N-TARP: A RANDOM PROJECTION BASED METHOD FOR SUPERVISED AND UNSUPERVISED MACHINE LEARNING IN HIGH-DIMENSIONS WITH APPLICATION TO EDUCATIONAL DATA ANALYSIS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Tarun Yellamraju

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

May 2019

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Mireille Boutin, Chair

School of Electrical and Computer Engineering

Dr. Fengqing M. Zhu

School of Electrical and Computer Engineering

Dr. Jonathan R. Peterson

Department of Mathematics

Dr. Peter A. Bermel

School of Electrical and Computer Engineering

Approved by:

Dr. Pedro Irazoqui

Head of the School Graduate Program

For Mom and Dad

ACKNOWLEDGMENTS

This dissertation is based upon work supported by the National Science Foundation under Grant No. EEC-1544244 and EEC-1826099. I would like to thank the following people for helping me complete this dissertation.

First, I would like to thank my advisor Prof Mireille Boutin, for her guidance, patience, support, encouragement and motivation throughout the four years of my PhD. I have learned a great deal from her and can think of no one better to have worked with for my PhD. Thanks for everything Prof Mimi!

Second, I would like to thank Prof Alejandra Magana for supervising the Educational Data Analysis and the development of the associated rubric presented in this dissertation. I would also like to thank Jonas Hepp for his significant contribution to the Benchmarks framework presented in this dissertation.

Third, I would like to thank my colleagues and fellow grad students, Kelsie, Eli, Emma, Sri Kalyan, Deena and Amrutha for their friendship and helpful advice over the course of my PhD.

Finally, I would like to thank my parents Jayashree and Sasidhar, for their support and encouragement. I would like to thank my mom for pushing me to chase the best opportunities in my education. I would like to thank my dad for cultivating a sense of curiosity for science and math in me since childhood. I would not be able to achieve what I have without them.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

ABSTRACT

Yellamraju, Tarun Ph.D., Purdue University, May 2019. n-TARP: A Random Projection based Method for Supervised and Unsupervised Machine Learning in High-dimensions with Application to Educational Data Analysis. Major Professor: Mireille Boutin.

Analyzing the structure of a dataset is a challenging problem in high-dimensions as the volume of the space increases at an exponential rate and typically, data becomes sparse in this high-dimensional space. This poses a significant challenge to machine learning methods which rely on exploiting structures underlying data to make meaningful inferences. This dissertation proposes the n-TARP method as a building block for high-dimensional data analysis, in both supervised and unsupervised scenarios.

The basic element, n-TARP, consists of a random projection framework to transform high-dimensional data to one-dimensional data in a manner that yields point separations in the projected space. The point separation can be tuned to reflect classes in supervised scenarios and clusters in unsupervised scenarios. The n-TARP method finds linear separations in high-dimensional data. This basic unit can be used repeatedly to find a variety of structures. It can be arranged in a hierarchical structure like a tree, which increases the model complexity, flexibility and discriminating power. Feature space extensions combined with n-TARP can also be used to investigate non-linear separations in high-dimensional data.

The application of n-TARP to both supervised and unsupervised problems is investigated in this dissertation. In the supervised scenario, a sequence of n-TARP based classifiers with increasing complexity is considered. The point separations are measured by classification metrics like accuracy, Gini impurity or entropy. The performance of these classifiers on image classification tasks is studied. This study provides an interesting insight into the working of classification methods. The sequence

of n-TARP classifiers yields benchmark curves that put in context the accuracy and complexity of other classification methods for a given dataset. The benchmark curves are parameterized by classification error and computational cost to define a benchmarking plane. This framework splits this plane into regions of "positive-gain" and "negative-gain" which provide context for the performance and effectiveness of other classification methods. The asymptotes of benchmark curves are shown to be optimal (i.e. at Bayes Error) in some cases (Theorem 2.5.2).

In the unsupervised scenario, the n-TARP method highlights the existence of many different clustering structures in a dataset. However, not all structures present are statistically meaningful. This issue is amplified when the dataset is small, as random events may yield sample sets that exhibit separations that are not present in the distribution of the data. Thus, statistical validation is an important step in data analysis, especially in high-dimensions. However, in order to statistically validate results, often an exponentially increasing number of data samples are required as the dimensions increase. The proposed n-TARP method circumvents this challenge by evaluating statistical significance in the one-dimensional space of data projections. The n-TARP framework also results in several different statistically valid instances of point separation into clusters, as opposed to a unique "best" separation, which leads to a distribution of clusters induced by the random projection process.

The distributions of clusters resulting from n-TARP are studied. This dissertation focuses on small sample high-dimensional problems. A large number of distinct clusters are found, which are statistically validated. The distribution of clusters is studied as the dimensionality of the problem evolves through the extension of the feature space using monomial terms of increasing degree in the original features, which corresponds to investigating non-linear point separations in the projection space.

A statistical framework is introduced to detect patterns of dependence between the clusters formed with the features (predictors) and a chosen outcome (response) in the data that is not used by the clustering method. This framework is designed

to detect the existence of a relationship between the predictors and response. This framework can also serve as an alternative cluster validation tool.

The concepts and methods developed in this dissertation are applied to a real world data analysis problem in Engineering Education. Specifically, engineering students' Habits of Mind are analyzed. The data at hand is qualitative, in the form of text, equations and figures. To use the n-TARP based analysis method, the source data must be transformed into quantitative data (vectors). This is done by modeling it as a random process based on the theoretical framework defined by a rubric. Since the number of students is small, this problem falls into the small sample high-dimensions scenario. The n-TARP clustering method is used to find groups within this data in a statistically valid manner. The resulting clusters are analyzed in the context of education to determine what is represented by the identified clusters. The dependence of student performance indicators like the course grade on the clusters formed with n-TARP are studied in the pattern dependence framework, and the observed effect is statistically validated. The data obtained suggests the presence of a large variety of different patterns of Habits of Mind among students, many of which are associated with significant grade differences. In particular, the course grade is found to be dependent on at least two Habits of Mind: "computation and estimation" and "values and attitudes."

1. INTRODUCTION

In the recent past, the progress made in machine learning techniques has been immense. Coupled with tremendous growth in computational power, the capabilities of machine learning techniques have reached new heights. These techniques are now able to handle increasingly larger quantities of complex data. Data science and analysis has also become increasingly popular over the last decade or so. Accompanying the progress in machine learning and data science has been a growth in the amount of data being collected, both in terms of quantity and dimensions. It is not uncommon to find high-dimensional datasets where the number of dimensions is larger than the number of samples. This leads to serious challenges in analyzing and understanding the structures underlying the data. Machine learning techniques have faced difficulties in working with very high-dimensional data relative to lower-dimension scenarios.

The growth in dimensionality of data has accentuated the phenomenon best known as the "Curse of Dimensionality" [1,2], which highlights the challenges associated with the exponentially increasing volume associated with increasing dimensions of data. In higher dimensions, data tends to be sparse and similarity metrics are not very effective at finding significant differences, as small changes in relevant dimensions can be hidden under cumulative noise in all the other dimensions. Thus, distance based methods that are perfectly valid for lower-dimensional scenarios perform poorly as the data becomes sparser with increasing dimensions. This has led to an approach to transform high-dimensional data to lower-dimensions. The techniques associated with this approach are termed as dimension reduction techniques. These methods are commonly used as a pre-processing step before applying other conventional machine learning techniques on the lower-dimensional data.

1.1 Dimension Reduction

A popular approach to dimension reduction is the Principal Component Analysis (PCA) technique [3], which aims to preserve distances by eliminating directions of low variance from the high-dimensional data. This is particularly useful for images that can be represented by far fewer dimensions than the total number of pixels. PCA acts as a pre-processing step that reduces the dimensionality of the problem, making it more tractable. Extensions of the PCA framework include kernel PCA [4] and non-linear PCA [5].

A more sophisticated approach to dimension reduction in a point proximity structure preservation scenario is through nonlinear dimensionality reduction techniques and manifold learning [6,7]. The premise for this approach is that data lies on a lower-dimensional manifold embedded in a high-dimensional space. For example, consider a thin spiralling band that has 2 dimensions in 3D space. The band can be considered a 2D manifold embedded in 3D space. Another example is a Mobius strip in 3D. The manifold learning methods seek to identify the underlying manifold in order to represent the high-dimensional data in the lower-dimensional manifold, thereby making the problem more tractable for other machine learning methods.

Research in Manifold Learning has resulted in many different techniques. For example, Isomap [8] computes geodesic distances on a manifold, which are utilized to determine positions of data points on the manifold. Another example is LLE (Locally Linear Embedding) [9] which seeks to express a data point as a linear combination of its neighbors. It then aims to determine a low-dimensional embedding such that the same linear combinations are maintained. Subspace clustering methods like the recent SSC-OMP [10] (Sparse Subspace Clustering - Orthogonal Matching Pursuit) are based on the premise that points lie on sub-manifolds and the sub-manifold structure can be exploited to form clusters. They depend on point-proximity structures and rely on the "self-expressiveness" concept, which in simple terms relates to a point expressed as a sparse linear combination of other points. Autoencoders [11] are neural

networks trained as an identity function. Half of the network is used for mapping from high-dimensions to low-dimensions and the other half is used for inverting this transformation. The intermediary low-dimensional space constitutes the manifold.

A few among many more popular Manifold Learning techniques include Diffusion Maps [12], Hessian Eigenmaps [13], Laplacian Eigenmaps [14], Semidefinite Embedding [15], Continuum Isomap [16], Tangent Space Alignment [17], Maximum Variance Unfolding [18] and t-SNE (t-distributed Stochastic Neighborhood Embedding) [19]. Most of the above methods have a common component of spectral decomposition, which can be slow in high-dimension scenarios. Some recent studies [20–24] have made progress in speeding up spectral decomposition in the Manifold Learning scenario.

Yet another approach to dimension reduction is through random projections. The goal is to transform data in high-dimensional space $\mathbb{R}^p$ to a lower-dimensional space $\mathbb{R}^{p'}$ where $p' < p$ through a random projection model, i.e. take the inner product with random vectors. The application of random projections to dimensionality reduction [25] is motivated by the Johnson-Lindenstrauss lemma [26] which relates to preservation of structure in high-dimensional space when transformed to lower dimensional spaces by preserving point distances. Through this approach, the data space is transformed from high-dimensions to lower-dimensions to make the problem more tractable, similar to what is achieved through manifold learning.

This idea has led to the use of random projections as a pre-processing step in several supervised and unsupervised machine learning problems. The concept of random projections [27,28] has previously been proposed as a basis for dimensionality reduction techniques [25, 29] with several applications in classification and clustering. For instance, [30–33] use random projections to reduce high-dimensional data into lower dimensional feature vectors for use with classifiers. Random projections have also been used in an iterative manner to find visual patterns of structure in data through dimension reduction [34]. Clustering methods based on random projections

like [35] project data to a lower dimensional space of dimensions greater than one followed by a point proximity based cluster assignment.

1.2 One-dimensional Random Projection: n-TARP

The conventional random projection techniques discussed above aim to preserve structures in high-dimensions in a lower-dimensional space, followed by application of point proximity metric based machine learning methods. In this dissertation, we propose a new random projection technique called n-TARP where TARP stands for Thresholding After Random Projection. The core idea of how this method works is quite simple: data is projected onto a randomly generated line through the origin. The projection is repeated until the projection points are found to form a bi-modal distribution.

This method is an extreme case of dimension reduction, wherein we project high-dimensional data down to just one dimension without concern for preserving distances. This is in contrast with the methods discussed previously. The kind of high-dimensional structure we are trying to exploit with this technique is quite different to the more conventional methods discussed previously. This method does not aim to preserve the high-dimensional structures, rather, its aim is to extract structures that are hidden in high-dimensions which can manifest as a point separation upon projection onto a trivial random 1D subspace (simplified illustrative example in Figure 1.1, reproduced from Chapter 3). Previous work [36, 37] has shown the effectiveness of this idea and the presence of hidden structure in data that is manifested as point separations with a high probability.

Our proposed n-TARP technique serves as a basic building block that can be used in both supervised and unsupervised learning scenarios. Several units of this building block can be combined to form more complex models as can be seen in [36–38]. We have also observed that the n-TARP method yields several different separation criteria instead of some unique "best" criteria [39, 40]. This leads to the interesting

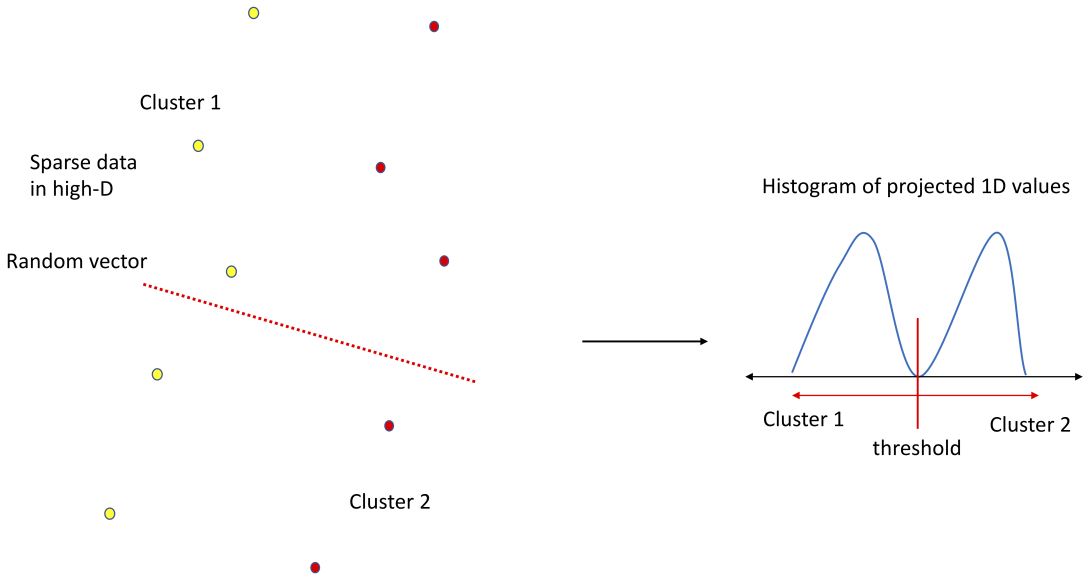


Fig. 1.1. Scenario where the data distributions are sparse in the original high-dimensional space and projection onto a random vector yields a binary separation

question of what do the various separation criteria actually represent and what kind of structure yields these criteria. We will go into further details and answer some of the questions regarding this method and its applications in the chapters to follow.

1.3 Supervised Learning in High-dimensions

One of the core challenges in supervised learning is to find the optimal classification method that achieves the lowest possible error. However, determining the lowest possible error is a challenge in itself. Determining this quantity requires complete knowledge of the probability distributions underlying the marginal class distributions of the data. Bayes Error is the probability of error of a classifier that assigns classes following Bayes Decision Rule and assuming full knowledge of the class (marginal) probability densities and priors. It is proven to be optimal, in the sense that no classifier can obtain a probability of error less than Bayes Error. Thus, Bayes Error

is a benchmark for the probability of error of a classifier which corresponds to the minimum possible probability of error that can be achieved for the given feature vector [3].

Computing Bayes Error from training data is difficult, especially in high-dimension. The task involves estimating the class probability densities and priors, and integrating these functions over potentially complex boundaries. While the case of univariate normally distributed class densities can be computed analytically, high-dimensional distributions (even normal ones) must be handled numerically. Computing Bayes Error when the problem does not follow any common probability model is even more difficult. There have been efforts to approximate Bayes Error and bound it. For the special case of binary classification where both class densities are multivariate Gaussians, Fukunaga et al [41] have presented an explicit mathematical expression that approximates Bayes Error. The Chernoff bound [42] and the Bhattacharyya bound [43] are well-known upper bounds; some closed form expressions for these bounds exist for the special case of Gaussian densities of the underlying classification data, but they are not necessarily tight or particularly insightful for distributions that deviate markedly from a Gaussian [3].

Therefore, while it would be ideal to know if a given classification method achieves Bayes Error, it is not feasible to evaluate. This issue becomes more complicated as the number of dimensions of the data becomes larger, where reliably estimating the probability distributions becomes quite challenging in addition to the data being sparse in high-dimensions.

One avenue to follow in this scenario is to find an upper bound on the Bayes Error. With that in mind, this dissertation introduces a benchmarking framework built with n-TARP classifiers. The n-TARP units are combined in a hierarchical tree structure that leads to a sequence of benchmarks with increasing model complexity. Each benchmark in the sequence is associated with a performance metric (error rate in this case). Hence, the benchmarks form curves in a space of error rate vs model complexity (computational cost is used as a proxy). An example illustration of the

benchmark curves is shown in Figure 1.2, reproduced from Chapter 2. While the tree based n-TARP structures are generated at random, they are completely different from the popular Random Forests technique [44]. The randomness with the n-TARP benchmark trees is derived from the random projection process, while in the case of Random Forests, it is based on dividing data into random subsets to train each tree of the forest.

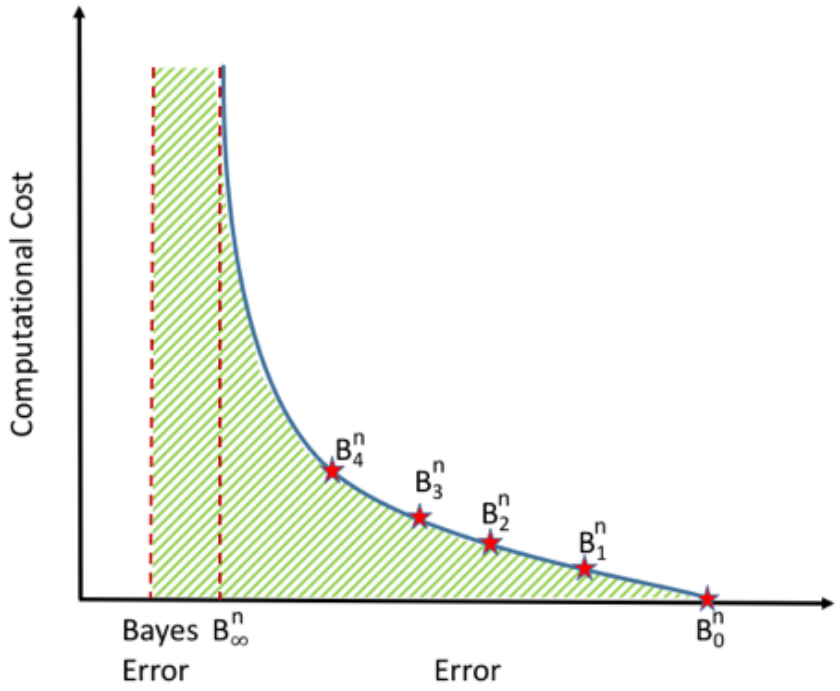


Fig. 1.2. The Benchmark Plane: The sequence of n-TARP classifiers $B_0^n, B_1^n, B_2^n, B_3^n, \ldots$ for some $n \in \mathbb{N}$ with their associated error rates and model complexity (computational cost serves as a proxy) define a benchmark curve with asymptote at B_∞^n that serves as an upper bound to the Bayes Error.

This framework of benchmark curves can be used to put in perspective the performance of a given classification method, like a DNN (Deep Neural Network) for example. The DNN will correspond to a point on the benchmark curve plane. The suitability of this method for the given dataset can be judged by its position relative

to the benchmark curves. An illustration of the benchmark curves is shown in Figure 1.2. As will be explained in Chapter 2, the benchmark curve divides the plane into 3 regions corresponding to different types of relative performances.

The goal of situating classification methods like the DNN with respect to the benchmark curves is to gain a better understanding of the trade-offs between model complexity and performance metrics. For instance, DNNs are inherently very complex models that seem to yield good results, but is their level of complexity justified? As model complexity increases, especially in high-dimensions, over-fitting the data available becomes a major risk. Gaining a better understanding of this trade-off can help in guiding choices of classifiers for a specific learning task. It also helps in understanding and minimizing the risks associated with that choice in terms of performance and generalization. These ideas are developed and discussed in detail in Chapter 2.

1.4 Unsupervised Learning in High-dimensions

As discussed previously, data science and analysis using machine learning techniques has become a very active area of research. Clustering data is a key ingredient of this process and is a challenging problem in high-dimensions. This is so because the high-dimensional data tends to be sparse, which causes the clustering problem to become ill-conditioned. Consequently, moving the points slightly can result in a seemingly different clustering structure. On the other hand, there is no reason to expect that there is a unique way to meaningfully cluster the data. In other words, several different cluster assignments may be valid instead of a unique one. This motivates our use of the n-TARP based clustering method to find several possible clustering structures, rather than a single one.

Another challenge associated with clustering in high-dimensions is the statistical validation of the formed clusters, as groupings found among samples may not be statistically meaningful. To validate the high-dimensional data groups, we usually

require a much larger number of samples than available to run multivariate statistical tests. For instance, estimating a covariance matrix (size 100×100) for a cluster of $m = 50$ data samples in $p = 100$ dimensional space is fraught with reliability and error concerns and can result in an ill-conditioned matrix.

The Hotelling T^2 statistic is a popular choice for multivariate statistical tests, that uses the covariance matrix. Specifically, it requires the inverse of the covariance matrix. However, in the $m < p$ scenario of high-dimensional problems, this matrix is singular and the test is not well defined. This behaviour is studied in [45], which also finds poor performance even when m and p are close. Various approaches have been proposed [46–48] that aim to estimate a diagonal covariance matrix as a work around. Similar to the motivation of dimension reduction techniques discussed previously, a random projection based approach has been developed [49] that aims to preserve distances in a lower-dimensional space (of dimension greater than 1) before employing the Hotelling T^2 test in the projected space. While the approach of [49] uses random projections, the lower dimensional space is far from one-dimensional, with a single random projection being used and aims to preserve distances, in contrast to n-TARP. This indicates exploitation of point proximity structures as opposed to hidden point separation structures.

The statistical validation approaches in the literature are not compatible with the unsupervised scenario combined with the n-TARP framework. To address the issue of statistical validity specific to the setup of n-TARP, we have proposed evaluation of statistical significance of the clusters in the one dimensional space of projected data associated with our n-TARP based clustering method, thereby bypassing the challenges associated with statistical validation in high-dimensions. Our statistical test procedure consists of two stages: Training and Validity Testing, wherein, half of the high-dimensional data is used in the n-TARP framework to find a point separation structure that yields clusters. The other half of the data is used to validate the separation criteria in a permutation test scenario with Monte-Carlo simulations [50]. A similar idea of evaluating statistical significance in a one-dimensional projected

space can be found in [51]. However, this work requires labels for data. Specifically, it involves repeated training to find linear binary classifiers, while using all of the available data. Consequently, while the one-dimensional analysis appears similar, this approach is fundamentally different from our n-TARP based statistical analysis framework, which will be discussed in further detail in Chapter 3.

The non-deterministic nature of the n-TARP method induces a stochastic distribution of statistically valid clusters. It is not feasible to study the characteristics of every individual grouping of the data resulting from n-TARP. Hence, we instead propose to characterize and study the collective empirically determined distribution of the clusters.

The n-TARP method is designed to find linear separations in high-dimensional data. However, the feature space can be extended to include monomial terms of increasing degree that changes the dimensionality of the problem. This allows n-TARP to find non-linear separations in the data, which may result in different distributions of the clustering assignments.

The clustering method and the approach for statistical validation will be discussed in detail in Chapter 3. While the clustering method presented in this dissertation is applicable to any sample size, we focus on a small sample size (20-30) setting which is motivated by a real world data problem that will be discussed in Chapter 5. While the sample sizes can vary quite a bit depending on the data being considered, we are almost always going to be in a small sample scenario since it is very unlikely that we will have enough data to not have a sparse distribution in high-dimensional space.

1.5 Predictor Response Patterns and Cluster Validation

The previous sections have introduced the utility of n-TARP based clustering. The cluster assignments resulting from n-TARP can be viewed as random variables induced by the random projection process. So the clusters can themselves be considered a random process, with several realized observations in our experiments. Given

the probabilistic nature of the clusters, it would be interesting to understand the significance of the resulting clusters. How does one interpret these clusters, or more generally, separations in a projected random one-dimensional space? Do the clusters have any meaning? These are interesting questions that we aim to address in a statistical manner through our proposed pattern dependence framework. This framework can also serve as an alternative means of validating clusters.

We have previously designed one-dimensional statistical tests to check the validity of the clusters and will employ a different approach for cluster evaluation. Consider the high-dimensional data sample as a multivariate vector, where the entries of a vector are considered as a collection of predictor variables. A separate variable of interest which is not part of the data that is clustered, is assigned as a response variable. Basically, the clustering method has no information about the response variable when the clusters are formed. We then investigate if the separation of data into clusters has any effect on the response variable. We do not seek a functional relationship between the predictors and response, rather, we investigate whether there is a dependence between the patterns underlying the clusters and the response variable. This is in contrast to approaches like regression which assume the existence of a relationship, enforce a functional form and determine parameters to best optimize some error metric. Examples of the regularized regression approaches include Ridge Regression [52], LASSO [53] (Least Absolute Shrinkage and Selection Operator) and Elastic Nets [54]. Instead, our goal with the pattern dependence framework is to determine the existence of a relationship.

It is possible that any effects we see are no different from the effects one would observe by partitioning the data at random. To check whether this is the case, our framework involves a statistical hypothesis test that considers empirically determined probability distributions of a measure of the response variable's behavior when associated with clusters. We test against a null hypothesis of random partition of samples into groups. Our pattern dependence framework along with the associated statistical test will be discussed in detail in Chapter 4.

1.6 Putting it all together in a Real World Problem

The concepts and ideas introduced over the previous sections are applied to a real world data analysis problem. A majority of the work presented in this dissertation was motivated by the problem of analyzing qualitative source data in the Engineering Education research domain with quantitative and statistical machine learning methods. The context for this work was qualitative student data based on peer reviewed presentations in a course on digital signal processing. The data consisted of excerpts of text, equations and figures. To run our n-TARP based data analysis methods, we must first convert this qualitative data to quantitative data that is compatible with the n-TARP techniques.

The qualitative data is mapped to quantitative data through the use of a rubric developed and refined in a cross-disciplinary effort with colleagues at Purdue University [55]. The rubric is based on Habits of Mind [56], which are modes of thinking required for STEM (Science Technology Engineering Mathematics) students to become effective problem solvers capable of transferring such skills to new contexts. The qualitative data is annotated by hand following the rubric. The annotations are subsequently modelled in a probabilistic framework that yields feature vectors that can be used in our n-TARP based analysis.

The resulting data fits into the scenario of small sample high-dimensional data, which is the scenario we have focused on for developing the tools and concepts introduced in the previous sections. The n-TARP clustering method is used to investigate the structures underlying the education data. We further investigate the statistical validity of the clusters we obtained and what meaning these clusters have. In particular, the student grade in the course is designated as a response variable while the data we acquired is treated as predictor data under the pattern dependence framework.

In Chapter 5, we present the details of the qualitative data we acquired, the process of mapping it to quantitative data and provide the context for our clustering results and what the clusters signify and represent. Results of our n-TARP based

unsupervised learning method are presented in the context of how student behavior and learning skills affect their course grade at the end of the semester under the pattern dependence framework. We hope that these results will eventually help identify students having difficulties in learning, generally help students achieve better results in their education, and aid in tailoring course design to improve student learning and to introduce mid-course interventions to optimize student learning and performance.

2. BENCHMARKS

The contents of this chapter appear in [38].

2.1 Introduction

In machine learning, the two-class pattern recognition paradigm is often formulated in terms of a discriminant function $g(x)$ whose sign should determine the class of the data point x with a high probability of accuracy. The entries of $x \in \mathbb{R}^p$ are features used to represent each object to be categorized, and the function $g(x)$ is estimated numerically using a training set of pre-labeled feature points $x_1, \ldots, x_N \in \mathbb{R}^p$. The function g(x), often concocted using a complicated non-linear combination of the original features x, can be viewed as a new feature. It is a distinguishing feature for the classes considered, and many efforts have been spent to develop powerful methods to effectively find good discriminant functions $g(x)$. Deep neural networks are a great example of such [57].

Pattern recognition in high-dimension can be quite challenging. Indeed, finding such a distinguishing feature $g(x)$ can be quite difficult when the feature vectors $x \in \mathbb{R}^p$ are in a space of high-dimension p. Unless there are several good choices of features, finding a good $g(x)$ is like looking for a needle in a haystack: without a good trick such as a prior model assumption or other prior information, one needs a considerable amount of numerical work to be successful.

However, recent work suggests that high-dimensional data representing images or videos have a lot of structure, "so much so that a mere random projection of the data is likely to uncover some of that structure" [36]. There is some evidence that this phenomenon extends to other types of high-dimensional data [37]. Thus, it is quite conceivable that there are many high-dimensional pattern recognition problems

where several and easily identifiable good choices of $g(x)$ exist. Such cases are much easier to deal with from a numerical and computational perspective.

It is not necessarily easy to tell from training data how many good classifiers $g(x)$ exist for a given pattern recognition problem, especially in high-dimension. More generally, there are no good all purpose methods for determining the difficulty of a given pattern recognition problem from training data. However, having such information would be useful, as it could guide the choice of method used to attack such problems in practice. In the context of machine learning research, that information could also be used to select good datasets for testing new pattern recognition methods: as good results in an easy dataset could give false impressions of success, one should develop new methodologies with the hard datasets in mind and focus the testing efforts on such datasets.

In the following, we provide a framework for quantifying the difficulty of a given pattern recognition problem. Specifically, we propose sequences of upper bounds for the probability of error of a binary classification. The bounds are obtained using simple heuristics based on random projection of the data on a one-dimensional linear subspace; the decisions are made by thresholding the resulting one-dimensional features. If a sophisticated pattern recognition method produces a worse outcome than these bounds, one must conclude that it is ill-suited for that particular pattern recognition problem. In other words, in order to justify its complexity and computational cost, a sophisticated pattern recognition method should yield a significantly smaller probability of error than our proposed error benchmarks for the given classification problem.

The sequences of bounds we propose are all monotonic decreasing (Theorem 2.5.1). Meanwhile the computational cost of the underlying classification heuristic starts from extremely modest and then increases gradually. One can use the computational cost of the method (either training cost or testing cost, depending on context) as a proxy for complexity. The trade off between complexity and accuracy for each sequence of bounds can be visualized by looking at the corresponding curve in the

error and computational cost plane, as illustrated in Figure 2.1. One can look at each of these curves in two ways: 1) for any given desired classification error, each curve provides an upper bound on tolerable complexity (computational cost), and 2) for any given computational cost allowed, each curve provides a maximum tolerable classification error. Thus the curve divides the relevant area of the benchmark plane (above the error axis and right of the vertical line through Bayes Error) into two regions : the "negative-gain" region, situated above the curve, and the "positive-gain" region (shaded), below the curve and right of the vertical line through Bayes error. Any method lying in the negative-gain region of any benchmark curve (i.e. any value of n) is ill-suited for the classification problem at hand.

Each sequence of bounds we propose converges to a limit; that limit can also be used as an error benchmark which effectively divides the positive-gain region into two: the region directly under the curve (right of the asymptote), which we call "computational-gain" region, corresponds to methods that only provide a computational advantage over the random heuristics. Indeed, since there exists a random heuristic with comparable error rate (right above the point of the method, on the benchmark curve), the complexity of the structure allowed by the method is not effectively needed to achieve the given error rate. In other words, a comparable or even smaller rate can be achieved with piece-wise linear separations chosen at random, and thus any non-linearity afforded by the method is not effectively exploited. In contrast, we call the region left of the asymptote the "structural-gain" region.

In some cases, the limit of our sequences is equal to Bayes Error (Theorem 2.5.2). Thus in those cases, the structure-gain region is empty. Such pattern recognition problems are extremely easy and do not warrant the use of any sophisticated method. Testing new pattern recognition methods on such datasets should be discouraged. To the contrary, one should look for datasets representing pattern recognition problems for which the structure-gain region is very large. For such datasets, there is a lot of room for improvement in accuracy and thus the use of complex and/or computationally expensive methods is justified.

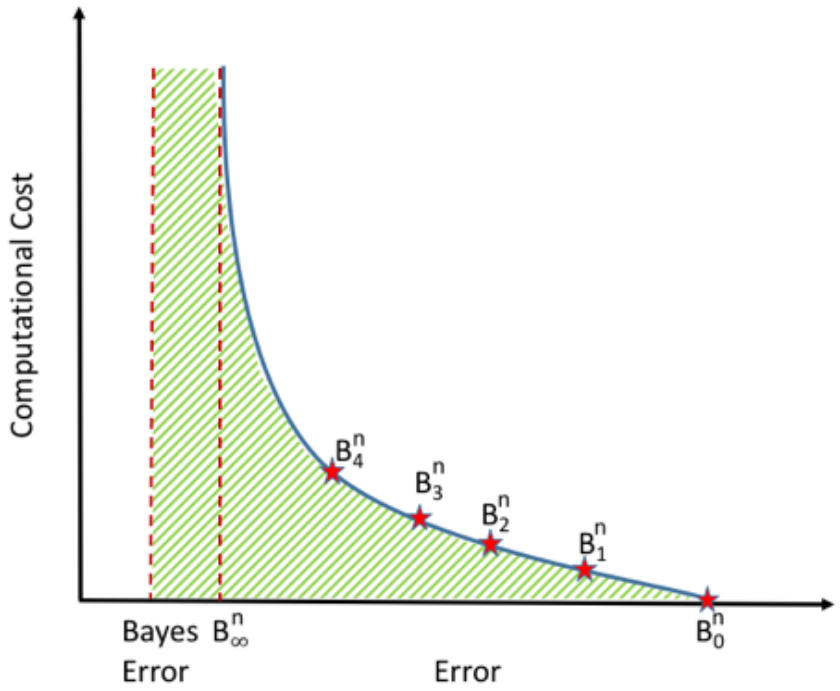


Fig. 2.1. The Benchmark Plane: For a given classification problem, the sequence of bounds $B_0^n, B_1^n, B_2^n, B_3^n, \ldots$ for some $n \in \mathbb{N}$ along with their (training) computation time define a curve with asymptote at B_∞^n. Methods whose sophistication is warranted for the problem must lie in the shaded region and to the left of the asymptote.

2.2 Benchmarking Pattern Recognition Problems

One benchmark traditionally used to put the results of a pattern recognition method in perspective is Bayes Error. Bayes Error is the probability of error of a classifier that assigns classes following Bayes Decision Rule and assuming full knowledge of the class (marginal) probability densities and priors. It is proven to be optimal, in the sense that no classifier can obtain a probability of error less than Bayes Error. Thus, Bayes Error is a benchmark for the probability of error of a classifier which corresponds to the minimum possible probability of error that can be achieved for the given feature vector [3].

Computing Bayes error from training data is difficult, especially in high-dimension. The task involves estimating the class probability densities and priors, and integrating these functions over potentially complex boundaries. While the case of univariate normally distributed class densities can be computed analytically, high-dimensional distributions (even normal ones) must be handled numerically. Computing Bayes error when the problem does not follow any common probability model is even more difficult. There have been efforts to approximate Bayes error and bound it. For the special case of binary classification where both class densities are multivariate Gaussians, Fukunaga et al [41] have presented an explicit mathematical expression that approximates Bayes error. The Chernoff bound [42] and the Bhattacharyya bound [43] are well-known upper bounds; some closed form expressions for these bounds exist for the special case of Gaussian densities of the underlying classification data, but they are not necessarily tight or particularly insightful for distributions that deviate markedly from a Gaussian [3].

Thus in practice, one seldom attempts to either estimate or bound Bayes error. Instead, one merely focuses on developing a classifier with the smallest possible probability of error. The accuracy of the chosen method on some test sets is used to measure the success of that method. For example, the large scale image classifier in [58] reports classification accuracy compared to other classifiers achieved on a subset of the ILSVRC 2010 dataset [59]. Similarly, the texture classifier in [60] reports and compares classification accuracies on three texture datasets : KTH-TIPS2 [61], FMD [62] and DTD [63]. Yet another example is the DCNN based image classifier in [64], which reports classification accuracies on the datasets : MIT-67 [65], Birds-200 [66], PASCAL-07 [67] and H3D [68]. If these test sets are appropriate, then one can conclude that Bayes error for each of these classification problems should be no larger than the error reported with each of the chosen methods.

But as we just mentioned, Bayes Error is merely a minimum bound for the probability of error of a classifier corresponding to the overlap between the probability distributions of the two underlying classes (class overlap): while it may be small

for some problems, it is not clear how difficult the underlying pattern recognition problem really is. For example, for the same value of Bayes Error, the optimal class separation could be a hyperplane (simple structure of data that can be easily found) or a hypersurface with highly varying curvatures and multiple connected components (a complex structure that requires a sophisticated pattern recognition method). Thus the value of Bayes Error provides no insight into the complexity of the classifier needed to solve the problem or the nature of the pattern recognition problem at hand.

Another benchmark that can be used to judge the performance of a classifier on a given classification problem is the value of the minimum class prior: $\min\{\text{Prob}\ (\omega_1), \text{Prob}\ (\omega_2)\}$, where ω_1 and ω_2 are the two possible classes for the given classification problem. For example, if Prob $(\omega_1) = 98\%$, then building a classifier with a 1.9% probability of error would not be particularly impressive. Indeed, picking the most likely class (ω_1) regardless of the value of x would nearly beat that classifier. This is relevant even if the class priors are equal Prob $(\omega_1) =$ Prob $(\omega_2) = 50\%$. Indeed classifiers with an estimated probability of error above 50% are not unheard of. Estimating the class priors from training data is not difficult. So while this benchmark also provides only a little bit of understanding about the nature of the pattern recognition problem at hand, it is useful to consider it when analyzing the results of a classifier.

2.3 Random Projections and TARP

As we mentioned earlier, the problem of building a two-class classifier is often formulated as a quest for a function $g(x) : \mathbb{R}^p \to \mathbb{R}$. The value of $g(x)$ can be viewed as a new feature; decisions are made by thresholding at zero the value of that feature. The difficulty of finding a good $g(x)$ tends to increase with the dimension of the feature vector x. Thus some methods first seek to decrease the dimension of x. Ideally, this should be done in such a way that the information that distinguishes the classes is preserved. However, this is a chicken and egg problem: it is hard to

preserve that information without knowing what it is, but one can hardly know what it is before decreasing the number of dimensions.

One way to decrease the dimension of x is to project it onto some lower-dimensional subspace. In the simplest cases, the lower-dimensional subspace is linear. Some approaches seek the best projection (in terms of some cost function) by careful numerical optimization. Other approaches project the data in some random fashion. The latter tends to be a lot less computationally extensive than the former. However, one might express concerns about the accuracy of mere random projections. While these concerns are justified, random projection methods are still popular because they have been shown to be surprisingly effective. For example, classification algorithms like [30], [31], [32] and [33] use random projections to transform high dimensional data into lower dimensional feature vectors as a pre-processing step before classification. Iterative random projections have also been used in the realm of Big Data to find visual patterns of structure within data by projecting from a high dimensional space to a low dimensional space [34]. An evaluation of the performance of random projections for dimensionality reduction can be found in [29].

The idea of using the random projections of high-dimensional data in a lower-dimensional space (of dimension greater than 1) is motivated by the Johnson-Lindenstrauss lemma [26] that relates to preservation of point distances of high-dimensional data when transformed to lower dimensions. Other examples of the utility of working with data in lower dimensions (greater than 1) instead of the original high-dimensional space include: [69] presents a framework to determine generalization error bounds for linear classifiers trained on the projected data, [70] explores the idea of using random projections as a regularization tool in scenarios where observations are fewer than dimensions, [71] presents a statistical perspective on the effects of using random projections in a least-squares problem in the lower-dimensional projected space of the data while [72] examines the problem of recovering the solution of optimization problems in high-dimensional space after finding a solution in randomly projected lower-dimensional space for reducing the computational cost. At

this point, it is important to note that all the above works use random projections as an intermediary dimension reduction tool (data is reduced to a low-dimensional space of dimensions greater than 1). In contrast to this approach, in this chapter, we propose to use random projections to reduce the dimensions of data down to one and extract classification structure in this trivial one-dimensional space.

Previous work suggests that images [36] and other high dimensional data [37] have so much hidden structure within that even randomly projecting down to just one dimension will likely uncover some of that structure. Thus, one can potentially construct a simple classifier based on the value of a feature vector $x \in \mathbb{R}^p$ by generating a random vector $r \in \mathbb{R}^p$ and thresholding the value of the projection $r \cdot x$. In mathematical terms, we can classify the data according to the rule (after relabeling the classes if needed)

$$r \cdot x \lessgtr_{\omega_2}^{\omega_1} t.$$

for some threshold value t. Observe that a threshold value of $t = \infty$ would classify all the points into the same class ω_1. Conversely, a threshold value of $t = -\infty$ would classify all the points into the same class ω_2. For optimal accuracy, the threshold $t \in \mathbb{R}$ should be chosen to minimize the probability of error of the classifier. We call the classifier obtained with the optimal value of the threshold a "TARP" (Thresholding After Random Projection), for short. Observe that, while the accuracy of such a classifier may not be particularly impressive in general, it is no worse than the accuracy of picking the most likely class within the population. In other words, regardless of the projection vector r, the probability of error of a TARP is no greater than $\min\{\text{Prob}\,(\omega_1), \text{Prob}\,(\omega_2)\}$. Furthermore, for some values of r, it can be quite low depending on the nature of the classification problem at hand. For example, if the data has been extended to a higher dimensional space using a kernel method (as with support vector machines) in such a way that the data became linearly separable, then the probability of error of a TARP classifier could be as low as 0%.

2.4 TARP-based Benchmarks

Let B_0 be the error rate achieved by always choosing the most likely class in the population, in other words

$$B_0 = \min\{\text{Prob }(\omega_1), \text{Prob }(\omega_2)\}.$$

This benchmark is our starting point. Let r be a sample of a p-dimensional random variable $\mathbf{r} \in \mathbb{R}^p$, and consider the projection of the feature vector x onto $\mathbf{r}$, namely $r \cdot x$. Consider the threshold t and the class labeling that minimizes the probability of error of the classifier

$$r \cdot x \lessgtr_{\omega_2}^{\omega_1} t.$$

Denote by $\varepsilon_1 = \epsilon_1(r)$ the probability of error of this TARP classifier. For an unspecified random vector $\mathbf{r}$, ϵ_1 becomes a random variable $\boldsymbol{\epsilon}_1$ whose probability density function is induced by that of $\mathbf{r}$. Denote by B_1 the expected value of $\boldsymbol{\epsilon}_1$:

$$B_1 = \int_{\mathbb{R}^p} \epsilon_1(r) f_r(r) dr.$$

The value of B_1, which represents the expected error of a TARP classifier under a given probability model for the projection vector $\mathbf{r}$, is our second benchmark. Observe that $B_1 \leq B_0$.

Now consider n samples $r_1, r_2, \ldots, r_n$ of the random vector $\mathbf{r}$ (i.i.d) and the TARP classifier for each projection $r_i \cdot x$, for $i = 1, \ldots, n$. Then pick the "best" TARP classifier among these n (for example, the one with the smallest probability of error, smallest impurity, lowest entropy, etc.) We call the resulting classifier an "n-TARP" (best among n TARP classifiers), and denote the probability of error of this classifier by $\varepsilon_1^n = \varepsilon_1^n(r_1, r_2, \ldots, r_n)$. For an unspecified set of random vectors (i.i.d.) $\mathbf{r}_1, \ldots, \mathbf{r}_n$, ε_1^n becomes a random variable $\boldsymbol{\epsilon}_1^n$ whose probability density function is induced by that of $\mathbf{r}$. Denote by B_1^n the expected value of $\boldsymbol{\epsilon}_1^n$:

$$B_1^n = \int_{\mathbb{R}^{p \times n}} \varepsilon_1^n(r_1, \ldots, r_n) f_r(r_1) \ldots f_r(r_n) dr_1 \ldots dr_n.$$

For any positive integer n, the value of B_1^n represents the expected error of an n-TARP classifier under a given probability model for the projection vector $\mathbf{r}$.

Our proposed sequences of benchmarks are built by contructing a k-layer binary decision tree using an n-TARP at every node of the tree. More specifically, we first fix $n \in \mathbb{N}$ and let k vary among all positive integers. We then consider a k-layer tree obtained by generating n random samples of the projection vector $\mathbf{r}$ for each node and using these projection vector samples to construct a n-TARP classifier at each node of the tree. Note that, although our proposed trees are built "at random," they are not the same as the trees defined by the well-known Random Forests [44]. Indeed, the randomness in our trees comes from the n-TARP used for each node, whereas in the case of random forests, it stems from the selection of random subsets of training data to form each decision tree.

We denote the probability of error of a given (sample) decision tree with k levels by ε_k^n. For an unspecified set of random projection vectors (i.i.d.), ε_k^n becomes a random variable $\boldsymbol{\epsilon}_k^n$ whose probability density function is induced by that of $\mathbf{r}$. Denote by B_k^n the expected value of $\boldsymbol{\epsilon}_k^n$, for $k = 1, \ldots, \infty$, and set $B_0^n = B_0$, for any positive integer n. We propose to use the sequence of bounds $\{B_k^n\}_{k=0}^{\infty}$ to analyze the nature of the structure of a pattern recognition problem.

2.5 Mathematical Properties of Proposed Benchmarks

Theorem 2.5.1 (Monotonicity in k) *For any integer $n_0 \in \mathbb{N}$, the bounds $\{B_k^{n_0}\}_{k=1}^{\infty}$ form a monotonic decreasing sequence*

$$B_{k+1}^{n_0} \leq B_k^{n_0}$$

converging to a limit

$$B_{\infty}^{n_0} := \lim_{k \to \infty} B_k^{n_0}$$

that is no smaller than Bayes Error

$$B_{\infty}^{n_0} \geq \textit{Bayes Error}.$$

Proof Let $k \geq 1,\ k \in \mathbb{Z}$. Consider a decision tree T with $k+1$ levels built using an n-TARP at each node. Observe that T is a random sample of a random tree $\mathbf{T}$ with a probability density function $\rho_T(T)$ that is induced by the random projection model for $\mathbf{r}$.

Let $\varepsilon^n_{k+1}(T)$ be the probability of classification error of decision tree T. Let $\varepsilon^n_k(T)$ be the probability of classification error of decision tree T restricted to its first k levels. By construction (since none of the n-TARPS considered increases the classification error from the previous level)

$$\epsilon^n_{k+1}(T) \leq \epsilon^n_k(T). \tag{2.1}$$

Observe that $\varepsilon^n_{k+1}(T)$ and $\varepsilon^n_k(T)$ are samples of the random variables $\boldsymbol{\epsilon}^n_{k+1}$ and $\boldsymbol{\epsilon}^n_k$ respectively. We have

$$\begin{aligned} B^n_{k+1} = \mathbb{E}(\boldsymbol{\epsilon}^n_{k+1}) &= \int_{\mathbb{T}} \epsilon^n_{k+1}(T)\rho_T(T)\,dT \\ &\leq \int_{\mathbb{T}} \epsilon^n_k(T)\rho_T(T)\,dT \\ &= \mathbb{E}(\boldsymbol{\epsilon}^n_k) \\ &= B^n_k \end{aligned}$$

where $\mathbb{T}$ is the set of all $k+1$ level decision tress produced by the random projection model. Therefore

$$B^n_{k+1} \leq B^n_k \quad \forall\ k \geq 0, k \in \mathbb{Z}, n \in \mathbb{N}, \tag{2.2}$$

and so $\{B^{n_0}_k\}$ is a monotone decreasing sequence in k.

By optimality of Bayes Error,

$$B^{n_0}_k \geq \mathit{Bayes\ Error} \quad \forall\ k \geq 0, k \in \mathbb{Z}, n_0 \in \mathbb{N}. \tag{2.3}$$

From (2.2), we know that the sequence $B^{n_0}_k$ is monotone decreasing and from (2.3), it is bounded below by the Bayes Error. Hence, using the Monotone Convergence Theorem, the sequence $B^{n_0}_k$ is convergent in k and it converges to its infimum

$$B^{n_0}_\infty = \lim_{k\to\infty} B^{n_0}_k, \quad B^{n_0}_\infty \geq \mathit{Bayes\ Error}.$$

■

Corollary 1 *The limits B_k^∞ form a monotonic decreasing sequence*

$$B_{k+1}^\infty \leq B_k^\infty$$

converging to a limit

$$B_\infty^\infty := \lim_{k\to\infty} B_k^\infty$$

that is no smaller than Bayes Error

$$B_\infty^\infty \geq Bayes\ Error.$$

Theorem 2.5.2 (Optimality of asymptotes) *Suppose that the probability density function $f_R(r)$ for the random projection vector $\mathbf{r} \in \mathbb{R}^p$ is such that*

$$p_u = \int_u f_R(r)dr \neq 0,$$

for any open set $u \in \mathbb{R}^p$. If the data points x to be classified are drawn from a mixture of two classes ω_1, ω_2

$$\rho(x) = Prob\ (\omega_1)\rho(x|\omega_1) + Prob\ (\omega_2)\rho(x|\omega_2)$$

such that the marginal distributions for both classes are normal with the same covariance matrix

$$\rho(x|\omega_1) = \mathcal{N}(\mu_1, \Sigma)\ ,\ \rho(x|\omega_2) = \mathcal{N}(\mu_2, \Sigma),$$

then all the asymptotes B_∞^n, $\forall n \in \mathbb{N}$, are optimal

$$B_\infty^n = B_k^\infty = Bayes\ Error\ \ \forall\ k, n \in \mathbb{N}.$$

Proof The optimal classifier in the case described is a linear separation hyperplane in $\mathbb{R}^p$. Let $\hat{N}_\rho \in \mathbb{R}^p$ be a unit normal vector to that hyperplane. Observe that if the random vector sample drawn is $r = \hat{N}_\rho$, then the TARP classifier will be optimal. Let u be an open neighborhood of $\hat{N}_\rho$ and let p_u be the probability that $\mathbf{r} \in u$. By assumption, $p_u \neq 0$. Consider n independent random vector samples $\{r_i\}_{i=1}^n$. The probability that none of the samples lie in u is $(1 - p_u)^n$, which approaches

zero as n goes to infinity. Thus, with probability one, the infinite random vector sequence $\{r_i\}_{i=1}^{\infty}$ contains a vector that is arbitrarily close to $\hat{N}_\rho$. This means that the probability of error of an n-TARP (best TARP among n trials) will converge to Bayes Error with probability one as n goes to infinity. Thus the expected value of that error, B_1^n, will approach Bayes Error as n goes to infinity. So the limit $B_1^\infty =$ Bayes Error. By Theorem 2.5.1, $B_{k+1}^n \leq B_k^n$. Taking the limit as $n \to \infty$, we have $B_{k+1}^\infty \leq B_k^\infty$ for any k, and thus $B_k^\infty =$ Bayes Error, for any $k \in \mathbb{N}$.

Now we look at B_k^n. So consider a tree with k levels constructed with a n-TARP at each node. Observe that each node at the last level of the tree is concerned with classifying a fraction of the original data space; if we picked $\hat{N}_\rho$ as the random vector at each of these nodes, then the overall k-level tree classifier would be optimal. In that case, adding further levels (more n-TARP classifiers) below any of these nodes would not decrease the accuracy of the classifier below that node, as it is already optimal. As we travel down each branch of the tree and let the number of levels k go to infinity, the random projection vector sample used for the n-TARP at each of the nodes we encounter along our path will form an infinite sequence of vectors. By the same argument as above, that infinite sequence contains a vector that is arbitrarily close to the optimal one $\hat{N}_\rho$, with probability one. Thus the probability of error of a k-level tree constructed with an n-TARP at each node approaches Bayes Error with probability one, as k approaches infinity. Therefore the expected value of that error also approaches Bayes Error, $B_\infty^n =$ Bayes Error.

■

2.6 n-TARP Implementation and Bound Estimation

We implemented an algorithm to estimate the bounds B_k^n for a binary decision problem using a dataset consisting of labeled (potentially high-dimensional) feature vectors. Our code is available at [73]. To estimate B_k^n, we build a k-level binary decision tree with an n-TARP at each node. We split the dataset set into two: 25%

is training data used to select the random vectors for each n-TARP, 25% is cross-validation data used to decide whether to apply a stopping criteria, and 50% of the data is the testing data used to estimate the error rate of the decision tree. Note that the stopping criteria does not effectively stop the tree from growing, but it prevents the data from being split at that node, so that the tree continues to grow, albeit artificially, to a length of k-layers,

For example, in order to estimate B_1^1, we construct a 1-level decision tree ($k = 1$) with a single 1-TARP. To do this, we generate $n = 1$ random vector(s) $\{r_i\}_{i=1}^n$ of the same dimension as the data, with each element of the vector drawn from a uniform distribution on [-1,1]. As described in Section 2.4, we take the inner product of every data point in the training set with the random vector r_1. We thus have $n = 1$ set(s) of projections of the training samples. For simplicity, we assume that the projected class distributions for the two underlying classes are 1D Gaussians with mean μ_1, μ_2 and standard deviations σ_1, σ_2, respectively. Bayes Decision Rule [74] in this case yields up to two thresholds: we pick the threshold t that lies between the two empirical means. Now, we use the threshold t along with the the random vector r_1 to classify the cross-validation data and record the error obtained. If this error is greater than the error observed before classifying the cross-validation set (the prior error in this case), then we choose t to be $\pm\infty$. i.e. we do not split the data into two and record the cross-validation error as the error achieved before classification (the prior error in this case). Having constructed a 1-level decision tree, we use it to classify the testing data and record the testing error. We repeat this construction and testing process 100 times and calculate the average testing error and use it as our estimate of B_1^1.

Now to estimate B_1^n, we build a 1-level decision tree ($k = 1$) with a single n-TARP. So we generate n random vectors of the same dimension as the features as before. Projecting the dataset onto each of these random vectors, we obtain n one-dimensional datasets and choose the best threshold using Bayes rule for each of these datasets. Let t^* be the threshold that produces the "best" split (we pick the one which decreases the Gini impurity [3] on the training data the most). Following the

same steps as above, we cross-validate the tree and compute the testing errors for 100 runs of the algorithm and calculate the average testing error. We declare this as our estimate for B_1^n.

For the general case of building a k-level decision tree with an n-TARP at each node of the tree. Basically, we apply an n-TARP at each node of the tree starting at the root. Based on the threshold found at that node, we split the data into two classes and pass one class split to the left child and the other to the right child of the current node. We now recursively apply the n-TARP construction process described above at the child nodes formed. At level k, we will have 2^k nodes and we find the average testing error across the data present at all of these nodes and declare it as our estimate of B_k^n. So by building a k-level tree in this manner, we can estimate $\{B_1^n, B_2^n, ..., B_k^n\}$. Note that we grow a new set of k-level trees and evaluate them for each value of the parameter k in order to estimate B_k^n. This means that a different set of trees are grown each time to estimate B_k^n for a specific k and n.

Of course, the results at any particular node in the decision tree are affected by the number of data points present at that node. When building a tree, the number of training samples at each node keeps decreasing as we move down the tree. Naturally, our estimates become unreliable when the number of samples at a node is too small. Hence, there is an upper limit on the values that k can take while also providing a good estimate of B_k^n. This limit is dependent on the dataset being used.

To test our implementation, we conducted the following experiment. We generated 6000 data samples from a mixture of two Gaussians in $\mathbb{R}^5$ and used 1500 data points for training, 1500 for validation, and the remaining 3000 for testing in the n-TARP algorithm. The results for this experiment are presented in Figure 2.2. We compute the sequence B_k^1 for this data. The parameters for the Gaussian mixture are $\mu_1 = (0, 0, 0, 0, 0)$, $\mu_2 = (10, 0, 0, 0, 0)$, $\Sigma_1 = \Sigma_2 = \mathbb{I}$. Bayes Error in this case is below machine precision (10^{-16}). We observe that the asymptote B_∞^1 is close to zero and equals Bayes Error as expected. This shows that Theorem 2.5.2 holds true and our bound estimates are accurate.

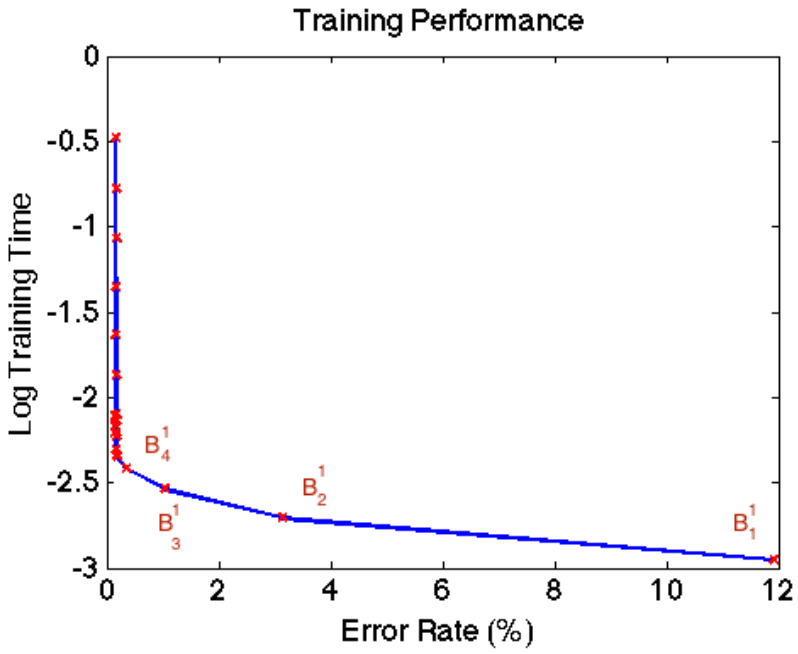


Fig. 2.2. A Benchmark curve for Normally Distributed Data: This empirical curve shows the asymptote defined by the limiting error bound B^1_∞. Here B^1_∞ is close to zero, as predicted by Theorem 2.5.2.

2.7 Experiments with Real Data

2.7.1 Analysis of Digit Recognition Problems

We first look at two different two-class digit recognition problems. The first problem is that of distinguising between the digit "0" and the digit "1" on a gray scale image. The other is that of determining whether the digit featured on a gray scale image is even or odd. Both problems will be studied using the MFEAT dataset [75], which contains 200 samples of each digit (0-9). The results of our analysis for each of these two problems are presented in Table 2.1 and Table 2.2, respectively. To capture the dependence of the recognition problem on the feature set used to represent the data, we consider three different feature sets: Fourier coefficients (76 features), Karhunen-Loeve coefficients (64 features) and Zernike Moments (47 features).

We used this data to estimate the first few terms of some of our proposed sequence of bounds. The first bound B_0 was estimated as 0.5 since the number of samples of each class for both classification tasks were equal. For the 0-vs-1 classification, the subsequent bounds B_k^n were calculated using the first 100 samples each of 0 and 1 as training data (first 50 of each for training and the remaining 50 for cross validation) and the next 100 samples as testing data for each feature set. Note that the ratio of class 0 and class 1 samples in the training, validation and test sets was the same. For the even-vs-odd classification, the subsequent bounds B_k^n were calculated using the first 100 samples of each digit which were split into even and odd classes as training data (first 50 of each digit for training and the remaining 50 for cross validation) and the next 100 samples as testing data for each feature set. Again, the ratio of class 0 (even) and class 1 (odd) samples in the training, validation and test sets was the same.

As expected, B_1^n and B_2^n for $n = 1, 10, 50$ are smaller than B_0. Note that for the 0-vs-1 classification, B_1^{50} is nearly as low as 2% which is a significant improvement over B_0 at 50%. Such a small value of the 50-TARP Benchmarks might be surprising, considering their low computational cost. Thus, even without having estimated any of the asymptotes, we can see that the positive-gain regions for each of these feature sets must be very small. On the other hand for the even-vs-odd classification, B_1^{50} is close to 20% for Karhunen-Loeve coefficients and 30% for the remaining two. The improvement over B_0 at 50% is not as great in this case but it is still significant considering it was achieved with naive random projections.

For comparison, we used MATLAB to build support vector machine (SVM) classifiers [76] using different kernels and parameter values. We also trained a deep neural network classifier [57] with 2 hidden layers. An Adaboost [77] classifier was also used for comparison, where the family of weak classifiers used was of the form $p.sgn(x_i - \theta)$ where p is a parity bit, x_i is the i^{th} component of the feature vector x and θ is a threshold. Training was done over 40 rounds in each case.

Table 2.1.
Classification of images of 0 and 1. The empirical error of support vector machines with various kernels and parameters, Deep Neural Network and Adaboost is compared to the empirical estimate for our proposed error bounds $B_0, B_1^1, B_2^1, B_1^{10}, B_2^{10}, B_1^{50}, B_2^{50}$.

Fourier Coefficients											
Method	DNN	Adaboost	SVM RBF	SVM Linear	B_2^1	B_1^1	B_2^{10}	B_1^{10}	B_2^{50}	B_1^{50}	B_0
parameter values	**	*	10								
Error	0%	0.5%	0%	0%	16.285%	25.125%	3.770%	6.855%	2.205%	2.340%	50%
Training Time (s)	9.198	15.537	2.280	0.078	0.0003	0.0006	0.0028	0.0016	0.0102	0.0058	-
Testing Time (s)	0.022	0.0020	0.0455	0.0025	0.0001	0.0001	0.00006	0.00005	0.00006	0.00008	-
Karhunen-Loeve Coefficients											
Method	DNN	Adaboost	SVM RBF	SVM Linear	B_2^1	B_1^1	B_2^{10}	B_1^{10}	B_2^{50}	B_1^{50}	B_0
parameter values	**	*	10								
Error	0.5%	2%	1%	0.5%	22.535%	28.775%	6.055%	10.580%	4.315%	5.905%	50%
Training Time (s)	8.9281	12.9592	0.0494	0.0460	0.0008	0.0003	0.0022	0.0011	0.0086	0.0052	-
Testing Time (s)	0.0247	0.0003	0.0052	0.0015	0.00008	0.00009	0.00008	0.00005	0.00011	0.00005	-
Zernike Moments											
Method	DNN	Adaboost	SVM RBF	SVM Linear	B_2^1	B_1^1	B_2^{10}	B_1^{10}	B_2^{50}	B_1^{50}	B_0
parameter values	**	*	10								
Error	1%	1.5%	1%	1%	10.925%	21.620%	2.775%	3.865%	2.465%	2.425%	50%
Training Time (s)	7.9931	9.6483	0.0212	0.0258	0.000002	0.0002	0.0016	0.0013	0.0069	0.0041	-
Testing Time (s)	0.0195	0.0003	0.0022	0.0012	0.00011	0.00009	0.00005	0.00002	0.00003	0.00003	-

* The family of weak classifiers used was of the form $p.sgn(x_i - \theta)$ where p is a parity bit, x_i is the i^{th} component of the feature vector x and θ is a threshold. Training was done over 40 rounds in each case. **The deep neural network consists of 2 hidden layers where the first layer has 35 components and the second layer has 15 components.

Table 2.2.
Classification of images of even and odd numbers. The empirical error of support vector machines with various kernels and parameters, Deep Neural Network and Adaboost is compared to the empirical estimate for our proposed error bounds $B_0, B_1^1, B_2^1, B_1^{10}, B_2^{10}, B_1^{50}, B_2^{50}$.

Fourier Coefficients											
Method	DNN	Adaboost	SVM RBF	SVM Linear	B_2^1	B_1^1	B_2^{10}	B_1^{10}	B_2^{50}	B_1^{50}	B_0
parameter values	**	*	10								
Error	12.3%	19.6%	15.2%	Div†	38.462%	42.854%	30.678%	35.141%	29.047%	32.058%	50%
Training Time (s)	14.0915	136.6714	0.2155	-	0.0025	0.0012	0.0081	0.0047	0.0330	0.0206	-
Testing Time (s)	0.0230	0.0016	0.0299	-	0.0005	0.0003	0.0005	0.0003	0.0005	0.0003	-
Karhunen-Loeve Coefficients											
Method	DNN	Adaboost	SVM RBF	SVM Linear	B_2^1	B_1^1	B_2^{10}	B_1^{10}	B_2^{50}	B_1^{50}	B_0
parameter values	**	*	10								
Error	1.4%	5.9%	2.2%	Div†	34.715%	40.071%	22.378%	27.481%	16.517%	20.687%	50%
Training Time (s)	16.4836	112.4335	0.1616	-	0.0022	0.0011	0.0096	0.0053	0.0289	0.0179	-
Testing Time (s)	0.0238	0.0015	0.0201	-	0.0005	0.0002	0.0006	0.0003	0.0005	0.0002	-
Zernike Moments											
Method	DNN	Adaboost	SVM RBF	SVM Linear	B_2^1	B_1^1	B_2^{10}	B_1^{10}	B_2^{50}	B_1^{50}	B_0
parameter values	**	*	10								
Error	15.1%	23.2%	16.3%	Div†	40.056%	43.148%	34.077%	36.210%	31.940%	33.440%	50%
Training Time (s)	16.9158	82.7901	0.3740	-	0.0017	0.0009	0.0065	0.0037	0.0237	0.0149	-
Testing Time (s)	0.0221	0.0014	0.1221	-	0.0004	0.0002	0.0004	0.0002	0.0004	0.0002	-

* The family of weak classifiers used was of the form $p.sgn(x_i - \theta)$ where p is a parity bit, x_i is the i^{th} component of the feature vector x and θ is a threshold. Training was done over 40 rounds in each case. **The deep neural network consists of 2 hidden layers where the first layer has 35 components and the second layer has 15 components. †Did not converge after 15000 iterations.

Observe that, in the case of the 0-vs-1 classification, the other classifiers are only slightly more accurate than the benchmarks despite being significantly more expensive. This is expected since the region of positive gain is very small and thus only modest gains, if at all, can be achieved with more complex methods. There are two factors at play here: an obvious structure in the data that can be found by simple heuristics (random projections) as well as a small class overlap and thus a small value of Bayes Error. Note that this holds for all three feature sets considered. The fact that the structure is obvious makes this particular classification problem not suitable for testing new classification methodologies.

The even-vs-odd classification problem is a lot more difficult, as can be seen by comparing the benchmark values in each case. The TARP benchmarks go down to about 30% for Fourier Coefficients and the Zernike Moments feature vectors and they go down to about 16% for the Karhunen-Loeve Coefficients feature vector. We would need to estimate the value of more terms in each sequence in order to estimate the asymptotes (which we shall do in our further analysis below). However, we can still observe the presence of a method (linear SVM) in the negative-gain region of the benchmark plane. Indeed, it is interesting to note that the method did not converge for any of the feature sets considering that reasonable piece-wise linear separations can be found at random. On the other hand, the other methods have a better accuracy than the benchmarks. Thus it is reasonable to imagine that the problem at hand has a hidden structure (i.e., a complex non-linear separation boundary) that cannot be found by simple random projections but that the machinery of an SVM, Adaboost, or DNN can reveal. While the structure is hidden for all feature sets considered, the class overlap appears to vary. We see a small class overlap in the case of the Karhunen-Loeve coefficients, and a potentially large class overlap in the case of the Fourier Coefficients and the Zernike moments. However the structural-gain region in each case might actually have a similar size, as the difference between the minimum benchmark values and the smallest of the complex methods accuracy is around 15-20% in all three cases. In other words, all three problems may actually

have a structure that is equally well hidden, despite the fact that one potentially has a much smaller Bayes Error than the others. Such datasets could thus be good candidates for developing and testing new classification methods.

2.7.2 Analysis of Pedestrian Detection Problems

We now look at the problem of detecting the presence of a pedestrian on a very low-resolution image. We study this problem using the Pedestrian Dataset [78]. This dataset contains low-resolution greyscale images divided into three training sets and two testing sets; we trained with Training Set #2 (splitting into two for training and validation) and tested on Testing Set #1. Each set contains 5000 samples without pedestrian and 4800 samples with pedestrian. We used two different feature sets to represent the images [73]. The first is a feature vector consisting of 648 discrete cosine transform coefficients. The second representation uses 10 rows of the Pascal Triangle of the image [79]. After removing the redundant right-hand-sides of the triangle and storing each complex entry as two real entries (the middle of the triangle is always real), we ended up with a total of 130 features for each image.

Again, we computed the first few terms of some of our proposed sequences of bounds. Our results are presented in Table 2.3. The first bound B_0 was estimated as $\frac{4800}{9800}$ (The ratio of the pedestrian samples to the total number of samples). The subsequent bounds B_k^n were estimated using Training Set #2 and Testing Set #1 for each set of feature. Note that the 4900 images we used for training contained the same ratio of no pedestrian (2500) to pedestrian (2400) pictures as the validation set (2500 no pedestrian, 2400 pedestrian). As expected, B_1^n and B_2^n for $n = 1, 10, 50$ are smaller than B_0 for both feature sets. In fact, B_1^{50} at 25.3% is nearly half of B_0 for the Pascal Triangle coefficient feature vectors.

For comparison, we used MATLAB to build support vector machine (SVM) classifiers [76] using different kernels and parameter values. We also trained a deep neural network classifier [57] with 2 hidden layers. Again, the linear SVM did not converge

Table 2.3.
Pedestrian Detection from Low-Resolution Pictures. The empirical error of support vector machines with various kernels and parameters along with Deep Neural Network is compared to the empirical estimate for our proposed error bounds $B_0, B_1^1, B_2^1, B_1^{10}, B_2^{10}, B_1^{50}, B_2^{50}$.

With DCT Coefficients											
Method	DNN	SVM RBF	SVM Linear	SVM MLP	B_2^1	B_1^1	B_2^{10}	B_1^{10}	B_2^{50}	B_1^{50}	B_0
parameter values	**	50		$[1,-1]^*$							
Error	22.5%	21.5%	Div†	38.7%	44.2%	45.5%	37.8%	39.4%	35.9%	36.8%	49.0%
Training Time (s)	577.659	177.399	-	21.801	0.141	0.067	0.440	0.268	0.181	0.117	-
Testing Time (s)	0.203	3.123	-	2.010	0.029	0.022	0.029	0.021	0.028	0.021	-
With Pascal Triangles											
Method	DNN	SVM RBF	SVM Linear	SVM MLP	B_2^1	B_1^1	B_2^{10}	B_1^{10}	B_2^{50}	B_1^{50}	B_0
parameter values	**	50		$[1,-1]^*$							
Error	25.2%	27.6%	Div†	49.50%	30.6 %	32.9%	27.1%	26.8%	25.5%	25.3%	49.0%
Training Time (s)	102.987	65.560	-	7.061	0.032	0.017	0.103	0.063	0.416	0.280	-
Testing Time (s)	0.089	2.101	-	0.628	0.006	0.004	0.005	0.004	0.004	0.005	-

*Default parameters in MATLAB. **The deep neural network consists of 2 hidden layers where for DCT, the first layer has 100 components and the second layer has 50 components. For Pascal Triangles, the first layer has 60 components and the second layer has 30 components.†Did not converge after 15000 iterations.

for either of the feature vectors within the default maximum number of iterations (15,000). Thus, the naive approach of finding a linear separation at random performs better than the iterative approach of the linear SVM in this case. Similarly, the error rate of the SVM with the MLP kernel is higher than some of the bounds listed. Thus both the linear SVM and the MLP kernel SVM lie in the negative-gain region of the benchmark plane. On the other hand, for the DCT coefficient feature vector, the error rate of the Deep Neural Network and the SVM with Radial Basis Function Kernel is significantly lower than any of the B_k^n presented in Table 2.3. While we do not have enough data to estimate the position of the asymptotes, we can conjecture that these two methods lie in the structure-gain region of the Benchmark plane. Thus the evidence suggests that the structure is hidden since simple heuristics do not perform as well as more sophisticated methods. In contrast, for the case of the Pascal Triangle coefficient feature vector, the bound B_1^{50} is only beat by the Deep Neural Network by a mere 0.1% which is quite small considering the much higher computational cost and sophistication for the DNN. The SVM with the RBF kernel is even worse than our bounds. Thus neither of these methods lie in the positive-gain region of the benchmark plane. It could be that the structure of the problem is obvious and that the overlap between the classes is fairly large (around 25%). It could also be that the overlap is smaller than that but the structure is very well hidden, so much so that even the DNN or the two SVMs we tried are unable to capture that structure. Thus this dataset could be a good candidate to develop and test new pattern recognition methods.

Estimation of Asymptotes for the Even-vs-Odd Digit Classification Problem

Our previous experiments illustrate how to use one, two or a few bounds to analyze the structure and class overlap of a pattern recognition problem. For a more complete analysis, one needs to estimate the asymptotes B_∞^n for some value(s) of n. To

illustrate how to do this, we used the MFEAT data and we focus on the even-vs-odd classification problem and use the Karhunen-Loeve Coefficient representation.

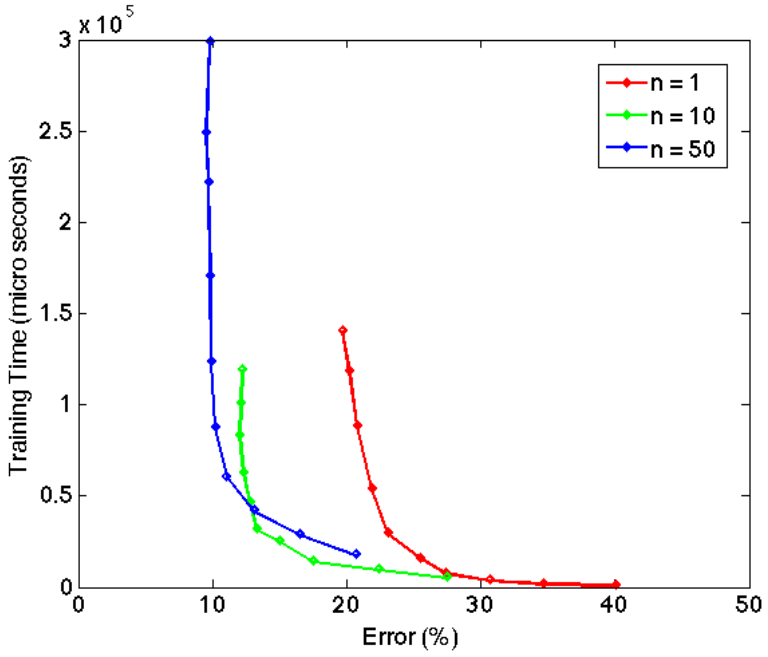


Fig. 2.3. Benchmark Curves for the even-vs-odd Classification Problem: The evolution of the sequence of bounds B_k^n for $n = 1, 10, 50$ using the Karhunen-Loeve Coefficients clearly shows the location of the asymptote with merely 10 terms.

Figure 2.3 shows the curves obtained after computing ten terms in the sequence B_k^n for n=1,10 and 50. This seems to be enough terms to estimate the asymptote for each curve, the smallest of which ($n = 50$) appearing to lie around 10%, still above the error rate of Adaboost (5.9%), DNN (1.4%) and the RBF kernel SVM (2.2%). This provides further evidence that the class separation structure is hidden and that a sophisticated classifier is required. For example, the sophistication of the DNN classifier can decrease the error by a factor greater than seven.

This convergence to a limiting B_∞^n indicates that our benchmarks are resistant to over-fitting the data as simply increasing the number of levels in the decision tree does not keep reducing the training error to 0%. Over the first few levels of the tree,

the natural structure present in the data has been found and further levels don't find any new structure that can reduce the error further. The first few levels where the structure within the data is being explored corresponds to the portion of the curves in Figure 2.3 where the error is reducing relatively quickly with the level parameter k. The later levels where the structure within the data has been completely detected corresponds to the portion of the curves where the errors remain more or less the same for increasing k with increasing computational cost (training time).

2.8 Conclusions

We observed that some pattern recognition problems, in particular high-dimensional ones, are a lot easier to solve than others. Indeed the structure of the data is sometimes so easy to find that simple heuristics can lead to a near optimal classification (i.e., with a probability of error close to Bayes error); the existence of such problems was proven in our experiments. Other problems are a lot more difficult; finding a way to accurately predict the class from a feature vector necessitates a sophisticated method.

In order to analyze the nature of the structure of the class distributions in a pattern recognition problem, we proposed simple heuristics to obtain upper bounds on the probability of error of a classifier. Our bounds are obtained using extremely low-computation methods based on random projections onto a one-dimensional subspace of the feature space and are particularly well-suited to analyze high-dimensional data sets. We use these bounds to construct a sequence of benchmark curves parameterized by probability of error and computational cost. Each curve determines an error asymptote which we proved to be optimal (equal to Bayes Error) in some cases. When using a non-trivial pattern recognition method on a specific classification problem, one should check that the computational time and probability of error of the method are situated on the left-hand-side of our proposed curves (i.e., in the positive-gain region of the benchmark plane); else their sophistication is either unwarranted (on or

near the benchmark curve) or unsuited (right of the curve, the negative-gain region of the benchmark plane) for the structure of the data.

To illustrate our proposed benchmarking framework, we looked at two types of digit recognition problems: the problem of distinguishing between "0" and "1" on a gray-scale image of a hand-drawn digit, and the problem of distinguishing between even and odd integers on a gray-scale image of a hand-drawn digit. Our analysis indicates clearly that the structure of the first problem is a lot easier to find than that of the second. Such simple classification problems should not be used for testing and developing new pattern recognition methods. We also looked at two pedestrian detection problems. For one of these problems, none of the popular pattern recognition methods we tried was found to lie in the positive-gain region of the benchmark plane. Thus the evidence suggests that the problem at hand has an obvious structure that can be found by random projection, while the class overlap is fairly large (Bayes Error near 25%). Based on evidence presented in [36], we expect many pattern recognition problems based on image/video data to have a similarly obvious structure.

3. N-TARP CLUSTERING

The contents of this chapter are an extension of [40].

3.1 Introduction

Clustering is a fundamental task of data mining and exploration which seeks to uncover structure within data and organize samples into groups of similar data in some sense. While many clustering approaches exist that work well in low-dimensional data space, the problem is considerably harder in higher dimensions. Methods like k-means [80], Expectation Maximization [81], BIRCH [82] and DBSCAN [83] work well in lower dimensional space, usually utilizing point proximity metrics, with the l_2 norm being a popular choice. However, such approaches are not feasible in higher dimensions, in part because of the phenomenon termed as the "Curse of Dimensionality" [1, 2]. Indeed, the sparsity of datasets in high-dimensional spaces makes clustering a challenging problem. This necessitates specially designed high-dimensional clustering methods like Proclus [84], Clique [85], Doc [86], Fires [87], INSCY [88], Mineclus [89], P3C [90], Schism [91], Statpc [92] and SubClus [93] among others. However, these methods may still not be effective in small data problems where the number of samples is much smaller than traditional big data problems.

Similarity functions tend to not distinguish classes well in high-dimensions. In rough terms, this is because small differences in information can be hidden under cumulated errors in non-relevant dimensions. High-dimensionality also poses a serious challenge to finding statistically significant results from the data. Hence, there is a need for effective clustering methods that find meaningful statistically significant clusters in high-dimensions with an emphasis on small data problems.

Another interesting and challenging aspect of high-dimensional clustering is the possibility of finding several different "good" clusters instead of a unique "best" grouping. This is explored in our recent works [36,37] wherein we formed a hypothesis that several different separations in the high-dimensional space can be found, so many in fact that just generating random linear hyper-plane separations can yield valid clusters.

Based on findings from [36, 37], we propose a novel randomized clustering algorithm called n-TARP, where TARP stands for "Thresholding After Random Projection". The core idea is to project the data onto randomly generated vectors and threshold the resulting projections so as to separate them into two groups. This forms the cluster assignment for the corresponding data points in the original high-dimensional space. Due to the randomized nature of this method, cluster assignments for each sample are random variables and the net result is a distribution of clusters which reflects the idea of the existence of several clusters rather than a unique cluster assignment. We incorporate a mechanism of statistical validation within this clustering framework that allows us to determine if the resulting clusters are statistically valid. We achieve this statistical validation by carrying out the necessary analysis in our projected one dimensional space rather than the original high-dimensional space, thereby bypassing the difficulties of statistical validation in high dimensions.

3.2 Random Projections in High-dimensions

The concept of random projections [27, 28] has previously been proposed as a basis for dimensionality reduction techniques [25, 29] with several applications in classification and clustering. For instance, [30–33] use random projections to reduce high-dimensional data into lower dimensional feature vectors for use with classifiers. Random projections have also been used in an iterative manner to find visual patterns of structure in data through dimension reduction [34].

The application of random projections to dimensionality reduction [25] is motivated by the Johnson-Lindenstrauss lemma [26] which relates to preservation of structure in high-dimensional space when transformed to lower dimensional spaces by preserving point distances. Other clustering methods based on random projections like [35] project data to a lower dimensional space of dimensions greater than one followed by a point proximity based cluster assignment. Our method however, uses random projections with a different purpose. We do not aim to preserve the structure, rather, we look to extract structure that is hidden in high dimensions which can manifest itself as a point separation upon projection onto a trivial random 1D subspace (simplified illustrative example in Figure 3.1). This is an important distinction as many other popular subspace clustering methods like the recent state-of-the-art SSC-OMP [10] rely on point proximity and try to utilize the concept of "self-expressiveness" to group data into clusters. The point proximity versus separation is a fundamental difference in the inherent geometries underlying subspace clustering methods as compared to n-TARP.

3.3 The n-TARP method

The crux of the n-TARP method, as summarized by the acronym TARP, is Thresholding After Random Projection. The idea is to generate a random vector in the data space and project the data onto it to obtain projection values in $\mathbb{R}$. This is motivated by the hypothesis that the resulting distribution of projection values will likely have a bi-modal distribution, which has been observed in [36, 37]. When a "good" bi-modal distribution is obtained, a threshold can easily be found to separate the projection values into two groups as shown in Figure 3.1. We will now go through the details of each step of the method in the following subsections.

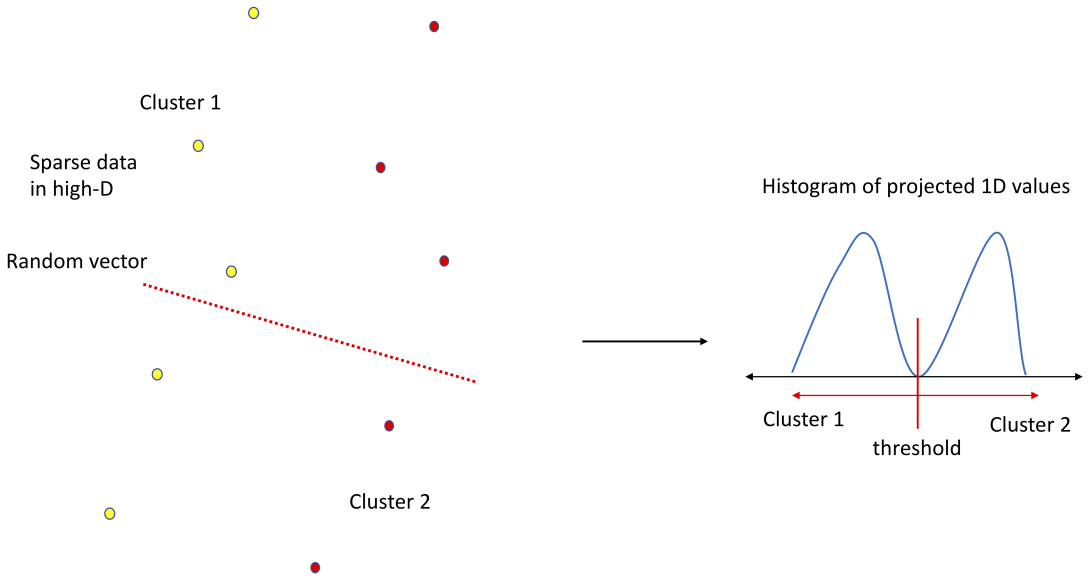


Fig. 3.1. The scenario where the data distribution is sparse in the original high-dimensional space and projection onto a random vector yields a binary clustering

3.3.1 Clusterability Quantification

Our method works when there exists structure in the high-dimensional space that, with a high likelihood manifests itself as a bi-modal distribution in a random projected 1D subspace. We quantify the clusterability of the probability density function $\rho(z)$ underlying the distribution of the projections $\mathbf{z}$ of the data using an empirical estimate of the quantity $\mathbf{S}$ introduced in [37], which is defined as:

$$S = S(\rho(z)) = \frac{1}{\sigma^2} \min_{T \in \mathbb{R}} \int_{-\infty}^{T} (z - \mu_-(T))^2 \rho(z) dz + \int_{T}^{\infty} (z - \mu_+(T))^2 \rho(z) dz,$$

where

$$\sigma^2 = \int_{-\infty}^{\infty} (z - \mu)^2 \rho(z) dz, \quad \mu = \int_{-\infty}^{\infty} z\rho(z) dz, \quad \mu_-(T) = \frac{1}{\pi_-(T)} \int_{-\infty}^{T} z\rho(z) dz,$$

$$\mu_+(T) = \frac{1}{\pi_+(T)} \int_{T}^{\infty} z\rho(z) dz, \quad \pi_-(T) = \int_{-\infty}^{T} \rho(z) dz, \quad \pi_+(T) = \int_{T}^{\infty} \rho(z) dz.$$

The quantity **S** measures to what extent a 1D distribution is divided into two clusters. Note that T corresponds to the position of the threshold and the $\pm$ signs correspond to clusters in Figure 3.1. A good binary clustering corresponds to a low value of S since this would imply that the within cluster variance is small. The probability density function of **S** should have a non-negligible mass below $S = 0.36$ to indicate presence of binary clusters as discussed in detail in [37]. Based on our empirical findings, if $S < 0.36$ for an instance of a random vector induced clustering, that instance of clustering is deemed "good" with a lower value of S indicating a better instance of clustering, i.e. sharper bi-modal distribution of projection values.

As discussed in [37], the $S = 0.36$ threshold is obtained by analyzing a null hypothesis, wherein data is generated from a single multivariate Gaussian, simulating absence of clusters. This null data is then projected onto random vectors following which S is estimated for the resulting projections. It was proved in [37] that the distribution of S for the null Gaussian data peaked at 0.36 irrespective of other parameter settings. Hence, the value of $S = 0.36$ is a rough threshold for deciding which projections are well clustered. The smaller the value of S below $S = 0.36$, the further (more clustered) the projections deviate from the null hypothesis.

3.3.2 Clustering

n-TARP is a non-deterministic method that is designed to seek and generate various statistically significant binary clusterings. We want to emphasize the possible existence of several possible clustering structures in high-dimensional space. Since not all random projections will yield a good separation, we use the TARP approach n times and pick the best quality cluster as determined by our empirical estimate of **S**. To check for statistical validity, the data is split into two sets at random: Training and Validity Testing. The purpose is to find the best cluster using S as a metric on the Training set in the 1D projection space and check if the resulting clustering rule generates statistically valid groupings on the Validity Testing set. Note that our

statistical test is performed in the 1D space of the projected values $\mathbb{R}$, thereby avoiding the challenges of statistical validity testing in higher dimensions. Presented below are the steps carried out in the two n-TARP phases of Training (implementation at [94]) and Validity Testing.

Training:

Let $x_1, \ldots, x_{m_1} \in \mathbb{R}^p$ be the training points.

1. For $i = 1$ to n:
2. Generate a random vector r_i in $\mathbb{R}^p$;
3. Project each x_j onto r_i by taking the dot product $z_{i,j} = (x_j \cdot r_i)$;
4. Empirical estimation of S
   (a) Use k-means with $k = 2$ to find 2 clusters in the projected space of z;
   (b) Estimate S_i for this clustering as

$$S_i = S\left(z_{i,1}, \cdots, z_{i,m_1}\right) = \frac{\sum_{l \in C_1} \left(z_{i,l} - \mu_1\right)^2 + \sum_{l \in C_2} \left(z_{i,l} - \mu_2\right)^2}{\tilde{\sigma}^2 \cdot m_1},$$

   where C_1, C_2 are the assigned clusters with their respective means μ_1, μ_2 and $\tilde{\sigma}$ is the empirical standard deviation of all the projections $z_{i,j}$.
5. End loop.
6. Store the vector r^* associated with the smallest S_i.
7. Compute and store the threshold t^* separating the two clusters such that $z \lessgtr t^*$ determines the cluster assignment;

Validity Testing:

Let $y_1, \ldots, y_{m_2} \in \mathbb{R}^p$ be the testing points.

1. Import r^* and t^* from the training phase;

2. Project each testing point y_j onto the vector r^* by taking the dot product $(y_j \cdot r^*)$;

3. Use the threshold t^* to assign a cluster to each of the y_j;

4. Perform permutation test with Monte-Carlo simulations [50] on the projected test data at statistical significance level of 99%.

There are many ways to generate the random vectors r_i. For simplicity, each coordinate is generated using a i.i.d standard normal probability model $\mathcal{N}(0, 1)$. There are also many ways to pick an appropriate threshold t^*. Again for simplicity, we compute the threshold as the halfway point between the extreme ends (the closest pair) of the projected values from the two clusters in the training phase. Note that n is a user selected parameter, with higher values of n more likely to yield better binary clusters as per the S metric. A simplified high level overview of the method is shown in Figure 3.2.

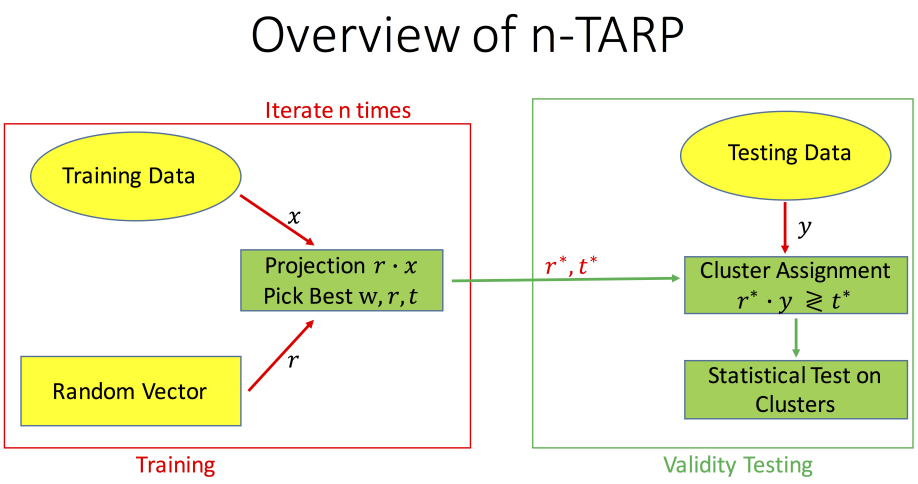


Fig. 3.2. Block diagram overview of n-TARP

3.3.3 Comments and insights on n-TARP

Note that each time the method is run, a different random vector is generated. Therefore, the criteria (feature) used to cluster is different in every attempt. Hence, every instance of n-TARP yields clusters resulting from different random vectors (i.e. different similarity measures). The probability of getting nearly identical random vectors in several iterations of n-TARP is negligible since the elements of the random vector are independent, continuous valued and drawn without bias. It is however still possible to get repeated cluster assignments resulting from distinct random vectors as different clustering criteria may result in the same grouping of samples. However, our experiments (Sections 3.4.2, 3.4.4, 3.4.5) show that a large number of the clusters are distinct.

The statistical significance test for the grouping using a permutation test is appropriate when the data set is small and the assumptions for other tests like the T-test do not hold. For other scenarios with larger datasets, an alternative statistical test can be picked/designed that is suitable to the application and the dataset. By performing the test in the projected space, we can validate if the clustering rule learned in the training phase generalizes to unseen data in the testing set and consequently is not biased by the training data. Performing statistical tests in high-dimensions is challenging due to the sparsity of data in that space and often requires a large sample size for reliable results. By following the projection space approach afforded by n-TARP, statistical significance is evaluated quickly. The advantages of n-TARP are most prominent on small data problems, but the principles involved apply equally to big data problems.

Further, n-TARP can be viewed as a basic unit similar to a single neuron/layer in a neural network. This unit can be combined or stacked together to make it more powerful. A tree structure based incorporation of n-TARP is presented in [36, 37] while an extension of the same framework to the task of classification is presented in [38]. In this chapter, we will focus on a single n-TARP unit for simplicity. In our

experience, a single n-TARP unit is more appropriate for small data problems, which will be the main focus of the rest of this chapter. Other architectures of clustering that combine several n-TARP units are appropriate for big data problems and the reader is referred to [37] for some examples of the same. Those examples do not contain checks for statistical validity, rather, they consider clustering accuracy to measure effectiveness of the method.

3.4 Experiments

Our main focus for n-TARP is on small data problems, wherein we have a small number of data samples in high-dimensional space. To this end, we will use an educational research dataset [55] to illustrate the effectiveness and utility of n-TARP. This dataset relates to attributes indicative of student learning comprising of data for 27 students. Details of how this data was acquired and processed to form feature vectors can be found in [39], and are detailed in Chapter 5. The dataset consists of qualitative data relating to students' Habits of Mind [56] that are evaluated based on a rubric [55]. The data is in the form of text, figures and equations which are annotated by hand following the developed rubric. These annotations are mapped to quantitative data by modeling them in a probabilistic framework. After processing, the dataset comprises 27 data samples, each represented by a 26 dimensional real-valued feature vector.

3.4.1 Clusterability of the Dataset

We first present some supporting results for our hypothesis that data in high dimensions has a lot of hidden structure that can be uncovered through simple random projections. This is illustrated by the distribution of $\mathbf{S}$ obtained through one run of n-TARP (without validation). Specifically, we set $n = 500$ and approximate S for each of the resulting 500 cluster groupings (steps 1 - 5 of Training phase) using roughly

half the data (13 samples) picked randomly. The empirical cumulative distribution function (CDF) for S is shown in Figure 3.3 with a vertical boundary at $S = 0.36$.

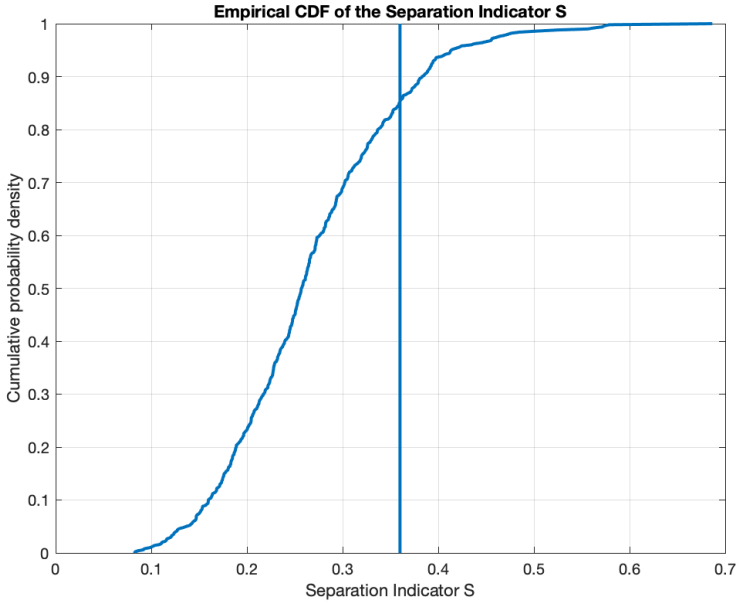


Fig. 3.3. Empirical cumulative density function of separation indicator S. The vertical line corresponds to $S = 0.36$

From Figure 3.3, we observe that approximately 85% of cluster groupings formed in this one run of n-TARP produced values of $S < 0.36$ and the distribution of S is skewed towards lower values. This indicates the existence of several different grouping criteria that yield binary clusters. The criteria are different for each of the 500 iterations since the probability of generating repeated identical random vectors is negligible.

Based on these results, we infer that the dataset has some kind of hidden structure in the 26 dimensional space that can be extracted through simple projections onto randomly generated vectors in the same 26D space with a high likelihood. Further, there are several possible ways of extracting this structure instead of a unique "best"

way, as evidenced by the large fraction of $S < 0.36$. Next, we present results on the statistical validity of clusters formed using n-TARP.

3.4.2 Statistically Significant Clusters

In this section, we will compare n-TARP against k-means [80] along with the following methods designed specifically for high-dimensions: Proclus [84], Clique [85], Doc [86], Fires [87], INSCY [88], Mineclus [89], P3C [90], Schism [91], Statpc [92] and SubClus [93]. The high-dimensional methods mentioned were used through the Weka [95] implementations.

The experiment is designed as follows: We will use the educational research dataset we briefly described in the previous section that consists of 27 data samples in 26 dimensional space. We will go into the details of how this data was acquired and processed in Chapter 5. Except for n-TARP, all the other methods will use all of the data samples available to form clusters. Due to the two phase approach described previously, n-TARP will use about half the dataset for training and the other half for validity testing. All the methods will be run 1000 times to analyze how many different clusters can be formed by each. The statistical validity of the resulting clusters will also be analyzed.

In our experiments, we found that Clique, Doc, INSCY, Mineclus and P3C did not form any clusters while running Schism, Statpc and SubClus always resulted in errors over many attempts. Fires produced overlapping cluster assignments which did not fit our experiment. We suspect these issues to have been a consequence of the small dataset size. Further, the Weka implementations of these methods did not allow us to easily run these methods in a loop to see results over 1000 attempts. Hence, we ran these methods manually and encountered the results we described above. It is important to note the difficulty of getting these methods to work in the small data scenario we are analyzing. We were successful in getting results only with Proclus and k-means along with n-TARP, where Proclus was run manually for a few attempts

until repeating clusters were observed, while k-means and n-TARP were run in an automated manner for 1000 attempts.

For all the methods we tested, if a cluster assignment cannot be formed, an invalid flag is raised. This event is counted as a clustering attempt but not as an instance of clustering. For example, with n-TARP, we found that for statistically invalid cluster instances, all samples were grouped into a single cluster which is not valid as a grouping was not produced. Hence, this counts as an attempt but not as a cluster assignment.

For n-TARP, we first set $n = 500$ (other values of n are experimented with in the next section) and ran the method a total of 1000 times. For each individual run of the clustering, we randomly split the students into one group of 13 for training and another group of 14 for testing. The vast majority (68%) of the attempts resulted in distinct clusters (Table 3.1). Here, distinct is quantified by at least one element of a cluster being different. For larger datasets, one should probably increase the number of differing elements needed to qualify distinctness of clusters. Since our sample size is very small, even a difference of one element is significant. Note that if the method repeatedly finds the same clusters with just one element being switched between them, then the number of distinct clusters formed would be very small. This is not the case with n-TARP as the number of distinct clusters is very high.

We found that about 80% of the n-TARP attempts resulted in statistically significant clusters. This is a very encouraging indicator that the grouping criteria learned by n-TARP on a training subset of the data generalizes well to the unseen testing data. Further, n-TARP is able to learn several distinct statistically valid grouping criteria instead of repeatedly finding just a finite few. Our analysis further revealed that in this data scenario, whenever a cluster assignment was formed by n-TARP, it was also statistically valid. Instances where the validity test was not passed were also the cases where cluster assignments could not be formed. Hence, the difference between "Stat Sig in proj 1D" column and the "Distinct Clusters" column reveals the fraction of repeated groupings in Table 3.1.

Table 3.1.
Comparison of nature of clusters formed by different methods in 26D feature space

Method	Distinct Clusters % (#)	Stat Sig in high D	Stat Sig in proj 1D
k-means	2.60 % (52)	22.7 %	N.A.
n-TARP	68.10 % (1362)	16.7 %	81.9 %
Proclus	100 % (4)	25 %	N.A.

In comparison, we also ran k-means with $k = 2$, 1000 times on the entire dataset. Unsurprisingly, distinct clusters were only obtained in 2.6% of the 1000 attempts (52 different clusters). We checked for the statistical significance of these clusters (in the original 26 dimensional space) using the high-dimensional version of the permutation test with Monte-Carlo simulations [50] and found that 22.7% of the total attempts resulted in statistically significant clusters in the original high-dimensional space. Overall, only 20 distinct and statistically significant groupings were obtained with this method with 6 repeated groupings.

Finally, for Proclus, since we were running the method manually, we observed that only 4 distinct groupings were being formed with many repetitions. Of those, only 1 was found to be statistically valid in the original high-dimensional space. The Weka software did not produce the projection space for analysis due to which we couldn't analyze this grouping in its projected state. Further, the projection space would not be a trivial one-dimensional space that we are considering in this experiment, hence, this comparison would not be appropriate.

It is important to note here that the statistical tests in 26D space may not be highly reliable due to the small sample size as factors like the estimation of the 26×26 covariance matrix may not be very reliable. This problem is compounded for n-TARP as only 14 samples are available for statistical tests instead of 27 for

the other methods. Due to this reason, the tests in 1D are more reliable but we could only perform these tests for n-TARP since k-means does not have a projection subspace while Proclus is unlikely to use a 1D subspace. Observe the huge gap between statistically valid clusters as determined in high-D vs 1D for n-TARP.

Ultimately, our goal with n-TARP is to extract a structure that was manifested as point separations in projected space. To that end, the 1D analysis is the correct approach; we provide high-dimensional analysis for further information only, since the other methods are not compatible for a comparable 1D analysis. We note that n-TARP creates significantly more distinct clusters than other clustering approaches, if they can function in the small data scenario to begin with. As far as we know, our proposed clustering method n-TARP is the only one that can reliably produce a large number of statistically significant and/or distinct clusters for such a small dataset.

3.4.3 Effect of n on Clusters

The number of trial attempts n is an important parameter of the clustering method. It influences the odds of finding a good binary clustering in terms of a low S score. If we allow a larger number of trials, we are more likely to find a clustering criterion with a lower S score. We now investigate the effects of this parameter on the clusters formed. Specifically, we focus on how the fraction of statistically significant clusters and distinct clusters varies as we change n. In our experiment, we varied the parameter over $n = 1, 2, 10, 50, 100, 300, 500$ and each value of n is used to generate 1000 clusters. The results are presented in Figure 3.4, where n is in logarithmic scale.

We observe that the fraction of statistically significant clusters initially increases with increasing n and then eventually saturates around 80% once $n \geq 10$. The initial sharp increase can be explained by a corresponding increase in the odds of finding good binary groupings that are statistically valid. A larger n affords more freedom to n-TARP in exploring the structure hidden in the data and finding statistically valid clusters. There is an eventual saturation point when n-TARP has fully explored the

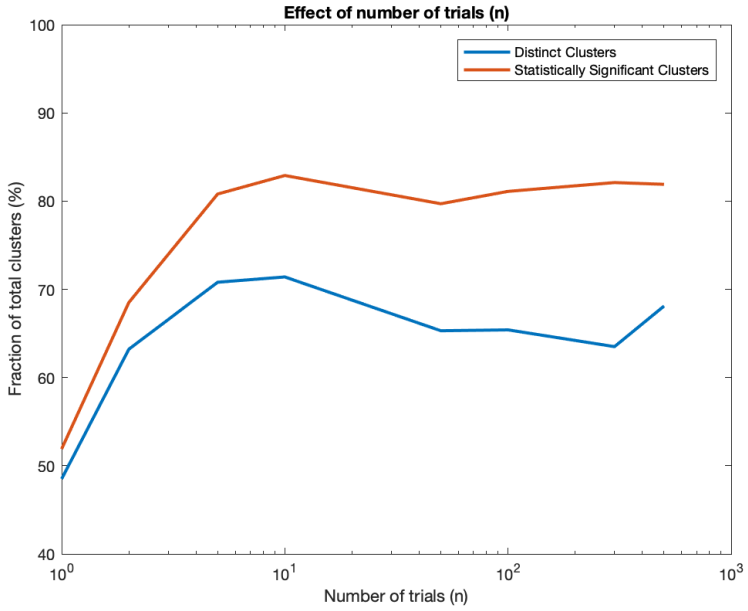


Fig. 3.4. Evolution of distinct clusters and statistically valid clusters with increasing number of trials n

hidden structure and identified all statistically valid groupings. Increasing n beyond this saturation point will cause diminishing returns in terms of finding more groupings and will likely not yield any significant changes as observed in Figure 3.4 for $n \geq 10$. Note that we are explicitly talking of saturation in terms of finding all statistically valid groupings and not criteria. There is a subtle distinction between the two as several different criteria resulting from distinct random vectors can yield repeated cluster assignments.

Next, we investigate the more interesting case of the fraction of distinct clusters. From Figure 3.4, we find that there is an initial sharp increase upto about $n = 10$ followed by a gradual decrease and eventual saturation as the value of n increases further. The initial increase can be explained by a similar argument as for the statistically valid clusters above. As n initially gets larger, there are higher chances of finding good binary groupings. Since n at this stage is not too large, there is

lower probability of finding repeated cluster assignments corresponding to the lowest S scores in each run of n-TARP. Hence, there is a higher chance of finding distinct clusters within the 1000 runs of this experiment.

In contrast, when n gets larger beyond a certain dataset dependent limit, the search space of n potential groupings is growing bigger and the likelihood of finding repeated cluster assignments is increasing over many attempts of n-TARP. Consequently, the fraction of distinct clusters goes down since the same low S scoring clusters are found repeatedly for higher values of n. This decrease is expected to continue until only the lowest S scoring clusters are found repeatedly in every run of n-TARP at which point there will be a saturation in the fraction of distinct clusters that will not change with increasing n. This effect is observed in Figure 3.4 for $n \geq 10$, with $n = 10$ being the maxima of the curve. We observe that for $n = 50$ to 500, the fraction of distinct clusters has small deviations around 65% and more or less exhibits a saturation region.

3.4.4 Feature Space Extension

The 26 dimensional features of the dataset can be considered as linear first order features representing the data. It is possible that combinations of these features may provide more information about the underlying structure of data, similar to switching from a linear kernel to a higher order polynomial kernel. For example, if we consider order 2 features (i.e. all monomials of degree two in the feature coordinates), in addition to the 26 linear order 1 features $f_i, i = 1, ..., 26$, we will add terms of the form $f_i f_j, \forall i, j = 1, ..., 26$ to the exisiting feature vector.

Following this idea, we expanded the feature space from order 1 to both orders 2 and 3 (monomials of degree 3) which expanded the feature space dimensions from 26 to 377 and 3003 respectively. The cluster analysis data using $n = 500$ and running the method 1000 times for these modified feature spaces is shown in Table 3.2.

Table 3.2.
Cluster Analysis for n-TARP with varying feature space orders including anomaly

Feature Order	Dimensions	Distinct Clusters	Stat Sig in proj 1D space
1	26	61.2 %	82.5 %
2	377	38.2 %	44.6 %
3	3003	20.5 %	26.5 %

We observe that as the order increases, the fraction of distinct clusters is decreasing. Similar to this trend, the total fraction of statistically significant clusters is also decreasing with increasing dimensions. This is indicative of a changing nature in the structure of the data which is possibly promoting some structures over others with increasing dimensions as fewer distinct groupings are formed. Note that the data for Table 3.2 was generated in a separate instance than for Table 3.1 which explains the differing data for order 1 n-TARP. For order 3, 20.5 % distinct groupings corresponds to 205 different instances of statistically valid cluster assignments in 3003 dimensional space with just 13 training samples to use for learning a clustering criterion, which is quite remarkable.

We further investigated the clusters formed for various orders and found that a significant number of clusters were formed by separating just 1 sample (index 12 in our dataset) from the rest of the data. The occurrence of this particular grouping increased with increasing order. Further, each instance of this grouping resulted from a different random vector, implying that several different criteria were found that separated sample 12 from the rest of the data. This is a curious case of an outlier in the dataset, especially considering that it occurred more frequently as the feature space order grew higher. To gain further insight into this outlier, we ran the same

experiments again with a single change of removing sample 12 from the dataset, shown in Table 3.3.

Table 3.3.
Cluster Analysis for n-TARP with varying feature space orders with anomaly removed

Feature Order	Dimensions	Distinct Clusters	Stat Sig in proj 1D space
1	26	56.7 %	77.5 %
2	377	42.7 %	52.2 %
3	3003	36.4 %	49.2 %

3.4.5 Outlier Detection

From Table 3.3, we notice that the number of distinct clusters found for order 1 was reduced after removing the suspected outlier. However, the number of distinct clusters has increased for orders 2 and 3 with the outlier removed. This indicates that the unique nature of sample 12 was predominant in orders 2 and 3, which influenced the learned clustering criteria to separate it from the rest of the dataset. When sample 12 was removed, we no longer observe this biased clustering criteria and the method is free to explore other candidate grouping criteria and consequently finds a larger number of distinct statistically valid clusters than previously. There is no longer a predominant repeated cluster assignment of 1 sample vs rest of the dataset.

With the removal of the outlier, for order 3 features, n-TARP is able to find about 36% or 360 instances of unique statistically valid grouping criteria in 3003 dimensional space with just 13 samples to learn a criterion. To our knowledge, no other method can yield this kind of performance in a small data problem. Considering the difficulties of running other methods for the purpose of comparison in order 1 features, it is highly unlikely that they would present any better performance, if they function at all, on

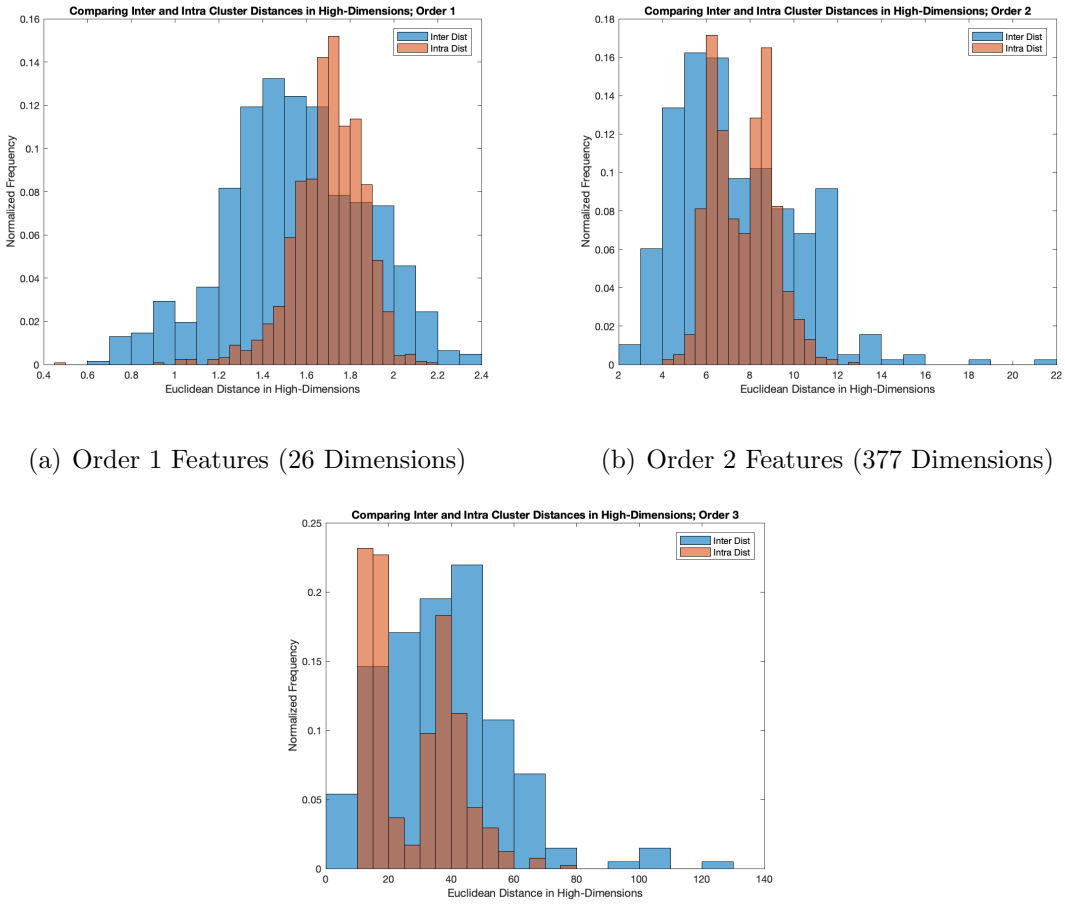
the same number of samples in a much larger dimensional feature space and hence we did not attempt such comparisons.

It is important to note that n-TARP without statistical validity checks can yield invalid clusters as evidenced by the dropping number of statistically valid clusters found as the feature space increased in Tables 3.2 and 3.3. Having a built in statistical test has proven to be insightful and informative, especially in high dimensions.

3.4.6 Analysis of Euclidean Distances

Our proposed n-TARP method depends on point separations in the projected 1D space to form clusters instead of utilizing euclidean distance metrics in the original high-D space. This is an important aspect of the method since euclidean distance as a metric for point similarity is not very reliable as the dimensions of the data keeps growing. To illustrate this, we examine the clusters formed in the feature extension experiment and summarized in Table 3.2. We consider the statistically valid distinct clusters that were found. Specifically, for every pair of clusters formed, we compare the inter-cluster and mean intra-cluster pairwise euclidean distances. The inter-cluster distance is calculated as the euclidean distance between the cluster means while the mean intra-cluster distance is calculated as the average euclidean distance of data samples from the mean of the cluster they are assigned to. These distances are calculated in the original high-D space, for all feature orders. This comparison is shown below in Figure 3.5 through normalized histograms.

The inter-cluster distances are represented in blue while the intra-cluster distances are represented in orange in Figure 3.5, while the overlap between the two is shown in a darker shade. We observe a common theme in all three graphs in Figure 3.5 which is a large overlap in the distribution of inter-cluster and intra-cluster distances corresponding to statistically valid clusters. This indicates that the distance between clusters is similar to the distances between data samples present within a cluster. In other words, based on euclidean distances alone, it is not feasible to determine if a



(a) Order 1 Features (26 Dimensions)

(b) Order 2 Features (377 Dimensions)

(c) Order 3 Features (3003 Dimensions)

Fig. 3.5. Inter-Cluster and Intra-Cluster Euclidean Distance Comparison for Varying Feature Extensions Orders

given data sample belongs within a cluster or outside it. This is an illustration of how similarity metrics based on euclidean distance measures are not effective in high-dimensions, which motivates the utility of our projection based clustering method.

As a closing remark on the experiments, we highlight the advantage of n-TARP in finding statistically valid clusters even in situations where conventional metrics like the l_2 norm can no longer provide good discriminating power as illustrated in the experimental results described above.

3.5 Conclusions

In this chapter, we introduced a novel clustering method called n-TARP, which is non-deterministic and based on point separations instead of point proximity. The method is designed to explore multiple clusters in data that are statistically significant rather than a unique "best" grouping of data into clusters. The method is geared towards high-dimensional data and its utility is most prominent in a high-dimensional setting with a small dataset, although it can easily be extended to lower dimensions and big data settings as well. This chapter focused on a small data problem in high-dimensional space [39, 55] as this was the context for the development of the method as presented in this chapter.

The central idea of the method is the projection of data onto randomly generated vectors. Our previous work [36, 37] has shown that data in high-dimensional space has hidden structure that can be uncovered through this projection process. We have shown empirical evidence to support this hypothesis for the dataset considered in this chapter through the quantification of clusterability illustrated in Figure 3.3.

This chapter proposed statistical validity evaluation in the projected 1D space where we hypothesised the hidden structure of high-dimensional data is manifested [37]. This approach allows us to bypass statistical validity testing in high dimensions with a small sample size. The clustering criterion is learned from half of the data followed by the statistical validity test on the remaining half. Through this two step procedure we are able to determine if the learned grouping criterion generalizes to the whole dataset. This allows us to generate a large number of distinct statistically significant clusters for a small dataset in high-dimensional space, something that to our knowledge has not been possible before.

Through experiments on the high-dimensional small data problem considered in this chapter, we have quantified the indicators of hidden structure in the data and produced several hundreds of distinct statistically valid clusters in a variety of high-dimensional settings, from 26 to 3003 dimensions. We have also shown an illustrative

analysis to detect outliers in high-dimensions, which is a challenging problem. Further, we illustrated that conventional similarity metrics like the l_2 distance is not a feasible option in high-dimensions and can lead to erroneous and statistically invalid results. The point separation approach used in n-TARP provides a good alternative and can find potentially a large number of statistically valid clusters in such extreme scenarios.

In conclusion, our work has shown that data in high-dimensions has a lot of structure that is not limited to a unique "best" set of clusters. This structure manifests itself as point separations in random projected 1D subspaces leading to several distinct sets of clusters. The cluster assignments themselves can be viewed as random variables induced by the random projections of n-TARP. Our results point to a need to study these cluster distributions to gain better understanding of high-dimensional structures.

4. PATTERN DEPENDENCE AND CLUSTER EVALUATION

The contents of this chapter are an extension of [40].

4.1 Introduction

So far, we have studied the utility of random projections for grouping high-dimensional data in Chapters 2 and 3. While the interpretation of the accuracy of the results in a supervised learning scenario is quite straightforward, the accuracy of the clusters in the unsupervised scenario is not. We are interested in the question of how one can interpret and evaluate clusters in the absence of labels, as well as how to use the clusters once they are obtained.

Chapter 3 showed that n-TARP based clustering yields a multitude of different sets of clusterings that are statistically valid. In this chapter, we will study how to determine the existence of a statistical effect caused by the separation of data into groups as determined by n-TARP. We propose to carry out this study in a predictor response setting, wherein the data used for clustering is treated as a set of predictors. An independent quantity of interest associated with every data sample, which was not used in the clustering process, is chosen as a response variable. In the following sections, we will describe our framework to study the effects of the clustered predictor variables on the response variable. The existence of an effect provides evidence that the clusters found are meaningful and truly present in the data.

Specifically, this chapter is concerned with the problem of investigating if a relationship exists between a response variable (in $\mathbb{R}$) modelled as a random variable and a vector of predictor variables (in $\mathbb{R}^p$) modelled as a random vector. The vector of predictor variables corresponds to the data samples that are clustered, where the

data is continuous and quantitative in nature. We further focus on the scenario of a very small sample size m (20-30 samples), corresponding to the data we have been working with in Chapter 3 and will discuss in greater detail in Chapter 5. This is a challenging problem as potential solutions and models might over-fit to the small amount of data available. The framework we present can be modified easily to extend to scenarios with larger sample sizes.

4.2 Background and Overview

A popular approach to the general problem of determining relationships between predictors and a response is regression, e.g. multiple regression. However, the problem is ill-posed and under-determined in the scenario where $m < p$, i.e., we have more variables than data-samples. In this case, the least-squares approach can be used to generate the minimum norm solution with the Moore-Penrose pseudo-inverse [96, 97]. This is just one of infinitely many solutions and exploring other solutions may prove useful. The problem at hand could also have a large condition number, which could lead to errors in the pseudo-inverse solution [98]. To address this, regularization in terms of prior models can be introduced, which inherently biases the solution. There exist several choices for priors. Popular examples include l_2 regularization as in ridge regression [52], l_1 regularization as in LASSO [53] and a combination of both in Elastic Nets [54], all of which promote sparsity of coefficients corresponding to the predictors. A limitation of this approach is that a particular functional relationship has to be chosen and imposed among several choices, followed by a suitable choice for the regularization parameters. An alternative is to use Bayesian Model Averaging approaches [99] wherein several models are combined in a Bayesian manner to predict the outcome variable from the predictor variables using various functional models as ingredients. Bayesian hierarchical shrinkage priors in regression are considered in [100]. Another approach is based on variable selection in a probabilistic manner,

like the SSVS (Stochastic Search Variable Selection) [101] and some others discussed in [102].

While these methods can work in finding a functional relationship that minimizes some error metric of prediction under the $m < p$ scenario, it is still quite challenging to find a solution when m is very small. As mentioned previously, solutions (if any) may have over-fitting issues leading to high variance of the model or under-fitting based on model assumptions leading to high bias. This challenging problem may swing between the extreme ends of the bias-variance trade-off, while ideally, we would like to find a sweet spot between the two. The statistical validation of these solutions may also be very challenging given the small sample size (20-30) and unknown distributions of the predictors and response variables, which limits our capability to invoke the central limit theorem and the law of large numbers [103], which is an underlying assumption in a variety of statistical tests.

Some other general statistical tests and approaches to evaluating a dependence between predictor and response are the Chi Square test [104] and Fisher's exact test [105, 106]. Fisher's exact test is more general and makes less assumptions than Pearson's Chi Square test, however, they are both designed for categorical variables which limits their utility for analyzing continuous quantitative data. These tests cannot be used to check for dependence between multivariate quantitative data and some response uni-variate quantitative data. The closest related approach to this is an element-wise dependence check that is limited in scope as it will ignore combination effects of the predictors that can cause changes in the response. Hence, approaches like element-wise correlation [103] of each predictor with the response is not feasible. Some studies on dependence between two variables include [107, 108].

In summary, we have a small data problem with the number of samples m in the 20-30 range with the data in high-dimensional space (large p), which prohibits the use of the central limit theorem and the law of large numbers. It also poses a challenge of finding statistically valid solutions that do not overfit the data. Next, we do not seek a precise functional relationship between the predictors and the response variables.

Instead, we seek to answer the more general question of whether a relationship exists between the predictors (as a whole vector entity) and the response without imposing a model or functional form. We ask the question, "Can the chosen response be predicted from the predictors?", rather than assuming that it can and imposing a functional relationship and determining its parameters. Building on this, we also study the effects of subsets of clustered predictors on the response to understand how certain predictors may/may not affect the response. Further, we also seek a statistical validity framework that is appropriate, general, non-parametric and has minimal assumptions, for our scenario of small data in high-dimensions.

4.3 The Pattern Dependence Framework

To study the existence of a relationship between multivariate predictors and a uni-variate response, we must first define a quantifiable uni-variate response. Since we are working with the same data used in Section 3.4, to be described in detail in Section 5.3, the data samples correspond to student learning indicators. We find a suitable numeric student performance metric associated with each data sample (course grade), which we will designate as our response variable associated with each student. The response variable is not part of the data used for clustering.

Our goal is to find a quantifiable measure of the difference in the response variable based on partitioning the predictors into groups. Consider that the predictors associated with students are used to form two groups: Group 1 and Group 2. Let the mean response variable associated with the students in Group 1 be μ_1. In other words, the average grade of the students in Group 1 is denoted by μ_1. Similarly, the mean response for Group 2 is μ_2. Let Δ be the quantity that measures differences in response based on the groups. For simplicity, Δ will be taken as the absolute difference between the means of the grouped response variables, i.e. $\Delta = |\mu_1 - \mu_2|$.

Consider the example shown in Table 4.1 (reproduced from Chapter 5) which contains the distribution of the response variable (grade) for an instance of n-TARP

Table 4.1.
Example distribution of response variable (student grade) based on clusters formed using predictors (student skills)

Response Variable Grade	All Students	Cluster 1	Cluster 2
A (4.0)	5	0	5
B (3.0)	10	2	8
C (2.0)	8	4	4
D (1.0)	2	2	0
F (0.0)	2	2	0
Mean Grade	2.51	1.60	3.05
Standard Deviation	1.12	1.07	0.74

clustering using the predictor variables (student skills) as discussed in Chapter 3. We observe a significant deviation of the response distribution between the response (grade) specific to the clusters as well as from the original total distribution (Column "All Students"). This difference in response corresponds to a value of $\Delta = |1.60 - 3.05| = 1.45$ for this instance of clustering.

The n-TARP clustering method discussed in Chapter 3 is used to find clusters using the students' predictor data. The measure of response difference, Δ, is calculated for each instance of the n-TARP binary cluster assignment. Since the n-TARP clustering method is stochastic in nature and it determines how the response variables are partitioned in our observed clusters, Δ can also be considered a random variable with a distribution that is induced by the n-TARP clustering process. The distribution of Δ can be empirically estimated through the observations of Δ for every instance of n-TARP clustering.

The pattern dependence framework seeks to investigate if the response variables are dependent on patterns detected in the predictors (clusters). Any change in re-

sponse based on partitioning the data into clusters is measured by Δ. Since Δ is a random variable, we investigate the dependence (if it exists) by studying the empirical CDF (Cumulative Distribution Function) of Δ that is estimated through observations of Δ for every instance of n-TARP clustering. If we observe a large proportion of clusters associated with non-negligible values of Δ, in the context of the dataset, we can conclude that the response variable does indeed depend on the predictors and that the groups formed based on patterns in the predictors leads to a quantifiable difference in response.

4.3.1 Statistical Validation

A natural question that arises is the statistical validity of the conclusions drawn from the empirical CDFs. It is important to investigate if the observed CDF is any different from what would be found if students were grouped at random or if it is a causal result of the clustering process (i.e. affected by patterns in predictors). This necessitates a hypothesis test setup to compare the CDFs of Δ resulting from n-TARP clusters against a null distribution of Δ where the response variables are randomly divided into groups of the same size as that of the clusters, independent of the predictor variables. This hypothesis test must be performed at a high level of significance (above 95%).

The alternate CDF curve (resulting from the clustering process) is compared to a null-hypothesis CDF curve corresponding to the case with no relationship or dependency between the response variable and the clustering of the data samples using the predictors. The null CDF curve is obtained by randomly partitioning the students into two groups and computing Δ for these groups. This partitioning is repeated several times and the resulting sample values of Δ are used to estimate a CDF curve. The size of the random groups are chosen to match the various sizes of the groups obtained by clustering.

The null CDF curve obtained as described above will vary from one trial to the next. Thus, we compute the average CDF curve over several trials, as well as the CDF curves lying five standard deviations above/below the average curve. Let us fix a value of $\Delta = \Delta_0$; the values of all the curves obtained through our trials can be used to estimate the exact probability α_0 that a curve value at Δ_0 would be below five standard deviations under the mean value when the curve follows the null distribution. Let μ denote the mean curve value at Δ_0 and σ denote the associated standard deviation. The significance level estimated by the pair of curves for $\mu \pm 5\sigma$ corresponds to a significance level of at least 96% based on Chebyshev's inequality [109,110]. The inequality states that for a random variable X with finite mean μ and finite non-zero variance σ^2 and any real number $k > 0$,

$$Pr(|X - \mu| \geq k\sigma) \leq \frac{1}{k^2}$$

which for $k = 5$ means that there is at most a 4% chance of a sample lying more than 5 standard deviations away from the mean. This inequality does not make any assumptions on the underlying distribution of the random variable X and so is a conservative general bound, thereby guaranteeing a minimum significance level of 96% if a realization lies outside the $\pm\ 5\sigma$ boundary of the mean, which corresponds to a maximum value of $\alpha_0 \leq 0.04$.

The values of Δ at which the experimental CDF curve lies five standard deviations below the average null-hypothesis curve are highlighted; if Δ_0 is a value in the highlighted region, and if the experimental CDF curve value at Δ_0 is p_0, then p_0 fraction of the patterns have a difference of Δ_0 or less, and thus $1 - p_0$ fraction of the patterns have a difference of at least Δ_0. This conclusion is valid with probability $1 - \alpha_0$. Consequently, the lower a CDF curve is below mean null CDF, the larger the proportion of clusters with a minimum Δ_0 difference in response. Hence, a lower alternate CDF curve is preferred to obtain clusters with more noticeable differences in the response variable.

By following this process, we are able to achieve a graphical hypothesis test with a minimum significance level of 96%, where the domain of significance is highlighted.

If the alternate CDF falls outside the $\mu \pm 5\sigma$ zone of the null CDF, we can conclude that the alternate CDF does not follow the same distribution as the null case, with a maximum probability of erroneous conclusion being 4%. Hence, if we are able to reject the null hypothesis in this scenario, we conclude that the dependence observed between the predictor groups and response is a statistically valid and quantifiable effect.

4.4 Experiments

As mentioned before, we will continue to work with the data described and used in Chapter 3 to analyze the clusters formed using n-TARP under the pattern dependence framework. Hence, the data consists of 27 samples in 26 dimensional space, where the data relates to students' learning skills called Habits of Mind [39, 55]. The response variable associated with student performance is the course grade. Further, the predictor data corresponding to students can be broken into five separate components A,B,C,D and E as will be discussed in Chapter 5 and can be found in [39, 55]. We will also consider the feature space extension data discussed in Chapter 3, where we extend our feature vectors by including terms of order two and three, growing the space dimension to 377 and 3003, respectively. This extension of feature space is carried out in the same way as in the previous chapter. For example, we can extend the dimensionality of the feature vector $(f_1, f_2, ..., f_p)$ by including terms of order k of the form $f_{i_1} \cdot f_{i_2} \cdots f_{i_k} \; \forall \; i_1, i_2, \cdots i_k \in \{1, 2, ...p\}$, to check for non-linear dependencies.

In order to estimate the alternate CDF, we use the n-TARP clustering method to generate 10000 sets of binary clusters with the data and calulate the Δ based on the average grades of the students in each group for each instance of binary clustering. To estimate the null CDF, we randomly divide the student grades into two groups where the group sizes are the same as for the sets of clusters formed with n-TARP. Hence, we form up to 10000 random groups of grades. We repeat this process 100 times to get 100 estimated CDF curves of the null hypothesis. From these 100 curves,

we calculate the mean (μ) and standard deviation (σ) null CDF curves, which are then used to create three curves to represent the null hypothesis: the μ null CDF and the $\mu \pm 5\sigma$ curves that define the rejection zone.

4.4.1 Hypothesis Test on Empirical CDF

We first investigate the pattern dependence with order 1 linear features in 26 dimensions. In Figure 4.1, the alternate CDF is plotted in blue, the mean null CDF in black with the 5σ zone represented in dashed black lines. A magenta indicator curve at the bottom represents the locations in the domain of Δ where the alternate CDF is in the null hypothesis rejection zone, i.e. lies outside the $\mu \pm 5\sigma$ null zone. For the magenta curve, a high step (at y-axis value of 0.05) indicates that the alternate CDF lies outside the null zone and we can reject the null hypothesis whereas a low step (at y-axis value of 0) indicates that the alternate curve lies within the null zone and the null hypothesis cannot be rejected.

As we can observe from Figure 4.1, the alternate CDF curve (blue) lies more than 5 standard deviations below the null CDF curve for a large range of values of Δ. Hence, we can reject the null hypothesis and declare that grouping students based on their learning skills data does indeed have a non-random effect on their course performance. This claim can be made at a minimum of 96% significance level as discussed in the previous section. This leads us to the conclusion that the response variable (course grade) does have a statistically valid dependence on patterns in the predictors (high-dimensional data samples).

This pattern dependence is investigated further in Figure 4.2, where we have extended the feature space to order 2 with 377 dimensions (Figure 4.2(a)) and order 3 with 3003 dimensions (Figure 4.2(b)). This allows us to study non-linear extensions of the pattern dependence and see if the extent of dependence changes with dimensionality. From Figure 4.2, we observe that for both orders 2 and 3, the significance region indicated by the magenta curve is quite large, with the maximum

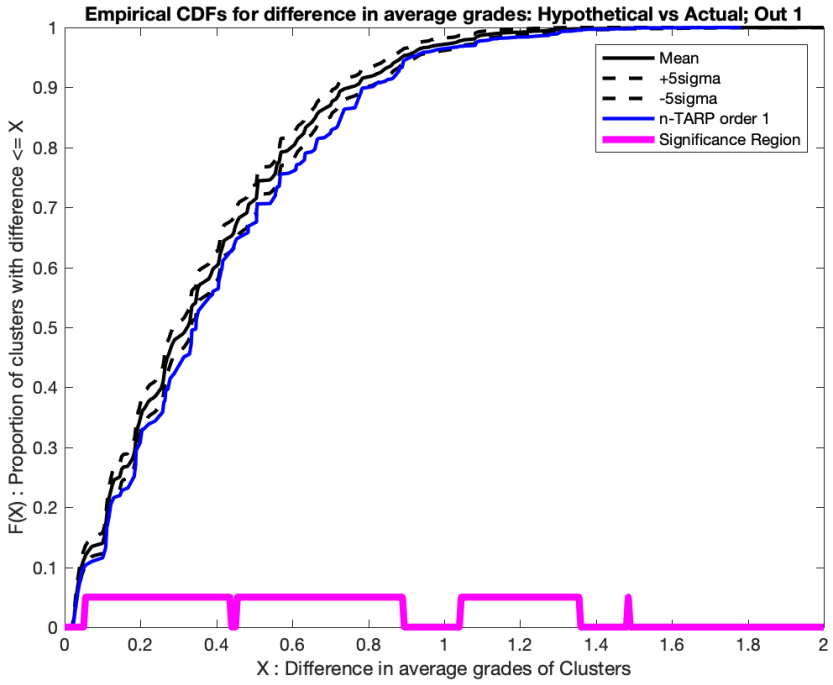
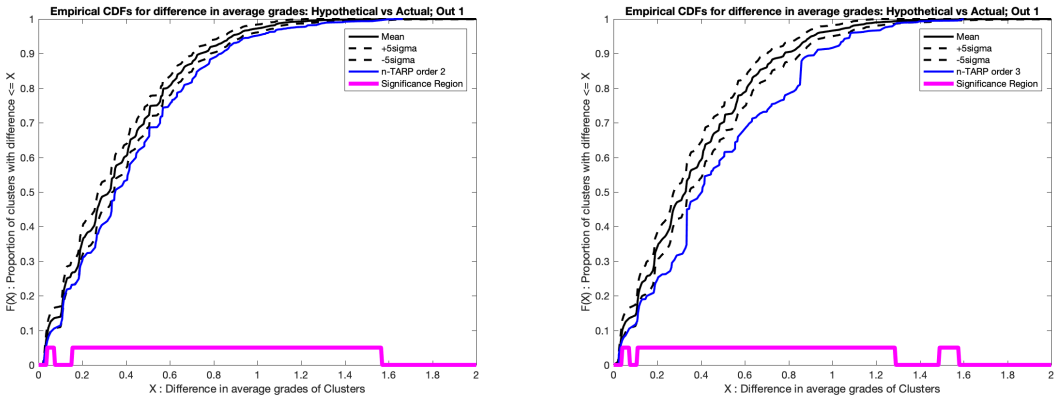


Fig. 4.1. The hypothesis test for pattern dependence with order 1 features (26 dimensions)

longest sequence being approximately 0.2 to 1.6 for order 2 in Figure 4.2(a). We also notice a larger margin between the null zone and the alternate CDF as the dimension grows, particularly in Figure 4.2(b). This larger margin may indicate a larger extent/strength of pattern dependence relative to the base line order 1 effect seen in Figure 4.1. Since a greater margin between the null and alternate CDFs implies a greater proportion of clusters that yield at least a given value of Δ, it can be interpreted as a relatively stronger dependence of the response on patterns in predictors. In other words, the curve of Figure 4.2(b) is the lowest of three curves considered and has the highest probability to yield clusters with large differences in their associated response variables.

Based on the experiments discussed above, we can infer that there is indeed a statistically valid pattern dependence effect observed between the predictors and re-



(a) order 2 features (377 dimensions)

(b) order 3 features (3003 dimensions)

Fig. 4.2. The hypothesis test for pattern dependence with higher order features

sponse for all orders of features. The large expanse of the significance region for all feature orders is an encouraging indicator of both the statistical validity of our clusters and the dependence between the predictors and response.

4.4.2 Feature Selection

As mentioned previously, each of the features of our data samples is related to one of 5 different components, denoted as A,B,C,D and E. These components of data arise from a rubric that will be discussed in greater detail in the next Chapter. We investigate whether the response is dependent on a given individual component by removing the feature coordinates corresponding to that component and recomputing the corresponding CDF curves. This CDF estimation is repeated 5 times, once each for the removal of all 5 components A,B,C,D and E in a sequential manner. The resulting CDF curves are compared to the baseline case wherein no components are removed.

The results using the set of order 1 features (linear relationship) in Figure 4.3 show that the response depends prominently on components A (red) and B (green) in

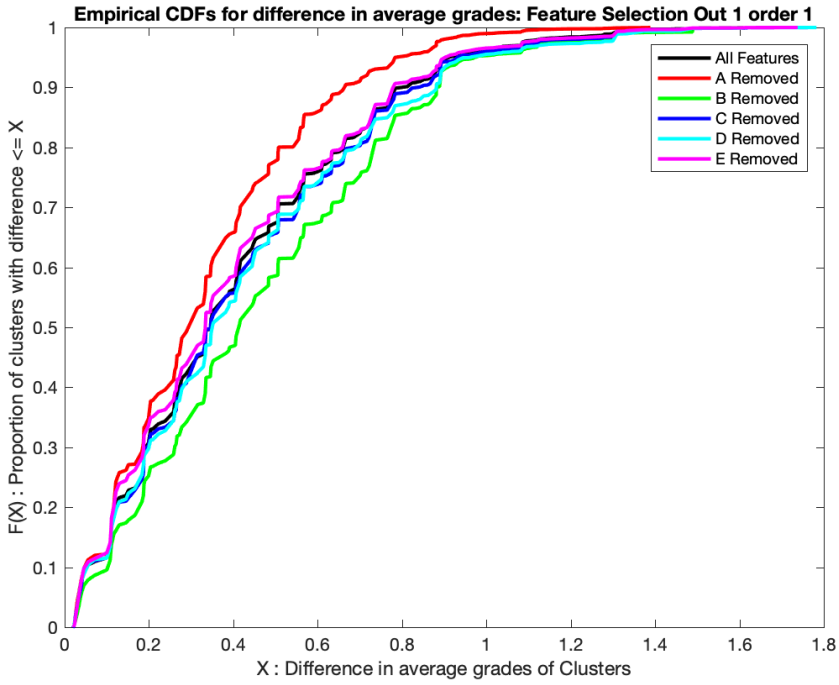


Fig. 4.3. Feature selection for pattern dependence with order 1 features (26 dimensions)

a linear fashion since their removal shifts the corresponding CDF curve significantly away from the baseline curve with all the components included (black). Removal of other feature components does not seem to affect the curves to a significant extent as the blue, cyan and magenta curves corresponding to removing components C,D and E respectively, seem to mostly overlap with the black curve. It is interesting that removing A seems to make it less probable to find a set of clusters with a contextually large $\Delta \geq 0.7$ for instance, whereas removing B seems to make it more probable. An analysis of this nature may help in determining which components of data contribute more significantly to an observed pattern dependence effect, which in this case appear to be components A and B.

We investigate if the effects of individual components change in scenarios with higher order features. Our results are shown for order 2 (Figure 4.4) and order

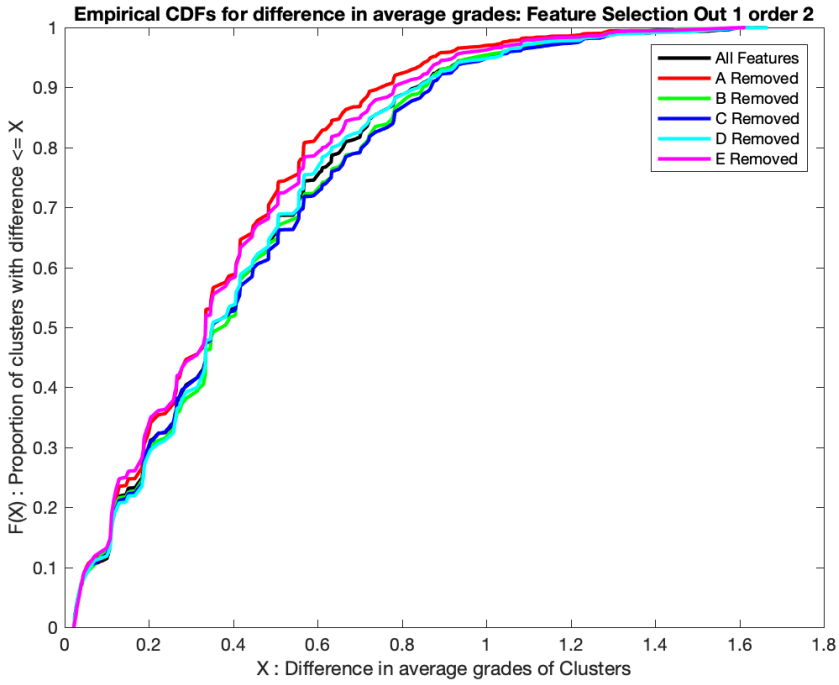


Fig. 4.4. Feature selection for pattern dependence with order 2 features (377 dimensions)

3 (Figure 4.5) features. Interestingly, we see that the effects observed for order 1 features in Figure 4.3 are not the same as those seen in higher order. For instance, in Figure 4.4, removing B (green) does not change the CDF curve much in comparison to the one with all features included (black) baseline. However, removing A (red) still shifts the CDF curve higher than the default one (black). In addition, removing E (magenta) also shifts the CDF curve higher than the default one, which was not the case with order 1. While some of the other curves are shifted from the baseline black curve, they exhibit that behaviour in a relatively smaller range of values in comparison to the red and magenta curves that span almost the whole range of values shown in Figure 4.4.

The case for order 3 features is also quite interesting. As seen in Figure 4.5, while all the colored curves seem to be shifted from the default curve (black) in small

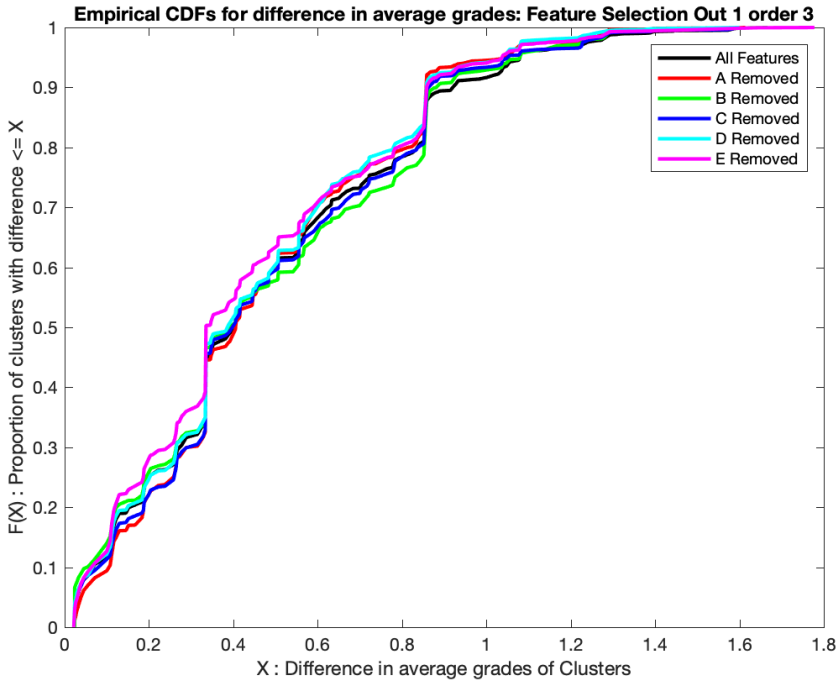


Fig. 4.5. Feature selection for pattern dependence with order 3 features (3003 dimensions)

regions, only the E removed (magenta) curve is consistently shifted higher than the default curve (black). This is quite interesting as it demonstrates an evolution of a different kind of pattern dependence effect as the dimensionality of data grows through increasing feature order. The red and green curves corresponding to A and B seem to be mostly overlapping or very close to the baseline curve for order 3 features (Figure 4.5), while they were prominently shifted much farther away for order 1 features (Figure 4.3). Based on the margin of shift away from the baseline black curve, the pattern dependence effect seems to be stronger for components A (red) and B (green) for lower order features (Figure 4.3) while this dependence seems to grow weaker as the order increases (Figures 4.4 and 4.5). The opposite appears to be the case for component E (magenta). The pattern dependence effect for component E appears weaker in lower order features and gradually becomes stronger in higher order features

as seen sequentially in Figures 4.3 - 4.5. This is indicative of a non-linear relationship between the predictors and the response for component E and linear relationships for components A and B, in the context of yielding significant differences in responses for the clusters resulting from the n-TARP process.

4.5 Conclusions

In this chapter, a statistical framework was developed to evaluate existence of a relationship between a set of predictors and a given response. Specifically, we investigated if an outcome was dependent on patterns found in the predictors, i.e. grouping the predictors into clusters. This framework also serves as an alternative approach to evaluating clusters and interpreting the grouping.

The experiments discussed have provided some important observations. The response was observed to be dependent on clusters in a statistically valid manner. The statistical validation was carried out through a hypothesis test on the distribution of a measure of the difference in response when the data was clustered with n-TARP. The extent of deviation from the null distribution appeared to increase as the dimensionality of the data grew larger through the extension of the feature space to higher order terms.

The CDF comparison approach was modified to investigate the dependence of the response on the clusters formed as well as on individual components of the data. We observed changes in the behavior of the distribution of the response when components of the data were sequentially removed. Some components' removal showed no significant change in behavior while some produced significantly shifted CDF curves, indicating a strong dependence of the response on those particular components. We also observed a change in dependence on components with changing feature order i.e. change in dimensionality of the problem. Components A and B seemed to have strong pattern dependence behavior for the response in lower dimensions but the effect seemed to weaken in higher dimensions. On the other hand, the dependence on

component E was weaker in the lower dimensions but significantly stronger in higher dimensions. These results present an interesting insight into the dependencies of the outcome on certain subsets of the data. The A and B components seem to yield a linear dependence while E yields a non-linear dependence.

In summary, we explored the statistical existence of relationships between predictors and response without enforcing a functional form. We also studied how the dependence was evolving with clusters of data and the dimensionality. Our results can potentially be used to introduce changes in certain dimensions of data prior to acquisition to cause desired effects on the response. They can also help in finding a model to represent the relationship between the predictors and response after the validation of the existence of a relationship. For instance, linear terms could be used to model the influence of components A and B of the feature vectors while cubic terms can be used for component E.

This study provides a better understanding and insight into the clusters formed through the non-deterministic n-TARP clustering framework and what the clusters represent in the context of the source data. The pattern dependence framework provides an avenue to statistically interpret the distribution of clusters obtained in a collective sense as analyzing them individually is not feasible.

5. CLUSTERING EDUCATIONAL DATA

The contents of this chapter appear in [39].

5.1 Introduction

Educators, policymakers and engineering education researchers have attempted to produce a clear understanding of the qualities and knowledge engineering graduates should possess [111]. Strong foundations in mathematics, engineering, and technology are highly emphasized in engineering programs [112]. Proficiency in areas such as good communication skills, persistence, curious learning capability, drive and motivation, and willingness to take calculated risks, among others is also important [111]. Bodies of accreditation have identified not only the required knowledge and skills that engineering graduates should exhibit, but also the attitudes and behaviors needed to confront complex problems. For instance, ABET criteria [113] stipulate student outcomes for engineering programs that consider the skills, knowledge, and behaviors that students are expected to know and be able to do by the time of graduation. Such criteria range from students' abilities to apply knowledge of mathematics, science and engineering, conduct experiments, analyze and interpret data, and design a system, component, or process to meet desired needs; to abilities to function on multidisciplinary teams, demonstrate professional and ethical responsibility, and communicate effectively, among others.

In order to identify solutions to current problems, engineering graduates must possess a professional mindset needed to shape the future, in addition to the technical knowledge and skill-set of a discipline. For example, the Engineer 2020 proposes a set of aspirations for engineering students needed to operate in societal, geopolitical, and professional contexts within which engineering and its technologies will occur [114].

These aspirations include traits such as strong analytical skills, creativity, ingenuity, professionalism, and leadership [114]. Such aspirations and traits relate to students' "Habits of Mind," which are defined in [115] as modes of thinking required for STEM students to become effective problem solvers capable of transferring such skills to new contexts. An example of Habit of Mind is a willingness to make mistakes while trying to solve a problem, an attitude that allows engineers to successfully attack complex problems.

The focus of this work are the Habits of Mind of students learning the theory and application of Signals and Systems Theory. More specifically, the work is focused on foundational concepts of digital signal processing taught to undergraduate engineering students. Two questions are investigated: 1) What are the different Habits of Mind patterns exhibited by the students?, and 2) Are some of these patterns associated with differences in course grades?

The rationale for centering the investigation around signals and systems is that these concepts are fundamental for electrical engineers and require a strong mathematical background [116, 117]. Furthermore, research has shown that the content of such courses is difficult to master [116, 118]. Previous studies of students learning signals and systems concepts have used quantitative approaches such as concept inventories [116] as well as qualitative approaches using textual analysis of students' responses [118]. Both of these approaches have advantages, but also limitations. The approach taken herein to characterize students' 'Habits of Mind' combines a qualitative method with random signal modeling and machine learning techniques. This method combines the advantages of qualitative approaches by first uncovering details of student performance from qualitative data and subsequently dividing students into groups (clusters) based on distinguishing characteristics. In contrast with global analysis methods, which reduce entire datasets into averages, percentages and other descriptive statistics, the method proposed in this chapter allows for the groups to be individually analyzed and compared so as to provide a more fine and detailed analysis of the set of students being studied. The groups are found by first trans-

forming the qualitative data into quantitative data. Specifically, the qualitative data is transformed into real-valued feature vectors by random signal modelling so it can be automatically clustered using machine learning approaches. The machine learning method selected is well-suited to analyze small datasets in high-dimensions. It also easily lends itself to statistical validation.

More specifically, the students' work is first annotated manually based on a custom-built rubric of Habits of Mind and skill levels. The annotations are vectors, which are stored in sequence for each student. The sequence of vectors of each student is modeled as a random process whose parameters are estimated from the observed data. The parameters of the random process associated with all the students are later clustered using a non-deterministic approach that yields several statistically significant patterns of Habits of Mind. These patterns correspond to binary groupings of the students, i.e. divisions of the students into two groups. The corresponding groups of students are then described using their Habits of Mind histogram as well as course grades. A more detailed statistical analysis is then given using the cumulative distribution function of the difference in average course grade of all the binary groupings. A statistical test is used to determine if the grade differences are significant. Repeating the analysis after removing certain individual Habits of Mind provides a visualization of the contribution of each Habit of Mind to the course grade.

The data analysis approach proposed is a new method that can generally be used to characterize and measure different aspects of professional formation processes in engineering education. The study itself provides a baseline for future efforts in engineering education research methods and assessment.

The rest of the chapter is organized as follows: The underlying conceptual framework is presented in Section 5.2 followed by a description of the methods used in Section 5.3. The experimental results are presented in Section 5.4 followed by discussions of the results and the conclusion of the work in Sections 5.5 and 5.6 respectively.

5.2 Conceptual Framework

This investigation is guided by the Scientific Habits of Mind conceptual framework. Habits of Mind are individuals' responses to situations and problems where the answers are not immediately known [56]. Specifically, scientific Habits of Mind refer to mathematical, logical and attitudinal modes of thinking required for science, mathematics, technology and engineering students to become effective problem solvers capable of transferring such skills to new contexts. Effective use of Habits of Mind can allow students to search for solutions moving from highly theoretical to the entirely concrete.

The implications of the conceptual framework for the design of this study relate to the operationalization and characterization of different Habits of Mind in an engineering context. The Habits of Mind explored and operationalized are described in Table 5.1.

5.3 Methods

The methodological framework for this investigation is a comparative case study method [119]. According to [120], a case study is a research strategy focused on understanding the dynamics within single settings. In [121], it is described as an empirical inquiry that investigates a contemporary issue in depth within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident. A case study approach was chosen because it facilitates in-depth investigations of student experienced Habits of Mind.

According to [121], a case study should include data collected from multiple data sources so as to allow the identification of individuals' behaviors, perceptions and attitudes. The use of multiple cases is a common strategy for identifying contextual variations [122]. By comparing cases, one can establish the range of generality of a finding or explanation, and at the same time, pin down the conditions under which that finding will occur [123]. The cases for this study were groups of students ex-

Table 5.1.
Definition of Habits of Mind as proposed by [115] and operationalization herein

Habits of Mind	Definition	Operationalization
Computation and estimation	Ability to judge an appropriate computation method to be used based on specific circumstances.	Ability to choose an appropriate computation method and carry out the mathematical procedure accurately.
Mathematical rigor	Ability for making careful observations and for handling information.	Ability to handle mathematical rigor and remember details of a definition.
Communication skills	Ability to communicate ideas and share information with fidelity and clarity.	Ability to communicate effectively, explain background and present a good and meaningful flow of ideas.
Critical-response skills	Ability to detect the symptoms of doubtful solutions, assertions and arguments.	Ability to detect the symptoms of doubtful solutions, assertions and arguments in ones own work and in peers work.
Values and attitudes	General social values and people's attitudes toward their own or others ability to understand science, engineering and mathematics.	Students attitude towards their peers work and their own ability to make assessments on others work.

hibiting Habits of Mind in similar ways. The groups were found using a clustering method called n-TARP [38, 40, 73] (previously discussed in Section 3.3) where TARP stands for "Thresholding After Random Projection"; the method is applied to feature vectors containing the parameters of a random process modeling a student's Habits of Mind expressed in an active learning activity. As described in the next section, the sources of data considered included student produced material, peer review material and course outcome data.

The clustering method looks for a good separation of the students into groups after a random projection of their representation (i.e., the feature vector containing the parameters of the student's own random process) down to one dimension [36–38, 40]. Since the structures of concern are found in a one-dimensional space, it is possible to find such groupings even if the number of points projected is fairly small. The one-

dimensionality of the data also greatly facilitates the statistical validation of these small groups [40]. These two groups were then analyzed as separate cases and their characteristics, including similarities and differences, were further explored.

5.3.1 Participants, Procedures and Dataset

The study context is a course on Signal Processing in which students were asked to produce learning material and share it on a public website [124]. Specifically, the instructor pre-defined nine topics covered in the course, and students prepared a slecture [125] explaining the course material for a topic of their choice in their own words. The term "slecture" is a concatenation of the words "student" and "lecture." Invented by Boutin in 2010, the idea is to have students create online learning material based on the teaching of a professor.

In addition to creating a slecture, the students were also instructed to review and comment on the slectures prepared by their peers (one slecture per topic for each student). Note that online discussion comments have been previously used to uncover students' Habits of Mind [126].

Specifically, the unit of analysis, the major entity that is being analyzed in a study, was each student's individual contribution to a public website. Two additional data sources were the feedback provided to their peers in the form of a review, and the final grade as a measure of performance. The cases for this study were groups of students exhibiting Habits of Mind in similar ways. Such cases were uncovered by the clustering method and were further compared and analyzed regarding their performance and observed Habits of Mind.

A total of 28 students participated: 27 students presented the slecture in written form, while one presented it as a video. The 27 written slectures were used in this study. There were 3.0 slectures per topic and 6.89 reviews per slecture, on average. This is because some students did not complete the review assignment while others provided more/less than 9 reviews. All students who completed the tasks received

full credit on the assignment, so the exercise in itself did not produce any difference in grades among the participants.

5.3.2 Data Scoring

The data scoring was performed using a rubric. The rubric was created and validated iteratively, starting with an inductive approach, followed by a deductive approach. For the inductive approach, one of the researchers with expertise in education research built a Habits of Mind focused criteria (see Table 5.1), which was derived from the literature [115]. The initial definitions were then further operationalized for the context of the study. Based on the initial operationalization of each construct or criterion, levels of performance were identified (see Table 5.2). A second researcher with expertise in signals and systems then used the rubric to annotate the slectures and the reviews. The first pass of the data scoring was then validated and reviewed by a third author. In the process, the rubric was modified to better capture students' patterns, and when modified, it was tested against the data following a deductive approach. The process and findings were discussed among the three researchers. This iterative approach was performed three times resulting in the rubric presented in Table 5.2 (reported in [55]).

The element "Values and Attitude" was initially focused on perceived importance or confidence in the subject domain. Traditionally, this habit of mind is assessed via surveys asking students to report their perceived confidence on the subject matter or their self-perceived abilities to understand the concepts. Because an opportunity to survey students was not available, "Values and Attitudes" focusing on students' abilities to evaluate their own and their peers' work was indirectly characterized. That is, it was found that the critical views of their own work and that of their peers was a proper indicator of students' confidence and abilities in their own knowledge and skills. So the focus was shifted to analyzing if students were able to provide

Table 5.2.

Rubric generated from student exhibited Habits of Mind.

	Description		Performance Level			
Tag	Element	Definition	Below Basic 1	Basic 2	Proficient 3	Advanced 4
A	Computation and Estimation	Ability to choose an appropriate computation method and carry out the mathematical procedure accurately.	Student selected an incorrect method and the solution was completely off.	Student selected a correct method but the solution was incorrect.	Student selected an appropriate method and the solution was correct. However, the student did not provide a justification for the method based on the circumstances, or the justification was inadequate.	Student selected an appropriate method, provided correct justification for the method selection based on the circumstances and the solution was correct.
B	Mathematical Rigor	Ability to handle mathematical rigor and remember details of a definition	Student was not at all rigorous in the involved mathematics.	Student displayed some rigor but there were major errors.	Student was very rigorous but made small errors.	Student was very rigorous and made no errors.
C	Communication Skills	Ability to communicate effectively, explain background and present a good and meaningful flow of ideas	Student presented an unclear and unjustified procedure.	Student presented a somewhat clear procedure but it was unjustified.	Student presented a clear procedure with a reasonable justification.	Student presented a clear procedure with a detailed justification based on the theory or principles.
D	Critical Response Skills	Ability to detect the symptoms of doubtful solutions, assertions and arguments in one's own work and in peers' work	Student was unable to identify incorrect procedures and provided no evidence of procedures for validation of their solution.	Student was able to identify incorrect procedures but was unable to correct them. Student provided no evidence of procedures for validation of their solution.	Student was able to identify incorrect procedures and corrected them properly. However, student provided no evidence of procedures for validation of their solution.	Student was able to identify incorrect procedures and correct them properly or did not demonstrate any incorrect procedures. In addition, student demonstrated evidence of applying procedures for validation of their solution.
E	Values and Attitudes	Students attitude towards their peers' work and their own ability to make assessments on others work	Student made negative comments about others' work or was indifferent to it.	Student made generic comments that do not provide any insight or critique.	Student made good comments providing insight and a somewhat reasonable critique.	Student made excellent comments, correcting mistakes and providing insightful critique.

a meaningful critique of their peers' work and how their attitude appeared in their feedback. Below are two examples of Values and Attitudes ratings.

- "I think specific outline is very helpful and make easy to follow the formula and graphs. Formulas and graphs are very clear to understand." − Basic Level rating

- "I think an important aspect that you did not include in your final answer is that the DTFT of a DT signal must be periodic. Your answer must be"rep-ed" to denote its periodicity. Otherwise your answer is only correct for $0 \leq \omega \leq 2\pi$. The DTFT of $x[n]$ is $rep_{2\pi}(2\pi\delta(\omega - \omega_0))$ Overall color coating was very helpful, and the slecture was concise and clear" − Advanced Level rating

Another example is the element "Computation and Estimation", which initially focused on the ability to choose an appropriate computation method and recognize when approximations can be made. The scenarios involving approximations were not present in the topics, but rather, the problems involved mathematical computations. Hence, the focus for this element was shifted to appropriateness of the computational method used and mathematical accuracy of the computation. A final example is the element "Mathematical Rigor", which in an earlier version of the rubric was called "Manipulation and Observation". Mathematical rigor in this case refers to the correctness of the notation and attention to details in the writing of mathematical expressions. This element was mostly present in definitions and mathematical statements either within a computation or standing on its own within the text of a slecture. (See the top of Figure 5.1, tagged as (B1,D1), for an example of poor rigor). The change was motivated by the fact that mathematical rigor within arguments and explanations was found to play a more critical role than handling basic mathematical manipulation and observation. In fact, manipulation and observation can be bundled in with computation and estimation. The final rubric is presented in Table 5.2. Please note the alphabetical labeling of the items, which is used in the annotation process described below.

5.3.3 Annotating Slectures and Comments with Rubric Tags

The annotated material of each student was recorded as a sequence of vectors representing the sequence of Habits of Mind elements and levels of performance. For example, one part of a slecture might have been tagged with the vector $(A4, B4, C2, D4)$ to denote that the student carried out the computations effectively with the necessary rigor and validation but the explanation was lacking in terms of communication. The length of the text used for each labelled block varied with the context and student communication style, so as to include separate ideas and concepts. The lengths varied from one sentence or equation for concise ideas to several paragraphs for lengthy or redundant explanations and were decided subjectively by the rater on a case to case basis.

Two examples are provided to illustrate how the slecture material is tagged / annotated. The first one (Figure 5.1) shows a low value tag because of errors made in the slecture. The second one (Figure 5.2) shows a high value tag since the material was almost flawless.

Table 5.3 presents a comparison of two examples of the identified Habits of Mind. Two examples from each are presented for comparison between different levels of performance. Example 1 for "Computation and Estimation" shows a computation error whereas for "Mathematical Rigor" and "Critical-response Skills", Example 1 shows wrong mathematical statements. For "Communication Skills", Example 1 provides a basic explanation of an idea through a set of mathematical equations without much context or explanation. For "Values and Attitudes", Example 1 shows a very vague comment that does not really provide any insight or critique. Reasons such as these justify a lower score for the scenarios presented under the Example 1 column of Table 5.3. In contrast, the corresponding scenarios in Example 2 are mostly correct and error free in terms of mathematics and provide greater insight, critique and context in terms of "Communication Skills" and "Values and Attitudes", which justifies a higher score.

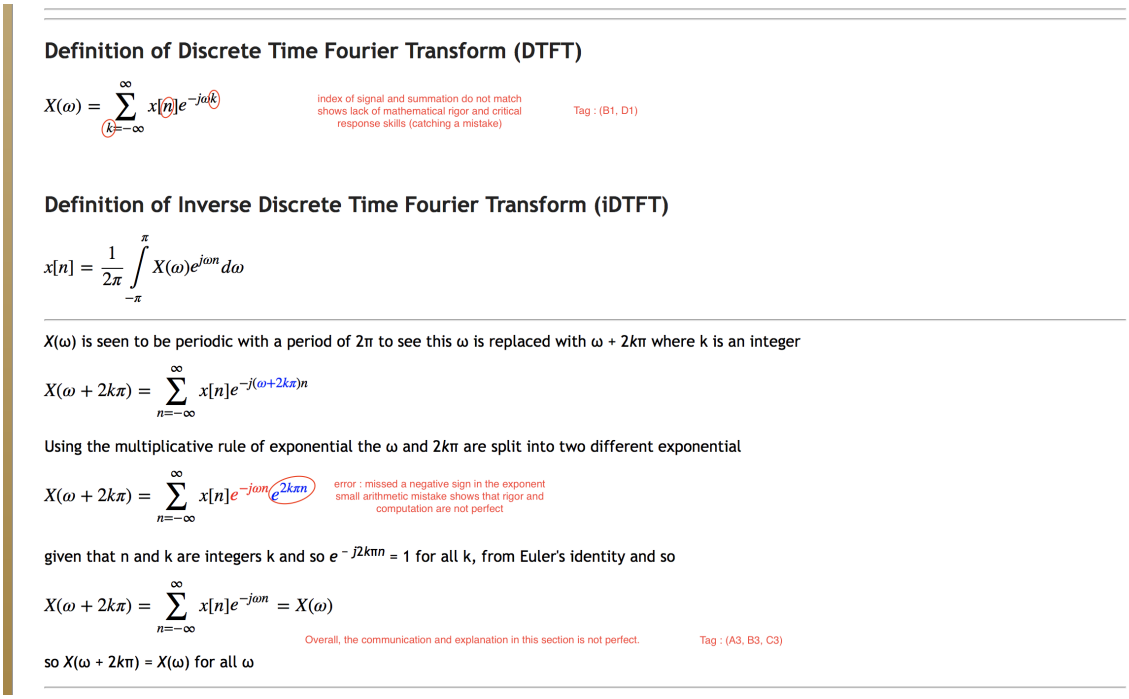


Fig. 5.1. Low value tag in a slecture

5.3.4 Inter-rater Reliability

The reliability of the data scoring was estimated by having another author annotate the slectures and reviews of 11 students in two phases. Data from 11 students was chosen at random to avoid bias and to span at least a third of the data. First, the rubric was consulted and discussed with the first rater. Then the slectures and reviews of five students were rated by the second rater. A discussion followed in which the two raters compared their scoring and discussed the reasons behind the differences. The second rater then rated the slectures and reviews of six other students. The reliability of the first (5 students) and second phase (6 students) were measured using two correlation coefficients (Pearson product-moment coefficients [127,128]) for each phase. One coefficient represents the reliability of the detection of the different rubric elements present in the work; the other coefficient represents the reliability of the accuracy of the scores for all the rubric elements.

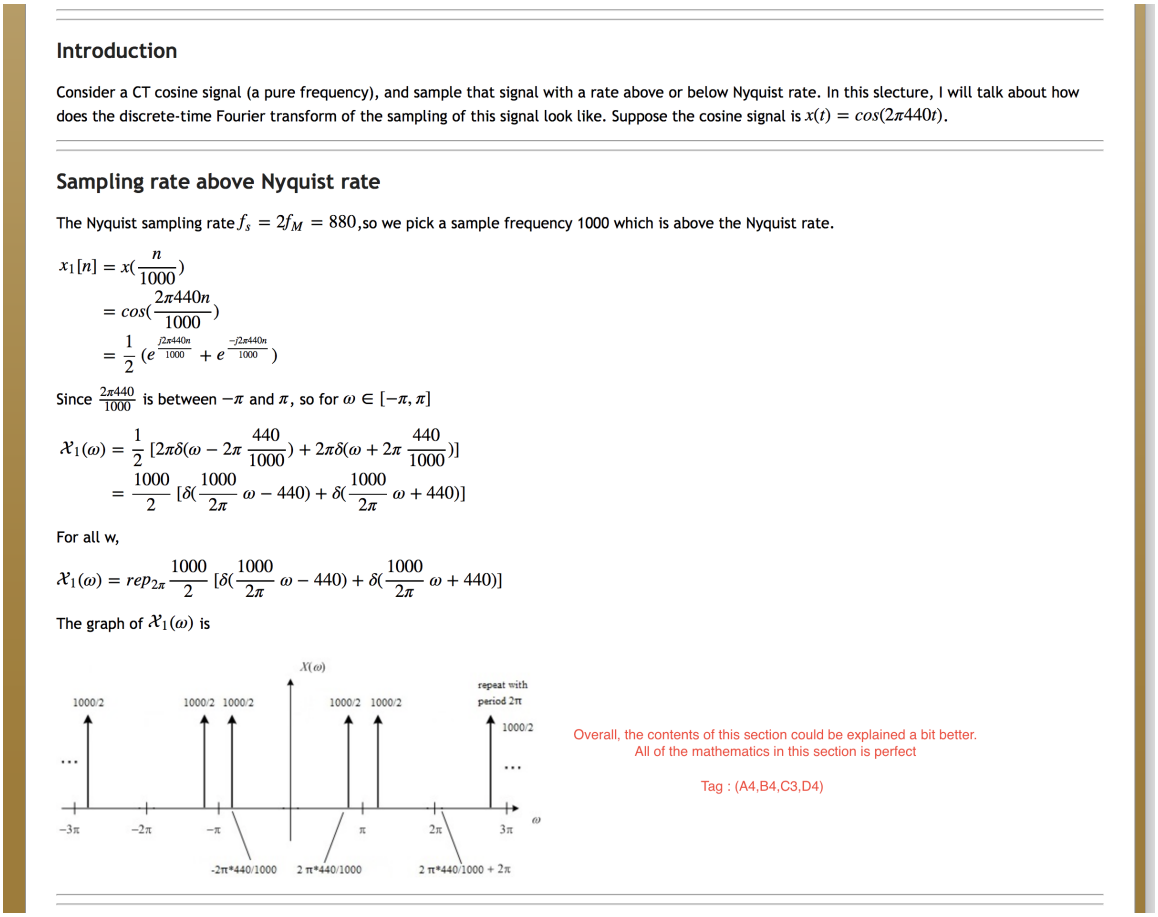


Fig. 5.2. High value tag in a slecture

More specifically, the reliability of the detection of the initial rater was estimated using the letter tags (without scale value) for both raters: when a part of text was coded with a given letter by a rater, a "detection", denoted by a "1", was recorded for that rater; if the other rater also coded the same text with that letter, then the other rater was considered to have also detected that event, and a "1" was recorded for that rater as well. Conversely, if the other rater did not code that text with that letter, then this rater was considered to have missed that event and a "0" was recorded. The reliability of this detection process was measured using the correlation coefficient [103] of the sequences of 0's and 1's for the two raters. The reliability of the accuracy of the labeling (letter and score) was only considered for those events

Table 5.3.
Examples of Habits of Mind enacted by Students

Habits of Mind	Example 1 (Low level)	Example 2 (High level)
Computation and Estimation	"$X(f) * f_s \sum_{k=-\infty}^{\infty} \delta(f - kf_s) = f_s \sum_{k=-\infty}^{\infty} X(f) * \delta(f - \frac{k}{f_s})$" −Basic Level	"$X_s(f) = F(\sum_{n=-\infty}^{\infty} x(nT)\delta(t - nT)) = \sum_{n=-\infty}^{\infty} x(nT)F(\delta(t - nT))) = \sum_{n=-\infty}^{\infty} x(nT)e^{-j2\pi fnT}$" −Advanced Level
Mathematical Rigor	"$X(2\pi f) = X(f)$" −Below Basic Level	"$\cdots X(f) = F(\delta(t - t_0)) = \int_{-\infty}^{\infty} \delta(t - t_0)e^{-i2\pi ft}dt = e^{-i2\pi ft_0} = e^{-i\omega t_0}$" −Advanced Level
Communication Skills	"Comb operator is used in time domain: $comb_T[x(t)] = \cdots = x(t) \cdot P_T(t) \cdots$" −Basic Level	"$\cdots x_s(t)$ is created by multiplying a impulse train $P_T(t)$ with the original signal $x(t)$ and actually $x_s(t)$ is $comb_T(x(t))$ where T is the sampling period $\cdots$" −Advanced Level
Critical-response Skills	"$x(t) = \int_{-\infty}^{\infty} \delta(t - \tau)d\tau$" −Below Basic Level	"$\cdots$the minimum repeating period T has to be $> a + b$ (a is the left boundary of the curve and b is the right boundary of the curve)." −Proficient Level
Values and Attitudes	"I think specific outline is very helpful and make easy to follow the formula and graphs. Formulas and graphs are very clear to understand." −Basic Level	"I think an important aspect that you did not include in your final answer is that the DTFT of a DT signal must be periodic. Your answer must be"rep-ed" to denote its periodicity. Otherwise your answer is only correct for $0 \leq \omega \leq 2\pi$. The DTFT of $x[n]$ is $rep_{2\pi}(2\pi\delta(\omega - \omega_0))$ Overall color coating was very helpful, and the slecture was concise and clear" −Advanced Level

(rubric elements) detected by both raters. A sequence of scores for each rater was built by concatenating all the numerical scores for all the commonly detected events of a given type (tag) into a vector; the correlation coefficient of the two vectors for that tag was then computed.

The reliability of the detection of the rubric elements in the first phase of the inter-rater reliability testing was found to be 0.6163 (correlation coefficient). In the second phase, that number increased to a much higher value of 0.8166. The reliability of the accuracy of the scoring was found to be 0.9430, already a very high value, which increased modestly to 0.9574 in the second phase. Thus, both the detection of rubric elements and accuracy of the data scoring were considered to be very reliable. These reliability estimates were computed after the two phases of rating were completed.

5.3.5 Data Analysis

Class Statistics

The Habits of Mind of the class are first summarized using a 2D histogram of tag values (in a 5×4 grid) in order to analyze the distribution of the annotation tags for the entire set of students in the study and look for some global trends in the class as a whole with regard to their Habits of Mind. The final grade distribution for the entire course is also examined in order to characterize their academic performance in the course as a whole.

Statistical Model Building

For a more in-depth analysis, a statistical model that describes each student's individual Habits of Mind is built; this model will later be used to cluster the students. The statistical model represents a random process underlying the sequence of annotation tag vectors. The parameters of the statistical model are estimated from the annotated data. For simplicity, consecutive vector tags are assumed to be independent. The different elements (A,B,C,D,E) are also assumed to be independent. However, in other circumstances, another perhaps more complicated model could be more appropriate. For example if the work was carried out over a long period of

time over which an improvement was expected, the consecutive vector tags could be modeled by a time-dependent process.

The statistical model consists of the likelihood of the tag scores for each element in the proposed rubric. In other words, it is represented by the discrete probabilities [103]: $P(k)$ and $P(j|k)$, for $k \in \{A, B, C, D, E\}$ and $j = 1, 2, 3, 4$. These probabilities are estimated by the relative frequencies of each tag in the scored data as follows.

$$\bar{P}(A) = \frac{\sum_{i=1}^{N_s} \mathbb{I}\{\textit{student } s \textit{ gets tag } A \textit{ in annotation } i\}}{N_s},$$
$$\bar{P}(1|A) = \frac{\sum_{i=1}^{N_s} \mathbb{I}\{\textit{student } s \textit{ gets tag } A(1) \textit{ in annotation } i\}}{\sum_{i=1}^{N_s} \mathbb{I}\{\textit{student } s \textit{ gets tag } A \textit{ in annotation } i\}}$$

where N_s is the number of annotations recorded for student s. A similar expression is used for the other score values 2, 3, and 4, for rubric element A. The probabilities for the other rubric elements B, C, D and E are computed in a similar fashion, except that the parameters N_s takes the value 26 for E (since the students could review a maximum of 26 slectures.)

Thus 5 model parameters for each of the 5 elements of the rubric are estimated, for a total of 25 parameters for each student. In addition to these, the parameter N_s (number of annotations received by the student) is added in order to highlight the difference between short and long slectures. Thus, 26 parameters are used to represent each student; these are stacked into a vector of dimension 26.

Clustering

Clustering a small number of points (27) in a high-dimensional space (26) is challenging and requires the use of an algorithm that is specially designed for small data. One such algorithm is "n-TARP" [40, 73], an algorithm that seeks good separations of the data after a projection onto a random line. The separation is obtained by projecting and thresholding the data n times, and picking the projection with the best separation among those n. It is a modification of the random projection

approach developed in [36–38], which has been empirically shown to work well for "real" high-dimensional data in general. The name TARP stands for "Thresholding After Random Projection." Instead of hierarchically clustering the data using a tree of thresholds after random projections (n-TARPs) as in [37], the method performs a single n-TARP on a fraction of the data given, and tests the statistical validity of any clustering identified using the remaining fraction of the data [40], as discussed in Chapter 3.

In general, clustering methods can be viewed as maps from the feature space (in high-dimensions for the data at hand) to one-dimensional space $\mathbb{R}$, followed by some thresholdings. Different methods have different ways of defining the "best" projection and thresholds. Projecting the data onto a line and thresholding corresponds to finding a linear separation between the clusters, which is the simplest form of clustering. Linear separations are well-suited for small data in high-dimensions because they can be found when only a small number of points are given. Previous work in [36–38, 40] has shown that many good linear separations can be found in real data by picking the line of projection at random. This is because real data often has a lot of hidden structure in high-dimensions that can be extracted through random projections [36, 37]. This observation, combined with the fact that only a small number of students are considered in this study, are the motivations for employing n-TARP to cluster the data.

In the experiments at hand, a binary clustering was performed using n-TARP with the parameter n set to 500 in order to divide the set of students into two groups, picking the best separation among the 500 projections performed. This was done in two phases: a training and a validity testing phase. Half of the data (randomly chosen every time) was used for each phase. Because the data size is so small (27 points), one would hardly expect to find any meaningful cluster in the original space. Looking for clusterings in a one-dimensional space addresses this issue because the projected points are closer together than in the original space. The extent to which the (training) projected points are clustered is measured using "normalized withinss

(W)", a renormalized version of the within class scatter of the data [36, 129]. More specifically the within class scatter of [129] is divided by the number of points and the empirical variance of the projected data. This insures that the measure is independent of the number of points considered and invariant under a rescaling of the dataset [36]. The definition of "normalized withinss (W)" for a set of projected points $x_1, x_2, \ldots, x_m \in \mathbb{R}$ is given below [37]

$$W = W(x_1, \cdots, x_m) = \min_{C_1, C_2} \frac{\sum_{i \in C_1} (x_i - \mu_1)^2 + \sum_{i \in C_2} (x_i - \mu_2)^2}{\tilde{\sigma}^2 \cdot m},$$

where C_1 and C_2 are a disjoint partition of the set of indices $\{1, \ldots, m\}$, μ_1 and μ_2 are the (empirical) mean of the points x_i whose indices are in C_1 and C_2, respectively, and $\tilde{\sigma}$ is the (empirical) standard deviation of the set of points $x_1, \cdots, x_m \in \mathbb{R}$.

Training Phase:

1. For $i = 1$ to n:
2. Generate a random vector r_i in 26 dimensional space;
3. Project the training data onto this vector r_i to form 1D projection values;
4. Use k-means (set $k = 2$) to find 2 clusters in the 1D projection values;
5. Find the normalized withinss w_i for this cluster assignment;
6. End loop.
7. Pick lowest w_i among the n measurements and store the random vector r^* associated with it and determine a threshold t^* that separates the classes formed in the 1D projected space.

Validity Testing Phase

1. Import r^* and t^* from the training phase

2. Project the testing data onto the vector r^*

3. Use the threshold t^* to assign clusters to each of the testing samples

4. Perform permutation test with Monte-Carlo simulations [50] on the projected test data at statistical significance level of 99%.

Pattern (Binary Clustering) analysis

Recall that the clustering is random, and thus can yield several different (and valid) binary clusterings. Each of these clusterings splits the students into two groups based on some distinctive Habits of Mind patterns. Although the pattern is described by the coefficients of the random projection vector r^* used for the projection, it is typically hard to make sense of the pattern directly from these coefficients. As an alternative, the histogram of Habits of Mind annotations for the two groups are considered and compared (i.e., the frequency of occurrence of each rubric annotation for both clusters). The distribution of the course grades for the two groups are also compared.

The number of different Habits of Mind patterns exhibited by students is quantified following the approach of [36, 37, 40]. Specifically, the distribution of the normalized withinss of the (random) projected data is plotted, and the area of the distribution below the value ≤ 0.36 (threshold value after which no clusters exist) is computed.

To quantify the relationship between Habits of Mind patterns and course grade, the empirical Cumulative Distribution Functions (CDF) [103] of the absolute difference between the average grades of both groups is constructed. In order to check the dependence of the different elements of the rubric and course grades, each element is removed one by one and a new CDF of absolute difference between average grades between groups is obtained: the resulting CDF curves are then compared.

Hypothesis Testing

It is conceivable that randomly grouping the students into two clusters could result in different grade distributions for the two clusters due to chance and not in any way related to the Habits of Mind of the students. In order to test the statistical significance of the observations, independence of the grades on the patterns (groupings) of Habits of Mind is set as the null hypothesis, and statistical significance of the observations is tested by comparing the CDF curves of the previously obtained clusterings with the CDF curves for random clusterings. In other words, the CDF of grade differences for the binary clusterings previously obtained is compared with the CDF of grade differences that one would obtain with random division of the students into two groups, as discussed in Section 4.3.

5.4 Results

In this section, the results of the data analysis are presented. After a brief comment on slectures, summary statistics like frequency of occurrence of the different levels of the elements of the rubric are presented. Following that, the results of the n-TARP clustering algorithm, which uses the feature vector formed through the model fitting described in the previous section, is presented. The frequency of occurrence of the rubric tags for the resulting groups are compared to identify the differences that led to the formation of the groups. Next, the results on the extent of clusterability of the data are presented. The various different clusters formed as a result of the random projection model underlying the n-TARP clustering algorithm are presented. Finally, the connections between 'Habits of Mind' patterns and the grades of students are examined.

5.4.1 Overall Patterns of Students' Habits of Mind

Table 5.4 shows the relative number of times each element/level of the rubric was tagged in the study. The most frequent tag is Values at a basic level (32.2%), followed by Values at a Proficient level (11.2%) and Computation at an Advanced level (7.7%). No Below Basic level was found with a frequency above 2%, and the only Habits of Mind element noted more than 3% of the time is Values. Overall, the elements other than Values tend to be tagged more frequently at the Proficient or Advanced level. Overall, a majority (64.8%) of the tags were given at the Proficient or advanced level.

Table 5.4.
Percentages of Exhibited Habits of Mind Among All 27 Students

Element/Level	Below Basic	Basic	Proficient	Advanced
Computation	0	1.06	1.32	7.71
Rigor	1.59	2.65	5.85	3.98
Communication	0.26	2.92	6.11	3.98
Critical Response	1.32	1.06	3.98	5.58
Values	1.06	32.18	11.17	6.11

The clustering method was repeated more than 1000 times to form groupings; one such (statistically significant) grouping, which was found to have a significant effect on the grades, is analyzed in Tables 5.5 and 5.6. Observe that students in Cluster 2 have much larger numbers of high level tags for all the elements of the rubric than Cluster 1, indicative of a higher level of Habits of Mind performance. Indeed, a majority (64.8%) of annotation tags for Cluster 2 are at the "Proficient" or "Advanced" level. In contrast, a majority (63.3%) of annotation tags for Cluster 1 are at the "Below Basic" and "Basic" level. Thus, the members of Cluster 2 are identified as the "Habits Developed" students (Case 2), and the members of Cluster 1 as "Habits Developing" students (Case 1).

5.4.2 Case Comparison

As stated earlier, two cases were identified. Case 1 is called the "Habits Developed" group, and Case 2 is called the "Habits Developing" group. The groups were characterized on the basis of the overall distribution of levels (more Advanced level tags for Case 1 than for Case 2). As observed from the sums of the columns of Tables 5.5 and 5.6 for each row element (Habit of Mind), the "Habits Developing" group (Cluster 1, Table 5.5) is also distinguished by a higher probability of expressing the "Values" element, 55% vs 48% for the "Habits Developed" group (Cluster 2, Table 5.6). Further, the "Habits Developing" group also shows a slightly lower probability of expressing the "Communication" element, 11% vs 14% for the "Habits Developed" group. On the other hand, the likelihood of exhibiting the "Computation" (9% versus 11%), "Rigor" (14% versus 14%) and "Critical Response" (11% versus 12%) elements are somewhat similar for both groups (Cluster 1, Table 5.5 vs Cluster 2, Table 5.6).

Table 5.5.
Percentages of Exhibited Habits of Mind for Case 1: Habits Developing (10 students)

Element/Level	Below Basic	Basic	Proficient	Advanced
Computation	0	1.66	3.33	4.16
Rigor	3.33	4.16	5.00	1.66
Communication	0	5.00	4.16	1.66
Critical Response	2.50	1.66	3.33	3.33
Values	3.33	41.66	9.16	0.83

5.4.3 Overall Course Performance and Performance by Case

The grade distributions for the clusters and the entire class are shown in Table 5.7. Observe the grade differences between the clusters (e.g., difference of 1.46 between

Table 5.6.
Percentages of Exhibited Habits of Mind for Case 2: Habits Developed (17 students)

Element/Level	Below Basic	Basic	Proficient	Advanced
Computation	0	0.78	0.39	9.37
Rigor	0.78	1.95	6.25	5.07
Communication	0.39	1.95	7.03	5.07
Critical Response	0.78	0.78	4.29	6.64
Values	0	27.73	12.10	8.59

average grades), indicating that having well developed Habits of Mind is associated with good course performance. Indeed, none of the Habits Developing students received an A in the course, whereas none of the Habits Developed students received an F or a D in the course, a majority receiving As or Bs.

Table 5.7.
Grade Distributions specific to clusters compared to each other and the distribution for all students together

Grade	All Students	Case 1: "Habits Developing"	Case 2: "Habits Developed"
A (4.0)	5	0	5
B (3.0)	10	2	8
C (2.0)	8	4	4
D (1.0)	2	2	0
F (0.0)	2	2	0
Mean Grade	2.51	1.60	3.05
Standard Deviation	1.12	1.07	0.74

However, there may be other patterns of Habits of Mind whose association to the course grade could be different. Figure 5.3 shows the distribution of normalized within-inss W for the dataset, which shows the very high clusterability of the dataset [36,37], as approximately 80% of the clusters found have a value of $W \leq 0.36$ (cluster present). The connection between these patterns and the grade is shown to be very strong in Figure 5.4. Specifically, the graph shows the CDF of the (absolute) difference in average grade between the two groups for a total of 1000 attempted binary groupings of which only valid statistically significant groupings are retained (about 90%). The lower the curve at a given point (grade value), the higher the proportion of patterns with an average grade difference at least as large as that grade value. For example, about 30% of the Habits of Mind patterns found were associated with an average grade difference of at least 0.5 (since the y-axis value for a difference in grades of 0.5 is about 0.7). The x-axis intercept is about 0.02 and thus no groups yield an average grade difference less than 0.02 (0.5%).

The elements of the rubric were removed one at a time: each time similar to above, a new set of 1000 clusterings was obtained of which only valid statistically significant groups are retained, and the CDF of the absolute value of the average grade difference between the groups was computed. The resulting curves are also shown in Figure 5.4. Observe that removing Element A shifts the CDF curve up (i.e. the new CDF curve is above the original CDF curve), and thus the relationship between Habits of Mind patterns not involving Computation are less strongly associated with different grade outcomes than Habits of Mind patterns involving Computation. This implies that Computation is related (dependent) to the final grade. The same is true, though to a lesser extent (less grade difference), with Values and Attitudes.

Hypothesis Testing

The CDF of grade differences for the binary clusterings is compared with the CDF of grade differences that one would obtain with random division of the students into

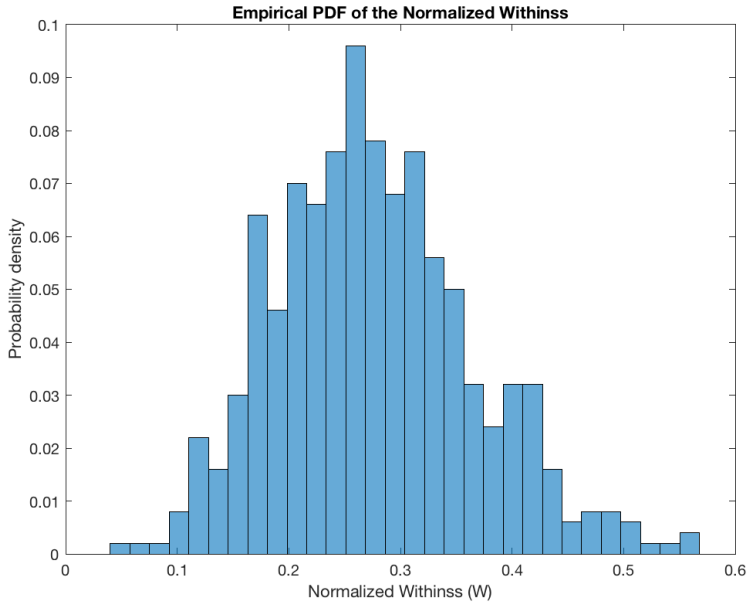


Fig. 5.3. Empirical probability distribution function of the normalized withinss W. The Clusterability of the data is measured by the pdf of Withinss.

two groups in Figure 5.4. The figure shows three CDF curves added to the plot of Figure 5.4, identified in the legend as mean and ± 5sigma.

To obtain these five curves, the students were randomly grouped into 2 clusters 10000 times to get 10000 differences in average grades of the resulting random clusters. Note that the Habits of Mind features were not utilized at any point in this process, just random divisions of the students into two groups. These 10000 differences were used to generate a CDF curve based on the null hypothesis. This process was repeated 100 times in order to get 100 CDF curves, which were used to form the mean null hypothesis curve (in solid black in Figure 5.4) along with the null hypothesis curves shifted five standard deviations away (in dashed black in Figure 5.4).

The significance level estimated by the pair of curves for mean ± 5 sigma corresponds to a significance level of at least 96% based on Chebyshev's inequality

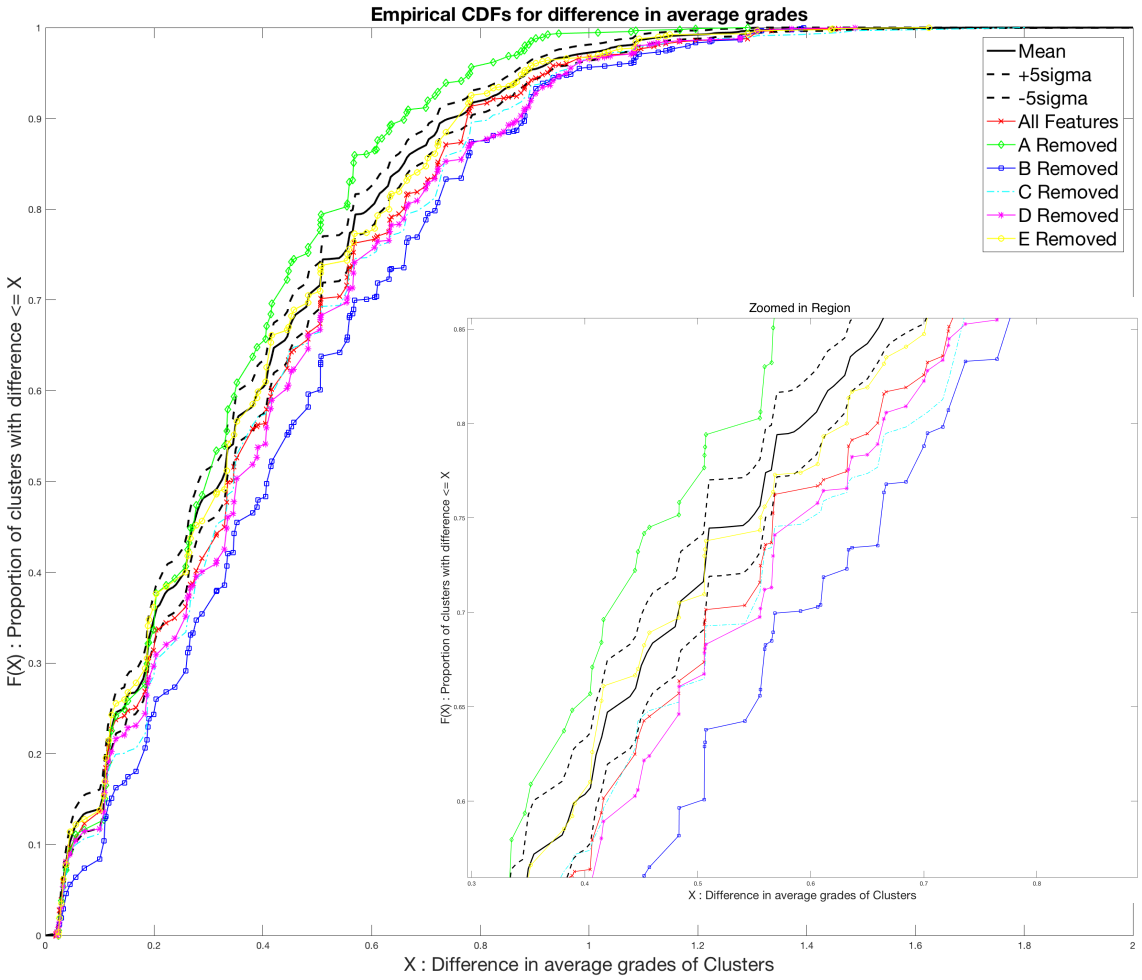


Fig. 5.4. Cumulative distribution functions (CDF) for absolute value of difference between average grades.

[109, 110]. The inequality states that for a random variable X with finite mean μ and finite non-zero variance σ^2 and any real number $k > 0$,

$$Pr(|X - \mu| \geq k\sigma) \leq \frac{1}{k^2}$$

which for $k = 5$ means that there is at most 4% chance of a sample lying more than 5 standard deviations away from the mean. This inequality does not make any assumptions on the underlying distribution of the random variable X and so is

a conservative general bound, thereby guaranteeing a minimum significance level of 96% if a realization lies outside the $\pm\ 5\sigma$ boundary of the mean.

As discussed previously in Section 4.3, in order for the null-hypothesis to be rejected (i.e., to say that the grade differences observed are dependent on patterns of Habits of Mind), the CDF curve obtained with a certain set of Habits of Mind should lie above/below the $\pm\ 5\sigma$ curves at a given point. More specifically, if the CDF curve for a grade difference of at least X based on patterns of Habits of Mind is, say, above the $+5\sigma$ curve or below the -5σ curve at X, then the probability that the observed grade difference of at least X for the proportion of Habits of Mind patterns indicated by the value of the CDF curve at X is due to chance is below 4%.

Recall that removing element A (green curve, Figure 5.4) not only shifted the Habits of Mind curve up, it also shifted it higher above the null-hypothesis curve (black curves). So in a statistically significant manner, removing element A reduces the association of the grades with the clusters. In other words, removing the "Computation and Estimation" Habit of Mind from the analysis results decreases the association of the grades with the patterns of the Habits of Mind. On the flip side, one see that removing element E (yellow curve) results in the CDF curve being pushed up and significantly overlapping with the null-hypothesis curve (black curves). This implies that the grade cluster associations from this experiment are not statistically significant. Hence, removing the element "Values and Attitudes" results in patterns of Habits of Mind that are not associated with a statistically valid grade difference. Thus, this element is a pivotal component of the patterns formed by the Habits of Mind associated with a significant grade difference since its removal results in statistically invalid patterns. Finally, one observes that curves corresponding to retaining all Habits of Mind (red curve), removing element B (blue curve), removing element D (magenta curve) and removing element C (cyan curve) one at a time result in curves that are below the null hypothesis curves (black curves) for a large range of grade difference values, indicating that the grade cluster associations displayed through these experiments are indeed statistically significant. Therefore, groups formed by either

including all Habits of Mind, or all Habits of Mind except "Mathematical Rigor", or all Habits of Mind except "Communication Skills" or all Habits of Mind except "Critical Response Skills" yield patterns that are associated with significant differences in grades in a statistically valid manner.

5.5 Discussion and Implications for Research, Teaching and Learning

Results from this study suggest that the course grade was dependent on at least two Habits of Mind: (a) Computation and Estimation and (b) Values and Attitudes. The dependency of course grade on computation and estimation is consistent with previous work that suggest that students' ability to choose an appropriate computation method and accurately carry out a mathematical procedure is a critical skill in engineering professionals [111]. Similarly, as reported in previous work on student learning of signals and systems, strong mathematical knowledge is important to succeed in this course [116,117]. A second dependency of course grade was on values and attitudes. In this study values and attitudes were operationalized as students' reactions and insights about others' work; that is, it was operationalized as peer-feedback. Student peer-feedback has been identified as a required skill to function properly in industry as well as educational settings [130]. It has also been identified as a critical form of effective communication skills, problem-solving skills, and professional responsibility to conduct the feedback. Although peer feedback has been widely implemented in engineering education as part of team performance [131], researchers have identified it as difficult to implement when the goal is improving students answers to open-ended problems [130,132,133]. However, when successfully integrated, peer feedback can result in better course performance and higher level thinking skill display such as critical thinking, planning, monitoring, and regulation [134].

Implications for research relate to the use of clustering methods to supplement traditional approaches for data analysis in education research. For instance, if only traditional approaches for qualitative analysis were followed for this study, the inves-

tigators would have been limited to characterizing the Habits of Mind as identified in Table 5.2. Specifically, following a traditional qualitative approach would have allowed to understand and describe how students' Habits of Mind were enacted by students in the context of a signals and systems course. Taking a step further by then utilizing quantitative approaches to data analysis, allowed the researchers to identify overall patterns of students' performance as depicted in Table 5.4. By utilizing the clustering method the investigators were able to identify several binary groupings (i.e. divisions of the students into two groups) that were found to be statistically significant. One particular grouping was highlighted. The patterns corresponding to the two groups (Habits Developed and Habits Developing students) were compared and contrasted based on their similarities and differences, both in terms of Habits of Mind elements and levels exhibited (Tables 5.5 and 5.6) and course performance (Table 5.7). The final step tested whether the grade differences observed for all the different patterns (clusterings) of Habits of Mind were statistically significant (Figure 5.4).

The implications for teaching and learning relate to the integration of pedagogies that not only focus on emphasizing the technical or mathematical elements of a course, but also those that integrate critical peer-feedback. The use of slectures appears to foster students' application of signals and systems knowledge along with other skills. That is, having students explaining the course material for a topic of their choice in their own words, as well as reviewing and commenting on the slectures prepared by their peers, may be an appropriate approach to help students develop Habits of Mind [125].

5.6 Conclusions, Limitations and Future Work

This chapter looked at how engineering students exhibited 'Habits of Mind' in the context of student-generated content for a course on signal processing. The five Habits of Mind investigated were Computation and Estimation, Mathematical Rigor,

Communication Skills, Critical Response Skills, and Values and Attitudes. A quantitative analysis based on random signal modeling and clustering was performed. The model assumed independence of the vector tags used to annotate the student slectures, which is a simplifying assumption for a first order model. A more complex model that relaxes this assumption and potentially models the data better, requires a larger number of data samples than were available.

Students were found to exhibit various different patterns of Habits of Mind (binary groupings). One such pattern (grouping) that was found to affect grade was analyzed: the main difference between these particular groups was found to be the level of proficiency of all the Habits of Mind elements. Thus the groups were designated as "Habits Developed" and "Habits Developing", respectively. Further analysis of the entire set of patterns (groupings) found by clustering revealed that many patterns of Habits of Mind affect grades, and that the grade is directly dependent on Computation and Estimation and Values and Attitudes. The main limitation of the study is the dependency of the proposed method on qualitative approaches to hand-scoring the data. The small sample size allows iterative scoring of the data by hand, and validation of such scoring by multiple raters. This step of the method will be harder to replicate with larger samples. While this study is limited in scope and size, it will be interesting to see if these results are confirmed in other electrical engineering core courses. It would also be interesting to conduct a comparative study between students who did slectures and the ones who did not. The analysis framework proposed is applicable in many other contexts and data types (e.g., video data or think-alouds) and could be used to study the relationship between other skills and educational outcomes.

REFERENCES

REFERENCES

[1] E. M. Wright and R. Bellman, *Adaptive Control Processes: A Guided Tour*. Princeton university press, may 1962, vol. 46, no. 356. [Online]. Available: https://www.jstor.org/stable/3611672?origin=crossref

[2] D. L. Donoho and Others, "High-dimensional data analysis: The curses and blessings of dimensionality," *AMS math challenges lecture*, vol. 1, no. 2000, p. 32, 2000.

[3] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. John Wiley & Sons, 2012.

[4] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998. [Online]. Available: https://doi.org/10.1162/089976698300017467

[5] C. L. Guy, F. Kaplan, J. Kopka, J. Selbig, and M. Scholz, "Non-linear PCA: a missing data approach," *Bioinformatics*, vol. 21, no. 20, pp. 3887–3895, 2005. [Online]. Available: https://doi.org/10.1093/bioinformatics/bti634

[6] N. D. Lawrence, "A Unifying Probabilistic Perspective for Spectral Dimensionality Reduction: Insights and New Models," *Journal of Machine Learning Research*, vol. 13, no. May, pp. 1609–1638, 2012. [Online]. Available: http://www.jmlr.org/papers/v13/lawrence12a.html

[7] J. A. Lee and M. Verleysen, *Nonlinear Dimensionality Reduction*, 1st ed. Springer Publishing Company, Incorporated, 2007.

[8] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[9] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[10] C. You, D. P. Robinson, and R. Vidal, "Scalable Sparse Subspace Clustering by Orthogonal Matching Pursuit," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2016, pp. 3918–3927. [Online]. Available: http://ieeexplore.ieee.org/document/7780794/

[11] Y. Bengio, M. Monperrus, and H. Larochelle, "Nonlocal Estimation of Manifold Structure," *Neural Computation*, vol. 18, no. 10, pp. 2509–2528, 2006. [Online]. Available: https://doi.org/10.1162/neco.2006.18.10.2509

[12] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps," *Proceedings of the National Academy of Sciences*, vol. 102, no. 21, pp. 7426–7431, 2005. [Online]. Available: https://www.pnas.org/content/102/21/7426

[13] D. L. Donoho and C. Grimes, "Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data," *Proceedings of the National Academy of Sciences*, vol. 100, no. 10, pp. 5591–5596, 2003. [Online]. Available: https://www.pnas.org/content/100/10/5591

[14] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, jun 2003. [Online]. Available: https://doi.org/10.1162/089976603321780317

[15] K. Q. Weinberger and L. K. Saul, "Unsupervised Learning of Image Manifolds by Semidefinite Programming," *International Journal of Computer Vision*, vol. 70, no. 1, pp. 77–90, oct 2006. [Online]. Available: https://doi.org/10.1007/s11263-005-4939-z

[16] H. Zha and Z. Zhang, "Continuum Isomap for manifold learnings," *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 184–200, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167947306004518

[17] Z. Zhang and H. Zha, "Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment (Dept of Computer Science, Pennsylvania State Univ, University Park, PA)," Tech Rep CSE-02-019, Tech. Rep., 2002.

[18] K. Q. Weinberger and L. K. Saul, "An introduction to nonlinear dimensionality reduction by maximum variance unfolding," in *AAAI*, vol. 6, 2006, pp. 1683–1686.

[19] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[20] P. Vepakomma and A. Elgammal, "A fast algorithm for manifold learning by posing it as a symmetric diagonally dominant linear system," *Applied and Computational Harmonic Analysis*, vol. 40, no. 3, pp. 622–628, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520315001451

[21] K. Zhang, I. W. Tsang, and J. T. Kwok, "Improved NystrÖM Low-rank Approximation and Error Analysis," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08. New York, NY, USA: ACM, 2008, pp. 1232–1239. [Online]. Available: http://doi.acm.org/10.1145/1390156.1390311

[22] K. Zhang and J. T. Kwok, "Clustered Nyström Method for Large Scale Manifold Learning and Dimension Reduction," *IEEE Transactions on Neural Networks*, vol. 21, no. 10, pp. 1576–1587, 2010.

[23] C. K. I. Williams and M. Seeger, "Using the Nyström method to speed up kernel machines," in *Advances in neural information processing systems*, 2001, pp. 682–688.

[24] A. Talwalkar, S. Kumar, and H. Rowley, "Large-scale manifold learning," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[25] E. Bingham and H. Mannila, "Random projection in dimensionality reduction," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '01*, ACM. New York, New York, USA: ACM Press, 2001, pp. 245–250. [Online]. Available: http://portal.acm.org/citation.cfm?doid=502512.502546

[26] W. B. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," *Contemporary mathematics*, vol. 26, no. 189-206, pp. 189–206, 1984. [Online]. Available: http://www.ams.org/conm/026/

[27] S. Dasgupta, "Learning mixtures of Gaussians," in *40th Annual Symposium on Foundations of Computer Science (Cat. No.99CB37039)*, IEEE. IEEE Comput. Soc, 1999, pp. 634–644. [Online]. Available: http://ieeexplore.ieee.org/document/814639/

[28] ——, "Experiments with random projection," in *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2000, pp. 143–151.

[29] S. Mylavarapu and A. Kaban, "Random projections versus random selection of features for classification of high dimensional data," in *2013 13th UK Workshop on Computational Intelligence (UKCI)*. IEEE, sep 2013, pp. 305–312. [Online]. Available: http://ieeexplore.ieee.org/document/6651321/

[30] L. Gondara, "RPC: An Efficient Classifier Ensemble Using Random Projections," in *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. IEEE, dec 2015, pp. 559–564. [Online]. Available: http://ieeexplore.ieee.org/document/7424375/

[31] Shuang Liu, Chunheng Wang, Baihua Xiao, Zhong Zhang, and Yunxue Shao, "Ground-based cloud classification using multiple random projections," in *2012 International Conference on Computer Vision in Remote Sensing*. IEEE, dec 2012, pp. 7–12. [Online]. Available: http://ieeexplore.ieee.org/document/6421224/

[32] M. Popescu, J. Keller, J. Bezdek, and A. Zare, "Random projections fuzzy c-means (RPFCM) for big data clustering," in *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, aug 2015, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/document/7337933/

[33] Atif Iqbal and A. M. Namboodiri, "Cascaded filtering for biometric identification using random projections," in *2011 National Conference on Communications (NCC)*. IEEE, jan 2011, pp. 1–5. [Online]. Available: http://ieeexplore.ieee.org/document/5734772/

[34] A. Anand, L. Wilkinson, and T. N. Dang, "Visual pattern discovery using random projections," in *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE, oct 2012, pp. 43–52. [Online]. Available: http://ieeexplore.ieee.org/document/6400490/

[35] X. Z. Fern and C. E. Brodley, "Random projection for high dimensional data clustering: A cluster ensemble approach," in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 2003, pp. 186–193.

[36] S. Han and M. Boutin, "The hidden structure of image datasets," in *2015 IEEE International Conference on Image Processing (ICIP)*, IEEE. IEEE, sep 2015, pp. 1095–1099. [Online]. Available: http://ieeexplore.ieee.org/document/7350969/

[37] T. Yellamraju and M. Boutin, "Clusterability and Clustering of Images and Other Real High-Dimensional Data," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 1927–1938, apr 2018. [Online]. Available: http://ieeexplore.ieee.org/document/8245810/

[38] T. Yellamraju, J. Hepp, and M. Boutin, "Benchmarks for Image Classification and Other High-dimensional Pattern Recognition Problems," *arXiv preprint arXiv:1806.05272*, jun 2018. [Online]. Available: http://arxiv.org/abs/1806.05272

[39] T. Yellamraju, A. J. Magana, and M. Boutin, "Investigating Students' Habits of Mind in a course on Digital Signal Processing," Accepted and under revision (available upon request), 2018.

[40] T. Yellamraju and M. Boutin, "Pattern Dependence Detection using n-TARP Clustering," *arXiv preprint arXiv:1806.05297*, jun 2018. [Online]. Available: http://arxiv.org/abs/1806.05297

[41] K. Fukunaga and T. Krile, "Calculation of Bayes' Recognition Error for Two Multivariate Gaussian Distributions," *IEEE Transactions on Computers*, vol. C-18, no. 3, pp. 220–229, mar 1969. [Online]. Available: http://ieeexplore.ieee.org/document/1671228/

[42] H. Chernoff, "A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the sum of Observations," *The Annals of Mathematical Statistics*, vol. 23, no. 4, pp. 493–507, dec 1952. [Online]. Available: http://projecteuclid.org/euclid.aoms/1177729330

[43] A. Bhattachayya, "On a measure of divergence between two statistical population defined by their population distributions," *Bulletin Calcutta Mathematical Society*, vol. 35, pp. 99–109, 1943.

[44] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[45] Z. Bai and H. Saranadasa, "EFFECT OF HIGH DIMENSION: BY AN EXAMPLE OF A TWO SAMPLE PROBLEM," *Statistica Sinica*, vol. 6, no. 2, pp. 311–329, 1996. [Online]. Available: http://www.jstor.org/stable/24306018

[46] M. S. Srivastava and M. Du, "A Test for the Mean Vector with Fewer Observations Than the Dimension," *J. Multivar. Anal.*, vol. 99, no. 3, pp. 386–402, mar 2008. [Online]. Available: http://dx.doi.org/10.1016/j.jmva.2006.11.002

[47] M. S. Srivastava, "A Test for the Mean Vector with Fewer Observations Than the Dimension Under Non-normality," *J. Multivar. Anal.*, vol. 100, no. 3, pp. 518–532, mar 2009. [Online]. Available: http://dx.doi.org/10.1016/j.jmva.2008.06.006

[48] S. X. Chen and Y.-L. Qin, "A two-sample test for high-dimensional data with applications to gene-set testing," *Ann. Statist.*, vol. 38, no. 2, pp. 808–835, 2010. [Online]. Available: https://doi.org/10.1214/09-AOS716

[49] M. E. Lopes, L. Jacob, and M. J. Wainwright, "A More Powerful Two-sample Test in High Dimensions Using Random Projection," in *Proceedings of the 24th International Conference on Neural Information Processing Systems*, ser. NIPS'11. USA: Curran Associates Inc., 2011, pp. 1206–1214. [Online]. Available: http://dl.acm.org/citation.cfm?id=2986459.2986594

[50] M. D. Ernst, "Permutation Methods: A Basis for Exact Inference," *Statistical Science*, vol. 19, no. 4, pp. 676–685, nov 2004. [Online]. Available: http://projecteuclid.org/euclid.ss/1113832732

[51] S. Wei, C. Lee, L. Wichers, and J. S. Marron, "Direction-Projection-Permutation for High-Dimensional Hypothesis Tests," *Journal of Computational and Graphical Statistics*, vol. 25, no. 2, pp. 549–569, 2016. [Online]. Available: https://doi.org/10.1080/10618600.2015.1027773

[52] J. B. Bell, A. N. Tikhonov, and V. Y. Arsenin, *Solutions of Ill-Posed Problems.* Winston Washington, DC, oct 1978, vol. 32, no. 144. [Online]. Available: https://www.jstor.org/stable/2006360?origin=crossref

[53] R. Tibshirani, "Regression Shrinkage and Selection Via the Lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, jan 1996. [Online]. Available: http://doi.wiley.com/10.1111/j.2517-6161.1996.tb02080.x

[54] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, apr 2005. [Online]. Available: http://doi.wiley.com/10.1111/j.1467-9868.2005.00503.x

[55] T. Yellamraju, A. J. Magana, and M. Boutin, "Board # 11 : Investigating Engineering Students Habits of Mind: A Case Study Approach," in *2017 ASEE Annual Conference & Exposition*. Columbus, Ohio: ASEE Conferences, jun 2017. [Online]. Available: https://peer.asee.org/27686

[56] A. L. Costa and B. Kallick, *Habits of mind across the curriculum: Practical and creative strategies for teachers.* ASCD, 2009.

[57] R. B. Palm, "Prediction as a candidate for learning deep hierarchical models of data," Master's thesis, Technical University of Denmark, 2012.

[58] M. Ristin, J. Gall, M. Guillaumin, and L. Van Gool, "From categories to subcategories: Large-scale image classification with partial class label refinement," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2015, pp. 231–239. [Online]. Available: http://ieeexplore.ieee.org/document/7298619/

[59] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, dec 2015. [Online]. Available: http://link.springer.com/10.1007/s11263-015-0816-y

[60] Yang Song, Weidong Cai, Qing Li, Fan Zhang, D. D. Feng, and H. Huang, "Fusing subcategory probabilities for texture classification," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2015, pp. 4409–4417. [Online]. Available: http://ieeexplore.ieee.org/document/7299070/

[61] B. Caputo, E. Hayman, and P. Mallikarjuna, "Class-specific material categorisation," in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, vol. 2. IEEE, oct 2005, pp. 1597–1604 Vol. 2. [Online]. Available: http://ieeexplore.ieee.org/document/1544908/

[62] L. Sharan, R. Rosenholtz, and E. Adelson, "Material perception: What can you see in a brief glance?" *Journal of Vision*, vol. 9, no. 8, pp. 784–784, sep 2010. [Online]. Available: http://jov.arvojournals.org/Article.aspx?doi=10.1167/9.8.784

[63] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing Textures in the Wild," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2014, pp. 3606–3613. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6909856

[64] L. Liu, C. Shen, and A. van den Hengel, "The treasure beneath convolutional layers: Cross-convolutional-layer pooling for image classification," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2015, pp. 4749–4757. [Online]. Available: http://ieeexplore.ieee.org/document/7299107/

[65] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2009, pp. 413–420. [Online]. Available: http://ieeexplore.ieee.org/document/5206537/

[66] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-UCSD birds 200," 2010.

[67] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, jun 2010. [Online]. Available: http://link.springer.com/10.1007/s11263-009-0275-4

[68] L. Bourdev, S. Maji, and J. Malik, "Describing people: A poselet-based approach to attribute classification," in *2011 International Conference on Computer Vision*. IEEE, nov 2011, pp. 1543–1550. [Online]. Available: http://ieeexplore.ieee.org/document/6126413/

[69] R. Durrant and A. Kaban, "Sharp Generalization Error Bounds for Randomly-projected Classifiers," in *Proceedings of the 30th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, S. Dasgupta and D. McAllester, Eds., vol. 28, no. 3. Atlanta, Georgia, USA: PMLR, 2013, pp. 693–701. [Online]. Available: http://proceedings.mlr.press/v28/durrant13.html

[70] R. J. Durrant and A. Kabán, "Random projections as regularizers: Learning a linear discriminant ensemble from fewer observations than dimensions," *JMLR: Workshop and Conference Proceedings*, vol. 29, pp. 17–32, 2013.

[71] G. Raskutti and M. W. Mahoney, "A statistical perspective on randomized sketching for ordinary least-squares," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 7508–7538, 2016.

[72] L. Zhang, M. Mahdavi, R. Jin, T. Yang, and S. Zhu, "Recovering the optimal solution by dual random projection," *JMLR: Workshop and Conference Proceedings*, vol. 30, pp. 1–23, 2013.

[73] J. Hepp, Y. Tarun, and M. Boutin, "Code and Dataset for Pattern Recognition Benchmarks," dec 2016. [Online]. Available: https://purr.purdue.edu/publications/2030/2

[74] R. FUKUNAGA, *Statistical pattern recognition*. Academic Press., 1990.

[75] M. Lichman, "{UCI} Machine Learning Repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[76] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, sep 1995. [Online]. Available: http://link.springer.com/10.1007/BF00994018

[77] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," in *Journal of Computer and System Sciences*, vol. 55, no. 1. Springer, aug 1997, pp. 119–139. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S002200009791504X

[78] S. Munder and D. Gavrila, "An Experimental Study on Pedestrian Classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 11, pp. 1863–1868, nov 2006. [Online]. Available: http://ieeexplore.ieee.org/document/1704841/

[79] M. Boutin, "The Pascal Triangle of a Discrete Image: Definition, Properties and Application to Shape Analysis," *Symmetry, Integrability and Geometry: Methods and Applications*, vol. 9, p. 31, apr 2013. [Online]. Available: http://www.emis.de/journals/SIGMA/2013/031/

[80] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-Means Clustering Algorithm," *Applied Statistics*, vol. 28, no. 1, p. 100, 1979. [Online]. Available: https://www.jstor.org/stable/10.2307/2346830?origin=crossref

[81] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data Via the EM Algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, sep 1977. [Online]. Available: http://doi.wiley.com/10.1111/j.2517-6161.1977.tb01600.x

[82] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH," in *Proceedings of the 1996 ACM SIGMOD international conference on Management of data - SIGMOD '96*, vol. 25, no. 2, ACM. New York, New York, USA: ACM Press, 1996, pp. 103–114. [Online]. Available: http://portal.acm.org/citation.cfm?doid=233269.233324

[83] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96, vol. 96. AAAI Press, 1996, pp. 226–231. [Online]. Available: http://dl.acm.org/citation.cfm?id=3001460.3001507

[84] C. C. Aggarwal, J. L. Wolf, P. S. Yu, C. Procopiuc, and J. S. Park, "Fast algorithms for projected clustering," in *Proceedings of the 1999 ACM SIGMOD international conference on Management of data - SIGMOD '99*, vol. 28, no. 2, ACM. New York, New York, USA: ACM Press, 1999, pp. 61–72. [Online]. Available: http://portal.acm.org/citation.cfm?doid=304182.304188

[85] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, *Automatic subspace clustering of high dimensional data for data mining applications.* New York, New York, USA: ACM Press, 1998, vol. 27, no. 2. [Online]. Available: http://portal.acm.org/citation.cfm?doid=276304.276314

[86] C. M. Procopiuc, M. Jones, P. K. Agarwal, and T. M. Murali, "A Monte Carlo algorithm for fast projective clustering," in *Proceedings of the 2002 ACM SIGMOD international conference on Management of data - SIGMOD '02*, ACM. New York, New York, USA: ACM Press, 2002, p. 418. [Online]. Available: http://portal.acm.org/citation.cfm?doid=564691.564739

[87] H.-P. Kriegel, P. Kroger, M. Renz, and S. Wurst, "A Generic Framework for Efficient Subspace Clustering of High-Dimensional Data," in *Fifth IEEE International Conference on Data Mining (ICDM'05)*, IEEE. IEEE, 2005, pp. 250–257. [Online]. Available: http://ieeexplore.ieee.org/document/1565686/

[88] I. Assent, R. Krieger, E. Müller, and T. Seidl, "INSCY: Indexing Subspace Clusters with In-Process-Removal of Redundancy," in *2008 Eighth IEEE International Conference on Data Mining*, IEEE. IEEE, dec 2008, pp. 719–724. [Online]. Available: http://ieeexplore.ieee.org/document/4781168/

[89] Man Lung Yiu and Nikos Mamoulis, "Frequent-pattern based iterative projected clustering," in *Third IEEE International Conference on Data Mining*, IEEE. IEEE Comput. Soc, 2003, pp. 689–692. [Online]. Available: http://ieeexplore.ieee.org/document/1251009/

[90] G. Moise, J. Sander, and M. Ester, "P3C: A Robust Projected Clustering Algorithm," in *Sixth International Conference on Data Mining (ICDM'06)*, IEEE. IEEE, dec 2006, pp. 414–425. [Online]. Available: http://ieeexplore.ieee.org/document/4053068/

[91] K. Sequeira and M. Zaki, "SCHISM: A New Approach for Interesting Subspace Mining," in *Fourth IEEE International Conference on Data Mining (ICDM'04)*, IEEE. IEEE, 2004, pp. 186–193. [Online]. Available: http://ieeexplore.ieee.org/document/1410283/

[92] G. Moise and J. Sander, "Finding non-redundant, statistically significant regions in high dimensional data," in *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD 08*, ACM. New York, New York, USA: ACM Press, 2008, p. 533. [Online]. Available: http://dl.acm.org/citation.cfm?doid=1401890.1401956

[93] K. Kailing, H.-P. Kriegel, and P. Kröger, "Density-Connected Subspace Clustering for High-Dimensional Data," in *Proceedings of the 2004 SIAM International Conference on Data Mining*, SIAM. Philadelphia, PA: Society for Industrial and Applied Mathematics, apr 2004, pp. 246–256. [Online]. Available: https://epubs.siam.org/doi/10.1137/1.9781611972740.23

[94] Y. Tarun and M. Boutin, "n-TARP Binary Clustering Code," may 2018. [Online]. Available: https://purr.purdue.edu/publications/2973/1

[95] E. Müller, S. Günnemann, I. Assent, and T. Seidl, "Evaluating clustering in subspace projections of high dimensional data," in *Proceedings of the VLDB Endowment*, vol. 2, no. 1, aug 2009, pp. 1270–1281. [Online]. Available: http://dl.acm.org/citation.cfm?doid=1687627.1687770

[96] R. Penrose and J. A. Todd, "A generalized inverse for matrices," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, no. 03, p. 406, jul 1955. [Online]. Available: http://www.journals.cambridge.org/abstract{_}S0305004100030401

[97] ——, "On best approximate solutions of linear matrix equations," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 52, no. 01, p. 17, jan 1956. [Online]. Available: http://www.journals.cambridge.org/abstract{_}S0305004100030929

[98] R. Hagen, *C* - Algebras and Numerical Analysis*. CRC Press, sep 2000. [Online]. Available: https://www.taylorfrancis.com/books/9781482270679

[99] J. A. Hoeting, D. Madigan, A. E. Raftery, and C. T. Volinsky, "Bayesian model averaging: a tutorial," *Statistical science*, vol. 14, no. 4, pp. 382–401, 1999. [Online]. Available: http://www.jstor.org/stable/2676803

[100] D. Korobilis, "Hierarchical shrinkage priors for dynamic regressions with many predictors," *International Journal of Forecasting*, vol. 29, no. 1, pp. 43–59, jan 2013. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0169207012000817

[101] E. I. George and R. E. McCulloch, "Variable Selection via Gibbs Sampling," *Journal of the American Statistical Association*, vol. 88, no. 423, pp. 881–889, sep 1993. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/01621459.1993.10476353http://www.tandfonline.com/doi/abs/10.1080/01621459.1993.10476353

[102] ——, "APPROACHES FOR BAYESIAN VARIABLE SELECTION," *Statistica Sinica*, vol. 7, no. 2, pp. 339–373, 1997. [Online]. Available: http://www.jstor.org/stable/24306083

[103] A. Papoulis and S. U. Pillai, *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.

[104] W. G. Cochran, "The χ^2 Test of Goodness of Fit," *The Annals of Mathematical Statistics*, vol. 23, no. 3, pp. 315–345, sep 1952. [Online]. Available: https://doi.org/10.1214/aoms/1177729380http://projecteuclid.org/euclid.aoms/1177729380

[105] R. A. Fisher, "On the Interpretation of χ 2 from Contingency Tables, and the Calculation of P," *Journal of the Royal Statistical Society*, vol. 85, no. 1, p. 87, jan 1922. [Online]. Available: https://www.jstor.org/stable/2340521?origin=crossref

[106] A. Agresti, "A Survey of Exact Inference for Contingency Tables," *Statistical Science*, vol. 7, no. 1, pp. 131–153, feb 1992. [Online]. Available: https://doi.org/10.1214/ss/1177011454http://projecteuclid.org/euclid.ss/1177011454

[107] N. Blomqvist, "On a Measure of Dependence Between two Random Variables," *The Annals of Mathematical Statistics*, vol. 21, no. 4, pp. 593–600, dec 1950. [Online]. Available: https://www.jstor.org/stable/2236609http://projecteuclid.org/euclid.aoms/1177729754

[108] N. Balakrishna and C. D. Lai, "Concepts of Stochastic Dependence," in *Continuous Bivariate Distributions*. New York, NY: Springer New York, 2009, pp. 105–140. [Online]. Available: http://link.springer.com/10.1007/b101765{_}4

[109] P. L. Chebyshev, "Des valeurs moyennes, Liouville's," *J. Math. Pures Appl.*, vol. 12, pp. 177–184, 1867.

[110] W. Feller, *An introduction to probability theory and its applications*. John Wiley & Sons, 2008, vol. 2.

[111] A. S. for Engineering Education, "Transforming Undergraduate Education in Engineering (TUEE)," *ASEE*, 2013.

[112] N. R. Council and Others, *Transforming undergraduate education in science, mathematics, engineering, and technology*. National Academies Press, 1999.

[113] E. A. C. ABET, "Criteria for Accrediting Engineering Programs Effective for Reviews during the 2017-2018 Accreditation Cycle," 2016.

[114] G. W. Clough and Others, "The engineer of 2020: Visions of engineering in the new century," *National Academy of Engineering, Washington, DC*, 2004.

[115] American Association for the Advancement of Science, "Project 2061: Science literacy for a changing future: A decade of reform." American Association for the Advancement of Science, 1995.

[116] K. Wage, J. Buck, C. Wright, and T. Welch, "The Signals and Systems Concept Inventory," *IEEE Transactions on Education*, vol. 48, no. 3, pp. 448–461, aug 2005. [Online]. Available: http://ieeexplore.ieee.org/document/1495653/

[117] M. D. Campbell, R. C. Houts, and E. A. Reinhard, "A Computer Utility Incorporating the FFT Algorithm for a Signal and System Theory Course," *IEEE Transactions on Education*, vol. 16, no. 1, pp. 42–47, 1973. [Online]. Available: http://ieeexplore.ieee.org/document/4320788/

[118] A. M. Goncher, D. Jayalath, and W. Boles, "Insights Into Students' Conceptual Understanding Using Textual Analysis: A Case Study in Signal Processing," *IEEE Transactions on Education*, vol. 59, no. 3, pp. 216–223, aug 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7403926/

[119] S. B. Merriam, *Case study research in education: A qualitative approach.* Jossey-Bass, 1988.

[120] K. M. Eisenhardt, "Building Theories from Case Study Research," *Academy of Management Review*, vol. 14, no. 4, pp. 532–550, oct 1989. [Online]. Available: http://journals.aom.org/doi/10.5465/amr.1989.4308385

[121] R. K. Yin, *Case study research: Design and methods vol 5*. Sage publications, 2009.

[122] M. Q. Patton, *Qualitative Research and Evaluation Methods, 2 ed.* SAGE Publications, inc, 2002.

[123] M. B. Miles and A. M. Huberman, *Qualitative data analysis: An expanded sourcebook*. sage, 1994.

[124] A. W. Haddad and M. Boutin, "Rhea: a student-driven tool for enhancing the educational experience," *Journal of Computing Sciences in Colleges*, vol. 26, no. 1, pp. 59–66, 2010.

[125] M. Boutin and J. Lax, "Engaging graduate students through online lecture creation," in *2015 IEEE Frontiers in Education Conference (FIE)*, IEEE. IEEE, oct 2015, pp. 1–4. [Online]. Available: http://ieeexplore.ieee.org/document/7344169/

[126] C. Steinkuehler and S. Duncan, "Scientific Habits of Mind in Virtual Worlds," *Journal of Science Education and Technology*, vol. 17, no. 6, pp. 530–543, dec 2008. [Online]. Available: http://link.springer.com/10.1007/s10956-008-9120-8

[127] K. Pearson, "Note on regression and inheritance in the case of two parents," *Proceedings of the Royal Society of London*, vol. 58, pp. 240–242, 1895.

[128] S. M. Stigler, "Francis Galton's account of the invention of correlation," *Statistical Science*, pp. 73–79, 1989.

[129] H. Wang and M. Song, "Ckmeans.1d.dp: Optimal k-means Clustering in One Dimension by Dynamic Programming." *The R journal*, vol. 3, no. 2, pp. 29–33, dec 2011. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/27942416http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5148156

[130] K. J. Rodgers, H. A. Diefes-Dux, and M. E. Cardella, "The nature of peer feedback from first-year engineering students on open-ended mathematical modeling problems," in *American Society for Engineering Education*. American Society for Engineering Education, 2012.

[131] J. Smith, G. Hoffart, and T. O'Neill, "Peer Feedback on Teamwork Behaviors: Reactions and Intentions to Change," in *American Society for Engineering Education*. American Society for Engineering Education, 2016.

[132] J. McGourty, P. Dominick, and R. Reilly, "Incorporating student peer review and feedback into the assessment process," in *FIE '98. 28th Annual Frontiers in Education Conference. Moving from 'Teacher-Centered' to 'Learner-Centered' Education. Conference Proceedings (Cat. No.98CH36214)*, vol. 1, IEEE. IEEE, 1998, pp. 14–18. [Online]. Available: http://ieeexplore.ieee.org/document/736790/

[133] R. Rada and Ke Hu, "Patterns in student-student commenting," *IEEE Transactions on Education*, vol. 45, no. 3, pp. 262–267, aug 2002. [Online]. Available: http://ieeexplore.ieee.org/document/1024619/

[134] Eric Zhi-Feng Liu, S. Lin, Chi-Huang Chiu, and Shyan-Ming Yuan, "Web-based peer review: the learner as both adapter and reviewer," *IEEE Transactions on Education*, vol. 44, no. 3, pp. 246–251, 2001. [Online]. Available: http://ieeexplore.ieee.org/document/940995/

VITA

VITA

Tarun Yellamraju received his Bachelors degree in Electrical Engineering with Honors, from the Indian Institute of Technology - Bombay, India in 2015. During his time as an undergraduate, he worked at INRIA Sophia Antipolis, in France for the summer of 2014, as a research intern. His project was on multiple object detection under the marked point process framework. Back at IIT Bombay, he worked on a compressed sensing based machine learning method for low bit rate video compression, that formed his undergraduate research project. His introductory research experiences at INRIA and IIT Bombay motivated him to pursue a PhD.

Tarun joined the PhD program in the school of Electrical and Computer Engineering at Purdue University, USA in 2015. Under the guidance of Dr. Mireille Boutin, he worked on statistical high-dimensional data analysis for his dissertation. The techniques developed were applied to Educational Data Analysis in a collaborative effort with colleagues at Purdue University. During his time as a graduate student at Purdue, he interned at Qualcomm in the summer of 2018, working on lens distortion correction in virtual reality headsets.

He is currently finishing his PhD at Purdue University in 2019. His current research interests include high-dimensional machine learning, image processing and computer vision. He is joining Qualcomm to work on Virtual and Augmented Reality systems.