# Identifying Discrete Fracture Networks by Clustering with Microseismic Data

Scott McKean, Jeffrey A. Priest, David W. Eaton
*University of Calgary, Calgary, Alberta, Canada*

ABSTRACT
Microseismic monitoring is common in the mining and petroleum industries. Pattern recognition to identify faults, fractures, and damage zones in the microseismic data can be a challenge due to the density and number of observations and their multiple attributes. This study applies unsupervised machine learning techniques to identify features in a dataset from a hydraulic fracturing program in the Duvernay Formation. It compares partitional, hierarchical, and density based clustering methods using various validation statistics and with multiple subsets of data attributes. The study shows that lower dimensional datasets tend to yield the best results, and that it is possible to cluster microseismic data using unsupervised techniques.

RÉSUMÉ
La surveillance microsismique est importante pour les industries minières et pétrolières. La reconnaissance des formes pour identifier les failles, les fractures et les zones endommagées dans les observations microsismiques est difficile en raison de la densité et le nombre d'observations et de leurs multiples attributs. Cette étude applique des techniques d'apprentissage machine sans supervision pour identifier les caractéristiques des observations d'un programme de fracturation hydraulique dans la Formation Duvernay. Il compare les méthodes de classification par partition, hiérarchique, et basée sur la densité en utilisant diverses statistiques de validation et avec plusieurs sous-ensembles d'attributs de données. L'étude montre que les ensembles de données avec moins d'attributs donner les meilleurs résultats, et qu'il est possible de regrouper les observations microsismiques en utilisant des techniques non supervisées.

## 1 INTRODUCTION

This study investigates the applicability of unsupervised machine learning for identifying discrete fracture networks (DFNs) from microseismic data. DFNs can be the primary control on numerous subsurface processes. Contaminant transport in fractured rock occurs almost exclusively through fractures (Berkowitz 2002). Fracture networks are essential for enhanced geothermal energy and unconventional resources (Tezuka and Nittsuma 2000). They control the stability of most underground mining structures and block caving (Hudyma 2008). Pressure propagation through fractures is also one of the most likely causes of induced seismicity during hydraulic fracturing (Maxwell et al. 2015).

Subsurface characterization and engineering produce a lot of temporal and geospatial data. A 'lot' is a relative term that ranges from several hundred data points to big datasets that quickly overwhelm the memory of a single processor. Engineers use pattern recognition as a key tool for interpreting this data, but the amount and variety of the data sources can quickly overcome conventional or manual pattern identification techniques. This problem is also compounded by multivariate or multidimensional data sets, where engineers need to interpret multiple attributes with each other.

The large number of hydraulic fracturing stages and density of microseismic events makes manual pattern recognition nearly intractable in microseismic data from hydraulic fracturing or block caving problems. The objective of this study is to show that an unsupervised machine-learning pipeline can be used to group microseismic events for further analysis and DFN characterization. This type of approach is essential for providing an unbiased, automated, and scalable method of interpreting microseismic data.

Fortunately, this problem is routinely tackled by machine learning - a set of rules that allows systems to learn directly from examples, data, and experience (Royal Society 2017). Machine learning can be supervised or unsupervised. Supervised learning consists of training models to predict or classify new data based on labeled training data sets. Unsupervised learning, which is also referred to as clustering or pattern recognition, groups data based on similarities in attributes.

### 1.1 Study Area & Dataset

The data in this study came from an operator-processed dataset that was collected during an eight well hydraulic fracturing program in the Duvernay Formation. The microseismic cloud generated during the stimulation is illustrated in Figure 1. The Duvernay is an unconventional oil and gas play located at a depth of approximately 2,500 metres below sea level (mbsl). Its lithology roughly consists of siliceous clay-rich mudstones, interbedded carbonate and mudstone, and dolomitic carbonate facies. The Duvernay is a prolific producer due to its high reservoir pressure and susceptibility to hydraulic stimulation.

The dataset consisted of 17,681 microseismic observations, 153 hydraulic fracturing stages, and numerous attributes including location, time, magnitude, and moment tensor data, etc. This microseismic catalogue is depicted by a cumulative distribution and probability density histogram in Figure 2. The magnitude of completeness of the dataset was approximately -1.5, with a b-value between 0.8 and 1.1. The catalogue of observations below -1.5 (the highest density of observations) is considered incomplete due to the detection threshold of the buried surface array.

Observations above magnitude 2.0 exceed the saturation limit of the geophones and should be considered non-conservative (i.e. the events are likely larger than reported in the microseismic data). This catalogue, with the above limitations, will be used to characterize the subsurface DFN, since the magnitude of the microseismic events can be roughly correlated to the size of discontinuity slip (Brune, 1970).

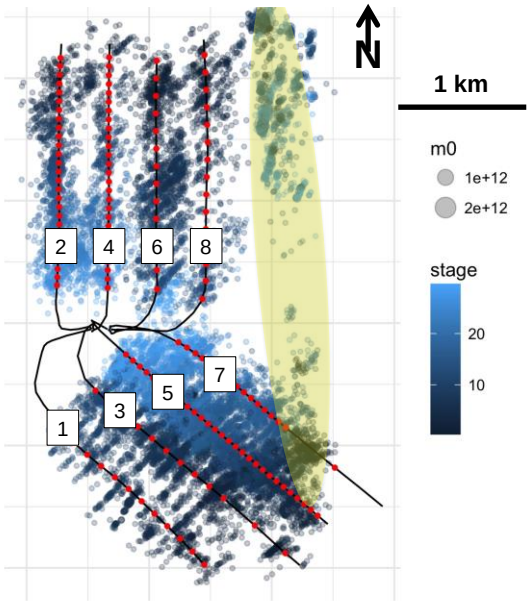


Figure 1. Plan view showing the study's microseismic observations that are coloured by stage and sized by seismic magnitude. Wells are numerated in the white boxes. Black lines illustrate the well trajectories and red dots show the hydraulic fracturing stages. Anomalous triggered events are highlighted in yellow.

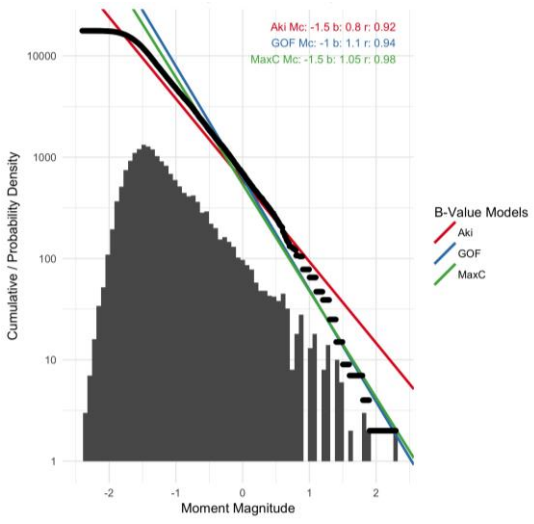


Figure 2. Cumulative magnitude distribution plot of the microseismic data. The probability density is plotted in grey and the cumulative distribution in black.

## 2 METHODS

There are many types of unsupervised learning methods. This study uses portioning, hierarchical, and density-based methods, the most common types of clustering for geospatial data. A dimensionality reduction technique is also used to explore the data. The analysis was conducted using R (Becker at al. 1988) with several external packages that used vectorized operations that are essential to reduce computational time. For example, a non-vectorized distance calculated between the hydraulic fracturing stages and microseismic events (2.7 million operations) took 45 minutes whereas vectorized distance between microseismic events and themselves (31.3 million operations) took 72 seconds.

### 2.1 Pre-Processing and Standardization

The dataset used in this study was already processed to be free of null or non-numeric values. The date and time were formatted into a numeric POSIXct format, required for numerical interpretation in R. The data and operator selected stage clustering are visualized three-dimensionally using Plotly (Sievert et al. 2017).

Standardization is essential for clustering due to the widely differing range and variance of each attribute. Standardization also makes the similarity matrix more consistent across different distance metrics (Kassambara 2017). Each attribute of the data was standardized using a z-score transform, subtracting each attribute value ($x_i$) by its mean ($\hat{x}$) and dividing by its standard deviation ($sd(x)$) (Equation 1).

$$\frac{x_i - \hat{x}}{sd(x)} \qquad [1]$$

Well surveys and hydraulic fracturing intervals were imported with GeoSCOUT™ using publically available data. A non-linear spline function was used to interpolate between survey stations.

### 2.2 Correlation Analysis

A correlation matrix was used to investigate and visualize the dependence between attributes. The Pearson parametric correlation (Pearson 1897) is the most commonly applied criteria in data science, but it requires a standardized and normally distributed dataset. The Spearman non-parametric correlation (Spearman 1904) can be applied to non-normal datasets. A modified Shapiro-Wilk normality test (Shapiro and Wilk 1965, Rahman and Govidarajulu 1997) was used to assess the data for normality and select the appropriate analysis. Razali and Wah (2011) showed that the Shapiro-Wilk analysis is the most powerful normality test. It is very sensitive to outliers and large sample sizes, so a low significance level (0.01) was used to accept or reject the null hypothesis of normality and differentiate between the Pearson and Spearman analysis. A correlation plot was generated using the *corrplot* package after correlation matrix computation (Wei and Simko 2013).

### 2.3 Principal Component Analysis

PCA (Pearson 1901, Hotelling 1933) is a widely used data science technique that produces linear combinations of attributes, analogous to deriving principal stresses in mechanics. These combinations, or principal components (PCs), reorganize the attributes such that each PC is orthogonal to itself. This eliminates cross-correlation between the PCs and allows them to explain a high amount of the data variance with fewer dimensions. In many clustering applications it also separates the data into distinct clusters for easier clustering. PCA is used to reduce the dimensionality of datasets with many attributes and/or identify the most important factors influencing the observations in the data.

This study used PCA through the *prcomp* function to identify covariates (i.e. related attributes) and focus the clustering analysis on influential variables. PCA was also used to identify and remove the influence of attributes that lacked physical significance, such as uncertainty in the microseismic locations. Since the dataset was standardized, these non-physical parameters could easily have as much of an influence on the clustering as location, magnitude, or time. PCA was not used to reduce the dimensionality of the dataset for further clustering analysis at this time, since the physical significance of the parameters was important.

Each PC can be quantified by its eigenvalue or how much variance it captures in the data. The complete variance of the dataset is explained when the number of PCs equals the number of variables. A Scree Plot can visually display this, and it is common to use the principal components that capture large portions of the dataset variance, illustrated by an inflection in the scree plot (Peres-Neto et al. 2005), as illustrated in Figure 3).

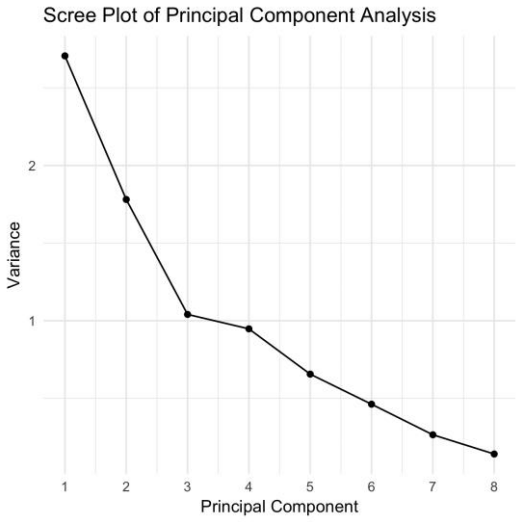


Figure 3. A Scree plot showing the variance captures by each principal component from a PCA of eight attributes. An inflection point is visible at PC3.

PCA results are routinely illustrated using a unit-circle plot, showing the coordinates (or loadings) of each attribute in principal component space (Figure 4). The *ggplot2* package (Wickham 2016) was used to draw the unit circle. The closer an attribute is located towards the unit circle, or the higher its weighting, the more influence it has on the first two principal components, and therefore the data variance.

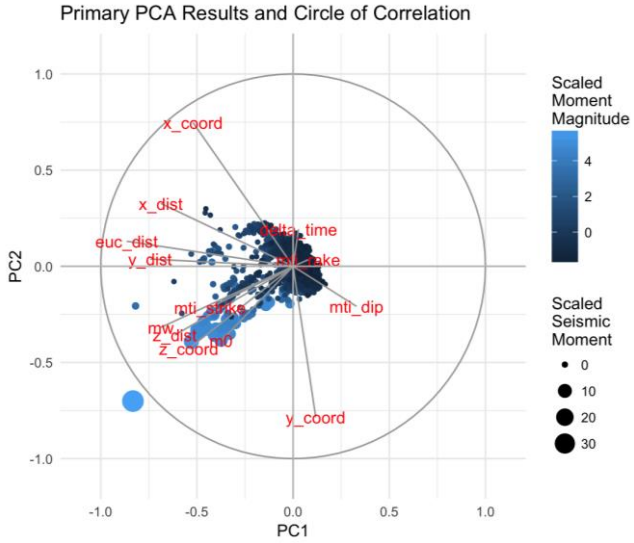


Figure 4. A unit circle plot showing the loadings of each attribute over the first two PCs. Microseismic data is plotted in PC space, differentiated by moment magnitude.

## 2.4 Partitional Clustering using K-Medoids

Partitional Clustering divides observations into non-overlapping clusters. K-means (MacQueen 1987) is a partitioning method routinely used in unsupervised machine learning, however it is sensitive to outliers and non-normality in observations. K-medoids (Kaufman and Rousseeuw 1990), or the most centrally located point in a cluster as measured by the dissimilarity between it and other members of the cluster, was used in this study to increase robustness. Dissimilarity is a key concept to all clustering analysis – it is generally defined as the multi-dimensional distance between all the attributes of one observation and the attributes of every other observation. An abstraction for larger datasets (clara) was used for the analysis. There are several dissimilarity measures, but the Euclidean distance, which is the most common, was used in this study and defined in Equation 2, where $d(a,b)$ is the dissimilarity or distance, $a_i$ and $b_i$ are attributes of observations *a* and *b*, and *n* is the total number of attributes.

$$d(a,b) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2} \qquad [2]$$

The *fpc* package (Hennig 2013) was used for K-Medoids clustering, along with the other clustering methods used in this study. There are numerous indices used to validate the number of clusters used in partitional clustering. For example, *NbClust* (Charrad et al. 2014) uses majority rule from 30 indices to decide for example. In this study, two of the most common validation statistics

were used to determine the optimal number of clusters – the Calinski-Harabasz index (CH index, Calinski and Harabasz 1974) and the average silhouette width (ASW, Rousseeuw 1987).

The CH index (Equation 3) is defined by the between-ground dispersion matrix ($B_q$) for data clustered into q clusters (Equation 4) and the within-group dispersion matrix ($W_q$) for data clustered into q clusters (Equation 5), where $c_k$ is the centroid of cluster $C_k$.

$$\frac{\mathrm{trace}(\mathrm{B_q})(\mathrm{q}-1)}{\mathrm{trace}(\mathrm{W_q})(\mathrm{n}-\mathrm{q})} \quad [3]$$

$$\sum_{\mathrm{k}=1}^{\mathrm{q}} \mathrm{n_k}\,(\mathrm{c_k}-\hat{\mathrm{x}})(\mathrm{c_k}-\hat{\mathrm{x}})^{\mathrm{T}} \quad [4]$$

$$\sum_{\mathrm{k}=1}^{\mathrm{q}} \sum_{\mathrm{i}\epsilon \mathrm{C_k}} (\mathrm{x_i}-\mathrm{c_k})(\mathrm{x_i}-\mathrm{c_k})^{\mathrm{T}} \quad [5]$$

The silhouette index is defined in Equation 6, where $c(i)$ is the average dissimilarity of each object to all of objects its cluster $C_k$ and $d(i)$ is the minimum average dissimilarity of each object to all other clusters (Charrad et al. 2014).

$$\sum_{i=1}^{n} \left[\frac{c(i)-d(i)}{max(c(i),d(i))}\right] \quad [6]$$

The *fpc* algorithm provides a clustering classification by maximizing each index. One more analysis was conducted using the number of fracturing stages in the well as specified by the operator.

## 2.5 Hierarchical Clustering using Ward's Method

This study used the *hclust* function and *fpc* package for agglomerative hierarchical clustering. This technique is computationally and memory intensive relative to partitional clustering. Each observation starts as it's own cluster and pairs of clusters are merged step-wise as the hierarchy tree grows. The nodes with the minimal pairwise distances are merged and the distance recomputed at every hierarchy step. This process repeats until all nodes are merged into a single cluster. There are many agglomeration methods including single-linkage (Florek 1951), complete-linkage (Sorensen 1948), centroid based (Sokal and Michener 1958), and minimum energy based (Ward 1963). Several trials indicated that Ward's method, which is the minimum energy based hierarchical equivalent of k-means, yielded the most distinct clusters relative to those selected by the operators.

As with partitioning based methods, the dissimilarity matrix is key for successful clustering. Additionally, it is important to select appropriate criteria for "cutting" the hierarchical tree – or identifying how many clusters to select. This process is illustrated in Figure 5. The same metrics that were applied to the K-Medoids technique (the CH index and ASW) were applied to cut the hierarchical tree. Unfortunately, these criteria tended to minimize the numbers of clusters regardless of agglomeration method.

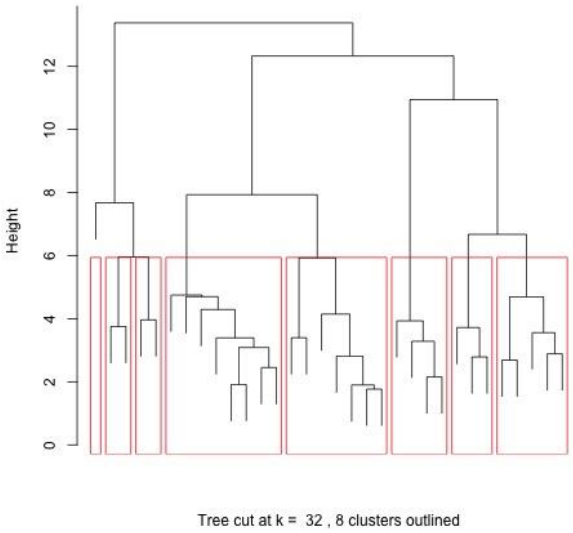


Figure 5. A dendogram showing the hierarchical agglomeration of 32 clusters using Ward's method. Eight clusters are outlined, based on the maximization of the CH index and ASW.

## 2.6 Density Based Spatial Clustering with Noise

Partitioning and hierarchical clustering methods can fail when clusters aren't spherical or convex, and can be severely affected by noise and outliers. Ester at al. (1996) developed an algorithm titled 'density-based spatial clustering with noise' (DBSCAN), which is based on human intuitive clustering where points of high density are visually clustered. It is one of the most common clustering algorithms and can identify arbitrary and linear shapes with a common density. It has a notion of noise and is robust to outlier. DBSCAN requires two parameters – the reachability distance ($\boldsymbol{\varepsilon}$) and the minimum number of points for forming a cluster ($p_{min}$). It is sensitive to differences in density and it can struggle with datasets that display large differences in density, as well as the reachability distance; where the minimum number of points simply serves to designate core clusters for further agglomeration. Two points are considered to be a cluster if the distance between the points is lower or equal to $\boldsymbol{\varepsilon}$. As a general rule, the minimum number of points should be greater than the number of dimensions / attributes in a data set and was set as the number of attributes +1 for this study.

Setting bounds for the reachability distance is the key constraint for implementing DBScan. A plot of k-Nearest Neighbours (kNN) distance versus connections between points provides a measure of the appropriate bounds for $\boldsymbol{\varepsilon}$. The distance from one point towards kNNs ranging from 1 to $p_{min}$ is plotted against the connections, as illustrated in Figure 6. The inflection of the curve serves to distinguish meaningful connections / clustering from noise, with the bounds being carried forward into this study's analysis. Once the $\boldsymbol{\varepsilon}$ value was bounded, a 'hybrid' DBSCAN algorithm was used, which computed partial distance matrices from raw input data to speed computation. The *dbscan* package (Hahsler 2017) was used for this purpose.

The CH index and ASW were computed from the results of the DBSCAN analysis with each incremental $\varepsilon$ value to optimize the results.

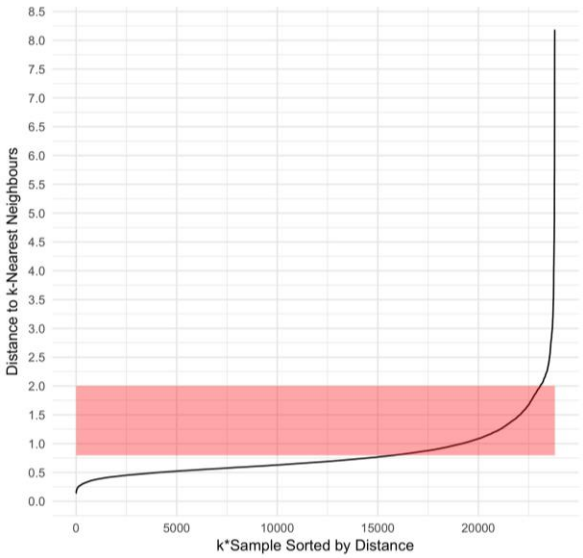


Figure 6. A plot showing the distance to k-nearest neighbours against the k times number of samples, sorted by distance and with k = 9. The $\varepsilon$ values carried forward in the DBSCAN analysis are annotated by the red rectangle.

## 3 RESULTS

The dataset used was complicated by the fact that the fracturing stages occurred nearly simultaneously across numerous wellbores with the exact stage timing unavailable. The nearly simultaneous completions (called a 'zipper frac' in industry) are illustrated in Figure 7. Even with exact start-stop times, it is difficult to cluster the events and attribute them to a single stage based on time alone. A k-nearest neighbour (kNN) algorithm was used to select the ten closest microseismic events to each based on Euclidian distance in order to label these. When compared to operator selected stage classifications, the kNN algorithm correctly identified 93% of stages. The results of the KNN classification are illustrated in Figure 8.

The results were relatively insensitive to the distance metric chosen, but a consistent distance metric with scaled attributes was used to generate consistent results. Due to the poor time control for the stages, it wasn't possible to compute a differential time attribute without excessive correlation and inaccuracy.

The combination of analysis (partitional, hierarchical, or density based) and validation statistic (CH index, ASW, etc.) resulted in widely varying results. Validation statistics are used for hypertuning in most of these methods, where the number of clusters is varied to maximize the chosen validation statistic. The results of this study, an example of which is discussed below, showed that a consistent method is essential for using clustering to generate DFNs.

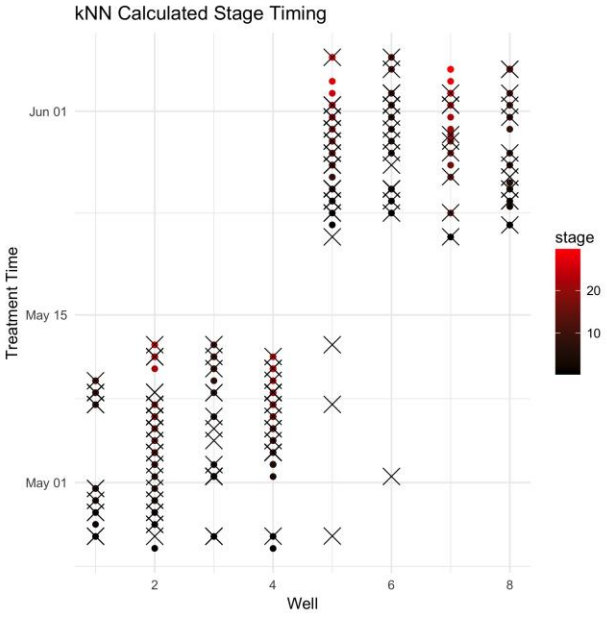


Figure 7. Plot of treatment time versus well with marker. Coloured dots indicate operator stage classification. The kNN classifier results are indicated by "X".

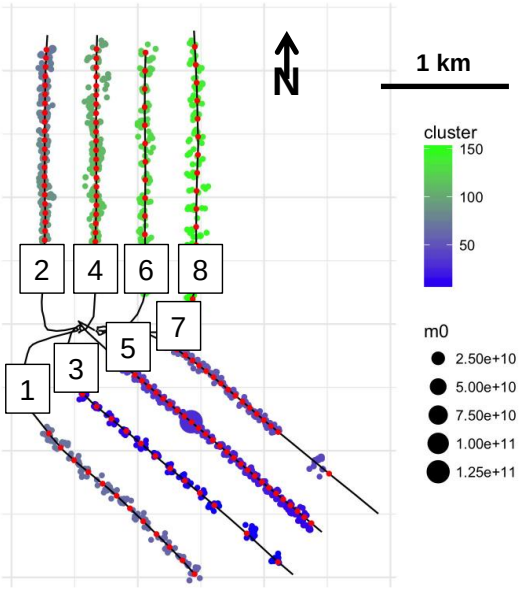


Figure 8. Results of automated kNN clustering analysis showing the 10 closest neighbours to the hydraulic fracturing time based on Euclidean distance.

### 3.1 Single Well, Single Stage Example

The example that follows focuses on one stage from a single well. The kNN classification scheme yielded 369 events for this stage, which are illustrated in Figure 9.

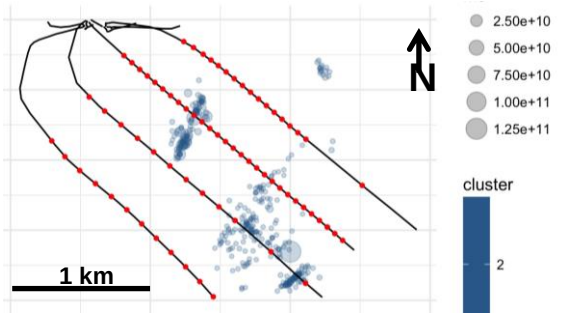


Figure 9. Single well, single stage example. Points are scaled by seismic moment, with a 50% transparency to illustrate dense overlays of events.

The results of several trials indicated that lower dimensionality datasets produced more consistent and reliable results. For example, the average silhouette width, Dunn's index, and CH index were highest for points clustered based on the time difference (t_dist) alone. This makes sense, since this is the main method used by operators to manually cluster into stages. This method may be adequate for separating data into stages. However, it doesn't help differentiate closely spaced clusters and a more robust set of attributes is required for using clustering to generate DFNs. Figure 10 illustrates this, by showing the average silhouette width for multiple data subsets with various attributes.

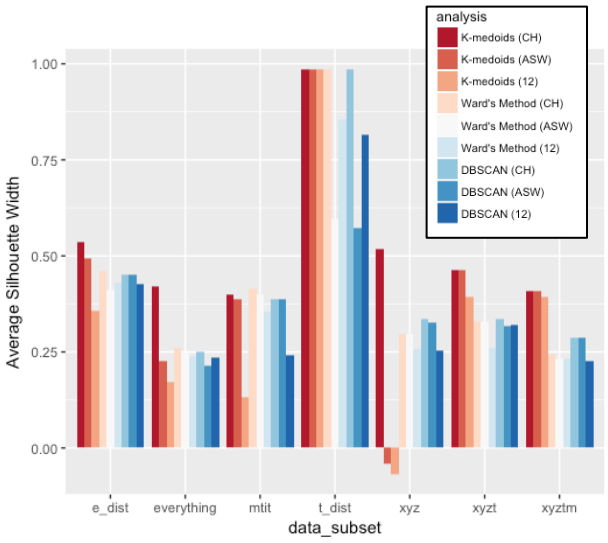


Figure 10. The average silhouette width for various methods (denoted by bar colour). Data subsets (horizontal axis). included euclidean distance (e_dist), all attributes (everything), moment tensor information (mtit), t_dist, coordinates (xyz), coordinates and time (xyzt), coordinates, time, and moment magnitude (xyztm).

The PCA unit circle for the example (Figure 11) shows that distinct groups of data do exist and that the magnitude (mw) is correlated with the moment tensor strike (mti_strike). It also shows that it should be possible to seperate clusters using coordinates, time, and magnitude as attributes, since they account for a large portion of the variance in the first two principal components. These principal components account for 56% of the dataset's variance.

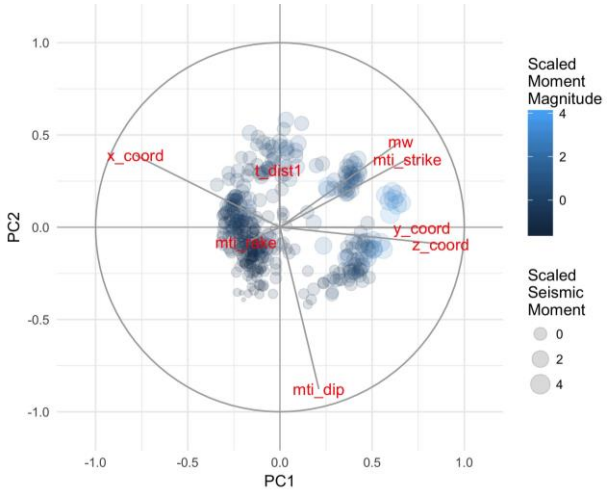


Figure 11. Principal component analysis of the example stage data. Red text denotes attributes with coordinates indicating principal component weightings and proximity to unit circle indicating influence on parameter variance.

The optimal method varied depending on what validation statistic was used. The silhouette index, within-cluster distance, and between-cluster distance indices were all highest for the DBSCAN method with 6 clusters. The Dunn index was highest for Ward's Hierarchical clustering method with 13 clusters. The CH Index indicated that the K-medoids method with 4 clusters was optimal. Once again, a lower dimensional dataset with only x, y, and z coordinates and time attributes yielded the best results. This is illustrated by comparing the results from a K-medoids analysis with a nine attribute data (Figure 12) to a K-medoids analysis with only four attributes (Figure 13). In each of the figures, different colours illustrate the clustering results. The overlap of different colours show that the large dimensional dataset failed to group lineations and discrete clusters in the data whereas the lower dimensional dataset did a much better job of geospatially grouping points. This difference is far more significant than the inter-method differences, although these too can be significant. The results of the hypertuned hierarchical clustering (Figure 14) and DBSCAN (Figure 15) results from the four attribute data set show how the method differences affect the clustering results. There are still overlaps between groups; however, the methods bear close resemblance to each other.

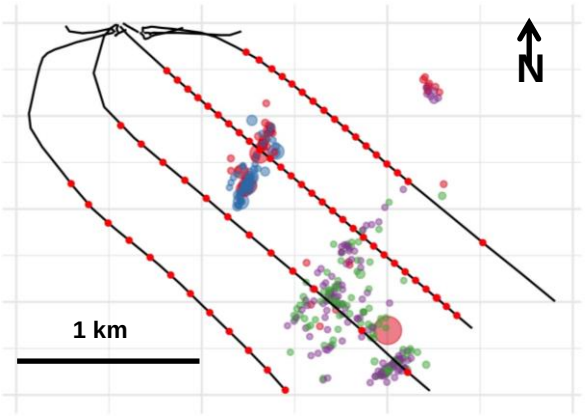


Figure 12 Clustering results (differentiated by colours) from the K-medoids analysis using nine attributes (x,y, and z coordinates; moment tensor inversion strike, dip, and rake; time, and magnitude)

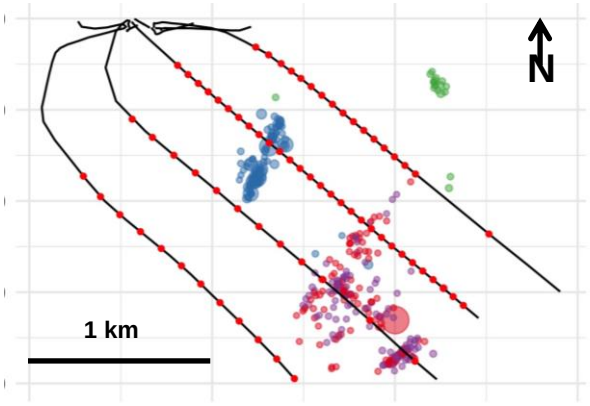


Figure 13. Clustering results (differentiated by colours) from a K-medoids analysis using four attributes (x,y, and z coordinates; and time)

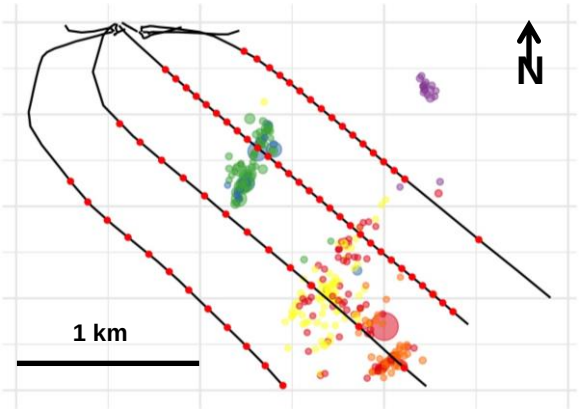


Figure 14. Example clustering results (differentiated by colours) from a hierarchical clustering analysis using Ward's method with four attributes (x,y, and z coordinates; and time)

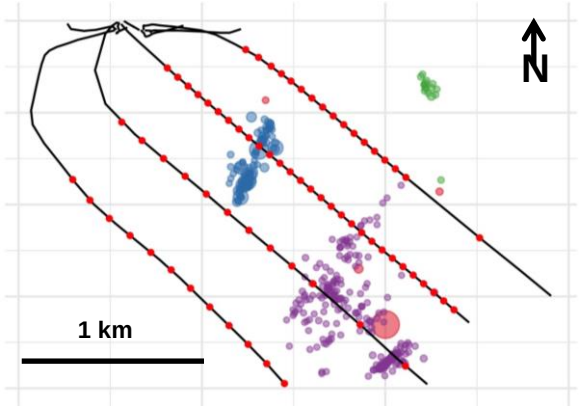


Figure 15. Example clustering results (differentiated by colours) from a DBSCAN analysis with four attributes (x,y, and z coordinates; and time)

Ultimately, these methods will need to applied automatically to large datasets of microseismic events and their success will depend on computational efficiency, robustness to varying microseismic clouds, and ease of applicability. In a trial of 2644 events with 24 attributes, the partitional clustering took 51 seconds, the hierarchical clustering took 251 seconds, and the DBSCAN took 183 seconds. The hypertuning tended to generate more clusters with K-medoids, which may limit its applicability to finding linear features. The hierarchical clustering gave excellent results but was computationally intensive. The DBScan was able to pick up noise/outliers, but required a manual step for each stage (selecting the eps value).

## 4 CONCLUSIONS AND FUTURE WORK

This study applied several unsupervised learning techniques to cluster microseismic events. It presented several validation statistics based on cluster betweenness and withinness and used these statistics to hypertune the unsupervised methods. Principal component analysis and correlation analysis was used for exploratory data analysis and to select an appropriate number of attributes for the analysis. Lower dimensionality datasets tended to yield better results in terms of validation statistics, but miss the purpose of the clustering (picking out lineation and dense clouds). When a low dimensional dataset with adequate features was used, the methods tended to perform adequately; however, the hierarchical and DBScan methods seem to generate the most reasonable results.

The ultimate objective of this study is to generate a discrete fracture network from a dense and irregular dataset with tens of thousands of observations. The results indicated that it is indeed possible to identify discrete clusters in the data, but that the methods are very sensitive to the dataset dimensionality (i.e. number of attributes used) and validation statistics.

Moving forward, the hierarchical clustering algorithm with hypertuning will be applied to the entire dataset using an automated algorithm. The hierarchical method is computationally demanding; however, it can be fully

automated. It can also be linked with Bayesian inferential methods (Heller and Ghahramani, 2005) and makes more physical sense in terms of seismicity generation, with several events linking together to form a large fault slip. This study indicated that the difference in methods were less significant than the difference in attributes and validation statistics used to hypertune the methods. Future work will look at optimizing the validation statistics and applying various other hierarchical methods such as graph theory, Bayesian methods, and structural methods linked the earthquake rupture physics.

## 5 ACKNOWLEDGMENTS

The authors would like to thank the Microseismic Industry Consortium (MIC), NSERC and the Canada First Research Excellence Fund (CFREF) for funding this work. Our sincere thanks go out to Chevron for providing the excellent pre-processed microseismic dataset.

## 6 REFERENCES

Becker, R.A., Chambers, J.M. & Wilks, A.R., 1988. The new S language. Pacific Grove, Ca.: Wadsworth \& Brooks, 1988.

Berkowitz, B., 2002. Characterizing flow and transport in fractured geological media : A review. , 25, pp.861–884.

Brune, J.N., 1970. Tectonic stress and the spectra of seismic shear waves from earthquakes. Journal of geophysical research, 75(26), pp.4997–5009.

Caliński, T. & Harabasz, J., 1974. A dendrite method for cluster analysis. Communications in Statistics-theory and Methods, 3(1), pp.1–27.

Charrad, M. et al., 2014. NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set. Journal of Statistical Software, 61(6). Available at: http://www.jstatsoft.org/v61/i06/.

Ester, M. et al., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd. pp. 226–231.

Florek, K. et al., 1951. Sur la liaison et la division des points d'un ensemble fini. In Colloquium Mathematicae. pp. 282–285.

Hahsler, M. & Piekenbrock, M., 2017. dbscan: Density Based Clustering of Applications with Noise (DBSCAN) and Related Algorithms. R package version, pp.0–1.

Heller, K.A. & Ghahramani, Z., 2005. Bayesian Hierarchical Clustering, London, UK.

Hennig, C., 2013. fpc: Flexible procedures for clustering. R package version 2.1-5.

Hotelling, H., 1933. Analysis of a complex of statistical variables into principal components. Journal of educational psychology, 24(6), p.417.

Hudyma, M.R., 2008. Analysis and Interpretation of Clusters of Seismic Events in Mines. University of Western Australia.

Kassambara, A., 2017. Practical Guide to Cluster Analysis in R: Unsupervised Machine Learning, STHDA.

Kaufman, L. & Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis, John Wiley \& Sons.

MacQueen, J., 1967. Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. pp. 281–297.

Pearson, K., 1897. Mathematical contributions to the theory of evolution - on a form of spurious correlation which may arise when indices are used in the measurement of organs. Proceedings of the royal society of london, 60(359–367), pp.489–498.

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.

Peres-Neto, P.R., Jackson, D.A. & Somers, K.M., 2005. How many principal components? Stopping rules for determining the number of non-trivial axes revisited. Computational Statistics \& Data Analysis, 49(4), pp.974–997.

Rahman, M.M. & Govindarajulu, Z., 1997. A modification of the test of Shapiro and Wilk for normality. Journal of Applied Statistics, 24(2), pp.219–236.

Razali, N.M., Wah, Y.B. & others, 2011. Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. Journal of statistical modeling and analytics, 2(1), pp.21–33.

Rousseeuw, P.J., 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics, 20, pp.53–65.

Shapiro, S.S. & Wilk, M.B., 1965. An analysis of variance test for normality (complete samples). Biometrika, 52(3/4), pp.591–611.

Sievert, C. et al., 2016. plotly: Create Interactive Web Graphics via plotly.js. R package version, 3(0).

Sokal, R.R., 1958. A statistical method for evaluating systematic relationship. University of Kansas science bulletin, 28, pp.1409–1438.

Sørensen, T., 1948. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. Biol. Skr., 5, pp.1–34.

Spearman, C., 1904. The Proof and Measurement of Association between Two Things. The American Journal of Psychology, 15(1), pp.72–101.

Tezuka, K. & Niitsuma, H., 2000. Stress estimated using microseismic clusters and its relationship to the fracture system of the Hijiori hot dry rock reservoir. Engineering Geology, 56(1–2), pp.47–62.

The Royal Society, 2017. Machine learning: the power and promise of computers that learn by example, London, UK.

Ward Jr, J.H., 1963. Hierarchical grouping to optimize an objective function. Journal of the American statistical association, 58(301), pp.236–244.

Wei, T. & Simko, V., 2013. corrplot: Visualization of a correlation matrix. R package version 0.73, 230(231), p.11.

Wickham, H., 2016. ggplot2: elegant graphics for data analysis, Springer.