BRIDGE ELEMENTS' WEIGHT DETERMINATION AND COMPONENTS' CONDITION PREDICTION USING MACHINE LEARNING APPROACH

by

Qozeem Olatunbosun Abiona

A thesis submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Master of Civil Engineering

Winter 2024

BRIDGE ELEMENTS' WEIGHT DETERMINATION AND COMPONENTS' CONDITION PREDICTION USING MACHINE LEARNING APPROACH

by

Qozeem Olatunbosun Abiona

Approved: ____
Monique Head, Ph.D., P.E.
Professor in charge of thesis on behalf of the Advisory Committee

Approved: ____
Jack A. Puleo, Ph.D.
Chair, Department of Civil and Environmental Engineering

Approved: ____
Levi T. Thompson, Ph.D.
Dean, College of Engineering

Approved: ____
Louis F. Rossi, Ph.D.
Vice Provost for Graduate and Professional Education and
Dean, Graduate College

ACKNOWLEDGMENTS

All glory to God almighty for seeing me through this master's degree study in civil engineering and I am elated by this great achievement. I also want to thank my dad, mum, and siblings for their support and care during my study. My sincere gratitude goes to my Advisor, Professor Monique Head, for her guidance and unwavering support towards the completion of the studies. It is indeed a great privilege to work under her tutelage and I have benefitted immensely from her wealth of experience. I am truly refined.

I also want to use this opportunity to show my gratitude to all the members of the Head Research Group (HRG) for being supportive and for their genuine acts of selflessness. Working with the research group allowed me to benefit from the experience of other group members because of the collaborative team spirit and made me more productive in my tasks. I could not ask for a better research group to work with, and I will greatly miss having you guys around.

Finally, I want to thank the entire civil engineering faculty including the graduate academic advisor, Jacquee, and the Professors for their kindness and support during my study. Special thanks also go to Professor Shenton and Professor Chajes for being on my thesis committee, I have benefitted greatly from their experience and willingness to bring out the best in students. More power to you.

TABLE OF CONTENTS

LIST OF TABLES vii
LIST OF FIGURES x
ABSTRACT xv

Chapter

1 OVERVIEW 1

1.1 Background of Study 1
1.2 Objectives of Research 6
1.3 Thesis Outline 6

2 LITERATURE REVIEW 7

2.1 Introduction 7

2.1.1 Bridge Inspection 7
2.1.2 Evaluation of Bridge Elements' Health Index 9
2.1.3 National Bridge Inventory 11

2.2 Bridge Elements' Weight 13
2.3 Synthesis of Component and Element-Level Data 17
2.4 Bridge Condition Prediction Models 18

2.4.1 Stochastic Model 19
2.4.2 Deterministic Model 20
2.4.3 Mechanistic Model 20
2.4.4 Machine Learning Models 21

2.4.4.1 Artificial Neural Networks 23
2.4.4.2 Activation Function 25
2.4.4.3 Tensorflow 27
2.4.4.4 Optimizers 28
2.4.4.5 Regularization Parameters 29

2.5 Features Affecting Bridge Components Condition Identified from Literature ... 30
2.6 Principal Component Analysis (PCA) 39

2.7 Random Forest Algorithm 44
2.8 Knowledge Gaps and Relevance of Research 46

3 METHODOLOGY 49

3.1 Description of Research Approach/Framework 49
3.2 Bridge Data Handling..50

3.2.1 Data Collection Procedure...50

3.2.1.1 Data Collection for Prediction Model 50
3.2.1.2 Data Collection for Bridge Elements' Weight Model 55

3.2.2 Data Pre-Processing 56
3.2.2.1 Data Cleaning for Prediction Model..56
3.2.2.2 Data Scaling for Prediction Model 57
3.2.2.3 Data Cleaning and Synthesis for Bridge Elements' Weight Model 58
3.2.3 Data Selection..60
3.2.3.1 Bridge Data Partitioning for Elements' Weight........................60
3.2.3.2 Deck Data for Model Development 64
3.2.3.3 Superstructure Data for Model Development..........................65
3.2.3.4 Substructure Data for Model Development.............................65

3.3 Data Analysis...66

3.3.1 Principal Component Analysis (PCA) for Prediction Model 66
3.3.2 Base Case Prediction Model...67
3.3.3 Artificial Neural Networks (ANN) Prediction Model Setup.................67
3.3.4 Prediction Model Performance Measurement 72
3.3.4.1 Model Performance 72
3.3.4.2 Confusion Matrix 73
3.3.5 Determination of Bridge Components' and Elements' Weight..............74
3.3.6 Formation Of Bridge Health Index (BHI) Equations 76

4 RESULTS AND DISCUSSION 77

4.1 Data Analysis Results for Prediction Model..77

4.1.1 Deck PCA Result...77
4.1.2 Deck Deterioration Model Results 81
4.1.3 Superstructure PCA Result..96
4.1.4 Superstructure Deterioration Model Results 99

4.1.5 Substructure PCA Result 113
4.1.6 Substructure Deterioration Model Results 116

4.2 Data Analysis Results for Bridge Weight 131

4.2.1 Deck Elements' Weight 147
4.2.2 Superstructure Elements' Weight 149
4.2.3 Substructure Elements' Weight..151
4.2.4 Overall Components' Weight 152
4.2.5 Resultant Bridge Elements' Weight..153

5 CONCLUSION AND FUTURE WORK 155

5.1 Summary 155
5.2 Recommendations and Future Work 159

REFERENCES...161

Appendix

A PERMISSION AND OTHER MATERIALS 166

LIST OF TABLES

Table 2-1: Condition rating description [24] 12

Table 2-2: Deck Features Identified from Literature 1. 35

Table 2-3: Deck Features Identified from Literature 2. 36

Table 2-4: Superstructure Features Identified from Literature. 37

Table 2-5: Substructure Features Identified from Literature. 38

Table 3-1: Bridge features downloaded for deck deterioration Model. 52

Table 3-2: Bridge features downloaded for superstructure deterioration Model. 53

Table 3-3: Bridge features downloaded for substructure deterioration Model. 54

Table 3-4: Bridge data format. 59

Table 3-5: Bridge categories in the analysis. 61

Table 3-6: The data frame format for each major component and the associated elements. 62

Table 3-7: The data frame format for components and overall rating. 62

Table 3-8: ANN Model Architecture for Deck. 69

Table 3-9: ANN Model Architecture for Superstructure. 70

Table 3-10: ANN Model Architecture for Substructure. 71

Table 4-1: Variance explained by principal components for the bridge deck. 78

Table 4-2: Classification report for deck deterioration model with 2 PC. 88

Table 4-3: Classification report for deck deterioration model with 3 PC. 89

Table 4-4: Classification report for deck deterioration model with 4 PC. 89

Table 4-5: Classification report for deck deterioration model with 5 PC. 90

Table 4-6: Classification report for deck deterioration model with 6 PC. 91

Table 4-7: Classification report for deck deterioration model with 7 PC. 91

Table 4-8: Classification report for deck deterioration model with 8 PC. 92

Table 4-9: Classification report for deck deterioration model with 9 PC. 92

Table 4-10: Classification report for deck deterioration model with all 14 features (Base Model). 93

Table 4-11: Variance is explained by the principal components of the bridge superstructure. 96

Table 4-12: Classification report for superstructure deterioration model with 2 PC. 100

Table 4-13: Classification report for superstructure deterioration model with 3 PCs......... 101

Table 4-14: Classification report for superstructure deterioration model with 4 PCs......... 101

Table 4-15: Classification report for superstructure deterioration model with 5 PC. 102

Table 4-16: Classification report for superstructure deterioration model with 6 PCs......... 102

Table 4-17: Classification report for superstructure deterioration model with 7 PCs......... 103

Table 4-18: Classification report for superstructure deterioration model with 8 PCs......... 104

Table 4-19. Classification report for superstructure deterioration model with 9 PCs. 104

Table 4-20: Classification report for superstructure deterioration model with 15 features (Base Model). 105

Table 4-21: Variance explained by principal components for bridge substructure. 114

Table 4-22: Classification report for substructure deterioration model with 2 PCs............ 118

Table 4-23: Classification report for substructure deterioration model with 3 PCs............ 118

Table 4-24: Classification report for substructure deterioration model with 4 PCs............ 119

Table 4-25: Classification report for substructure deterioration model with 5 PCs............ 120

Table 4-26: Classification report for substructure deterioration model with 6 PCs........... 120

Table 4-27: Classification report for substructure deterioration model with 7 PCs........... 121

Table 4-28: Classification report for substructure deterioration model with 8 PCs........... 121

Table 4-29: Classification report for substructure deterioration model with 9 PCs........... 122

Table 4-30: Classification report for substructure deterioration model with 10 PCs.......... 123

Table 4-31. Classification report for substructure deterioration model with 14 features (Base Model). 123

LIST OF FIGURES

Figure 1-1: Bridge Hierarchy [5]. 4

Figure 2-1: Hierarchy of Bridge Parts. 8

Figure 2-2: Typical Artificial Neural Networks (ANN) Layers. 25

Figure 2-3: Illustration of Principal Component Analysis [61]. 41

Figure 2-4: Sample Scree plot for PCA generated in R-programming language. 42

Figure 3-1: Research Framework. 49

Figure 3-2: Bridge Components Hierarchy. 63

Figure 3-3: Deck Hierarchy. 63

Figure 3-4: Superstructure Hierarchy 64

Figure 3-5: Substructure Hierarchy 64

Figure 3-6: Confusion Matrix Illustration. 74

Figure 4-1: Bridge Components Condition Prediction Model Flow-Chart. 77

Figure 4-2: Scree plot for deck PCA. 79

Figure 4-3: Relationship between deck features and principal components. 80

Figure 4-4: (a) Learning curve for the deck model with 2 Principal Components (b) Confusion matrix for deck model 2 PC on the test set. 82

Figure 4-5: (a) The learning curve for the deck model with 3 Principal Components (b) Confusion matrix for deck model 3 PC on the test set. 83

Figure 4-6: (a) Learning curve for the deck model with 4 Principal Components (b) Confusion matrix for deck model 4 PC on the test set. 84

Figure 4-7: (a) Learning curve for the deck model with 5 Principal Components (b) Confusion matrix for deck model 5 PC on the test set. 85

Figure 4-8: (a) Learning curve for the deck model with 6 Principal Components (b) Confusion matrix for deck model 6 PC on the test set. 86

Figure 4-9: (a) Learning curve for the deck model with 7 Principal Components (b) Confusion matrix for deck model 7 PC on the test set. 87

Figure 4-10: (a) The learning curve for the deck model with 8 Principal Components (b) Confusion matrix for deck model 8 PC on the test set. 88

Figure 4-11: (a) The learning curve for the deck model with 9 Principal Components (b) Confusion matrix for deck model 9 PC on the test set. 94

Figure 4-12: (a) The learning curve for the deck base model with all features (b) Confusion matrix for the deck base model with all features on the test set. 95

Figure 4-13: Scree plot for superstructure PCA. 97

Figure 4-14: Relationship between superstructure features and principal components. 98

Figure 4-15: (a) Learning curve for the superstructure model with 2 Principal Components (b) Confusion matrix for superstructure model 2 PC on the test set. 106

Figure 4-16: (a) Learning curve for the superstructure model with 3 Principal Components (b) Confusion matrix for superstructure model 3 PC on the test set. 107

Figure 4-17: (a) Learning curve for the superstructure model with 4 Principal Components (b) Confusion matrix for superstructure model 4 PC on the test set. 108

Figure 4-18: (a) Learning curve for the superstructure model with 5 Principal Components (b) Confusion matrix for superstructure model 5 PC on the test set. 109

Figure 4-19: (a) Learning curve for the superstructure model with 6 Principal Components (b) Confusion matrix for superstructure model 6 PC on the test set. 110

Figure 4-20: (a) Learning curve for the superstructure model with 7 Principal Components (b) Confusion matrix for superstructure model 7 PC on the test set. 111

Figure 4-21: (a) Learning curve for the superstructure model with 8 Principal Components (b) Confusion matrix for superstructure model 8 PC on the test set. 112

Figure 4-22: (a) Learning curve for the superstructure model with 9 Principal Components (b) Confusion matrix for superstructure model 9 PC on the test set. 113

Figure 4-23: (a) The learning curve for the superstructure base model with all features (b) Confusion matrix for the superstructure base model with all features on the test set.......... 113

Figure 4-24: Scree plot for substructure PCA. 115

Figure 4-25: Relationship between substructure features and principal components. 116

Figure 4-26: (a) Learning curve for the substructure model with 2 Principal Components (b) Confusion matrix for substructure model 2 PC on the test set. 125

Figure 4-27: (a) The learning curve for the substructure model with 3 Principal Components (b) Confusion matrix for substructure model 3 PC on the test set.......... 125

Figure 4-28: (a) The learning curve for the substructure model with 4 Principal Components (b) Confusion matrix for substructure model 4 PC on the test set.......... 126

Figure 4-29: (a) The learning curve for the substructure model with 5 Principal Components (b) Confusion matrix for substructure model 5 PC on the test set.......... 127

Figure 4-30: (a) The learning curve for the substructure model with 6 Principal Components (b) Confusion matrix for substructure model 6 PC on the test set.......... 127

Figure 4-31: (a) The learning curve for the substructure model with 7 Principal Components (b) Confusion matrix for substructure model 7 PC on the test set.......... 128

Figure 4-32: (a) The learning curve for the substructure model with 8 Principal Components (b) Confusion matrix for substructure model 8 PC on the test set.......... 129

Figure 4-33: (a) The learning curve for the substructure model with 9 Principal Components (b) Confusion matrix for substructure model 9 PC on the test set.......... 129

Figure 4-34: (a) The learning curve for the substructure model with 10 Principal Components (b) Confusion matrix for substructure model 10 PC on the test set.......... 130

Figure 4-35: (a) The learning curve for the substructure base model with all features (b) Confusion matrix for the substructure base model with all features on the test set.......... 131

Figure 4-36: Bridge Elements' Weight Flow-Chart. 132

Figure 4-37: Bridge Components' and Elements' Importance for Stringer/Multi-beam or Girder Bridge. 132

Figure 4-38: Bridge elements' weight for Stringer/Multi-beam or Girder Bridge. 133

Figure 4-39: Bridge Components' and Elements' Importance for Box Beam or Girder (Single or Spread) Bridge. 133

Figure 4-40: Bridge Elements' Weight for Box Beam or Girders (Single) Bridge. 134

Figure 4-41: Bridge Components' and Elements' Importance for Box Beam or Girder (Multiple) Bridge. 134

Figure 4-42: Bridge Elements' Weight for Box Beam or Girders (Multiple) Bridge. 135

Figure 4-43: Bridge Components' and Elements' Importance for Tee Beam Bridge. 135

Figure 4-44: Bridge Elements' Weight for Tee Beam Bridge. 136

Figure 4-45: Bridge Components' and Elements' Importance for Girder and Floor Beam System Bridge. 136

Figure 4-46: Bridge Elements' Weight for Girder and Floor Beam System Bridge. 137

Figure 4-47: Bridge Components' and Elements' Importance for Truss-Thru Bridge. 137

Figure 4-48: Bridge Elements' Weight for Truss-Thru Bridge. 138

Figure 4-49: Bridge Components' and Elements' Importance for Frame Bridge. 138

Figure 4-50: Bridge Elements' Weight for Frame Bridge. 139

Figure 4-51: Bridge Components' and Elements' Importance for Arch-Deck Bridge. 139

Figure 4-52: Bridge Elements' Weight for Arch Deck Bridge. 140

Figure 4-53: Bridge Components' and Elements' Importance for Channel Beam Bridge. .. 140

Figure 4-54: Bridge Elements' Weight for Channel Beam Bridge. 141

Figure 4-55: Bridge Components' and Elements' Importance for Truss-Deck Bridge. 141

Figure 4-56: Bridge Elements' Weight for Truss-Deck Bridge. 142

Figure 4-57: Bridge Components' and Elements' Importance for Movable Bascule Bridge. 142

Figure 4-58: Bridge Elements' Weight for Movable Bascule Bridge. 143

Figure 4-59: Bridge Components' and Elements' Importance for Arch-Thru Bridge. 143

Figure 4-60: Bridge Elements' Weight for Arch-Thru. 144

Figure 4-61: Bridge Components' and Elements' Importance for Segmental Box Girder Bridge. 144

Figure 4-62: Bridge Elements' Weight for Segmental Box Girder Bridge.......................... 145

Figure 4-63: Bridge Components' and Elements' Importance for Suspension Bridge......... 145

Figure 4-64: Bridge Elements' Weight for Suspension Bridge. 146

Figure 4-65: Bridge Components' and Elements' Importance for Stayed Girder Bridge..... 146

Figure 4-66: Bridge Elements' Weight for Stayed Girder Bridge. 147

Figure A1: Permission to reprint figure 1-1.. 166

Figure A2: Code for condition prediction models. 168

Figure A3: Code for bridge elements' weight.. 170

ABSTRACT

Managing the deterioration of the bridge components and elements has continued to be one of the major concerns for the Departments of Transportation (DOTs) in the United States due to the huge cost needed for constructing new bridges. To avert disasters that could lead to severe losses, deterioration models have been created to predict the future condition of the deck while attributing the deterioration to different factors elicited by engineering and statistical techniques. Previous deterioration models have been more like linear regression models and did not predict the discretized value of the deck condition rating. A prediction of 6.51 was approximated to be a condition rating of 7, which is inappropriate for a discrete data type. This research project uniquely combines all the bridge features identified in the literature by applying principal component analysis (PCA) to capture the variance in the dataset needed to predict the condition of the bridge components. Feedforward Artificial Neural Network (ANN) models were created using different numbers of principal components and the performances were compared with that of the base model that uses all the features collected from the literature. It was observed that 9, 9, and 10 principal components are needed to create a deterioration prediction model that gives a better prediction accuracy than the base model that uses all the bridge features in the Deck, Superstructure, and Substructure respectively. The deterioration of the bridge

elements is also known to influence the condition rating of the bridge components and the overall condition of the bridge. The weight or importance of the bridge elements influences the maintenance, repair, and replacement (MRR) schedule of the Departments of Transportation (DOTs) and the resource allocation to the structures. The DOTs currently use a cost-based approach to assign weight to bridge elements which can be in terms of the loss accrued during downtime or the amount needed for the replacement of the element. However, this approach does not consider the bridge element's structural relevance to the bridge's overall performance. This research also uniquely uses the Random Forest (RF) algorithm, an ensemble of decision trees, to evaluate the importance of different elements to the condition of the bridge components and the overall condition of the bridge. The analysis focused on 15 bridge design types in Delaware, Maryland, Pennsylvania, Virginia, and West Virginia and discovered that the weight of bridge elements is not constant as insinuated by the cost-based approach but varies based on its relevance to the bridge's structural performance. The resultant bridge elements' weight can be used to construct the Bridge Health Index (BHI) equations for the different bridge types. The novel approach herein provides the DOT personnel with data-driven evidence to determine which set of bridge elements to prioritize in their maintenance actions to improve the components' condition, and the overall condition of their bridge inventory and to ascertain if the elements receiving the highest priority in the MRR schedule and budget allocation are also the same set of elements that bridge inspectors regard as needing attention. Furthermore, the technique presented also serves as an approach for synthesizing the bridge component and element-level data and aids the conversion process between the two important datasets.

Chapter 1

OVERVIEW

1.1 Background of Study

Deterioration is inherent in bridges like many other transportation structures that support the movement of goods and services and promote economic activities. Bridge owners including the Departments of Transportation (DOTs) who receive federal funding for bridge maintenance in the United States are mandated to collect data and rate the condition of the bridge components based on the bridge deck, superstructure, and substructure at the minimum. As such, there are more than three decades worth of condition rating data that shows the health condition trend of in-service bridges in the nation. In 1997, the idea of collecting element-level data in addition to condition rating data on the bridge deck, superstructure, and substructure was first introduced via the AASHTO Commonly Recognized (CoRe) structural elements. The purpose of including element-level data was to make the bridge data collection process more consistent and quantitative [1] and it was not mandatory at this point. However, some DOTs in the United States of America started collecting element-level data in the early 1990s as part of their bridge management operations and only collected for their internal operations. The Moving Ahead for Progress in 21st Century (MAP-21) Act in 2014 mandated all bridge owners receiving federal funds to start collecting element-level data to help engineers monitor the life-cycle of a bridge and improve Bridge Management Systems (BMS) [2].

As of today, the bridge data collection in the U.S. is getting more robust with a large volume of component and element-level data collected yearly. There are over 600,000 bridges in the country and this number continues to rise due to the creation of new transportation routes, and more data related to the condition and features of bridges will be collected. Similarly, determining these bridges' deterioration remains a priority to avert disasters as about 7.5% of these bridges are in poor condition [3].

The deck, superstructure, and substructure are the major components of the bridge, and they have several other elements nested under them. The lingering issue has continued to be the effect of the deterioration of the bridge elements on the condition of the major components and the bridge as a whole. The importance of bridge elements is known to influence their prioritization in the maintenance, replacement, and repair (MRR) process. It becomes pertinent to determine which sets of elements have the highest impact on the condition rating of existing bridge components. The current method used by most departments of transportation (DOTs) in the United States to determine the bridge elements' weight is based on the failure cost, which is determined by experts in terms of the amount needed to replace the element or the amount that would be lost if such elements were not functioning optimally [4]. However, this approach does not consider the bridge element's structural relevance to the major components' overall performance.

FHWA [5] provided a hierarchy showing the connection/relationship between the major bridge components and the elements as shown in Figure 1-1. However, this hierarchy does not consider the different bridge design types that could occur and only shows the general connection between the elements and the major components. Different bridge design types have distinct sets of elements that are responsible for their structural performance; for example, the cables in a suspension bridge are important elements in the superstructure, however, they have no relevance in other bridge types like the Stringer/Multi-beam, Truss-thru, Box beam, etc. This makes it pertinent to consider the bridge elements' importance based on the design type of the bridge.

The bridge deck, which provides a travel surface for vehicles is susceptible to deterioration effects because of the high level of contact (abrasive force), traffic volume, and quicker exposure to adverse weather conditions like snowfall and freezing effects [6]. This prompts the usage of de-icing salts and other chemicals which have been linked to causing rapid deterioration of bridge decks [7-10]. Although there is no actual data on the frequency of salt added to the bridge deck, this can be implied by using other climatic factors like snowfall and freeze index as references [7]. For this reason, it becomes pertinent to give close monitoring to the bridge deck for proper maintenance, replacement, and repair (MRR) planning. The reconstruction of a bridge deck accrues a lot of costs associated with traffic jams, rerouting, minimizing noise and dust, and the actual reconstruction cost that may be averted by taking actions that will mitigate the full deterioration of the deck [11].

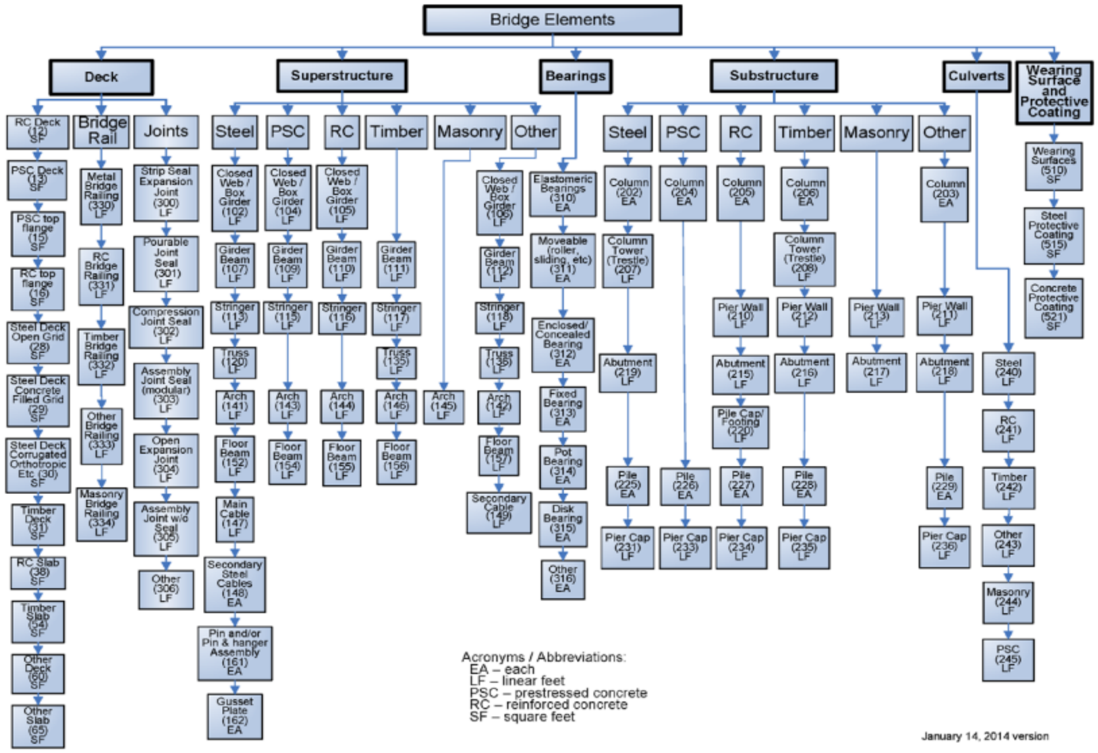


Figure 1-1: Bridge Hierarchy [5].

Also, the other bridge components i.e., the superstructure and substructure also experience deterioration which limits their load-carrying capacity and affects the overall bridge performance. Recently, more attention has been directed to the maintenance of the bridge deck because of the immediate discomfort it causes road users when it is in bad condition. However, the other components (i.e., superstructure and substructure) are as important to the overall condition of the bridge. The deterioration of any of the three major bridge components affects the overall condition rating of the bridge as this rating is taken as the lowest of the condition rating of the deck, superstructure, and substructure [12]. By foreseeing a bridge's imminent collapse or adverse deterioration, quicker actions can be taken to minimize the consequences [6, 13].

The study first computes the weight (or importance) of individual bridge elements and calculates the overall bridge health index (BHI) which is another measure of a bridge performance using the random forest algorithm. This was done to replicate the different bridge design types that occur in the National Bridge Inventory (NBI) to give a better understanding of the specific elements that influence the deterioration of the different types of bridges the most.

Many deterioration models have been created to predict the future condition of bridge components with many researchers attributing this drop in condition rating to different sets of features and recorded a reasonable level of accuracy [6, 7, 14-18]. However, combining all bridge features to create a standard model creates noise in the data because of redundancies in some of the features due to inaccurate data and low variability in certain instances, which does not help the prediction accuracy of the model created [17]. This research aims to apply principal component analysis (PCA), a dimensionality reduction technique, to find a good set of new features or principal components that represent the important information in the dataset and create a more standardized model. Artificial Neural Network (ANN) models will be created using these principal components to predict the future condition of the bridge components. This model is intended to have a lower computation cost in terms of the complexity of the architecture when compared to using the original set of features without having any significant effect on prediction accuracy. Also, the models created will predict the actual discrete value of the deck condition rating using an appropriate classification algorithm framework.

1.2 Objectives of Research

The proposed research objective for this research is to develop bridge components' deterioration models using the components' inspection data to predict the future condition of bridge components in appropriate classes while exploring the effectiveness of dimensionality reduction in bridge data when creating the prediction models. Principal components (PC), which are less computationally expensive and produce equal or better prediction accuracy compared to the base model (i.e., with the original bridge feature set), will be used to create prediction models. A classification model will also be created instead of the regression-based approach and prevent the need for approximating the predicted condition rating values. The analysis will go a step further to determine the bridge elements that influence the deterioration of the bridge components and overall bridge condition the most. The novelty of the proposed approach identifies critical quality bridge features to be harnessed using principal component analysis to filter data rather than considering irrelevant feature selection techniques. Correlations between the component and element-level data are determined to define bridge weights from this data-driven analysis to redefine bridge health index (BHI) equations for different bridges based on main-span design types. The broader impact of this study supports BHI equations using a data-driven approach given the wealth of bridge inspection data now available at the bridge element level for evaluating bridge health.

1.3 Thesis Outline

This section is illustrated in the table of contents, highlighting several parts of this thesis including the knowledge gaps and the approaches that were adopted to solve the existing problems.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

The importance of bridge elements to the overall condition the of bridge and the ability to monitor the deterioration of the major bridge components are important topics that drive the effort for the continuous improvement of bridge infrastructure. The need for bridges that are performing optimally cannot be overemphasized because they form a core part of the transportation network that drives the economy and other societal needs. It becomes essential to discuss the rudiments of this important infrastructure and several efforts that have gone into putting it in good condition.

2.1.1 Bridge Inspection

Bridges are made up of several parts that act together as a unit to facilitate transportation. To keep these parts in good working condition, bridges are inspected regularly to observe their condition and ability to perform optimally according to the specified design. Most bridges are inspected every two (2) years except in special cases where closer monitoring is required for heavily deteriorated bridges [12]. Bridges are made up of three major components i.e., deck, superstructure, and substructure, and each of these components is made up of elements that are responsible for its structural configuration and influence its condition. Most of the bridge elements are nested under these three major components termed the National Bridge Inventory (NBI) rating or General Condition Rating as illustrated in Figure 2-1.

The NBI ratings have been in use since the 1970s and a lot of data have been accrued to date [19].

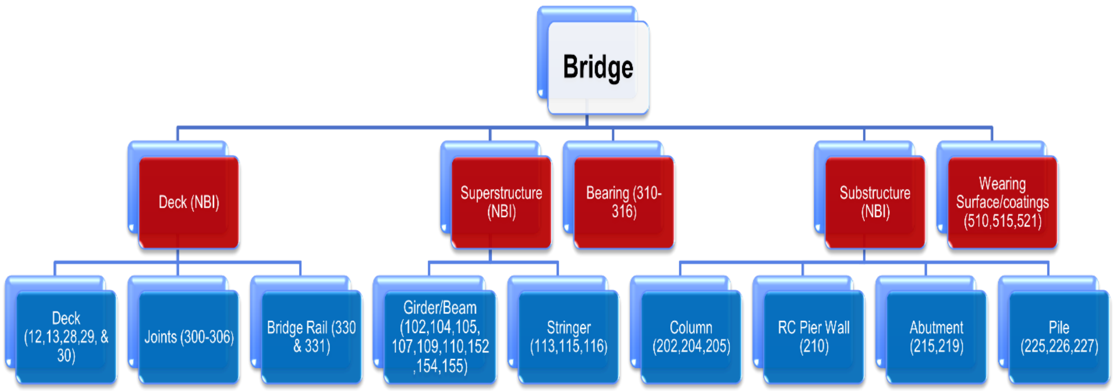


Figure 2-1: Hierarchy of Bridge Parts.

Bridge elements are divided into three categories, according to the AASHTO Manual for Bridge Element Inspection (MBEI): 1) National Bridge Element (NBE), 2) Bridge Management Element (BME), and 3) Agency-defined Element (ADE). This categorization is important to have a simplified technique for element condition assessment that can be adopted nationwide [12]. The NBE is the primary load-carrying element of the bridge. BME are elements used to improve the performance of the NBE, and ADE are elements specific to a DOT that are considered important to the bridge condition. Furthermore, a sub-element of the NBE and BME can also be defined [1].

The collection of element-level condition data is gaining acceptance among State DOTs, where there is a need to document the condition states of bridges to efficiently utilize these data for an improved Bridge Management System (BMS). It is essential to have a consistent scale for measuring bridge conditions that will help to establish accurate bridge evaluation [20]. Lin, Pan [20] also noted that without the element-level data, existing manuals only introduce material distresses for condition rating but overlook bridge element conditions that affect the bridge performance and indicate the likelihood of failure.

Fiorillo and Nassif [19] further clarified the mode of identifying elements using a unique identification (ID) called an Element Number (EN). Elements with IDs of less than 100 are in the deck region of the bridge. IDs between 100 and 199 are for elements in the superstructure and IDs between 200 and 299 are for the substructure elements. Bridge Management Elements like the joints are assigned IDs between 300 and 399 while ADEs like the wearing surface are assigned IDs of 500 upwards. In some DOTs, element numbers 800 and above are reserved for agency-defined elements [1]. This helps sort elements and prevents mismatches when trying to correlate the performance of related elements, which is important for having a more definitive maintenance schedule and budget allocation for structure improvement.

2.1.2 Evaluation of Bridge Elements' Health Index

A bridge element's status is represented in terms of its quantity in each condition state (CS) as specified by AASHTO [12]. CS1 signifies a good state, CS2- is a fair state, CS3- is a poor state, and CS4- is a severe state. These condition states are all that are required to determine the health index, HI_e, of each element, where a weighted factor is assigned to each of them. For example, CS1 has the greatest effect on the health index and is assigned a factor of 1, CS2 – 0.67, CS3 – 0.33, and CS4 – 0, assuming an equally-weighted distribution. Other weight distributions like 1:0.5:0.25, 1:0.4:0.1:0, and 1:0.8:0.4:0 have also been used in practice and comparisons are made to determine which of them results in a health index value that most accurately depict the actual physical condition of the bridge elements. The number of elements in CS4 is assumed to have no impact on the health index of the element [12]. The condition state data of each bridge element are converted into an element health index (HI_e), which is a single value between 0 and 100%. Equation 2.1 illustrates how to compute the health index of individual bridge elements.

Jiang and Rens [21] and Jiang and Rens [22] examined a bridge deck element whose health index was computed using the linear health index coefficient ratio of 1:0.67:0.33:0 and observed that the health index did not change significantly even as the bridge got older and with apparent deterioration. The authors suggested that using the health index coefficient ratio 1:0.4:0.1:0 for computing the elements' health index is more realistic and better represents the actual condition of the bridge elements. In their research, Inkoom, Sobanjo [23] evaluated the bridge elements' health index using 1:0.5:0.25:0, 1:0.67:0.33:0, and 1:0.8:0.4:0 separately and concluded that the weight factor ratio of 1:0.5:0.25:0 and 1:0.67:0.33:0 provide a more conservative computation of the elements' health index and may represent the present realities of these elements. In this research project, the health index of the bridge elements will be computed using the 1:0.4:0.1:0 weight factor ratio as it was observed that the linear approach (1:0.67:0.33:0) does not produce a realistic estimate of the elements' health index when juxtaposed with the condition rating of the associated major component. For instance, a bridge deck that is rated 5 has the associated deck element having a health index greater than 90% in many instances.

$$Hi_e = \sum \text{((health index coefficient) * (\% of the element in each condition state)} \quad (2.1)$$

2.1.3 National Bridge Inventory

The National Bridge Inventory (NBI) is the most complete and reliable source of bridge condition data in the United States of America (USA), providing the general condition rating of the deck, superstructure, and substructure on a scale of 0 (failed condition) to 9 (excellent condition) [12]. The indicator for each of the condition states is shown in Table 2-1. The NBI database does not only contain data on the condition ratings of the deck, superstructure, and substructure, but also data on factors like bridge properties, climatic conditions, and traffic volume that could influence the deterioration and distress of the bridges. This database also helps in prioritizing funding allocations for bridge maintenance, repair, and replacement (MRR), and provides a source of reporting to Congress on the status of bridges in the country [24]. In 2014, the Moving Ahead for Progress in 21st Century (MAP-21) Act mandated that all state DOTs collect element-level data, which makes the NBI database larger and helps to capture the relationship between major deterioration components and their sub-elements, also making the Bridge Management Systems (BMS) more robust [2].

The bridge components condition rating assessment is carried out by trained bridge inspectors who check for some physical attributes of the component and assign the appropriate condition rating value. The condition rating of the deck, superstructure, and substructure is alternatively called the general condition ratings (GCR) and can further be divided into categories. Ratings of 0-4 are categorized as poor, 5 and 6 – as fair, and 7 through 9 – as good [12]. The overall condition rating of a bridge is taken as the minimum of the rating assigned to the deck, superstructure, and substructure, e.g., a bridge with the deck at 5, superstructure at 6, and substructure at 7 will have an overall condition rating of 5 [12]. Other data available in the NBI database include the load rating of the bridge, dimensions of spans, traffic volume, highway functional class, etc. The NBI database has a total of 116 items that represent data on the bridge geometry, climatic and environmental conditions, deterioration conditions, traffic conditions, and the location of the structure. The full description of these items is available in the NBI data dictionary.

Table 2-1: Condition rating description [24]

Code	Condition	Description
N	Not Applicable	Component does not exist.
9	Excellent	Isolated inherent defects.
8	Very Good	Some inherent defects.
7	Good	Some minor defects.
6	Satisfactory	Widespread minor or isolated moderate defects.
5	Fair	Some moderate defects; strength and performance of the component not affected.
4	Poor	Widespread moderate or isolated major defects; strength and/or performance of component is affected.
3	Serious	Major defects; strength and/or performance of the component is seriously affected. Condition typically necessitates more frequent monitoring, load restrictions, and/or corrective actions.

2	Critical	Major defects; component is severely compromised. Condition typically necessitates frequent monitoring, significant load restrictions, and/or corrective actions in order to keep the bridge open.
1	Imminent Failure	Bridge is closed to traffic due to component condition. Repair or rehabilitation may return the bridge to service.
0	Failed	Bridge is closed due to component condition, and is beyond corrective action. Replacement is required to restore service.

2.2 Bridge Elements' Weight

Knowledge of the element weights of bridge components is a requirement to determine the significance of their deterioration to the overall bridge health. The overall condition state of a bridge cannot be derived from the simple addition of effective element deficiencies because of the complexity of the bridge system, having a range of material types and components [25].

Efforts have been made over the past decades to convert the health index of elements into the general health index of the bridge. AASHTO [12] introduced the AASHTO Bridge Management (BrM) software that combines the individual health index of elements into a single rating on a scale of 0 to 100. The effect or impact of an element's deterioration on the overall health index of the bridge still depends on the weighted factors adopted by the state DOTs which are mostly based on the failure or replacement cost. The overall health index of a bridge is termed the Bridge Health Index (BHI) and can be calculated as illustrated in Equation 2.2.

$$\text{BHI} = \sum ((\text{Element Weight}) * (\text{Element Health Index, } Hi_e)) \quad (2.2)$$

The California Bridge Health Index (BHI) method posited that the health index of an element is proportional to its quantity in a particular condition state and the economic impact of the element's failure [4]. In the study, failure cost was introduced as the weight emphasizing the importance of the element to the overall bridge health since depreciation in element value can be measured over time and the associated replacement cost can be estimated by experts. However, the limitation of this method is that element failure cost varies across states and agencies--a model cannot be universally adopted to calculate bridge health index across states using this technique because of varying economic conditions. Also, this technique is more fixated on the dollar value of the bridge elements rather than their functionality to the optimum performance of the bridge. This approach is the most widely adopted element condition index by the DOTs in the United States which raises the need for other data-driven alternatives that solely use the condition state and rating data collected to determine the importance of bridge elements and eliminate the bias that can be associated with the failure and replacement cost method normally determined by experts.

Inkoom and Sobanjo [26] developed a bridge element importance weighting system that used the availability of the element and the criticality of its downtime to the performance of the bridge. The availability index gives the residual strength of the element expressed in terms of uptime and downtime. Failure and repair rates of important elements were also considered in the study as major factors because their failure will have the greatest effect on the performance of the bridge. However, the computation of the availability index is highly probabilistic and there is limited actual data that has been collected to measure the downtime and uptime of bridge elements proposed in the study.

Abiona, Head [27] applied artificial neural networks (ANN) to determine which bridge elements are more important to the overall condition of the bridge. This was done by setting the health index values of the bridge elements as the independent variables and the overall condition of the bridge as the dependent variable. The relevance of each bridge element in predicting whether the overall condition of the bridge is poor, fair, or good is used to assign importance to the elements. In this study, it was observed that the bridge abutment has the highest level of importance to the overall condition of the bridge while the gusset plate has the least importance when considering the aggregation of all the bridges in Delaware, Maryland, Pennsylvania, Virginia, and West Virginia. However, the bridges were not classified based on the design type which means that bridge elements in some less frequent bridge types will have low representation in the data set causing their importance to be underestimated.

Inkoom, Sobanjo [23] developed an importance factor based on the element's replacement costs, long-term costs, vulnerability to hazard risk, and the combination of all three measures. The replacement cost approach was done by calculating the unit cost of each element and finding the ratio to that of the total unit cost of all elements in the bridge. The long-term cost approach was done similarly to the replacement cost approach, but the long-term cost was computed using the product of the unit cost and the element's quantity. The ratio between the long-term cost of each element to the total long-term cost of all the elements in the bridge gives the importance factor. The elements' importance factor based on vulnerability to the hazard was computed using a vulnerability index scale based on the type of hazard that is predominant in the bridge location. It was observed that the elements' importance based on the vulnerability to hazard is higher than that of the replacement cost and long-term cost. Finally, the combined approach allows the addition of some level of importance to bridge elements that were deemed irrelevant by the individual approaches. However, the approaches adopted by the authors are mainly based on cost and not the importance of the element to the functionality of the bridge. Given the amount of bridge element data collected since 2015, there is a need to consider the role and impact of these elements on the bridge vulnerability using a data-driven approach.

Jiang and Rens [21] developed an alternative approach for determining the bridge elements' importance based on the conventional failure and replacement cost. The bridge elements' importance is made flexible so that elements in a deplorable state are assigned a higher weight greater than what was initially assigned making the overall condition of the bridge more sensitive to reflect the condition of small bridge elements. In this approach, the increase in the severity of the distress of the bridge elements leads to higher importance on the overall condition of the bridge. Adjustment factors were used such that if the health index of an element reduces below 40%, the importance of the element increases by eight times. However, like Inkoom, Sobanjo [23], the basis of the analysis is rooted in the cost of the elements, which might introduce some initial bias and can vary from one state to another depending on the economic condition. Also, increasing the importance of the elements in a poor state by eight times might lead to being overly conservative in stating the overall condition of the bridge.

By comparing with what is being done internationally, in the United Kingdom, elements' significance to the overall bridge health is represented using an assigned

Element Importance Factor (EIF) and this can be combined with the Element Condition Score (ECS) to determine the overall condition index of the bridge [4]. The major setback of this approach is that the EIF is subjective; in other words, an element that is perceived by the inspector to be of less importance might carry more weight depending on the complexity and current state of the bridge. Furthermore, Chase, Adu-Gyamfi [4] also described the weighted averaging method of prioritizing elements' importance adopted by the Finnish Road Administration. The bridge is divided into nine structural parts and a weight factor is assigned to each part using experts' opinions. Although this method can capture the element-level defects in the overall bridge condition rating, there is no index integrating the structural condition and function of the elements.

2.3 Synthesis of Component and Element-Level Data

As it is already known that the condition of the major bridge components (i.e., deck, superstructure, and substructure) are influenced by conditions of the sub-elements that are nested under them, it becomes pertinent to synthesize these two important data sets. According to the bridge inspection reference manual, the condition rating of the major components is assigned based on the inspected condition of the sub-elements that are nested under them, and can be deduced that the deterioration of the major components and associated bridge elements are interconnected [28].

In a survey conducted by Bektaş [1] on 51 Departments of Transportation (DOTs) in the United States to observe if they have a prescribed set of activities for comparing the element-level data and general condition rating data. The result of the survey shows that 26 of these DOTs which is equivalent to 51% indicated that they do not compare the element-level data and general condition rating data, 13 DOTs (25%) indicated that they developed a conversion profile/model, but it needs further improvement, 6 DOTs (12%) indicated that they use the default conversion profile available in the Bridge Management System (BMS), and only 6 DOTs (12%) indicated that they developed a conversion profile they are confident in. This goes to show the amount of benefit the DOTs will get from having a framework that can synthesize these two distinct data sets. Apart from helping them to gain better intuition from the bridge data that were collected, it also helps to validate their maintenance, repair, and replacement (MRR) schedule.

This research explores techniques for capturing how different bridge elements influence the deterioration of the major components and also setting up a reliable prediction model for forecasting the future condition of the major components (i.e., deck, superstructure, and substructure).

2.4 Bridge Condition Prediction Models

Since the commencement of data collection by most DOTs in the 1990s, different kinds of deterioration models have been developed to monitor the condition of bridges in the country. Many researchers have applied different statistical tools, and more recently, machine learning algorithms to develop deterioration models that utilize these data for making forecasts on the future condition of bridges. These models form the backbone of many bridge management systems (BMS) and have a great influence on the maintenance, repair, and replacement (MRR) plan [15]. The major types of models used are stochastic, deterministic, mechanistic, and machine learning models.

2.4.1 Stochastic Model

This is a type of model that allows uncertainties and randomness in some quantities to be built into it, and it is usually time or condition-state-bound. The deterioration process for the condition state-based is modeled using the probability of transition from one state to another while the time-based is modeled using the amount of time a component remains in a condition without transitioning [29]. The condition state-based model is more widely adopted in bridge deterioration modeling. Ranjith, Setunge [30] used a stochastic Markov chain to develop a deterioration model for predicting the future condition of timber bridge elements in Australia in the form of a probabilistic estimate. Ranjith, Setunge [30] used the condition rating data to develop transition probabilities and applied a percentage prediction method, regression-based optimization method, and nonlinear optimization technique to predict transition matrices and transient probabilities. It was observed that the nonlinear optimization technique was more mathematically acceptable and predicted the progress of deterioration with better accuracy than the other two methods. However, Madanat, Mishalani [31] developed an econometric method, which was based on an ordered probit technique for making transition probabilities using condition rating data. This method has the advantage of treating the deterioration of the components as a latent variable and recognizes the discrete nature of the condition rating data. Madanat, Mishalani [31] also concluded that the ordered probit technique produced a more accurate estimation of the transition probabilities when compared to the regression-based method. However, this model is limited because of the assumption that the future condition only depends on the present condition state and neglects the historical data of the bridge.

2.4.2 Deterministic Model

Deterministic models assume that there is a constant relationship between the factors causing deterioration and the condition rating of the bridge components. This model is very simple to apply in predicting the future condition of bridges and can be adopted at the network level [29]. Moomen, Qiao [7] created deterministic models for bridge components to support Indiana's bridge management system (BMS). This model was able to capture the complex relationship between some factors like bridge age, length of span, number of freeze-thaw cycles, etc., and the condition rating of these bridge components. Trans [32] used data mining and geographical information systems (GIS) with the NBI data to create a prediction model. Trans [32] created several generalized linear models, generalized additive models, and a combination of the two models. It was observed that the generalized linear model gave a better prediction and was more accurate. However, random errors in the prediction are usually neglected and do not give room for uncertainty or the influence of undeclared variables [33].

2.4.3 Mechanistic Model

Mechanistic models are different from other statistical models because they do not use the historical data of the bridges to make predictions but rather the mathematical description of the phenomenon involved in the degradation [34]. To describe the deterioration mechanism mathematically, the underlying causes of deterioration must be known and studied throughout the life of the bridge as against the early deterioration process that was only measured during the onset of visible defects [34]. The deterioration predictions of the mechanistic models are more quantitative and are more appropriate for project-level analysis [29]. Lu, Liu [35] developed a mathematical model that can predict the time from corrosion initiation in a structural member to the appearance of the first crack on the concrete. This model established a relationship between the expansive pressure and the amount of corrosion in the steel using the mechanics of elasticity and was able to predict the time to the appearance of the cover crack with a reasonable level of accuracy. However, this model requires a large amount of data and is computationally expensive for a large bridge network, making it difficult to adopt in bridge management systems [29].

2.4.4 Machine Learning Models

Machine learning is an artificial intelligence (AI) system that allows computers to solve problems involving knowledge of the real world by extracting patterns from raw data to make decisions that appear subjective [36]. The advent of different machine learning algorithms has seen great acceptance in bridge deterioration modeling. This has caused a significant shift from the stochastic and deterministic-based model that was widely adopted in different applications. Srikanth and Arockiasamy [29] did an extensive review of models created with stochastics, deterministic, and Artificial Neural Network (ANN) based approaches for timber and concrete bridges. Srikanth and Arockiasamy [29] concluded that the stochastic-based method like the Markov chain provided a lower level of accuracy because of the assumption that the future condition only depends on the present condition which resulted in many unused condition rating data. Overall, the authors concluded that the ANN-based model gave the highest level of accuracy in predicting the future condition of the bridge components.

Althaqafi [37] used historical bridge condition data in Ohio from 1992 to 2019 to develop deterioration models for the bridge deck, superstructure, and substructure using both ANN and Markov Chain. The analysis result shows that the ANN model is far superior to the Markov chain model in terms of prediction accuracy and provides a more reliable means to forecast the future condition of bridge components. Miao and Yokota [38] developed a bridge deterioration prediction model using both Markov chain and recurrent neural network (RNN). The analysis result shows that the RNN model predicts the bridge deck deterioration faster than the Markov chain model. Nguyen and Dinh [13] applied ANN to predict the condition rating of bridge decks using 2572 bridge samples in the state of Alabama collected from the National Bridge Inventory (NBI) and observed a prediction accuracy of 73.6% which is more than what is obtainable from any deterministic, stochastic, or mechanistic model.

Apart from predicting the future condition of bridge components, machine learning algorithms have been applied to bridge data to improve usability and help in making informed decisions on bridge preservation. Fiorillo and Nassif [19] applied deep convolutional neural networks (CNN) to convert between the bridge elements condition and the NBI condition ratings. This study is important because the element level data collection was only started recently while the NBI condition rating data runs over three decades, improving the conversion between the two datasets will help to reveal the historical trend of the bridge elements' condition and improve their maintenance schedule. The CNN model was able to predict the ratings of the bridges with an accuracy of up to 90%. Similarly, Bektas, Carriquiry [39] used classification and regression trees (CART) to predict the NBI condition ratings from the element-level condition data. The method and machine learning algorithm presented by the authors gives another approach for converting between the two important bridge datasets. Wang, Yao [40] developed a machine learning framework for predicting bridge defect detection cost by using a random forest algorithm to determine the importance of the major factors affecting detection cost and used a combination of genetic algorithm and multilayer neural network to develop the detection cost prediction model.

2.4.4.1 Artificial Neural Networks

Among the different machine learning algorithms, the artificial neural network has seen the most application in bridge deterioration prediction. Miao [41], Huang [15], Ali, Elsayegh [42], Nguyen and Dinh [13], Assaad and El-adaway [14], Althaqafi and Chou [43], etc. have all applied ANN to predict the deterioration of bridge components using bridge inspection data collected from the national bridge inventory. The quick acceptability of ANN among researchers can be attributed to its ease of use and result interpretation.

The artificial neural network (ANN) is a form of deep learning algorithm that was developed to mimic the human brain where several neurons can interact and share information. Like the human brain, the ANN can also function even when some of the neurons are dead or disabled [44]. The model is created through an iterative learning process when it is given a set of examples to study the pattern in which they are organized. Weights are assigned to the neurons (or units) and are continuously evaluated until there are no significant changes in them again. The impulse received by each neuron is computed as the weighted sum of the input signal received from a preceding layer [45].

The learning method of the ANN can be in three major forms which are supervised, unsupervised, and reinforcement learning. Supervised learning involves a dataset with a known output or response variable whereas unsupervised learning involves a dataset without an output variable and the work of the algorithm is to detect patterns or any other interesting aspect of the data. Reinforcement learning is used for a situation where the task is too complicated to be programmed in the computer so the learner is not told about the actions to take but uses the rewards and penalties from actions taken to determine which direction is the best [45].

The ANN framework usually contains three categories of layers i.e., input layer, hidden layer, and output layer. The signal flows from the input units to the output units in a forward direction and this framework is referred to as a feed-forward network or multilayer perceptron (MLP) [45]. The MLP consists of the input and output layer at the extreme, and multiple hidden layers in the middle [46]. Figure 2-2 shows a typical arrangement of the layers in an ANN. The input layer is the first layer in the neural network and corresponds to the actual features that describe the dataset. The number of neurons in the input layer must correspond to the number of features that are fed into the model. The hidden layer is the middle layer of the neural network, and the number of layers and neurons can be modified depending on the intended complexity of the model. The output layer is the final layer of the neural network and the number of neurons in this layer is dependent on the data type of the response variable. For a regression problem where the response variable is a continuous data type and can take an infinite number of possible outcomes, the number of neurons in the output layer is 1 and in a binary classification problem where the response variable can take only two possible values which can be 0 or 1, the number of neurons in the output layer for this kind of model is also 1. For a multi-class classification problem where the output can only be a set of discrete numbers within a particular range of values, the number of neurons must be equal to the number of possible discrete values. E.g., a bridge deterioration model that intends to predict the condition rating of bridge components into a value from 0 to 9 must have exactly 10 neurons in the output layer. An important term that helps to differentiate how each of the layers is treated concerning the nature of data they handle is called the 'activation function'.

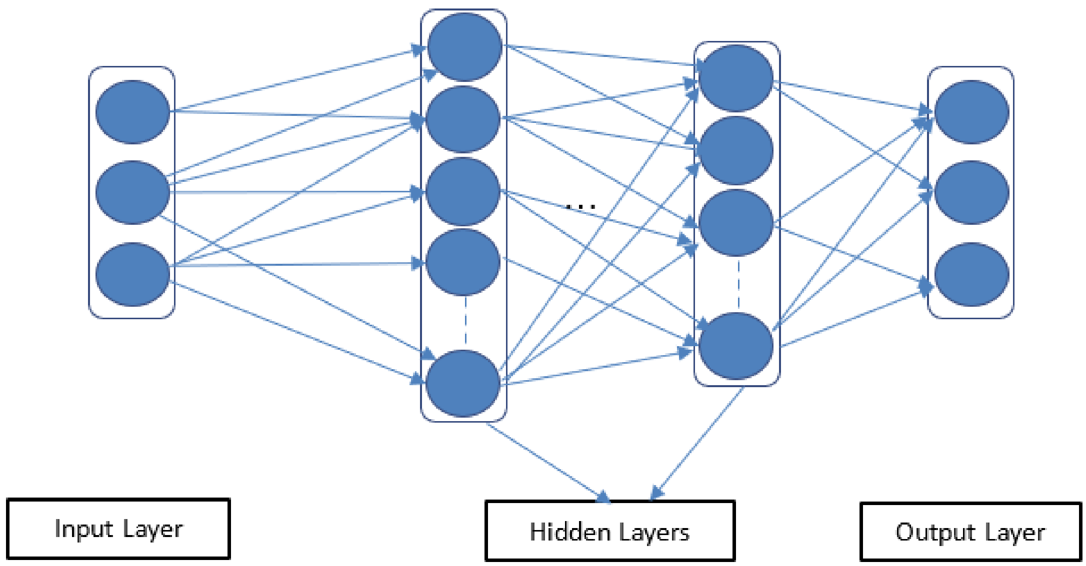


Figure 2-2: Typical Artificial Neural Networks (ANN) Layers.

2.4.4.2 Activation Function

The activation functions in artificial neural networks are needed to relay information in an interpretable format between interconnected layers. They help in transforming input signals received by a layer into a suitable output that will serve as an input for the next layer in the model assembly [47]. Nwankpa, Ijomah [48] described activation functions as functions that help to compute the weighted sum of input and biases to know if a neuron can be activated or not. Activation functions are known to generally improve the prediction accuracy of models and if they are not used, the output signal will be a simple linear function that is not capable of understanding and recognizing complex mappings from data [47]. Also, non-linear activation functions are needed in neural networks because neural networks are expected to be able to solve and interpret complex functions to achieve optimum performance.

According to Sharma, Sharma [47], there are different types of activation functions such as Binary step function, Linear, Sigmoid, Tanh, Rectified linear unit (ReLu), Softmax, etc. with different approaches for mapping between the input and output signals. The binary step function is a simple activation function that is used for a binary classification problem i.e. there are only two possible classes in the dataset represented as 0 or 1. Linear activation functions represent the signals using a simple linear function with a single gradient and the error in the model does not improve with training because of the constant gradient [47, 49]. The sigmoid activation is a more acceptable method for binary classification problems with an S-shaped function that runs from 0 to 1. They are used in the output layer of a binary classification problem and have also been applied in other logistic regression models [48]. Sigmoid functions are rarely used in the hidden layers of a neural network except for a shallow network [50].

Tanh function is like the sigmoid function but symmetric around the origin and the S-shaped function runs between -1 and 1. This helps in mapping the negative inputs to negative outputs and mapping inputs that are close to zero to outputs with similar characteristics [49]. The rectified linear unit (ReLu) activation is a more efficient activation function that only activates neurons that are in use while the others are deactivated. ReLu activations are widely adopted in the hidden layer of neural networks because of their efficiency in mapping linear combinations of inputs into non-linear outputs, which is very suitable for most neural network architecture to perform optimally in handling complex functions. The Softmax activation works like multiple sigmoid functions, and it is the most appropriate activation function for multiclass classification problems. They are often used as the activation function in the output layer of a multiclass classification model.

2.4.4.3 Tensorflow

TensorFlow owned by Google is an open-source deep learning software library for defining, training, and deploying machine learning models [51]. The multilayer model of the ANN has been simplified through a framework called TensorFlow which can help to easily specify the model parameters ranging from the number of layers and neurons, learning rate, activation function, etc. The process of computing weight and bias in the loss function, gradient descent, and other parameters is very complex especially when it has to be done over several iterations, TensorFlow provides a dashboard to implement this process in a computationally efficient way [46]. TensorFlow has a function called tf.gradients () that can automatically compute gradients and has different types of optimizers in its library that can be used to regulate the learning rate of the model [46]. Other available deep-learning frameworks include Theano, Torch, Caffe, etc. [46].

2.4.4.4 Optimizers

Determining the appropriate learning rate that regulates the weight (w) and bias (b) of the gradient descent in each iteration can be a very difficult task because choosing the wrong learning rate can cause the model to infinitely without reaching its optimum performance. The gradient descent gets stuck in a local minima irrespective of the change in the weight and bias. Equations 2.3 and 2.4 show how the weight and bias are updated in each iteration through a multiplication factor i.e., the learning rate (α).

$$\text{Weight, } w = w_o - \alpha.\left(\frac{\partial J(w,b)}{\partial w}\right) \tag{2.3}$$

$$\text{Bias, } b = b_o - \alpha.\left(\frac{\partial J(w,b)}{\partial b}\right) \tag{2.4}$$

Where,

$J(w,b)$ is the loss function.

w_o is the initialized weight carried on from the previous iteration.

b_o is the initialized bias carried on from the previous iteration.

Using optimizers helps to automatically tune the learning rate to help the deep learning model achieve optimum performance [52]. Although different optimizers occur in practice such as SGD, Adam Momentum, RMSProp, Nesterov, Adagrad, Nag, Adadelta, etc. no theory guides in making an appropriate comparison between the available optimizers [52, 53]. Schmidt, Schneider [53] posited that the performance of optimizers is task-dependent, and the analysis conducted shows that the Adam optimizer is one of the contenders for the best-performing optimizer across the majority of assigned tasks.

2.4.4.5 Regularization Parameters

One of the major problems with neural network models is overfitting which is a situation where the model performs well on the data it was trained on and reports a poor performance on a new dataset. Regularization allows the deep learning model to generalize well to unseen data even when training on a finite training set or with an imperfect optimization procedure [54]. This poor generalization on a new dataset is associated with the model trying to memorize the training data without a firm understanding of the different possible patterns. Using too many features in the neural network model has been associated as one of the major causes of overfitting. Some of the features can be polynomial and result in a higher-order function that the model tries to fit in. The use of the regularization parameter (λ) helps to suppress the effect of features with proclivity to cause overfitting in the model. The regularization parameter is only added in the computation of the weight during gradient descent and not the bias because it is a constant in the model evaluation function as shown in Equation 2.5. The method for computing the weight during the model training is now updated as shown in Equation 2.6 to suppress the effect of the features (x).

$$f\,(w,b) = wx + b \tag{2.5}$$

$$\text{Weight, } w = w_o - \alpha.\left(\frac{\partial J(w,b)}{\partial w} + \frac{\lambda}{m} wo\right) \tag{2.6}$$

m represents the number of training examples available for creating the model.

Some of the regularization techniques used in practice include dataset augmentation involving the addition of simulated data to the model and preventing the model from trying to fit into all the available data points, early stopping which involves stopping the model training after there is no more significant reduction in error and reduces the propensity for overfitting, bagging (or ensemble method) which involves averaging several models since all the models cannot make the same errors on the test set, dropout which involves the random deactivation of some neurons in the neural network layers and prevents the model from memorizing the training examples, Lasso regularization (or L1 norm) involves shrinking the model parameters towards zero, Ridge regularization (L2 norm) shrinks the weight to be very small without making them exactly zero, etc. [55].

2.5 Features Affecting Bridge Components Condition Identified from Literature

One of the most important steps in creating a machine-learning model that makes good predictions is feature selection. These are potential factors that influence the condition rating of the bridge components, and the created model can study their structure to make predictions on a new set of data. Miao [41] used two sensitivity analysis techniques which are Shapley value and sobol indices to determine the bridge features that have the greatest influence on the condition deterioration and observed a similar level of feature importance from the two techniques. Ali, Elsayegh [42] used engineering judgment to select 15 features from the NBI database for an ANN model and adopted linear correlation to validate the selected features. However, the resulting model gave a low coefficient of determination, R^2, for the deck, superstructure, and substructure with a value of 0.35, 0.47, and 0.37 respectively. This does not show any significant improvement in accuracy when compared with the linear model. The low accuracy level is strongly attributed to the features selection method, and the linear correlation method that was adopted by Ali, Elsayegh [42] to validate the feature selection is not appropriate for discrete data types like the components condition rating. Liu, Nehme [56] depended on the auto-feature mining characteristics of the deep learning framework and used a total of 23 features that cut across geographical location indicators, bridge attributes, structure configurations, and climatic factors for making a convolutional neural network (CNN) model. Liu, Nehme [56] observed an average prediction error of 56% for condition rating data from 1993 to 2019, which seems high. Although this result was better than that of the Markov chain-based percentage prediction method (PPM), expected-value method, Bayesian approach, and MUSTEM model, the low accuracy further reinforces the need for a more appropriate feature selection procedure.

Liu and Zhang [57] used different combinations of features to generate deterioration models for the deck, superstructure, and substructure of bridges in Maryland and Delaware and observed a variation in the performance of these models signifying that all the features are not on the same level of importance to the deterioration of the bridge components. Zhu and Wang [17] in their work proposed the ReliefF algorithm as a method that can be used in feature selection based on the assigned weight by using the correlation between the features and the rank in the ReliefF algorithm as the determinant. The aim was to remove the subjectivity in engineering judgment used when selecting features for the machine learning model. The algorithm was able to select the right set of features, and a combination of recurrent neural network (RNN) and convolutional neural network (CNN) was used to make forecasts for 3 to 4 years. However, no individual prediction was made for the bridge components; rather, the authors adopted the overall condition rating of the bridge as the response variable which does not help the maintenance agencies know which component needs urgent rehabilitation.

Winn and Burgueño [16] used a combination of linear correlation, chi-square, and engineering judgment to make a preliminary selection of features and then used the trial-and-error method to select the optimum set of features. This is done by creating a multi-layer perceptron (MLP) model for different sets of nominated features and the set of features with the best predictive property was selected. However, the trial and error method does not guarantee that those selected are the best set of features because other combinations could give a better result. Nguyen and Dinh [13] adopted seven of the features discovered by Winn and Burgueño [16] in their ANN deterioration for bridge deck and cited the uncertainty in the NBI data as the reason for dropping some of the features used by Winn and Burgueño [16]. Nguyen and Dinh [13] added the average daily traffic (ADT) growth rate to the initially selected seven features to construct an ANN model that gives the maximum prediction accuracy of 73.6%. The eight features that were ultimately adopted are bridge age, ADT, design load, structure type, approach design type, number of spans, percent truck traffic, and ADT growth rate. However, Nguyen and Dinh [13] constructed the model more like linear regression and did not predict the class of the deck condition. This is done by assuming that the condition ratings are continuous, whereas they are discrete and can only assume a whole number value between 0 and 9. A prediction of 6.51 was approximated by Nguyen and Dinh [13] to be a condition rating of 7.

Huang [15] developed an ANN model to predict the condition of concrete bridge decks in Wisconsin, USA. Huang [15] used the Analysis of Variance (ANOVA) method to select features that influence the condition of the bridge deck, and 8 features were identified at the end of this process, which resulted in a model with a prediction accuracy of 75%. However, more emphasis was placed on the age of the bridge when it deteriorates from one state to another, and other features were selected based on the level of relationship associated with it. Hasan and Elwakil [18] developed a regression model to predict the condition of the prestressed concrete bridge deck. The authors conducted the best subset analysis to identify features that influence the deck condition rating the most by pegging the minimum coefficient of determination, R^2, at 70%. Features that resulted in a lower R^2 for any iteration were categorized as insignificant and then omitted from the analysis. The optimum model at the end of this analysis gave a coefficient of determination of 80%. Like many other papers previously cited, this model treats the deck condition rating like a continuous data type when in reality, ratings are discrete values.

Assaad and El-adaway [14] adopted the Boruta algorithm, which was based on a wrapper method, in selecting features for their bridge deck deterioration model. This algorithm can select the relevant features out of a long list irrespective of the complex and non-linear relationship with the output variable. Assaad and El-adaway [14] created an Artificial Neural Network (ANN) and K-nearest Neighbor (KNN) with the nominated features and then discovered that the ANN model produced better prediction accuracy of the deck. However, an unbalanced dataset was observed in the output prediction of the model where only a few bridges are in condition ratings of 2 and 3. This could lead to low prediction accuracy on a new set of data for bridges in a critical state. Moomen, Qiao [7] nominated features for the deterioration model based on the long-term study of the effect of the features on the deterioration of the bridge component. Kong, Li [6] adopted the Shapley additive explanation (SHAP) to investigate the association between various factors and bridge deck deterioration. The authors created an XGBoost model that can identify young bridges (less than 20 years) with poor or failing deck condition and old bridges (30 – 40 years) with good condition. This is aimed at identifying factors that influence the quick deterioration of the bridge deck regardless of the age of the bridge.

Chang, Maguire [58] developed a framework to mitigate human bias in feature selection during the development of deterioration models for bridge components using the least absolute shrinkage and selection operator (LASSO) which is a penalized regression and covariance analysis. The process is aimed at removing redundant features that can occur in the form of repetition or contribute nothing to the informative description of the dataset. The authors used the features selected by LASSO to develop a deterministic deterioration model for the bridge components. Moomen and Siddiqui [59] in developing a probabilistic deterioration model for bridge components adopted a marginal effect technique that quantifies the effect of changes in the explanatory variables (bridge features) on the response variable (component condition rating). The effect of each of the explanatory variables on the response variable is observed while the other explanatory variables are held constant. Moomen, Qiao [7] developed probabilistic deterioration models for bridge components in the state of Indiana using a binary probit approach that was based on the LIMDEP platform. The significance of the bridge features to the components' condition rating was estimated using a hypothesis test at a 5% significance level. The continuous explanatory variables highlighted by these papers were elicited for further analysis in this research project.

Table 2-2, Table 2-3, Table 2-4, and Table 2-5 show the list of features selected by several papers using different analytical methods for the bridge components deterioration model. This shows that there is no standard list of features that are generally accepted as the factors influencing the deterioration of the bridge components the most. Combining all the features and using them in a deterioration model greatly increases the dimensionality of the data and does not guarantee greater prediction accuracy. Liu, Nehme [56] used all the features on the NBI database which resulted in a high prediction error of 50% with a CNN model.

Table 2-2: Deck Features Identified from Literature 1.

Huang [15], ANOVA	Winn and Burgueño [16], Trial & Error	Moomen, Qiao [7], Long-term study of effect	Assaad and El-adaway [14], Boruta Algorithm
District	Age	Age	Deck width
Design load	Year built	Highway class	Bridge age
Average daily traffic (ADT)	Average daily traffic (ADT)	Service under bridge	Structural length
Environment	Percent truck traffic	Number of freeze-thaw cycles	Average daily traffic
Skew	Average daily truck traffic (ADTT)	Freeze Index	Maximum span
Deck length	Number of spans	Average daily truck traffic	Operation rating
Deck area	Region	Number of spans	Inventory rating

Number of spans	Steel reinforcement protection	Skew
	Structure Type	Bridge length
	Design load	
	Approach surface type	

Table 2-3: Deck Features Identified from Literature 2.

Hasan and Elwakil [18], Iterative Regression	Zhu and Wang [17], ReliefF Algorithms	Kong, Li [6], SHAP
Skew	Latitude	Structure width
Maximum span length	Longitude	Average daily traffic
Structural length	Lanes on structure	Number of snowfall days
Roadway width	Skew	Max length of span
Deck width	Design/Material Type	Structure length
Inspection frequency	Number of spans	Material type
Percent truck traffic	Type of Wearing Surface	Percent truck

		traffic
	Average daily truck traffic	Number of spans
	Future average daily traffic	Design type
	Bridge condition in the previous year	
	Age	
	Age from reconstruction	

Table 2-4: Superstructure Features Identified from Literature.

Chang, Maguire [58], LASSO	Moomen, Qiao [7], Statistical Significance	Moomen and Siddiqui [59], Marginal Effects	Zhu and Wang [17], ReliefF Algorithms
Age	Age	Age	Lanes on structure
Bridge roadway width	Number of freeze-thaw cycle	Number of annual precipitation days	Skew
Length of maximum span		Average daily traffic	Number of spans
Structure length			Average daily truck traffic (ADTT)

Average temperature			Future average daily traffic
Total Precipitation			Age
Number of spans			
Lanes on structure			
Lanes under structure			
Skew			

Table 2-5: Substructure Features Identified from Literature.

Chang, Maguire [58], LASSO	Moomen, Qiao [7], Statistical Significance	Moomen and Siddiqui [59], Marginal Effects	Zhu and Wang [17], ReliefF Algorithms
Age	Age	Age	Lanes on structure
Bridge roadway width	Number of freeze-thaw cycle	Number of annual precipitation days	Skew
Average daily truck traffic		Total precipitation	Number of spans

(ADTT)	
Structure length	Average daily truck traffic (ADTT)
Average temperature	Future average daily traffic
Total Precipitation	Age
Number of spans	
Lanes under structure	
Skew	

2.6 Principal Component Analysis (PCA)

PCA is a statistical technique used to explain the variation in a dataset by using uncorrelated linear combinations called principal components. It has been used in different applications to suppress or remove features with low amounts of variation since they do not give much information about the dataset. PCA extracts the most important information from a dataset which makes it a tool for dimensionality reduction and easier data interpretation [60]. PCA projects the original data into other axes while retaining most of the variation in the dataset. Wetherell, Costamagna [61] showcased the importance of PCA in Figure 2-3 below. The original data points in red color consist of horizontal and vertical coordinates which can be projected to a new axis PC1. This resulted in a single coordinate on the PC1 axis while also giving a clearer representation of the variation between the data points. However, if the points were to be projected on axis PC2, the data points would be clustered around a region and not much of the variation in the data would be explained by this component. This explains why different components explain different amounts of variation in the dataset. The first principal component explains the largest variation in the dataset and decreases progressively until the last principal component. The number of principal components that could be generated is as much as the number of features in the dataset that are orthogonal to each other. This orthogonality means that the principal components are independent of each other.

PCA is a very useful tool in feature selection for creating a model because the principal components are influenced by the original features to different degrees to explain the variation in the data. Selecting a few principal components that sufficiently explain the variation in the dataset to replace the original set of features is equivalent to selecting the features that have more information about the dataset and removing redundancies. PCA has been used in different fields such as engineering, agriculture, health, commerce, etc. to determine the optimum set of features for their analytical models. Song, Guo [62] applied PCA for feature selection in a face recognition model and observed an increase in the classification accuracy when the right set of features was used while also reducing the dimensionality of the samples. Lasisi and Attoh-Okine [63] applied principal component analysis to reduce the dimensionality of the track geometry parameter and built a machine-learning model for predicting defects in tracks. It was observed by the authors that three principal components are more efficient in predicting defects in tracks compared to using the whole track quality index.

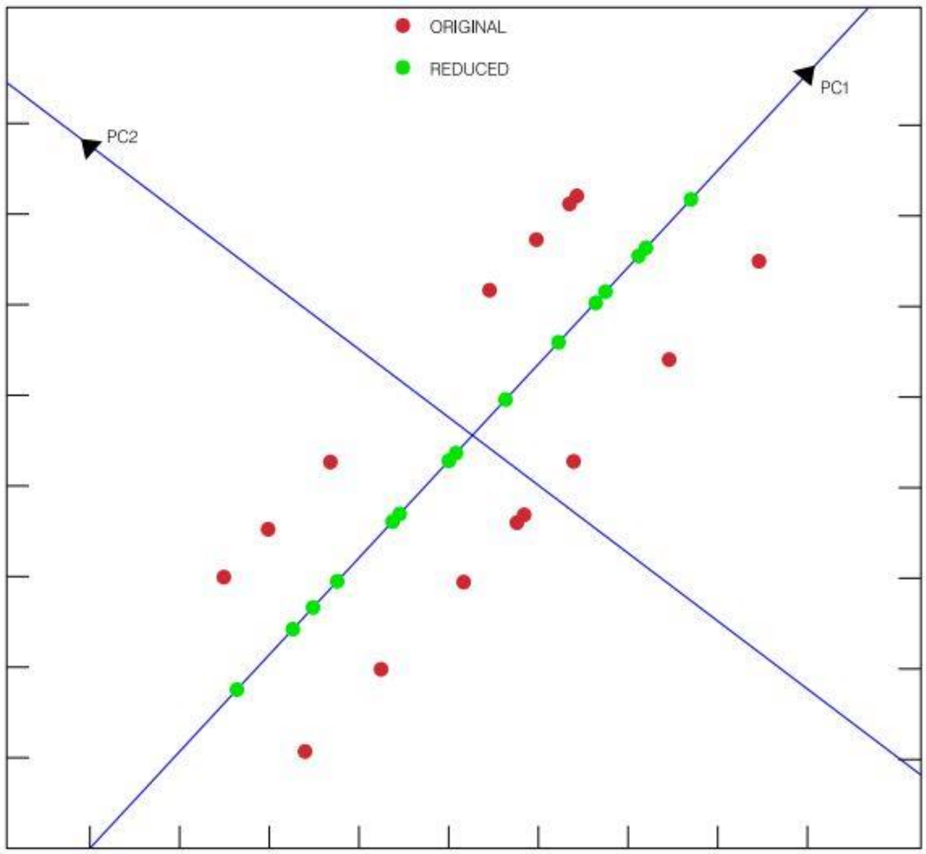


Figure 2-3: Illustration of Principal Component Analysis [61].

The choice of the number of principal components to select depends on the purpose of the analysis to be conducted [64]. To select the number of principal components to adopt in some cases, a scree plot is used to select as many principal components as possible until the curve flattens out [60]. Figure 2-4 shows a sample scree plot where most of the variation in the dataset is explained by the first four principal components as signified by the inertia (measure of variance) after which there was no significant increase in the amount of variation explained. In other cases, the number of principal components is selected based on the quality of data available, which is expressed in terms of the amount of noise in the dataset [64]. In a case where the data has a large quantity of noise, choosing a number of principal components that explain 90 to 95% of the data will amount to modeling the noise, as such, a smaller number of components is usually recommended [64]. It is not uncommon to see instances where principal components are selected based on eigenvalues because eigenvalues are mostly ordered from the highest to lowest corresponding to each of the components. Principal components with eigenvalues less than 1 or less than the overall average could be dropped and tagged as insignificant.

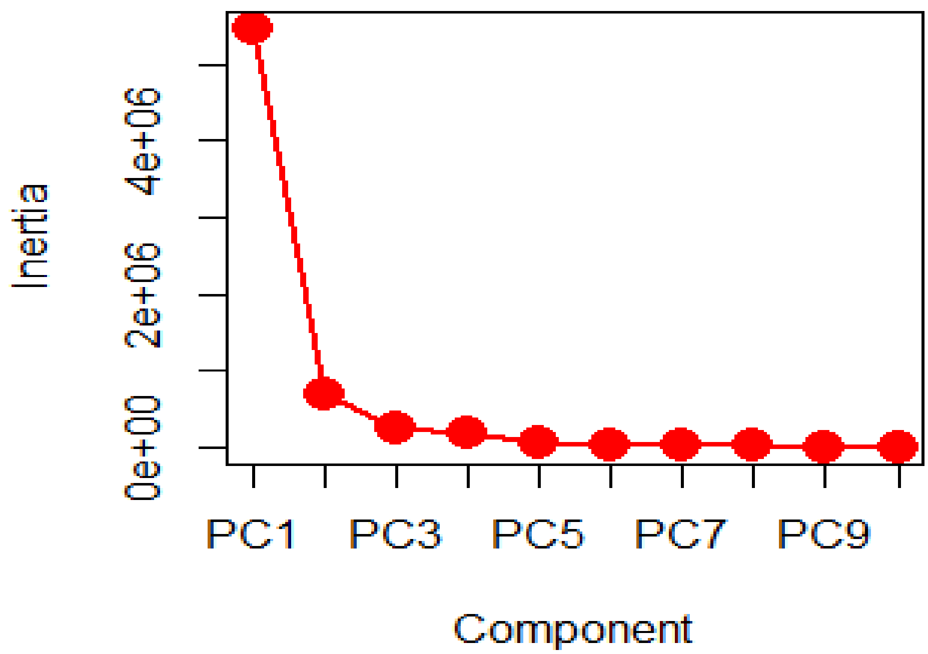


Figure 2-4: Sample Scree plot for PCA generated in R-programming language.

Like many other statistical techniques, PCA has its limitations in the sense that it only re-expresses the data as a linear combination and neglects other more complex relationships [65]. It is also worth noting that PCA cannot be applied to categorical variables since they do not have a variance structure that is needed by the principal components.

Omuya, Okeyo [66] developed a hybrid feature selection model, which uses both PCA and information gain. The features selected at the end of this process were used to make a classification machine learning model. Omuya, Okeyo [66] observed a large reduction in training time and better prediction accuracy when compared to the original dataset. Zhang [67] applied a varying number of principal components in their malware detection model using ANN. The author observed that selecting 32 principal components produced the same level of accuracy as 48 principal components while also producing a 33% reduction in dimensionality and a 22% reduction in training time. This further showcases the importance of PCA in removing redundancies from a dataset and even generates better model accuracies in cases where the less important features constitute noise in the data. Yuce, Mastrocinque [68] similarly used PCA for selecting features in an ANN model for detecting defects in wood. This model achieved an 18% increase in accuracy while using 61 fewer epochs, which translated to a reduction in the training time. Generally, PCA can help to remove features with low variation, remove multicollinearity in the features of a model, suppress noise in the data, and speed up model training.

By synthesizing the ways PCA has been implemented by all the literature cited, it is evident that it can serve as a good technique for capturing important information or variance in a dataset that is required for developing a good deterioration model. It can also eliminate the need for complex model architecture since good performance is guaranteed without using all the features and saves some computing effort.

2.7 Random Forest Algorithm

Random forest is an effective classification algorithm that uses an ensemble of decision trees and bagging techniques to make predictions. It is also very efficient in handling data with a high number of variables, large noise content, and missing data without affecting the stability of the resulting model [69]. A good level of performance has been achieved with this algorithm because of its low propensity for overfitting and easy result interpretation. The decision tree works by splitting the dataset based on the available features and evaluates the purity of the groups after the split. Each member of a group is expected to be as similar as possible and dissimilar from the members of other groups. The features that result in the highest amount of purity after splitting are regarded as the most important features. In an ensemble (group) of decision trees, the overall importance of the features can be evaluated by averaging the importance across all the trees. It is worth noting that the individual decision trees are uncorrelated and reduce the likelihood of making biased decisions.

In each of the decision trees making up the random forest algorithm, each of the features (nodes) is used to split the dataset into two groups and the decrease in impurity for each split is computed using Equation 2.7. The amount of decreased impurity derived from using each of the features is combined using Equation 2.8 to derive the overall feature importance in the decision tree and the resulting value is normalized to a value between 0 and 1 using Equation 2.9. The overall importance of each feature using the random forest algorithm is derived from averaging over all the trees as illustrated in Equation 2.10 [70]. It is worth noting that the number of trees in a random forest algorithm is typically between 64 and 128 to achieve optimum performance and good processing time [71].

Nodal Reduction in Impurity (Ni) = H (P_i^{root}) – (w^{left} * H (P_i^{left}) + w^{right} * H (P_i^{right})) (2.7)

Where, H (P_i^{root}) is the measure of impurity at the initial stage before the split.

w^{left} is the fraction of the total sample in group a.

H (P_i^{left}) is the measure of impurity in group a.

w^{right} is the fraction of the total sample in group b.

H (P_i^{right}) is the measure of impurity in group b.

$$\text{Feature Importance (f)} = \frac{\sum Ni\ for\ nodes\ that\ are\ splitted\ using\ feature\ i}{\sum N\ for\ all\ nodes} \quad (2.8)$$

$$\text{Normalized Feature Importance (nf)} = \frac{f}{\sum f} \quad (2.9)$$

$$\text{Overall Feature Importance (F)} = \frac{\sum nf}{Total\ number\ of\ trees} \quad (2.10)$$

Another method to determine which feature to split on is called the 'Gini Index' and is widely used in machine learning libraries like Scikit-Learn because of its computational efficiency. The Gini Index (gini) computes the probability of getting a misclassification when a sample is drawn randomly, and it is mathematically illustrated in Equation 2.11 [72]. The lower the Gini Index, the lower the likelihood of misclassification, and the feature importance approach follows the same procedure as illustrated in equations 3, 4, and 5.

$$\text{gini} = 1 - \sum_{i=1}^{j} \mathrm{P}(i)^2 \quad (2.11)$$

where j is the number of classes in the dataset

P(i) is the probability of selecting each class when drawn randomly.

This algorithm has seen some level of application in bridge engineering for both prediction and feature selection purposes. Alipour, Harris [73] used a combination of decision trees and random forest to determine the load rating capacity of bridges and were able to rapidly determine which set of bridges required load restriction or load removal. Soleimani [74] applied random forest to determine the importance of some bridge portfolios or features to the seismic response of the bridge. The analysis was performed on different bridge design types and bridges with special characteristics like taller piers, and the random forest was able to highlight the major parameters that influence the seismic response of the bridge. Chun, Ujike [75] used the random forest to evaluate the extent of internal damage due to rebar corrosion in reinforced concrete. This model was tested on an actual bridge and was able to detect internal damage that was not obvious on the exterior surface. Zhang [67] developed an economical damage detection model for old short-span arc bridges in rural areas using the random forest algorithm. The algorithm takes the vehicle-induced excitation as an input and can determine the damage index on the bridge.

In the case of this research, the bridge elements are the features because they are first inspected to help in assigning condition ratings to the major components. As such, the condition rating of the bridge components is dependent on the condition of the associated bridge elements.

2.8 Knowledge Gaps and Relevance of Research

From the literature review conducted, the method for the determination of the importance (or weight) of bridge elements to the overall condition of the bridge adopted by the DOTs and other researchers is not based on the actual inspection data that describes the condition of the different bridge parts but rather human judgment that is subjected to bias and error in assumptions. Moreover, the methods that were previously adopted in assigning importance to bridge elements did not consider the different bridge design types that could occur in reality and the fact that the importance of the elements is dependent on the structural configuration of the bridge. The literature review also shows that there is a need to synthesize the component and element-level data to help validate the bridge inspection process and the maintenance, repair, and replacement (MRR) schedule of the DOTs.

Furthermore, in various attempts by researchers to forecast the future condition of bridge components, the literature review conducted shows that there is no generally acceptable list of features that certainly influence the condition of the bridge components, different techniques adopted by the previous researchers give a different list of most influential bridge features. Also, it was observed that most of the machine learning prediction models that were previously developed did not adopt the right technique that is suitable for the nature of the bridge components data. The models have been more like linear regression models and did not predict the discretized value of the components' condition rating. A prediction of 6.51 was approximated to be a condition rating of 7, which is inappropriate for a discrete data type.

As such, this research intends to determine the importance of the bridge elements using a data-driven approach by adopting the random forest algorithm to synthesize the different bridge data and come up with the overall bridge health index (BHI) equation for the different bridge types. Also, in predicting the future condition of bridge components this research intends to apply the principal component analysis

(PCA) technique to extract the important information in the dataset and develop a more suitable prediction model that is more appropriate for the nature of available component data.

Chapter 3

METHODOLOGY

3.1 Description of Research Approach/Framework

The activity in this research can be divided into two main distinct objectives namely: (a) Developing bridge component condition prediction model. (b) Determining bridge elements weight and developing bridge health index (BHI) equations. Figure 3-1 below shows a pictorial representation of the framework to be followed to achieve the stated objectives.

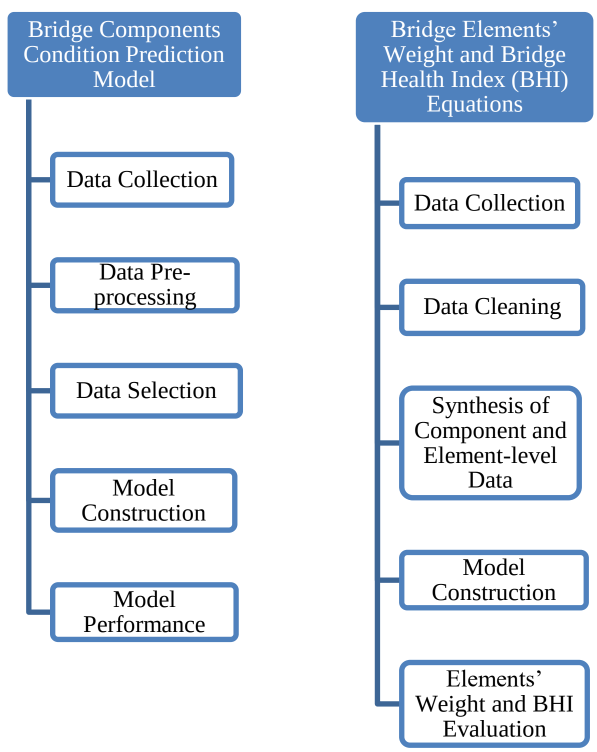


Figure 3-1: Research Framework.

3.2 Bridge Data Handling

The approach adopted as related to getting the bridge data ready for analysis is discussed in the following sections.

3.2.1 Data Collection Procedure

Large amounts of data were collected from different sources to achieve the itemized goals of this research and it is important to discuss the associated process.

3.2.1.1 Data Collection for Prediction Model

The bridge data for five states in Region 3 as per USDOT categorization (Delaware, Maryland, Pennsylvania, Virginia, and West Virginia) from 1992 to 2022 were downloaded from the info-bridge Long Term Bridge Performance (LTBP) web portal after selecting the appropriate bridge features that will be used in the analysis. The bridge features used by the authors listed in Table 2-2, Table 2-3, Table 2-4, and Table 2-5 are a combination of discrete and continuous data types e.g., the design and material type are categorical data types while features like the operating rating, inventory rating, traffic volume, etc. are continuous. Some papers that have developed deterioration models for direct application by DOTs e.g., Moomen, Qiao [7] have regarded these factors as grouping factors and would normally develop multiple deterioration models for all the sub-groups. This means that if we have six material types (i.e., steel, concrete, prestressed concrete, steel continuous, concrete continuous, and prestressed concrete continuous) six models will have to be created. To move away from this approach, the deterioration model was developed to be robust and should be able to be applied irrespective of the design type, material type, and other groupings. Moreover, some features like the deck area are redundant because it is a factor of the deck width and deck length. A combination of all three features will lead to multicollinearity in the dataset, which is a situation where two or more features are proportional to each other and affect the model performance. Features like the longitude and latitude were removed because it was observed from the data analysis that most of the values are zero (0) and further inquiry shows that bridges that are over water have their longitude and latitude recorded as zero. The zeros could also be attributed to incomplete bridge data collection. Table 3-1, Table 3-2, and Table 3-3 show the aggregation of features that were identified from the literature review conducted for the identification of prevalent factors influencing the deterioration of the bridge deck, superstructure, and substructure respectively [6, 7, 13-18, 58, 59]. Data for culverts were excluded from the download because they are not explicitly separated into deck, superstructure, and substructure. The portal has an interactive display that helps in making proper selections and allows a lot of flexibility in the data that needs to be downloaded. The condition rating data of the bridge deck, superstructure, and substructure are collected through visual inspection by bridge inspectors, and this makes the data collected highly subjective. To minimize the effects of this subjectivity on the performance of the deterioration model, the data selected was restricted to only 2 states with enough data distribution among the possible condition rating values. An analysis will be conducted to determine the two states with enough distribution for each of the three bridge components. It was assumed that bridge inspectors in a particular state go through the same mode of training and the data they collect go through the same quality assurance and quality control framework, making the disparity minimal. Combining data from multiple states will worsen the variation in the data and make it difficult for the deterioration model to learn from these extremely varied cases.

Table 3-1: Bridge features downloaded for deck deterioration Model.

S/N	Bridge features
1	Average daily traffic
2	Number of spans in main unit
3	Structure length
4	Bridge age
5	Skew
6	Length of maximum span
7	Deck width
8	Operating rating
9	Inventory rating
10	Lanes on the structure
11	Future average daily traffic
12	Average daily truck traffic
13	Number of freeze-thaw cycles

14	Number of snowfall days

Table 3-2: Bridge features downloaded for superstructure deterioration Model.

S/N	Bridge features
1	Bridge roadway width
2	Number of spans in main unit
3	Structure length
4	Bridge age
5	Skew
6	Length of maximum span
7	Lanes under structure
8	Average temperature
9	Total precipitation
10	Lanes on the structure
11	Future average daily traffic
12	Average daily truck traffic
13	Number of freeze-thaw cycles

14	Average daily traffic
15	Number of days with measurable precipitation

Table 3-3: Bridge features downloaded for substructure deterioration Model.

S/N	Bridge features
1	Bridge roadway width
2	Number of spans in main unit
3	Structure length
4	Bridge age
5	Skew
6	Length of maximum span
7	Lanes under structure
8	Average temperature
9	Total precipitation
10	Lanes on the structure
11	Future average daily traffic
12	Average daily truck traffic
13	Number of freeze-thaw cycles

14	Number of days with measurable precipitation

3.2.1.2 Data Collection for Bridge Elements' Weight Model

In this research, bridges in the Region 3 (U.S. DOT categorization) states namely Delaware, Maryland, Pennsylvania, Virginia, and West Virginia were considered in the analysis. The long-term bridge performance (LTBP) web portal was used to identify bridges in these states with valid element-level data. This process is important because the collection of the element-level data was only mandated in 2014 and has not been fully implemented in all the bridges in these states. After identifying these bridges, the general condition rating data for the deck, superstructure, and substructure were downloaded from the LTBP portal for the years 2015 to 2022. Included with the components' condition rating data are the overall condition rating of the bridges and other bridge-specific features like the structural number and the main span design type. Also, with the help of the interactive interface of the LTBP web portal, culverts were filtered out of the download since they do not have a distinct deck, superstructure, or substructure. However, on the LTBP web portal, the element-level data for these bridges were only presented in a visual format and cannot be downloaded. A separate Federal Highway Administration (FHWA) web portal was utilized to get the element-level data for bridges in each state from 2015 to 2022.

3.2.2 Data Pre-Processing

3.2.2.1 Data Cleaning for Prediction Model

The NBI condition rating for the deck, superstructure, and substructure was checked for missing values which either appear as 'N' or the space is empty. This could be attributed to the reconstruction period of the bridges or some bridge design types that merge two components. An example of this case appears in some slab bridges where the deck and superstructure are given a single rating because they are built monolithically. Bridge samples with any of these cases were removed from the analysis. It was observed that most of the bridge components have condition ratings of 4 to 8 while condition ratings of 0, 1, 2, 3, and 9 only represent a small percentage of the total bridge sample. The case of low representation for condition ratings 0, 1, 2, and 3 can be attributed to the fact that the DOTs mostly do not allow the bridge components to deteriorate to this level before carrying out rehabilitation efforts. Condition rating 9 also has a low representation because it represents bridges in perfect condition, which can only be found in a few new bridges. Most bridges are already in existence and new bridges are not built as often, making the number 9 condition rating have low representation in the dataset. Thus, bridge components in condition ratings 0, 1, 2, and 3 were merged to condition rating 4 and those in condition rating 9 were merged with 8. The data from other states was not added to supplement the condition ratings with low frequency of occurrence because the ratings assigned are subjective to the inspectors. As such, having a high level of variability in the judgment of the inspectors on the condition rating of the components will significantly affect the prediction accuracy.

Also, the age of the reconstructed bridges was recomputed to show a new age count and not continue from the initially deteriorated structure. The data frame was restructured to include only the features to be used in developing the deterioration models while other irrelevant data columns like the structural number and state name automatically downloaded with the dataset were deleted. Finally, all the bridge features downloaded were checked for invalid and missing values. The length of maximum span, deck width, and the number of spans in any bridge are not meant to be 0. These situations could be a result of input error, and such bridges were removed from the analysis.

3.2.2.2 Data Scaling for Prediction Model

The different bridge features take on a wide range of possible values and it becomes essential to scale data to help the models' gradient descent run faster and reach the local minima earlier. There are different types of feature scaling which include the Min-max method, Mean method, Z-score method, etc. The min-max method of scaling a feature involves subtracting the minimum value from each of the data points and dividing the result by the difference between the minimum and maximum value as illustrated in Equation 3.1. The mean method involves subtracting the mean value from each of the data points and dividing the result by the difference between the minimum and maximum value as illustrated in Equation 3.2. The z-score method involves subtracting the mean value from each of the data points and dividing the result by the standard deviation as illustrated in Equation 3.3. It is worth noting that the standard scaler module of the scikit-learn library which will later be utilized in this research uses the z-score scaling method.

$$X_{scaled} = \frac{X - Xmin}{Xmax - Xmin} \tag{3.1}$$

$$X_{scaled} = \frac{X - \mu}{Xmax - Xmin} \tag{3.2}$$

$$X_{scaled} = \frac{X - \mu}{\delta} \tag{3.3}$$

Where μ is the mean value and δ is the standard deviation.

3.2.2.3 Data Cleaning and Synthesis for Bridge Elements' Weight Model

The general condition rating data of the bridges were inspected for missing or invalid condition rating values in the three major components. In some instances, a value of 'N' was observed in place of an expected condition rating between 0 and 9, this could be a result of current rehabilitation on the bridge component or a specific type of bridge design where two components are built monolithically and are rated as a single unit. This was observed in some slab bridges where the deck and superstructure are assigned a single condition rating. As such, bridge data with these conditions or with completely missing condition rating values were removed from the analysis.

The health index of all the bridge elements was computed using the weight factor ratio 1:0.4:0.1:0 and added as a column in the final because the element-level data only show the number of elements in each condition state. Since the element-level data of the bridges and the general condition rating data are downloaded separately, it becomes important to match these data sets into a single unit. The structural number which is a unique identifier for each bridge was used to connect the data sets and reformatted as shown in Table 3-4. The table was filled with the health index value and condition rating value for each corresponding bridge element and major component. The structural number, main span design, and overall condition rating of the bridges were also included in the data table. It is worth noting that in cases where a bridge has more than one type of an element e.g., a bridge having both steel and reinforced concrete rail or a bridge having different joint types, the effective element health index that is selected in the analysis is the lowest of the available health index because this represents the most critical health index condition.

Table 3-4: Bridge data format.

Structural Number	Main Span Design	Element (1) Health Index	…	Element (n) Health Index	Deck rating	Superstructure rating	Substructure rating	Overall Rating
Bridge 1								
Bridge 2								
:								
Bridge n								

3.2.3 Data Selection

3.2.3.1 Bridge Data Partitioning for Elements' Weight

As it is already noted, different bridge design types have separate sets of elements that are important for their structural performance. The resulting dataset in Table 3-4 was divided into groups based on the main span design type of the bridges; 15 groups were derived as shown in Table 3-5 to represent most of the bridge span design types found in the states considered in the analysis. The bridge elements were nested under the associated major component, Figure 3-2, Figure 3-3, Figure 3-4, and Figure 3-5 show a general hierarchy of the bridge parts to depict the bridge component and the elements connected with them. The hierarchy was adapted from the Federal Highway Administration (FHWA) guideline for bridge elements and modified to show the connection between the major components and the element irrespective of the material type of the bridge [5]. Based on this hierarchy, the dataset for each main span design group was divided such that only the elements associated with a given major component appear in the data frame as shown in

Table 3-6. For each main span design group, three of these data frames were created to represent the deck, superstructure, and substructure and the bridge elements associated with them. Also, to determine the importance of each of the three major components to the overall condition of the bridge, the data frame shown in Table 3-7 was set up for each of the main span design type groups.

Table 3-5: Bridge categories in the analysis.

Main Span Design Types	Notation in Analysis	Number of Observations
Stringer/Multi-beam or Girder	SMG	55878
Box Beam or Girders (Single)	BBGS	19864
Box Beam or Girders (Multiple)	BBGM	13371
Tee Beam	TBM	10254
Girder and Floor beam System	GFS	2549
Frame	FRM	1075
Truss-Thru	TRT	1175
Channel Beam	CBM	677
Arch-Deck	ARD	533
Truss-Deck	TRD	277
Movable Bascule	MVB	83
Arch-Thru	ART	112
Segmental Box Girder	SBG	37
Suspension	SPS	59
Stayed Girder	STG	16

Table 3-6: The data frame format for each major component and the associated elements.

Structural Number	Element (1) Health Index	…….	.. Element (n) Health Index	Major Component Condition Rating
Bridge 1				
Bridge 2	SAMPLE			
:				
Bridge n				

Table 3-7: The data frame format for components and overall rating.

Structural Number	Deck Rating	Superstructure Rating	Substructure Rating	Overall Rating
Bridge 1		SAMPLE		
Bridge 2				
:				
Bridge n				

To avoid leaving out the wearing surface and bearing from the bridge elements' weight analysis, they were nested under the Deck and Superstructure respectively. Although this new arrangement is different from the Federal Highway Administration

(FHWA) hierarchy in Figure 1-1, the reason for this adjustment can be justified by the bridge inspection manual. It states that for situations where the surface of the deck is not visible due to the presence of a wearing surface, the condition of the Deck is assessed based on the observed crack or settlement in the wearing surface. This affirms the relationship between the Deck condition rating and the wearing surface. Also, for the inspection of the superstructure, the bridge inspection manual states that the problem areas to be inspected include the location of maximum moment, maximum shear, bearing, section change, and connection. This statement affirms the relationship between the Superstructure condition rating and the bearing.

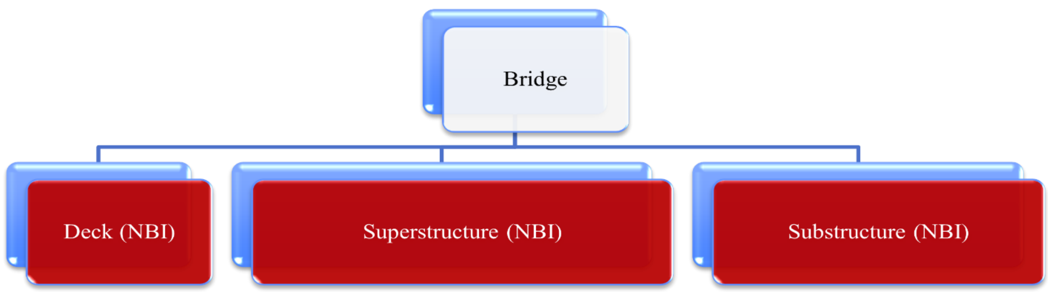


Figure 3-2: Bridge Components Hierarchy.

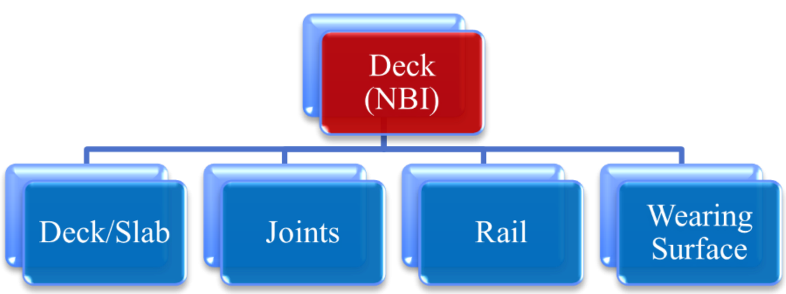


Figure 3-3: Deck Hierarchy.

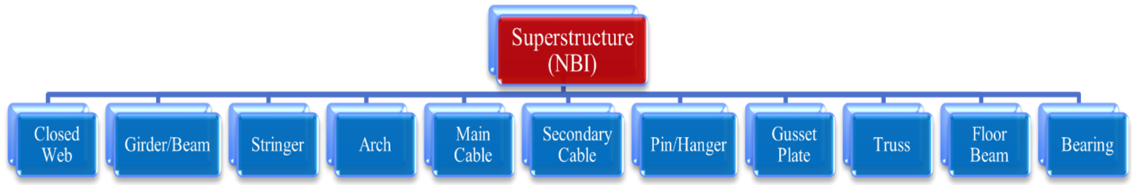


Figure 3-4: Superstructure Hierarchy

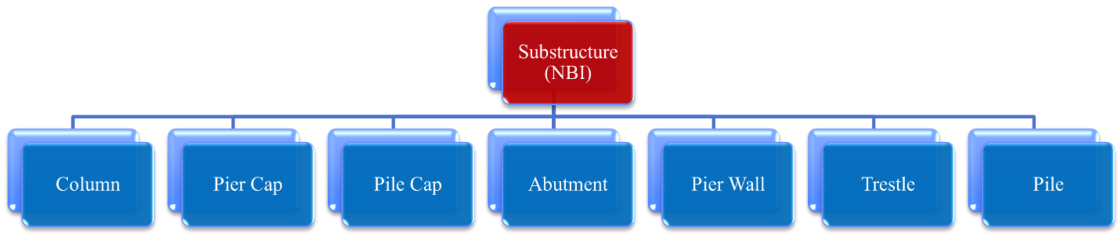


Figure 3-5: Substructure Hierarchy

3.2.3.2 Deck Data for Model Development

A preliminary analysis was conducted on the deck condition rating data collected for the 5 states to determine how the condition rating values are distributed. It was observed that a combination of deck condition rating data in Maryland and Virginia gives a good distribution for the creation of an optimum model. A total of 53,000 bridge samples were used and it was ensured that there was almost equal representation of each condition state in the dataset. This represents an average of 10600 bridge samples in each condition state, and these bridges were drawn randomly from the total dataset. However, during the preliminary training and testing of the models, it was observed that most of the misclassifications occurred for condition ratings 5, 6, and 7. Thus, the number of bridge samples used is 10000, 11000, 11000, 11000, and 10000 for condition ratings 4, 5, 6, 7, and 8 respectively.

3.2.3.3 Superstructure Data for Model Development

The preliminary analysis was also conducted for the superstructure and like the deck, it was observed that a combination of superstructure condition rating data in Maryland and Virginia gives a good distribution for the creation of optimum model. A total of 63,000 bridge samples were used and it was ensured that there was almost equal representation of each condition state in the dataset. This represents an average of 12600 bridge samples in each condition state, and these bridges were drawn randomly from the total dataset. However, during the preliminary training and testing of the models, it was observed that most of the misclassifications occurred for condition ratings 6 and 7. Thus, the number of bridge samples used is 12000, 12000, 13500, 13500, and 12000 for condition ratings 4, 5, 6, 7, and 8 respectively.

3.2.3.4 Substructure Data for Model Development

The preliminary analysis conducted on the substructure condition rating data collected for the 5 states shows that a combination of substructure data for Maryland and West Virginia gives a good distribution for the creation of an optimum model. A total of 69,000 bridge samples were used and it was ensured that there was almost equal representation of each condition state in the dataset. This represents an average of 13800 bridge samples in each condition state, and these bridges were drawn randomly from the total dataset. However, during the preliminary training and testing of the models, it was observed that most of the misclassifications occurred for condition ratings 5, 6, and 7. Thus, the number of bridge samples used is 13000, 14500, 14500, 14000, and 13000 for condition ratings 4, 5, 6, 7, and 8 respectively.

3.3 Data Analysis

3.3.1 Principal Component Analysis (PCA) for Prediction Model

The features were first scaled using the standard scaler module of the scikit-learn library to make sure they assumed the same range of values [76]. The scaled data for the 14 features for the deck and substructure and 15 features for the superstructure were then fit into the PCA module which automatically does the mean normalization to ensure that the data is distributed around the mean. Mean normalization is an integral part of PCA because it projects the original data points into different axes (principal components) to maximize the variation in the dataset, and without first centering the data around the mean, the PCA is biased towards the features with initial high variance and underrepresent the contribution of the other features to the principal components. A scree plot was generated to visualize the amount of variation explained by each of the components. The numbers of components from 2 to 14 or 2 to 15 were stored separately to represent different data sets and the number of components retained in each set represents the new features. It is important to know that selecting 2 principal components means that only the projection of the original features into 2 axes was retained while the other axes that explain a smaller amount of the variation in the data were removed. Also, the total amount of variation explained by selecting 2 principal components (features) is the sum of the variation explained by the first and second components. This guide also applies when selecting 3, 4, and up to 14 or 15 principal components. Finally, the PCA attribute ‘components_’ was used to extract the linear correlation between the original features and the principal components.

3.3.2 Base Case Prediction Model

A base model was created for the deck, superstructure, and substructure using all the original 14, 15, and 14 features in the dataset respectively to compare with the models created using the different number of cumulative principal components. All the features were scaled using the standard scalar module of the scikit-learn library to ensure that they all have the same range of values. This technique helps the model to converge faster and save some training time.

3.3.3 Artificial Neural Networks (ANN) Prediction Model Setup

The datasets from the PCA were used to develop ANN models to compare the efficacy of the different number of features (principal components) in predicting the deterioration of the bridge deck to that of the base case. Models were created for different numbers of principal components in incremental versions until the performance matches that of the base case and to showcase the importance of PCA as a dimensionality reduction technique. The loss function used in all the models is ‘Sparse Categorical Crossentropy’ because it is a classification task with different integer labels. Rectified linear unit (‘relu’) activation was adopted for the hidden layers of the neural network while the output layer utilizes the ‘softmax’ activation which is very suitable for multiclass classification problems. Adaptive moment estimate (Adam) was selected as the optimizer in all the models to regulate the learning rate and ensure quick convergence. To combat overfitting in the models, a combination of L2 regularization (ridge regression) and dropout was adopted, and the values were adjusted appropriately to ensure good generalization on the cross-validation and test set. The number of layers and neurons in the model was adjusted to achieve good prediction accuracy with minimum overfitting. In all the models, the data was split into train, cross-validation, and test set at 68%, 17%, and 15% respectively.

It is very important to know how many iterations (epochs) are required to train the model before there is no significant improvement in the accuracy. As such, the 'Early Stopping' module of the sci-kit learn library was utilized to stop training whenever the increase in accuracy did not surpass a particular level [76]. This approach helps to train the models more efficiently and reduce the training time which is one of the goals for using principal components instead of the original feature set. A value of 0.1% was set as a benchmark for training to stop if improvement in the validation accuracy does not surpass this level. However, it is not uncommon to see cases where the accuracy does not increase for some period as the model is trying to locate the 'local minima'. In this case, the training will be stopped even though the validation accuracy would have increased if a longer time was allotted to the training. To solve this problem, the 'patience value' was set at 50 iterations to permit the model to run for more time in case the accuracy might increase after which the training will be stopped. Each of the models was initially set at large epochs to allow them to effectively run until the condition for early stopping is reached. The final number of epochs used for each of the models were those obtained from the early stopping procedure after conducting the test multiple times to ensure that the output was stable. It is also worth noting that the early stopping technique helps to combat overfitting in the model.

The GridSearchCv module of the Scikit-learn library was first used to conduct an initial parameter search for the bridge components deterioration models to know the range of values that would give the optimum model performance. The result from the grid search formed the foundation for selecting initial values for the optimizers (for regulating the learning rate), hidden layers, and regularization parameters. Table 3-8**,**

Table 3-9**,** and Table 3-10 shows the ANN model parameters used for the deck, superstructure, and substructure deterioration models respectively including the number of layers and the associated number of neurons (or units) in each layer. The first and the last layer in each of the models represent the input and output layers respectively while all other layers in between are the hidden layers.

Table 3-8: ANN Model Architecture for Deck.

Model	ANN Layers and Neurons	Dropout	L2 Regularizer	Optimizer (Adam)
2-PC	2-50-150-200-50-5	0.01	0.00015	0.0001
3-PC	3-50-150-200-150-50-5	0.08	0.00015	0.0001
4-PC	4-50-150-200-200-100-50-5	0.08	0.00015	0.0001
5-PC	5-50-150-200-150-100-50-5	0.08	0.00015	0.0001
6-PC	6-50-150-200-200-150-50-5	0.08	0.00015	0.0001
7-PC	7-50-150-300-300-300-300-150-50-5	0.1	0.00015	0.0001

8-PC	8-50-150-300-300-300-300-300-150-150-50-5	0.09	0.00015	0.0001
9-PC	9-50-150-300-300-300-300-300-150-50-5	0.09	0.00015	0.0001
Base-Case	14-50-150-300-300-450-450-300-150-50-5	0.1	0.00015	0.0001

Table 3-9: ANN Model Architecture for Superstructure.

Model	ANN Layers and Neurons	Dropout	L2 Regularizer	Optimizer (Adam)
2-PC	2-50-150-300-50-5	0.1	0.001	0.0001
3-PC	3-50-150-300-300-50-5	0.1	0.001	0.0001
4-PC	4-50-150-300-300-150-50-5	0.085	0.0003	0.0001
5-PC	5-50-150-300-300-300-150-50-5	0.15	0.0003	0.0001
6-PC	6-50-150-300-300-300-300-150-50-5	0.12	0.0002	0.0001
7-PC	7-50-150-300-300-300-300-300-150-50-5	0.09	0.0002	0.0001
8-PC	8-50-150-300-300-450-450-300-	0.075	0.00015	0.0001

	150-150-50-5			
9-PC	9-50-150-300-300-450-600-450-300-150-150-50-5	0.075	0.00015	0.0001
Base-Case	15-50-150-300-300-450-450-600-600-300-150-150-50-5	0.09	0.00015	0.0001

Table 3-10: ANN Model Architecture for Substructure.

Model	ANN Layers and Neurons	Dropout	L2 Regularizer	Optimizer (Adam)
2-PC	2-50-100-200-50-5	0.15	0.003	0.0001
3-PC	3-50-100-300-100-50-5	0.15	0.0003	0.0001
4-PC	4-50-100-300-300-100-50-5	0.15	0.0003	0.0001
5-PC	5-50-100-300-300-300-100-50-5	0.12	0.0003	0.0001
6-PC	6-50-100-300-300-300-300-100-50-5	0.12	0.0003	0.0001
7-PC	7-50-150-300-300-300-300-300-150-50-5	0.1	0.0003	0.0001
8-PC	8-50-150-300-300-450-450-300-150-150-50-5	0.1	0.00015	0.0001

9-PC	9-50-150-300-300-450-450-450-300-150-150-50-5	0.09	0.00015	0.0001
10-PC	10-50-150-300-300-450-600-450-300-150-150-50-5	0.08	0.00015	0.0001
Base-Case	14-50-150-300-300-450-600-600-450-300-150-150-50-5	0.1	0.00015	0.0001

3.3.4 Prediction Model Performance Measurement

3.3.4.1 Model Performance

The effectiveness of the models was measured using their performance on the test set which is equivalent to 15% of the whole dataset i.e., 7950, 9450, and 10350 bridge samples for the deck, superstructure, and substructure respectively. The metrics used are precision, accuracy, and the F1-score of the models in predicting each class of bridge. The precision of a model in predicting a bridge class is the ratio of the number of accurate predictions to the number of total predictions for that class. The recall for a bridge is the ratio of the number of accurate predictions to the actual number of samples in the bridge class. Also, a combination of the precision and recall for a bridge class was used to calculate the F1-score of the model for a particular bridge class. Equations 3.4, 3.5, and 3.6 illustrate how to compute the precision, accuracy, and F1 score respectively. The number of bridge samples that were used in computing the precision and recall of a model for a bridge class is not explicitly stated in the classification report of the model. The classification report only shows the 'Support' for each of the classes which represents the total number of samples allotted to it in the test set. However, the precision and recall values can be validated by checking the confusion matrix, which clearly shows the amount of correct and incorrect classifications for each bridge class in a model.

$$\text{Precision (P)} = \frac{\text{True Positive}}{\text{Predicted Postive}} \tag{3.4}$$

$$\text{Recall (R)} = \frac{\text{True Positive}}{\text{Actual Postive}} \tag{3.5}$$

$$\text{F1-score} = \frac{2*P*R}{P+R} \tag{3.6}$$

3.3.4.2 Confusion Matrix

The confusion matrix is a detailed chart that shows the actual number of samples that were used in computing the precision, recall, and F1-score that appeared on the classification report. As illustrated in Figure 3-6, the confusion matrix consists of 2 axes representing the true (or actual) class and the predicted class for a particular test data. The model performance is illustrated by the number of test samples it accurately classified (i.e., true positive and true negative) into the proper class and the amount it misclassified (i.e., false negative and false positive). This is particularly useful to identify which neighboring classes are misclassified by a model to help in taking the right actions to improve the model performance.

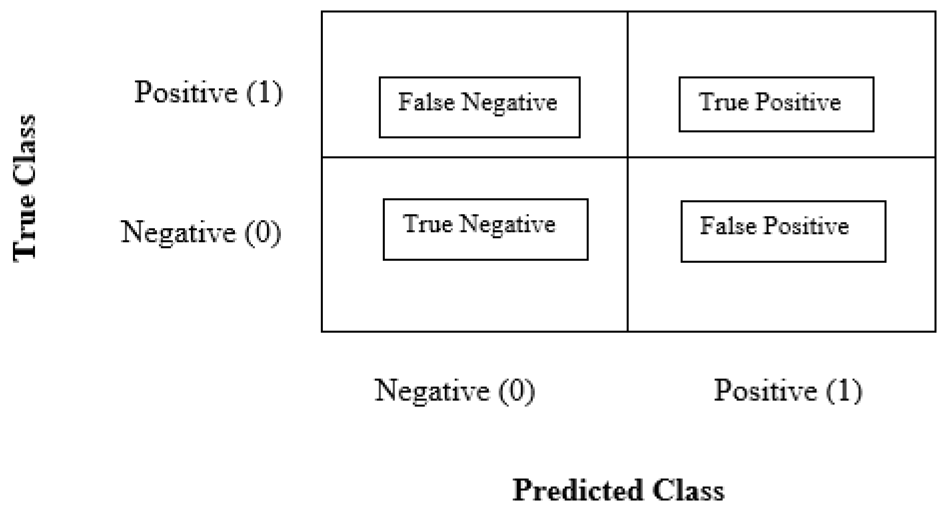


Figure 3-6: Confusion Matrix Illustration.

3.3.5 Determination of Bridge Components' and Elements' Weight

To determine the weight (or importance) of the bridge elements, the random forest classifier of the Scikit-Learn library was used to evaluate the changes in the condition rating value of the major components while having the associated bridge elements' health index as the features [76]. The weight of the bridge components was also determined by using the random forest classifier to evaluate the changes in the overall rating of the bridge while having the three components' condition ratings as the features. Since the condition of the bridge elements influences the condition rating of the major components they are nested under, it makes sense to take the elements' health index as the explanatory variable (independent variable) and the condition rating values as the response variable (dependent variable). The same analogy goes for the relationship between the components' condition rating and the overall rating of the bridge where the components' condition ratings are the independent variables, and the overall rating is the dependent variable. A total of 60 models were created where 45 of the models represent the elements and major components (i.e., Deck, Superstructure, and Substructure) relationship for each of the 15 bridge design types considered in the analysis, and the remaining 15 represent the major components and the overall rating relationship.

The number of decision trees in the random forest was set to 100 for each of the models created. After implementing the random forest classifier model, the feature selection module of the Scikit-Learn library was used to rank the bridge elements based on their importance to the condition rating of the major component. The models were observed to be very stable even after running them multiple times, there was no significant change in the output of the bridge elements' importance rank. The bridge elements' importance rank and components' importance rank for all the main span design groups will be compared to draw out similarities and conclusions.

It is worth noting that the random forest algorithm was not used in predicting the condition rating of the bridge components, only the powerful feature selection framework was utilized to evaluate the relationship between the bridge data. Having to use the algorithm to predict the condition rating of the bridge components based on the health index of the associated elements will require a balanced dataset to achieve a good level of performance. A balanced dataset means an equal or almost equal number of representations of the different component condition rating classes. A total of 10 classes is available in this case representing the component condition ratings from 0 to 9 and the data available is not sufficient to evenly divide the different bridge design types into condition rating categories since the first set of element-level data was collected in 2015.

3.3.6 Formation Of Bridge Health Index (BHI) Equations

After getting the bridge elements and components' importance rank for the 15 bridge design types, it becomes essential to consider how to synthesize this data to come up with the resultant weight of the bridge elements and the overall bridge health index equation for all bridge design types. Since the importance of the individual bridge elements to the condition of the associated bridge component is known and the importance of the bridge component to the overall rating of the bridge is also known, the resultant importance (or weight) of the bridge elements to the whole bridge can be mathematically computed. For example, if an element's importance to a particular component is computed to be 0.25 and the component's importance to the overall rating of the bridge is computed to be 0.30, the overall (resultant) importance of the bridge element to the bridge can be calculated as 0.075.

The resultant bridge elements' weight was computed for all the elements in the 15 bridge design types and a BHI equation was developed for each of the bridge design types. Equation 3.7 shows the format for representing the BHI equation for each of the bridge design types.

$$BHI = \sum ((\text{Element Weight}) * (\text{Element Health Index, } Hi_e)) \quad (3.7)$$

Chapter 4

RESULTS AND DISCUSSION

4.1 Data Analysis Results for Prediction Model

The data analysis result for the prediction model includes the initial Principal Component Analysis (PCA) that was conducted on the Deck, Superstructure, and Substructure data and also the Artificial Neural Network (ANN) models created for each of these three components. The scope of the analysis conducted to develop reliable condition prediction models for the bridge components is illustrated in Figure 4-1.

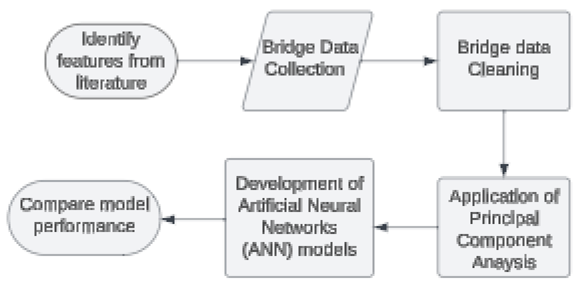


Figure 4-1: Bridge Components Condition Prediction Model Flow-Chart.

4.1.1 Deck PCA Result

The amount of variation in the bridge deck feature set explained by each of the principal components and the cumulative values when multiple principal components (PC) are combined is shown in Table 4-1. Figure 4-2 also shows the graphical trend of the cumulative values of the principal components.

Table 4-1: Variance explained by principal components for the bridge deck.

Principal Components (PC)	Variance Explained (%)	Cumulative (%)
1	23.37	23.37
2	14.15	37.52
3	11.95	49.47
4	11.09	60.56
5	10.13	70.69
6	6.86	77.55
7	6.14	83.69
8	4.53	88.22
9	3.50	91.72
10	3.06	94.78
11	2.62	97.40
12	1.21	98.61
13	0.95	99.56
14	0.45	100

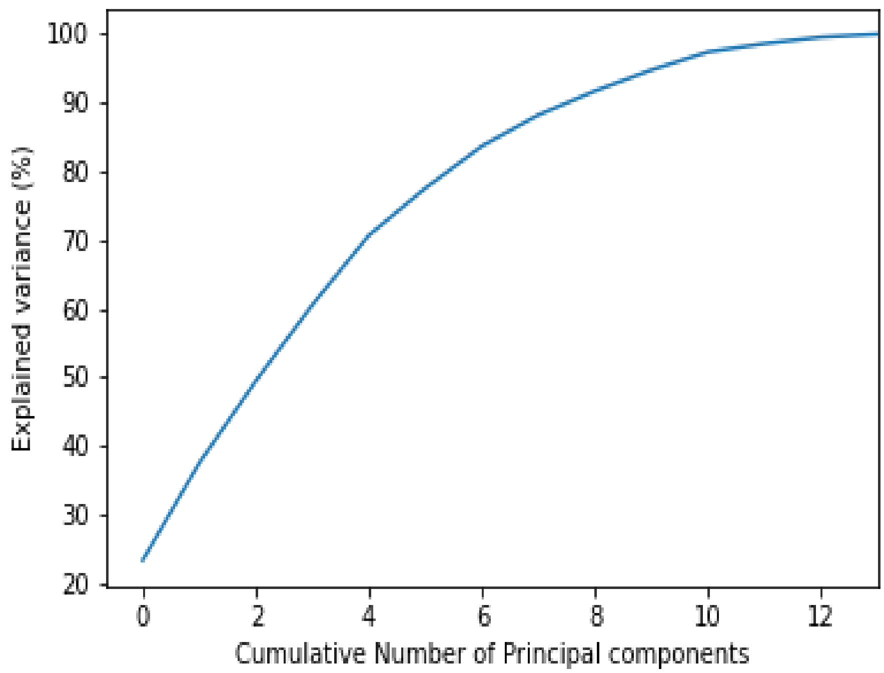


Figure 4-2: Scree plot for deck PCA.

The amount of variance explained is highest in the first PC and reduces progressively in subsequent PCs. The cumulative amount of variance explained by all the PCs must be 100% in all cases. It is worth noting that selecting a particular set of PCs for analysis signifies combining all the PCs and the amount of variation explained is the cumulative variance up to the desired point. E.g., selecting 5 PCs for analysis will result in a cumulative explained variance of 70.69% as shown in Table 4-1. The correlation between the original deck feature set and the principal components is shown in Figure 4-3.

As shown in Figure 4-3, PC1 is more influenced by the average daily traffic (ADT) and the average daily truck traffic (ADTT), this shows that the traffic condition on the bridge deck holds very important information about the state of the bridge because PC1 has the highest amount of explained variance as shown in Table 4-1 and affirms the relevance of the traffic-related data. PC2 is more influenced by the operating and inventory rating of the bridge, PC3 is more influenced by the number of freeze-thaw cycles, PC4 is more influenced by the number of snowfall days, PC5 is more influenced by the structure length, PC6 is more influenced by the skew angle, PC7 is more influenced by the bridge age, PC8 is more influenced by the deck width, and PC9 is more influenced by the structure length. As the scree-plot in Figure 4-2 flattens after the ninth principal component, it makes sense not to consider any principal component after this point because the added explained variance becomes insignificant. This result highlights the most important and dominating deck features that hold the relevant information useful in predicting its future condition.

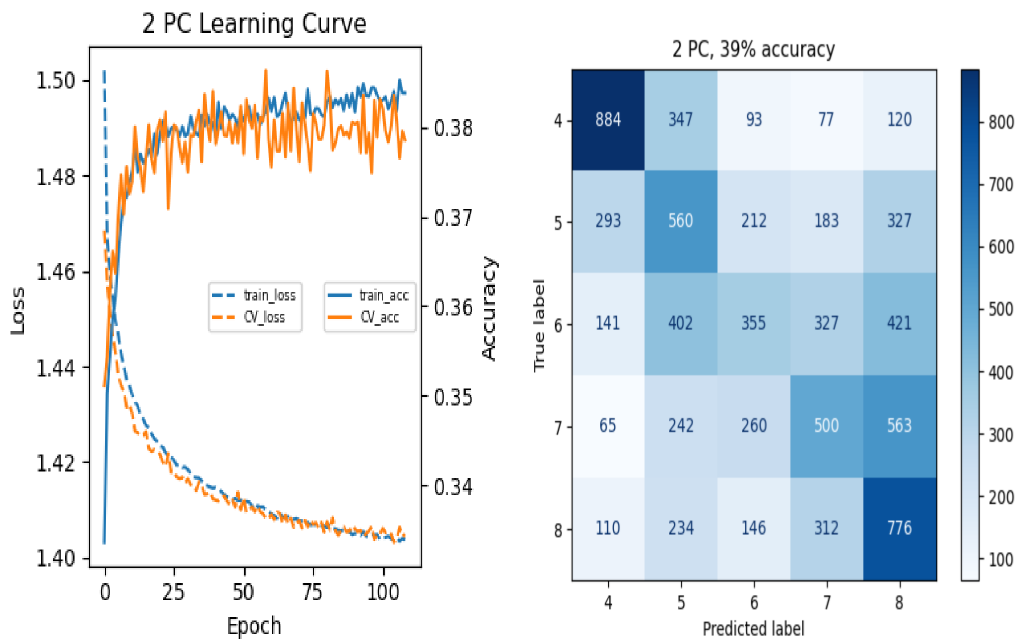
	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
Average Daily Traffic	0.513726	0.097805	0.065922	-0.13934	-0.04996	-0.08238	0.067932	0.190126	-0.03684
Number of Spans in Main Unit	0.086525	0.14867	-0.1182	0.350247	0.564451	-0.05974	0.159155	0.181278	0.64654
Structure Length	0.115686	0.150694	-0.087837	0.334896	0.578098	0.000269	0.051203	-0.129713	-0.664578
Bridge Age	-0.152509	0.265633	0.035611	-0.130578	-0.116848	-0.226216	0.867123	-0.236704	-0.052116
Skew Angle	0.115716	-0.013388	0.130738	-0.081975	0.058941	0.941838	0.256036	-0.042924	0.050666
Length of Maximum Span	0.014331	0.213214	0.460625	0.390195	-0.205419	0.041643	-0.020814	0.342748	-0.256566
Deck Width	0.333847	0.13589	0.198518	0.101336	-0.032549	-0.029932	-0.227933	-0.82674	0.147177
Operating Rating	0.124989	-0.5947	0.181803	0.206006	-0.00408	-0.10033	0.24497	-0.04387	-0.01658
Inventory Rating	0.133285	-0.61369	0.140893	0.209542	0.004778	-0.06924	0.153986	-0.03541	0.010249
Lanes on the Structure	-0.03085	0.259777	0.483676	0.343375	-0.19822	-0.04863	-0.00951	0.031078	0.211973
Future Average Daily Traffic	0.492355	0.049085	-0.0247	-0.21431	0.0087	-0.08289	0.050992	0.136108	-0.00283
Average Daily Truck Traffic	0.499823	0.079837	0.0365	-0.181	-0.02443	-0.09754	0.061774	0.179563	-0.01969
Number of Freeze-Thaw Cycles	-0.13736	-0.03701	0.487097	-0.35162	0.324712	-0.11019	-0.04767	0.011199	0.018856
Number of Snowfall Days	-0.14651	-0.05517	0.421865	-0.39101	0.372662	-0.05505	-0.07133	0.016763	-0.01142

Figure 4-3: Relationship between deck features and principal components.

4.1.2 Deck Deterioration Model Results

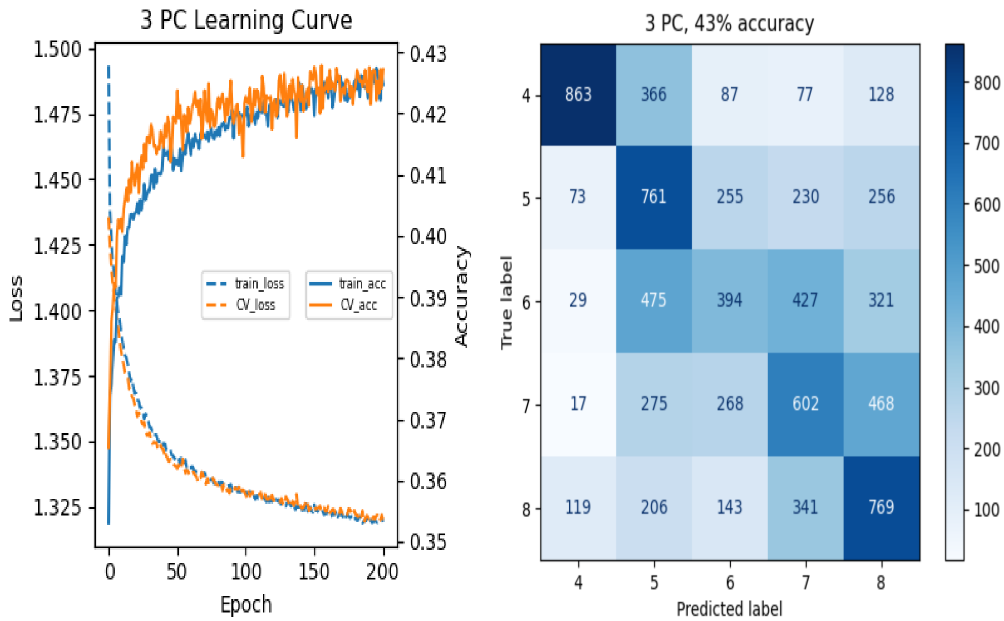
The learning curves and confusion matrix for the bridge deck prediction models created from different sets of PCs are shown in Figure 4-4 to Figure 4-11 and that of the base model is shown in Figure 4-12. The learning curves show the trend in the loss (or error) and accuracy over a series of iterations (epochs) during the model training process. The confusion matrix shows the model performance on the difference condition rating classes in the test set. Table 4-2 to Table 4-9 shows the classification report for the PC models and Table 4-10 shows the classification report for the base model.

By comparing the variation explained by each set of principal components in Table 4-1 with the performance of the models, it was observed that the overall F1 score continues to increase with an increasing number of principal components. This signifies the importance of an added component that gives more information about the dataset on the performance of the models. Increasing the complexity of the neural network in terms of the number of hidden layers and neurons does not have as much effect on the performance of the model as an additional component does. An illustration of this is the 4-PC and model 5-PC with the architecture shown in Table 3-8. Model 4-PC has 50 more neurons in the fifth layer while the other layers were the same as that of model 5-PC. The classification reports from the two models in Table 4-4 and Table 4-5 show that Model 5-PC has an overall F1-score of 61% while Model 4 PC has an overall F1-score of 51%.



(a) (b)

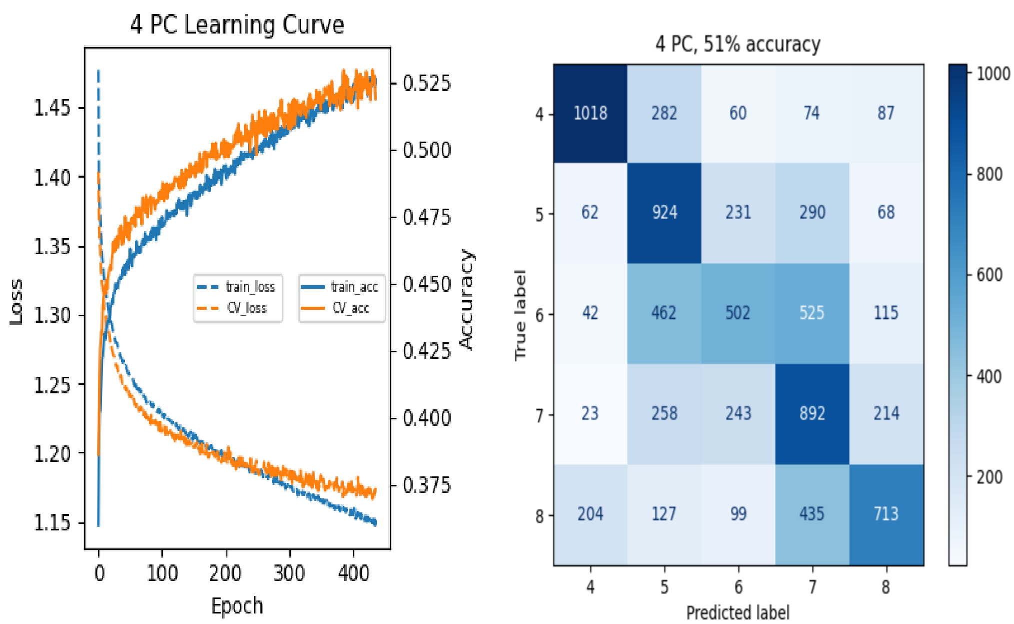
Figure 4-4: (a) Learning curve for the deck model with 2 Principal Components (b) Confusion matrix for deck model 2 PC on the test set.



(a) (b)

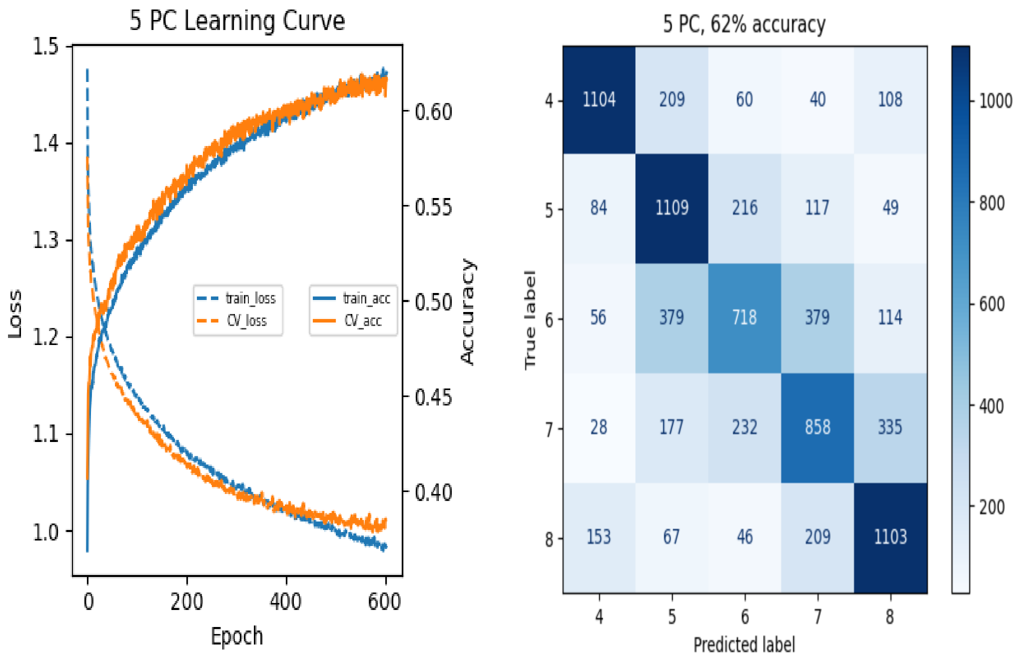
Figure 4-5: (a) The learning curve for the deck model with 3 Principal Components (b) Confusion matrix for deck model 3 PC on the test set.

As shown in Table 4-1 the cumulative variation explained by the 4 and 5 principal components is 60.56% and 70.69% respectively. This shows that an additional component that contributes 10.13% to the explained variance of the data has an almost equal contribution to the model performance (i.e., an 11% increase in accuracy). Models 8-PC and 9-PC also show the same relationship exhibited by 4 PC and 5 PC. Table 3-8 shows that model 8-PC has an additional layer of 150 neurons more than model 9-PC. However, the overall F1-score is 74% and 76% for models 8-PC and 9-PC respectively as seen in the classification report in Table 4-8 and Table 4-9. The cumulative variance explained by 8-PC and 9-PC is 88.22% and 91.72% respectively. The idea of an increase in the model performance as a result of an added principal component does not hold indefinitely as it can be shown in the model performance result that the addition of lower-level principal components does not have as much impact on the model performance as the upper-level principal components. Many PC models were created until the performance level matched that of the base model. More PC models were not created after matching the base model performance so as not to circumvent the goal of having a condition prediction model with reduced dimensionality.



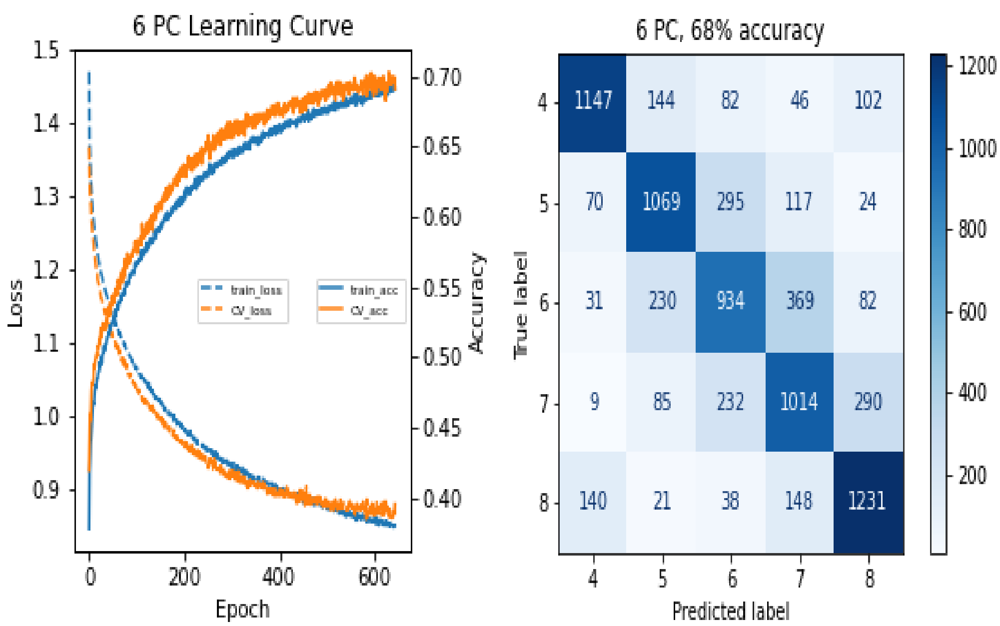
(a) (b)

Figure 4-6: (a) Learning curve for the deck model with 4 Principal Components (b) Confusion matrix for deck model 4 PC on the test set.



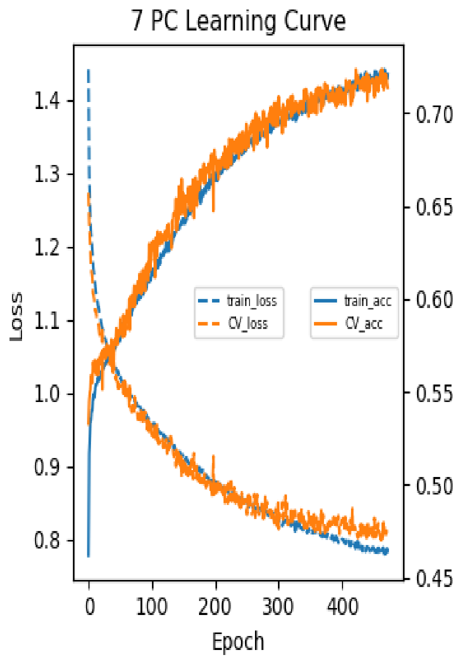
(a) (b)

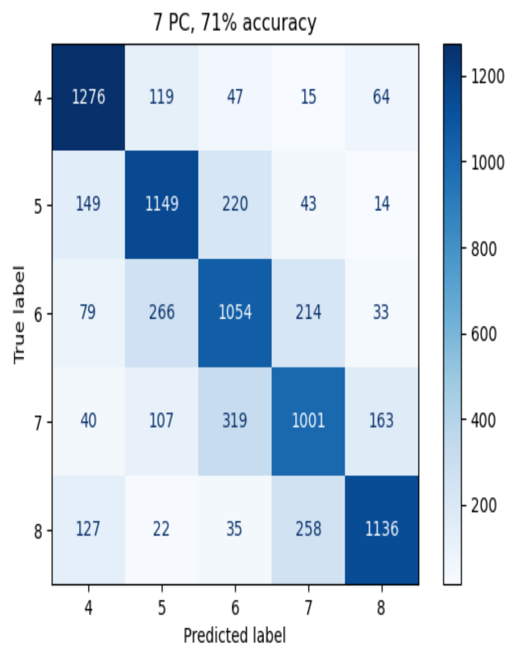
Figure 4-7: (a) Learning curve for the deck model with 5 Principal Components (b) Confusion matrix for deck model 5 PC on the test set.



(a) (b)

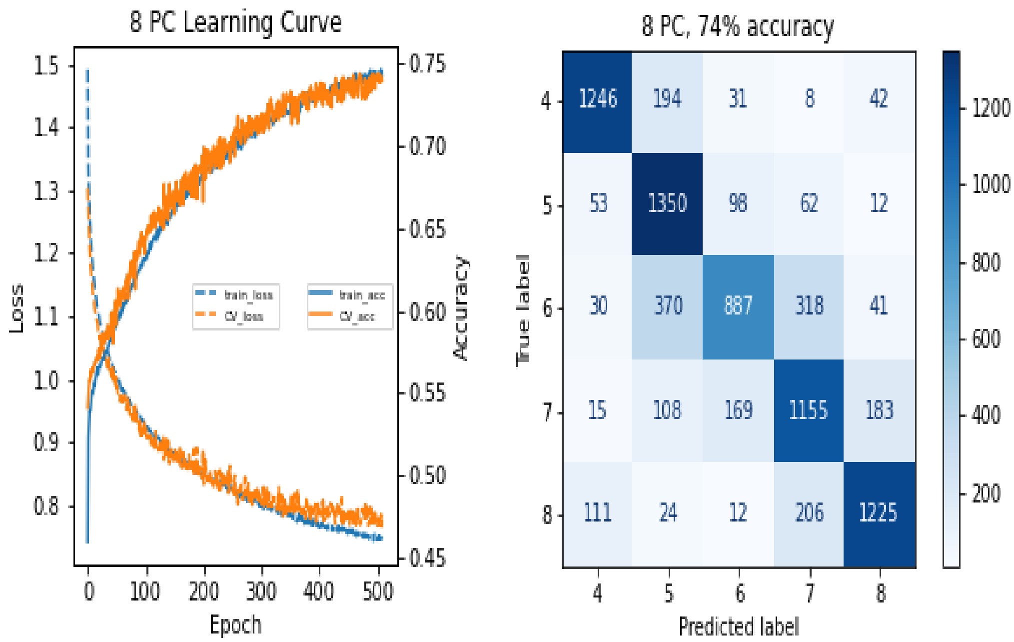
Figure 4-8: (a) Learning curve for the deck model with 6 Principal Components (b) Confusion matrix for deck model 6 PC on the test set.





(a) (b)

Figure 4-9: (a) Learning curve for the deck model with 7 Principal Components (b) Confusion matrix for deck model 7 PC on the test set.



(a) (b)

Figure 4-10: (a) The learning curve for the deck model with 8 Principal Components (b) Confusion matrix for deck model 8 PC on the test set.

Table 4-2: Classification report for deck deterioration model with 2 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.59	0.58	0.59	1521
5	0.31	0.36	0.33	1575
6	0.33	0.22	0.26	1646
7	0.36	0.31	0.33	1630
8	0.35	0.49	0.41	1578

Accuracy			0.39	7950
Macro Average	0.39	0.39	0.38	7950
Weighted Average	0.39	0.39	0.38	7950

Table 4-3: Classification report for deck deterioration model with 3 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.78	0.57	0.66	1521
5	0.37	0.48	0.42	1575
6	0.34	0.24	0.28	1646
7	0.36	0.37	0.36	1630
8	0.40	0.49	0.44	1578
Accuracy			0.43	7950
Macro Average	0.45	0.43	0.43	7950
Weighted Average	0.45	0.43	0.43	7950

Table 4-4: Classification report for deck deterioration model with 4 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.75	0.67	0.71	1521
5	0.45	0.59	0.51	1575

6	0.44	0.30	0.36	1646
7	0.40	0.55	0.46	1630
8	0.60	0.45	0.51	1578
Accuracy			0.51	7950
Macro Average	0.53	0.51	0.51	7950
Weighted Average	0.53	0.51	0.51	7950

Table 4-5: Classification report for deck deterioration model with 5 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.77	0.73	0.75	1521
5	0.57	0.70	0.63	1575
6	0.56	0.44	0.49	1646
7	0.54	0.53	0.53	1630
8	0.65	0.70	0.67	1578
Accuracy			0.62	7950
Macro Average	0.62	0.62	0.61	7950
Weighted Average	0.62	0.62	0.61	7950

Table 4-6: Classification report for deck deterioration model with 6 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.82	0.75	0.79	1521
5	0.69	0.68	0.68	1575
6	0.59	0.57	0.58	1646
7	0.60	0.62	0.61	1630
8	0.71	0.78	0.74	1578
Accuracy			0.68	7950
Macro Average	0.68	0.68	0.68	7950
Weighted Average	0.68	0.68	0.68	7950

Table 4-7: Classification report for deck deterioration model with 7 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.76	0.84	0.80	1521
5	0.69	0.73	0.71	1575
6	0.63	0.64	0.63	1646
7	0.65	0.61	0.63	1630
8	0.81	0.72	0.76	1578

Accuracy			0.71	7950
Macro Average	0.71	0.71	0.71	7950
Weighted Average	0.71	0.71	0.71	7950

Table 4-8: Classification report for deck deterioration model with 8 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.86	0.82	0.84	1521
5	0.66	0.86	0.75	1575
6	0.74	0.54	0.62	1646
7	0.66	0.71	0.68	1630
8	0.82	0.78	0.80	1578
Accuracy			0.74	7950
Macro Average	0.75	0.74	0.74	7950
Weighted Average	0.75	0.74	0.74	7950

Table 4-9: Classification report for deck deterioration model with 9 PC.

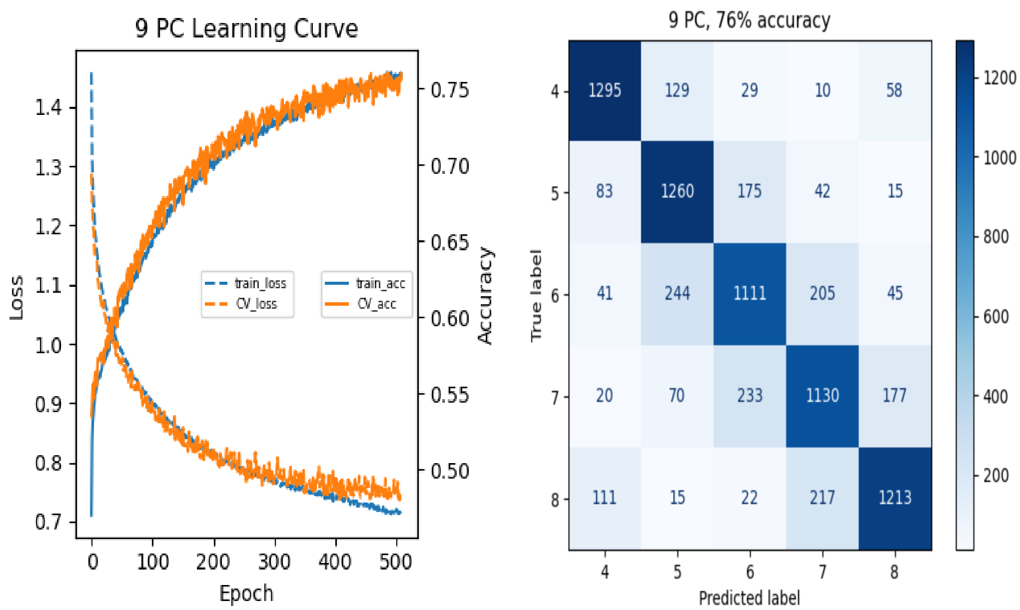
Condition Rating	Recall	Precision	F1-score	Support

4	0.84	0.85	0.84	1521
5	0.73	0.80	0.77	1575
6	0.71	0.67	0.69	1646
7	0.70	0.69	0.70	1630
8	0.80	0.77	0.79	1578
Accuracy			0.76	7950
Macro Average	0.76	0.76	0.76	7950
Weighted Average	0.76	0.76	0.76	7950

Table 4-10: Classification report for deck deterioration model with all 14 features (Base Model).

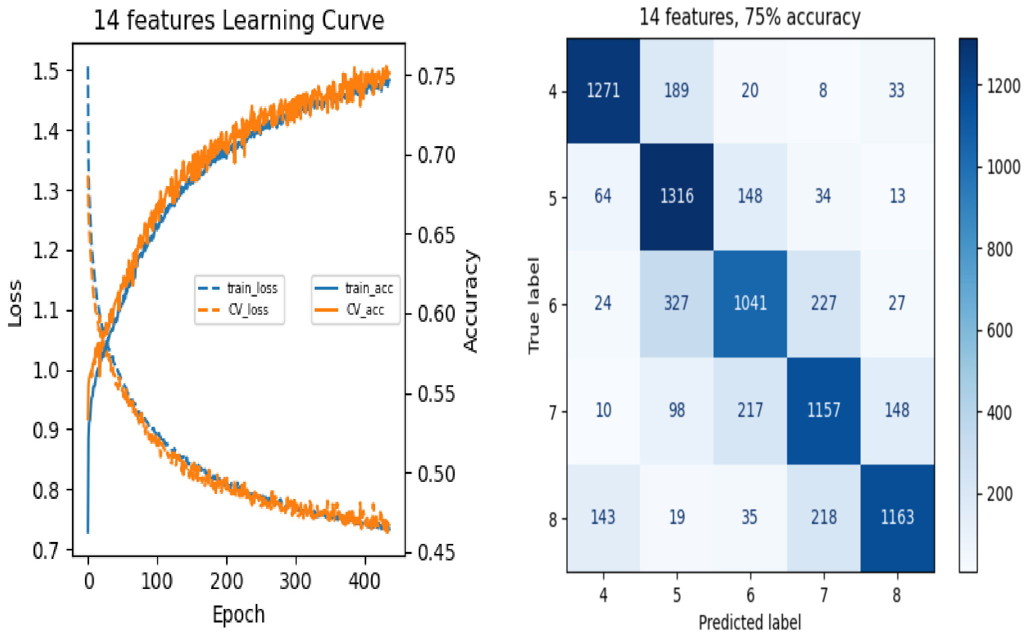
Condition Rating	Recall	Precision	F1-score	Support
4	0.84	0.84	0.84	1521
5	0.68	0.84	0.75	1575
6	0.71	0.63	0.67	1646
7	0.70	0.71	0.71	1630
8	0.84	0.74	0.79	1578
Accuracy			0.75	7950
Macro Average	0.75	0.75	0.75	7950

Weighted Average	0.75	0.75	0.75	7950



(a) (b)

Figure 4-11: (a) The learning curve for the deck model with 9 Principal Components (b) Confusion matrix for deck model 9 PC on the test set.



(a) (b)

Figure 4-12: (a) The learning curve for the deck base model with all features (b) Confusion matrix for the deck base model with all features on the test set.

Figure 4-12 shows the learning curve and confusion matrix for the base model with all 14 bridge features. When compared with all the PC models, it performs better in terms of F1-score and accuracy than all the models except for model 9 PC as shown in the classification report in Table 4-10. Although the model 8 PC, which corresponds to 88.22% of the variation in the dataset has the same average recall as the base model, the overall F1-score and accuracy were 1% short. However, model 9 PC with 91.72% of the variation in the dataset performed better than the base model with an overall F1-score and accuracy of 76%. As such creating a prediction model with 9 principal components will produce a better performance than using the 14 bridge features.

4.1.3 Superstructure PCA Result

The amount of variation in the bridge superstructure feature set explained by each of the principal components and the cumulative values when multiple principal components (PC) are combined is shown in Table 4-11. The scree-plot in Figure 4-13 also shows the graphical trend of the cumulative values of the principal components.

Table 4-11: Variance is explained by the principal components of the bridge superstructure.

Principal Components	Variance Explained (%)	Cumulative (%)
1	27.99	27.99
2	15.48	43.47
3	10.06	53.53
4	8.15	61.68
5	7.26	68.94
6	6.25	75.19
7	6.08	81.27
8	5.42	86.69
9	4.61	91.30
10	3.52	94.82
11	1.47	96.29

12	1.25	97.54
13	1.12	98.66
14	0.95	99.61
15	0.39	100

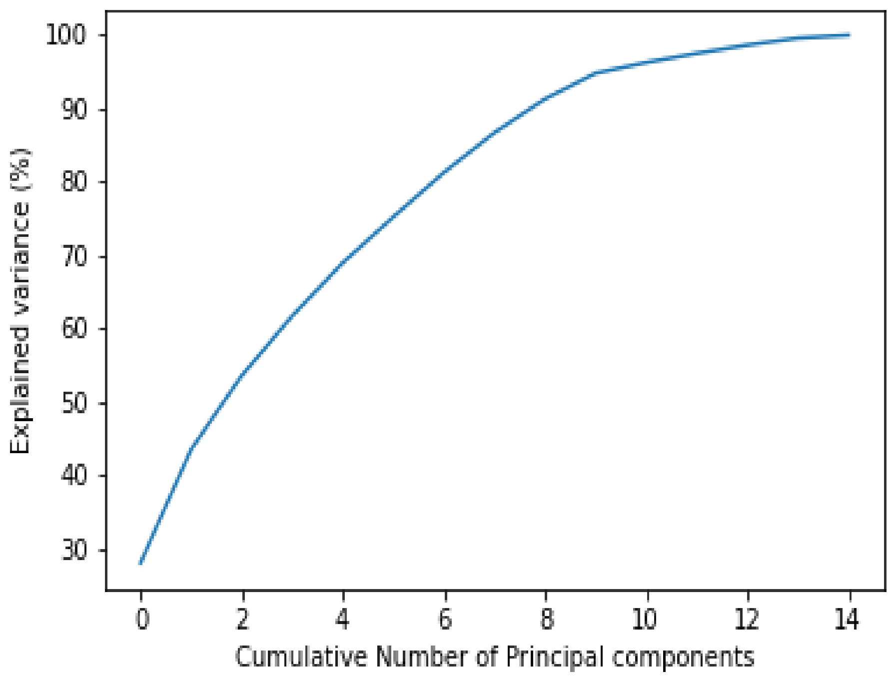


Figure 4-13: Scree plot for superstructure PCA.

As with the normal trend for any principal component analysis, the amount of variance explained is highest in the first PC and reduces progressively in subsequent PCs. The cumulative amount of variance explained by all the PCs is equal to 100%. The correlation between the original superstructure feature set and the principal components is shown in Figure 4-14.

Like the deck principal component analysis, PC1 is more influenced by the average daily traffic (ADT) and the average daily truck traffic (ADTT), which further reinforces the importance of the traffic-related data as shown in Figure 4-14. PC2 is more influenced by the number of days with measurable precipitation, PC3 is more influenced by the structure length, PC4 is more influenced by the total precipitation, PC5 is more influenced by the bridge age, PC6 is more influenced by the skew angle, PC7 is more influenced by the length of maximum span, PC8 is more influenced by the bridge age, and PC9 is more influenced by the number of lanes under the structure. This showcases the most important superstructure features that hold the relevant information useful in predicting its future condition.

	PC-1	PC-2	PC-3	PC-4	PC-5	PC-6	PC-7	PC-8	PC-9
Average daily traffic	0.433725	0.127402	-0.13417	0.018513	0.178247	0.000494	0.107496	-0.20394	0.164742
Number of spans in main unit	0.092583	-0.00684	0.542411	0.126097	0.382586	-0.2169	-0.32235	-0.00325	-0.01495
Structure length	0.109111	0.017185	0.630744	0.031583	0.244672	-0.03111	0.050833	-0.01098	-0.12343
Bridge age	-0.12555	0.005451	-0.19113	0.18489	0.446974	-0.38848	0.499136	0.546897	0.09613
Skew	0.103185	0.061935	0.03861	-0.31322	-0.33638	-0.79602	0.146488	-0.23488	-0.24742
Bridge Roadway Width	0.39354	0.124237	-0.06459	0.010912	-0.12089	0.069783	-0.12833	0.326532	-0.36694
Length of maximum span	0.076685	0.023661	0.386449	-0.17128	-0.22301	0.360244	0.721442	0.008594	-0.10227
Lanes on the structure	0.36876	0.134137	-0.07955	0.045485	-0.08333	0.078556	-0.13926	0.424164	-0.42145
Lanes under the structure	0.214344	0.020899	0.22941	-0.13091	-0.40855	-0.10689	-0.13666	0.432637	0.704045
Future average daily traffic	0.406618	0.131437	-0.10818	0.015932	0.169459	0.000691	0.094397	-0.21776	0.196897
Average daily truck traffic	0.413234	0.134294	-0.14088	0.015083	0.193372	0.018423	0.112326	-0.247	0.167026
Average Temperature	0.165024	-0.45323	0.006575	0.469284	-0.18483	-0.08127	0.094499	-0.0476	-0.00608
Number of freeze-thaw cycles	-0.16356	0.486348	-0.01839	-0.40072	0.167645	0.061068	-0.05565	0.081544	0.032908
Number of days with measurable precipitation	-0.15402	0.528609	0.056465	0.338867	-0.12854	-0.03829	0.028092	-0.04327	0.023923
Total Precipitation	-0.09223	0.43411	0.065849	0.552894	-0.2568	-0.05517	0.044675	-0.10495	0.013678

Figure 4-14: Relationship between superstructure features and principal components.

4.1.4 Superstructure Deterioration Model Results

The learning curves and confusion matrix for the bridge superstructure prediction models created for different sets of PCs are shown in Figure 4-15 to Figure 4-22 and that of the base model is shown in Figure 4-23. The learning curves show the trend in the loss (or error) and accuracy over a series of iterations (epochs) during the model training process. The confusion matrix shows the model performance on the difference condition rating classes in the test set. Table 4-12 to Table 4-19 show the classification report for the PC models and Table 4-20 shows the classification report for the base model.

Like the PC models for the bridge deck, the performance of the PC models for the bridge superstructure improves with every added component until it reaches a point where there is no significant improvement in the F1-score of the models due to the limited amount of variation contributed by the added components. It was observed in all the PC models and the base model (i.e., with 15 features) that the best performance in terms of F1-score is observed in condition ratings 4 and 8, and the worst performance is observed in condition rating 6. This can be attributed to the effect of the subjectivity of the data collection process, condition ratings 4 and 8 are at the extreme points and are easily identifiable because poor bridge components in serious deplorable condition and good bridge components in almost perfect condition are easily elicited whereas condition ratings in between this two extremes are subject to more variations in quantifying the extent of damage which overall affect the model performance on this portion of the data.

The model performance result shows that the model 8-PC which corresponds to 86.69% of the variation in the dataset has the same accuracy and F1-score (i.e., 74%) as the base model that uses all the 15 bridge features and going a step further to add another component to make the model 9-PC which corresponds to 91.3% of the variation in the dataset result in a higher overall accuracy and F1-score (i.e., 76%). This shows that suppressing the redundancies in the dataset by using the principal components will lead to a better model performance while also minimizing the data dimensionality. An in-depth look at the model classification reports for model 9-PC and the base model shows that model 9-PC performed better on all the condition rating classes than the base model as depicted by the F1-scores in Table 4-19 and Table 4-20.

Table 4-12: Classification report for superstructure deterioration model with 2 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.43	0.63	0.51	1751
5	0.26	0.02	0.04	1821
6	0.30	0.44	0.36	1974
7	0.26	0.22	0.24	2036
8	0.29	0.32	0.30	1868
Accuracy			0.32	9450
Macro Average	0.31	0.33	0.29	9450
Weighted Average	0.31	0.32	0.29	9450

Table 4-13: Classification report for superstructure deterioration model with 3 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.50	0.55	0.53	1751
5	0.45	0.30	0.36	1821
6	0.34	0.21	0.26	1974
7	0.29	0.37	0.33	2036
8	0.37	0.49	0.42	1868
Accuracy			0.38	9450
Macro Average	0.39	0.39	0.38	9450
Weighted Average	0.39	0.38	0.38	9450

Table 4-14: Classification report for superstructure deterioration model with 4 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.60	0.51	0.55	1751
5	0.47	0.34	0.39	1821
6	0.32	0.45	0.37	1974
7	0.33	0.34	0.34	2036
8	0.44	0.44	0.44	1868

Accuracy			0.41	9450
Macro Average	0.43	0.41	0.42	9450
Weighted Average	0.43	0.41	0.41	9450

Table 4-15: Classification report for superstructure deterioration model with 5 PC.

Condition Rating	Recall	Precision	F1-score	Support
4	0.63	0.58	0.60	1751
5	0.49	0.47	0.48	1821
6	0.41	0.33	0.37	1974
7	0.40	0.39	0.39	2036
8	0.49	0.67	0.57	1868
Accuracy			0.48	9450
Macro Average	0.49	0.49	0.48	9450
Weighted Average	0.48	0.48	0.48	9450

Table 4-16: Classification report for superstructure deterioration model with 6 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.68	0.63	0.66	1751

5	0.55	0.60	0.57	1821
6	0.48	0.52	0.50	1974
7	0.50	0.53	0.52	2036
8	0.71	0.59	0.65	1868
Accuracy			0.57	9450
Macro Average	0.58	0.57	0.58	9450
Weighted Average	0.58	0.57	0.58	9450

Table 4-17: Classification report for superstructure deterioration model with 7 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.75	0.75	0.75	1751
5	0.61	0.75	0.67	1821
6	0.67	0.55	0.61	1974
7	0.67	0.58	0.62	2036
8	0.70	0.78	0.74	1868
Accuracy			0.68	9450
Macro Average	0.68	0.68	0.68	9450
Weighted Average	0.68	0.68	0.67	9450

Table 4-18: Classification report for superstructure deterioration model with 8 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.76	0.86	0.80	1751
5	0.73	0.73	0.73	1821
6	0.75	0.60	0.66	1974
7	0.69	0.72	0.71	2036
8	0.76	0.79	0.78	1868
Accuracy			0.74	9450
Macro Average	0.74	0.74	0.74	9450
Weighted Average	0.74	0.74	0.73	9450

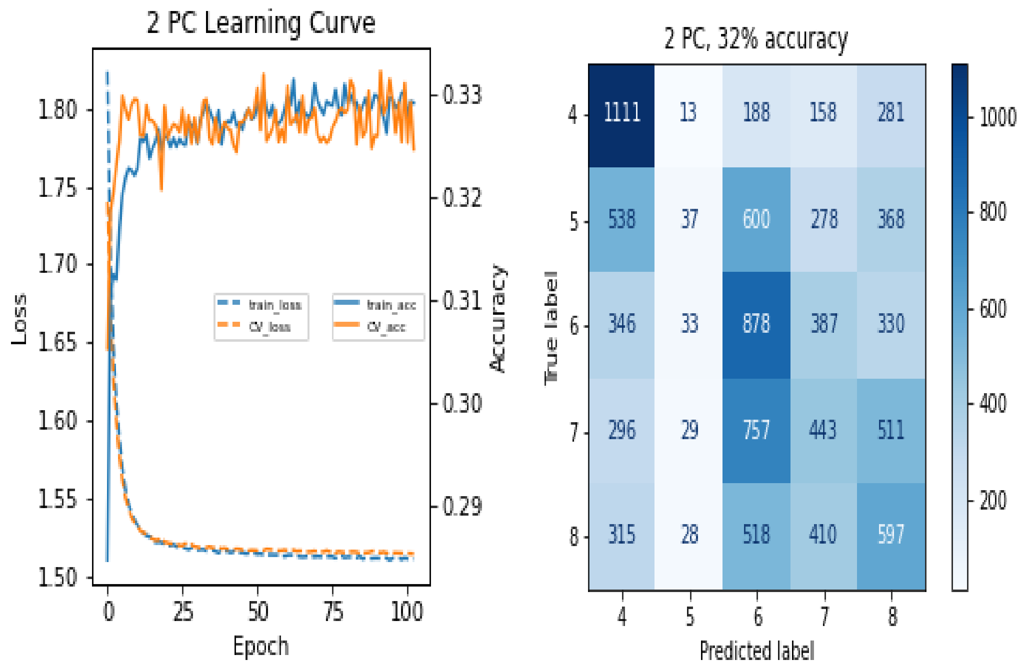
Table 4-19. Classification report for superstructure deterioration model with 9 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.81	0.84	0.83	1751
5	0.72	0.80	0.76	1821
6	0.78	0.62	0.69	1974
7	0.71	0.72	0.72	2036
8	0.77	0.81	0.79	1868

Accuracy			0.76	9450
Macro Average	0.76	0.76	0.76	9450
Weighted Average	0.76	0.76	0.75	9450

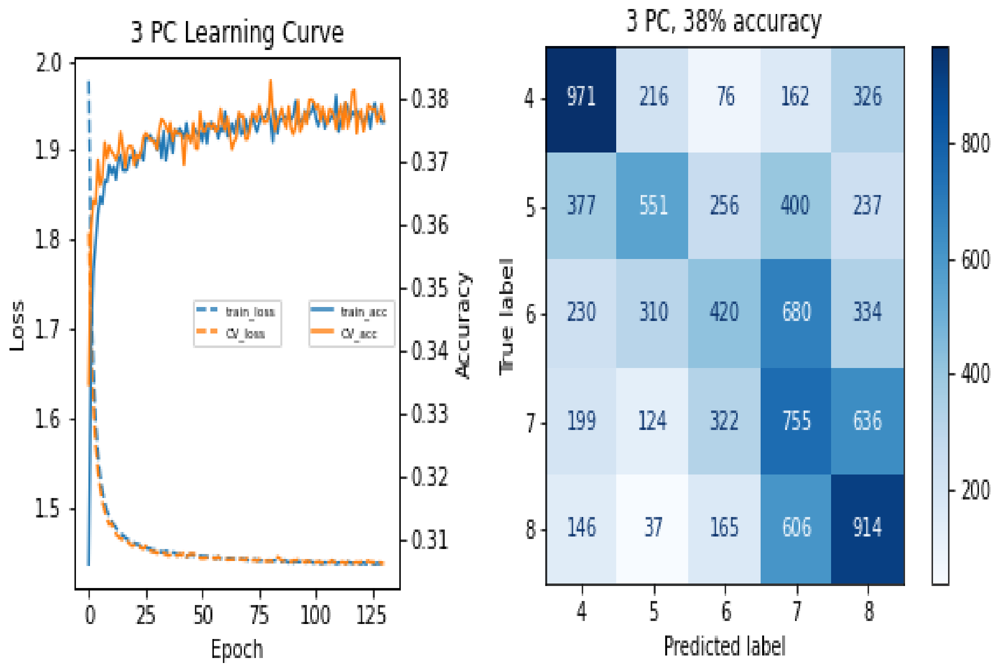
Table 4-20: Classification report for superstructure deterioration model with 15 features (Base Model).

Condition Rating	Recall	Precision	F1-score	Support
4	0.83	0.81	0.82	1751
5	0.67	0.81	0.73	1821
6	0.75	0.60	0.66	1974
7	0.73	0.68	0.71	2036
8	0.75	0.82	0.78	1868
Accuracy			0.74	9450
Macro Average	0.75	0.74	0.74	9450
Weighted Average	0.74	0.74	0.74	9450



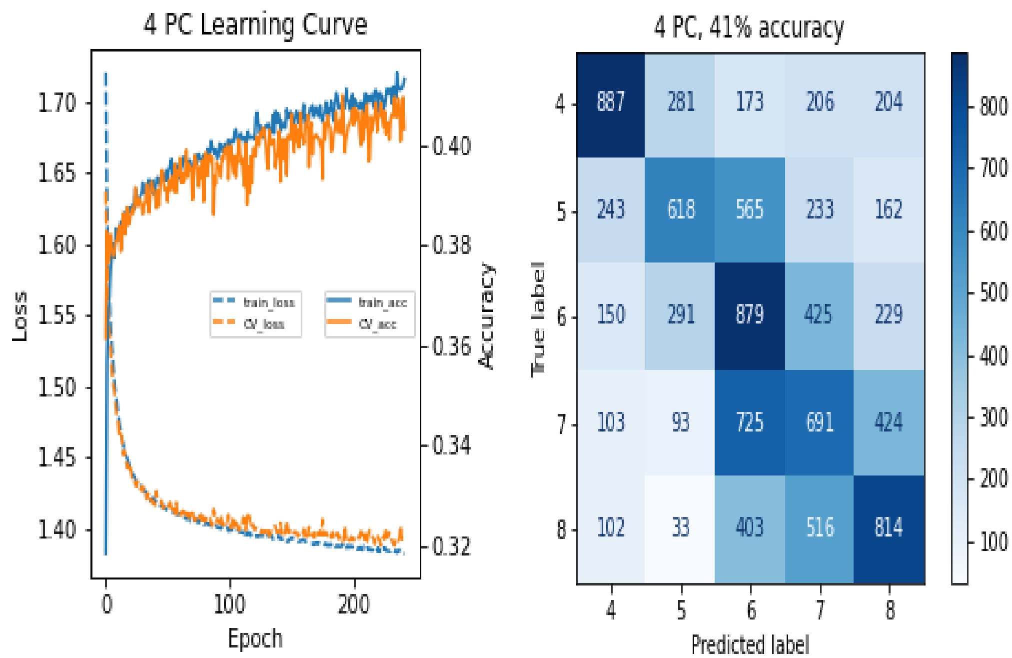
(a) (b)

Figure 4-15: (a) Learning curve for the superstructure model with 2 Principal Components (b) Confusion matrix for superstructure model 2 PC on the test set.



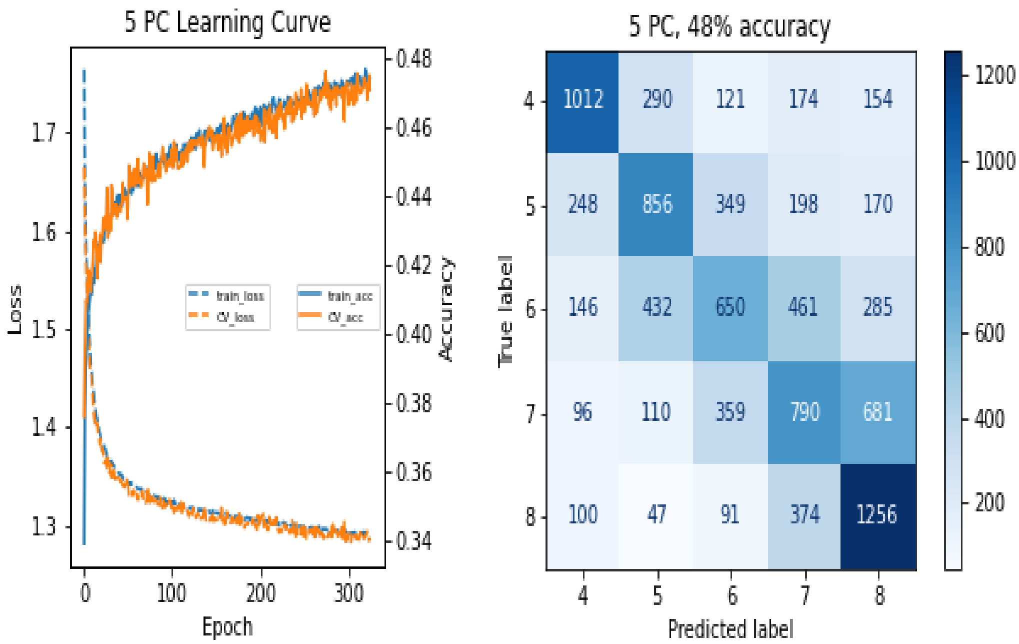
(a) (b)

Figure 4-16: (a) Learning curve for the superstructure model with 3 Principal Components (b) Confusion matrix for superstructure model 3 PC on the test set.



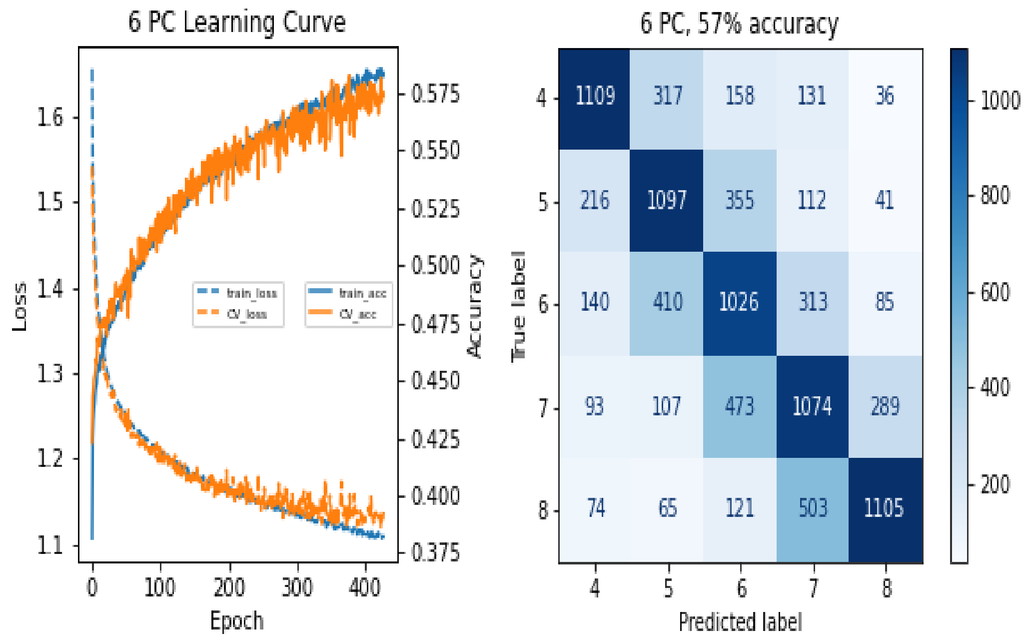
(a) (b)

Figure 4-17: (a) Learning curve for the superstructure model with 4 Principal Components (b) Confusion matrix for superstructure model 4 PC on the test set.



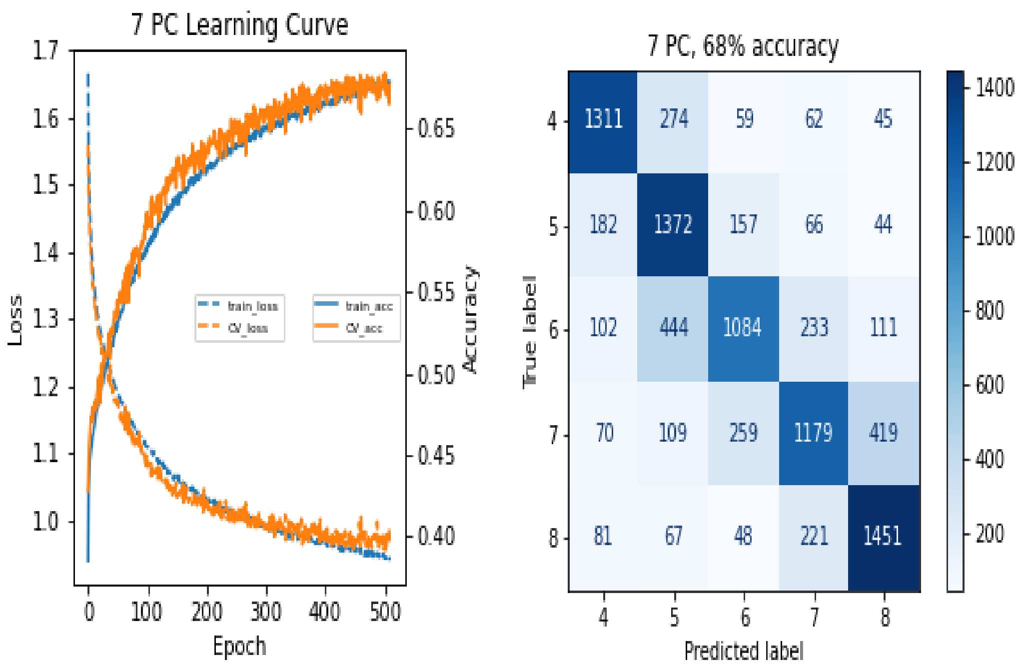
(a) (b)

Figure 4-18: (a) Learning curve for the superstructure model with 5 Principal Components (b) Confusion matrix for superstructure model 5 PC on the test set.



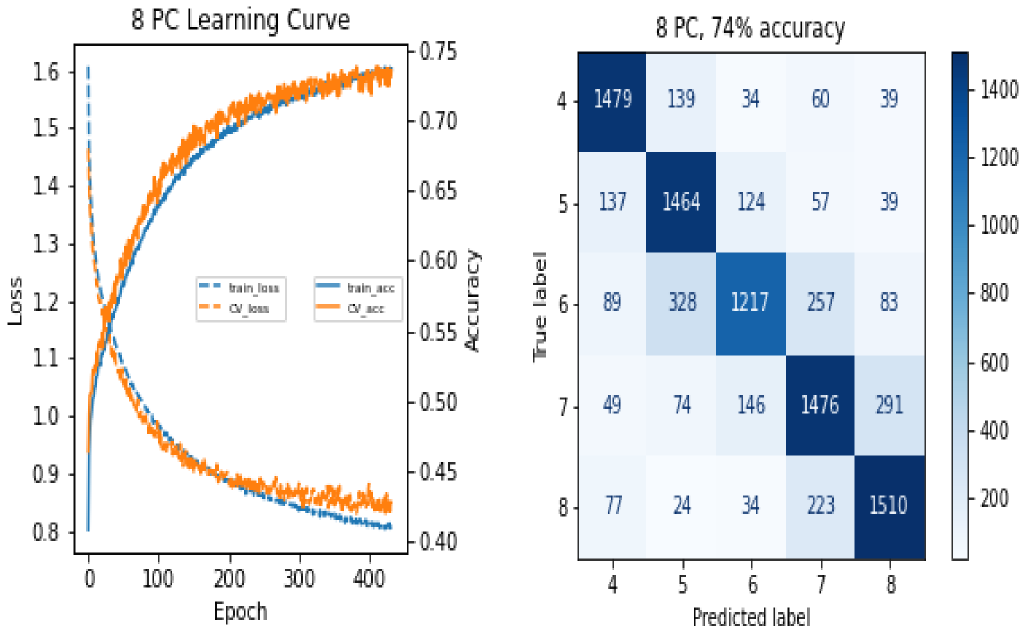
(a) (b)

Figure 4-19: (a) Learning curve for the superstructure model with 6 Principal Components (b) Confusion matrix for superstructure model 6 PC on the test set.



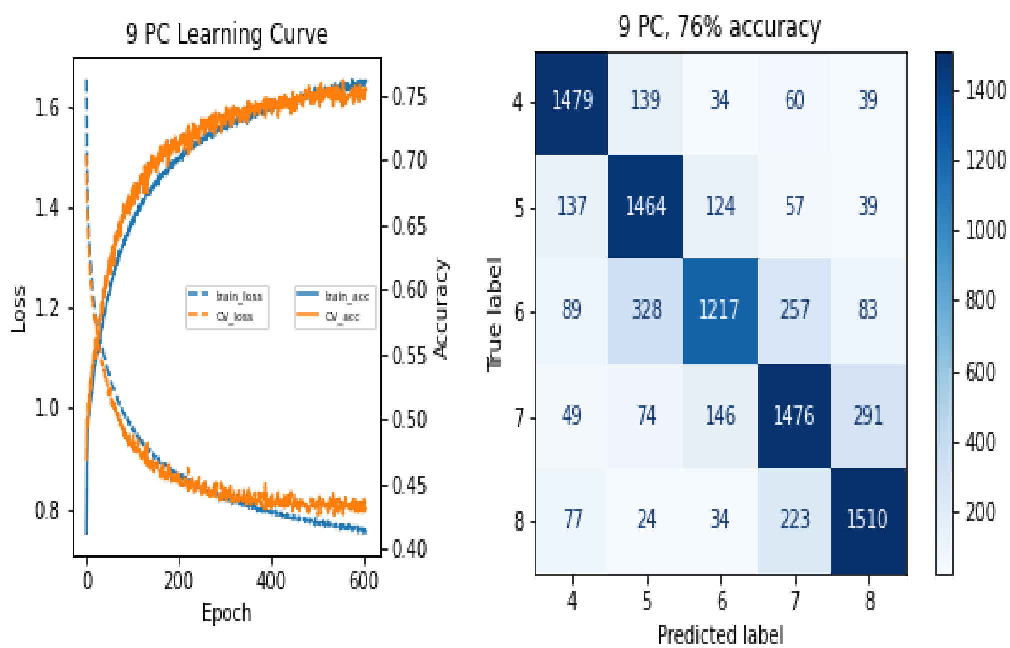
(a) (b)

Figure 4-20: (a) Learning curve for the superstructure model with 7 Principal Components (b) Confusion matrix for superstructure model 7 PC on the test set.



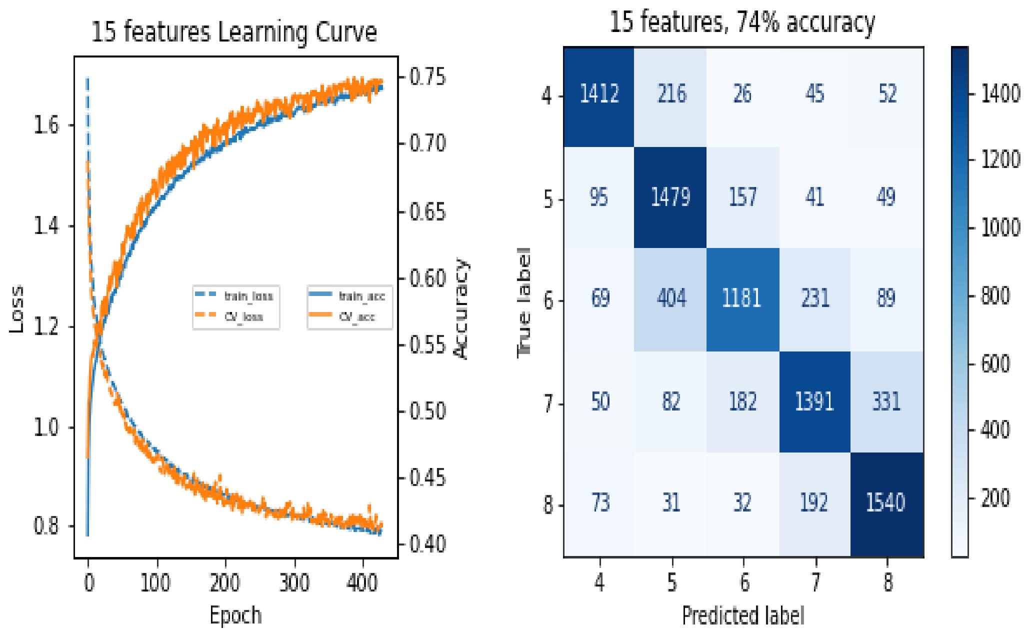
(a) (b)

Figure 4-21: (a) Learning curve for the superstructure model with 8 Principal Components (b) Confusion matrix for superstructure model 8 PC on the test set.



(a) (b)

Figure 4-22: (a) Learning curve for the superstructure model with 9 Principal Components (b) Confusion matrix for superstructure model 9 PC on the test set.



(a) (b)

Figure 4-23: (a) The learning curve for the superstructure base model with all features (b) Confusion matrix for the superstructure base model with all features on the test set.

4.1.5 Substructure PCA Result

The amount of variation in the bridge substructure feature set explained by each of the principal components and the cumulative values when multiple principal components (PC) are combined is shown in Table 4-21. The scree-plot in Figure 4-24 also shows the graphical trend of the cumulative values of the principal components.

Table 4-21: Variance explained by principal components for bridge substructure.

Principal Components	Variance Explained (%)	Cumulative (%)
1	27.82	27.82
2	15.50	43.32
3	10.70	54.02
4	7.86	61.88
5	7.09	68.97
6	6.84	75.81
7	6.37	82.18
8	5.27	87.45
9	4.40	91.85
10	3.43	95.28
11	1.46	96.74
12	1.14	97.88
13	1.09	98.97
14	1.03	100

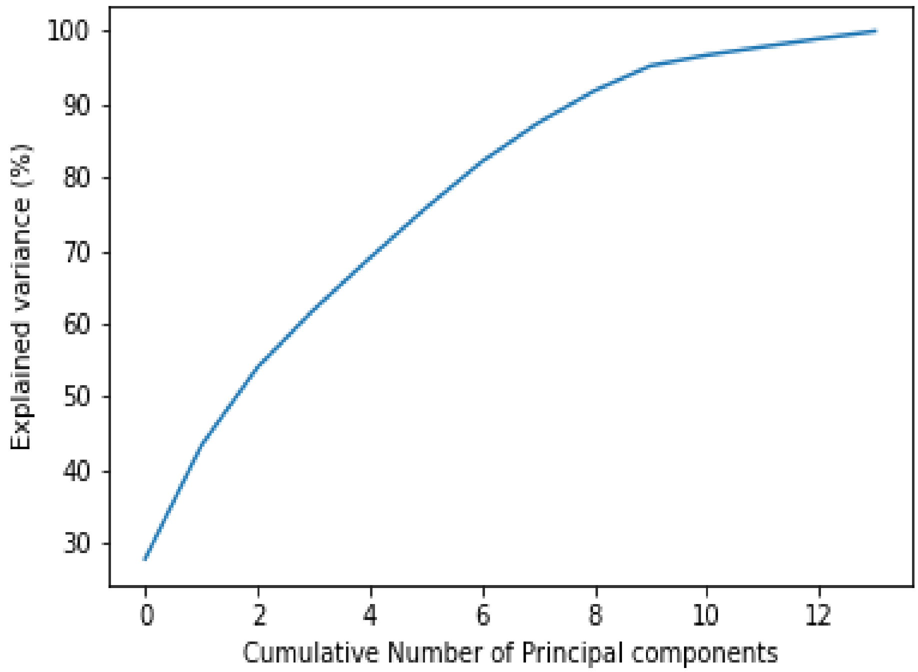


Figure 4-24: Scree plot for substructure PCA.

As with the normal trend for any principal component analysis, the amount of variance explained is highest in the first PC and reduces progressively in subsequent PCs. The cumulative amount of variance explained by all the PCs is equal to 100%. The correlation between the original substructure feature set and the principal components is shown in Figure 4-25.

As shown in Figure 4-25, PC1 is more influenced by the bridge roadway width. PC2 is more influenced by the number of days with measurable precipitation, PC3 is more influenced by the structure length, PC4 is more influenced by the total precipitation, PC5 is more influenced by the bridge age, PC6 is more influenced by the skew angle, PC7 is more influenced by the length of maximum span, PC8 is more influenced by the lanes under the structure, PC9 is more influenced by the lanes on the structure, and PC10 is more influenced by the structure length. As the scree-plot in Figure 4-24 flattens after the tenth principal component, it makes sense not to consider any principal component after this point because the added explained variance becomes insignificant. This result highlights the most important substructure features that hold the relevant information useful in predicting its future condition.

	PC-1	PC-2	PC-3	PC-4	PC-5	PC-6	PC-7	PC-8	PC-9	PC-10
Number of spans in main unit	0.131946	0.019307	0.467664	-0.32018	-0.34646	-0.06019	-0.51421	-0.08269	0.002808	0.514941
Structure length	0.14837	0.110226	0.622564	-0.10236	-0.17211	-0.02522	0.055608	0.148514	0.087022	-0.71014
Bridge age	-0.12265	-0.05013	-0.13612	-0.27048	-0.56924	-0.4639	0.521295	-0.27606	0.032004	0.028488
Skew	0.082891	0.103332	0.01514	0.447811	0.109627	-0.80874	-0.22657	0.218796	0.123847	0.027341
Bridge roadway width	0.407664	0.225067	-0.16104	-0.02443	0.013534	0.071242	0.005471	-0.11599	0.414668	0.012306
Length of maximum span	0.120695	0.117161	0.467233	0.216287	0.186575	0.084168	0.630702	0.200954	0.05893	0.471832
Lanes on the structure	0.377763	0.23051	-0.17673	-0.0582	-0.02402	0.083515	0.009839	-0.15427	0.557311	-0.00112
Lanes under the structure	0.259448	0.103498	0.178273	0.175866	0.261731	-0.10226	0.014362	-0.79191	-0.38034	-0.07121
Future average daily traffic	0.391468	0.222395	-0.17347	-0.05045	-0.14874	0.006477	0.033924	0.233835	-0.41032	0.009211
Average daily truck traffic	0.379487	0.228975	-0.19368	-0.05583	-0.16216	0.012703	0.041091	0.270225	-0.41937	-0.00212
Average temperature	0.269881	-0.40519	-0.00843	-0.35037	0.304022	-0.18068	0.075215	0.070024	-0.00318	-0.0075
Number of freeze-thaw cycles	-0.25493	0.439976	-0.00266	0.31958	-0.28006	0.15806	-0.05034	-0.09087	-0.00885	0.015038
Number of days with measurable precipitation	-0.28202	0.458592	-0.00081	-0.27589	0.204993	-0.10008	0.013115	0.028505	-0.02318	-0.01136
Total Precipitation	-0.1824	0.419653	-0.00284	-0.4743	0.385904	-0.17041	0.009812	0.024465	-0.02302	0.025035

Figure 4-25: Relationship between substructure features and principal components.

4.1.6 Substructure Deterioration Model Results

The learning curves and confusion matrix for the bridge substructure prediction models created for different sets of PCs are shown in Figure 4-26 to Figure 4-34 and that of the base model is shown in Figure 4-35. The learning curves show the trend in the loss (or error) and accuracy over a series of iterations (epochs) during the model training process. The confusion matrix shows the model performance on the difference condition rating classes in the test set.

Table 4-22 to

Table 4-30 shows the classification report for the PC models and Table 4-31 shows the classification report for the base model.

Like the PC models for the bridge deck and superstructure, the performance of the PC models for the bridge substructure improves with every added component until it reaches a point where there is no significant improvement in the F1-score of the models due to the limited amount of variation contributed by the added components. Also like the deck and superstructure condition prediction models, the best performance in the PC models and the base model (i.e., with 14 features) in terms of F1-score are observed in condition ratings 4 and 8 and the worst performance was observed in condition rating 6. This can also be attributed to the subjectivity of the data collection process where the condition ratings at the extreme points (i.e., condition ratings 4 and 8) are easily identifiable.

The model performance result shows that it takes up to 9 principal components corresponding to 91.85% of the variation in the dataset to match the performance of the base model (i.e., with 14 features) with both models having an accuracy and F1-score of 72% on the test set. Taking a step further to develop a prediction model with 10 principal components corresponding to 95.28% of the variation in the dataset results in a slightly better model with an accuracy and F1-score of 73%. As such, it takes 9 and 10 PCs to develop models that match and surpass the performance of the base model.

Table 4-22: Classification report for substructure deterioration model with 2 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.50	0.64	0.56	1904
5	0.33	0.16	0.21	2147
6	0.32	0.54	0.40	2189
7	0.30	0.17	0.22	2175
8	0.36	0.37	0.36	1935
Accuracy			0.37	10350
Macro Average	0.36	0.37	0.35	10350
Weighted Average	0.36	0.37	0.35	10350

Table 4-23: Classification report for substructure deterioration model with 3 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.57	0.61	0.59	1904

5	0.41	0.25	0.31	2147
6	0.37	0.32	0.34	2189
7	0.34	0.46	0.39	2175
8	0.45	0.52	0.48	1935
Accuracy			0.42	10350
Macro Average	0.43	0.43	0.42	10350
Weighted Average	0.42	0.42	0.42	10350

Table 4-24: Classification report for substructure deterioration model with 4 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.58	0.60	0.59	1904
5	0.43	0.22	0.29	2147
6	0.38	0.39	0.38	2189
7	0.37	0.44	0.40	2175
8	0.45	0.56	0.50	1935
Accuracy			0.44	10350
Macro Average	0.44	0.44	0.43	10350
Weighted Average	0.44	0.44	0.43	10350

Table 4-25: Classification report for substructure deterioration model with 5 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.60	0.67	0.63	1904
5	0.49	0.32	0.39	2147
6	0.44	0.44	0.44	2189
7	0.45	0.44	0.44	2175
8	0.56	0.73	0.63	1935
Accuracy			0.51	10350
Macro Average	0.51	0.52	0.51	10350
Weighted Average	0.50	0.51	0.50	10350

Table 4-26: Classification report for substructure deterioration model with 6 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.64	0.71	0.68	1904
5	0.50	0.30	0.38	2147
6	0.46	0.47	0.46	2189
7	0.47	0.59	0.53	2175

8	0.68	0.69	0.69	1935
Accuracy			0.55	10350
Macro Average	0.55	0.55	0.55	10350
Weighted Average	0.55	0.55	0.54	10350

Table 4-27: Classification report for substructure deterioration model with 7 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.65	0.81	0.72	1904
5	0.60	0.55	0.58	2147
6	0.62	0.54	0.57	2189
7	0.61	0.62	0.61	2175
8	0.76	0.75	0.76	1935
Accuracy			0.65	10350
Macro Average	0.65	0.65	0.65	10350
Weighted Average	0.65	0.65	0.64	10350

Table 4-28: Classification report for substructure deterioration model with 8 PCs.

Condition Rating	Recall	Precision	F1-score	Support

4	0.71	0.75	0.73	1904
5	0.63	0.58	0.61	2147
6	0.65	0.61	0.63	2189
7	0.67	0.68	0.67	2175
8	0.72	0.79	0.76	1935
Accuracy			0.68	10350
Macro Average	0.68	0.68	0.68	10350
Weighted Average	0.68	0.68	0.68	10350

Table 4-29: Classification report for substructure deterioration model with 9 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.71	0.82	0.76	1904
5	0.66	0.68	0.67	2147
6	0.73	0.62	0.67	2189
7	0.76	0.66	0.71	2175
8	0.75	0.84	0.79	1935
Accuracy			0.72	10350
Macro Average	0.72	0.72	0.72	10350

Weighted Average	0.72	0.72	0.72	10350

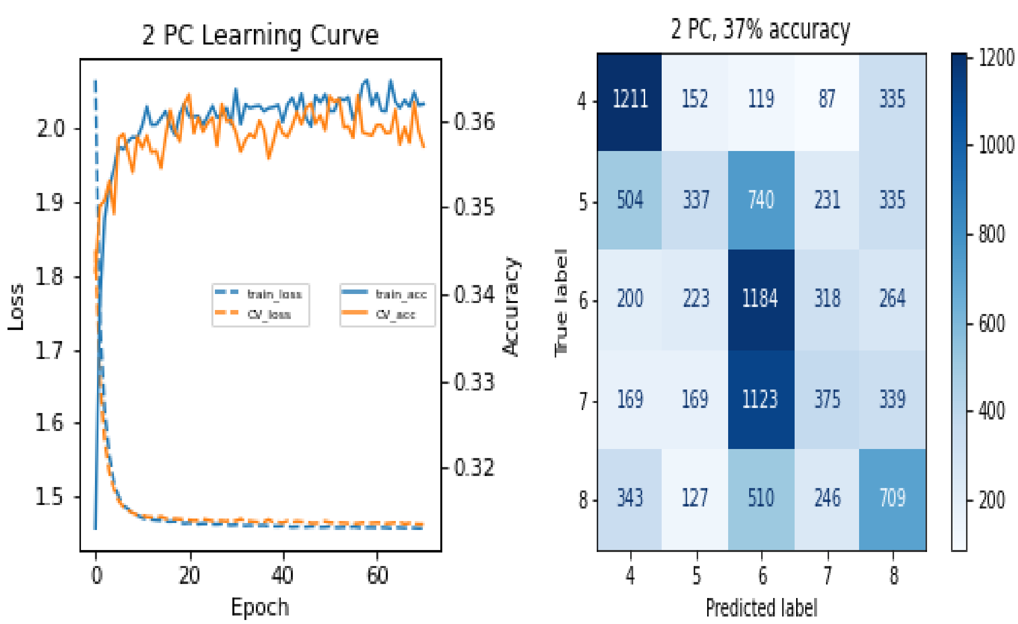
Table 4-30: Classification report for substructure deterioration model with 10 PCs.

Condition Rating	Recall	Precision	F1-score	Support
4	0.72	0.82	0.77	1904
5	0.69	0.64	0.67	2147
6	0.69	0.68	0.68	2189
7	0.74	0.69	0.72	2175
8	0.79	0.81	0.80	1935
Accuracy			0.73	10350
Macro Average	0.73	0.73	0.73	10350
Weighted Average	0.73	0.73	0.72	10350

Table 4-31. Classification report for substructure deterioration model with 14 features (Base Model).

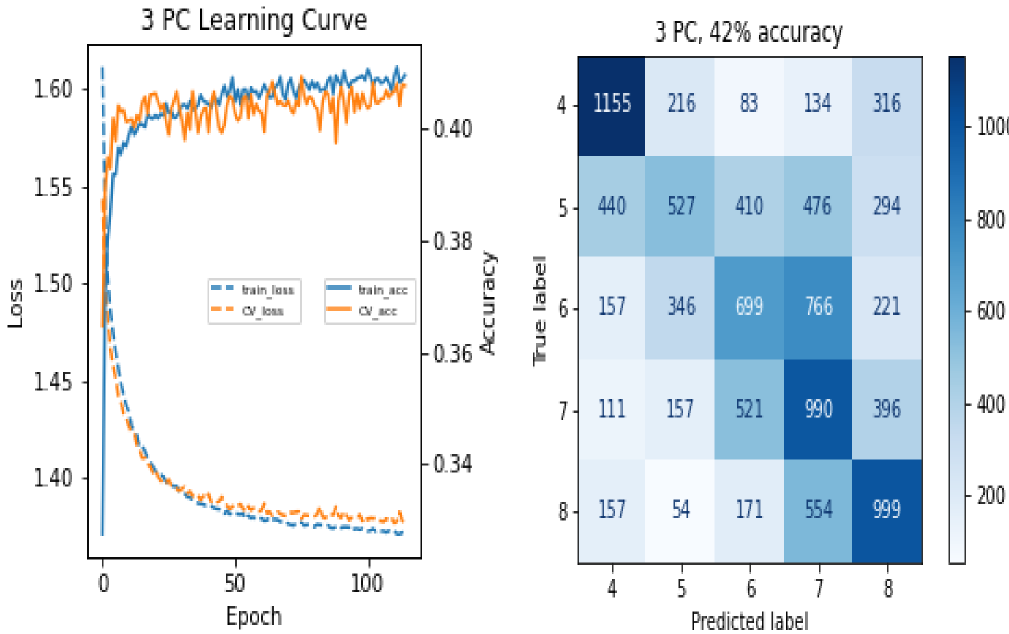
Condition Rating	Recall	Precision	F1-score	Support

4	0.73	0.82	0.77	1904
5	0.69	0.61	0.65	2147
6	0.74	0.62	0.67	2189
7	0.70	0.75	0.72	2175
8	0.76	0.85	0.80	1935
Accuracy			0.72	10350
Macro Average	0.72	0.73	0.72	10350
Weighted Average	0.72	0.72	0.72	10350



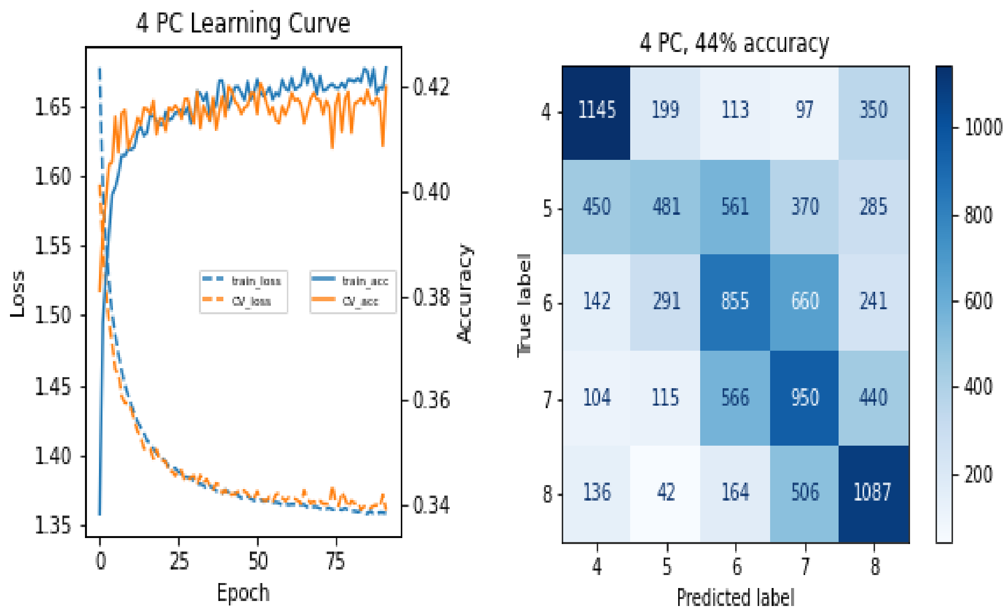
(a) (b)

Figure 4-26: (a) Learning curve for the substructure model with 2 Principal Components (b) Confusion matrix for substructure model 2 PC on the test set.



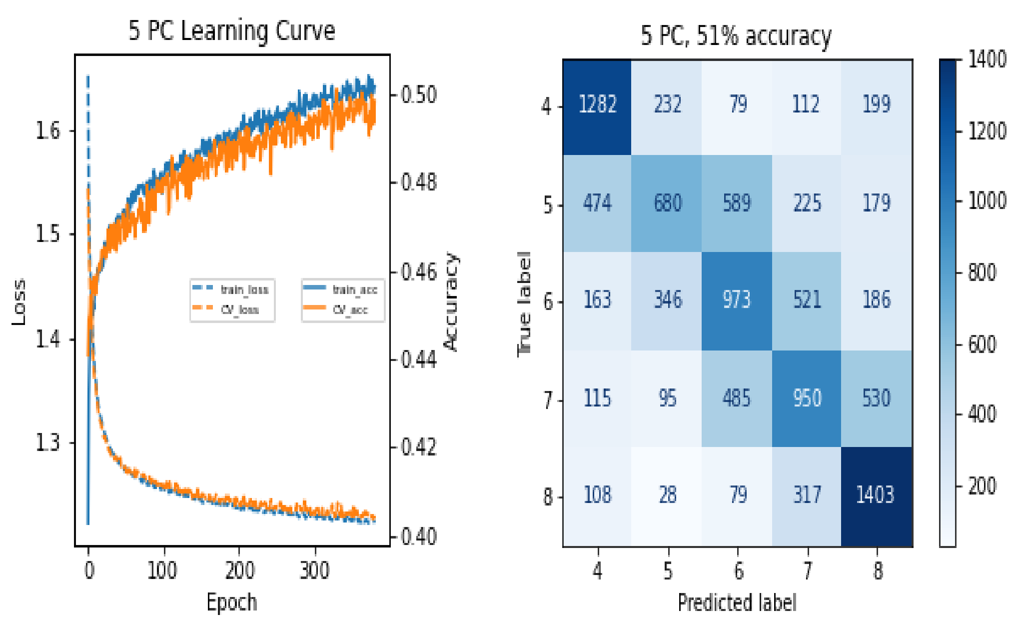
(a) (b)

Figure 4-27: (a) The learning curve for the substructure model with 3 Principal Components (b) Confusion matrix for substructure model 3 PC on the test set.



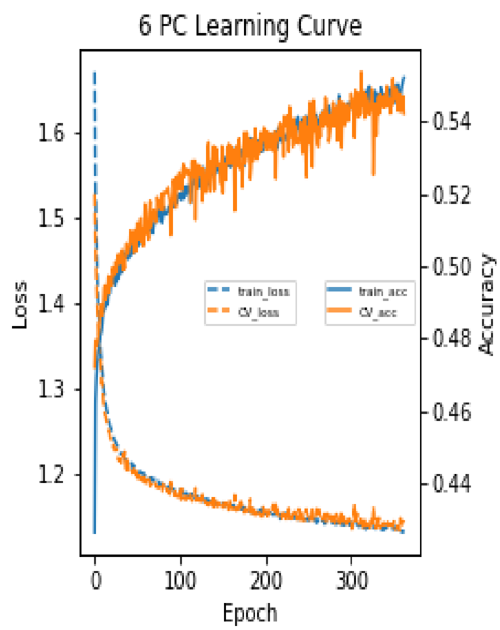
(a) (b)

Figure 4-28: (a) The learning curve for the substructure model with 4 Principal Components (b) Confusion matrix for substructure model 4 PC on the test set.

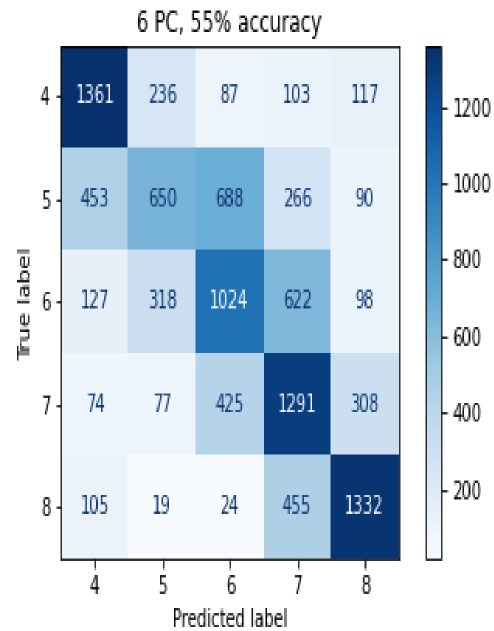


(a) (b)

Figure 4-29: (a) The learning curve for the substructure model with 5 Principal Components (b) Confusion matrix for substructure model 5 PC on the test set.

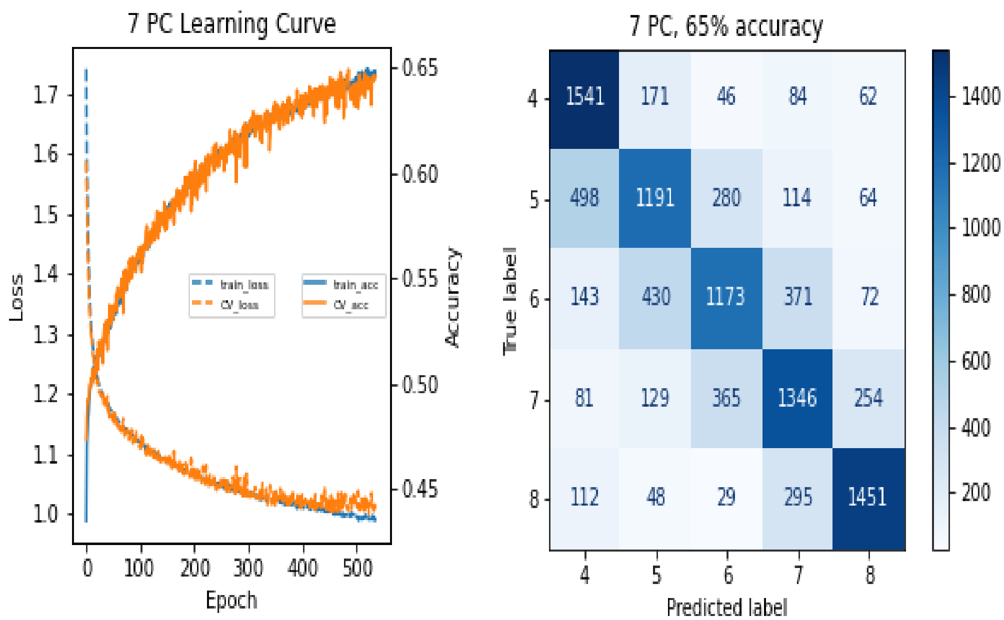


(a)



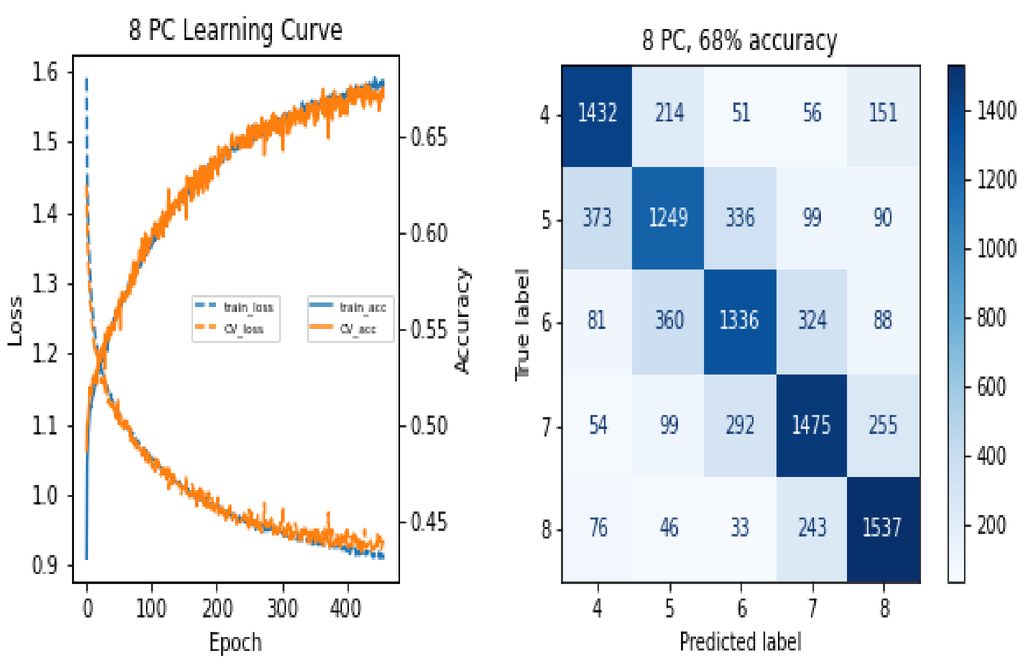
(b)

Figure 4-30: (a) The learning curve for the substructure model with 6 Principal Components (b) Confusion matrix for substructure model 6 PC on the test set.



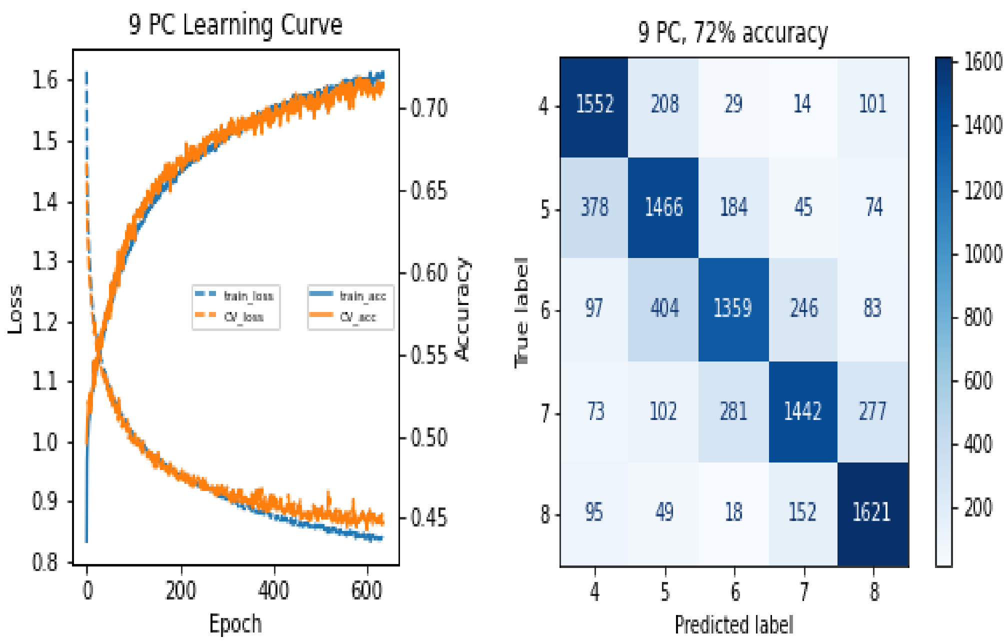
(a) (b)

Figure 4-31: (a) The learning curve for the substructure model with 7 Principal Components (b) Confusion matrix for substructure model 7 PC on the test set.



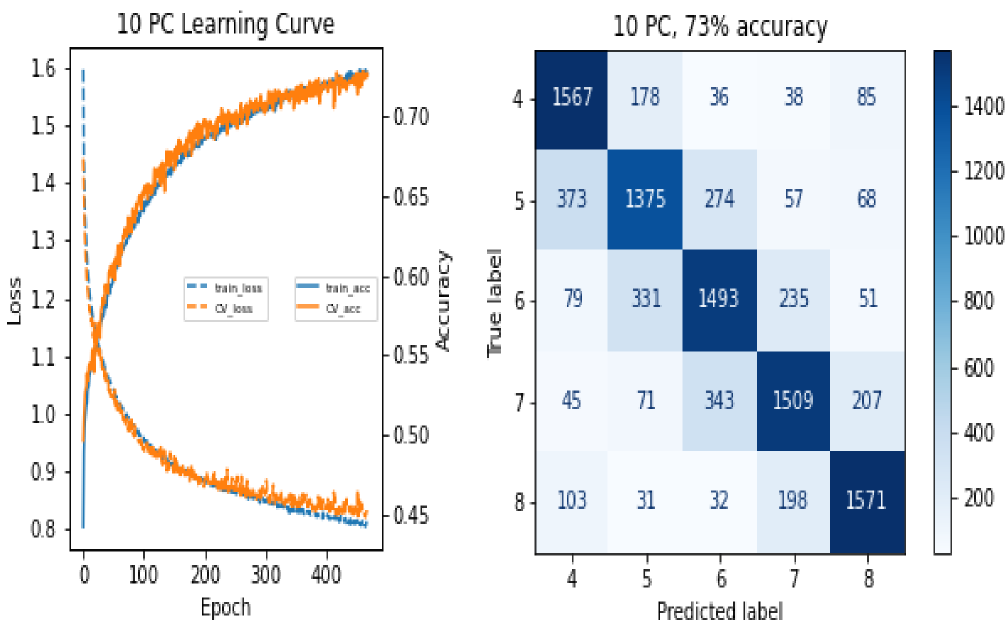
(a) (b)

Figure 4-32: (a) The learning curve for the substructure model with 8 Principal Components (b) Confusion matrix for substructure model 8 PC on the test set.



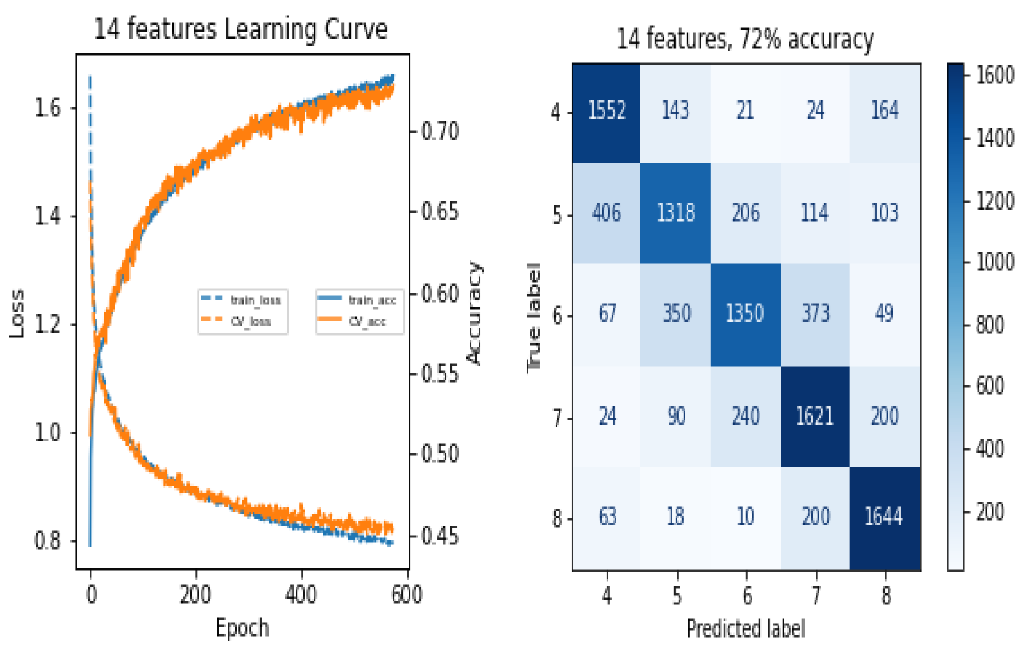
(a) (b)

Figure 4-33: (a) The learning curve for the substructure model with 9 Principal Components (b) Confusion matrix for substructure model 9 PC on the test set.



(a) (b)

Figure 4-34: (a) The learning curve for the substructure model with 10 Principal Components (b) Confusion matrix for substructure model 10 PC on the test set.



(a) (b)

Figure 4-35: (a) The learning curve for the substructure base model with all features (b) Confusion matrix for the substructure base model with all features on the test set.

4.2 Data Analysis Results for Bridge Weight

The result of the bridge elements' importance to the condition of the associated major component and the major components' importance to the overall condition of the bridge for the different bridge design types are discussed below. Figure 4-37 to Figure 4-66 show the elements', components', and resultant elements' weight for all the bridge design types. A total of 15 bridge categories were considered based on the main span design and the random forest algorithm was able to extract the importance of the bridge elements to the bridge components based on how the changes in the individual element health index influence the changes in the condition rating of the associated major component. The importance of the bridge components to the overall bridge rating is also evaluated based on how the changes in the individual component condition ratings influence the changes in the overall condition rating of the bridge. This process of evaluating the bridge elements' weight using the Random Forest (RF) algorithm is illustrated in the flow chart in Figure 4-36.

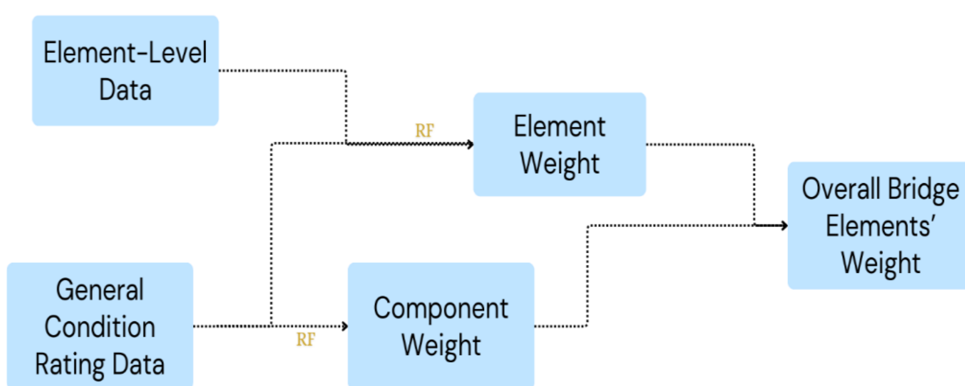


Figure 4-36: Bridge Elements' Weight Flow-Chart.

The analysis result for the Deck, Superstructure, Substructure, and Overall bridge condition is discussed below and it shows that the importance of bridge elements is dependent on the design type and the structural configuration of the bridge, and it is not constant all through. The result represents a total of 60 models created for the Deck, Superstructure, Substructure, and Overall bridge condition in each of the 8 bridge design types.

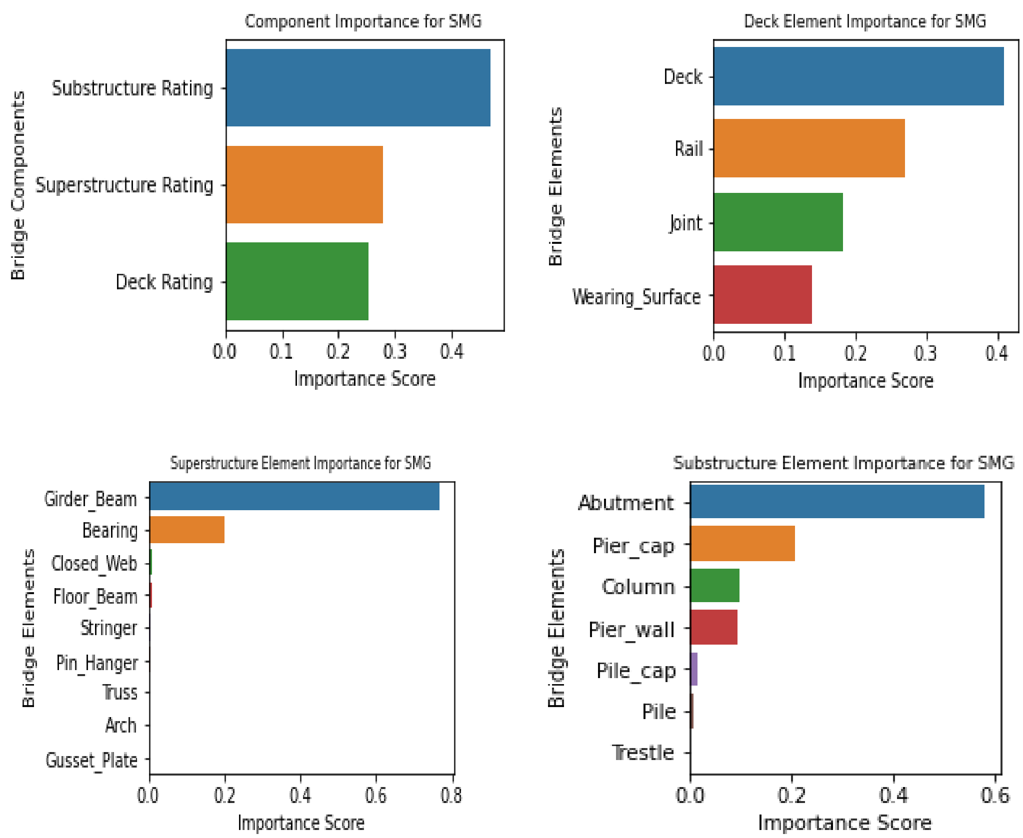


Figure 4-37: Bridge Components' and Elements' Importance for Stringer/Multi-beam or Girder Bridge.

Deck = 0.252856 * 0.409167 = 0.10346	Truss = 0.278476 * 0.001374 = 0.00038
Rail = 0.252856 * 0.270910 = 0.06850	Arch = 0.278476 * 0.000367 = 0.000102
Joint = 0.252856 * 0.181820 = 0.04597	Gusset_Plate = 0.278476 * 0.000066 = 0.000018
Wearing Surface = 0.252856 * 0.138103 = 0.03492	Abutment = 0.468668 * 0.581387 = 0.27248
Girder_Beam = 0.278476 * 0.768170 = 0.21392	Pier_Cap = 0.468668 * 0.205923 = 0.09651
Bearing = 0.278476 * 0.201852 = 0.05621	Column = 0.468668 * 0.095670 = 0.04484
Closed_Web = 0.278476 * 0.008746 = 0.00244	Pier_wall = 0.468668 * 0.095436 = 0.04473
Floor_Beam = 0.278476 * 0.007394 = 0.00206	Pile_cap = 0.468668 * 0.014562 = 0.00682
Stringer = 0.278476 * 0.006501 = 0.00181	Pile = 0.468668 * 0.006961 = 0.00326
Pin_Hanger 0.278476 * 0.005530 = 0.00154	Trestle = 0.468668 * 0.000060 = 0.000028

Figure 4-38: Bridge elements' weight for Stringer/Multi-beam or Girder Bridge.

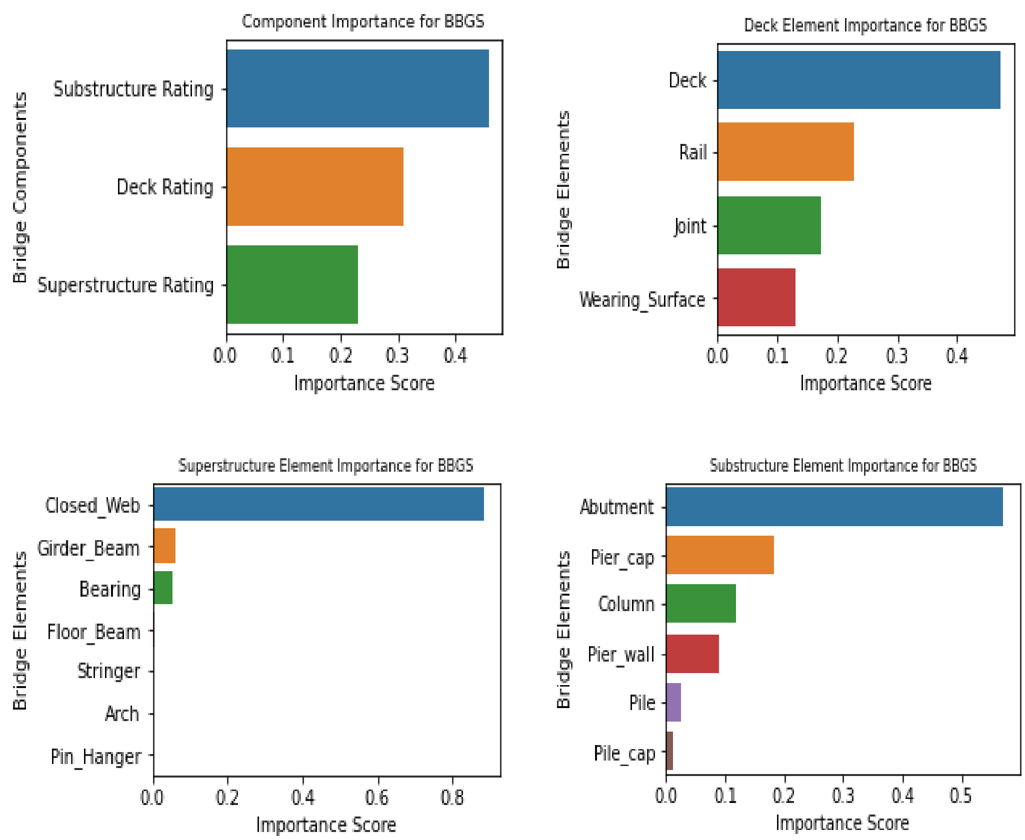


Figure 4-39: Bridge Components' and Elements' Importance for Box Beam or Girder (Single or Spread) Bridge.

Deck = 0.309813 * 0.471433 = 0.14606	Arch = 0.231429 * 0.000609 = 0.00014
Rail = 0.309813 * 0.227362 = 0.07044	Pin_Hanger = 0.231429 * 0.000341 = 0.00008
Joint = 0.309813 * 0.172073 = 0.05331	Abutment = 0.458758 * 0.570581 = 0.26176
Wearing Surface = 0.309813 * 0.129132 = 0.040	Pier_Cap = 0.458758 * 0.183619 = 0.08424
Closed_Web = 0.231429 * 0.883742 = 0.20452	Column = 0.458758 * 0.118686 = 0.05445
Girder_Beam = 0.231429 * 0.059058 = 0.01367	Pier_wall = 0.458758 * 0.090091 = 0.04133
Bearing = 0.231429 * 0.051558 = 0.01193	Pile = 0.458758 * 0.024148 = 0.01108
Floor_Beam = 0.231429 * 0.003405 = 0.00079	Pile_cap = 0.458758 * 0.012874 = 0.00591
Stringer = 0.231429 * 0.001287 = 0.00030	

Figure 4-40: Bridge Elements' Weight for Box Beam or Girders (Single) Bridge.

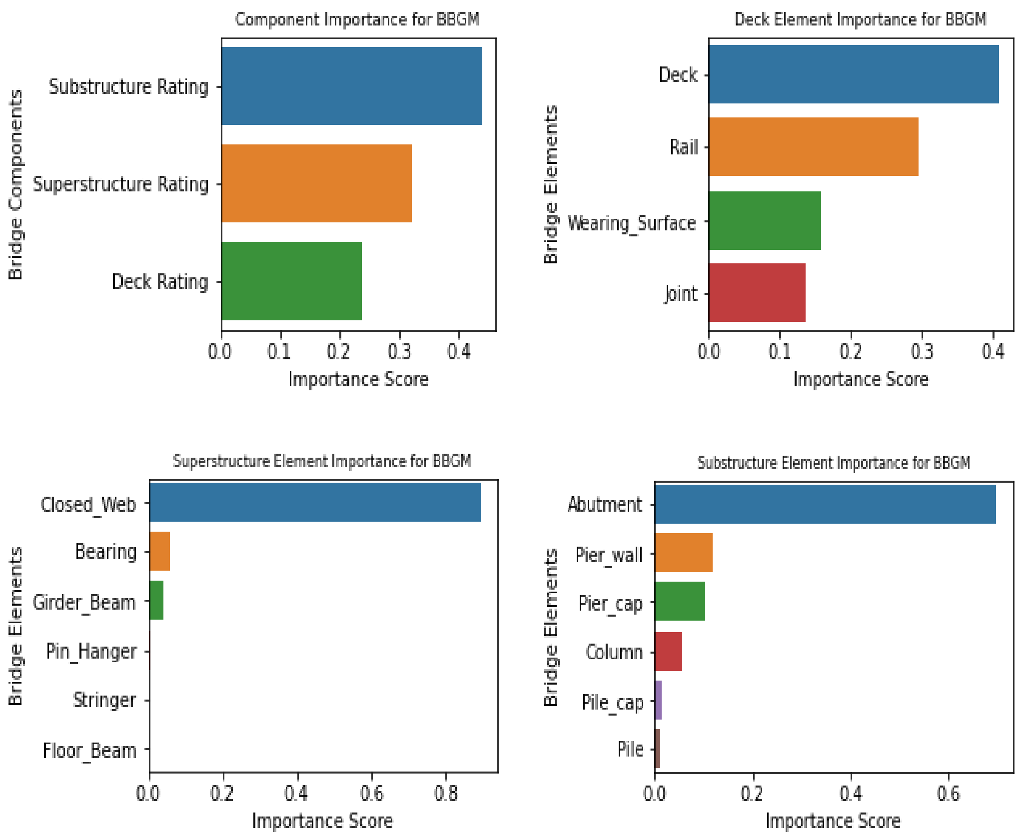


Figure 4-41: Bridge Components' and Elements' Importance for Box Beam or Girder (Multiple) Bridge.

Deck = 0.237403 * 0.408540 = 0.0970	Stringer = 0.322248 * 0.001191 = 0.00038
Rail = 0.237403 * 0.295769 = 0.07022	Floor_Beam = 0.322248 * 0.000967 = 0.00031
Wearing Surface = 0.237403 * 0.158042 = 0.03752	Abutment = 0.440349 * 0.696688 = 0.30679
Joint = 0.237403 * 0.137649 = 0.03268	Pier_wall = 0.440349 * 0.118259 = 0.05208
Closed_Web = 0.322248 * 0.895405 = 0.28854	Pier_Cap = 0.440349 * 0.104484 = 0.04601
Bearing = 0.322248 * 0.057420 = 0.01850	Column = 0.440349 * 0.054848 = 0.02415
Girder_Beam = 0.322248 * 0.041459 = 0.01336	Pile_cap = 0.440349 * 0.013770 = 0.00606
Pin_Hanger = 0.322248 * 0.003557 = 0.00115	Pile = 0.440349 * 0.011951 = 0.00526

Figure 4-42: Bridge Elements' Weight for Box Beam or Girders (Multiple) Bridge.

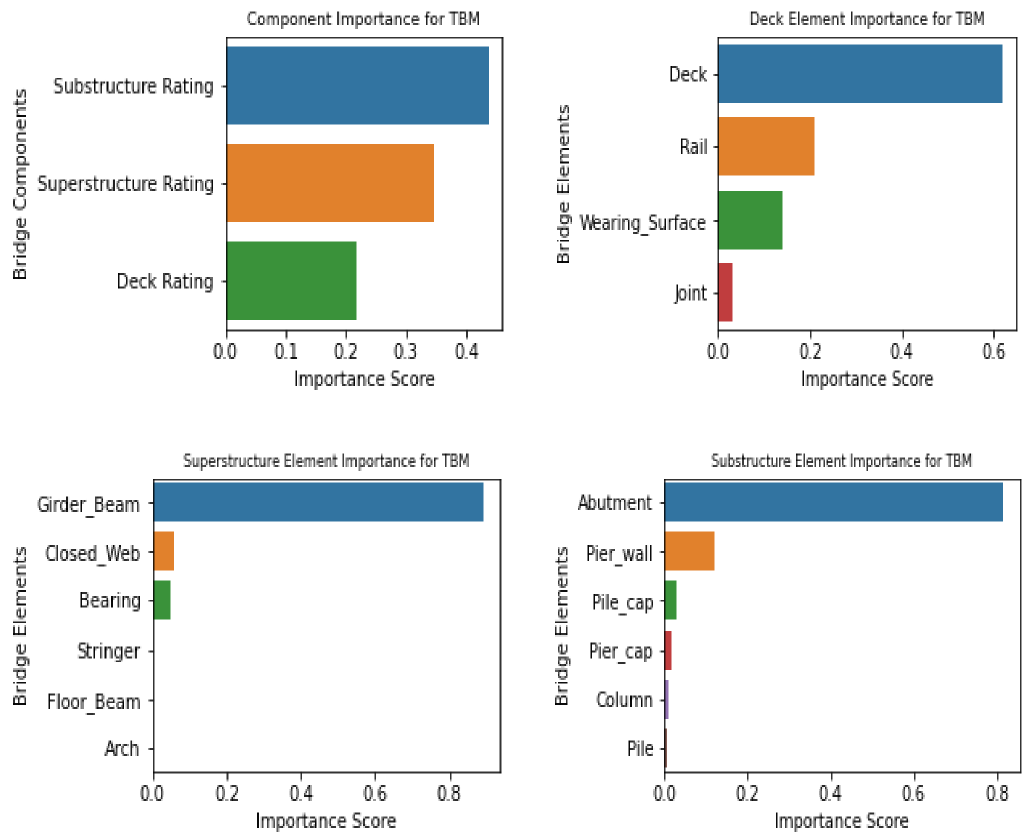


Figure 4-43: Bridge Components' and Elements' Importance for Tee Beam Bridge.

Deck = 0.217049 * 0.617944 = 0.134124	Floor_Beam = 0.346055 * 0.001750 = 0.00061
Rail = 0.217049 * 0.208895 = 0.04534	Arch = 0.346055 * 0.000428 = 0.00015
Wearing Surface = 0.217049 * 0.141752 = 0.03077	Abutment = 0.436896 * 0.815039 = 0.35609
Joint = 0.217049 * 0.031409 = 0.00682	Pier_wall = 0.436896 * 0.119790 = 0.05234
Girder_Beam = 0.346055 * 0.892473 = 0.30884	Pile_cap = 0.436896 * 0.030256 = 0.01322
Closed_Web = 0.346055 * 0.056558 = 0.01957	Pier_Cap = 0.436896 * 0.018262 = 0.00798
Bearing = 0.346055 * 0.046352 = 0.01604	Column = 0.436896 * 0.010584 = 0.00462
Stringer = 0.346055 * 0.002438 = 0.00084	Pile = 0.436896 * 0.006069 = 0.00265

Figure 4-44: Bridge Elements' Weight for Tee Beam Bridge.

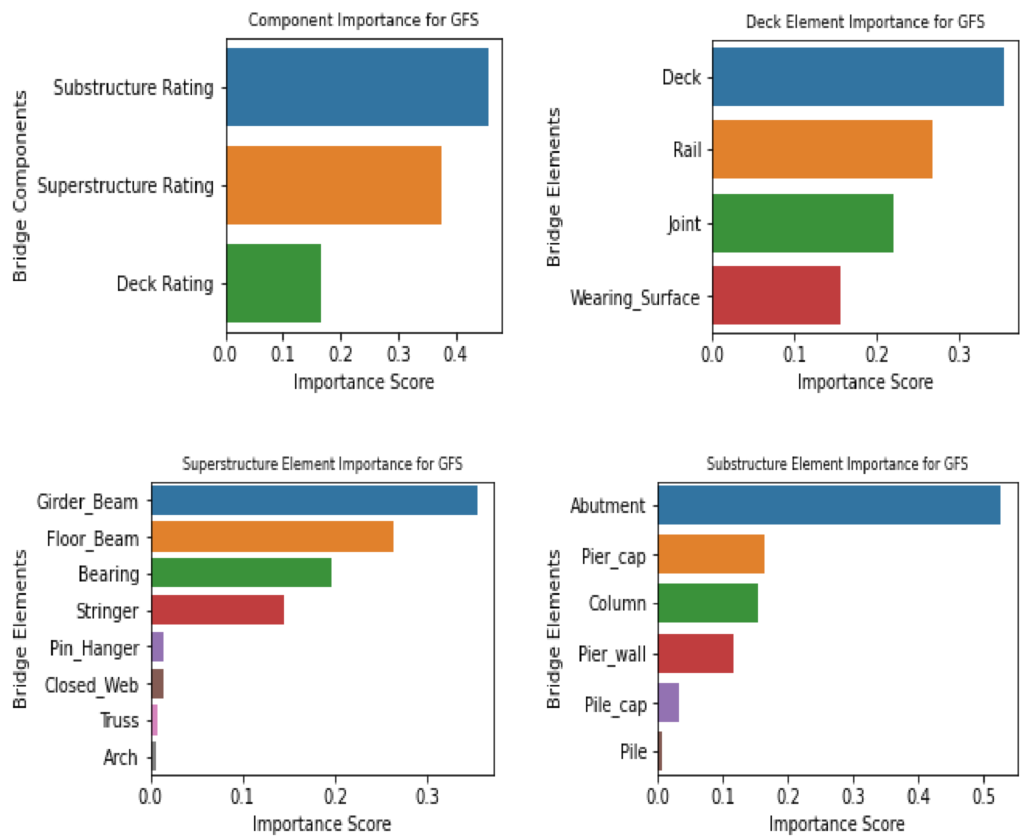


Figure 4-45: Bridge Components' and Elements' Importance for Girder and Floor Beam System Bridge.

Deck = 0.167062 * 0.354555 = 0.05923	Closed_Web = 0.376082 * 0.013841 = 0.00521
Rail = 0.167062 * 0.267067 = 0.04462	Truss = 0.376082 * 0.006854 = 0.00258
Joint = 0.167062 * 0.221188 = 0.03695	Arch = 0.376082 * 0.004695 = 0.00177
Wearing Surface = 0.167062 * 0.157189 = 0.02626	Abutment = 0.456856 * 0.525681 = 0.24016
Girder_Beam = 0.376082 * 0.354960 = 0.13349	Pier_Cap = 0.456856 * 0.163381 = 0.07464
Floor_Beam = 0.376082 * 0.263062 = 0.09893	Column = 0.456856 * 0.153842 = 0.07028
Bearing = 0.376082 * 0.197226 = 0.07417	Pier_wall = 0.456856 * 0.116977 = 0.05344
Stringer = 0.376082 * 0.145128 = 0.05458	Pile_cap = 0.456856 * 0.032567 = 0.01488
Pin_Hanger = 0.376082 * 0.014236 = 0.00535	Pile = 0.456856 * 0.007553 = 0.00345

Figure 4-46: Bridge Elements' Weight for Girder and Floor Beam System Bridge.

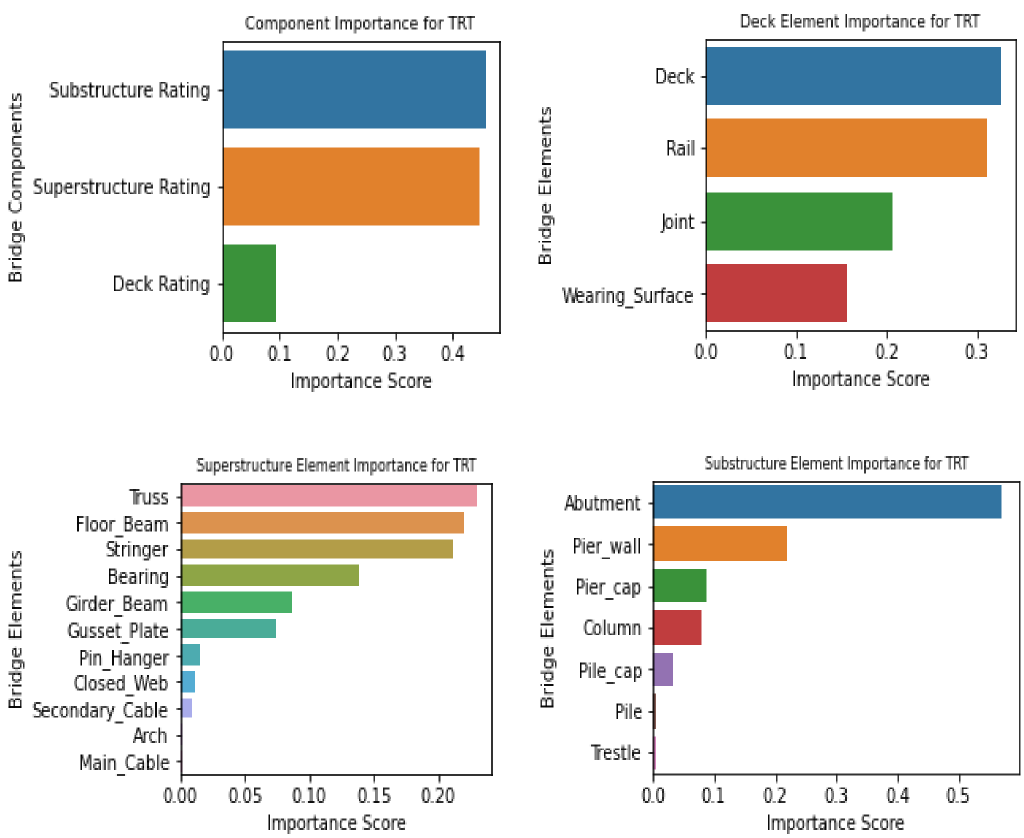


Figure 4-47: Bridge Components' and Elements' Importance for Truss-Thru Bridge.

Deck = 0.093431 * 0.326154 = 0.03047	Closed_Web = 0.447915 * 0.011854 = 0.00531
Rail = 0.093431 * 0.311272 = 0.02908	Secondary_Cable = 0.447915 * 0.008984 = 0.00402
Joint = 0.093431 * 0.206828 = 0.01932	Arch = 0.361292 * 0.001813 = 0.00081
Wearing Surface = 0.093431 * 0.155746 = 0.01455	Main_Cable = 0.447915 * 0.001277 = 0.00057
Truss = 0.447915 * 0.229825 = 0.10294	Abutment = 0.458654 * 0.571313 = 0.26203
Floor_Beam = 0.447915 * 0.219803 = 0.09845	Pier_wall = 0.458654 * 0.218309 = 0.10013
Stringer = 0.447915 * 0.211791 = 0.09486	Pier_Cap = 0.458654 * 0.088544 = 0.04061
Bearing = 0.447915 * 0.138638 = 0.0621	Column = 0.458654 * 0.079274 = 0.03636
Girder_Beam = 0.447915 * 0.086352 = 0.03868	Pile_cap = 0.458654 * 0.033958 = 0.01557
Gusset_Plate = 0.447915 * 0.074618 = 0.03342	Pile = 0.458654 * 0.004811 = 0.00221
Pin_Hanger = 0.447915 * 0.015046 = 0.00674	Trestle = 0.458654 * 0.003792 = 0.00174

Figure 4-48: Bridge Elements' Weight for Truss-Thru Bridge.

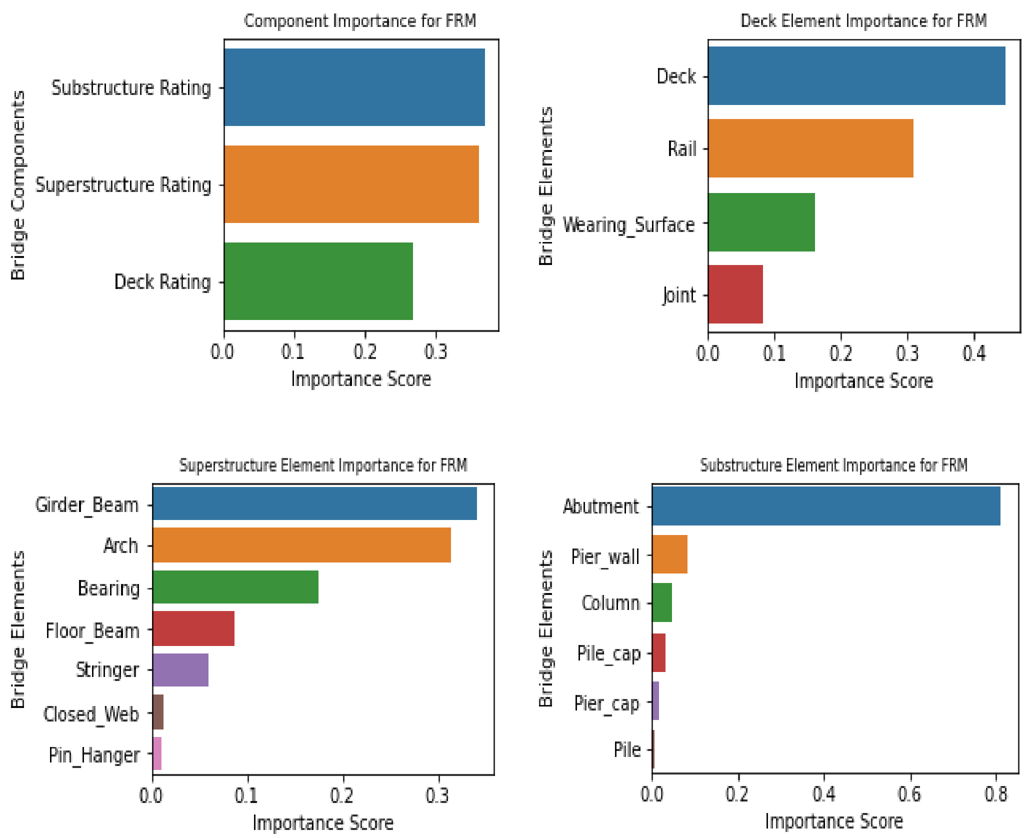


Figure 4-49: Bridge Components' and Elements' Importance for Frame Bridge.

Deck = 0.268546 * 0.446640 = 0.11994	Closed_Web = 0.361292 * 0.012505 = 0.00452
Rail = 0.268546 * 0.309893 = 0.08322	Pin_Hanger = 0.361292 * 0.011309 = 0.00409
Wearing Surface = 0.268546 * 0.160812 = 0.04319	Abutment = 0.370163 * 0.811563 = 0.30041
Joint = 0.268546 * 0.082655 = 0.02220	Pier_wall = 0.370163 * 0.084397 = 0.03124
Girder_Beam = 0.361292 * 0.341797 = 0.12349	Column = 0.370163 * 0.045999 = 0.01703
Arch = 0.361292 * 0.314719 = 0.11371	Pile_cap = 0.370163 * 0.033421 = 0.01237
Bearing = 0.361292 * 0.174338 = 0.06299	Pier_Cap = 0.370163 * 0.016785 = 0.00621
Floor_Beam = 0.361292 * 0.086109 = 0.03111	Pile = 0.370163 * 0.007836 = 0.0029
Stringer = 0.361292 * 0.059224 = 0.02140	

Figure 4-50: Bridge Elements' Weight for Frame Bridge.

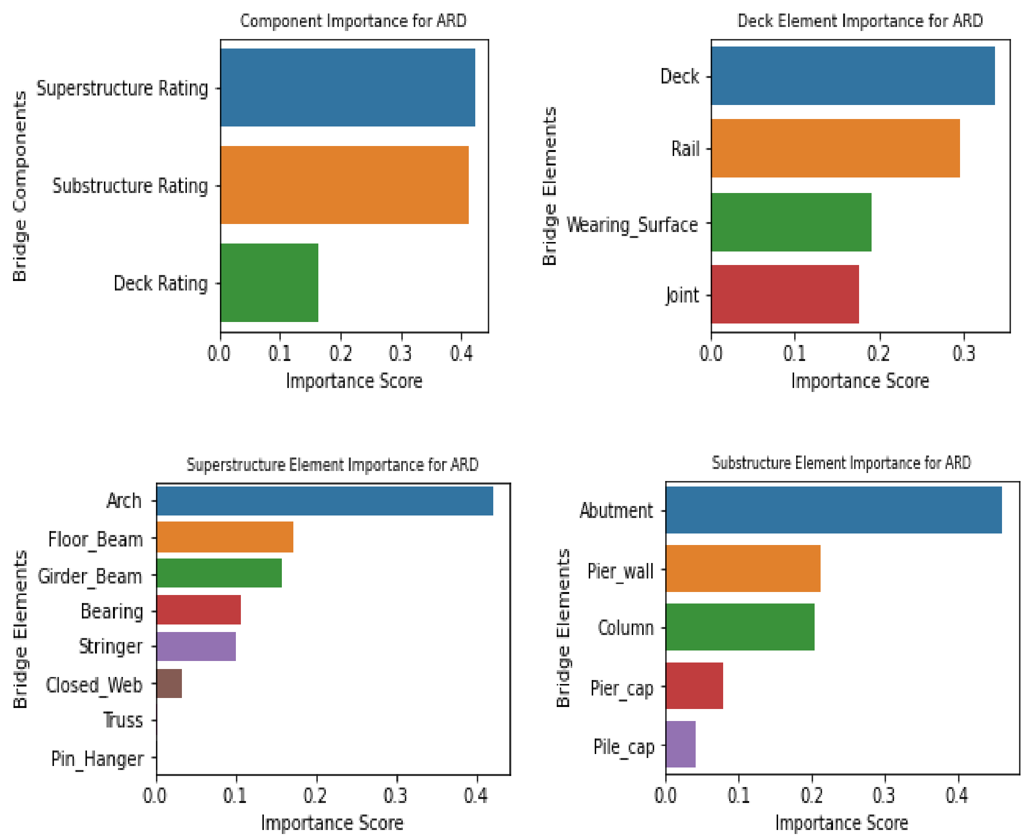


Figure 4-51: Bridge Components' and Elements' Importance for Arch-Deck Bridge.

Deck = 0.163804 * 0.337001 = 0.05520	Closed_Web = 0.423095 * 0.033435 = 0.01415
Rail = 0.163804 * 0.295880 = 0.04847	Truss = 0.423095 * 0.003403 = 0.00144
Wearing Surface = 0.163804 * 0.190458 = 0.0312	Pin_Hanger = 0.423095 * 0.002987 = 0.00126
Joint = 0.163804 * 0.176661 = 0.02894	Abutment = 0.413101 * 0.461340 = 0.19058
Arch = 0.423095 * 0.420784 = 0.17803	Pier_wall = 0.413101 * 0.212868 = 0.08794
Floor_Beam = 0.423095 * 0.172137 = 0.07283	Column = 0.413101 * 0.204196 = 0.08435
Girder_Beam = 0.423095 * 0.158431 = 0.06703	Pier_Cap = 0.413101 * 0.079520 = 0.03285
Bearing = 0.423095 * 0.107463 = 0.04547	Pile_cap = 0.413101 * 0.042076 = 0.01738
Stringer = 0.423095 * 0.101360 = 0.04288	

Figure 4-52: Bridge Elements' Weight for Arch Deck Bridge.

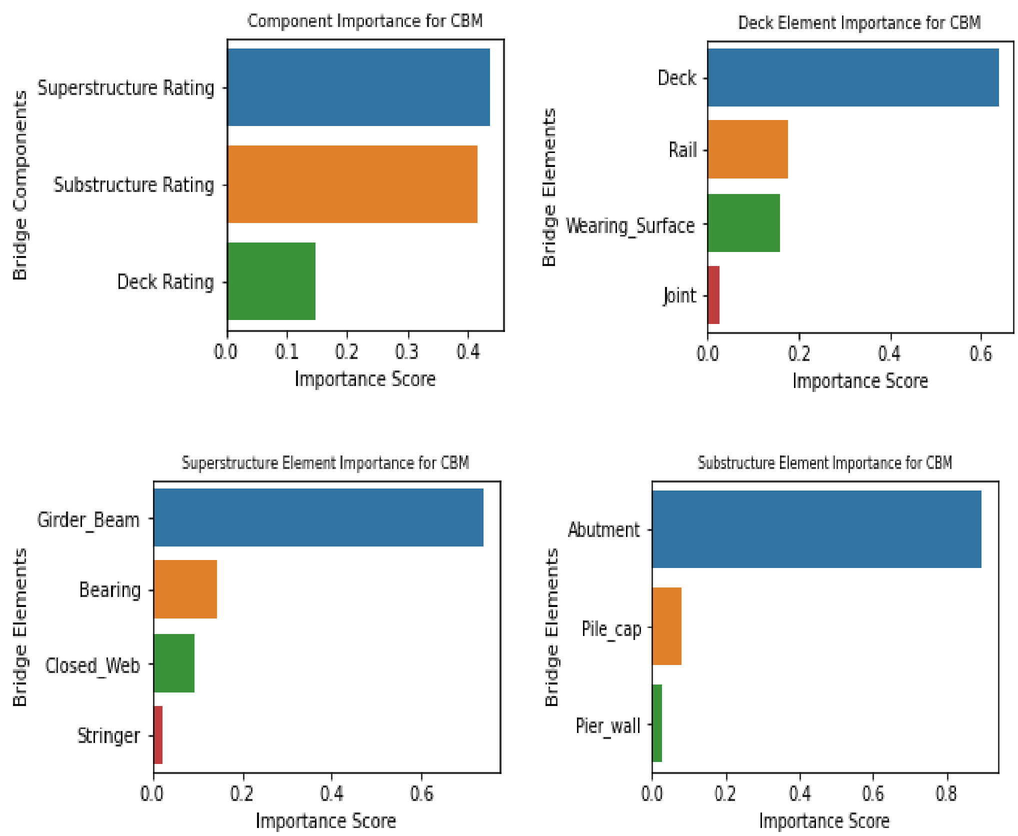


Figure 4-53: Bridge Components' and Elements' Importance for Channel Beam Bridge.

Deck = 0.147642 * 0.639281 = 0.09438	Closed_Web = 0.436833 * 0.093297 = 0.04076
Rail = 0.147642 * 0.175059 = 0.02585	Stringer = 0.436833 * 0.023293 = 0.01018
Wearing Surface = 0.147642 * 0.159576 = 0.02356	Abutment = 0.415525 * 0.895140 = 0.37195
Joint = 0.147642 * 0.026084 = 0.00385	Pile_cap = 0.415525 * 0.079058 = 0.03285
Girder_Beam = 0.436833 * 0.739374 = 0.32298	Pier_wall = 0.415525 * 0.025802 = 0.01072
Bearing = 0.436833 * 0.144036 = 0.06292	

Figure 4-54: Bridge Elements' Weight for Channel Beam Bridge.

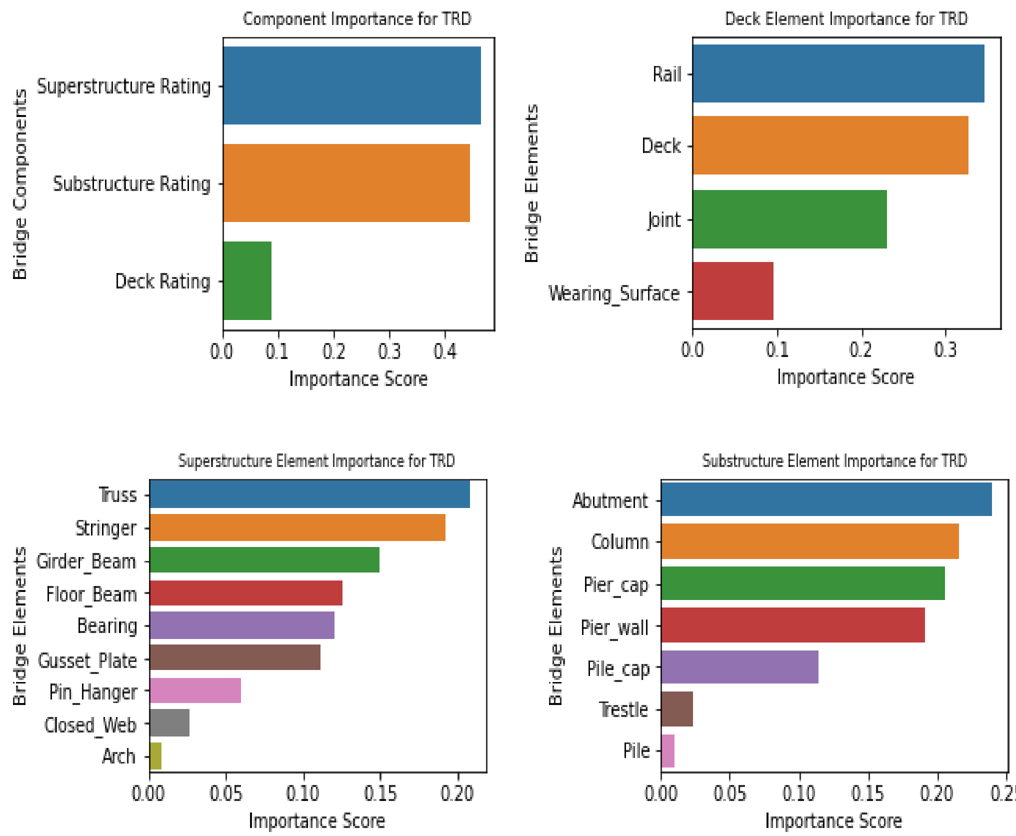


Figure 4-55: Bridge Components' and Elements' Importance for Truss-Deck Bridge.

Rail = 0.090237 * 0.346677 = 0.03128	Pin_Hanger = 0.465151 * 0.060051 = 0.02793
Deck = 0.090237 * 0.326751 = 0.02949	Closed_Web = 0.465151 * 0.026240 = 0.01221
Joint = 0.090237 * 0.229691 = 0.02073	Arch = 0.465151 * 0.008071 = 0.00375
Wearing Surface = 0.090237 * 0.096880 = 0.00874	Abutment = 0.444613 * 0.239122 = 0.10632
Truss = 0.465151 * 0.207904 = 0.09671	Column = 0.444613 * 0.215777 = 0.09594
Stringer = 0.465151 * 0.191477 = 0.08907	Pier_Cap = 0.444613 * 0.206100 = 0.09163
Girder_Beam = 0.465151 * 0.149079 = 0.06934	Pier_wall = 0.444613 * 0.190914 = 0.08488
Floor_Beam = 0.465151 * 0.125605 = 0.05843	Pile_cap = 0.444613 * 0.113890 = 0.05063
Bearing = 0.465151 * 0.120639 = 0.05612	Trestle = 0.444613 * 0.023876 = 0.01062
Gusset_Plate = 0.465151 * 0.110935 = 0.05160	Pile = 0.444613 * 0.010319 = 0.00459

Figure 4-56: Bridge Elements' Weight for Truss-Deck Bridge.

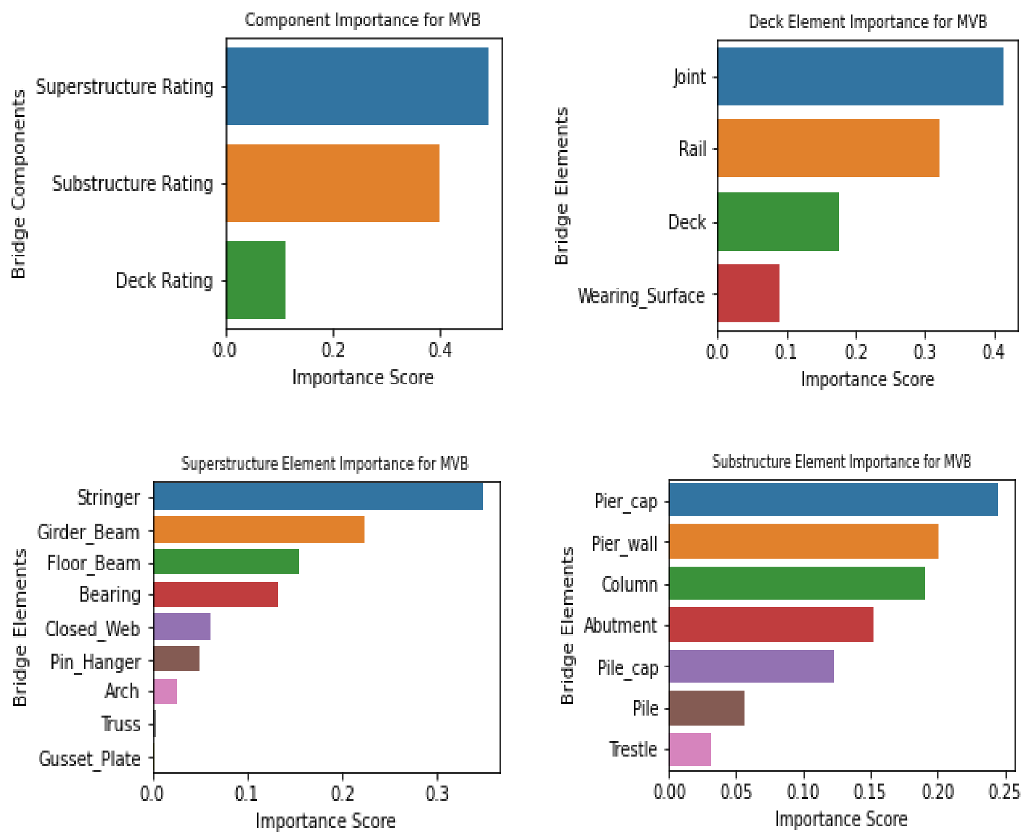


Figure 4-57: Bridge Components' and Elements' Importance for Movable Bascule Bridge.

Joint = 0.111167 * 0.413309 = 0.04595	Arch = 0.489437 * 0.024955 = 0.01221
Rail = 0.111167 * 0.321285 = 0.03572	Truss = 0.489437 * 0.003215 = 0.00157
Deck = 0.111167 * 0.174864 = 0.01944	Gusset_Plate = 0.489437 * 0.001722 = 0.00084
Wearing Surface = 0.111167 * 0.090541 = 0.01007	Pier_Cap = 0.399396 * 0.244430 = 0.09762
Stringer = 0.489437 * 0.348665 = 0.17065	Pier_wall = 0.399396 * 0.201227 = 0.08037
Girder_Beam = 0.489437 * 0.223611 = 0.10944	Column = 0.399396 * 0.190823 = 0.07621
Floor_Beam = 0.489437 * 0.153940 = 0.07534	Abutment = 0.399396 * 0.152668 = 0.06097
Bearing = 0.489437 * 0.132091 = 0.06465	Pile_cap = 0.399396 * 0.122993 = 0.04912
Closed_Web = 0.489437 * 0.062107 = 0.03040	Pile = 0.399396 * 0.056713 = 0.02265
Pin_Hanger = 0.489437 * 0.049693 = 0.02432	Trestle = 0.399396 * 0.031147 = 0.01244

Figure 4-58: Bridge Elements' Weight for Movable Bascule Bridge.

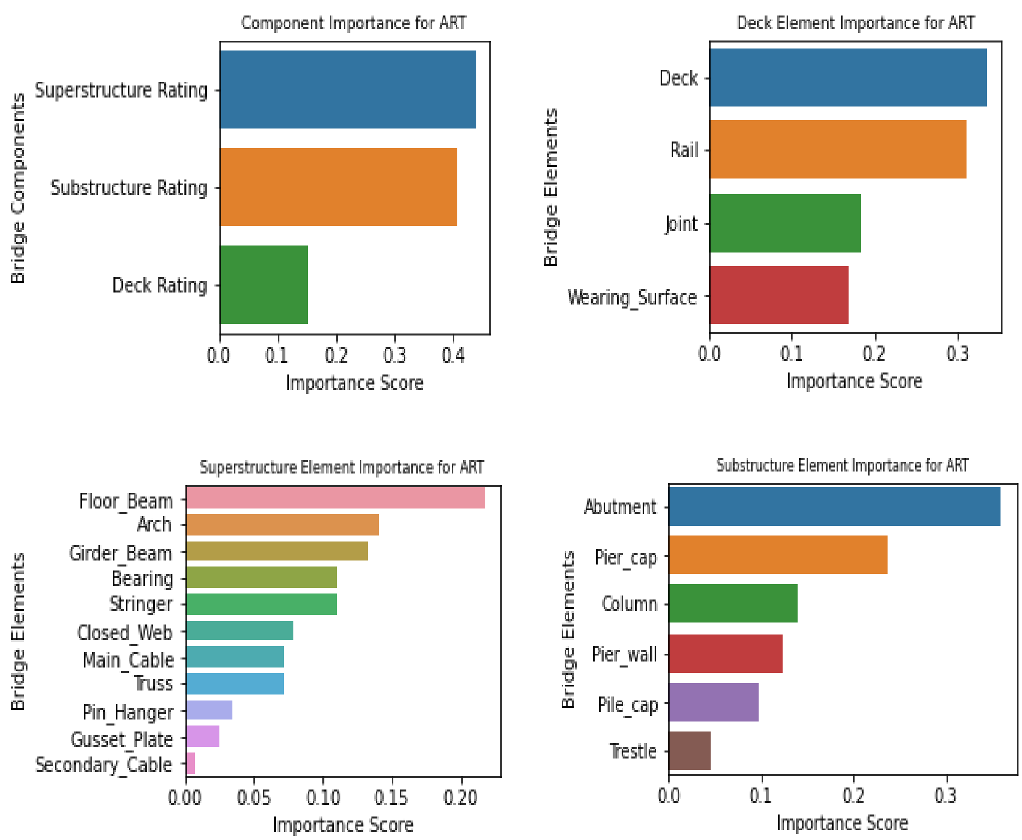


Figure 4-59: Bridge Components' and Elements' Importance for Arch-Thru Bridge.

Deck = 0.152176 * 0.336546 = 0.05121	Truss = 0.439962 * 0.071750 = 0.03157
Rail = 0.152176 * 0.311629 = 0.04742	Pin_Hanger = 0.439962 * 0.034045 = 0.01498
Joint = 0.152176 * 0.183564 = 0.02793	Gusset_Plate = 0.439962 * 0.024469 = 0.01077
Wearing Surface = 0.152176 * 0.168262 = 0.02561	Secondary_Cable = 0.439962 * 0.006783 = 0.00298
Floor_Beam = 0.439962 * 0.217960 = 0.09589	Abutment = 0.407862 * 0.359547 = 0.14665
Arch = 0.439962 * 0.140860 = 0.06197	Pier_Cap = 0.407862 * 0.236460 = 0.09644
Girder_Beam = 0.439962 * 0.132391 = 0.05825	Column = 0.407862 * 0.138955 = 0.05667
Bearing = 0.439962 * 0.110836 = 0.04876	Pier_wall = 0.407862 * 0.122704 = 0.05005
Stringer = 0.439962 * 0.110768 = 0.04873	Pile_cap = 0.407862 * 0.097090 = 0.03960
Closed_Web = 0.439962 * 0.078283 = 0.03444	Trestle = 0.407862 * 0.045243 = 0.01845
Main_Cable = 0.439962 * 0.071855 = 0.03161	

Figure 4-60: Bridge Elements' Weight for Arch-Thru.

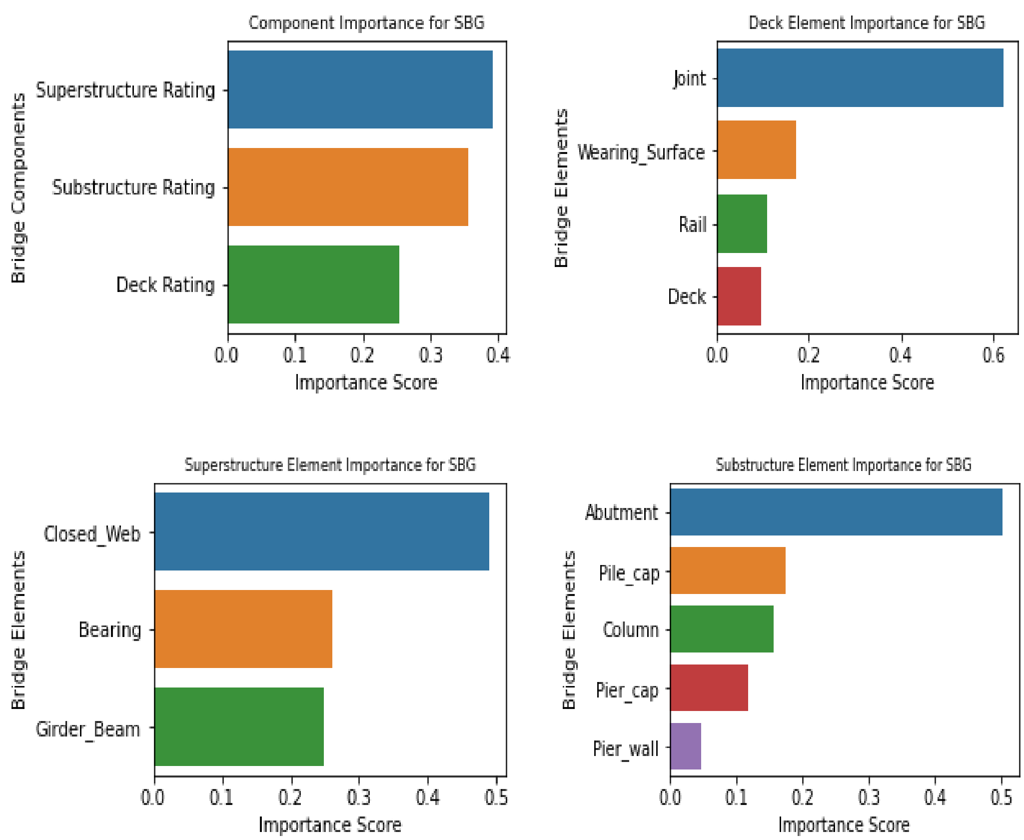


Figure 4-61: Bridge Components' and Elements' Importance for Segmental Box Girder Bridge.

Joint = 0.253318 * 0.622655 = 0.15773	Girder_Beam = 0.391754 * 0.249300 = 0.09766
Wearing Surface = 0.253318 * 0.172229 = 0.04363	Abutment = 0.354928 * 0.502784 = 0.17845
Rail = 0.253318 * 0.109027 = 0.02762	Pile_cap = 0.354928 * 0.175147 = 0.06216
Deck = 0.253318 * 0.096090 = 0.02434	Column = 0.354928 * 0.156627 = 0.05559
Closed_Web = 0.391754 * 0.490114 = 0.192	Pier_Cap = 0.354928 * 0.117972 = 0.04187
Bearing = 0.391754 * 0.260586 = 0.10209	Pier_wall = 0.354928 * 0.047471 = 0.01685

Figure 4-62: Bridge Elements' Weight for Segmental Box Girder Bridge.

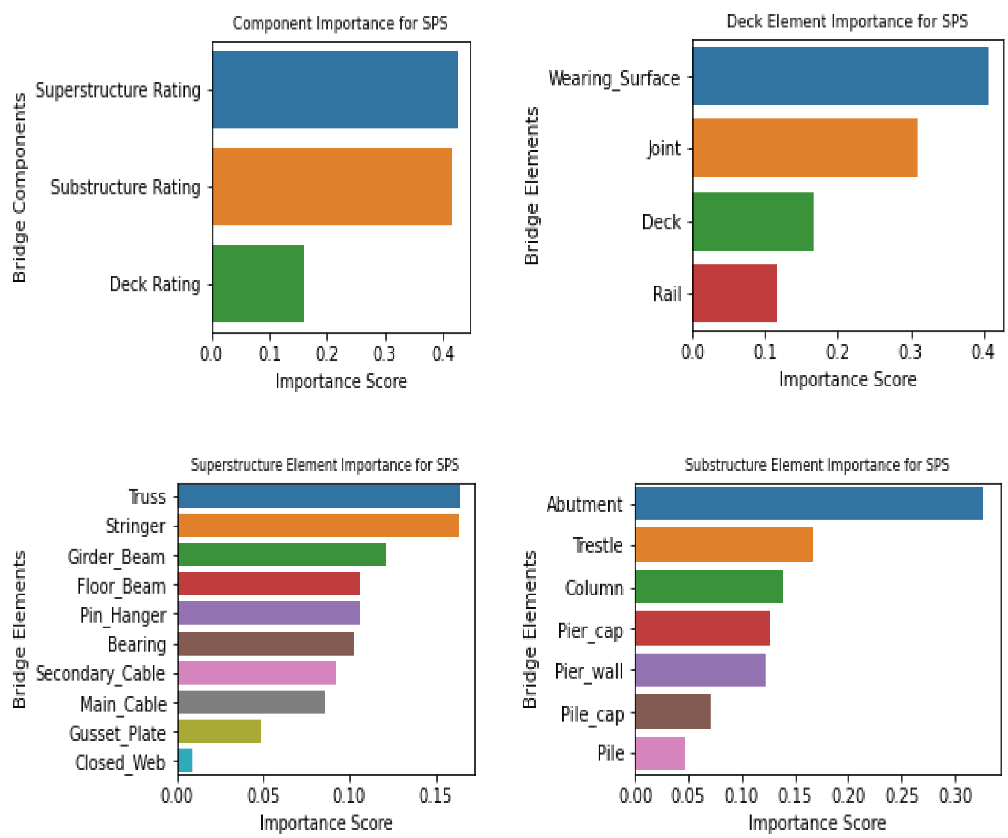


Figure 4-63: Bridge Components' and Elements' Importance for Suspension Bridge.

Wearing Surface = 0.158923 * 0.406812 = 0.06465	Main_Cable = 0.425471 * 0.085674 = 0.03645
Joint = 0.158923 * 0.309558 = 0.0492	Gusset_Plate = 0.425471 * 0.048677 = 0.02071
Deck = 0.158923 * 0.167541 = 0.02663	Closed_Web = 0.425471 * 0.008692 = 0.0037
Rail = 0.158923 * 0.116089 = 0.01845	Abutment = 0.415607 * 0.326665 = 0.13576
Truss = 0.425471 * 0.164385 = 0.06994	Trestle = 0.415607 * 0.167103 = 0.06945
Stringer = 0.425471 * 0.164019 = 0.06979	Column = 0.415607 * 0.138246 = 0.05746
Girder_Beam = 0.425471 * 0.121143 = 0.05154	Pier_Cap = 0.415607 * 0.126622 = 0.05262
Floor_Beam = 0.425471 * 0.106228 = 0.0452	Pier_wall = 0.415607 * 0.122992 = 0.05112
Pin_Hanger = 0.425471 * 0.105837 = 0.04503	Pile_cap = 0.415607 * 0.071201 = 0.02959
Bearing = 0.425471 * 0.102918 = 0.04379	Pile = 0.415607 * 0.047170 = 0.01960
Secondary_Cable = 0.425471 * 0.092427 = 0.03933	

Figure 4-64: Bridge Elements' Weight for Suspension Bridge.

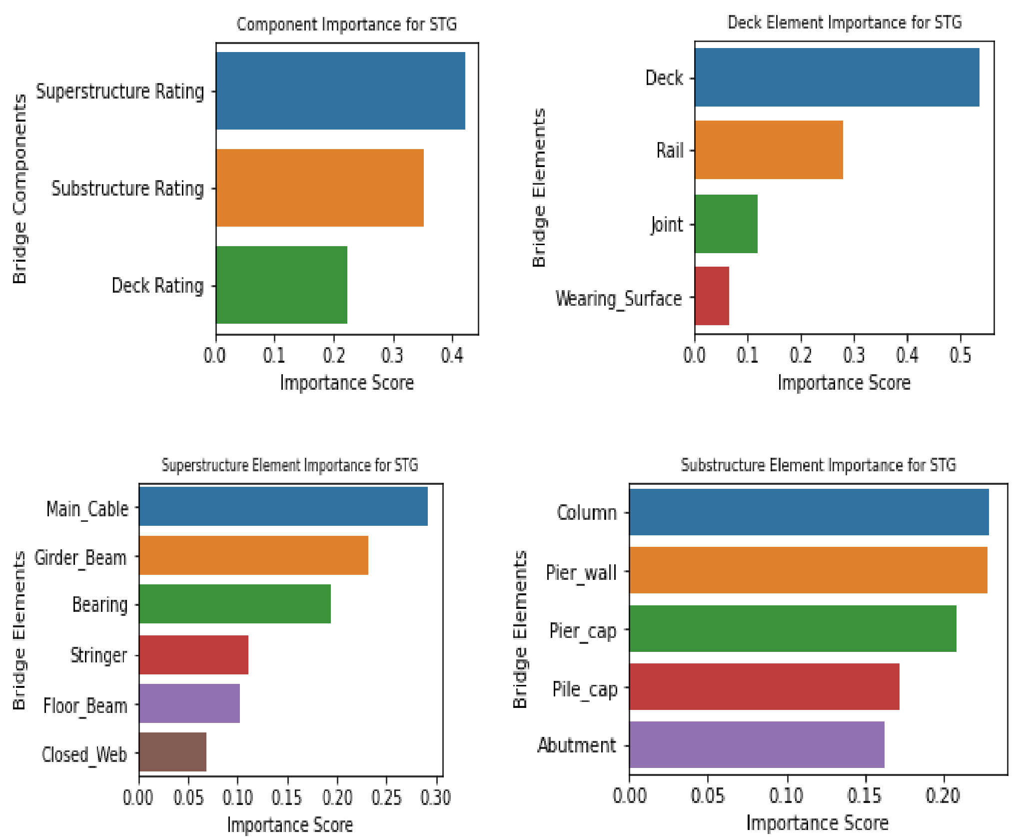


Figure 4-65: Bridge Components' and Elements' Importance for Stayed Girder Bridge.

Deck = 0.223253 * 0.536206 = 0.11971	Floor_Beam = 0.423007 * 0.102452 = 0.04334
Rail = 0.223253 * 0.279057 = 0.06230	Closed_Web = 0.423007 * 0.069225 = 0.02928
Joint = 0.223253 * 0.119634 = 0.02671	Column = 0.353740 * 0.229135 = 0.08105
Wearing Surface = 0.223253 * 0.065102 = 0.01453	Pier_wall = 0.353740 * 0.227653 = 0.08053
Main_Cable = 0.423007 * 0.291471 = 0.12329	Pier_Cap = 0.353740 * 0.208630 = 0.0738
Girder_Beam = 0.423007 * 0.232257 = 0.09825	Pile_cap = 0.353740 * 0.171834 = 0.06078
Bearing = 0.423007 * 0.193819 = 0.08199	Abutment = 0.353740 * 0.162748 = 0.05757
Stringer = 0.423007 * 0.110777 = 0.04686	

Figure 4-66: Bridge Elements' Weight for Stayed Girder Bridge.

4.2.1 Deck Elements' Weight

The deck element is more important to the overall condition rating of the Deck in all the main span design types except for the Truss-Deck (TRD), Movable-Bascule (MVB), Segmental Box Girder (SBG), and Suspension (SPS) design types. The bridge joint element has the highest importance for the Movable-Bascule (MVB)and Segmental Box Girder (SBG), the wearing surface has the highest importance for Suspension (SPS) bridges, and the rail element is the most important in the Deck for Truss-Deck (TRD) bridges. This is a rather unexpected result for these bridge types and can be deduced that the rail and joint elements in these bridge types are in a far worse condition than the deck elements which makes them influence the Deck condition rating more. The same analogy can be drawn for the Truss-Deck (TRD) which has the rail element as the most important element. This also goes to show that the bridge inspectors drop or increase the condition ratings of the whole Deck more due to the observed condition of these elements which further emphasizes their condition. It is also worth noting that in all the main span design types where the deck element is the most important bridge element in the Deck, the bridge rail happens to be the second most important element in such a structure which also emphasizes the importance of the rail element to the overall deck condition rating. The high importance of the deck element to the Deck condition rating in most bridge types can be attributed to the fact that it receives direct traffic load from the vehicles and has a large surface area that is exposed to extreme weather conditions. The bridge rail is subjected to collision effects from vehicles, and this explains why it has significant importance to the Deck condition rating. Furthermore, this result indicates a clear distinction between the deck element and the wearing surface that serves as its protective coating. The random forest algorithm can distinguish between these two elements in the weight calculation to help justify and reaffirm the bridge inspection procedure.

Also, according to Table 3-5, the number of bridge samples for the Movable-Bascule (MVB), Segmental Box Girder (SBG), and Suspension (SPS) design types are 83, 37, and 59 respectively which is a relatively lower number of observations compared to the other design types. Overall, the model will benefit from an increase in the amount of input data and as more inspection data is collected the model can be re-run to observe the changes in the level of importance of the bridge elements to the components' condition rating and help guide the DOTs on their bridge inspection policies and the maintenance, repair, and replacement (MRR) schedules for the different bridge design types. This will also serve as a note for bridge inspectors to see how the health index of the elements they inspect influences the general condition rating of the Deck and guide them for future inspection activity.

4.2.2 Superstructure Elements' Weight

For the superstructure, the most important element varies largely based on the structural configuration of the bridge. The Girder/Beam element is by far the most important in a Stringer/Multi-beam or Girder (SMG), Tee beam (TBM), and channel beam (CBM) bridge design type while the Closed web is by far the most important element in a Box beam or Girder (Single) (BBGS) and Box beam or Girder (Multiple) (BBGM). This outcome makes sense because the SMG, TBM, and CBM design types have the Girder/Beam element as their base design element, and, logically, they influence the superstructure condition rating more. The results for BBGS and BBGM also make sense because the closed web which is also known as the 'box girder' forms the base design of the superstructure and it is expected to have the highest importance to the superstructure rating. This confirms the synchronization between the elements' health index and the component's condition rating and showcases the efficiency of the random forest algorithm in distinguishing between the bridge elements based on their structural configuration.

In the Girder and Floor Beam System (GFS) design type, the Girder/beam, floor beam, bearing, and stringer elements are the most significant elements when compared to the rest of the elements in the superstructure. This is also a logical result because it is expected that the Girder/beam, floor beam, and stringer will have more importance to the superstructure condition rating of a signature bridge like the GFS and improves the confidence in the approach showcased in this paper for synthesizing the component and element-level data. This further reinforces the need to consider bridge elements based on their importance to the structural performance of the bridge component and not based on the perceived cost of replacement. In the Frame (FRM) design type, the girder/beam, arch, bearing, and floor beam elements are the most significant elements to the condition rating of the superstructure. This result is also reasonable for the FRM design type.

For the Truss-Thru (TRT) design type, the truss, floor beam, and stringer are the most important elements in the superstructure. This is yet another logical result that further confirms the synchronization between the bridge superstructure rating and the condition of its associated elements. The arch is the most important element in the arch-deck (ARD) design type while the truss is the most important element in the truss-deck (TRD) design type. The stringer, girder/beam, and floor beam are the most important superstructure elements in the movable-bascule (MVB) design type. The floor beam, girder/beam, arch, and stringer are the most significant elements in the arch-thru (ART) type while the closed web and girder/beam are the only significant elements in the segmental box girder (SBG) design type. So far, the bridge elements' importance keenly agrees with the expected output based on the bridge design type.

In the suspension (SPS) design type, the stringer, truss, girder/beam, floor beam, pin/hanger, secondary cable, main cable, and gusset plate have a significant level of importance to the condition of the superstructure. For the stayed girder (STG) design type, the main cable, girder/beam, and floor beam are the most important superstructure elements. This set also shows a reasonable result that can be further improved when additional inspection data are available.

Overall, the elements' weight of the superstructure elements for the different bridge design types considered in the analysis agrees mostly with the expected result and this confirms the reliability of the random forest approach for synthesizing the component and element-level data. The bridge inspection procedure/guideline is also validated with these results as it shows that the bridge inspectors are considering the structural configuration of the bridges in assigning importance to the bridge elements while assigning condition rating values to the superstructure.

4.2.3 Substructure Elements' Weight

In the substructure, most of the span design types have the abutment element as the most significant to the condition of the substructure except for movable bascule (MVB) and stay-girder (STG) where the pier cap and column have a higher importance respectively. The reason for this slight change in result might be that the pier cap and column are in a far worse condition than the abutment elements which makes them influence the Substructure condition rating more. The number of bridge observations used in creating the models in the MVB and STG design types are 83 and 16 respectively and the reliability of the results for these three design types will benefit from the availability of more inspection data.

This result is logical because the abutment is subjected to earth and hydrostatic pressure from the embarkment as it connects the bridge span to the land, and this explains why the condition rating of the substructure is greatly influenced by the abutment element. The abutment element provides an important support for the superstructure which also makes it subjected to the traffic load. All these affirm the importance of the abutment element which is showcased by the random forest algorithm. The importance of all other elements i.e., pier wall, pier cap, column, pile, pile cap, and trestle to the substructure rating varies significantly with the different main span design types. Thus, different bridge design types have different elements that dictate their structural performance as evidenced by the analysis conducted. This calls for a review cost-based approach that focuses more on the economic loss and not the structural impact of the elements' deterioration on the condition of the bridge component.

4.2.4 Overall Components' Weight

A bridge's Overall condition rating is taken as the lowest of the deck, superstructure, and substructure condition ratings as posted on the National Bridge Inventory (NBI) database. The analysis conducted with the random forest algorithm shows that the overall condition rating of the bridge is influenced more by the condition rating of the Superstructure and Substructure. This result shows that the Superstructure and Substructure are in a poorer condition compared to the Deck. The Deck has the least importance to the overall condition of the bridge in all the bridge design types except for the BBGS design type which has the Deck as the second most important component. This signifies that the Deck component is in a much better condition than all the other components and can be attributed to the more intensive maintenance action carried out on the Deck due to the direct discomfort caused to the road users when it is not in a good condition.

This analysis is a step toward achieving a new method for determining the overall Bridge Health Index (BHI) of different bridge design types as against the cost-based method that uses a constant weight in computing the Bridge Health Index BHI of a bridge. As a result of the approach showcased here, it is evident that each of the different bridge design types will have a distinct Bridge Health Index (BHI) equation that better represents the relevance of each of the elements to the structural performance of the bridge.

4.2.5 Resultant Bridge Elements' Weight

The bridge elements' weight for the 15 bridge design types as illustrated in Figure 4-37 to Figure 4-66 show that the abutment is the most important bridge element in all the bridge design types based on the data-driven analysis except for the MVB, SBG, and STG design types that have the Stringer, Closed web, and Main cable respectively as the most important bridge element. This result emphasizes the importance of the abutment to the overall performance of the bridge which can be related to a similar work done by Abiona, Head [27] stating the abutment is the most important element in the bridge. The ranking of the resultant weight of all the other bridge elements varies significantly with the different bridge design types. The bridge elements' weight observed for the different bridge design types helps to guide the DOTs on which bridge elements need maintenance action the most to boost the overall condition rating of their bridge inventory.

When maintenance action is conducted and the bridge components and elements are re-rated, the model can be re-run with the inclusion of the newly collected data to observe the changes in the level of importance of the bridge elements to the overall bridge condition and help guide the DOTs on their bridge inspection policies and the maintenance, repair, and replacement (MRR) schedules for the different bridge design types. This will also serve as a note for bridge inspectors to see how the health index of the elements they inspect influences the general condition rating of the components and guide them for future inspection activity.

Finally, the analysis conducted provides a promising approach that can be useful in converting the general condition rating values of the major components into the associated element-level health indexes for the different bridge design types. When the importance (or weight) of the elements is known, the condition rating of the components can be used to evaluate the possible condition of the associated elements. This will help in saving the cost of data collection as there will not be any need to collect large amounts of bridge data when the condition of the bridge elements can be analytically calculated.

Chapter 5

CONCLUSION AND FUTURE WORK

5.1 Summary

In the deterioration prediction of the bridge components using Artificial Neural Networks (ANN), principal component analysis (PCA) which is a statistical technique for capturing the important information in a dataset was successfully applied to create condition rating prediction models for the bridge Deck, Superstructure, and Substructure. In each of the bridge components, Principal Component (PC) models were created until the scree-plot showing the amount of cumulative variation explained flattened out, signifying no significant improvement in the information gained. The performance of the PC models in each of the components was compared with that of the base model that uses all the selected bridge features. In the Deck, Superstructure, and Substructure, the number of relevant features influencing their condition identified from literature are 14, 15, and 14 respectively, and were used directly in the base models. It was observed that as the number of principal components used in a model increases, the performance of the model also increases until the variance explained by subsequent principal components becomes insignificant (flattening of the scree-plot). Increasing the complexity of the model in terms of the number of hidden layers and neurons does not translate to better performance over adding more principal components. Adam optimizer was used to regulate the learning rate in all the models and an initial value of 0.0001 was sufficient to ensure quick convergence. A combination of dropout and L2 regularizer was successfully deployed in all the models to minimize the effects of overfitting. 'Relu' activation was used in all the hidden layers and 'Softmax' activation was used in the output layers of all the models since it is a classification problem. The models were created by training them on 68% of the dataset, cross-validating on 17% of the dataset to check for overfitting and testing the model performance on 15% of the dataset.

The Deck, using 9 principal components (9-PC model) which corresponds to 91.72% of the variation in the dataset to make a deterioration prediction model for the bridge deck has a better prediction accuracy and F1-score of 76% when compared to the base model, 75% which uses all the 14 bridge features. Apart from the improved model performance, the dimensionality of the data is also reduced from 14 features to 9 features (or PCs). The model with 9 principal components also has fewer neurons in some of the hidden layers when compared to the base model, which translates to lower computational costs. In the Superstructure, comparing the performance of the PC models with the base model that used all the 15 bridge features shows that 8 principal components corresponding to 86.69% of the variation in the data have the same accuracy and F1-score (i.e., 74%) as the base model. Going a step further to compare the performance of the model with 9 principal components (corresponding to 91.3% of the variation) with the base model, it shows that the model 9-PC has an accuracy and F1-score of 76%, surpassing that of the base model. In the Substructure, model 9-PC has the same accuracy and F1-score (72%) as the base model that used all the 14 bridge features. Model 10-PC has an accuracy and F1-score of 73% which surpasses that of the base model. These results emphasize the efficiency of PCA in maximizing the explained variance in a dataset to help improve the model performance while reducing the dimensionality of the data. To improve the accuracy of subsequent models, it is imperative to consider the following:

i. Minimize the subjectivity in assigning condition ratings to bridge decks as it can be confirmed by Moomen, Qiao [7] that ratings assigned by different inspectors are randomly distributed around the real condition of the bridge component.

ii. Low variability in the assigned condition rating by different inspectors on the same bridge deck will generally lead to more prediction accuracy in the models created.

iii. Other standardized means of inspection like computer-aided vision can be considered to improve the quality of bridge data collected.

The random forest algorithm which consists of multiple decision trees was used to evaluate the importance of bridge elements and components to the overall condition of the bridge. The analysis was conducted on 15 different bridge design types for the inventory of bridges in Delaware, Maryland, Pennsylvania, Virginia, and West Virginia. The analysis result shows that the deck and rail elements are the most important bridge elements in the Deck in most of the bridge design types and this is attributed to the fact that the deck elements receive the direct traffic load while the rail element is exposed to frequent collisions with vehicles. The most important element in the Superstructure varies depending on the structural configuration of the bridge with the Closed Web (or Box Girder) being the most important element in Box Beam bridges, the Truss being the most important element in Truss-Thru bridges, the Arch being the most important element in Arch-Deck bridges, etc. The most important substructure element in all the bridge design types except for the movable bascule (MVB) and stay-girder (STG) bridge is the abutment element. In contrast, the ranking of all other substructure elements varies depending on the design type. The results from the bridge component-element analysis are logical from a structural point of view and affirm the random forest algorithm's efficiency in determining the bridge elements' importance. The superstructure and substructure condition rating has the most influence on the overall condition rating in all the bridge design types, and this signifies that they are in a poorer condition compared to the Deck. The combination of the bridge components and elements' importance shows that the abutment element has the highest resultant weight in all the bridge design types except for the MVB, SBG, and STG design types that have the Stringer, Closed web, and Main cable respectively as the most important bridge element. The ranking of the resultant weights of all the other bridge elements varies largely depending on the bridge design type. This result shows the need to evaluate the importance of bridge elements based on their structural relevance to the performance of the bridge and not a constant value all through. The weight of the respective elements in the different bridge design types can be used to construct the overall Bridge Health Index (BHI) equation and provide a guide for the Departments of Transportation (DOTs) on which set of elements to prioritize in their maintenance actions to improve the overall condition of their bridge inventory. Finally, the results obtained from this approach show promise towards helping the departments of transportation (DOTs) to ascertain if the elements they give the highest priority in the maintenance, repair, and replacement (MRR) schedule and budget allocation are also the same set of elements the bridge inspector accord this level of importance during inspections. The models can be re-run as more inspection data becomes available to observe the changes in the level of importance of the bridge elements to the components' condition rating and help guide the DOTs on their bridge inspection policies.

5.2 Recommendations and Future Work

The result of this research shows that the quality and reliability of the bridge components' condition prediction model can be improved by investing more in computerized inspection to ensure consistency in the data collection process. Also, the bridge owners can adopt the resultant bridge elements' weight computed in this research to formulate the bridge health index (BHI) equation for the different bridge types in their jurisdiction. Furthermore, the cost-based elements' weight approach currently in use can be transformed by combining it with the results showcased in this study to develop a more robust bridge management system that optimizes the repair budget and the overall bridge condition.

In the deterioration prediction of the bridge components, future studies could include careful consideration of the evolution of bridge performance degradation or lifespan problems if the data becomes available. Other data improvement techniques will also be considered for analysis to observe any improvements in the performance of the bridge components' condition prediction models. The bridge weight analysis provides a novel approach for converting the general condition rating of the components to the element-level health index by using the derived weights of the elements based on the design type to redistribute the condition rating among the elements that make up the bridge component. The future work involved showcasing how this weight redistribution can be conducted to improve the data collection process of the DOTs.

REFERENCES

1. Bektaş, B., *Bridge Element Data Collection and Use*. 2022.
2. Fiorillo, G. and H. Nassif, *Application of machine learning techniques for the analysis of national bridge inventory and bridge element data.* Transportation Research Record, 2019. **2673**(7): p. 99-110.
3. ASCE, *Report Card for America's Infrastructure*. 2021.
4. Chase, S.B., et al., *Synthesis of national and international methodologies used for bridge health indices.* 2016.
5. FHWA, *Specification for the national bridge inventory bridge elements.* 2014.
6. Kong, X., et al., *Bridge Deck Deterioration: Reasons and Patterns.* Transportation Research Record, 2022: p. 03611981221080140.
7. Moomen, M., et al., *Bridge deterioration models to support Indiana's bridge management system.* 2016.
8. Yehia, S.A. and C.Y. Tuan, *Bridge deck deicing.* 1998.
9. Kreislova, K. and H. Geiplova, *Evaluation of corrosion protection of steel bridges.* Procedia Engineering, 2012. **40**: p. 229-234.
10. Stewart, M.G. and D.V. Rosowsky, *Structural safety and serviceability of concrete bridges subject to corrosion.* Journal of Infrastructure systems, 1998. **4**(4): p. 146-155.
11. Kim, K.H., et al., *Prediction of remaining life for bridge decks considering deterioration factors and propose of prioritization process for bridge deck maintenance.* Sustainability, 2020. **12**(24): p. 10625.
12. AASHTO, *Manual for Bridge Evaluation*. 2018, AASHTO Washington, DC.
13. Nguyen, T.T. and K. Dinh, *Prediction of bridge deck condition rating based on artificial neural networks.* Journal of Science and Technology in Civil Engineering (STCE)-HUCE, 2019. **13**(3): p. 15-25.
14. Assaad, R. and I.H. El-adaway, *Bridge infrastructure asset management system: Comparative computational machine learning approach for evaluating and predicting deck deterioration conditions.* Journal of Infrastructure Systems, 2020. **26**(3): p. 04020032.
15. Huang, Y.-H., *Artificial neural network model of bridge deterioration.* Journal of Performance of Constructed Facilities, 2010. **24**(6): p. 597-602.
16. Winn, E.K. and R. Burgueño, *Development and validation of deterioration models for concrete bridge decks-phase 1: artificial intelligence models and bridge management system*. 2013, Michigan. Dept. of Transportation.
17. Zhu, J. and Y. Wang, *Feature Selection and Deep Learning for Deterioration Prediction of the Bridges.* Journal of Performance of Constructed Facilities, 2021. **35**(6): p. 04021078.

18. Hasan, S. and E. Elwakil, *National bridge inventory data-based stochastic modeling for deck condition rating of prestressed concrete bridges.* Practice Periodical on Structural Design and Construction, 2020. **25**(3): p. 04020022.
19. Fiorillo, G. and H. Nassif, *Improving the conversion accuracy between bridge element conditions and NBI ratings using deep convolutional neural networks.* Structure and Infrastructure Engineering, 2020. **16**(12): p. 1669-1682.
20. Lin, Z., et al., *Improved element-level bridge inspection criteria for better bridge management and preservation.* Fargo, ND: Mountain-Plains Consortium, 2019.
21. Jiang, X. and K.L. Rens, *Bridge health index for the city and county of Denver, Colorado. II: Denver bridge health index.* Journal of performance of constructed facilities, 2010. **24**(6): p. 588-596.
22. Jiang, X. and K.L. Rens, *Bridge health index for the city and county of Denver, Colorado. I: Current methodology.* Journal of performance of constructed facilities, 2010. **24**(6): p. 580-587.
23. Inkoom, S., et al., *Bridge health index: Study of element condition states and importance weights.* Transportation Research Record, 2017. **2612**(1): p. 67-75.
24. FHWA, *Specifications for the national bridge inventory*. 2022, Draft, US Department of Transportation Washington, DC.
25. Gattulli, V. and L. Chiaramonte, *Condition assessment by visual inspection for a bridge management system.* Computer-Aided Civil and Infrastructure Engineering, 2005. **20**(2): p. 95-107.
26. Inkoom, S. and J. Sobanjo, *Availability function as bridge element's importance weight in computing overall bridge health index.* Structure and Infrastructure Engineering, 2018. **14**(12): p. 1598-1610.
27. Abiona, Q.O., M.H. Head, and Y. Yoon. *Bridge Element Weights Based on Data-Driven Model with Artificial Neural Networks*. in *Structures Congress 2023*. 2023.
28. Ryan, T.W., et al., *Bridge inspector's reference manual (BIRM)*. 2022.
29. Srikanth, I. and M. Arockiasamy, *Deterioration models for prediction of remaining useful life of timber and concrete bridges: A review.* Journal of traffic and transportation engineering (English edition), 2020. **7**(2): p. 152-173.
30. Ranjith, S., et al., *Deterioration prediction of timber bridge elements using the Markov chain.* Journal of Performance of Constructed Facilities, 2013. **27**(3): p. 319-325.
31. Madanat, S., R. Mishalani, and W.H.W. Ibrahim, *Estimation of infrastructure transition probabilities from condition rating data.* Journal of infrastructure systems, 1995. **1**(2): p. 120-125.
32. Trans, G., *An in-depth analysis of the national bridge inventory database utilizing data mining, GIS and advanced statistical methods.* 1999.
33. Agrawal, A.K., A. Kawaguchi, and Z. Chen, *Bridge element deterioration rates*. 2008, New York (State). Dept. of Transportation.

34. Nickless, K., *Investigation of mechanistic deterioration modeling for bridge design and management*. 2017: Colorado State University.
35. Lu, C., R. Liu, and W. Jin, *A model for predicting time to corrosion-induced cover cracking in reinforced concrete structures.* J. of Fracture Mechanics of Concrete, 2010. **53**(4): p. 967-976.
36. Goodfellow, I., Y. Bengio, and A. Courville, *Deep learning*. 2016: MIT press.
37. Althaqafi, E., *Developing Bridge Deterioration Model Using Artificial Neural Network and Markov Chain*. 2021: The University of Toledo.
38. Miao, P. and H. Yokota, *Comparison of Markov chain and recurrent neural network in predicting bridge deterioration considering various factors.* Structure and Infrastructure Engineering, 2022: p. 1-13.
39. Bektas, B.A., A. Carriquiry, and O. Smadi, *Using classification trees for predicting national bridge inventory condition ratings.* Journal of infrastructure systems, 2013. **19**(4): p. 425-433.
40. Wang, C., et al., *A machine learning framework for predicting bridge defect detection cost.* Infrastructures, 2021. **6**(11): p. 152.
41. Miao, P., *Prediction-based maintenance of existing bridges using neural network and sensitivity analysis.* Advances in Civil Engineering, 2021. **2021**: p. 1-17.
42. Ali, G., et al., *Artificial neural network model for bridge deterioration and assessment.* 2019.
43. Althaqafi, E. and E. Chou, *Developing Bridge Deterioration Models Using an Artificial Neural Network.* Infrastructures, 2022. **7**(8): p. 101.
44. Krogh, A., *What are artificial neural networks?* Nature biotechnology, 2008. **26**(2): p. 195-197.
45. Abraham, A., *Artificial neural networks.* Handbook of measuring system design, 2005.
46. Pang, B., E. Nijkamp, and Y.N. Wu, *Deep learning with tensorflow: A review.* Journal of Educational and Behavioral Statistics, 2020. **45**(2): p. 227-248.
47. Sharma, S., S. Sharma, and A. Athaiya, *Activation functions in neural networks.* Towards Data Sci, 2017. **6**(12): p. 310-316.
48. Nwankpa, C., et al., *Activation functions: Comparison of trends in practice and research for deep learning.* arXiv preprint arXiv:1811.03378, 2018.
49. Szandała, T., *Review and comparison of commonly used activation functions for deep neural networks.* Bio-inspired neurocomputing, 2021: p. 203-224.
50. Ding, B., H. Qian, and J. Zhou. *Activation functions and their characteristics in deep neural networks*. in *2018 Chinese control and decision conference (CCDC)*. 2018. IEEE.
51. Goldsborough, P., *A tour of tensorflow.* arXiv preprint arXiv:1610.01178, 2016.
52. Choi, D., et al., *On empirical comparisons of optimizers for deep learning.* arXiv preprint arXiv:1910.05446, 2019.

53. Schmidt, R.M., F. Schneider, and P. Hennig. *Descending through a crowded valley-benchmarking deep learning optimizers*. in *International Conference on Machine Learning*. 2021. PMLR.
54. Kukačka, J., V. Golkov, and D. Cremers, *Regularization for deep learning: A taxonomy.* arXiv preprint arXiv:1710.10686, 2017.
55. Goodfellow, I., Y. Bengio, and A. Courville, *Regularization for deep learning.* Deep learning, 2016: p. 216-261.
56. Liu, H., J. Nehme, and P. Lu, *An application of convolutional neural network for deterioration modeling of highway bridge components in the United States.* Structure and Infrastructure Engineering, 2021: p. 1-14.
57. Liu, H. and Y. Zhang, *Bridge condition rating data modeling using deep learning algorithm.* Structure and Infrastructure Engineering, 2020. **16**(10): p. 1447-1460.
58. Chang, M., M. Maguire, and Y. Sun, *Framework for mitigating human bias in selection of explanatory variables for bridge deterioration modeling.* Journal of Infrastructure Systems, 2017. **23**(3): p. 04017002.
59. Moomen, M. and C. Siddiqui, *Probabilistic deterioration modeling of bridge component condition with random effects.* Journal of Structural Integrity and Maintenance, 2022. **7**(3): p. 151-160.
60. Abdi, H. and L.J. Williams, *Principal component analysis.* Wiley interdisciplinary reviews: computational statistics, 2010. **2**(4): p. 433-459.
61. Wetherell, J., et al., *Insights into one-body density matrices using deep learning.* Faraday Discussions, 2020. **224**: p. 265-291.
62. Song, F., Z. Guo, and D. Mei. *Feature selection using principal component analysis*. in *2010 international conference on system science, engineering design and manufacturing informatization*. 2010. IEEE.
63. Lasisi, A. and N. Attoh-Okine, *Principal components analysis and track quality index: A machine learning approach.* Transportation Research Part C: Emerging Technologies, 2018. **91**: p. 230-248.
64. Bro, R. and A.K. Smilde, *Principal component analysis.* Analytical methods, 2014. **6**(9): p. 2812-2831.
65. Shlens, J., *A tutorial on principal component analysis.* arXiv preprint arXiv:1404.1100, 2014.
66. Omuya, E.O., G.O. Okeyo, and M.W. Kimwele, *Feature selection for classification using principal component analysis and information gain.* Expert Systems with Applications, 2021. **174**: p. 114765.
67. Zhang, J., *Machine learning with feature selection using principal component analysis for malware detection: a case study.* arXiv preprint arXiv:1902.03639, 2019.
68. Yuce, B., et al., *Neural network design and feature selection using principal component analysis and Taguchi method for identifying wood veneer defects.* Production & Manufacturing Research, 2014. **2**(1): p. 291-308.

69. Jaiswal, J.K. and R. Samikannu. *Application of random forest algorithm on feature subset selection and classification and regression*. in *2017 world congress on computing and communication technologies (WCCCT)*. 2017. IEEE.

Appendix A

PERMISSION AND OTHER MATERIALS

McAbee, Wendy (FHWA) <Wendy.McAbee@dot.gov> Thu, Dec 21, 2023, 10:50 AM
to me, Dawn, Semme, Wendy

Qozeem:

The National Bridge Inventory Bridge Elements manual is a document from the FHWA Office of Bridges and Structures. The Turner-Fairbank Highway Research Center forwarded your request to us and we do not have any issues with you reprinting/using the image in Appendix B as long as you cite the appropriate source for the image.

Wendy L. McAbee, P.E.

Senior Bridge and Tunnel Inventory Engineer
FHWA Office of Bridges and Structures
1200 New Jersey Ave SE, E75-122
Washington, DC 20590
202-366-5658
Wendy.mcabee@dot.gov

Figure A1: Permission to reprint figure 1-1.

```
import pandas as pd
import numpy as np
from sklearn import model_selection
from sklearn import metrics
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
import tensorflow as tf
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.losses import SparseCategoricalCrossentropy
from tensorflow.keras.optimizers import Adam
from sklearn.metrics import ConfusionMatrixDisplay
from sklearn.metrics import confusion_matrix
from tensorflow.keras.layers import Input
from tensorflow.keras.regularizers import L2
from tensorflow.keras.layers import Dropout
from tensorflow.keras.callbacks import EarlyStopping
md = pd.read_csv('deck_pca.csv') ##Import condition rating data for maryland
va = pd.read_csv('va.csv') ##Virginia
## Supplemenent condition rating data with data from another state
df1 = md
df_list_1 = [v4, v8] # df1 has been intialized

for i in range(len(df_list_1)):
    df1 = df1.append(df_list_1[i], ignore_index = True)
```

```
##draw out 53000 bridge samples
samples = []
for group in df1.deckCR.unique():
    if group == 4:
        s = df1.loc[df1.deckCR== group].sample(n=10000, random_state = 2).reset_index(drop=True)
        samples.append(s)
    elif group == 5:
        s = df1.loc[df1.deckCR== group].sample(n=11000, random_state = 2).reset_index(drop=True)
        samples.append(s)
    elif group == 6:
        s = df1.loc[df1.deckCR== group].sample(n=11000, random_state = 2).reset_index(drop=True)
        samples.append(s)
    elif group == 7:
        s = df1.loc[df1.deckCR== group].sample(n=11000, random_state = 2).reset_index(drop=True)
        samples.append(s)
    elif group == 8:
        s = df1.loc[df1.deckCR== group].sample(n=10000, random_state = 2).reset_index(drop=True)
        samples.append(s)

sample = pd.concat(samples, axis=0)
sample.reset_index(drop = True, inplace = True)
## Reformat data and apply PCA
sample['deckCR'] = sample['deckCR'].replace([4,5,6,7,8],[0,1,2,3,4])
X = sample.iloc[:, :-1]
Y = sample.iloc[:, 14]
scaler = StandardScaler()
X_norm = scaler.fit_transform(X)
pca_2 = PCA(n_components=2)
Xr = pca_2.fit_transform(X_norm)
y= np.array(Y)
Yr = y.reshape(53000,1)
#Split Data
X_train, X_test, y_train, y_test = model_selection.train_test_split(Xr, Yr, test_size= 0.15, random_state = 1)

print('X_train dimension= ', X_train.shape)
print('X_test dimension= ', X_test.shape)
print('y_train dimension= ', y_train.shape)
```

```
##ANN
early_stopping = EarlyStopping(monitor='val_accuracy',patience=50,min_delta=0.001,mode='max')
model = Sequential([Input(2),
                    Dense(50, 'relu',kernel_regularizer= L2(0.00015) ),
                    Dropout(0.01),
                    Dense(150, 'relu',kernel_regularizer= L2(0.00015) ),
                    Dropout(0.01),
                    Dense(200, 'relu',kernel_regularizer= L2(0.00015) ),
                    Dropout(0.01),
                    Dense(50, 'relu',kernel_regularizer= L2(0.00015) ),
                    Dropout(0.01),
                    Dense(5, 'softmax' )])
model.compile(loss = SparseCategoricalCrossentropy(), optimizer = Adam(0.0001), metrics = ['accuracy'])
history = model.fit(X_train,y_train, validation_split=0.2, epochs=800,callbacks=[early_stopping])
#Get learning curve
fig, ax1 = plt.subplots(figsize = (4,4))
ax1.set_xlabel('Epoch')
ax1.set_ylabel('Loss')
ax1.plot(history.history['loss'],'--')
ax1.plot(history.history['val_loss'],'--')
ax1.legend(['train_loss', 'CV_loss'], loc= 10,prop= {'size':6})
ax2 = ax1.twinx()
ax2.set_ylabel('Accuracy')
ax2.plot(history.history['accuracy'])
ax2.plot(history.history['val_accuracy'])
ax2.legend(['train_acc', 'CV_acc'], loc= 7, prop= {'size':6})
plt.title('2 PC Learning Curve')
fig.tight_layout()
#plt.savefig('2-PC-curve.png')
plt.show()
#Get classification report
pred = np.array(np.argmax(f_x, axis =1))
pred.reshape(7950,1)
print(metrics.classification_report(y_test, pred))
#Get confusion matrix
labels = ["4", "5", "6",'7','8']
cm = confusion_matrix(y_test, pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=labels)
disp.plot(cmap=plt.cm.Blues)
plt.title('2 PC, XX accuracy')
disp.figure_.savefig('2-PC-CM.png')
```

Figure A2: Code for condition prediction models.

```python
import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestClassifier
from sklearn.feature_selection import SelectFromModel
import matplotlib.pyplot as plt
%matplotlib inline
MD_BI = pd.read_csv('MD_NBI.csv') ## Import the NBI data
NBI = MD_BI[MD_BI['Year'] >= 2015] ## Take out NBI data from 2015 onwards
NBI.reset_index(drop = True, inplace = True) ## Reset the numbering
##The next 5 lines checks the structure number for extra spaces leading to error
space = []
for i in range(len(NBI)):
    ot = NBI.iloc[i,2].strip()
    space.append(ot)
NBI['8 - Structure Number'] = space ## Check ends here
Miss_deck = NBI[NBI['58 - Deck Condition Rating'] == 'N'] ## Isolate bridges with missing condition rating
NBI = NBI[~NBI['8 - Structure Number'].isin(Miss_deck['8 - Structure Number'].unique())] ## Remove the bridges
NBI.reset_index(drop = True, inplace = True)
##Check for bridges with invalid condition ratings
boc = set()
for i in range(len(NBI)):
    if NBI.loc[i,'60 - Substructure Condition Rating'] not in str([1,2,3,4,5,6,7,8,9]):
        boc.add(NBI.loc[i,'8 - Structure Number']) ## Check ends here.
#Ensure ratings are integers
NBI['58 - Deck Condition Rating'] = NBI['58 - Deck Condition Rating'].astype(str).astype(int)
NBI['59 - Superstructure Condition Rating'] = NBI['59 - Superstructure Condition Rating'].astype(str).astype(int)
NBI['60 - Substructure Condition Rating'] = NBI['60 - Substructure Condition Rating'].astype(str).astype(int)
##The following function filters the element-level data to ensure same bridge repository
def filter_element(x):
    Del = set()
    for i in range(len(x)):
        if x.loc[i,'STRUCNUM'] not in NBI['8 - Structure Number'].tolist():
            Del.add(x.loc[i,'STRUCNUM'])
    x = x[~x['STRUCNUM'].isin(Del)]
    x.reset_index(drop = True, inplace = True)
    x['HI']= ((x['CS1']/x['TOTALQTY'])*100)+((x['CS2']/x['TOTALQTY'])*67)+((x['CS3']/x['TOTALQTY'])*33)+((x['CS4']/x['TOTALQTY'])*0)
    return x, Del
```

```python
## After getting the element level data for each state, merge using the following code
TBE = DEBE ## This is the first state, other states will be merged
Lset2 = [MDBE,PABE,VABE,WVBE]
for i in range(len(Lset2)):
    TBE = TBE.append(Lset2[i], ignore_index = True)
#Merge the condition rating data for each state and seperate into bridge types
TBI = DEBI
Lset = [MDBI,PABI,VABI,WVBI]
for i in range(len(Lset)):
    TBI = TBI.append(Lset[i], ignore_index = True)
SMG = TBI[TBI['43B - Main Span Design']== 'Stringer/Multi-beam or Girder'] ## Ex. taking out one bridge type
SMGBE = TBE[TBE['STRUCNUM'].isin(SMG['8 - Structure Number'])] ## synchronize the component and element-level data
SMGBE = SMGBE[~SMGBE['EN'].isin([241,244,515,521])] ## remove this elements from the analysis culvert and coatings
SMGBE.reset_index(drop = True, inplace = True)
SMG_Piv = pd.pivot_table(SMGBE, values = 'HI', index=['Year', 'STRUCNUM'], columns = 'EN').reset_index() #Pivot table
dcol = [12,13,15,16,28,29,30, 31,38,60,65]
SMG_Piv = SMG_Piv.assign(Deck = SMG_Piv[dcol].min(1)).drop(dcol, 1) #Tke the minimum in case many element type occur
##Repeat the above for all elements to form distint columns
## The following code merges the component and element data.
SMG['Year'] = SMG['Year'].astype(str)
SMG['8 - Structure Number'] = SMG['8 - Structure Number'].astype(str)
SMG_Piv['STRUCNUM'] = SMG_Piv['STRUCNUM'].astype(str)
SMG_Piv['Year'] = SMG_Piv['Year'].astype(str)
SMGBI = SMG[SMG.columns[[0,2,-4,-3,-2,-1]]]
SMGBI['merge']= SMGBI['8 - Structure Number'] + '-'+ SMGBI['Year']
SMGE = SMG_Piv
SMGE['merge']= SMGE['STRUCNUM'] + '-' + SMGE['Year']
SMGT = SMGE.merge(SMGBI, on='merge') ## Merged dataframe
rem = ['merge', 'Year_y', '8 - Structure Number']
SMGT.drop(rem, inplace = True, axis = 1)
SMGT = SMGT.rename(columns = {'Year_x':'Year'}) ## clean up
SMGT_deck = SMGT[SMGT.columns[[0,1,2,3,4,5,-4]]] ## Group deck elements
SMGT_sup = SMGT[SMGT.columns[[0,1,6,7,8,9,10,11,12,13,14,-3]]] ## Group superstructure elements
SMGT_sub = SMGT[SMGT.columns[[0,1,15,16,17,18,19,20,21,-2]]] ## Group substructure elements
SMG_bridge = SMGT[SMGT.columns[[0,1,-4,-3,-2,-1]]] ## Group components with overall condition rating
```

```
## Applying random forest, do the same for the 3 bridge components
SMG_bridge_X = SMG_bridge.iloc[:, 2:-1]
SMG_bridge_Y = SMG_bridge.iloc[:, -1]
SMG_bridge_feat = RandomForestClassifier(n_estimators=100, random_state=0)
SMG_bridge_feat.fit(SMG_bridge_X, SMG_bridge_Y)
SMG_bridge_feature_scores = pd.Series(SMG_bridge_feat.feature_importances_, index=SMG_bridge_X.columns).sort_values(ascending=False)
SMG_bridge_feature_scores # To show the weights
#Creat the plots
f, ax = plt.subplots(figsize=(4.5, 3))
ax = sns.barplot(SMG_bridge_feature_scores, SMG_bridge_feature_scores.index, dodge = False)
ax.set_title("Component Importance for SMG", size = 9)
ax.set_yticklabels(SMG_bridge_feature_scores.index)
#, data= SMGT_deck.iloc[:, 2:]
ax.set_xlabel("Importance Score")
ax.set_ylabel("Bridge Components")
f.tight_layout()
#plt.savefig('SMG_comp.png')
plt.show()
```

Figure A3: Code for bridge elements' weight.

Data Sources:

- https://infobridge.fhwa.dot.gov/
- https://www.fhwa.dot.gov/bridge/nbi/element.cfm