Semblance-based velocity picking using unsupervised machine learning

Ninoska Amundaray and Daniel Trad

ABSTRACT

Normal moveout (NMO) correction depends on the identification of optimal velocity-time pairs to flatten the hyperbolic character associated with seismic reflections. Semblance panels are ideal attributes to accomplish this task. However, they are highly affected by the level of noise in the data and poor calculations for short offsets. These are often overcome with additional seismic attributes, which we suggest to replace with the introduction of velocity trends based on semblance. In this study, we demonstrate that our method works as an adequate filtering technique, capable to generate inputs for applications of unsupervised machine learning (ML) in velocity analysis. The performance of three different types of clustering methods known as K-Means, Gaussian Mixture Models and DBSCAN, is investigated by the identification of velocity-time pairs to guide the NMO correction in two datasets simulated for the Marmousi model. In both tests, deep reflectors are corrected independently of the clustering technique used; whereas shallow events are only flattened at near and mid offsets, and under corrected or stretch at far offsets.

INTRODUCTION

Velocity analysis workflows rely on finding the best velocity-time picks to correct the hyperbolic character of seismic reflections. To achieve this, they scan different values of velocity using various methods to flatten seismic gathers and generate a final velocity spectrum (Yilmaz, 2000). One the most used attributes in this process is semblance, which is a measure of similarity of a number of traces after normal moveout (NMO) has been applied. Although in theory, only effective moveout velocities should maximize the likelihood of flattening seismic events, this attribute is heavily affected by the level of noise in the data. Hence, semblance panels tend to display velocity trends that might be driven by a combination of factors aside from geology.

While authors recognize some of the downsides of semblance, its utility perseveres. Throughout the last years, both, auto-pickers and machine learning (ML) algorithms (Smith, 2017; bin Waheed et al., 2019) have been tested with synthetic and real seismic data. Corrected records using attributes, as semblance and others, have reproduced coherent seismic records with lateral continuity. This suggests them as a favourable method to reduce human time expenditure in velocity picking. However, some of these applications depend in more than one attribute and that it is not always available for some datasets.

The statement above motivates us to investigate other strategies to generate velocity-time pairs for NMO correction utilizing semblance panels as the only attribute. In this study, we propose filtering our training data with a "soft" threshold based on velocity trends and percentiles. Taking these as inputs, we applied three distinct clustering analyses to guide our moveout correction and compare their performance on common midpoint (CMP) gathers.

UNSUPERVISED MACHINE LEARNING

Unsupervised machine learning refers to algorithms that learn to identify patterns and features in training data without external aid or prior information. This self-training process relies on finding correlations and dissimilarities from the inputs to a system. Cluster analysis are a common application of unsupervised learning because they aim to organize unclassified data into labeled groups. To find a notion of similarity, clustering can use measures of distance between points, probability models, density-based approaches or graph distance techniques.

There exists a large collection of cluster analysis tools with proven applications in multi-dimensional signal analysis. For this study, we examined the following three cluster algorithms with continuous semblance panels.

K-Means

In this technique, unclassified data is partitioned based on a user-defined number of cluster (referred as *K*). This is an optimization problem, where the objective function is to minimize the distance between all identified cluster centroids and their surrounding points. Hence, convergence is accomplished when either all centroids do not change in-between iterations or after a number of maximum iterations is reached.

Traditionally, in this analysis, the Euclidean distance has been used as a metric to compute distances, following:

$$d_{ik} = |x_i - \mu_k|^2 = (x_i - \mu_k)^T \sum_k^{-1} (x_i - \mu_k) = (x_i - \mu_k)^T (x_i - \mu_k) \tag{1}$$

Where μ_k is the mean (centroid) of the *K*-cluster, x_i are input points and the summation is the covariance matrix, which in Euclidean space is a diagonal matrix. The latter assumes an equal variance between values, making it difficult to detect clusters with non-spherical shapes or with widely different sizes or densities (Tan et al., 2006), which might require a transform and standardization of the data space.

Gaussian Mixture Models (GMM)

GMM evaluates the likelihood of a sample point belonging to a superposition of Gaussian distributions given the mean, the covariance matrix and the weight of each distribution (i.e., K clusters). Hence, it is a probabilistic clustering approach, which can be defined using:

$$p(x_n|K_k) = \frac{1}{(2\pi)^{N/2} |\sum_k^{-1}|} exp[-\frac{1}{2}(x_i - \mu_k)^T \sum_k^{-1} (x_i - \mu_k)] \tag{2}$$

Where p is the probability of a training point x_i, given a kth cluster, which is iteratively updated following an expectation maximization algorithm until convergence (Russell, 2020). Notice, this approach allows variance between points because it utilizes the full covariance

matrix and weighting factors. Hence, GMM can deal with training data with non-spherical shapes without the need to standardize it.

Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

Density-based methods rely on the location of regions of high-density and low-density training samples in the data space to determine a cluster (Tan et al., 2006). Delineation of these regions depends in two hyper-parameters known as Epsilon (*Eps*) and number of points (*minPts*). Here, the former states the radius of a circle around each point to be analyzed; whereas the latter indicates the minimum number of points that will delimit a "dense" region. Using both, training samples will be labeled as core points when they satisfy both parameters, borders points when they are reachable following the Eps distance and noise point when they do not meet neither criterion.

Although, this technique can handle clusters of different shapes, the Euclidean distance is utilized to locate data points in space through Epsilon. Hence, it is advisable to transform and standardize datasets with large variance.

METHODOLOGY

Figure 1 summarizes the general workflow employed during this investigation. Each column represents the followed steps in each clustering analysis after a common phase of data input and initial threshold definition. From left to right in the workflow, left column comprises *K*-Means clustering, center column corresponds to Gaussian Mixture Models and last column is DBSCAN clustering.

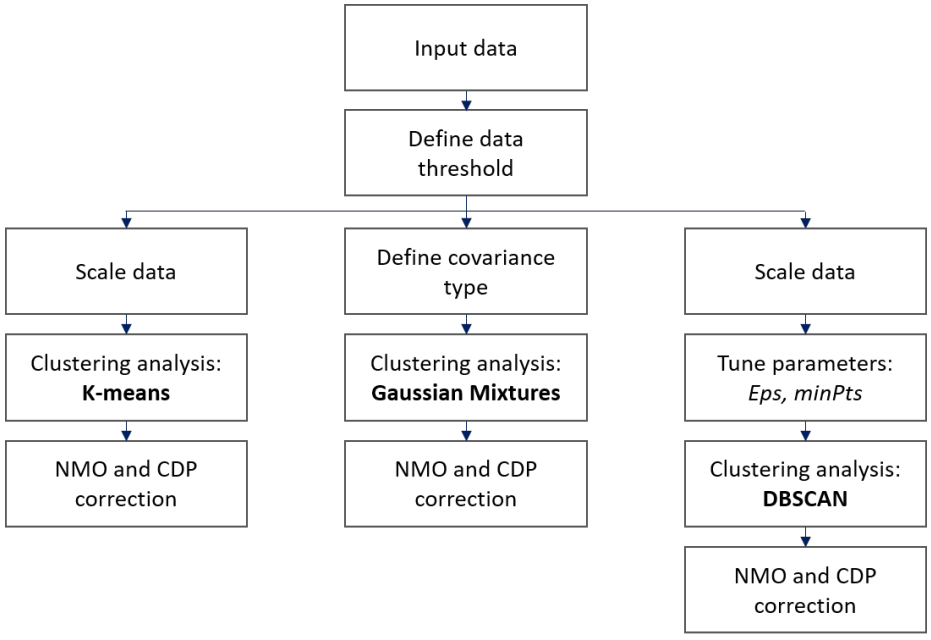

FIG. 1. Workflow used during this investigation. Cluster analyses shared initial steps. After defining a data threshold, bottom columns represent the steps used for each clustering technique. Here, left column resumes *K*-Means Clustering, Center column summarizes Gaussian Mixture Models and right column resumes DBSCAN.

Semblance panels were generated using common midpoint gathers in Seismic Unix. After an initial inspection, we manually classified defining a limit our semblance into binary maps. However, this proved to be a "hard" threshold because several meaningful points did not pass the initial classification, as it can be seen in Figure 2(b) and Figure 5(b). To solve this, we set a "softer" threshold integrating two velocity trends and the usage of percentiles to define our data limits, following:

$$s\_threshold = \begin{cases} 1, & vel_{low} + tol \leq r \leq vel_{up} + tol \\ 0, & otherwise \end{cases} \tag{3}$$

Here, vel_{low} is the low velocity tolerance, vel_{up} is the upper velocity tolerance and r is given by:

$$r = \frac{PxN}{100} \tag{4}$$

Where P is the percentile selected by the user, N is the number of data points and r is the rank given by the quotient of these two parameters divided by 100.

Following the "soft" approach, we allowed more sample points to pass our initial filtering; while we still restricted our data space from non-meaningful information (e.g., highly coherence zones on the top and bottom semblance), as Figure 2(c) and Figure 5(c) display.

For clustering analyses that rely on Euclidean distance for their performance, we scaled our filtered semblance on both variables (i.e., velocity and time) and used the data in the transformed space until we reach the final normal moveout correction. Without this step, attributes in the velocity domain were dominating cluster definitions, overlooking any information provided in the time domain.

We tested different number of clusters for *K*-Means and GMM, keeping the ones that better described our data and conditioning our results, which in a well-known weakness of these clustering analyses. Meanwhile for DBSCAN, we tuned *Eps* and *minPts* parameters by plotting the sorted distances of the fifth nearest neighbour of each point on a k-nearest neighbour graph, and identifying the values where there was a sharp increase, as suggested by Tan et al. (2006). Through this process, clusters were computed identifying noisy samples by the algorithm itself and decreasing external biases.

After each cluster centers was found, we utilized a linear interpolation between these and estimated the NMO correction to use in their corresponding CMP gathers with the expectations to flatten all the hyperbolic events associated with reflections.

STUDY CASE

We studied the performance of clustering analyses using two surface seismic data simulations for the Marmousi model, with and without surface multiples (two models with shallow or deep water). The goal was to compare the introduction of coherent noise (multiples) which were present in the deep-water setting.

Marmousi model test

Figure 2(a) shows the original semblance. Here, the velocity-time pairs suggested to optimally correct traces are given a value of one or close to one; hence, these would be the ideal candidates to be picked as cluster centers. However, as Figure 2(b) illustrates, setting a relevant threshold to initiate clustering is not trivial but still compulsory, because semblance is vulnerable to noise and many clustering algorithms do not handle well noise and outliers, especially if Euclidean distance is utilized. Still, we observed that our "soft" limit allows us to fairly discriminate between outliers and relevant data samples, and we used this as input data for clustering.

Cluster outputs from analyses are overlaying the original semblance in Figure 3. Two distinct outcomes should be highlighted in our solution. First, incorporating prior knowledge from the user, through the definition of the number of clusters, might lead to results that differ widely from data-driven approaches. In this case, DBSCAN identified as cluster centers only one-third of the number that we deemed as optimal from the initial semblance plot. Secondly, once data is properly scaled, GMM and K-Means will converge towards a similar solution, proving that removing variance in the training dataset will satisfy Equation 1.

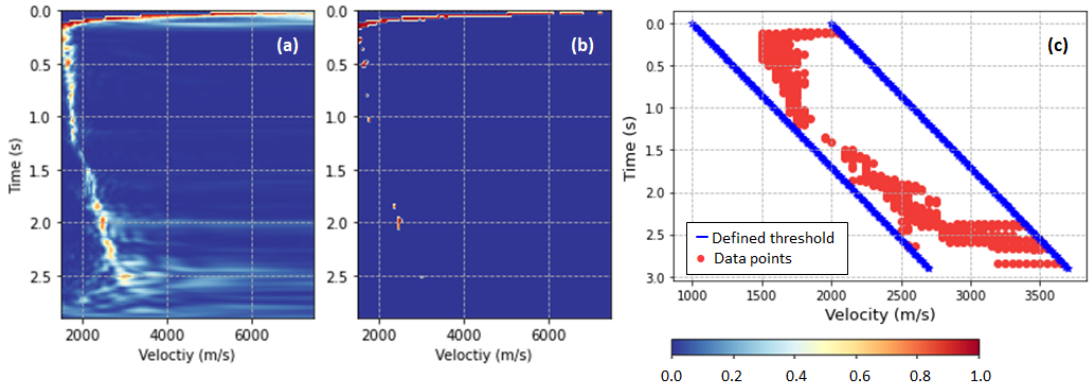

FIG. 2. Semblance panels from Marmousi model. (a) It is the original semblance calculated using Seismic Unix, (b) shows the results after filtering the initial semblance with a "hard" threshold, and (c) shows the results after filtering the initial semblance with a "soft" threshold.

As a last step in our workflow, we computed NMO profiles utilizing cluster centers to guide our correction. Figure 4 shows the comparison between the original CMP gather and the computed gathers after normal moveout. In general terms, we observe that shallow events at mid and far offset are still under corrected when GMM and DBSCAN are used as solutions; whereas, when K-Means results are applied, most reflectors are flattened or slightly stretched at far offsets. Although, we recognize that density-based analysis oversimplifies the velocity-time picks required for the correction. Probabilistic clustering, with two times more information to guide the velocity correction, reproduced a comparable final image. Lastly, deep reflectors in all corrected gathers were properly flattened independently of which machine learning algorithm was used because similar clusters centers were identified, as Figure 8(a) shows.

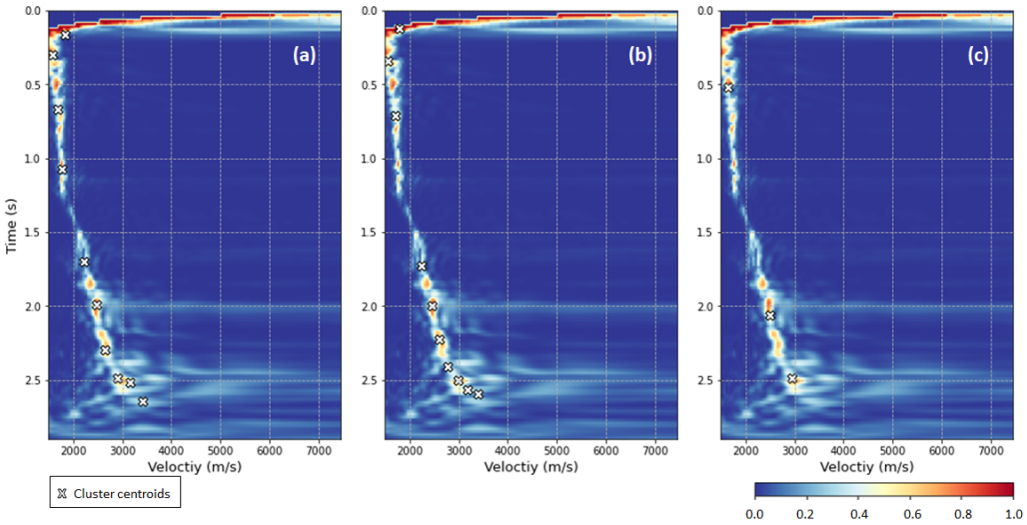

FIG. 3. Original semblance panels overlaid with the identified cluster centers using unsupervised learning. (a) Shows results using *K*-Means, (b) displays results using Gaussian Mixture Models, and (c) shows results using DBSCAN. In all panels, white markers represent cluster centers.

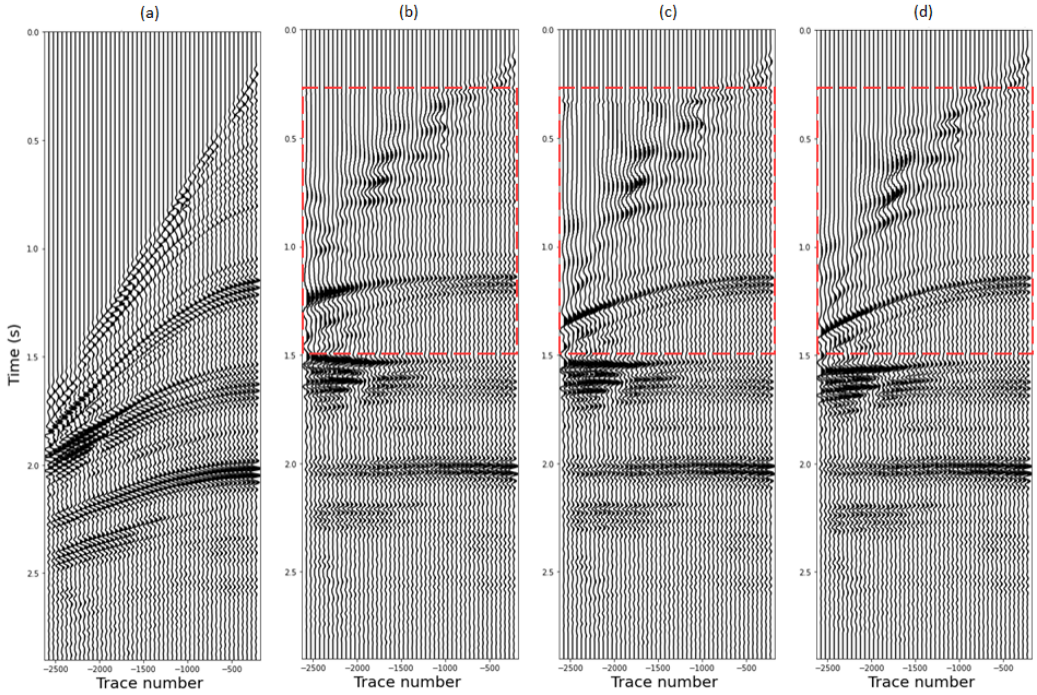

FIG. 4. Common midpoint gathers. (a) Shows original gather, (b) displays corrected gather using cluster centers from *K* -Means, (c) shows corrected gather using cluster from Gaussian Mixture Models, and (d) displays corrected gather using cluster centers from DBSCAN. Red dashed boxes illustrate under correction of shallow events.

Marmousi2 model test

Marine seismic data have surface multiples usually with stronger amplitudes than primaries. These are traceable in semblance panels because strong maxima will be repeated following the primary trend but at later time, deviating the real velocity trend. Figure 5(a)

displays a semblance panel with multiples. Here, areas of high correlation are horizontally stretched from top to bottom. At these locations, the attribute suggests several velocity-time pairs to employ in the correction, leading to the formation of a wide tail in the horizontal axis (i.e., velocity) within a short time window.

After initial filtering, the binary map set as input for machine learning still possesses areas influenced by multiples at the bottom sections, as Figure 5(c) illustrates. Some clusters centers are identified in those problematic areas, as highlighted with black boxes in Figure 6(a)-(b) for GMM and *K*-Means solutions. Both, distance-based and probabilistic algorithms regard most training samples while assigning clusters, giving them some weight. Therefore, these results should not be a surprise. Likewise, when hyper-parameters are properly tuned, our study demonstrates that DBSCAN can distinguish between outliers and meaningful training samples. Only if those values are not located in a highly dense area where it could still erroneously pick cluster centroids, as indicated with a black box at the lowermost section in Figure 6(c).

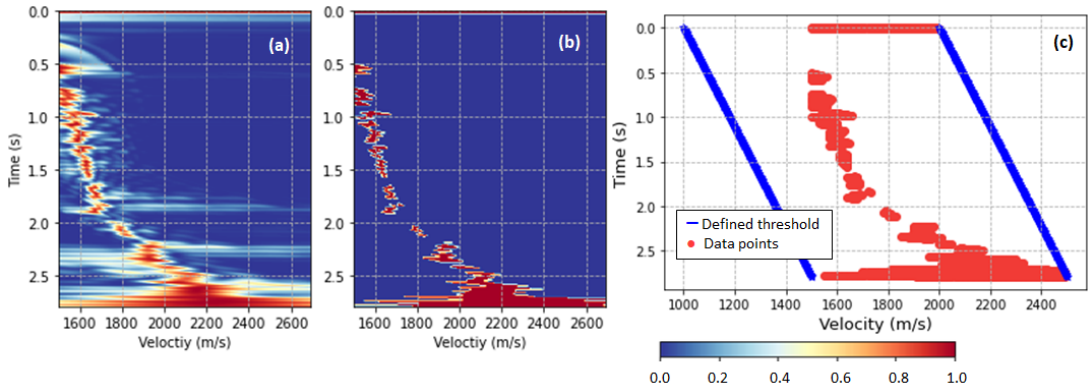

FIG. 5. Semblance panels from Marmousi model in a deep-water setting. (a) original semblance, (b) semblance after "hard" thresholding, and (c) semblance after "soft" thresholding.

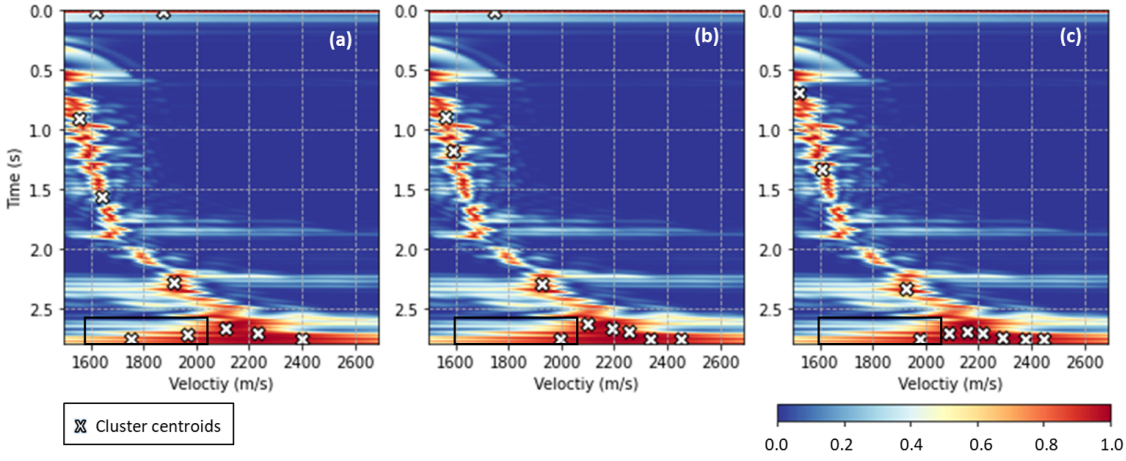

FIG. 6. Original semblance panels for Marmousi model in a deep-water environment overlaid with the identified cluster centers using unsupervised learning. (a) Shows results using *K*-Means, (b) displays results using Gaussian Mixture Models, and (c) shows results using DBSCAN. In all panels, white markers represent cluster centers; whereas black boxes highlight clusters resulting from low resolution semblance due to short offsets and stretch mute.

As in the previous test, we estimated the normal-moveout curve to correct our CMP gathers. In this case, we observe the linear event associated with refracted waves highlighted with a red shape in Figure 7. This effect is usually corrected through other processing steps in a velocity analysis workflow; hence, it is not our concern in this investigation.

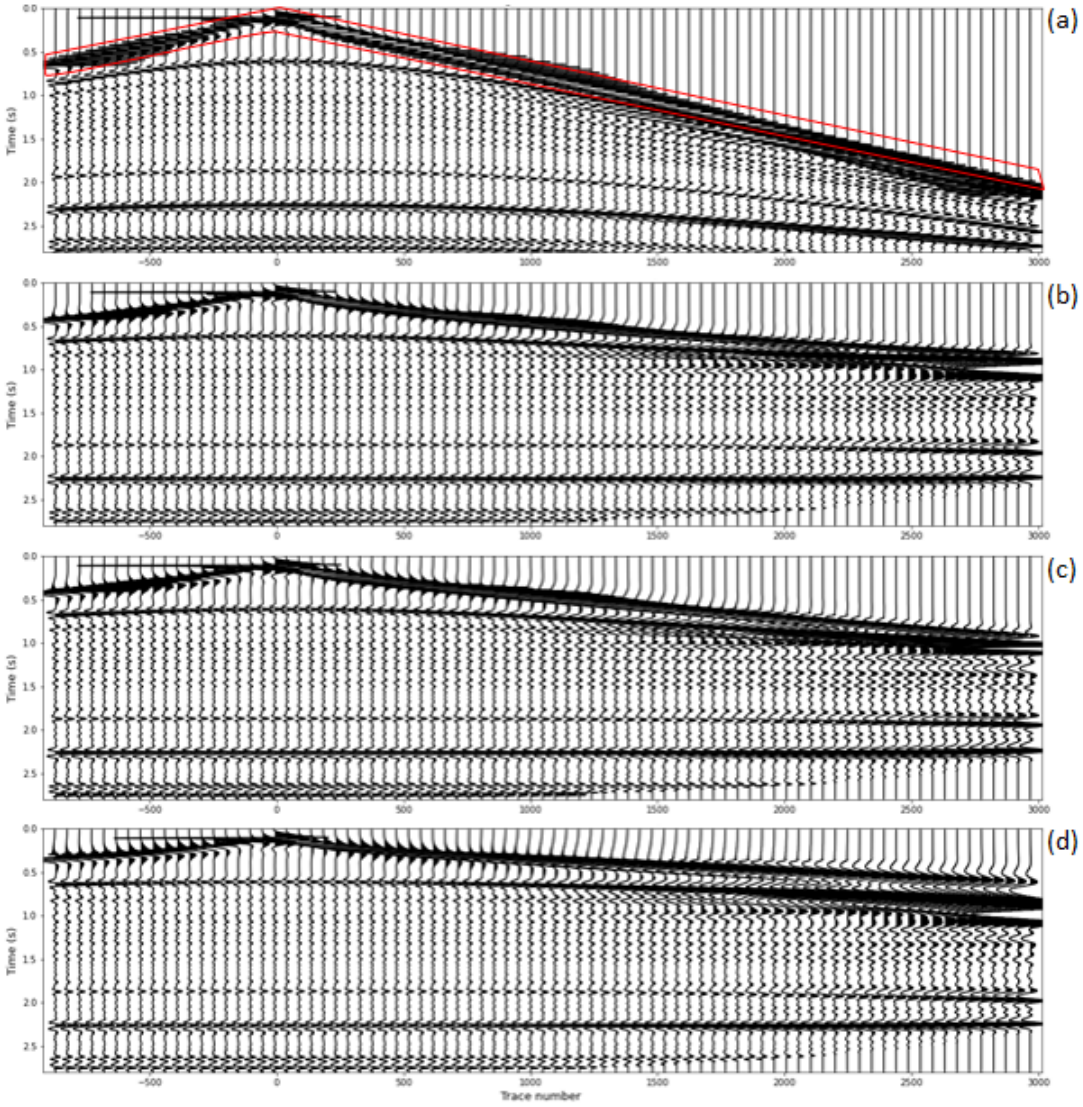

FIG. 7. Common midpoint gathers for Marmousi model in a deep-water setting. (a) Shows original gather, (b) displays corrected gather using cluster centers from *K*-Means, (c) shows corrected gather using cluster from Gaussian Mixture Models, and (d) displays corrected gather using cluster centers from DBSCAN. Red shape on the top panel signals the refracted wave in the seismic record.

Comparison between common mid points gathers after the velocity correction is applied using clustering reproduces analogous records. Most events are evenly flattened at near and mid offsets; whereas seismic traces are under corrected and stretched at far offsets. Nevertheless, there is a noticeable difference for the first reflector in all seismic images. For this event, discrimination of noisy data and outliers as suggested by DBSCAN, appears to improve the correction of the shallow hyperbolic trajectory.

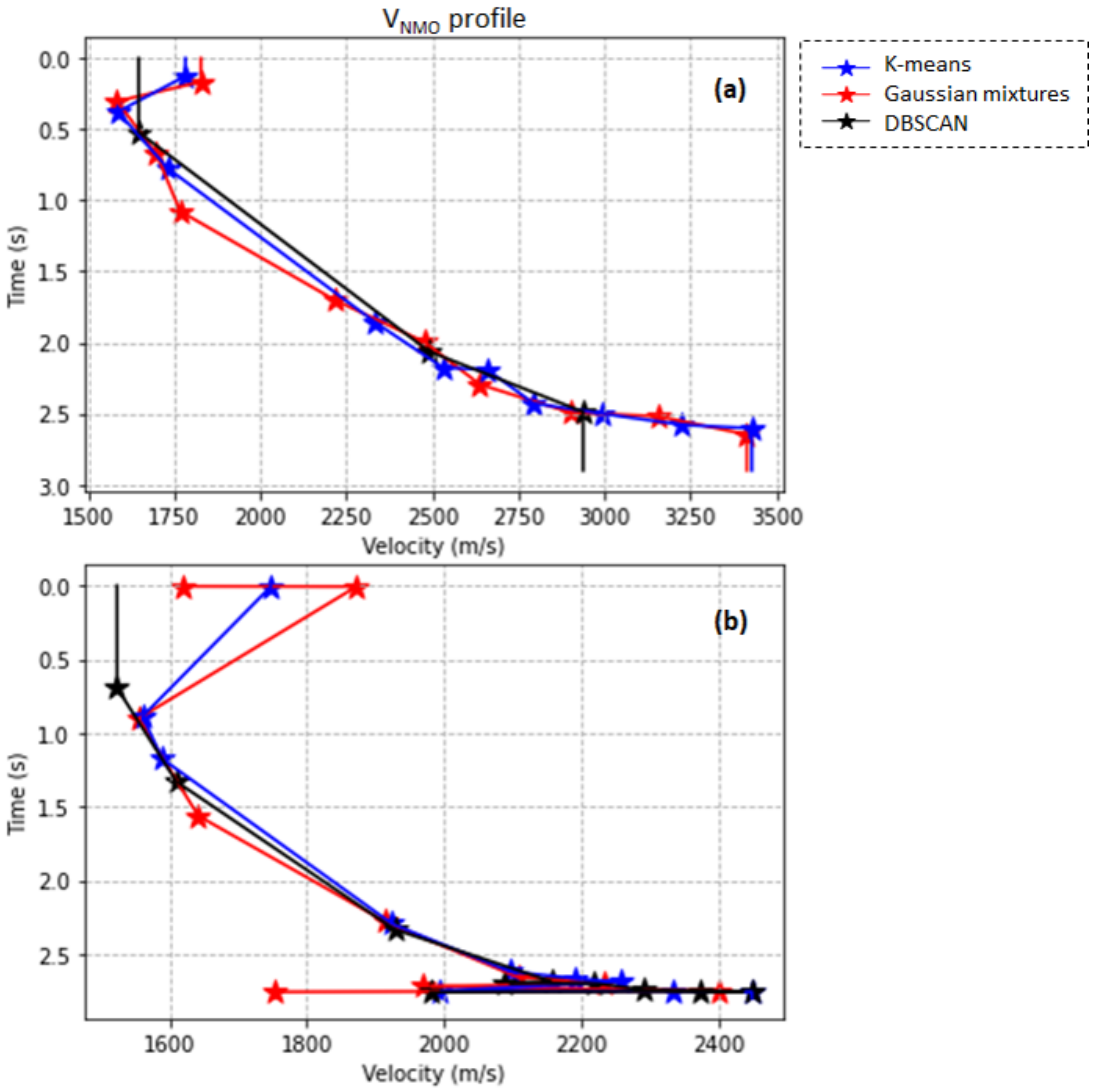

FIG. 8. Computed Normal moveout curves using interpolation between cluster centers (a) Shows the comparison between velocity profiles for Marmousi model with a shallow water layer on top, and (b) shows the comparison between velocity profiles for Marmousi model in a deep-water environment. In both panels, blue lines were calculated using *K*-Means solution, red lines were computed using Gaussian Mixture Models, and black lines were calculated using DBSCAN solution.

CONCLUSIONS

In this investigation, we applied three clustering algorithms and analyzed the performance of each, to generate velocity picks from semblance panels for NMO correction. We defined a workflow adjusting our data space and estimating the required hyper-parameters governing each analysis. To input training samples for unsupervised learning, we proved that a "soft" threshold can be used as only attribute to filter noisy data from semblance, maintaining a large number of relevant velocity-time pairs. This filter does not eliminate all the outliers but it lessens their existence and limit clustering failure.

Noisy samples are treated differently in each algorithm. Both, K-Means and Gaussian Mixture Models include them in their identification process of cluster centers; whereas DBSCAN labels them as outliers and ignore their location to assign clusters. Hence, further semblance filtering could be done with this algorithm if it is properly tuned. However, this step will introduce additional assumptions from a user's standpoint, and it could oversimplify the problem.

Applications of normal moveout correction in CMP gathers validate the performance of the identified cluster centers as velocity picks to flatten hyperbolic events in seismic. In both tests, deep reflectors are corrected better than shallow events. In the latter case, results suggest that seismic traces would need higher velocity picks to avoid being stretched or under corrected.

Lastly, future work could benefit from supervised machine learning to generate smoother NMO curves, and potentially, improving our solutions. This could be tested using human-guided picks or data-guide picks from clustering analyses to assist the algorithm and compare learning between a human interpreter and a machine.

ACKNOWLEDGEMENTS

We thank the sponsors of CREWES for continued support. This work was funded by CREWES industrial sponsors and NSERC (Natural Science and Engineering Research Council of Canada) through the grant CRDPJ 543578-19.

REFERENCES

bin Waheed, U., Al-Zahrani, S., and Hanafy, S. M., 2019, Machine learning algorithms for automatic velocity picking: K-means vs. dbscan: SEG Technical Program Expanded Abstracts, 5110–5114.

Russell, B., 2020, Unsupervised seismic facies classification using k-means and gaussian mixture modeling: CREWES Research Report, **32**, No. 51.

Smith, K., 2017, Machine learning assisted velocity autopicking: SEG Technical Program Expanded Abstracts, 5686–5690.

Tan, P.-N., Steinbach, M., and Kumar, V., 2006, Cluster Analysis: Basic Concepts and Algorithms: Addison-Wesley.

Yilmaz, O., 2000, Seismic data analysis: Soc. of Expl.