



HAL
open science

Assessment of Machine Learning algorithms for predicting autoignition and ignition delay time in microscale supercritical water oxidation process

Deewakar Sharma, Carole Lecoutre-Chabot, Fabien Palencia, Olivier Nguyen, Arnaud Erriguible, Samuel Marre

► To cite this version:

Deewakar Sharma, Carole Lecoutre-Chabot, Fabien Palencia, Olivier Nguyen, Arnaud Erriguible, et al.. Assessment of Machine Learning algorithms for predicting autoignition and ignition delay time in microscale supercritical water oxidation process. *Fuel*, 2023, 352, pp.129098. 10.1016/j.fuel.2023.129098 . hal-04153992

HAL Id: hal-04153992

<https://hal.science/hal-04153992>

Submitted on 6 Jul 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 **Assessment of Machine Learning algorithms for predicting autoignition and ignition delay**
2 **time in microscale supercritical water oxidation process**

3 Deewakar Sharma^a, Carole Lecoutre^a, Fabien Palencia^a, Olivier Nguyen^a, Arnaud Erriguible^{a,b}, Samuel
4 Marre^{a*}

5

6 ^a Univ. Bordeaux, CNRS, Bordeaux INP, ICMCB – UMR5026, F-33600 Pessac, France.

7 ^b Univ. Bordeaux, CNRS, Bordeaux INP, I2M – UMR5295, F-33600 Pessac, France.

8

9 **Abstract**

10 With recent advancements in space technology, there is a need to develop technologies to ensure a
11 sustainable environment for human survival. Among these, treatment of human and organic waste aboard
12 manned space missions is a challenging task for which supercritical water oxidation using hydrothermal
13 flames has been proposed as a possible solution. The critical step in readily adopting this technology from
14 established ground-based setups is scaling the process to microscale. In addition to the challenge of
15 physical realization of the microreactors at these high pressure and temperature ($P > 22 \text{ MPa}, T >$
16 350 C) conditions, the need to explicitly analyze the process dynamics at microscale is inevitable owed to
17 the size of the reactors under consideration, the physics being significantly different from meso/mini scale
18 systems. One of the primary objectives is to identify the operating physical parameters for which
19 formation of hydrothermal flames can be obtained. Before proceeding with an expensive computational
20 or experimental approach to determine the exact ignition map, an initial estimate based on physical
21 arguments can help in providing insights into the process. We address this problem using homogeneous
22 ignition calculations to develop machine learning models to predict autoignition as well as ignition delay
23 time. The ingenuity of the work lies in defining autoignition criteria in relation to flow time scales expected

24 at microscale. Various classification models were trained and tested for predicting autoignition and
25 regression models were demonstrated to predict the ignition delay time. While predicting autoignition is
26 a straightforward process, a two-step approach is proposed for ignition delay time. Finally, how machine
27 learning can be used more explicitly, particularly for understanding and designing efficient microreactors,
28 is presented which highlights that machine learning approach is not merely restricted to prediction but
29 can also have real implications on improving the process as a whole.

30 *Keywords: Supercritical water oxidation, hydrothermal flames, machine learning, flame ignition,*
31 *microscale.*

32 **1. Introduction**

33 Over the recent years, there has been an unprecedented growth in the space industry and it is envisioned
34 to have deep space missions involving humans in the near future. It is thus inevitable to develop
35 technologies which can aid in sustaining life aboard these missions. Among these, treatment of organic
36 and human waste and converting them back to reusable resources is identified as one of the essential
37 technologies. The generated by-products can be reused in some form thereby reducing the burden of
38 carrying additional resources during space launches. Tapping on the potential of supercritical water
39 oxidation (SCWO) with hydrothermal flames, this process has been proposed to meet the aforementioned
40 technological requirements [1, 2]. Hydrothermal flames are the flames which exist in supercritical (near
41 critical) water due to autoignition of organic matter. The phenomenon is attributed to reduction in
42 autoignition temperature of organic matter, which also acts as a fuel, in supercritical water conditions (
43 $P > 22.1 \text{ MPa}, T > 374 \text{ C}$). While water is polar at ambient conditions, it becomes non-polar at
44 supercritical conditions. This facilitates dissolution of several non-polar gases such as oxygen, nitrogen, as
45 well as organic matter. With both organic matter and oxygen dissolved in supercritical water (SCW), it
46 provides a uniform medium for the oxidation of organic matter. In these conditions, when the

47 concentration of organic matter (fuel) exceeds a certain limit, *i. e.* threshold concentration at a given
48 pressure and temperature, the oxidation reaction leads to autoignition and thus the formation of
49 hydrothermal flames. The motivation to use hydrothermal flames for SCWO are two folds. Firstly, it
50 permits the injection of reactants at lower temperature than the critical point of water which circumvents
51 the problems of clogging in the inlet section of the reactor – one of the major drawbacks of SCWO [3].
52 Secondly, a higher heat generated due to hydrothermal flames promote a faster reaction and
53 decomposition of waste, recalcitrant molecules, etc. This process has been successfully demonstrated for
54 ground-based applications [4-7]. An important constraint in readily adapting the designs and
55 understanding of hydrothermal flames from ground-based applications to the space sector is the size of
56 reactors under consideration. Most of the reactors for ground applications are at meso/mini scale with
57 the volume of reactor being of the order of a few liters. However, with volume and weight being a
58 limitation in space applications, it is inevitable to have a device with similar functionalities but which is
59 compact and light in weight. In pursuit of advancing the technology of SCWO using hydrothermal flames
60 for space applications, we have proposed to carry out this process at microscale [8]. We define this process
61 as $\mu SCWO - H$. The feasibility to realize such microreactors (made of sapphire) which can withstand such
62 harsh pressure and temperature conditions along with being chemically compatible with SCW has already
63 been demonstrated [8]. This was made possible due to patented developed technology to etch sapphire
64 [9]. In addition to fulfilling the criteria of compact design, working at microscale augments the SCWO
65 process capabilities by tapping the advantages of a classical microfluidic system, such as further
66 reductions in reaction time, more uniformity in the reactions, etc.

67 The need to develop this technology, as evident from the above discussion, necessitates an explicit study
68 of hydrothermal flames at microscale owing to differences in the physics with respect to meso/mini scale
69 reactors. One of the prime reasons can be ascribed to a high surface area to volume ratio in a typical
70 microsystem resulting in a higher heat transfer, which can eventually render a different behavior to

71 hydrothermal flames when compared to meso/mini scale systems. Though, it can be argued that for utility
72 in space, the absence of any gravity can result in negligible heat loss due to natural convection, analysis
73 on the ground being the first step before moving to space applications necessitates such an investigation.
74 The second difference arises due to the small reactor volume which implies a large amount of heat being
75 generated in a smaller region. Consequently, the effect of heat constrained within the system will
76 significantly impact the flame dynamics. Thirdly, owing to low flow rates encountered at microscale, the
77 operating condition is primarily expected to be in laminar regime as compared to the majority of the
78 pertinent literature operating in turbulent regime for hydrothermal flames, for example in [4, 6].

79 In order to develop this technology, it is vital to identify operating regimes (flow rates, fuel & oxidizer
80 concentration, operating temperature, pressure) for which autoignition and thus, the formation of
81 hydrothermal flames can be attained. While development of an exact ignition map can be achieved from
82 experiments or high-fidelity numerical simulations, understanding autoignition conditions based on
83 homogeneous 0D calculations can aid in narrowing down the operational window to search upon the
84 optimum conditions. Homogeneous ignition analysis involves solving the reaction kinetics in time
85 assuming a zero-dimensional reactor using the reaction scheme for a given fuel under consideration. The
86 conventional utility of this approach is limited, such as predicting autoignition parameters (ignition delay,
87 maximum heat release etc.) for given set of operating conditions. However, it holds enormous potential
88 in the context of the present problem, which could be tapped by using data-driven models. For instance,
89 upon successful training the machine learning models, these can be used to predict the desired output
90 parameters for new operating conditions. Further, direct mapping of field parameters to ignition related
91 parameters can help to better understand the formation of ignition kernel, flame dynamics, design of
92 reactors, etc.

93 Machine learning over the recent years has garnered significant attention in diverse fields of physical
94 systems – fluids, thermal, and combustion. Brunton *et al.* [10] have presented an overview on how various

95 machine learning methodologies can find their applicability in the various areas of fluid mechanics,
96 addressing a wide range of problems, such as modeling and controlling fluid flows. The authors described
97 how optimization performance as well as convergence time can be improved by posing several objectives
98 in fluid mechanics as optimization and regression problems. Brenner *et al.* [11] and Brunton [12] in their
99 recent articles have further provided insights into advancing fluid mechanics using machine learning.
100 Machine learning models have also been used to predict thermodynamic properties of pure fluids and
101 mixtures, equilibrium compositions in liquid-vapor flash calculations, heat transfer coefficients, etc. [13-
102 18]. In the context of combustion, Zheng *et al.* [19] reviewed the recent progress and applicability of
103 machine learning in combustion studies. The authors highlighted several key pertinent sub-domains
104 where machine learning could have significant impact, such as detection of combustion oscillations due
105 to thermoacoustic, investigating physiochemical properties of fuels and subsequently designing a new
106 generation of fuels, and reconstruction of cellular surface of gaseous detonation, to name a few. A more
107 comprehensive review has been recently presented by Zhou *et al.* [20]. The authors highlighted
108 developing relations between input parameters (temperature, pressure, species etc.) and output
109 parameters of interest (chemical kinetics etc.) for combustion simulations using various machine learning
110 models. Here again, feasibility of fuel design, predicting physical properties of fuel (such as density,
111 heating value), their ignition delay time, flash point, etc. using neural networks was presented. In addition,
112 the authors addressed the challenges of combining machine learning with combustion research and aptly
113 mentioned that one of the key challenge lies in judiciously framing the physical problem in the framework
114 of machine learning. This not only requires in depth understanding of the machine learning aspects, such
115 as data selection and pre-processing, appropriate choice of machine learning models, tuning the models,
116 etc., but most importantly, an in-depth understanding of the physical problem. In the context of the
117 current problem, relevant work in auto-ignition is scarce and only a few of them can be found in literature.
118 Pan *et al.* [21] used *Support Vector Machine* model to predict autoignition temperature of organic fuels

119 based only on their molecular structure. The relationship was developed between autoignition
120 temperature in terms of descriptors which were selected using a genetic algorithm. This algorithm is
121 known to mimic natural selection evolutionary processes primarily, selection, gene crossover, and
122 mutation to obtain global optima solutions. The first step comprises of initially creating a population,
123 which is a set of individuals with each individual representing the solution that is to be solved. In order to
124 decide which individuals will be selected from the population, a fitness function is defined to evaluate the
125 fitness score. Using the fitness score, two pairs of individuals are selected with individuals having a higher
126 fitness score are more likely to be selected. Subsequently, offspring's are created from genes (here gene
127 represents variables on which individuals are dependent) after selecting a crossover point. The offsprings
128 are continually added to the population and the algorithm terminates when convergence is attained, *i. e.*
129 when no new offspring significantly different from existing ones can be formed. A similar work has been
130 reported by Suleiman *et al.* [22] where using molecular weight along with the number of carbon,
131 hydrogen, and oxygen atoms as parameters, the authors were able to predict autoignition temperature
132 of organic energetic compounds using hybrid support vector regression. More recently, Shah *et al.* [23]
133 used *Random Forest* and deep learning algorithms to predict autoignition and flame properties (flame
134 speed, fuel octane ratings, ignition delay time) in multicomponent fuels. Both the methodologies were
135 found to have high predictability even using a small data set. Cui *et al.* [24] developed models for ignition
136 delay using Back propagation neural network for n-butane/hydrogen mixtures. In addition, genetic
137 algorithm optimized back propagation model was also developed and performance of both were
138 compared over a wide range of operating conditions (pressures varying from 20 –25 bar, temperature
139 varying from 722 - 987 K, equivalence ratio from 0.5 - 15) and the performance metric (average correlation
140 coefficient) in case of the latter was found to be superior. Lehn *et al.* [25] used neural networks to predict
141 the laminar burning velocity using pressure, temperature, fuel/air ratio, and twelve molecular groups as
142 the input variables. Using sensitivity analysis to analyze the impact of functional groups, the laminar

143 burning velocity was found to increase with unsaturation, which was coherent with observations in
144 literature. Cui *et al.* [26] used back propagation neural network to predict the ignition delay time in
145 surrogate fuels where data for training and validation was generated using 0-D ignition calculations. An
146 interesting result reported was comparison of computational time using 0D simulation and trained back
147 propagation neural network. While the former took 28 s for a single calculation, 176 cases were evaluated
148 in 3.2 s using the latter highlighting significant gain using neural networks. More recently, Huang *et al.*
149 [27] studied the impact of hydrogen addition on the ignition delay time for aviation fuel using artificial
150 neural network (ANN). It was found that a large local relative error occurs when the ignition delay time
151 was very small. In order to improve the performance, a sub-ANN model was developed by training on the
152 subset of data points corresponding to the poor predictions from the initial ANN. Thus, if the predicted
153 ignition delay time from the initial ANN was below the selected threshold of $10^3 \mu\text{s}$, the sub-ANN was
154 used to update the output value.

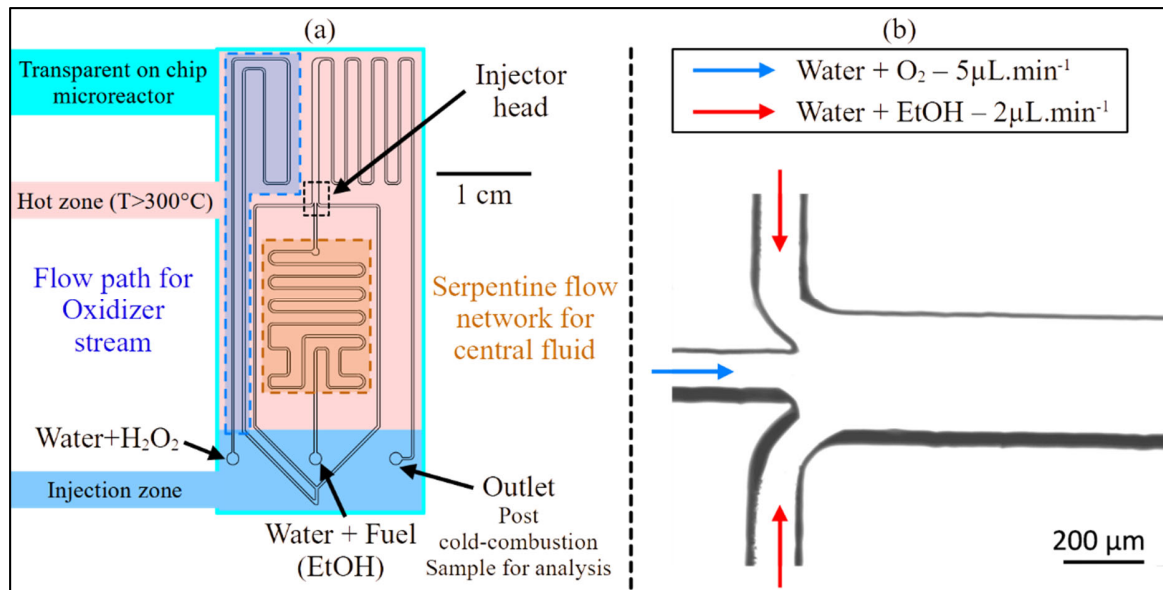
155 In this work, we demonstrate the use of machine learning models to predict the autoignition
156 characteristics of ethanol as a model fuel for the formation of hydrothermal flames at microscale. Owing
157 to microscale dimensions under considerations, the ingenuity of the current work lies in defining
158 autoignition by accounting for residence (flow) time and subsequently developing machine learning
159 models to predict the output parameters – possibility of autoignition and ignition delay time.
160 Furthermore, we also highlight how these data driven models can be used to predict zone of formation of
161 the ignition kernel from real physical data, such as CFD simulations. This can eventually be used for
162 understanding the dynamics of hydrothermal flames as well as designing better microreactors. The prime
163 focus of the article is thus to exhibit how by using a simple homogeneous ignition calculation in
164 conjunction with machine learning models, initial estimate of autoignition conditions and corresponding
165 ignition delay time can be predicted for microscale SCWO process. We begin by describing the physical
166 problem and means of data generation. Subsequently, various machine learning algorithms are applied

167 on the data to predict the autoignition condition as well as ignition delay time. Finally, we illustrate how
168 this approach can have physical implications for better understanding of the process dynamics for
169 development of this technology.

170 **2. Problem description**

171 The physical process to be investigated was described in the previous section. In order to provide readers
172 with more insights into the microfluidic process of SCWO, a schematic of the microreactor design
173 proposed for the aforementioned application is shown in Figure 1 (a).

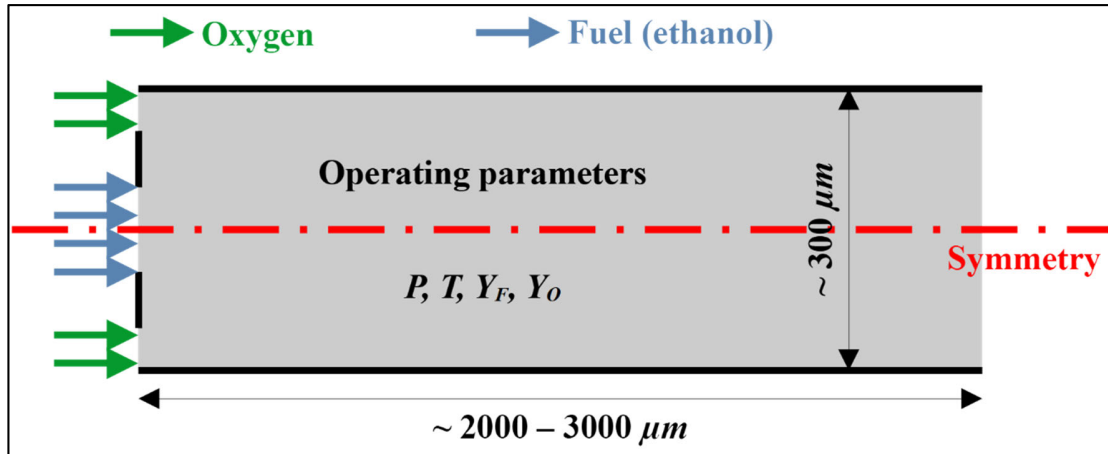
174 The microreactor consists of two inlets feeding the system with oxidizer ($\text{H}_2\text{O} + \text{H}_2\text{O}_2$) and fuel ($\text{H}_2\text{O} +$
175 ethanol). Two serpentine microchannels help preheating the two fluids (and ensure for the oxidizing fluid
176 the decomposition of H_2O_2 in H_2O and O_2) before they are put in contact in the injector. While details have
177 been presented elsewhere [8], we present an example illustrating phenomenon of phase-change of water
178 at high pressure and temperature (Figure 1(b)) conditions supporting the feasibility to realize such
179 extreme conditions at microscale. For the current problem, the focus lies in the injection zone where fuel
180 and oxidizer streams mix with each other. The schematic of this current physical process under
181 consideration (injection zone) can thus be described as shown in Figure 2 and the long-term objective is
182 to obtain a sustainable hydrothermal flame in the presented microreactor.



183

184 Figure 1: (a) Schematic of the microreactor system proposed to be used for $\mu SCWO - H$. (b) Microscopy image of
 185 the injector head microfabricated in sapphire.

186 The elementary design considerations as well as experimental observations [8] provided insights into how
 187 the considered physical dimensions (microscale reactor) could implicitly impact the phenomenon of
 188 autoignition in the current system and thus finds relevance in the context of the present problem. This
 189 can be explained as follows. Upon injection of the fuel (ethanol in the present case) and oxidizer into the
 190 reactor (microchannel), the species mix with each other resulting in a chemical reaction.



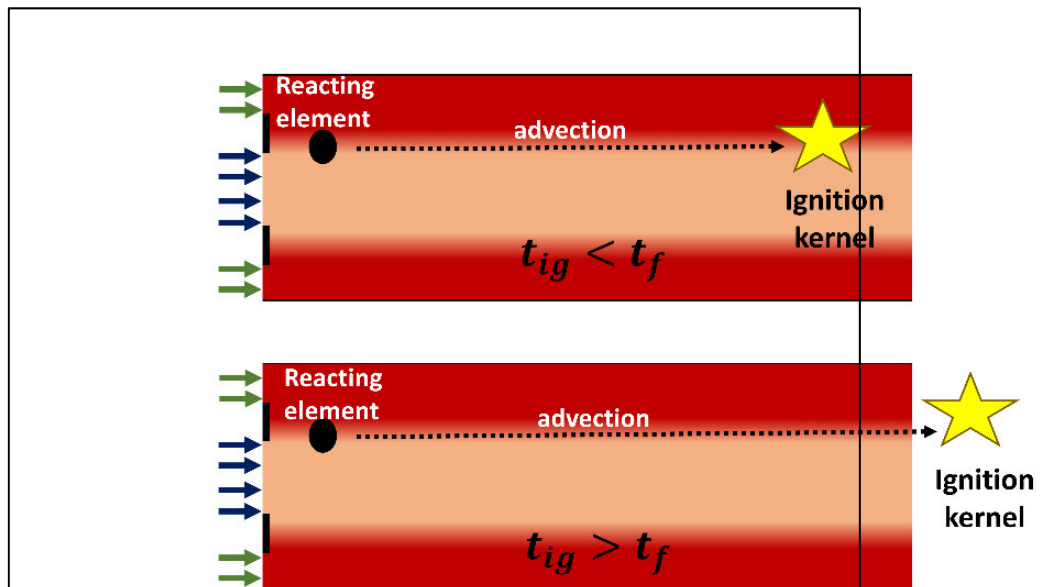
191

192 Figure 2: Schematic of microscale reactor for μ SCWO – H.

193

194 Even though the reaction between fuel and oxidizer is omnipresent in these operating conditions,
 195 autoignition will only occur under certain conditions governed by the coupled phenomena,
 196 hydrodynamics, thermodynamics and chemical kinetics of the system. An important characteristic of the
 197 autoignition is the ignition delay, which is defined as the time taken by a reacting mixture of a given
 198 composition to auto-ignite. The ignition delay can thus be interpreted as the time when the first ignition
 199 kernel may be expected to form. This ignition delay in the current problem can be related to the constraint
 200 on the permissible autoignition conditions due to the scale of the reactor under consideration. It is evident
 201 that in order to have hydrothermal flame at microscale, the ignition kernel needs to be formed within the
 202 reactor channel. This implies that the ignition delay for a given mixture composition at a given pressure
 203 and temperature should be less than the time taken by this reacting mixture element to flow through the
 204 reactor microchannel. Failure to meet this criterion implies that even though the mixture is ignitable, it
 205 will not serve the purpose for the given physical system under investigation. This is illustrated in Figure 3
 206 where we define the flow time as $t_f = L/u$ with L being the length of the channel and u the axial velocity.

207 The above explanation thereby substantiates presenting the schematic of the microreactor with physical
208 dimensions in the present homogeneous reaction/0D reactor study.



209

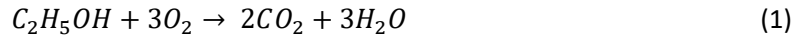
210 Figure 3: Schematic illustration of how ignition delay (t_{ig}) being larger than flow residence time (t_f) implies failure
211 of autoignition in $\mu SCWO - H$ application.

212 Thus, despite considering homogeneous calculation in the present context, reference to the physical
213 system and hydrodynamics is self-explanatory as it will govern the criterion for deciding whether or not
214 we have autoignition. This forms one of the major differences in identifying autoignition conditions at
215 microscale in comparison to meso/mini scale reactors. Before proceeding further, it is worth mentioning
216 that in a real scenario where hydrodynamics will be considered, the ignition limits (which will be presented
217 below) will be modified owing to several local dynamics playing their role. However, the utility of the
218 current approach lies in providing an initial estimate on the limits along with physical implications it can
219 have as discussed in §5.

220 3. Methodology

221 3.1 Data generation

222 In order to develop a machine learning model, having reliable and sufficient data is of prime importance.
 223 In the present work, data is generated using homogeneous ignition calculations based on a single step
 224 reaction for oxidation of ethanol in SCW conditions as given in [8]:



225 with reaction rate evaluated as,

$$Q_{EtOH} = -10^{17.23} \exp\left(\frac{-E_a}{RT}\right) [EtOH]^{1.34} [O_2]^{0.55} \quad (2)$$

226

227 The activation energy in the above expression is, $E_a = 214 \frac{kJ}{mol}$ [28]. The governing equations of
 228 conservation of mass, species, and energy were solved in time using a simple first order Euler scheme. As
 229 the physical system under consideration is at microscale, the relative change in pressure is expected to
 230 be very small when compared to operating pressure which is greater than $22.1 MPa$. Thus, homogeneous
 231 calculations in the current work correspond to a constant pressure reactor system with density varying in
 232 the system. Appendix 1 in the supplementary information presents the governing equations used for data
 233 generation. The parameters (independent variables) considered were the mass fraction of ethanol (Y_f),
 234 mass fraction of oxygen (Y_o), the temperature (T), and the pressure (P). Table 1 presents the range of
 235 each parameter considered in the present work. The upper limit for fuel and oxygen concentration arises
 236 due to the validity of reaction rate up to these limits as mentioned in [8].

237

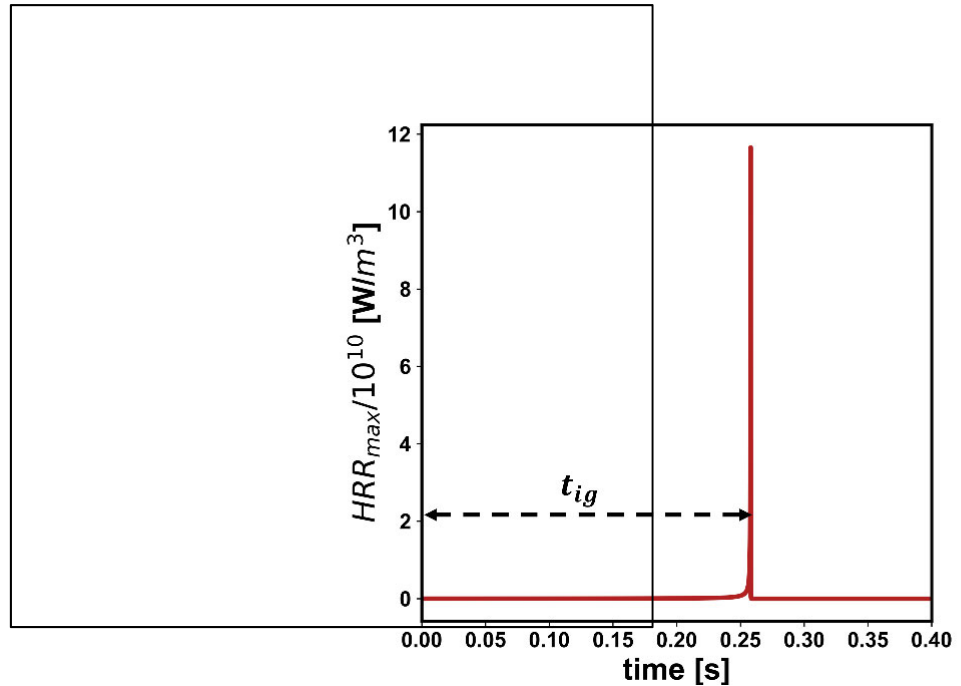
Table 1: Range of input parameters considered

Parameter	Range
Pressure (P)	225 – 250 <i>bar</i>
Temperature (T)	350 – 450 [C]
Fuel (ethanol) (Y_F)	0.5 – 4.5 [<i>in percent</i>]

Oxidizer (Y_o) 0.5 – 9.5 [*in percent*]

238

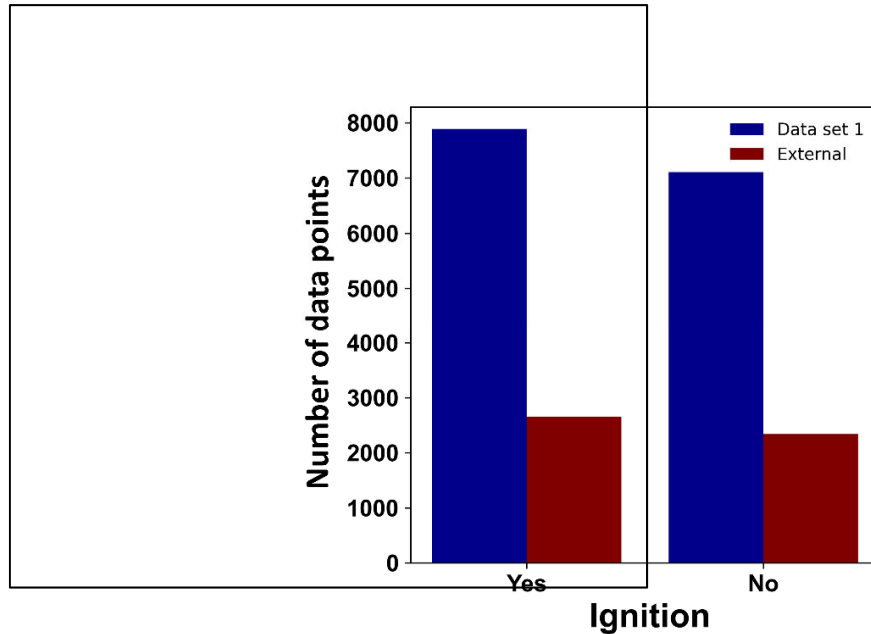
239 Each operating point, denoted by $[Y_{f,i}, Y_{o,i}, T_i, P_i]$ for i^{th} data point, was randomly generated using
240 Python. The data set consists of 15000 data points. The entire data set, termed as data set 1, comprised
241 15000 data points and was used for training, validation, and testing various models. It is intuitive to expect
242 that once the models are trained, these can further be used to predict the output (autoignition or not,
243 ignition delay time) for new conditions not included in this data set, such as those coming from
244 experiments. This could further aid in testing the trained machine learning models. However, in the
245 absence any such data at present, another data set comprising 5000 data points was created which can
246 be considered as an example of data from experiments. We term this data set as external data set and
247 this is used only for testing the trained models. The objective of doing this is to further test the robustness
248 of the trained models. For each data point, the time taken for autoignition, *i. e.*, ignition delay time, was
249 evaluated as the time when a sudden jump in the heat release rate is observed as shown for one case in
250 Figure 4.



251

252 Figure 4: Representative case illustrating how ignition delay time was defined. The plot is for $P = 250 \text{ bar}$, $T =$
 253 400 C , $Y_f = 2 \%$, $Y_o = 9.5 \%$

254 The output or dependent variables of interest are whether we have autoignition or not (I_g), and if yes,
 255 what is the ignition delay time (t_g). While the former is a categorical variable, the latter represents a
 256 continuous output variable. As per previous discussion, categorizing I_g will depend on flow or
 257 hydrodynamic time scales under consideration. Based on preliminary investigation on design aspects as
 258 presented in [8], 1s is chosen as the upper limit in the present case. This corresponds to flow rates in the
 259 range $5 - 100 \mu\text{l}/\text{min}$ for channel dimensions of $\sim 300 \mu\text{m}$ width and $\sim 20 \mu\text{m}$ etch depth and length
 260 $\sim 2500 \mu\text{m}$. The time limit presented is thus the maximum tentative flow time in these conditions which
 261 nevertheless can be adapted for different flow conditions in the future. Based on this criterion, the two
 262 data sets comprised of 52% and 53.2% cases corresponding to autoignition as shown in Figure 5, which
 263 represents a well distributed data.



264

265 Figure 5: Bar graph illustrating number of data points in the data sets (data set 1 and external) which correspond to
 266 conditions of autoignition (blue bar) and which do not (dark red bar).

267

268 3.2 Data treatment & preprocessing

269 One of the initial steps before developing machine learning models is data preprocessing which includes
 270 several aspects such as checking for any duplicate entries. Both the data sets, data set 1 and external data
 271 set, were checked if any duplicate entries of operating conditions $[Y_f, Y_o, T, P]$ were present and none was
 272 found. In terms of the range of values of the individual parameters, there exists a large variation in their
 273 magnitudes. Here mass fractions, owing to their definition, lie between 0 and 1, while temperature
 274 (350 C – 450 C) and pressure (225 bar to 250 bar) are $O(10^2)$. Training the models directly on this
 275 data can result in model inaccuracies due to bias towards features with larger values. Several standard
 276 scaling methods, such as normalization (scaling between minimum and maximum value) and
 277 standardization (scaling such that data exhibits properties of Gaussian distribution with zero mean and
 278 unit variance), exist to address this issue. The primary objective of this step is to ensure that all the

279 features are of the same order. In the present work, however, we resort to scaling based on physical
280 aspects of the problem – operating conditions. It is evident from the aforementioned range of the
281 variables that scaling is required primarily for temperature and pressure. As in supercritical water
282 oxidation, the critical point of water ($T_{c,w} = 374\text{ C}$) serves as a landmark to define the process, we scale
283 the feature temperature (T) as $T = (T - T_{c,w})/100$. With temperatures varying from $350\text{ C} - 450\text{ C}$,
284 this results in T being the same order as mass fractions. Similarly, pressure, which in the current scenario
285 varies from $22.5 - 25\text{ MPa}$ ($225\text{ bar} - 250\text{ bar}$), was scaled by 100. These scaling ensures that all the
286 variables (features) are now same $\sim O(1)$ in the current problem. The motivation to opt for the proposed
287 scaling is its simplicity in implementation. This will be particularly useful when using these models with
288 external data from physical analysis, such as from experiments or entire 2D/3D space from CFD
289 simulations, wherein the physical variables can be obtained in the proposed scaled manner and thus be
290 directly fed into the models for predicting autoignition and ignition delay time rendering ease of using
291 machine learning methodology for real process. For output parameters, I_g was set to 1 when we had
292 autoignition else it was assigned the value of 0. In addition to identifying whether or not we have
293 autoignition, predicting the ignition delay time is also a parameter of interest. One of the key challenges
294 is how to describe ignition delay time when the input operating parameters corresponded to no
295 autoignition condition. This is because, in a physical sense, no autoignition implies that despite having left
296 the reacting mixture element for considerable long duration, we do not observe any sudden increase in
297 HRR. Thus, a very large value could be used to define no autoignition. However, in the present context no
298 autoignition is identified when $t_{ig} > t_f$ and thus, in principle, any value greater than t_f could be assigned
299 to t_{ig} . One of the primary concerns with this approach is that many data points with this same value
300 could create a bias towards this assigned value. Thus, errors may creep in during training and subsequent
301 testing. In order to minimize this, we assign value 0 to these instances. It is to be mentioned that assigning
302 0 does not have any physical meaning and is just an indicator of no-autoignition event. It may be argued

303 that this may also create a bias towards lower values of ignition delay time, and this is what will be
304 explored in the subsequent section and an appropriate strategy to predict ignition delay time will be
305 discussed.

306 3.3 Machine learning models and performance criteria

307 The first part of the presented problem represents a classification problem wherein the objective lies to
308 predict whether we have autoignition. Various supervised machine learning models were trained to
309 predict the autoignition (I_g) for a given set of input variables. These include *logistic regression (Logistic)*,
310 *Decision Tree (DT)* [29], *Random Forest (RF)* [30], *Support Vector Machine Classifier (SVC)* [31], *k-Nearest*
311 *neighbors (kNN)* [32], and *AdaBoost classifier (Ada)*. The motivation to use different models was to analyze
312 which model could well capture the desired trend and be subsequently used for further analysis. The
313 models were implemented in Scikit-learn [33] in Python 3.9. The performance of each model is known to
314 depend on several parameters and values of these parameters need to be tuned to have their optimum
315 values, the process known as *hyperparameter tuning*. The objective of this process is primarily to ensure
316 that the model achieves a global minimum, which may otherwise be skipped when considering the default
317 values. These optimal parameters were obtained by using the *GridSearchCV* function in Python using an
318 8-fold cross validation. A similar methodology was adopted for predicting ignition delay time. Since
319 ignition delay time is a continuous variable, here regression models were used, namely *Ridge regression*
320 *[34]*, *Decision Tree regressor*, *Random Forest regressor*, *k-nearest neighbor regressor*, *Adaboost*, and
321 *Gradient boost regressor (GB)*. While details of each model can be found in the cited reference, we briefly
322 describe each model in simple terms.

- 323 • **Linear models:** Ridge and logistic models form a part of generalized linear models. Logistic
324 regression is a classification algorithm used to predict the probability of an event. It is one of the
325 easiest models to implement, train, and test. The model assumes there is no or very minimal

326 multicollinearity between the independent variables. The probability of an event (p) is converted
327 to odds (ratio of success and failure probability, $p/(1 - p)$) on which a logit transformation is
328 applied. This is known as log of odds and is described by $\log\left(\frac{p}{1-p}\right) = \beta_0 + \sum_{i=1}^k \beta_i X_i$. The
329 coefficients (β_i) are obtained by maximizing the log of likelihood. **Ridge regression** is a form of a
330 linear regression model where the model is penalized using sum of squared of the weights
331 (coefficients) in order to prevent overfitting. Here, the residual sum of squares can be evaluated
332 given by $\sum_{i=1}^N (y_n - \hat{y}_n)^2 + \lambda \sum_{i=1}^k \beta_i^2$, where $\hat{y}_n = \sum_{i=1}^k \beta_i X_i$ and λ is a parameter which governs
333 the severity of the penalization.

334 • **Tree based models:** *Decision Tree*, *Random Forest*, *AdaBoost*, and *GradientBoost* models fall
335 under the category of Tree Based models. In these models, a tree like structure is developed based
336 on certain conditions applied on the input variables and a decision is made to predict the output.
337 Thus, the prediction can be obtained using simple if-else conditions rendering these models quite
338 intuitive to interpret. These kinds of models can be used both for classification as well as
339 regression problems.

340 **Decision Tree** is the simplest model in this category. Starting with the base of the tree, also termed
341 as the root node, the data is split into branches leading to decision nodes. The terminal node
342 where a decision is made, are called the leaves of the tree. Usually, the tree is not allowed to grow
343 to its full depth to prevent overfitting and some parts of the tree are removed, the process being
344 termed as pruning.

345 **Random Forest** comprises of several decision trees operating collectively to predict the output.
346 The trees are developed by bootstrapping the data and each tree predicts an output.
347 Subsequently, the output with maximum number of votes is taken to be the model prediction.
348 Random Forest forms a part of ensemble models, *i. e.* the models where several models (trees in
349 this case) are generated and the output is governed collectively from these models. While

350 Random Forest operates in parallel, averaging out error over all trees, **AdaBoost** and
351 **GradientBoost** work sequentially to reduce the errors. In these models, the errors in the previous
352 models (trees) are corrected in the successive models. In *Adaboost*, only a single split is
353 permissible in each tree resulting in two leaf nodes and the trees are called stumps. Initially, equal
354 weight is assigned to each data set and the weights are adjusted while developing subsequent
355 stumps. This is attained by assigning a higher weight to incorrectly classified data point in order
356 to correct in subsequent classification. Unlike *Adaboost*, no limitation is posed on the number of
357 leaves and splits in *GradientBoost* algorithm. The peculiarity of latter method is that leaf nodes
358 predict residuals and with addition of tree, the residuals are reduced.

359 • **k-Nearest Neighbor** It is a form of non-parametric (*i. e.* it does not assume any form of mapping
360 function between input and output) and supervised learning classifier which uses proximity
361 between data set to classifications. “*k*” signifies how many neighbors will be used to predict the
362 output. The proximity is calculated in terms of distance metrics, the most common being the
363 Euclidean distance. These models are easy to implement and have very few parameters to hyper-
364 tune. However, it fails to perform well with the increase in dimensionality of the data.

365 • **Support Vector Machine (Classifier)** This method can be used both for classification and
366 regression analysis, the former being used in the current study. It sorts the data into respective
367 classes by defining margins / hyperplanes between the classes. In the case of linearly separable
368 data, the best decision boundary (hyperplane) is identified as the one which has the largest
369 distance from the classes. The points nearest to the hyperplane, termed as support vectors, are
370 used to evaluate the distance. In the case of non-linearly separable data, data is transformed to
371 higher dimensions using kernels, such as polynomial, radial basis, in order to define linear decision
372 boundary.

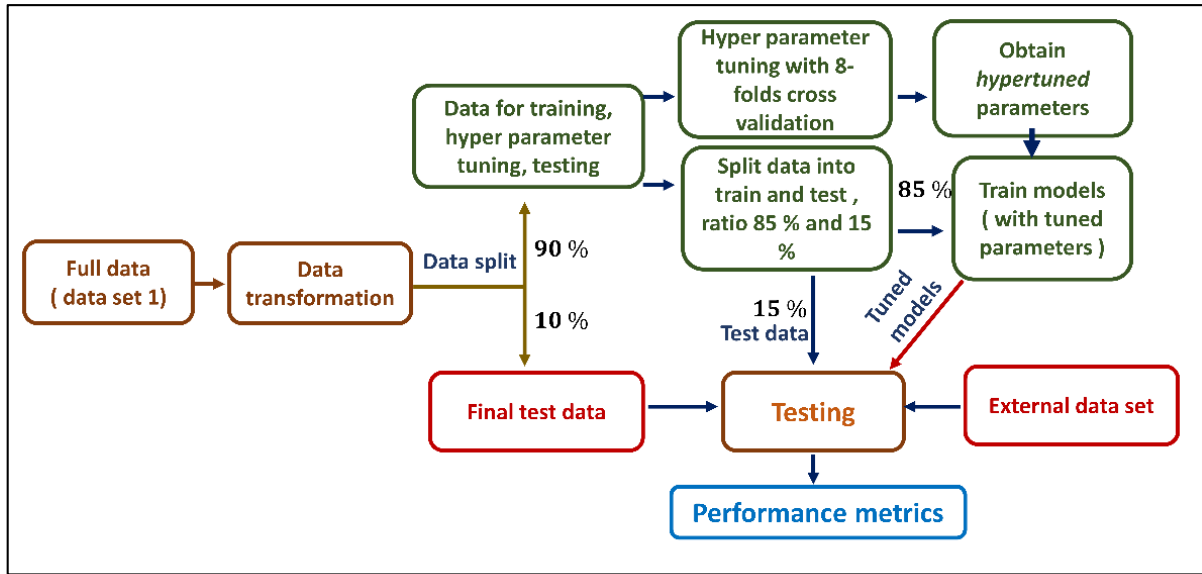
373 Several metrics exist to quantify the performance of machine learning models [35], such as the absolute
374 error (*MAE*), the mean squared error (*RMSE*), and the coefficient of determination (R^2) for regression
375 problems and accuracy, F1-score, etc., for classification problems. In the present case, performance of
376 classification models was evaluated in terms of accuracy, which is defined as the ratio of correct
377 predictions to total predictions while R^2 was opted for regression analysis and can be defined
378 mathematically by (where symbols have their usual meanings),

$$R^2 = 1 - \frac{\sum_{i=1}^N (\hat{y}_n - y_n)^2}{\sum_{i=1}^N (y_n - \bar{y}_n)^2} \quad (3)$$

379

380 3.4 Training, cross validation, and testing of models

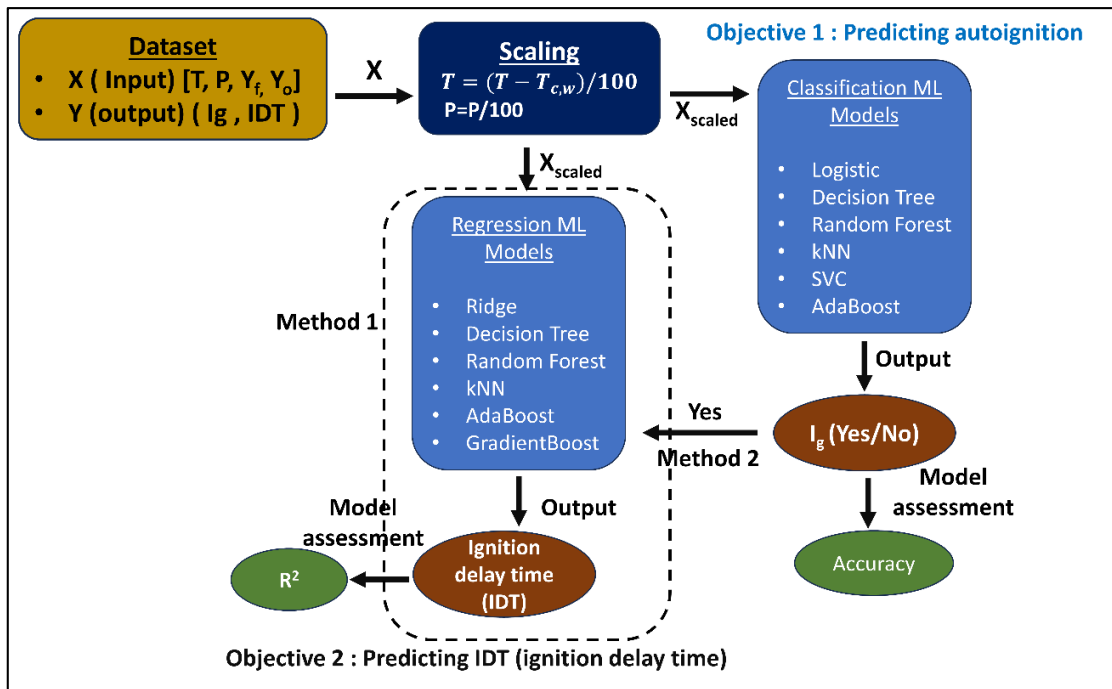
381 The various machine learning models were built on data set 1 by dividing it into training and testing
382 subsets in the ratio of 9: 1. Here, 10 % of the data was reserved for final testing (termed as final test data
383 henceforth) and none of these data points were used at any stage during the training/validation process
384 to make sure that performance metrics obtained for each model are fair by all means.



385

386 Figure 6: Schematic of procedure adopted in the current work for training, validation, and testing machine learning
 387 models.

388 The other 90 % of the data served two purposes. Firstly, it was used to obtain hyper-tuned parameters
 389 using *GridsearchCV* function with 8-folds cross-validation. Hypertuned parameters for all the models are
 390 presented in Appendix 2. Once the tuned parameters were obtained, this data set was split into and
 391 test in the ratio of 85 % and 15 % for training and testing, respectively. The latter is termed as test data
 392 henceforth. Subsequently, models were trained with best performing parameters and tested on this 15%
 393 data (test data). Finally, these models were tested on the initial 10 % data (final test data) in addition to
 394 the external data. A schematic of this procedure is illustrated in Figure 6.



395

396 Figure 7: Schematic of the flow process for predicting autoignition and ignition delay time from input variables. For
 397 predicting ignition delay time, two methods are proposed (see text below). Models are evaluated based on accuracy
 398 as well as R² for autoignition and ignition delay time prediction, respectively.

399

400 As mentioned previously, the two objectives of interests are to predict whether there is autoignition and
 401 ignition delay time. Figure 7 illustrates a schematic of these two objectives highlighting the inputs,
 402 predicted variables, and performance metrics. While training/testing the models for whether we have
 403 ignition or not is pretty straightforward, it is not so evident for ignition delay time. The complexity arises
 404 because in the entire data, there are cases which do not correspond to autoignition and for which the
 405 values have been set to zero. Here, two different approaches are possible. Firstly, models can be trained
 406 (and hyper-tuned) using the entire data set *i. e.* including zero as ignition time for no autoignition cases.
 407 This implies that predicting ignition time will be independent of whether we have ignition or not. In the
 408 second approach, the models for predicting ignition time can be trained only using the real ignition delay

409 time data, *i. e.* data points corresponding to autoignition cases. These methodologies lead to two different
410 possibilities of predicting the output variable on an entirely new data set (external data set). For the sake
411 of clarity, these are termed as method 1 (M1), where models are developed on the entire data set, and
412 method 2 (M2), where models are developed only using the actual ignition delay time. The advantage of
413 former lies in its simplicity to predict the output as in case of the latter, two models need to be developed
414 and used, one for predicting autoignition and second for predicting the ignition time where the outcome
415 of first model (classification) will have influence on the overall outcome of the second model (regression).
416 However, the first method could be prone to more error due to manual bias introduced. We thereby test
417 both these approaches and compare the performance metrics of the models, which are presented in the
418 next section.

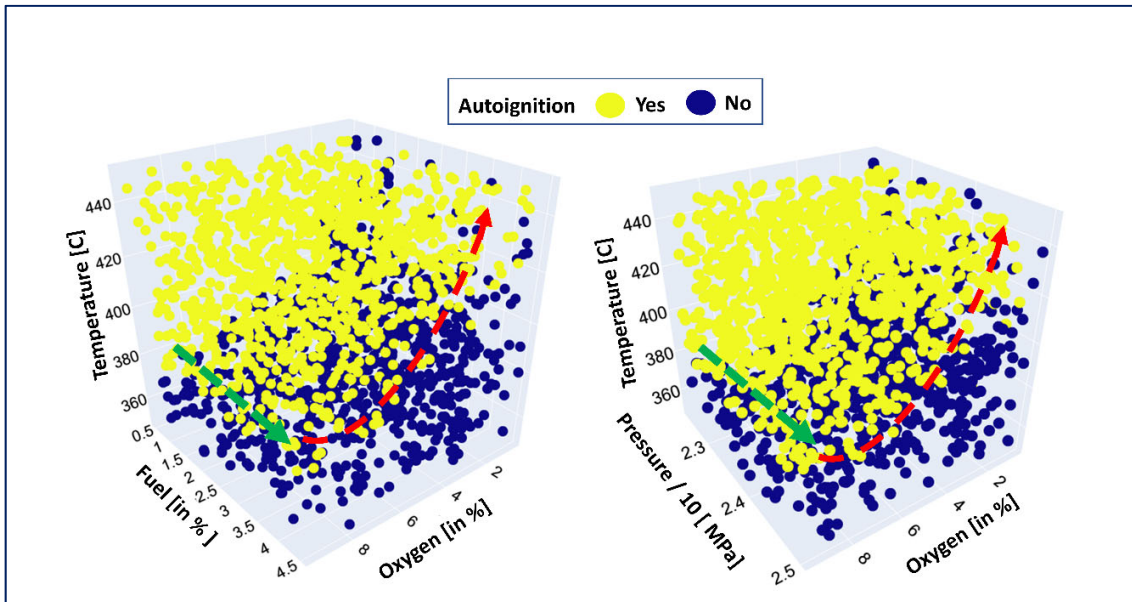
419 **4 Results and discussions**

420 In this section, the performance of various machine learning models is presented. However, before that,
421 we do some data analysis of the results from homogeneous ignition calculation to understand how the
422 data behaves and if the output yields meaningful results. This will further aid in ensuring that proposed
423 implications of using machine learning models can be successfully applied as described in §5.

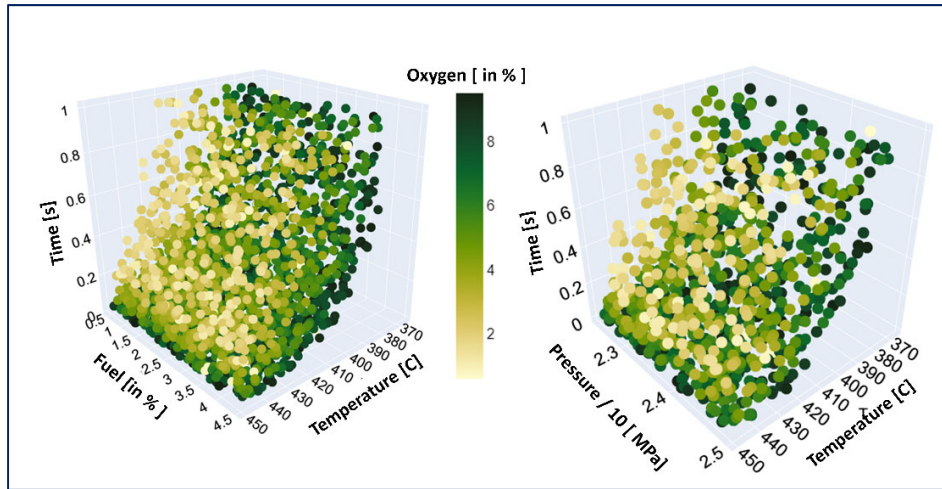
424 4.1. Data Analysis

425 We first try to seek the effect of various independent parameters on whether we have ignition or not. It
426 can be seen from Figure 8 that with increase in temperature, the likelihood of autoignition increases.
427 Here, for the sake of clarity, only 20% of the data points from data set 1 were plotted by selecting every
428 5th data point. The trend in Figure 8 can be explained as increase in temperature represents a higher heat
429 release and thus a faster reaction rate, which results in a smaller autoignition time. This is further
430 supported from the ignition delay plot as shown in Figure 9. Further, both the outputs seem to be affected
431 more by the oxidizer (oxygen) percentage as compared to fuel concentration while there is a very little

432 impact of pressure. This can be inferred from Figure 8 and Figure 9 where the trend for autoignition as
433 well as ignition delay remains nearly flat with fuel percentage while it exhibits a non-linear trend (nearly
434 exponential) with oxygen percentage.



435
436 Figure 8: Scatter plot illustrating dependence of various input parameters on output variable (I_g). Green and red
437 dashed lines illustrate the trend of ignition likelihood with fuel and oxidizer concentration, respectively (left) and
438 pressure and oxidizer concentration (right) with temperature.



439

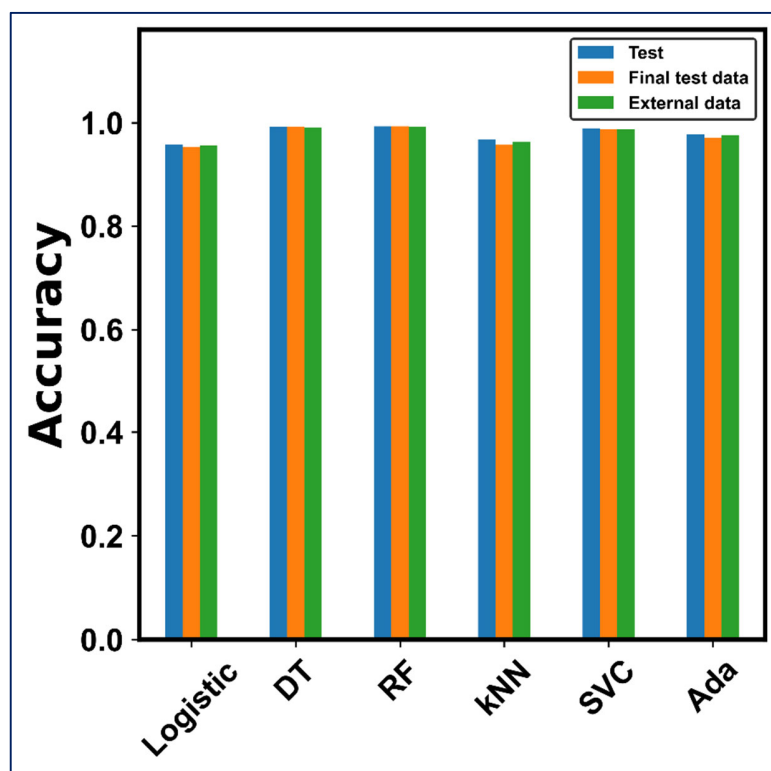
440 Figure 9: Ignition delay plot as function of various parameters.

441 From a process engineering perspective, this trend is thereby advantageous as we have the possibility of
 442 autoignition without much varying of the fuel concentration. This implies that we can nearly keep this
 443 parameter constant when attempting to seek autoignition conditions in real experiments and focus more
 444 on adapting the oxidizer concentration.

445 4.2 Performance of various machine learning models

446 Different models presented in the above section were first tested for the classification problem of
 447 predicting autoignition. Figure 10 illustrates the performance metrics of different models for predicting
 448 I_g . The metrics presented for each model correspond to three different test data sets as was explained in
 449 §3.4. In most of the cases, the performance metrics for all the three test cases remain nearly the same
 450 illustrating that the models were trained, have a small variance and were not overfitted. While the
 451 performance metrics of all the models seem to be quite satisfactory, *Random Forest* is able to make
 452 predictions most accurately. The performance is slightly better than *Decision Tree*, which as a similar
 453 approach, but *Random Forest* goes a step further to bootstrap random samples and eventually predicts
 454 the outcome for the tree with highest votes. Averaging over all the trees subsequently averages out the

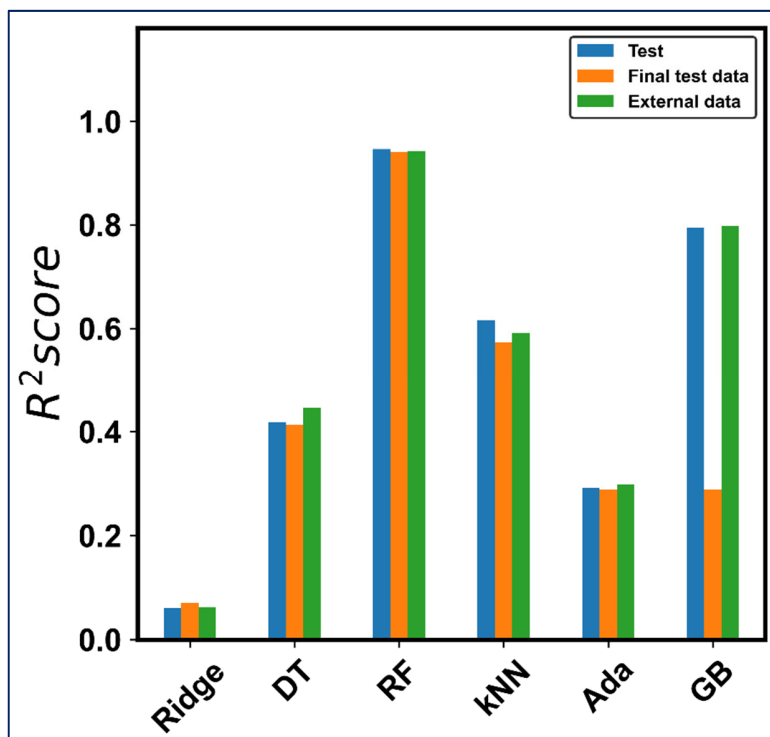
455 variance in each tree thereby optimizing the bias and variance of the model in this classification problem.
456 The performance of the support vector machine classifier (SVC) is also comparable with that of *Random*
457 *Forest*. This can be ascribed to very clear distinguishment between the classification variable, *i. e.* whether
458 there will be autoignition or not, and this model is known to perform quite well in such cases. Further,
459 this can also be attributed to no overlapping classes in the present case as each instance of input
460 parameter is assigned only a single class.



461
462 Figure 10: Accuracy of different models for the classification problem of identifying whether we have autoignition
463 or not. For abbreviations, refer to the table after the conclusion.

464 The second part of the problem is to predict the ignition delay time. As was explained previously, here
465 two different approaches are followed. We first present the results with method 1, *i. e.* where ignition
466 time is predicted independently of whether we have ignition or not. Figure 11 shows the performance
467 metrics of various models used in this case. Here, it can be seen that except *Random Forest* most of the

468 models perform quite poorly in predicting the ignition time. The objective to illustrate these not so
469 convincing results is to highlight that if not dealt with and applied appropriately, despite their enormous
470 potential, machine learning models can result in very poor outcomes. One possible reason in the current
471 case could arise due to training the models with exact numerical value and expecting the prediction to be
472 exact, which is very unlikely especially within the limits of numerical accuracy. One means to improve the
473 performance could be to classify the ignition delay time within certain ranges to make bins. This would
474 reduce the error as instead of predicting the exact value, we have a certain tolerance to the exact value.
475 This could also explain why *Random Forest* performs better as compared to other models. Nevertheless,
476 the poor performance of most of the models could primarily arise due to imposition of zero as ignition
477 time for no autoignition cases. These cases, which even in training set seem to have incurred manual bias,
478 further worsen the model performance owing to likelihood of predicted values being close to zero which
479 otherwise would resemble no autoignition condition.

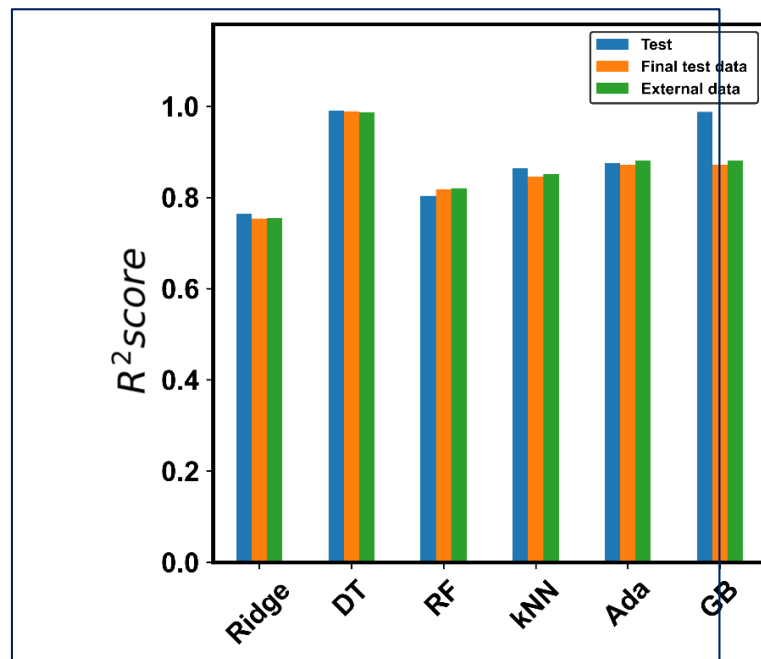


480

481 Figure 11: R^2 values for different models for predicting ignition delay using method 1 as described in the text.

482

483 In order to be coherent with physical reasoning that ignition delay is to be predicted only when
484 autoignition is predicted, we test method 2 as was described previously. The models in this approach are
485 trained using only the data points which correspond to autoignition and thus all the cases corresponding
486 to $I_g = 0$ are excluded. This results in 7891 data points from data set 1. The rest of the procedure to
487 define the training/validation, testing, and final test data is the same as in the previous case and explained
488 in Figure 6. The only difference was that the number of data points were reduced. Hyper tuning of the
489 parameters was performed as in the previous case using 8-fold cross-validation. Figure 12 shows
490 performance metrics of different models on various test data sets. Here, for testing the external data,
491 data points corresponding to only autoignition were used which comprised 2659 data points from the
492 total 5000.



493

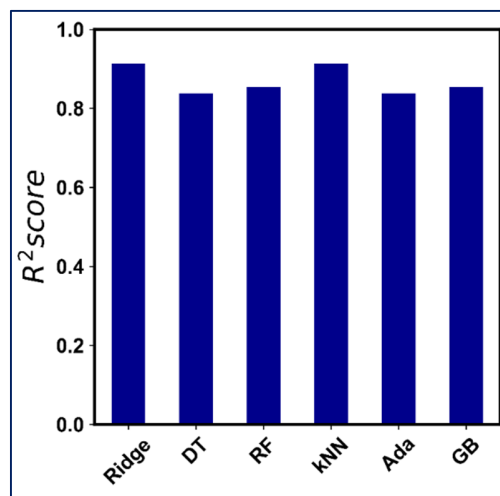
494 Figure 12: R^2 values for different models for predicting ignition delay using method 2 as described in the text.

495

496 Comparing the performance of various models using method 1 (where we used the entire data to predict
497 autoignition irrespective of autoignition and as shown in Figure 10), the performance of all the models is
498 significantly better using method 2 as can be seen in Figure 12 (*i. e.* predicting ignition delay time in a two-
499 step process, firstly predicting autoignition and if found to occur, only then predict the ignition delay time).
500 This shows that there was a substantial effect of inaccuracies that arose due to accounting for imposed
501 ignition delay time in no autoignition cases. It may be argued that using some other value instead of zero
502 could improve the performance using method 1. However, this may not make a large difference because
503 there will be a significant percentage of the population with the same (hypothetical) ignition time delay
504 which introduces manual bias in the data set itself. Therefore, in order to predict the ignition delay, the
505 second approach seems to be more reasonable, given the condition that autoignition is well known. Thus,
506 if the operating parameter is known to yield autoignition, we can calculate the ignition delay time quite
507 precisely with various regression models as shown in Figure 12. However, in such case, attention must be
508 paid as the error in the output would be a combined error due to two models, firstly due to predicting
509 autoignition and subsequently in ignition delay time.

510 As a final step, we tested this approach to check how well the machine learning models as presented so
511 far could find their utility in a real application. Here, the external data set serves as the test case while a
512 more realistic case is presented in the next section. The input data (external data) is first tested for
513 whether we have ignition or not. Subsequently, the data points for which autoignition is predicted are
514 passed as input to the models predicting the ignition delay time. Let us define this output as V_1 . For the
515 remaining data points, *i. e.* for which no autoignition was predicted, the corresponding output ignition
516 delay time was set to zero to be coherent with how non ignition is defined in input data. Let us define this
517 output as V_2 . It is to be noted here that zero as ignition time serves no purpose for model development
518 and is just a numerical value to complete the output vector. These two sets (V_1, V_2) are combined to
519 obtain the final predicted ignition delay time output (\hat{y}) and this is eventually compared with true values

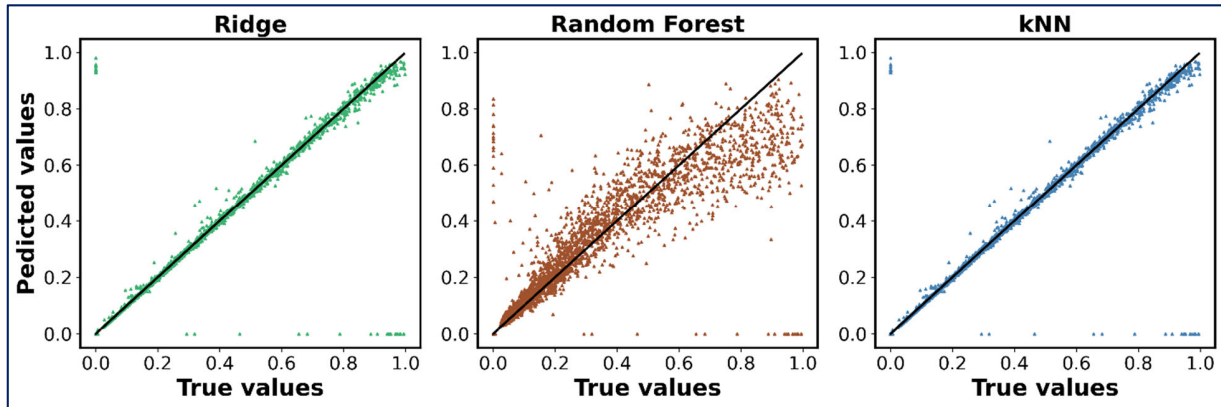
520 (y). As *Random Forest* had the best performance for predicting autoignition, this model was used in the
521 initial step of predicting for autoignition. Subsequently, all the aforementioned models were tested, and
522 their performance metrics are shown in Figure 13. It can be seen that there is a significant improvement
523 over the model performance as compared to method 1. We further present a comparison between actual
524 values and predicted ones for three different models. This is shown in Figure 14 where in an ideal case all
525 the points should like on $y = x$ straight line.



526

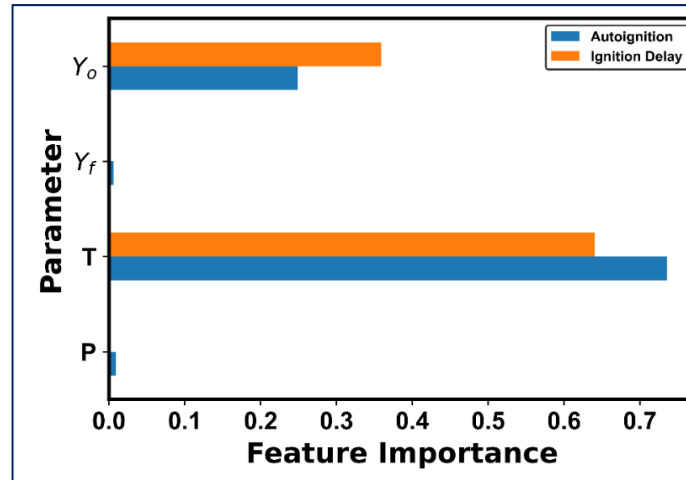
527 Figure 13: R² values for different models for predicting ignition delay for external data set.

528 This thereby presents that a two-model sequential approach needs to be followed to predict the desired
529 output parameters with satisfactory accuracy.



530
531 Figure 14: Comparison between actual and predicted values of ignition delay on external data set using methods as
532 described in the text using three different models as mentioned therein.

533 As a final step, we finally analyzed the importance of various features on the model outputs. While this
534 was briefly presented in the above section based on graphical visualization, here we used feature
535 importance metrics available in *Random Forest* algorithm. Figure 15 presents the feature importance
536 metrics in case of autoignition as well as ignition delay. It can be observed that temperature plays one of
537 the major roles in governing the outcome of both the parameters followed by oxygen mass fraction. While
538 there is a small difference in the extent to which these features are important, the overall trend remains
539 the same and is coherent with physical understanding of the physical process considered in the present
540 work.



541

542 Figure 15: Feature importance metrics for autoignition and ignition delay time.

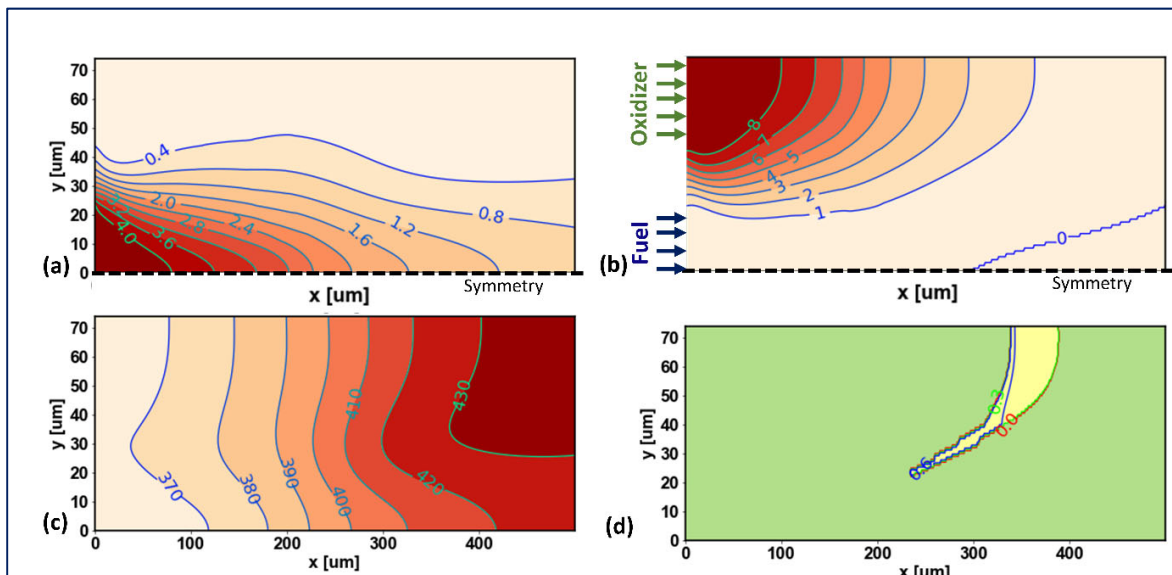
543 So far, we have highlighted how homogenous ignition calculations coupled with tentative flow time scale
 544 can be used to build data-driven machine learning models, which when trained judiciously can predict
 545 whether we can have autoignition as well as ignition delay time for a given set of operating parameters.
 546 However, a more important question which arises here is how these models can be further used to
 547 develop understanding as well improving the SCWO process at microscale. We discuss this aspect in next
 548 section.

549 **5 Potential implication of ML models**

550 In this section, we present a perspective on how these models and the approach in general can be used
 551 to better understand and improve upon SCWO process apart from the default utility in predicting the
 552 autoignition as well as ignition delay time. One of utilities of this approach is that the trained models can
 553 be used to map the flow field (in terms of temperature, fuel concentration, etc.) at a given time from CFD
 554 simulations to predict where in the flow domain autoignition is likely to occur and what could be the
 555 corresponding ignition delay time. This can provide insights into the probable region in the domain where
 556 ignition kernel could form, an essential and important aspect in understanding the flame dynamics. In
 557 order to explain this in a more qualitative way, we perform CFD simulations for the schematic shown in

558 Figure 2. Owing to symmetry in the domain (as highlighted in the figure) only half of the domain is
 559 simulated. The modeling approach is similar to the one described in [36] and is not been described here
 560 in detail as the prime objective is to demonstrate the described implication without any quantitatively
 561 interpretations. Figure 16 shows an example of the generated 2D field of fuel and oxygen concentration
 562 along with temperature contours. It is to be mentioned that since the objective lies in predicting the
 563 autoignition condition, we are not concerned about what happens after the ignition using this
 564 methodology. The input data is fed to the machine learning model following which regions where ignition
 565 is likely to happen is identified. For the given configuration, it is likely that ignition will happen in regions
 566 dominated by oxidizer concentration which is expected. Mapping the ignition delay time on this contour
 567 plot shows regions with small ignition delay time thereby highlighting the likely regions for the formation
 568 of an ignition kernel.

569



570

571 Figure 16: Illustration of how ML models can assist in predicting ignition kernel region. 2D field generated from
 572 simulation. (a) Fuel mass percentage (b) Oxygen mass percentage (c) Temperature [in C] (d) Machine learning model
 573 prediction of autoignition zone (yellow) and ignition delay time.

574 Further implications of this methodology are briefly described as follows. In a real scenario, autoignition
575 will depend on how the flow field evolves and thus hydrodynamics is bound to play an important role.
576 When the fuel and the oxidizer are injected into the reactor microchannel, the temperature of the mixture
577 element will start to increase due to the reaction as the fluid element advects downstream. Thus, even if
578 we started with conditions corresponding to probability of no ignition as per homogeneous calculation,
579 the conditions can eventually change as the fluid element is advected and thus we can have conditions
580 leading to autoignition later in time. In such a scenario, it will be intuitive to have the design of reactors,
581 such as extending flow paths, using bluff bodies, etc. to alter hydrodynamics and mixing dynamics, which
582 ensure that such conditions are attained as close as possible to the inlet region. Thus, rather than running
583 a large number of simulations or experiments, mapping the simulation fields from several designs to
584 predict ignition as illustrated ~~previously~~ can significantly aid in designing efficient microreactors.
585 Furthermore, a similar mapping can be done for ignition delay time. This can provide insights for better
586 understanding the impact of hydrodynamics on the autoignition phenomenon. Besides, the methodology
587 can be extended in analyzing the impact of various process parameters, such as minimum inlet
588 temperature at which ignition can occur, minimum fuel and oxidizer concentration for wide range of
589 physical dimensions of the microreactor. A more important implication would lie in 3D analysis of the
590 system. Owing to the small scale of the system, running a 3D simulation can be computationally very
591 expensive. This may lead to limiting the parametric space of various geometric parameters to be explored
592 and thus constraint optimum microreactor design. However, extending the methodology from 2D
593 systems, an initial estimate can be made in 3D design configurations. Thus, the implications of the current
594 approach of developing machine learning models are not only restricted to the trivial application of
595 predicting autoignition conditions but can significantly improve the understanding and design
596 considerations of microreactor in $\mu SCWO - H$. The presented approach will be used in the future with
597 real simulation data to understand the dynamics of hydrothermal flames at microscale.

598 **6 Conclusion and perspectives.**

599 Machine learning based models were trained and tested to predict autoignition for a given set of
600 operating parameters (in terms of pressure, temperature, and fuel and oxidizer concentration) leading to
601 the formation of hydrothermal flames for supercritical water oxidation at microscale for its application in
602 the space industry. The autoignition criteria was defined in relation to residence time of the fluid/reacting
603 element in the microreactor. Thus, despite homogeneous reaction cases which may always yield
604 autoignition, this was restricted in the present case owing to limitation by flow time scales involved.
605 Among several classification models, *Random Forest* and *Support Vector Machine* were able to predict
606 outcomes with high accuracy. Subsequently, regression models were used to predict ignition delay time
607 where a two-step sequential strategy consisting of first predicting autoignition followed by ignition delay
608 time for the corresponding cases. The developed approach was further tested on an example of a simple
609 model system with 2D simulation to highlight the potential of using Machine Learning models beyond
610 simple prediction of autoignition and ignition delay time. Several further implications and utility of the
611 machine learning methodology were presented in the context of present problem highlighting how it can
612 assist in understanding the onset of hydrothermal flames at microscale and eventually design efficient
613 microreactors. As a future perspective, it is intended to couple the hydrodynamics of jets or co-axial flow
614 to account for more intricate hydrodynamic time scales into the machine learning models. This will further
615 improve the predictability of these models.

616 **Acknowledgements**

617 The authors are grateful to the French space agency CNES for funding the postdoctoral position of
618 Deewakar Sharma and for financial support awarded through GdR MFA.

619 **Abbreviations used for various Machine learning models**

<u>Abbreviation</u>	<u>Name</u>
Ridge	Ridge regression model
DT	Decision Tree
RF	Random Forest
kNN	k-nearest neighbor
Ada	AdaBoost
GB	Gradient Boost
SVC	Support vector classifier

620

621 **References**

- 622 [1] Hicks MC, Hegde UG, Fisher JW. Investigation of Supercritical Water Phenomena for Space and
623 Extraterrestrial Application. NASA; 2012.
- 624 [2] Hicks MC, Hegde UG, Kojima JJ. Hydrothermal ethanol flames in Co-flow jets. *The Journal of*
625 *Supercritical Fluids* 2019;145:192-200.
- 626 [3] Bermejo MD, Cocero MJ. Supercritical water oxidation: A technical review. *AIChE Journal*
627 2006;52(11):3933-51.
- 628 [4] Serikawa RM, Usui T, Nishimura T, Sato H, Hamada S, Sekino H. Hydrothermal flames in
629 supercritical water oxidation: investigation in a pilot scale continuous reactor. *Fuel*
630 2002;81(9):1147-59.
- 631 [5] Augustine C, Tester JW. Hydrothermal flames: From phenomenological experimental
632 demonstrations to quantitative understanding. *The Journal of Supercritical Fluids* 2009;47(3):415-
633 30.
- 634 [6] Sierra-Pallares J, Teresa Parra-Santos M, García-Serna J, Castro F, José Cocero M. Numerical
635 modelling of hydrothermal flames. Micromixing effects over turbulent reaction rates. *The Journal*
636 *of Supercritical Fluids* 2009;50(2):146-54.
- 637 [7] Wellig B, Weber M, Lieball K, Príkopský K, Rudolf von Rohr P. Hydrothermal methanol diffusion
638 flame as internal heat source in a SCWO reactor. *The Journal of Supercritical Fluids* 2009;49(1):59-
639 70.
- 640 [8] Deewakar Sharma, Arnaud Erriguible, Olivier Nguyen, Carole Lecoutre, Yves Garrabos, et al.. A
641 preliminary investigation of microreactor designs for supercritical water oxidation using
642 hydrothermal flames for space applications. *13th International Symposium on Supercritical Fluids,*
643 *ISASF (International Society for the Advancement of Supercritical Fluids), May 2022, Montréal,*
644 *Canada. 2 p. [hal-04129176](#)*
- 645 [9] A. Cario, C. Fauveau, O. Nguyen, C. Lecoutre, Y. Garrabos, Marre. S. Microreacteurs en saphir
646 (CNRS et CNES). 2020.
- 647 [10] Brunton SL, Noack BR, Koumoutsakos P. Machine Learning for Fluid Mechanics. *Annual Review of*
648 *Fluid Mechanics* 2020;52(1):477-508.

- 649 [11] Brenner MP, Eldredge JD, Freund JB. Perspective on machine learning for advancing fluid
650 mechanics. *Physical Review Fluids* 2019;4(10):100501.
- 651 [12] Brunton SL. Applying machine learning to study fluid mechanics. *Acta Mechanica Sinica*
652 2021;37(12):1718-26.
- 653 [13] Zhang T, Li Y, Li Y, Sun S, Gao X. A self-adaptive deep learning algorithm for accelerating multi-
654 component flash calculation. *Computer Methods in Applied Mechanics and Engineering*
655 2020;369:113207.
- 656 [14] Liu Y, Hong W, Cao B. Machine learning for predicting thermodynamic properties of pure fluids
657 and their mixtures. *Energy* 2019;188:116091.
- 658 [15] Zhu B, Zhu X, Xie J, Xu J, Liu H. Heat Transfer Prediction of Supercritical Carbon Dioxide in Vertical
659 Tube Based on Artificial Neural Networks. *Journal of Thermal Science* 2021;30(5):1751-67.
- 660 [16] Sun F, Xie G, Song J, Li S, Markides CN. Thermal characteristics of in-tube upward supercritical CO₂
661 flows and a new heat transfer prediction model based on artificial neural networks (ANN). *Applied*
662 *Thermal Engineering* 2021;194:117067.
- 663 [17] Lashkarbolooki M, Vaferi B, Shariati A, Zeinolabedini Hezave A. Investigating vapor-liquid
664 equilibria of binary mixtures containing supercritical or near-critical carbon dioxide and a cyclic
665 compound using cascade neural network. *Fluid Phase Equilibria* 2013;343:24-9.
- 666 [18] Wang K, Luo J, Wei Y, Wu K, Li J, Chen Z. Practical application of machine learning on fast phase
667 equilibrium calculations in compositional reservoir simulations. *Journal of Computational Physics*
668 2020;401:109013.
- 669 [19] Zheng Z, Lin X, Yang M, He Z, Bao E, Zhang H, et al. Progress in the Application of Machine Learning
670 in Combustion Studies. *ES Energy & Environment* 2020;9:1-14.
- 671 [20] Zhou L, Song Y, Ji W, Wei H. Machine learning for combustion. *Energy and AI* 2022;7:100128.
- 672 [21] Pan Y, Jiang J, Wang R, Cao H, Cui Y. Predicting the auto-ignition temperatures of organic
673 compounds from molecular structure using support vector machine. *Journal of Hazardous*
674 *Materials* 2009;164(2):1242-9.
- 675 [22] Suleiman MA, Owolabi TO, Adeyemo HB, Olatunji SO. Modeling of autoignition temperature of
676 organic energetic compounds using hybrid intelligent method. *Process Safety and Environmental*
677 *Protection* 2018;120:79-86.
- 678 [23] Shah N, Zhao P, DeVescovo D, Ge H. Prediction of Autoignition and Flame Properties for
679 Multicomponent Fuels Using Machine Learning Techniques. SAE International; 2019.
- 680 [24] Cui Y, Wang Q, Liu H, Zheng Z, Wang H, Yue Z, et al. Development of the ignition delay prediction
681 model of n-butane/hydrogen mixtures based on artificial neural network. *Energy and AI*
682 2020;2:100033.
- 683 [25] vom Lehn F, Cai L, Copa Cáceres B, Pitsch H. Exploring the fuel structure dependence of laminar
684 burning velocity: A machine learning based group contribution approach. *Combustion and Flame*
685 2021;232:111525.
- 686 [26] Cui Y, Liu H, Wang Q, Zheng Z, Wang H, Yue Z, et al. Investigation on the ignition delay prediction
687 model of multi-component surrogates based on back propagation (BP) neural network.
688 *Combustion and Flame* 2022;237:111852.
- 689 [27] Huang Y, Jiang C, Wan K, Gao Z, Vervisch L, Domingo P, et al. Prediction of ignition delay times of
690 Jet A-1/hydrogen fuel mixture using machine learning. *Aerospace Science and Technology*
691 2022;127:107675.
- 692 [28] Koido K, Hirotsuka K, Kubo T, Fukayama M, Ouryouji K, Hasegawa T. Numerical study on premixed
693 hydrothermal combustion in tube reactor. *Combustion Theory and Modelling* 2009;13(2):295-
694 318.
- 695 [29] Quinlan JR. Induction of decision trees. *Machine Learning* 1986;1(1):81-106.
- 696 [30] Breiman L. Random Forests. *Machine Learning* 2001;45(1):5-32.

- 697 [31] Cortes C, Vapnik V. Support-vector networks. *Machine Learning* 1995;20(3):273-97.
- 698 [32] Cover T, Hart P. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*
699 1967;13(1):21-7.
- 700 [33] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine
701 Learning in Python. *Journal of Machine Learning Research*;12:2825-30.
- 702 [34] Hoerl AE, Kennard RW. Ridge Regression: Biased Estimation for Nonorthogonal Problems.
703 *Technometrics* 2000;42(1):80-6.
- 704 [35] Hossin M, Sulaiman MN. A Review on Evaluation Metrics for Data Classification Evaluations.
705 *International Journal of Data Mining & Knowledge Management Process* 2019;5(2):1-11.
- 706 [36] Zhang F, Marre S, Erriguible A. Mixing intensification under turbulent conditions in a high pressure
707 microreactor. *Chem Eng J* 2020;382:122859.

708

709

710

711

712

713