Proceedings of the ASME 2024 43rd International Conference on Ocean, Offshore & Arctic Engineering
OMAE2024
June 9-14, 2024, Singapore EXPO, Singapore

OMAE2024-123362

UNSUPERVISED MACHINE LEARNING FOR WELL LOG DEPTH ALIGNMENT

Sushil Acharya
Department of Geoscience
Norwegian University of Science and Technology
Trondheim, Norway
Email: sushil.acharya@ntnu.no

Karl Fabian
Department of Geoscience
Norwegian University of Science and Technology
Trondheim, Norway
Email: karl.fabian@ntnu.no

ABSTRACT

Depth alignment as a prerequisite for subsurface geology interpretation aims to match signals of two logs from the same well to a common depth scale. Here, three consecutive unsupervised machine-learning techniques are applied to achieve this goal. First, data vectors are reduced through pre-processing with PCA and autoencoder methods, and clustered by K-means with K-means++ initialization. In a second step, corresponding cluster boundaries are identified to globally align lithologic units. This assumes that cluster boundaries represent physical changes in lithology, without direct lithological interpretation. The third step uses cross-correlation to obtain the final log alignment. Additional linear or non-linear interpolation between matching boundaries is sufficient to align well logs. Performance and accuracy of various techniques are compared by visual inspection and manually determined boundaries. Reducing data dimension with an autoencoder and K-means clustering performed best among tested methods. The study suggests that cluster-based methods can automate depth alignment of well logs without human intervention.

1 INTRODUCTION

Well logs are high-resolution (0.1-1 m) multivariate records of physical properties of subsurface formations, generated by moving multi-sensor instrument chains through a borehole. Common measurement parameters are natural gamma ray activity, electric resistivity, density, and porosity of the formation. Geologists and engineers rely on well logs to assess crucial geological and geotechnical aspects, like lithology, fluid saturation, porosity, permeability, and pressure within a borehole [1].

For a single well, several well logs are acquired at different times. Typically, a logging while drilling (LWD) record is obtained in real-time, followed by a later electrical wireline log (EWL). These methods have different depth measurement systems, which introduce errors and discrepancies in the depth values [2, 3, 4]. Borehole conditions such as diameter, shape, inclination, and mud properties also affect depth measurements and lead to errors. Wire tension and friction forces stretch or squeeze the measurement string and displace the logging equipment from its intended depth. Discrepancies in signal patterns resulting from different logging techniques and sensors further complicates the aligning of corresponding data to a uniform reference depth [2, 3, 4]. These challenges associated with depth alignment necessitate the exploration of corrective methodologies. Previously, labor-intensive techniques like manual cross-correlation involving expert intervention have been employed, proving to be time-consuming [5, 6]. An automated approach is required to process the large volume of subsurface data in a short amount of time.

Various studies have investigated employing machine learning techniques to automate well-log depth alignment. Zimmermann et al. [6] and Le et al. [7] introduced a supervised machine-learning method using a 1D-CNN to align well-logs to the same depth reference. Their approach involved choosing peaks (anchor points) on a reference log, creating windows centered around these peaks, and training the neural network to find windows with similar peaks on a desynchronized log. This

method enhances synchronization in well-log signals from vertical or low-deviation wells. However, its effectiveness relies on selected peak values. Moreover, it is not suitable for logs with substantial shifts or spanning greater depth ranges. Wang et al. [8] introduced a unique multitask learning approach based on deep neural networks for optimizing depth-matching algorithms for multiple gamma-ray logs in the same field. They proposed a sliding window method in which the best matching portion is found by moving a segment of the reference log (query) across the target log (tracker). The size of the tracker determines how effective the model is. But beyond the gamma-ray log, there isn't much documentation on how to apply this method, which makes it difficult to apply it in broader settings. Caceres et al. [5] introduced a new regression-based framework employing seven 1D CNNs to estimate depth shifts in seven different well logs from the same well. They combined predicted shifts using diverse fusion techniques. While effective for well logs with uniform depth errors, the method can not handle well logs with nonuniform errors. To align well logs affected by non-uniform depth errors, Acharya et al. [9] modified the methodology presented by Caceres et al. by utilizing better data preparation techniques with smaller windows. Their modification performs well in depth aligning GR logs with minimal noise. However, this method was only tested with gamma rays.

Considering the limitations identified in supervised techniques, an unexplored avenue lies in unsupervised machine learning algorithms using various suites of logs, which might offer novel solutions for handling depth matching of well logs. In recent studies by [10] and [11], the application of unsupervised machine learning techniques in lithology identification from well log data has shown promising results. These studies have explored clustering algorithms and their effectiveness in classifying lithology types without explicit supervision.

Based on these findings, the utility of unsupervised machine learning methods for clustering relevant lithological information is here employed to provide robust global boundaries for aligning LWD logs with reference EWL logs from the same well through cluster-based matching. Various unsupervised machine-learning techniques are presented and compared for their use in pre-processing for well-log depth alignment using standard well-log measurements, such as natural gamma-ray activity, deep resistivity, bulk density, and neutron porosity logs.

2 DATASETS

To develop and test different alignment techniques, two wells, 16/1-9 and 16/1-6 A, from the DISKOS [12] database are used that both provide the same four-parameter EWL and LWD data sets. Both wells were drilled in the Norwegian North Sea as exploration wells. For well 16/1-6 A the '2020 FORCE Machine Learning Contest,' [13], provides an additional lithology column. For all data, flagged outliers or missing values were removed, and the records were normalized as outlined in the table 1.

TABLE 1: PHYSICAL PARAMETERS FOR WELLS 16/1-9 AND 16/1-6 A USED TO DEVELOP AND TEST THE ALIGNMENT TECHNIQUES. FOR EACH PARAMETER, THE COMMONLY USED MEASUREMENT UNIT AND TYPICAL ABBREVIATIONS IN THE DISKOS DATABASE ARE LISTED. BEFORE ALIGNMENT, THE DATA ARE TRANSFORMED TO A NORMALIZED UNIT-FREE QUANTITY AS LISTED IN THE LAST COLUMN.

Parameter	Unit	Abbreviation	Transform	Normalization
Bulk density ρ	g/cm^3	SBDC, RHOB	linear	DEN $= \frac{\rho-\bar{\rho}}{\sigma(\rho)}$
Deep resistivity r	Ωm	RDEP, SEDP	logarithmic	RES $= \frac{\log r-\overline{\log r}}{\sigma(\log r)}$
Natural γ-ray activity γ	API	KBGR,GR	linear	GR $= \frac{\gamma-\bar{\gamma}}{\sigma(\gamma)}$
Neutron porosity Φ	%	TNPL, NPHI	linear	NP $= \frac{\Phi-\bar{\Phi}}{\sigma(\Phi)}$

3 PRE-PROCESSING METHODS

Successful alignment of well-log data requires compatible and robustly comparable data sets. Several pre-processing methods are available to improve the original data sets by removing or weighting outliers or noise, and by diminishing the intrinsic redundancy of multi-parameter logs. Unsupervised machine learning provides several standard methods that focus on identifying patterns and structures within large data sets without explicit supervision or manually labeled training data [14]. Here, the implementations of the Python libraries scikit-learn [15] and TensorFlow [16] were used.

3.1 K-means Clustering

A standard unsupervised machine learning technique to reduce data complexity is K-means clustering. It splits a dataset into K unique and non-overlapping clusters [17, 14]. Starting from K randomly placed centroids in the data space, the algorithm iteratively assigns each data point to its nearest centroid and then updates the centroid position as the mean of its cluster.

The original dataset consists of $N \approx 10^5$ measurements for each log, denoted by

$$x_i = (\mathrm{GR}_i, \mathrm{RES}_i, \mathrm{DEN}_i, \mathrm{NP}_i) \in \mathbb{R}^4, \quad i = 1, \ldots, N.$$

With the notation

$$\overline{\sum_{x\in A}} f(x) := \frac{1}{\#A}\sum_{x\in A} f(x),$$

for averaging $f(x)$ over all $x \in A$, where $\#A$ is the number of elements in A, K-means clustering proceeds in the following steps:

1. **Initialization:** For some $K \ll N$ initialize K cluster centroids $\mu_j \in \mathbb{R}^4$, $j \in \{1,\dots,K\}$ either randomly, or by some other fast method.
2. **Data assignment:** For each data point x_i calculate the Euclidean distances $||x_i - \mu_j||$ to all centroids. Assign x_i to the cluster C_j with the closest centroid μ_j, $(j \in \{1,\dots,K\}$.
3. **Centroid update:** Update centroids μ_j as the mean of all data points in their respective clusters:

$$\mu_j \leftarrow \overline{\sum_{x_i\in C_j}} x_i.$$

The K-means algorithm iteratively executes the assignment step and update step until meeting one of the following stopping criteria:

- centroids no longer exhibit significant changes,
- a specified maximum number of iterations is reached,
- the assignments of data points to clusters stabilize.

Upon convergence, the algorithm returns centroids that minimize within-cluster variation, ensuring stable assignments of data points to clusters.

For initialization, the K-means++ algorithm is used, which enhances the cluster initialization process by improving the selection of initial centroids [18]. The optimal number of iterations for centroid initialization is defined by the *n_init* parameter, consequently optimizing the clustering process. The same K-means clustering setup was separately applied to each LWD and EWL record.

The main issue with K-means clustering is to determine the optimal number K of clusters. This is stepwise resolved here by the elbow method [19] and the silhouette score. The first uses the "elbow" shaped plot of the variance explained (or inertia) $J(K)$ of the data set to identify an optimal K where adding further clusters does not significantly reduce J.

Inertia is defined as the Within-Cluster Sum of Squares (WCSS)

$$J(K) = \sum_{j=1}^{K}\sum_{x_i\in C_j} ||x_i - \mu_j||^2.$$

The silhouette value [20] of a point $x \in C_j$ is determined using

$$a(x) = \overline{\sum_{x'\in C_j\setminus\{x\}}} ||x-x'||, \text{ and } b(x) = \min_{k\neq j} S = \overline{\sum_{x'\in C_k}} ||x-x'||,$$

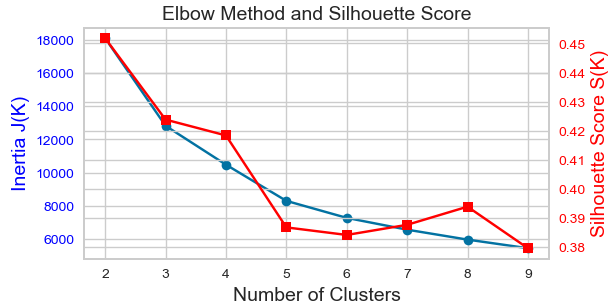
as

$$s(x) = \frac{b(x)-a(x)}{\max(b(x),a(x))}.$$

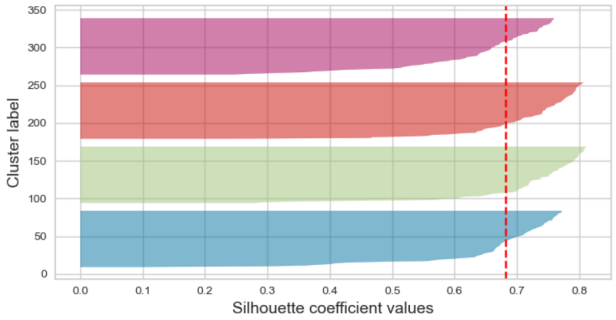
The silhouette score of a cluster assignment is the average silhouette value

$$S = \overline{\sum_{x}} s(x),$$

which for optimal clustering should be as close to 1 as possible.



(a) INERTIA $J(K)$ AND SILHOUETTE SCORE $S(K)$ FOR K-MEANS CLUSTERING OF EWL LOGS OF WELL 16/1-9.



(b) SILHOUETTE ANALYSIS FOR $K = 4$ CLUSTERS

FIGURE 1: COMPARATIVE ANALYSIS OF ELBOW METHOD AND SILHOUETTE SCORE FOR DETERMINING THE OPTIMAL 'K' VALUE IN K-MEANS CLUSTERING OF EWL WELL LOGS FROM WELL 16/1-9.

For the K-means clustering of EWL and LWD logs of well 16/1-9, the $J(K)$ curve in Fig. 1a suggests potential K values between 3 and 5. The analysis based on the silhouette score shown in Fig. 1b indicates that utilizing $K = 4$ clusters is optimal. This conclusion is drawn from the observation that the silhouette score for each cluster is higher than the average silhouette score (vertical red dotted line). Moreover, the clusters are very similar in size.

3.2 Principal Component Analysis

The standard tool for dimensionality reduction in data analysis is principal component analysis (PCA). It determines the unique linear transformation of the high-dimensional data into a sequence of lower-dimensional spaces each covering as much of the remaining variance as possible [21].

For the data matrix $X \in \mathbb{R}^{N\times n}$, which for each of the N depth values contains $n = 4$ normalized measurement parameters, PCA proceeds in five steps:

1. With the mean values
$$\bar{x}_j = \frac{1}{N}\sum_{i=1}^{N} x_{ij}, \quad j \in \{1,\ldots,n\},$$
it is possible to calculate the centered data $x_{ij} - \bar{x}_j$.
2. Then the covariance matrix $\Sigma \in \mathbb{R}^{n\times n}$ is calculated as
$$\Sigma = \frac{1}{N} X^T X.$$
3. The principal directions are the eigenvectors $\mathbf{v}_i$ of Σ and represent orthogonal directions of data variability, while the corresponding eigenvalues λ_i indicate the variance along these directions.
4. The eigensystem for the general covariance matrices is determined by singular value decomposition (SVD) of the centered data matrix
$$X = U\Sigma' V^T,$$
providing matrices U, Σ', and V^T, where U and V contain the left and right singular vectors respectively, and Σ' is a diagonal matrix containing the singular values. After sorting the eigenvectors by decreasing eigenvalues the top k principal directions can be selected.
5. Projecting the data onto these k principal directions yields a new dataset of the selected principal components.

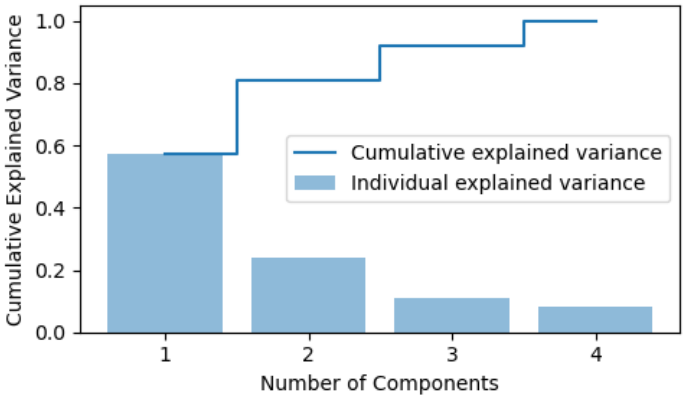


FIGURE 2: FRACTION OF DATA VARIANCE OF EWL LOG 16/1-9 EXPLAINED BY THE PRINCIPAL COMPONENTS AFTER PCA. BLUE BARS INDICATE THE EXPLAINED VARIANCE OF THE INDIVIDUAL COMPONENTS, WHILST THE BLUE LINE REPRESENTS THE CUMULATIVE EXPLAINED VARIANCE.

For the well log datasets, the plot of variance explained versus the number of principal components in Fig. 2 reveals that the first two principal components cover over 80% of the total variance. Because for dimensionality reduction it is suggested in [22] to select principal components that collectively cover at least 80% of the total variance, the first two components are used.

PCA thus reduces the dimensionality of the data from 4 to 2. Following this, K-means clustering was applied on EWL and LWD. The elbow method also in this case indicated that $K = 4$ clusters are optimal.

3.3 Autoencoders

An autoencoder is a neural network that learns to encode redundant input data vectors $x \in \mathbb{R}^n$ into a lower-dimensional representation $y \in \mathbb{R}^m$ with $m < n$ by decoding it back to an output $\hat{x} \in \mathbb{R}^n$ of original length and minimizing a loss function $\mathscr{L}(x,\hat{x})$ [23] [24]. It thus learns to compress the input data x in a lower-dimensional *latent space* representation y. Fig. 3 sketches the schematic architecture of an autoencoder. A commonly used loss function is the mean squared error (MSE)

$$\mathscr{L}(x,\hat{x}) = \frac{1}{N}\sum_{i=1}^{N} ||x_i - \hat{x}_i||^2, \tag{1}$$

where x_i and $\hat{x}_i$ represent individual input and reconstructed output vectors, and N is the total number of elements in the training data set (depth measurements).

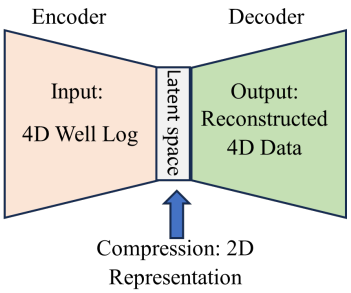


FIGURE 3: SCHEMATIC ARCHITECTURE OF AN AUTOENCODER WITH $N = 4$ DIMENSIONAL INPUT DATA, AND A $M = 2$ DIMENSIONAL LATENT SPACE.

To optimize the autoencoder architecture for well log data, hyperparameter tuning with a random search technique results in a simple architecture having an input layer with four features that gradually reduces dimensionality through encoder layers employing ReLU activation functions. The encoder has two dense layers with 8 and 4 neurons, respectively, resulting in a two-neuron encoding layer (latent space). The decoder repeats this structure to reconstruct the input. The model is constructed using the MSE loss function and the Adam optimizer (learning rate = 0.0001). Training is optimized using callbacks, including a learning rate scheduler and model checkpoint. This architecture is intended to capture data patterns across 100 epochs with a batch size of 16, allowing the extraction of relevant information from the data.

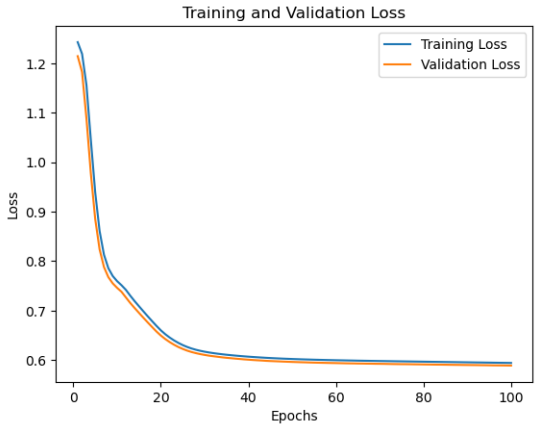


FIGURE 4: TRAINING AND VALIDATION LOSS (OR MSE) OF THE AUTOENCODER TRAINED USING DATA FROM WELL 16/1-9.

The autoencoder was trained on the data from well 16/1-9 with $n = 4$ dimensions. The data were split into training and validation sets, allocating 80% for training and 20% for model validation.

To assess the tradeoff between variance and bias, both the training loss and validation loss were plotted in Fig. 4. The figure reveals that there are no signs of overfitting in the model. The proximity or overlap between the training and validation loss curves suggests that the model has not excessively adapted to the training data to the detriment of generalization.

4 MATCHING CLUSTER BOUNDARIES

Once the cluster boundaries of the individual records are obtained, it is crucial to reliably match corresponding boundaries in both records. After retrieving the depth intervals associated with each identified cluster all depth intervals consisting of 1-5 identical cluster points are deleted, and the preceding cluster interval is extended to include the deleted depth interval. This successfully removes spurious boundaries and random subdivisions of large cluster intervals.

The upper and lower border depths of the $L = 30$ largest clusters in each log are identified as 'predicted boundaries', leading to $2L = 60$ predicted boundaries in each log. Small clusters are deliberately disregarded, because they are more likely to reflect random variability of measurements, logging process, or geological features in the well logs.

Refinement of these predicted boundaries involves sequential sorting of the depths in ascending order, and the detection of inconsistencies, specifically the presence of a boundary in one log that cannot be matched to a corresponding boundary in the other log.

Matching is performed by maximizing cross-correlation coefficients between adjacent 11-unit windows (approximately 28 cm) around predicted boundaries in the EWL and LWD Gamma-ray logs. Windows exhibiting cross-correlation values surpassing the 0.8 threshold are selected as matching windows, and corresponding boundaries are selected as matching final boundaries. The correlation threshold was deliberately set below 1 to accommodate realistic noise and inherent data variability in the well-log data. Remaining unmatched boundaries are removed after this step, such that $L' \leq L$ matching boundary pairs in both EWL and LWD are found, which then serve as reference points for the final well-log depth alignment. Subsequently, interpolation techniques to linearly or dynamically align the well logs need to be applied only within the intervals defined by the L' matching boundary pairs.

5 RESULTS

5.1 K-means Clustering

To evaluate the original K-means clustering algorithm for the normalized EWL and LWD data from well 16/1-9, the $L = 30$ largest clusters were selected. For a log length of $d = 1300$ m, the resulting $2L = 60$ boundary depths, even after deletion of some remaining spurious boundaries, provide potential matching points for interval lengths $\Delta d \approx d/L \approx 32$ m on EWL and LWD logs. This suffices for robustly define the global alignment of the pre-processing. The cross-correlation tests removed 18 inconsistent boundaries from both the EWL and LWD logs, that did not cross the minimum-correlation threshold, leaving 42 robust boundaries.

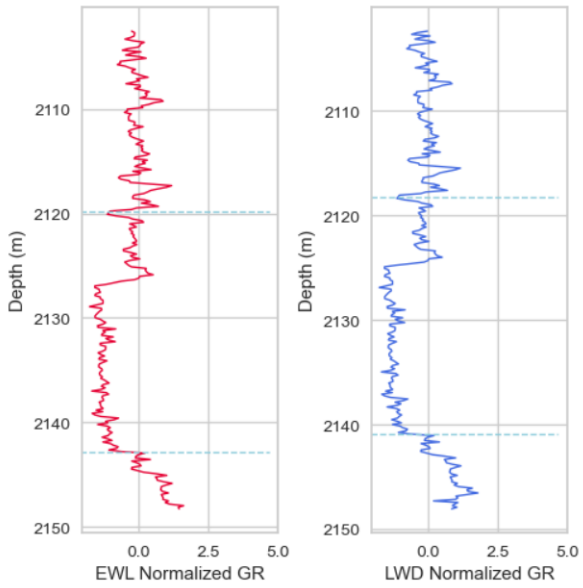


FIGURE 5: K-MEANS CLUSTERING WAS UTILIZED TO ANALYZE NORMALIZED WELL LOGS (EWL AND LWD) FROM WELL 16/1-9, HIGHLIGHTING DELINEATED BOUNDARIES (REPRESENTED BY DOTTED HORIZONTAL LINES) THAT IDENTIFY DISTINCT GEOLOGICAL ZONES. TO ENHANCE CLARITY, A CONCISE 46-METER SEGMENT OF THE LOG IS DISPLAYED IN THE IMAGE. THE COMPLETE IMAGE IS ACCESSIBLE IN THE APPENDIX.

Fig. 5, displays a specific 46 m interval of the 1300 m gamma-ray well logs, in which several visually apparent boundaries have not been selected by the algorithm. For instance, the distinct step at about 2127 m in both the EWL and LWD logs is missing.

Figure 6 illustrates the result of the global K-means clustering boundaries for linear alignment in the post-processing. The gamma-ray logs on the left display the initial state before alignment. The logs on the right show the post-alignment state, with improved but not perfect alignment of the linearly distorted LWD depth scale.

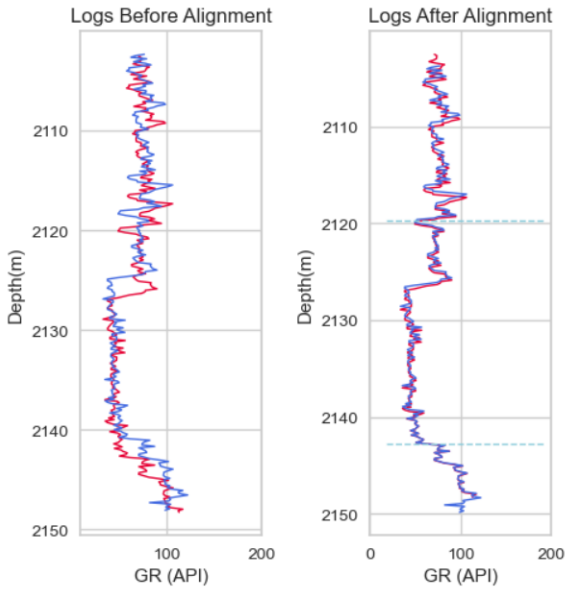


FIGURE 6: WELL LOG ALIGNMENT USING K-MEANS METHOD. LEFT: GAMMA RAY (GR) LOGS FOR EWL (RED), AND LWD (BLUE) BEFORE ALIGNMENT, EACH ON ITS OWN DEPTH SCALE. RIGHT: EWL (RED), AND LWD (BLUE) LOGS AFTER ALIGNMENT TO THE EWL DEPTH.

To assess the performance of the model with respect to linear post-processing, the gamma-ray logs from EWL and LWD were divided into 27 windows of up to 46 m length. Table 3 shows the resulting alignment based on Pearson's correlation and visual inspection of each slice. Out of the 27 slices (or windows), the majority, 23 pairs, exhibited noticeable improvement in depth alignment. Additionally, 2 pairs of slices remained unchanged, while only 2 pairs showed a worse alignment following the alignment procedure.

5.2 PCA-Based Clustering

Figure 7 displays a small section of the logs obtained when the K-means clustering is applied to the two leading principal components of the PCA. This leads to enhanced performance compared to the previous method as indicated by the additional boundary around the depth of 2130m on the EWL log, and approximately 2128m on the LWD log.

Because Figure 7 only provides the gamma-ray logs, while boundary identification involves the PCA of four well logs, Fig-

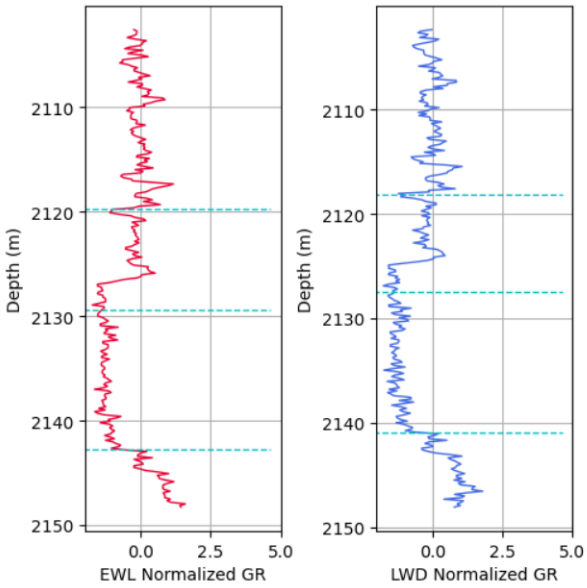


FIGURE 7: PCA-BASED CLUSTERING WAS UTILIZED TO ANALYZE NORMALIZED WELL LOGS (EWL AND LWD) FROM WELL 16/1-9, HIGHLIGHTING DELINEATED BOUNDARIES (REPRESENTED BY DOTTED HORIZONTAL LINES) THAT IDENTIFY DISTINCT GEOLOGICAL ZONES. TO ENHANCE CLARITY, A CONCISE 46-METER SEGMENT OF THE LOG IS DISPLAYED IN THE IMAGE. THE COMPLETE IMAGE IS ACCESSIBLE IN THE APPENDIX.

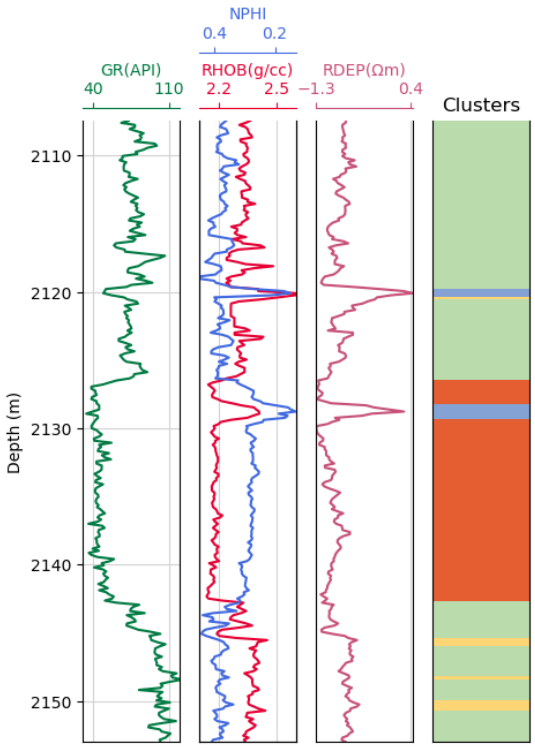


FIGURE 8: PCA-BASED CLUSTERS WITHIN A 46 M LONG SUBSECTION OF THE ABOUT 1300 M LONG EWL LOG FROM WELL 16/1-9. CLUSTERING IS PERFORMED ON NORMALIZED DATA, BUT CLUSTER ASSIGNMENT IS SHOWN HERE WITH MEASURED RECORDS.

ure 8 relates the identified boundaries to the cluster results from the same log section, and all four logs. These clusters highlight the identified boundaries on the EWL logs from well 16/1-9 with a noticeable anomaly at 2130m, that is not influencing the gamma-ray log. This exemplifies that the identified cluster boundary represents an actual geological boundary, even if it is not visible in each individual data log.

Out of the 60 boundaries suggested by PCA-based clustering, the boundary finalization approach removes 17 boundaries from each EWL and LWD, and depth alignment of EWL and LWD uses only the remaining 43 matching boundaries. Figure 9 demonstrates the alignment of the LWD log with the EWL log from well 16/1-9. Based on Pearson's correlation and visual inspection, this section appears appropriately depth-aligned. On inspection across 27 different windows, Table 3 shows the alignment of two sections got worse after the alignment procedure, one section remains unchanged, and the alignment of 24 windows has improved following the alignment procedure. This showcases the superior performance of PCA-based results compared to using only K-means on four-dimensional data. The full alignment result is available in the appendix.

5.3 Autoencoder-Based Clustering

A neural network-based autoencoder with a bottleneck layer comprising two neurons is trained using the log data from the well 16/1-9. Both our EWL and LWD log data were fed through the encoder portion of the trained autoencoder and the output from the bottleneck layer was used as a two-dimensional compressed signal for K-means clustering with four clusters, similar to the first two PCA components. Figure 10 shows three cluster boundaries as in the previous PCA-based approach, but in slightly different positions.

The autoencoder-based approach provides 60 boundaries. Our depth finalization approach removes five boundaries, leaving 55 matching boundaries in both EWL and LWD. Linear alignment post-processing aligns the LWD log with the EWL log from the same well, 16/1-9 as shown in Figure 11. Table 3 reveals that out of twenty-seven sections, the algorithm deteriorates the alignment in only one section, whilst alignment improves in all other sections. All alignment results are available in the appendix.

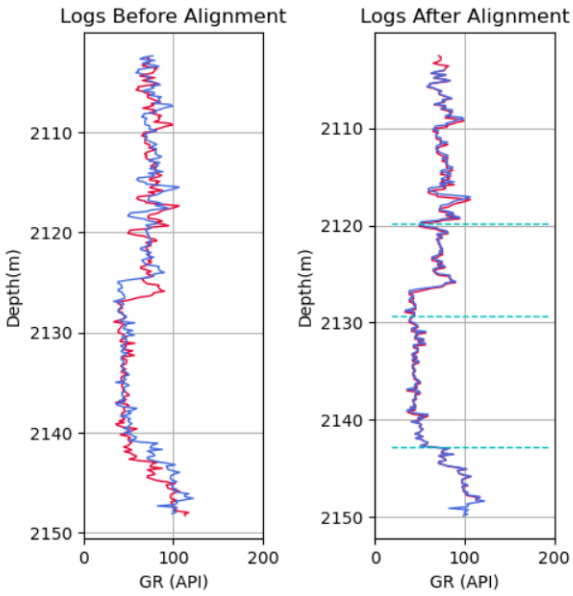


FIGURE 9: WELL LOG ALIGNMENT USING PCA-BASED METHOD. THE EWL LOG IS PLOTTED IN RED, AND THE LWD LOG IS PLOTTED IN BLUE. LOGS BEFORE ALIGNMENT ARE SHOWN ON THE LEFT, WHILE THOSE AFTER ALIGNMENT ARE ON THE RIGHT. THE COMPLETE IMAGE IS ACCESSIBLE IN THE APPENDIX.

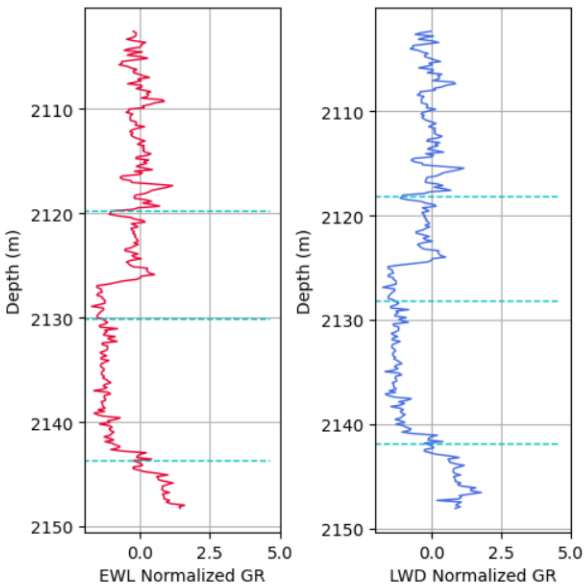


FIGURE 10: AUTOENCODER-BASED CLUSTERING WAS UTILIZED TO ANALYZE NORMALIZED WELL LOGS (EWL AND LWD) FROM WELL 16/1-9, HIGHLIGHTING DELINEATED BOUNDARIES (REPRESENTED BY DOTTED HORIZONTAL LINES) THAT IDENTIFY DISTINCT GEOLOGICAL ZONES. TO ENHANCE CLARITY, A CONCISE 46-METER SEGMENT OF THE LOG IS DISPLAYED IN THE IMAGE. THE COMPLETE IMAGE IS ACCESSIBLE IN THE APPENDIX.

6 DISCUSSION

Alignment of long well logs must avoid spurious misfits of phase shifted similarly looking signal subsequences. The three-step procedure presented here to reach this goal critically depends on the initial pre-processing step, which aims to identify robust and reliable matching boundaries in both records. It turns out that cluster boundaries from K-means clustering of multivariate datasets from well 16/1-6 A, reliably correspond to physical changes in the records, many of which are authentic lithological boundaries. PCA-based, and autoencoder-based pre-processing of the measurement data reduces the input dimension for K-means clustering from 4 to 2, which significantly improves reliability and robustness of the clustering result. The reason for this improvement seems to be that outliers, noise, data errors, lithological transitions and uncertainties are effectively removed or suppressed by the dimensional reduction, leaving primarily the robust and lithologically relevant information in the remaining data dimensions.

To demonstrate this effect, in Fig. 12 cluster boundaries are compared to independently determined lithological boundaries to distinguish authentic from spurious cluster boundaries. To acknowledge real uncertainty in lithological transition boundaries, a cluster boundary is considered 'true' if it lies within a 3-inch (7.6 cm) vicinity of the nearest lithological boundary. Cluster boundaries deviating more than 3 inches from the nearest lithological boundary are categorized as 'spurious' boundaries. The uncertainty limit is chosen because lithology is not directly observable from the surface, and 3 inch approximates the typical seismic wavelength and uncertainty. Even well log data have uncertainties in this range [25], because lithological interfaces commonly have fuzzy borders, and the resolution of most measuring methods also lies in this length scale. Table 2 compiles the results of comparing cluster and lithological boundaries and provides a comprehensive analysis of the true and spurious boundaries predicted by all three unsupervised methods.

Table 2 summarizes the validation results of three proposed algorithms applied to identify lithological boundaries in the well log data obtained from well 16/1-6 A.

The 'True boundaries' and 'Spurious boundaries' columns show the performance of each algorithm in detecting accurate and false boundaries, respectively. The autoencoder-based method outperformed others, detecting 37 true boundaries with only 3 spurious ones, indicating higher accuracy compared to K-means and PCA-based clustering.

This validation supports the use of unsupervised ML for

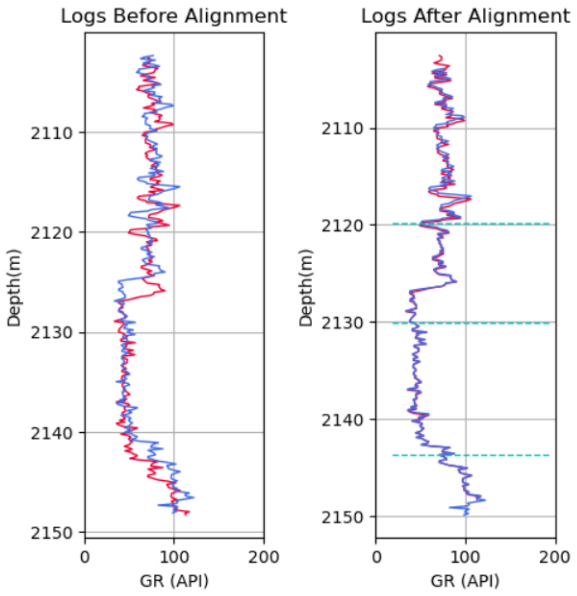


FIGURE 11: WELL LOG ALIGNMENT USING AUTOENCODER-BASED METHOD. THE EWL LOG IS PLOTTED IN RED, AND THE LWD LOG IS PLOTTED IN BLUE. LOGS BEFORE ALIGNMENT ARE SHOWN ON THE LEFT, WHILE THOSE AFTER ALIGNMENT ARE ON THE RIGHT. THE COMPLETE IMAGE IS ACCESSIBLE IN THE APPENDIX.

TABLE 2: NUMBER OF TRUE AND SPURIOUS BOUNDARIES PREDICTED BY THE UNSUPERVISED MODELS *K*-MEANS, PCA-BASED *K*-MEANS, AND AUTOENCODER-BASED *K*-MEANS ON DATA FROM WELL 16/1-6 A.

Method	True boundaries	Spurious boundaries
K-means clustering	30	10
PCA-based clustering	33	7
Autoencoder-based clustering	37	3

lithology detection, opening the door for possible depth alignment applications.

The test well logs were divided into 27 subsections, with each subsection measuring 46 meters in length. The performance of the various pre-processing methods was evaluated using Pearson's correlation and visual analysis. The comparative examination of three models is presented in Table 3.

Of the 27 subsections, the *K*-means model improved alignment in 23, worsened alignment in two, and had no effect in two instances. The PCA-based strategy resulted in a deterioration in alignment in 1 sample and did not produce any change in alignment in 2 samples. Significantly, the autoencoder-based technique demonstrated superior performance, with only one occurrence of inferior alignment. This result demonstrates that an autoencoder-based approach can effectively and automatically align desynchronized LWD logs with the reference EWL log from the same well.

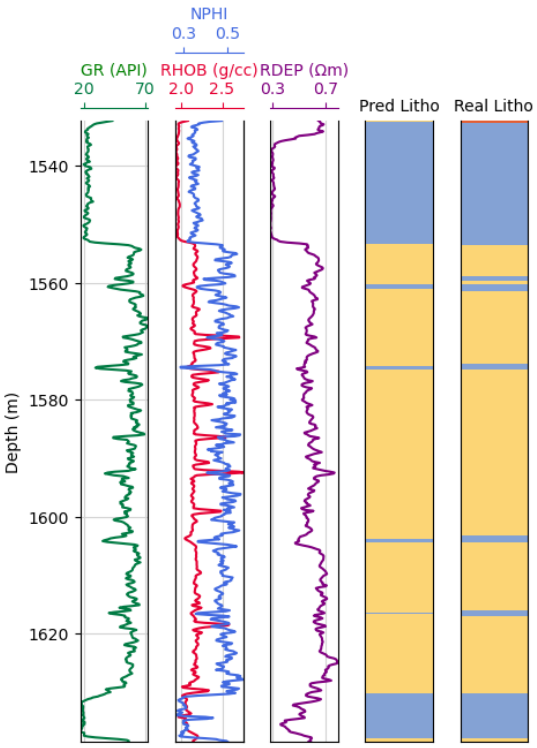


FIGURE 12: COMPARISON OF LITHOLOGY CLUSTERS: MEASURED LOGS INCLUDING GAMMA RAY (GR) IN GREEN, BULK DENSITY (RHOB) IN RED, NEUTRON POROSITY (NPHI) IN BLUE, AND DEEP RESISTIVITY (RDEP) IN PURPLE. THE 'PRED LITHO' COLUMN DISPLAYS PREDICTED LITHOLOGY CLUSTERS, WHILE THE 'REAL LITHO' COLUMN PRESENTS ACTUAL LITHOLOGY CLUSTERS.

TABLE 3: COMPARING THE EFFECTIVENESS OF THREE PROPOSED APPROACHES IN ENHANCING ALIGNMENT BETWEEN WELL LOG SUBSECTIONS.

Model	Log	Improved Samples	Worsened Samples	Not Changed Samples
K-means	GR	23	2	2
PCA-based	GR	24	1	2
Autoencoder-based	GR	26	1	0

An advantage of utilizing unsupervised machine learning for depth alignment is its ability to obviate the necessity of data labeling for algorithm training, resulting in substantial time savings. Significant amounts of time are only required for the first unsupervised training of the autoencoder. After determining the trained autoencoder weights, loading the weights and lowering the dimension of the well log data becomes highly time-efficient. The Python code used to process the 1300m long test well logs completes execution within 1 minute on a standard laptop equipped with 32 GB of RAM and 1 GB of graphics memory.

7 SUMMARY

To align desynchronized multivariate LWD logs with reference EWL logs from the same well, an unsupervised machine-learning technique is used for pre-processing by clustering, boundary detection, and boundary alignment before the final within-interval alignment of the logs. Pre-processing by K-means clustering on the original, and reduced dimensions of the four-dimensional well-log data has been analyzed, and evaluated using various metrics. The autoencoder-based clustering algorithm performed better than the PCA-based and the original K-means algorithms in detecting robust lithological boundaries on both well logs. A correlation-based matching approach to filter the boundaries was successfully compared with visual alignment results and Pearson's correlation. The technique so far successfully handles erroneous well logs from vertical wells, but still allows for improvement in dealing with subtle lithological variations, and in refining the boundary identification process. To reach production-ready quality will require to expand the training dataset by more well logs, applying different weights to different logs, refining the correlation-based matching algorithm, and implement an optimal algorithm for post-processing of the within-interval alignment. Further improvements are possible by evaluating the performance on various datasets, and exploring alternative clustering methods. Already the current technique demonstrates the power and potential of unsupervised machine learning for clustering and aligning well logs, and provides a foundation for future research on this topic.

ACKNOWLEDGMENT

This work was conducted as part of the Ph.D. program at the Norwegian University of Science and Technology (NTNU), Trondheim, Norway, within the BRU21 research program. We kindly acknowledge funding and support by AkerBP. Well log data were made available through the Norwegian National Data Repository DISKOS of the Norwegian Offshore Directorate.

REFERENCES

[1] Ellis, D. V., and Singer, J. M., 2007. *Well logging for earth scientists*, Vol. 692. Springer.

[2] Bolt, H., 2016. "Wireline logging depth quality improvement: Methodology review and elastic-stretch correction". *Petrophysics,* ***57***(03), pp. 294–310.

[3] Theys, P. P., 1999. *Log data acquisition and quality control*. Editions Technip.

[4] Bishop, C. M., and Nasrabadi, N. M., 2006. *Pattern recognition and machine learning*, Vol. 4. Springer.

[5] Torres Caceres, V. A., Duffaut, K., Yazidi, A., Westad, F., and Johansen, Y. B., 2022. "Automated well log depth matching: Late fusion multimodal deep learning". *Geophysical Prospecting*.

[6] Zimmermann, T., Liang, L., and Zeroug, S., 2018. "Machine-learning-based automatic well-log depth matching". *Petrophysics,* ***59***(06), pp. 863–872.

[7] Le, T., Liang, L., Zimmermann, T., Zeroug, S., and Heliot, D., 2019. "A machine-learning framework for automating well-log depth matching". *Petrophysics,* ***60***(05), pp. 585–595.

[8] Wang, S., Shen, Q., Wu, X., and Chen, J., 2020. "Automated gamma-ray log pattern alignment and depth matching by machine learning". *Interpretation,* ***8***(3), pp. SL25–SL34.

[9] Acharya, S., and Fabian, K., 2023. "Dynamic depth alignment of well-bore measurements using machine learning". In 84th EAGE Annual Conference & Exhibition, Vol. 2023, European Association of Geoscientists & Engineers, pp. 1–5.

[10] Mishra, A., Sharma, A., and Patidar, A. K., 2022. "Evaluation and development of a predictive model for geophysical well log data analysis and reservoir characterization: machine learning applications to lithology prediction". *Natural Resources Research,* ***31***(6), pp. 3195–3222.

[11] Singh, H., Seol, Y., and Myshakin, E. M., 2020. "Automated well-log processing and lithology classification by identifying optimal features through unsupervised and supervised machine-learning algorithms". *SPE Journal,* ***25***(05), pp. 2778–2800.

[12] Diskos database. Accessed: 22.12.2023.

[13] Bormann, P., Aursand, P., Dilib, F., Dischington, P., and Manral, S., 2022. 020 force machine learning contest.

[14] James, G., Witten, D., Hastie, T., Tibshirani, R., et al., 2013. *An introduction to statistical learning*, Vol. 112. Springer.

[15] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., 2011. "Scikit-learn: Machine learning in Python". *Journal of Machine Learning Research,* ***12***, pp. 2825–2830.

[16] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

[17] Hartigan, J. A., and Wong, M. A., 1979. "Algorithm as 136: A k-means clustering algorithm". *Journal of the royal statistical society. series c (applied statistics),* ***28***(1), pp. 100–108.

[18] Arthur, D., and Vassilvitskii, S., 2007. "K-means++ the advantages of careful seeding". In Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, pp. 1027–1035.

[19] Ketchen, D. J., and Shook, C. L., 1996. "The application of cluster analysis in strategic management research: an analysis and critique". *Strategic management journal,* ***17***(6), pp. 441–458.

[20] Rousseeuw, P. J., 1987. "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis". *Journal of Computational and Applied Mathematics,* ***20***, Nov, p. 53–65.

[21] Pearson, K., 1901. "Liii. on lines and planes of closest fit to systems of points in space". *The London, Edinburgh, and Dublin philosophical magazine and journal of science,* ***2***(11), pp. 559–572.

[22] Jolliffe, I. T., 2002. "Choosing a subset of principal components or variables". *Principal component analysis*, pp. 111–149.

[23] Kramer, M. A., 1991. "Nonlinear principal component analysis using autoassociative neural networks". *AIChE journal,* ***37***(2), pp. 233–243.

[24] Hinton, G. E., and Zemel, R., 1993. "Autoencoders, minimum description length and helmholtz free energy". *Advances in neural information processing systems,* ***6***.

[25] Wellmann, J. F., Horowitz, F. G., Schill, E., and Regenauer-Lieb, K., 2010. "Towards incorporating uncertainty of structural data in 3d geological inversion". *Tectonophysics,* ***490***(3-4), pp. 141–151.

Appendix A: Supplementary Materials

The EWL and LWD gamma-ray logs acquired from well 16/1-9, were sliced with a window size of 301 units, approximately equivalent to 46 meters. This uniform window size was consistently applied across all three algorithms employed. The presented table summarizes the results obtained from the utilization of *K*-means, PCA-based, and Autoencoder-based algorithms.

The initial column shows the distinct window numbers from 1 to 27, while the subsequent columns show the results produced by each respective algorithm. '+' signifies an improved refinement in well log alignment after applying the algorithm; 'o' indicates no discernible alteration post-alignment; and '-' denotes a deterioration in alignment following the algorithm's implementation.

All graphs presented in this paper are only partial representations of the complete well logs; the full logs and alignment are available via the following links:

1. *K*-means clusters: `http://htmlpreview.github.io/?https://github.com/Sushil-Acharya/Graphs/blob/main/K-means%20based%20clusters.html`
2. PCA-based clusters: `http://htmlpreview.github.io/?https://github.com/Sushil-Acharya/Graphs/blob/main/PCA%20based%20clusters.html`
3. Autoencoder-based clusters `http://htmlpreview.github.io/?https://github.com/Sushil-Acharya/Graphs/blob/main/Autoencoder%20based%20clusters.html`
1. *K*-means-based alignment: `http://htmlpreview.github.io/?https://github.com/Sushil-Acharya/Graphs/blob/main/K_means_based_log_alignment.html`
2. PCA-based alignment: `http://htmlpreview.github.io/?https://github.com/Sushil-Acharya/Graphs/blob/main/PCA_based_log_alignment.html`
3. Autoencoder-based alignment `http://htmlpreview.github.io/?https://github.com/Sushil-Acharya/Graphs/blob/main/Autoencoder_based_log_alignment.html`

Window	*K*-means	PCA	Autoencoder
1	+	+	+
2	-	-	+
3	-	-	+
4	+	+	+
5	+	+	+
6	+	+	+
7	+	+	+
8	+	+	+
9	+	+	+
10	+	+	+
11	o	+	+
12	+	+	+
13	+	+	+
14	+	+	-
15	+	o	+
16	+	+	+
17	+	+	+
18	+	+	+
19	+	+	+
20	+	+	+
21	+	+	+
22	+	+	+
23	+	+	+
24	+	+	+
25	+	+	+
26	+	+	+
27	o	+	+

TABLE 4: COMPARISON OF THREE ALIGNMENT APPROACHES ON DIFFERENT SLICES (OR WINDOWS) OF THE GAMMA-RAY LOGS FROM THE WELL 16/1-9. '+' SIGNIFIES IMPROVED WELL LOG ALIGNMENT AFTER APPLYING THE ALGORITHM, 'O' INDICATES NO DISCERNIBLE EFFECT, AND '-' DENOTES A DETERIORATION IN ALIGNMENT.