Chemometrics and Intelligent Laboratory Systems 206 (2020) 104161

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

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

Tutorial article

Prediction of sulfur content in diesel fuel using fluorescence spectroscopy and a hybrid ant colony - Tabu Search algorithm with polynomial bases expansion

Lucas Ranzan *, Luciane F. Trierweiler, Jorge O. Trierweiler

Group of Intensification, Modeling, Simulation, Control, and Optimization of Process, GIMSCOP, Federal University of Rio Grande do Sul, Chemical Engineering Department, 90040-040, Porto Alegre, RS, Brazil

ARTICLE INFO

Keywords:
Ant colony optimization
Tabu search
Polynomial combination
EEM Fluorescence
Diesel
Sulfur

ABSTRACT

It is widely accepted that feature selection is an essential step in predictive modeling. There are several approaches to feature selection, from filter techniques to meta-heuristics wrapper methods. In this paper, we propose a compilation of tools to optimize the fitting of black-box linear models. The proposed AnTSbe algorithm combines Ant Colony Optimization and Tabu Search memory list for the selection of features and uses l1 and l2 regularization norms to fit the linear models. In addition, a polynomial combination of input features was introduced to further explore the information contained in the original data. As a case study, excitation-emission matrix fluorescence data were used as the primary measurements to predict total sulfur concentration in diesel fuel samples. The sample dataset was divided into S10 (less than 10 ppm of total sulfur), and S100 (mean sulfur content of 100 ppm) groups and local linear models were fit with AnTSbe. For the Diesel S100 local models, using only 5 out of the original 1467 fluorescence pairs, combined with bases expansion, we were able to satisfactorily predict total sulfur content in samples with MAPE of less than 4% and RMSE of 4.68 ppm, for the test subset. For the Diesel S10 local models, the use of 4 Ex/Em pairs was sufficient to predict sulfur content with MAPE 0.24%, and RMSE of 0.015 ppm, for the test subset. Our experimental results demonstrate that the proposed methodology was able to satisfactorily optimize the fitting of linear models to predict sulfur content in diesel fuel samples without need of chemical of physical pre-treatment, and was superior to classic PLS regression methods and also to our previous results with ant colony optimization studies in the same dataset. The proposed AnTSbe can be directly applied to data from other sources without need for adaptations.

1. Introduction

Nowadays, predictive modeling permeates every knowledge field. The ability to quantify an output of interest, classify occurrences, or predict future outcomes using auxiliary measures are base principles of many technological advances of the last decades. From detecting cancer [1,2], social media behavior [3], to industrial production [4], new machine learning algorithms are changed and developed daily to predict information that would be too costly, invasive, or impossible to access directly.

Black box models establish a functional relationship between system inputs and outputs [5]. The parameters of these functions do not need to have any phenomenological significance (*e.g.*, heat or mass transfer coefficients or reaction kinetics), but are very efficient in faithfully representing trends in the process behavior [6]. With a collection of empirical or simulated data of a system, a model can be fitted to find the correlation of that information with one or more outputs of interest.

In many applications, the number of available input features can reach hundreds or even thousands of variables (*e.g.*, image classification [7–9], spectral data [10–12], and industrial processes [13,14]). However, data can be associated with a high level of noise, collinearity, and be filled with irrelevant or redundant variables [15]. Several selection techniques were developed to address the problem of extracting valid information from the features which can efficiently describe the input data while reducing noise and useless variables [16]. These techniques can be categorized into *Feature extraction* and *Feature selection*.

Feature extraction approaches project features into a new space with lower dimensionality by combining the original feature space. Examples

* Corresponding author.
E-mail addresses: lranzan@enq.ufrgs.br (L. Ranzan), jorge@enq.ufrgs.br (J.O. Trierweiler).

https://doi.org/10.1016/j.chemolab.2020.104161
Received 12 May 2020; Received in revised form 23 August 2020; Accepted 10 September 2020
Available online 15 September 2020

include Principal Component Analysis (PCA) [17,18], Linear Discriminant Analysis (LDA) [19], and Partial Least Squares (PLS) [20], which are the most widely applied techniques to deal with high dimensionality data [21]. However, it is difficult to link the features from the original space with the new features since there is no physical meaning for the transformed variables.

On the other hand, the Feature selection approaches aim to select a small subset of features that minimize redundancy and maximize relevance to the target, maintaining the physical meanings [15]. Metaheuristics search algorithms (as Genetic Algorithm [22], Particle Swarm Optimization [23] and Ant Colony Optimization [24]) use the predictor as a black box and the predictor performance as the objective function to evaluate a feature subset. The feature search component will produce a set of features that will be used by the learning algorithm to predict an output. The performance of the prediction returns to the feature search component for the next iteration of subset selection.

The Ant Colony Optimization (ACO) is a metaheuristic multi-agent algorithm used to solve hard combinatorial optimization problems. There are several review and survey papers [25–27] dedicated to ACO applications, that include areas as fluid dynamics, telecommunications, bioinformatics, system modeling, simulation, image processing, routing, scheduling, and production problems, logistics, transportation and supply chain management. ACO algorithms try to mimic the food foraging behavior of real ants adapting a pheromone memory model that governs the way agents wander the search space. The management of this pheromone influences the diversification (i.e., exploration) and the intensification (i.e., exploitation) of the search process. A well-design ACO algorithm use strategies to balances exploration and exploitation (E&E) to find high-quality solutions for problems [10,28–30]. A natural evolution of E&E techniques was the hybridization of metaheuristic algorithms [31,32]. The incorporation of principles from other searching algorithms into ACO was used to solve the Traveling Salesman Problem [33], Vehicle Routing problem [34–36], Tasks to Workstations Assignment problem [37], Multi-skill Resource-constrained Project Scheduling problem [38], and Quadratic Assignment problems [39,40].

The two major contributions of our methodology is the proposal of a hybrid variable selection algorithm based on Ant Colony Optimization and Tabu Search (TS) [41,42], to solve early stagnation and avoid redundant calculations, and the use of the *expansion of bases* to further explore the information contained in the data. After the selection of inputs (and before model fitting), the selected variables are expanded as a new feature matrix consisting of all polynomial combinations of features with degree equal or less than a defined value. This expansion can capture non-linear and combinatorial information that may not be perceived otherwise.

To evaluate the methodology, a case study is presented, where Excitation-Emission Matrix (EEM) fluorescence spectroscopy is used as primary information to predict total sulfur concentration in diesel fuel.

2. Methodology

All implementations in this work were done in Python v3.5.4.1 in combination with the readily available modules, especially from the SciKit Learn library version 0.20.2 54.

2.1. Preprocessing

The first necessary step for any modeling procedure is the pretreatment of the data. If misleading information, outliers, and unscaled data are given to the optimizer, there will be a detriment in the efficiency of the algorithm. The preprocessing routine comprehends i) outliers detection; ii) data segmentation into training, validation, and test data sets; and iii) data scaling.

2.1.1. Outliers detection – Hoteling's T^2 statistic

This method is coupled with Principal Component Analysis (PCA). All input data is first scaled to mean zero and standard deviation one; then, PCA is applied, and the eigenvalues and eigenvectors are used to calculate the T^2 statistic for each sample, expressed as Eq. (1) [43]:

$$T^2 = X^T \widehat{W} \widehat{\Lambda}^{-1} \widehat{W}^T X \quad (1)$$

where $\widehat{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_l)$ is a diagonal matrix containing the eigenvalues related to the l retained PCs; X the matrix of scaled inputs; and $\widehat{W}$ the matrix of the l retained eigenvectors. In this implementation, the number of retained PCs is the one able to capture at least 95% of the original variance of the data. The T^2_α threshold is computed as Eq. (2):

$$T^2_\alpha = \frac{l(N-1)}{(N-l)} F_{l,N-l,\alpha} \quad (2)$$

where l is the number of retained principal components, N the number of samples, α the level of significance (defined as 5%) and $F_{l,N\text{-}l,\alpha}$ is the Fisher distribution with l and (N-l) degrees of freedom. Any sample with T^2 higher than T^2_α is considered an outlier and removed from the dataset [44].

2.1.2. Data set splitting based on a modified version of K-rank

The methodology for splitting the data into calibration (cal), validation (val), and testing (test) subsets is the one implemented by Santos et al. [45]. This methodology is especially useful when dealing with multiple solutions problems: situations where standardizes combinations of the input variables can yield the same output y.

First, the user chooses a k number of clusters, ranging from 1 to N-1 (number of samples - 1). Then, a k-means algorithm [46] with k centroids is run using only the input variables to split the dataset into k_i similar groups. For each cluster: the $k_{i,\text{samples}}$ are sorted in ascending order for a selected output y; the proportions of each subset are chosen (*e.g.,* 60% calibration – 20% validation – 20% testing), and the methodology adapts a pattern to select, in order, the samples to their respective subsets (*e.g.,* cal-cal-cal-val-test …). In this implementation, the extremes samples (with minimum and maximum values of y) in each cluster are always selected for the training subset, to avoid extrapolation. This clustering of data is especially useful when dealing with cases that have a multiplicity of solutions.

Although other splitting methodologies (as cross-validation) can be appealing when fitting black-box models (to avoid overfitting and the influence of abnormal samples), the selection of variables can be a very time-demanding task, and the use of multiple cal/val/test subsets can, for some cases, render the total computational time prohibitive.

2.1.3. Data scaling

Standardization is a common requirement for many machine learning estimators. Many elements used in the objective function of a learning algorithm (as $l1$ and $l2$ regularizes of linear models) assume that all features are centered on zero and have variance in the same order. If a feature has a variance that is orders of magnitude larger than others, it might dominate the objective function and make the estimator unable to learn correctly from other features as expected [47]. The scaler function must be fitted using the training subset, and then the other subsets are scaled accordingly. The user can define which scaler is better for their specific case.

After the pre-processing stage, we end up with clean and scaled data, with all samples already divided into subgroups, ready to be processed by the AnTSbe algorithm, the core optimizer that will venture through the possible combinations of input features to fit linear models and predict the desired output.

2.2. AnTSbe – ant colony optimizer hybridized with Tabu Search and bases expansion

The AnTSbe algorithm is based in Ant Colony Optimization [48], a type of stochastic optimization where multiple parallel processes (ants) take different 'routes' to minimize an objective function. When applied to optimization, the algorithm ascribes a quality indicator - called pheromone - to each input variable. The pheromone of each input is incremented after each iteration based on how well a model using this variable could predict the desired output (the lower the error, the higher the increment in pheromone). The more pheromone a variable has, the higher the chance ants in the next iteration will select it. One of the problems of swarm intelligence algorithms is the stagnation in local minimums: after a number of iterations, some inputs can dominate the pheromone trail in such a way that it loses its exploration capabilities. There are several proposals of ACO adaptations to avoid stagnation, as pheromone reset, reactive memory, smoothing, and max-min bounds [49]. In this work, we will incorporate principles from Tabu Search into the algorithm to avoid early stagnation. TS is a neighborhood search-based method that uses a memory structure to avoid being trapped in local optima. It improves the efficiency of the searching process by storing a tabu list of local solutions that were used to restrict the search by forbidding moves to some poor neighbor solutions that already have been visited [50]. One feature of tabu exploration is diversification, responsible for moving the exploration process over different regions of the search space. In this implementation, TS is hybridized with ACO not to search for neighbor solutions directly, but to construct a short-term memory tabu list to avoid previously tested input combinations. The memory is short-term because it only considers tested input combinations of the last z iterations (after z iterations the input combinations are not forbidden anymore). Forbidding previously tested combinations for some iterations encourages exploration, and the evaporation of pheromone can help to avoid stagnation in local optima.

To compare and evaluate models, three metrics are used throughout the algorithm:

(i) Mean Absolute Percentage Error (MAPE), defined as Eq. (3):

$$MAPE = \frac{100\%}{n}\sum_{i}^{n}\left|\frac{(y_i - \hat{y}_i)}{y_i}\right| \quad (3)$$

where n is the number of samples, y the real value of the output and $\hat{y}$ the predicted value of the output;

(ii) Root Mean square error (RMSE) – Eq. (4),

$$RMSE = \sqrt{\frac{1}{n}\sum_{i}\left(y_i - \hat{y}_i\right)^2} \quad (4)$$

and (iii) the Coefficient of Determination (R^2) – Eq. (5),

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2} \quad (5)$$

where $\bar{y}$ is the mean value of y.

The AnTSbe is divided into three phases: Phase One is the initialization stage, where the optimization parameters are chosen; Phase Two is the core of the optimization, where models are fitted, evaluated and compared; and, in Phase Three, the best-fitted models are presented, with their corresponding variables and all comparative metrics.

2.2.1. Phase One – initialization

The initialization process starts with the selection of all needed optimization parameters:

Model Size – *Nw*: the number of input variables in each model. The user can define Nw_0 and Nw_n to be the initial and final model sizes, respectively (with $Nw_0 \leq Nw_n$). If defined, *Nw* will assume values within Nw_0 and Nw_n. The algorithm as a whole (Phase one to three) will be run individually for each *Nw*.

Type of model: the type of regression to be fitted. The Ridge Regression (through the scikit-learn function sklearn.linear_model.Ridge) solves a regression model where the loss function is the Linear Least Squares function, and the regularization is given by the $l2$-norm, which aims to reduce the magnitude of coefficients [51]. The Lasso Regression, similarly, uses $l1$-norm regularization, which penalizes the number of total parameters in the model, reducing some coefficients to absolute zero [52]. The used Scikit-learn LassoLarsIC (criterion = 'bic') implementation solves the Lasso model using Least Angle Reduction (Lars), and the selection of the regularization parameter α is based on the Bayesian information criterion [53], making a trade-off between the goodness of fit and the complexity of the model. The use of regularization will foment the fitting of models more robust to overfitting, discarding inefficient input variables and avoiding singularity issues during a model fitting in cases where there are more variables than observations.

Base Expansion – σ: generate a new feature matrix consisting of all polynomial combinations of features with degree equal or less than σ. Even though the models are linear in the parameters, the expansion of the selected input variables into polynomial combinations can capture information that could improve the prediction metrics. It was essential to only expand the bases after the selection of inputs, because, in cases where there are thousands of variables, if we simply expand all the features before selection, the complexity of the problem and the number of local minimums would increase exponentially.

Optimization Metric – *OptMetric*: the metric involved in the loss function the algorithm will minimize. It can be RMSE or MAPE. If MAPE is selected, it has the tendency to minimize prediction errors of outputs with values closer to zero. If RMSE is selected, it has the tendency to minimize prediction errors of outputs with higher values.

Number of runs – μ: number of times the algorithm resets the pheromone trail to its initial value τ_0.

Number of iterations – *t*: total number of iterations the algorithm performs in each run.

Number of ants – N_{ants}: number of models fitted in each iteration.

Tabu memory size – *z*: number of past iterations where the tested combinations are part of the tabu memory.

Initial pheromone value – τ_0: the initial amount of pheromone for each variable.

Pheromone gain – *k*: numerator of the expression of pheromone increment each ant will add to the variables it has selected in that iteration.

Pheromone evaporation rate – ρ: defines how much of the current pheromone will not be kept for the next iteration. If close to one, it will heavily penalize unselected inputs or inputs that fitted models with high prediction errors.

Once all parameters are established, the Global Solution is initialized by fitting a model with random *Nw* variables. This Global Solution will be latter compared to future fitted models. The tabu memory is also initialized as an empty list.

2.2.2. Phase Two – optimization

During Phase Two, the ant army will evaluate possible combinations of input variables that could minimize the loss function. At the beginning of each t_i iteration, each of the N_{ants} ants will select *Nw* input variables. This selection is based on the pheromone trail and in a random factor. The random factor is represented by a random trigger, generating values between 0 and 1. The pheromone trail is transformed in a pheromone density vector γ, by dividing the pheromone of each input by the sum of all pheromone, and then accumulated as $C\gamma$ (Eq. (6)):

$$C\gamma_i = \sum_{1}^{i}\gamma_i \therefore \gamma_i = \frac{\tau_i}{\sum \tau} \quad (6)$$

The random trigger is fired, and its value compared to the accumulated pheromone density. The closest input with accumulated density

higher than the trigger is selected, removed from the selection pool, and the γ is updated without that input. The procedure is repeated until *Nw* features are selected. The combination between the random trigger and the density of pheromone guarantees that every input has a chance into being selected, but the higher the pheromone density, the greater the chance of an input being selected. Fig. 1 shows an example of input selection with a random trigger of 0.856. In this example, input X_9 would be the one selected.

After the ant has selected *Nw* inputs, its selection is compared to the combinations stored in the tabu memory. If the ant has selected a forbidden combination, the ant resets and starts the selection once again. If the ant combination is not forbidden, the selected inputs are expanded as all polynomial combination of features with degree equal or less than σ (*e.g.*, if $[X_1, X_2]$ are the chosen inputs, an expansion with $\sigma = 2$ will result in a new matrix $[X_1, X_2, X_1^2, X_2^2, X_1{*}X_2]$).

The ant will use the chosen variables (original or expanded, depending on the σ) and samples from the calibration subgroup to fit a linear model of the selected regression type. The fitted model is used to predict the output of interest for all subgroups (calibration, validation, and test), and all metrics are evaluated. The ant compares the quality of its model with the Global Solution, based on a loss function ψ_q (Eq. (7)),

$$\psi_q = [OptMetric]_{cal} + [OptMetric]_{val} + \frac{\max[OptMetric]_{cal}, [OptMetric]_{val} + 1}{\min[OptMetric]_{cal}, [OptMetric]_{val} + 1} \quad 7$$

if the ant's ψ_q is smaller than the Global Solution's ψ_q, the ant's model becomes the new Global Solution.

Finally, the ant deposits pheromone in each of the variables it has selected according to Eq. (8):

$$\tau_{i,t+1} = \tau_{i,t} + \frac{k}{(\psi_q + 1)^3} \quad 8$$

where *k* is the pheromone gain. The higher the prediction error the model has, the smaller the increment in pheromone. If by any chance (as in regressions with *l1-norm*), the fitted parameter of an input is absolute zero, then no pheromone is added to that variable. As all ants run in parallel, the increment of pheromone each ant deposits in their selected variables will only be perceived in the next iteration, not affecting the variable selection of other ants in the current t_i.

After all N_{ants} have fitted their models and deposited their pheromones, the pheromone trail is evaporated, multiplying the trail by the evaporation rate ρ (Eq. (9)):

$$\tau_{t+1} = \tau_{t+1} \cdot (1 - \rho) \quad 9$$

The tabu memory list is updated, adding all N_{ants} combinations tested in the current iteration (and removing combinations from any other than the last *z* iterations).

The routine is repeated *t* times, and at each iteration, the pheromone trail is updated. The final pheromone trail can indicate which of the input variables had a higher correlation with the desired output, been part of models with smaller predictive errors. To avoid local minima, Phase Two is repeated μ times, each time restarting the pheromone trail to the initial τ_0 (reset pheromone memory) and emptying the tabu memory list, but carrying on the Global Solution. At the end of every μ run, the final pheromone trail – $\tau_{F\mu}$ – and the best predictive model of each particular run are saved for future reference.

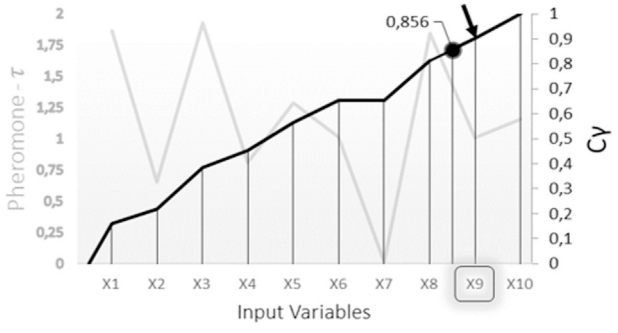

Fig. 1. Example of pheromone-based input selection with a random trigger of 0.856.

2.2.3. Phase Three – global solution

After μ runs, each with *t* iterations and N_{ants} models fitted at each iteration, the algorithm returns the Global Solution - the fitted model with lowest ψ_q - along with its metrics for all subgroups, selected input variables, and corresponding parameters. If the metrics of the test subset are equivalent to the calibration and validation metrics, then the fitted model can be considered robust when dealing with samples never seen before. In addition, the global pheromone trail – τ_G – is presented, as the sum of all τ_F, normalized between 0 and 1 (dividing the vector by its maximum value), as can be seen in Eq. (10). The global pheromone trail is an indicator of which input variables had greater success predicting the output of interest throughout the optimization, been part of models that had better metrics and smaller prediction errors.

$$\tau_G = \frac{\sum_1^\mu \tau_F}{\max\left(\sum_1^\mu \tau_F\right)} \quad 10$$

Fig. 2 presents a schematic representation of the AnTSbe algorithm.

If different model sizes were fitted, the user could select the one that fits better to their needs, based on all calculated metrics for each of the returned Global Solutions. Ideally, if there is a clear correlation between the inputs and the desired output, different sized models should mostly select the same variables.

In cases where there are many highly correlated input variables, the AnTSbe algorithm can be used as a variable filter for further analyses. After running the AnTSbe, it is possible to filter the input variables that had greater success in predicting the desired output by selecting the inputs with higher pheromone (quality indicator) concentration. If the user wants to filter inputs from optimization with only one specific set of parameters, a filter vector is created directly as the global pheromone trail of that optimization. If the user wants to filter inputs considering multiple sets of parameters (*e.g.,* various model sizes for *Ridge* models and the same σ), a multi-filter vector is created as the sum of all individual global pheromone trails. The filter/multi-filter vector is sorted in descending order, and the user defines how many of the first variables should be selected as filtered inputs.

With these filtered inputs, the AnTSbe can be re-run, with any selected parameters, but with the restriction of only selecting inputs within this filtered base. This procedure can be recursively done.

To evaluate the proposed methodology, a case study will be presented next, making use of excitation−emission matrix fluorescence spectroscopy to quantify sulfur concentration in diesel fuel.

3. Case study – Quantifying total sulfur content in diesel fuel samples using EEM fluorescence spectroscopy

The dispersion of sulfur oxides in the atmosphere from the combustion of fossil fuels is a sensitive topic in environmental laws [54]. The sulfur contained in the fuel is directly responsible for the emission of sulfate particulates and SO_3/SO_2 (that causes acid rain) during combustion, the poisoning of refining catalysts, and the corrosion of pipes, storage units and motors [55]. In the last 20 years, most developed countries severely changed their legislations regarding maximum sulfur content in fuels, going from ten thousand ppm to near-zero levels, being less than 15 ppm the typical limit for transportation diesel and gasoline worldwide [56]. Hydrodesulphurization (HDS) is the most common catalytic chemical process used to remove sulfur from refined petroleum

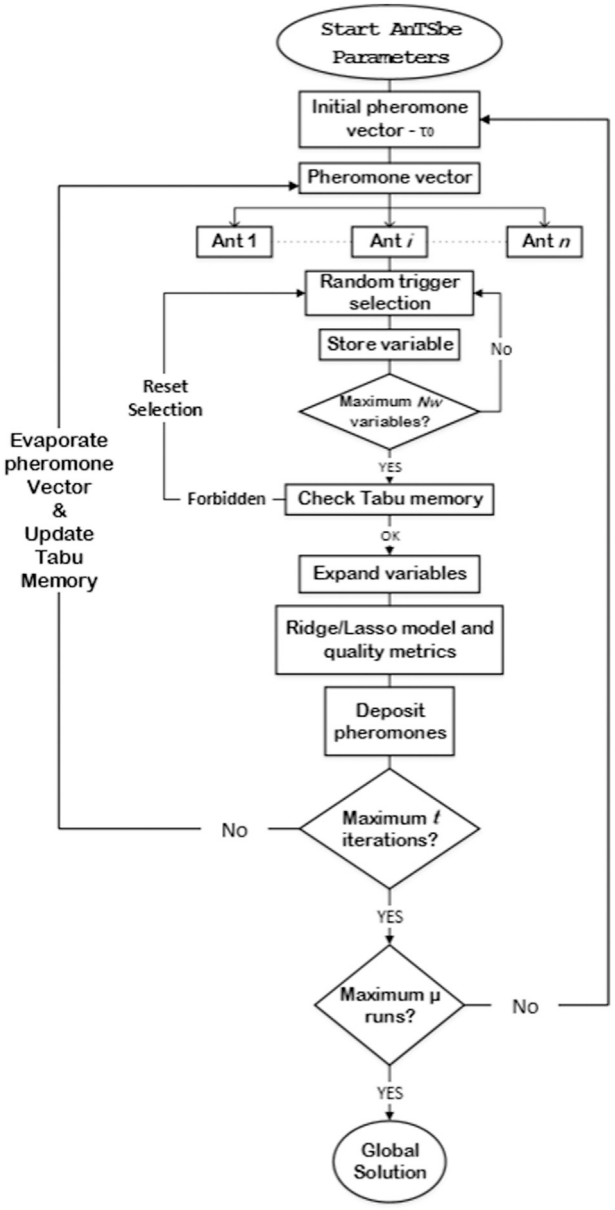

Fig. 2. Schematic representation of the AnTSbe algorithm.

products. In a refinery, the reaction takes place in a fixed-bed reactor at elevated temperatures (from 300 to 400 °C) and elevated pressures (30–130 atm of absolute pressure), typically using alumina base impregnated with cobalt and molybdenum as a catalyst, combining the feed with a hydrogen-rich stream and retrieving hydrogen sulfide (H_2S) [57]. HDS is a costly procedure and should be fine-tuned to avoid inefficiency, producing streams out of specification that demand rework.

Presently, to quantify total sulfur concentration in a stream after HDS, a sample must be collected and taken to a laboratory where qualified people will handle it and run the specific certification test for sulfur quantification in their legislation (as the ASTM D-4294 [58], in Brazil). This quantification procedure, besides being expensive, usually takes hours in a typical refinery. That means that when the operator finally receives feedback, it can be too late to make any control action.

To implement advanced controlling techniques to the HDS, it is of utmost importance to be able to predict in real-time the sulfur concentration of the desulfurized product. As can be seen in works as [11, 59–61], EEM fluorescence spectroscopy can be used as a non-invasive, fast, and sensitive technique to capture information about total sulfur content in diesel fuel. After HDS, most of the oil's sulfur is contained in stable polycyclic aromatic molecules as benzothiophenes and dibenzothiophenes, being the latter the hardest sulfur to remove in diesel oil [62, 63]. These compounds and their derivatives present luminous proprieties, being natural fluorophores [64–67].

The objective of this case study is to apply the AnTSbe methodology to optimize the selection, among thousands of fluorescence excitation-emission pairs, of the ones able to predict total sulfur content in diesel samples satisfactorily.

3.1. Dataset

The dataset used in this study is the one gathered by Ranzan et al. [11]: excitation-emission matrix fluorescence spectra of sixty-one samples of diesel fuel, provided and certified by a Brazilian petroleum refinery. The samples were characterized as Diesel S10 (samples with total sulfur lower than 10 ppm) and Diesel S100 (samples with average total sulfur around 100 ppm). The referred work used the data in a purely classificatory study, being able to label all samples correctly, but without any regard about sulfur quantification.

Diesel S10 – eleven samples were characterized as Diesel S10; they had between 5.1 and 6.4 ppm of total sulfur, with an average of 5.8 ppm. The sulfur content of these samples was certified according to the ASTM Standard D-7039 [68], using a Sindie 7039 bench analyzer by XOS®.

Diesel S100 – fifty samples were characterized as Diesel S100, with total sulfur content between 73.7 and 118 ppm, and an average of 99.5 ppm. All samples were certified according to the ASTM Standard D-4294, using a LABX-3000 by Oxford®.

EEM fluorescence spectra – the fluorescence spectra were collected using a Horiba® Fluoromax-4, equipped with a xenon lamp of 150 W. The measurements were made in a range of excitation wavelengths between 260 and 600 nm and emission wavelengths between 290 and 850 nm. The geometry of measurements was 90°. Both excitation and emission wavelengths used an increment of 10 nm. With these arrangements, each fluorescence spectra was obtained as a 57 × 35 matrix, containing the fluorescence intensity of 1995 excitation/emission (Ex/Em) pairs. As no excitation can lead to emission with a smaller wavelength, there were 1467 valid fluorescence pairs in each spectrum. Each EEM spectra was later unfolded into a row vector, the row representing the sample, and each column representing one of the Ex/Em pairs. Measurements were made in triplicate, and all samples were stabilized at 25 °C using a thermostatic bath. Fig. 3 presents the average EEM fluorescence spectra for the Diesel S10 and Diesel S100 sample groups.

In this work, we focused on the development of local models, one for Diesel S10 and another for Diesel S100, mainly because the difference in total sulfur concentration and quantity of samples between groups was very significant. If only one global model was fitted, the Diesel S100 samples would dominate the optimization.

3.2. AnTSbe pre-processing and parameters

The following parameters were applied to the optimization of both Diesel S10 and Diesel S100 local models.

For the data pre-processing, as the number of inputs vastly surpluses the number of samples, no outlier was removed in the pre-treatment. For the splitting of subgroups, total sulfur concentration (in ppm) was the selected output to sort the data, and the chosen proportions for the calibration, validation, and testing subsets were 60%, 20%, and 20%, respectively. The number of clusters was defined as one because there was no multiplicity of solutions. The *StandardScaler* was defined as the scaler function, removing the mean and scaling to unit variance (independently in each feature).

The general AnTSbe optimization parameters can be seen in Table 1.

For the Diesel S100 models, Model Size (*Nw*) will range from 3 to 5, and the Type of Model and bases expansion (σ) were arranged as *Ridge* and σ = 1 and *LassoLarsIC* and σ = 2. With this arrangement, the

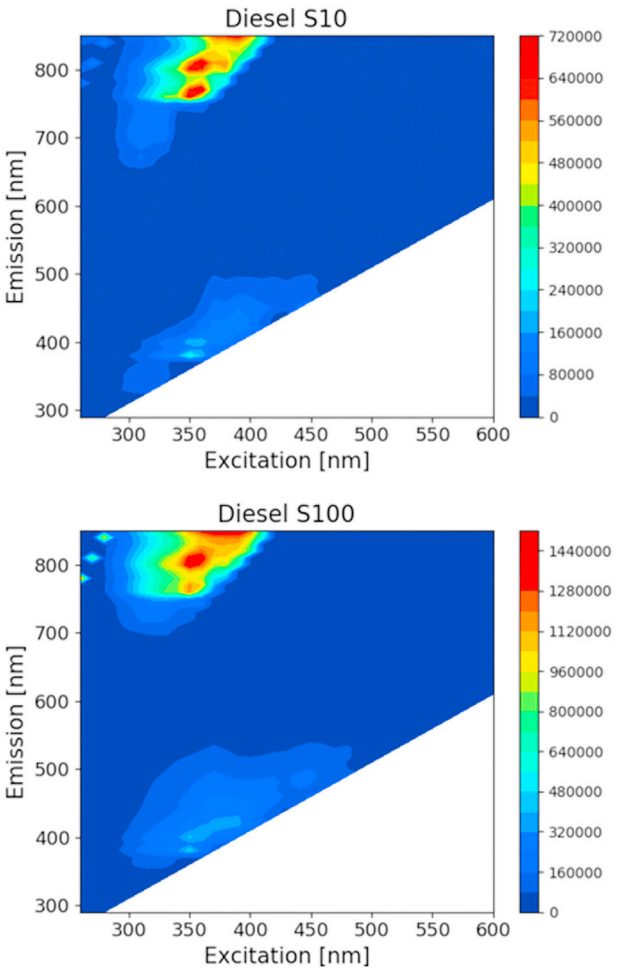

Fig. 3. Average EEM fluorescence spectra for the Diesel S10 and Diesel S100 sample groups.

Table 1
AnTSbe general optimization parameters for diesel fuel local models.

Parameter	Value
OptMetric	RMSE
N. Runs - μ	50
N. iterations - t	150
N_{ants}	200
Tabu Memory Size - z	5
Initial pheromone value – τ0	1000
Pheromone gain – k	100
Pheromone evaporation rate – ρ	0.1

algorithm as a whole will be run six times: three model sizes with *Ridge* and σ = 1 and three model sizes with *LassoLarsIC* and σ = 2. We combined the use of Lasso Regression when applying bases expansion (σ > 1), because the expansion considerably increases the number of model parameters to be fitted, and the use of *l1*-norm regularization helped to keep the models more comprehensible.

As the information contained in the EEM can be highly correlated (many Ex/Em pairs in a region could contain similar information about the desired output), this first round of experiments was also used as a filter to reduce the number of input variables. Then, the algorithm is re-run, to evaluate if we could improve the quality of the predictive models.

To filter the inputs, the multi-filter approach will be applied: the global solution for all model sizes of each specific model type will be evaluated. A multi-filter vector will be created as the sum of the global pheromone trails of each model size (3–5). Considering that each EEM had more than 1400 pairs, we will filter the 100 input variables with higher pheromone concentration and re-run the AnTSbe using only these filtered inputs.

For the Diesel S10 models, Model Size (*Nw*) will range from 2 to 5, and the Type of Model and bases expansion (σ) were selected as *Ridge* and σ = 1, considering the smaller number of samples. As before, the multi-filter approach will be applied to filter the 100 input variables with higher pheromone concentration and re-run the algorithm.

4. Results and discussions

4.1. Diesel S100

The metrics for the first optimization of the Diesel S100 local models can be seen in Table 2. Both the *Ridge* as the *LassoLars* models achieved similar results for the calibration and validation subsets, with MAPE smaller than 4% and RMSE around 4 ppm. For the test subset, the *Ridge* models achieved slightly better results. The base expansion increases the number of variables in the model, and, directly, the number of parameters to be fitted. Using σ = 2, the model sizes 3, 4, and 5 expand to 9, 14, and 20 variables, respectively. All *LassoLars* global solutions kept (non-zero parameters) a combination of primary and expanded variables, indicating that the use of base expansion could be beneficial for Diesel S100 predictive models. In this run, the number of non-zero parameters in *LassoLars* global solutions was 5, 6, and 7 (model sizes 3, 4, and 5).

To filter the results, all final pheromone trails of every model size (3–5), using *Ridge* and σ = 1, were normalized between 0 and 1 and summed. The hundred input variables with higher pheromone concentration were selected as the filtered inputs. In addition, the variables that participated in each global solution were added to this filter vector.

The AnTSbe was re-run, using the same general parameters as the first run, but only selecting variables within the filtered inputs. As for the model size, type, and base expansion, for this filtered run, we choose 3–6, *Ridge* regression and bases expansion 2, to evaluate if the expansion of the variables could improve the prediction metrics. Table 3 presents the metrics for this filtered run.

As can be seen, comparing the metrics in Tables 2 and 3, the filtered run achieved better results than the previous optimizations. There was also a consensus throughout the internal runs of the algorithm about the selected pairs in each model size. Common pairs between different model sizes are bolded in Table 3. There was no significant improvement between selecting 5 or 6 input variables, so, by the principle of parsimony, there was no need to venture further into even bigger models.

As a direct comparison, linear models were built with classical partial least squares (PLS) regression using the same calibration, validation, and test subsets. Table 3 also presents the metrics for the PLS regression model using 7 latent variables (optimized number of LV's). The performance of the AnTSbe model with 5 inputs was significantly better than the PLS regression model for the validation and test subsets, requiring only a fraction of the information (PLS uses the whole spectra).

Fig. 4 presents the measured *vs.* predicted outputs for the *Ridge, LassoLars,* and Filtered *Ridge* global solutions, with model size 5. For a more visual comprehension, Fig. 5 presents all the global solutions' (Nw = 5) selected fluorescence pairs plotted upon the average Diesel S100 EEM.

4.2. Diesel S10

As both the first and the filtered optimization of the Diesel S10 models selected a combination of the same excitation/emission pairs, and the filtered run used the same optimization parameters, we will focus the results directly in the filtered global solutions.

Table 4 Presents the global solutions' metrics and selected fluorescence pairs for the Diesel S10 filtered optimization.

Evaluating the global solutions, the predictive model using only 4 out

Table 2
Global solutions' metrics and selected fluorescence pairs for Diesel S100 - *first optimization*.

Diesel S100 *Ridge* and σ = 1									
	Calibration			Validation			Test		
Model Size	R^2	MAPE	RMSE	R^2	MAPE	RMSE	R^2	MAPE	RMSE
3	0.678	5.16	6.38	0.876	2.62	3.50	0.720	4.25	4.98
4	0.781	4.59	5.27	0.858	3.34	3.75	0.631	4.98	5.72
5	0.823	3.94	4.73	0.855	3.14	3.79	0.758	3.88	4.64
Selected Excitation/Emission Pairs									
3	Ex310/Em330 Ex480/Em590 Ex560/Em580								
4	Ex290/Em350 Ex310/Em330 Ex340/Em730 Ex370/Em720								
5	Ex310/Em330 Ex310/Em680 Ex360/Em370 Ex480/Em590 Ex560/Em580								
Diesel S100 *LassoLars* and σ = 2									
	Calibration			Validation			Test		
Model Size	R^2	MAPE	RMSE	R^2	MAPE	RMSE	R^2	MAPE	RMSE
3	0.857	3.83	4.26	0.810	3.39	4.34	0.480	5.33	6.80
4	0.834	3.74	4.59	0.739	3.91	5.09	0.676	4.81	5.37
5	0.858	3.85	4.24	0.882	2.79	3.42	0.519	5.39	6.53
Selected Excitation/Emission Pairs									
3	Ex310/Em330 Ex370/Em750 Ex440/Em840								
4	Ex310/Em330 Ex370/Em740 Ex500/Em790 Ex520/Em650								
5	Ex300/Em430 Ex310/Em720 Ex340/Em760 Ex400/Em550 Ex470/Em510								

Table 3
Global solutions' metrics and selected fluorescence pairs for Diesel S100 - *filtered optimization* and PLS regression (7 LV) metrics for comparison.

Diesel S100 Filtered *Ridge* and σ = 2									
	Calibration			Validation			Test		
Model Size	R^2	MAPE	RMSE	R^2	MAPE	RMSE	R^2	MAPE	RMSE
3	0.833	3.72	4.61	0.880	2.69	3.45	0.534	4.93	6.43
4	0.923	2.63	3.12	0.824	3.75	4.17	0.648	4.68	5.59
5	0.952	2.21	2.47	0.939	2.15	2.45	0.753	4.00	4.68
6	0.953	2.13	2.43	0.947	1.98	2.29	0.75	4.20	4.72
PLS_7	0.990	0.83	1.12	0.332	6.30	8.13	0.39	6.51	7.38
Selected Excitation/Emission Pairs									
3	**Ex310/Em440 Ex310/Em710 Ex440/Em840**								
4	**Ex270/Em470 Ex280/Em530 Ex310/Em710 Ex560/Em590**								
5	**Ex270/Em470 Ex280/Em530 Ex310/Em710 Ex500/Em790 Ex560/Em590**								
6	**Ex270/Em470 Ex280/Em530 Ex290/Em670 Ex310/Em710 Ex500/Em790 Ex560/Em590**								

of the original 1467 fluorescence pairs (less than 0.3%) is able to correctly quantify total sulfur in diesel S10 samples with considerable low errors. In addition, the validation and test metrics are similar to the calibration metrics, signalizing that the model could deal with unseen data satisfactorily. There was no significant improvement between models with 4 and 5 input variables.

As for Diesel S100, PLS regression models were also built using Diesel S10 whole spectral data and the same calibration, validation, and test subsets, and the model metrics can be seen in Table 4. Comparing the AnTSbe and the PLS models, we can see that the later had metrics one order of magnitude higher than the former. Although non-intuitive, R^2 (as defined in Eq. (5)) can assume values between -∞ and 1 [69]. This can happen when the fitted model has predictions that are worse than a horizontal line equal to the mean value of the output in that subset. Generally, R^2 is nonnegative for any linear regression with intercept, and that will always occur in the calibration subset. When using the fitted model to predict the validation and test subset, if the model is overfitted or unable to deal with this unseen data, its predictions can be actually worse than a constant (equal to the mean value of the output). When R^2 is negative, it indicates a complete lack of fit. In the Diesel S10 case, there is a small amount of validation/test samples, and their total sulfur concentration has a very small amplitude within the group. This way, the predictive model must have very small prediction errors to be better than the group means. The AnTSbe models achieved the necessary accuracy, but not the PLS model.

Fig. 6 Presents the measured *vs.* predicted outputs for the Filtered *Ridge* global solution, with model size 4. Finally, Fig. 7 shows the global solutions' (Nw = 4) selected fluorescence pairs plotted upon the average Diesel S10 EEM.

Analyzing Figs. 5 and 7, we can see that both local models selected fluorescence pairs in regions that are far from the peak of fluorescence intensity. This can be explained by remembering that the black-box models fitted here use empirical data without any need for phenomenological significance. Even though those pairs appear to be in a region of noise and no fluorescence, after normalization, they follow linear trends that can be correlated to sulfur in the samples. The algorithm treats all pairs equally, no matter their relative intensity, and seeks the model with the smallest errors, as can be seen by the metrics presented. Pairs, especially the ones between Ex 260 to 400 and Em 300 to 500, could be correlated to the fluorescence of benzothiophenes and dibenzothiophenes observed in other works [64–67]. However, those same works state that the differences in solvents and radicals attached to the compounds can shift the fluorescence peak to other regions. This way, it is hard to directly link fluorescence pairs to sulfur-containing molecules based only on the fitted models. For future works, if the chemical meaning of the fluorescence pairs is relevant, it is possible to spike diesel fuel samples with known diesel sulfur-containing compounds and evaluate models based on this controlled changes. Also, the valid region to select features of the spectra can be trimmed to known areas with high fluorescence intensity.

4.3. Tabu memory activations

To study the impact of the tabu memory list, we follow how many times, on average, a forbidden combination was chosen by the ants throughout all performed optimizations. As expected, some parameters

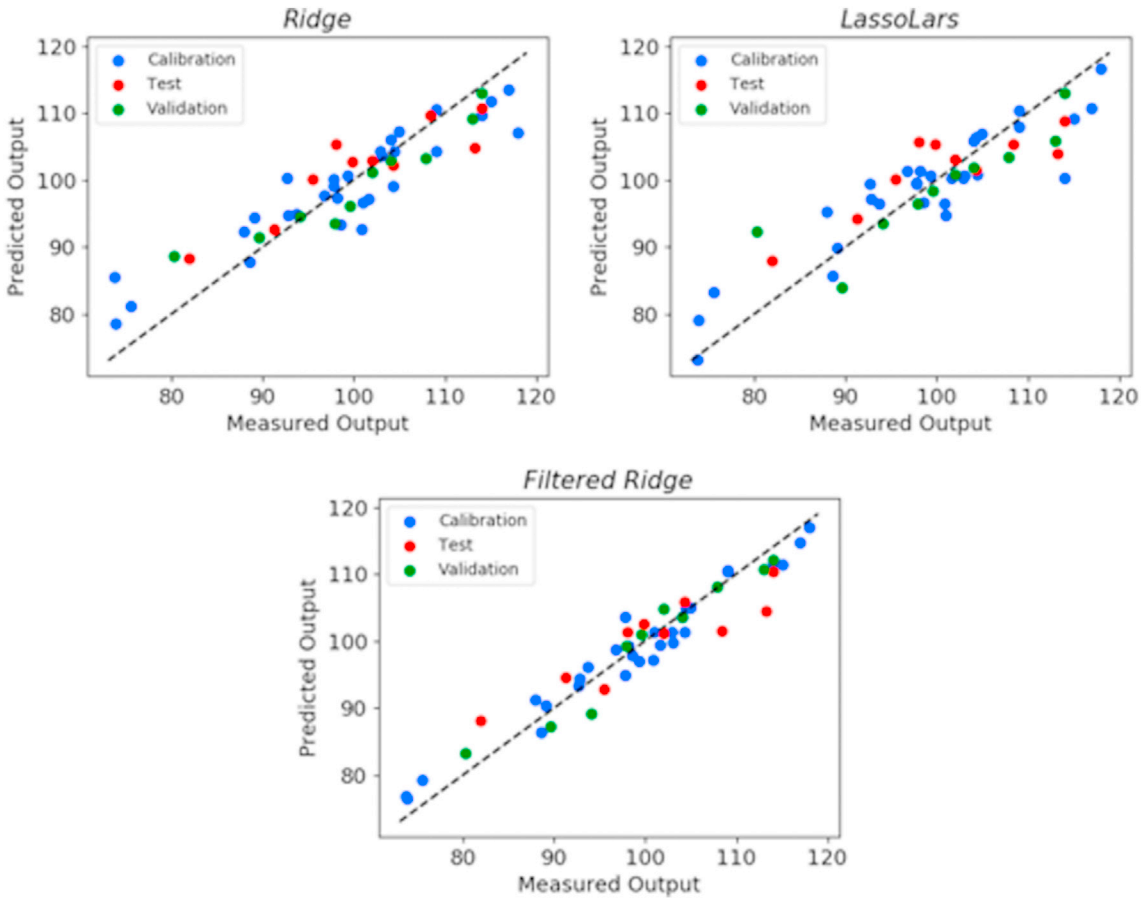

Fig. 4. Diesel S100 Measured *vs* Predicted outputs for the *Ridge, LassoLars,* and Filtered *Ridge* global solutions, with model size 5.

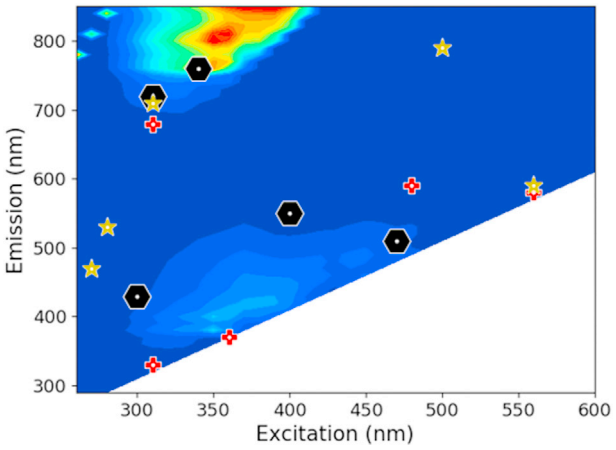

Fig. 5. Global solutions' (*Nw* = 5) selected fluorescence pairs for Diesel S100 (Black hexagon – *LassoLars*; Red cross – *Ridge*; Yellow star – Filtered *Ridge*). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

had a direct correlation with tabu memory activations. First, the number and source of inputs matter: the higher the number of inputs, the rarer will be the situation of the stochastic selection of the same elements by ants. The source is also relevant. In cases like the one presented (using EEM fluorescence), many neighbor inputs carry similar information. In cases where inputs are more linear-independent, the activations would be more frequent. Other very influential parameter is the model size. Smaller models have more chances to activate de tabu memory, and that is corroborated both by the number of possible combination of inputs by permutation (that considerably increases with model size) as by the way the pheromone vector is constructed: the pheromone distribution is uneven in the top performers and more equilibrated in the other inputs. This can be seen in Fig. 8, which illustrates the individually normalized final pheromone trail of optimizations with diesel S100 and model sizes 3, 4, and 5.

After the two first inputs with higher pheromone concentration (inputs 23 and 1416), there is a close competition for the subsequent top performers. This balance in pheromone distribution reflects in the input selection by the ants, meaning that bigger models have smaller chances of selecting forbidden combinations.

For the number of activations of the Tabu memory, in the first optimizations (1467 inputs), model sizes 3, 4, and 5 had 500, 100, and 10 activations, respectively, for diesel S10. Diesel S100 had in general, only a fifth of that – 100, 20, ~1. This difference is due to the higher number of fluorescent-sulfured components present in diesel S100, making the information contained in the spectra more disperse. In the filtered runs (100 inputs), both diesel S10 and S100 had, on average, 6.000, 2.000, and 200 activations for model sizes 3, 4, and 5. Considering that each run fits 30.000 models (200 ants and 150 iterations), 6.000 activations are equivalent to repeating 20% of all fitted models. For the reasons discussed above, in this case study, the tabu memory had no influence in model sizes 5 or bigger.

4.4. Contrast with previous works

Our research group also studied the application of fluorescence spectroscopy as a tool to predict sulfur content in diesel fuel in a previous work [61]. In it, the chosen optimization strategy applied for the selection of fluorescence pairs and the fitting of linear models was called Pure

Table 4
Global solutions' metrics and selected fluorescence pairs for Diesel S10 - *filtered optimization*. PLS regression (4 LV) metrics for comparison.

Diesel S10 Filtered *Ridge* and $\sigma = 1$									
	Calibration			Validation			Test		
Model Size	R^2	MAPE	RMSE	R^2	MAPE	RMSE	R^2	MAPE	RMSE
2	0.889	1.86	0.134	0.813	1.49	0.086	0.784	0.71	0.046
3	0.952	1.25	0.088	0.968	0.62	0.036	0.750	0.72	0.050
4	0.980	0.83	0.057	0.949	0.73	0.045	0.978	0.24	0.015
5	0.986	0.71	0.048	0.994	0.26	0.016	0.991	0.15	0.009
PLS_4	0.997	0.12	0.008	−8.58	9.22	0.619	−5.12	4.19	0.247
Selected Excitation/Emission Pairs									
	Ex310/Em640 **Ex400/Em760**								
2									
3	**Ex260/Em790 Ex340/Em810** Ex390/Em570								
4	**Ex340/Em810** Ex380/Em450 Ex520/Em650 Ex600/Em740								
5	**Ex260/Em790** Ex290/Em450 **Ex340/Em810 Ex400/Em760** Ex410/Em660								

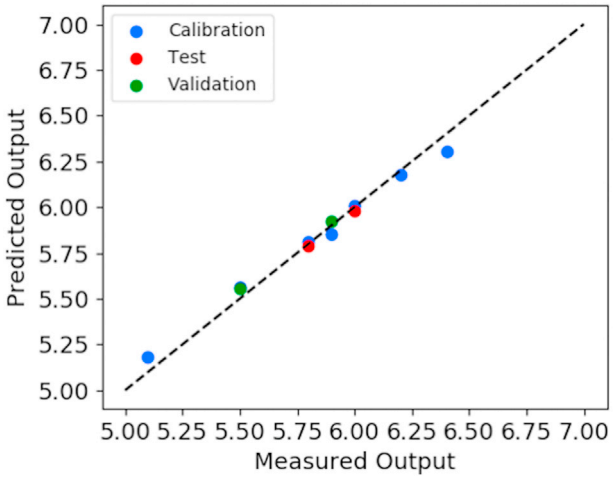

Fig. 6. Diesel S10 Measured *vs.* Predicted outputs for the Filtered *Ridge* global solution, with model size 4.

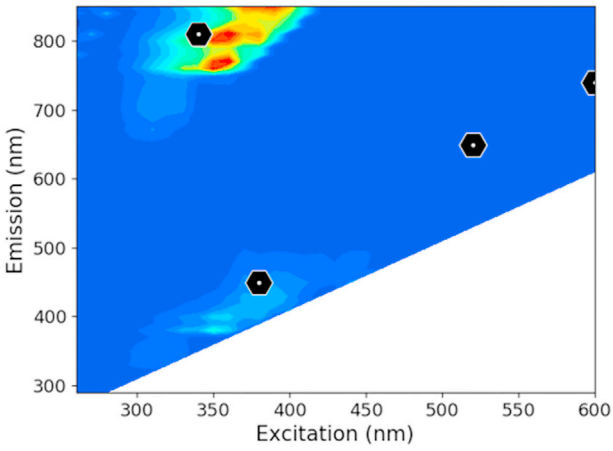

Fig. 7. Global solution' ($Nw = 4$) selected fluorescence pairs for Diesel S10.

Spectra Chemometric Modeling (PSCM): an ACO based algorithm developed for the selection of spectral elements to predict state variables. The AnTSbe can be seen as an evolution of the PSCM, but for general use. The major differences between algorithms are the intended contributions of this work: the introduction of regularized linear models, hybridization with Tabu Search, the adjustable mechanisms of pheromone manipulation, the use of polynomial input expansion, and filtration runs. Both works used the same diesel S10 and S100 datasets, being the only

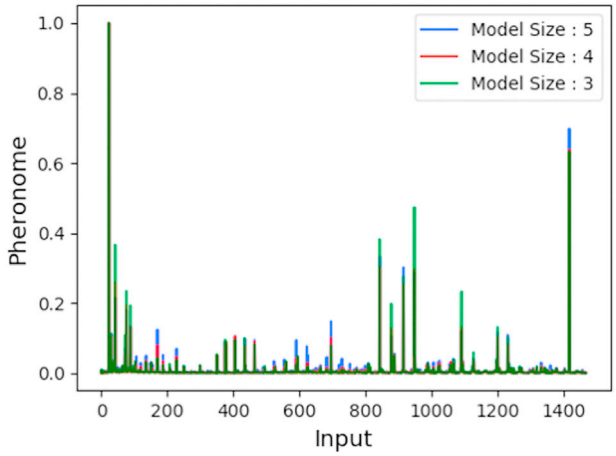

Fig. 8. Individually normalized final pheromone trail of optimizations with Diesel S100 and model sizes 3, 4, and 5.

difference that the PSCM diesel S100 group had one more sample, containing 138 ppm of total sulfur. In addition, the splitting of the data was not the same: The PSCM study applied systematic sampling to split the datasets into only training and test subsets, in a 2:1 proportion. Disregarding these small differences, the results of both works can be reasonably compared. Table 5 presents a compilation of the Global Solutions of both methodologies for diesel S100 and S10, with model sizes 5 and 4, respectively. As can be seen in Table 5, the proposed modifications made a positive impact in the general quality of the final predictive models.

In both studies, the same Diesel S10 dataset was used as a case to evaluate and present the evolution in the methodologies. However, the small number of samples must be taken into account when applying the predictive models to other researches or industrial applications. The authors strongly recommend that future works that intend to predict sulfur concentration in ultra-low-sulfur diesel increase the size of the dataset to reparameterize the presented models or even rerun the algorithm in search of more suitable case-specific models.

5. Conclusions

Fluorescence EEM is a fast and viable source of information that can be correlated to many properties in diesel fuel. Applying the AnTSbe methodology, we were able to optimize the selection of input variables and fit predictive models that use a small amount of fluorescence data to estimate total sulfur concentration in diesel samples.

For the Diesel S100 local models, using only 5 out of the original 1467 fluorescence pairs, combined with bases expansion, we were able to satisfactorily predict total sulfur content in samples with mean absolute

Table 5
Comparison between Global Solutions of AnTSbe and PSCM [61] methodologies for diesel S100 and S10, with model sizes 5 and 4, respectively.

S100	Calibration		Validation		Test	
	R^2	RMSE	R^2	RMSE	R^2	RMSE
$AnTSbe_5$[a]	0.952	2.47	0.934	2.45	0.753	4.68
$PSCM_5$	0.660	7.41	- - -	- - -	0.410	8.94
S10	Calibration		Validation		Test	
	R^2	RMSE	R^2	RMSE	R^2	RMSE
$AnTSbe_4$[b]	0.980	0.057	0.949	0.045	0.978	0.015
$PSCM_4$	0.940	0.09	- - -	- - -	0.690	0.16

[a] Filtered *Ridge* $\sigma = 2$ and[b] Filtered *Ridge* $\sigma = 1$.

prediction errors of less than 4% and root mean squared errors of 4.68 ppm, for the test subset. For the Diesel S10 local models, the use of 4 Ex/Em pairs was sufficient to predict sulfur content with MAPE 0.24%, and RMSE of 0.015 ppm, for the test subset.

Comparing the AnTSbe global solutions with classical PLS regression models and previous optimization studies (PSCM), the proposed methodology proved superior in the task of predicting sulfur content in real diesel fuel samples without the need for any physical or chemical pretreatment. Furthermore, the proposed AnTSbe methodology can deal not only with fluorescence data, but could be used for optimizations with any source of information, as infrared/Raman spectrum, industrial measurements, physicochemical properties, or even a combination of those. The introduction of the tabu memory list was useful to avoid early stagnation and redundant calculations, especially in models with smaller sizes. The technique could be even more fruitful in studies where the input variables are less correlated. The use of bases expansion also proved beneficial for the predictive models.

Using only a small selected fraction of the original spectra, we reduce the time required to acquire data, the influence of noisy inputs that do not carry valid information, and simplify the equipment specification. The proposed methodology is a further step for the construction of custom sensors that could be coupled directly into refinery streams to predict sulfur or even other compounds that present natural fluorescence. This predictive information can be used by operators to take effective controlling actions in hydrodesulphurization processes.

CRediT authorship contribution statement

Lucas Ranzan: Conceptualization, Methodology, Software, Investigation, Writing - original draft. **Luciane F. Trierweiler:** Supervision, Conceptualization, Writing - review & editing. **Jorge O. Trierweiler:** Project administration, Supervision, Conceptualization, Writing - review & editing.

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.

Acknowledgment

The authors are grateful for the scholarship provided by CAPES and would like to thank Petrobras for financial and technical support.

Abbreviations

ACO	Ant Colony Optimization
Ex/Em	excitation/emission
EEM	excitation-emission matrix
HDS	hydrodesulphurization
LV	latent variable
MAPE	mean absolute prediction error
PLS	Partial Least Squares
PCA	Principal Component Analysis
PSCM	Pure Spectra Chemometric Modeling
RMSE	root mean squared error
S10	sample group with less than 10 ppm of sulfur
S100	sample group with average 100 ppm of sulfur
TS	Tabu Search.

References

[1] A. Einipour, Corresponding, A fuzzy-ACO method for detect breast cancer, Global J. Health Sci. 3 (2011), https://doi.org/10.5539/gjhs.v3n2p195.
[2] L. Sun, X. Kong, J. Xu, Z. Xue, R. Zhai, S. Zhang, A hybrid gene selection method based on ReliefF and ant colony optimization algorithm for tumor classification, Sci. Rep. 9 (2019), https://doi.org/10.1038/s41598-019-45223-x.
[3] S. Moro, P. Rita, B. Vala, Predicting social media performance metrics and evaluation of the impact on brand building: a data mining approach, J. Bus. Res. 69 (2016) 3341–3351, https://doi.org/10.1016/j.jbusres.2016.02.010.
[4] T. Beltramo, M. Klocke, B. Hitzmann, Prediction of the biogas production using GA and ACO input features selection method for ANN model, Inf. Process. Agric. 6 (2019) 349–356, https://doi.org/10.1016/j.inpa.2019.01.002.
[5] R. Guidotti, A. Monreale, F. Turini, D. Pedreschi, F. Giannotti, A survey of methods for explaining black box models, ACM Comput. Surv. 51 (2018), https://doi.org/10.1145/3236009.
[6] Front Matter, in: P. Zhang (Ed.), Adv. Ind. Control Technol., William Andrew Publishing, Oxford, 2010, p. iii, https://doi.org/10.1016/B978-1-4377-7807-6.10020-8.
[7] T. Kavzoglu, Chapter 33 - object-oriented random forest for high resolution land cover mapping using quickbird-2 imagery, in: P. Samui, S. Sekhar, V.E. Balas (Eds.), Handb. Neural Comput., Academic Press, 2017, pp. 607–619, https://doi.org/10.1016/B978-0-12-811318-9.00033-8.
[8] B. Johnson, Z. Xie, Classifying a high resolution image of an urban area using super-object information, ISPRS J. Photogrammetry Remote Sens. 83 (2013) 40–49, https://doi.org/10.1016/j.isprsjprs.2013.05.008.
[9] S. Mohapatra, D. Patra, S. Satpathy, An ensemble classifier system for early diagnosis of acute lymphoblastic leukemia in blood microscopic images, Neural Comput. Appl. 24 (2014) 1887–1904, https://doi.org/10.1007/s00521-013-1438-3.
[10] C. Ranzan, A. Strohm, L. Ranzan, L.F. Trierweiler, B. Hitzmann, J.O. Trierweiler, Wheat flour characterization using NIR and spectral filter based on Ant Colony Optimization, Chemometr. Intell. Lab. Syst. 132 (2014) 133–140, https://doi.org/10.1016/j.chemolab.2014.01.012.
[11] L. Ranzan, C. Ranzan, L.F. Trierweiler, J.O. Trierweiler, Classification of diesel fuel using two-dimensional fluorescence spectroscopy, Energy Fuels 31 (2017) 8942–8950, https://doi.org/10.1021/acs.energyfuels.7b00954.
[12] J.A. Sebben, J. da Silveira Espindola, L. Ranzan, N. Fernandes de Moura, L.F. Trierweiler, J.O. Trierweiler, Development of a quantitative approach using Raman spectroscopy for carotenoids determination in processed sweet potato, Food Chem. 245 (2018) 1224–1231, https://doi.org/10.1016/j.foodchem.2017.11.086.
[13] J.I.S. da Silva, A.R. Secchi, AN approach to optimize costs during ultra-low hydrodesulfurization OF a blend consisting OF different oil streams, Braz. J. Chem. Eng. 35 (2018) 1293–1304.
[14] C. Zheng, S. Schwartz, R. Chapkin, R. Carroll, I. Ivanov, Feature selection for high-dimensional integrated data, in: Proc. 2012 SIAM Int. Conf. Data Min., Society for Industrial and Applied Mathematics, 2012, pp. 1141–1150, https://doi.org/10.1137/1.9781611972825.98.
[15] J. Tang, S. Alelyani, H. Liu, Feature selection for classification: a review. Data Classify. Algorithms Appl., 2014, pp. 37–64, https://doi.org/10.1201/b17320.
[16] G. Chandrashekar, F. Sahin, A survey on feature selection methods, Comput. Electr. Eng. 40 (2014) 16–28, https://doi.org/10.1016/j.compeleceng.2013.11.024.
[17] J. Camacho, J. Picó, A. Ferrer, Data understanding with PCA: structural and variance information plots, Chemometr. Intell. Lab. Syst. 100 (2010) 48–56, https://doi.org/10.1016/j.chemolab.2009.10.005.
[18] M.E. Tipping, C.M. Bishop, Probabilistic principal component analysis, J. R. Stat. Soc. Ser. B Statistical Methodol. 61 (1999) 611–622. http://www.jstor.org/stable/2680726.
[19] A.C. da Silva, S.F.C. Soares, M. Insausti, R.K.H. Galvão, B.S.F. Band, M.C.U. de Araújo, Two-dimensional linear discriminant analysis for classification of three-way chemical data, Anal. Chim. Acta 938 (2016) 53–62, https://doi.org/10.1016/j.aca.2016.08.009.
[20] S. Wold, M. Sjöström, L. Eriksson, PLS-regression: a basic tool of chemometrics, Chemometr. Intell. Lab. Syst. 58 (2001) 109–130, https://doi.org/10.1016/S0169-7439(01)00155-1.
[21] J.L. Godoy, J.R. Vega, J.L. Marchetti, Relationships between PCA and PLS-regression, Chemometr. Intell. Lab. Syst. 130 (2014) 182–191, https://doi.org/10.1016/j.chemolab.2013.11.008.
[22] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Kluwer Academic Publishers, Boston, 1989.
[23] L. Hu, C. Yin, S. Ma, Z. Liu, Vis-NIR spectroscopy combined with wavelengths selection by PSO optimization algorithm for simultaneous determination of four quality parameters and classification of soy sauce, Food Anal. Methods. 12 (2019) 633–643, https://doi.org/10.1007/s12161-018-01407-1.

[24] C.M. Pessoa, C. Ranzan, L.F. Trierweiler, J.O. Trierweiler, Development of ant colony optimization (ACO) algorithms based on statistical analysis and hypothesis testing for variable selection, IFAC-PapersOnLine. 48 (2015) 900–905, https://doi.org/10.1016/j.ifacol.2015.09.084.
[25] M. Dorigo, C. Blum, Ant colony optimization theory: a survey, Theor. Comput. Sci. 344 (2005) 243–278, https://doi.org/10.1016/j.tcs.2005.05.020.
[26] T. Stützle, M. López-Ibáñez, M. Dorigo, A concise overview of applications of ant colony optimization. https://doi.org/10.1002/9780470400531.eorms0001, 2011.
[27] B. Chandra Mohan, R. Baskaran, A survey: ant Colony Optimization based recent research and implementation on several engineering domain, Expert Syst. Appl. 39 (2012) 4618–4627, https://doi.org/10.1016/j.eswa.2011.09.076.
[28] B. Bullnheimer, R. Hartl, C. Strauss, A new rank based version of the ant system - a computational study, Cent. Eur. J. Oper. Res. 7 (1999) 25–38.
[29] L. Gambardella, M. Dorigo, Ant-Q: a Reinforcement Learning approach to the traveling salesman problem, Proc. ML-95, 12th Int. Conf. Mach. Learn. 170 (2000), https://doi.org/10.1016/B978-1-55860-377-6.50039-6.
[30] K. Ku-Mahamud, M. Alobaedy, New Heuristic Function in Ant Colony System Algorithm for Optimization, 2013.
[31] C. Blum, M. Blesa, A. Roli, M. Sampels, Hybrid metaheuristics: an emerging approach to optimization. https://doi.org/10.1007/978-3-540-78295-7, 2008.
[32] C. Blum, J. Puchinger, G.R. Raidl, A. Roli, Hybrid metaheuristics in combinatorial optimization: a survey, Appl, Soft Comput 11 (2011) 4135–4151, https://doi.org/10.1016/j.asoc.2011.02.032.
[33] X. Zhang, L. Tang, A new hybrid ant colony optimization algorithm for the traveling salesman problem, in: D.-S. Huang, D.C. Wunsch, D.S. Levine, K.-H. Jo (Eds.), Adv. Intell. Comput. Theor. Appl. With Asp. Artif. Intell., Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 148–155.
[34] M.M.S. Abdulkader, Y. Gajpal, T.Y. ElMekkawy, Hybridized ant colony algorithm for the multi compartment vehicle routing problem, Appl. Soft Comput. 37 (2015) 196–203, https://doi.org/10.1016/j.asoc.2015.08.020.
[35] S. Balseiro, I. Loiseau, J. Ramonet, An ant colony algorithm hybridized with insertion heuristics for the time dependent vehicle routing problem with time windows, Comput. Optic 38 (2011) 954–966, https://doi.org/10.1016/j.cor.2010.10.011.
[36] X.Y. Li, P. Tian, S.C.H. Leung, An ant colony optimization metaheuristic hybridized with tabu search for open vehicle routing problems, J. Oper. Res. Soc. 60 (2009) 1012–1025, https://doi.org/10.1057/palgrave.jors.2602644.
[37] A. Serbencu, V. Minzu, Hybridized ant colony system for tasks to Workstations assignment, IEEE Symp. Ser. Comput. Intell., 2016, pp. 1–7, https://doi.org/10.1109/SSCI.2016.7850060.
[38] P.B. Myszkowski, M.E. Skowroński, Ł.P. Olech, K. Oślizło, Hybrid ant colony optimization in solving multi-skill resource-constrained project scheduling problem, Soft Comput 19 (2015) 3599–3619, https://doi.org/10.1007/s00500-014-1455-x.
[39] F. Arito, G. Leguizamón, Incorporating tabu search principles into ACO algorithms. https://doi.org/10.1007/978-3-642-04918-7_10, 2009.
[40] S. Tsutsui, N. Fujimoto, ACO with tabu search on a GPU for solving QAPs using move-cost adjusted thread assignment. https://doi.org/10.1145/2001576.2001785, 2011.
[41] Y. Niu, Z. Yang, P. Chen, J. Xiao, A hybrid tabu search algorithm for a real-world open vehicle routing problem involving fuel consumption constraints, Complexity (2018) 1–12, https://doi.org/10.1155/2018/5754908.
[42] H. Pirim, E. Bayraktar, B. Eksioglu, Tabu search: a comparative study. Tabu Search, I-Tech Education and Publishing, 2008, https://doi.org/10.5772/5637.
[43] E.L. Russell, L.H. Chiang, R.D. Braatz, Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis, Chemometr. Intell. Lab. Syst. 51 (2000) 81–93, https://doi.org/10.1016/S0169-7439(00)00058-7.
[44] M. Mansouri, M. Sheriff, R. Baklouti, M. Nounou, H. Nounou, A. Ben Hamida, N. Karim, Statistical fault detection of chemical process - comparative studies, J. Chemical (2016), https://doi.org/10.4172/2157-7048.1000282.
[45] P.V.J.L. Santos, L. Ranzan, M. Farenzena, J.O. Trierweiler, K. Rank, AN evolution OF Y-rank for multiple solutions problem, Braz. J. Chem. Eng. 36 (2019) 409–419.
[46] D. Raschka, Python machine learning, Packt Publishing Ltd., Birmingham B3 2 PB, UK., 2015.
[47] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, Scikit-learn: machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[48] M. Dorigo, L.M. Gambardella, Ant colonies for the travelling salesman problem, Biosystems 43 (1997) 73–81, https://doi.org/10.1016/s0303-2647(97)01708-5.
[49] R. Sagban, K. Ku-Mahamud, M. Abu Bakar, Reactive memory model for ant colony optimization and its application to TSP. https://doi.org/10.1109/ICCSCE.2014.7072736, 2014.
[50] J. Kluabwang, D. Puangdownreong, S. Sujitjorn, Multipath adaptive tabu search for a vehicle control problem, J. Appl. Math. 2012 (2012), https://doi.org/10.1155/2012/731623.
[51] A.N. Tikhonov, V.I.A. Arsenin, Solutions of Ill-Posed Problems, Winston, 1977. https://books.google.com.br/books?id=ECrvAAAAMAAJ.
[52] L. Massaron, A. Boschetti, Boschetti, Regression Analysis with Python, Packt Publishing Ltd., Birmingham B3 2PB, UK, 2016, 8770.
[53] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, Least angle regression, Ann. Stat. 32 (2004) 407–499, https://doi.org/10.1214/009053604000000067.
[54] V. Chandra Srivastava, An evaluation of desulfurization technologies for sulfur removal from liquid fuels, RSC Adv. 2 (2012) 759–783, https://doi.org/10.1039/c1ra00309g.
[55] Y. Murakami, Analysis of corrosive wear of diesel engines: relationship to sulfate ion concentrations in blowby and crankcase oil, JSAE Rev. 16 (1995) 43–48. http://www.scopus.com/inward/record.url?eid=2-s2.0-0029230944&partnerID=40&md5=5efb46a1b319782f1c1ff982a30118cb.
[56] F. Lima, J. Gouvenaux, L.C. Branco, A.J.D. Silvestre, I.M. Marrucho, Towards a sulfur clean fuel: deep extraction of thiophene and dibenzothiophene using polyethylene glycol-based deep eutectic solvents, Fuel 234 (2018) 414–421, https://doi.org/10.1016/j.fuel.2018.07.043.
[57] S. Brunet, D. Mey, G. Pérot, C. Bouchy, F. Diehl, On the hydrodesulfurization of FCC gasoline: a review, Appl. Catal. Gen. 278 (2005) 143–172, https://doi.org/10.1016/j.apcata.2004.10.012.
[58] ASTM, D4294-10 Standard Test Method for Sulfur in Petroleum and Petroleum Products by Energy Dispersive X-Ray Fluorescence Spectrometry, 2010, https://doi.org/10.1520/D4294-10.
[59] P. Aburto, K. Zuñiga, J. Campos-Terán, J. Aburto, E. Torres, Quantitative analysis of sulfur in diesel by enzymatic oxidation, steady-state fluorescence, and linear regression analysis, Energy Fuels 28 (2014) 403–408. http://www.scopus.com/inward/record.url?eid=2-s2.0-84892762748&partnerID=40&md5=129a874180bf234fa616ad3558a0bd0a.
[60] A.T. Campos, C.M. Quintella, M. Meira, S. Luna, Prediction of sulfur content in diesel/biodiesel blends using LED-induced fluorescence associated with multivariate calibration, J. Braz. Chem. Soc. 29 (2018) 1367–1372.
[61] C. Ranzan, L. Ranzan, L.F. Trierweiler, J.O. Trierweiler, Sulfur determination in diesel using 2D fluorescence spectroscopy and linear models, IFAC-PapersOnLine. 48 (2015) 415–420, https://doi.org/10.1016/j.ifacol.2015.09.003.
[62] R. Hua, Y. Li, W. Liu, J. Zheng, H. Wei, J. Wang, X. Lu, H. Kong, G. Xu, Determination of sulfur-containing compounds in diesel oils by comprehensive two-dimensional gas chromatography with a sulfur chemiluminescence detector, J. Chromatogr., A 1019 (2003) 101–109. http://www.scopus.com/inward/record.url?eid=2-s2.0-0142165009&partnerID=40&md5=530e9bdc6c5d4b812ccfa7c32815740c.
[63] F.C.Y. Wang, W.K. Robbins, F.P. Di Sanzo, F.C. McElroy, Speciation of sulfur-containing compounds in diesel by comprehensive two-dimensional gas chromatography, J. Chromatogr. Sci. 41 (2003) 519–523. http://www.scopus.com/inward/record.url?eid=2-s2.0-0344899991&partnerID=40&md5=9a1a60c42cedb8e778b86f8acd0e4108.
[64] J.J. Aaron, Z. Mechbal, A. Adenier, C. Parkanyi, V. Kozmik, J. Svoboda, Luminescence properties of new fused benzothiophene derivatives and their conductive oligomers structural and solvent effects. J. Fluoresc., Springer, 2002, pp. 231–239, https://doi.org/10.1023/A:1016869002735.
[65] Y. Hou, Y. Li, Y. Liu, G. Li, Z. Zhang, Effects of polycyclic aromatic hydrocarbons on the UV-induced fluorescence spectra of crude oil films on the sea surface, Mar. Pollut. Bull. 146 (2019) 977–984, https://doi.org/10.1016/j.marpolbul.2019.07.058.
[66] P.K. Nayak, N. Agarwal, N. Periasamy, Synthesis, Photophysical and Electrochemical Properties of 2,8-Diaryl-Dibenzothiophene Derivatives for Organic Electronics, 2010.
[67] A. Bree, R. Zwarich, Electronic spectra of dibenzothiophene, Spectrochim. Acta Part A Mol. Spectrosc. 27 (1971) 621–630, https://doi.org/10.1016/0584-8539(71)80264-7.
[68] ASTM ASTM, D7039 - 15a standard test method for sulfur in gasoline, diesel fuel, jet fuel, kerosine, biodiesel, biodiesel blends, and gasoline-ethanol blends by monochromatic wavelength dispersive X-ray fluorescence spectrometry. https://www.astm.org/Standards/D7039.htm, 2015 accessed May 5, 2020.
[69] T.O. Kvålseth, Cautionary note about R2, Am. Statistician 39 (2012) 279–285, https://doi.org/10.1080/00031305.1985.10479448.