Contents lists available at ScienceDirect

Computers and Geosciences

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

Research paper

Probabilistic logging lithology characterization with random forest probability estimation

Yile Ao [a,1], Liping Zhu [a,*,2], Shuang Guo [b,3], Zhongguo Yang [c,3]

[a] China University of Petroleum-Beijing, Changping, Beijing, China
[b] China University of Petroleum (Karamay Campus), Karamay, Xingjiang, China
[c] North China University of Technology, Shijingshan, Beijing, China

ARTICLE INFO

Keywords:
Machine learning
Classification probability estimation
Probabilistic random forest
Logging lithology interpretation
Fuzzy lithology characterization journal:
Computers & Geoscience

ABSTRACT

Borehole lithology discrimination is the foundation of formation evaluation and reservoir characterization. Due to the limitation of costing or accuracy, direct discrimination methods such as borehole core and drilling cutting analysis are unable to be deployed to every well, while logging lithology interpretation provides an alternative solution for this. Recently, several machine learning algorithms such as the neural network, support vector machine, decision tree, and random forest have already been employed by researchers for automatic logging lithology interpretation. However, the vast majority of these studies belong to the category of deterministic lithology characterization. In this article, we propose a probability based fuzzy characterization method for more effective logging lithology interpretation. Moreover, to improve the accuracy of lithology probability estimation, we propose the probabilistic random forest algorithm and investigate its advantages referred to 8 existing probability estimation algorithms. Through the comparative experiments on 9 real-world logging lithology interpretation tasks, the feasibility and advantage of the proposed method are confirmed. Application case demonstrates that compared with traditional deterministic lithology characterization methods, probabilistic lithology characterization is able to provide more information about rhythm, heterogeneity, and formation properties, which worths further application and promotion to improve the fineness of formation evaluation and reservoir characterization.

1. Introduction

As the foundation of formation evaluation and reservoir characterization, the discrimination of the borehole lithology is extremely important for petroleum exploration and engineering. Through the observation and analysis of borehole core or drilling cutting, lithology conclusions of subsurface formations are able to obtain directly. However, borehole cores are generally expensive and are often only available for limited key wells, while drill cuttings have drawbacks in measuring depth accuracy and thin layer resolution. As a result, lithology columns based on those methods are insufficient for wide applications to all wells.

Since logging curves are usually available for all wells, logging lithology interpretation provides an alternative way for this. Early studies (Burke et al., 1969; Porter et al., Whitmanet al.; Clavier and Rust et al.; Serra et al., 1985) attempted to establish theoretical mappings between logging responses and formation lithology based on the principle of logging measurement, which formed the theoretical foundation (Serra, 1983) of logging lithology interpretation. However, these theoretical relationships are often established based on idealized theoretical assumptions, which will bring subjective bias inevitably and lead to a greatly discounted in accuracy and feasibility. To mitigate the influence of subjective bias, statistical or machine learning algorithms are employed to establish empirical mapping models under the calibration of cuttings or cores. From the perspective of pattern recognition, this is equivalent to a typical classification task from logging responses to lithology types. Until now, several algorithms such as statistical discriminant (Bosch et al., 2002), neural network (Shao et al., 2008),

* Corresponding author.
E-mail address: zhuliping@cup.edu.cn (L. Zhu).
[1] The proponent and implementer of the presented method in this article.
[2] The supervisor of studies, who provides macro guidance for the completion of this article.
[3] Data collation and experiment facilitator, who also gives many suggestions for the writing of this article.

https://doi.org/10.1016/j.cageo.2020.104556
Received 18 September 2018; Accepted 18 June 2020

fuzzy logic (Bosch et al., 2013a), support vector machine (Deng et al., 2017), decision tree (Tan et al., 2010), boosted trees (Kadkhodaie Ilkhchi et al., 2010) and random forest (Xie et al., 2018) have already been applied for this purpose, while the potential of employing machine learning techniques to assist the process of logging lithology interpretation is confirmed.

However, the vast majority of these applications all intend to give deterministic lithology conclusions, which has no ability to reflect gradual lithology transitions and subtle in-layer formation property differences. In this article, we demonstrate that fuzzy lithology characterization using posterior classification probability for borehole lithology characterization is able to provide more information about rhythm, heterogeneity, and formation properties. Moreover, to improve the accuracy of lithology probability estimation, we propose the *Probabilistic Random Forest* (PrRF) algorithm and investigate its feasibility and advantage compared to 8 existing algorithms in well logging based lithology probability estimation.

The remainder of this article is structured as follows. After this brief introduction, we summarize the state-of-art of machine learning assisted logging lithology interpretation in section 2. Then section 3 presents the methodologies of existing classification probability estimation algorithms and the proposed probabilistic random forest algorithm. In section 4, the feasibility and advantage of the probabilistic random forest referred to 8 existing algorithms in probabilistic logging lithology characterization are illustrated by experiments on 9 real-world tasks of different areas. Then demonstrated by a real-world application case, the practicality and advantages of PrRF based probabilistic logging lithology characterization are revealed in section 5. Finally, section 6 presents the conclusions of our study.

. t d or

Applying statistical or machine learning algorithms to assist the interpretation of borehole lithology has a long history. As early as the 1980s, researchers (Delfiner et al., 1987; Busch et al., 1987; Ingerman, 1995) began to use statistical discriminant algorithms (such as *Linear Discriminant Analysis* (LDA) (Fisher, 1938) and *Quadratic Discriminant Analysis* (QDA) (Devijver and Kittler, 1982)) instead of manual analysis to conduct logging lithology interpretation. Although the implementations of these algorithms are relatively simple, their methodologies are consistent with the principle of machine learning assisted logging lithology. However, statistical algorithms often make different degrees of normal distribution hypothesis for the logging responses of each lithology type, which is difficult to be satisfied in practice. With the development of machine learning, multifarious more sophisticated algorithms were proposed, which provided alternative ways for better logging lithology modeling.

The first used machine learning algorithm in logging lithology interpretation belongs to the family of *Neural Networks* (NNET) (Rogers et al., 1992). Advantages of neural networks referred to statistical discriminant algorithms are confirmed by many studies (Wong et al., 1995; Benaouda et al., 1999; Dubois et al., 2007). In addition to the most representative back propagation neural network algorithm, neural fuzzy system (a hybrid of neural network and fuzzy expert system) (Chang et al., 1997), neural network with feature construction (Zhang et al., 1999), modular neural network (Bhatt and Helle, 2002), fuzzy neural network (Qi and Carr, 2006), self-adapting neural networks (Maiti et al., 2007; Wang and Zhang, 2008) have also been applied later. Since neural network algorithms have been widely investigated in logging lithology interpretation and recognized by the industry, it is often accepted as a benchmark in comparative studies of algorithms (Dubois et al., 2007; Kadkhodaie Ilkhchi et al., 2010; Horrocks et al., 2015).

Support Vector Machine (SVM) is another widely applied algorithms in logging lithology interpretation (Sebtosheikh et al., 2015; Mou and Wang, 2015; Deng et al., 2017; Al-Anazi and Gates, 2010). Comparative studies in the context of logging lithology classification show that SVM has significant advantages referred to statistical discrimination and neural networks (Al-Anazi and Gates, 2010; Cheng et al., 2010). Meanwhile, support vector machine with different feature extractions (Li et al., 2010a), support vector machine with global optimization arithmetic difference evolutionary (Annan and Lu, 2009), *Least Squares Support Vector Machine* (LSSVM) with particle swarm optimization (Cheng et al., 2010) and coupled simulated annealing optimization (Salehi and Honarvar, 2014) were proposed and applied in logging lithology discrimination later.

Recently, more algorithms such as *Fuzzy Logic* (FL) (Hsieh et al., 2005; Kadkhodaie Ilkhchi et al., 2010; Bosch et al., 2013b), *Decision Tree* (DT) (Li et al., 2010b; Tan et al., 2010; Xiongyan et al., 2012; Zhang et al., 2012) and ensemble methods (such as *LogitBoost* (Kadkhodaie Ilkhchi et al., 2010; Dev and Eden, 2018), *AdaBoost* (Xie et al., 2018; Tewari and Dwivedi), and *Random Forest* (RF) (Xie et al., 2018; Tewari and Dwivedi)) are also employed to assist the process of logging lithology interpretation. In addition to applying new algorithms, integrating and combining existing algorithms provides another way to build more complex but accurate models (de Oliveira et al., 2013). A representation of them is the *Committee Machine* (CM) (Masoudnia and Ebrahimpour, 2014), which is an integration of multiple classifiers (also named *Mixture of Experts*). For example, a committee machine of naive Bayes, support vector machine, and neural network are present by Horrocks and Holden et al. (Horrocks et al., 2015) for automated lithology interpretation. Extremely, Gifford and Agah (2010) proposed a committee machine with 11 different based algorithms for the lithology/rock-facies interpretation of wells. Besides, standard ensemble learning technique such as bagging, arcing, and boosting of neural networks (Santos et al., 2003) and support vector machines (Leite et al., 2013) are also applied for logging lithology modeling.

In order to clarify the advantages and disadvantages of machine learning algorithms in logging lithology discrimination, several comparative studies are executed. The conclusions of them are summarized in Table 1. Although the algorithms compared and the datasets used are different, some common conclusions can still be obtained: using more sophisticated machine learning algorithms (NNET or SVM) always outperform simple algorithms such as LDA, QDA, KDA (Kernel Discriminant Analysis), KNN (K Nearest Neighborhoods), and NBAYES (Naive Bayes) (Benaouda et al., 1999; Dubois et al., 2007; Al-Anazi and Gates, 2010), while the advantages of decision tree based ensemble methods such as boosted trees and random forest are confirmed (Kadkhodaie Ilkhchi et al., 2010; Xie et al., 2018; Tewari and Dwivedi). However, these are only conclusions for deterministic lithology discrimination, and it is still worthwhile to investigate whether tree-based ensemble methods are able to maintain its advantages in fuzzy lithology characterization. In this article, referred to the famous

1

Summary of algorithm comparisons for logging lithology interpretation.

Citation	Compared Algorithms	Winner
Benaouda 1999 (Benaouda et al., 1999)	LDA, QDA, KDA, NNET	NNET
Santos 2003 (Santos et al., 2003)	NNET, Bagging NNET, Arcing NNET	Arcing NNET
Dubois 2007 (Dubois et al., 2007)	QDA, FL, KNN, NNET	NNET
Kadkhodaie 2010 (Kadkhodaie Ilkhchi et al., 2010)	FL, NNET, LogitBoost	LogitBoost
Al-Anazi 2010 (Al-Anazi and Gates, 2010)	LDA, PNN, SVM	SVM
Zhang 2012 (Zhang et al., 2012)	Association Rule, DT, SVM	DT
Xie 2018 (Xie et al., 2018)	NBAYES, NNET, SVM, Boosted Trees, RF	Boosted Trees, RF
Tewari 2018 (Tewari and Dwivedi)	Bagging, AdaBoost, Rotation Forest, Subspace, DECORATE	Rotation Forest

random forest algorithm, we propose the *Probabilistic Random Forest* (PrRF) algorithm and investigate its feasibility and advantage in probabilistic lithology characterization, which will fill this gap in related studies.

. t odo o

In this section, the basic concepts of probabilistic lithology characterization are presented at first, followed by a brief introduction of existing probability estimation algorithms. Then the algorithms of the proposed probabilistic decision tree and probabilistic random forest are presented in detail.

3.1. Probabilistic lithology characterization

Probabilistic lithology characterization refers to the technology of using lithology probabilities instead of deterministic lithology conclusions to describe the distribution of formation lithology. As mentioned earlier, using probabilistic lithology characterization is able to provide more information than deterministic discrimination for many circumstances. Specifically, there are two situations that make probabilistic characterization more effective than deterministic discrimination:

1. *Gradient Interfaces*: For formations with shale-sand or glutenite lithology sequences, there are a large number of positive or negative rhythms. For these cases, formation lithologies are often a smooth transition rather than a direct mutation since the mineral composition or granularity change gradually. Deterministic discrimination lacks the ability to reflect these gradual changes, while fuzzy characterization is able to describe these gradual changes by the transition of lithology probabilities.

2. *Formation Heterogeneity*: Influenced by the heterogeneity of formations, there are still several fluctuations of composition and granularity even for a single lithology layer. However, these difference are not strong enough to change the conclusion of deterministic lithology discriminations. For probabilistic fuzzy characterization, these small changes will cause fluctuations of lithology probabilities, which enables it to describe the heterogeneity phenomenon to a certain extent.

In our proposed method, we employ the posterior classification probabilities for fuzzy lithology characterization. Assuming that there are K kinds of lithology types $(Y_1, Y_2, \ldots, Y_K)$, denote the input logging responses as vector $\mathbf{x}$, then for the kth lithology its posterior classification probability is $Prob(y = Y_k \mid \mathbf{x})$, which represents that when logging responses are $\mathbf{x}$, how large is the probability that this sample belongs to the kth lithology. The prediction of posterior classification probabilities can be performed by a supervised multi-classes probabilities estimation task. After the learning of the specified algorithm, a multi-class probability estimation model $\mathscr{P}$ is established, which consists of K approximation functions $\mathscr{P}_1(\mathbf{x}), \mathscr{P}_2(\mathbf{x}), \ldots \mathscr{P}_K(\mathbf{x})$ for the prediction of posterior classification probabilities. Theoretically speaking, these approximation functions satisfy:

$$\begin{cases} \mathscr{P}_i(\mathbf{x}) \approx P \quad b(\quad = Y_i \mid \mathbf{x}) \\ \mathscr{P}_i(\mathbf{x}) \in [0.0,\ 1.0] \\ \sum_{i=1}^{K} \mathscr{P}_i(\mathbf{x}) = 1.0 \end{cases} \tag{1}$$

The classification probabilities estimation task has a very close relationship with the classification task. In fact, probability estimation functions $\mathscr{P}_1(\mathbf{x}), \mathscr{P}_2(\mathbf{x}), \ldots \mathscr{P}_K(\mathbf{x})$ are able to form a classifier:

$$\mathscr{F}(\mathbf{x}) = a \underset{i \in Y_1, \ldots, Y_K}{\quad a_i} (\mathscr{P}_i(\mathbf{x})) \tag{2}$$

which discriminates the sample $\mathbf{x}$ as the class with the highest posterior probability estimation. In general, the goodness of fit for a learned probability estimation model $\mathscr{P}$ is evaluated from two aspects: 1) the probabilities fitting error of $\mathscr{P}_1(\mathbf{x}), \mathscr{P}_2(\mathbf{x}), \ldots \mathscr{P}_K(\mathbf{x})$ for each class; 2) the effectiveness of constructed classifier $\mathscr{F}(\mathbf{x})$. However, for the samples of logging lithology discrimination, the true values of posterior classification probability $Prob(y = Y_k \mid \mathbf{x})$ are availablev

For statistical probability estimation methods, the main drawback sources from the fact that they all estimate the classification probabilities based on parameterized distribution assumptions. If the assumption for each class is satisfied, then these algorithms will give very accurate probability estimations. However, if these assumptions have deviated seriously, the effectiveness of the their probability estimations cannot be guaranteed.

3.2.2. Kernel density estimation

In order to mitigate the bias caused by parameterized distribution assumptions, *Kernel Density Estimation* (KDE) (Silverman, 1986) provides a nonparametric way for the estimation of probability density functions. KDE is firstly introduced for univariate density estimation (Rosenblatt, 1956; Parzen, 1962) and then generalized to multivariate applications (Silverman, 1986; Simonoff, 1996). For N samples $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, the kernel density estimation of their probability density function is:

$$\mathscr{P}_{\mathscr{K}}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\mathscr{K}(\mathbf{x}-\mathbf{x}_i) \tag{6}$$

In the above definition, $\mathscr{K}(\mathbf{x})$ is the pre-selected kernel function. For multivariate context, the most widely used kernel function is the *multivariate gaussian kernel*, which is defined as:

$$\mathscr{K}(\mathbf{x}) = \frac{e^{\left(-\frac{1}{2}\mathbf{x}^T\mathbf{H}^{-1}\mathbf{x}\right)}}{\sqrt{(2\pi)^P|\mathbf{H}|}} \tag{7}$$

where $\mathbf{H}$ represents the *bandwidth matrix* and P is the dimensions number of inputs. For KDE based classification probability estimation, the definition in Eq (6) is substituted into Eq (5) for nonparametric approximations of $Prob(\mathbf{x} \mid y = Y_k)$ and $Prob(\mathbf{x})$, then the posterior classification probabilities are predicted in the same way with statistical probability estimation methods.

The successful application of KDE depends on the selection of kernel function (or the bandwidth matrix $\mathbf{H}$ if we use the multivariate gaussian kernel). Even though Silverman (1986) suggested some rules of thumb, it's still a hard task for the selection of kernel function in practice. Besides, as can be observed from Eq (6), the kernel function $\mathscr{K}(\mathbf{x})$ will be applied to every sample to compute its contribution. When the count

partition process. For each partition, denoting the input space for parent node $\mathcal{N}_C$ as $\mathbb{X}_C$, PrDT tries to find the best split to partition $\mathbb{X}_C$ into two subspace $\mathbb{X}_L$ and $\mathbb{X}_R$, which is corresponding to the left child node $\mathcal{N}_L$ and the right child node $\mathcal{N}_R$. Denotes the split based on the pth input by specified threshold v as $S_{p,v}$, for numerical inputs such as logging curves, $S_{p,v}$ is expressed as:

$$S_{,v}: \mathbb{X}_L = \{ \quad < v \mid \mathbf{x} \in \mathbb{X}_C \}; \mathbb{X}_R = \{ \quad \geq v \mid \mathbf{x} \in \mathbb{X}_C \} \tag{8}$$

Actually, $S_{p,v}$ is equivalent to an IF-ELSE rule. For each subspace, the classification probabilities are estimated by the proportion of samples in each class, which provide simple probability estimation model for samples in this subspace. During the training phase, a cascaded logic chain is learned and stored as a binary tree by recursive binary partition. Denote the probability estimation model for $\mathbb{X}_L$ and $\mathbb{X}_R$ as $\mathscr{P}_L$ and $\mathscr{P}_R$ respectively, then the split for each node is optimized with the minimization of joint impurity, which is defined as:

$$JI(S_{,v}) = \frac{||\mathbb{X}_L||}{||\mathbb{X}_C||} APE(\mathscr{P}_L) + \frac{||\mathbb{X}_R||}{||\mathbb{X}_C||} APE(\mathscr{P}_R) \tag{9}$$

In the above definition of Eq. (9), $||\mathbb{X}||$ represents the sample count in input space $\mathbb{X}$, while $APE(\mathscr{P}_L)$ and $APE(\mathscr{P}_R)$ are the averaged probability error for model $\mathscr{P}_L$ and $\mathscr{P}_R$. It's easy to see that the split with the minimum $JI(S_{p,v})$ will bring the highest reduction of $APE(\mathscr{P})$ of $\mathbb{X}_C$. Thus, the selection of $S_{p,v}$ is equivalent to the minimization of $APE(\mathscr{P})$ for the current node. By traversing every possible split in all the P inputs, the best split with the minimum joint impurity will be selected for the partition of $\mathbb{X}_C$. Then for the next steps, $\mathbb{X}_L$ and $\mathbb{X}_R$ will be regarded as the parent nodes for further partition recursively, until any of the following conditions are met:

1. No split is necessary since the target values of all samples in $\mathbb{X}_C$ are the same.
2. No split is available since input values of all samples in $\mathbb{X}_C$ are the same.
3. The tree depth exceeds the preset maximum depth limitation.
4. The samples count in $\mathbb{X}_C$ is less than the preset minimum leaf size limitation.

After fully growth with optimization, the whole input space is partitioned into L subspace $\mathbb{X}_1 \ldots \mathbb{X}_L$ defined by L leave nodes in the tree structure. The for the whole tree, the classification probability estimation for the kth class is:

$$\mathscr{P}_{ee}\left(\quad = Y_{\cdot} \,\middle|\, \mathbf{x} \right) = \sum_{=1}^{L} \left(I\left(\mathbf{x} \in \mathbb{X}_{\cdot} \right) \frac{\sum_{=1}^{N} I\left(_{,} = Y_{\cdot} \,\middle|\, \mathbf{x}_{,} \in \mathbb{X}_{\cdot} \right)}{\sum_{=1}^{N} I\left(\mathbf{x}_{,} \in \mathbb{X}_{\cdot} \right)} \right) \tag{10}$$

For the observation with input vector $\mathbf{x}^*$, by substituting $\mathbf{x}^*$ into the logic chain to find out the corresponding leaf node and subspace, the classification probability for each class is estimated by Eq (10). It's easy to see that $\mathscr{P}_{tree}$ is a piecewise constant function, which may be too rough referred to the true probability function. Besides, PrDT models are very unstable. A small change in the training samples will engender a significant change in the tree structure. However, with the benefits of ensemble learning (Zhou, 2012), these drawbacks can be eliminated to a certain extent.

3.3.2. Probabilistic random forest

The formal definition of random forest was first made by Breiman (2001) in 2001, which is a bagging of uncorrelated decision trees learned with randomized node optimization. The probabilistic random forest algorithm follows almost the same formula of Breiman's random forest, and the only difference is that PrRF uses the randomized PrDT instead of the randomized CART as the base algorithm.For the construction of a PrRF model, M sample subsets are generated from the whole sample set $\mathbb{S}$ by bootstrap sampling at first, then M randomized PrDTs are constructed from these subset independently and then integrated as a bagging ensemble. The randomness of these trees is embodied in two aspects:

- *Random Sample Selection*: The construction of the mth decision tree $\mathscr{T}_m(\mathbf{x})$ is based on a randomly generated sample subset $\mathbb{S}_m$, which is obtained by bootstrap sampling from the whole sample set $\mathbb{S}$.
- *Random Feature Selection*: During the partition of tree nodes, instead of traversing every possible split in all P inputs, the randomized tree only consider splits in a randomly selected input subset. The size of this subset is specified by the user defined hyper-parameter.

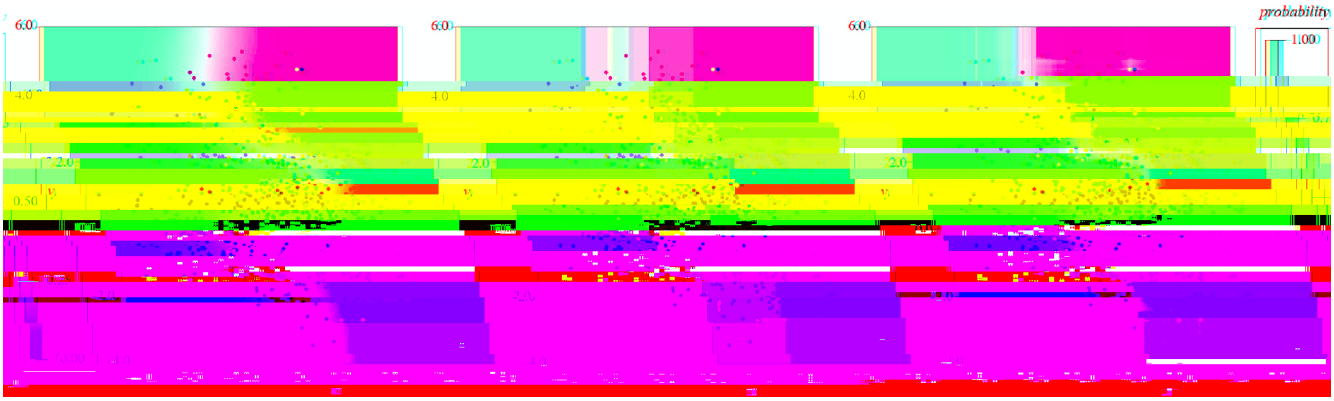
After the independent construction in parallel, M based tree models $\mathscr{P}_1, \mathscr{P}_2, \ldots, \mathscr{P}_M$ are integrated as the final random forest model $\mathscr{P}_{forest}$. For the prediction of sample $\mathbf{x}$, the predictions of M trees are determined at first, then the classification probability for the kth class of $\mathbf{x}$ is estimated by averaging them:

$$\mathscr{P}_{f\ e\text{-}}\left(\quad = Y_{\cdot} \mid \mathbf{x} \right) = \frac{1}{M} \sum_{=1}^{M} \mathscr{P}\ \left(\quad = Y_{\cdot} \mid \mathbf{x} \right) \text{intrlapp9} \tag{11}$$

For bagging style ensemble, the diversity between base models has been proved beneficial for the performance and robustness of ensemble models (Breiman, 1996; Ueda and Nakano, 1996; Brown et al., 2005). As a special version of bagging probabilistic trees, probabilistic random forest inherits this nature. Due to the additional randomized node optimization, the diversity of based trees is increased significantly, which makes probabilistic random forest more accuracy and robust than the common bagging ensemble without additional randomization.

For an intuitive understanding of probabilistic decision tree and probabilistic random forest, a 2-dimensional artificial problem is presented for demonstration. 1000 samples of two classes (orange and blue points) are generated under the pre-defined mixture distributions, and the true probability function (deduced from the theoretical distribution of each class) for the orange points is visualized in Fig. 1 (a) by gradual colors. Since these samples are labeled by probabilities, there is significant class overlapping which makes it hard to distinguish them by deterministic classification.

Based on these samples, classification probability functions fitted by PrDT and PrRF are visualized into Fig. 1 (b) and (c) respectively. Although it's effective enough overall, as a stepwise function the fitted probability function of PrDT is too rough compared to the true function, which is insufficient to provide reasonable probability predictions. However, this drawback is alle e



i . 1. Probability estimation results of PrDT and PrRF on the artificial dataset.

cores) of 9 different areas[4] from the Daqing, Shengli, Dagang, and Karamay oilfields are collected to set up 9 real-world lithology probability estimation modeling tasks for experiments. Geological information including sedimentary environment, lithologic sequence, and lithology types (abbreviations are used for short, see Appendix A) of the involved 9 areas is listed in Table 2. The formations of these areas are dominated by various sandstones or conglomerates, and the main differences in lithology are reflected in mineral composition and grain size, which are able to distinguish by conventional logging curves.

For each area, after some pre-processes such as normalization of logging curves (Shier et al.) and depth localization of cuttings and core, the curve values of multiple wells are collected as the input part and calibrated by available drill cuttings or borehole cores for the construction of training sample set. In order to avoid possible mislabeling, measurement points near the lithological interfaces are excluded from the collection. Information such as calibration source, size of samples and input curves is illustrated in Table 3. Due to the continuity of the drilling record, more than thousands of samples are calibrated by the drilling cuttings for 7 of these 9 tasks, which provide sufficient data for further lithology probability estimation.

Nine probability estimation algorithms including LDA, QDA, GMM, KDE, NBAYES, TAN, AODE, PNN, and PrRF[5] are applied to these tasks for comparison. The hyper-parameter setting of each algorithm for each task is determined by a grid search process, while 10-folds cross-validation is employed to evaluate the generalization performance with the optimized hyper-parameter setting. In our comparison, classification accuracy $ACC(\mathscr{T})$ and averaged probability error $APE(\mathscr{P})$ are employed to measure the fitting goodness of models.

Geological information of the involved 9 areas.

Area	Sedimentary Environment	Lithologic Sequence	Lithology Types
A1	Delta Plain/Front	Shale-Sand	MuS, PeS, SiS, CaS
A2	Fluvial Sediment	Shale-Sand	MuS, SiM, SiS, FiS
A3	Delta Front	Shale-Sand/Lime	MuS, SiS, FiS, LmS, Lm
A4	Fluvial Sediment	Shale-Sand	MuS, PeS, SiS, FiS, CoS
A5	Fluvial Sediment	Shale-Sand	MuS, PeS, SiS, FiS
A6	Delta Plain	Shale-Sand	Sh, MuS, PeS, SiS, CaS, Gy
A7	Proluvial Fan	Glutenite	MuS, ShC, SaC, BrC, FiC, CoS
A8	Proluvial Fan	Glutenite	MuS, ShC, SaC, FiC, CoS
A9	Proluvial Fan	Glutenite	MuS, ShC, SaC, FiC, CoS

[4] For the purposes of data confidentiality, the real names of areas are replaced with codes.

[5] Availability of implementations for these algorithms are declared in Appendix B.

Dataset information of the involved 9 modeling tasks.

Area	Calibration	#Well	#Sample	Input Curves
A1	Cuttings, Cores	22	14620	γ_n, Δt, ρ_b, R_{MN}, R_{LLS}, R_{LLD}
A2	Cuttings, Cores	5	4764	γ_n, Δt, ρ_b, Φ_N, P_e, R_{MN}, R_{LLS}, R_{LLD}
A3	Cuttings	3	5224	γ_n, Δt, ρ_b, R_{MN}, R_{LLS}, R_{LLD}
A4	Cuttings	7	3160	γ_n, Δt, ρ_b, R_{MN}, R_{LLS}, R_{LLD}
A5	Cuttings, Cores	6	5680	γ_n, Δt, ρ_b, R_{LLS}, R_{LLD}
A6	Cuttings	4	2430	γ_n, Δt, ρ_b, R_{LLS}, R_{LLD}
A7	Cuttings	7	3540	γ_n, Δt, ρ_b, R_{LLS}, R_{LLD}
A8	Cores	5	761	γ_n, Δt, ρ_b, Φ_N, R_{LLS}, R_{LLD}
A9	Cores	6	530	γ_n, Δt, ρ_b, Φ_N, R_{LLS}, R_{LLD}

4.2. Experimental results

To eliminate the influence of randomness in fold splitting, we repeat the cross-validation evaluation process 30 times and take the average of obtained $ACC(\mathscr{T})$ and $APE(\mathscr{P})$ values as the final result. The evaluated classification accuracy and averaged probability error are listed in Table 4 and Table 5 respectively. For each task, the maximum accuracy or minimum error are marked with underlines. Besides, bar charts of these results are also visualized in Fig. 2 for more intuitive comparison.

Although there are differences in the magnitude of $ACC(\mathscr{T})$ and $APE(\mathscr{P})$ for different tasks, some general conclusions are still able to summarize according to the relative performance difference of algorithms for the same task:

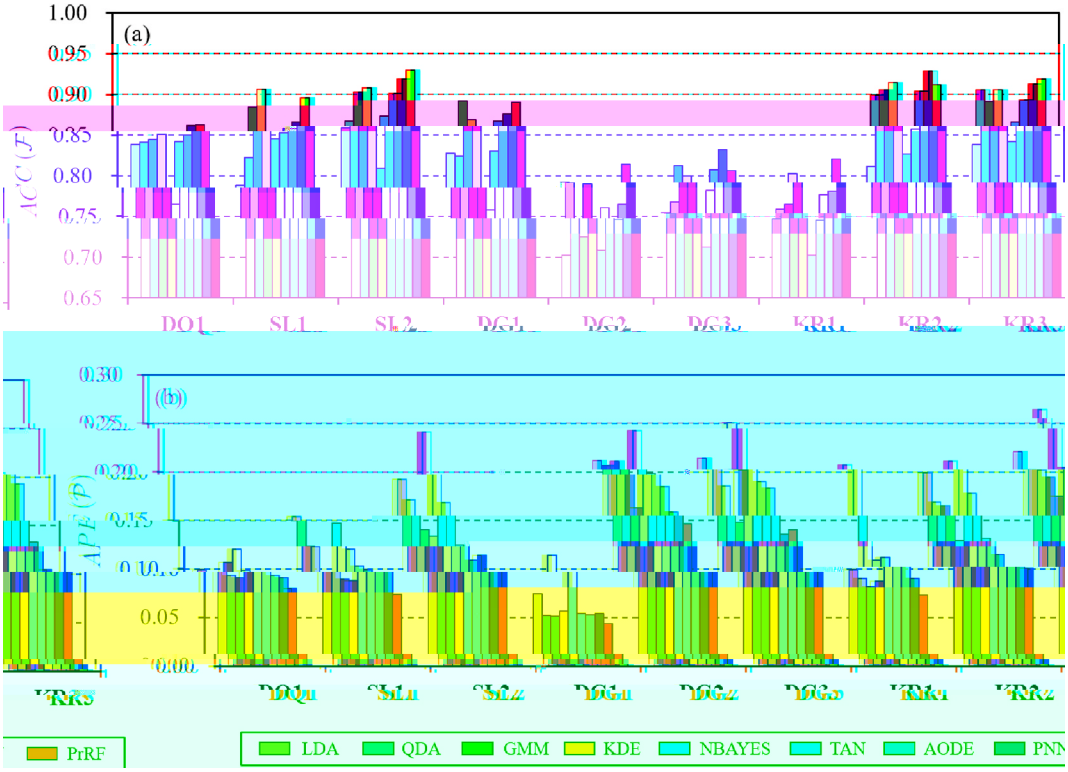
1. The advantage of PrRF in performance is highlighted, which wins the maximum accuracy for 5 times and the minimum averaged probability error for 7 times. Meanwhile, for the tasks that PrRF fails to perform the best, the performance gaps between PrRF and the best algorithm are quite small.
2. NBAYES and LDA are not good choices for logging lithology discrimination since there are significant performance gaps between them and other algorithms. This phenomenon can be attributed to the fact that their algorithmic assumptions are too strict, which makes them unsuitable for the classification or probability estimation of lithologies.
3. The poor performances of LDA and QDA are able to be improved by GMM and KDE with more flexible assumptions. However, the prerequisite for this improvement is that the models of GMM and KDE are learned with the optimized hyper-parameter setting, while the hyper-parameter tuning process for GMM and KDE are very troublesome and time costs.
4. Compared to NBAYES, much better performances are achieved by TAN, AODE, and PNN. Among 4 Bayesian network algorithms, PNN outperforms the rest algorithms for most of the 9 tasks, which is quite

Classification accuracy of algorithms on 9 modeling tasks.

Area	LDA	QDA	GMM	KDE	NBAYES	TAN	AODE	PNN	PrRF
A1	0.8388	0.8415	0.8449	0.8512	0.7649	0.8419	0.8496	0.8620	0.8624
A2	0.7880	0.8220	0.8840	0.9060	0.8460	0.8523	0.8577	0.8660	0.8960
A3	0.8587	0.8668	0.9032	0.9078	0.8091	0.8737	0.9011	0.9190	0.9295
A4	0.8277	0.8242	0.8921	0.8691	0.7576	0.8301	0.8674	0.8762	0.8905
A5	0.7024	0.7916	0.7249	0.7904	0.7085	0.7604	0.7464	0.7651	0.8140
A6	0.7548	0.7677	0.8129	0.8000	0.7120	0.7817	0.8067	0.8323	0.8061
A7	0.7476	0.7588	0.7650	0.8023	0.7024	0.7453	0.7771	0.7808	0.8202
A8	0.8112	0.8992	0.9052	0.9140	0.8264	0.8574	0.9037	0.9288	0.9120
A9	0.8380	0.9052	0.8908	0.9056	0.8424	0.8658	0.8926	0.9124	0.9192

Averaged probability error of algorithms on 9 modeling tasks.

Area	LDA	QDA	GMM	KDE	NBAYES	TAN	AODE	PNN	PrRF
A1	0.0991	0.0932	0.0909	0.1065	0.1204	0.0985	0.0930	0.0912	0.0796
A2	0.1542	0.1231	0.0892	0.0877	0.1470	0.1203	0.1029	0.0992	0.0738
A3	0.1923	0.1716	0.1407	0.1336	0.2411	0.1683	0.1248	0.1090	0.1146
A4	0.0741	0.0517	0.0511	0.0568	0.1145	0.0540	0.0529	0.0540	0.0434
A5	0.2118	0.2068	0.2108	0.1631	0.2426	0.1985	0.1846	0.1587	0.1463
A6	0.2140	0.1857	0.1481	0.2022	0.2509	0.1902	0.1836	0.1361	0.1408
A7	0.0989	0.0904	0.0822	0.0874	0.2071	0.1091	0.1120	0.0901	0.0731
A8	0.1991	0.1692	0.1649	0.1293	0.2108	0.1783	0.1315	0.1151	0.1006
A9	0.2209	0.2020	0.1944	0.1751	0.2645	0.2044	0.1927	0.1332	0.1041



i . . Bar charts for the comparison of classification accuracies and averaged probability errors on 9 logging lithology modeling tasks.

competitive to PrRF especially from the perspective of classification accuracy.

Relatively speaking, the performance advantage of probabilistic random forest in probability estimation is more significant than classification. We reason that this observation can be attributed to the fact that PrRF itself is designed for the optimization of classification probability estimation, which is able to obtain better prediction with less averaged probability error. Overall, the feasibility and advantage of PrRF are confirmed based on the experimental results we presented, which substantiates that PrRF is a reliable algorithm for both logging lithology discrimination and probabilities prediction.

. ic tion c

In this section, to validate the practicality of the proposed algorithm, we apply it to the A1 area for probabilistic lithology characterization for demonstration. The selected A1 area is located in the Songliao Basin,

Northeast China, which is composed of lake delta sediments mainly. The pelitic siltstone and siltstone deposited in the delta plain and the former delta environment constitute the main body of reservoirs. However, due to the limitation of deposition and hydrodynamic conditions, sandbodies of the selected area tend to be thin layers with poor dispersion of planar distribution. These bring great challenges to the formation evaluation and reservoir characterization.

For the selected area, the conventional logging curve of each well is relatively complete, while drilling cutting lithology descriptions of 20 wells and core descriptions of 4 wells are available. These provide a large number of samples for the machine learning assisted logging lithology interpretation. In our application, the training set is constructed based on 22 wells and the rest 2 core wells were reserved for verification. Based on the collected sample set (related information is listed in the first row of Table 3), a lithology probabilities estimation model is learned by the PrRF algorithm with the optimized hyper-parameter setting obtained in section 4. Then the established PrRF model is applied to available wells for prediction.

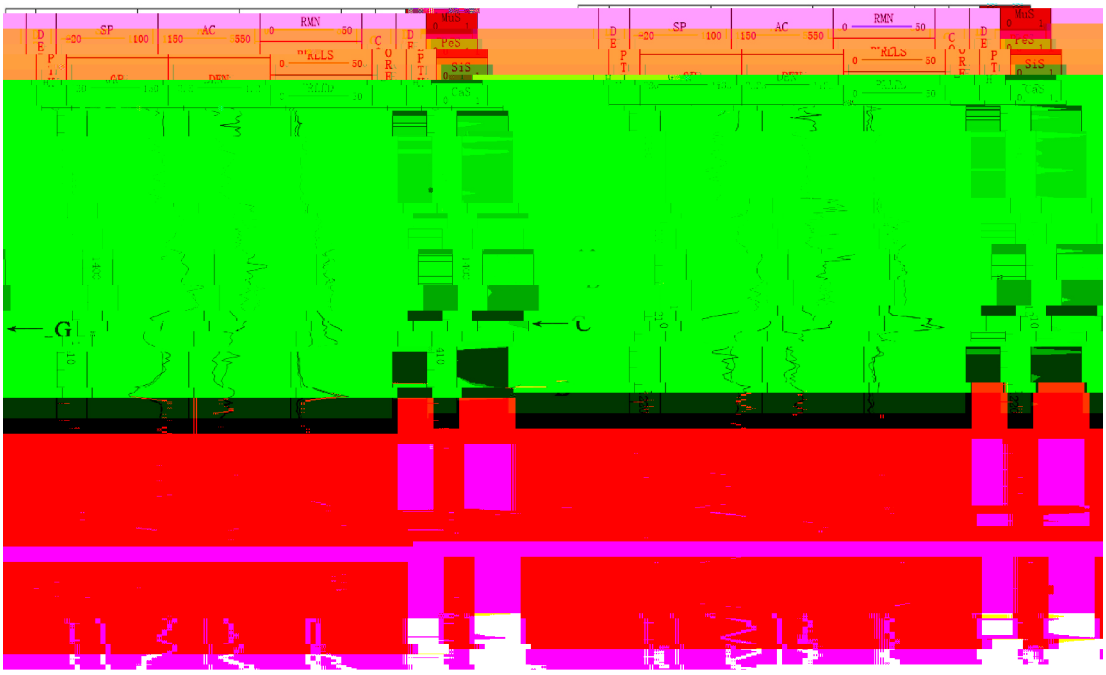
The prediction results of two verification wells are visualized in Fig. 3 for analysis. In Fig. 3, logging curves and core lithology descriptions are plotted at the left side. Since the sum of probabilities is identical to 1, the predicted lithology probabilities are visualized in the same way with rock volume percentages at the right side. Compared the predicted lithology probabilities with the core lithology descriptions, it's easy to see that the consistency between them is very well, which further demonstrates the feasibility of using PrRF for the probabilities prediction of logging lithology. Moreover, the advantages of probabilistic lithology characterization are also revealed by the demonstration of Fig. 3:

1. For the marked layers C, D, E, and H, based on the morphology of the natural gamma and resistivity curves, we conclude that these layers have obvious positive rhythm characteristics. For deterministic lithology characterization, even if the lithology discriminations is completely consistent with the coring descriptions, these rhythms are unable to be reflected. However, for probabilistic lithology characterization, the gradual changes of granularity are described by the trends of lithology probabilities.
2. Although the overall lithology conclusion of a formation remains the same, this does not mean that the mineral composition or granularity in it is constant. For the marked layers E, F, G, and I, the existence of lithology heterogeneity is evidenced by fluctuations in the log responses. For deterministic discriminations, these fluctuations are ignored since they are not significant enough to cause the change of lithology conclusions, while probabilistic lithology characterization is able to characterize them by the fluctuations of probabilities.
3. Differences in lithology probabilities also reflect the properties difference of formations in the same lithology type. For example, even though layer A, B, and H are all PeS layers, A and H have significant probabilities of SiS while the SiS probability of B is closed to zero. This phenomenon indicates that B is a typical PeS layer, but A and H are more like a transition between PeS and SiS.

Through the analysis of two core wells, the feasibility and effectiveness of PrRF based probabilistic lithology characterization are substantiated. The application in A1 area also reveals that in addition to determining the lithology types accurately, probabilistic lithology characterization is also able to fuse related information in multiple curves to provide more details of borehole lithology changes, which improves the accuracy and fineness of lithology characterization effectively.

. onc u ion

In this article, we propose a probability based fuzzy method for borehole lithology interpretation and characterization. To improve the



accuracy of lithology probability estimation, we also propose the probabilistic random forest algorithm and investigate its feasibility and advantage compared to 8 existing algorithms in logging lithology interpretation. The research conclusions are summed up as follows:

1. The advantage of the proposed probabilistic random forest referred to other 8 algorithms is substantiated by the comparative experiments on 9 real-world logging lithology modeling tasks, which confirms that the proposed probabilistic random forest algorithm is a significantly better way for logging lithology probability estimation.
2. Through the application case of A1 area, the feasibility and practicality of probabilistic based fuzzy lithology characterization are confirmed. Comparison between probabilities lithology characterization and deterministic lithology discrimination illustrates that the former is able to provide more information about rhythm, heterogeneity, and formation properties, which can be used for finer reservoir characterization.

Overall, through the research in this paper, we reveal the advantage and feasibility of the probabilistic random forest based lithology characterization in logging lithology interpretation, which worths further application and promotion to improve the fineness of reservoir characterization. However, due to the limitation of available data, our study still has its imperfections. The main drawback is that we only investigate the practicality of the proposed method for shale-sand or glutenite formations. For carbonate reservoirs or volcanic reservoirs, whether it is able to maintain the feasibility and advantage still deserve further study.

c r tion o co tin int r t

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.

ndi . u nt r d t

Supplementary data to this article can be found online at https://doi.org/10.1016/j.cageo.2020.104556.

ndi . r i tion or it o o

- Sh: Shale
- MuS: Mudstone
- SiM: Silty Mudstone
- PeS: Pelitic Siltstone
- SiS: Silty Sandstone
- FiS: Fine Sandstone
- CoS: Coarse Sandstone
- CaS: Calcareous Sandstone
- LmS: Lime Sandstone
- Lm: Limestone
- Gy: Gypsum
- ShC: Shaly Conglomerate
- SaC: Sandy Conglomerate
- BrC: Breccia Conglomerate
- FiC: Fine Conglomerate

ndi . t t nt out od i i it

The experiments in this article are carried out under the R language environment, which is available on www.r-project.org. Codes of our study is mainly focused on the implementation of various probability estimation algorithms. Most of them have been implemented by R packages which are available on *The Comprehensive R Archive Network*:

- The LDA and QDA algorithms has been implemented by lda() and qda() in the R package MASS (https://cran.r-project.org/web/packages/MASS/index.html).
- The GMM algorithm has been implemented by MclustDA() in the R package mclust (https://cran.r-project.org/web/packages/mclust/index.html).
- The NBAYES and TAN algorithms have been implemented by the function naive.bayes() and tree.bayes() in the R package bnlearn (https://cran.r-project.org/web/packages/bnl earn/index.html).
- The PNN algorithm has been implemented by the function learn() in the R package pnn (https://cran.r-project.org/web/packages/pnn/index.html).

For the algorithms of KDE, AODE, and PrRF, we implement them by ourselves since there is no suitable implementation in the R environment available for now. Although we are unable to publish these implementations due to the limitation of technical confidentiality agreement, we still give some tips and suggestions for readers to facilitate their re-implementation. Specifically:

- For reasons of computational efficiency, the KDE algorithm is implemented with c++ by ourselves based on the definition of Eq (6) and Eq (7). Then with the help of R package Rcpp, the c++ implementation is wraped as R function for application.
- The AODE algorithm has already been implemented by the R package AnDE (https://cran.r-project.org/web/packages/AnDE/index.html). However, we find this implementation is too time-costing during the application. Thus, we re-implement it refer to the implementation of Weka (https://www.cs.waikato.ac.nz/ml/weka/index.html) with c++ and then wrap as R function with Rcpp.
- The proposed PrRF algorithm is implemented based on the available implementation of random forest in the R package ranger (https://cran.r-project.org/web/packages/ranger/ind ex.html). Attentive readers can discover that all we need to do is modify the split selection and model integration implementation in ranger.

r nc

Al-Anazi, A., Gates, I., 2010. On the capability of support vector machines to classify lithology from well logs. Nat. Resour. Res. 19 (2), 125–139.

Annan, J., Lu, J., 2009. Studying the lithology identification method from well logs based on de-svm. In: 2009 Chinese Control and Decision Conference. IEEE, pp. 2314–2318.

Benaouda, D., Wadge, G., Whitmarsh, R.B., Rothwell, R.G., MacLeod, C., 1999. Inferring the lithology of borehole rocks by applying neural network classifiers to downhole logs: an example from the ocean drilling program. Geophys. J. Int. 136 (2), 477–491.

Bhatt, A., Helle, H.B., 2002. Determination of facies from well logs using modular neural networks. Petrol. Geosci. 8 (3), 217–228.

Bosch, M., Zamora, M., Utama, W., 2002. Lithology discrimination from physical rock properties. Geophysics 67 (2), 573–581.

Bosch, D., Ledo, J., Queralt, P., 2013a. Fuzzy logic determination of lithologies from well log data: application to the ktb project data set (Germany). Surv. Geophys. 34 (4), 413–439.

Bosch, D., Ledo, J., Queralt, P., 2013b. Fuzzy logic determination of lithologies from well log data: application to the ktb project data set (Germany). Surv. Geophys. 34 (4), 413–439.

Breiman, L., 1996. Bagging predictors. Mach. Learn. 24 (2), 123–140.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.

L. Breiman, J. H. Friedman, R. Olshen, C. Stone, Classification and Regression Trees.

Brown, G., Wyatt, J.L., Tiňo, P., 2005. Managing diversity in regression ensembles. J. Mach. Learn. Res. 6 (1), 1621–1650.

Burke, J., Campbell Jr., R., Schmidt, A., et al., 1969. The litho porosity cross plot: a new concept for determining porosity and lithology from logging methods. In: Proceedings in the 10th SPWLA Annual Logging Symposium. Society of Petrophysicists and Well-Log Analysts.

Busch, J., Fortney, W., Berry, L., et al., 1987. Determination of lithology from well logs by statistical analysis. SPE Form. Eval. 2, 412–418, 04.

Chang, H.C., Chen, H., Fang, J., 1997. Lithology determination from well logs with fuzzy associative memory neural network. IEEE Trans. Geosci. Rem. Sens. 35 (3), 773–780.

Cheng, G., Guo, R., Wu, W., 2010. Petroleum lithology discrimination based on pso-lssvm classification model. In: 2010 2nd International Conference on Computer Modeling and Simulation, vol. 4. IEEE, pp. 365–368.

Chow, C., Liu, C., 1968. Approximating discrete probability distributions with dependence trees. IEEE Trans. Inf. Theor. 14 (3), 462–467.

C. Clavier, D. Rust, et al., Mid plot: a new lithology technique, Log. Anal. 17 (06).
Cooper, G.F., 1990. The computational complexity of probabilistic inference using bayesian belief networks. Artif. Intell. 42 (2–3), 393–405.
Dagum, P., Luby, M., 1993. Approximating probabilistic inference in bayesian belief networks is np-hard. Artif. Intell. 60 (1), 141–153.
de Oliveira, J.M., dos Santos, E.M., Carvalho, J.R.H., de Vasconcelos Marques, L.A., 2013. Ensemble of heterogeneous classifiers applied to lithofacies classification using logs from different wells. In: The 2013 International Joint Conference on Neural Networks (IJCNN). IEEE, pp. 1–6.
Delfiner, P., Peyret, O., Serra, O., et al., 1987. Automatic determination of lithology from well logs. SPE Form. Eval. 2, 303–310, 03.
Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the em algorithm. J. Roy. Stat. Soc. 1–38.
Deng, C., Pan, H., Fang, S., Konaté, A.A., Qin, R., 2017. Support vector machine as an alternative method for lithology classification of crystalline rocks. J. Geophys. Eng. 14 (2), 341.
Dev, V.A., Eden, M.R., 2018. Evaluating the boosting approach to machine learning for formation lithology classification. In: Computer Aided Chemical Engineering, vol. 44. Elsevier, pp. 1465–1470.
Devijver, P.A., Kittler, J., 1982. Pattern Recognition: A Statistical Approach. Prentice Hall.
Dubois, M.K., Bohling, G.C., Chakrabarti, S., 2007. Comparison of four approaches to aRo