 0.3, …, 2.0 'linear', 'square', 'exponential'	1000 0.1 'linear'
5	GB	n_estimators	100, 500, 1000, 3000, 5000	1000

		learning_rate max_depth subsample	0.001, 0.01, 0.1, 1.0, 1.3 1, 2, 3, …, 10 0.1, 0.2, 0.3, …, 1.0	0.1 3 1.0
6	SVR	criterion	Kernel: 'linear', 'poly', 'rbf', 'sigmoid', 'precomputed'	'linear'

3.2. Đánh giá mức độ quan trọng của các biến độc lập

Hiện nay, có hai phương pháp để xác định mức độ ảnh hưởng hay điểm quan trọng của các biến độc lập, bao gồm phương pháp thống kê (đơn giản nhất) và phương pháp tầm quan trọng. Trong nghiên cứu này, phương pháp tầm quan trọng của đặc trưng (feature importance) được sử dụng để tính điểm quan trọng (xem Bảng 3). Điểm quan trọng này được tính toán cho từng thuộc tính trong tập dữ liệu đầu vào, nhờ đó các thuộc tính được xếp hạng và so sánh với nhau. Điểm của từng thuộc tính cho biết mức độ hữu ích hoặc có giá trị của nó trong việc xây dựng cây quyết định của mô hình.

Nghiên cứu này sử dụng năm thuật toán ML, bao gồm DT, RF, ET, Ada và GB, để đánh giá mức độ tác động của các yếu tố thủy lực đến độ dài tương đối của nước nhảy. Kết quả tính toán trong Bảng 3 cho thấy, cả năm thuật toán đều đánh giá số Fr_1 có ảnh hưởng nhiều nhất, vượt trội so với các yếu tố khác. Bốn thuật toán cho rằng số Re_1 có ảnh hưởng thứ hai sau số Fr_1, nhưng ảnh hưởng này không lớn. Chiều rộng và độ nhám tương đối có ảnh hưởng ít nhất. Kết quả này đã giải thích vì sao các công thức thực nghiệm tính chiều dài nước nhảy tương đối chỉ phụ thuộc vào số Fr_1. Đây là cơ sở để sắp xếp thứ tự các biến đầu vào mô hình ML, giúp cho kết quả tính toán chính xác hơn.

Bảng 3: Mức độ ảnh hưởng của các biến độc lập

Biến đầu vào	DT	RF	ET	Ada	GB
Fr_1	0.9914	0.9810	0.83001	0.9693	0.9878
Re_1^*	0.0043	0.0091	0.02640	0.0143	0.0058
h_1/b	0.0027	0.0041	0.04937	0.0038	0.0017
e/h_1	0.0016	0.0057	0.09422	0.0126	0.0048

4. KẾT QUẢ NGHIÊN CỨU VÀ THẢO LUẬN

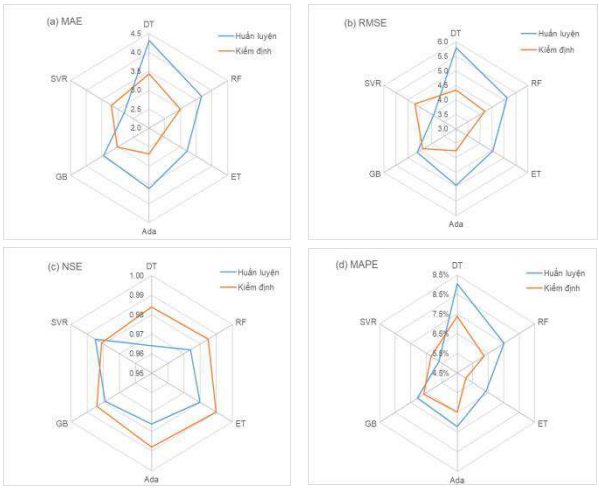
4.1. Kết quả huấn luyện mô hình

Sau khi cập nhật các siêu tham số đã chọn, các mô hình ML được huấn luyện bằng tập dữ liệu huấn luyện. Kết quả đánh giá giai đoạn này dựa trên các chỉ số thống kê được trình bày trong Bảng 4.

Bảng 4: Kết quả huấn luyện sáu mô hình ML

No	Mô hình	MAE	RMSE	NSE	MAPE
1	DT	4.317	5.800	0.964	9.05 %
2	RF	3.672	5.131	0.974	7.52 %
3	ET	3.211	4.533	0.980	6.35 %
4	Ada	3.594	4.955	0.976	7.22 %
5	GB	3.453	4.616	0.979	7.06 %
6	SVR	**2.794**	**3.929**	**0.985**	**5.68 %**

Mô hình SVR có kết quả huấn luyện tốt nhất, với hệ số Nash đạt gần 0.99 và sai số tương đối trung bình MAPE dưới 6 %, sau đó đến mô hình ET với sai số trên 6 %. Mô hình DT có hệ số Nash thấp nhất và sai số tương đối MAPE cao nhất, trên 9 %. Điều này phản ánh đúng tính chất của thuật toán DT. Các thuật toán khác, trừ SVR, đều được cải tiến dựa trên DT nên có kết quả tốt hơn DT. Ba mô hình RF, Ada, GB có kết quả xấp xỉ nhau, với hệ số Nash gần 0.98 và sai số tương đối trên 7 %. Các sai số MAE, RMSE cũng có chung quy luật như sai số tương đối MAPE. Tất cả sáu mô hình đều có hệ số Nash trên 0.95, chứng tỏ kết quả dự báo có độ chính xác cao và rất đáng tin cậy.



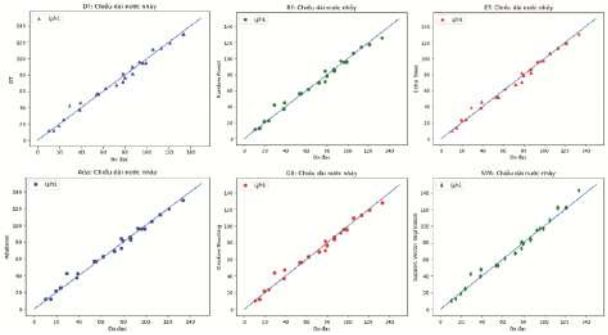
Hình 2: So sánh kết quả huấn luyện và kiểm định mô hình

Kết thúc giai đoạn huấn luyện, các mô hình ML được chuyển sang giai đoạn kiểm định. Hình 2 so sánh kết quả huấn luyện và kiểm định mô hình, cho thấy trong giai đoạn kiểm định, mô hình SVR có kết quả dự báo kém chính xác nhất. Kết quả kiểm định mô hình này có các chỉ số thống kê đều thua kém kết quả huấn luyện. Năm mô hình còn lại cho kết quả dự báo trong giai đoạn kiểm định chính xác hơn khi huấn luyện. Việc này cho thấy hiệu suất cao của năm mô hình DT, RF, ET, Ada và GB trong dự báo chiều dài nước nhảy tương đối.

4.2. Kết quả kiểm định mô hình

Mặc dù có kết quả huấn luyện cao nhất, nhưng mô hình SVR lại có kết quả kiểm định thấp nhất so với năm mô hình ML khác. Điều này hoàn toàn trái ngược với các nhận xét trước đây về thuật toán SVR. Như vậy, có thể thấy rằng kết quả của nghiên cứu này là rất cần thiết cho việc lựa chọn mô hình ML để dự báo chiều dài nước nhảy. Hình 3 so sánh kết quả dự báo của sáu mô hình ML với số liệu thực đo trong thí nghiệm mô hình vật lý. Các điểm trên Hình 3 đều bám sát đường phân giác của góc 90^o, cho thấy độ chính xác của kết quả dự báo và hiệu suất mô hình rất cao.

Bảng 5 thống kê kết quả kiểm định sáu mô hình ML và bốn công thức kinh nghiệm. Điều đáng mừng là tất cả các mô hình toán và công thức kinh nghiệm đều có hệ số Nash từ 0.98 trở lên, chứng tỏ kết quả tính có độ chính xác rất cao. Chính xác nhất là kết quả của mô hình ET với hệ số Nash đạt 0.99 và sai số tương đối MAPE khoảng 5 %. Tiếp theo sau ET là Ada, RF, GB, DT, SVR theo thứ tự độ chính xác giảm dần. Mô hình DT có sai số MAE và MAPE cao nhất trong số các mô hình ML. Tuy vậy, tất cả các mô hình ML đều cho kết quả kiểm định chính xác hơn các công thức kinh nghiệm. Các mô hình ML có trị số RMSE và MAPE thấp hơn, và hệ số Nash cao hơn so với các công thức kinh nghiệm. Cá biệt, các công thức Pikalov và Chertausov có sai số MAPE lên trên 20%, Silvester trên 10%. Công thức Hager có độ chính xác cao hơn ba công thức kể trên. Kết quả kiểm định được thể hiện bằng hình ảnh trong Hình 4. Rõ ràng là các mô hình ML đã làm lu mờ các công thức kinh nghiệm bởi độ chính xác và tính hiệu quả trong dự báo.



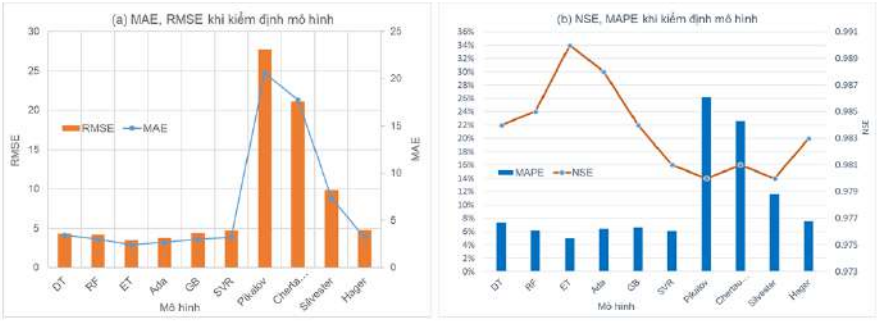
Hình 3: So sánh kết quả kiểm định mô hình ML với số liệu thực đo

Bảng 5: Kết quả kiểm định mô hình ML và công thức kinh nghiệm

No	Mô hình	MAE	RMSE	NSE	MAPE
1	DT	3.433	4.331	0.984	7.39 %
2	RF	3.002	4.213	0.985	6.23 %
3	ET	**2.399**	**3.459**	**0.990**	**5.05 %**
4	Ada	2.687	3.750	0.988	6.48 %
5	GB	3.013	4.380	0.984	6.66 %
6	SVR	3.197	4.710	0.981	6.18 %
7	Pikalov	20.586	27.720	0.980	26.27%
8	Chertausov	17.759	21.129	0.981	22.70%
9	Silvester	7.345	9.860	0.980	11.65%
10	Hager	3.219	4.786	0.983	7.57%

Từ kết quả kiểm định mô hình có thể nhận thấy các ưu điểm của mô hình ML là: có độ chính xác và hiệu suất dự báo cao, xét đến nhiều yếu tố ảnh hưởng, có thể dễ dàng cập nhật khi có thêm dữ liệu mới, dễ tiếp cận, tốc độ tính toán nhanh. Trong các ưu điểm trên thì độ chính xác cao và tính dễ cập nhật là ưu điểm vượt trội của mô hình ML so với các công thức kinh nghiệm. Rõ ràng, các công thức kinh nghiệm không thể cập nhật và phạm vi ứng dụng phụ thuộc vào điều kiện thí nghiệm trên mô hình vật lý.

Tuy nhiên, mô hình ML cũng có hạn chế, đó là sự không kiên định trong các lần dự báo, phụ thuộc vào người sử dụng mô hình. Nghĩa là mỗi lần chạy có thể cho kết quả khác nhau, độ chính xác có thể tăng nhưng cũng có thể giảm trong lần chạy sau. Khi dự báo mức độ quan trọng của các biến độc lập, các mô hình có thể cho ra những xu hướng khác nhau trong kết quả tính. Những hạn chế này có thể được khắc phục bằng phương pháp thử dần, mỗi mô hình cần chạy nhiều lần để tìm ra kết quả tốt nhất. Trong số sáu mô hình ML, SVR có sự kiên định cao nhất, kết quả dự báo của nó không đổi sau các lần chạy. Để làm rõ hơn về vấn đề này cần tiếp tục nghiên cứu sâu hơn.



Hình 4: Kết quả kiểm định mô hình và công thức

5. KẾT LUẬN

Bài báo này trình bày một phương pháp nghiên cứu mới để tính toán chiều dài nước nhảy tương đối trong kênh chữ nhật đáy bằng. Trong nghiên cứu này, Định lý π-Buckingham đã được sử dụng để tìm các tham số không thứ nguyên làm đầu vào và đầu ra của mô hình toán. Không chỉ số Froude trước nước nhảy, độ nhám và chiều rộng kênh cũng như độ nhớt của chất lỏng đã được xem xét khi tính chiều dài nước nhảy. Nghiên cứu này đã thiết lập sáu mô hình ML để đánh giá tầm quan trọng của các biến đầu vào, sau đó sử dụng các mô hình này để dự báo chiều dài nước nhảy tương đối. Ngoài ra, các ưu điểm và hạn chế của mô hình ML cũng được chỉ ra trong bài báo này.

Kết quả kiểm định mô hình cho thấy rằng, mô hình ET có độ chính xác cao nhất. Sau đó là Ada, RF, GB, DT, SVR theo thứ tự độ chính xác giảm dần. Tất cả các mô hình ML đều cho kết quả tính toán chính xác hơn các công thức kinh nghiệm. Vì vậy, mô hình ET có thể thay thế các công thức kinh nghiệm trong dự báo chiều dài nước nhảy.

TÀI LIỆU THAM KHẢO

[1] Abbaspour, A., Farsadizadeh, D., & Ghorbani, M. A. (2013). Estimation of hydraulic jump on corrugated bed using artificial neural networks and genetic programming. *Water Science and Engineering*, *6*(2), 189–198. https://doi.org/10.3882/j.issn.1674-2370.2013.02.007

[2] Baharvand, S., Jozaghi, A., Fatahi-Alkouhi, R., Karimzadeh, S., Nasiri, R., & Lashkar-Ara, B. (2021). Comparative Study on the Machine Learning and Regression-Based Approaches

to Predict the Hydraulic Jump Sequent Depth Ratio. *Iranian Journal of Science and Technology - Transactions of Civil Engineering*, *45*(4), 2719–2732. https://doi.org/10.1007/s40996-020-00526-2

[3] Brakeni, A., P, G., M, C., V, M., & S, V. (2021). STUDY OF A STILLING BASIN WITH A SWIRLING FLOW. *Larhyss Journal*, *46*, 115–130.

[4] Brunton, S. L., Noack, B. R., & Koumoutsakos, P. (2020). Machine Learning for Fluid Mechanics. *Annual Review of Fluid Mechanics*, *52*(1), 477–508. https://doi.org/10.1146/annurev-fluid-010719-060214

[5] Hager, W. H. (1992). *Classical Hydraulic Jump BT - Energy Dissipators and Hydraulic Jump* (W. H. Hager (ed.); pp. 5–40). Springer Netherlands. https://doi.org/10.1007/978-94-015-8048-9_2

[6] Hager, W. H., & Bremen, R. (1989). Sequent depths: Le ressaut hydraulique classique: étude des hauteurs conjuguées. *Journal of Hydraulic Research*, *27*(5), 565–585. https://doi.org/10.1080/00221688909499111

[7] Ho, H. V., Nguyen, D. H., Le, X. H., & Lee, G. (2022). Multi-step-ahead water level forecasting for operating sluice gates in Hai Duong, Vietnam. *Environmental Monitoring and Assessment*, *194*(6), 1–27. https://doi.org/10.1007/s10661-022-10115-7

[8] Houichi, L., Dechemi, N., Heddam, S., & Achour, B. (2013). An evaluation of ANN methods for estimating the lengths of hydraulic jumps in U-shaped channel. *Journal of Hydroinformatics*, *15*(1), 147–154. https://doi.org/10.2166/hydro.2012.138

[9] Kenda, K., Peternelj, J., Mellios, N., Kofinas, D., Čerin, M., & Rožanec, J. (2020). Usage of statistical modeling techniques in surface and groundwater level prediction. *Journal of Water Supply: Research and Technology - AQUA*, *69*(3), 248–265. https://doi.org/10.2166/aqua.2020.143

[10] Khosravinia, P., Sanikhani, H., & Abdi, C. (2018). Predicting Hydraulic Jump Length on Rough Beds Using Data-Driven Models. *Journal of Rehabilitation in Civil Engineering*, *6*(2), 139–153. https://doi.org/10.22075/JRCE.2017.11047.1180

[11] Kisi, O., Khosravinia, P., Nikpour, M. R., & Sanikhani, H. (2019). Hydrodynamics of river-channel confluence: toward modeling separation zone using GEP, MARS, M5 Tree and DENFIS techniques. *Stochastic Environmental Research and Risk Assessment*, *33*(4–6), 1089–1107. https://doi.org/10.1007/s00477-019-01684-0

[12] Mammadov, A. (2017). Hydraulic jump on smooth and uneven bottom. *International Journal of Advanced Engineering Research and Science*, *4*(11).

[13] Naseri, M., & Othman, F. (2012). Determination of the length of hydraulic jumps using artificial neural networks. *Advances in Engineering Software*, *48*(1), 27–31. https://doi.org/10.1016/j.advengsoft.2012.01.003

[14] Peterka, A. J. (1984). Hydraulic Design of Stilling Basins and Energy Dissipators. *Monograph E, Editor. A Water Resources Technical Publication, USBR*, *25*, 240. https://ntrl.ntis.gov/NTRL/dashboard/searchResults/titleDetail/PB95139457.xhtml

[15] Rezaee, A., Bozorg-Haddad, O., & Chu, X. (2023). Comparison of data-driven methods in the prediction of hydro-socioeconomic parameters. *Aqua Water Infrastructure, Ecosystems and Society*, *72*(4), 438–455. https://doi.org/10.2166/aqua.2023.161

[16] Silvester R. (1964). Hydraulic jump in all shapes of horizontal channels. *ASCE, Journal of the Hydraulics Division*, *90*(HY1), 23–55. https://doi.org/10.1061/JYCEAJ.0000977

[17] Truong, V. H., Ly, Q. V., Le, V. C., Vu, T. B., Le, T. T. T., Tran, T. T., & Goethals, P.

(2021). Machine learning-based method for forecasting water levels in irrigation and drainage systems. *Environmental Technology and Innovation*, *23*, 101762. https://doi.org/10.1016/j.eti.2021.101762.