THESIS

MACHINE LEARNING PREDICTION OF DEEPWATER SLOPE-CHANNEL FACIES USING CORE-ANALOGOUS OUTCROP OBSERVATIONS

Submitted by

Patrick Ronnau

Department of Geosciences

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Fall 2024

Master's Committee:

Advisor: Lisa Stright

Michael Ronayne
Sean Gallen
Nikhil Krishnaswamy

ABSTRACT

MACHINE LEARNING PREDICTION OF DEEPWATER SLOPE-CHANNEL FACIES USING CORE-ANALOGOUS OUTCROP OBSERVATIONS

Sedimentological (SED) data is often qualitative, making combining it with Machine Learning (ML) workflows challenging. SED data in subsurface exploration incorporates qualitative interpretations that remain valuable to subsequent exploration efforts. These exploration projects often have access to geologic core data that is limited spatially, making subsurface interpretation difficult and highly uncertain. Incorporating core-like data into ML workflows provides a framework to generate consistent interpretation over large datasets. ML, a technique already employed in well-log interpretation, represents an advantage over manual interpretation methods which are time intensive and introduce errors and bias. This research investigates methods to automate geologic interpretation (specifically of sedimentary facies) through ML techniques. Sedimentological observations (grain size, bed thickness) from outcrop measured sections in the deepwater slope strata of the Magallanes Basin provide training and testing features to make ML predictions (classifications) of human-interpreted geologic facies.

The study employs seven ML techniques (K-Means, Least Squares Regression, Logistic Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Random Forest, and Neural Networks) to investigate the problem of facies prediction from multiple methodological angles. The results show that some ML methods are not suitable for this classification problem due to their architecture or the qualitative aspects of manually collected SED data. Supervised methods generally provide better results than unsupervised methods (PCA and K-Means). Supervised ML both produces better raw performance metrics (Accuracy, BedThickness Normalized, Accuracy, Recall) than K-Means, and generates qualitatively better predictions of measured sections (Fig. 48; Fig. 49). Among methods that are suitable, a random forest model generates the best facies prediction performance.

ACKNOWLEDGEMENTS

This study is an extension of the comprehensive sedimentological research conducted by the Chile Slope Systems (CSS) Joint Industry Project, a collaboration involving Colorado State University, Virginia Tech University, and the University of Calgary along with various industry sponsors including Anadarko, BHP Billiton, Chevron, ConocoPhillips, Equinor, Hess, Nexen/CNOOC, Petrobras, Repsol, and Shell. The fieldwork, data collection, and interpretation of the Laguna Figueroa outcrop, which served as this study's foundation, were carried out by Brian Romans, Steve Hubbard, Ryan Macauley, Sean Fletcher, and Sarah Southern.

TABLE OF CONTENTS

ABSTRACT ii
ACKNOWLEDGEMENTS iii
LIST OF TABLES vi
LIST OF FIGURES vii
LIST OF EQUATIONS x
1. INTRODUCTION 1
1.1 Overview of SED database at Laguna Figueroa 4
1.2 Research Motivation 6
1.3 Machine Learning in Geoscience 7
2. GEOLOGIC DATABASE DESCRIPTON 9
2.1 Geologic Setting 9
2.1.1 Deepwater Stratigraphy 9
2.1.2 Magallanes Basin 11
2.1.3 Measured Sections and Sedimentary Facies 16
2.2 Database Description 20
2.2.1 Database Class Statistics 21
3. MACHINE LEARNING METHODS 25
3.1 Unsupervised Methods 25
3.1.1 Principal Component Analysis 26
3.1.2 K-Means Clustering 26
3.2 Supervised Methods 27
3.2.1 Least Squares Solution 28
3.2.2 Logistic Regression 29
3.2.3 Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) 30
3.2.4 Random Forest 31
3.2.5 Neural Network Classifier 33
3.3 Performance Metrics 35
3.3.1 Accuracy 35
3.3.2 Recall 35
3.3.3 Precision 36
3.3.4 Inertia 36
3.3.5 Silhouette Score 37
3.3.6 Davies-Bouldin Index 38
3.3.7 Calinski-Harabasz Score 38
3.3.8 Feature Importance 38
3.3.9 Confusion Matrix 39
4. RESULTS AND PREDICTION EVALUATION 41
4.1 Unsupervised Methods 41
4.1.1 K-Means Clustering 42
4.2 Supervised Methods 47
4.2.1 Least Squares Solution 47
4.2.2 Logistic Regression 54
4.2.3 Linear Discriminant Analysis (LDA) 60
4.2.4 Quadratic Discriminant Analysis (QDA) 64
4.2.5 Random Forest 68
4.2.6 Neural Network Classifier 74

5. DISCUSSION AND INTERPRETATION 80
5.1 Unsupervised Methods 80
5.1.1 K-Means Clustering 80
5.2 Supervised Methods 82
5.2.1 Least Squares Solution 83
5.2.2 Logistic Regression 85
5.2.3 Linear Discriminant Analysis (LDA) 86
5.2.4 Quadratic Discriminant Analysis (QDA) 87
5.2.5 Random Forest 88
5.2.6 Neural Network Classifier 90
5.3 Comparative Results of All Methods 91
5.3.1 Quantitative Comparison 91
5.3.2 Visual Results in Training and Testing Area 94
5.3.3 Visual Results in Prediction Area 97
6. CONCLUSIONS AND FUTURE WORK 99
6.1 Conclusions 99
6.2 Future Work 100
7. REFERENCES 101

LIST OF TABLES

Table 1: Database Features Utilized in this Study 17

Table 2: Total Database Facies Count 21

Table 3: Logistic Regression Hyperparameters 30

Table 4: Random Forest Hyperparameters 33

Table 5: Neural Network Hyperparameters 34

Table 6: Comparison of ML Method Performance Metrics 93

Table 7: Per Facies Performance Metrics for Logistic Regression Model 93

Table 8: Per Facies Performance Metrics for Random Forest Model 93

Table 9: Per Facies Performance Metrics for Neural Network Model 93

LIST OF FIGURES

Figure 1: Deepwater channel systems (A from Hubbard et al., 2014), (C from Daniels et al., 2024) 3

Figure 2: Section data collected at Laguna Figueroa. 6

Figure 3: The Bouma sequence, modified slightly from Bouma, 1962 (Strekeisen, A., 2020) 10

Figure 4: Sedimentary facies model of the turbidite fan (Sprague et al., 2005). 11

Figure 5: Location and strata of the study area examined in this thesis (Modified from Romans, 2011). . 12

Figure 6: Schematic diagrams depicting style of deepwater deposition (Romans et al., 2011). 15

Figure 7: Bed-scale target facies of the Tres Pasos Formation. Adapted from (Merovitz, 2024). 18

Figure 8: Example measured section data from Fig 100 section in the Laguna Figueroa. 19

Figure 9: Total Database Distribution Between Target Facies (Classes) 22

Figure 10: Total Database Feature Distribution, Differentiated by Interpreted FaciesCode 24

Figure 11: Example decision tree. Modified from (Biau, 2012). 33

Figure 12: Schematic diagram connecting neurons. Modified from (Robinson, 2023). 34

Figure 13: Typica confusion matrix configuration (Evidently AI, 2024). 39

Figure 14: Explained Variance by Number of Components in PCA. 41

Figure 15: Evaluation of Cluster Quality Using Inertia and Silhouette Score. 42

Figure 16: Comparison of Davies-Bouldin and Calinski-Harabasz Scores for Clustering Evaluation. 43

Figure 17: Scatter Plot; BedThickness vs. P50 Grain Size. 44

Figure 18: Scatter Plot; Grian Size Max vs. Min. 44

Figure 19: Comparison of Interpreted Class and K-Means, Differentiated by FaciesCode 46

Figure 20: Normalized Confusion Matrix for *highamalsand* vs. All Other Classes. 48

Figure 21: Accuracy Variation on BedThickness Feature Normalization; *highamalsand.* 49

Figure 22: Normalized Confusion Matrix for *semiamal* vs. All Other Classes. 50

Figure 23: Accuracy Variation on BedThickness Feature Normalization; *semiamal.* 50

Figure 24: Normalized Confusion Matrix for *nonamalsand* vs. All Other Classes. 51

Figure 25: Accuracy Variation on BedThickness Feature Normalization; *nonamalsand.* 52

Figure 26: Normalized Confusion Matrix for *nonamalsilt* vs. All Other Classes 53

Figure 27: Accuracy Variation on BedThickness Feature Normalization; *nonamalsilt.* 54

Figure 28: Logistic Regression Feature Coefficients per FaciesCode. 56

Figure 29: Normalized Confusion Matrix for Logistic Regression Model. 57

Figure 30: BedThickness Normalized Accuracy for Logistic Regression Model. 58

Figure 31: Comparison of Interpreted Class and Logistic Regression, Differentiated by FaciesCode 59

Figure 32: Normalized Confusion Matrix for Linear Discriminant Analysis Model. 61

Figure 33: BedThickness Normalized Accuracy for Linear Discriminant Analysis Model. 62

Figure 34: Comparison of Interpreted Class and Linear Discriminant Analysis. 63

Figure 35: Normalized Confusion Matrix for Quadratic Discriminant Analysis Model. 65

Figure 36: BedThickness Normalized Accuracy for Quadratic Discriminant Analysis Model. 66

Figure 37: Comparison of Interpreted Class and Quadratic Discriminant Analysis 67

Figure 38: Random Forest Accuracy for Varying Number of Trees in the Model. 69

Figure 39: Random Forest Feature Importance. 69

Figure 40: Normalized Confusion Matrix for Random Forest Model. 71

Figure 41: BedThickness Normalized Accuracy for 10 Random Forest Models. 72

Figure 42: Comparison of Interpreted Class and Random Forest, Differentiated by FaciesCode. 73

Figure 43: Normalized Confusion Matrix for Neural Network (5 Model) Classification. 75

Figure 44: BedThickness Normalized Accuracy for 5 Neural Network Models. 76

Figure 45: Comparison of Interpreted Class and Neural Network, Differentiated by FaciesCode 77

Figure 46: Normalized confusion matrix for Multi-Layer Perceptron method. 79

Figure 47: Accuracy Comparison by Method; Total Accuracy and BedThickness Normalized 92

Figure 48: CACH1 -- Drafted measured section, interpreted facies, and predictions. 95

Figure 49: FIG100 -- Drafted measured section, interpreted facies, and predictions. 96

Figure 50: Random Forest prediction in A) PCS19, and B) DCMS25. 98

LIST OF EQUATIONS

Equation 1: Data points for observations and centroids are in the form *d(x, y, z)*. 26

Equation 2: The sum of all distances in a class, divided by the number of observations.. 27

Equation 3: Sigmoid (Logistic) function (Scott Long, 1997). 29

Equation 4: Linear function with "*t*" class relationships (Scott Long, 1997). 29

Equation 5: Softmax function (Bishop, 2006). 29

Equation 6: Facies classes can be represented the vector "*c*" in ten dimensions (10-D). 31

Equation 7: Equation for classification accuracy. 35

Equation 8: Equation for classification recall (Hall, 2016). 36

Equation 9: Equation for classification precision (Hall, 2016). 36

Equation 10: General Form of Inertia "*I*" Equation (sum of squared minimum difference). 36

Equation 11: General Form of Equation for Silhouette Score (Rousseeuw, 1987). 37

1. INTRODUCTION

Sedimentary processes break down rocks at the Earth's surface, transporting sediments along gravitational gradients into basins. These sediments can lithify into strata that preserve information about the environment and the basin's history. In deepwater settings (often oceans), when a basin develops enough relief to initiate density currents, sediments are distributed across the basin floor in complex sub-sea systems (McHargue et al., 2011). These processes result in sedimentary layers that, despite accumulating slowly, compress millions of years into relatively thin sections of rock, which can be studied today (Lee, 1952). Spatial variations in deposition within the system give rise to facies, or distinct rock units reflecting environmental and process conditions through characteristics such as grain size, sedimentary structures, and fossil content. Predicting sedimentary facies in deepwater environments is thus crucial for reconstructing past depositional patterns (geomodelling) and their governing processes. By understanding how basin geometry, sediment supply, and deepwater dynamics interact, geologists can better interpret the subsurface, aiding in resource exploration and management.

Deepwater slope-channel systems, more specifically, transport sediments from marine-continental shelf environments to the basin floor, scouring and filling channel elements (Fig. 1) through transport, sorting, and deposition of successive flows (McHargue et al., 2011). These sandstone- and siltstone-dominated channel systems exhibit bed- to geobody-scale rock property distributions (facies, porosity, and permeability) that have a high abundance of thick-bedded sandstone toward the center of the channel element to thin-interbedded sandstone and siltstone toward the edges or margins (Fig. 1A). These data are measured by a field geologist in a vertical transect, called a measured section (Fig. 1B). The larger scale architecture of the deepwater channel system is comprised of stacked and grouped channel elements (Fig. 1C; Sprague et al., 2005; Mayall et al., 2006; Hubbard et al., 2014; Meirovitz et al., 2020). The stacking patterns of channel elements and internal architecture directly affect the flow of fluids when these channelized systems are in the subsurface and contain fluids (oil/gas, water, carbon dioxide).

While these architectures and patterns can be readily observed and interpreted in outcropping rock formations (e.g., Fig. 1A), it is much more difficult to interpret in the subsurface (typically > 5000 m depth, with limited whole core samples) with sparse data or low-resolution data, which leads to ambiguity in the interpretation of the system organization and property distributions. Ambiguity directly impacts the economic risk, often leading to low success rates in exploration projects. Many other studies have focused on the larger scale architecture to understand how the ambiguity is related to risk associated with the volumes of fluid present, as well as the ability for the fluid to be recovered to the surface (Ruetten, 2021; Escobar Arenas, 2023). The aim of this research is to step down in scale, and focus on the classification of grouped rock properties, herein referred to as "facies," from 1D data (outcrop measured sections here as an analog to wellbore core from the subsurface). These are a foundational building block to predict reservoir sandstone distribution in the subsurface and reduce interpretation ambiguity within the framework of deepwater slope-channel systems utilizing machine learning methods.

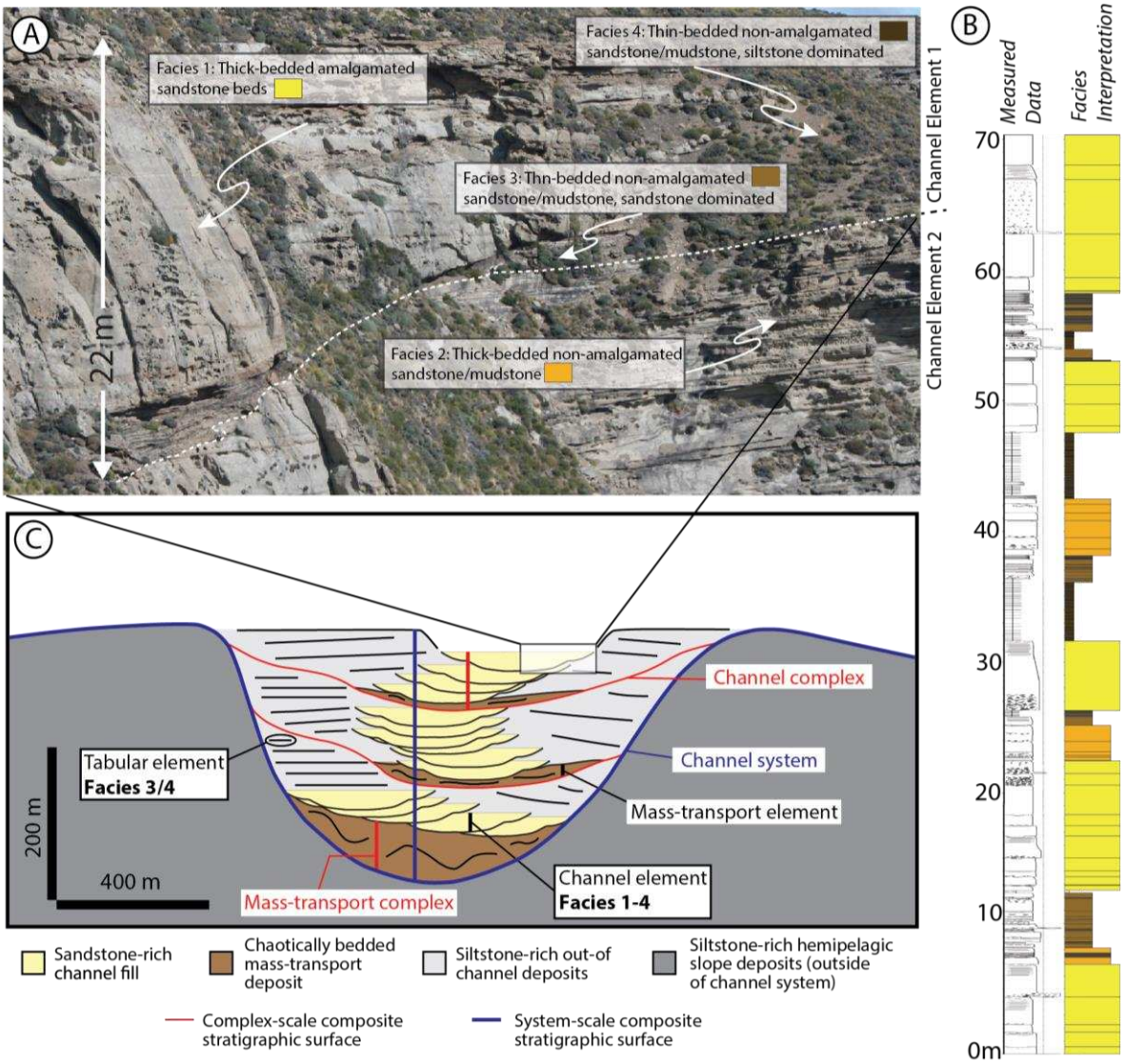

Figure 1: Deepwater channel systems are comprised of a series of interoperative building blocks including sandstone/mudstone beds of different thickness distributions (A, modified after Hubbard et al., 2014), that field geologists measure in vertical sections and interpret grouped beds into categories called facies (A, B). The facies are organized into hierarchical scales (C, modified after Daniels et al., 2024). The smallest scale is comprised of beds organized related to depositional energy – high energy: channel element axis of center (A). These beds can be grouped into facies – beds of similar thickness, amalgamation, and grain size – and are fundamental building blocks for elements that are the core of channel complexes and systems.

Traditional geologic interpretation is time consuming and laborious. Modern geologic exploration methods generate large databases of information on earth's subsurface while only capturing heterogeneity in coarse detail (hundreds of meters). Inherent variability in geologic properties is often described as a

change in sedimentary facies. Quantitative methods like ML can assess the vast quantities of physical data collected as a part of exploration practices rapidly, without introducing bias to interpretation. These approaches are robust even as data varies in quantity, precision, and spatial density. ML methods represent a relative reduction in labor and resources compared to manual interpretation, given the training database is adequate and the ML problem is well posed.

The objectives of this research are threefold:

1) To evaluate the quantitative performance of ML workflows for facies prediction. *"How does the computer perform?"*

2) To evaluate the qualitative performance of ML workflows for predicting facies from fundamental sedimentary observations. *"Does it match what a sedimentologist would interpret?"*

3) To evaluate the Chile Slope Systems measured section database (and similar core-analogous data) as training data to extend existing facies interpretation to additional up- and down-dip, measured outcrop sections. *"Can ML be used to interpret our missing classifications in nearby areas?"*

This thesis is structured into six chapters to address the research objectives. Chapter 1 introduces the scope of this analysis including the facies concept and stratigraphic architecture. Chapter 2 provides a detailed geological context of the dataset, focusing on the architectural framework interpretation at the Laguna Figueroa outcrop. Chapter 3 discusses the Machine Learning (ML) algorithms, methods used in the analyses, and the metrics for evaluating their performance. Chapter 4 presents facies predictions and performance metrics for each method. Chapter 5 compares the metrics and the relative quality of predictions in the framework of sedimentary facies prediction in measured sections. Finally, Chapter 6 concludes the thesis, synthesizing research findings and suggesting potential future research directions.

1.1 Overview of SED database at Laguna Figueroa

The outcrop exposures of deepwater architecture in the Cretaceous Tres Pasos Formation at Laguna Figueroa in Southern Chile offer unparalleled visibility of channelized turbidite deposits (Macauley and Hubbard, 2013; Southern et al., 2017; Pemberton et al., 2018; Jackson et al., 2019). This exposure is

recorded as measured stratigraphic sections. Data in the CSS database is manually collected at the outcrop in the form of measured sections (MS) recorded vertically through stratigraphy. Grain sizes are assessed by individuals using hand lenses and grain size index cards. This imparts unique characteristics to the dataset that most closely resemble vertical, 1-D data similar to core samples.

Measured sections from this location include sedimentary facies interpreted manually and represent the foundation of training and testing ML algorithms for this study. The measured data (grain size profiles and bed thickness; Fig. 1B) is recorded in the Chile Slope Systems (CSS) measured section database. This database contains over 10,000 meters of measured sections from deepwater slope channel strata. Sedimentary facies, or groupings of geologic strata with similar grain sizes and bed thickness patterns, are interpreted for 3,400 meters of section collected at the Laguna Figueroa outcrop (Fig 2). Beyond Laguna Figueroa, the broader CSS database (>100 additional measured sections) contains sections without manually interpreted facies. These manual interpretations are time and labor intensive, providing an opportunity for ML methods to optimize the interpretation workflow.

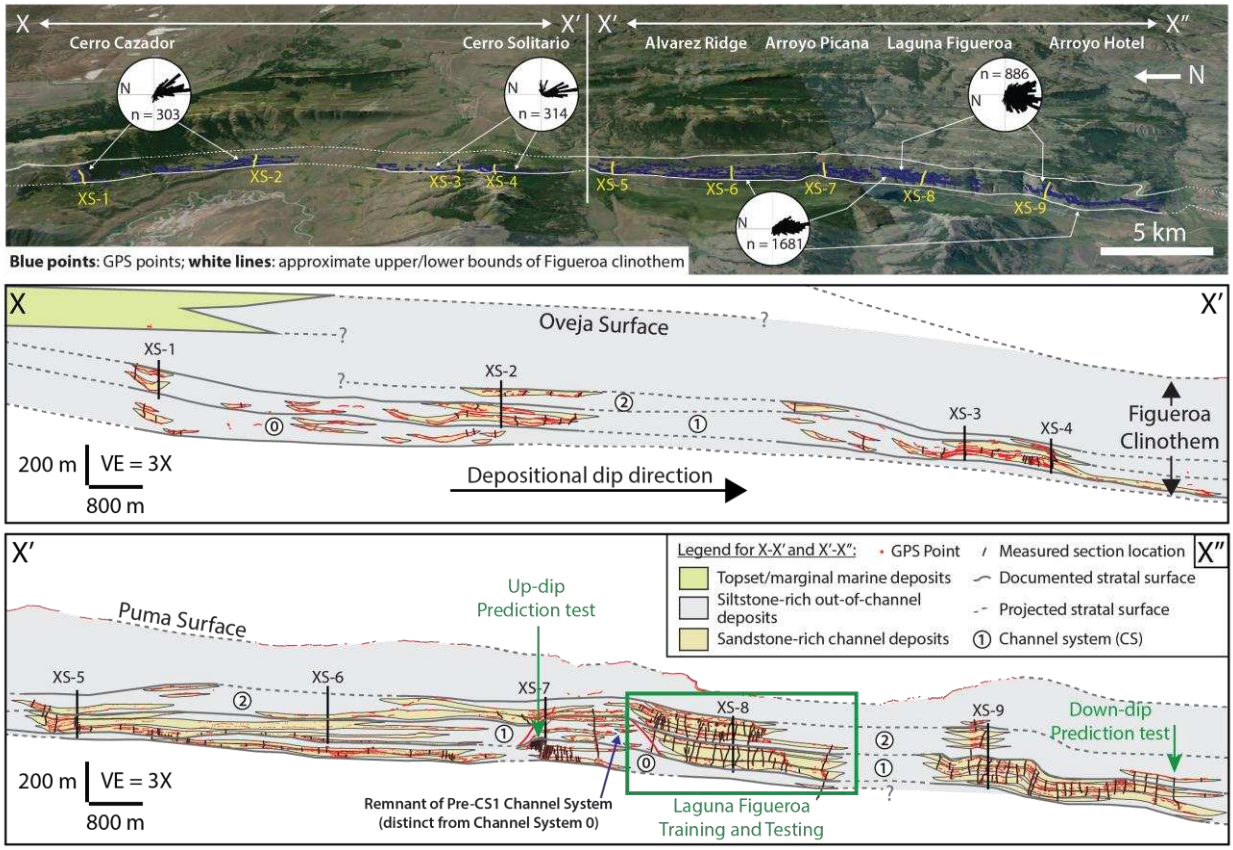

Figure 2: Section data collected at Laguna Figueroa include statistics related to sedimentary grain size and bed thickness, as well as interpretations of facies and depositional architecture. Facies are interpreted at the bed scale and serve as the training data and prediction target value. Additional computed statistics include bulk sandstone proportion, and bed to bed amalgamation ratio. This semi-qualitative, interpreted dataset, is a novel training data type for ML methods, but simultaneously represents a substantial portion of data available to modern geoscientists. Using this dataset, seven different supervised and un-supervised ML methods predict the facies of each bed sample across the entire CSS database. The resulting performance metrics and qualitative prediction performance are then assessed to recommend the best facies-prediction method.

1.2 Research Motivation

Research into deepwater slope-channel systems is partly driven by the petroleum industry, which is actively engaged in enhancing understanding of stratigraphic architecture and its impact on risk during exploration, development, and production (Deptuck et al., 2007). Predictions of facies in deepwater channel systems are critical to understanding channel architecture and its impact on reservoir performance (Romans, 2011). Industry analytical methods commonly include direct reservoir-rock observations through the collection of cores. Core samples subsequently undergo analysis to generate datasets analogous to the grain

size statistics and facies interpretation available at the Laguna Figeroa outcrop. These core-type datasets represent a significant quantity of geologic data that is challenging to incorporate into modern geologic exploration workflows.

In the same way that industry collects core samples from wells, outcrop studies serve as analogs to the subsurface, contributing a wealth of bed- to field-scale observations to our understanding of facies distribution and heterogeneity (Macauley and Hubbard, 2013). This research presents a workflow for the incorporation of core-type data by expanding the interpretation present at Laguna Figeroa to other up- and down-dip measured sections in the CSS database (Fig. 2). Abundant quantitative data and statistics can be derived from outcrop, making this type of ML prediction workflow possible (Southern et al., 2017; Daniels et al., 2019).

1.3 Machine Learning in Geoscience

Developments in ML geologic prediction have recently focused on the application of deep learning methods in lieu of traditional ML techniques. Deep learning incorporates both supervised and unsupervised methods, with a primary emphasis on Neural Networks. These are algorithms designed to mimic the human brain, as initially proposed by McCulloch and Pitts (1943). Utilization of Neural Networks has revolutionized computationally challenging problems like the interpretation of geologic heterogeneity, as demonstrated by Cheng et al. (2019), Mohammadpoor and Torabi (2019), and Na and Fox (2019). Notably, in specific instances, these networks have achieved accuracies exceeding 83%, particularly when incorporating petrographic and textural information into well-log analysis for facies prediction (Saporetti et al., 2018).

Despite diverse applications in the petroleum industry, the utilization of ML algorithms on field observations, such as measured section, remains infrequent. Limited adoption of this data type for quantitative ML analysis can be attributed, in part, to the semi-qualitative nature of sedimentologic interpretation. Integrating field-collected geologic data and ML methods, thus, still holds potential to

enhance prediction of facies. Previous CSS research has applied this workflow in predicting relative channel position of a group of beds denoting an architectural position within a channel (axis vs. margin), (Vento, 2020). One of the key features used in Vento (2020) is facies.

Adjacent research has focused on sedimentary facies interpretation using core imagery (Martin et al., 2021; Falivene et al., 2022). The workflow involved in image recognition utilizes the numeric red, green, and blue color channels of each core image pixel to train and make predictions of facies per image position. Additional workflow stages allow similar adjacent pixels to amalgamate into bed features in the interpreted core result. These studies find that interpretation performance varies between 60% and 70% accuracy with performance at its best in highly distinct facies classes (i.e. sandstone and mudstone) (Martin et al., 2021). Similar work incorporates petrophysical logs with imagery to achieve similar total performance in more complex depositional settings (Houshmand et al., 2022).

In this study, properties observed and measured directly from rocks using field-collected measured sections provide a multi-variable feature dataset to predict facies in comparison with using RGB image color channels in previous studies. The results demonstrate facies prediction in measured section data as a core-analogous data type. This workflow is not applied to a subsurface core dataset, however, the groundwork is in place to do so in future work.

2. GEOLOGIC DATABASE DESCRIPTON

2.1 Geologic Setting

Sedimentological data derived from over a decade of research from the Tres Pasos Formation of the Magallanes Basin of Chile forms the basis of this study. The strata in the Tres Pasos Formation are deepwater slope channel deposits, generated from sediment transported from up-system deltas into the Cretaceous ocean basin.

2.1.1 Deepwater Stratigraphy

Deepwater stratigraphic architecture encompasses the complex arrangement of sedimentary structures and facies deposited in deep marine environments. Typically, these formations occur beyond the continental shelf edge. The architecture is primarily shaped by gravity-driven, submarine flow processes such as turbidity currents and debris flows which transport sediment from shallow to deepwater settings. The key elements of deepwater stratigraphic architecture to this investigation are the sedimentary facies. The work of Arnold H. Bouma introduced what is commonly referred to as the "Bouma Sequence," a descriptive model for turbidite deposits that revolutionized the understanding of deep-sea sedimentation (Fig. 3). The sequence, characterized by a division into five distinct facies (Ta, Tb, Tc, Td, and Te), provided a framework for interpreting the depositional environments of ancient turbidite systems and has been instrumental in the exploration of submarine fans and turbidite reservoirs (Bouma, A.H., 1962).

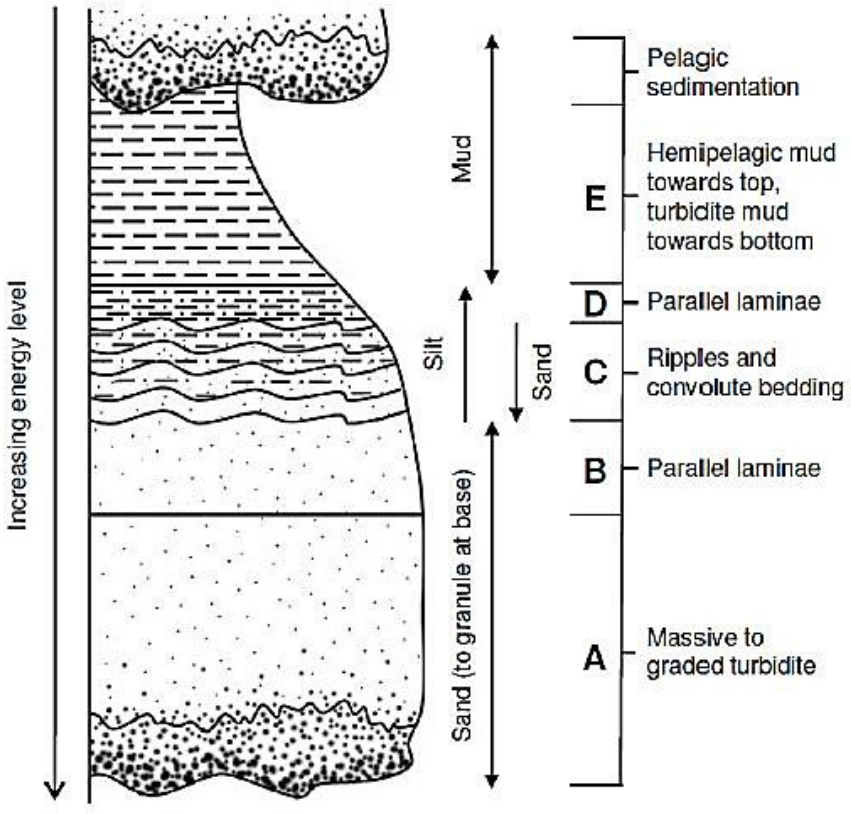

Figure 3: The Bouma sequence (Ta, Tb… etc.) of turbidites, modified slightly from Bouma, 1962 (Strekeisen, A., 2020; Bouma, A.H., 1962)

Beyond facies, stratigraphic architecture of deepwater systems is further complicated by the interplay between sediment supply, basin topography, and sea level changes (Walker, R.G., 1978). Fundamental contributions in understanding this architecture came from Roger G. Walker's (1978) work on deepwater sandstone facies and the concept of ancient submarine fans provided a crucial model for the exploration of stratigraphic traps. His contributions laid the groundwork for understanding the complexity of submarine fan systems and facilitate modern facies models such as the turbidite fan (Fig. 4) (Sprague, 2005).

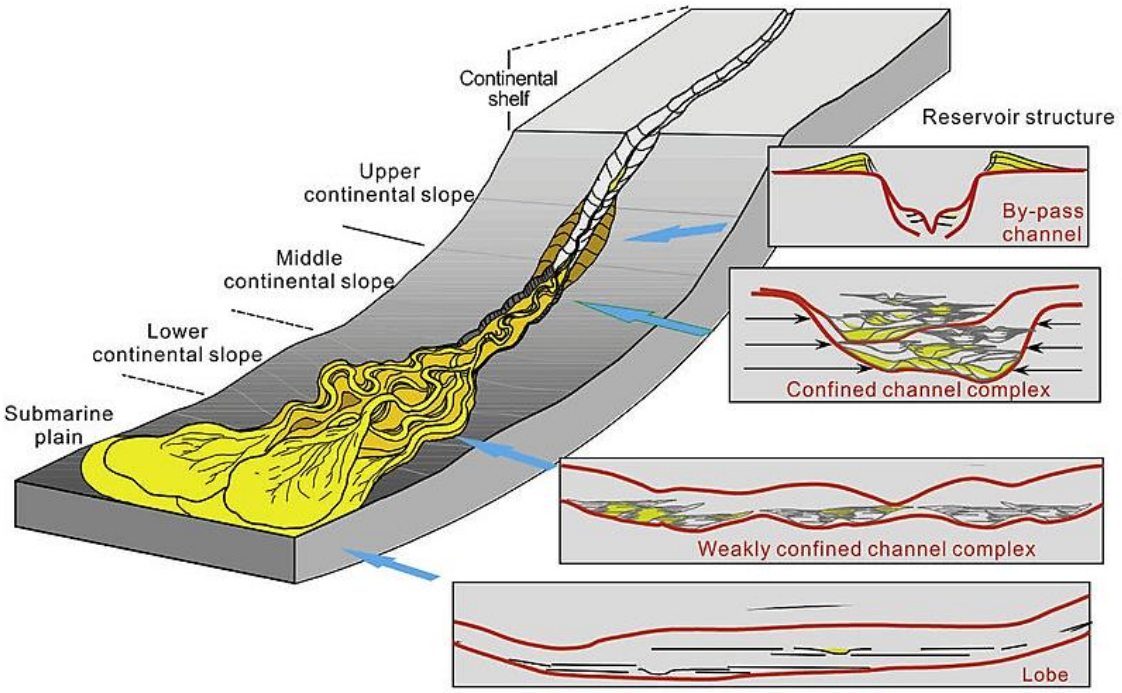

Figure 4: Sedimentary facies model of the turbidite fan. Structural cross-sections emphasize confinement and incision in the continental shelf and upper slope environments. Channel complexes then develop on the slope and gradationally lose confinement, terminating in lobes in on the submarine plain (Sprague et al., 2005).

Modern sedimentology has refined the deepwater depositional system model since the advent of modern process observations and the addition of more field outcrop analog studies. In addition, advancements in understanding eustatic controls on sedimentation have allowed deeper understanding of basin wide systems characteristics through seismic interpretation (Posamentier and Vail, 1988). This work laid foundations for later research that would focus on the basin-scale evolution of these hidden processes.

2.1.2 Magallanes Basin

The focus of this study is the deepwater slope strata of the Magallanes Basin (olive green Ktp shade in Fig. 5A). The prograding slope system of the Tres Pasos Formation (Fig. 5B) and more specifically, the deposits along the Figueroa clinoform surface (Fig. 5C) will be described in detail in the following section.

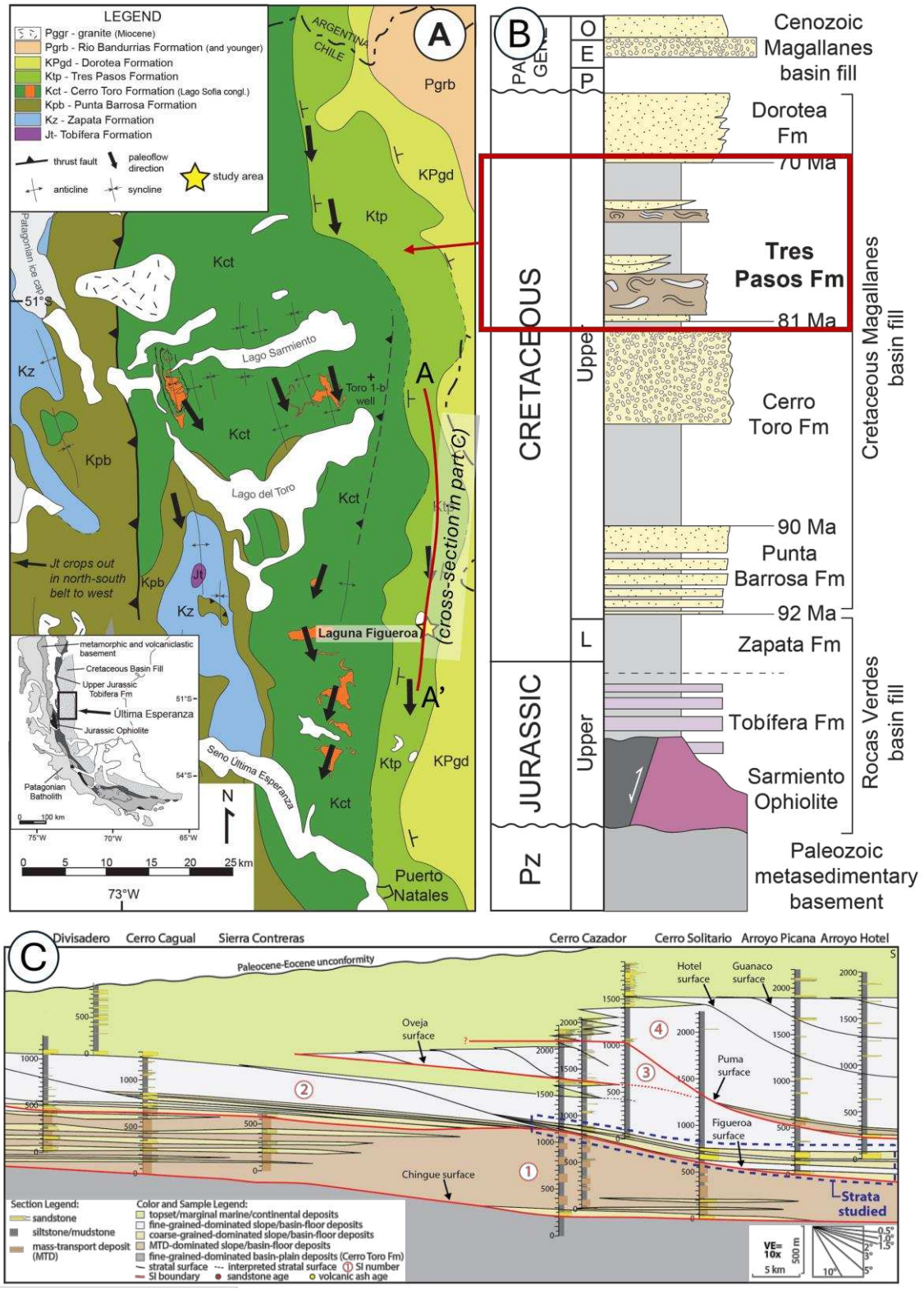

Figure 5: Location and strata of the study area examined in this thesis. A) Geologic map of the outcropping Cretaceous strata in the Magallanes basin, B) Stratigraphic column of the deepwater to deltaic strata, and C) Cross-section of the mapped Tres Pasos Formation with the study area highlighted (Modified from Romans, 2011).

Katz (1972) first recognized the Magallanes Basin as a retroarc foreland basin, having its long axis aligned with the Andean fold-thrust belt in Patagonia (Fildani and Hessler, 2005; Romans et al., 2011). The basin experienced phases of Jurassic extension and compression after the breakup of Gondwana, leading to the opening and subsequent closure of the Rocas Verdes Basin and the development of the retroarc basin (Dalziel et al., 1974; Fildani and Hessler, 2005; Romans et al., 2011; Wilson, 1991).

The depositional history of the Magallanes Basin initiates with the formation of a back-arc basin due to subduction of the Pacific plate beneath the South American plate (Late Jurassic – Early Cretaceous). The initial rifting phase is marked by a dark gray to black shale known as the Zapata Formation. The Zapata Formation can be infrequently interbedded with thin-sandy turbidites, together signaling a deepwater environment of deposition. This shale overlies marine volcaniclastics (Tobifera Formation) sourced from the adjacent arc (Fildani and Hessler, 2005). These early deposits indicate the initial basin accommodation from extensional tectonics.

As subduction continued (Late Cretaceous), the Magallanes Basin transitioned into a foreland basin setting (Fildani and Hessler, 2005). This dual history of thinning due to extension followed by compression leads to significant subsidence. This phase is characterized by marine transgression, leading to the deposition of extensive sandy turbidites which are well-documented in the Punta Barrosa Formation (Fildani and Hessler, 2005). Represented in Figure 5B, the transition from the Zapata Formation to the Punta Barrosa (~92-85 Ma) signifies the onset of the Andean orogeny in the Magallanes Basin (Wilson 1991; Romans et al., 2011). The Punta Barrosa exhibits characteristics of an unconfined depositional setting, featuring relatively thin sheets or fan-like lobate sand deposits that occupied a trough approximately 100 km wide, parallel to the fold-thrust belt (Fildani and Hessler, 2005; Romans et al., 2011).

The Late Cretaceous to Paleogene marks a third phase, characterized by confined deepwater deposition through channelized systems. This occurs concurrent with tectonic inversion and uplift in the Magallanes Basin, driven by continued subduction and the Andean orogeny (Romans et al., 2011). The beginning of this phase is marked the shift from Punta Barrosa to the Cerro Toro Formation (~86–80 Ma).

This transition is identified by the absence of coarse-grained beds and the appearance of the dark Cerro Toro mudstone (Katz, 1963; Romans et al., 2011). The Cerro Toro, predominantly shale, includes the conglomeratic Lago Sofia Member near the center of the stratigraphic package (Katz, 1963; Hubbard et al., 2008; Romans et al., 2011). Turbidite sandstones and debris flow deposits are locally present, with the Cerro Toro representing a channel-levee complex along the basin's axis, where channel bodies amalgamate further south along the paleo-slope (Jobe et al., 2009; Romans et al., 2011) (Fig. 5B). As marine transgression continued through the Upper Cretaceous, the Cero Toro Formation transitioned to a progradational slope depositional system. The Tres Pasos Formation (~80–70 Ma), the focus of this study, initially records this shift as a prominent sandstone layer present above the uppermost Cerro Toro (Katz, 1963; Romans et al., 2011). The lower Tres Pasos is characterized by lenticular to tabular sandstone packages and mass transport deposits (MTDs), while the predominantly fine-grained upper section incorporates coarse-grained deposits, including turbidites, structureless sandstone, and mudstone-clast conglomerates (Shultz and Hubbard, 2005; Romans et al., 2011). The Tres Pasos displays channel complex geometry and amalgamation strongly influenced by slope position (Hubbard et al., 2010; Romans et al., 2011) (Fig. 5B).

Following the deposition of Tres Pasos, the Magallanes basin enters a fourth phase dominated by shallow marine deposits which also include fluvial systems. The Dorotea Formation (~72–65 Ma) conformably overlies the Tres Pasos, concluding the filling stage of the Magallanes Basin (Covault et al., 2009; Romans et al., 2011). Characterized by a significant sandstone layer overlying the mudstone-rich upper Tres Pasos Formation, the Dorotea consists primarily of shallow-water sandstone (Covault et al., 2009; Hubbard et al., 2010). The interpretation suggests an upward-shallowing trend, transitioning from upper slope to shallow marine, deltaic, and non-marine strata at the uppermost section (Macellari et al., 1989; Hubbard et al., 2010).

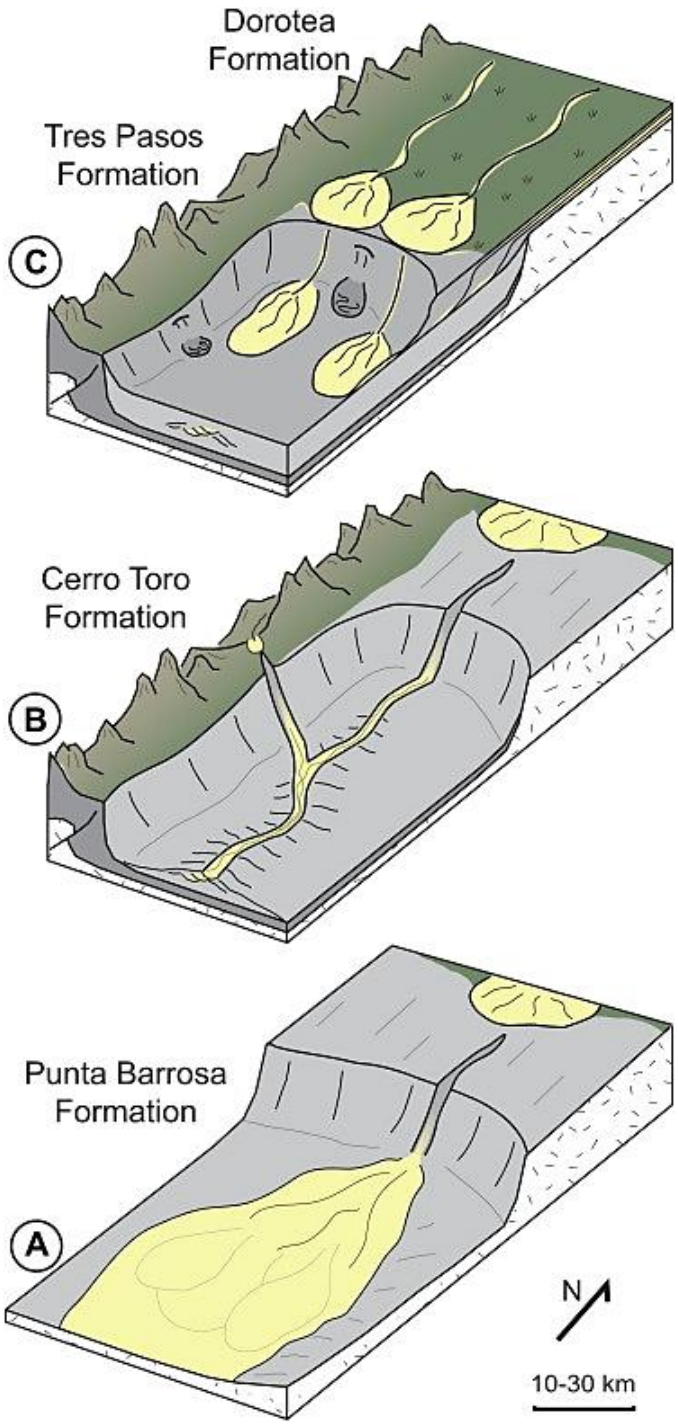

Figure 6: Schematic diagrams depicting style of deepwater deposition and architecture in the Magallanes Basin. Transition from fully continental crust (north) to attenuated crust (south) inherited from older Rocas Verdes backarc basin. (A) The oldest phase of deepwater sedimentation is represented by the Punta Barrosa Formation, which is expressed as tabular to slightly lenticular sandstone bodies in an unconfined setting. (B) The Cerro Toro Formation is characterized by a foredeep-axial channel-levee system filled with conglomeratic deposits (C) The final phase of deepwater sedimentation is genetically linked Tres Pasos (slope) and Dorotea (delta/shelf) Formations. Deepwater accommodation filled as the Tres Pasos slope systems prograded southward. Modified from (Romans et al., 2011).

2.1.3 Measured Sections and Sedimentary Facies

The Chile Slope System consortium has contributed significantly to the understanding of the evolution of channelized-deepwater slope systems, through their leveraging of the unique outcrops of the Magallanes Basin, in Chile (Romans et al., 2011). The database generated by this research is indexed as measured sections of stratigraphy. Figure 2 shows the broader interpretation recorded across multiple outcrops in the database. Training data originates from the Laguna Figueroa outcrop of the Tres Pasos Formation at the "XS-8" position in the first and third panels. This displays the broader interpretation generated by correlation between the sections in the CSS database.

Each measured section contains bed-by-bed samples, each with twenty-one features (Table 1). Among these are ten numeric statistics and one facies interpretation. Total sample counts and per-facies features counts can be found in Table 1 and 2. For MS at the Laguna Figueroa outcrop, each sample is assigned a facies code from among 7 ("out of channel," non-amalgamated siltstone dominated, non-amalgamated sandstone dominated, semi-amalgamated, highly amalgamated sandstone, mass transport deposits, and "covered section"). Figure 1B and Figure 8 both depict graphic representations of this facies code system, styled like a measured section drawing typically recorded in field observations: *highamalsand* = Yellow, *semiamal* = Orange, *nonamalsand* = Grey, *nonamalsilt* = Black).

Outcrops of Tres Pasos are interpreted into multiple sedimentary facies which are subsequently recorded in the Chile Slope Systems database. The primary lithologies are sandstone and shale with individual grains ranging from fine-upper sand to clay. These lithologies interbed to form associations of facies in the channel system framework (Fig. 1). Channels present the most prominent architectural elements in deepwater depositional environments, and their interplay with stratigraphic architecture gives rise to facies associations.

Table 1: Database Features Utilized in this Study.

Database Features: (*training features)	**Testing and Training** *Total Count (NA values)*	**Prediction**	**Description**
BedThickness*	15915 (0)	31521	Total thickness values for each bed (Jac-staff)
gs_min*	15915 (237)	31521	Minimum observed grain size (hand lens/grain size card)
gs_max*	15915 (237)	31521	Maximum observed grain size (hand lens/grain size card)
gs_p10*	15915 (253)	31521	Tenth percentile grain size (hand lens/grain size card)
gs_p50*	15915 (253)	31521	Fiftieth percentile grain size (hand lens/grain size card)
gs_p90*	15915 (253)	31521	Ninetieth percentile grain size (hand lens/grain size card)
gs_mean*	15915 (237)	31521	Mean grain size (hand lens/grain size card)
gs_at_bed_bottom*	15915 (240)	31521	Grain size above the lower boundary (hand lens/grain size card)
gs_at_bed_top*	15915 (252)	31521	Grain size below the upper boundary (hand lens/grain size card)
gs_sand_thickness*	15915 (237)	31521	Equivalent bed of only the sand portion in a sample (Net to Gross)
FaciesCode	15915 (0)	**TARGET**	Interpreted sedimentary facies encoded as an indicator variable (1, 2, 3, 4) for supervised machine learning workflows

The bed scale characterization of the Laguna Figueroa channelized outcrop from Macauley and Hubbard (2013) identified several key facies groupings [F1, F2, F3, F4]. These include thick-bedded amalgamated *highamalsand* (Facies 1; F1), the coarsest facies (Fig. 7.1). Coarse sandstones represent permeable and porous containers that can host hydrocarbon accumulations. Less amalgamated, sandstone dominated facies (F2) are *semiamal*. Facies 2 represent thick- to thin-bedded semi-amalgamated sandstones (Fig. 7.2). As the proportion of siltstone increases, facies changes to 3. *Nonamalsand* (Facies 3; F3) is thin-bedded non-amalgamated sandstone and siltstone, sandstone dominated (Fig. 7.3). Originally, Facies 3 and 4 represented the sandstone and siltstone dominated members of the non-amalgamated class but now are distinct classes in this analysis. *Nonamalsilt* (Facies 4; F4) is thin-bedded, non-amalgamated sandstone and siltstone, siltstone dominated. This facies captures some of the draping beds at the base of channel elements (Fig. 7.4).

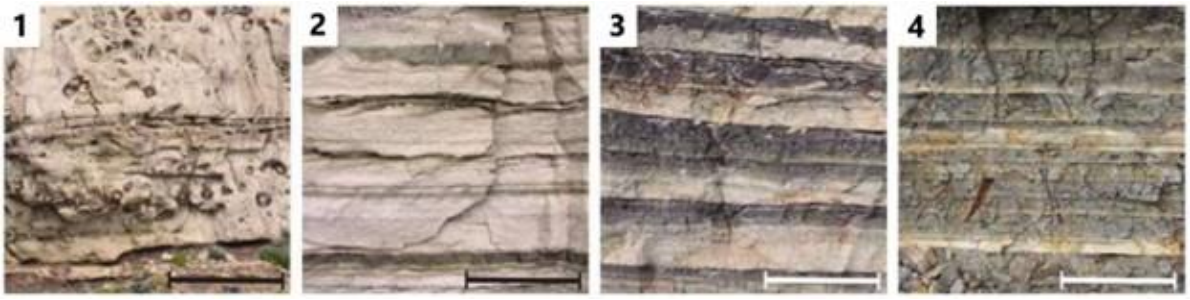

Figure 7: Bed-scale target facies of the Tres Pasos Formation. (1) thick-bedded sandstone of FaciesCode 1 (scale bar is 2 m long); (2) thick-bedded semi-amalgamated sandstone of FaciesCode 2 (scale bar is 70 cm long); (3) non-amalgamated sandstone and mudstone of FaciesCode 3 (scale bar is 20 cm long); and (4) thin-bedded mudstone with sandstone of FaciesCode 4 (scale bar is 20 cm long). Adapted from (Merovitz, 2024).

These facies configurations constitute the internal fill of channel elements, with a strong correlation to the position within the channel. F1 is predominantly found along the channel axis, whereas F3 is typically associated with the channel margins. F2 is located in transitional "off-axis" areas, situated between F1 and F3, and F4 is dominantly at the base (Macauley and Hubbard, 2013; Meirovitz et al., 2020).

Figure 8: Example measured section data from Fig 100 section in the Laguna Figueroa outcrop. The left image is the drafted field data, the right image is the interpreted facies for this section. This is one of the measured sections used in training and testing.

For the purposes of measuring section, previous studies incorporate three additional classes (covered section, out of channel, and mass transport deposits (MTDs) to the facies scheme. The “covered section” class shows when data was unavailable. The "out of channel" class includes areas outside of channel elements or MTDs which is characterized by a lack of flow capabilities and storage properties. Mass transport deposits (MTDs) then result from gravity-induced mass-wasting events and display varied lithologies acting as reservoirs, seals, or sources. Within the Tres Pasos Formation, MTDs are characterized by chaotic bedding of mudstone and sandstone with occasional large clasts and organic matter, marking transitions between sandy channel elements or at the base of channel complexes (Hubbard et al., 2009; Fletcher, 2013; Romans et al., 2011).

2.2 Database Description

The CSS measured section database contains sections from the Tres Pasos Formation from multiple outcrops along the north-south striking belt (Fig. 2). The database records twenty-one data features at each bed observed in the field (Table 1). Among these features, 12 are observed or recorded including depth at bed bottom, depth at bed top, BedThickness, grain size minimum, grain size maximum, P10 of grain size, P50 of grain size, P90 of grain size, grain size mean, grain size sample count, grain size at bed bottom, and grain size at bed top. A further eight are interpreted features. These include, measured section name, bed number, element number, complex number, complex-set number, facies code, architectural element position, facies association. Lastly, a single feature is computed (Net-to-Gross). Net-to-Gross (NTG) evaluates the proportion of productive or sandstone thickness relative to the overall thickness of the observed section. NTG is determined by the ratio of the net value to the gross value for each channel sample (Vento, 2020). It is derived from 5 cm-scale observations made within individual bed samples to determine the grain size distribution at finer than individual bed scale.

Other features to note are the Bed Number, Element Number, Complex Number, and Complex-Set. These together form the detailed architecture interpretation of the Laguna Figueroa outcrop. They represent geologic interpretations beyond the scope of this study and are excluded from training/testing datasets.

2.2.1 Database Class Statistics

The dataset collected on the Tres Pasos Formation across multiple outcrops contains roughly 33,000 individual bed samples. The measured sections used in this study are digitized tables of sedimentologic grain size observations. General database statistical characteristics are presented in Table 1. With non-facies classes removed and the target facies remaining, the training/testing database is distributed as shown in Figure 9. Table 2 shows the original facies interpretation from the Laguna Figeroa outcrop with the training facies for this study denoted with a check mark.

Table 2: Total Database Facies Count. Refer to Figure 7 for descriptions of facies.

Classification Facies Names	FaciesCode	Used in ML?	Database Count (w/out NA)	Database Count	Percent Total	Percent Missing	Missing Count
outofchannel	0	X	323	324	2.10%	0.30%	1
highamalsand	1	✓	2067	2070	13.10%	0.10%	3
semiamal	2	✓	1939	1955	12.30%	0.80%	16
nonamalsand	3	✓	4688	4744	29.80%	1.10%	56
nonamalsilt	4	✓	6654	6733	42.30%	1.10%	79
masstransport	5	X	40	40	0.30%	0.00%	0
covered	8	X	9	49	0.00%	82.00%	40
uninterpreted	-99	X	9469	NA	NA	NA	NA

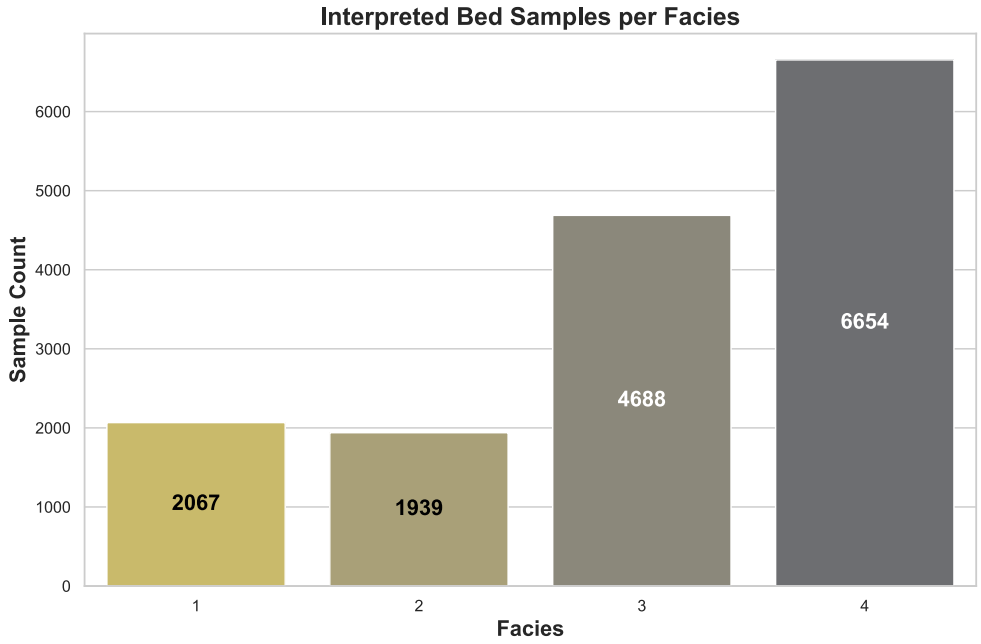

Figure 9: Total Database Distribution Between Target Facies (Classes)

Focusing on the four target facies, target classes are significantly more abundant in Facies 3 and 4 classes than in Facies 1 and 2 classes. The "sandier" facies (1 and 2) represent roughly 26% of the total sample count while the "siltier" facies (3 and 4) dominate with ~74%.

Figure 10 depicts the distribution of features within each target facies. Each facies presents a distinct distribution in the feature space. Facies 1 shows the tightest inter-quartile ranges of all facies. All features tended to 10^0 magnitude with similar median values. The other three target facies (*semiamal*, *nonamalsand*, *nonamalsilt*) have much more similar distributions of features. For example, Facies 2 and 3 have similar interquartile range (10^{-2} to 10^{-1}) in all features but differ in median values. The third class (*nonamalsand*) has a slightly lower median value in all features than class two (*semiamal*). In a comparable way, the fourth class (*nonamalsilt*) has most features' interquartile range fall within 10^{-2} to 10^{-1} but median values are now much lower, the lowest among all classes.

Important data features to note are the variability of both BedThickness and gs_sand_thickness between classes. In the same way, the similarity of the remaining data features (gs_min, gs_max, gs_p10, gs_p50, gs_p90, gs_mean, gs_at_bed_bottom, gs_at_bed_top) is significant for this type of quantitative analysis.

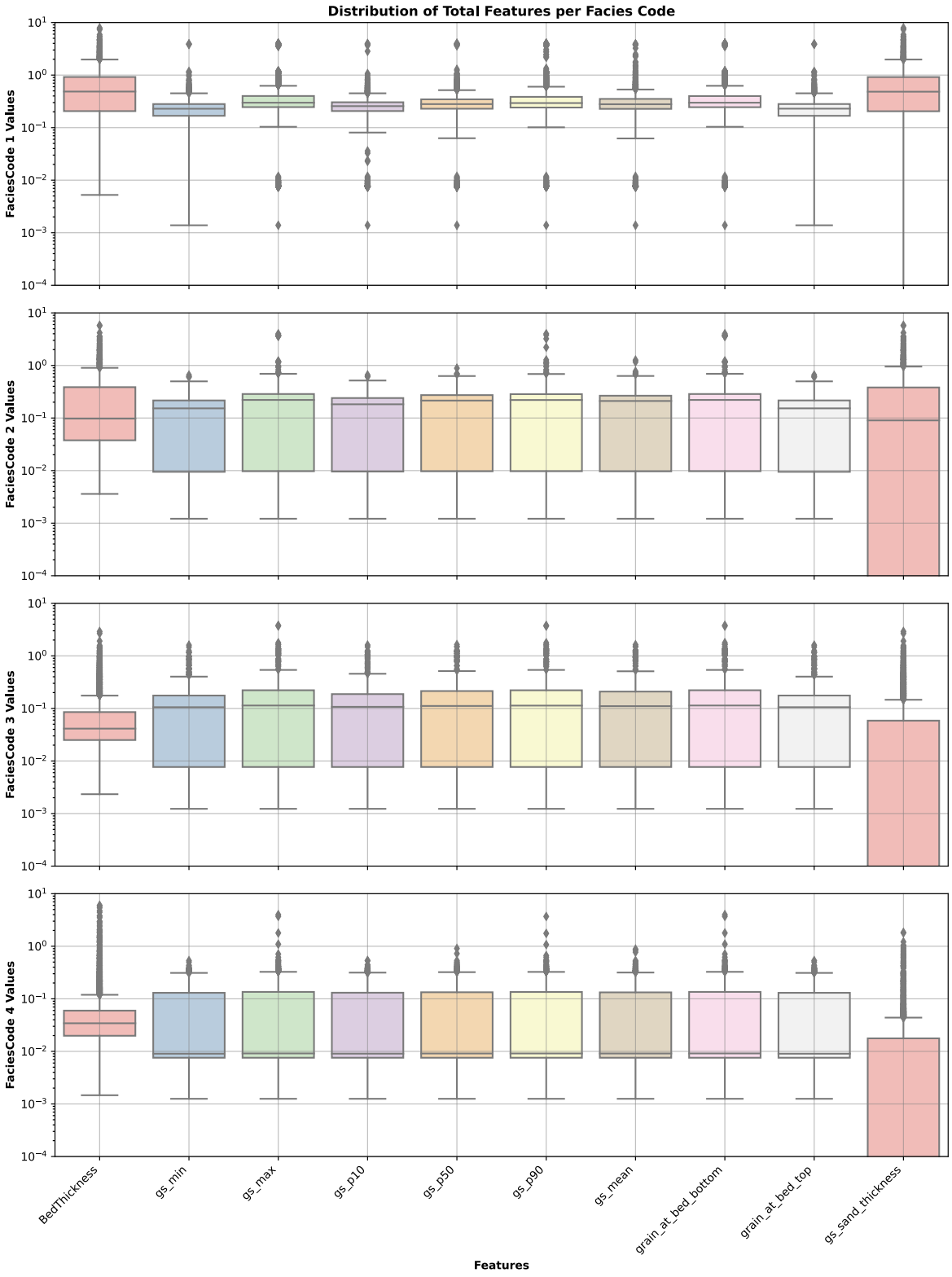

Figure 10: Total Database Feature Distribution, Differentiated by Interpreted FaciesCode

3. MACHINE LEARNING METHODS

Methods used in this study are broadly distinguished into Supervised and Unsupervised methods. The interplay of both types allows for a robust investigation. Each method presents a unique window into this facies prediction challenge. All methods utilized in this study are prepared as python scripts, isolated from one another. Each technique requires bespoke data preparation prior to classification. The python software used was Jupyter Notebooks and all data is hosted on an Amazon web services server. Scripts associated with each method can be found on GitHub under the user “r0nnau.” Measured section visualizations are generated by code adapted from the open-source project “StratLoggerPlotter” available on GitHub.

3.1 Unsupervised Methods

In the context of machine learning, the term "unsupervised" refers to a type of algorithm or learning process that operates on data without explicit instructions on what to predict or infer. The term is often used in contrast to "supervised learning," where algorithms are trained on a labeled dataset, meaning that each input in the training set is paired with the correct output. Unsupervised learning algorithms seek to identify patterns, structures, or insights within a dataset without the guidance of a predetermined outcome or target variable. These algorithms are useful for exploratory data analysis, clustering, and dimensionality reduction, among other tasks. The key characteristic of unsupervised learning is its ability to discover the underlying structure of the data autonomously.

Techniques like Principal Component Analysis (PCA) are used to reduce the number of variables under consideration by obtaining a set of principal variables. Clustering involves grouping a set of objects in such a way that objects in the same group (called a cluster) are more like each other than to those in other groups. K-Means clustering is a popular example of a clustering algorithm. The application of unsupervised learning is particularly beneficial in situations where collecting or labeling data is impractical, expensive, or when it is uncertain what patterns the data may possess.

3.1.1 Principal Component Analysis

PCA is a statistical technique used in machine learning to simplify data by reducing its dimensions while preserving as much variance as possible (Jolliffe, I.T., 1990). This is achieved by transforming the original variables into a new set of uncorrelated variables known as principal components. These components are ordered so the first few retain most of the variation present in all the original variables. PCA helps in analyzing data by identifying patterns, removing redundancies, and compressing data without significant loss of information— this technique is often used for exploratory data analysis and predictive modeling

3.1.2 K-Means Clustering

The K-Means algorithm is a widely used method in unsupervised machine learning for partitioning a dataset into *k* distinct, ideally non-overlapping, clusters. Introduced in 1967, it aims to group data points into clusters in such a way that each point belongs to the cluster with the nearest mean, minimizing the within-cluster variances (MacQueen, 1967). The algorithm is particularly favored for its simplicity and efficiency in processing large datasets. The procedure of the K-Means algorithm can be described as follows:

1. Initialization: Select *k* initial centroids randomly or based on a heuristic from the dataset. These centroids are the initial centers of the clusters.

2. Assignment Step: Assign each data point to the nearest cluster by calculating the distance between each data point and each centroid. The distance metric used is the Euclidean distance, though others can be used depending on the nature of the data and the desired outcome.

$$\boldsymbol{E(d_{obvs}, d_{cent}) = \sqrt{(x_d - x_c)^2 + (y_d - y_c)^2 + (z_d - z_c)^2}}$$

Equation 1: As data points for observations and centroids are in the form *d(x, y, z)*, then the function *E* provides their 3-dimensional distance.

3. Update Step: Calculate the new centroids as the mean of all points assigned to each cluster, effectively moving the centroid to the center of the cluster.

4. Iteration: Repeat the Assignment and Update steps iteratively until the centroids no longer change significantly, indicating that the algorithm has converged, or a predetermined number of iterations is reached.

$$\boldsymbol{c_i}^{(\boldsymbol{t+1})} = \frac{\mathbf{1}}{\left|\boldsymbol{E_i}^{(\boldsymbol{t})}\right|} \sum_{\boldsymbol{d_j \epsilon E_i}^{(\boldsymbol{t})}} \boldsymbol{d_j}$$

Equation 2: The sum of all distances in a class up to "*j*", divided by the number of observations "*i*" gives the new centroid "*c*" at time "*t*" in (x, y, z) form.

The objective of the K-Means method is to minimize the sum of squared distances between each point and its centroid, known as the within-cluster sum of squares (WCSS). The algorithm's efficiency and simplicity make it a popular choice for cluster analysis in various applications. However, the K-Means algorithm has several limitations including the number of clusters (*k*) must be specified in advance. The number of clusters (*k*) is often determined using a PCA analysis. Additionally, K-Means is sensitive to the initial selection of centroids, which can lead to different results on different runs as well as outliers, which can distort the cluster formation. These conditions all limit the data types that can be accurately classified. Despite these limitations, K-Means remains a fundamental tool in the data scientist's toolkit, especially for exploratory data analysis where the aim is to gain insights into the underlying structure of the data.

3.2 Supervised Methods

As stated above, "supervised" machine learning refers to a type of algorithm or learning process that operates on data with explicit instructions on how to make a classification. Supervised learning, in the realm of ML, denotes a workflow where the algorithm is trained on a labeled dataset. This dataset comprises input-output pairs, where each input (or feature vector in *n*-D dimensions) is associated with the correct output (also known as the target or label). The principal objective of supervised learning is for the algorithm to devise a model that, given new, unseen inputs, can accurately predict their corresponding outputs.

The supervised learning process involves two main phases, the first of which is training. During this phase, the algorithm iteratively makes predictions on the training data and is corrected by comparing

its predictions against the actual outcomes. The model adjusts its parameters to minimize the difference between its predictions and the true outputs. This phase continues until the model achieves a satisfactory level of accuracy or meets other stopping criteria. The subsequent, second primary phase is testing. Once trained, the model's performance is evaluated using a separate dataset not seen by the model during training. This evaluation phase helps to gauge how well the model generalizes to new data. Supervised learning can be further categorized into two primary types, based on the nature of the output variable.

The first type is regression. In regression tasks, the output variable is continuous, and the model predicts a quantitative output. An example is house price prediction, where the model estimates a house's price based on features like size, location, and the number of bedrooms. The second type is classification. In classification tasks, the output variable is categorical, meaning the model predicts which category or class the new input belongs to. A common example is email spam detection, where each email is classified as either "spam" or "not spam." This type of supervised learning best resembles the process of manual geologic interpretation.

3.2.1 Least Squares Solution

Linear regression analysis is a foundational statistical method for modeling the relationship between a dependent variable and one or more independent variables. It allows for the assessment of the strength and nature of the relationship between the dependent variable (Facies) and each independent variable (Database Features) (Neter et al., 1996). The Least Squares solution in classification attempts to adapt the linear regression framework to a binary classification problem by minimizing the sum of squared errors between the predicted scores and actual class labels (Suykens and Vandewalle, 1999). While it offers simplicity and interpretability, it lacks the probabilistic interpretation and performance robustness of more specialized classification methods like Logistic Regression. It is used more as a conceptual bridge between regression and classification rather than as a primary classification method in practical applications.

3.2.2 Logistic Regression

Logistic Regression is a statistical method used for binary classification problems, where the outcome variable (target) is categorical and typically represents one of two classes. The method is a part of the broader family of generalized linear models (GLMs) (Scott Long, 1997). By modeling the probability of a binary outcome using the logistic function (Eqn. 3, Eqn. 4), it allows for clear insights into the relationships between database features and the outcomes (facies) (Scott Long, 1997). Its simplicity and effectiveness make it a popular choice in various domains where binary classification is required.

$$\boldsymbol{\sigma(z)} = \frac{\mathbf{1}}{1 + \mathrm{e}^{-\boldsymbol{z}}}$$

Equation 3: Sigmoid (Logistic) function used for Logistic Regression classification, where "*z*" (Eqn. 4) is the linear combination of the input feature (Scott Long, 1997).

$$\boldsymbol{z} = w^{\boldsymbol{T}}x + b$$

Equation 4: Linear function with "*t*" class relationships, where "*w*" is the vector of weights, "*x*" is the vector of input features, and "*b*" is the bias vector (Scott Long, 1997).

In the case where there are more than two classes in the system (e.g. 4 facies), the activation function for Logistic Regression must change. The softmax function (Eqn. 5) generalizes the sigmoid function, converting vectors of raw scores into a probability function over multiple class labels (Bishop, 2006). The classification with the highest probability is then the prediction result.

$$\boldsymbol{\sigma(z)} = \frac{\mathrm{e}^{\boldsymbol{zi}}}{\sum_{j=1}^{K} \mathrm{e}^{\boldsymbol{zj}}}$$

Equation 5: Softmax function used for multinomial Logistic Regression classification, where "z" is the vector combination of the input features. Softmax applies the exponential function to elements of the input vector, applying weight roughly to the vector's maximal element (soft-max) (Bishop, 2006).

In addition to altering the activation function for multinomial regression, the Logistic Regression model requires some hyperparameter values be specified ahead of training and testing. To achieve a higher level of performance than an unoptimized baseline, an open-source python library's (sklearn) hyperparameter grid search tool tested the classification performance over multiple configurations of possible model parameters. The following table (Table 3) displays the parameters used in the grid search and the resulting best configuration of hyperparameters.

Table 3: Logistic Regression Hyperparameters

Logistic Regression	C	max_iterations	penalty	solver
Hyperparameters Searched	0.1, 1, 10	10, 100	l1, l2	liblinear, saga
Best Hyperparameters	0.1	10	l1	liblinear

3.2.3 Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA)

Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) are two classical methods in statistical and machine learning for classification tasks, both deriving from the general framework of discriminant analysis. They aim to predict the category or group to which a new observation belongs, based on a set of predictor variables. The primary difference between LDA and QDA lies in the assumptions they make about the distribution of the predictor variables and the decision boundary between classes.

LDA assumes that the predictor variables have the same variance-covariance matrix across all classes (Fisher, 1936). This implies that different classes have similar shape and orientation in the predictor space, differing only in their means (Bishop, 2006). Due to this assumption, the decision boundary between any two classes in LDA is linear, which is where the method gets its name. In contrast, QDA does not make this assumption. It allows each class to have its own variance-covariance matrix. This flexibility means that QDA can model classes with different shapes and orientations (in *n*-D space) by adjusting the covariance structure for each class (Duda & Hart, 1973). Consequently, the decision boundary between classes in QDA

can be quadratic (or curvilinear), allowing for more complex relationships between the predictor variables and the class labels.

These distinctions lead to important variations in complexity and flexibility for these methods. LDA is less flexible than QDA because it constrains the classes to have the same covariance matrix. However, this simplicity can be an advantage when the sample size is small (not the case in this study) or when the linear decision boundary assumption is approximately correct, as it tends to be more robust and less prone to overfitting in these scenarios (Bishop, 2006). QDA is more flexible due to its ability to model different covariance structures for each class, which can lead to a better fit when the true decision boundaries are non-linear. However, this increased flexibility comes at the cost of a higher risk of overfitting, especially in scenarios with limited training data relative to the number of predictor variables (Bishop, 2006). The choice between LDA and QDA often hinges on the underlying distribution of the data, and the specific requirements of the classification task. LDA is preferred for when linear boundaries are sufficient or when data is limited, while QDA may offer better performance in capturing complex relationships at the expense of requiring more data to avoid overfitting (Bishop, 2006).

$$\vec{C} = (\boldsymbol{f_1}, \boldsymbol{f_2}, \boldsymbol{f_3}, \cdots, \boldsymbol{f_n})$$

Equation 6: Each facies association class can be represented the vector "*c*", a matrix of data features "*f*", equal to the dimensionality of the dataset. In this study, ten dimensions (10-D).

3.2.4 Random Forest

The Random Forest algorithm is a learning method used for classification tasks which operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the predicted classes (Breiman, 2001). Introduced by Breiman (2001), random forests correct for individual decision trees' habit of overfitting to their training set, providing a more generalized and robust model. The algorithm can be described through the following key steps.

Random Forest employs a technique called “bagging” where multiple subsets of the original dataset are created, known as bootstrap samples (Breiman, 2001). Each subset is used to train a separate decision tree. The randomness introduced by sampling with replacement ensures that the trees are diverse. For classification tasks, the prediction by the Random Forest is made by majority voting. Each tree votes for a class, and the class receiving the majority of votes becomes the model’s prediction.

Random Forest stands out among learning algorithms currently available, demonstrating robust performance across a multitude of problems. It can manage thousands of input variables without the need for variable deletion and offers insights into which variables are significant in the classification process (Biau, 2012). Additionally, Random Forest is adept at estimating missing data, retaining its accuracy even when a sizable portion of the data is missing. Despite these strengths, there are some limitations to consider. The algorithm can demand considerable computational resources, particularly with a vast number of trees and/or extensive datasets, though it generally remains fast (Breiman, 2001). Unlike a single decision tree, Random Forest's complexity, derived from hundreds or thousands of trees contributing to the final decision, makes it challenging to interpret. Furthermore, it tends to overfit datasets that are exceptionally noisy. Nevertheless, Random Forest's versatility allows it to handle various data types, including numerical and categorical, cementing its status as a valuable asset in the arsenal of machine learning practitioners.

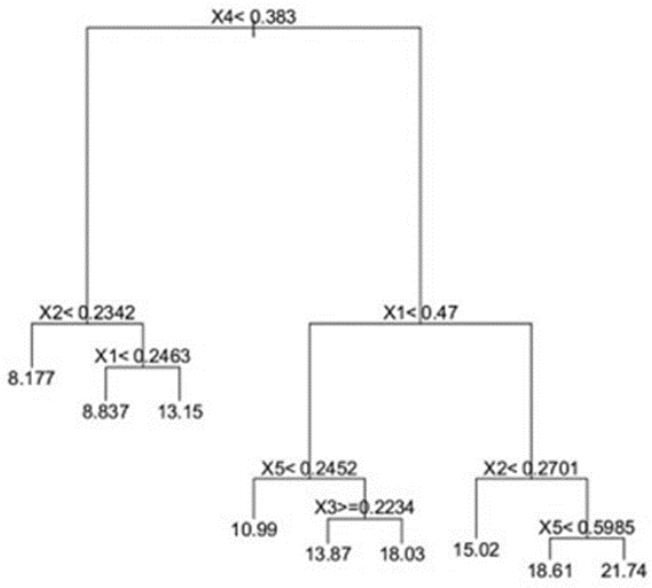

Figure 11: Example decision tree, display the binary decision architecture and outputs at decision points. Modified from (Biau, 2012).

The Random Forest model requires some hyperparameter values to be specified ahead of training and testing. To achieve a higher level of performance than an unoptimized baseline, the same sklearn grid search tool tested possible configurations for a Random Forest model. The following table (Table 4) displays the parameters used as well as the best configuration of hyperparameters.

Table 4: Random Forest Hyperparameters.

Random Forest	n_estimators	max_depth	min_samples to split	min_samples to leaf
Hyperparameters Searched	10, 20, 30, 40, 50, 60	None, 10, 20, 30, 40, 50	1, 2, 6, 10, 20	1, 2, 6, 8, 14
Best Hyperparameters	50	20	10	1

3.2.5 Neural Network Classifier

A Neural Network is composed of layers: an input layer, one or more hidden layers, and an output layer. Each neuron (individual calculation) in a layer is connected to every neuron in the subsequent layer through weights. Additionally, each neuron (except those in the input layer) has a bias term. The weights

and biases are initially set to random values and are adjusted during the training process. Once initiated, forward propagation begins computing the weighted sum of the inputs, adding the bias, and applying an activation function to the result to obtain the output of each neuron. The process starts from the input layer and moves through the hidden layers until the output layer is reached, producing the network's prediction.

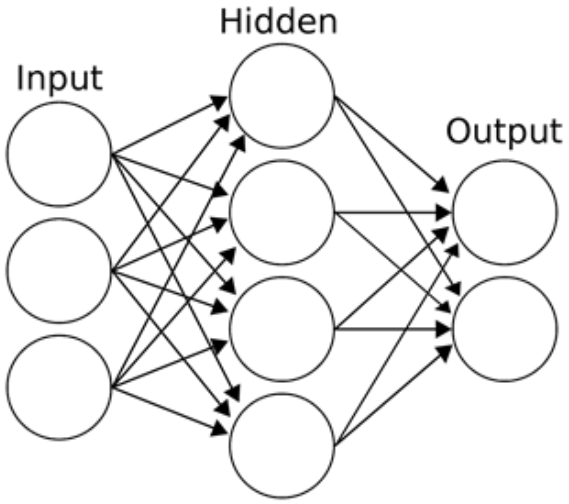

Figure 12: Schematic diagram connecting neurons (circles) through bias (arrows) to subsequent layers. Modified from (Robinson, 2023).

The Neural Network model requires some hyperparameter values be specified ahead of training and testing. The same hyperparameters optimization workflow utilized in Logistic Regression and Random Forest models generates the following best configuration of hyperparameters for a Neural Network model (Table 5).

Table 5: Neural Network Hyperparameters.

Neural Network	n_epochs	n_hiddens	activation	solver	alpha	learning_rate
Hyperparameters Searched	10, 50, 100	20, 50, 100, (20, 20), (50, 50), (100, 100), (20, 50), (50, 100)	'tanh', 'relu'	'sgd', 'adam'	0.0001, 0.05	'constant', 'adaptive'
Best Hyperparameters	10	(100, 100)	relu	adam	0.0001	constant

3.3 Performance Metrics

Utilizing many types of machine learning methods necessitates various kinds of evaluation metrics be used for each method. The following are explanations of the metrics used in this study.

3.3.1 Accuracy

Accuracy is the simplest common metric, defined as the proportion of correctly classified observations to the total observations. It is a good measure when the target classes are well balanced but can be misleading when dealing with imbalanced classes.

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{Correct\ Classifications}}{\boldsymbol{Total\ Classifications}}$$

Equation 7: Equation for classification accuracy.

This metric is effective in comparing class to class performance or performance variation due to a particular database feature. In this analysis, facies correlate well with BedThickness, with thick *highamalsand* facies representing an important class. To interpret performance for this class and varying BedThickness, a normalized accuracy metric is generated by weighing each classification by its proportion of total BedThickness for a section. This applies higher weights to thicker beds giving a measure of performance in *highamalsand* facies along with the thinner, finer facies classes.

3.3.2 Recall

Recall is another common metric. It is the ratio of correctly predicted positive observations to all observations in the actual class. It measures the ability of the classifier to capture all relevant instances in a class. A false negative is an incorrect classification of the class of interest as another class (Hall, 2016).

$$\boldsymbol{Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}}$$

Equation 8: Equation for classification recall. The likelihood that you make a correct classification for a given class (Hall, 2016).

3.3.3 Precision

Precision (Positive Predictive Value) is a similar ratio, measuring correctly predicted positive observations relative to the total predicted positives (Eqn. 7). It is a measure of the quality of the positive class predictions (Hall, 2016). A false positive is an incorrect classification where another class is predicted to be the class of interest.

$$\boldsymbol{Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}}$$

Equation 9: Equation for classification precision. The likelihood that a classified sample has the correct class (Hall, 2016).

3.3.4 Inertia

Inertia, specifically in the context of methods like K-Means clustering, refers to a metric for internal cohesion of the clusters formed by the algorithm. It quantifies the sum of squared distances between each sample in a cluster and the centroid of that cluster. It is a measure of how internally similar the samples of each cluster are, which indirectly reflects how well the clustering algorithm has performed in grouping similar data points together.

$$I = \sum(Sample - Centroid)^2$$

Equation 10: General Form of Inertia "*I*" Equation (sum of squared minimum difference).

Inertia serves as a crucial criterion for evaluating the quality of the cluster formations. A lower value of inertia indicates that clusters are dense and well-separated, which suggests a better clustering structure. However, it is important to note that using inertia alone to determine the optimal number of clusters can sometimes be misleading, especially with data that is not inherently spherical or if the clusters have different variances. In such cases, other metrics such as the silhouette score might be used alongside inertia to get a more comprehensive understanding of the cluster quality.

3.3.5 Silhouette Score

The concept of the silhouette score in clustering analysis was introduced by Peter J. Rousseeuw in 1987, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis." In the paper, Rousseeuw presents the silhouette score as a measure to describe how well each sample has been classified when assigned to clusters, by comparing the tightness and separation of the clusters.

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

Equation 11: General Form of Equation for Silhouette Score (Rousseeuw, 1987).

Equation 10 depicts silhouette score (s), where a is the mean distance between a sample and all other points in the same cluster (intra-cluster distance), and b is the mean distance between a sample and all other points in the nearest cluster to which the sample does not belong (nearest-cluster distance). The silhouette score for a set of samples is then the average of the individual scores. This score ranges from -1 to $+1$. A score close to $+1$ indicates that the data point is well-matched to its own cluster and poorly matched to neighboring clusters. A score of 0 indicates that the data point is on or close to the decision boundary between two neighboring clusters. A score close to -1 indicates that the data point may have been assigned to the wrong cluster (Rousseeuw, 1987). Overall, the silhouette score is a powerful and popular tool for evaluating clustering quality, helping in both validating the results of specific clustering assignments and guiding the selection of the number of clusters in unsupervised learning scenarios.

3.3.6 Davies-Bouldin Index

The Davies-Bouldin Index (DBI) is a metric for evaluating clustering algorithms. It is especially useful as an internal evaluation scheme where the true cluster labels are not known. The index provides a measure of clustering evaluating both the separation between clusters and the compactness of the clusters themselves. The Davies-Bouldin Index is defined as the average similarity measure between each cluster and its most similar cluster, where similarity is a function that compares the distance between clusters with the size of the clusters themselves (Davies and Bouldin, 1979).

Lower index values indicate a better clustering arrangement, suggesting that clusters are both compact (low intra-cluster distance) and well-separated (high inter-cluster distance). High values indicate clusters that are either overlapping significantly or very dispersed.

3.3.7 Calinski-Harabasz Score

The Calinski-Harabasz Score, also known as the Variance Ratio Criterion, is a method for evaluating the quality of clustering in a dataset (Calinski and Harabasz, 1974). This score is an internal evaluation metric that assesses clustering quality based on the inter-cluster dispersion and intra-cluster cohesion. The metric is calculated using the ratio of the sum of between-cluster dispersion and within-cluster dispersion for all clusters.

A higher Calinski-Harabasz Score indicates that the clusters are dense and well-separated, which corresponds to a model with better-defined clusters. The score is designed to reach its maximum value when the clusters are compact and far from each other (Calinski and Harabasz, 1974).

3.3.8 Feature Importance

Feature importance in classification models involves identifying which features (variables) in a dataset have the most significant influence on the predictive accuracy of the model. This concept is crucial

for understanding, improving, and interpreting classification models, especially in domains where model transparency and understanding are essential.

There are several approaches to evaluating feature importance in classification tasks, varying by the type of classifier used. Coefficient magnitudes (linear models) in models such as Logistic Regression, the magnitude of the coefficients associated with each feature can indicate importance, assuming all features are on the same scale. In decision trees and ensemble methods like Random Forest s and gradient boosting machines use metrics like Gini impurity or entropy for classification tasks to evaluate the importance of each feature. These models provide a built-in mechanism for feature importance which is calculated by the amount a feature is used to make key splits in the trees.

3.3.9 Confusion Matrix

Though not a metric per se, the confusion matrix is another useful tool for understanding the performance of a classification model, providing insight into the types of errors made by the classifier. It shows the actual versus predicted classifications in a matrix format, highlighting true positives, true negatives, false positives, and false negatives. Figure 13 depicts a typical confusion matrix.

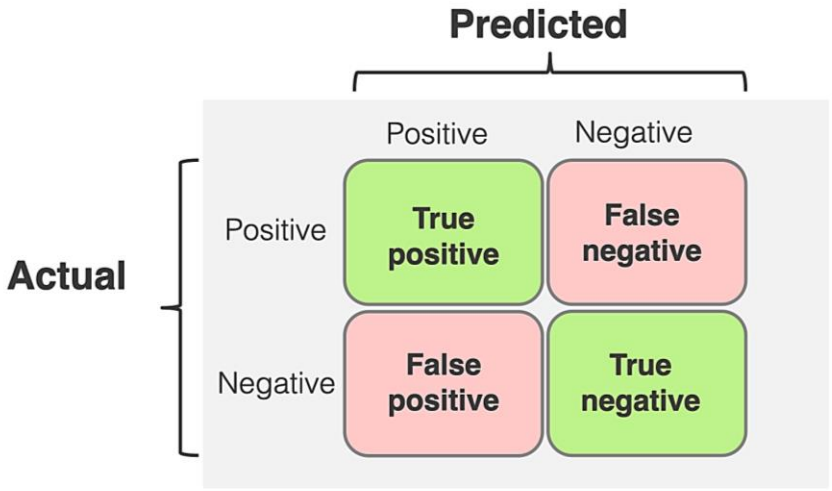

Figure 13: Typica confusion matrix configuration, depicting a 2-class scheme, positive-correct classifications running along the green diagonal (Evidently AI, 2024).

Each of these metrics provides different insights into the classifier's performance, and the choice of which to use depends on the specific objectives of the machine learning task and the nature of the data. For instance, in applications where false negatives carry a higher cost than false positives, recall might be emphasized over precision. Conversely, in situations where the cost of a false positive is high, precision could be more important. Balancing these metrics, often through the F1 score or by customizing a performance function specific to the application's needs, is key to developing effective machine learning models.

4. RESULTS AND PREDICTION EVALUATION

All methods results are prepared as matplotlib plots from method python scripts. Varying methods results, together, generate a more complete picture of machines learning's facies prediction (classification) performance.

4.1 Unsupervised Methods

Beginning with principal component analysis, the CSS database shows decreasing incremental variation as the number of components is increased. Figure 14 shows that more than 95% of all variances in the database can be accounted for in a 4-dimensional system. The plot reveals that with just two components, approximately 83% of the variance is explained.

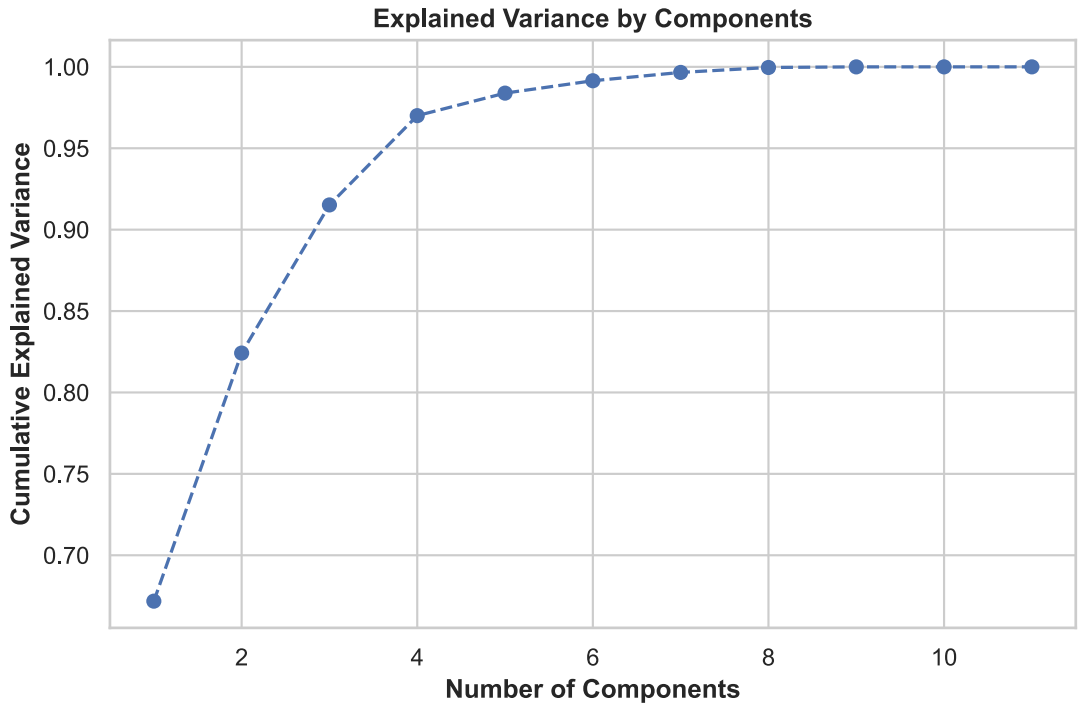

Figure 14: Explained Variance by Number of Components in PCA.

4.1.1 K-Means Clustering

As K-Means is an unsupervised technique, the clustering is performed without knowledge of the interpreted class. This means that K-Means clusters are not directly tied to the original facies interpretation and may not represent physically “real” facies. This makes the number of clusters being picked significant to the classification. In determining the number of clusters to assign to the dataset, it is necessary to plot Inertia, Silhouette Score, Davies-Bouldin Score, and Calinski-Harabasz Score. Figures 15 and 16 depict these metrics across varying numbers of clusters on the x-axis. Particularly, Figure 15 shows Inertia plotted next to Silhouette Score.

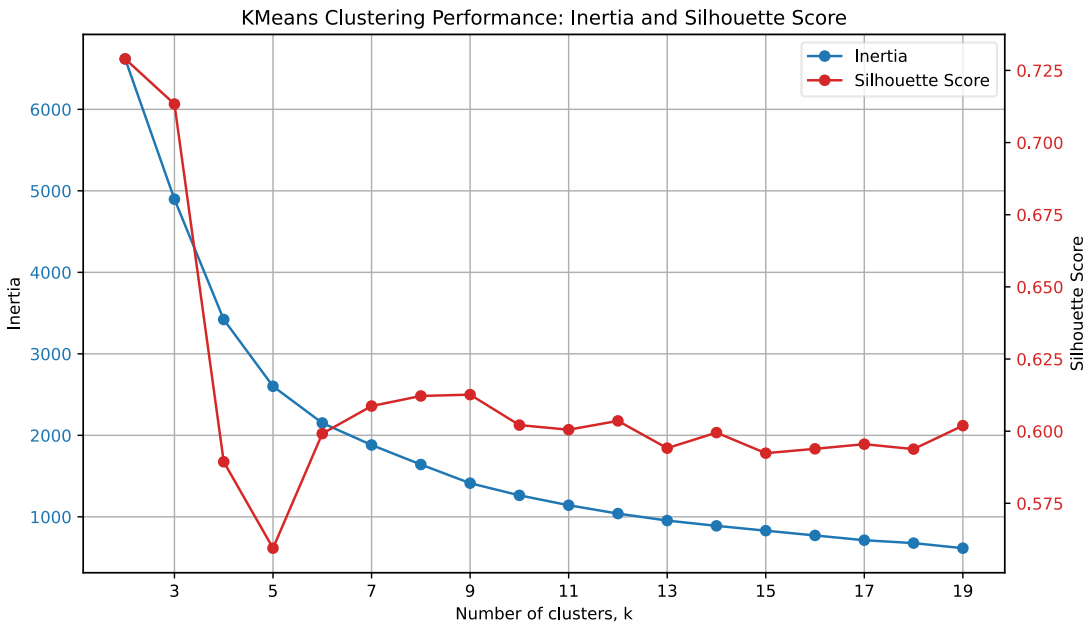

Figure 15: Evaluation of Cluster Quality Using Inertia and Silhouette Score.

For smaller numbers of clusters, both Inertia and Silhouette Score are high with a steep decreasing trajectory. As the number of clusters increases to five, the Silhouette Score finds a minimum below 0.575. The Inertia in this dataset for five clusters begins to show significant decrease in the slope of the line.

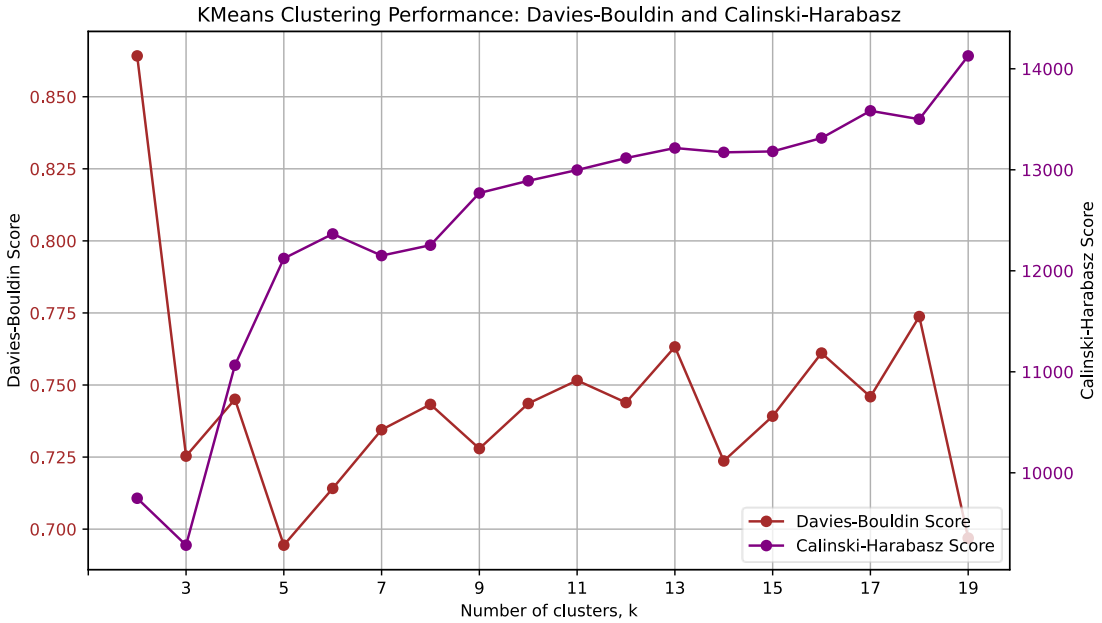

Figure 16: Comparison of Davies-Bouldin and Calinski-Harabasz Scores for Clustering Evaluation.

Similar, changing behavior is displayed as the number of clusters goes beyond five in Figure 16. Here the Davies-Bouldin Score is plotted next to the Calinski-Harabasz Score for varying numbers of clusters. The Davie-Bouldin Score begins remarkably high (0.85) before falling to 0.725 at three clusters and more-or-less stabilizing for increasing cluster count. The Calinski-Harabasz Score has the reverse character, as early values are low (~10000) but increase drastically beyond 3 clusters (>12000).

Using the four-class (4 facies) system, plotting all samples in a 2-D sense shows how clusters separate in feature space. Figures 17 and 18 show two 2-D sample projections (BedThickness vs. median grain size, max vs. min grain size. These results illustrate how K-Means handles distributions of manually collected (gradational and containing artifacts) geologic data. Sample clusters appear at notably “round” BedThickness values (1 m, 0.1 m, 0.01 m). In Figure 18, data samples cluster at grain size maximum values (1 mm, 0.1 mm, 0.01 mm) that are found on standard grain size cards as well as along the line gs_max = gs_min (indicating well sorted beds).

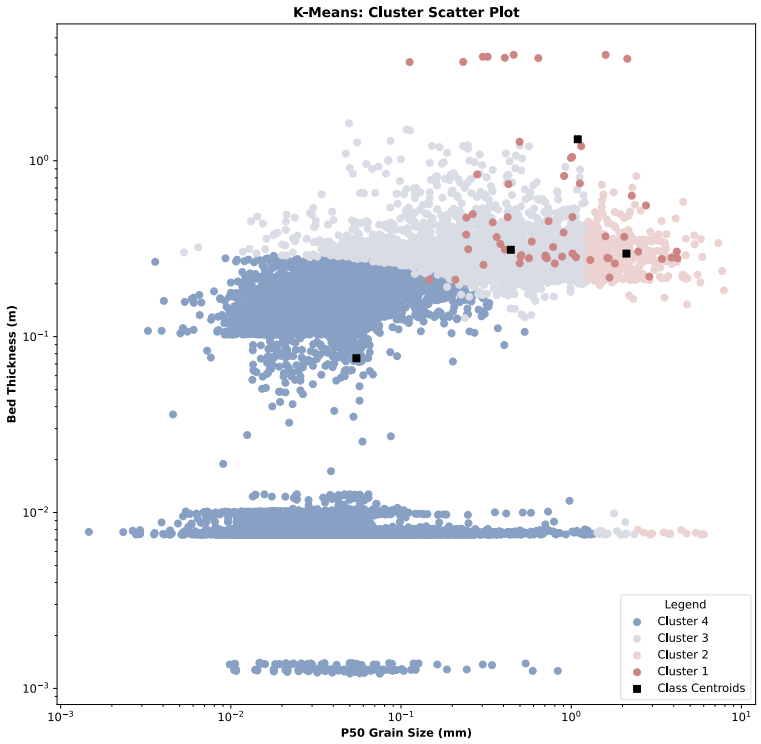

Figure 17: Scatter Plot of Clustered Data Points with Class Centroids; BedThickness vs. P50 Grain Size.

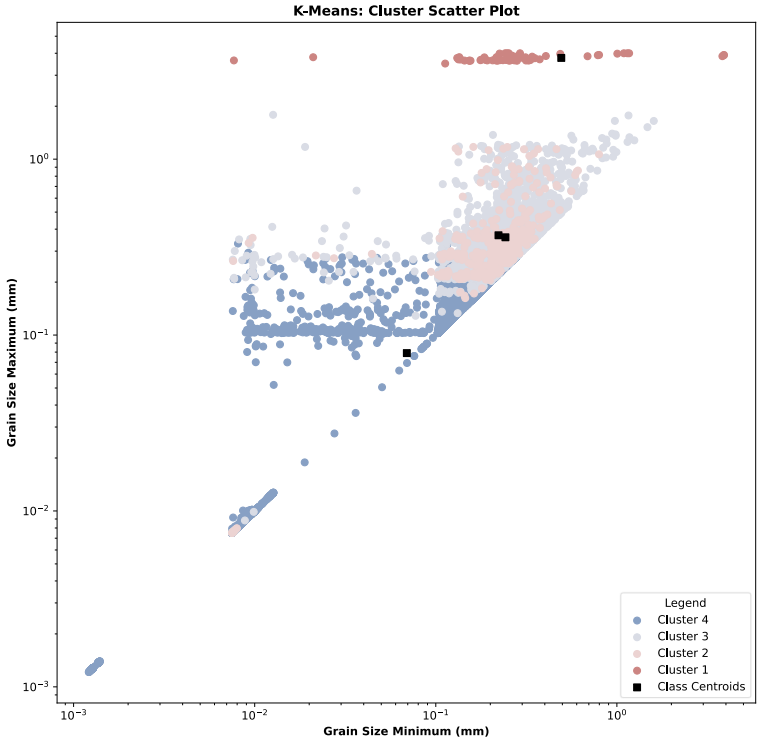

Figure 18: Scatter Plot of Clustered Data Points with Class Centroids; Grian Size Max vs. Min.

To generate typical performance metrics like accuracy for a K-Means classification, the target classes must be interpreted to fit the existing four class system of facies. This is performed once clusters are classified by reading class centroids from the K-Means algorithm and overprinting their equivalent facies from the facies system. This also allows for Figure 19 to compare the feature distributions between the existing "Training Data" and the equivalent K-Means facies giving insight into how these classes appear in this method.

Assuming a 1:1 K-Means cluster to facies interpretation, the total accuracy of the classifier was 48% in testing. Taking the total accuracy and normalizing it to the database feature BedThickness results in a score of 29%. Figure 19 shows some of the feature characteristics of that performance. Facies 1 shows some notable disagreement between green and blue. The median of the training data is lower, and the spread is narrower compared to the K-Means clusters, which have a higher median and wider spread. Facies 2 shows the opposite scenario. The training facies are broadly distributed with high medians, but the K-Means cluster shows a tighter distribution and even higher medians. This is true for both Facies 1 and 2. On the other hand, Facies 4 displays a strong correlation between Training Class and K-Means Cluster.

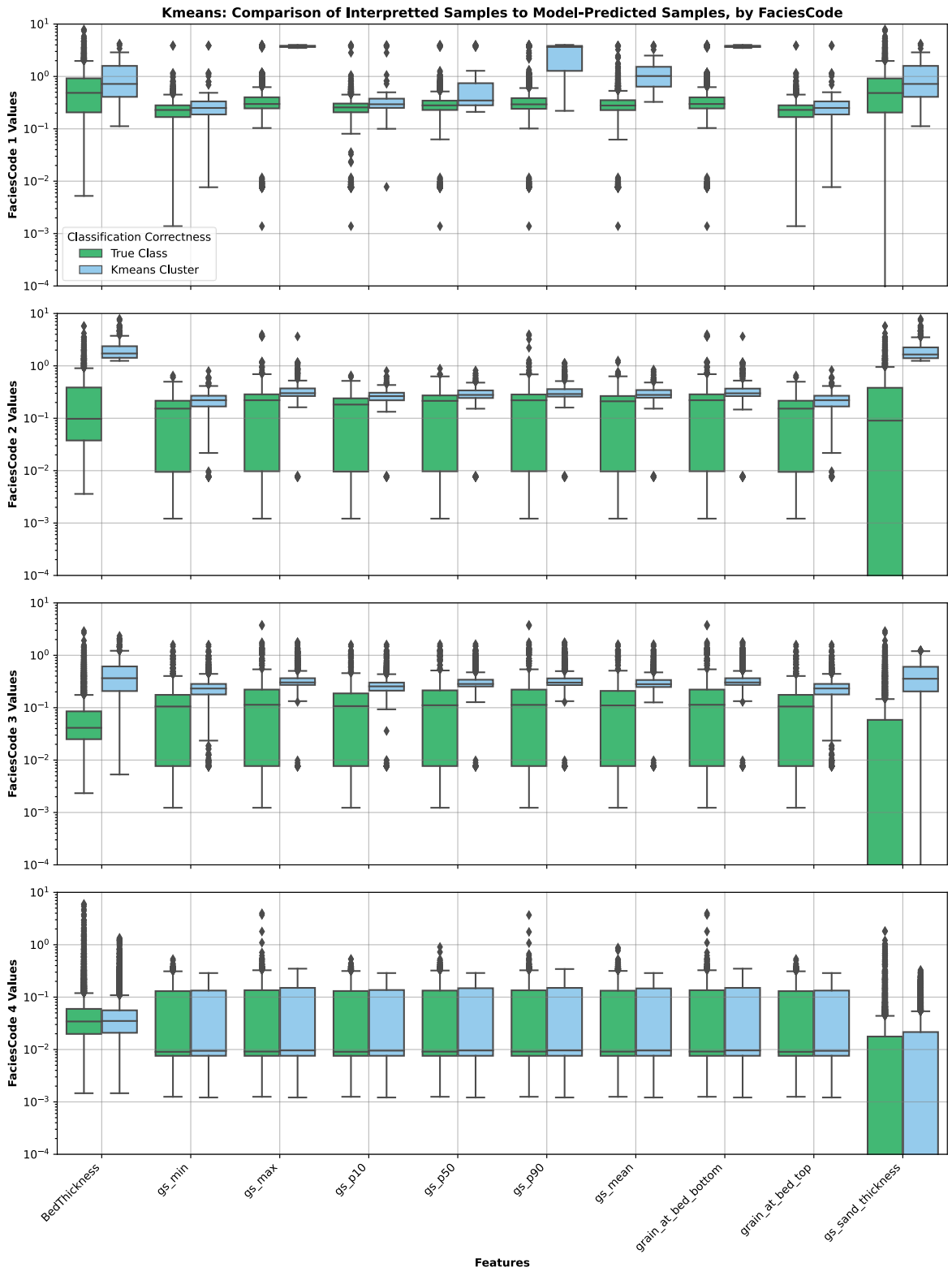

Figure 19: Comparison of Interpreted Class and K-Means Cluster Classification Across All Features, Differentiated by FaciesCode.

4.2 Supervised Methods

In addition to unsupervised methods, his study applies supervised machine learning techniques to predict facies. Methods including Least Squares Regression, Logistic Regression, Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Random Forest, and Neural Networks, are applied to classify geologic facies. This section presents the predictive accuracy, confusion matrices, and feature importance analyses for each model, highlighting the efficacy of the supervised learning approach in delineating facies distribution.

4.2.1 Least Squares Solution

The Least Squares solution (binary classification) requires four scenarios to account for the four facies system. The four scenarios are the permutations of a particular facies (e.g. F1, F2, F3, F4) classified against the sum of all other facies.

Facies 1: highamalsand

The first scenario is Facies 1 classified against all other classes. This produces a total accuracy score of 89% and a BedThickness normalized accuracy of 71%. Figure 20 shows a confusion matrix for this facies system.

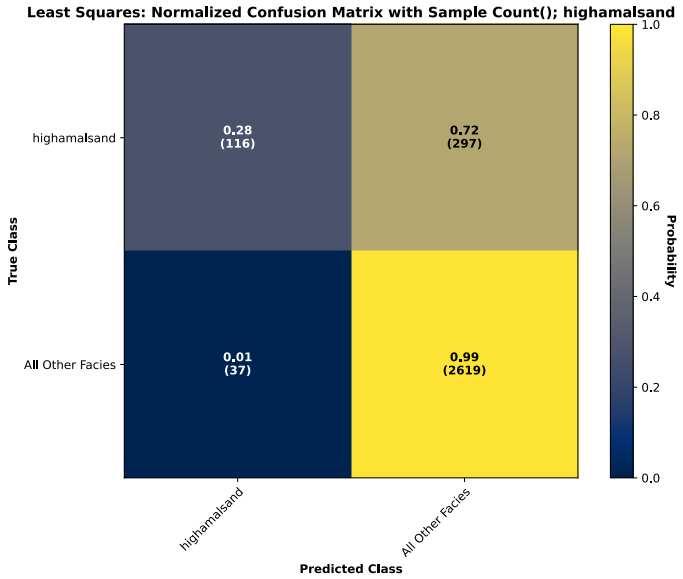

Figure 20: Normalized Confusion Matrix for Binary Classification of *highamalsand* vs. All Other Classes.

The top-left square (Fig. 20) represents the True Positive (TP) rate for the class *highamalsand*. It shows a value of 0.28, indicating that 28% of all the instances that truly belong to *highamalsand* were correctly identified as such by the classifier. The raw count of these instances is 116. The top-right square represents the False Negative (FN) rate for the class *highamalsand*. It shows a value of 0.72, suggesting that 72% of the instances that actually belong to *highamalsand* were incorrectly classified as All Other Facies. The raw count here is 297. The bottom-left square represents the False Positive (FP) rate for the class *highamalsand*, reflecting how often other classes were incorrectly identified as *highamalsand*. It has a value of 0.01, which means that 1% of the instances that were not *highamalsand* were incorrectly labeled as being in this class. The raw count for these instances is thirty-seven. The bottom-right square represents the True Negative (TN) rate for the class *highamalsand*. It shows a value of 0.99, indicating that 99% of the instances that do not belong to *highamalsand* were correctly classified as All Other Facies. The raw count here is 2619. Overall, matrix indicates that the classifier is highly effective at correctly identifying instances of All Other Facies (high TN rate) but struggles with correctly classifying *highamalsand* instances (low TP rate).

Plotting BedThickness normalized accuracy for varying BedThickness allows for another perspective on performance. Figure 21 depicts this metric.

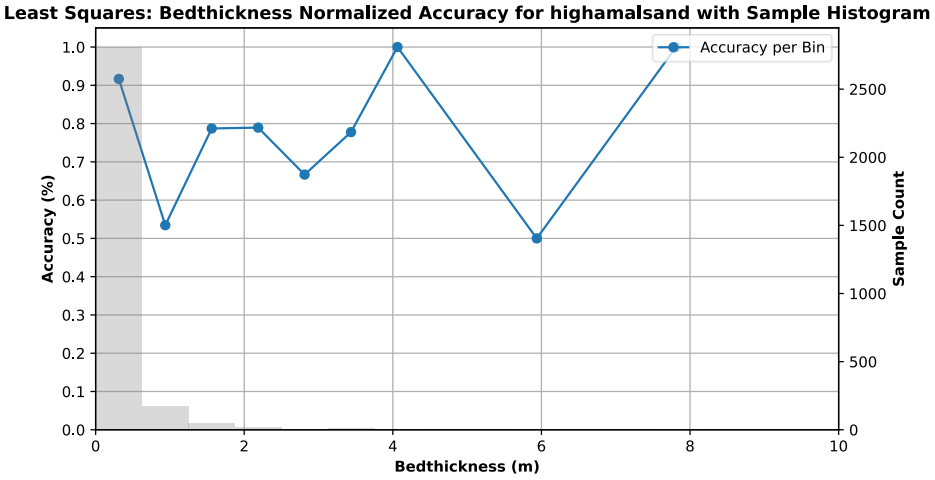

Figure 21: Accuracy Variation Across Bins Based on BedThickness Feature Normalization; *highamalsand.*

The graph shows significant variability in accuracy across all bins of BedThickness. This is plotted over a sample count bar graph to emphasize bins with significant numbers of samples. It begins with a high accuracy (~90%) in the first bin, dips in the second (~50%), and maintains a stable accuracy through the next few bins (~80%). There is a notable drop in accuracy at the penultimate bin (~50%) before it rises sharply to reach the highest accuracy in the last bin (100%).

Facies 2: semiamal

The second scenario is Facies 2 classified against all other classes. This produces a total accuracy score of 87% and a BedThickness normalized accuracy of 79%. Figure 22 shows a confusion matrix for this facies system.

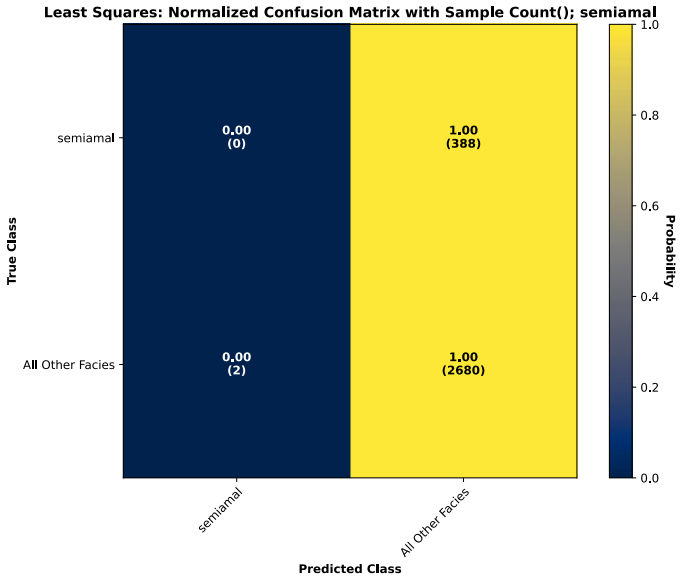

Figure 22: Normalized Confusion Matrix for Binary Classification of *semiamal* vs. All Other Classes.

This model's True Negative score (1.00) indicates that it is phenomenally successful in predicting all other facies, in conjunction with not predicting any True Positives and almost no False Negatives. Of all 3070 classifications made, only two were the *semiamal* class. When the model predicted not Facies 2 (i.e., one of 1, 3, or 4), it misclassified samples with a true class of *semiamal* as All Other Facies 13% of the time. This performance is notably distinct from that of the *highamalsand* scenario.

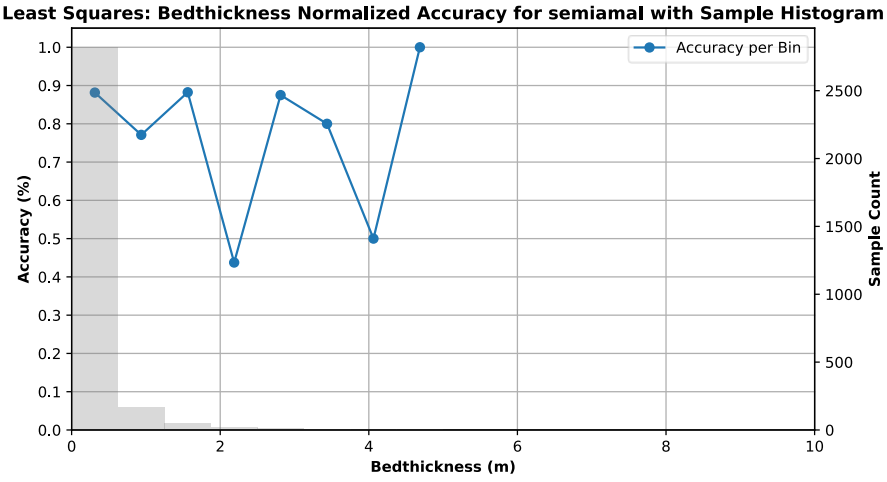

Figure 23: Accuracy Variation Across Bins Based on BedThickness Feature Normalization; *semiamal.*

Figure 23 reflects how the classification accuracy for distinguishing Facies 2 from other classes changes as a function of the BedThickness. Similar in character to *highamalsand*, *semiamal* shows varying levels of predictability depending on its BedThickness value. In the "thinnest" bin (lowest BedThickness), the model shows robust performance (~85%) that persists until the bin at roughly 2 meters BedThickness. Here the sample count bars show minimal samples and performance becomes more variable in accuracy (40% to 100%).

Facies 3: nonamalsand

The third scenario is Facies 3 classified against all other classes. This produces a total accuracy score of 69% and a BedThickness normalized accuracy of 85%. Figure 24 shows a confusion matrix for this facies system.

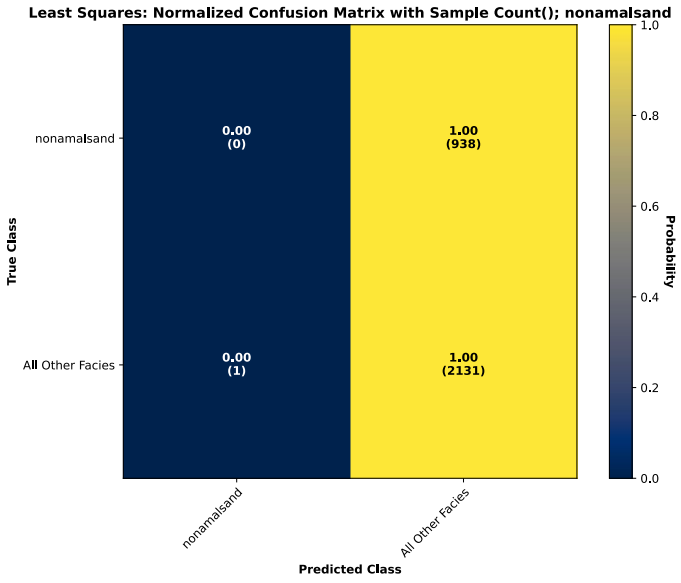

Figure 24: Normalized Confusion Matrix for Binary Classification of *nonamalsand* vs. All Other Classes.

In this confusion matrix for Facies 3 versus all other classes, like the previous facies system (Facies 2 vs. All Other Facies), almost all classifications are made for the All Other Facies class with only one

sample being classified as *nonamalsand*. This gives exceedingly high False Positive and True Negative scores and incredibly low True Positive and False Negative scores. This confusion matrix, like the previous ones, is useful for understanding the model's predictive performance regarding a specific class (Facies 3 in this case) relative to all other classes. This result is distinct from the performance observed in the *highamalsand* scenario.

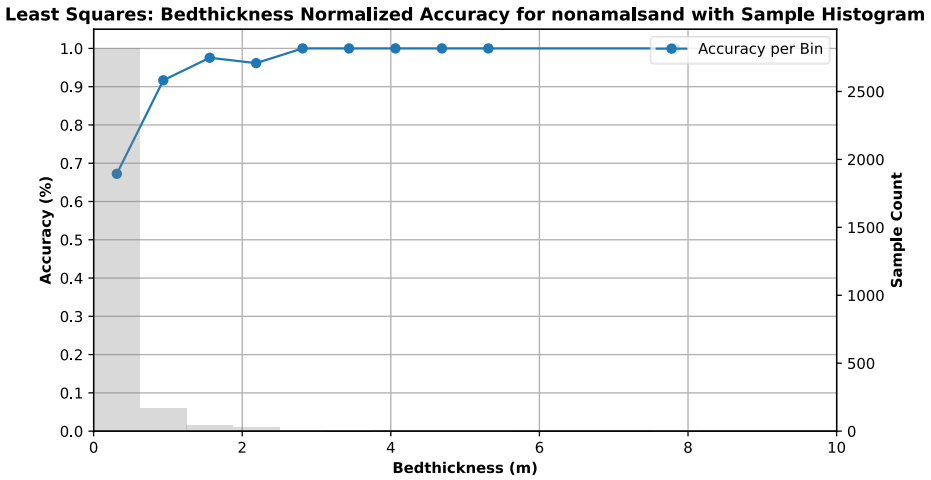

Figure 25: Accuracy Variation Across Bins Based on BedThickness Feature Normalization; *nonamalsand.*

In Figure 25, depicting the accuracy per bin for the classification of Facies 3 versus all other classes, we observe a stable accuracy trend across the bins. After the initial low performance (~65%), the accuracy remains stable across the subsequent bins (95%). This analysis demonstrates that the model is effective in distinguishing All Other Classes from Facies 3, with high and stable accuracy across most bins. The consistency across later bins coincides with a lack of meaningful sample count beyond two meters of BedThickness.

Facies 4: nonamalsilt

The last scenario is Facies 4 classified against all other classes. This produces a total accuracy score of 64% and a BedThickness normalized accuracy of 91%. Figure 26 shows a confusion matrix for this facies system.

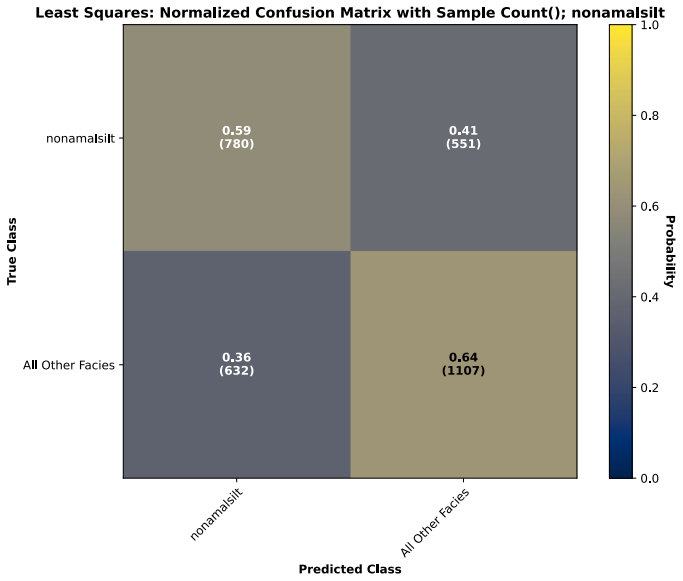

Figure 26: Normalized Confusion Matrix for Binary Classification of *nonamalsilt* vs. All Other Classes.

This confusion matrix illustrates the performance of a classification with Facies 4 and all other classes. The *nonamalsilt* scenario shows the most balanced performance of the four Least Squares Solution scenarios. Both True Positive (59%) and True Negative (64%) Classifications dominate and show robust performance. The top-right box represents false negatives for Facies 4. It indicates that 41% of the cases that were actually Facies 4 were incorrectly predicted as not Facies 4, with 551 instances misclassified. False positives for Facies 4 show that 36% of the predictions made as Facies 4 were incorrect, actually belonging to other classes, with 632 instances incorrectly labeled. At 59%, the model has a moderate sensitivity to Facies 4 when it is present.

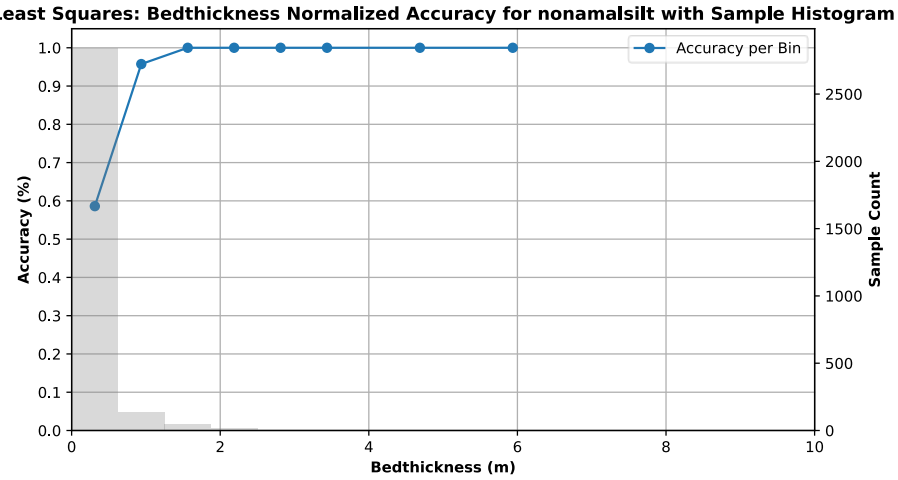

Figure 27: Accuracy Variation Across Bins Based on BedThickness Feature Normalization; *nonamalsilt.*

Figure 27 illustrates accuracy per bin for the classification of Facies 4 (*nonamalsilt*) versus all other classes demonstrates a pattern of increasing performance with increasing BedThickness value. The first bin, containing the majority of samples, displays the lowest performance (~60%). Accuracy quickly rises in the second (~95%) and third bins (to 100%). After the initial increase, accuracy stabilizes and maintains a consistently high level across the remaining bins. This plot suggests that the classification model for Facies 4 vs. All Other Facies is robust, with good performance across most of the dataset bins.

4.2.2 Logistic Regression

Logistic Regression Classification does not require the same scenarios as the Least Squares Solution above. This allows for direct classification in the four facies system, because output values are discreet instead of continuous. In total the model was able to generate a classification accuracy of 55%. Normalized to BedThickness, this accuracy increases to 64%.

The Logistic Regression model also allows for some degree of feature analysis through its adjustment of feature coefficients. In "Applied Logistic Regression", the coefficients in Logistic Regression are defined as the weights assigned to the features in the model (Hosmer Jr. et al., 2013). These coefficients

determine the contribution of each feature to the model's prediction. The book explains how these coefficients are used to model the log odds of the dependent variable as a linear combination of the independent variables. Each coefficient represents the change in the log odds ratio of the dependent variable for a one-unit change in the corresponding independent variable, holding all other variables constant.

The interpretation of these coefficients involves examining the sign and magnitude of each coefficient, which indicates the direction and strength of the relationship between each feature and the likelihood of the outcome variable. Positive coefficients increase the log odds of the outcome, indicating a positive association with the likelihood of the outcome, while negative coefficients decrease the log odds, indicating a negative association (Hosmer Jr. et al., 2013).

Figure 28 uses the coefficients of the trained Logistic Regression model to display the relationships of each feature to each facies. Notable characteristics of this data set can be broken up by facies. Facies 1 displays almost only positive associations with the magnitude of most features. This is distinct from Facies 2 and 3 which show a mixture of positive and negatively associated features. Notably, both classes have a positive relationship to gs_sand_thickness and gs_p90. Most other features show a negative association for these classes. Conversely, Facies 4 shows the strongest relationships of all classes with a strong negative relationship to gs_sand_thickness. Other important associations appear to be BedThickness (Positive) and gs_mean/gs_p90/gs_p50 (Negative).

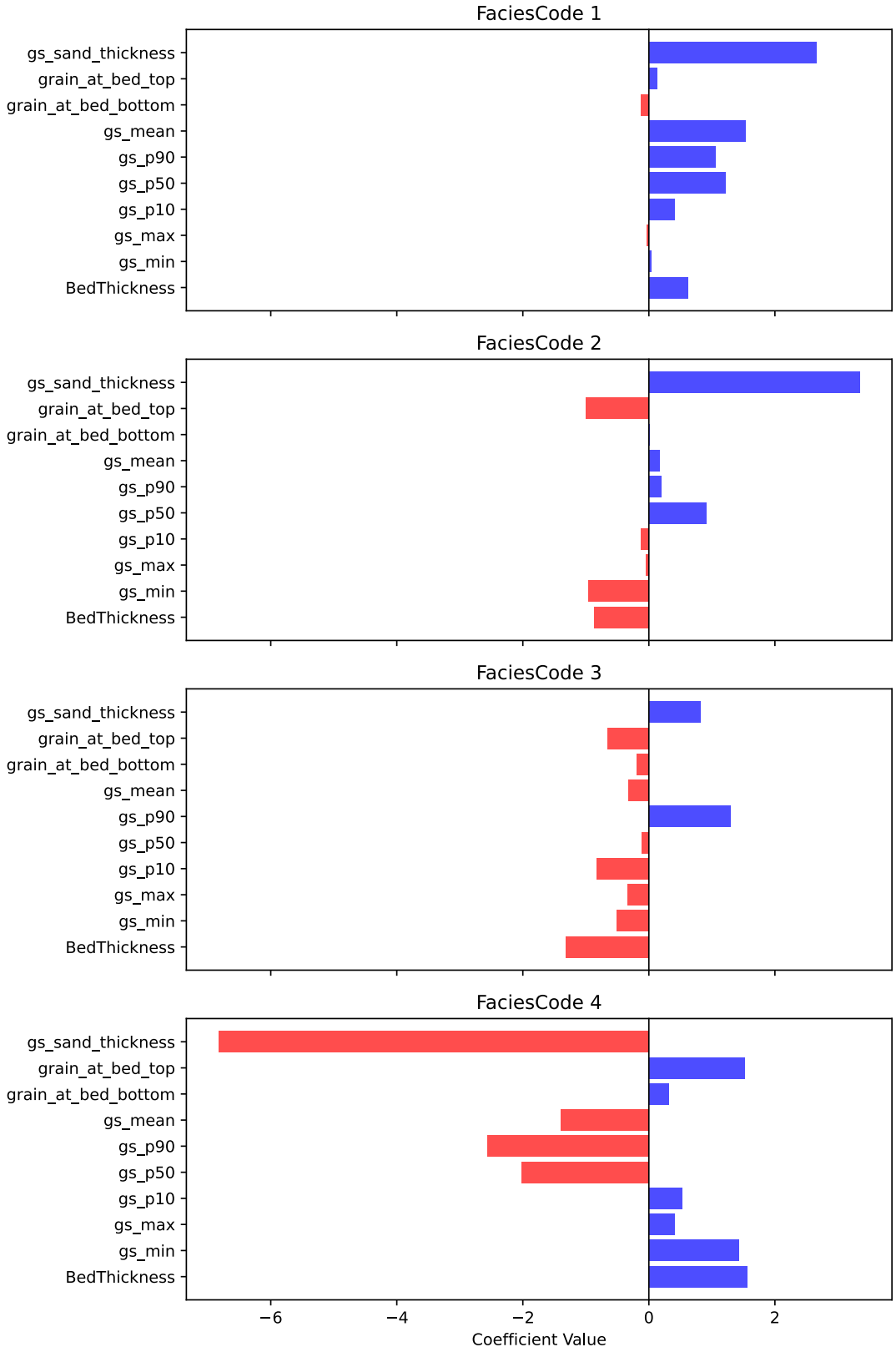

Figure 28: Logistic Regression Feature Coefficients per FaciesCode.

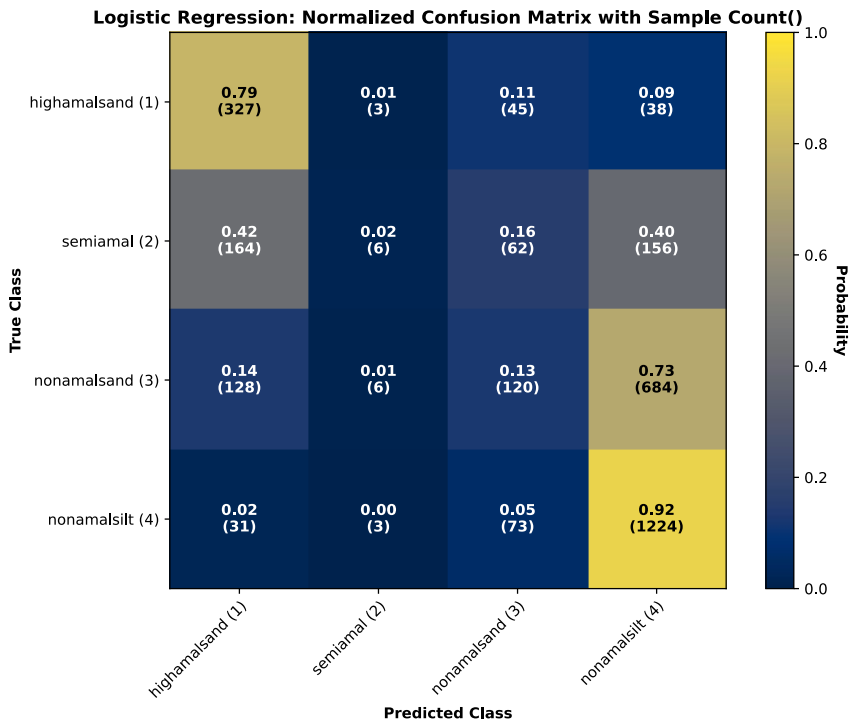

Figure 29: Normalized Confusion Matrix for Logistic Regression Model.

Returning to the confusion matrix format, Figure 29 displays performance results for the Logistic Regression Model. Observing the diagonal values, which represent correct classifications, the model excels particularly with the *nonamalsilt* class, showing a high accuracy (92%) with 1224 samples correctly classified. This is followed by the *highamalsand* class, with an accuracy of 79% and 327 correct predictions. The accuracy dips for the *nonamalsand* class at 13% with 120 correct instances, and it is notably lower for the *semiamal* class at 2%, where six samples are correctly identified. Misclassifications are evident in *semiamal*, and with *highamalsand* at 42% (164 samples) and *nonamalsilt* at 40% (156). *Nonamalsand* is also commonly misclassified as *nonamalsilt* at 73% (684 samples).

Another important feature to note is the relative lack of predictions of the *semiamal* class. This class is nearly absent; most samples having a True Class of Facies 2 are mostly classified as Facies 3 and 4.

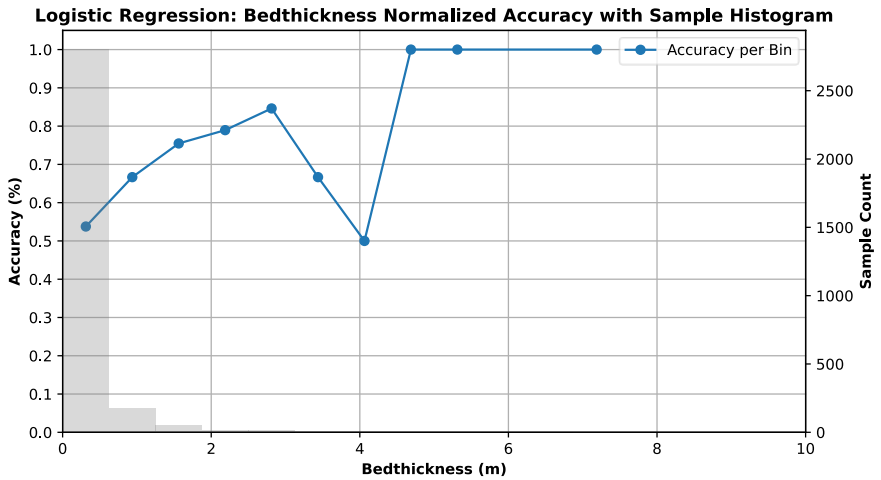

Figure 30: BedThickness Normalized Accuracy for Logistic Regression Model.

The normalized accuracy plot for Logistic Regression model provides insights into how the model's performance varies by BedThickness (Fig. 30). Performance begins low (~55%), displays an initial rise, a significant dip at roughly 4 meters BedThickness, and subsequent recovery to a stable level of high accuracy (100%). Here is a breakdown of the model's behavior as depicted in the graph:

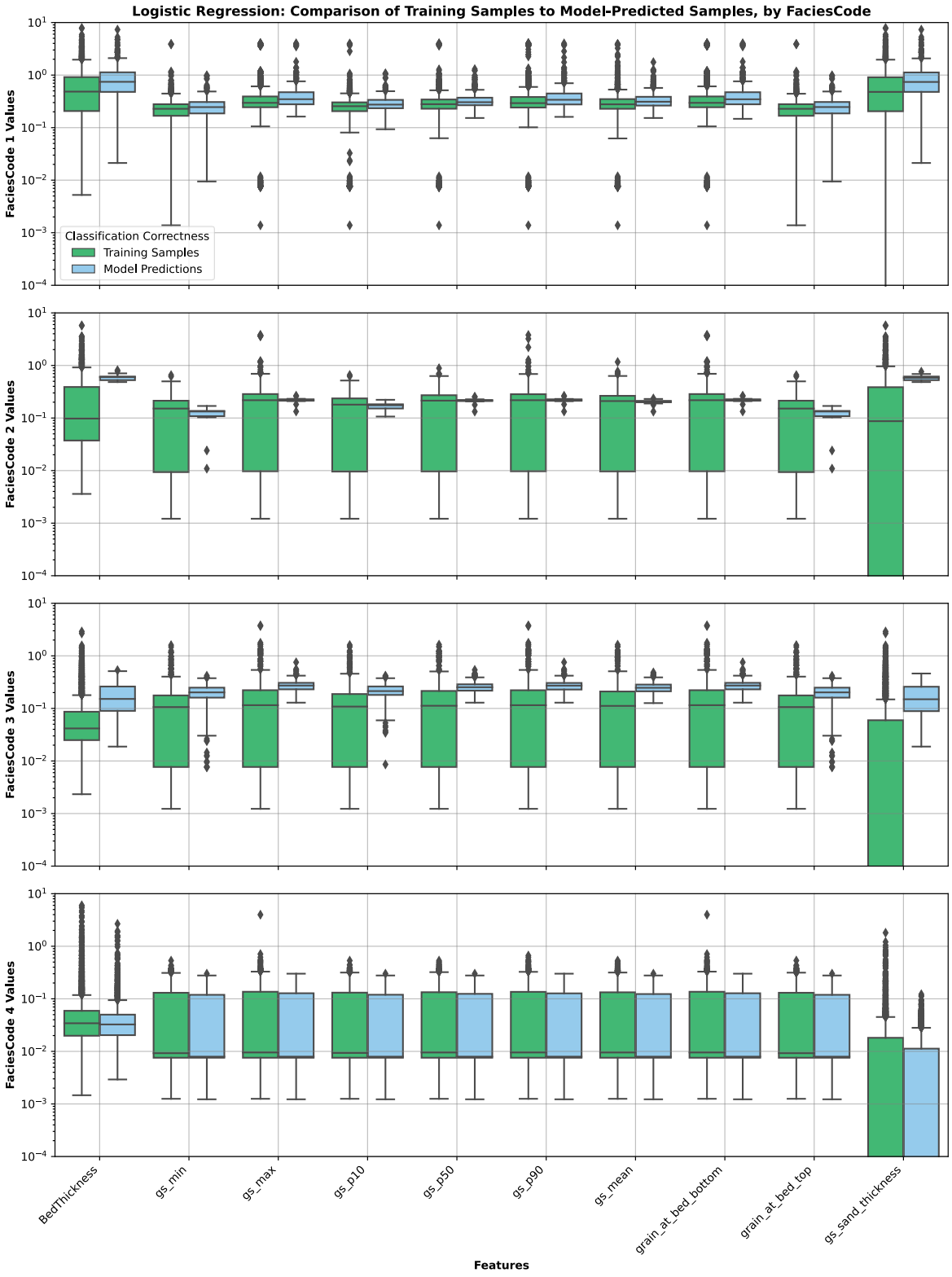

Figure 31: Comparison of Interpreted Class and Logistic Regression Classification Across All Features, Differentiated by FaciesCode.

Figure 31 depicts feature distribution matches between the training data (green) and the predicted data (blue) across the four classes of facies. The first and last facies show the best agreement between the True and Predicted Classes, with overlapping interquartile-range (IQR) and similar medians. The two middle classes (Facies 2 and 3) show significant disagreement. Both have broad IQR in the training data but the samples the Logistic Regression model has classified are a much tighter spread, all on the thick and coarse (high magnitude) end of the distribution.

Overall, features like BedThickness, and gs_sand_thickness tend to show less alignment between the training and predicted data for different facies. Features related to grain size extremes and specific percentiles (gs_min, gs_max, gs_p10, gs_p50, gs_p90) and grain size at specific bed layers (gs_at_bed_bottom and gs_at_bed_top) show more alignment of training and predicted data.

4.2.3 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis classification yielded a total classification accuracy of 46% and a BedThickness normalized accuracy of 60%. This is less than Logistic Regression but is more balanced across the four classes, notably having 3 of 4 classes likely to be classified correctly. Figure 32 is the confusion matrix of this model's performance (notice the bright tone along the diagonal).

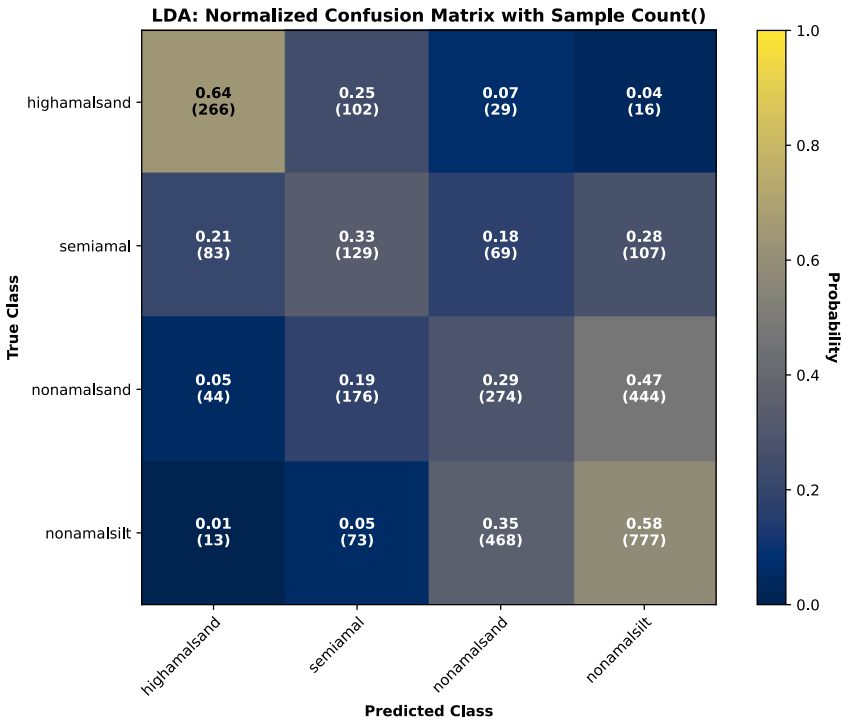

Figure 32: Normalized Confusion Matrix for Linear Discriminant Analysis Model.

The model correctly identifies *highamalsand* best with a probability of 64%, with 266 samples accurately classified. For *semiamal*, the model shows an accuracy of 33%, correctly classifying 129 out of the total samples. *Nonamalsand* class has a correct classification rate of 29%, with 274 samples correctly identified. The performance here is the lowest among all classes. The model performs second best with the *nonamalsilt* class, achieving a high accuracy of 58% with 777 samples correctly classified, indicating a strong capability to distinguish this class from others.

The model frequently misclassifies samples from the *semiamal* class as *nonamalsilt* (28% probability with 107 samples) and *highamalsand* (21% probability with eighty-three samples). There are also significant misclassifications of *nonamalsand* as *nonamalsilt*. This is the same pattern of misclassification as the Logistic Regression model. Misclassifications between *highamalsand* and *semiamal* are also significant, with a probability of 25% (102 samples) classified as *semiamal*. Another noteworthy error is the misclassification of *nonamalsilt* as *nonamalsand* with a high probability of 35% and

468 samples. Overall, the LDA model shows strengths in identifying the *highamalsand* and *nonamalsilt* classes but struggles between *semiamal* and *nonamalsand*.

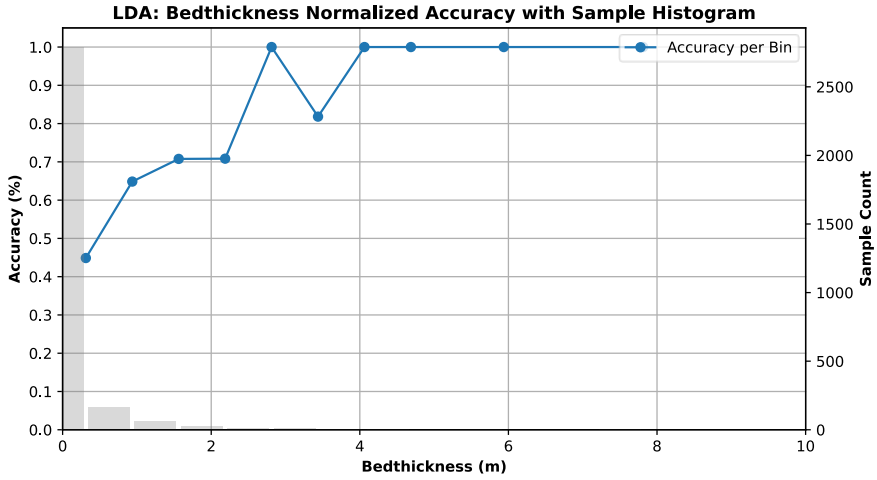

Figure 33: BedThickness Normalized Accuracy for Linear Discriminant Analysis Model.

Figure 33 displays the performance of LDA model over varying BedThickness. Low BedThickness values show steady rise in accuracy (~45% to 70%) from the first bin onwards. There is a peak in accuracy at 3 meters BedThickness as performance reaches 100%. Following this peak there is a notable dip (down to 80%) in accuracy. After the drop, the accuracy recovers to 100%.

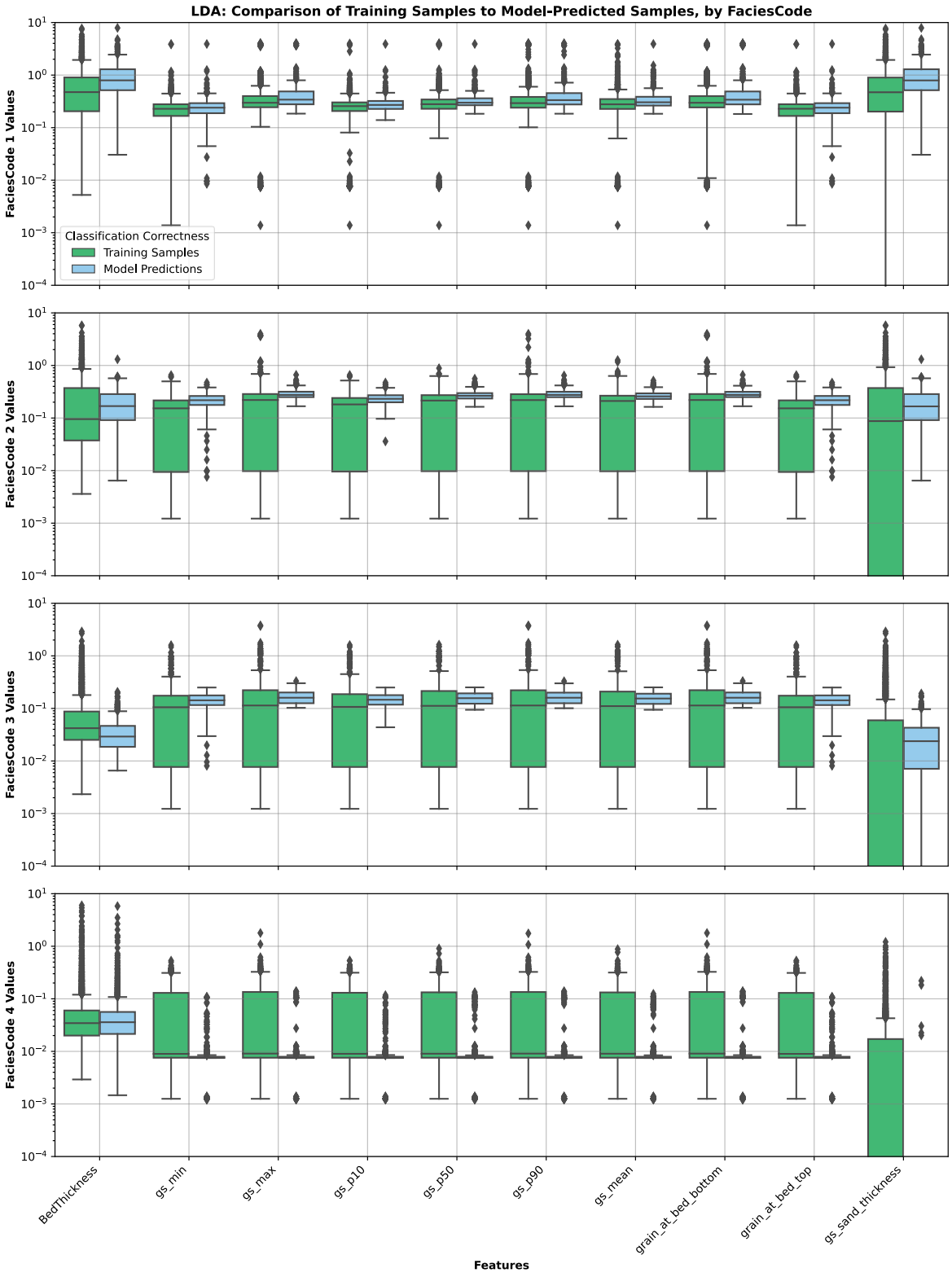

Figure 34: Comparison of Interpreted Class and Linear Discriminant Analysis Classification Across All Features, Differentiated by FaciesCode.

Figure 34 offers a comparison of the feature distributions across the four facies classes. In several classes and features, the blue box plots (LDA model predictions) closely align with the green box plots (training data), particularly for features like BedThickness and gs_sand_thickness. These plots exhibit similar medians, interquartile ranges, and outliers. Features such as gs_min, gs_max, and the percentile features (gs_p10, gs_p50, gs_p90) show more variability between the predicted and training data. For example, the blue plots sometimes have narrower distributions or shifted median values compared to the green plots, indicating discrepancies in how the model captures extreme values or central tendencies. In certain cases, there is a noticeable shift in the median between the training and predicted data (for instance, in gs_max and gs_mean across some classes). There are also varied numbers of outliers in the predicted data compared to the training data, but this is expected and related to the number of samples in the training set (12278) vs. the testing set (3070).

4.2.4 Quadratic Discriminant Analysis (QDA)

Quadratic Discriminant Analysis classification yielded a total classification accuracy of 44% and a BedThickness normalized accuracy of 45%. This is worse than Linear Discriminant Analysis and on-par with the performance of the K-Means clustering method. Figure 35 depicts the confusion matrix for this method.

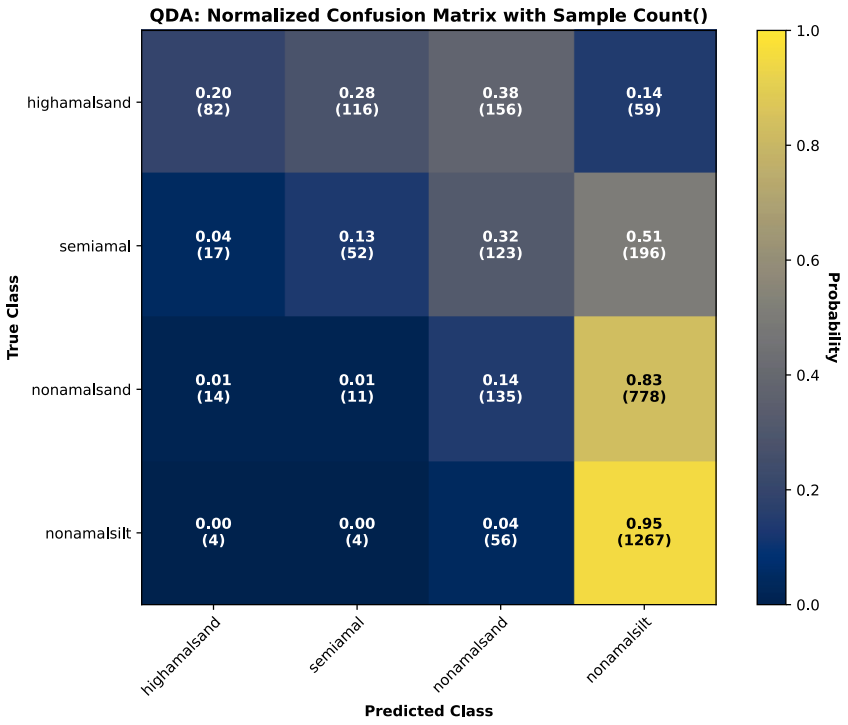

Figure 35: Normalized Confusion Matrix for Quadratic Discriminant Analysis Model.

The confusion matrix presented for the QDA model (Fig. 35) reveals both strengths and weaknesses in classifying the four facies. The *highamalsand* class achieves a correct classification probability of 20% with eighty-two samples, indicating a low effectiveness in identifying this class accurately. *Semiamal* accuracy worsens slightly at 13% with fifty-two samples correctly classified. The model's performance for *nonamalsand* is somewhat better at 14% correct classification probability with 135 samples. The model excels in classifying *nonamalsilt*, with an exceedingly high accuracy of 95%, correctly identifying 1267 samples. This indicates a robust capability to distinguish this class from the others.

There is a notable misclassification of *highamalsand* samples as *nonamalsand* (38% probability with 156 samples) and as *semiamal* (28% probability with 116 samples). These errors highlight a confusion in the model in distinguishing *highamalsand* from other classes. A considerable number of *semiamal* (51% probability with 196 samples) and *nonamalsand* (83% probability with 778 samples) samples are

misclassified as *nonamalsilt*. Misclassifications are also evident in lower probabilities such as *highamalsand* as *nonamalsand* (38% probability with 156 samples).

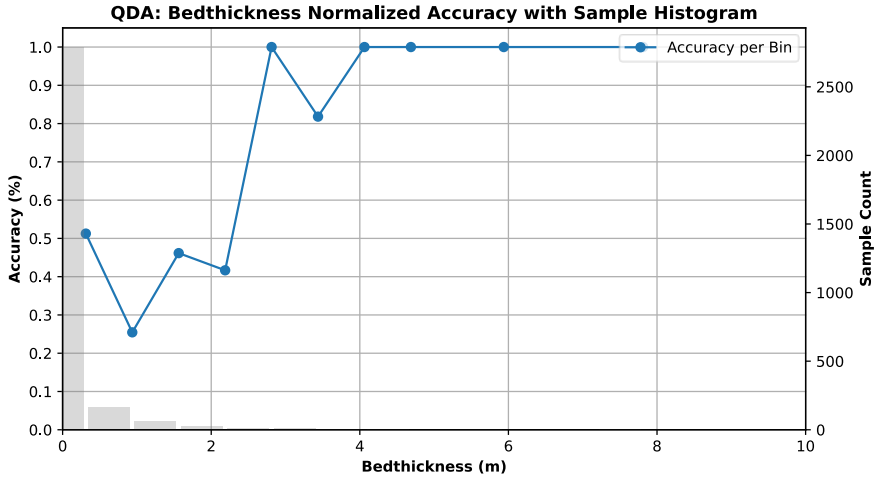

Figure 36: BedThickness Normalized Accuracy for Quadratic Discriminant Analysis Model.

Figure 36 illustrates the performance of a Quadratic Discriminant Analysis (QDA) model over various bins, highlighting the model's accuracy fluctuations. The plot begins with a relatively low accuracy (~25% to 50%). The early points show considerable variability, with accuracy initially dipping before rising sharply. After the initial variability, there is a sharp increase in accuracy (100%). However, this peak is followed by a significant drop (to 80%) and recovery, where accuracy increases and then stabilizes. This pattern indicates that the QDA model performs well for samples with higher values of BedThickness.

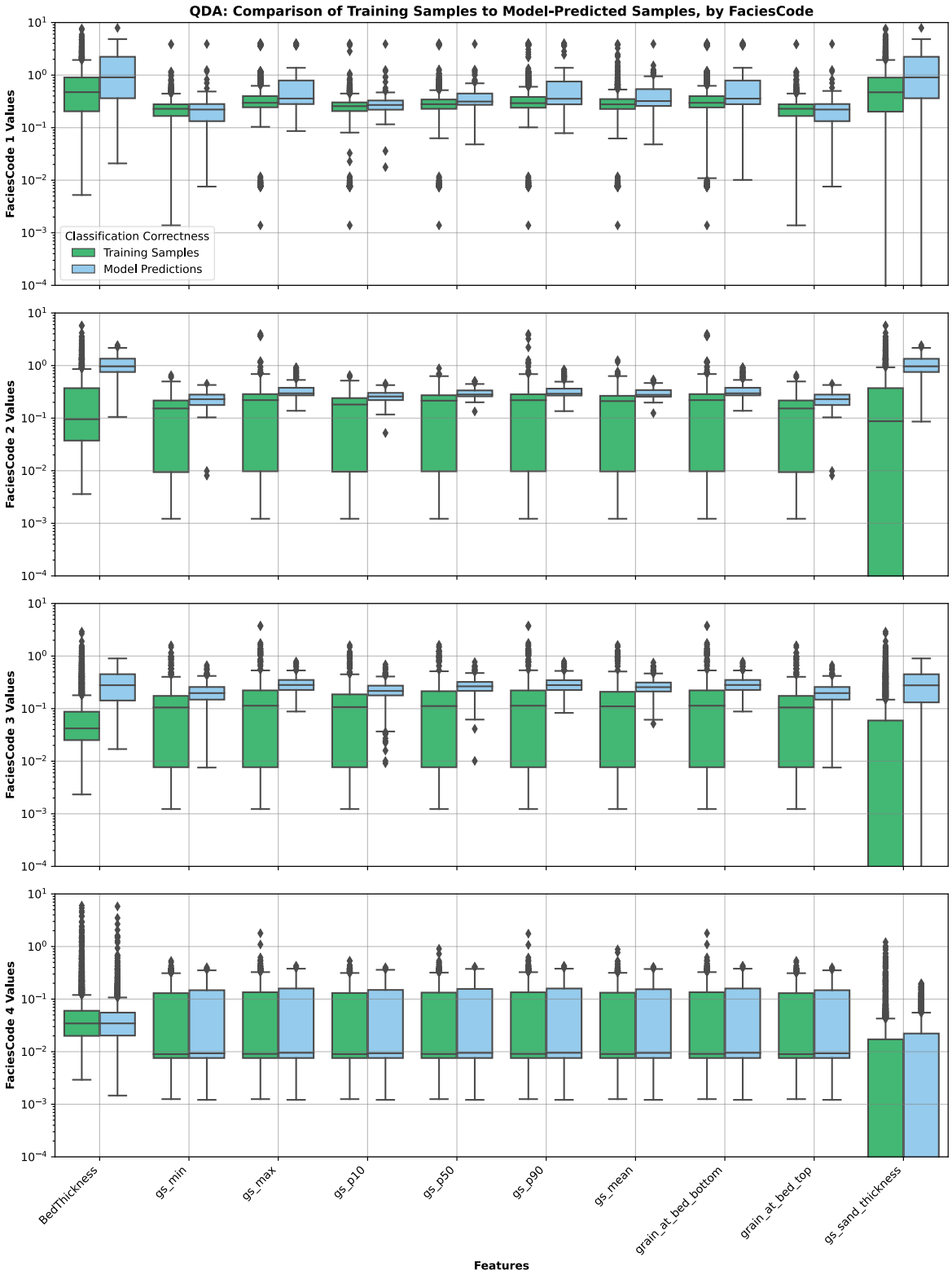

Figure 37: Comparison of Interpreted Class and Quadratic Discriminant Analysis Classification Across All Features, Differentiated by FaciesCode.

Figure 37 analyzes the facies for the QDA model, comparing feature distributions in the training data (green box plots) and the predictions (blue box plots). For certain features, such as gs_min and gs_at_bed_top, the blue box plots (QDA predictions) align closely with the green box plots (training data), both in terms of median and interquartile ranges. Other features like gs_max, and the percentile-based measures (gs_p10, gs_p50, gs_p90) show more noticeable discrepancies. These include differences in median, range, or the presence of outliers. Differences in the spread (width of the box plots) and the range (whiskers) are particularly visible in features like BedThickness and gs_sand_thickness for Facies 1, 2, and 3.

The alignment between training and predicted distributions varies across different facies classes. In summary, the Quadratic Discriminant Analysis model demonstrates variable performance across distinctive features and classes. While it captures some feature distributions well, others show significant deviations.

4.2.5 Random Forest

The Random Forest model produced a total accuracy score of 58% and a BedThickness normalized accuracy of 64%. Before training the model, it is necessary to select hyperparameters including the number of trees. Figure 38 shows how total classification accuracy varies with tree number. Initial models (utilizing few trees) provide poor performance (50%), but accuracy increases rapidly to a plateau at roughly 55%. The plot shows values above roughly ten trees provide the best performance so forty is used in this analysis.

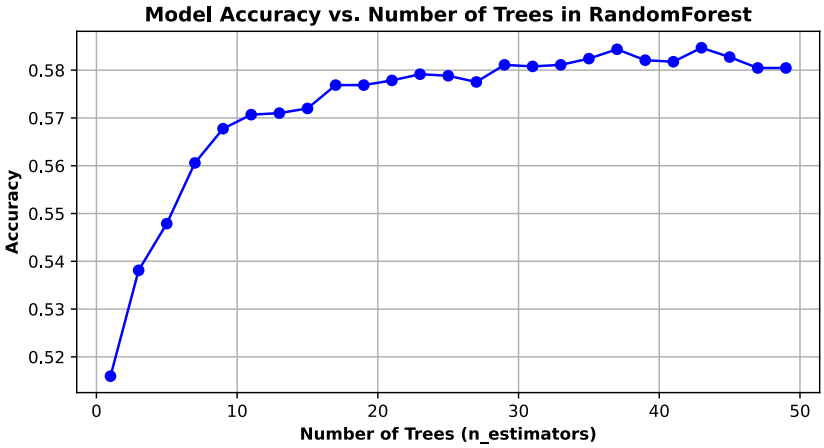

Figure 38: Random Forest Accuracy for Varying Number of Trees in the Model.

Random Forest also provides a way to understand the utility of each statistic in the classification. In Figure 39 (importance plot derived from a Random Forest model), the performance of the features can be described best in three groups.

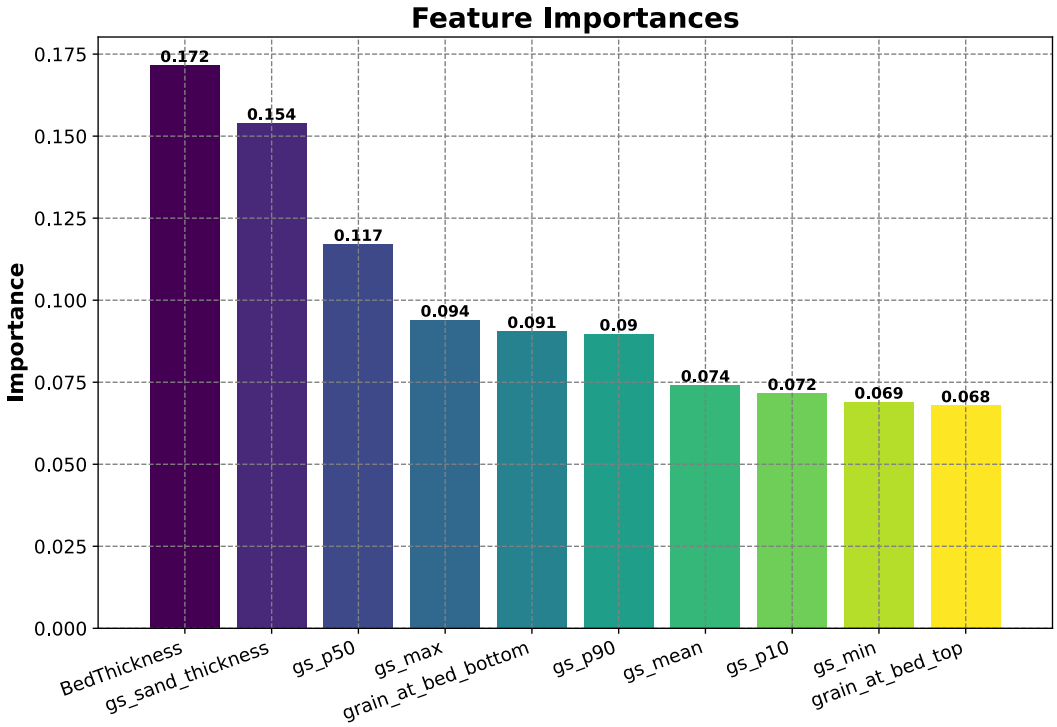

Figure 39: Random Forest Feature Importance.

The first three features (BedThickness, gs_sand_thickness, gs_p50) dominate importance. BedThickness has the highest importance score in this group and overall, at 0.172, indicating it is the most significant predictor among the features plotted. This is a similar result to the coefficient analysis used in the Logistic Regression model. The gs_sand_thickness feature follows with a score of 0.154, also showing high relevance in the model. Then gs_p50 has a score of 0.117, somewhat lower than the first two but still relatively influential in the model. Collectively, these three features are quite prominent in predicting the outcome, with each playing a strong role in the model's decision-making process. They are the most critical features based on their importance scores.

The second group includes the next three features (gs_max, gs_at_bed_bottom, gs_p90). The gs_max feature has an importance score of 0.094, marking a decrease from the earlier group but still significant. Then gs_at_bed_bottom follows closely with a score of 0.091 and gs_p90 has a similar score of 0.090. This middle group shows relatively less importance compared to the first group but still holds moderate influence within the model.

Lastly, the final four features form a group (gs_mean, gs_p10, gs_min, gs_at_bed_top). The mean grain size, gs_mean, is at 0.074, starting the final group with a lesser influence compared to previous features. The remaining features are gs_p10 with a score of 0.072, gs_min with a decrease to 0.069, and gs_at_bed_top has the lowest importance in this group and overall at 0.068. These four features show the least importance among the ten features considered. Their impact on model performance is lower, suggesting they are less critical in the model's decision-making process compared to the earlier groups. From this model, classification of the CSS database generates the Figure 40 confusion matrix. The matrix offers detailed insights into its performance across four classes.

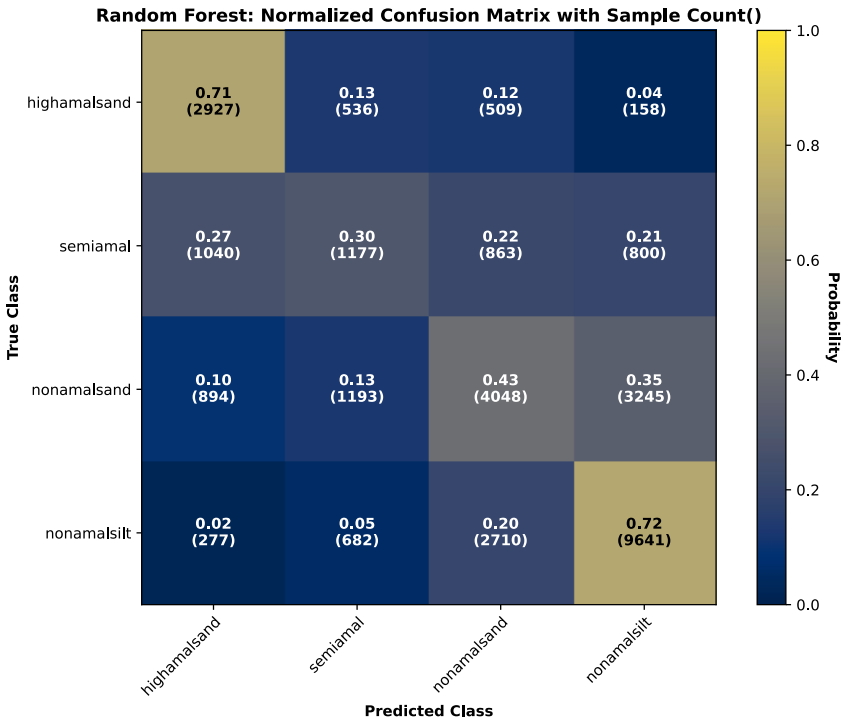

Figure 40: Normalized Confusion Matrix for Random Forest Model.

The matrix shows a relatively high accuracy for *highamalsand*, correctly classifying 71% of the samples (2927 out of a total), indicating a good ability to identify this class. For *semiamal*, the model has a lower accuracy rate of 30% (1177 correct classifications). *Nonamalsand* is classified correctly in 43% of cases (4048 samples), showing low accuracy. The model performs best with the *nonamalsilt* class, achieving an accuracy of 72% with 9641 samples correctly identified. There are notable misclassifications with *semiamal* samples being incorrectly classified as *highamalsand* (27% with 1040 samples), *nonamalsand* (22% with 863 samples), and *nonamalsilt* (21% with eight hundred samples). *Nonamalsand* also shows significant misclassification, particularly into *nonamalsilt* (35% with 3245 samples). A considerable portion of *nonamalsilt* samples are incorrectly predicted as *nonamalsand* (20% with 2710 samples).

Overall, the Random Forest model excels in identifying *nonamalsilt* but exhibits challenges with *semiamal* and *nonamalsand*, with notable misclassifications that could affect the practical application of

this model in scenarios requiring precise class separations. The robust performance in *highamalsand* and *nonamalsilt* suggests that features relevant to these classes are well-captured by the model. However, improvements might be necessary to enhance its capability to distinguish *semiamal* and *nonamalsand* more effectively, potentially through parameter tuning, more extensive training, or incorporating additional or more distinctive features.

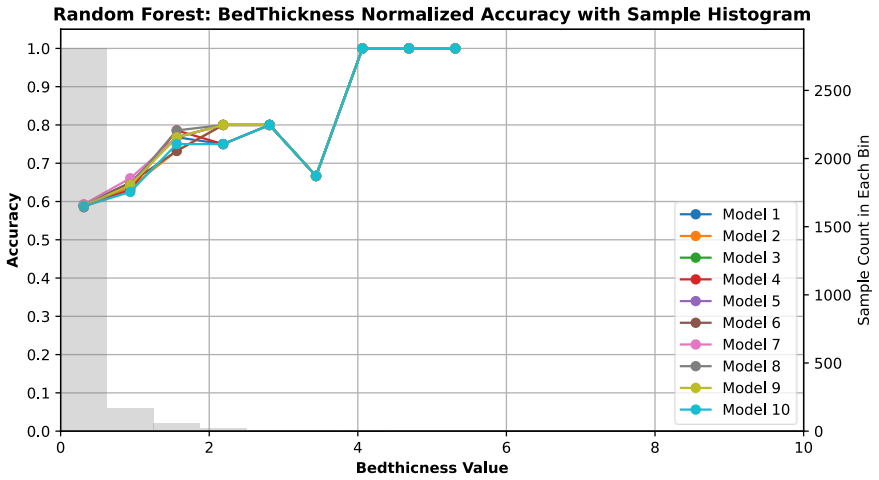

Figure 41: BedThickness Normalized Accuracy for 10 Random Forest Models.

Figure 41 begins with all ten models performing relatively similarly (~60%), and subsequently improving in accuracy for several bins. Then, all models experience a substantial drop in accuracy at roughly 3.5 meters BedThickness. Following the drop, there is a point of recovery (4 meters) where accuracy improves, with models converging in performance. In the final data points, there is a stabilization in the models' performance, reaching a higher level of accuracy (100%). The overall trajectory and differences among these models indicate that Random Forest is generally robust while its performance can vary somewhat based on random initiation.

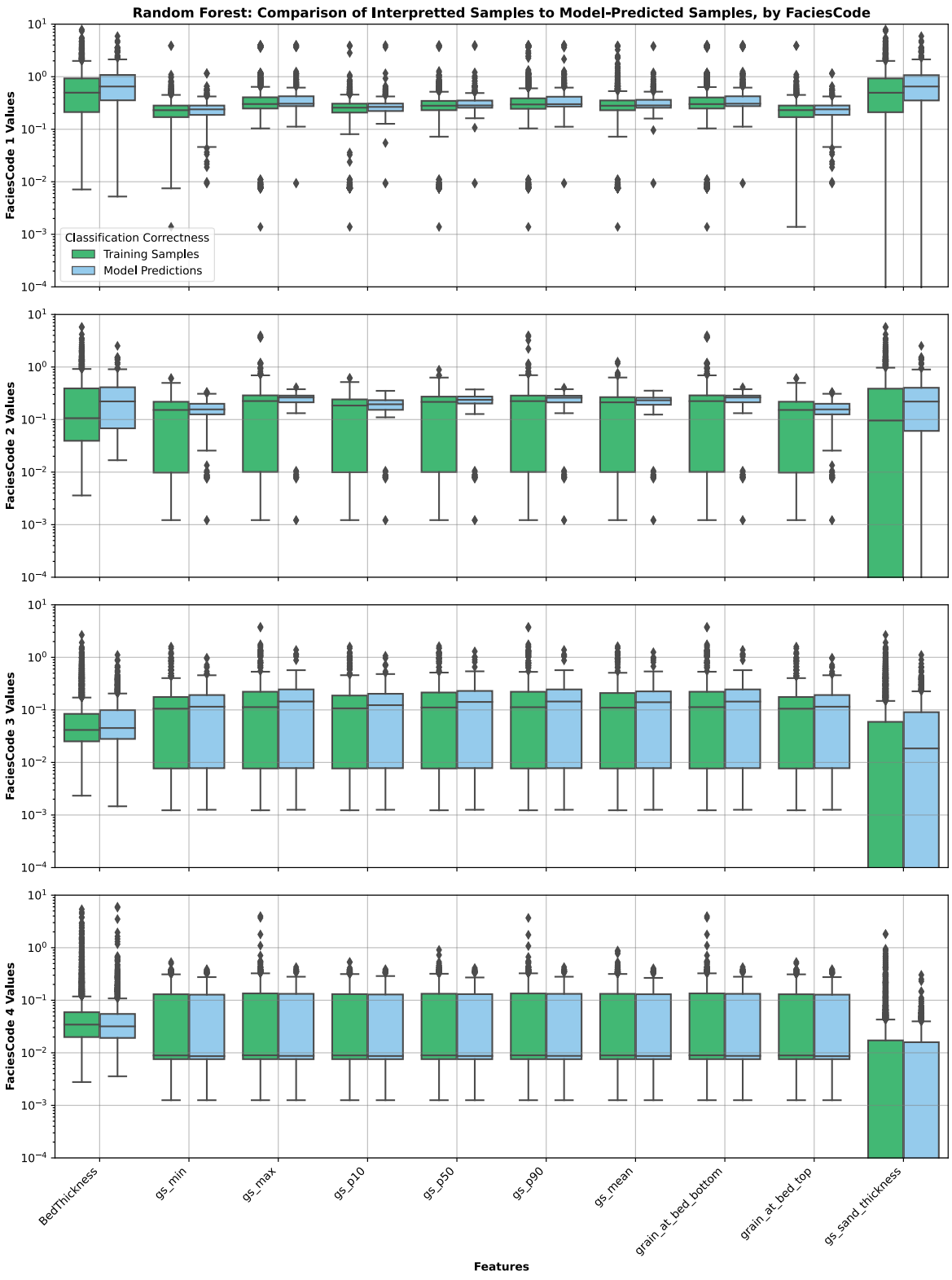

Figure 42: Comparison of Interpreted Class and Random Forest Classification Across All Features, Differentiated by FaciesCode.

Figure 42 offers a look at how the Random Forest model handles the classification of facies, comparing the distributions of various features in the training data (green box plots) and the model predictions (blue box plots). This analysis aids understanding the model's effectiveness. In some features (BedThickness, gs_sand_thickness), there is a good alignment between the predicted and training distributions. This is evident where the medians are closely matched, and the interquartile ranges (the boxes) are similar. Some features show significant differences (gs_min, gs_max, gs_p10), particularly in the spread and the presence of outliers.

BedThickness shows a strong match in distributions across most classes. The features, gs_min and gs_max have some deviations, especially in the outliers and the spread of the distributions. Percentile features (gs_p10, gs_p50, gs_p90) show significant differences in IQR in Facies 2, 3, and 4. Grain size means a good match in all but Facies 2, gs_at_bed_bottom and gs_at_bed_top also miss the IQR on Facies 2. Lastly, gs_sand_thickness shows a high degree of alignment. In summary, the Random Forest model performs well in capturing the distribution characteristics of various features across different classes, although there are noticeable areas of worse performance (Facies 2).

4.2.6 Neural Network Classifier

The Neural Network is the final method used in this analysis. In testing, the model generates a total 5-network accuracy of 55% and a BedThickness normalized accuracy of 63%. Plotting the performance in a confusion matrix (Fig. 43), displays areas of good and poor classification.

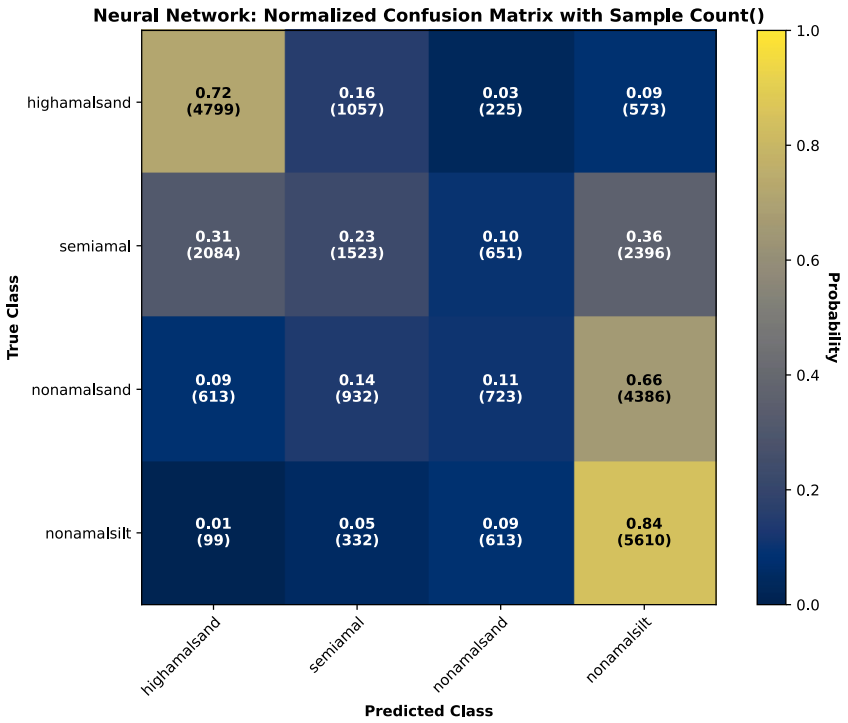

Figure 43: Normalized Confusion Matrix for Neural Network (5 Model) Classification.

The confusion matrix for the Neural Network Classifier shows best performacne in identifying *nonamalsilt* where it excels. *Nonamalsilt* achieves a very high accuracy of 90%, with 6016 samples correctly identified. Next best is *highamalsand*, with a high accuracy of 71%, correctly predicting 1465 out of the total samples for this class. Falling off from there, *nonamalsand* achieves an accuracy of 21%, with 980 samples correctly classified. And lastly, *semiamal* drops in accuracy to 1%, with 552 correct classifications.

There are significant misclassifications with *highamalsand* being incorrectly classified as *nonamalsand* (19% probability with 384 samples) and *nonamalsilt* (10% probability with 203 samples). A substantial number of *semiamal* samples are incorrectly predicted as nonamalsily (40% probability with 782 samples), *nonamalsand* (28% probability with 552 samples), and *highamalsand* (30% probability with 552 samples). The model also frequently misclassifies *nonamalsand* as *nonamalsilt* (70% probability with 3269 samples).

Overall, the Neural Network Classifier shows excellent performance with *nonamalsilt* and *highamalsand* but struggles considerably with *semiamal* and *nonamalsand*. In particular it performs very poorly in classifying Facies 2. These characteristics are somewhat reflected in the BedThickness normalized accuracy plot for the Neural Network model (Fig. 44).

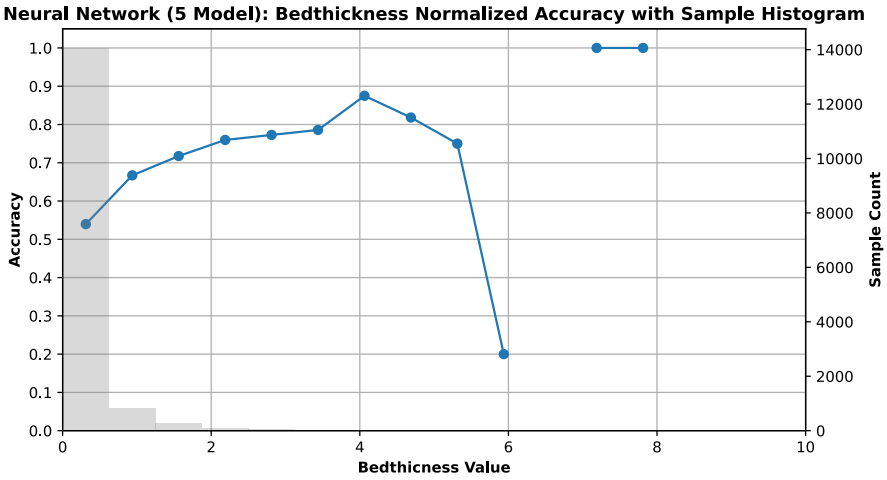

Figure 44: BedThickness Normalized Accuracy for 5 Neural Network Models.

The normalized accuracy plot shows a gradual increase in accuracy as BedThickness increases, starting from the left. The highest accuracy is achieved by the model at roughly 4 meters BedThickness value. Following the peak, there is a sharp decline in accuracy as BedThickness continues to increase towards 6 meters. After the sharp decline, the accuracy re-appears at 100%, maintaining a constant level.

Figure 45 shows the feature distribution for the training samples plotted against the Neural Network-classifed samples. This plot shows some of the unique performance of the Neural Network.

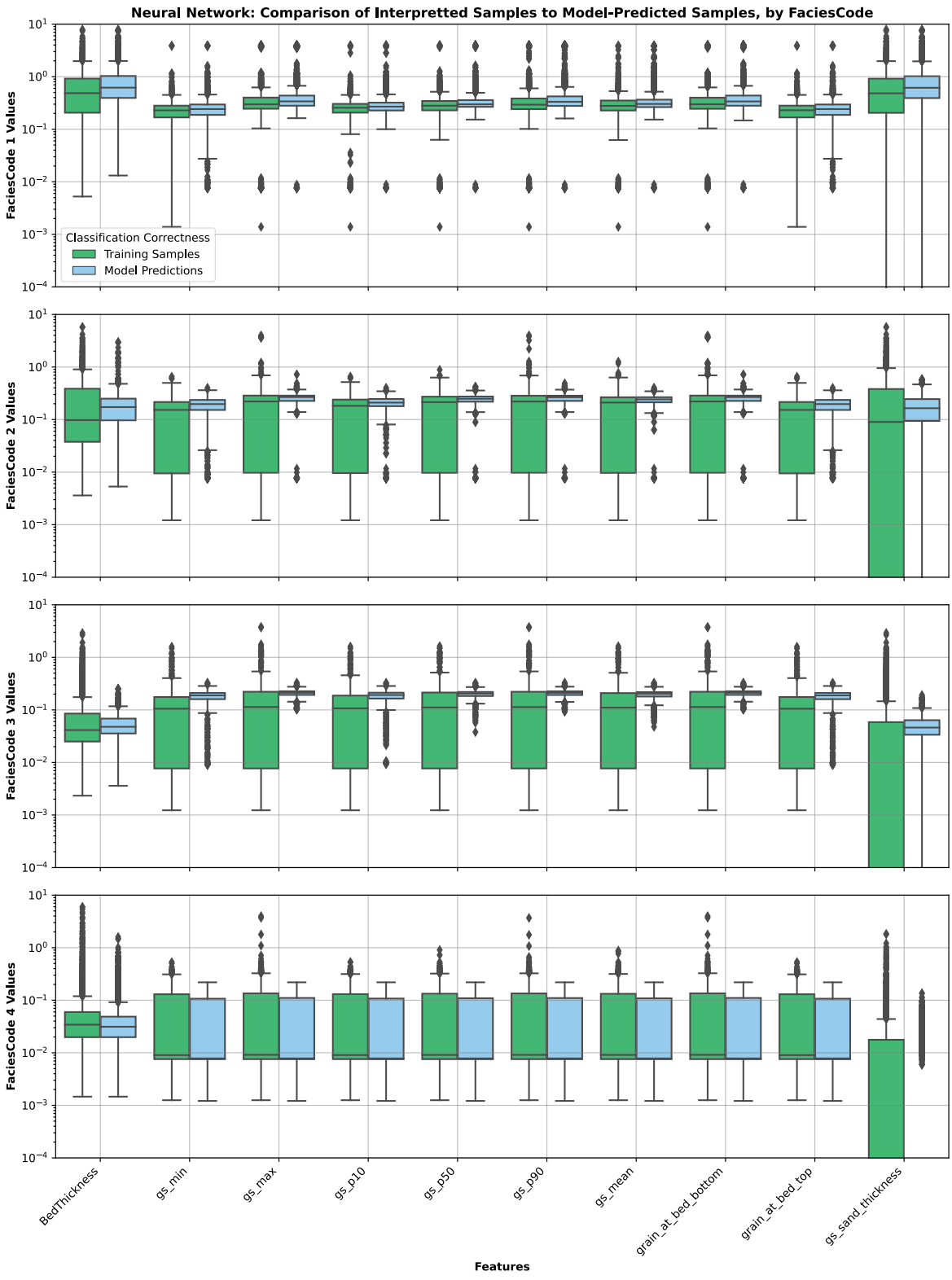

Figure 45: Comparison of Interpreted Class and Neural Network Classification Across All Features, Differentiated by FaciesCode.

Analyzing the plot for the Neural Network model in terms of how it classifies the CSS dataset offers valuable insights into the performance and characteristics of the Neural Network. The model shows varying degrees of alignment between the predicted and training data distributions. BedThickness shows a strong match in distributions across Facies 1 and 4, and struggles more with Facies 2 and 3. The features gs_min and gs_max show some deviations, especially in IGR in Facies 2 and 3. Percentile features (gs_p10, gs_p50, gs_p90) display a mix of alignment and slight discrepancies, suggesting nuances in the data that the model might not fully capture. Grain size mean generally matches well. Gs_sand_thickness, which usually shows high alignment, is poorly matched in all classes. This is unique among the methods assessed in this study.

Neural Network results were verified against the results of an open source classifier from the sklearn kit called the Multi-Layer Perceptron (MLP). This method showed total accuracy of 50% and a BedThickness normalized accuracy of 68%. This moderately improves on the earlier method but shows similar per facies accuracy, meaning the intermediate classes do not perform well. The following confusion matrix is for the MLP method.

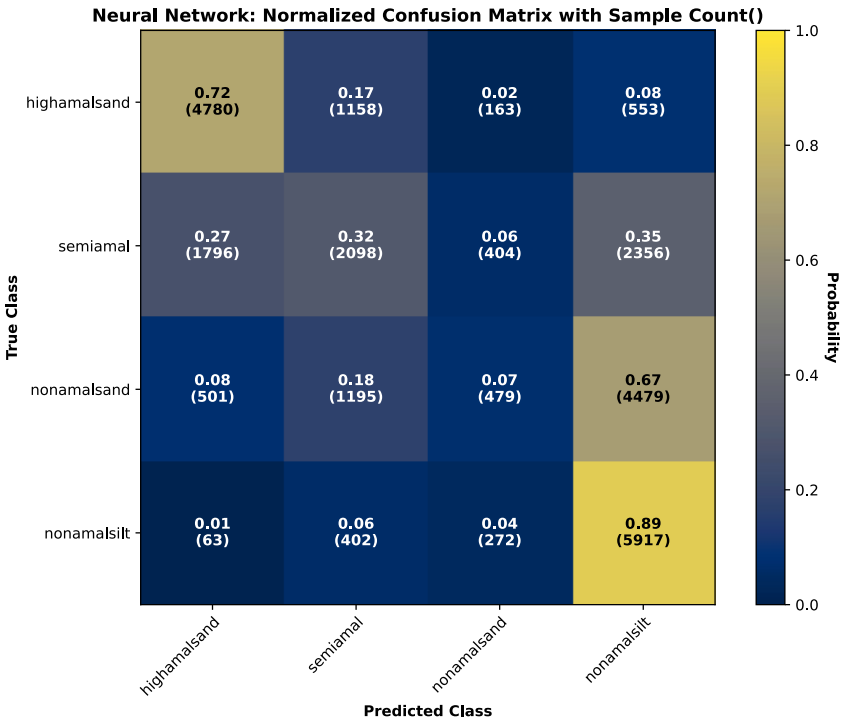

Figure 46: Normalized confusion matrix for Multi-Layer Perceptron method.

In conclusion, the Neural Network generally performs competently in capturing the distribution characteristics of various features across different classes. However, noticeable areas of poor performance exist for Facies 2 and 3.

5. DISCUSSION AND INTERPRETATION

Plotting the ML results as graphs (previous section) is useful for method evaluation but fails to give any direct insight into the geologic significance/correctness of the results. In the context of facies prediction, it is the case that coarser facies (Facies 1 and 2) are the most critical for calculating reservoir fluid volumes and the finer, thinner-bedded facies (Facies 3 and 4) are critical for their impact as barriers to fluid flow. This makes correct prediction of the less abundant *highamalsand* and *semiamal* more significant for volume estimation, which is critical for assessing the high exposure to financial risk at the exploration stage. In later development stages, *nonamalsand* and *nonamalsilt* are more important for fluid flow considerations. The ML methods also provide information about the database's utility in this classification problem. Leveraging ML results against one another provides a clearer picture of facies prediction performance as a whole.

5.1 Unsupervised Methods

Results from principal component analysis in the CSS database show that most of the database's variance (>95%) can be accounted for with 4 dimensions. This suggests that some features may not contribute strongly to classification as their variance is much less than that of more determinant features. This finding is later verified in Logistic Regression coefficients and Random Forest importance.

5.1.1 K-Means Clustering

Within K-Means, four metrics are used to determine the proper number of classes (Inertia, Silhouette Score, Davies-Bouldin Index, Calinski-Harabasz Score). Figure 15 shows decreasing slope-magnitude for Inertia at 5 clusters. This decrease can indicate the number of distinct clusters in a dataset. The silhouette score shows that as more classes (beyond five) are differentiated, the average silhouette width stabilizes at about 0.6. Both Inertia and Silhouette Score suggest that the ideal number of clusters for this dataset is between 4 and 5. Figure 16 gives additional insights into the distribution of CSS database samples *n*-D space. Both Davies-Bouldin and Calinski-Harabasz suggest that the proper number of classes

is likely between 3 and 4. These metrics together verify that it is reasonable to classify the facies data into a 4-class system.

Figures 17 and 18 hint at the tendencies of the researchers collecting measured sections. Figure 17 shows clustering of BedThickness values that is not diagnostic of the rock formation but of BedThicknesses that are readily estimated. For instance, we see clustering at BedThickness values of 0.01, 0.1, and 1, which have no physical basis for being this dominant in the data. We can also observe the impact of typical, sedimentology-grain size interpretation cards in Figures 17 and 18. Many grain sizes occur near to the common delineations of this widely adopted tool.

Figure 17 also distinctly shows that K-Means is dividing classes with distinct boundaries in the feature space. From intuition about the data collection methods as well as Figure 10, we know that facies often overlap in feature space, intermingling at class fringes. This means that distinct boundaries like K-Means cannot fully differentiate this dataset.

The plot of database features per facies also aids in interpreting the K-Means performance (Figure 19). We can see that the K-Means Facies 1 is coincident between the training data and the model predictions. The model appears to distinguish the coarsest samples well from the background siltstones. These similar distributions are also true for the Facies 4 class, but the performance is vastly different for the other two classes. Facies 2 and 3 are much more broadly distributed in the training than in the prediction data. This shows that those classes are more variable in their lithology than the more distinct end members. The two classes significantly overlap in feature space, lowering the K-Means performance. Overall, K-Means performs best in the grain size endmembers and poorly in the more similar gradational classes.

This study's results do not directly compare to the literature in accuracy or data type. Similar investigations can generate accuracies of greater than 80% (e.g., Bhattacharya et al., 2016) in multiclass facies prediction schemes but rely on physical well logs tied to core for facies interpretation. The study by Bhattacharya, Carr, and Pal titled "Comparison of supervised and unsupervised approaches for mudstone lithofacies classification: Case studies from the Bakken and Mahantango-Marcellus Shale, USA" compares

the efficacy of supervised and unsupervised machine learning techniques for classifying mudstone lithofacies. The authors used log data from the Bakken and Mahantango-Marcellus Shale formations. They found that supervised methods, particularly Neural Networks, performed better in terms of accuracy and reliability compared to unsupervised methods like K-Means clustering (Bhattacharya et al., 2016). This study achieves an accuracy of 71.3% in a similar unsupervised cluster method, Multi Resolution Graph-based Clustering (MRGC). This is much higher than the K-Means application in this study due to the statistical overlap of the intermediate facies (Facies 2 and 3) in the CSS Database.

Key findings from the application of K-Means Clustering include the following: (i) The 4-facies system is reasonable for clustering analysis, as indicated by several metrics of clustering performance in the training database; (ii) Facies 2 and 3 exhibit similar feature distributions after K-Means classification (indistinct classes), with the model distinguishing samples primarily based on distinct BedThickness values between the two facies; and (iii) Clustering proves insufficient for classifying facies from these features, partly due to the overlap of facies classes in 10-D (feature space).

5.2 Supervised Methods

The findings of the K-Means method inform the application of supervised ML methods on the same database. In the 4-class system the overlap between samples in feature space is challenging for unsupervised methods. Supervised methods provide additional learning architectures that can account for these challenges to the classification.

The consensus of current literature is that supervised ML methods deliver superior performance in geologic interpretation tasks (Bhattacharya et al., 2016; Fadokun et al., 2020). This study also finds this to be the case with the best results metrics coming from the group of Logistic Regression, Random Forest, and Neural Network methods. In conjunction with other supervised methods, this study constrains solutions to the 4-class facies prediction problem.

5.2.1 Least Squares Solution

Least Squares solution provides insight into the performance of simpler ML methods. The results showed that, like the K-Means method, the gradational "middle" facies are more challenging to predict than the Facies 1 and 4 endmembers.

Facies 1: highamalsand

The initial results of supervised classification provide some insight into the most sand-rich of the bed samples. The model (Fig. 20) shows a high rate of false negatives suggesting that the *highamalsand* (Facies 1) is commonly mistaken for other facies, likely Facies 2 or 3. This is expected due to the overlap of high magnitude-outlier samples in Facies 2 (Fig. 10) with the IQR of Facies 1 features. Some of the coarsest samples of Facies 2 (*semiamal*) are not wholly distinct from samples of Facies 1 (*highamalsand*). However, the model has a particularly good True Negative rate at 99%. This trait could be useful for identifying False Positives among other methods.

Significant false negatives explain the variability of the BedThickness normalized accuracy in Figure 21. The variable accuracy in samples with BedThickness greater than one meter is problematic for predicting sandier facies, making this method poorly suited to finding coarse facies.

Facies 2: semiamal

Performance suggests that *semiamal* samples are not as statistically distinct as *highamalsand*. High accuracy with high sample counts suggests robust model performance in those ranges. The lowest point in accuracy might correspond to a range of BedThickness where the characteristics of FaciesCode 2 blend with other classes, making discrimination difficult. Figure 22 shows that the mixing of features with *nonamalsilt* (Facies 4) is reducing performance. *Semiamal* also shows variable performance relative to BedThickness. Accuracy varies, suggesting that there are particular BedThicknesses where the Facies 2 class is more distinct than others.

Facies 3: nonamalsand

The Facies 3 case shows similar performance to Facies 2. The classification fails to make significant classifications on the side of the target class and instead favors "All Other Classes." Both the intermediate facies (2 and 3) are less statistically distinct than the endmembers. The Least Squares solution is not sufficient to differentiate these classes.

Facies 4: nonamalsilt

In this final scenario, the model demonstrates balanced but not optimal performance. The moderate false negative rate and a relatively high false positive rate affect the overall reliability and precision of the model. Even in these conditions the model can correctly predict the facies most of the time. With total accuracy of 64% and BedThickness accuracy of 91%, this configuration of Least Squares is a moderately effective *nonamalsilt*-classifier. This is significant as these samples represent a majority of the training dataset.

Figure 26 indicates that while the model is competent at distinguishing Facies 4 from other classes, it does not excel in either accuracy or recall. The considerable proportion of false negatives and false positives suggests that the model could benefit from further tuning or feature engineering to enhance its discrimination ability for Facies 4 versus other classes.

The accuracy plot of Figure 27 shows a significant increase in accuracy from the first to the second bin. This suggests that the model initially struggles to classify Facies 4 correctly but quickly improves in thicker beds. This is expected as the most Facies 4 beds exist in the first bins of the sample count plot. The lower accuracy in the first bin points to samples where the coarser facies (1, 2, 3) have a lower BedThickness value or more feature variance within that range of the bin.

Key findings from the application of the Least Squares Solution include the following: (i) Facies 1 accuracy is poor, but the model almost never mislabels Facies 1 when the true class is another facies; (ii) Facies 2 and 3 are challenging to classify from the total training set using the Least Squares method; and

(iii) Facies 4 shows balanced performance in the binary system, demonstrating utility in a sandstone/shale (lithological) binary classification.

5.2.2 Logistic Regression

The first utility for the Logistic Regression method is to analyze the feature influence in the model after training. Training the Logistic Regression model adjusts the model weights used in the linear combination of input features, W (Eqn. 4). This matrix of coefficients, plotted in Figure 28, shows the model dependance on gs_sand_thickness and the percentile statistics. This corroborates the PCA analysis findings that most database variance is accounted for in a 4-dimensional system. The primary dimensions likely correspond to gs_sand_thickness, the percentile statistics, and a few others. Significantly, facies exhibit a positive or negative relationship to the percentile statistics as a group. Percentile features have positive group relationships in the coarser facies (Facies 1 and 2) and negative in the finer facies (3 and 4). This pattern, in conjunction with the overall lower strengths of relationships, suggests that the percentile features do not represent a large part of the data variance. It may be possible to achieve similar model performance without having to incorporate all these features.

Once predictions are made, a confusion matrix (Fig. 29) for the Logistic Regression model shows the first full 4-facies classification of this study. This performance is variable between the classes. The lack of *semiamal* (Facies 2) is a challenge to the prediction of coarser and sandier facies in this framework. Among the misclassifications, the vast majority are finer-siltier facies, mostly *nonamalsilt* (Facies 4). This is not related to the biasing of the training data towards *nonamalsilt*, as training data is balanced before use. To accomplish this, smaller classes are re-sampled to a higher count, making them less variable than a class filled with unique samples.

The normalized accuracy metric shows a trend of better performance with increasing BedThickness value. This trend is disrupted by two bins with much lower performance at BedThickness values of 3.5 to 4 meters. This characteristic of variability persists between methods and often occurs at roughly 4 meters

BedThickness. This suggests there are samples in that range that are particularly challenging to classify accurately. Following the drop, accuracy quickly recovers and stabilizes at a high level. The overall pattern observed implies that Logistic Regression is effective but can experience significant sensitivity to certain data characteristics.

Comparison of training data features to predicted facies features in Figure 31 shows the challenge in classifying Facies 2 and 3. The intermediate classes are much more variable in their grain size distribution than the model is selecting. Taking this in conjunction with the Figure 29 results, it is apparent that Logistic Regression is not sufficient to distinguish the intermediate classes.

This method appears in the literature in adjacent research classifying beach sand grain size. Peter Vincent makes environment of deposition classifications using a Logistic Regression model, trained on the grain size distributions of known sand samples (Vincent, 1986). This method is effective in differentiating beaches from coastal dunes using hyperbolic distribution parameters, like the grain size statistics used in this analysis. This analysis suggests that classification is less robust using Logistic Regression as the number of facies or classes increases.

Key findings from the application of Logistic Regression include the following: (i) Facies 1 and 4 accuracies are higher relative to the intermediate facies; (ii) Facies 2 and 3 are more challenging to classify, with Facies 2 being almost absent from classifications and notably absent from measured section visualizations; and (iii) Logistic Regression shows better performance in BedThickness weighted accuracy over accuracy (preferred performance characteristic).

5.2.3 Linear Discriminant Analysis (LDA)

Discriminant Analysis requires assumptions about the dataset. It assumes a normal distribution within data features as well as with roughly equivalent variance-covariance matrices between classes (facies). A quick glance at the training data feature-distributions (Fig. 10), reveals that none of these conditions are met. Facies classes of this set tend to overlap in feature space and exhibit significant differences in variance. This explains the poor classification performance in both discriminant model types.

Within the Linear Discriminant model, the best performing facies are the endmembers (Facies 1 and 4). This model is uniquely capable of distinguishing Facies 2 with a True Positive rate of 33%. Even though this is the most likely prediction among the four facies, misclassifications are evenly distributed to the other classes. Facies 2 performance is only mildly better than a random 1 in 4 chance of predicting the right facies. This is still not optimal performance for differentiating the intermediate facies (Facies 2 and 3) or coarser facies (Facies 1 and 2)

The LDA model performs differently across the four classes, with some classes showing better alignment between training and predicted data than others. The intermediate classes again are predicted as much tighter distributions of features than they appear in the training samples. The observed differences suggest areas where the LDA model might benefit from further tuning for features where the model predictions consistently diverge from the training data.

LDA models have been utilized in facies classification before. Zheng et al., 2021 applied the method to karsted reservoirs, predicting the vug and karst facies near well-bore. Their models achieved an accuracy of 92%. However, this system failed to resolve karsted reservoirs in other intra-basin settings, reducing its utility in exploration applications. This study finds similar performance constraints in the turbidite slope-channel system setting. Total performance is lower than published examples with significant bias to end-member facies.

Key findings from the application of Linear Discriminant Analysis (LDA) include the following: (i) Facies 1 and 4 accuracies are the highest; (ii) Facies 2 is predicted correctly more often than any other classification, despite being a notably difficult facies (absent in other methods); and (iii) there is a positive correlation between BedThickness and accuracy, which is important for fluid volume estimations.

5.2.4 Quadratic Discriminant Analysis (QDA)

The second discriminant analysis method showed worse performance than LDA. QDA consistently predicts siltier and finer facies than the true facies. The QDA model demonstrates strong discriminative

power for the *nonamalsilt* class, which might be attributed *nonamalsilt* representing more than one third of the training dataset. However, it struggles significantly with *highamalsand*, *semiamal*, and *nonamalsand*.

This method avoids the dropped class problems that Least Squares solution and Logistic Regression see with the intermediate facies (Facies 2 and 3). The QDA normalized confusion matrix (Fig. 35) shows the trend of prediction plotting above and right of the diagonal of true positives. This leads to the lowest total accuracy of this analysis.

Understanding the nature of the algorithms, both LDA and QDA assume that each class has a Gaussian distribution with the same covariance matrix. If the classes overlap significantly, the model might struggle to distinguish them. Results of plotting feature distributions (Fig. 10) show significant overlap among facies classes, particularly in Facies 2 and 3. This explains the poor accuracy metrics and the increased variability in the BedThickness normalized accuracy plot (Fig. 36).

Interestingly, the LDA method is more accurate and robust in the intermediate classes (Facies 2 and 3). This implies something about the nature of boundaries in the CSS Database feature-space. LDA will use linear functions to approximate a hyperplane, or multi-dimensional plane between classes in feature space. The inferior performance of the QDA suggests that the more complex discriminant function does not capture the character of the class boundary as well as the linear function.

Key findings from the application of Quadratic Discriminant Analysis (QDA) include the following: (i) Facies 4 dominates all predictions (most abundant); (ii) all predictions are siltier than the true values, with a tendency towards Facies 4 (*nonamalsilt*); and (iii) the CSS Database facies vary significantly in feature distribution and variance, making discriminant analysis unsuitable for this dataset.

5.2.5 Random Forest

The Random Forest model is the most successful model in accuracy among the methods classifying the full four class scheme. Notably, all four classes are most predicted as their true class (correct prediction), something that no other method achieved. Halotel et al. (2020), showed that Random Forest Classifiers can perform well with interpreted geologic input as a part of the training data. There is also evidence to suggest

that Random Forest Classifiers outperform linear boundary-type Support Vector Machines (SVMs) (Halotel et al., 2020). This aligns well with this study's findings of the Random Forest as the most robust performance in an interpretive geologic classification.

This method also allows for some feature analysis using the feature importance metric. This is a measure of how often the feature is used in the algorithm to branch the decision tree. The plot, Figure 39, corroborates the result of the earlier Logistic Regression coefficient analysis. BedThickness, gs_sand_thickness, and gs_p50 are the most determinant features in this decision tree classification (>45% of total importance). This mirrors what you would expect a sedimentologist to use in interpretation. Metrics like BedThickness, the median grain size, and the proportion of sand are indicative of the processes that form the geologic beds, indirectly acting as a proxy to the sedimentary facies being predicted.

This could also be why the Random Forest architecture displays the best 4-facies accuracy of this study. The typical method of interpretation for field geologists can be modelled like a series of binary decisions. *Is the bed thick or thin? Are the grains sandy or silty? Are coarse or fine grains more abundant?* The Random Forest uses a branching network that resembles, at least in structure, a human interpretation process.

The normalized accuracy plot (Fig. 41) reflects the better overall performance and displays a positive relationship between BedThickness and accuracy. Figure 42 compares feature distribution and shows that Random Forest has the best feature-fit to Facies 3 of all methods. Most commonly Facies 3 is predicted to be a much tighter distribution towards the coarse Facies, 1 and 2. Here the model distinguishes it as more similar in distribution to Facies 4.

Key findings from the application of Random Forest include the following: (i) Random Forest demonstrates the best accuracy in intermediate classes (*semiamal* and *nonamalsand*); (ii) performance for endmembers (Facies 1 and 2) is slightly reduced compared to Neural Network and Logistic Regression; and (iii) there is a positive correlation between BedThickness and accuracy.

5.2.6 Neural Network Classifier

The Neural Network implementation relative to other methods, does not perform as well with endmember facies as Logistic Regression but avoids the dropped-class issue with *semiamal*. What it does have over methods like LDA and QDA in accuracy, it lacks robust classification in the more challenging intermediate facies like LDA and Random Forest. The overall accuracy score for this method was lower than Random Forest and Logistic Regression.

Neural networks represent a popular choice in facies prediction in datasets like seismic acquisitions (Zhao et al., 2015), as well as grain sizes (Zhang et al., 2021). The Zhang et al. study of 2021, "Applying convolutional Neural Networks to identify lithofacies of large-n cores from the Permian Basin and Gulf of Mexico: The importance of the quantity and quality of training data" provides a valuable reference for this analysis. Their researcher found that Neural Network classification can produce facies accuracy in the 70% to 80% range. This fits closely to the performance of individual classes within this study's model, although this occurs only in the endmember (Facies 1 and 4). Neural Network classification in the CSS database produced much lower accuracies for the intermediate facies (23% and 11% for Facies 2 and 3 respectively), following the trend of worse performance among Facies 2 and 3 relative to Facies 1 and 4. Notably, the Zhang, 2021 results are in core-derived grain size data analogous to the CSS database.

Another example of this method in facies prediction includes a notable publication by Qi and Carr. The 2006 paper applies a Neural Network to predicting geologic interpretation in the Hugoton gas-field of southwest Kansas. The team employed a Neural Network trained using a dataset of well logs and corresponding lithofacies classifications derived from core data (Qi and Carr, 2006). The authors collected log data, including gamma-ray, neutron porosity, density, and sonic logs. These logs provide indirect measurements related to grain size and other sedimentological properties. Their research found predicted litho-facies display accuracies of 70-90% depending on the facies and sets a reasonable baseline for studies of similar data quality.

Figure 44 shows that the Neural Network in this study achieves a positive correlation from BedThickness to accuracy but also displays a significant drop in performance at higher values of BedThickness. This shows the network struggles to classify some thicker samples. This feature is present in normalized accuracy plots of other methods, but the Neural Network displays more pronounced drop in the offending bin (20% accuracy). The feature distribution comparison (Fig. 45) shows the significant difference between the training data to the model predictions. The network distinguishes samples with any coarse grains as likely Facies 1, 2, and 3, which does not capture the whole feature spread in the training data.

Key findings from the application of Neural Network include the following: (i) Facies 1 and 4 accuracies are high, similar to Logistic Regression and Random Forest; (ii) Facies 2 and 3 accuracies are low, but still present, with visualization being better than the Logistic Regression method; and (iii) there is a positive correlation between BedThickness and accuracy, although total accuracy is notably lower than Logistic Regression or Random Forest.

5.3 Comparative Results of All Methods

In the context of geologic exploration, sandier target facies often exhibit thicker bedforms. This makes it important that methods positively correlate accuracy to BedThickness. This is the case in all of the 4-facies methods to varying degrees. Unfortunately, the K-Means and the coarser facies (Facies 1 and 2) of the Least Squares method show the opposite trend, making them sub-optimal for this task.

5.3.1 Quantitative Comparison

Comparing all method accuracies against one another reveals the different classes of methods as discussed in Chapter 3. Figure 47 shows K-Means as strictly lower than the other methods. Quadratic Discriminant Analysis also falls below the other methods distinctly. Together, these methods are not suitable for facies classification in this core-analogous data type. Least Squares represents another

performance group that is problematic for reasons explained in the methods section and are therefore sub-optimal for this classification scheme.

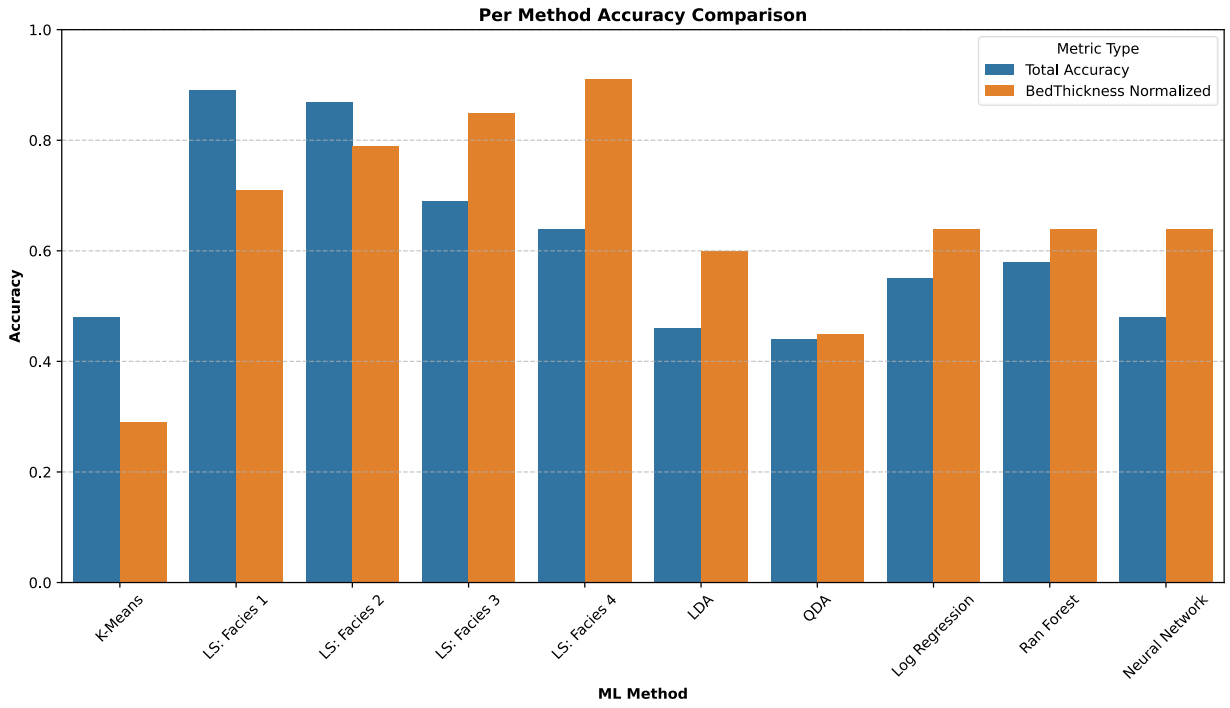

Figure 47: Accuracy Comparison by Method; Total Accuracy in Blue and BedThickness Normalized Accuracy in Orange.

This leaves Logistic Regression, Linear Discriminant Analysis, Random Forest, and Neural Network as multi-class classifiers with reasonable performance. Among them, Random Forest has the best total accuracy and comparable BedThickness normalized accuracy, making it the best method among the seven of this study in mimicking the original facies interpretation of the CSS measured section database. Logistic Regression is the second-best method by performance.

Table 6 shows additional metrics beyond accuracy for all the ML methods. Cells of the table are shaded to highlight high performance. Least Squares, Logistic Regression, and Random Forest all show more yellow (50%-69%) and green (70%-100%) cells than other methods. Table 7 breaks down the Logistic Regression model's performance by Facies class. This highlights the difficulty in classifying the intermediate facies. Table 8 and 9 then show a similar breakdown in Random Forest and Neural Network

performance, respectively. Among these three, more metrics show higher performance values in the Random Forest model.

Table 6: Comparison of ML Method Performance Metrics (grey =0%-29%, red=30%-49%, yellow=50%-69%, green=70%-100%).

Performance Metrics by ML Method	Accuracy	BedThickness Normalized Accuracy	Precision (Weighted)	Recall (Weighted)	F1 Score (Micro)	F1 Score (Macro)
K-Means	48.0%	29.5%	47.5%	48.0%	48.0%	28.4%
Least Squares (Facies 4 - Binay)	64.3%	91.5%	64.5%	64.3%	62.2%	61.7%
Logistic Regression	54.5%	63.5%	47.2%	54.5%	54.4%	38.1%
LDA	47.7%	62.0%	48.0%	47.7%	45.6%	43.9%
QDA	49.2%	46.4%	48.4%	49.2%	37.5%	28.8%
Random Forest	59.0%	64.2%	58.8%	59.0%	57.6%	53.0%
Neural Network	46.8%	63.2%	43.7%	46.8%	46.8%	41.8%

Table 7: Per Facies Performance Metrics for Logistic Regression Model (grey =0%-29%, red=30%-49%, yellow=50%-69%, green=70%-100%).

Performance Metrics by Facies Class (Logistic Regression)	Precision	Recall	F1 Score
Facies 1: *highamalsand*	50.8%	82.5%	62.9%
Facies 2: *semiamal*	39.3%	3.3%	6.1%
Facies 3: *nonamalsand*	41.8%	11.8%	18.4%
Facies 4: *nonamalsilt*	58.9%	93.0%	72.1%

Table 8: Per Facies Performance Metrics for Random Forest Model (grey =0%-29%, red=30%-49%, yellow=50%-69%, green=70%-100%).

Performance Metrics by Facies Class (Random Forest)	Precision	Recall	F1 Score
Facies 1: *highamalsand*	55.0%	71.6%	62.2%
Facies 2: *semiamal*	29.8%	28.3%	29.0%
Facies 3: *nonamalsand*	52.3%	43.1%	47.3%
Facies 4: *nonamalsilt*	69.9%	73.0%	71.4%

Table 9: Per Facies Performance Metrics for Neural Network Model (grey =0%-29%, red=30%-49%, yellow=50%-69%, green=70%-100%).

Performance Metrics by Facies Class (Neural Network)	Precision	Recall	F1 Score
Facies 1: *highamalsand*	61.5%	73.7%	67.1%
Facies 2: *semiamal*	40.0%	19.0%	25.7%
Facies 3: *nonamalsand*	30.2%	12.9%	18.1%
Facies 4: *nonamalsilt*	43.0%	81.6%	56.3%

In a per facies sense, it is most important to perform well in the Facies 1 (highly amalgamated sandstone). Among all methods, this class has a the highest true-positive rate in the Logistic Regression method with 79%. Similarly, Facies 2 performs a few percentage points better in Logistic Regression (33%) than in Random Forest (30%). Random Forest is then best in classifying Facies 3 at 43%. Lastly Logistic Regression generates the highest True Positive rate for Facies 4 at 92% of samples.

5.3.2 Visual Results in Training and Testing Area

The results for the ML predictions by measured sections (CACH1 and FIG100; Figs. 48 and 49), provide a visual comparison between data, original interpretation, and the predictions of the seven methods. Importantly, the Least Squares-measured section utilizes the Facies 4 vs. all other classes scenario. Among the four high performing, 4-facies classifiers in Figure 47, all classify the Facies 1 and 4 portions of each section well. As before, where Random Forest had better accuracy in intermediate facies, Random Forest correctly classifies the most Facies 2 samples. Most methods lack many predictions of Facies 2. Quadratic Discriminant Analysis does predict many Facies 2, but often where there should be Facies 1. Random Forest can make these predictions without misclassifying Facies 1. The Neural Network also over predicts Facies 2 to its detriment.

While Logistic Regression had high performance metrics, plotting the predicted measured sections shows its poor discernment in intermediate classes. Conversely to before, Logistic Regression under-predicts Facies 2. The predicted MS looks like the Least Squares, in that it lacks many Facies 2 and 3 predictions. For similar performance metrics and a more accurate measured section print, Random Forest is the better classification method for this interpreted facies-prediction.

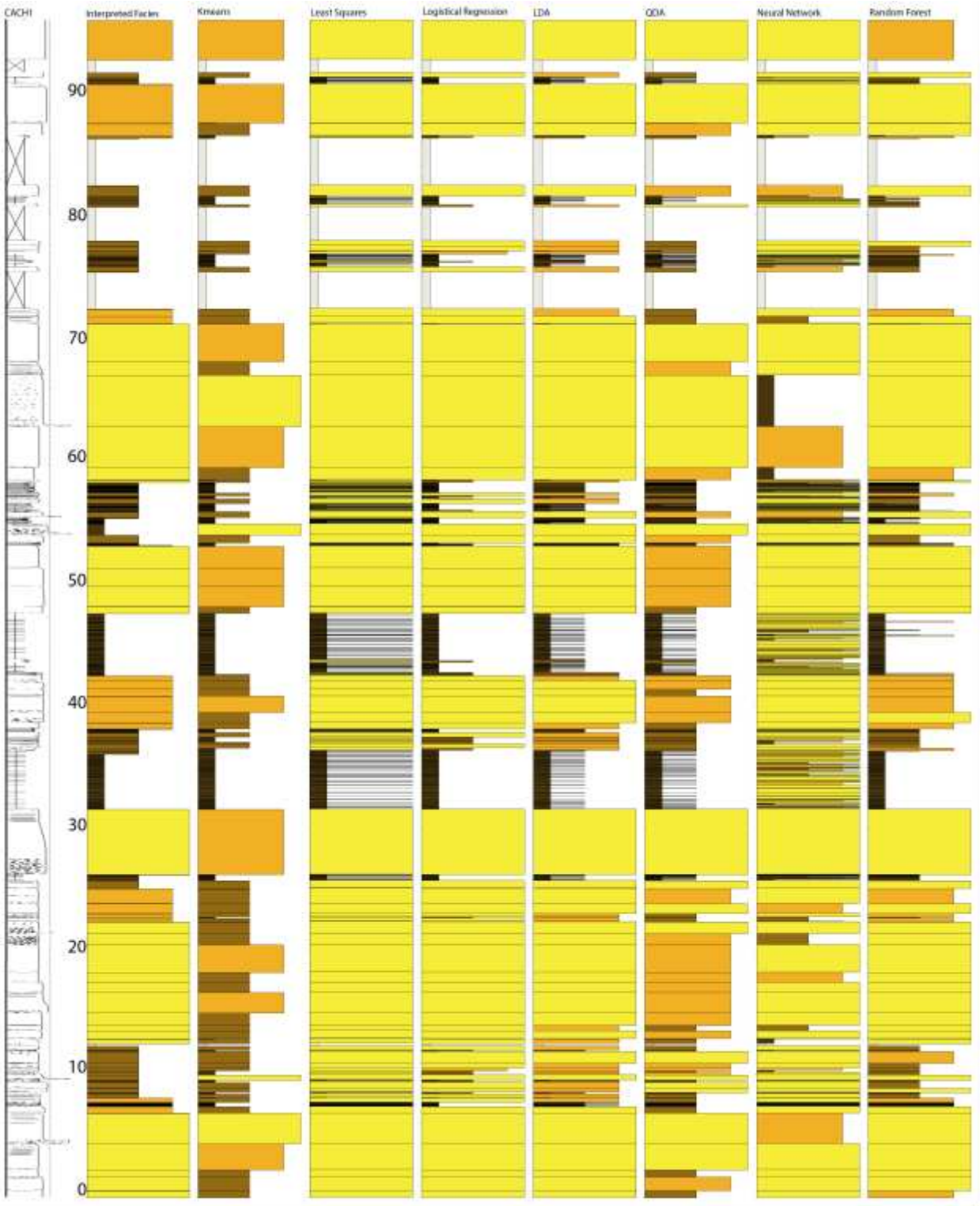

Figure 48: CACH1 -- Drafted measured section (column 1), interpreted facies (column 2), and seven prediction methods (columns 3-9). Color scheme for facies delineation is provided in Figure 1A.

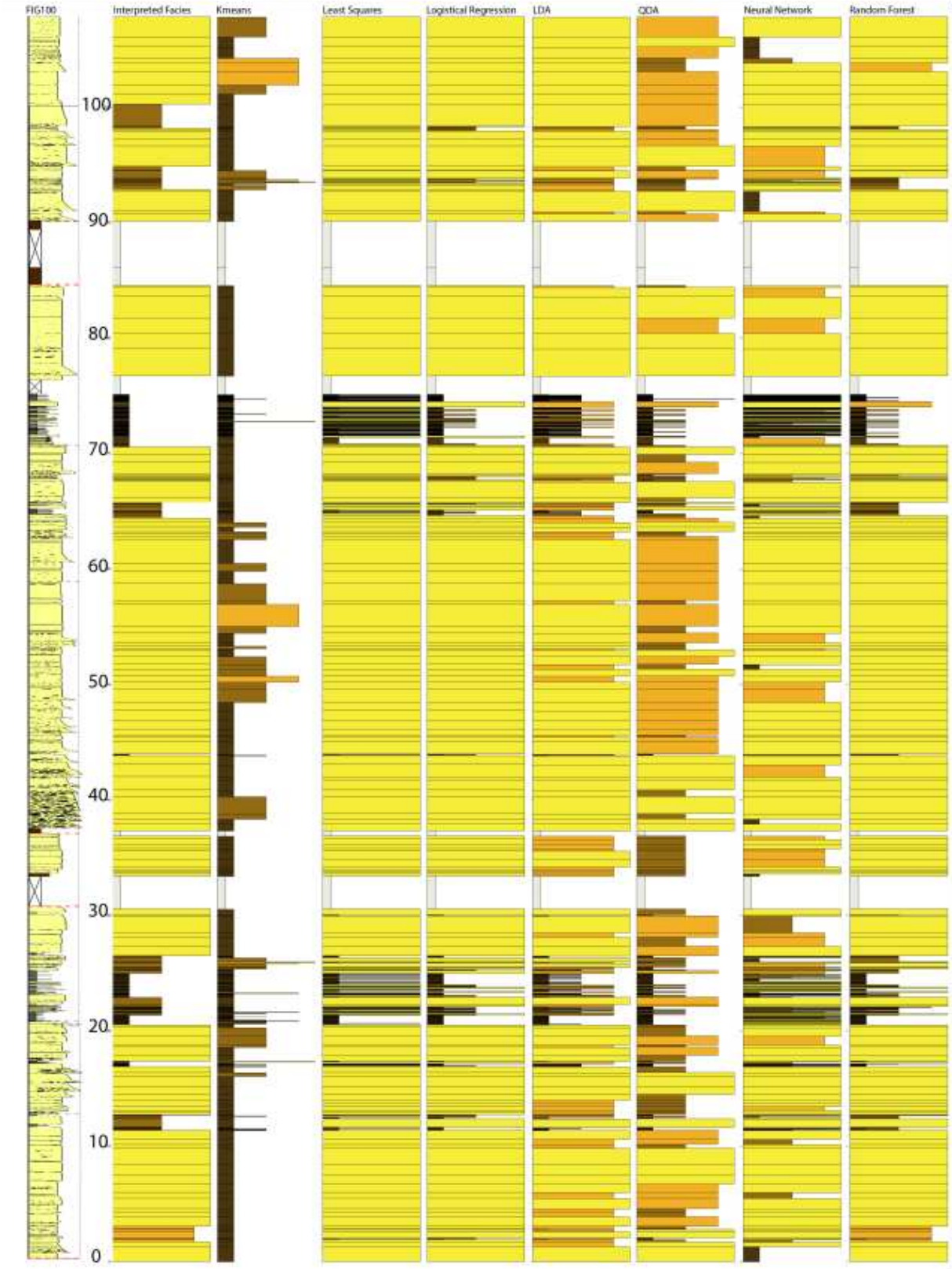

Figure 49: FIG100 -- Drafted measured section (column 1), interpreted facies (column 2), and seven prediction methods (columns 3-9).

5.3.3 Visual Results in Prediction Area

The Random Forest model is used to predict the facies interpretation for measured section moving up- and down-dip of the Laguna Figueroa outcrop/training data locality. These are MS PCS19 from Arroyo Picana and DCMS25 from Arroyo Hotel. The goal of this work was to expand the interpretation at Laguna Figeroa to other outcrops in the database. These sections show that the workflow is effective at mirroring the original interpretation in bed samples from similar deepwater outcrops. Notably, PCS19 represents a slightly different environment of deposition relative to Laguna Figeroa. Superior performance may relate to the similarity of sedimentary facies deposited in both similar environments.

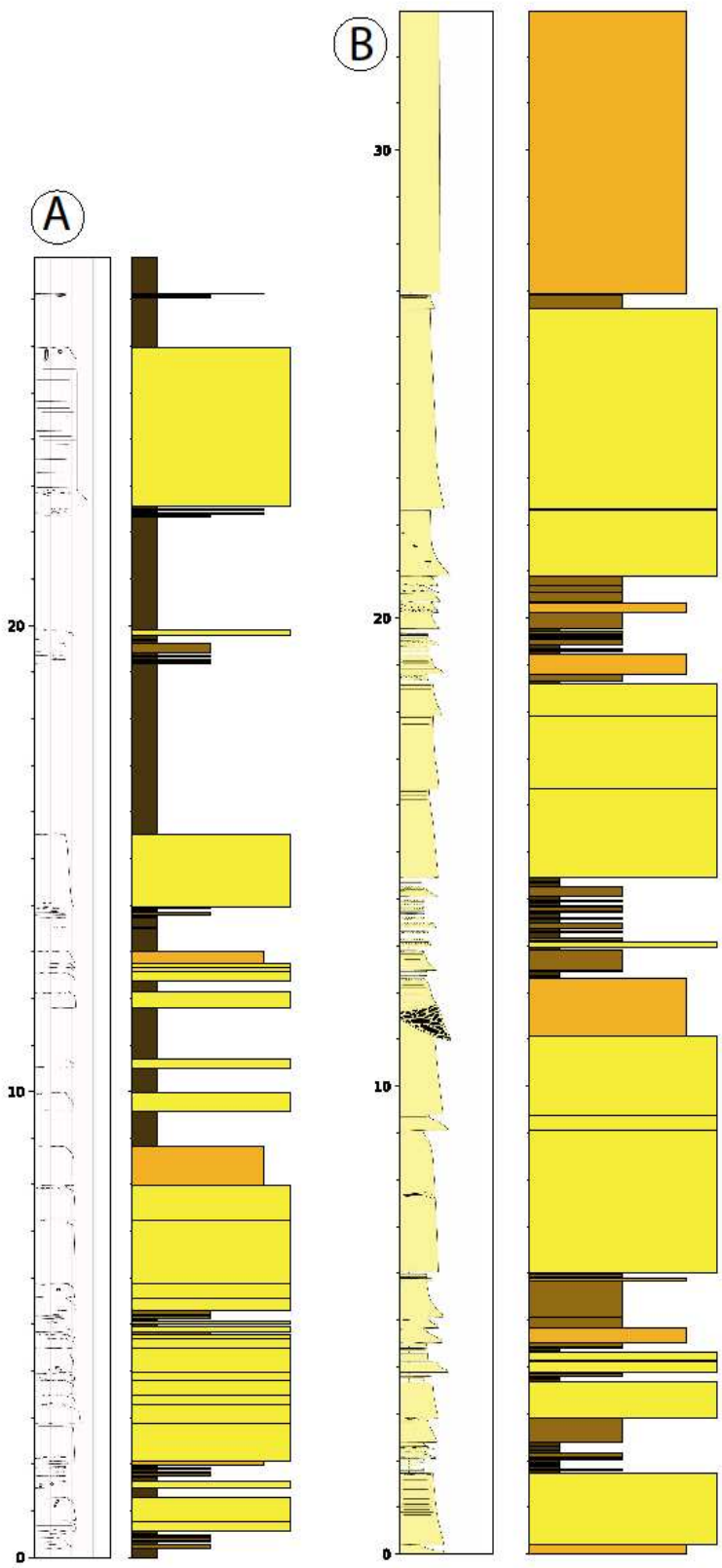

Figure 50: Random Forest prediction in A) up-dip location PCS19, and B) down-dip location DCMS25. See Figure 2 for the relative locations of these data to the training and testing area.

6. CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

In this thesis, machine learning methods are used to predict a geologic-facies interpretation from data collected in measured sections. Prediction of facies through machine learning provides an efficient workflow for consistent subsurface modeling in core-analogous 1-D data from field observations. Grain size observations (similar to core data) are sufficient feature data to interpret (predict) facies of deepwater strata from the Magallanes Basin of southern Chile.

The first objective of this study was to evaluate the quantitative performance of ML workflows for facies prediction. In classifying facies from manually collected grain size data, the method must respect the nature of the database. The statistical character of how we observe geologic data in the field negates the utility of several ML methods, fundamentally (K-Means, Least Squares Regression, Discriminant Analysis). Other methods provide reasonable accuracy (Logistic Regression, Neural Networks) but fail to distinguish statistically similar classes representing intermediate facies. Random Forest captures most of the facies-class variance of any method. The architecture of decision trees in general mimics the workflow of a field geologist and provides the best performance in the measured section facies interpretation application.

The second objective was to evaluate the subjective or qualitative performance in facies classification. This work found that the balanced performance in the Random Forest model generates the best quality graphical depictions of measured sections. Other methods (Least Squares and LDA/QDA) fail to classify many intermediate facies. Even where performance metrics are good (Logistic Regression and Neural Network), bias in overclassifying end-member facies obscures the underlying pattern in the sedimentary strata. In the case of measured section facies prediction, evenly distributed misclassifications are much less distracting or detrimental to the interpretation than a significant bias to one class or another.

Lastly, the third objective sought to gain insight into the nature of the database and datatype as they relate to ML methods. This objective also sought to expand the Laguna Figeroa Interpretation to other

outcrops in the database. Unsupervised techniques like K-Means clustering show that facies classes have significant spatial overlap in feature space. Samples of one class are often statistically similar to samples of another class. Further investigation using hyperplane-based delineation in LDA and QDA shows that simple partitioning methods cannot be used in this data. Later methods (Logistic Regression and Random Forest) suggest that certain features (BedThickness, gs_50) are more deterministic in facies classification than others (gs_min, gs_max). Additionally, we can see preferential clustering of some features around regular gradations (e.g. BedThickness at 0.1, 0.2, 0.3 meters) and grain size card references (Figs. 17 and 18). Altogether, the measured section derived grain size statistics represent a valuable resource for the predictive classification of sedimentary facies. Testing of this workflow on additional sections shows strong classification performance, matching the general character of each section (Fig. 50).

6.2 Future Work

Heterogenous methods are popular in recent research in the field of sedimentary facies prediction (Ippolito et al., 2021). Neural networks are common for similar classification problems. Adjacent research suggest further analysis of this study's neural network could yield higher intermediate facies performance than was observed in these experiments. A thorough review of hidden layer activation values would provide insights into this network's performance characteristics. Ippolito (2021) suggests that creating an integrated workflow across two or more of this studies methods may yield enhanced classification performance. The Least Squares method having particularly good selectivity in one class could indicate that using it in conjunction with other methods would be valuable.

Similar work in ML facies prediction from well log and seismic data have shown success using Support Vector Machine algorithms (SVMs) (Alexsandro et al., 2017; Liu et al., 2020). This method also represents a direction of further research in this outcrop data, facies-classification problem.

7. REFERENCES

Alexsandro, G.C., Carlos, A.D.P. and Geraldo, G.N., 2017, August. Facies classification in well logs of the Namorado oilfield using support vector machine algorithm. In 15th International Congress of the Brazilian Geophysical Society & EXPOGEF, Rio de Janeiro, Brazil, 31 July-3 August 2017 (pp. 1853–1858). Brazilian Geophysical Society.

Bhattacharya, S., Carr, T.R. and Pal, M., 2016. Comparison of supervised and unsupervised approaches for mudstone lithofacies classification: Case studies from the Bakken and Mahantango-Marcellus Shale, USA. Journal of Natural Gas Science and Engineering, 33, pp.1119–1133.

Biau, G., 2012. Analysis of a random forests model. The Journal of Machine Learning Research, 13(1), pp.1063–1095.

Bishop, C.M., 2006. Pattern recognition and machine learning. Springer google schola, 2, pp.5–43.

Bouma, A.H., 1962. Sedimentology of some flysch deposits. A graphic approach to facies interpretation, 168.

Breiman, L., 2001. Random forests. Machine learning, 45, pp.5–32.

Calinski, T., & Harabasz, J. 1974. "A dendrite method for cluster analysis". Communications in Statistics, 3(1): 1–27.

Cheng, B., Xiao, R., Wang, J., Huang, T., and Zhang, L., 2019, High frequency residual learning for multi-scale image classification: British Machine Vision Association, p. 1–14.

Covault, J. A., Romans, B. W., & Graham, S. A., 2009, Outcrop expression of a continental margin-scale shelf-edge delta from the Cretaceous Magallanes Basin, Chile. Journal of Sedimentary Research, 79(7–8), 523–539. https://doi.org/10.2110/jsr.2009.053

Dalziel, I.W.D., de Wit, M.J., and Palmer, K.F., 1974, Fossil marginal basin in the southern Andes: Nature, v. 250, p. 291–294.

Daniels, B.G., Hubbard, S.M., Stright, L., and Romans B.W., 2019, Downslope variability in deep-water slope channel fill and stacking patterns: Insights from outcrop and shallow seismic analogs: In ACE 2019 Annual Convention & Exhibition

Daniels, B.G., Hubbard, S.M., Stright, L. and Romans, B.W., 2024. Downslope variability in deep-water slope channel fill facies and stacking patterns. Marine and Petroleum Geology, 165.

Davies, D. L., and Bouldin, D. W. 1979. "A Cluster Separation Measure". IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1(2): 224–227.

Deptuck, M.E., Sylvester, Z., Pirmez, C., and O'Byrne, C., 2007, Migration-aggradation history and 3-D seismic geomorphology of submarine channels in the Pleistocene Benin-major Canyon, western Niger Delta slope: Marine and Petroleum Geology, v. 24, p. 406–433, doi:10.1016/j.marpetgeo.2007.01.005.

Duda, R.O. and Hart, P.E., 1973. Pattern classification and scene analysis (Vol. 3, pp. 731–739). New York: Wiley.

Escobar Arenas, L. C., 2023, Modeling of channel stacking patterns controlled by near wellbore modeling [M.S. thesis]: Fort Collins, CO, Colorado State University.

Evidently AI, 2024, How to interpret a confusion matrix for a machine learning model, Open-Source ML Monitoring and Observability. Available at: https://www.evidentlyai.com/classification-metrics/confusion-matrix (May 2024).

Fadokun, D.O., Oshilike, I.B. and Onyekonwu, M.O., 2020, August. Supervised and unsupervised machine learning approach in facies prediction. In SPE Nigeria Annual International Conference and Exhibition (p. D013S014R007). SPE.

Falivene, O., Auchter, N.C., de Lima, R.P., Kleipool, L., Solum, J.G., Zarian, P., Clark, R.W. and Espejo, I., 2022. Lithofacies identification in cores using deep learning segmentation and the role of geoscientists: Turbidite deposits (Gulf of Mexico and North Sea). AAPG Bulletin, 106(7), pp.1357–1372.

Fildani, A., and Hessler, A.M., 2005, Stratigraphic record across a retroarc basin inversion: Rocas Verdes-Magallanes Basin, Patagonian Andes, Chile: Bulletin of the Geological Society of America, v. 117, p. 1596–1614, doi:10.1130/B25708.

Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Annals of eugenics, 7(2), pp.179–188.

Hall, B., 2016, Facies classification using machine learning: The Leading Edge, v. 35, p. 906–909, doi:10.1190/tle35100906.1.

Halotel, J., Demyanov, V. and Gardiner, A., 2020. Value of geologically derived features in machine learning facies classification. Mathematical Geosciences, 52(1), pp.5–29.

Hosmer Jr., D. W., Lemeshow, S. & Sturdivant, RX 2013, Applied Logistic Regression, 3rd edn, Wiley, viewed May 2024, p.272.

Houshmand, N., GoodFellow, S., Esmaeili, K. and Calderón, J.C.O., 2022. Rock type classification based on petrophysical, geochemical, and core imaging data using machine and deep learning techniques. Applied Computing and Geosciences, 16.

Huang, Y., 2018. Sedimentary characteristics of turbidite fan and its implication for hydrocarbon exploration in Lower Congo Basin. Petroleum Research, 3(2), pp.189–196.

Hubbard, S.M., Covault, J.A., Fildani, A., and Romans, B.W., 2014, Sediment transfer and deposition in slope channels: Deciphering the record of enigmatic deep-sea processes from outcrop: Geological Society of America Bulletin, v. 126, no. 5–6, p. 857–871, doi:10.1130/B30996.1.

Hubbard, S.M., Romans, B.W. and Graham, S.A., 2008. Deep-water foreland basin deposits of the Cerro Toro Formation, Magallanes basin, Chile: architectural elements of a sinuous basin axial channel belt. *Sedimentology*, *55*(5), pp.1333–1359.

Ippolito, M., Ferguson, J. and Jenson, F., 2021. Improving facies prediction by combining supervised and unsupervised learning methods. Journal of Petroleum Science and Engineering, 200.

Jackson, A., Stright, L., Hubbard, S. M., & Romans, B. W., 2019. Static connectivity of stacked deep-water channel elements constrained by high-resolution digital outcrop models. AAPG Bulletin, 103(12), 2943–2973. https://doi.org/10.1306/03061917346

Jolliffe, I.T., 1990. Principal component analysis: a beginner's guide—I. Introduction and application. Weather, 45(10), pp.375–382.

Katz, H. R. 1963. Revision of Cretaceous Stratigraphy in Patagonian Cordillera of Ultima Esperanze, Magallanes Province, Chile. AAPG Bulletin, 47(3), 506–524.

Katz, H. R. 1972. Plate Tectonics and Orogenic Belts in the South-east Pacific. Nature, 237, 331.

Lee, H.D.P., 1952. Meteorologica by Aristotle, Book 1 Part 14. Translated from ancient Greek. Cambridge, MA: Harvard University Press.

Liu, X., Chen, X., Li, J., Zhou, X. and Chen, Y., 2020. Facies identification based on multikernel relevance vector machine. IEEE Transactions on Geoscience and Remote Sensing, 58(10), pp.7269–7282.

Macauley, R. V., & Hubbard, S. M. 2013. Slope channel sedimentary processes and stratigraphic stacking, Cretaceous Tres Pasos Formation slope system, Chilean Patagonia. Marine and Petroleum Geology, 41(1), 146–162. https://doi.org/10.1016/j.marpetgeo.2012.02.004

MacQueen, J., 1967, Some methods for classification and analysis of multivariate observations: University of California Press, Berkeley, California, Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, v. 1, p. 281–297.

Martin, T., Meyer, R. and Jobe, Z., 2021. Centimeter-scale lithology and facies prediction in cored wells using machine learning. Frontiers in Earth Science, 9.

Mayall, M., Jones, E., and Casey, M. 2006. Turbidite channel reservoirs - Key elements in facies prediction and effective development.

McCulloch, W.S., and Pitts, W.A., 1943, Logical calculus of the ideas immanent in nervous Activity: Bulletin of Mathematical Biophysics, v. 5, p. 115–133, doi:10.1007/BF02478259.

McHargue, T., Pyrcz, M. J., Sullivan, M. D., Clark, J. D., Fildani, A., Romans, B.W., and Drinkwater, N.J., 2011, Architecture of turbidite channel systems on the continental slope: Patterns and predictions: Marine and Petroleum Geology, v. 28, p. 728–743, doi:10.1016/j.marpetgeo.2010.07.008.

Meirovitz, C. D., Stright, L., Hubbard, S. M., & Romans, B. W. 2020. The Influence of Inter- and Intra-channel Architecture on Deep-water Turbidite Reservoir Performance. Petroleum Geoscience. https://doi.org/https://doi.org/10.1144/petgeo2020-005

Meirovitz, C. D., 2024, From Rocks to Models: Use of Outcrop Characterization for Improved Reservoir Performance Characterization [Ph.D. dissertation]: Salt Lake City, UT, The University of Utah.

Mohammadpoor, M., and Torabi, F., 2019, Big Data analytics in oil and gas industry: An emerging trend: Petroleum, doi:10.1016/j.petlm.2018.11.001.

Na, B., and Fox, G.C., 2019, Object detection by a super-resolution method and a convolutional neural networks: Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018, p. 2263–2269, doi:10.1109/BigData.2018.8622135.

Neter, J., Kutner, M.H., Nachtsheim, C.J. and Wasserman, W., 1996. Applied linear statistical models.

Pemberton, E. A. L., Stright, L., Fletcher, S., & Hubbard, S. M. 2018. The influence of stratigraphic architecture on seismic response: Reflectivity modeling of outcropping deepwater channel units. Interpretation, 6(3), T783–T808. https://doi.org/10.1190/int-2017-0170.1

Qi, L. and Carr, T.R., 2006. Neural network prediction of carbonate lithofacies from well logs, Big Bow and Sand Arroyo Creek fields, Southwest Kansas. Computers & Geosciences, 32(7), pp.947–964.

Posamentier, H.W. and Vail, P.R., 1988. Eustatic controls on clastic deposition II—sequence and systems tract models.

Ruetten, A., 2021. Evaluating the impact of hierarchical deep-water slope channel architecture on fluid flow behavior, Cretaceous Tres Pasos Formation, Chile [M.S. thesis]: Fort Collins, CO, Colorado State University.

Robinson, S. 2023. Introduction to neural networks with Scikit-Learn, 2013–2024 Stack Abuse. https://stackabuse.com/introduction-to-neural-networks-with-scikit-learn/

Romans, B.W., Fildani, A., Hubbard, S.M., Covault, J.A., Fosdick, J.C., and Graham, S.A, 2011, Evolution of deep-water stratigraphic architecture, Magallanes Basin, Chile: Marine and Petroleum Geology, v. 28, p. 612–628, doi:10.1016/j.marpetgeo.2010.05.002.

Rousseeuw, Peter J. 1987. "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis". Journal of Computational and Applied Mathematics, 20: 53–65.

Saporetti, C.M., da Fonseca, L.G., Pereira, E., and de Oliveira, L.C, 2018, Machine learning approaches for petrographic classification of carbonate-siliciclastic rocks using well-logs and textural information: Journal of Applied Geophysics, v. 155, p. 217–225, doi:10.1016/j.jappgeo.2018.06.012.

Scott Long, J., 1997. Regression models for categorical and limited dependent variables. Advanced quantitative techniques in the social sciences, 7.

Southern, S.J., Stright, L., Jobe, Z.R., Romans, B., and Hubbard, S., 2017, The stratigraphic expression of slope channel evolution: insights from qualitative and quantitative assessment of channel fills from the Cretaceous Tres Pasos Formation, southern Chile: AAPG Annual Convention, Houston, TX, April 3–5, 2017.

Sprague, A.R.G., Garfield, T.R., Goulding, F.J., Beaubouef, R.T., Sullivan, M.D., Rossen, C., Campion, K.M., Sickafoose, D.K., Abreu, V., Schellpeper, M.E. and Jensen, G.N., 2005. Integrated slope channel depositional models: the key to successful prediction of reservoir presence and quality in offshore West Africa. Veracruz, México, Colegio de Ingenieros Petroleros de México, pp.1–13

Strekeisen, A., 2020. Turbidite: The Bouma sequence, modified slightly from Bouma 1962.

Suykens, J.A. and Vandewalle, J., 1999, July. Multiclass least squares support vector machines. In IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No. 99CH36339) (Vol. 2, pp. 900–903). IEEE.

Vento, N., 2020, Hypothesis-based machine learning for deep-water channel systems [M.S. thesis]: Fort Collins, CO, Colorado State University, 168 p.

Vincent, P., 1986. Differentiation of modern beach and coastal dune sands—a logistic regression approach using the parameters of the hyperbolic function. Sedimentary Geology, 49(3-4), pp.167–176.

Walker, R.G., 1978. Deep-water sandstone facies and ancient submarine fans: models for exploration for stratigraphic traps. AAPG Bulletin, 62(6), pp.932–966.

Wilson, T. J., 1991, Transition from back-arc to foreland basin development in the southernmost Andes: Stratigraphic record from the Ultima Esperanza District, Chile: Geological Society of America Bulletin, v. 103, p. 98–111, doi:10.1130/00167606(1991)103<0098:TFBATF>2.3.CO;2

Zhang, J., Ambrose, W. and Xie, W., 2021. Applying convolutional neural networks to identify lithofacies of large-n cores from the Permian Basin and Gulf of Mexico: The importance of the quantity and quality of training data. Marine and Petroleum Geology, 133.

Zheng, W., Tian, F., Di, Q., Xin, W., Cheng, F. and Shan, X., 2021. Electrofacies classification of deeply buried carbonate strata using machine learning methods: A case study on ordovician paleokarst reservoirs in Tarim Basin. Marine and Petroleum Geology, 123.

Zhao, T., Jayaram, V., Roy, A. and Marfurt, K.J., 2015. A comparison of classification techniques for seismic facies recognition. Interpretation, 3(4), pp.SAE29–SAE58.