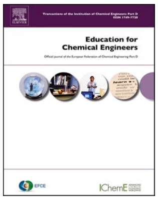
Education for Chemical Engineers 46 (2024) 10–21

Contents lists available at ScienceDirect

Education for Chemical Engineers

journal homepage: www.elsevier.com/locate/ece



Machine learning in chemical engineering: Hands-on activities

Vitor Lavor [a,b], Fernando de Come [a], Moisés Teles dos Santos [a], Ardson S. Vianna Jr. [a,*]

[a] Department of Chemical Engineering, University of São Paulo, Av. Prof. Luciano Gualberto, 380, 05508-010, São Paulo, Brazil
[b] School of the Built Environment, University of Reading, Reading, UK

ARTICLE INFO

Keywords:
Machine Learning
Hands-on
Clustering
Classification
Regression
ANN

ABSTRACT

A set of hands-on activities, that were proposed in an introduction course to machine learning in a Chemical Engineering undergraduate course, are presented. The activities aimed to introduce basic concepts of unsupervised learning (e.g., clustering) and supervised learning (e.g., classification and regression). Google Colaboratory, a cloud service provided by Google for free to promote research in Artificial Intelligence and Machine Learning, was used to develop these activities, but the proposed activities can be run similarly in a local Python environment. The datasets used in the activities are publicly available on websites such as Kaggle and University of California (UCI), and a specific example in chemical engineering for the ore grinding process was also used. The student's response to the ML topic within the course was very positive.

1. Introduction

1.1. *Artificial intelligence and society*

Artificial Intelligence (AI) and Machine Learning (ML) have been impacting society (Bryson, 2019). According to Amy Stapleton: “We are entering a new world. The technologies of machine learning, speech recognition, and natural language understanding are reaching a nexus of capability. The end result is that we'll soon have artificially intelligent assistants to help us in every aspect of our lives.” Mark Cuban also affirmed: “Artificial Intelligence, deep learning, machine learning — -whatever you're doing if you don't understand it — learn it. Because otherwise, you're going to be a dinosaur within 3 years.” Besides, Professor Aleksander Madry, director of the MIT Center for Deployable Machine Learning said: “Machine learning is changing, or will change, every industry, and leaders need to understand the basic principles, the potential, and the limitations.”

But what is ML? ML is a subset, one of the most significant, of AI. That's when one mimics human learning. The starting point is a known, and meaningful, data set so that you can learn from it. From there you can sort, group, and estimate information from the data. Mitchel (2017) put the idea in the form of a question: “How can one construct computer systems that automatically improve through experience?”.

In fact, an inversion of the direction of programming occurs (Girmsom (2020)). The conventional modeling and simulation process traditionally used, also in chemical engineering, is composed of the following steps: 1- Observe the physical phenomenon; 2- Assume hypotheses; 3- Generate a mathematical model from the fundamental equations; 4-Generate data from the model; 5- Compare the model with experimental data.

The construction of a model by machine learning reverses this order. The model is developed from the experimental data. An adapted figure from Professor Grimson's presentation is shown in Fig. 1:

One can argue: is it a good approach? Big companies have been using Machine learning. For example, Watson from IBM recommends treatment for different types of cancer. Malluba from Microsoft develops deep learning for Natural Language Processing (NLP). Google uses networks to create relationships between datasets.

Engineering has also become another major field where AI has been making a big impact. Industrial processes usually generate a large volume of data, and often it's needed to deal with very complex processes where usual mathematical modelling based on phenomenological models and physical principles becomes nearly impossible to perform. That's where de Data-Driven Modeling comes in handy, requiring just a lot of data to accurately describe very complex systems. It's also possible, though not required, to include phenomenological knowledge about the system into the IA model, achieving even higher accuracies.

1.2. *Machine learning in chemical engineering*

The work of Professor Venkatasubramanian serves as a valuable starting point (Venkatasubramanian, 2019). He highlights several

[*] Corresponding author.
E-mail address: ardson@usp.br (A.S. Vianna).

https://doi.org/10.1016/j.ece.2023.09.005
Received 30 December 2022; Received in revised form 31 July 2023; Accepted 28 September 2023
Available online 4 October 2023

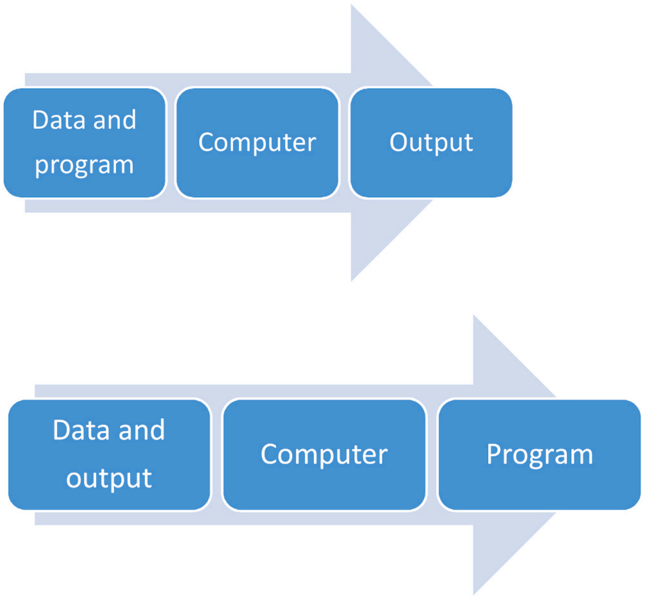


Fig. 1. Inversion towards programming.

challenging problems, including the development of conceptual frameworks such as hybrid models, mechanism-based causal explanations, domain-specific knowledge, discovery engines, and analytical theories of emergence. These advanced issues are exemplified by past examples like process synthesis and design, modeling environments, molecular structure search engines, automatic reaction network generators, and chemical entity extraction systems. Furthermore, Professor Venkatasubramanian suggests promising innovations for the future, namely: 1) recurrent neural networks, where memory is incorporated through long short-term memory (LSTM) units; 2) reinforcement learning; and 3) statistical machine learning.

Some works are summarized here in order to show the extent of the applications in Chemical Engineering, without trying to exhaust the subject. Quaglio et al. (2020) developed an artificial neural network with the goal of identifying kinetic models based on experimental data. To evaluate the network, the authors conducted a case study involving a complex reaction consisting of three compounds and three reactions. Alhajeri et al. (2021) studied the dynamic behaviour of nonlinear chemical processes. They estimated states using machine learning, to apply predictive control. Mowbray et al. (2021) feature a review of applications of ML in bioengineering. Some algorithms era discussed as Multivariate statistical analysis, Principal component analysis, Support vector machines, RNN. Applications can be on bioreactor systems, metabolic engineering and biomaterials engineering, or biosensors and biodevices. Final models can be used as digital twins, even in high dimensional, nonlinear, and stochastic domains.

ML has been used widely to represent complex models, in order to reduce computational time and make the dynamic model treatable. Alves et al. (2022) developed a machine learning-based framework for process operability using Gaussian Process Regression.

Some items may be considered by chemical engineering, or, because of their importance, should be considered. Campos et al. (2019) discussed the importance of synthetic chemistry in the pharmaceutical industry. Some computational tools were evaluated to direct drug synthesis. For example, ML makes it possible to find new chemical routes not yet found and, with this, to synthesize new drugs. Specifically, it is cited as “the use of algorithms (ML) for synthetic route planning to a target molecule”. Therefore, ML stands out as a tool for the synthetic optimization of targets much higher.

1.3. Hands-on experiences

Hands-on activities have always accompanied undergraduate engineering courses since it is necessary to talk about theoretical knowledge and hands-on capability to solve practical problems. One classic example is the use of pilot plants being operated directly by students, such as the study of membrane technology by Souto-Melgar et al. (2022).

Another example is to use of computational activities, such as the use of CFD, which can be incorporated into the engineering curriculum (Adair et al., 2014). Thus, students may better understand the basic concepts.

Broader use of hands-on activities is defining the market and product design needs. This was suggested by Galán et al. (2018) in the proposal for market needs a strategy in the product project based on the fundamentals of transport phenomena.

1.4. Data science in undergraduate chemical engineering course

Professor Venkatasubramanian is a forerunner in artificial intelligence for the undergraduate degree in chemical engineering (Venkatasubramanian, 1986). For this, he used the concept of knowledge-based expert systems (KBES). An introductory-level course in Computational Intelligence was proposed by Venayagamoorthy (2005). Despite the growth in the number of publications in ML, Dobbelaere et al. (2021) claim that chemical engineers still have limited training in artificial intelligence, computer science, and data analysis.

In the present work, we aim to discuss the design and implementation of a hands-on learning framework developed at the Department of Chemical Engineering, for an undergraduate course in Chemical Engineering. The hands-on activities covered basic concepts of unsupervised learning – clustering – and several techniques in supervised learning – classification and regression. The activities also included the study of Artificial Neural Networks (ANN), a powerful tool for modelling complex relationships in data. The Colab tool was used for the development of hands-on activities and validated its effectiveness in training students to be confident and self-directed learners.

2. Methodology

The methodology used in the case studies can be seen as the basic paradigm of ML (Guttag, 2016):

Set the problem:

This initial step in the engineering model-building process, which may not necessarily involve machine learning, focuses on observing the phenomenon, identifying relevant variables, and understanding the responses. The goal is to detect any particularities or singularities associated with the phenomenon.

Upload the dataset:

ML activities rely heavily on substantial datasets, demanding a good quality set of information for meaningful outcomes. In real-world applications, feature engineering becomes essential to manipulate the data effectively. However, for academic courses, utilizing pre-existing datasets can be advantageous, as it allows students to prioritize the learning of ML algorithms such as clustering, classification, and regression techniques.

Understand its structure:

When the data comes from public datasets (e.g., Kaggle), the entire dataset can be visualized, providing valuable insights into the total number of columns (i.e., number of features) and rows (i.e., number of observations). Additionally, data visualization methods allow a deeper understanding of the inputs’ interrelationships using techniques such as heat maps.

Evaluate the algorithms:

This activity involves making informed decisions about the algorithm selection, implementation, and pre-assessment of results. For instance, when performing a classification task on a dataset, multiple algorithms, such as decision trees and Support Vector Machines (SVM), can be considered. To evaluate the quality of the algorithm, the dataset needs to be partitioned into training and testing sets, providing the parameters necessary for assessment.

Evaluate the results:

Evaluating the results is a crucial activity in all ML applications, with each algorithm having its distinct evaluation criteria. For instance, in classification tasks, the results can be evaluated using metrics like accuracy, precision and the confusion matrix (showing the counts of true positives, false positives, true negatives, and false negatives), while regression evaluations rely on metrics like mean absolute error (MAE) and R2 score. Comparison between different methods is a good practice to ensure that the results are consistent and aligned with the desired outcomes. However, it is important to remember that the modelling and simulation process can always have inherent errors.

Show the results:

There are several effective ways to present ML results, depending on the modelling approach and the characteristics of the data. Heatmaps are valuable visualization tools that can reveal relationships and patterns between variables. For regression tasks, a 2D plot and graphs (e.g., line plots, scatterplots) can offer highly informative representations, especially when focusing on one attribute at a time. In the course, we place significant emphasis on the importance of result discussion. Students have the opportunity to create various plots and are encouraged to provide meaningful insights derived from them. As part of the weekly activities, the students are asked to reflect on the results they obtained and participate in discussions related to their findings. This practice fosters a deeper understanding of the ML process and its implications.

The datasets used in this study are readily available on the Internet, from sources such as Kaggle and the University of California repository (UCI). These easily accessible datasets have captured students' attention, leading them to focus on problems with less complexity, rather than utilizing process variables, which demand a deeper understanding of the underlying processes. Nevertheless, the study also involved the examination of a typical grinding process to develop an Artificial Neural Network (ANN).

3. Contextualization

The activities presented here are part of the course PQI 3403 Analysis of Chemical Process Industry, which is a topic of the syllabus of the four-month course of Chemical Engineering at the University of Sao Paulo, Brazil. It has a workload of 60 h and is taught in the first quarter of the 4th year.

The Machine Learning topic covered various concepts and algorithms, including clustering, classification and regression. Additionally, specific algorithms such as ANN, logistic regression, support vector machine (SVM), and model metrics were discussed within this context. The practical activities were carried out using Google Colab, which is a cloud-based environment for Jupyter Notebooks, within the Google ecosystem and it is implemented in Python. The course opted for Python programming within Google Colab notebooks, not only because of students' familiarity with it, developed since the first year of their undergraduate degree but also due to its open-source nature.

The syllabus of the course PQI 3403 Analysis of Chemical Process Industry is:

1. Presentation of the course; Introduction to Scilab; Tutorial of Scilab.
2. Linear Systems of Algebraic Equations. Sparse matrix.
3. Qualitative method for solving ODE. Critical points: nodes, saddle, center, spiral points. Geometric analysis of linear systems.
4. Almost linear systems. Phase plane. Phase portrait. Classic dynamic systems: chemical reacting, pendulum, and population balances.
5. Numerical Methods for Non-linear ODEs with Initial Conditions. Euler, Runge-kutta, multistep, and BDF methods.
6. Numerical Methods for PDE. Finite differences. Fictitious domain method. 2D heat diffusion. MOL.
7. Introduction to Artificial Intelligence: search, learning, reasoning.
8. Machine Learning: clustering, classification, and regression. Datasets available on the internet. Ranking metrics. Logistic function.
9. Deep Learning: Artificial Neuronal Networks (ANN).
10. Stochastic processes: Brownian movement, Wiener process, stochastic differential equations. Applications using Python library.

The details of item 10, Stochastic Processes, can be seen elsewhere (Oliveira et al., 2022; Nakama et al., 2017).

The learning objectives for the hands-on activities are specifically:

1. Recognize ML basic concepts;
2. Use the Colab tool to develop the ML basic concepts;
3. Analyze the datasets;
4. Carry out clustering by implementing different techniques;
5. Executing classification using decision tree and SVM;
6. Implement linear and multilinear regressions;
7. Generate ANN using the Keras.

4. Case studies

Four main topics within machine learning were covered in the classroom and transformed into tasks to be developed by students in pairs using the Google Colaboratory environment (Bisong, 2019), which was introduced in an introductory class. Each task aimed to present the students with some of the basic concepts of unsupervised learning – clustering – and supervised learning – classification and regression. Within supervised learning, particular emphasis was placed on ANN.

Google Colaboratory, popularly called Colab, is a cloud service offered by Google for free to encourage Artificial Intelligence and Machine Learning research.

Some of the main features of Colab are:

- It is already configured, and it is not necessary to have a powerful personal computer.
- It's simple to share, like any file stored in Drive. To understand a little better about the environment, one should access the following link:https://colab.research.google.com/notebooks/intro.ipynb

4.1. Clustering

The main task involved in clustering analysis is to divide the population into specific groups, so that those belonging to the same group have similar characteristics. Several algorithms perform the task based on different techniques, such as connectivity-based models (e.g., hierarchical clustering), centroid-based models (e.g., K-Means), distribution-based models (e.g., Gaussian), and density-based models (e.g., DBSCAM).

All models are especially useful for certain types of datasets and variables, however, the most common ones, and the ones have chosen to be presented in the classroom, are the hierarchical algorithms and K-Means.

4.1.1. Dataset

Two datasets were used to introduce the clustering algorithms to the students. The first was the very common Iris flower and consisted of 150 samples of 3 different species, and the second dataset is a modified dataset, consisting of data from NBA 2020 season players. However, only the iris dataset analysis will be informed. The details of each dataset are presented in Tables 1 and 2. General dataset information, as well as the notebook with the expected analysis, can be found at: https://github.com/vitorlavor/education/tree/main/clustering.

4.1.2. K-means

The K-Means algorithm performance is based on the inertia criterion (or within-cluster sum-of-squares – wcss), Eq. 1. Unlike other algorithms, K-Means requires the number of clusters as input to the model.

Basically, the algorithm divides the dataset of N samples X into user-specified K clusters, which is described by μ_j - the average point of the dataset within the cluster. This average point is called centroid. The K-means algorithm aims to determine centroids that minimize the inertia, or within-cluster sum-of-squares criterion:

$$wcss = \sum_{i=0}^{n} \min(\|x_i - \mu_j\|^2) \tag{1}$$

The tasks performed by the students were:

1. Determine the ideal number of clusters using the Knee/Elbow method;
2. Build a K-Means model using the ideal cluster number determined in Task 1;
3. Compare the labels provided by the model with the real labels.

The expected analysis is demonstrated in Figs. 2 and 3. The Knee plot in Fig. 2 displays the behavior of the wcss metric for different numbers of clusters. The optimal number of 3 clusters was determined using a knee point detection algorithm Satopaa et al. (2011).

Fig. 3 shows the distinction between species/clusters based on petal length and sepal width characteristics. It is possible to observe that the K-Means model - Fig. 3b - presented a consistent grouping compared to the real labels - Fig. 3a.

4.1.3. Hierarchical clustering

The hierarchical algorithm, as the name already implies, seeks to build clusters based on hierarchical classes of similarities between samples. There are two main methods within the algorithm family:

- Agglomerative: a bottom-up approach, where each sample starts in its isolated cluster and merges with other clusters as they move up the hierarchy.
- Divisive: a top-down approach, where all samples start in a single big cluster and split as they go down the hierarchy.

The results of the hierarchical clustering are usually presented in a dendrogram, which is a tree diagram.

The tasks performed by the students were:

1. Build a hierarchical clustering model, using the clustering method known as single linkage;
2. Build a dendrogram resulting from the model developed in Task 1;
3. Analyze the construction of the dendrogram.

Table 1
Iris dataset details.

Variable	Sample (N)
Sepal length [cm]	150
Sepal width [cm]	150
Petal length [cm]	150
Petal width [cm]	150
Specie [class]	150

Table 2
NBA 2020 players dataset details.

Variable	Sample (N)
Name	426
Height [m]	426
Weight [kg]	426
Position [class]	426

The expected analysis is shown in Fig. 4. It is possible to understand the similarities between the samples, as well as to determine the number of clusters depending on the desired hierarchical similarity level.

4.2. Classification

Classification is supervised learning, where a set of features x_i is related to labels y_j. they form a set of feature/label pairs. From this, it is possible to find a rule that allows one to associate a label with a feature not yet known. Thus, it is a discrete data set, such as sunny, cloudy, or rainy, it's not a continuous function.

The classification may or may not be successful. False positives or false negatives may occur. Hence several parameters arise that help in the evaluation of the efficiency of the classification such as the confusion matrix and the receiver operating characteristic curve.

4.2.1. Dataset

In this activity, the dataset *red_wine_quality.csv* (Dua and Graff (2017)) was used. The link was given for students to access the data set. It is present in Kaggle and UCI repositories.

The dataset is related to red variants of the Portuguese "Vinho Verde" wine. The input variables are 1 - fixed acidity, 2 - volatile acidity, 3 - citric acid, 4 - residual sugar, 5 – chlorides, 6 - free sulphur dioxide, 7 - total sulphur dioxide, 8 – density, 9 – pH, 10 – sulphates, 11 – alcohol; and the output variable (based on sensory data), 12- quality (score between 0 and 10).

It is not necessary to be a wine connoisseur to understand the importance of each of these parameters. An exploratory data analysis was done using the heatmap (Fig. 5), which is also an effective visualization tool. From the heatmap, it was possible to identify that the alcoholic content is the most important (i.e., more correlated) feature to predict the wine quality of the wine.

4.2.2. Decision tree and SVM modeling

The classification was developed by two approaches: 1- Decision Tree and 2-Support Vector Machine (SVM). For data partition, the percentage of 80–20% was chosen. Activities have been requested:

a. Evaluate the result of confusion matrices;
b. Calculate Accuracy, Precision, and Recall metrics for each model;
c. Analysis of the results. What was the best model?

a. Evaluate the result of confusion matrices;

The Seaborn library was used to build the heatmap tool from the build confusion matrices, as can be seen in Fig. 5. The two heatmaps obtained are relatively similar, presenting the same pattern but with slightly different results. Moreover, both models demonstrated superior true positive (TP) and true negative (TN) rates, while the false negative (FN) and false positive (FP) were comparatively lower, indicating a favourable performance. (Fig. 6).

b. Calculate Accuracy, Precision, and Recall metrics for each model using the test dataset;

Table 3 presents the accuracy, precision, and recall for the two models, the decision tree and SVM. The decision tree model showed

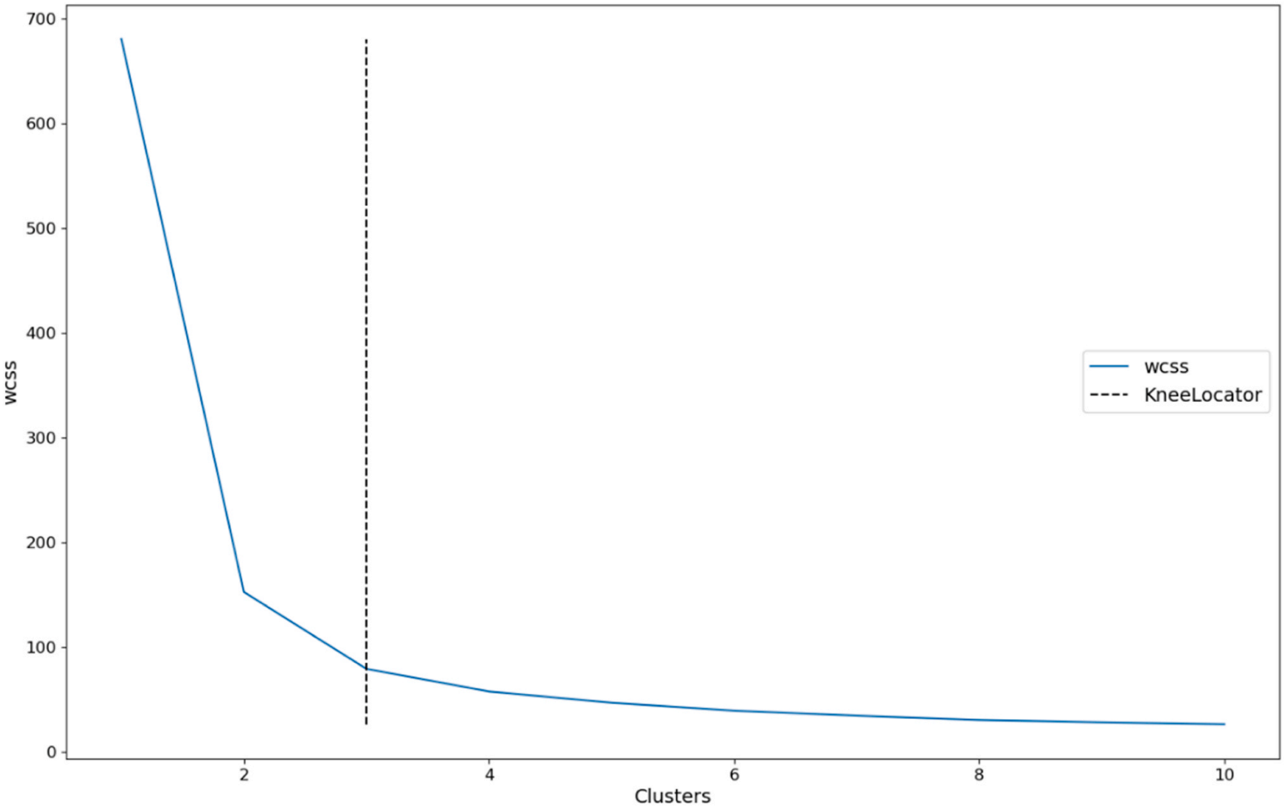


Fig. 2. Knee plot for the Iris dataset.

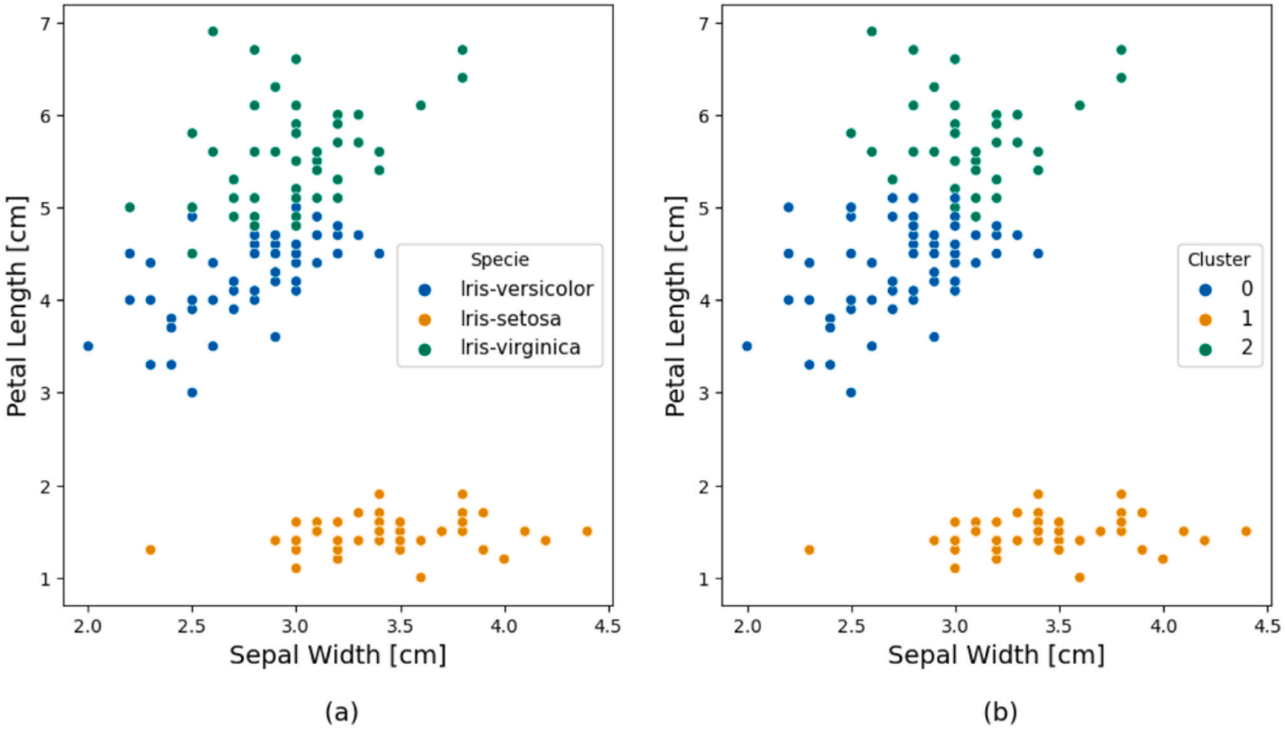


Fig. 3. Comparison of petal width against petal length for the Iris dataset, using (a) actual labels and (b) predicted labels from a KMeans clustering model. The scatter plots show the distribution of data points for each class.

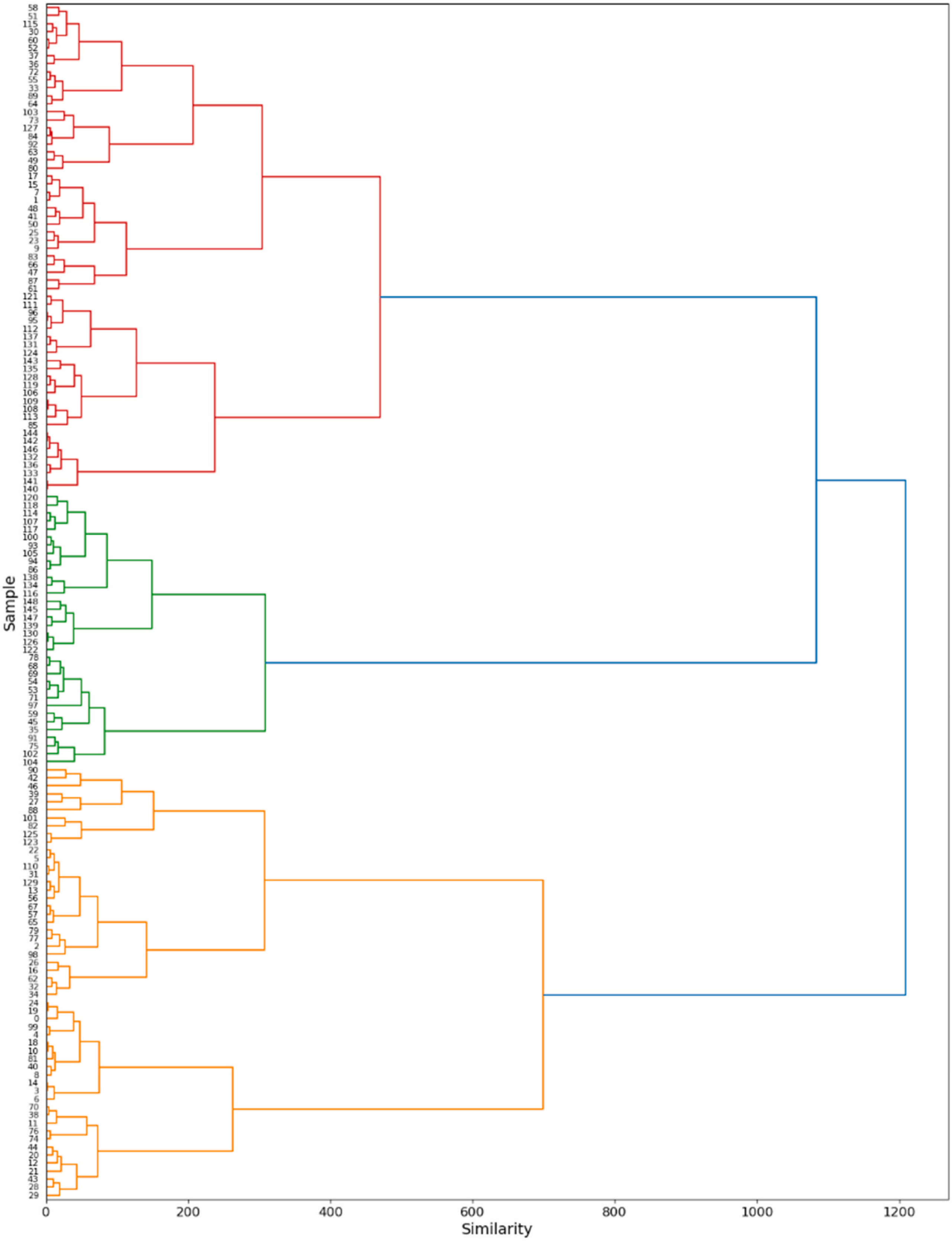


Fig. 4. Dendogram diagram for the Iris dataset.

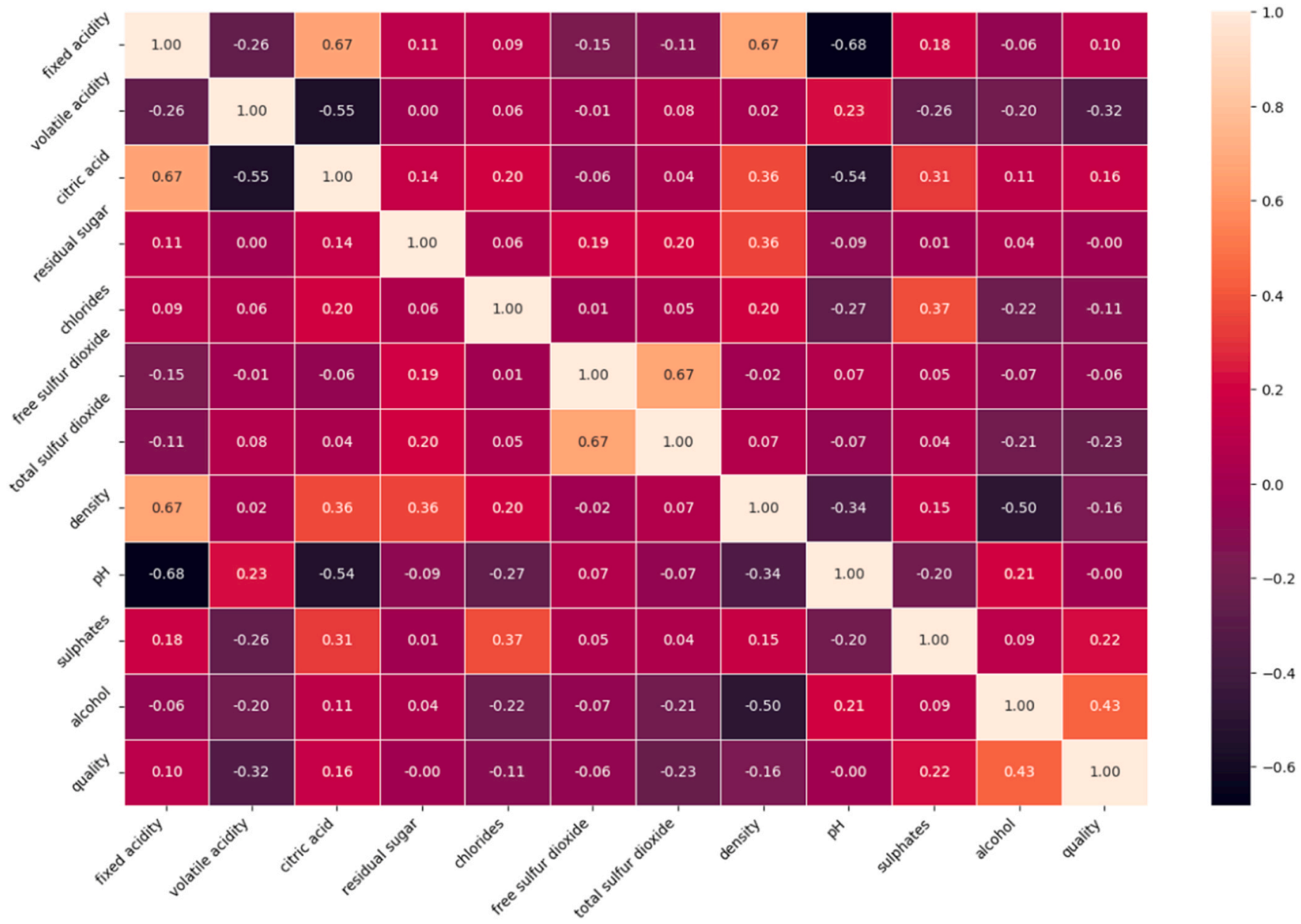


Fig. 5. Heat map of red variants of the Portuguese "Vinho Verde" wine.

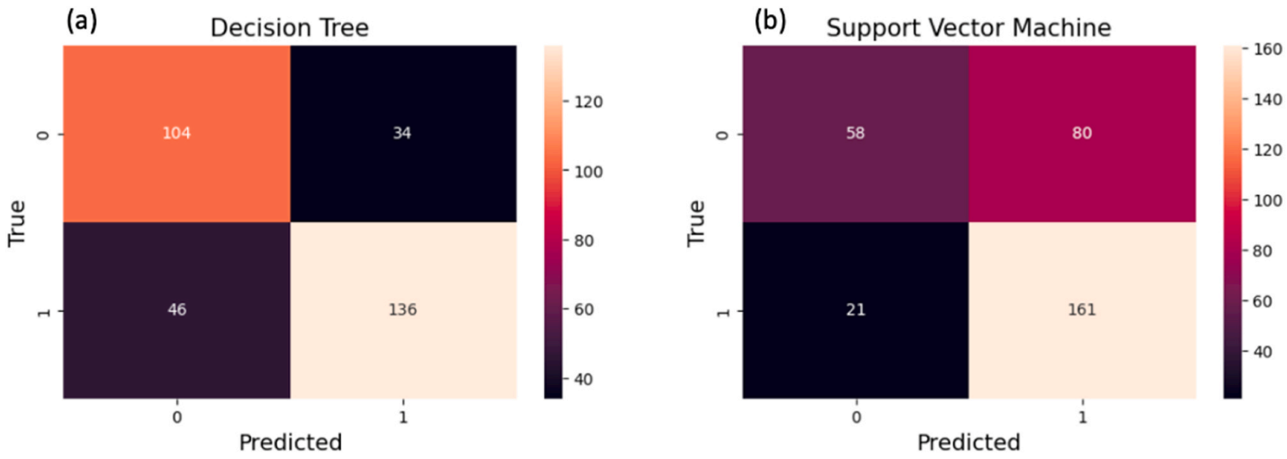


Fig. 6. Confusion matrix for (a) decision tree model and (b) SVM model.

Table 3
Model metrics calculated on the test dataset.

Metric	Decision tree	SVM
Accuracy	0.77	0.65
Precision	0.77	0.64
Recall	0.80	0.87

better accuracy and precision than SVM. However, the SVM model presented a better recall metric.

c. Analysis of the results. What was the best model?

Accuracy indicates how much of examples were correctly classified, both positive and negative. From this perspective, the decision tree represented better the data set.

The precision indicates how many positives were correctly classified within the universe of positives. In this context, the decision tree also presented better performance.

The recall answers the question of how many positives were answered correctly as positive. This is an important question when the result is a medical diagnosis. In this area, the SVM model was better. As the quality of wine does not involve lives, we can indicate that the decision tree model was more efficient.

4.3. Regression

Regression is supervised learning as well, but unlike Classification, the output takes the form of a continuum function. Linear regression is a basic mathematical concept, widely used in various areas of teaching. Russell and Norvig (2002) present an equation (Eq. 2) that already considers the learning process:

$$h_w = w_0 + w_1 x \tag{2}$$

Where h_w is the label associated with the feature x and weights w_0 and w_1, which will be changed by the learning process.

But the label (h_{sw}) can be a function of more than one feature (x_i), as is the case of the quality of the wine in the previous item. It now has a multivariate linear regression, Eq.3 (Russell and Norvig (2002)):

$$h_{sw} = w_0 + \sum_i w_i x_i \tag{3}$$

The idea of regression can be used for other functions, that is, what should be done is a process of minimizing error when comparing the chosen function with the actual data set.

There is no limit to creativity, especially to solve complex problems. Russell and Norvig (2002) discuss regression trees, which are associations between classification by decision tree and regression. Here a regression is made for each leaf in the tree. The regression tree was suggested in the activity, but students could not alone develop this more advanced approach.

4.3.1. Regressions with public dataset

Here, the dataset *wholesale_customers_data.csv* (Kaggle (2021)) was used.

The list of questions was:

a. Create a heat map and identify the variable with the highest correlation with the consumption variable;
b. Build a simple linear regression model using the identified variable.

The heatmap points out that the annual spent on cleaning products (input Detergent_Paper) is more strongly correlated with the total spent on groceries (Fig. 7) since it has the largest correlation module. The input Detergent_Paper was then chosen to develop a simple linear regression model the predict the total groceries spent with a coefficient of determination R2 = 0.8475. This low coefficient of correlation is associated with the scattering of the data set, as can be seen in Fig. 8.

4.3.2. Regressions with generated dataset

A dataset was generated based on a trend that follows a sine curve, with random errors added to the dataset. The dataset points and shape can be observed in Fig. 9.

For the generated dataset, four regression models were developed using the *Scikit-learn* library. The prediction generated by the models using a test dataset that consisted of a sequence of numbers starting with 0 up to 5 with a step of 0.01. The models are:

a. Linear regression.
b. Decision tree (max_depth parameter = 2).
c. Decision tree (max_depth parameter= 5).
d. KNN regressor (K-nearest neighbors).

The linear regression did not effectively capture the characteristics of the dataset, as expected. Consequently, other approaches were implemented, yielding better predictions. The decision trees proved effective

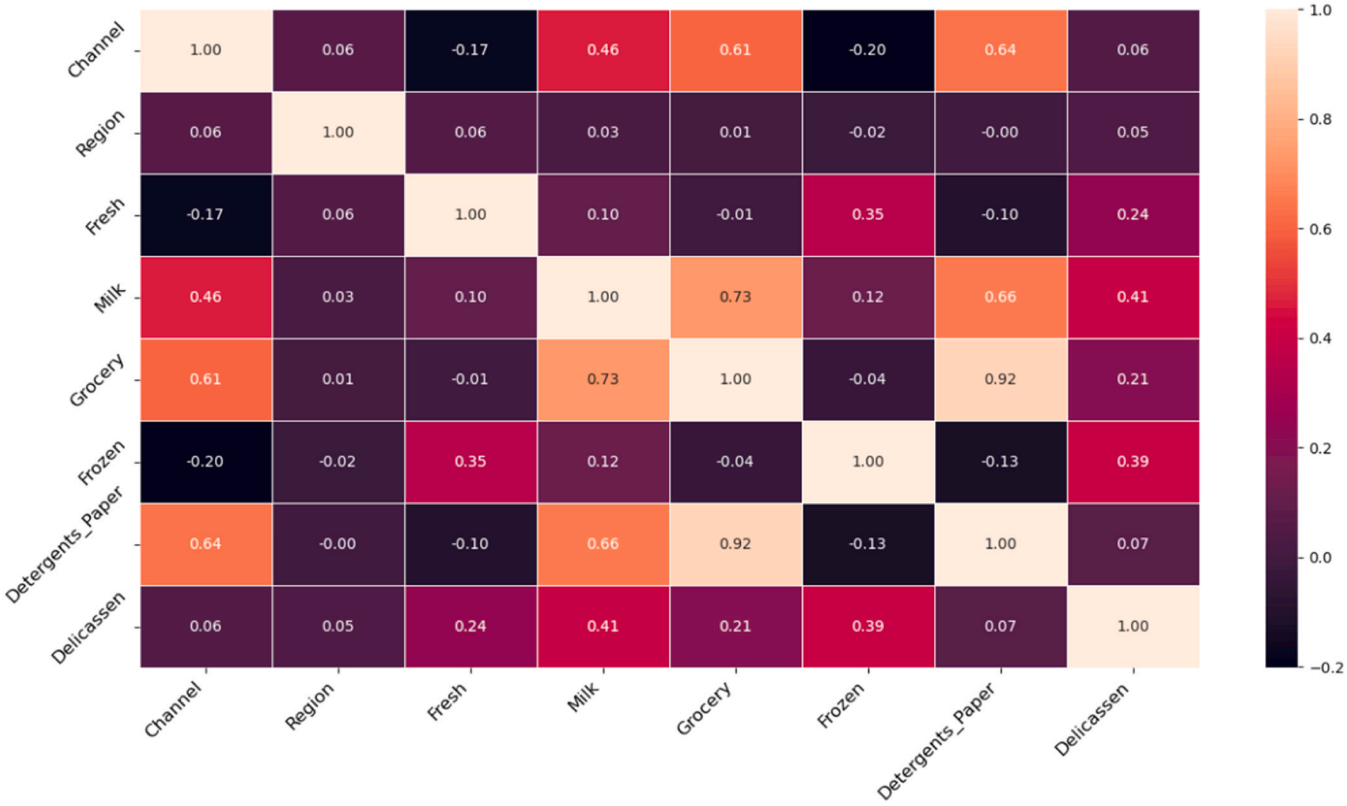


Fig. 7. Heatmap for the wholesale customers dataset.

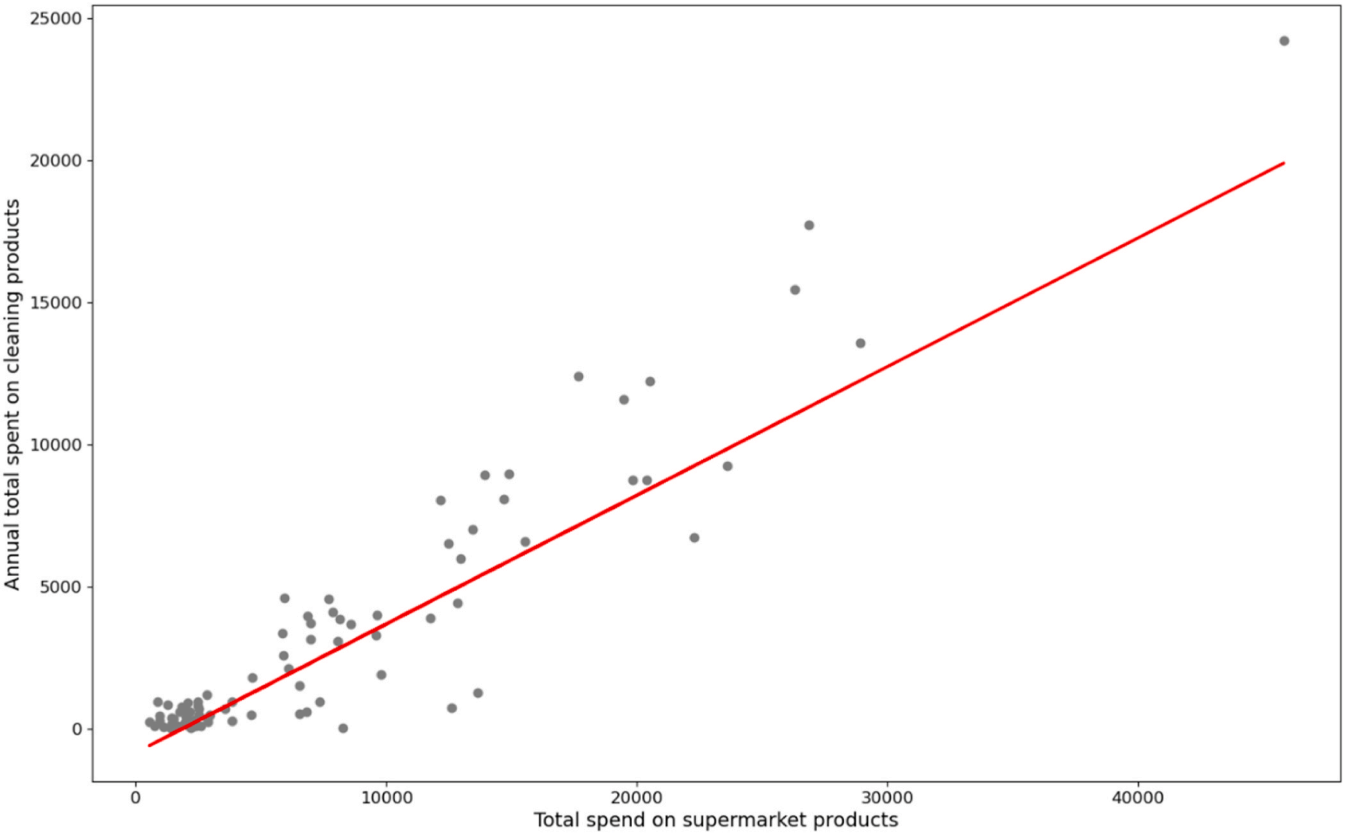


Fig. 8. Simple linear regression to predict the total spent on groceries based on the consumption of cleaning products.

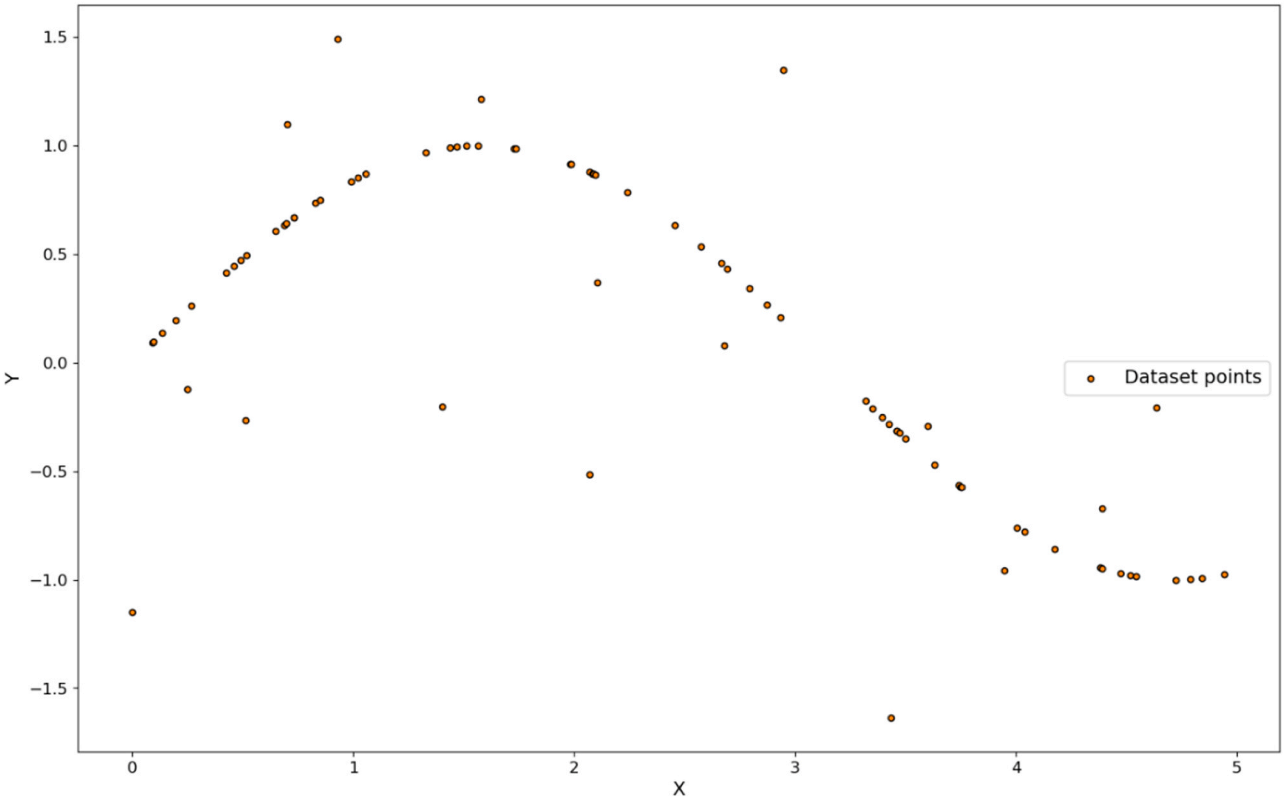


Fig. 9. Dataset generated based on a sine curve with random errors.

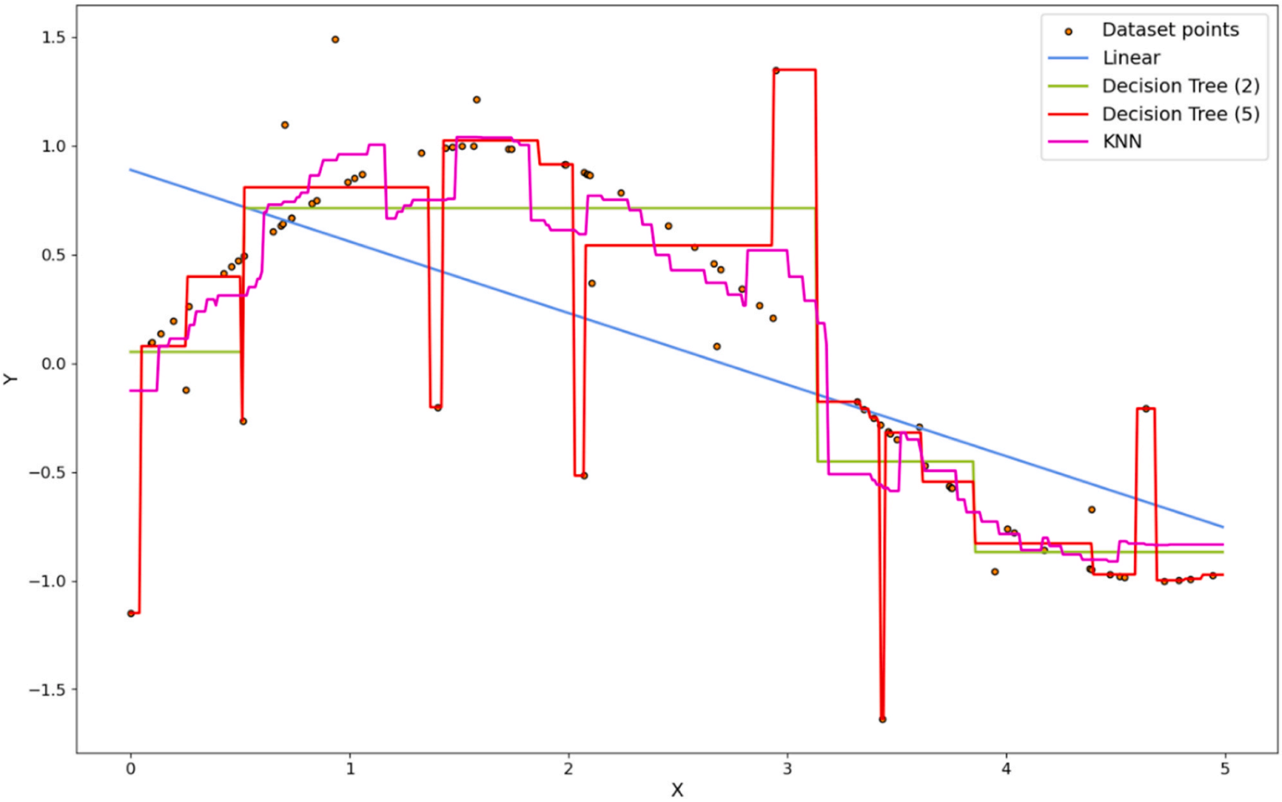


Fig. 10. Comparison between linear regression, decision tree and KNN regression models.

in identifying the errors, and the deeper decision tree (max_depth = 5), as depicted by the red line in Fig. 10, demonstrated a better ability in capturing the error patterns in the dataset. Similarly, the KNN algorithm demonstrated a similar capacity, as illustrated by the magenta line in Fig. 10.

4.3.3. ANN

A database related to a real mining processing industry, specifically a milling process, was utilized. In the context of mining, milling refers to the process of grinding and crushing ore into smaller particles for further processing. The grinding process is usually composed of two operations: the mill itself and a classification operation, which is usually represented by a cyclone. The new ore is fed into the mill together with the cyclone underflow (also known as circulating load) and dissolution water. This underflow represents the fraction of the ground ore that has not been fully ground to the correct particle size and is returned to the mill for a new milling operation. The ore that has been correctly ground leaves the cyclone as the overflow and proceeds to the next operation. Fig. 11 shows the flowchart of the grinding process used in the activities.

Dealing with solids is always a difficult task because the laws

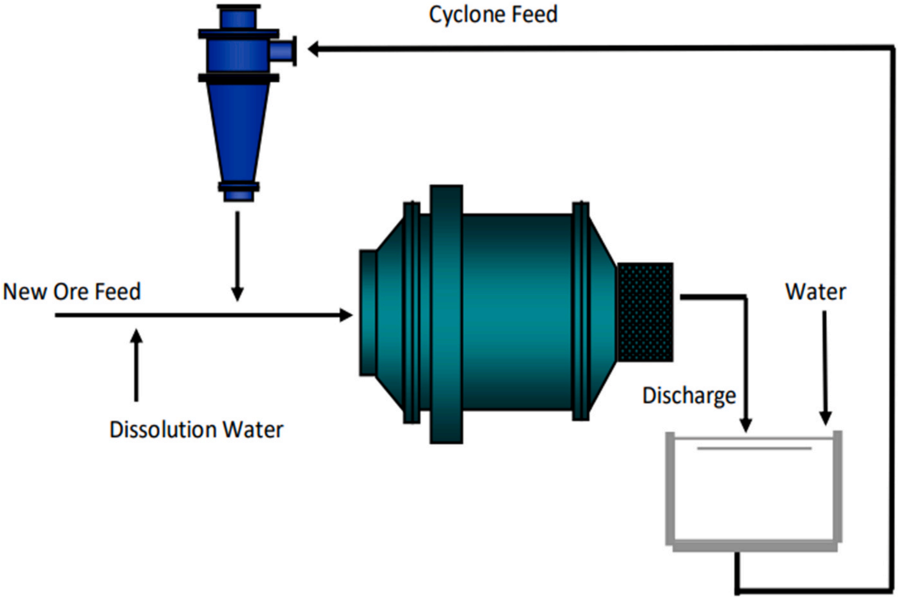


Fig. 11. Flowchart of the grinding process.

governing the mechanics of solids are complex. For instance, one cannot accurately estimate the amount of energy required to carry out the size reduction. Therefore, a good choice is to use the history of the process data, taking advantage of the Data-Driven modeling approach.

The idea behind the use was to optimize the process efficiency. The hydrocyclones were operating with a much higher solids percentage than the ideal, which made its classification worsen and a lot of already ground ore was being re-fed into the mill, making the process a lot less efficient. However, in order to add the right amount of water to the stream feeding the cyclone, it was needed to know the amount of solids in this stream. Measuring this quantity is no easy task, and the most common equipment for that measurement is based on radioactive beam emission which faces severe norms and legislation. To mathematically model this process is another difficult task since the model for the milling process is both very complex and requires a lot of fitting parameters, which requires a lot of data to be determined. So, by using a Neural Network as a black-box model for the system we mitigated this physical complexity.

The process variables measured in real-time and that were used are:

- Fresh ore feed.
- Solution water flow rate for the crusher.
- Solution water flow rate for the hydrocyclone.
- Crushed solid stream with a certain granulometry.
- Pressure in the hydrocyclones.
- Power consumed by crusher.

Using the dataset contained in the Dados_Moagem.xlsx, the following tasks were performed:

a. Build a Neural Network model using the Keras (TensorFlow) library to predict the percentage of solids. Use 60% of the data for training and 40% for testing. Note: Before training the model, normalize the independent variables.
b. Adjust the model using 100 epochs.
c. Make an analysis of the evolution of the mean absolute error (MAE) in each epoch.
d. Perform the forecast of % solids for the following dataset:
   i. Fresh ore feed = 230 kg
   ii. Solution water flowrate for crusher = 20 m^3/h
   iii. Solution water flowrate for hydrocyclone = 200 m^3/h
   iv. Crushed solid stream with a certain granulometry = 65%;
   v. Pressure in hydrocyclones = 0.55 psi;
   vi. Power consumed by crushers = 3300 W.

First, the data file is downloaded and normalized using the Z-score method, which is one feature engineering technique. Then, the data set is divided into training and testing, with a ratio of 60/40%. The next step is to build a neural network with 2 inner layers with 64 neurons each. The activation function is RELU. The model is generated with 100 epochs, Fig. 12. The mean absolute error (MAE) was equal to 0.0096. The forecast with normalized variables was 80.07% of solids.

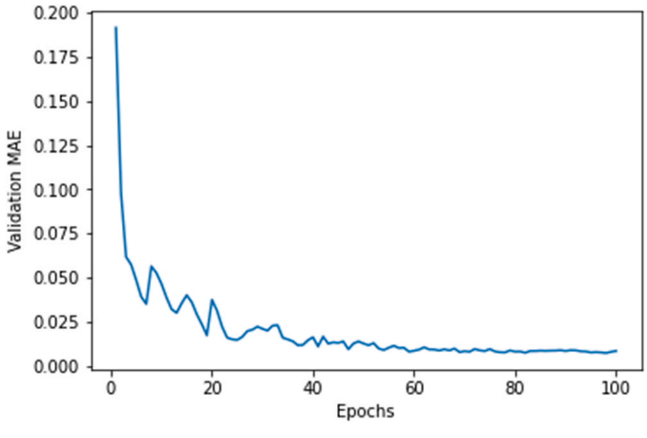


Fig. 12. Evolution of the MAE with the epochs.

5. Students' evaluation

An anonymous online survey was conducted to evaluate students' perceptions regarding three key features: 1) acceptance of the flipped learning strategy, 2) level of engagement, and 3) the influence of the COVID-19 pandemic on their outcomes. Out of a total of 51 students, 27 chose to participate in the survey, providing valuable insights. The results can be seen in Fig. 13.

Regarding the flipped learning strategy, the whole class considered it to be adequate, with 77.8% finding it absolutely appropriate, and 22.2% finding it appropriate. It's worth noting that this year, all activities had to be conducted remotely due to the COVID-19 pandemic.

The level of engagement required by the course was found to be as expected by 70.4% of the students, while 25.9% perceived it to be higher than expected. This positive response indicates that the students were actively involved in the course.

An unexpected result emerged from the survey regarding the influence of the pandemic on the course. Surprisingly, 30.8% of the students felt that the pandemic impaired the course, while 38.4% were indifferent to its influence. Besides, 30.8% indicated that the pandemic improved the course. The remarkably similar responses suggest that students have likely adapted to this new reality, finding ways to cope with the challenges posed by the pandemic.

6. Conclusions

The hands-on ML activities proposed for this chemical engineering course, encompassing clustering, classification, and regression (including Artificial Neural Networks (ANN)), present a compelling and relevant set of skills for undergraduate students in Chemical Engineering.

Clustering serves as an excellent starting point for representing datasets using Machine Learning (ML) techniques. The evaluation of the iris dataset proved to be an exceptional choice, allowing students to analyze the results quickly and accurately. This widely available dataset provides ample information that aids in contextualizing the modeling process. The impactful results obtained from dendrograms further enrich the ML analysis.

Classification holds significant importance within ML. The wine dataset captured the attention and curiosity of the students, elevating the activity's intrigue. The heatmap emerged as a highly meaningful method for assessing correlations between attributes. We applied decision trees in conjunction with SVM, evaluating the classification parameters using the confusion matrix, which includes the analysis of true positives, false positives, true negatives, and false negatives. Employing multiple algorithms contributes to the robustness of the classification process, as one method can corroborate the results of another. This led to a captivating discussion regarding the precautions necessary when applying ML algorithms.

Furthermore, we developed a neural network for a grinding process using data retrieved from an industrial plant unit. This activity provided students with the opportunity to evaluate the variables of an actual chemical process. Despite its simplicity, the exercise sparked a valuable discussion around the direct application of Machine Learning in chemical engineering applications.

Ultimately, we have successfully achieved our learning objectives, with students acquiring fundamental concepts about Machine Learning (ML). Throughout the learning process, Google Colab proved invaluable, offering an easy-to-use environment for thorough dataset evaluation, with the appropriate implementation, and analysis of various

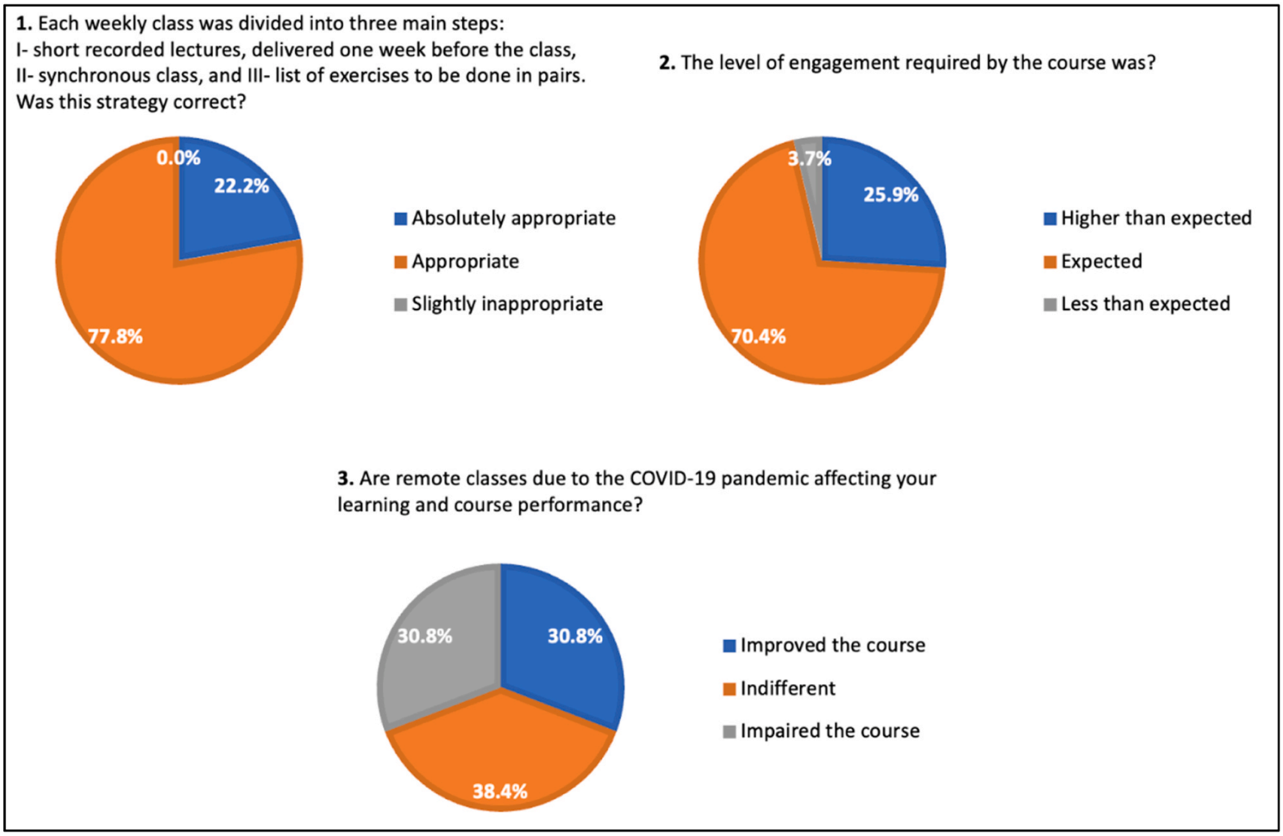


Fig. 13. Pie graphs for the surveys, with the questions and respective answers.

algorithms. By covering this range of activities, students gain valuable insights into various ML techniques while applying them to relevant real-world scenarios, enhancing their understanding of ML within the context of Chemical Engineering.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - Projeto Institucional de Modernização (PIM) – Capes/Fulbright (PMG-2018981270P).

References

Adair, D., Bakenov, Z., Jaeger, M., 2014. Building on a traditional chemical engineering curriculum using computational fluid dynamics. Educ. Chem. Eng. 9 (4), e85–e93. https://doi.org/10.1016/j.ece.2014.06.001.

Alhajeri, M.S., Wu, Z., Rincon, D., Albalawi, F., Christofides, P.D., 2021. Machine-learning-based state estimation and predictive control of nonlinear processes. Chem. Eng. Res. Des. 167, 268–280. https://doi.org/10.1016/j.cherd.2021.01.009.

Alves, V., Gazzaneo, V., Lima, F.V., 2022. A machine learning-based process operability framework using Gaussian processes. Comput. Chem. Eng. 163, 107835 https://doi.org/10.1016/j.compchemeng.2022.107835.

Bisong, E., 2019. Google Colaboratory. Building Machine Learning and Deep Learning Models on Google Cloud Platform. Apress, Berkeley, CA.,. ⟨https://doi.org/10.1007/978-1-4842-4470-8_7⟩.

Bryson, J.J., 2019. The Future of AI's Impact on Society. MIT Technology Review,.

Campos, K.R., Coleman, P.J., Alvarez, J.C., Dreher, S.D., Garbaccio, R.M., Terrett, Tillyer, R.D., Truppo, M.D., N.K.,.Parmee, E. R., 2019 The importance of synthetic chemistry in the pharmaceutical industry. Science, 363(6424), eaat0805. DOI: 10.1126/science.aat0805.

Dobbelaere, M.R., Plehiers, P.P., Van de Vijver, R., Stevens, C.V., Van Geem, K.M., 2021. Machine learning in chemical engineering: strengths, weaknesses, opportunities, and threats. Engineering 7 (9), 1201–1211. https://doi.org/10.1016/j.eng.2021.03.019.

Dua, D., Graff, C., 2017. {UCI} Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences,. ⟨http://archive.ics.uci.edu/ML⟩.

Galán, F.S., Mora, H.M.B., Heredia, A.T.J., Peñuela, L.L.S., Barrios, A.F.G., 2018. Everyday chemical product design as platform for teaching transport phenomena. Educ. Chem. Eng. 25, 9–15. https://doi.org/10.1016/j.ece.2018.09.001.

Girmsom, E., 2016, MIT OpenCourseWare, 6.0002 Introduction to Computational Thinking and Data ScienceFall, ⟨https://ocw.mit.edu⟩, (Acessed 29 December 2020).

Guttag, J., 2016. Introduction to Computation and Programming Using Python: with Application to Understanding Data. MIT Press,, Cambridge.

Kaggle, ⟨https://www.kaggle.com/⟩. (Acessed 06 March 2021).

Mowbray, M., Savage, T., Wu, C., Song, Z., Cho, B.A., Del Rio-Chanona, E.A., Zhang, D., 2021. Machine learning for biochemical engineering: A review. Biochem. Eng. J. 172, 108054 https://doi.org/10.1016/j.bej.2021.108054.

Nakama, C.S.M., Siqueira, A.F., Vianna Jr., A.S.V., 2017. Stochastic axial dispersion model for tubular equipment. Chem. Eng. Sci. 171, 131–138. https://doi.org/10.1016/j.ces.2017.05.024.

Oliveira, C.J., dos Santos, M.T., Vianna Jr, A.S., 2022. A proposal to cover stochastic models in chemical engineering education. Educ. Chem. Eng. 38 (2022), 86–96. https://doi.org/10.1016/j.ece.2021.12.002.

Quaglio, M., Roberts, L., Jaapar, M.S.B., Fraga, E.S., Dua, V., Galvanin, F., 2020. An artificial neural network approach to recognise kinetic models from experimental data. Comput. Chem. Eng. 135, 106759 https://doi.org/10.1016/j.compchemeng.2020.106759.

Russell, S., Norvig, P., 2002. Artificial intelligence. A Modern Approach. Prentice Hall,, Upper Saddle River, NJ, USA.

Satopaa, V., Albrecht, J., Irwin, D., Raghavan, B., 2011. Finding a" kneedle" in a haystack: Detecting knee points in system behavior. 2011 31st international conference on distributed computing systems workshops. IEEE,, pp. 166–171.

Web references

Souto-Melgar, Natacha, Jackqueline Steinman-Ptacek, and Andie Veeder. "A hands-on experience to study membrane technology developed by undergraduate chemical engineering students." 2022 ASEE Annual Conference & Exposition. 2022.

Venkatasubramanian, V., 2019. The promise of artificial intelligence in chemical engineering: Is it here, finally? AIChE J. 65 (2), 466–478. https://doi.org/10.1002/aic.16489.