Applications of machine learning in supercritical fluids research

Lucien Roach, Gian-Marco Rignanese, Arnaud Erriguible, Cyril Aymonier

Applications of Machine Learning in Supercritical Fluids Research

Lucien Roach,[a] Gian-Marco Rignanese,[b] Arnaud Erriguible,[a,c] Cyril Aymonier*,[a]

[a] *Univ. Bordeaux, CNRS, Bordeaux INP, ICMCB, UMR 5026, F-33600 Pessac, France.*
[b] *UCLouvain, Institute of Condensed Matter and Nanosciences, Chemin des Étoiles 8, B-1348 Louvain-la-Neuve, Belgium.*
[c] *I2M-UMR5295, Université de Bordeaux, Bordeaux INP, CNRS, site ENSCBP, 16 avenue Pey-Berland, Pessac Cedex, France*
* Corresponding author: cyril.aymonier@icmcb.cnrs.fr.

ABSTRACT

Machine learning has seen increasing implementation as a predictive tool in the chemical and physical sciences in recent years. It offers a route to accelerate the process of scientific discovery through a computational data-driven approach. Whilst machine learning is well established in other fields, such as pharmaceutical research, it is still in its infancy in supercritical fluids research, but will likely accelerate dramatically in coming years. In this review, we present a basic introduction to machine learning and discuss its current uses by supercritical fluids researchers. In particular, we focus on the most common machine learning applications; including: (1) The estimation of the thermodynamic properties of supercritical fluids. (2) The estimation of solubilities, miscibilities, and extraction yields. (3) Chemical reaction optimization. (4) Materials synthesis optimization. (5) Supercritical power systems. (6) Fluid dynamics simulations of supercritical fluids. (7) Molecular simulation of supercritical fluids and (8) Geosequestration of CO_2 using supercritical fluids.

Keywords: *Machine Learning, Optimization, Supercritical Fluids, Data Intensive Computing, Regression*

HIGHLIGHTS

- Supercritical fluids research using machine learning has grown rapidly in recent years
- In the past machine learning was used mainly for thermodynamics, now prominent in new areas
- Machine learning can accurately estimate thermophysical properties of supercritical fluids
- Modelling of supercritical fluids can be accelerated using machine learning models
- Many unexplored opportunities exist for machine learning in supercritical fluids research

1. Introduction

Data-intensive computing has been hailed as the fourth paradigm of science, a new approach in which big data and computational approaches will revolution how scientists make discoveries [1,2]. The preceding three paradigms; *empirical observation and experimentation*, *analytical and theoretical frameworks*, and *computational science and simulation* being the first, second, and third, respectively. In the fourth paradigm, science advances through the collection, curation, and analysis of large data sets. Machine learning is an approach that will inevitably play a significant role in the analysis of these datasets.

Machine learning is finding increasing application in the sciences and is better established in certain fields such as organic chemistry and pharmaceutical research [3,4]. Interest in the use of machine learning has increased dramatically over the last twenty years as the required computational power for its application has become progressively more affordable. Research implementing machine learning can be found throughout the sciences, with many review articles discussing its use in disparate range of fields including the chemical sciences [5–10], biological sciences [11–13], condensed matter physics [14], fluid dynamics [15,16], nanoscience [17–19], and materials science [20–27].

In this Review, we discuss the use of machine learning as a tool for the modelling and estimation of systems and processes using supercritical fluids. In **Section 1.2,** we discuss what machine learning is and, in **Section 1.3,** we discuss the different machine learning approaches that have been implemented in supercritical fluids research. In **Section 2**, we review the specific applications of machine learning that have been used in this domain. These include: (1) The estimation of solubilities, miscibilities, and extraction yields. (2) The estimation of the thermodynamic properties of supercritical fluids. (3) Chemical reaction optimization. (4) Materials synthesis optimization (4) Supercritical power systems. (5) Fluid dynamics simulations of supercritical fluids. (5) Molecular simulation of supercritical fluids and (6) Geosequestration of CO_2 using supercritical fluids. Finally, in **Section 3**, we offer our conclusions and perspectives on the implementation of machine learning in the research of supercritical fluids.

1.1 Introduction to Machine Learning.

Machine learning refers to a group of programming techniques that allow models to be 'learned' from input data. It can be used for two key tasks « *classification* » and « *regression* ». In regression, a mathematical model which estimates outputs from a set of input data is trained using an existing dataset. Regression models can approximate the relationships between input parameters allowing prediction of outputs throughout a studied range of inputs. In this context, 'prediction' is used according the statistical definition, whereby the information gained from a sample of a dataset is applied to the entire range of inputs covered by the dataset. This makes them a powerful tool for identifying optimized experimental conditions, which can be used alongside other techniques, such as theoretical modelling, to accelerate the process of scientific discovery. By comparison, in classification models, data is sorted into discrete categories based on a set of properties specific to each datapoint, the thresholding conditions for classification are learned by the model from a dataset. The large majority of examples in this Review are examples of regression, where a machine learning algorithm has been used to estimate the outcome of a physical process based on a given set of input conditions, although some examples of classifiers are discussed.

A distinction needs to be made between « *supervised* » and « *unsupervised* » learning. In supervised machine learning, the objective is to estimate a specific output from input data. For classification tasks, this means the training dataset has already been labeled. In unsupervised learning, machine learning is being used to analyze an unlabeled dataset, by identifying clustering or relationships between variables in the dataset. Unsupervised learning is particularly important in tasks such as identifying trends in large datasets, image processing, or multivariate analysis. The vast majority of the examples discussed in this review use supervised machine learning, however unsupervised machine learning will likely find much wider use in the future applications.

In a well-designed machine learning procedure, a large dataset is split into three subsets, the training, testing, and validation sets. Commonly used proportions range from 60 - 20 - 20% to 80 - 10 - 10% for these, respectively. The training set is used for optimizing the model

such that the difference between the model's outputs and the training data is minimized. The testing set is then used to evaluate the performance of the trained model. The validation set is used for the optimization of the model hyperparameters, which are values that control either the complexity of the model, such as the architecture of the neural network, or the learning process, such as the learning rate of the same.

The data quality and quantity used to train a machine learning model are imperative to the reliability of trained models. The required dataset size is dependent on the complexity of the problem to be modelled, accurately capturing the character of the relationships of many variables with high levels of interdependency and complex behaviors requires more data than simple problems. This is complicated by data availability, acquiring large datasets may be prohibitively expensive, due to the nature of the required experiments or only limited literature data may exist.

The quality of the dataset can be improved through data curation (or preprocessing) strategies such as; (1) Data cleaning, the removal of clearly erroneous datapoints which could mislead the training process. (2) Normalization (or *z*-score standardization), the rescaling of data to as a fraction of the total range (or as the number of standard deviations from the mean), which allows more rapid convergence during the training process. (3) Feature selection, eliminating redundant variables in the dataset can increase the efficiency of the learning process. Multivariate analysis techniques, such as principal component analysis are powerful tools for this, allowing data to be represented in a more compact fashion.

A common problem encountered when analyzing noisy datasets with machine learning is overfitting. This is essentially fitting the noise in a particular system over the 'true' underlying relationship between inputs and outputs. An example of overfitting by a non-linear regression model is shown in the red curve of Figure 1b. Overfitting is most commonly resolved through *k*-fold cross-validation, in which the dataset is partitioned into *k* subsets (or 'folds'). The model is initially trained on *k* - 1 subsets, before being validated against the remaining subset, this process is repeated *k* times, such that each subset is used as the validation set. The performance metrics are collected after each iteration and averaged to give the performance

of the model overall. This process reduces the bias from individual subsets, and provides an estimate of how well the model generalizes to new data. While the vast majority of the literature reviewed here resorts to *k*-fold cross-validation, other suitable approaches exist, such as the leave-*p*-out, leave-one-out, holdout and Monte Carlo cross-validations methods, see Ref [28] for a deeper discussion of cross-validation methods and the criteria for selecting them.

A key goal of many applications of machine learning is the construction of generalizable models. Specifically, this means, training models to make accurate estimations beyond the relatively narrow range of conditions within the training set (although still with the actual ranges of the input parameters within the training set). This requires large high-quality datasets, featuring parameters that allow the system to be understood in general terms. For instance, in the case of solubility estimation, this requires including the description of the systems in terms of the physical properties of the solute and solvent molecules, such as the acentric factor, dipolar moment, molar volumes, molecular composition, etc. Allowing the estimation of the solubilities of many solutes in many different solvents, rather than just estimations for a narrow range of solutes and solvents. Being able to make predictions beyond currently available knowledge makes machine learning an extremely powerful tool for advancing the frontiers of science.

1.2 Machine Learning Algorithms.

Several common methods exist for the implementation of machine learning. In this Review, the vast majority of discussed examples use regressors for their analysis, hence we only introduce regressors here. The use of machine learning for classification is a broad topic in its own right and is important in tasks such as image analysis, text recognition, or data partitioning, but is largely beyond the scope of this Review. Regressors can be broadly grouped into regression algorithms, decision tree algorithms, nearest-neighbor methods, and artificial neural networks.

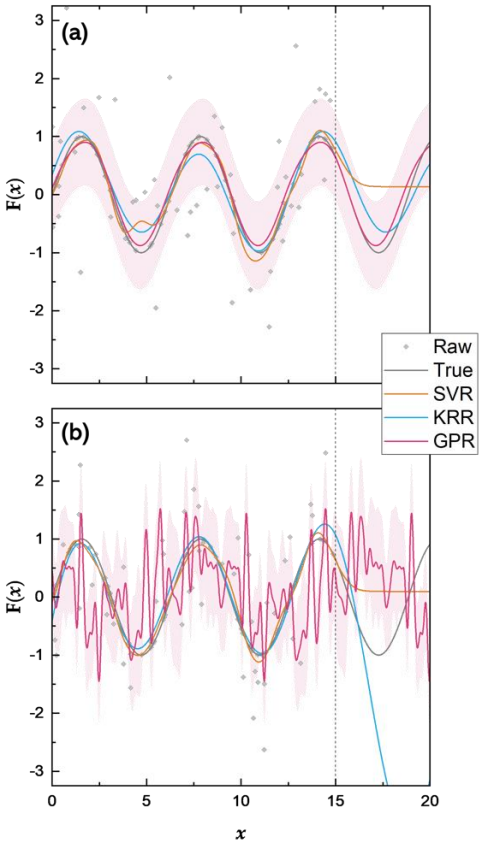

Figure 1. Example implementations of support vector regression (SVR), kernel ridge regression (KRR), and Gaussian process regression (GPR) non-linear regression modeling to a sine featuring generate noise between $0 < x < 15$. The seed used to generate the random noise has been changed between (a) and (b), and the same models are applied to both datasets. It can be seen that SVR fails to estimate the function outside the training range, and KRR also fails outside the training range in (b). GPR suffers from overfitting in (b). Data in these figures were generated using the *scikit-learn* Python module [29,30].

Regression Algorithms.

These algorithms fit a mathematical function to estimate a continuous output based on a set of inputs. These can be grouped into linear and non-linear regression.

Linear regression assumes a linear relationship between independent inputs and the output [31]. Several linear regression algorithms exist such as ordinary least squares (OLS), least absolute shrinkage and selection operator (LASSO), and multi-linear regression (MLR).

Non-linear regression conversely assumes a non-linear relationship between inputs and outputs. A common example used in machine learning studies is support vector regression (SVR) which uses a kernel function to transform the data into a higher-dimensional feature

space where the data may be a linear function or hyperplane which fits the transformed data [32,33]. A range of SVR kernels exist (such as linear, radial basis function, polynomial, sigmoidal, Gaussian…) which describe the expected relationship between the inputs and the output and should be selected dependent on the particular problem. An alternative variant of SVR which is commonly used is the least-squares support vector regression (LSSVR) algorithm which optimizes the model by minimizing the sum of the squared errors between the output and 'true' result (rather than maximizing the distance between the hyperplane and closest transformed points in SVR) [33,34]. LSSVR is more robust to outliers, but struggles with non-linear relationships between inputs and outputs. Additionally, kernel ridge regression (KRR) is another common non-linear regression method, which is similar to SVR, but uses linear combinations of non-linear kernel functions to transform the data. The validity of the models produced by SVR, LSSVR, and KRR is highly dependent on the choice of kernel functions used [33,35]. Another example, Gaussian process regression (GPR), assumes that the input data has a Gaussian distribution about a kernel covariance function [36]. This algorithm is typically robust when used with noisy input data, but can have a high computational load when used with large datasets. Comparisons of SVR, KRR, and GPR applied to noisy sinusoidal data are shown in Figure 1.

Tree-Based Algorithms.

Named after their tree-like structure, decision tree (DT) algorithms can be used for both classification and regression [37]. In the case of regression, the DT algorithm splits data using a series of « If $\{x\}$, Then $\{y\}$, Else $\{z\}$ » decision rules. Hence, DT outputs are a piecewise approximation of the 'true' output function, rather than a continuous function. (Figure 2) With each section of the approximation corresponding to the satisfaction of a criteria set. Unlike many machine-learning approaches, the decision-making rules can be observed making the interpretation of the trained DT model relatively straightforward. DTs suffer several drawbacks; they are not well suited for datasets with high dimensionality or large numbers of features, where they can easily overfit data. They can be highly sensitive to the training set, training with different parts of the same dataset can result in radically different tree structures. For noisy datasets, precautions must be taken to ensure that a suitable minimum number of data

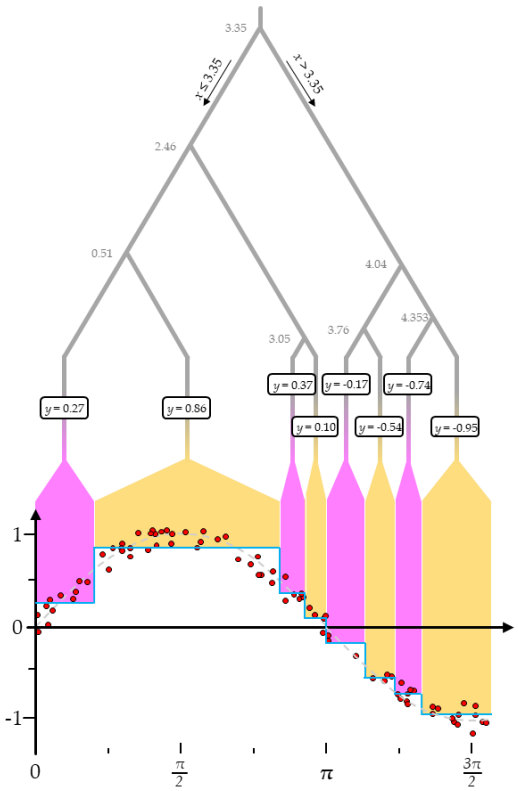

Figure 2. Example implementation of a decision tree (DT) on a noisy sine wave (red dots = fitted data, grey dashed line = 'true' sine function). A decision is made at each fork as to whether the input is less than (left branch) or more than (right branch) the value shown. The net result is a piecemeal estimation of the function (blue line). Data in this figure was generated using the scikit-learn Python module *[29]*.

points per node is used. Each additional layer of if-then-else decision rules doubles the required number of datapoints in the training set. DTs are highly sensitive to biases within the dataset, data must be well balanced in the parameter space being explored.

Ensemble methods that combine several DTs have been developed which avoid several of these potential problems. For instance, random forest (RF) algorithms, work through the construction of several DTs during the training process [38]. The output is the mean of all the values returned by the DTs within the RF correcting for problems due to overfitting of the training set. Alternatively, gradient-boosted (GB) algorithms, train an additional DT on the erroneous outputs of the initial DT and repeat this process until the DT which accurately represents the content of the testing/validation sets is produced [39,40]. Also, the AdaBoost (AB) algorithm, follows a similar process to GB algorithms, however increasing weight is given to erroneous outputs on each iteration of the process, forcing the algorithm to focus on these instances [41].

Nearest Neighbor Methods.

These algorithms use the closest data points in the training set to make predictions [42–44]. The most commonly used example is the *k*-nearest neighbors (*k*NN) algorithm, where the *k*-closest data points are used to predict the output value for a given set of inputs. Each neighbor's distance from the given inputs can be used as weights. *k*NN analysis is simple to implement, but it requires the storage of all training data in memory, and for large, high-dimensionality datasets, distance calculations can become prohibitively expensive.

Artificial Neural Networks.

Artificial neural networks (ANNs) consist of a series of interconnected nodes (or 'neurons') organized in layers (Figure 3) [45]. These layers are arranged into an input layer, which consists of the independent variables in the system, the hidden layers, which process the input data, and the output layer, which consists of the dependent variables. Each neuron in the hidden layer(s) receives inputs and produces a single output which can be received by multiple other neurons. Each neuron's output is generated by processing the inputs with a mathematical 'activation' function. The inputs of each neuron (or equivalently the outputs of the previous neuron) are weighted. These weights can be adjusted to decrease or increase the influence of the input on the activation function. The relationships between inputs and outputs is then represented as the superposition of the respective activation functions located at each node in the network [46]. During the training of the neural network, it is these parameters that are modified to minimize the error between the ANN output and the true output signal. The choice of optimizer affects the training speed and accuracy of the ANN. Several optimization algorithms exist such as gradient descent, Adam, or Adagrad, among others, and the choice of the algorithm is largely dictated by the nature of the problem and ANN. Simple optimizers such as gradient descent methods are effective, but can struggle with noisy datasets or complex network architectures, where an algorithm which is capable of varying the learning rate, such as Adam, may perform better. Ultimately, the best strategy for identifying the best optimizer is testing the ANN performance using different optimizers on a validation dataset.

Most neural networks propagate in the forward direction (feedforward ANNs), that is all outputs from a layer are the inputs for the neurons in the next layer in the direction of the output layer. The most common method of training feedforward ANNs is backpropagation,

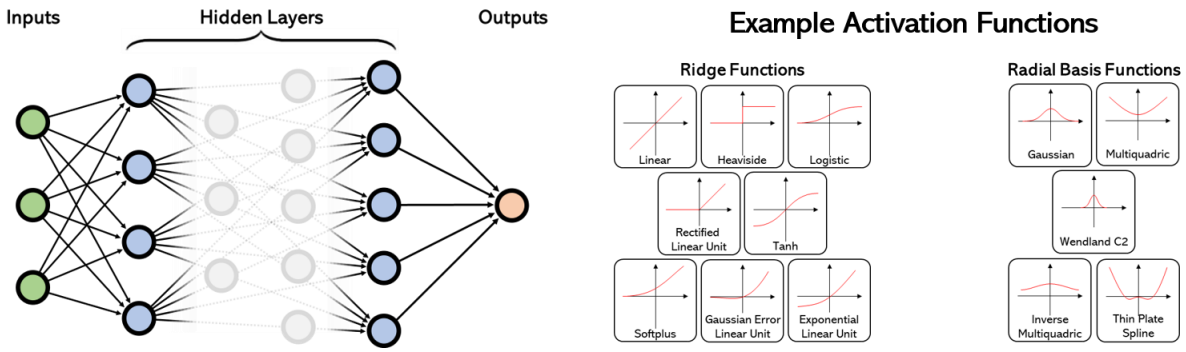

Figure 3. Typical representation of an artificial neural network (ANN). A given set of inputs (green circles) are fed into a network of nodes organized in layers (blue circles) before calculating an output (orange circle). Each node within the hidden layer represents a mathematical 'activation' function. Typically, these are either ridge or radial basis functions. Examples of activation functions are given to the right of the main diagram. The exact forms and weights used in the functions within the hidden layers are determined during training.

where error is calculated for the output, and the weights of the inputs in the last hidden layer are adjusted. Error is propagated backward through the network adjusting the weights layer-by-layer in reverse order. ANNs exist where the outputs of neurons later in the network are the inputs for neurons earlier in the network allowing for the outputs of some nodes to affect their own future output, such networks are called recurrent neural networks.

A tradeoff is made through the use of ANNs, since they can be used to create accurate models of complex relationships within datasets, but this comes at the cost of potentially losing an understanding of the underlying relationships. With the trained model offering only limited interpretability in the relative importance of the inputs, but without a direct expression of this relationship to the output. It is for this reason that ANNs are sometimes referred to as a 'black box' methodology, although the use of this term is not universally accepted as appropriate for ANNs [47]. This could be problematic within domains such as the physical sciences, where understanding the physical and chemical processes are often key objectives. The development of applications of machine learning which do not act as black boxes are thus desirable if understanding of the nature of trained models is to be prioritized. However, in systems where the relationships between dependent variables are complex and no accurate physical models exists, ANN models can be used to allow work to advance without this physical underpinning (i.e. by predicting previously unknown results which can be verified and investigated further through complementary techniques). There has been increasing amounts

of work seeking to use trained machine learning models to discover the underlying physical principles driving modeled systems through 'inverse design' [19,48,49].

A wide variety of activation functions exist which can be grouped roughly into ridge functions, radial basis functions, and "folding" functions (the latter are primarily used in convolutional neural networks for image recognition problems and are not discussed further here). Ridge functions typically produce outputs that are close to zero for small inputs, and output larger values for large inputs. However, the mathematical relationship between inputs and outputs can vary greatly, with some functions saturating with increasing input, and others growing indefinitely. Most ridge functions produce values close to zero for negative inputs, although a few exceptions such as the linear and tanh ridge functions exist. Radial basis functions are symmetric about the y-axis and typically output large values close to $x = 0$ and small values far from the origin (or *vice versa*). Neural networks using these are known as radial basis function neural networks [50]. Linear combinations of radial basis functions are very efficient at approximating complex mathematical functions making them well suited for used as activation functions.

A common application of ANNs is in hybrid systems such as adaptive neuro-fuzzy inference systems (ANFIS), which combine ANNs and fuzzy logic systems [51,52]. Fuzzy logic differs from DT style « If $\{x\}$, Then $\{y\}$, Else $\{z\}$» binary logic statements, by allowing intermediate values between extremes (i.e., partial membership in categories), allowing for imprecision and uncertainty to be included in the logic system. In ANFIS, the data is first transformed using fuzzy logic statements, and an ANN is then used to learn the parameters of the model. After training on the fuzzy logic model, the trained ANN can then be used to calculate outputs for new input data.

The number of hidden layers in an ANN and the number of neurons in each layer are key to determining the complexity of the relationships that it can model. ANNs featuring few hidden layers and a small number of neurons require less training time but may fail to accurately describe complex relationships (i.e., data will be 'underfitted'). Conversely, having many hid-

den layers, enables complex relationships to be modeled by the network, but can lead to overfitting without a large enough training set. There is no universal strategy to determining the correct number of hidden layers and neurons in an ANN, however several common strategies exist. Firstly, there are brute-force techniques, such as simple trial-and-error testing of different combinations of layers and neurons which are evaluated using a validation set. Grid searching, which systematically searches all possible combinations of ANN structures from a predefined set of possible layer numbers and neuron numbers. Randomized searching performs the same process but with random combinations of these. Alternatively, there are techniques which use machine learning to select the optimal number of hidden layers and neurons and optimal hyperparameters for an ANN, such as AutoML.

Deep Learning.

ANNs featuring more than three hidden layers are often referred to as 'deep' neural networks (DNNs) [53,54]. The use of many layers allows DNNs to approximate much more complex patterns and relationships that smaller ANNs, how this comes at a significantly increased computational cost, typically require large datasets to the 'pre-train' the network. The initial layers of such networks typically capture basic patterns in the data, with later layers capturing more subtle behaviors with higher order dependencies, allowing a more precise approximation of the data. Examples of DNNs include: Graph ANNs which feature connections between the nodes of each hidden layer, allowing the neighboring nodes in the layer to affect each other's value through a message-passing algorithm [55]. Or convolutional ANNs, which are capable of processing higher dimensional data (such as images) in which the inputs are matrices. These are significantly more complicated than the majority of machine learning approaches used in this review, neurons are organized in convolutional, activation, pooling, and fully connected layers. Which combine to extract features from the input dataset and reduce its dimensionality. Readers are referred to Refs. [54,56,57] for a more comprehensive review of these forms of machine learning.

Interpretative Language Models.

Interpretative language models are a very recent development in the field of machine learning combing natural language processing and machine learning techniques. These models are capable of interpreting text and generating a natural language response. These models are trained on very large datasets allowing them to learn language structures, typically using deep learning algorithms to build their language models. The most obvious applications of these models are in roles requiring natural language such as chatbots, but they have found application in the physical sciences in roles such as chemical property estimation, drug discovery, reaction optimization, and toxicology [58]. These models have a number of advantages over other approaches such as DNNs, such as handling variable-length inputs and utilizing the pre-training of such models on large text datasets. The implementation of these models is very recent, hence there are not many examples within the domain of supercritical fluids research, but they will likely become much more prominent in the coming years.

1.3 Optimization from Trained Machine Learning Models.

Trained machine learning models can be coupled to optimization algorithms to allow improved conditions to be identified. The fast computation and generalizability offered by properly trained models allow for outputs to be used as figures of merit for optimization. This can be true for multiple outputs simultaneously if multiparameter optimization is performed. A wide range of optimization algorithms exist which are suitable for this, such as particle swarm optimization [59–63], genetic algorithms [64,65], gradient descent algorithms [66], and Adam [67,68] amongst others. The combination of generalizable machine learning models and optimization offers a direct route toward accelerating the discovery of improved conditions for a range of processes in supercritical fluids research. These range from improved or novel syntheses to higher efficiency power systems, or higher yield extractions.

All of the aspects of building and training a machine learning model discussed above are demonstrated as a workflow in Figure 4. This applies to all the machine learning models discussed so far.

1.4 Online Machine Learning.

So far, our discussion of machine learning has focused entirely on offline machine learning, where a model is trained in a single instance from a pre-generated dataset. However, machine learning can be for online optimization also, where data is generated sequentially (e.g. because of experiments being performed in series) and the model trained continuously on the incoming data. Online versions of several of the models discussed above exist for this purpose as well as, prominent optimization methods such as stochastic gradient descent. The outputs of such models can be used to direct future experiments allowing systems to incrementally improve towards an optimized state.

Because online machine learning requires retraining at each data acquisition step, these approaches can be much more resource-intensive than offline approaches, although potentially at the benefit of finding optimized conditions using less experimental resources. Online approaches face a number of challenges, in case where the identification of global minima is non-trivial, these approaches tend toward identifying local minima. Mitigations against this include the introduction of a level of randomness to condition selection (such as in stochastic gradient descent methods) where a wider range of conditions are explored or sampling multiple new data points at each training step. These approaches can also be sensitive to noise or outliers during data generation, which is commonly dealt with through the incorporation of Bayesian probability into these models [69]. These approaches are extremely interesting for the development of autonomous labs which can perform experiments and develop syntheses without user input. We could find no published examples of online machine learning for supercritical fluids research, however such approaches have appeared in a wide range of other domains, and are likely to appear in the near future [17,69,70].

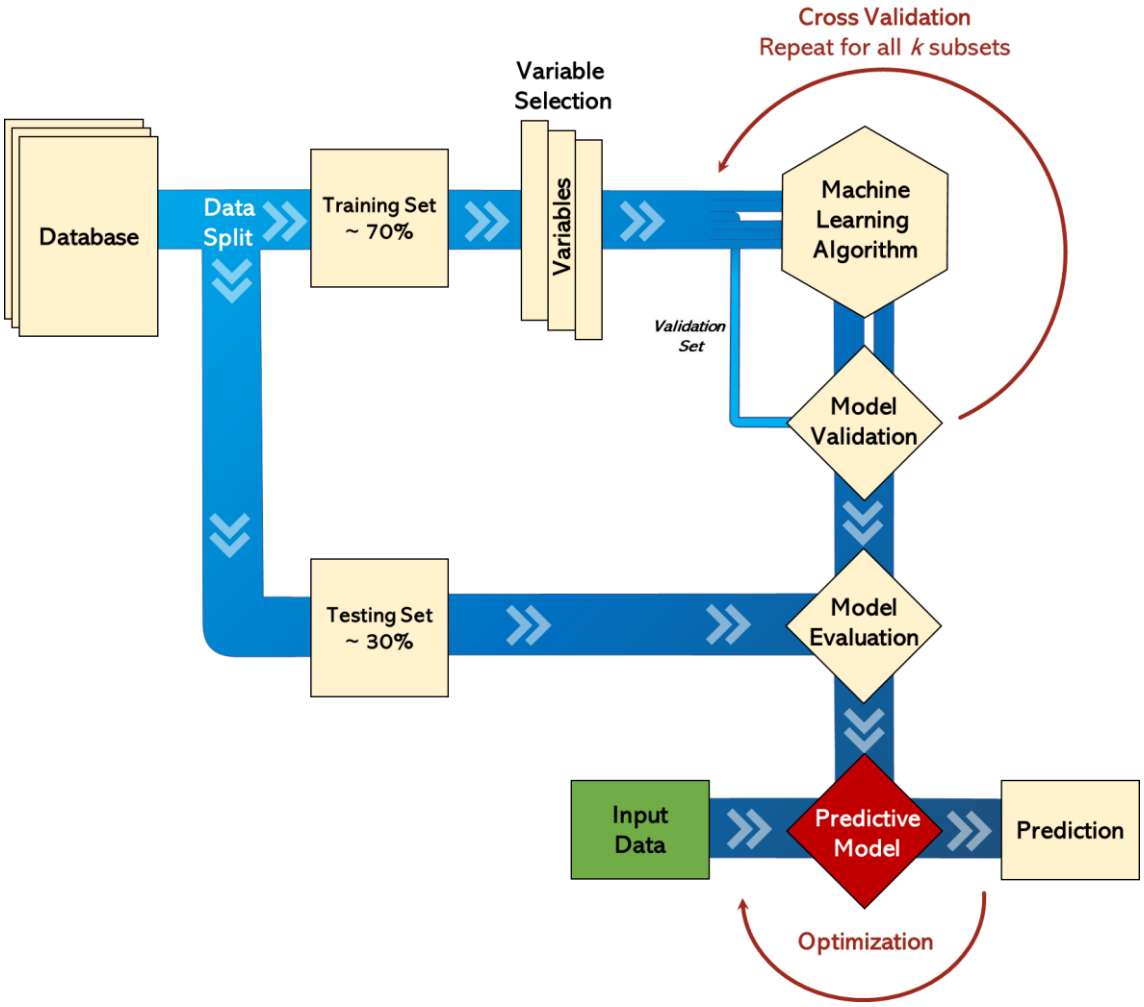

Figure 4. Example work flow for the implementation of offline machine learning models. A dataset is prepared and then split into a testing and training set. Input (and output variables) are identified). The training data is then split into *k* subsets, and a single subset kept as the validation set. The machine learning algorithm is trained on the remaining *k*-1 subsets, before being validated against the validation subset. This process is repeated *k*-fold times. After cross-validation the best model is tested against the unseen testing set. Finally, the trained model can make predictions for new previously unseen input conditions. If coupled with an optimization algorithm, this model can be used to find ideal conditions for studied systems. Figure modified with permission from Ref. *[71]* Copyright 2021 MDPI.

2. Machine Learning in Supercritical Fluids Research.

In this section, we review the various areas of supercritical fluids research in which machine learning has been implemented including (1) The estimation of the thermodynamic properties of supercritical fluids. (2) The estimation of solubilities, miscibilities, and extraction yields. (3) Chemical reaction optimization. (4) Materials synthesis optimization. (5) Supercritical power systems. (6) Fluid dynamics simulations of supercritical fluids. (7) Molecular

simulation of supercritical fluids and (8) Geosequestration of CO_2 using supercritical fluids. The distribution of this studies is not homogeneous, there were far more machine learning studies applied to estimation of solubilities and extraction yields (52 references) than any other domain. Other areas such as thermodynamic property estimation (29 references), hydrothermal gasification (14 references), biodiesel production (7 references) and supercritical power systems (13 references). This distribution of is largely reflective of two things the availability of data to train the models and the relative complexity of the problems being modeled in each domain.

2.1 Thermodynamic Properties of Supercritical Fluids.

Machine learning has been applied to predict the properties of supercritical fluids. These studies cover a wide range of fluid properties including state variables, thermodynamic equilibria, diffusivities, and interfacial tensions among others. Knowledge of these properties is crucial to a wide range of applications and hence, machine learning has great potential to accelerate and improve our current approaches to their calculation. A summary of several examples is given in Table 1 [72–94].

State Variables.

The most fundamental properties of a fluid are those of the thermodynamic state variables (P; molar volume, V; T, and particle number, n), from which the equilibrium state of the system can be described. Several authors have sought to apply machine learning to the prediction of these for a range of supercritical fluids. For instance, Liu *et al.* have used SVR to predict *PVT* curves from existing data for pure fluids (H_2O, CO_2, H_2) and their mixtures in the near- and supercritical regions of phase space [88]. Experimental and simulation data regarding the thermodynamic properties of H_2O-CO_2-H_2 ternary mixtures are presently scarce. The authors obtained data for the pure fluids from the National Institute of Standards and Technology (NIST) standard reference database and produced their own data for mixtures using

Table 1.
Properties of supercritical fluids calculated using machine learning.

Input Parameters	Predicted Parameter(s)	Supercritical Fluid	Machine Learning Technique(s)	Dataset Size	Train - Test - Valid. Split (%)	REF
T_r, P_C, ω, ϕ_{CO_2}	T_B, T_D	CO_2 mixtures	ANN	316	75 - 0 - 25	[72]
T, P	ρ	CO_2	ANN	5895	not specified	[73]
T, P, N_C	ρ, D	Alkanes	GPR	1200	80 - 0 - 20	[74]
T, P, ρ	ρ, κ, η	CO_2	ANN	30	70 - 15 - 15	[75]
T, P	ρ, C_P, v_s, κ, η, H, S	CO_2	DNN	32994	80 - 0 - 20	[76]
T, P	ρ, Pr, C_P, Γ, v_s, κ, η	China RP-3 Kerosene	ANN	49086	90 - 0 - 10	[77]
T, P	C_P, H	H, O_2, CH_4, CO_2	ANN	not specified	not specified	[78,79]
T, P	ρ, C_P, C_V, v_s, κ, η	CO_2	ANN	not specified	not specified	[80,81]
T, ρ_{solv}, M_w, P_C, ω	D	CO_2	MLR, kNN, DT, RF, GB	4917	70 - 0 - 30	[82]
T, ρ, P, η, $M_{w,2}$, ϕ_2, $M_{w,3}$, $\phi 3$	D	Ternary mixtures containing $sc$$H_2O$	DNN	1220	80 - 0 - 20	[83]
Voronoi cell density, N_{faces}	πgas	Lennard-Jones fluids (Ar, H_2O, CO_2)	DNN	21,970,000	80 - 0 - 20	[84,85]
T, P, ϕ_{N_2}, ϕ_{CH_4}, [Na+], [K^+], [Ca^{2+}], [Mg^{2+}]	γ	CO2	ANN	1716	70 - 15 - 15	[86]
$g(r)$	State variables (P, T)	Lennard-Jones fluids	ANN	10201	80 - 10 - 10	[87]
T, ϕ_{H_2O}, ϕ_{CO_2}	State variables (P, V, T)	H_2O-CO_2-H_2	SVR	2490	80 - 0 - 20	[88]
T, P, ρ	κ	CO_2	ANN, LSSVR	586	70 - 15 - 15	[89]
T, P	κ	CO_2	ANFIS	1042	80 - 0 - 20	[90]
T, ρ	κ	CO_2	ANN	5893	80 - 0 -20	[91]
T, ϕ_{solv}, 10 params.	Vapor-Liquid Equilibrium, $\gamma\infty$	CO_2, CHF_3	ANN	1567	92 - 0 -8	[92]
T, ρ	μ	CO_2	ANFIS, LSSVR, ANN	1124	80 - 0 - 20	[93]
m_s, α	T_C, P_C	Mie fluids	ANN	500	80 - 0 - 20	[94]
m_s, α, T	P_v, ρ_{liq},ρ_{vap}.	Mie fluids	ANN/GPR	16077	80 - 0 - 20	[94]
m_s, α, T, P	ρ_{sc}	Mie fluids	ANN	27000	80 - 0 - 20	[94]

C_P, specific heat at constant pressure; D, diffusion coefficient; $g(r)$, radial distribution function; H, enthalpy; N_C, number of carbons; M_w, molecular weight; T, temperature; T_B, bubble point; T_C, critical temperature; T_D, dew point; T_r, reduced temperature (= T/T_C); P, pressure; P_C, critical pressure; Pr, Prandtl number; P_V, vapor pressure; S, entropy; v_s, velocity of sound; V, volume; η, viscosity; γ, interfacial tension; γ^∞, activity coefficient at infinite dilution; Γ, specific heat ratio; κ, thermal conductivity; π_{gas}, fraction of gas-like molecules; ρ, density; ρ_{liq}, saturated liquid density; ρ_{sc}, supercritical density; ρ_{vap}, saturated vapor density; ω, acentric factor; ϕ_x, fraction of component x;
ANFIS, adaptive neuro-fuzzy inference system; ANN, artificial neural network; DNN, deep neural network; DT, decision tree; GB, gradient-boosted; GPR, Gaussian process regression; MLR, multilinear regression; LSSVR, least squares support vector regression; kNN, k-nearest neighbors; RF, random forest

molecular dynamics simulations. The SVR model used molar volume, T, and the molar fractions of each component, ϕ_i, as inputs to predict the corresponding value of P. When trained with NIST data, the SVR model predicted *PVT* curves with extremely low error ($R^2 > 0.999$ in the testing set). For ternary systems, the trained SVR model predicted *PVT* data in the test set with an $R^2 = 0.9999$. Hence, this model could accurately predict the molecular dynamics predicted *PVT* properties of ternary fluid mixtures. This model is highly specific to the ternary system studied, but the empirical equation-of-state model produced by this is approach is potentially time-saving compared to experimental determination of *PVT* models.

Que-Salinas *et al.* used multilayer ANNs to predict bulk properties of fluids described by Lennard-Jones potentials under sub- and supercritical conditions from calculated radial dis-

tribution functions, $g(r)$ [87]. The only input was $g(r)$, consisting of 100 neurons, and predictions were made for T and P independently (with P and T fixed, respectively) and for both T and P simultaneously. After training, the P, T, and PT models typically had relative percentage errors in the predicted variables of < |3%|, with a few conditions (low T, low P) presenting errors < |7%|. This model showed that machine learning based approaches can accurately predict the macroscopic properties of fluids from a microscopic description of their molecular distribution. The authors then used their trained PT model to predict P at constant T as a function of fluid density, ρ. The calculated $P(\rho)$ curves closely agree with theoretically calculated isothermal curves, demonstrating that this approach can be used to build a complete thermodynamic description of a system from microscopic descriptions.

Zhu and Müller investigated the use of multilayer ANNs as an alternative to equation-of-state approaches to predict multiple fluid properties; the critical coordinates, T_C and P_C; the subcritical vapor-liquid equilibrium, and the supercritical density, ρ [94]. These problems having increasing complexity, required 2, 3, and 4 variables, respectively. The authors used statistical associating fluid theory (SAFT) with a Mie potential as an equation of state to predict sub- and supercritical fluid properties. SAFT models molecules as assemblies of spherical segments and consider the interactions between molecules in terms of an interaction potential. The authors used a Mie potential to model the interactions in their fluid. (1) For the prediction of T_C and P_C; an ANN with three hidden layers (15, 10, 5) was used with the number of segments, m_s, and a parameter describing the repulsive part of the Mie potential, α, as inputs. The trained ANN was able to predict the contents of the validation set with an R^2 of 0.9999. (2) For predicting the vapor-liquid equilibrium, an ANN featuring three hidden layers (48, 24, 12) was used, with m_s, α, and $1/T^*$ as inputs. The trained ANN was able to predict the vapor pressure, $\ln(P_V^*/P_C^*)$; saturated liquid density, ρ_L^*; and saturated vapor density ρ_V^* in the validation set with R^2 values of 0.9985, 0.9995 and 0.9987 respectively (Figure Figure(a)). The prediction of $\ln(P_V^*/P_C^*)$ became increasingly inaccurate close to $P_V^*/P_C^* = 1$. (3) In the case of supercritical density, an ANN with 4 hidden layers (48, 24, 12, 6) was used, with m_s, α, T, and P as inputs. The trained ANN predicted the contents of the validation set with an R^2 of 0.997 (Figure Figure(b)). In all cases, the machine learning approach was able to replicate the prediction of the SAFT model with *AARD*s below 4.7%. These models offer a route for predicting fluid

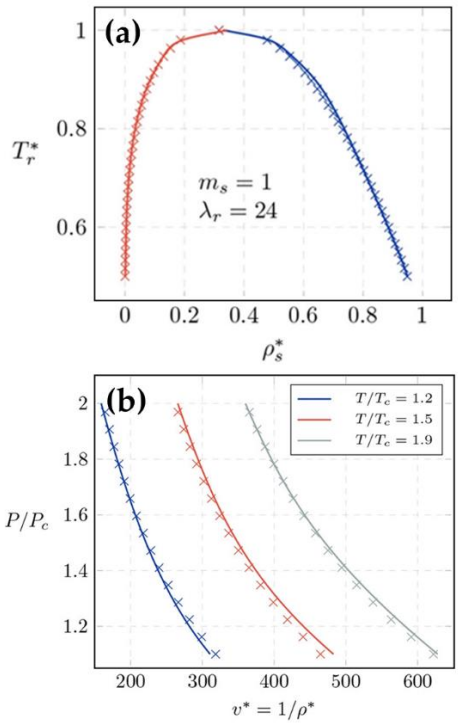

Figure 5. (a) Comparison between SAFT and ANN predicted VLE envelopes. Red indicates saturated vapor densities, while blue indicates saturated liquid densities, with a solid line representing ANN predicted values. **(b)** Plot of ANN-predicted v^* (= $1/\rho^*$) for a fluid previously unseen by the ANN (solid line) compared with SAFT predicted values (symbols) for three isotherms. Reprinted with permission from Ref. [94]. Copyright 2020 American Chemical Society.

properties outside of the training set without the high computational cost associated with SAFT equation-of-state models.

Diffusivities.

Diffusion coefficients describe the mobility of chemical species through a different material, they are important for describing the mixing and homogenization of mixtures. There is a wide range of supercritical fluid applications in which the prediction of these coefficients is important, such as describing mixed solvent systems, extractions, and chemical reactions. The use of machine learning in this task has the potential to allow the description of novel mixtures and processes without lengthy experimental characterization or simulation.

Zhao *et al.* used a DNN to predict diffusion coefficients for binary and ternary supercritical H_2O mixtures [83]. They sought to calculate self-diffusion coefficients (relating to the displacement of individual molecules) and Fick diffusion coefficients (relating to the transport of a group of molecules due to a driving force) [95]. Initially, they used molecular dynamics sim-

ulations to calculate diffusion coefficients for binary and ternary supercritical H_2O-based mixtures (N_{binary} = 140, $N_{ternary}$ = 160). For binary and ternary mixtures of H_2O with H_2, CH_4, CO, O_2, and CO_2 under supercritical conditions were considered (5 binary and 10 ternary mixtures). They trained their ANN (3 hidden layers) with the simulated results using T (200 – 400°C), ρ (calculated from P = 25 MPa), and viscosity, η, as well as the molecular weight, $M_{w,2}$, and fraction, ϕ_2 (1 – 30 mol. %), of the second species as inputs to predict the corresponding self-diffusion (D_1 and D_2) and Fick diffusion (D_{FK}) coefficients. Their trained model was able to predict the contents of the testing data set (20% of data) with an R^2 value of 0.99.

The authors then used 'transfer learning' to improve prediction accuracy for the ternary mixtures. In transfer learning, a model is first trained and optimized using large relevant dataset, such as the binary mixture dataset in this study. The optimized ANN model is then used as the initial conditions for training on a smaller dataset of interest, such as the ternary mixture dataset. The likelihood of successfully training on the second smaller dataset is increased because the optimized models are likely to have similar hyperparameters. The ternary mixture dataset covered the same temperature range, with $60 \leq \phi_{H_2O} \leq 90$ mol. %. An additional two inputs were included, $M_{w,3}$, and ϕ_3. An additional self-diffusion coefficient, D_3 was added to the outputs and D_{FK} was replaced by the 2 diagonal elements of the ternary Fick diffusion coefficient matrix. The trained binary mixture ANN was used as an initial state for the ternary mixture ANN, allowing the model to be 'fine-tuned' during the training process. As a demonstration of the superior ability of transfer learning-assisted ANNs, the authors only used 20% (N = 32) of the ternary mixture dataset for training, reserving the other 80% (N = 128) for testing. The transfer learning-assisted ANN performed significantly better than a randomly initialized ANN of the same structure for this small training set (R^2 = 0.86 vs. R^2 = 0.63). Training with larger fractions of the dataset resulted in improved R^2 values for both ANNs, but the transfer-assisted model consistently performed better than the randomly initialized ANN. The models presented by Zhao *et al.* offer a method to rapidly and accurately predict diffusion coefficients for multicomponent mixtures even for systems where relatively small amounts of data exist.

Aniceto *et al.* used five machine learning algorithms (MLR, *k*NN, DT, RF, and GB) to predict the diffusivities of a large range of polar and non-polar solute molecules in supercritical CO_2 [82]. They used the database of Vaz *et al.* ($N = 4917$) which includes 13 properties of 174 solutes in supercritical CO_2 [96]. The model was trained using T and $\rho_{solv.}$, as inputs as well as the M_w, P_C, and ω of the solute and predicted D as its output. After optimization and training with 70% of the dataset, the GB model was found to perform best. It was able to predict data in the testing set with an *AARD* value of 2.6%. The trained GB model was shown to outperform several classical diffusivity equations such as Wilke-Chang, Lai-Tan, and Dymond-Hildebrand-Batschinski (*AARD*s of 12%, 26%, and 4.3% respectively). This model is likely generalizable to previously unseen solutes, although this was not demonstrated by the authors. "This model is likely generalizable to previously unseen solutes, although this was not demonstrated by the authors. The authors have made they model available online [97].

Freitas *et al.* used GPR and a multilayer ANN to predict the densities and diffusivities of pure alkanes (*n*-octane, *n*-nonane, *n*-decane, *n*-dodecane, and *n*-hexadecane) at sub- and supercritical conditions from molecular dynamics modeled datasets ($N = 1200$) and the NIST standard reference database [74]. They used T (27 – 627°C), P (3 – 150 MPa), and the number of carbons in the alkane, N_C as inputs. The trained GPR and ANN models were able to predict the densities in the validation set with an R^2 of 0.999 in both cases. The GPR model was less accurate when trained using only a fraction (10%) of the training set, predicting the contents of the testing set with an R^2 of 0.85 compared with 0.94 for the trained ANN. Both models were consistently less accurate in the trans-critical region at low pressure (i.e. close to the critical point). When the models were applied to predict diffusivities the same drop in accuracy close to the critical point was observed, although the ANN performed better within this region.

Vapor-Liquid Equilibria.

Some work has focused on using machine learning to predict the vapor-liquid equilibria (VLE) of fluid mixtures where at least one component is under supercritical conditions. For instance, Mohanty used a single-layer ANN to predict the VLE for binary mixtures of supercritical CO_2 and three fatty acid ethyl esters (ethyl caproate, ethyl caprylate, and ethyl caprate)

using T and P as the model inputs [98]. After training their ANN was able to predict the molar fractions of CO_2 in the liquid (and vapor) phases with R^2 values of 0.998 (0.835), 0.997 (0.810), and 0.995 (0.882) for ethyl caproate, ethyl caprylate, and ethyl caprate containing mixtures, respectively.

Alvarez and Saldaña used a combination of a multilayer ANN and an equation-of-state model to predict the vapor-liquid equilibria of binary mixtures of ionic liquids, and supercritical CO_2 or supercritical CHF_3 [92]. First, they used an ANN to predict the activity coefficient at infinite dilution, γ^{∞} for a range of organic, aqueous, and ionic liquid solutions, including 283 different solutes and 65 different solvents, totaling 1567 data points, covering. a range of temperatures between 288 and 363°C. The input parameters for their ANN were temperature, the residual part of the activity coefficient (8 parameters calculated by using the conductor-like screening model-segment activity coefficient (COSMO-SAC) model), the combinatorial contribution to the activity coefficient (2 parameters calculated using the Staverman-Guggenheim model), and the solute-solvent volume ratio. The trained ANN was able to predict the infinite dilution activity coefficients with an average MSE of 0.056 on the test dataset (8% of the total dataset containing 126 binary systems). The authors then used the ANN-predicted values of γ^{∞} for binary mixtures featuring supercritical CO_2 or CHF_3 were then used to calculate Henry's constant. This was then used in turn to calculate the vapor-liquid equilibrium using the perturbed-chain statistical associating fluid theory (PC-SAFT) equation-of-state. The calculated phase behaviors of mixtures of supercritical CO_2 or CHF_3 with ionic liquids matched closely at pressures closer to P_C (< 15 MPa) although diverged increasingly above this, whilst qualitatively describing the shape of the vapor-liquid equilibrium. The authors suggest that using modified versions of the PC-SAFT model for associating substances could correct these deviations. Because of the wide range of binary mixtures in the training set, this ANN model is likely generalizable to other ionic liquids unseen by the ANN, although this was not demonstrated in the paper.

The research group of Sun has produced several studies implementing machine learning as a replacement for flash (or phase equilibrium) calculation in the prediction of vapor-liquid

equilibria for multicomponent mixtures [99–101]. Flash calculations are important in the calculation of equilibria in separation processes and multiphase mixtures and are important in compositional reservoir simulations. Equation-of-state flash calculations are extremely computationally expensive, acting as a bottleneck to these techniques. Hence, there is interest in the implementation of machine learning to improve the speed of these calculations without compromising accuracy. In their most recent study, Zhang *et al.* used a DNN to predict equilibrium constants in five-, eight-, and fourteen-component reservoir fluids for which 40401 data points had been generated by flash calculation [101]. Their DNN used T; P; overall molar concentration, C; and $T_{C,i}$, $P_{C,i}$, ω_i, and ϕ_i of each component as inputs and predicted the number of phases and molar fractions in the vapor, ϕ_{vapor} (mol. %), and liquid phases, ϕ_{liquid} (mol. %) as outputs. After training (90% of the dataset) the DNN was able to predict the contents of the training set (10%) with a relative *MSE* of ~2%. The DNN was able to correctly molar compositions of the liquid and gas components under subcritical conditions and identify the transition to supercritical conditions for the five-component oil. This model has the potential to be made generalizable to a greater range of oil mixtures.

Interfacial Tensions.

Zhang *et al.* used a two-layer ANN to predict the interfacial tension, γ, of CO_2 – brine mixtures in sub- and supercritical conditions [86]. The authors gathered 1712 data points from the literature covering both pure CO_2, and impure CO_2 containing CH_4 and N_2 (0 – 80 mol. %) and 5 solutes: NaCl, KCl, Na_2SO_4, $MgCl_2$, and $CaCl_2$. Their ANN used T, P, ϕ_{N_2} (mol. %), and ϕ_{CH_4} (mol. %), as well as [Na^+], [K^+], [Ca^{2+}], and [Mg^{2+}] as inputs. The trained ANN significantly outperformed empirical correlations and provides accurate reproduction of γ for pure CO_2–H_2O, pure CO_2–brine, and impure CO_2 systems, with only a 3.4% error in the testing set. Their ANN model appears to be generalizable to other monovalent salts containing Na^+ and K^+, and bivalent salts containing Ca^{2+} and Mg^{2+}, but this was not demonstrated.

Bubble and Dew Points.

Lashkarbolooki *et al.* used a single-layer ANN to predict the bubble and dew point pressures of binary CO_2 mixtures containing *n*-pentadecane, 2-ethyl-*1*-butene, decafluorobutane, *1*-hexene, *n*-hexane, or *n*-butane [72]. The authors gathered 316 data points from the literature

covering 199 bubble points and 117 dew points with T and P in the ranges -10 – 120°C and 0.18 – 12.06 MPa, respectively. Their ANN used properties of the non-CO_2 compound as inputs; reduced temperature, T_r (= T/T_c), P_C, ω, and ϕ_{CO_2} (in the liquid and gas phases) as inputs. After training (N = 237) and optimization, the ANN was able to predict the contents of the test set (N = 79) with an R^2 value of 0.992. The ANN outperformed an equation of state models in all instances. Because this model is trained using the co-solvent's characteristics, it is likely generalizable to other co-solvents unseen by the model, although the authors do not demonstrate this.

Activity Coefficients.

Winter *et al.* have recently developed SPT-NRTL, a natural language processing model to predict activity coefficients of binary mixtures in the liquid phase [102]. Their model uses T, ϕ_x, and the characters of the SMILES codes of each molecule as inputs to predict non-random two liquid (NRTL) model parameters which can in turn be used to predict activity coefficients, $\ln(\gamma)$, and VLEs. The authors collected data from on concentration-dependent activity coefficients from Brouwer *et al.* [103] (20870 data points covering 349 solvents and 373 solutes) and the Dortmund Data Bank [104] (77053 data points covering 506 substances). They pretrained their model using a synthetic database containing 13 million data points which substantially improved convergence time and learning rates compared to a randomly initialized model. Their trained SPT-NTRL model was able to predict binary activity coefficients with a mean absolute error between 0.1 and 0.2 (water and carboxylic acids produced errors between 0.4 and 0.5). Its generalizability was demonstrated by predicting previously unseen compounds with mean absolute error of 0.2. The trained VLE model was accurate to within ± 1°C. The VLE model was extrapolated to predict liquid-liquid equilibria (LLE), without training on LLE data, the predicted LLEs were less accurate than the VLE model, but were surprisingly accurate given the lack of directly comparable training data.

2.2 Solubilities, Miscibilities, and Supercritical Extractions.

The use of supercritical fluids as a solvent is one of their most important applications. Supercritical fluids have a range of attractive solvent properties, such as low viscosity, high

diffusivity, and low surface tensions which make them well suited for the extraction of compounds. The most commonly used supercritical solvent is supercritical CO_2 which is cheap and safe to use. The low polarity of CO_2 means that co-solvents such as methanol and ethanol must be used to dissolve polar species.

Solubilities in Supercritical Solvents.

One of the most common applications of machine learning in the domain of supercritical fluids research has been the estimation solubilities of individual compounds in supercritical fluids (and the solubility of supercritical fluids in other fluids). These studies vary greatly in complexity, with most studies using machine learning to estimate the solubility of individual compounds in a single supercritical solvent over a range of pressures and temperatures. Examples of such studies are given in Table 2 [105–131], although receive no further discussion in the main text. A more interesting approach is the use of machine learning to study the solubility of families of compounds in supercritical solvents, using the physical properties of the solute as inputs for the model. Such studies have the potential to create models with broad generalizability applicable to a wide range of solutes, although most examples in the literature do not demonstrate this [132,133]. Such models have the potential to estimate molecular solubilities for a broad range of molecules and solvents from their properties alone.

A recent demonstration of the generalizability of such approaches was published recently by Aminian and ZareNezhad, who used an ANN to estimate the phase behavior of supercritical CO_2 and fatty oils [132]. Using temperature, T; pressure, P; alongside fatty acid properties (critical temperature, T_c; and critical pressure, P_c, and their acentric factor, ω) as inputs. 678 data points were collected from the literature covering 33 different fatty acids. Their trained ANN model estimated the molar fractions in the testing set (number of data points, $N = 120$) with a coefficient of determination, R^2 of 0.995, and lower average absolute relative deviations, *AARD*s than equation-of-state models for all datasets. Importantly, they then demonstrated the generalizability of their trained model by estimating liquid-phase CO_2 fraction ϕ_{CO_2} (mol. %) for four fatty acids previously unseen by the neural network with *AARD*s between 5.4 and 21%. Suggesting that this approach would be well suited to further estimations of the solubilities of many other compounds in a wide range of supercritical fluids.

Another recent example of a demonstrably generalizable model is that of Osada *et al.*, who used machine learning to estimate the solubility of organic compounds in high-temperature H_2O (100–250°C) [133]. They took experimental conditions (temperature and solvent density) alongside 194 molecular structure descriptors as regression analysis and estimated the solubility of organic compounds in H_2O. Published solubility data from 1280 organic compounds at room temperature and ambient pressure were used alongside 54 data points for 10 organic compounds at high temperature and above saturation pressure to train the neural network. 80% of this dataset was used to train the machine learning model and 20% was used for testing. Regression analysis was performed using OLS, Lasso, and SVR, as well as a model which combined Lasso and SVR analysis (Lasso + SVR). Their Lasso + SVR method performed better than other regression techniques, producing an R^2 of 0.92 and a root mean square error, *RMSE* of 0.58 on the test data set. The generalizability of the model was demonstrated on an additional 54 data points for 7 organic compounds which had not been previously seen by the model. The model was able to estimate the experimental values in this verification dataset with an R^2 of 0.95 and an *RMSE* of 0.46, demonstrating that it could be to any organic molecule to estimate its solubility in H_2O between 100 and 250°C.

Supercritical Fluid Extractions.

Another closely related application of machine learning is the estimation of supercritical extraction yields of natural product. The extraction of products from natural sources using supercritical CO_2 can yield products free of organic solvents at low temperatures with low operational costs [134,135]. Thus, there is great interest in the optimization of these processes using novel approaches such as machine learning. Machine learning analysis of supercritical fluid extractions has been mostly limited to extraction yield prediction of single products under a limited range of conditions, with no generalizability to other products. In the papers reviewed in this section, machine learning approaches are often reported to perform better than other multivariate analysis techniques, such as response surface analysis, although a

Table 2. Supercritical solubility and miscibility studies.

Extracted Compound	Solvent (+ co-solvent)	*P* (MPa) Min	Max	*T* (°C) Min	Max	Input Parameters	Estimated Parameter	Dataset Size	Train - Valid. - Test Split (%)	Comment	REF
α-pinene	***sc*CO₂**	3	10	40	55	*P*, *T*	Solubility (ϕ_{solute})	65	59 - 16 - 20	Single solute in *sc*CO2	[105]
Lenalidomide	***sc*CO₂**	12	30	35	65	*T*, *P*, ρ_{scCO_2}	Solubility (ϕ_{solute})	28	79 - 0 - 21	Single solute in *sc*CO2	[106]
Salsalate, decitabine	***sc*CO₂**	12	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	64	84 - 0 - 16	Individual solutes in *sc*CO2	[107]
Tamoxifen	***sc*CO₂**	12	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	32	84 - 0 - 16	Single solute in *sc*CO2	[108]
Busulfan	***sc*CO₂**	12	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	32	100 - 0 - 0	Single solute in *sc*CO2	[109]
Fenoprofen	***sc*CO₂**	15	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	32	84 - 0 - 16	Single solute in *sc*CO2	[110]
Chloroquine	***sc*CO₂**	12	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	32	66 - 0 - 33	Single solute in *sc*CO2	[111]
Busulfan	***sc*CO₂**	12	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	32	62 - 0 - 38	Single solute in *sc*CO2	[112]
Salsalate	***sc*CO₂**	12	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	32	81 - 0 - 19	Single solute in *sc*CO2	[113]
Oxaprozin	***sc*CO₂**	12	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	32	69 - 0 - 31	Single solute in *sc*CO2	[114]
Oxaprozin	***sc*CO₂**	12	40	35	65	*T*, *P*	Solubility (ϕ_{solute})	32	84 - 0 - 16	Single solute in *sc*CO2	[115]
8 polymers	***sc*CO₂**	7.4	44	177	100	*T*, *P*	Solubility (ϕ_{CO_2})	327	70 - 15 - 15	*sc*CO2 in several polymers.	[116]
Anthracene	***sc*CO₂** (+ Ace, EtOH, cyhex)	10	30	35	55	*P*, *T*, co-solvent, $\phi_{cosolvent}$	Solubility (ϕ_{solute})	60	50 - 20 - 30	Single solute in *sc*CO2 + co-solvents	[117]
Disperse dyes	***sc*CO₂** (+ EtOH)	15.9	30.1	80	120	*T*, *P*, T_C, Pc, ρ_{dye}, M_w, ω	Solubility (kg·m^{-3})	48	70 - 15 - 15	Individual solutes in *sc*CO2 + EtOH	[118]
Anti-cancer drugs	***sc*CO₂**	12	40	35	65	*T*, *P*, ρ_{scCO_2}, M_w	Solubility (ϕ_{solute})	311	64 - 16 - 20	Several solutes in *sc*CO2 using properties as inputs.	[119]
46 organic compounds	***sc*CO₂** (+ H₂O)	7.5	31.5	1	11	*T*, *P*, 200 parameters	Partition coefficients	447	90 - 10 - 0	Several solutes in *sc*CO2 using properties as inputs.	[120]
Polycyclic aromatic hydrocarbons	***sc*CO₂**	8	13.3	35	60	*P*, *T*, $v^s(T,P)$, E_{HOMO}, μ, $N_=$	Solubility (ϕ_{solute})	89	79 - 21 - 0	Several solutes in *sc*CO2 using properties as inputs.	[121]
Solid aromatic compounds	***sc*CO₂**	1.5	44.8	35	70	*T*, *P*, T_C, P_C, ω	Solubility (ϕ_{solute})	198	95 - 0 - 5	Several solutes in *sc*CO2 using properties as inputs.	[122]
Organic solutes	***sc*CO₂**	8	35.5	25	65	*T*, *P*, ρ_{scCO_2}, T_C, P_C, ω	Solubility (ϕ_{solute})	439	70 - 15 - 15	Several solutes in *sc*CO2 using properties as inputs.	[123]
Solid organic compounds	***sc*CO₂**	3.6	46.75	35	250	*T*, *P*, T_C, P_C, $v^s(T,P)$, ω	Solubility (ϕ_{solute})	795	80 - 0 - 20	Several solutes in *sc*CO2 using properties as inputs.	[124]
Acids	***sc*CO₂**	7	30	28	75	*T*, *P*, pK_a, M_w, N_C, N_H	Solubility (kg·m^{-3})	180	86 - 0 - 14	Several solutes in *sc*CO2 using properties as inputs.	[125]
Bioactive compounds	***sc*CO₂**	23.9	56.3	0	90	*T*, *P*, ρ_{scCO_2}, M_w, T_m	Solubility (ϕ_{solute})	1074	100 - 0 - 0	Several solutes in *sc*CO2 using properties as inputs.	[126]
Non-steroidal anti-inflammatory drugs	***sc*CO₂**	8.9	40	30	65	*T*, *P*, ρ_{scCO_2}, M_w, T_m	Solubility (ϕ_{solute})	254	not specified	Several solutes in *sc*CO2 using properties as inputs.	[127]
23 aromatic and aliphatic organic compounds	***sc*CO₂**	7.8	200	35	95	*T*, *P*, T_C, P_C, ω	Solubility (ϕ_{solute})	970	66 - 0 - 33	Several solutes in *sc*CO2 using properties as inputs.	[128]
11 alcohols, ketones, and glycol ethers	***sc*CO₂**	0.01	57	10	100	*T*, *P*, T_C, P_C, ω	Solubility (ϕ_{CO_2})	810	80 - 0 - 20	*sc*CO2 solubility in several solvents using properties as inputs.	[129]
14 ionic liquids	***sc*CO₂**	0.01	100	25	100	*T*, *P*, T_C, P_C, M_w, ω	Solubility (ϕ_{CO_2})	728	70 - 0 - 30	*sc*CO2 solubility in several solvents using properties as inputs	[130]
20 ionic liquids	***sc*CO₂**	0.1	100	177	100	*T*, *P*, T_C, P_C, M_w	Solubility (ϕ_{CO_2})	1386	70 - 15 - 15	*sc*CO2 solubility in several solvents using properties as inputs	[131]
Fatty oils	***sc*CO₂**	0.7	30	33	211	*P*, *T*, T_C, P_C, ω	ϕ_{CO_2} in gas and liquid phases	673	70 - 15 - 15	Generalized to 3 oils not in the training set. **Demonstrated as generalizable**	[132]
organic compounds	**H₂O**	--	--	100	250	*T*, ρ_{H_2O}, 194 params.	Solubility (mol·kg^{-1})	1290	64 - 16 - 20	Generalized to 7 organic compounds not in the training set. **Demonstrated as generalizable**	[133]

P, pressure; *T*, temperature; ϕ_x, fraction of *x*; $v^s(T, P)$, molar volume; ω, acentric factor; M_w, molecular weight; T_m, melting point, T_C, critical temperature; P_C, critical pressure; pK_a, equilibrium constant; V_m, molecular volume; μ, dipolar moment; N_C, carbon number; N_H, hydrogen number; $N_=$, number of double bonds; E_{HOMO}, energy of highest occupied molecular orbital;Ace, acetone; cyhex, cyclohexane; EtOH, ethanol; *sc*CO₂, supercritical CO₂, *sc*H₂O, supercritical H₂O

Table 3.
Machine learning predicted extractions from natural sources.

Source	Extracted Compound	Solvent (+co-solvent)	*P* (MPa)		*T* (°C)		Input Parameters	Estimated Parameter	Dataset Size	Train-Valid.-Test (%)	Comment	REF
			Min	Max	Min	Max						
Raspberry *Rubusidaeus L.*	Seed oil	***sc*CO₂**	25	35	40	60	P, T, d_P, Q_{CO_2}	Yield	19	60 - 20 - 20	Single extract in $scCO_2$	[136]
Black pepper *Piper nigrum*	Essential oils	***sc*CO₂**	15	30	30	50	P, T, t_d, d_P, Q_{CO_2}	Yield	24	50 - 25 - 25	Single extract in $scCO_2$	[137]
Aniseed *Pimpinella anisum*	Essential oils	***sc*CO₂**	8	18	30	30	P, t_d, Q_{CO_2}	Yield	369	60 - 20 - 20	Single extract in $scCO_2$	[138]
Passion fruit *Passiflora edulis*	Seed oil	***sc*CO₂**	17	33	47	63	P, T, t_d	Yield	16	70 - 0 - 30	Single extract in $scCO_2$	[139]
Diplotaenia *Diplotaenia cachrydifolia*	Essential oils	***sc*CO₂** (+MeOH)	10.1	30.4	35	75	P, T, t_d, V_{MeOH}	Yield	42	62 - 14 - 24	Single extract in $scCO_2$ + MeOH	[140]
Green tea *Camellia sinensis*	Epigallocatechin gallate	***sc*CO₂** (+EtOH)	10	30	40	60	P, T, t_d, Q_{CO_2}	Yield	31	67 - 0 - 33	Single extract in $scCO_2$ + EtOH	[141]
Green tea *Camellia sinensis*	Epigallocatechin gallate	***sc*CO₂** (+EtOH)	10	30	40	55	P, T, t_d	Yield	62	60 - 8 - 32	Single extract in $scCO_2$ + EtOH	[142]
Spearmint *Mentha spicata*	Spearmint oil	***sc*CO₂**	8.5	12	38	50	P, T, t_d, d_P, Q_{CO_2}	Yield	405	75 - 0 - 25	Single extract in $scCO_2$	[143]
Persian rose *Rosa damascene*	Quercetin	***sc*CO₂** (+EtOH)	10	30	35	55	P, T, t_d, Q_{CO_2}	Yield	31	34 - 33 - 33	Single extract in $scCO_2$	[144]
Chavir *Ferulago angulata*	Essential oils	***sc*CO₂**	12	20	35	55	P, T, t_d, d_P,	Yield	31	Not specified	Single extract in $scCO_2$	[145]
Kuntze *Launaea acanthodes*	Essential oils	***sc*CO₂** (+EtOH)	12	24	35	55	P, T, V_{EtOH}, Q_{CO_2}	Yield	30	70 - 15 - 15	Single extract in $scCO_2$	[146]
Bertoni *Stevia rebaudiana*	Stevioside, Rebaudioside-A, Other phenols	***sc*CO₂** (+EtOH)	7.5	37.5	25	65	P, T, ϕ_{EtOH}	Yields	16	70 - 15 - 15	Multiple extracts in $scCO_2$ + EtOH	[147]
Sage *Salvia officinalis*	Essential oils	***sc*CO₂**	10	30	40	60	P, T, t_d, Q_{CO_2}	Apparent solubility	17	85 - 0 - 15	Single extract in $scCO_2$	[148]
Annatto *Bixa Orellana*	Seed oil	***sc*CO₂** (+EtOH)	5.7	19.3	25	60	P, T, ϕ_{CO_2}, ϕ_{EtOH}, Equilib. Type	Yield	40	60 - 20 - 20	Single extract in $scCO_2$ + EtOH	[149]
Mexican prickly poppy *Argemone Mexicana*	Seed oil	***sc*CO₂** (+EtOH)	20	35	60	100	P, T, d_P, Q_{CO_2} ϕ_{E-tOH}	Phase equilibria	46	70 - 15 - 15	$scCO_2$ – EtOH – seed oil phase equilibria	[150]
Tomato paste waste *Solanum lycopersicum*	Lycopene, β-carotene	***sc*CO₂** (+EtOH)	10	30	20	100	P, T, t_d, Q_{CO_2}, ϕ_{E-tOH}	Yields	81	90 - 0 - 10	Two extracts in $scCO_2$ + EtOH	[151]
Ulan Ulan *Cuscuta reflexa*	Coumarin	***sc*CO₂** (+MeOH)	35	75	15.2	34.5	P, T, t_d	Yield	20	80 - 0 - 20	Single extract in $scCO_2$	[152]
Myrobalan *Terminalia chebula*	Various phytochemicals	***sc*CO₂** (+EtOH-H_2O)	10	17.5	40	60	P, T, t_d, $Q_{H_2O\text{-}EtOH}$	Yields	30	70 - 15 - 15	Multiple extracts in $scCO_2$ + H_2O + EtOH	[153]
Pomegranate *Punicagranatum L.*	Pomegranate oil	***sc*CO₂**	20	40	40	60	P, T	Yield	37	70 - 15 -15	Single extract in $scCO_2$	[154]
Licorice *Glycyrrhiza glabra*	Glycyrrhizic acid	***sc*CO₂** (+H_2O)	10	34	45	85	P, T, Q_{CO_2}, t_d	Yield	65	81 - 11 - 8	Single extract in $scCO_2$	[155]
Galega *Galega officinalis L.*	Galegine	***sc*CO₂**	35	55	10	30	P, T, Q_{CO_2}, t_d	Yield	31	71 - 16 -13	Single extract in $scCO_2$	[156]
Coriander *Coriandrum sativum L.*	Seed oils	***sc*CO₂**	10	20	40	70	P, T, Q_{CO_2}	Yield	21	70 - 15 - 15	Single extract in $scCO_2$	[157]
Aniseed *Pimpinella anisum*	Seed oils	***sc*CO₂**	8	18	30	30	P, Q_{CO_2}, t_d	Yield	369	75 - 0 -25	Single extract in $scCO_2$	[158]
Gynura *Gynura procumbens*	Various phytochemicals	***sc*CO₂** (+EtOH-H_2O)	18	24	60	70	P, T, ϕ_{H_2O}	Yield	20	70 - 15 -15	Total extraction yield in $scCO_2$ + H_2O + EtOH	[159]
Sugarcane bagasse *Saccharum officinarum*	Lignin, Glycose, Xylose	***sc*CO₂** (+EtOH)	7.5	30	35	100	P, T, ϕ_{EtOH}	Yields	12	70 - 0 - 30	Three extracts in $scCO_2$	[160]

P, pressure; T, temperature; t_d, dynamic extraction time; Q, flow rate; d_P, particle diameter (after milling); ϕ_x, fraction of x; V_x, volume of x; EtOH, ethanol; MeOH, methanol; $scCO_2$, supercritical CO_2

small number of the studies conclude the opposite. For instance, recently Pavlić *et al*. predicted optimized conditions for the extraction of raspberry seed oil with higher precision than the values calculated from response surface optimization [136]. A summary of similar recent supercritical extraction studies utilizing machine learning is given in Table 3, detailing the extraction sources, extracts, solvents, and ranges of conditions explored [136–160].

2.3 Chemical Reaction Optimization.

Supercritical fluids have found application in a wide range of chemical syntheses. Two particularly important applications are the production of biodiesel by supercritical transesterification and the synthesis of fuel gases by supercritical water gasification. Machine learning has found application in the improvement of yield in both of these applications in recent years.

Biodiesel Production by Supercritical Transesterification.

Biodiesel consists of mixtures of fatty acid alkyl esters which are prepared from biologically sourced oils [161]. Supercritical biodiesel production is typically performed through the transesterification of triglycerides or the esterification of fatty acids with an alcohol. In the case of transesterification, the use of supercritical alcohols has become an area of active research because it offers several advantages over other approaches which often require high-energy input, expensive short-lived catalysts, long reaction times, and pretreatment of the precursors. By comparison, supercritical transesterification of triglyceride-containing oils in alcohols offers a rapid, catalyst-free, more energy-efficient synthesis that is compatible with a wide range of precursors [161]. First proposed by Kusdiana and Saga in 2001 [162], in their protocol, methanol and triglyceride-containing oils were mixed and brought to supercritical conditions. As the methanol becomes supercritical, there is a rapid drop in its dielectric constant resulting in the two phases becoming miscible forming a single phase. Methanol molecules then attack the carbons located in the carbonyl groups of the triglycerides, resulting in decomposition into a free fatty acid methyl ester ('biodiesel') and a diglyceride (Figure 6) [163]. This process can then be repeated for the resulting diglyceride, and then the resulting monoglyceride, until only a glycerol molecule and three free fatty acid methyl esters remain. For complete conversion of the precursor oil to free fatty acid methyl esters, either a large molar excess of methanol or long residence times (close to the stoichiometric ratio) are required. The use of other solvents, co-solvents, and catalysts have been proposed as routes to further improve these approaches [161]. Machine learning has been deployed as a route toward optimizing this process further.

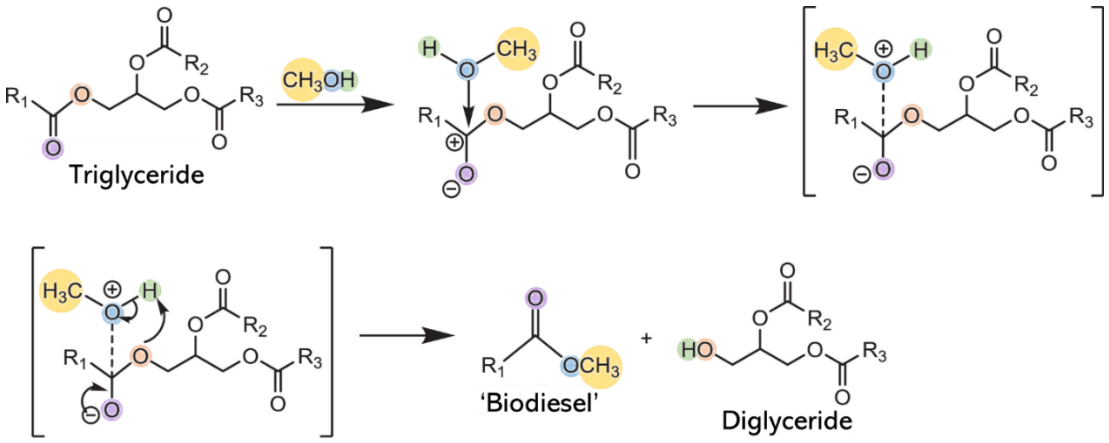

Figure 6. Reaction mechanism between triglycerides and methanol in supercritical conditions. R_1, R_2, and R_3 are alkyl chains. The methanol attacks the carbon atom of a carbonyl group leading to the transfer of a methoxide and the fracturing of the triglyceride into a free fatty acid methyl ester ('biodiesel') and a diglyceride. The diglyceride can then be reacted with another methanol molecule to form a monoglyceride, and again reacted with another methanol molecule to finally form a glycerol molecule and three free fatty acid methyl esters. Reprinted (and adapted) with permission from Ref. [161]. Copyright 2015 Elsevier.

The first study to use a single layer ANN to predict biodiesel yield *via* supercritical transesterification was by Farobie *et al* [164]. They studied the transesterification of canola oil into biodiesel in supercritical methanol/ethanol using a spiral flow reactor. The ANN inputs were T (270 – 400°C), P (10 – 20 MPa), ϕ_{oil} (2.4 – 4.8 vol%), and reaction time, t_r (3 to 30 min). The yield of free fatty acid methyl esters (in methanol) and free fatty acid ethyl esters (in ethanol) measured using gas chromatography-mass spectrometry were used as outputs. Their dataset (N = 42) was divided into training (N = 36), testing (N = 3), and validation (N = 3) sets. Their ANN predicted the contents of the test set with an R^2 of 0.998 in both ethanol and methanol. These conditions were observed to produce biodiesel from canola oil with ~100% yield.

The dataset generated by Farobie *et al.* has been revisited by the group of Baghban [165,166], who have used alternative machine learning models, such as the ANFIS and LSSVR approaches which predicted the yields in their test datasets with R^2 values of 0.998 [165] and 0.996 [165] respectively. Farobie and Hansanah also demonstrated the use of the same ANN in predicting yields in the transesterification of canola oil in supercritical tert-butyl methyl ether [167]. Using T (200 – 500°C), P (6 – 15 MPa), t_r (3 – 15 min), and ϕ_{oil} (2 – 4.8 vol%) as inputs, their ANN predicted biodiesel yields in in the testing set with an R^2 of 0.97.

Sarve *et al.* used a single layer ANN to predict yields of biodiesel from mahua (*Madhuca indica*) oil using supercritical ethanol with CO_2 as a co-solvent [168]. The presence of CO_2 decreases T_C and P_C of ethanol [169]. They used T (250 – 350°C), P (10 – 50 MPa), t_r (10 – 50 min), and ϕ_{oil} (3 – 6 mol. %) as input variables and the free fatty ethyl ester yield as determined by gas chromatography-mass spectrometry as the output. The ANN was able to predict yields in the unseen testing data with an R^2 of 0.87. By comparison, response surface analysis produced an R^2 of 0.66 in the same dataset. Optimum conditions of T = 304°C, P = 4 MPa, t_r = 36 min, and ϕ_{oil} = 3.4% were predicted to produce a free fatty ethyl ester yield of 95.08%. This was experimentally found to be 97.42% close to the output value and above the minimum free fatty ethyl content for use as biodiesel of 96.5% (EN 14214 [170]).

Srivastava *et al*. used a single layer ANN coupled with a genetic algorithm optimization to predict conditions for the transesterification of microalgae oil to fatty acid methyl ester (biodiesel) using supercritical methanol [171]. They cultivated microalgae at a large-scale (100 L) and extracted the non-polar/lipid components. For the transesterification, a methanol – microalgae oil (ϕ_{oil} = 2.2 – 6.3 vol%) mixture was brought to supercritical conditions (T = 240 – 300°C) with a reaction time of (t_r = 15 – 60 min). The fatty acid yield was then calculated using nuclear magnetic resonance spectroscopy and gas chromatography-mass spectrometry. The ANN was able to predict data in the testing set with an R^2 of 0.97. A globally best-optimized condition generated by a mixture of an ANN and genetic algorithm for supercritical methanol transesterification predicted that T = 285.2°C, t_r = 26.5 min, ϕ_{oil} = 4 vol. % would yield a conversion efficiency of 99%, this was found experimentally to be 98%. Furthermore, the synthesized biodiesel was found to have similar properties to international standards on biodiesel, making it suitable for commercial use (ASTM D6751 [172] and EN 14214 [170]).

Selvan *et al*. used a single-layer ANN coupled with genetic algorithm optimization to predict the biodiesel yield from the supercritical methanol transesterification of *Aegle marmelos* oil [173]. They used T (230 – 350°C), ϕ_{oil} (2 – 4 wt. %), and t_r (5 – 25 min) as inputs. They were able to predict the contents of the testing set with an R^2 value of 0.9998. Optimization of the ANN using a genetic algorithm predicted optimal synthesis conditions of T = 325.47°C, ϕ_{oil} = (2.4 wt. %), and t_r = 22.35 min. The ANN model output a yield of 1.01 for these conditions, which experimentally

produced a yield of 0.984 (with an absolute difference between the output and experimental result of 0.027) demonstrating the accuracy of the ANN model. The synthesized biodiesel was found to comply with the ASTM D6751 specification [172].

These studies have all shown potential in the prediction of biodiesel yields and as platforms for the optimization of biodiesel production. However, published results have only focused on using machine learning to maximize yield, this is somewhat redundant given that several studies already report 100% yields. Other forms of optimization should be prioritized, such as making the process greener and lowering costs, through reductions in energy consumption by using lower temperatures and reaction times.

Supercritical Enzymatic Interesterification.

A related process to supercritical transesterification is that of supercritical interesterification, where the fatty acids in a mixture of triglycerides are rearranged by breaking and reforming the ester bonds connecting the fatty acids to the glycerol moiety. This has application in domains such as food science, where there is active interest in controlling the fatty acid content of food ingredients, such as cocoa butter.

Shekarchizadeh *et al*. used a multilayer ANN coupled with a genetic algorithm to predict and optimize the production of cocoa butter analog *via* enzymatic interesterification of camel hump fat and tristearin in supercritical CO_2 [174]. Cocoa butter increasingly faces supply limitations and growing prices, leading to interest in the development of alternative products. The fatty acids in camel hump fat are predominantly palmitic, stearic, and oleic acids. By comparison, 70% of cocoa butter is comprised of three triacylglycerols (*1,3*-palmitin-*2*-olein (POP), *1*-palmitin-*2*-olein-*3*-stearin (POS), and *1,3*-stearin-*2*-olein (SOS)). Camel hump fat is comparably deficient in stearin containing fatty acids, hence to produce a product similar to cocoa butter, interesterification much be performed with a triglyceride featuring stearin in excess. Hence, they interesterified the camel hump fat alongside tristearin (SSS) using *thermomyces lanuginosus 1,3*-specific lipase (added at 10% weight of substrate). Their ANN used T (30 – 70°C), P (3 – 31 MPa), ϕ_{SSS} (11 – 66 wt. %), ϕ_{H_2O} (0 – 20 wt. %), and t_r (1 – 7 h) as inputs to predict the yield of POP, POS, and SOS after interesterification as determined by high-performance liquid chromatography. After training on all data available (N = 32), the

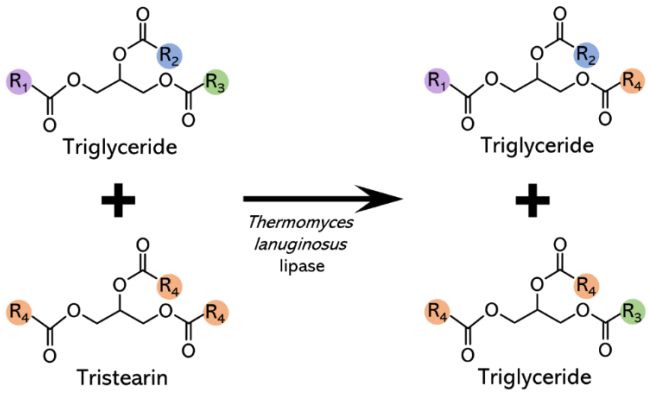

Figure 7. Reaction mechanism between triglycerides and tristearin during supercritical enzymatic interesterification. Either of the fatty acids, R_1 and R_3, are liberated by the *thermomyces lanuginosus* lipase as well as a steric acid from tristearin and exchanged leading to the formation of different triglycerides.

trained model was able to predict the POP, POS, and SOS yields of a previously untested set of experimental conditions with errors of 0.5, 0.8, and 1.3% respectively. A genetic algorithm was then used to optimize the synthesis to produce a triglyceride composition close to that of cocoa butter. The genetic algorithm predicted that conditions of T = 40°C, P = 10 MPa, $\phi_{\text{tristearin}}$ = 37.5 wt. %, ϕ_{H_2O} = 13 wt%, and t_r = 4.5 h would produce POP, POS, and SOS fractions of 24.55%, 43.65%, and 31.80% respectively (c.f. ϕ_{POP} = 23.82 wt. %, ϕ_{POS} = 44.81 wt. %, and ϕ_{SOS} = 31.37 wt. % for natural cocoa butter). Although these conditions were not tested by the authors.

Supercritical Water Gasification.

Supercritical H_2O gasification is a thermochemical process that uses supercritical H_2O as a reaction medium and reactant in the processing of organic waste to produce fuel gases [175,176]. It offers a route to process waste from a large number of sectors (municipal, agricultural, industrial, and forestry waste, as well as wastewater) into high-quality fuel gases at a lower temperature than traditional gasification with low tar production [177–181]. The gasification of organic waste by supercritical H_2O occurs through a complex set of concurrent reactions which include (but are not limited to) [182]:

Hydrolysis

$$C_nH_mO_y + (n - y)\cdot H_2O \rightarrow n\cdot CO + (n - y + \tfrac{1}{2}\cdot m)\cdot H_2$$
$$C_nH_mO_y + (2n - y)\cdot H_2O \rightarrow n\cdot CO_2 + (2n - y + \tfrac{1}{2}\cdot m)\cdot H_2$$

Methanation

$$CO + 3\cdot H_2 \leftrightarrow CH_4 + H_2O$$
$$CO_2 + 4\cdot H_2 \leftrightarrow CH_4 + 2\cdot H_2O$$

Water-gas shift reaction

$$CO + H_2O \leftrightarrow CO_2 + H_2$$

Water is an active participant in several of these reactions, as well as being the solvent, meaning that H_2 is generated from the solvent alongside the organic components. This process can be performed with and without catalysts. The development of supercritical H_2O gasification is likely to prove extremely important as a green technology for the production of fuel gases and waste management.

Several researchers have now shown that machine learning approaches can be used to accurately predict gas yields of data collected from the literature on supercritical H_2O gasification of solid waste validated by unseen data [183–187], simulation [188], or further experimentation [189]. These typically use reactor conditions (T, P, t_r, $\phi_{solvent}$) and feedstock composition (i.e., elemental composition, as well as water, ash, and volatile content) as inputs to calculate the yields of various gases of interest (typically H_2, CO, CO_2, and occasionally CH_4) [190,191]. These studies have shown that a wide variety of machine learning approaches can be effectively trained on datasets concerning supercritical H_2O; these approaches include ANN, GB, GPR, SVR, DT, RF, and AB supervised-learning models [183–187,189]. These models can accurately fit the pre-existing datasets, but typically are not used subsequently to predict optimized conditions in the specific system studied, or shown to have any generalizability beyond the training set.

Other authors have attempted to demonstrate that this is possible. For instance, recently Li *et al*. have produced multiple studies implementing machine learning in the prediction and optimization of supercritical H_2O gasification of organic wastes [192–194]. They have approached this through two machine learning approaches; multilayer ANNs [193] and GB [192,194].

In the case of the GB approach, Li *et al*. studied the sub- and supercritical H_2O gasification of organic waste, the authors collected data from the literature, collecting 295 data points from 29 peer-reviewed papers [192,194]. This data was processed using the GB approach to predict H_2, CH_4, CO_2, and CO yields ($mol \cdot kg^{-1}$) during gasification. Organic waste composition (ϕ_C (22.1 – 65.5 wt. %), ϕ_H (2.1 – 7.3 wt. %), ϕ_N (0 – 7.3 wt. %), ϕ_O (0.2 – 54.8 wt. %), ϕ_{ash} (0 – 57.8 wt. %), and ϕ_{solid} (0.76 – 30 wt. %)), T (200 – 850°C), P (10.4 – 32 MPa), and t_r (2 – 120 min) were used as input conditions. Their trained model was able to predict the yields of H_2, CH_4, CO_2, and CO in the testing set with R^2 values of 0.93, 0.92, 0.95, and 0.95 respectively, and *RMSE* values of 1.9, 0.51, 1.3, and 0.29 respectively,

demonstrating the accuracy of their model. The authors then used a particle swarm optimization (PSO) algorithm to optimize the trained GB model and produce optimized organic waste compositions and gasification conditions. The authors validated the output optimum conditions using Aspen Plus simulation software. The difference between the PSO-optimized GB outputs and the Aspen Plus simulated results were 6 and 5 % for CO_2 and H_2, respectively. The authors then used a combination of their trained GB model and Aspen Plus to infer a greater understanding of the role of each reaction parameter in the system and the chemical processes which occur during supercritical H_2O gasification of solid waste.

In the implementation of their multilayer ANN approach, Li *et al*. developed a generalized ANN to predict and optimize non-catalyzed, alkali-catalyzed, transition metal-catalyzed supercritical H_2O gasification of solid waste [193]. They initially created three datasets; firstly, the 295 data points mentioned above, secondly, an additional 117 data points covering alkali-catalyzed supercritical H_2O gasification, and finally, 75 data points covering transition metal-catalyzed supercritical H_2O gasification, each of these groupings was treated separately. The inclusion of catalysts added several inputs to the ANNs including the catalyst ratio and catalyst descriptors (for alkali salts (equilibrium constant, pK_a; solubility; M_w; and alkali metal atomic mass, radius, valency, ionization energy, electron affinity, and conductivity), and transition metals (catalyst size, alongside surface area, pore size, and total pore volume (as determined by Brunauer–Emmett–Teller analysis)). The datasets for each condition (non-catalyzed, alkali-catalyzed, and transition metal-catalyzed) were used to train three separate ANNs, which were able to predict data in the test set with R^2 values of 0.96, 0.85, and 0.93, respectively. They then combined all the datasets ($N = 527$) and trained a single generalized ANN model, which was able to predict the contents of the test set with an R^2 value of 0.86. Using the trained generalized ANN, the authors found that Fe and Fe-compounds were predicted to outperform more commonly used and more expensive catalysts such as Ni and Ru at catalyzing supercritical H_2O gasification of solid waste whilst maximizing H_2 yield and minimizing CO_2 production. They were also able to predict optimized supercritical H_2O gasification conditions for these catalysts. Li *et al*. have made their ANN trained with the combined dataset available to the community in a user-friendly format [195].

Fózer *et al*. have similarly applied a multilayer ANN to a combined dataset of both non-catalyzed and NaOH-catalyzed supercritical H_2O gasification of microalgae, which can then be used as feedstock for biomethanol production [196]. They then used their ANN model to predict optimized conditions and then performed a life-cycle analysis to assess the sustainability of these processes. The authors gathered 55 data points from 6 publications covering 6 different feedstocks. They used T (380 – 700°C), P (1 – 30 MPa), t_r (2 – 60 min), ϕ_{NaOH} (0 – 5 wt. %), $\phi_{biomass}$ (1 – 30 wt. %), and biomass composition (ϕ_C (34 – 50 wt. %), ϕ_H (5 – 6.2 wt. %), ϕ_N (0.1 – 7.3 wt%), ϕ_O (19 – 53 wt. %), and ϕ_S (0 – 3.7 wt. %)) as input variables and the various gas yields (ϕ_{H_2}, ϕ_{CO}, ϕ_{CO_2}, ϕ_{CH_4}, $\phi_{C_2H_4}$, $\phi_{C_2H_6}$ ($mol \cdot kg^{-1}$)) as outputs. After training, the ANN was able to predict data in the test set with an R^2 of 0.99. The ANN model indicated that both non-catalytic and catalytic supercritical H_2O gasification were suitable methods for producing syngas for methanol production. They then selected two highly performing conditions for the non-catalytic and NaOH-catalyzed supercritical H_2O gasification were produced with H_2, CO_2, and CO in suitable yields for methanol production (and minimal hydrocarbon presence), and simulated them using Aspen Plus. Life cycle analysis of the ANN-guided simulations indicated that the process could be used for carbon fixing and have negative greenhouse gas emissions.

2.4 Materials Synthesis Optimization.

Solid Lipid Microparticles.

López-Iglesias *et al.* have recently used an ANN to optimize the size and polydispersity of solid lipid microparticles synthesized by the 'particles from gas-saturated solutions' (PGSS) technique [197]. Solid lipid microparticles are of interest for biomedical, food, and antimicrobial applications [198–200]. The PGSS technique offers a solvent-free approach to produce solid lipid microparticles, by using supercritical CO_2 to melt and solubilize phospholipids, before atomization into particles. supercritical CO_2 depresses the melting point of the phospholipids in this process, lowering the melting point of glyceryl monostearate from 61°C to 52°C by varying the pressure between ambient and 13 MPa. This is advantageous because it reduces energy usage during the process and ensures the rapid solidification of the particles after atomization.

An ANN was used to predict the optimum conditions for producing solid lipid microparticles *via* the PGSS technique, using atomizer nozzle diameter, temperature, and pressure as inputs and mean particle diameter, and fine particle fraction (%) as outputs. For each set of conditions explored, mean particle size and fine particle fraction were characterized. Variable modeling was carried out using an ANN and fuzzy logic integrated into neuro-fuzzy software, the results of which indicated that temperature was the main parameter affecting particle diameter and that nozzle diameter were the main parameter affecting the solid lipid microparticle size distribution. Particle diameters were estimated with an R^2 of 0.92, whereas the standard deviation of the diameters and fine particle fraction had relatively poor R^2 values of 0.58 and 0.75 respectively, suggesting that other variables in the system beyond temperature, pressure, and nozzle diameter should be used as model inputs. The authors did not demonstrate the accuracy of their model outside of the training set or use the trained ANN to predict optimized conditions.

2.5 Supercritical Power Systems.

Supercritical CO_2 has in recent years drawn increasing attention as a next-generation heat transfer fluid for use in power generation [201,202]. When used in a heat engine, supercritical CO_2 offers a high Brayton cycle efficiency, because of its high density and specific heat capacity (relative to gaseous CO_2) and small compressibility factor [203]. CO_2 is inexpensive, safe, and offers increased efficiency in power systems making it a greener working fluid than many alternatives. The implementation of the supercritical CO_2 Brayton cycle power systems has largely been limited by the high pressures and temperatures required by the system. Supercritical CO_2 Brayton cycle power systems have the potential to improve the heat exchange efficiency in a wide range of power systems, including coal-fired, gas-turbine, concentrated solar, and nuclear power plants [202]. Their implementation may prove to be most important in emerging green technologies such as concentrated solar plants, where it could significantly improve their economic viability [204].

A range of potential supercritical CO_2 Brayton cycles have been proposed, such as the supercritical recuperated, supercritical reheated, and supercritical recompression power cycles (Figure 7) [205]. The most basic case of a supercritical CO_2 Brayton cycle is the simple closed-loop recuperated cycle, consisting of a heat exchanger, turbine, recuperator, compressor, and cooler. In which CO_2 is heated, exerts work on a turbine, and waste heat is recovered by the recuperator, followed by the CO_2 being cooled, recompressed, and heated by the recovered lost heat. Reheating cycles add a

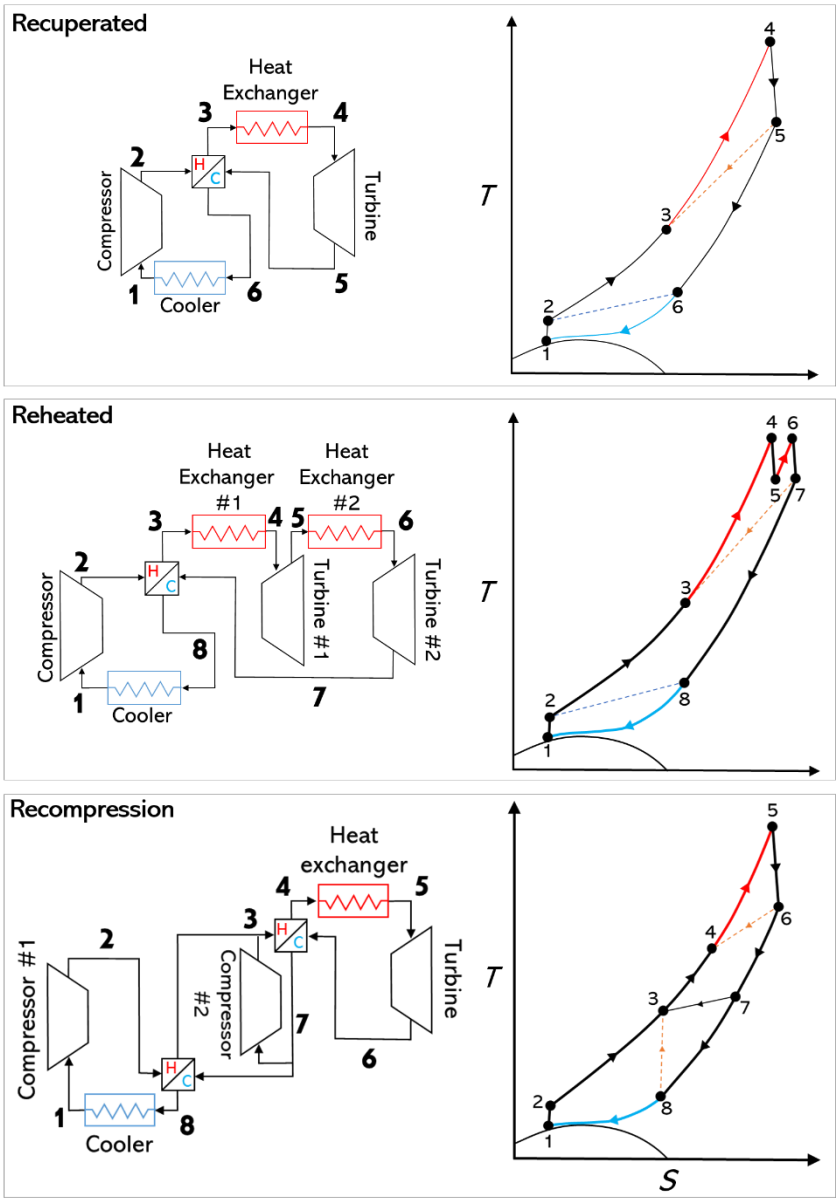

Figure 8. Supercritical power cycles. Recuperated, reheated, and recompression cycles. Blue lines indicate heat rejection processes, and red lines indicate heat injection processes. Adapted with permission from Ref. [205]. Copyright 2021 Elsevier.

heating step after the turbine before work is performed on a second turbine. Recompression cycles have an additional compressor that recompresses a fraction of the main flow before the cooler, and reinjects it back into the cycle before heat rejection in the cooler, improving the efficiency of the cycle. Machine learning has been recently been applied to the optimization of individual cycle components and to accelerate the search for optimized cycle running conditions.

Table 4. ANN estimated vertical tubular heat exchanger heat transfer characteristics.

Input Parameters	Estimated Parameter(s)	Fluid	Machine Learning Technique(s)	Dataset Size	Train - Valid. - Test (%)	REF
P, $\dot{m}$, Q, d, H	h	CO_2	ANN	5895	86 - 0 - 14	[73]
T_{in}, P, Q, $\dot{m}$, d, H	T_w	CO_2	ANN	403	100 - 0 - 0	[206]
T_{in}, P, Q, $\dot{m}$, d, H	T_w	CO_2	ANN	7313	80 - 0 - 20	[207]
P, Q, $\dot{m}$, d, H	T_w (cool side)	H_2O	ANN	5280	73 - 24 - 3	[208]
T_{in}, P, Q, $\dot{m}$, d, H, Γ, $\bar{C}_P/C_{P,fl}$, Pr, Re, $\overline{Pr}$ $\rho_w:\rho_{fl}$, $\mu_w:\mu_{fl}$, and $\kappa_w:\kappa_{fl}$ ratios	Nu, δT_{HTD}	CO_2	DNN	11539	64 - 16 - 20	[209]
Γ, Re, Pr, $\rho_w:\rho_{fl}$, $\mu_w:\mu_{fl}$, and $\kappa_w:\kappa_{fl}$ ratios	Nu	CO_2	ANN	2646	not specified	[210]
Q, $\dot{m}$, Re, Bo*, x^+	h	CO_2	ANN	46	80 - 0 - 20	[211]

Nu, Nusselt number; Re, Reynolds number; Pr, Prandtl number; Γ, specific heat ratio; ρ_i, density; μ_i, viscosity; κ_i, thermal conductivity; T, temperature; P, pressure, d, tube diameter, Q, heat flux; $\dot{m}$, mass flux; H, enthalpy; C_P, specific heat at constant pressure; v_{gas}, flue gas velocity; $\bar{C}_P$, C_P integrated over the range between the heatsink wall and T_{fl}; $\overline{Pr}$, Pr calculated using $\bar{C}_P$; Bo*, Buoyancy number; x^+, dimensionless axial co-ordinate; δT_{HTD}, T overshoot due to heat transfer deterioration. *Subscripts:* in, inlet; fl, fluid; w, wall

Heat exchangers.

Several of the machine learning implementations in this area have looked at vertical tubular heat exchangers. These are typically relatively simple systems that estimate either the wall temperature, T_w, the heat transfer coefficient, h, or the Nusselt number, Nu (= hL/κ) [73,206–211]. These studies are typically not generalizable and are only demonstrated to accurately estimate these properties for single fluids and tube geometries limiting their potential for optimization. They are normally demonstrated to be more accurate in estimating experimental data than published empirical models. These studies have been summarized in Table 4.

Saeed *et al.* have produced several recent studies looking at the use of multilayer ANNs to optimize the design of supercritical CO_2 printed circuit board heat exchangers for use in supercritical CO_2 power systems [212,213]. They investigated two heat-exchanger designs, a zigzag channel, and a C-channel. They parameterized the geometry and used the parameters as ANN inputs (Figure 8). This allowed the optimization of their heat-exchanger designs through coupling with an optimization algorithm.

In the case of the zigzag channel study, Saeed *et al.* used a 3D Reynolds averaged Navier–Stokes model to compute the thermal and hydraulic properties of a zig-zag printed circuit heat exchanger containing supercritical CO_2 close to the critical point in the cool side and H_2O in the hot side (See Figure 8 for geometry) [212]. The pre-cooler geometry was constant throughout the simulations, and properties were calculated for a wide range of CO_2 mass flow rates (Reynolds numbers, Re, between

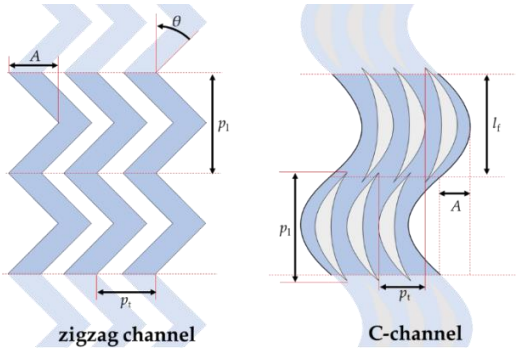

Figure 9. Heat sink channel geometries and geometric parameters investigated by Saeed *et al.* in Refs. [212] and [213].

2500 and 70000) and Prandtl numbers, Pr, between 0.7 and 13 (covering nearly all values of P and T used in the supercritical CO_2 Brayton cycle). Their ANN used Pr, Re, ρ, μ, and thermal conductivity, κ, as inputs, and Nusselt number, Nu, and fanning friction factor, f, were outputs. 99% of the data in the validation set could be predicted with 90% confidence. When the results of the trained ANN were used in conjunction with a precooler design and analysis code, the computational requirements and time were reduced significantly. Requiring only a single processor on a local machine for 2 – 10 min, compared with 10 – 26 h on a high-performance computation facility for the precooler design and analysis coupled with a computational fluid dynamics model. The ANN-assisted code was then used to improve the design of the precooler by finding precooler operating conditions that avoided the creation of 'pinch points' which adversely affect heat exchange.

More recently, Saeed *et al.* applied the same approach to the optimization of a C-shaped (sinusoidal fins) printed circuit heat exchanger [213]. These heat exchangers consist of a sinusoidal channel featuring discontinuous staggered C-shaped fins which repeat longitudinally, with a pitch, p_l, and are staggered transversely with a pitch, p_t, (See Figure 8 for geometry). In this study, p_l; p_t; and the channel periodicity, l_f; and amplitude, A, were used as inputs. The outputs were Nu; and f from which the performance of the heat exchanger could be calculated. The authors used a 3D Reynolds averaged Navier–Stokes model to calculate these parameters for 81 different geometries. These were then used to train the ANN. The trained ANN predicted the contents of the testing set (N = 13) with an R^2 of 0.996 and 0.994 for Nu and f respectively. The trained ANN was then coupled to a genetic algorithm to allow optimization of the C-shaped channel geometry, producing a heat exchanger design with a higher performance than any of the initially simulated designs within a few minutes. The machine learning optimized C-shaped channel heat exchanger outperformed other common

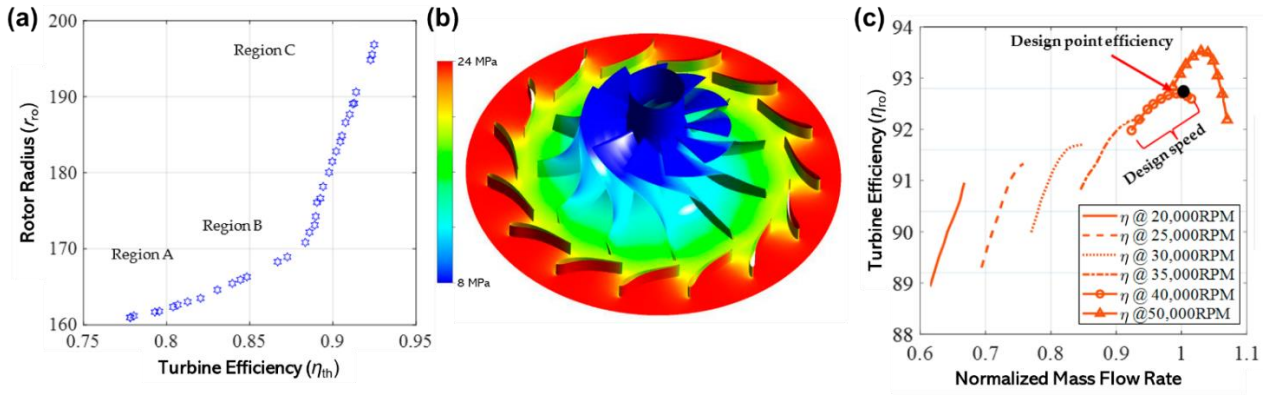

Figure 10. ANN modelled supercritical CO_2 turbine efficiencies. **(a)** Optimized efficiencies for a range of rotor radii calculated from a multi-objective genetic algorithm optimization of an ANN model. **(b)** Multiphysics simulation of pressure distribution of the on the surface of the optimized hub/rotor geometry. **(c)** Simulated off-design performance of the optimized turbine geometry. Adapted with permission from Ref. [214]. Copyright 2021 MDPI.

printed circuit heat-exchangers in simulations. This approach towards the optimization of individual components with power systems offers a route to rapidly converge on highly efficient designs.

Turbines.

Saeed *et al*. have also applied this approach to the design of turbines for use in supercritical CO_2 power systems [214]. The authors generated 600 data points using a 3D Reynolds averaged Navier–Stokes model over a range of turbine geometries. They parameterized the design of their turbine in terms of the shroud-to-rotor radius ratio, r_{sh}/r_{ro}; the hub-to-rotor radius ratio, r_{hub}/r_{ro}; and the inlet flow angle, α. These were all used as inputs to their multilayer ANN alongside the blade-to-fluid velocity ratio, σ, to predict the thermal efficiency η_{th} and rotor radius, r_{ro}. They were able to predict the contents of the testing and validation datasets with R^2 values of 0.997 in both cases. They then performed a multiparameter optimization using the trained ANN and a multi-objective genetic algorithm, to maximize η_{th} while minimizing r_{ro}. Through this approach, they identified three groups of design parameters (min r_{ro}, max η_{th}, and compromised efficiency) that satisfied their multi-objective optimization (Figure 9a). The authors conclude that higher η_{th} should be prioritized because the size of the turbine is small compared to other components in the power cycle. Simulations of the geometric parameters optimized for η_{th} performed using the multiphysics package ANSYS showed that the design could maintain efficiencies over 90% over a wide range of turbine rotational velocities (Figure 9b&c). This ANN is not generalizable to expanders with different power outputs, but the use of ANNs in this fashion offers an accelerated route to find optimized turbine designs without performing large numbers of computationally-expensive Multiphysics simulations.

Predictions of full-power cycles.

Several authors have sought to use machine learning to study the efficiency of full supercritical CO_2 Brayton cycles. There are a few examples demonstrating accurate prediction of experimental [215] and theoretically modeled efficiencies [216] by machine learning models, with no attempts at optimization. However, the more interesting application of this approach is to use the trained machine learning model to design optimized power systems.

For instance, Diao *et al.* used both a DNN and a convolutional NN to predict the performance of supercritical Brayton power cycles under different operating conditions, focusing on the examples of the recompression and intercooling cycles [217]. They calculated theoretical performance values for 600,000 randomly selected operating parameters through the use of a thermodynamic model. Their DNN used 16 inputs including; the main compressor and turbine inlet temperatures, T_i, and pressures P_i; reheating and intercooling pressures, P_i; pinch point temperature differences in high and low T recuperator, ΔT_i; ambient temperature, T_{atm}, power output, W; and the compressor, turbine, motor, and generator efficiencies (η_{com}, η_{tur}, η_M, and η_G respectively). The outputs were the thermal efficiency, η_{th}, the exergy efficiency, η_{ex}, and levelized energy cost, LEC. They found that the convolutional NN outperformed the DNN below 20,000 data points in the training set. The prediction accuracy of the convolutional NN was 99.6%. They then coupled the convolutional NN to a genetic algorithm and performed a multiparameter optimization for η_{th}, η_{ex}, and LEC. The optimized model predicted operations close to the Pareto frontier (optimal condition of the power cycle), at a much lower computational cost than searching through large-scale calculations (100,000× faster).

Mishamandani *et al*. used RF and SVR to predict the performance of recuperating, precompression, and reheating supercritical power cycles [218]. The authors calculated thermal efficiencies for each configuration using thermodynamics generating 80,000 data points for each condition. For all models they used the temperature of the air, T_a (-4 – 40°C); and the first turbine, $T_{tur,1}$ (450 – 550°C); the mass flux of the air, $\dot{m}_a$ (45 – 55.8 kg·s^{-1}); fuel, $\dot{m}_F$ (4.9 – 9.9 kg·s^{-1}); and CO_2, $\dot{m}_{CO_2}$ (55 – 70 kg·s^{-1}); and the efficiency of the compressor, η_{com} (0.8 – 0.89); the isentropic turbine, $\eta_{tur,I}$ (0.85 – 0.9); the recuperator, η_{rc} (0.86 – 0.96); and waste heat recovery unit, η_{WU} (0.9 – 1) as inputs. In the case of the reheating model, the temperature of the second turbine, $T_{tur,2}$ (350 – 450°C), was also used as an input. Their model predicted the total thermal efficiency, η_{th}, as its output. After training, the SVR and

RF models were able to predict the contents of the testing set to within 10% and 3% respectively. They then coupled both models and the thermodynamic model to a particle swarm optimization algorithm, which resulted in similarly optimized solutions demonstrating the accuracy of the machine learning models in predicting the thermodynamic model. The optimized machine learning model was between 10 and 120× faster than the thermodynamic model.

The high number of possible configurations, components, and variables in these systems makes machine learning potentially a very powerful tool for optimizing their power output and efficiency. As machine-learning models and more complete data becomes available for these systems, optimization of the entire system will be possible, allowing these technologies to mature faster and become competitive with other power systems more quickly than would be otherwise possible.

2.6 Computational Fluid Dynamics Solvers with Machine Learning.

Hydrodynamic phenomena in supercritical fluids are studied numerically with computational fluid dynamics (CFD) solvers as it provides easy access to local information (velocity, pressure, concentration…) to complement experiments. For this purpose, it is necessary to precisely estimate the thermophysical properties of supercritical fluids. It is in this role that machine learning is primarily employed. The computational burden of calculating fluid properties for CFD flow solvers can be reduced through the implementation of machine learning models for this task.

An ANN was used to predict a range of properties for a single fluid [77,81,219] or for complex mixtures [220–222]. Supercritical fluids are known for the high non-linearity of the properties, particularly close to the pseudo-boiling point. The study of hydrodynamical phenomena in this region requires very high precision for property estimation. Usually, property library such as the NIST database [223] offered very good properties estimation but are very costly. For that, Longmire and Banuti [81,219] proposed to use an ANN based on NIST properties, to combine fast calculation and high precision. The ANN are trained using the PyTorch library [224] on NIST reference data [223]. The only input for the model was T, and the outputs were ρ, κ, η, C_P, and C_V. The authors demonstrate that their ANN significantly reduced computation time and offers parameter estimates of excellent precision comparable to those proposed by the NIST library. This method has revealed a

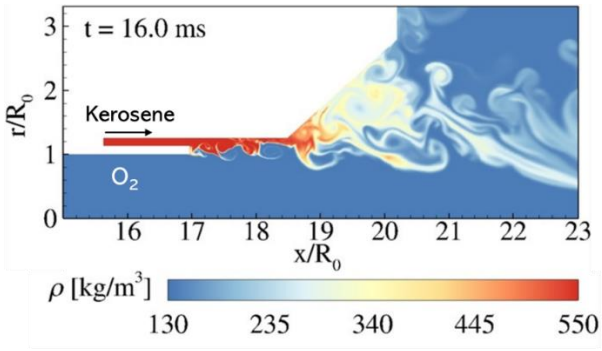

Figure 11. Density field of kerosene injection into a swirl coaxial injector under supercritical conditions as calculated using ANN-estimated thermodynamic variables. The simulation compared favorably to brute force calculations. Reprinted with permission from Ref. [221]. Copyright 2021 Elsevier.

physical mechanism as fine as the deterioration of heat transfer near the pseudo-boiling line, reminiscent of the phenomenon of boiling crisis below the critical point.

For CFD simulations involving multi component systems, Milan *et al.* developed a DNN for the evaluation of thermophysical properties of the supercritical fluid flows [220,221]. They sought to reduce the computational burden of solving generalized equations of state for such problems, which typically require large amounts of computational power and time, or by using tabulation which requires prohibitively large amounts of memory. The amount of memory used by the trained DNN model was around 4 to 5 orders of magnitude smaller than that required when implementing a tabulation approach. Analysis of the computational cost showed that the trained DNN could calculate fluid properties between 2.4 and 3.7× faster for the swirl coaxial injector and counterflow diffusion flame examples respectively, compared to the baseline computational fluid dynamics case. Hence, the implementation of machine learning to this problem has been demonstrated to significantly reduce the computation time and resource requirements for these simulations. The predicted fluid properties made by the trained DNNs were then coupled to a custom computational fluid dynamics flow solver [225]. Dependent on whether the solver required primitive or conservative variables, the inputs used were P and T, or, ρ and mass-specific energy, e, alongside the component mass fractions, ϕ_i. The outputs were also dictated by the flow solver but included a wide range of thermodynamic variables, such as P; T; ρ; e; specific heat at constant pressure, C_P; η…etc. (up to 27 in the most complex example shown). The authors used the calculated flows for two model systems. Firstly, a gas-centered liquid-swirl coaxial injector featuring the supercritical turbulent mixing of

oxygen and kerosene. Secondly, a laminar counterflow diffusion flame, consisting of a hydrogen/nitrogen mixture featuring 7 species. In both cases, the results obtained by the CFD—DNN approach are close to simulations that used properties calculated *via* computational fluid dynamics brute force approaches.

Machine learning has also been used for predicting spatiotemporal fields inside a supercritical water-fluidized bed reactor for coal gasification by Xie *et al* [226]. They used a recurrent neural network to predict the complex multiphase flow fields inside the reactor. The predicted flow fields correlated strongly with those calculated by CFD simulations. The trained recurrent neural network was hundreds of times faster than CFD simulation. Despite the training data in this study being simple 2D simulations, this approach is very promising and will be arguably more powerful with training data obtained by full 3D simulations.

Another example concerns the estimation of closure relations for statistical turbulence model. There is a scarcity of literature concerning turbulent flow with heat transfer under supercritical conditions. Building on the work of Ling *et al.* [227,228], Cao *et al.* trained a DNN with direct numerical simulation data, including all necessary physics, to improve the Reynolds-averaged Navier–Stokes turbulent model [229]. At each time iteration, the DNN model was used to estimate unclosed turbulence quantities in function of the variable fields estimated by the CFD solver. This work is a promising as it demonstrates that machine learning models can satisfactorily complement physics-based turbulent models especially in supercritical fluids.

2.7 Molecular Simulation of Supercritical Fluids.

Techniques such as density functional theory and molecular dynamics are highly reliant on the energy functionals and potential energy surfaces assumed in these models, respectively [230,231]. Machine learning has been implemented as a route to decrease simulation times and increase accuracy by using it to model these interactions. The implementation of machine learning to these modeling techniques has great potential to estimate the properties of supercritical fluids with higher accuracy and with lower computation time.

Cats *et al.* used a non-linear regression model to predict the free-energy functionals of Lennard-Jones fluids at supercritical temperatures ($k_BT = 2\varepsilon$, where k_B is the Boltzmann constant and ε is the depth of the Lennard-Jones potential well) from which density profiles, $g(r)$, and bulk equations of state could be derived [232]. Lennard-Jones fluids are a suitable model for water under supercritical conditions due to reduced hydrogen bonding [233]. They used the van der Waals mean-field approximation containing additional quadratic and cubic correction terms as their model. The exact forms of the correction terms were determined by the optimization of a loss function. They trained their model using Monte Carlo simulated density profiles under a variety of chemical and external potentials in 2D. The predicted free-energy functionals were more accurate than the standard mean-field approximation at predicting the density profiles and bulk pressure as calculated by grand-canonical Monte Carlo simulations. $g(r)$ was calculated by their machine learning model with similar accuracy to the mean-field approximation. The model trained using planar geometries could be accurately used with the 3D fluid.

Li – Cl ion pairs in supercritical H_2O.

Zhang *et al.* used a DNN to model the potential of Li – Cl ion pairs in high T aqueous solutions [234]. They used the DP-GEN software package [235] to generate *ab initio* molecular dynamics training data for LiCl-H_2O ratios between 1:55 and 20:100 and T between 30°C and 1130°C. $g(r)$ distributions calculated from the trained DNN potential for Li – Cl, Li – O, Cl – H, and O – O closely matched *ab initio* molecular dynamics simulations and were observed to change with temperature and density. Several metastable states were identified in the simulations such as H_4O_2, $H_5O_3^+$, and $H_7O_3^+$. Association constants calculated from the ANN potential were comparable with those observed experimentally, suggesting that this approach could be used where no experimental data exists.

Supercritical Hydrogen.

Recently, Cheng *et al.* predicted a previously unobserved supercritical state for H_2 at extremely high pressures (T_C, P_C = 143°C, 350 GPa) [236]. They used machine learning to predict the potential energy surfaces and interatomic forces between hydrogen atoms. The potential was output by an ANN trained using electronic structure methods. This prediction has proved controversial [237,238].

Karasiev *et al.* found that the machine learning potential predicted supercritical state was inconsistent with the subcritical behavior calculated in their own large (2,048 atoms), and long (10 ps) density functional theory molecular dynamics simulations [232]. Cheng *et al.* have disputed this criticism [238]. The pressures and temperatures for this state are experimentally achievable using approaches such as diamond anvil cells, hence this controversy could be resolved experimentally [239].

2.8 Autoignition in Supercritical Water Oxidation Processes.

Sharma *et al.* recently using several machine learning approaches to estimate the autoignition characteristics of ethanol in the oxidative environment of supercritical water within a microreactor [240]. The modelled microreactors consisted of an inlet supplying oxidizer (H_2O_2 + H_2O) and an inlet supplying fuel (H_2O and ethanol). The authors used homogeneous ignition calculations to generate 20,000 data points (15,000 in the training set and 5,000 in the validation set). This was a two-step problem, initially a classification problem (did autoignition occur?) and then a regression problem (estimation of ignition time). For classification, the authors used logistic regression, DT, RF, SVM, *k*NN, AB. Whereas for regression, they used KRR, DT, RF, *k*NN, AB, and GB. Inputs for the machine learning models were pressure (22.5 - 25 MPa), temperature (350 - 450 °C), fuel fraction (0.5 - 4.5 wt. %) and oxygen fraction (0.5 - 9.5 wt. %). In the case of the classifiers, the output was whether autoignition will occur expressed by a Boolean parameter [True/False]. In the case of the regressors the output was ignition time, for the conditions where autoignition occurred. All investigated classifiers were able to predict autoignitions with an accuracy > 95 %, with the RF and SVM classifiers being the most accurate in determining if autoignition occurred (99.3 % and 98.8 % accuracy respectively). For the regression part of the problem, using only the conditions where autoignition occurred, ignition time was predicted with an $R^2 > 0.78$ in all cases, with RF and GB models performing the best ($R^2 = 0.996$ and $R^2 = 0.993$).

2.9 Inhomogeneities in Supercritical Fluids.

Some implementations of machine learning have allowed us to build theoretical models of the nature of supercritical fluids. Ha *et al.* used a DNN to directly classify individual molecules within supercritical fluids as belonging to a liquid-like or gas-like state, based on local structural changes

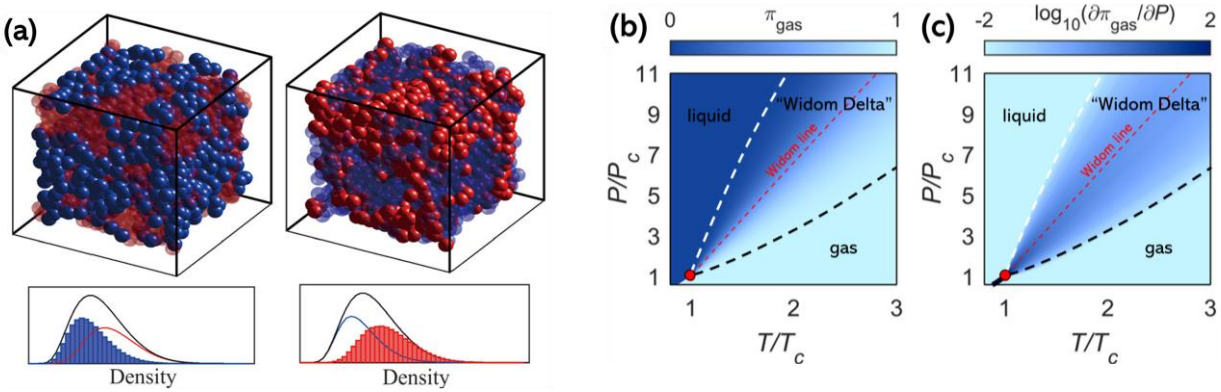

Figure 12. (a) Molecular dynamics simulation of a supercritical Lennard-Jones fluid ($T = 1.05T_c$ and $P = 1.24P_c$) showing a mixture of ANN-classified gas-like (blue spheres, left) and liquid-like (red spheres, right) particles. Histograms of the local density distribution of gas-like (blue) and liquid-like (red) particles. **(b)** Supercritical phase diagram of π_{gas}. Below the critical point (red dot), π_{gas} jumps discontinuously across the vapor-liquid equilibrium line. Beyond the critical point, π_{gas} varies continuously across the "Widom delta" (between the white and black dashed lines). The red dashed line indicates, $\pi_{gas} = \pi_{liq} = 0.5$. **(c)** Pressure derivative of π_{gas} (log-scale), in which the Widom delta is more visible. The derivative diverges at the $\pi_{gas} = \pi_{liq} = 0.5$ contour and decays in the deeply supercritical regime. (a) reprinted with permission from Ref. [84]. Copyright 2018 American Chemical Society. (b&c) reprinted with permission from Ref. [85]. Copyright 2020 American Chemical Society.

seen in their molecular dynamics simulations [84,85]. They used machine learning to directly classify liquid-like and gas-like molecules in the simulated data, based on their local density and number of nearest neighbors (as defined by Voronoi segmentation) with 100% accuracy ($N = 21{,}970{,}000$, Figure 12a). From these studies, the authors identified a region of pressures and temperatures extending from the critical point on either side of the Widom line [241], forming the shape of a deltoid in which the numbers of gas-like and liquid-like particles were seen to continually transition suggesting that supercritical fluids should be viewed as a heterogeneous mixture of liquid-like and gas-like microstructures (Figure 12b&c) [84]. These results are consistent with a number of experimental studies over the last three decades looking at the inhomogeneous nature of supercritical fluids [242–249].

Furthermore, from their classification, they extracted the fraction of gas-like molecules, π_{gas}, and demonstrated its utility as an order parameter that can be used to derive material-independent scaling laws for the supercritical state of simple fluids. They were able to calculate macroscopic properties of the supercritical fluid such as T_c and P_c, from the results of the classifier despite it only being trained with information on the local structure of individual particles. They used the machine learning derived values of π_{gas} to demonstrate that isothermal curves of bulk density for Ar, CO_2, and H_2O were found to collapse to a single master curve when rescaled using parameters derived from

π_{gas} [85]. These results have important implications for our understanding of the microscopic nature of supercritical fluids, suggesting that they are a heterogeneous mixture of microstates which dictate their macroscopic properties.

Recent work has also looked at the inhomogeneous nature of supercritical fluids using unsupervised machine learning. Banuti recently published the hybrid ergodic lattice gas model which can estimate the macroscopic fluid pressure along the critical isotherm [250]. This model used classical equation of states to calculate the kinetic and repulsive pressure components, whereas the attractive component was calculated by analysis of clustering in an Ising-like 3D lattice gas approach. Clustering within this system was evaluated using DBSCAN, an unsupervised machine learning algorithm for discovering clusters in large spatial databases with noise [251]. When the equation of state and machine learning pressure components were combined, the model outperformed the Peng-Robinson equation of state for liquids at high density. In addition, Banuti identified three distinct stages of molecular behavior along the isotherm, cluster formation, cluster consolidation, and supercluster growth. The supercritical state belongs to the cluster consolidation regime, where clusters merge together to form a single supercluster. He also demonstrated that the results of his model could be used as the basis for an alternative derivation of the attractive term of a van der Waals-type equation of state.

2.9 Supercritical CO_2 Geosequestration and CH_4 extraction.

Meng *et al.* have used a single-layer ANN to study the absorption behavior of supercritical CO_2 onto three types of coal for applications in the geosequestration of CO_2 and CH_4 extraction from deep coal beds [252]. Their ANN estimated the excess CO_2 absorption onto three types of coal using *T*, *P*, and the physicochemical parameters of coal as inputs and was trained using literature data. The model was limited by the availability of data but was able to accurately estimate the absorption behavior of supercritical CO_2 onto the three types of coal at a single temperature of 53°C.

In the context of geological CO_2 storage, multiphase flow simulations are currently used to estimate the reservoir performance. Wu *et al*. proposed a DNN model to estimate the CO_2 saturation

and the fluid pressure distribution in the reservoir [253]. They trained their DNN using data generated from a physics-based numerical simulation (N = 460). The DNN used capillary pressure and the relative permeability as inputs. The results of the physics-based simulations and DNN outputs correlated strongly. The trained DNN was able to generate simulated data using 250× less CPU time than the numerical simulations.

3. Perspectives and Conclusions

In this section, we outline our opinions on the current state of the implementation of machine learning within the domains discussed, and what future work we would like to see as the field develops further. The implementation of machine learning in the domain of supercritical fluids research is still in its infancy and there is clearly great potential for it to advance the field. As shown in many of the examples discussed above, machine learning models and their potential for use in optimization will allow the acceleration of supercritical fluid technologies toward commercial viability in the coming years. Machine learning is a tool, which in conjunction with other techniques available to us, can help us more rapidly develop poorly understood systems. Much of the true potential of machine learning is yet to be realized, the application of machine learning techniques to high dimensionality problems in supercritical fluid research, could revolutionize the use of supercritical fluids across many different applications.

Problems in the Existing Literature.

Several common problems can be observed in the existing literature implementing machine learning to supercritical fluid systems. Machine learning is a good approach in contexts where the outputs are dependent on a large number of variables with complex interdependencies between them. However, many of the systems in the literature currently being studied by machine learning are not like this. They are often too simple, and the use of computationally-expensive tools such as ANNs are inappropriate. If varying a single input results in a change in the output parameter that is smooth, monotonic, and can be described by an elementary mathematical function, with no interdependency between the inputs; more basic less computationally-expensive curve fitting techniques probably exist.

Often the ability of machine learning models to estimate data outside of the training set was not demonstrated. Improper segregation of the training and testing sets is a common problem. Proper segregation is critical to demonstrating the predictive capacities of machine learning models. Models need to be demonstrated to be able to estimate conditions that it has not previously seen. A model being able to reproduce data with which it was trained is not noteworthy and does not demonstrate that problems such as overfitting have been avoided. Further to this, if a model has been used to predict an optimized set of inputs, these need to be tested to demonstrate the accuracy of the model's outputs. Too many of these problems seem to originate in an improper understanding of the implementation of either experimental approaches or the proper implementation of machine learning. Going forward this could be addressed through collaborative approaches combining the expertise of experimentalists and machine learning experts to support each other.

It is also often near impossible to reproduce the implementation of machine learning reported in the literature from the dataset and methods reported. Only a few of the studies mentioned made a copy of their model available after publication, or provided the coefficients required to replicate their results (e.g. Refs. [78,82,97,195]). Ideally authors would make both the dataset and the script used publicly available, alongside their manuscript to allow other researchers to probably interpret their results. The creation of a repository for machine learning models and their associated data could a be a valuable asset to enable this.

Another issue that is largely overlooked is the handling of experimental uncertainties. In the studies which provided their datasets discussed here, the vast majority only reported the measured value without any reference to the associated error in the measurement. One cause of this is that experimental uncertainties are often not reported in the source materials. But authors rarely comment upon how they dealt with uncertainties, with the most likely assumption being that they were neglected entirely. Methods exist for dealing with experimental uncertainties such as Bayesian neural networks which allow for uncertainties to be propagated through the network and provide a probabilistic estimation of the output. Such approaches should be strongly considered when training using data with associated uncertainties.

In more simple problems, generalizability should be a primary goal. For instance, in the case of solubility studies, it is more interesting and useful to be able to estimate the solubilities of many different molecules in many different solvents, rather than only being able to estimate the solubility of a single molecule in a single solvent. Such generalized models could be of great benefit in accelerating computational simulations or as part of experimental workflows. By comparison, in more complex systems, where many interdependent variables exist, the use of machine learning as a model for multiparameter optimization offers a route to accelerate the discovery of improved input conditions. These models will likely be unavoidably highly specialized, but their benefit is derived from the accelerated discovery of improved outputs.

Barriers to Progression and Opportunities.

There are several barriers to the implementation of machine learning being advanced in this domain, the biggest among them is data availability and quality. The datasets required to train machine learning models are often not available, and while the recent trend toward open data standards will address this, it will take time to accrue large enough datasets for many applications. The quality of this data is also a concern. In some of the fields discussed these datasets are small and heavily biased towards a narrow range of conditions. These datasets typically omit negative results entirely, partially as a result of the heavy bias against negative results seen in academic publishing. This is problematic as it prevents machine learning models from accurately representing the global parameter space, making the training of reliable training of machine learning models more difficult.

Much of the published data suffer from incompatibility issues. In some areas, there are issues with variability between reported datasets for the same conditions. Methodological differences in the collection of characterization data can lead to significant variation between datasets due to differences in equipment, noise, operating protocols, and sensitivity. Hence comparing and/or combining different datasets can be extremely difficult. Strategies such as normalization can partially compensate for these issues, but require a great deal of user input, and have limited success.

These issues coupled with the need for large, reliable, and compatible datasets have made high-throughput data generation an appealing option. In the case of materials science, for instance, cou-

pling automation, *in situ* characterization, and high-throughput synthesis would allow the autonomous rapid generation of large volumes of data. A large number of applications of supercritical fluids research are well-suited to this approach because they can be performed under flow and using *in situ* characterization allowing for real-time optimization of processes and the high-throughput generation of datasets for offline optimization. But these are opportunities that are yet to be realized.

The recent advent of new machine learning techniques such as interpretative language models has opened up exciting new possibilities that are yet to be explored. We hope to see these models being more widely implemented by the community to potentially improve upon the results presented within this review.

Several areas of active supercritical fluids research are conspicuously absent in this review, which could benefit from the application of machine learning. Primary among them is the synthesis of materials. Many of the materials currently synthesized under supercritical conditions have properties that would be well suited to analysis by machine learning algorithms. For example, quantum dots, which have been synthesized under supercritical conditions [254,255], and possess narrow well-defined fluorescence peaks from which meaningful information can be inferred on properties such as the quality, size distribution, and colloidal stability from *in situ* measurements. Additionally, the faster reaction kinetics of chemical reactions seen under supercritical conditions allow for rapid in situ characterization and faster feedback loops in online optimization than could be achieved in many subcritical processes. Approaches utilizing machine learning, robotic synthesis, and *in situ* characterization have already been implemented under subcritical conditions [256]. Such approaches could be applied to a wide range of other materials that have been synthesized under supercritical conditions; including semiconductor [254,255], metal oxide [257–264], metallic [257,265–267], organic [268], and insulator nanocrystals [269–272].

Some of the preliminary work toward the implementation of machine learning in such roles has already begun to appear in the literature. Such as that of Kløve *et al.*, who recently demonstrated the ability of machine learning approaches to identify crystal structures of unknown compounds from *in situ* X-ray characterization data for a range of inorganic materials in capillaries obtained with short

temporal resolution at elevated *T* and *P*, including the identification of metastable phases and stacking faults [273]. The coupling of machine learning and *in situ* characterization in this manner has great potential to accelerate the discovery of new materials and improve pre-existing syntheses. While no examples of online machine learning for materials synthesis in supercritical fluids are available in the literature, it is apparent that industry is actively developing this technology [274,275]. When combined with approaches such as microfluidics, which use very small volumes of materials, online machine learning approaches potentially offer a route to reduce the physical costs of developing syntheses. We expect such approaches to become more prominent in the literature in the coming years, as autonomous robotic labs become a more mature technology.

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.

Data Availability

Not applicable.

Acknowledgments

L. Roach, A. Erriguible, and C. Aymonier would like to thank the *Agence Nationale de la Recherche* (Grant No. ANR-21-CE50-0021) for financial support during the writing of this manuscript.

Abbreviations

AB, AdaBoost; **ANFIS**, adaptive neuro-fuzzy inference system; **ANN**, artificial neural network; **Ace**, acetone; **CFD**, computational fluid dynamics; **COSMO-SAC**, conductor-like screening model-segment activity coefficient; **cyhex**, cyclohexane; **DT**, decision tree; **EtOH**, ethanol; **GB**, gradient-boosted; **GPR**, Gaussian process regression; ***k*NN**, *k*-nearest neighbors; **Lasso**, least absolute shrinkage and selection operator; **LEC**, levelized energy cost; **LLE**, liquid-liquid equilibrium; **LSSVR**, least squares support vector regression; **NIST**, National Institute of Standards and Technology; **NRTL**, non-random two liquid; **MeOH**, methanol; **MLR**, multi-linear regression; **OLS**, ordinary least squares; **PC-SAFT**, perturbed-chain statistical associating fluid theory; **POP**, *1,3*-palmitin-*2*-olein; **POS**, *1*-palmitin-*2*-olein-*3*-stearin; **PSO**, particle swarm optimization; **PGSS**, particles from gas-saturated solutions; **RF**, random forest; **SAFT**, statistical associating fluid theory; ***sc*CO$_2$**, supercritical carbon dioxide; ***sc*H$_2$O**, supercritical water; **SOS**, *1,3*-stearin-*2*-olein; **SVR**, support vector regression; **VLE**, vapor-liquid equilibrium;

Notation

[*x*], Concentration of *x*; **Δ*x***, Change in *x*; $\bar{x}$, *x* integrated over the range between the heatsink wall and T_{fluid}.

Nomenclature

A, amplitude; ***AARD***, average absolute relative deviation; **Bo***, buoyancy number; ***C***, molar concentration; C_P, specific heat at constant pressure; d_p, particle diameter (after milling); ***D***, dipolar moment; ***D***, diffusion coefficient; ***e***, mass-specific energy; E_{HOMO}, energy of highest occupied molecular orbital; ***f***, fanning friction factor; ***g*(*r*)**, radial distribution function; ***h***, heat transfer coefficient; k_B, Boltzmann constant; l_f, periodicity; ***L***, characteristic length; $\dot{m}$, mass flow rate; m_s, number of segments; M_w, molecular weight; ***MSE***, mean square error; ***n***, particle number; ***N***, number of data points; $N_=$, number of double bonds; N_C, number of carbons; N_H, number of hydrogens; **Nu**, Nusselt number (= hL/κ); pK_a, equilibrium constant; p_l, longitudinal pitch; p_t, transverse pitch; ***P***, pressure; P_C, critical pressure; P_V, vapor pressure; ***PT***, pressure-temperature; ***PVT***, pressure-volume-temperature; **Pr**, Prandtl number; ***Q***, flow rate; R^2, coefficient of determination; **Re**, Reynolds number; ***RMSE***, root mean square error; ***S***, entropy; t_d, dynamic extraction time; t_r, residence/reaction time; ***T***, temperature; T_B, bubble point; T_C, critical temperature; T_D, dew point; T_m, melting point; T_r, reduced temperature (= T/T_C); ***V***, volume; v_{gas}, gas velocity; v_s, velocity of sound; ***W***, power output; x^+, dimensionless axial co-ordinate;

α, parameter describing repulsive part of Mie potential; δT, temperature overshoot due to heat transfer deterioration; ε, depth of energy well; ***H***, enthalpy; γ, interfacial tension; γ^∞, activity coefficient at infinite dilution; Γ, ratio of specific heats; η, viscosity; η_{xx}, efficiency of *xx*; κ, thermal conductivity; μ, dipolar moment; ρ, density; $\rho_{liq.}$ saturated liquid density; ρ_{sc}, supercritical density; $\rho_{vap.}$, saturated vapor density; π_{gas}, fraction of gas-like molecules; $v^s(T, P)$ molar volume; ϕ_x, *fraction of x;* ω, acentric factor; ω_r, rotational velocity;

Subscripts

a, air; **atm**, atmospheric; **com**, compressor; **ex**, exergy; **in**, inlet; **fl**, fluid; **F**; fuel; **G**, generator; **M**, motor; **rc**, recuperator; **ro**, rotor; **sh**, shroud; **th**, thermal; **tur**, turbine; **w**, wall; **WU**, waste heat recovery unit;

References

[1] T. Hey, S. Tansley, K. Tolle (eds)., The Fourth Paradigm: Data-Intensive Scientific Discovery, Microsoft Research, Redmond, WA, 2009.

[2] J. Key, eScience - A Transformed Scientific Method, *NRC-CSTB meeting*, Mountain View, CA (2007). jimgray.azurewebsites.net/JimGrayTalks.htm.

[3] P. Carracedo-Reboredo, J. Liñares-Blanco, N. Rodríguez-Fernández, F. Cedrón, F.J. Novoa, A. Carballal, V. Maojo, A. Pazos, C. Fernandez-Lozano, A review on machine learning approaches and trends in drug discovery, *Comput. Struct. Biotechnol. J.* **19** (2021) 4538. 10.1016/j.csbj.2021.08.011.

[4] V. Kaul, S. Enslin, S.A. Gross, History of artificial intelligence in medicine, *Gastrointest. Endosc.* **92** (2020) 807. 10.1016/j.gie.2020.06.040.

[5] O.V. Prezhdo, Advancing physical chemistry with machine learning, *J. Phys. Chem. Lett.* **11** (2020) 9656. 10.1021/acs.jpclett.0c03130.

[6] A.F. de Almeida, R. Moreira, T. Rodrigues, Synthetic organic chemistry driven by artificial intelligence, *Nat. Rev. Chem.* **3** (2019) 589. 10.1038/s41570-019-0124-0.

[7] K.T. Butler, D.W. Davies, H. Cartwright, O. Isayev, A. Walsh, Machine learning for molecular and materials science, *Nature*. **559** (2018) 547. 10.1038/s41586-018-0337-2.

[8] A. Mistry, A.A. Franco, S.J. Cooper, S.A. Roberts, V. Viswanathan, How machine learning will revolutionize electrochemical sciences, *ACS Energy Lett.* **6** (2021) 1422. 10.1021/acsenergylett.1c00194.

[9] M. Meuwly, Machine learning for chemical reactions, *Chem. Rev.* **121** (2021) 10218. 10.1021/acs.chemrev.1c00033.

[10] Z.J. Baum, X. Yu, P.Y. Ayala, Y. Zhao, S.P. Watkins, Q. Zhou, Artificial intelligence in chemistry: Current trends and future directions, *J. Chem. Inf. Model.* **61** (2021) 3197. 10.1021/acs.jcim.1c00619.

[11] J.G. Greener, S.M. Kandathil, L. Moffat, D.T. Jones, A guide to machine learning for biologists, *Nat. Rev. Mol. Cell Biol.* **23** (2022) 40. 10.1038/s41580-021-00407-0.

[12] S.J. Goodswen, J.L.N. Barratt, P.J. Kennedy, A. Kaufer, L. Calarco, J.T. Ellis, Machine learning and applications in microbiology, *FEMS Microbiol. Rev.* **45** (2021) 1. 10.1093/femsre/fuab015.

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

[14] E. Bedolla, L.C. Padierna, R. Castañeda-Priego, Machine learning for condensed matter physics, *J. Phys.: Condens. Matter*. **33** (2020) 053001. 10.1088/1361-648X/abb895.

[15] S.L. Brunton, B.R. Noack, P. Koumoutsakos, Machine learning for fluid mechanics, *Annu. Rev. Fluid Mech*. **52** (2020) 477. 10.1146/annurev-fluid-010719-060214.

[16] M.P. Brenner, J.D. Eldredge, J.B. Freund, Perspective on machine learning for advancing fluid mechanics, Phys. Rev. Fluids. 4 (2019) 100501. 10.1103/PhysRevFluids.4.100501.

[17] H. Tao, T. Wu, M. Aldeghi, T.C. Wu, A. Aspuru-Guzik, E. Kumacheva, Nanoparticle synthesis assisted by machine learning, *Nat. Rev. Mater.* **6** (2021) 701. 10.1038/s41578-021-00337-5.

[18] J.-F. Masson, J.S. Biggins, E. Ringe, Machine learning for nanoplasmonics, *Nat. Nanotechnol.* (2023) 1. 10.1038/s41565-022-01284-0.

[19] P.R. Wiecha, A. Arbouet, C. Girard, O.L. Muskens, Deep learning in nano-photonics: inverse design and beyond, *Photon. Res.* **9** (2021) 182. 10.1364/PRJ.415960.

[20] J. Wei, X. Chu, X.-Y. Sun, K. Xu, H.-X. Deng, J. Chen, Z. Wei, M. Lei, Machine learning in materials science, *InfoMat.* **1** (2019) 338. 10.1002/inf2.12028.

[21] G.R. Schleder, A.C.M. Padilha, C.M. Acosta, M. Costa, A. Fazzio, From DFT to machine learning: Recent approaches to materials science – A review, *J. Phys. Mater.* **2** (2019) 032001. 10.1088/2515-7639/ab084b.

[22] Y. Hong, B. Hou, H. Jiang, J. Zhang, Machine learning and artificial neural network accelerated computational discoveries in materials science, *WIREs Comput. Mol. Sci.* **10** (2020) 1450. 10.1002/wcms.1450.

[23] C. Gao, X. Min, M. Fang, T. Tao, X. Zheng, Y. Liu, X. Wu, Z. Huang, Innovative materials science *via* machine learning, *Adv. Funct. Mater.* **32** (2022) 2108044. 10.1002/adfm.202108044.

[24] C. Chen, Y. Zuo, W. Ye, X. Li, Z. Deng, S.P. Ong, A critical review of machine learning of energy materials, *Adv. Energy Mater.* **10** (2020) 1903242. 10.1002/aenm.201903242.

[25] Y. Liu, O.C. Esan, Z. Pan, L. An, Machine learning for advanced energy materials, *Energy AI.* **3** (2021) 100049. 10.1016/j.egyai.2021.100049.

[26] G.L.W. Hart, T. Mueller, C. Toher, S. Curtarolo, Machine learning for alloys, Nat. Rev. Mater. 6 (2021) 730. 10.1038/s41578-021-00340-w.

[27] S. Das, H. Pegu, K.K. Sahu, A.K. Nayak, S. Ramakrishna, D. Datta, S. Swayamjyoti, Machine learning in materials modeling - Fundamentals and the opportunities in 2D materials, in: E.-H. Yang, D. Datta, J. Ding, G. Hader (Eds.), *Synthesis, Modeling, and Characterization of 2D Materials, and Their Heterostructures*, Elsevier, (2020) 445. 10.1016/B978-0-12-818475-2.00019-2.

[28] O.A. Montesinos López, A. Montesinos López, J. Crossa, Overfitting, model tuning, and evaluation of prediction performance, in: O.A. Montesinos López, A. Montesinos López, J. Crossa (Eds.), *Multivariate Statistical Machine Learning Methods for Genomic Prediction*, Springer International Publishing, Cham, (2022) 109. 10.1007/978-3-030-89010-0_4.

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay, Scikit-learn: Machine learning in Python, *J. Mach. Learn. Res*. **12** (2011) 2825. jmlr.org/papers/v12/pedregosa11a.html (accessed February 13, 2023).

[30] J.H. Metzen, G. Lemaitre, Comparison of kernel ridge and Gaussian process regression, (2023). scikit-learn/stable/auto_examples/gaussian_process/plot_compare_gpr_krr.html (accessed February 13, 2023).

[31] G. James, D. Witten, T. Hastie, R. Tibshirani, Linear regression, in: G. James, D. Witten, T. Hastie, R. Tibshirani (Eds.), *An Introduction to Statistical Learning: With Applications in R*, Springer US, New York, NY, (2021) 59. 10.1007/978-1-0716-1418-1_3.

[32] B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers, in: *Proc. 5th ACM Workshop Comput. Learn. Theory*, Association for Computing Machinery, New York, NY, (1992) 144. 10.1145/130385.130401.

[33] N. Cristianini, J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, Cambridge, 2000. 10.1017/CBO9780511801389.

[34] J.A.K. Suykens, J. Vandewalle, Least squares support vector machine classifiers, *Neural Process. Lett.* **9** (1999) 293. 10.1023/A:1018628609742.

[35] V. Vovk, Kernel ridge regression, in: B. Schölkopf, Z. Luo, V. Vovk (Eds.), *Empirical Inference*, Springer, Berlin, (2013) 105. 10.1007/978-3-642-41136-6_11.

[36] C.E. Rasmussen, C.K.I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, (2005) 10.7551/mitpress/3206.001.0001.

[37] G. James, D. Witten, T. Hastie, R. Tibshirani, Tree-based methods, in: G. James, D. Witten, T. Hastie, R. Tibshirani (Eds.), *An Introduction to Statistical Learning: With Applications in R*, Springer US, New York, NY, (2021) 327. 10.1007/978-1-0716-1418-1_8.

[38] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5. https://doi.org/10.1023/A:1010933404324.

[39] J.H. Friedman, Stochastic gradient boosting, *Comput. Stat. Data Anal*. **38** (2002) 367. 10.1016/S0167-9473(01)00065-2.

[40] B. Greenwell, Gradient boosting, in: B. Boehmke, B. Greenwell (Eds.), *Hands-On Machine Learning with R*, CRC Press, Boca Raton, FL, 2020.

[41] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, *J. Comput. Syst. Sci.* **55** (1997) 119. doi.org/10.1006/jcss.1997.1504.

[42] G. Shakhnarovich, D. Darrell, P. Indyk, (Eds.), *Nearest-Neighbor Methods in Learning and Vision*, MIT Press, Cambridge, MT, 2006.

[43] E. Fix, J.L. Hodges, Discriminatory analysis - Nonparametric discrimination: Consistency properties, *Int. Stat. Rev.* **57** (1989) 238. 10.2307/1403797.

[44] T. Cover, P. Hart, Nearest neighbor pattern classification, *IEEE Trans. Inf. Theory.* **13** (1967) 21. 10.1109/TIT.1967.1053964.

[45] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, Hoboken, NJ, 1998.

[46] G. Cybenko, Approximation by superpositions of a sigmoidal function, *Math. Control Signal Systems*. **2** (1989) 303. 10.1007/BF02551274.

[47] N. Savage, Breaking into the black box of artificial intelligence, *Nature*. (2022). 10.1038/d41586-022-00858-1.

[48] A. Franceschetti, A. Zunger, The inverse band-structure problem of finding an atomic configuration with given electronic properties, *Nature*. **402** (1999) 60. 10.1038/46995.

[49] C. Kuhn, D.N. Beratan, Inverse strategies for molecular design, *J. Phys. Chem.* **100** (1996) 10595. 10.1021/jp960518i.

[50] P.V. Yee, S. Haykin, *Regularized radial basis function networks: Theory and applications*, Wiley, Hoboken, NJ, 2001.

[51] A. Abraham, Adaptation of fuzzy inference system using neural learning, in: N. Nedjah, L. de Macedo Mourelle (Eds.), *Fuzzy Systems Engineering: Theory and Practice*, Springer, Berlin, (2005) 53. 10.1007/11339366_3.

[52] J.-S.R. Jang, ANFIS: adaptive-network-based fuzzy inference system, *IEEE Trans. Systems, Man, and Cybernetics.* **23** (1993) 665. 10.1109/21.256541.

[53] J. Schmidhuber, Deep learning in neural networks: An overview, *Neural Netw.* **61** (2015) 85. 10.1016/j.neunet.2014.09.003.

[54] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, *Nature*. **521** (2015) 436. 10.1038/nature14539.

[55] F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner, G. Monfardini, The graph neural network model, *IEEE Trans. Neural Networks*. **20** (2009) 61. 10.1109/TNN.2008.2005605.

[56] R. Venkatesan, B. Li, *Convolutional Neural Networks in Visual Computing: A Concise Guide*, CRC Press, 2017.

[57] U. Michelucci, *Advanced Applied Deep Learning: Convolutional Neural Networks and Object Detection*, Apress, 2019.

[58] Y. Hu, M.J. Buehler, Deep language models for interpretative and predictive materials science, *APL Machine Learning*. **1** (2023) 010901. 10.1063/5.0134317.

[59] M.R. Bonyadi, Z. Michalewicz, Particle swarm optimization for single objective continuous space problems: A review, *Evol. Comput.* **25** (2017) 1. 10.1162/EVCO_r_00180.

[60] J. Kennedy, R. Eberhart, Particle swarm optimization, in: *Proc. Int. Conf. Neural Networks*, (1995) 1942. 10.1109/ICNN.1995.488968.

[61] K.E. Parsopoulos, M.N. Vrahatis, Particle swarm optimization method in multiobjective problems, in: *Proceedings of the 2002 ACM Symposium on Applied Computing*, Association for Computing Machinery, New York, NY, (2002) 603. 10.1145/508791.508907.

[62] C.A. Coello Coello, M.S. Lechuga, MOPSO: A proposal for multiple objective particle swarm optimization, in: *Proc. CEC, IEEE Computer Society*, USA, (2002) 1051.

[63] K. Mason, J. Duggan, E. Howley, Multi-objective dynamic economic emission dispatch using particle swarm optimisation variants, *Neurocomputing*. **270** (2017) 188. 10.1016/j.neucom.2017.03.086.

[64] W. Banzhaf, *Genetic programming : an introduction on the automatic evolution of computer programs and its applications*, Morgan Kaufmann Publishers ; Heidelburg : Dpunkt-verlag, San Francisco, CA, (1998).

[65] D.E. Goldberg, *Genetic algorithms in search, optimization and machine learning*, 13th ed., Addison-Wesley Professional, Boston, (1989).

[66] E.K.P. Chong, S.H. Zak, *Gradient methods, in: An Introduction to Optimization*, John Wiley & Sons, Hoboken, NJ, (2013) 131.

[67] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, ArXiv. **412.6980** (2017). 10.48550/arXiv.1412.6980.

[68] Z. Zhang, Improved Adam optimizer for deep neural networks, in: *IEEE/ACM 26th Int. Symp. Qual. Serv.*, (2018) 1. 10.1109/IWQoS.2018.8624183.

[69] B.J. Shields, J. Stevens, J. Li, M. Parasram, F. Damani, J.I.M. Alvarado, J.M. Janey, R.P. Adams, A.G. Doyle, Bayesian reaction optimization as a tool for chemical synthesis, *Nature.* **590** (2021) 89. 10.1038/s41586-021-03213-y.

[70] Y. Xie, K. Sattari, C. Zhang, J. Lin, Toward autonomous laboratories: Convergence of artificial intelligence and experimental automation, *Progress in Materials Science*. **132** (2023) 101043. 10.1016/j.pmatsci.2022.101043.

[71] J.P.S. Aniceto, B. Zêzere, C.M. Silva, Predictive models for the binary diffusion coefficient at infinite dilution in polar and nonpolar fluids, *Materials*. **14** (2021) 542. 10.3390/ma14030542.

[72] M. Lashkarbolooki, Z.S. Shafipour, A.Z. Hezave, H. Farmani, Use of artificial neural networks for prediction of phase equilibria in the binary system containing carbon dioxide, *J. Supercrit. Fluids.* **75** (2013) 144. 10.1016/j.supflu.2012.12.032.

[73] F. Sun, G. Xie, S. Li, An artificial-neural-network based prediction of heat transfer behaviors for in-tube supercritical CO_2 flow, *Appl. Soft Comput.* **102** (2021) 107110. 10.1016/j.asoc.2021.107110.

[74] R.S.M. Freitas, Á.P.F. Lima, C. Chen, F.A. Rochinha, D. Mira, X. Jiang, Towards predicting liquid fuel physicochemical properties using molecular dynamics guided machine learning models, *Fuel*. **329** (2022) 125415. 10.1016/j.fuel.2022.125415.

[75] C. Zhang, J. Jiang, X. Peng, W. Zhao, J. Li, Comparison and correction of CO_2 properties model in critical region, *CIESC J*. **70** (2019) 3058. 10.11949/0438-1157.20190184.

[76] C. Ni, X. Wang, H. Liu, K. Zhang, X. Zheng, Y. Duan, Gradient-harmonizing-based deep learning for thermophysical properties of carbon dioxide, *J. Thermophys. Heat Transfer.* **37** (2022) 1. 10.2514/1.T6729.

[77] B. Li, Y. Lee, W. Yao, Y. Lu, X. Fan, Development and application of ANN model for property prediction of supercritical kerosene, *Comput. Fluids.* **209** (2020) 104665. 10.1016/j.compfluid.2020.104665.

[78] D.T. Banuti, A critical assessment of adaptive tabulation for fluid properties using neural networks, in: *AIAA Scitech 2021 Forum*, AIAA, (2021) 0927. 10.2514/6.2021-0927.

[79] N.P. Longmire, D.T. Banuti, Limits of fluid modeling for high pressure flow simulations, *Aerospace.* **9** (2022) 643. 10.3390/aerospace9110643.

[80] N.P. Longmire, D.T. Banuti, Modeling of the supercritical boiling curve by forced convection for supercritical fluids in relation to regenerative cooling, in: *AIAA Scitech 2021 Forum*, AIAA, (2021) 1406. 10.2514/6.2021-1406.

[81] N. Longmire, D.T. Banuti, Onset of heat transfer deterioration caused by pseudo-boiling in CO_2 laminar boundary layers, *Int. J. Heat Mass Transfer.* **193** (2022) 122957. 10.1016/j.ijheatmasstransfer.2022.122957.

[82] J.P.S. Aniceto, B. Zêzere, C.M. Silva, Machine learning models for the prediction of diffusivities in supercritical CO_2 systems, *J. Mol. Liq.* **326** (2021) 115281. 10.1016/j.molliq.2021.115281.

[83] X. Zhao, T. Luo, H. Jin, Predicting diffusion coefficients of binary and ternary supercritical water mixtures *via* machine and transfer learning with deep neural network, *Ind. Eng. Chem. Res.* **61** (2022) 8542. 10.1021/acs.iecr.2c00017.

[84] M.Y. Ha, T.J. Yoon, T. Tlusty, Y. Jho, W.B. Lee, Widom delta of supercritical gas–liquid coexistence, *J. Phys. Chem. Lett.* **9** (2018) 1734. 10.1021/acs.jpclett.8b00430.

[85] M.Y. Ha, T.J. Yoon, T. Tlusty, Y. Jho, W.B. Lee, Universality, scaling, and collapse in supercritical fluids, *J. Phys. Chem. Lett.* **11** (2020) 451. 10.1021/acs.jpclett.9b03360.

[86] J. Zhang, Q. Feng, S. Wang, X. Zhang, S. Wang, Estimation of CO_2–brine interfacial tension using an artificial neural network, *J. Supercrit. Fluids*. **107** (2016) 31. 10.1016/j.supflu.2015.08.010.

[87] U. Que-Salinas, P. E. Ramírez-González, A. Torres-Carbajal, Determination of thermodynamic state variables of liquids from their microscopic structures using an artificial neural network, *Soft Matter*. **17** (2021) 1975. 10.1039/D0SM02127J.

[88] Y. Liu, W. Hong, B. Cao, Machine learning for predicting thermodynamic properties of pure fluids and their mixtures, *Energy*. **188** (2019) 116091. 10.1016/j.energy.2019.116091.

[89] R. Shams, S. Esmaili, S. Rashid, M. Suleymani, An intelligent modeling approach for prediction of thermal conductivity of CO_2, *J. Nat. Gas Sci. Eng.* **27** (2015) 138. 10.1016/j.jngse.2015.08.050.

[90] A. Tatar, A. Barati-Harooni, A. Najafi-Marghmaleki, B. Norouzi-Farimani, A.H. Mohammadi, Predictive model based on ANFIS for estimation of thermal conductivity of carbon dioxide, *J. Mol. Liq.* **224** (2016) 1266. 10.1016/j.molliq.2016.10.112.

[91] M. Nait Amar, A. Jahanbani Ghahfarokhi, N. Zeraibi, Predicting thermal conductivity of carbon dioxide using group of data-driven models, *J. Taiwan Inst. Chem. Eng.* **113** (2020) 165. 10.1016/j.jtice.2020.08.001.

[92] V.H. Alvarez, M.D.A. Saldaña, Thermodynamic prediction of vapor–liquid equilibrium of supercritical CO_2 or CHF_3 +ionic liquids, *J. Supercrit. Fluids.* **66** (2012) 29. 10.1016/j.supflu.2012.02.011.

[93] S. Abdolbaghi, A. Barati-Harooni, A. Najafi-Marghmaleki, Improving the prediction ability of reference correlation for viscosity of carbon dioxide, *J. CO_2 Util.* **31** (2019) 106. 10.1016/j.jcou.2019.02.022.

[94] K. Zhu, E.A. Müller, Generating a machine-learned equation of state for fluid properties, *J. Phys. Chem. B.* **124** (2020) 8628. 10.1021/acs.jpcb.0c05806.

[95] X. Liu, S.K. Schnell, J.-M. Simon, P. Krüger, D. Bedeaux, S. Kjelstrup, A. Bardow, T.J.H. Vlugt, Diffusion coefficients from molecular dynamics simulations in binary and ternary mixtures, *Int. J. Thermophys.* **34** (2013) 1169. 10.1007/s10765-013-1482-3.

[96] R.V. Vaz, A.L. Magalhães, C.M. Silva, Improved hydrodynamic equations for the accurate prediction of diffusivities in supercritical carbon dioxide, *Fluid Phase Equilib.* **360** (2013) 401. 10.1016/j.fluid.2013.09.052.

[97] J.P.S. Aniceto, *Machine Learning models for D12 prediction*, (2022). github.com/EgiChem/ml-D12-app.

[98] S. Mohanty, Estimation of vapour liquid equilibria of binary systems, carbon dioxide–ethyl caproate, ethyl caprylate and ethyl caprate using artificial neural networks, *Fluid Phase Equilib.* **235** (2005) 92. 10.1016/j.fluid.2005.07.003.

[99] Y. Li, T. Zhang, S. Sun, X. Gao, Accelerating flash calculation through deep learning methods, *J. Comput. Phys.* **394** (2019) 153. 10.1016/j.jcp.2019.05.028.

[100] Y. Li, T. Zhang, S. Sun, Acceleration of the NVT flash calculation for multicomponent mixtures using deep neural network models, *Ind. Eng. Chem. Res.* **58** (2019) 12312. 10.1021/acs.iecr.9b00527.

[101] T. Zhang, Y. Li, S. Sun, H. Bai, Accelerating flash calculations in unconventional reservoirs considering capillary pressure using an optimized deep learning algorithm, *J. Pet. Sci. Eng.* **195** (2020) 107886. 10.1016/j.petrol.2020.107886.

[102] B. Winter, C. Winter, T. Esper, J. Schilling, A. Bardow, SPT-NRTL: A physics-guided machine learning model to predict thermodynamically consistent activity coefficients, *Fluid Phase Equilib.* **568** (2023) 113731. 10.1016/j.fluid.2023.113731.

[103] T. Brouwer, S.R.A. Kersten, G. Bargeman, B. Schuur, Trends in solvent impact on infinite dilution activity coefficients of solutes reviewed and visualized using an algorithm to support selection of solvents for greener fluid separations, *Sep. Purif. Technol.* **272** (2021) 118727. 10.1016/j.seppur.2021.118727.

[104] *Dortmund Data Bank*, (2023). ddbst.com/.

[105] M.J. Kamali, M. Mousavi, Analytic, neural network, and hybrid modeling of supercritical extraction of α-pinene, *J. Supercrit. Fluids.* **47** (2008) 168. 10.1016/j.supflu.2008.08.005.

[106] R.M. Alzhrani, A.H. Almalki, S.I. Alaqel, S. Alshehri, Novel numerical simulation of drug solubility in supercritical CO_2 using machine learning technique: Lenalidomide case study, *Arabian J. Chem.* **15** (2022) 104180. 10.1016/j.arabjc.2022.104180.

[107] H. Chinh Nguyen, F. Alamray, M. Kamal, T. Diana, A. Mohamed, M. Algarni, C.-H. Su, Computational prediction of drug solubility in supercritical carbon dioxide: Thermodynamic and artificial intelligence modeling, *J. Mol. Liq.* **354** (2022) 118888. 10.1016/j.molliq.2022.118888.

[108] B. Huwaimel, A. Alobaida, Anti-cancer drug solubility development within a green solvent: Design of novel and robust mathematical models based on artificial intelligence, *Molecules.* **27** (2022) 5140. 10.3390/molecules27165140.

[109] A. Sadeghi, C.-H. Su, A. Khan, M. Lutfor Rahman, M. Sani Sarjadi, S.M. Sarkar, Machine learning simulation of pharmaceutical solubility in supercritical carbon dioxide: Prediction and experimental validation for busulfan drug, *Arabian J. Chem.* **15** (2022) 103502. 10.1016/j.arabjc.2021.103502.

[110] F. An, B.T. Sayed, R.M.R. Parra, M.H. Hamad, R. Sivaraman, Z. Zanjani Foumani, A.A. Rushchitc, E. El-Maghawry, R.M. Alzhrani, S. Alshehri, K. M. AboRas, Machine learning model for prediction of drug solubility in supercritical solvent: Modeling and experimental validation, *J. Mol. Liq.* **363** (2022) 119901. 10.1016/j.molliq.2022.119901.

[111] Y. Cao, A. Khan, S. Zabihi, A.B. Albadarin, Neural simulation and experimental investigation of chloroquine solubility in supercritical solvent, *J. Mol. Liq.* **333** (2021) 115942. 10.1016/j.molliq.2021.115942.

[112] H. Zhu, L. Zhu, Z. Sun, A. Khan, Machine learning based simulation of an anti-cancer drug (busulfan) solubility in supercritical carbon dioxide: ANFIS model and experimental validation, *J. Mol. Liq.* **338** (2021) 116731. 10.1016/j.molliq.2021.116731.

[113] W. Liu, R. Zhao, X. Su, A. Mohamed, T. Diana, Development and validation of machine learning models for prediction of nanomedicine solubility in supercritical solvent for advanced pharmaceutical manufacturing, *J. Mol. Liq.* **358** (2022) 119208. 10.1016/j.molliq.2022.119208.

[114] E.V. Kostyrin, V.V. Ponkratov, A. Salah Al-Shati, Development of machine learning model and analysis study of drug solubility in supercritical solvent for green technology development, *Arabian J. Chem.* **15** (2022) 104346. 10.1016/j.arabjc.2022.104346.

[115] S.M. Alshahrani, A.A. Saqr, M.M. Alfadhel, A.S. Alshetaili, B.K. Almutairy, A.M. Alsubaiyel, A.H. Almari, J.A. Alamoudi, M.A.S. Abourehab, Application of CO_2 supercritical fluid to optimize the solubility of oxaprozin: Development of novel machine learning predictive models, *Molecules.* **27** (2022) 5762. 10.3390/molecules27185762.

[116] L. Mengshan, L. Liang, H. Xingyuan, L. Hesheng, C. Bingsheng, G. Lixin, W. Yan, Prediction of supercritical carbon dioxide solubility in polymers based on hybrid artificial intelligence method integrated with the diffusion theory, *RSC Adv.* **7** (2017) 49817. 10.1039/C7RA09531G.

[117] Y. Bakhbakhi, Phase equilibria prediction of solid solute in supercritical carbon dioxide with and without a cosolvent: The use of artificial neural network, *Expert Syst. Appl.* **38** (2011) 11355. 10.1016/j.eswa.2011.03.003.

[118] A. KhazaiePoul, M. Soleimani, S. Salahi, Solubility prediction of disperse dyes in supercritical carbon dioxide and ethanol as co-solvent using neural network, *Chin. J. Chem. Eng.* **24** (2016) 491. 10.1016/j.cjche.2015.11.027.

[119] M. Najmi, M.A. Ayari, H. Sadeghsalehi, B. Vaferi, A. Khandakar, M.E.H. Chowdhury, T. Rahman, Z.H. Jawhar, Estimating the dissolution of anticancer drugs in supercritical carbon dioxide with a stacked machine learning model, *Pharmaceutics.* **14** (2022) 1632. 10.3390/pharmaceutics14081632.

[120] T. Fujii, M. Kobune, Prediction of partition coefficient in high-pressure carbon dioxide–water systems using machine learning, *J. Supercrit. Fluids.* **179** (2022) 1. 10.1016/j.supflu.2021.105421.

[121] T. Khayamian, M. Esteki, Prediction of solubility for polycyclic aromatic hydrocarbons in supercritical carbon dioxide using wavelet neural networks in quantitative structure property relationship, *J. Supercrit. Fluids.* **32** (2004) 73. 10.1016/j.supflu.2004.02.003.

[122] B. Vaferi, M. Karimi, M. Azizi, H. Esmaeili, Comparison between the artificial neural network, SAFT and PRSV approach in obtaining the solubility of solid aromatic compounds in supercritical carbon dioxide, *J. Supercrit. Fluids.* **77** (2013) 44. 10.1016/j.supflu.2013.02.027.

[123] A. Aminian, Estimating the solubility of different solutes in supercritical CO_2 covering a wide range of operating conditions by using neural network models, *J. Supercrit. Fluids.* **125** (2017) 79. 10.1016/j.supflu.2017.02.007.

[124] M.R. Dadkhah, A. Tatar, A. Mohebbi, A. Barati-Harooni, A. Najafi-Marghmaleki, M.M. Ghiasi, A.H. Mohammadi, F. Pourfayaz, Prediction of solubility of solid compounds in supercritical CO_2 using a connectionist smart technique, *J. Supercrit. Fluids*. **120** (2017) 181. 10.1016/j.supflu.2016.06.006.

[125] X. Wang, Rigorous modeling of solubility of acid in supercritical carbon dioxide using connectionist approach: Comparison between ANN and density based modeling, *Energy Sources Part A.* **43** (2021) 1. 10.1080/15567036.2020.1871445.

[126] Y. Zhang, X. Xu, Machine learning bioactive compound solubilities in supercritical carbon dioxide, *Chem. Phys.* **550** (2021) 111299. 10.1016/j.chemphys.2021.111299.

[127] T. Rezaei, V. Nazarpour, N. Shahini, S. Bahmani, A. Shahkar, M. Abdihaji, S. Ahmadi, F.T. Shahdost, A universal methodology for reliable predicting the non-steroidal anti-inflammatory drug solubility in supercritical carbon dioxide, *Sci. Rep.* **12** (2022) 1043. 10.1038/s41598-022-04942-4.

[128] M. Lashkarbolooki, B. Vaferi, M.R. Rahimpour, Comparison the capability of artificial neural network (ANN) and EOS for prediction of solid solubilities in supercritical carbon dioxide, *Fluid Phase Equilib.* **308** (2011) 35. 10.1016/j.fluid.2011.06.002.

[129] H. Bahmaninia, M. Shateri, S. Atashrouz, K. Jabbour, A. Hemmati-Sarapardeh, A. Mohaddespour, Predicting the equilibrium solubility of CO_2 in alcohols, ketones, and glycol ethers: Application of ensemble learning and deep learning approaches, *Fluid Phase Equilib*. **567** (2023) 113712. 10.1016/j.fluid.2022.113712.

[130] A. Baghban, M.A. Ahmadi, B. Hashemi Shahraki, Prediction carbon dioxide solubility in presence of various ionic liquids using computational intelligence approaches, *J. Supercrit. Fluids.* **98** (2015) 50. 10.1016/j.supflu.2015.01.002.

[131] M. Mesbah, S. Shahsavari, E. Soroush, N. Rahaei, M. Rezakazemi, Accurate prediction of miscibility of CO_2 and supercritical CO_2 in ionic liquids using machine learning, *J. CO_2 Util.* **25** (2018) 99. 10.1016/j.jcou.2018.03.004.

[132] A. Aminian, B. ZareNezhad, A generalized neural network model for the VLE of supercritical carbon dioxide fluid extraction of fatty oils, *Fuel*. **282** (2020) 118823. 10.1016/j.fuel.2020.118823.

[133] M. Osada, K. Tamura, I. Shimada, Prediction of the solubility of organic compounds in high-temperature water using machine learning, J. Supercrit. Fluids. **190** (2022) 105733. 10.1016/j.supflu.2022.105733.

[134] H. Sovová, Apparent solubility of natural products extracted with near-critical carbon dioxide, *Am. J. Analyt. Chem*. **3** (2012) 958. 10.4236/ajac.2012.312A127.

[135] H. Ahangari, J.W. King, A. Ehsani, M. Yousefi, Supercritical fluid extraction of seed oils – A short review of current trends, *Trends Food Sci. Technol.* **111** (2021) 349. 10.1016/j.tifs.2021.02.066.

[136] B. Pavlić, L. Pezo, B. Marić, L.P. Tukuljac, Z. Zeković, M.B. Solarov, N. Teslić, Supercritical fluid extraction of raspberry seed oil: Experiments and modelling, *J. Supercrit. Fluids.* **157** (2020) 104687. 10.1016/j.supflu.2019.104687.

[137] M. Izadifar, F. Abdolahi, Comparison between neural network and mathematical modeling of supercritical CO_2 extraction of black pepper essential oil, *J. Supercrit. Fluids.* **38** (2006) 37. 10.1016/j.supflu.2005.11.012.

[138] A. Shokri, T. Hatami, M. Khamforoush, Near critical carbon dioxide extraction of Anise (*Pimpinella Anisum L.*) seed: Mathematical and artificial neural network modeling, *J. Supercrit. Fluids*. **58** (2011) 49. 10.1016/j.supflu.2011.04.011.

[139] G. Zahedi, A. Azarpour, Optimization of supercritical carbon dioxide extraction of *Passiflora* seed oil, *J. Supercrit. Fluids.* **58** (2011) 40. 10.1016/j.supflu.2011.04.013.

[140] M. Khajeh, M.G. Moghaddam, M. Shakeri, Application of artificial neural network in predicting the extraction yield of essential oils of *Diplotaenia cachrydifolia* by supercritical fluid extraction, *J. Supercrit. Fluids*. **69** (2012) 91. 10.1016/j.supflu.2012.05.006.

[141] S.M. Ghoreishi, E. Heidari, Extraction of Epigallocatechin-3-gallate from green tea *via* supercritical fluid technology: Neural network modeling and response surface optimization, *J. Supercrit. Fluids*. **74** (2013) 128. 10.1016/j.supflu.2012.12.009.

[142] E. Heidari, S.M. Ghoreishi, Prediction of supercritical extraction recovery of EGCG using hybrid of adaptive neuro-fuzzy inference system and mathematical model, *J. Supercrit. Fluids.* **82** (2013) 158. 10.1016/j.supflu.2013.07.006.

[143] M. Lashkarbolooki, Z.S. Shafipour, A.Z. Hezave, Trainable cascade-forward back-propagation network modeling of spearmint oil extraction in a packed bed using SC-CO_2, *J. Supercrit. Fluids.* **73** (2013) 108. 10.1016/j.supflu.2012.10.013.

[144] S.M. Ghoreishi, A. Hedayati, S.O. Mousavi, Quercetin extraction from *Rosa damascena* Mill *via* supercritical CO_2: Neural network and adaptive neuro fuzzy interface system modeling and response surface optimization, *J. Supercrit. Fluids*. **112** (2016) 57. 10.1016/j.supflu.2016.02.006.

[145] G. Sodeifian, S.A. Sajadian, N. Saadati Ardestani, Evaluation of the response surface and hybrid artificial neural network-genetic algorithm methodologies to determine extraction yield of *Ferulago angulata* through supercritical fluid, *J. Taiwan Inst. Chem. Eng*. **60** (2016) 165. 10.1016/j.jtice.2015.11.003.

[146] G. Sodeifian, S.A. Sajadian, N. Saadati Ardestani, Optimization of essential oil extraction from *Launaea acanthodes* Boiss: Utilization of supercritical carbon dioxide and cosolvent, *J. Supercrit. Fluids.* **116** (2016) 46. 10.1016/j.supflu.2016.05.015.

[147] K. Ameer, B.-S. Chun, J.-H. Kwon, Optimization of supercritical fluid extraction of steviol glycosides and total phenolic content from *Stevia rebaudiana* (Bertoni) leaves using response surface methodology and artificial neural network modeling, *Ind. Crops Prod.* **109** (2017) 672. 10.1016/j.indcrop.2017.09.023.

[148] B. Pavlić, O. Bera, S. Vidović, L. Ilić, Z. Zeković, Extraction kinetics and ANN simulation of supercritical fluid extraction of sage herbal dust, *J. Supercrit. Fluids.* **130** (2017) 327. 10.1016/j.supflu.2017.06.015.

[149] R. Vardanega, I.A. Dalmolin, G.C. Nogueira, T. Hatami, M.A.A. Meireles, Phase behaviour and mathematical modelling for the system annatto seed oil in compressed carbon dioxide + ethanol as co-solvent, *J. Supercrit. Fluids.* 130 (2017) 56. 10.1016/j.supflu.2017.07.038.

[150] B. Suryawanshi, B. Mohanty, Application of an artificial neural network model for the supercritical fluid extraction of seed oil from *Argemone mexicana* (L.) seeds, *Ind. Crops Prod*. **123** (2018) 64. 10.1016/j.indcrop.2018.06.057.

[151] S.X. Yang, W. Shi, J. Zeng, Modelling the supercritical fluid extraction of lycopene from tomato paste waste using neuro-fuzzy approaches, *Advances in Neural Networks - ISNN 2004*. (2004) 88. 10.1007/978-3-540-28648-6_140.

[152] P. Mitra, P.C. Barman, K.S. Chang, Coumarin extraction from *Cuscuta reflexa* using supercritical fluid carbon dioxide and development of an artificial neural network model to predict the coumarin yield, *Food Bioprocess Technol.* **4** (2011) 737. 10.1007/s11947-008-0179-2.

[153] A.K. Jha, N. Sit, Comparison of response surface methodology (RSM) and artificial neural network (ANN) modelling for supercritical fluid extraction of phytochemicals from *Terminalia chebula* pulp and optimization using RSM coupled with desirability function (DF) and genetic algorithm (GA) and ANN with GA, *Ind. Crops Prod*. **170** (2021) 113769. 10.1016/j.indcrop.2021.113769.

[154] J. Sargolzaei, A.H. Moghaddam, Predicting the yield of pomegranate oil from supercritical extraction using artificial neural networks and an adaptive-network-based fuzzy inference system, *Front. Chem. Sci. Eng.* **7** (2013) 357. 10.1007/s11705-013-1336-3.

[155] A. Hedayati, S.M. Ghoreishi, Artificial neural network and adaptive neuro-fuzzy interface system modeling of supercritical CO_2 extraction of glycyrrhizic acid from *Glycyrrhiza glabra L.*, *Chem. Prod. Process Model.* **11** (2016) 217. 10.1515/cppm-2015-0048.

[156] P. Davoodi, S.M. Ghoreishi, A. Hedayati, Optimization of supercritical extraction of galegine from *Galega officinalis L.*: Neural network modeling and experimental optimization *via* response surface methodology, *Korean J. Chem. Eng*. **34** (2017) 854. 10.1007/s11814-016-0304-2.

[157] Z. Zeković, O. Bera, S. Đurović, B. Pavlić, Supercritical fluid extraction of coriander seeds: Kinetics modelling and ANN optimization, *J. Supercrit. Fluids.* **125** (2017) 88. 10.1016/j.supflu.2017.02.006.

[158] M. Davoody, G. Zahedi, M. Biglari, M.A.A. Meireles, A. Bahadori, Expert and gray box modeling of high pressure liquid carbon dioxide extraction of *Pimpinella anisum L.* seed, *J. Supercrit. Fluids.* **72** (2012) 213. 10.1016/j.supflu.2012.09.002.

[159] S.A. Idris, M. Markom, N. Abd Rahman, J. Mohd Ali, Prediction of overall yield of *Gynura procumbens* from ethanol-water + supercritical CO_2 extraction using artificial neural network model, *Case Stud. Chem. Environ. Eng.* **5** (2022) 100175. 10.1016/j.cscee.2021.100175.

[160] I.C. Valim, A.S.C. Rego, A. Queiroz, V. Brant, A.A.F. Neto, C. Vilani, B.F. Santos, Use of artificial intelligence to experimental conditions identification in the process of delignification of sugarcane bagasse from supercritical carbon dioxide, *Comput. Aided Chem. Eng.* **43** (2018) 1469. 10.1016/B978-0-444-64235-6.50256-4.

[161] O. Farobie, Y. Matsumura, State of the art of biodiesel production under supercritical conditions, *Prog. Energy Combust. Sci.* **63** (2017) 173. 10.1016/j.pecs.2017.08.001.

[162] S. Kusdiana, S. Saka, Methyl esterification of free fatty acids of rapeseed oil as treated in supercritical methanol, *J. Chem. Eng. Jpn.* **34** (2001) 383. 10.1252/jcej.34.383.

[163] D. Kusdiana, S. Saka, Effects of water on biodiesel fuel production by supercritical methanol treatment, *Bioresour. Technol.* **91** (2004) 289. 10.1016/S0960-8524(03)00201-3.

[164] O. Farobie, N. Hasanah, Y. Matsumura, Artificial neural network modeling to predict biodiesel production in supercritical methanol and ethanol using spiral reactor, *Procedia Environ. Sci*. **28** (2015) 214. 10.1016/j.proenv.2015.07.028.

[165] J. Guo, A. Baghban, Application of ANFIS strategy for prediction of biodiesel production using supercritical methanol, *Energy Sources Part A.* **39** (2017) 1862. 10.1080/15567036.2017.1380731.

[166] A. Baghban, Computational modeling of biodiesel production using supercritical methanol, *Energy Sources Part A.* **41** (2018) 14. 10.1080/15567036.2017.1344748.

[167] O. Farobie, N. Hasanah, Artificial neural network approach to predict biodiesel production in supercritical tert-butyl methyl ether, *Indones. J. Sci. Technol.* **1** (2016) 23. 10.17509/ijost.v1i1.2218.

[168] A.N. Sarve, M.N. Varma, S.S. Sonawane, Response surface optimization and artificial neural network modeling of biodiesel production from crude mahua (*Madhuca indica*) oil under supercritical ethanol conditions using CO_2 as co-solvent, *RSC Adv.* **5** (2015) 69702. 10.1039/C5RA11911A.

[169] J. Zhang, J. Zhang, W. Cao, Relationship between the density of supercritical CO_2 + ethanol binary system and its critical properties, *Sci. China, Ser. B Chem.* **46** (2003) 290. 10.1360/02yb0189.

[170] Committee for Standardization, Automotive fuels—fatty acid methyl esters (FAME) for diesel engines—requirements and test methods, **EN 14214:2008**, European Committee for Standardization, Brussels (2008).

[171] G. Srivastava, A.K. Paul, V.V. Goud, Optimization of non-catalytic transesterification of microalgae oil to biodiesel under supercritical methanol condition, *Energy Convers. Manage.* **156** (2018) 269. 10.1016/j.enconman.2017.10.093.

[172] ASTM, Standard specification for diesel fuel oil, biodiesel blend (B6 to B20), **ASTM D7467-18**, ASTM International, West Conshohocken, PA, 2008.

[173] S. Sindhanai Selvan, P. Saravana Pandian, A. Subathira, S. Saravanan, Artificial neural network modeling-coupled genetic algorithm optimization of supercritical methanol transesterification of *Aegle marmelos* oil to biodiesel, *Biofuels.* **12** (2021) 797. 10.1080/17597269.2018.1542567.

[174] H. Shekarchizadeh, R. Tikani, M. Kadivar, Optimization of cocoa butter analog synthesis variables using neural networks and genetic algorithm, *J Food Sci Technol.* **51** (2014) 2099. 10.1007/s13197-012-0695-y.

[175] A. Loppinet-Serani, C. Aymonier, F. Cansell, Current and foreseeable applications of supercritical water for energy and the environment, *ChemSusChem.* **1** (2008) 486. 10.1002/cssc.200700167.

[176] A. Loppinet-Serani, C. Aymonier, F. Cansell, Supercritical water for environmental technologies, *J. Chem. Technol. Biotechnol.* **85** (2010) 582. 10.1002/jctb.2323.

[177] P. Casademont, M.B. García-Jarana, J. Sánchez-Oneto, J.R. Portela, E.J.M. Ossa, Supercritical water gasification: A patents review, *Rev. Chem. Eng.* **33** (2017) 237. 10.1515/revce-2016-0020.

[178] F.J. Gutiérrez Ortiz, Biofuel production from supercritical water gasification of sustainable biomass, *Energy Convers. Manage.: X.* **14** (2022) 100164. 10.1016/j.ecmx.2021.100164.

[179] P. Murugesan, V. Raja, S. Dutta, J.A. Moses, C. Anandharamakrishnan, Food waste valorisation *via* gasification – A review on emerging concepts, prospects and challenges, *Sci. Total Environ.* **851** (2022) 157955. 10.1016/j.scitotenv.2022.157955.

[180] L. Leng, W. Zhang, S. Leng, J. Chen, L. Yang, H. Li, S. Jiang, H. Huang, Bioenergy recovery from wastewater produced by hydrothermal processing biomass: Progress, challenges, and opportunities, *Sci. Total Environ.* **748** (2020) 142383. 10.1016/j.scitotenv.2020.142383.

[181] H. Shahbeik, W. Peng, H. Kazemi Shariat Panahi, M. Dehhaghi, G.J. Guillemin, A. Fallahi, H. Amiri, M. Rehan, D. Raikwar, H. Latine, B. Pandalone, B. Khoshnevisan, C. Sonne, L. Vaccaro, A.-S. Nizami, V.K. Gupta, S.S. Lam, J. Pan, R. Luque, B. Sels, M. Tabatabaei, M. Aghbashlo, Synthesis of liquid biofuels from biomass by hydrothermal gasification: A critical review, *Renewable Sustainable Energy Rev.* **167** (2022) 112833. 10.1016/j.rser.2022.112833.

[182] N. Wei, D. Xu, B. Hao, S. Guo, Y. Guo, S. Wang, Chemical reactions of organic compounds in supercritical water gasification and oxidation, *Water Res.* **190** (2021) 116634. 10.1016/j.watres.2020.116634.

[183] S. Shenbagaraj, P.K. Sharma, A.K. Sharma, G. Raghav, K.B. Kota, V. Ashokkumar, Gasification of food waste in supercritical water: An innovative synthesis gas composition prediction model based on artificial neural networks, *Int. J. Hydrogen Energy.* **46** (2021) 12739. 10.1016/j.ijhydene.2021.01.122.

[184] W. Wang, H. Jin, Artificial neural network prediction model for supercritical water gasification data of discarded circuit boards, *Chem. Eng. Trans.* **94** (2022) 1303. 10.3303/CET2294217.

[185] S. Zhao, J. Li, C. Chen, B. Yan, J. Tao, G. Chen, Interpretable machine learning for predicting and evaluating hydrogen production *via* supercritical water gasification of biomass, *J. Clean. Prod.* **316** (2021) 128244. 10.1016/j.jclepro.2021.128244.

[186] Z.U. Haq, H. Ullah, M.N.A. Khan, S.R. Naqvi, M. Ahsan, Hydrogen production optimization from sewage sludge supercritical gasification process using machine learning methods integrated with genetic algorithm, *Chem. Eng. Res. Des.* **184** (2022) 614. 10.1016/j.cherd.2022.06.020.

[187] S. Liu, Y. Yang, L. Yu, F. Zhu, Y. Cao, X. Liu, A. Yao, Y. Cao, Predicting gas production by supercritical water gasification of coal using machine learning, *Fuel.* **329** (2022) 125478. 10.1016/j.fuel.2022.125478.

[188] A. Mohammadidoust, M.R. Omidvar, Simulation and modeling of hydrogen production and power from wheat straw biomass at supercritical condition through Aspen Plus and ANN approaches, *Biomass Conv. Bioref.* **12** (2022) 3857. 10.1007/s13399-020-00933-5.

[189] P.V. Gopirajan, K.P. Gopinath, G. Sivaranjani, J. Arun, Optimization of hydrothermal gasification process through machine learning approach: Experimental conditions, product yield and pollution, *J. Clean. Prod.* **306** (2021) 127302. 10.1016/j.jclepro.2021.127302.

[190] W. Zhang, Q. Chen, J. Chen, D. Xu, H. Zhan, H. Peng, J. Pan, M. Vlaskin, L. Leng, H. Li, Machine learning for hydrothermal treatment of biomass: A review, *Bioresour. Technol.* **370** (2023) 128547. 10.1016/j.biortech.2022.128547.

[191] G.C. Umenweke, I.C. Afolabi, E.I. Epelle, J.A. Okolie, Machine learning methods for modeling conventional and hydrothermal gasification of waste biomass: A review, *Bioresour. Technol. Rep.* **17** (2022) 100976. 10.1016/j.biteb.2022.100976.

[192] J. Li, L. Pan, M. Suvarna, Y.W. Tong, X. Wang, Machine learning prediction of syngas composition of hydrothermal gasification from wet organic wastes, *Proc. Int. Conf. Appl. Energy*. **10** (2020) 199. 10.46855/energy-proceedings-7161.

[193] J. Li, L. Pan, M. Suvarna, X. Wang, Machine learning aided supercritical water gasification for H_2-rich syngas production with process optimization and catalyst screening, *Chem. Eng. J.* **426** (2021) 131285. 10.1016/j.cej.2021.131285.

[194] J. Li, M. Suvarna, L. Pan, Y. Zhao, X. Wang, A hybrid data-driven and mechanistic modelling approach for hydrothermal gasification, *Appl. Energy.* **304** (2021) 117674. 10.1016/j.apenergy.2021.117674.

[195] J. Li, L. Pan, M. Suvarna, X. Wang, Composition prediction of syngas from supercritical water gasification (SCWG) of wet waste, *Xianon SSE Group*. (2021). https://180.76.158.202:10002/ .

[196] D. Fózer, A.J. Tóth, P.S. Varbanov, J.J. Klemeš, P. Mizsey, Sustainability assessment of biomethanol production *via* hydrothermal gasification supported by artificial neural network, *J. Clean. Prod.* **318** (2021) 128606. 10.1016/j.jclepro.2021.128606.

[197] C. López-Iglesias, E.R. López, J. Fernández, M. Landin, C.A. García-González, Modeling of the production of lipid microparticles using PGSS® technique, *Molecules*. **25** (2020) 4927. 10.3390/molecules25214927.

[198] Nidhi, M. Rashid, V. Kaur, S.S. Hallan, S. Sharma, N. Mishra, Microparticles as controlled drug delivery carrier for the treatment of ulcerative colitis: A brief review, *Saudi Pharm. J.* **24** (2016) 458. 10.1016/j.jsps.2014.10.001.

[199] S. Jaspart, G. Piel, L. Delattre, B. Evrard, Solid lipid microparticles: Formulation, preparation, characterisation, drug release and applications, *Expert Opin Drug Deliv*. **2** (2005) 75. 10.1517/17425247.2.1.75.

[200] M. Lengyel, N. Kállai-Szabó, V. Antal, A.J. Laki, I. Antal, Microparticles, microspheres, and microcapsules for advanced drug delivery, *Sci. Pharm.* **87** (2019) 20. 10.3390/scipharm87030020.

[201] J.-Q. Guo, M.-J. Li, Y.-L. He, T. Jiang, T. Ma, J.-T. Xu, F. Cao, A systematic review of supercritical carbon dioxide (S-CO_2) power cycle for energy industries: Technologies, key issues, and potential prospects, *Energy Convers. Manage.* **258** (2022) 115437. 10.1016/j.enconman.2022.115437.

[202] X. Wang, R. Wang, X. Bian, J. Cai, H. Tian, G. Shu, X. Li, Z. Qin, Review of dynamic performance and control strategy of supercritical CO_2 Brayton cycle, *Energy AI.* **5** (2021) 100078. 10.1016/j.egyai.2021.100078.

[203] T. Neises, C. Turchi, A comparison of supercritical carbon dioxide power cycle configurations with an emphasis on CSP applications, *Energy Procedia*. **49** (2014) 1187. 10.1016/j.egypro.2014.03.128.

[204] F. Crespi, D. Sánchez, G.S. Martínez, T. Sánchez-Lencero, F. Jiménez-Espadafor, Potential of supercritical carbon dioxide power cycles to reduce the levelized cost of electricity of contemporary concentrated solar power plants, *Appl. Sci.* **10** (2020) 5049. 10.3390/app10155049.

[205] M.T. White, G. Bianchi, L. Chai, S.A. Tassou, A.I. Sayma, Review of supercritical CO_2 technologies and systems for power generation, *Appl. Therm. Eng.* **185** (2021) 116447. 10.1016/j.applthermaleng.2020.116447.

[206] R. Dhanuskodi, R. Kaliappan, S. Suresh, N. Anantharaman, A. Arunagiri, J. Krishnaiah, Artificial neural networks model for predicting wall temperature of supercritical boilers, *Appl. Therm. Eng*. **90** (2015) 749. 10.1016/j.applthermaleng.2015.07.036.

[207] K. Ye, Y. Zhang, L. Yang, Y. Zhao, N. Li, C. Xie, Modeling convective heat transfer of supercritical carbon dioxide using an artificial neural network, *Appl. Therm. Eng*. **150** (2019) 686. 10.1016/j.applthermaleng.2018.11.031.

[208] W. Chang, X. Chu, A.F. Binte Shaik Fareed, S. Pandey, J. Luo, B. Weigand, E. Laurien, Heat transfer prediction of supercritical water with artificial neural networks, *Appl. Therm. Eng.* **131** (2018) 815. https://doi.org/10.1016/j.applthermaleng.2017.12.063.

[209] X. Zhu, R. Zhang, X. Yu, Q. Qiu, L. Zhao, Study on artificial neural network-based prediction of thermal characteristics of supercritical CO_2 in vertical channels, *Int. Commun. Heat Mass Transfer*. **139** (2022) 106502. 10.1016/j.icheatmasstransfer.2022.106502.

[210] X. Lei, J. Zhang, L. Gou, Q. Zhang, H. Li, Experimental study on convection heat transfer of supercritical CO_2 in small upward channels, *Energy*. **176** (2019) 119. 10.1016/j.energy.2019.03.109.

[211] S.M. Pesteei, M. Mehrabi, Modeling of convection heat transfer of supercritical carbon dioxide in a vertical tube at low Reynolds numbers using artificial neural network, *Int. Commun. Heat Mass Transfer.* **37** (2010) 901. 10.1016/j.icheatmasstransfer.2010.05.018.

[212] M. Saeed, M.I. Radaideh, A.S. Berrouk, K. Alawadhi, Machine learning-based efficient multi-layered precooler design approach for supercritical CO_2 cycle, *Energy Convers. Manage.: X.* **11** (2021) 100104. 10.1016/j.ecmx.2021.100104.

[213] M. Saeed, A.S. Berrouk, Y.F. Al Wahedi, M.P. Singh, I.A. Dagga, I. Afgan, Performance enhancement of a C-shaped printed circuit heat exchanger in supercritical CO_2 Brayton cycle: A machine learning-based optimization study, *Case Stud. Therm. Eng.* **38** (2022) 102276. 10.1016/j.csite.2022.102276.

[214] M. Saeed, A.S. Berrouk, B.M. Burhani, A.M. Alatyar, Y.F.A. Wahedi, Turbine design and optimization for a supercritical CO_2 cycle using a multifaceted approach based on deep neural network, *Energies*. **14** (2021) 7807. 10.3390/en14227807.

[215] S. Son, Y. Jeong, S.K. Cho, J.I. Lee, Development of supercritical CO_2 turbomachinery off-design model using 1D mean-line method and deep neural network, *Appl. Energy*. **263** (2020) 114645. 10.1016/j.apenergy.2020.114645.

[216] L. Sun, T. Liu, D. Wang, C. Huang, Y. Xie, Deep learning method based on graph neural network for performance prediction of supercritical CO_2 power systems, *Appl. Energy.* **324** (2022) 119739. 10.1016/j.apenergy.2022.119739.

[217] C. Diao, T. Liu, Z. Yang, Y. Duan, Comparison between deep learning and fully connected neural network in performance prediction of power cycles: Taking supercritical CO_2 Brayton cycle as an example, *Int. J. Intell. Syst.* **36** (2021) 7682. 10.1002/int.22603.

[218] A.S. Mishamandani, M. Mojaddam, A. Mohseni, Performance improvement of supercritical CO_2 power cycles by machine learning method, (2022). Preprint. SSRN Repository. 4218407. 10.2139/ssrn.4218407.

[219] N.P. Longmire, D. Banuti, Extension of SU2 using neural networks for thermo-fluids modeling, *AIAA Propul. Energy Forum.* (2021). 10.2514/6.2021-3593.

[220] P.J. Milan, X. Wang, J.-P. Hickey, Y. Li, V. Yang, Accelerating numerical simulations of supercritical fluid flows using deep neural networks, *AIAA Scitech 2020 Forum*. (2020). 10.2514/6.2020-1157.

[221] P.J. Milan, J.-P. Hickey, X. Wang, V. Yang, Deep-learning accelerated calculation of real-fluid properties in numerical simulation of complex flowfields, *J. Comput. Phys.* **444** (2021) 110567. 10.1016/j.jcp.2021.110567.

[222] Y. Ge, M. Hansinger, C. Traxinger, M. Pfitzner, Deep residual learning applied to real-gas thermodynamics, *AIP Conf. Proc.* **2040** (2018) 150004. 10.1063/1.5079207.

[223] P.J. Linstrom, W.G. Mallard, *NIST chemistry webbook, NIST Standard Reference Database Number 69*, National Institute of Standards and Technology, Gaithersburg, MD, (2001) 10.18434/T4D303.

[224] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: an imperative style, high-performance deep learning library, in*: Proc. 33rd Int. Conf. Neural Information Processing Systems*, Curran Associates Inc., Red Hook, NY, (2019) 8026.

[225] M.X. Yao, Thermoacoustic Instabilities in Counterflow Diffusion Flames, Master Thesis, University of Waterloo, 2019. uwspace.uwaterloo.ca/handle/10012/15104 (accessed January 17, 2023).

[226] X. Xie, X. Wang, P. Zhao, Y. Hao, R. Xie, H. Liu, Learning time-aware multi-phase flow fields in coal-supercritical water fluidized bed reactor with deep learning, *Energy.* **263** (2023) 125907. 10.1016/j.energy.2022.125907.

[227] J. Ling, R. Jones, J. Templeton, Machine learning strategies for systems with invariance properties, *J. Comput. Phys*. **318** (2016) 22. 10.1016/j.jcp.2016.05.003.

[228] J. Ling, A. Kurzawski, J. Templeton, Reynolds averaged turbulence modelling using deep neural networks with embedded invariance, *J. Fluid Mech*. **807** (2016) 155. 10.1017/jfm.2016.615.

[229] Y. Cao, R. Xu, P. Jiang, Physics-informed machine learning based RANS turbulence modeling convection heat transfer of supercritical pressure fluid, *Int. J. Heat Mass Transfer*. **201** (2023) 123622. 10.1016/j.ijheatmasstransfer.2022.123622.

[230] N. Fedik, R. Zubatyuk, M. Kulichenko, N. Lubbers, J.S. Smith, B. Nebgen, R. Messerly, Y.W. Li, A.I. Boldyrev, K. Barros, O. Isayev, S. Tretiak, Extending machine learning beyond interatomic potentials for predicting molecular properties, *Nat. Rev. Chem*. **6** (2022) 653. 10.1038/s41570-022-00416-3.

[231] F. Noé, A. Tkatchenko, K.-R. Müller, C. Clementi, Machine learning for molecular simulation, *Annu. Rev. Phys. Chem.* **71** (2020) 361. 10.1146/annurev-physchem-042018-052331.

[232] P. Cats, S. Kuipers, S. de Wind, R. van Damme, G.M. Coli, M. Dijkstra, R. van Roij, Machine-learning free-energy functionals using density profiles from simulations, *APL Mater*. **9** (2021) 031109. 10.1063/5.0042558.

[233] P. Schienbein, D. Marx, Supercritical water is not hydrogen bonded, *Angew. Chem. Int. Ed*. **59** (2020) 18578. 10.1002/anie.202009640.

[234] W. Zhang, L. Zhou, B. Yang, T. Yan, Molecular dynamics simulations of LiCl ion pairs in high temperature aqueous solutions by deep learning potential, *J. Mol. Liq.* **367** (2022) 120500. 10.1016/j.molliq.2022.120500.

[235] Y. Zhang, H. Wang, W. Chen, J. Zeng, L. Zhang, H. Wang, W. E, DP-GEN: A concurrent learning platform for the generation of reliable deep learning based potential energy models, *Comput. Phys. Commun.* **253** (2020) 107206. 10.1016/j.cpc.2020.107206.

[236] B. Cheng, G. Mazzola, C.J. Pickard, M. Ceriotti, Evidence for supercritical behaviour of high-pressure liquid hydrogen, *Nature*. **585** (2020) 217. 10.1038/s41586-020-2677-y.

[237] V.V. Karasiev, J. Hinz, S.X. Hu, S.B. Trickey, On the liquid–liquid phase transition of dense hydrogen, *Nature*. **600** (2021) 12. 10.1038/s41586-021-04078-x.

[238] B. Cheng, G. Mazzola, C.J. Pickard, M. Ceriotti, Reply to: On the liquid–liquid phase transition of dense hydrogen, *Nature*. **600** (2021) 15. 10.1038/s41586-021-04079-w.

[239] E.F. O'Bannon, Z. Jenei, H. Cynn, M.J. Lipp, J.R. Jeffries, Culet diameter and the achievable pressure of a diamond anvil cell: Implications for the upper pressure limit of a diamond anvil cell, *Rev. Sci. Instrum*. **89** (2018) 111501. 10.1063/1.5049720.

[240] D. Sharma, C. Lecoutre, F. Palencia, O. Nguyen, A. Erriguible, S. Marre, Assessment of machine learning algorithms for predicting autoignition and ignition delay time in microscale supercritical water oxidation process, *Fuel*. **352** (2023) 129098. 10.1016/j.fuel.2023.129098.

[241] H.E. Stanley, S.V. Buldyrev, N. Giovambattista, Static heterogeneities in liquid water, *Physica A*. **342** (2004) 40. 10.1016/j.physa.2004.06.045.

[242] K. Nishikawa, I. Tanaka, Correlation lengths and density fluctuations in supercritical states of carbon dioxide, *Chem. Phys. Lett.* **244** (1995) 149. 10.1016/0009-2614(95)00818-O.

[243] K. Nishikawa, K. Kusano, A. Ayusawa Arai, T. Morita, Density fluctuation of a van der Waals fluid in supercritical state, *J. Chem. Phys*. **118** (2003) 1341. 10.1063/1.1526469.

[244] K. Nishikawa, A.A. Arai, T. Morita, Density fluctuation of supercritical fluids obtained from small-angle X-ray scattering experiment and thermodynamic calculation, *J. Supercrit. Fluids*. **30** (2004) 249. 10.1016/j.supflu.2003.09.003.

[245] K. Nishikawa, T. Morita, Inhomogeneity of molecular distribution in supercritical fluids, *Chem. Phys. Lett*. **316** (2000) 238. 10.1016/S0009-2614(99)01241-5.

[246] S.C. Tucker, Solvent density inhomogeneities in supercritical fluids, *Chem. Rev*. **99** (1999) 391. 10.1021/cr9700437.

[247] F. Maxim, K. Karalis, P. Boillat, D.T. Banuti, J.I.M. Damian, B. Niceno, Ludwig, C., Thermodynamics and dynamics of supercritical water pseudo-boiling, *Adv. Sci*. **8** (2020) 2002312. 10.1002/advs.202002312.

[248] D. Kajiya, K. Nishikawa, K.-I. Saitow, Time evolution of density fluctuation in the supercritical region. 2. Comparison of hydrogen- and non-hydrogen-bonded fluids, *J. Phys. Chem. A*. **109** (2005) 7365. 10.1021/jp0522056.

[249] F. Maxim, C. Contescu, P. Boillat, B. Niceno, K. Karalis, A. Testino, C. Ludwig, Visualization of supercritical water pseudo-boiling at Widom line crossover, *Nature Commun*. **10** (2019) 4114. 10.1038/s41467-019-12117-5.

[250] D.T. Banuti, The hybrid ergodic lattice gas model for critical fluids and the molecular nature of the critical point, *J. Supercrit. Fluids*. **183** (2022) 105505. 10.1016/j.supflu.2021.105505.

[251] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: *Proc. 2nd Int. Conf. Knowl. Discovery Data Min.*, AAAI Press, Portland, OR, (1996) 226.

[252] M. Meng, Z. Qiu, R. Zhong, Z. Liu, Y. Liu, P. Chen, Adsorption characteristics of supercritical CO_2/CH_4 on different types of coal and a machine learning approach, *Chem. Eng. J*. **368** (2019) 847. 10.1016/j.cej.2019.03.008.

[253] H. Wu, N. Lubbers, H.S. Viswanathan, R.M. Pollyea, A multi-dimensional parametric study of variability in multi-phase flow dynamics during geologic CO_2 sequestration accelerated with machine learning, *Appl. Energy*. **287** (2021) 116580. 10.1016/j.apenergy.2021.116580.

[254] B. Giroire, A. Garcia, S. Marre, T. Cardinal, C. Aymonier, Chemistry platform for the ultrafast continuous synthesis of high-quality III–V quantum dots, *Chem. Eur. J*. **27** (2021) 12965. 10.1002/chem.202101802.

[255] P.W. Dunne, C.L. Starkey, M. Gimeno-Fabra, E.H. Lester, The rapid size- and shape-controlled continuous hydrothermal synthesis of metal sulphide nanomaterials, *Nanoscale*. **6** (2014) 2406. 10.1039/C3NR05749F.

[256] N. Munyebvu, E. Lane, E. Grisan, P. D. Howes, Accelerating colloidal quantum dot innovation with algorithms and automation, *Mater. Adv*. **3** (2022) 6950. 10.1039/D2MA00468B.

[257] A. Yoko, T. Aida, N. Aoki, D. Hojo, M. Koshimizu, S. Ohara, G. Seong, S. Takami, T. Togashi, T. Tomai, T. Tsukada, T. Adschiri, Application 53 - Supercritical Hydrothermal Synthesis of Nanoparticles, in: M. Naito, T. Yokoyama, K. Hosokawa, K. Nogi (Eds.), *Nanoparticle Technology Handbook*, 3rd Ed., Elsevier, (2018) 683. 10.1016/B978-0-444-64110-6.00060-3.

[258] G. Philippot, E.D. Boejesen, C. Elissalde, M. Maglione, C. Aymonier, B.B. Iversen, Insights into $BaTi_{1-y}Zr_yO_3$ ($0 \le y \le 1$) synthesis under supercritical fluid conditions, *Chem. Mater*. **28** (2016) 3391. 10.1021/acs.chemmater.6b00635.

[259] E. Lester, P. Blood, J. Denyer, D. Giddings, B. Azzopardi, M. Poliakoff, Reaction engineering: The supercritical water hydrothermal synthesis of nano-particles, *J. Supercrit. Fluids*. **37** (2006) 209. 10.1016/j.supflu.2005.08.011.

[260] T. Adschiri, Y. Hakuta, K. Arai, Hydrothermal synthesis of metal oxide fine particles at supercritical conditions, *Ind. Eng. Chem. Res*. **39** (2000) 4901. 10.1021/ie0003279.

[261] J. Zhang, S. Ohara, M. Umetsu, T. Naka, Y. Hatakeyama, T. Adschiri, Colloidal ceria nanocrystals: A tailor-made crystal morphology in supercritical water, *Adv. Mater*. **19** (2007) 203. 10.1002/adma.200600964.

[262] T. Sasaki, S. Ohara, T. Naka, J. Vejpravova, V. Sechovsky, M. Umetsu, S. Takami, B. Jeyadevan, T. Adschiri, Continuous synthesis of fine $MgFe_2O_4$ nanoparticles by supercritical hydrothermal reaction, *J. Supercrit. Fluids*. **53** (2010) 92. 10.1016/j.supflu.2009.11.005.

[263] A. Auxéméry, G. Philippot, M.R. Suchomel, D. Testemale, C. Aymonier, Stabilization of tetragonal zirconia nanocrystallites using an original supercritical-based synthesis route, *Chem. Mater*. **32** (2020) 8169. 10.1021/acs.chemmater.0c01550.

[264] H.L. Hellstern, J. Becker, P. Hald, M. Bremholm, A. Mamakhel, B.B. Iversen, Development of a dual-stage continuous flow reactor for hydrothermal synthesis of hybrid nanoparticles, *Ind. Eng. Chem. Res*. **54** (2015) 8500. 10.1021/acs.iecr.5b02899.

[265] T. Gendrineau, S. Marre, M. Vaultier, M. Pucheault, C. Aymonier, Microfluidic synthesis of palladium nanocrystals assisted by supercritical CO_2: Tailored surface properties for applications in boron chemistry, *Angew. Chem. Int. Ed*. **51** (2012) 8525. 10.1002/anie.201203083.

[266] M. Bondesgaard, A. Mamakhel, J. Becker, H. Kasai, G. Philippot, M. Bremholm, B.B. Iversen, Supercritical flow synthesis of $Pt_{1-x}Ru_x$ nanoparticles: Comparative phase diagram study of nanostructure versus bulk, *Chem. Mater*. **29** (2017) 3265. 10.1021/acs.chemmater.7b00586.

[267] G. Aksomaityte, M. Poliakoff, E. Lester, The production and formulation of silver nanoparticles using continuous hydrothermal synthesis, *Chem. Eng. Sci*. **85** (2013) 2. 10.1016/j.ces.2012.05.035.

[268] T. Jaouhari, F. Zhang, T. Tassaing, S. Fery-Forgues, C. Aymonier, S. Marre, A. Erriguible, Process intensification for the synthesis of ultra-small organic nanoparticles with supercritical CO_2 in a microfluidic system, *Chem. Eng. J*. **397** (2020) 125333. 10.1016/j.cej.2020.125333.

[269] A. Cabañas, J. Li, P. Blood, T. Chudoba, W. Lojkowski, M. Poliakoff, E. Lester, Synthesis of nanoparticulate yttrium aluminum garnet in supercritical water–ethanol mixtures, *J. Supercrit. Fluids*. **40** (2007) 284. 10.1016/j.supflu.2006.06.006.

[270] Y. Hakuta, T. Haganuma, K. Sue, T. Adschiri, K. Arai, Continuous production of phosphor YAG:Tb nanoparticles by hydrothermal synthesis in supercritical water, *Mater. Res. Bull.* **38** (2003) 1257. 10.1016/S0025-5408(03)00088-6.

[271] E. Glogic, M. Claverie, M. Jubayed, V. Musumeci, C. Carême, F. Martin, G. Sonnemann, C. Aymonier, Greening pathways for synthetic talc production based on the supercritical hydrothermal flow process, *ACS Sustainable Chem. Eng*. 9 (2021) 16597. 10.1021/acssuschemeng.1c05120.

[272] V. Musumeci, P.S. Camacho, K. Xu, P.J.M. Monteiro, J.S. Dolado, C. Aymonier, Sub- and supercritical hydrothermal route for the synthesis of xonotlite nanofibers for application to green concrete materials, *J. Supercrit. Fluids*. **184** (2022) 105583. 10.1016/j.supflu.2022.105583.

[273] M. Kløve, S. Sommer, B.B. Iversen, B. Hammer, W. Dononelli, Machine learning based approach for solving atomic structures of nanomaterials combining pair distribution functions with density functional theory, *Adv. Mater*. **35** (2023) 2208220. 10.1002/adma.202208220.

[274] E.H. Lester, Self optimizing supercritical flow reactors, European Meeting on Supercritical Fluids, Budapest, Hungary (2023). emsf2023.com/ (accessed May 23, 2023).

[275] *Promethean Particles*, Promethean Particles. (2023). prometheanparticles.co.uk/ (accessed May 23, 2023).