The Use of Artificial Intelligence in Pavement Engineering

FINAL REPORT
by

Natalie Mikels, Emad Kassem, Ahmed Muftah, and Abu Sufian
University of Idaho

Sponsorship
PacTrans

for

Pacific Northwest Transportation Consortium (PacTrans)
USDOT University Transportation Center for Federal Region 10
University of Washington
More Hall 112, Box 352700
Seattle, WA 98195-2700

In cooperation with U.S. Department of Transportation,
Research and Innovative Technology Administration (RITA)

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The Pacific Northwest Transportation Consortium, the U.S. Government and matching sponsor assume no liability for the contents or use thereof.

Technical Report Documentation Page		
1. Report No.	**2. Government Accession No.** 01784894	**3. Recipient's Catalog No.**
4. Title and Subtitle The Use of Artificial Intelligence in Pavement Engineering		**5. Report Date** November 2023 **6. Performing Organization Code**
7. Author(s) and Affiliations Natalie Mikels, Emad Kassem, 0000-0002-4331-6692; Ahmed Muftah, and Abu Sufian University of Idaho Moscow, ID 83844-1022		**8. Performing Organization Report No.** 2021-M-UAF-2
9. Performing Organization Name and Address PacTrans Pacific Northwest Transportation Consortium University Transportation Center for Federal Region 10 University of Washington More Hall 112 Seattle, WA 98195-2700		**10. Work Unit No. (TRAIS)** **11. Contract or Grant No.** 69A3551747110
12. Sponsoring Organization Name and Address United States Department of Transportation Research and Innovative Technology Administration 1200 New Jersey Avenue, SE Washington, DC 20590		**13. Type of Report and Period Covered** **14. Sponsoring Agency Code**
15. Supplementary Notes Report uploaded to: www.pactrans.org		

16. Abstract

The performance of asphalt pavements decreases with time because of traffic loading and environmental conditions. Performance decay models are needed in pavement management systems to program pavement preservation and rehabilitation treatments to extend the service life and improve the performance of flexible pavements. Many factors affect pavement performance, including the material properties and thickness of each layer, applied traffic, and environmental conditions. Performance models, including those for rutting, cracking, roughness, are often developed and used to forecast the future conditions of pavements. Meanwhile, to develop reliable performance models, numerous variables are needed in such models, and historical performance data are required. This study investigated and developed multiple types of artificial intelligence models to predict pavement performance.

The study results demonstrated that random forests regression was best suited for the data utilized in this study. Multiple random forests regression models were developed to predict various indicators of pavement performance, such as the International Roughness Index (IRI), rutting, and cracking. These models utilized a theoretical dataset generated with the Pavement ME software and field data collected from the Long-Term Pavement Performance (LTPP) Program. There were good correlations between all the theoretical and predicted performance indicators. In addition, the predicted performance decay curves were found to closely simulate the measured decay curves. In addition, the results for the models developed with the field dataset demonstrated good correlations between measured and predicted performance indicators for some of the investigated performance indicators.

17. Key Words		**18. Distribution Statement**	
artificial intelligence, pavement performance prediction			
19. Security Classification (of this report)	**20. Security Classification (of this page)**	**21. No. of Pages**	**22. Price**
Unclassified.	Unclassified.	99	N/A

Form DOT F 1700.7 (8-72) **Reproduction of completed page authorized.**

SI* (Modern Metric) Conversion Factors

Symbol	When You Know	Multiply By	To Find	Symbol
APPROXIMATE CONVERSIONS TO SI UNITS				
		LENGTH		
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	square millimeters	mm^2
ft^2	square feet	0.093	square meters	m^2
yd^2	square yard	0.836	square meters	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	square kilometers	km^2
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	cubic meters	m^3
yd^3	cubic yards	0.765	cubic meters	m^3
		NOTE: volumes greater than 1000 L shall be shown in m^3		
		MASS		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2000 lb)	0.907	megagrams (or "metric ton")	Mg (or "t")
		TEMPERATURE (exact degrees)		
°F	Fahrenheit	5 (F-32)/9 or (F-32)/1.8	Celsius	°C
		ILLUMINATION		
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	candela/m^2	cd/m^2
		FORCE and PRESSURE or STRESS		
lbf	poundforce	4.45	newtons	N
lbf/in^2	poundforce per square inch	6.89	kilopascals	kPa
APPROXIMATE CONVERSIONS FROM SI UNITS				
Symbol	**When You Know**	**Multiply By**	**To Find**	**Symbol**
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	square millimeters	0.0016	square inches	in^2
m^2	square meters	10.764	square feet	ft^2
m^2	square meters	1.195	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	square kilometers	0.386	square miles	mi^2
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	cubic meters	35.314	cubic feet	ft^3
m^3	cubic meters	1.307	cubic yards	yd^3
		MASS		
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or "t")	megagrams (or "metric ton")	1.103	short tons (2000 lb)	T
		TEMPERATURE (exact degrees)		
°C	Celsius	1.8C+32	Fahrenheit	°F
		ILLUMINATION		
lx	lux	0.0929	foot-candles	fc
cd/m^2	candela/m^2	0.2919	foot-Lamberts	fl
		FORCE and PRESSURE or STRESS		
N	newtons	0.225	poundforce	lbf
kPa	kilopascals	0.145	poundforce per square inch	lbf/in^2

*SI is the symbol for the International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380. (Revised March 2003)

Table of Contents

List of Figures

List of Tables

Abbreviations

AADT	Average annual daily traffic
AAP	
AASHTO	American Association of State Highway and Transportation Official
AAT	
AC	Asphalt concrete
AI	Artificial intelligence
ANFIS	Adaptive neuro fuzzy inference system
ANN	Artificial neural network
ARA	Automatic Road analyzer
BPNN	Back-propagation neural network
CI	Crack index
COM	Composite (pavement)
ESAL	Equivalent single axle load
FWD	Falling weight deflectometer
GA	Genetic algorithm
GBDT	Gradient boosted decision tree
GMDH	Group methods of data handling
GPI	Global Pavement Index
GPS	Global Positioning System
HMA	Hot-mix asphalt
LCMS	Laser crack measurement system
IRI	International Roughness Index
LSTM	Long-term/short-term memory
LTPP	Long-Term Pavement Performance
PCC	Portland cement concrete
PCI	Pavement condition index
PCR	
PMS	Pavement management system
PPPM	Pavement performance prediction models
PT	
RF	
SEM	Structural equation modeling
SHRP	Strategic Highway Research Program
SQL	Structured query language
ST	
SVM	Support vector machine
TSD	Traffic speed deflectometer

CHAPTER 1. Introduction

1.1. Research Goal

This study had the following objectives:

- Investigate different types of artificial intelligence tools that can be used to model the performance decay in asphalt pavements.
- Develop artificial intelligence models using structural, environmental, and traffic data to predict pavement performance indicators.

1.2. Research Background and Problem Statement

Pavement management systems often use performance decay models to predict the future performance of pavements so that agencies can plan appropriate maintenance and extend the life of given sections. The performance of asphalt pavements is affected by various distresses such as cracking, rutting, and moisture damage. Their performance can also be affected by the material properties and thickness of each pavement layer, the traffic on the pavement, and environmental factors such as temperature and moisture. The layers of asphalt pavements include the asphalt layer, the base layer, the subbase layer, and the subgrade layer. The most significant material property of these layers is the modulus of elasticity. The modulus represents the stiffness of a pavement and characterizes its ability to withstand permanent deformation (Hossain et al., 2017). Permanent deformation is one of the many factors that affect a pavement's condition.

Pavement performance or condition is the ability of that pavement to serve traffic. There are many ways to measure the performance or condition of a pavement. Two of the most popular methods are the International Roughness Index (IRI), which is an indicator of ride quality, and the pavement condition index (PCI), which scores pavements on a zero to 100 scale intended to capture overall pavement quality. Cracking is one of the major distresses that can be inspected visually. Visual inspection involves inspectors quantifying the number of cracks or the severity of cracks on a pavement segment and then translating that information to a rating. There are many types of cracking, such as fatigue cracking, longitudinal cracking, transverse cracking, and thermal cracking (Hossain et al., 2017). These distresses must be managed to ensure that the road remains serviceable.

Decisions about the management of different pavements within pavement management systems are reliant on the methods used to report and predict pavement conditions. Pavement management systems that can effectively make these decisions support better use of public funds, better distribution of resources, and well-maintained roadways (Salini et al., 2015).

Conducting maintenance activities on pavement sections is essential for their performance. There are optimal times to perform maintenance that will reduce costs over time and also increase the longevity of pavement sections. Costs can be reduced by performing minor treatments on a pavement before major, more costly treatments are needed to restore it. However, performing these treatments too early can lead to an increase in cost because then they are done more frequently than necessary. The optimal times are based on the deterioration in performance of a given section.

Deterioration of pavements can be difficult to predict because numerous factors affect pavement condition. So a better understanding of this deterioration can aid pavement engineers in making better decisions regarding maintenance activities. Therefore, there is a need to develop models that can predict pavement conditions over time.

1.3. Research Approach

The objectives of this project were achieved by conducting the research tasks discussed below.

1.3.1. Task 1. Literature Review

The objective of this task was to conduct a comprehensive literature review of different types of artificial intelligence and their applications in civil and pavement engineering. The main subjects of the literature review were as follows:

- Different measures of pavement deterioration and how they are related to pavement management
- Various artificial intelligence methods and how those methods operate
- Applications of different artificial intelligence models in pavement engineering
- Applications of different artificial intelligence models in civil engineering.

1.3.2. Task 2. Collect Data

Data from several theoretical and field pavement segments were collected to create multiple datasets that could be used to train and test different models. Theoretical datasets were generated with the Pavement ME software and the Modulus 7 software so that a large range of sections could be represented. Field datasets were collected from the Long-Term Pavement Performance (LTPP) Program, in addition to falling weight deflectometer (FWD) and traffic speed deflectometer (TSD) deflection data that were collected in Idaho to validate that the models could be applied in practice.

The collected datasets were edited and organized so that they could be used to train and test the models. Artificial intelligence models are sensitive to data format, so proper organization of the datasets was vital to their performance. Descriptive parameters in the datasets were converted to numerical values to simplify the code used in the models. Cumulative traffic data were also compiled into the datasets based on the yearly traffic data provided. The datasets were also organized so that each time measurement for a given section was treated as a separate data point instead of as a collective section. This was done to develop performance decay curves for pavement sections and to use the largest amount of data possible to train and test the models.

1.3.3. Task 3. Analysis of the Results

Three different types of artificial intelligence were evaluated to determine which approach would perform best for the remaining tasks. The models selected for this investigation were a neural network, a random forests regression, and a support vector machine. These three models were selected as a result of information gathered in the literature review that supported their potential. These models were developed by using the same theoretical dataset generated by the Pavement ME software, and all were used to predict rutting. This allowed for direct comparison of their performance; the best performing model type was then used to develop the remaining models in this study.

The datasets collected in Task 2 and organized in Task 3 were used to develop multiple artificial intelligence models that could predict pavement performance measures and back-calculate the moduli of pavement layers. Performance was measured with IRI, total rutting, rutting of only the asphalt layer, and cracking. Models based on theoretical and field datasets containing structural, environmental, and traffic data were developed to predict these performance indicators. Performance decay curves were also obtained for the applicable models.

1.3.4. Task 4. Final Report

The research team prepared this final report to provide background on the subject, information on the selected chemical products, the testing matrix, the results, and recommendations.

1.4. Organization of This Report

This report documents the research methodology, presents the results and analysis, summarizes the findings, and provides recommendations for future studies and implementation. The report has five chapters.

Chapter 2 details the main findings from the literature review concerning different methods of artificial intelligence, the application of those methods to civil and pavement engineering, and how pavement deterioration relates to pavement management.

Chapter 3 provides information on the different datasets used in this study. It details the collection of these datasets and describes the different parameters found within the data. As this research study used multiple sources of both theoretical and field data, this chapter reviews the differences in these datasets to explain the purpose for using multiple sources of data.

Chapter 4 discusses the results for the artificial intelligence models developed using data comprising the material properties and structural information for each pavement section. These models were developed to predict various performance indicators of asphalt pavements, and different models were developed using theoretical and field datasets. This chapter also investigates different methods of artificial intelligence by comparing three models that were trained on the same dataset and developed to predict the same performance measure. This investigation was performed to determine the best method of artificial intelligence to use for the other models.

Chapter 5 summarizes the main findings and conclusions of this study and provides recommendations for future research. The appendices provide examples of the datasets and the Python codes for the artificial intelligence models used in this research.

CHAPTER 2. Background

2.1. Pavement Management

2.1.1. Pavement Characteristics and Deterioration

Asphalt pavements consist of several layers, including the asphalt layer at the top, followed by the base layer, subbase layer, and subgrade layer. The overall performance of the pavement is affected by the material properties of each layer, the thickness of each layer, applied traffic, and environmental conditions. Asphalt pavements experience various distresses in the field, including rutting (or permanent deformation), cracking, and moisture damage. Rutting, cracking, and roughness increase with time. Pavement engineers conduct regular distress surveys to rate the conditions of pavement and determine the need for surface treatments. Performance decay models are often used in an asset management system to predict future performance in response to maintenance and rehabilitation treatments.

Numerous factors cause pavement deterioration. In most cases, traffic level has a significant impact on pavement deterioration. Pavement characteristics such as the layers' elastic moduli and strength affect how the pavement deteriorates. Environmental factors also influence this deterioration. Environmental factors include moisture, how often freeze-thaw occurs, whether the sections are underlain with permafrost, and many others (Hossain et al., 2017).

2.1.2 Measures of Pavement Condition and Pavement Distresses

There are many ways to measure the performance or condition of a pavement. One of the most popular methods is the International Roughness Index (IRI), which can be used to indicate ride quality. The IRI is calculated from longitudinal road profiles and is typically reported in measurements of length per length such as inches per mile (Hossain et al., 2017).

Another common measure of pavement performance is the pavement condition index (PCI). The PCI is scored on a scale of zero to 100 and is intended to reflect the pavement's overall condition. This score is determined on the basis of visual inspection of the number and types of distresses in a pavement (Kumar et al., 2021).

Cracking is one of the major distresses that can be inspected visually. There are many types of cracking, such as fatigue cracking, longitudinal cracking, transverse cracking, bottom-up cracking, and top-down cracking. Fatigue cracking, also known as alligator cracking, is caused by inadequate structural support for the given loading and consists of a series of interconnected cracks that may look like the back of an alligator. Longitudinal cracking runs parallel with the centerline of the pavement. It may be caused by reflective cracking, poor joint construction, or top-down cracking. It can be measured for both the wheel-path and non-wheel-path. Transverse cracking, on the other hand, runs perpendicularly to the centerline of the pavement. This is usually a type of thermal cracking and can be caused by the shrinkage of the hot-mix asphalt (HMA). Top-down cracking starts at the surface of a pavement and begins to progress deeper within the HMA layer of the pavement. Bottom-up cracking begins to form at the bottom of the HMA layer and then travels upward toward the surface (Pavement Distresses, 2006).

It is important that a pavement management system (PMS) be used to collect, report, and predict the conditions of pavement sections so that agencies can make the best decisions to manage those sections. Good pavement management systems can save public funds (Salini et al., 2015). At a network level, this can only be achieved through proper planning. If the PMS offers a good prediction of what pavement sections' conditions will be in the coming years, it will greatly aid the planning process. This allows for a better distribution of resources, better use of public funds, and better maintained roadways (Salini et al., 2015).

2.2. Different Artificial Intelligence Methods

2.2.1. What is Artificial Intelligence?

Artificial intelligence (AI) is a branch of computer science in which computer programs perform tasks and solve problems that would typically require human intelligence (Artificial Intelligence (AI) vs. Machine Learning, 2022). The term was first coined in 1956 by a group of researchers at Dartmouth College (Dick, 2019). Since then, artificial intelligence has advanced and changed immensely. Today, there are many different subsets of AI, and the term AI is often used interchangeably with the term machine learning, although they are not synonymous. Machine learning is a subset of AI in which a program automatically learns insights and recognizes patterns in data through algorithms and then applies that learning to make increasingly better decisions (Artificial Intelligence vs. Machine Learning, 2022).

There are three groups of machine learning models: supervised learning, unsupervised learning, and reinforcement learning (Justo-Silva et al., 2021). The type of machine learning model that was investigated in this study was supervised learning. Supervised learning models use input and output data to make predictions for new data and are typically "used for project-level or network-level pavement management" (Justo-Silva et al., 2021).

Most machine learning techniques split data into training and testing sets. The model uses the training data to learn, and the testing set is used to validate that model to see whether it can accurately model unseen data. The percentage of data used in each of these sets is dependent on the variability of the data as well as the number of samples available.

2.2.2. Artificial Neural Networks (ANN)

An artificial neural network (ANN) model is a machine learning technique that uses supervised learning to solve problems. ANNs are designed to mimic the way neurons fire in the human brain, and they are a black box in which the operator does not get to see the model's decision-making process. They comprise an input layer, a specified number of hidden layers, and an output layer all containing nodes. These nodes, mimicking neurons, connect to one another, and each has a designated weight and threshold. If the node's output is above the threshold, then the node activates and sends data to the next layer (Bishop, 1994).

ANNs can have issues with overfitting. Overfitting describes a model that does not generalize data well. This is indicated by high accuracy with the training set and significantly decreased accuracy with the testing set. It is difficult to deal with overfitting in ANNs because they have a slow runtime. This slow runtime is due to their iterative process, and this process contributes to ANNs' vulnerability to overfitting. Dropout can be used to reduce overfitting and thus can improve neural networks. Dropout has been found to improve neural networks in object classification, speech recognition, analysis of computational biology data, digit recognition, and document classification. However, dropout increases the training time of the network by two to three times in comparison to a standard ANN with similar architecture (Srivastava et al., 1970).

2.2.3. Decision Trees and Random Forests

Decision trees consist of nodes that split into two or more directions depending on the given input. The nodes split a designated number of times to reach a conclusion. Decision trees have been used in applications outside of AI, but they are also a popular machine learning technique. On the basis of the inputs and number of splits decided by the operator, the decision tree finds the best criteria for these splits to come up with the most accurate results. Decision trees are commonly used for classification problems, and they have the benefit of giving the operator more knowledge about how the decision was reached (Rokach and Maimon, 2005).

Random forests algorithms are like decision trees, but instead of generating only one tree, they generate multiple trees. They then generate an output based on the majority voting or average answer of all the decision trees for classification or regression problems, respectively. They overcome some of the issues with decision trees such as overfitting and not being robust to outliers (Gong et al., 2018).

2.2.4. Support Vector Machines

Support vector machines (SVM) are another machine learning technique. They operate by creating hyperplanes that have an axis for each input parameter, and then they find the optimal plane for identifying and predicting points. The SVM algorithms are typically robust to outliers and have good generalization capabilities. They were initially developed for classification problems, but they can also be used for regression (Hasan and Ziari, 2009).

2.2.5. Expert Systems and Fuzzy Logic

Expert systems are a form of machine learning often combined with fuzzy logic. Expert systems are simply systems designed to mimic a human expert, and fuzzy logic is mean to resemble human reasoning. Fuzzy logic works by using true/false statements on multiple different levels to achieve the output. It can often aid in confronting uncertainty (Kaur and Pulugurta, 2008). Expert systems are efficient in problem solving because they use expert knowledge and human reasoning that cannot be implemented analytically because of the complex nature of the problems (Ismail et al., 2009).

2.2.6. Other Types of Machine Learning

There are countless different machine learning algorithms, such as genetic algorithms, long-term/short-term memory (LSTM), intelligent search algorithms, structural equation modeling (SEM), and group method of data handling (GMDH). Genetic algorithms are intended to mimic natural selection (Elhadidy et al., 2015). LSTM models are typically used for time series data, and they work by forgetting/removing unhelpful information, inputting useful information, and outputting the most important information (Hosseini et al., 2020). Intelligent search algorithms work by finding the shortest path between two cells on a grid (Tohidi et al., 2022). SEM is like factor analysis combined with multiple regression analysis, used to measure the relationship between independent variables and dependent variables (Chen et al., 2016). “The basic GMDH algorithm is a procedure for constructing a high-order polynomial of the form which relates m input variables x_1, x_2, ..., x_m to a single output variable, y” (Ziari et al., 2015).

2.3. Applications of Artificial Intelligence in Civil Engineering

2.3.1. General Information About Artificial Intelligence Usage in Civil Engineering Applications

Artificial intelligence models are gaining popularity in civil engineering applications and research studies have been conducted to explore various types of machine learning and how they can be used in various applications. A review article by Lu et al. (2012) discussed various methods of artificial intelligence that can be used in civil engineering. It summarized different methods of evolutionary computation, swarm intelligence, neural networks, fuzzy systems, expert systems, and a few methods that cannot be placed in a category. It also discussed the future trends for artificial intelligence, such as further development of fuzzy systems, the possibility of hybrid systems, and increased research in possible applications for these systems.

Justo-Silva et al. (2021) summarized information about the differences in various AI techniques for pavement performance prediction models. They reviewed many models used in the development of pavement performance prediction models (PPPM). PPPMs relate pavement condition to a set of variables such as traffic loading, environmental conditions, etc. PPPMs can be classified by the type of formulation, conceptual format, application level, and types of variables. Machine learning algorithms find generalizable predictive patterns in datasets, and can be classified as either supervised learning, unsupervised learning, or reinforcement learning. Both project-level and network-level pavement management applications can utilize supervised learning. Unsupervised learning can be used in exploratory and clustering analysis applications, and reinforcement learning can aid decision makers for both project-level and network-level pavement management (Justo-Silva et al., 2021). Table 2.1 summarizes the trade-off characteristics for various algorithms examined by Justo-Silva et al. (2021).

A study conducted by Piryonesi and El-Diraby (2021) compared the accuracy of various forms of AI when they were applied to predict the pavement condition index. They used the LTPP dataset to predict the reduced PCI values in the next two to six years based on the current PCI and climate data. They chose this timeframe because roads without maintenance for over six years often begin to deteriorate. The training set contained over 3,000 samples in their study. They demonstrated that their models were more accurate than previous models because of the size of the dataset in comparison to those of other studies. The larger dataset allowed the models to see a greater range of possibilities, which increased their overall accuracy. The highest accuracy was found by assembling machine learning algorithms based on decision trees and a naïve Bayes classifier coupled with kernel estimation. They calculated the current PCI based on the distresses and their severity levels, and they did not use highway sections that had undergone maintenance during the prediction horizon (Piryonesi and El-Diraby, 2021).

Table 2.1 Trade-Off Characteristics of Some Machine Learning Algorithms (after Justo-Silva et al. 2021).

Type of Algorithm	Prediction Speed	Training Speed	Memory Usage	Required Tuning
Linear Logistic Regression	Fast	Fast	Small	Minimal
Linear Support Vector Machines	Fast	Fast	Small	Minimal
Decision Trees	Fast	Fast	Small	Some
Nonlinear Support Vector Machines	Slow	Slow	Medium	Some
Nonlinear Logistic Regression	Slow	Slow	Medium	Some
Nearest Neighbor	Moderate	Minimal	Medium	Minimal
Naïve Bayes	Fast	Fast	Medium	Some
Ensembles	Moderate	Slow	Varies	Some
Neural Networks	Moderate	Slow	Medium to Large	Significant

2.3.2. Artificial Neural Networks in Pavement Engineering Applications

ANNs have been used in several applications of pavement engineering. For example, a case study conducted in Montreal by Zhang et al. (2021) created a convolutional neural network model for pavement distress detection. The study investigated an automated methodology for pavement distress detection and type classification by using low-cost video data paired with convolutional neural networks. The study was divided into four steps: data collection, dataset preparation, image labeling, and building deep neural networks to learn the patterns of different distress types in pavement images. The data used in this case study were images taken from a GoPro camera mounted on the front of a vehicle that took a photo every second. For this method to work with a deep learning model, it was necessary that the models had enough examples of each distress so they could correctly identify the distress based on the division categories.

Distress classes were categorized as patching, pothole, linear crack, and network crack (Zhang et al., 2021). The results of this study found that in comparison to a laser crack measurement system (LCMS), the ANN system achieved similar accuracy while being more economical and simpler to replicate. Also, the results showed that the use of an embedded system integrated with a deep neural network could be installed on vehicles and used for automatic pavement distress type detection as well as for classification of road facilities. The proposed system utilized a sports camera that could be easily installed on any vehicle and could be used in any location (Zhang et al., 2021).

Issa et al. (2021) used this type of model to predict the pavement condition index. They predicted PCI values based on data about pavement distresses, including the density of distresses, the severity of distresses, and the number of manholes present. There were four input variables: stress type, stress severity, section width, and number of manholes per section. This system was coded with 41 neurons in the input layer once redundant variables had been removed. This approach yielded results of an average of 25 percent standard error across all groups for predicting PCI, with R^2 values of 0.98 or higher for every group (Issa et al., 2021).

Another study by Dimeter et al. (2018) used an artificial neural network to calculate Global Pavement Index (GPI) and assign an appropriate maintenance strategy. They took measurements of various pavement parameters such as longitudinal evenness, rut depth, texture depth, surface cracks, and patches. These were obtained from national roads with a total length of 481.3 km, which were then divided into approximate 1-km sections (471 sections total). They divided these into three separate databases. The first database calculated GPI on the basis of IRI, rut depth, mean profile depth, cracks and patches (expressed as percentage of area affected). The second database determined a maintenance strategy based on the same parameters as the previous database. The final database was an ANN that was used to calculate both the GPI and a maintenance strategy, and the results of this database were compared to the results of the previous two. The results showed that the ANN was able to accurately predict GPI 87 percent of the time and was able to choose the correct maintenance strategy 95 percent of the time (Dimeter et al., 2018).

An ANN does not have to be created from scratch. Issa et al. (2021) used the neural network toolbox available in MATLAB (2015 version) to train a model to predict PCI. They had roughly 400 sections that were 10 km long. They evaluated three different ANN algorithms to predict PCI with various pavement distresses. The highest R value they achieved was 89 percent and the lowest mean squared error value was 4 percent. They reiterated that models can be improved by adding more data points and by incorporating other factors such as climatic condition, traffic volume, etc. (Issa et al., 2021).

Neural networks can also be used to predict IRI. Hossain et al. (2017) developed a prediction model that was used to predict IRI solely on the basis of climate and traffic data. They collected these data from the LTPP database. These data were then used in an ANN designed with the neural network toolbox in MATLAB. The ANN model was trained using half of the climate, traffic, and IRI data, and the other half of the data was used to validate the model. The model was validated by comparing the IRI prediction results from the ANN with the measured IRI values for flexible pavements. The ANN was a feed-forward back-propagation model. This model type was chosen because of its simplicity in training multiple inputs. The researchers were able to achieve a high correlation between the LTPP data and the model predictions (Hossain et al., 2017).

An advantage of neural networks is that they can analyze large amounts of data. Shahnazari et al. (2012) conducted a study in which they used a database comprising PCI results for over 1,250 km of highways in Iran, and they used 12,487 segments of those highways to develop the models. They used both an ANN-based model and a Gaussian Processes (GP)- based model to analyze the data. The inputs for these models were representative of the type, severity, and quantity of the distresses, and the output was a PCI estimation. They found that cases with a PCI value below 40 had relatively larger prediction errors than the remaining cases, which was likely due to a higher concentration of cases with PCI values above 60. Despite this increase in errors for lower PCI values, the errors remained within an acceptable range (Shahnazari et al., 2012).

Another study by Sollazzo et al. (2017) examined 342 different test sections from almost all the states available in the LTPP database. The sections they focused on were asphalt concrete sections, and they excluded maintenance and rehabilitation operations from the dataset (they considered only measurements for each section until the first maintenance operation was performed). For these sections, they considered 13 different parameters. The parameters were total pavement thickness; asphalt layer thickness; average annual equivalent single axle load (ESAL) values in the LTPP lane; average annual estimated daily number of trucks in the LTPP lane; average temperature (mean of annual average temperatures in chosen years); standardized temperature range; average number of days with an average temperature above 32 degrees Celsius; average number of days annually with a temperature below freezing; time passed since the first profilometer survey; first measured IRI; IRI at a specific time for each section, which was an average of the left and right wheel paths; SN_{eff} at a specific time; and average pavement surface temperature during the deflection test. Sollazzo et al. (2017) trained three ANNs, with each ANN developed for a different temperature range. All these ANNs contained 25 hidden neurons, and the records related to them were divided such that 70 percent of the data was used in the training group, 15 percent was used in the validation group, and the remaining 15 percent was used in the testing group. The training of these ANNs was performed with the Levenberg-Marquardt algorithm and measured by mean-squared error (Sollazzo et al., 2017).

Another parameter that can be modeled with ANNs is the present serviceability index of flexible pavements. Terzi (2007) gathered performance data from AASHTO test results that contained 74 sections. From these data, the researcher chose the input variables of slope variance, rut depth, cracking, patches, and longitudinal cracking. The output was chosen as panel data, and the results from three different ANNs were compared. The best results achieved an R^2 of 0.99 for the training set and 0.87 for the testing set (Terzi, 2007).

Artificial neural networks have also been used in pavement engineering to study the correlation between various pavement distresses and overall pavement roughness. Lin et al. (2003) examined the correlation between ten different types of common pavement distresses in Taiwan and pavement roughness. This study was conducted by using a back-propagation neural network. The ten types of distresses they investigated were rutting, alligator cracking, other cracking, digging/patching (digging due to pipe installation, electrical work, etc.), potholes, corrugation, manholes, stripping, patching, and bleeding. They used 125 sections that were each 1 km long. These sections included roadways on both highways and country roads. Information on these sections was collected by an automatic road analyzer (ARAN). Given these data, the neural network was able to use 100 records of training data to achieve a root mean square of 0.068 and a correlation coefficient of 0.84. For the testing records, they used 25 sections and were able to reach prediction R^2 values of 0.94, which exceeded that of the training data. The results of this study demonstrated that the IRI can be used to evaluate the quality of pavement projects and can fully represent the pavement deterioration process. This indicated that IRI can be used as the basis for road maintenance ranking evaluation (Lin et al., 2003).

Lou et al. (2001) developed back-propagation neural network (BPNN) models to forecast the short-term variation of the crack index (CI) of Florida's highway network. They used both single-year and multi-year BPNNs in their study. They gathered the data from the Florida Department of Transportation's pavement performance database, which included data from the previous 20 years. The network they developed contained an input layer, one hidden layer, and an output layer. The input layer, hidden layer, and output layer had seven neurons, 12 neurons, and a single neuron, respectively. This network was designed to forecast the CI for the following year when given data from the previous three years (Lou et al., 2001).

2.3.3 Decision Trees and Random Forests Usage in Pavement Engineering Applications

Rajagopal (2006) used a decision tree method to predict pavement performance for the city of Cincinnati. This research used the Information Management System (IMS) database and the ViPERS database as sources for all their data. These data were used to predict (PCR) in the first model. The decision tree was used to select a reasonable maintenance or rehabilitation alternative for each project. This would assist engineers in determining the best possible strategy for maintenance. The decision tree method split on the basis of the PCR value. Different maintenance strategies were then suggested based on that split (Rajagopal, 2006).

Like decision trees, random forests or random decision trees work by constructing a group of decision trees and then forecasting the classification or mean regression of the individual trees. They are ensemble learning methods that can be used for regression, classification, and other tasks. Gong et al. (2018) developed a random forest regression model to predict pavement IRI based on various inputs. These inputs included various distress types, traffic history, structural information of the pavement, and climate conditions. They used a dataset comprising over 12,300 distress data samples, 19,900 IRI data samples, and 28,7000 rutting data samples that were collected from the LTPP database (Gong et al., 2018). Using 18 variables from these data, the researchers were able to achieve an R^2 of 0.998 for the training set. This was 2 percent higher than that for test set, in which the R^2 was 0.975, which was still very high. A lower R^2 in the testing set could have been an indication of overfitting, but because both R^2 values were so high, this did not pose a serious problem (Gong et al., 2018).

Another variation of decision trees is the gradient boosted decision tree (GBDT). Guo et al. (2021) used a GBDT to predict IRI and rut depth. They gathered their data from the LTPP database, which comprised more than 1,600 records. The inputs chosen from this database covered the climatic, traffic, and structural factors for each section. The authors noted that the ANN and SVM methods could not determine which input factors had the greatest influence on the result and whether their impact was positive or negative. They addressed this issue by using the lightGBM package in Python (a gradient boosted decision tree framework) to develop their prediction model. These can produce better results because they construct multiple learners and combine them to optimize the results as opposed to using only a single learner (Guo et al., 2021).

2.3.4 Fuzzy Expert Systems Usage in Pavement Engineering Applications

Fuzzy expert systems are another popular form of machine learning that is often used in engineering applications, and they can even be combined with other forms of machine learning to improve the accuracy of those models. Kaur and Pulugurta (2008) created a fuzzy decision tree and compared its accuracy to that of logistic regression. They considered three separate fuzzy models based on subgrade type (clay, sandy clay, and sand). Other parameters of the fuzzy model included surface thickness, age of the road, and total traffic count. They used an adaptive neuro fuzzy inference system (ANFIS) to develop a pavement performance prediction model. The results concluded that the fuzzy decision tree had a higher accuracy than the logistic model. The fuzzy decision tree was also easier to develop than the logistic model (Kaur and Pulugurta, 2008).

Ismail et al. (2009) summarized the development and potential usage of expert systems in pavement management. They suggested that the use of expert systems can offer significant advantages over traditional computerized models. This is possible because expert systems are efficient in problem solving because they utilize human reasoning and extensive knowledge from experts that are too complex to be represented in an analytical fashion (Ismail et al., 2009).

2.3.5 Other Artificial Intelligence Model Usage in Pavement Engineering Applications

Genetic algorithms (GAs) have been used in pavement condition rating. Elhadidy et al. (2015) assessed pavement condition by using a zero to four scale that was developed by the Federal Highway Administration. This system assumed that pavements were serviceable until the rating reduced to a value of one, which indicated poor condition and that major maintenance was needed. A rating of zero would indicate that the pavement had failed and was beyond corrective action. They used a Markovian deterioration model in this research to estimate the future decline in pavement condition. This was necessary for a PMS to be able to select an appropriate rehabilitation strategy. Through these methods, they were able to obtain the optimal maintenance actions based on the prediction of the pavement condition (Elhadidy et al., 2015).

Hosseini et al. (2020) conducted a study in Iowa that used deep learning to model pavement deterioration. The data in this study were obtained between 1998 and 2018 and comprised dates of construction, dates of reconstruction, section identifiers, information about the highway system classification, pavement ride quality data, and pavemented distress data that were automated (Hosseini et al., 2020). The pavement types studied were asphalt concrete (AC), Portland cement concrete (PCC), and composite (COM) pavements. Pavement distress information was collected, including rutting, cracking, transverse cracking, longitudinal cracking, alligator cracking, wheel-path cracking, and patching. The severity levels of these distresses were also determined. Ride quality was also characterized by using the IRI for all pavement types. After these data had been collected and organized, condition indices were estimated by using riding index, rutting index (AC and COM only), cracking index, and faulting index (PCC only). The researchers concluded that prediction accuracy was higher for AC pavements with the long-term/short-term memory (LSTM) model than with the individual DOT regression models (Hosseini et al., 2020).

Tohidi et al. (2022) used intelligent search algorithms in the cost optimization of road pavement thickness design. This study used optimization techniques to find a systematic solution for a variety of decision-making problems so that an optimal solution could be found that required spending as little time as possible on calculations. The dataset used contained classification of the vehicles, their weight and axle properties, the tolerable range of thickness for different pavement layers, the equivalent load factor for different types of axles, the level of reliability, the standard deviation, and the ultimate serviceability index. Through a comparison of various techniques, the article suggested that particle swarm optimization was superior to a GA for pavement system management applications.

Bianchini and Bandini (2010) combined multiple forms of AI to improve the accuracy of a model. They used fuzzy theory to perform the reasoning part and an ANN to determine the numerical components of the membership functions through an adaptive process extracted from the data. They used the MnROAD test site database available on Infopave as the source of input and output parameters for the training and validation phases. They chose seven input variables, including the surface curvature index, the deflection measured at 36 inches from the point of load application, the area under the pavement profile, the rut depth, the percentage of the area of the section affected by fatigue cracking, the thickness of the asphalt pavement layer, and the traffic in terms of ESALs. They had different models for different seasons (deep frost, beginning of spring, late spring, summer, and fall). The accuracy of their results was shown as goodness of fit statistics, and most of the R^2 values for their models ranged from 0.987 to 0.809 (Bianchini and Bandini, 2010).

Chen et al. (2016) examined another form of AI, structural equation modeling (SEM), to develop a distress condition index of asphalt pavements. SEM deals with the relationship between observed variables measured to reflect latent variables and the actual latent variables using a statistical approach (Chen et al., 2016). SEM comprises two major components: the measurement model and the structural model. The measurement model is used to determine how well latent variables are measured by various observed variables. The structural model describes how the latent variables relate to each other. They had to modify this model to improve the goodness of fit until it was deemed acceptable (Chen et al., 2016). For this model, they chose their latent variable to be the overall pavement condition index. Other factors such as pavement age, material, layer thickness, environmental factors, and traffic data were treated as endogenous and exogenous variables. According to their prediction model, there were twelve distress types that had a significant impact on expressing the latent condition index. These included fatigue cracking, longitudinal cracking in both the wheel path and not in the wheel path, block cracking, transverse cracking, patches, bleeding, and edge cracking. Rutting was also shown to be important, but because of a lack in rutting data in the LTPP database, it was not taken into consideration (Chen et al., 2016).

Piryonesi and El-Diraby (2021) conducted a study to compare the impact of performance indicators on flexible pavement deterioration modeling. This study used the LTPP database to predict the IRI and PCI of asphalt pavements by training machine learning algorithms. From the LTPP database, they gathered and prepared 30,274 IRI records and 3,227 PCI records to train the models. The data were retrieved with structured query language (SQL) and automated with Python. The researchers used two decision trees, RFs, GBDTs, linear regression, and random forests regression algorithms. With these algorithms, the highest cross-validation accuracy they were able to achieve with IRI was 0.95 and with PCI was 0.84. They noted the lack of research comparing how machine learning performs for IRI versus PCI and that this was likely due to a lack of measurements of PCI data in major databases. They indicated that the LTPP database does not measure it. By comparing PCI and IRI prediction results, they deduced that IRI values were easier to predict than PCI values. The prediction of PCI values was improved by using more complex algorithms. They also noted that the initial IRI value had a greater impact on the IRI prediction than the initial PCI did for the PCI prediction. They also noted that they were able to achieve the highest accuracy in a dry climate with no freezing (Piryonesi and El-Diraby, 2021).

Group methods of data handling (GMDH) algorithms are another form of AI that has been used in pavement engineering. Ziari et al. (2015) used the LTPP database to train and test this algorithm. They considered only asphalt concrete pavement on a granular base for this research. They also selected pavements that had not received any maintenance or rehabilitation because they would have a continuous change in IRI. Then they eliminated any sections that had an average annual daily traffic (AADT) of greater than 1,000 or less than 100 in their life cycle. Given the remaining dataset, they chose the nine input variables of AAT, AAP, average annual daily truck traffic (AADTT), AADT, PT, ESALs, and ST. They chose the output variable to be IRI. Once all these parameters had been determined, they prepared 26 sections comprising 206 rows of annual data. They ran these data through a GMDH algorithm in MATLAB and achieved an R^2 of 0.9 (Ziari et al., 2015).

Hosseini (2020) used the long-term/short-term memory (LSTM) method to develop a pavement prediction model. This research was divided into two major parts. The first described the process and outcome of deterioration modeling for three pavement types. The second described how the accuracy of prediction models could affect the decisions made in terms of cost. This study used results from a pavement prediction model developed with LSTM. Five scenarios were assumed, from maximum to minimum error rate, to investigate the impact that increasing error had on decision making. Different rates of error (10 percent, 30 percent, 50 percent, 70 percent, and 90 percent) were added to the predicted values of performance indicators for each scenario. The results indicated that increased error had a significant correlation with the cost of maintenance activities. The researchers indicated that transportation agencies need to improve the prediction accuracy of their models to reduce unnecessary costs (Hosseini, 2020).

2.3.6 Artificial Intelligence Applications in Other Areas of Civil Engineering

AI has also been used in other various applications in civil engineering. For example, Shaheen et al. (2009) used fuzzy expert systems as a predictive tool in construction. They discussed how fuzzy expert systems work best when enough expert knowledge is available. In contrast, they also discussed that it has a weakness when if-then rules do not have a defined structured approach (Shaheen et al., 2009). They described how to optimize these systems by using neuro fuzzy techniques that use a combination of the explicit knowledge represented by fuzzy expert systems and adaptive neural networks' learning powers (Shaheen et al., 2009). They then described a case study on how a fuzzy expert system was used to determine tunneling time based on several inputs (Shaheen et al., 2009).

Machine learning has also been used in geotechnical engineering. However, Shahin et al. (2009) demonstrated that there has been little improvement in ANN development since the mid-1990s. A possible solution to this would be utilizing a systematic approach in ANN development to improve model performance. Since ANNs are a black box, an approach for improvement would need to address the determination of adequate models, choice of suitable network architecture, data division and preprocessing, selection of parameters that control optimization, model validation, stopping criteria, and other major factors. Model robustness, transparency and knowledge extraction, uncertainty, and extrapolation are also areas in which ANNs fall short. The authors stated that if these matters were addressed and improved, then ANNs would be considered an alternative to conventional methods as opposed to a supplementary method as they are currently. However, despite these flaws, ANNs have had success in geotechnical engineering and other disciplines. They have been used to model axial and lateral load capacities of deep foundations in both compression and uplift, as well as shallow foundation behavior such as settlement estimation and bearing capacity (Shahin et al., 2009).

Ebid (2020) noted that over 600 papers have been published since 1984 covering topics concerning the application of AI in geotechnical engineering. The author discussed the various applications and different types of AI covered by research, and which topics and AI types were the most popular. This research summarized various types of artificial intelligence and discussed how and why ANN is the most popular technique. It also noted how research involving artificial intelligence in geotechnical engineering appears to be increasing at a near exponential rate (Ebid, 2020).

Chapter 3. Selection of Datasets

3.1 Introduction

This study utilized multiple theoretical and field datasets to develop models that predict pavement performance and back-calculate the moduli of different pavement layers. The theoretical datasets utilized in this research covered a large number of sections with a wide range of properties. These types of datasets are ideal for developing artificial intelligence models because they introduce the models to large numbers of possibilities and allow the models to analyze the direct relationship between the model inputs and outputs without the inconsistencies found in field datasets. The models developed with these theoretical datasets allowed the researchers to determine the best type of models to use for similar field data because they indicated the models' potential. However, field datasets were also a necessary part of the research to validate the models and ensure that they could work in practice. Field datasets present inconsistencies in the data, and capturing these inconsistencies is important for determining the accuracy of performance models.

3.2 Long-Term Pavement Performance Dataset

The Long-Term Pavement Performance (LTPP) data are publicly available and were extracted from the Infopave website in 2021. The LTPP program was founded by the Strategic Highway Research Program (SHRP) in 1987 and is now managed by the Federal Highway Administration (FHWA nd). It aims to study in-service pavement sections and their performance to better understand why they perform the way that they do. The LTPP database contains information from 2,509 pavement sections throughout North America, and the LTPP program continues to collect data from over 700 sections(LTPP 2009).

In this project, all of the available pavement sections from the Pacific Northwest were selected. These comprised 98 sections, including six sections from Alaska, 22 sections from Idaho, 35 sections from Montana, six sections from Oregon, 16 sections from Washington, and 13 sections from Wyoming. From these 98 sections, eight had to be removed because of a lack of sufficient information regarding either the pavement structure or performance data.

The remaining sections were all flexible pavements. For each section, climate data were collected regarding moisture and temperature. Each section had data for the average annual precipitation measured in inches, the average annual temperature in degrees Fahrenheit, the Freeze Index in degrees Fahrenheit times degrees days, and annual maximum and minimum humidity in percentage. These climate data were summarized into four different climate zones to simplify the model inputs and reduce the chances of overfitting. The four different climate zones were wet, freeze; dry, freeze; wet, no-freeze; and dry, no-freeze.

The sections collected from the LTPP database also included their Global Positioning System (GPS) coordinates measured in latitude and longitude and the date the sections were constructed. In addition, the years of all minor and major maintenance treatments on the sections were recorded, as well as the types of maintenance that occurred. Types of construction events included aggregate seal coat, single layer surface treatments, pothole patching, spot patching, crack sealing, milling off the existing asphalt layer and overlaying with a new asphalt layer, asphalt concrete overlay, fog seal coat, shoulder restoration, mechanical premix patch, overlay with hot mix recycled asphalt, and slurry seal coat. Minor maintenance treatments were not considered in the dataset used to train and test the models, but major maintenance treatments were captured by treating sections where major treatments occurred as "new" sections after the treatment occurred. Because an overlay would have such a major impact on the performance of the pavement, the age of the overlaid section was reset to zero, but the cumulative traffic on that section was carried over. This allowed the model to better understand the other traits of the sections and led to better decision making.

The data for the pavement structure were also available for these sections. The thicknesses of the asphalt and base layers were recorded, as well as the types of material(s) for the asphalt, base, and subgrade layers. The material type of each pavement layer was used to represent the material properties of that layer and was represented by using a number code for each material type. The thicknesses of the asphalt and base layers were used as structure information.

Information was also recorded regarding the traffic and usage of these sections. The functional class of these sections was included, which contained information regarding whether the section was a major or minor road, whether it was rural or urban, and whether it was principal or arterial. The traffic on these sections was recorded in equivalent ESALs, AADT, and AADTT. These traffic measurements were recorded for some of the years that the sections were in service. For years that traffic data were not recorded, estimations were made on the basis of the average traffic from the nearest previous and following years. This allowed for the cumulative traffic on each section to be estimated.

Lastly, there was also information regarding the performance of the pavement sections. The LTPP program collects International Roughness Index (IRI) measured in inches per mile, fatigue cracking measured in squared feet, longitudinal cracking for both the wheel path and non-wheel path measured in feet, transverse cracking measured by count, and rutting measured in inches. These data were available for some of the years the sections were in service.

The data were organized into a single Excel sheet with each year containing performance data for each section listed in a separate row. Included on the sheet were the age of the section, the cumulative traffic on the section in ESALs, the climate zone, the major type of material in each of the pavement layers, the thicknesses of the asphalt and base layers, and the performance data. The performance data were used as provided, and the years for each section that did not contain performance information were excluded. All of the data that were not numerical were coded into different numerical values, and those values were used for the given parameter. Table 3.1 shows an example of data from a few different pavement sections in this dataset that were used in the rutting prediction model. Labels were not included in the dataset that was entered into the model because they would not have any effect on the performance of the sections, but they were added here to allow a better review of the data. More examples of this dataset can be found in Appendix A.

Table 3.1 Sample Data LTPP Dataset

Section Label	Section Age (years)	Climate Zone	Subgrade Layer Material	Base Layer Thickness (in)	Asphalt Layer Thickness	Asphalt Layer Material	Base Layer Material	Cumulative Traffic (KESAL)	Rutting (in)
Section A	16	2	254	9.2	5.1	1	308	1693	0.3
Section A	17	2	254	9.2	5.1	1	308	1841.5	0.2
Section A	18	2	254	9.2	5.1	1	308	1990	0.2
Section A	21	2	254	9.2	5.1	1	308	2435.5	0.3
Section A	22	2	254	9.2	5.1	1	308	2546.5	0.3
Section B	14	2	267	11.6	4.2	1	304	469	0.4
Section B	15	2	267	11.6	4.2	1	304	532.5	0.4
Section B	19	2	267	11.6	4.2	1	304	786.5	0.3
Section C	17	1	282	19.4	7.5	1	304	682	0.4
Section C	18	1	282	19.4	7.5	1	304	726	0.4
Section C	19	1	282	19.4	7.5	1	304	770	0.4
Secton D	15	1	141	9.2	11.2	1	304	1926	0.4
Secton D	16	1	141	9.2	11.2	1	304	2318	0.5

3.3 Pavement ME Dataset

This dataset was a theoretical dataset generated by the AASHTOWare Pavement ME software. This software uses a mechanistic-empirical approach to design and analyze the performance of pavements (ME Design Guide, 2008). A parametric study was performed that contained 243 pavement designs with different layer thicknesses and moduli, as presented in Table 3.2. These pavement structures represented typical pavement sections for low-, intermediate-, and high-volume traffic roads in Idaho. The pavement structures were generated on the basis of a historical database of Idaho pavement sections created in a previous study (Bayomy et al. 2018). The AASHTOWare Pavement ME predicted and reported the performance of these test sections recorded every 0.08 years for their respective 20 total years of service. This totaled 57,120 total data points. These sections were designed to have no maintenance activities during their service life so that performance decay curves could be captured accurately using predictions from the artificial intelligence models. They were also designed to be in the same location so that climate data could be excluded from the models. The Pavement ME data used in this research was produced in ITD RP 294 – Simplified Analysis Methods of Traffic Speed Deflectometer (TSD) and Falling Weight Deflector (FWD) Data.

These sections contained information on the structure of the pavement. This included the thicknesses of the asphalt and base layers of the pavement measured in inches. It also included information regarding the material properties, such as the moduli of the asphalt, base, and subgrade layers measured in ksi.

These sections also contained data on the cumulative traffic for each section. The cumulative traffic was measured every 0.08 years for the entire 20-year lifespan of the pavements, and it was measured in ESALs. These traffic measurements were compiled cumulatively to achieve performance decay models as a function of traffic.

In addition to traffic, these sections also had data that measured performance with different parameters. The parameters used were the depth of thermal cracking in inches, the IRI in inches per mile, the total pavement deformation (rutting) in inches, the permanent deformation of the asphalt layer only in inches, the top-down fatigue cracking in percentage of lane area, and the bottom-up fatigue cracking in percentage of lane area.

The data were organized into a single Excel sheet, with each time interval containing new performance and traffic data for each section listed in a separate row. The cumulative traffic on the section in ESALS, the modulus of elasticity of each of the pavement layers in ksi, the thicknesses of the asphalt and base layers in inches, and the performance data were included on this sheet. Table 3.3 shows an example of data from a few different pavement sections in this dataset as they would be organized to be entered into the rutting prediction model. Labels were not included in the dataset that was entered into the model because they would not have any effect on the performance of the sections, but they were added here to provide a better review of the data. More examples of this dataset can be found in Appendix B.

Table 3.2 Design Factors in the Parametric Study

Traffic Level	Thickness (inch)		Moduli (ksi)		
	AC	Base	AC	Base	Subgrade
Low	2.5	8	200	15	7
Intermediate	7	14	500	50	20
High	10	25	1000	200	35

Table 3.3 Sample Data for Pavement ME Dataset

Section Label	Cumulative Traffic (ESALs)	Asphalt Layer Thickness	Base Layer Thickness (in)	Modulus of Asphalt Layer (ksi)	Modulus of Base Layer (ksi)	Moduli of Subgrade Layer (ksi)	Rutting (in)
Section A	63744	2.5	8	200	15	7	0.313
Section A	128974	2.5	8	200	15	7	0.341
Section A	185172	2.5	8	200	15	7	0.356
Section A	238723	2.5	8	200	15	7	0.366
Section A	291087	2.5	8	200	15	7	0.369
Section B	63744	10.0	14	500	200	20	0.058
Section B	128974	10.0	14	500	200	20	0.062
Section B	185172	10.0	14	500	200	20	0.064
Section B	238723	10.0	14	500	200	20	0.065
Section B	291087	10.0	14	500	200	20	0.066
Section C	63744	2.5	14	500	200	7	0.119
Section C	128974	2.5	14	500	200	7	0.131
Section C	185172	2.5	14	500	200	7	0.138
Section C	238723	2.5	14	500	200	7	0.143
Section C	291087	2.5	14	500	200	7	0.146
Section D	63744	7.0	8	500	15	7	0.157
Section D	128974	7.0	8	500	15	7	0.169
Section D	185172	7.0	8	500	15	7	0.173
Section D	238723	7.0	8	500	15	7	0.177
Section D	291087	7.0	8	500	15	7	0.179

Chapter 4. AI Performance Prediction Models based on Pavement Material Properties and Structure Information

4.1 Introduction

This chapter discusses how pavement materials' properties (e.g., modulus and subgrade type) and structure information (e.g., number and thickness of each layer), along with field performance data, were used to develop various AI models to predict the deterioration of performance, including rutting, IRI, and cracking. Various types of AI models were examined and developed to determine the best model type for performance prediction. After the proper type of AI model had been identified, it was then modified and applied to predict various performance measures for both theoretical and field datasets. These models were trained on a randomized subset of sections in a given dataset and then were validated using the remaining number of sections in the test set.

4.2 Investigation of Various AI Models

As discussed in Chapter 2, many types of AI models have been used in civil engineering applications. For this study, multiple model types were tested to determine the appropriate model for given datasets and the nature of the problem. The first model investigated was a neural network. Neural networks are one of the most popular forms of AI and can be applied to a wide range of problems. The next model investigated was an SVM model, in particular, a support vector regression model. This model type was selected because these models tend to work well with smaller groups of data because they have good generalization capabilities and tend to be robust to outliers. The final model type investigated was the random forests model, in particular, random forests regression. This model type was selected because random forests regression applications have not been investigated much in civil engineering applications despite the fact that they can be applied to many problems and work well with many different kinds of data. All these models were developed using Python code executed through the Google Colaboratory browser.

To determine the best model for different datasets and kinds of problems, each model type was developed on the basis of the theoretical data, and their ability to predict performance was compared with one another The data used to test these models were generated from the Pavement ME that included performance data for 266 pavement sections. Permanent deformation or rutting predicted by using various AI models was selected as the performance indicator because this parameter was determined to likely be the simplest measure to predict.

All of the models tested used as inputs material properties that included the moduli of the asphalt, base, and subgrade layers; the thicknesses of the asphalt and base layers; and the cumulative traffic for each section. The accuracy of these models was measured by using the coefficient of regression (R^2) between the predictions from the AI models and the actual data from the Pavement ME software for the testing datasets. The testing sets of the data were made from 40 percent of the dataset, and the remaining 60 percent of the dataset was used to train the models. The sections from the dataset chosen for the training and testing sets were selected randomly with Python code to reduce bias.

As shown in figures 4.1, 4.2, and 4.3, the coefficients of regression of the testing sets for the neural network, random forests, and SVM models were 0.9991, 0.9999, and 0.9162, respectively, for predicting rutting. The coefficients of regression for the training sets of these models were not graphed because they all had a coefficient of regression of 0.9999. The decrease in accuracy between the testing sets and the training sets was determined to be negligible because they all decreased by only a minor amount. They likely all had perfect coefficients of regression for the training sets because the data were theoretical and thus easily captured by an algorithm. All of the test models proved to be fairly accurate, but the random forests model performed the best of the three test models. The random forests model outperformed the neural networks model, which typically required a larger number of samples to be adaptable to the testing data. Also, the SVM model was unable to adapt to the complexity of the problem as well as the random forests model could. This is exemplified in Figure 4.4, which shows that the SVM model did not mimic the intricacies in the performance decay model based on theoretical data to as high a degree as the random forests model.

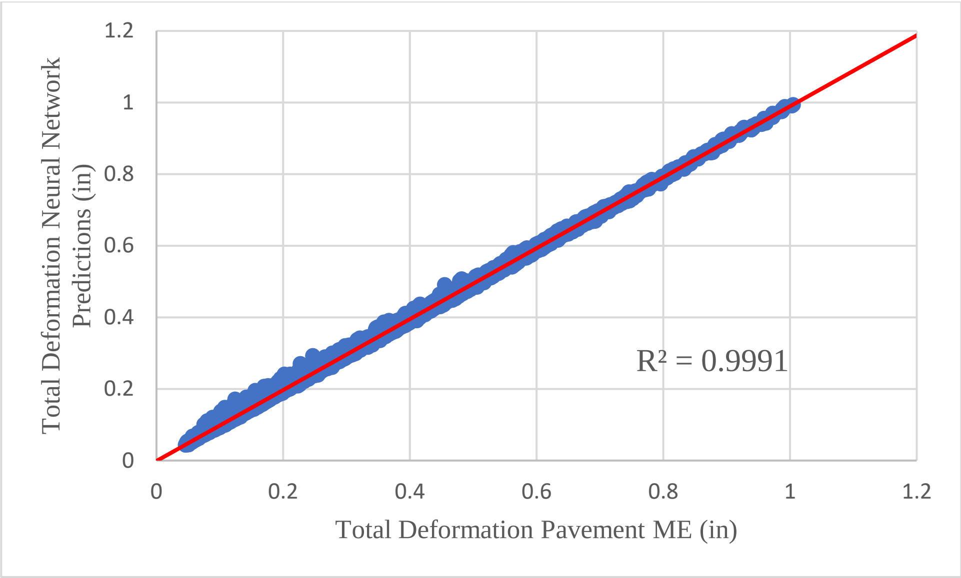

Figure 4.1 Neural Network Prediction Accuracy for Total Deformation Using Pavement ME Theoretical Data

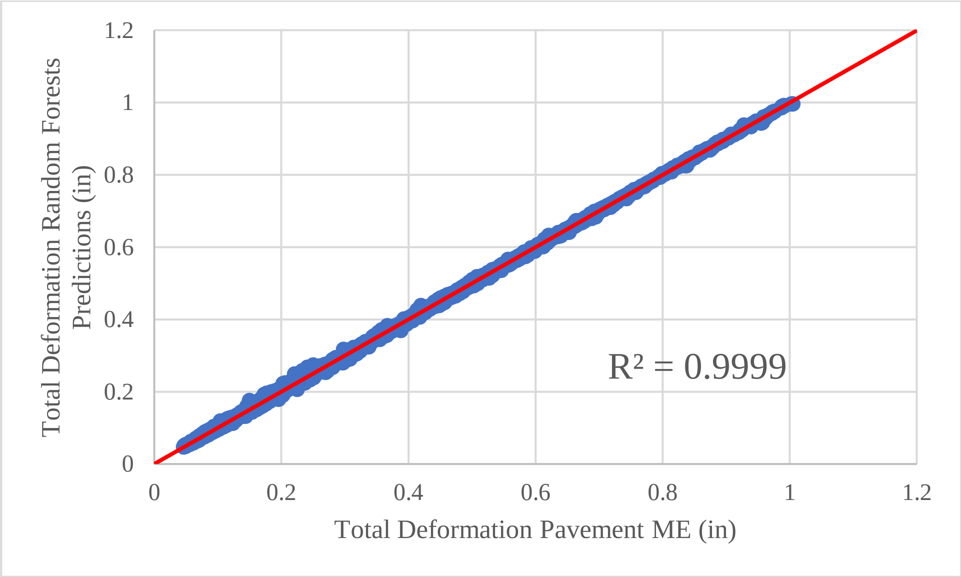

Figure 4.2 Random Forests Regression Prediction Accuracy for Total Deformation Using Pavement ME Theoretical Data

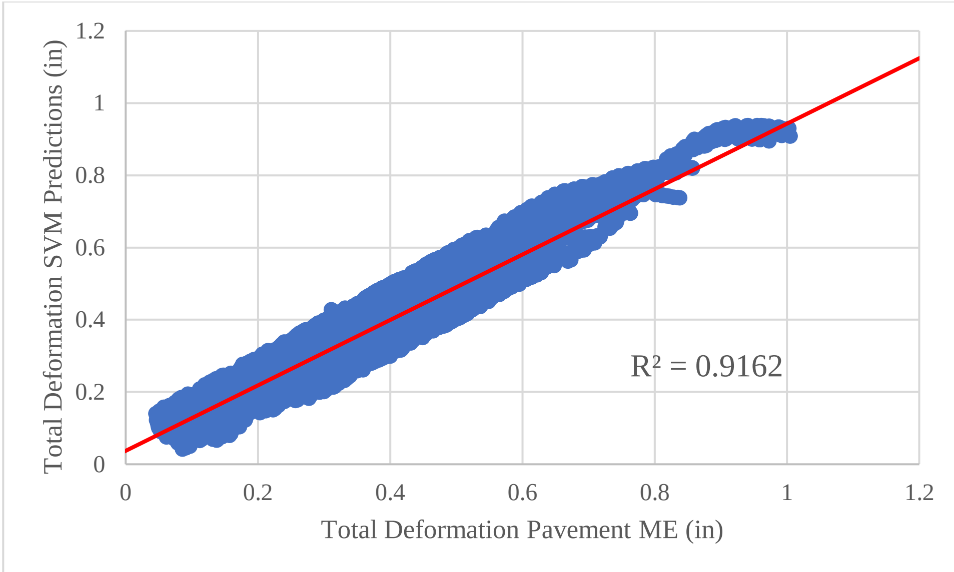

Figure 4.3 SVM Prediction Accuracy for Total Deformation Using Pavement ME Theoretical Data

Rutting increases with traffic because of the permanent deformation within various layers of the pavement structure Rutting versus the cumulative traffic and rutting performance over time (performance decay model) were examined for the test models. Figure 4.4 shows the change in rutting over time for the test models. In this particular example, the modulus values for the asphalt, base, and subgrade were 200, 15, and 7 ksi, respectively. The thicknesses of the asphalt and base layers were 2.5 inches and 8 inches, respectively.

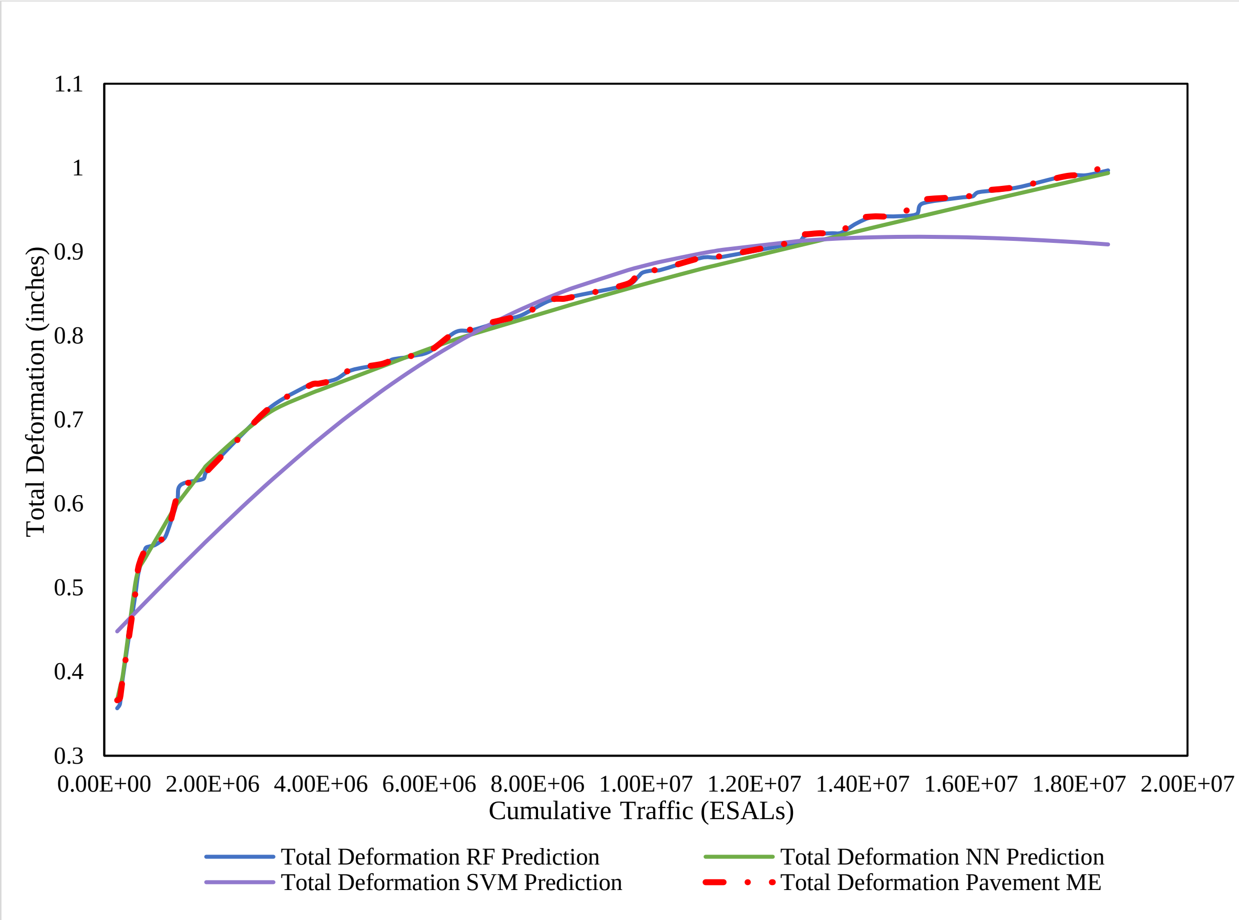

Figure 4.4 Comparison of Total Deformation Performance Decay Curves Using Various AI Methods

As shown in Figure 4.4, the random forests regression model fit the data much better than the other models. The model was able to predict and account for minor changes in rutting values, whereas the other models predicted change with smooth curves and were not able to capture minor changes.

The results of the three test models demonstrated that the random forests regression provided the highest coefficient of regression and best captured the performance decay curve. In addition, the random forests regression model was also simple and straightforward, which minimized possibilities of user error. Therefore, this model was selected for predicting the other performance parameters such as cracking and IRI.

4.3 Performance Decay Models Using the Theoretical Data

This section discusses the results of the performance decay models (i.e., rutting, cracking, and roughness) that used the theoretical data generated by the Pavement ME software. The models utilized the performance, material, and structure information for 266 test sections. The Pavement ME software predicted monthly performance over 20 years. Using this dataset, the total deformation (i.e., rutting), deformation of only the asphalt layer, IRI, bottom-up fatigue cracking, and top-down fatigue cracking were predicted by the random forests regression models. For all of these models, 60 percent of the data were used in the training set and the remaining 40 percent of the data were used in the testing set. The sections chosen for each data set were randomized in Python to reduce bias. To make predictions, they all used the thicknesses of the asphalt and base layers; the moduli of the asphalt, base, and subgrade layers; and the cumulative traffic on the sections.

4.3.1 Rutting Prediction Models

In predicting the deformation of only the asphalt layer of the pavement sections, the random forests regression model had an R^2 of 0.9999 and 0.9997 for the training and testing sets, respectively. The difference in these accuracies was negligible because it was so small. Figure 4.5 shows the correlation between the predicted rutting with the random forests regression model and rutting in the asphalt layer calculated with the Pavement ME software. The high prediction accuracy is attributed to the theoretical nature of the data, since the rutting in the asphalt layer was not measured but rather was calculated on the basis of material properties and pavement structure information.

This model was also used to examine the increase in deformation of the asphalt layer as a function of traffic. An example of this decay curve is shown in Figure 4.6. The example included in Figure 4.6 has an asphalt thickness of 2.5 inches, a base thickness of 8 inches, an asphalt layer modulus of 200 ksi, a base layer modulus of 15 ksi, and a subgrade layer modulus of 7 ksi. Figure 4.6 shows the increase in deformation or rutting of the pavement section over a 20-year span as a function of the traffic with no surface treatments over the service life (i.e., 20 years). As expected, the rutting increased with traffic. The random forests regression model closely simulated the theoretical rutting data. Such rutting performance curves could be utilized to determine the most economical times to perform treatments on a given section.

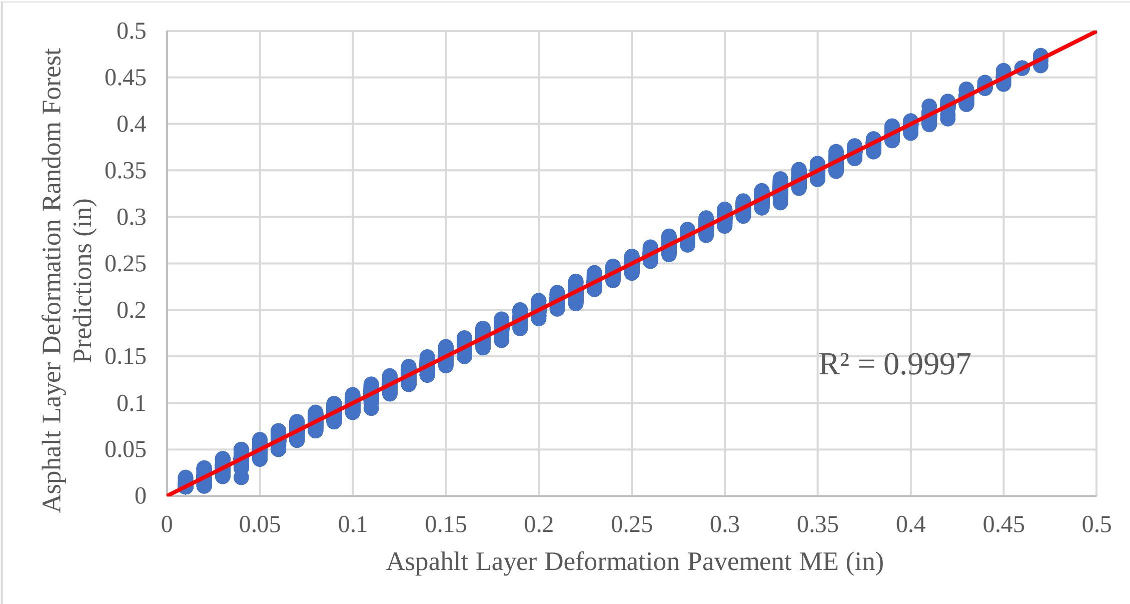

Figure 4.5 Random Forests Predictions vs Pavement ME Asphalt Layer Deformation

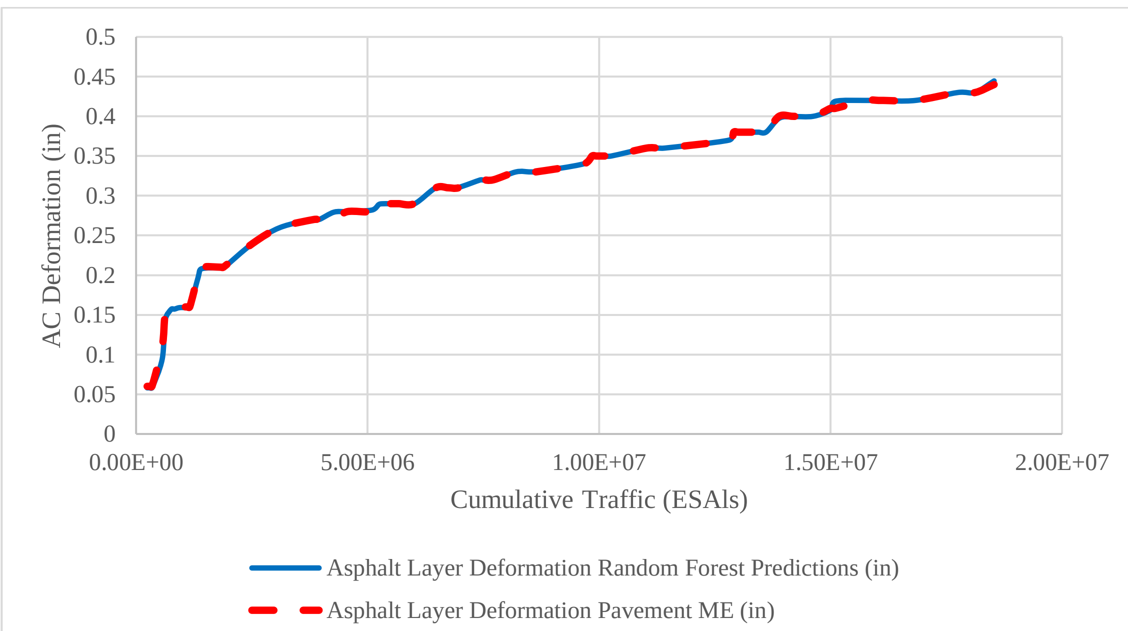

Figure 4.6 Asphalt Layer Deformation Decay Curve

Like the deformation of the asphalt layer, the total deformation or total rutting for the entire pavement section was also studied with a random forests regression model. In predicting the total deformation, the model had an R^2 of 0.9999 for the training set and an R^2 of 0.9999 for the testing set. There was no change in the accuracy between the training set and the testing set because the model was able to completely capture the relationship between total deformation and the input parameters for the entire dataset. However, this model would need to be trained with field data to fully investigate the complexities found in practice. Figure 4.7 shows the correlation between the predictions of the total deformation and the theoretical deformation for the testing group.

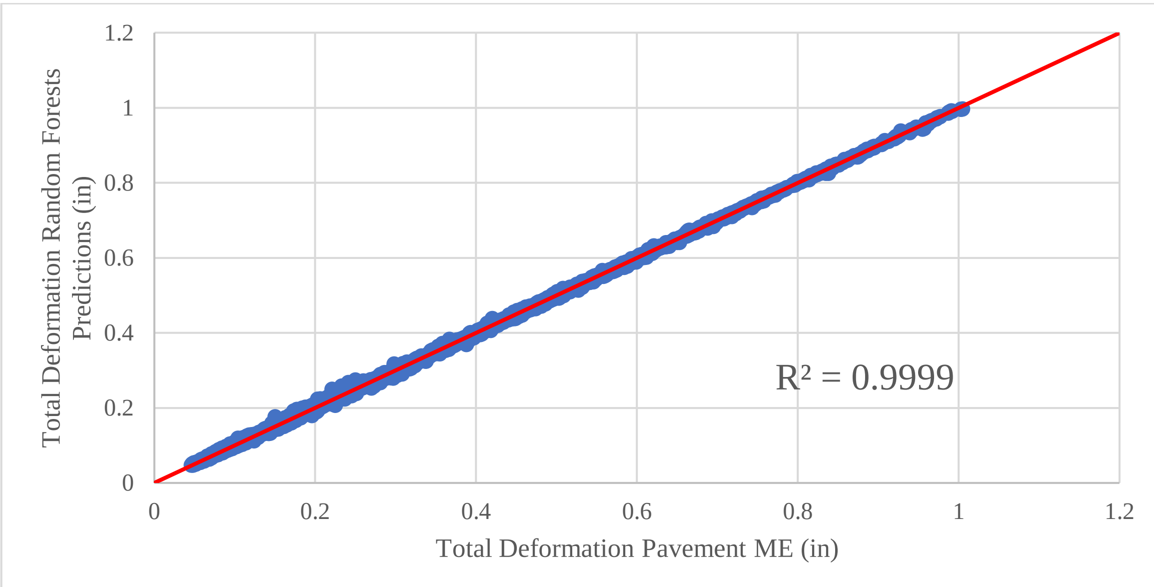

Figure 4.7 Random Forests Predictions vs Pavement ME Total Deformation

The results from this model were also used to create a decay curve of the increase in total deformation as a function of increasing traffic, as shown in Figure 4.8. The example shown in Figure 4.8 had an asphalt thickness of 2.5 inches, a base thickness of 8 inches, an asphalt layer modulus of 200 ksi, a base layer modulus of 15 ksi, and a subgrade layer modulus of 7 ksi. This figure shows the decay of the pavement section over a 20-year span as a function of the traffic on that section with no treatments applied over that period.

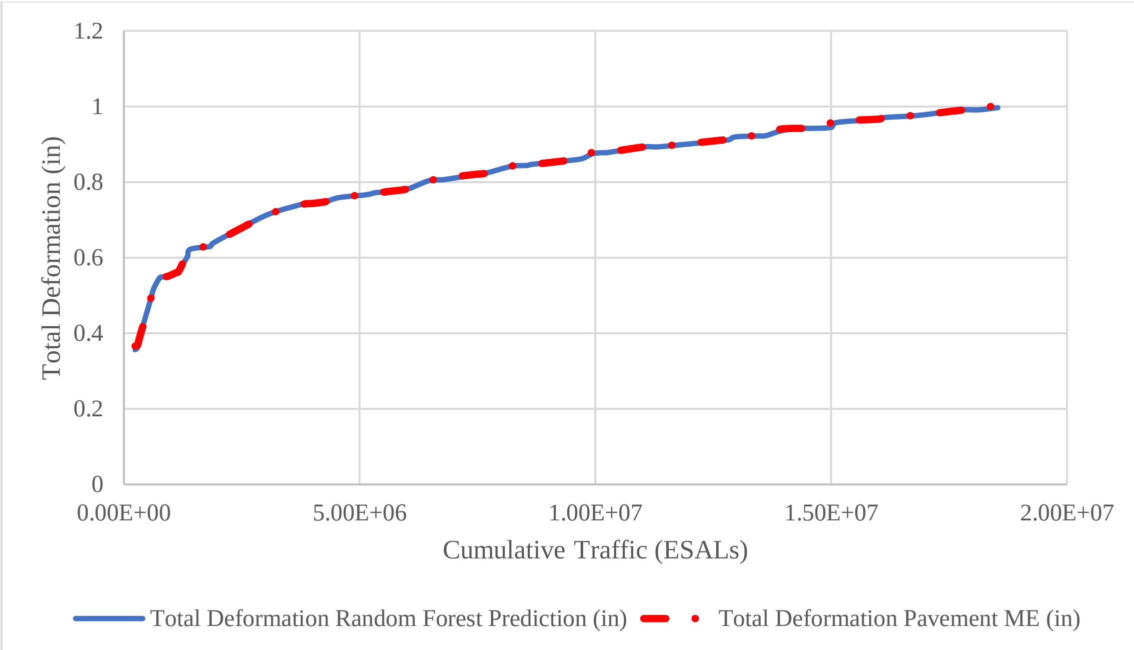

Figure 4.8 Total Deformation Decay Curve

The random forests regression models were found promising for predicting total rutting as well as rutting of the asphalt layer. These models were able to capture minor changes in rutting of the theoretical dataset generated with the Pavement ME software. This indicated the potential ability of these models to capture changes in rutting in the field. However, they should be revised to capture the effects of surface treatments. The results clearly indicated that the random forests regression models work well with these types of data and predict deformation based on the input parameters. All the random forests regression models used for the rutting predictions are summarized and presented in Appendix C.

4.3.2 IRI Prediction Results

A random forests regression was also used to predict the IRI of the test pavement sections. This model had an R^2 of 0.9999 for the training set and an R^2 of 0.9997 for the testing set. The slight decrease from the training set to the testing set was not enough to indicate any concern for the generalization capabilities of the model. Figure 4.9 shows the correlation between the random forests predictions of the IRI and the Pavement ME IRI for the testing set.

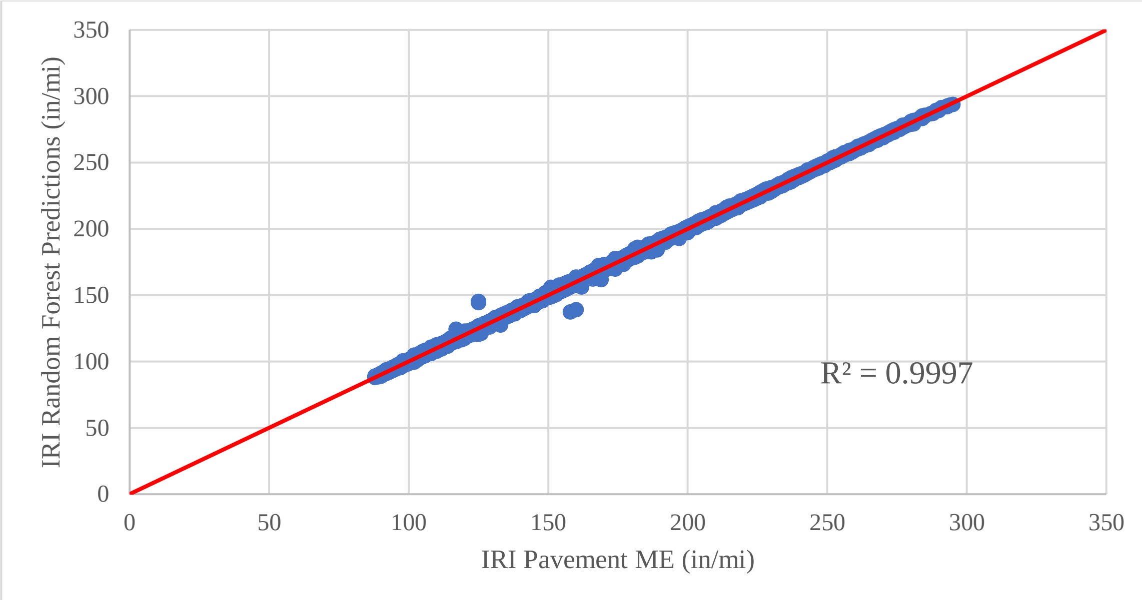

Figure 4.9 Random Forests Predictions vs. Pavement ME IRI

The random forests regression was also used to study the performance decay curve for IRI (i.e., the change in IRI) as a function of cumulative traffic on given sections. The IRI of flexible pavement increased with traffic because of the distresses induced by traffic and environmental conditions. Figure 4.10 shows an example of a performance decay curve for the IRI for a given pavement section. This section had an asphalt layer modulus of 200 ksi, a base layer modulus of 15 ksi, a subgrade layer modulus of 7 ksi, an asphalt layer thickness of 2.5 inches, and a base layer thickness of 8 inches. The IRI curve spanned over a 20-year period during which no treatments or construction events were conducted on the section.

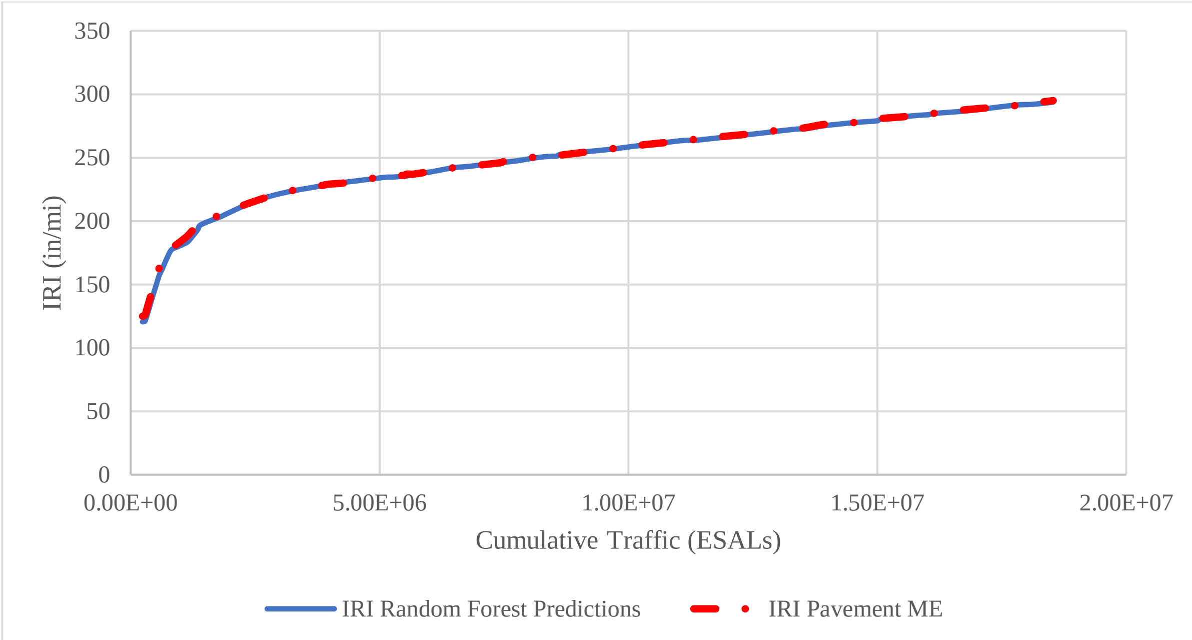

Figure 4.10 The Change in IRI with Traffic

Given the results of the IRI prediction model, the random forests technique was able to predict IRI at a high accuracy for the Pavement ME dataset. Because this dataset was theoretical, it yielded a higher accuracy than if this model had been applied to field data, as discussed later in this chapter. Field data have more variability and follow a less clear pattern than the theoretical data used in this model. However, the use of the theoretical data in this model showed that random forests regression has a comprehensive understanding of this type of data and is able to recognize patterns in using structural pavement data and cumulative traffic to predict a pavement's IRI at a given time. All random forests regression models used for the IRI predictions are summarized and presented in Appendix C.

4.3.3 Cracking Prediction Results

Random forests regression models were also developed to predict both bottom-up and top-down fatigue cracking. Cracking can be difficult to predict because it does not decay as consistently as some other performance measures do, especially with field data. For top-down fatigue cracking predictions, the model had an R^2 of 0.9996 for the training set and an R^2 of 0.9978 for the testing set. The difference between these numbers was small enough that it did not raise any issues or concerns. In Figure 4.11, the correlation between the random forests predictions and Pavement ME top-down fatigue cracking is shown for the testing set. From this model, performance decay curves for various pavement sections could be obtained. These decay curves were functions of the cumulative traffic on the given section, and they spanned over a period of 20 years. Like the rutting and IRI models, no treatments or construction events were carried out over this analysis period for the test sections. One example of the performance decay curves for top-down fatigue cracking is shown in Figure 4.12. For this example, the pavement section had an asphalt layer thickness of 2.5 inches, a base layer thickness of 8 inches, an asphalt layer modulus 200 ksi, a base layer modulus of 15 ksi, and a subgrade layer modulus of 7 ksi. One can observe that once the top-down cracking started, it propagated downward at much faster rate than IRI or rutting, which changed gradually. Also, the maximum percentage of cracking predicted by the Pavement ME was about 80 percent.

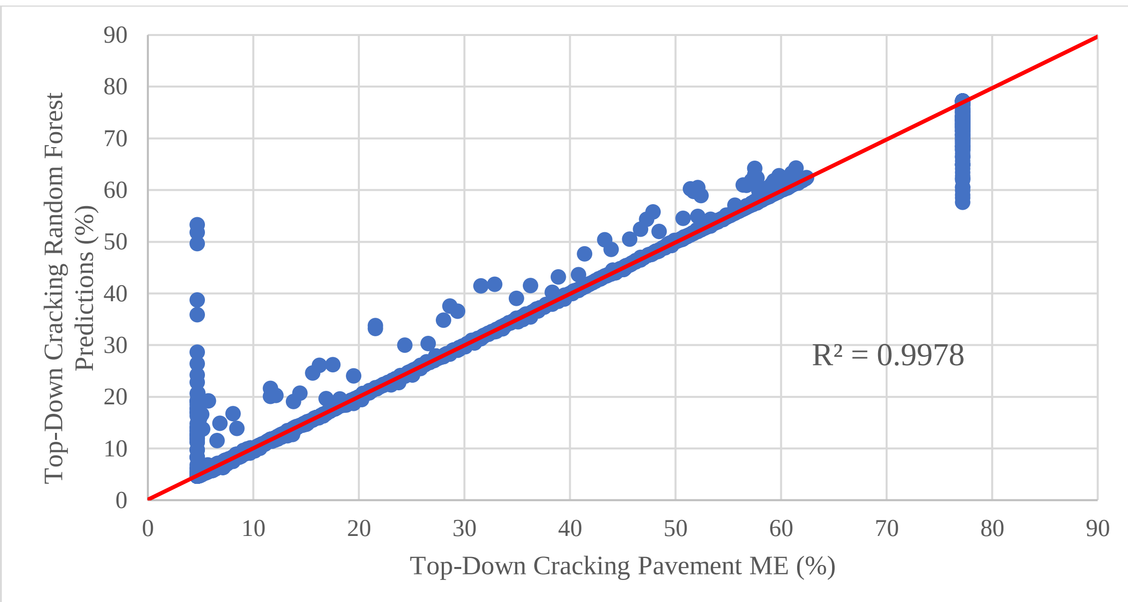

Figure 4.11 Random Forests Predictions vs. Pavement ME Top-Down Fatigue Cracking

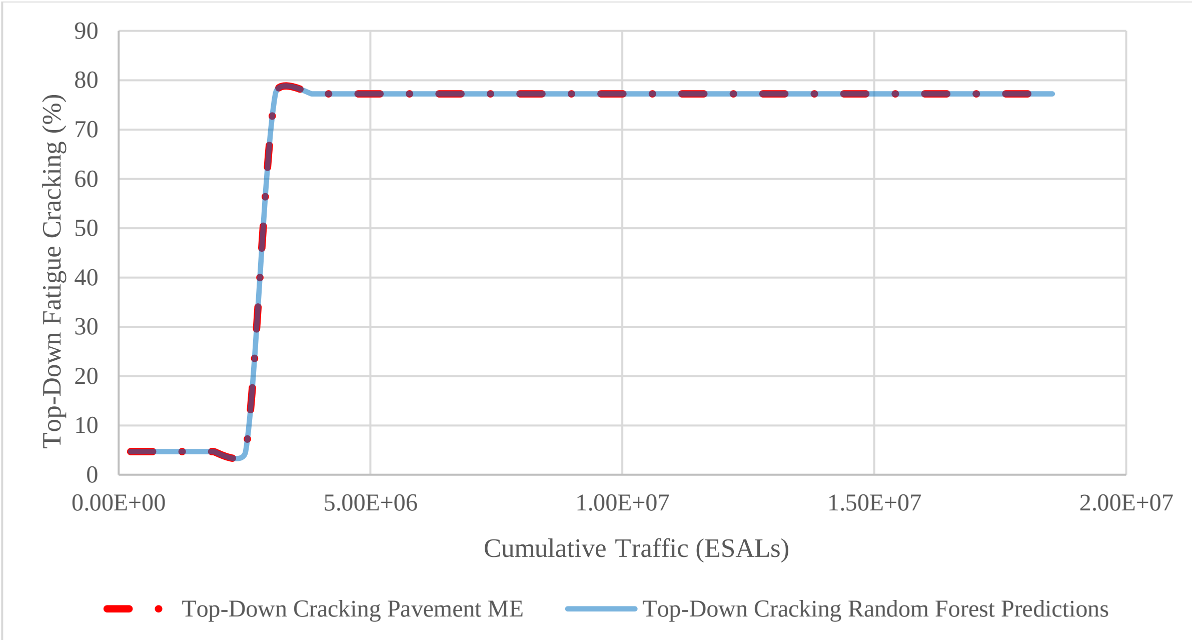

Figure 4.12 Top-Down Fatigue Cracking Decay Curve

While the R^2 of the predictions against the Pavement ME data was high, there appeared to be more variance toward the lower and higher amounts of cracking, as shown in Figure 4.11. This was likely due to the data and not caused by an issue within the AI model. In the dataset, the top-down fatigue cracking stayed stable and then suddenly increased drastically. This is shown in Figure 4.12. This rapid increase ws difficult for the model to capture, and therefore, the percentage of cracking was overestimated in some of the sections before this jump. After this rapid increase, the top-down cracking reached a maximum predicted value of about 80 percent. In this instance, the AI model underestimated some of these values. This stability led to the model not being able to identify a pattern based on changing inputs as accurately because once the cracking percentage had reached a certain threshold, it remained stable regardless of the increase in cumulative ESALs. However, the model was still able to closely predict a section's decay, as shown in Figure 4.12, so this slight variance was not of concern. The model also still accurately captured a high percentage of the dataset because these outliers represented only a small portion of the dataset.

The issue of these outliers did not appear in the model developed to predict bottom-up fatigue cracking. While there also appeared to be a rapid increase in cracking at a certain point for bottom-up fatigue cracking in this dataset, the cracking still steadily increased before it reached 100 percent. This difference in the datasets indicated that the outliers when predicting top-down cracking were a result of the dataset and not the ability of the random forests regression models to predict cracking. The model developed to predict bottom-up fatigue cracking had an R^2 of 0.9999 for the training set and an R^2 of 0.9998 for the testing set. The correlation between the random forests regression predictions and the Pavement ME bottom-up fatigue cracking for the testing set is shown in Figure 4.13.

An example of the change in bottom-up fatigue cracking or bottom-up fatigue cracking performance decay is shown in Figure 4.14. The pavement section shown had an asphalt layer thickness of 2.5 inches, a base layer thickness of 8 inches, an asphalt layer modulus of 200 ksi, a base layer modulus of 15 ksi, and a subgrade layer modulus of 7 ksi. The decay curve spanned a period of 20 years, and no construction events or treatments were conducted on the pavement section during this analysis period. This decay curve displays the model's ability to accurately predict a pavement's deterioration as a function of cumulative traffic on the section.

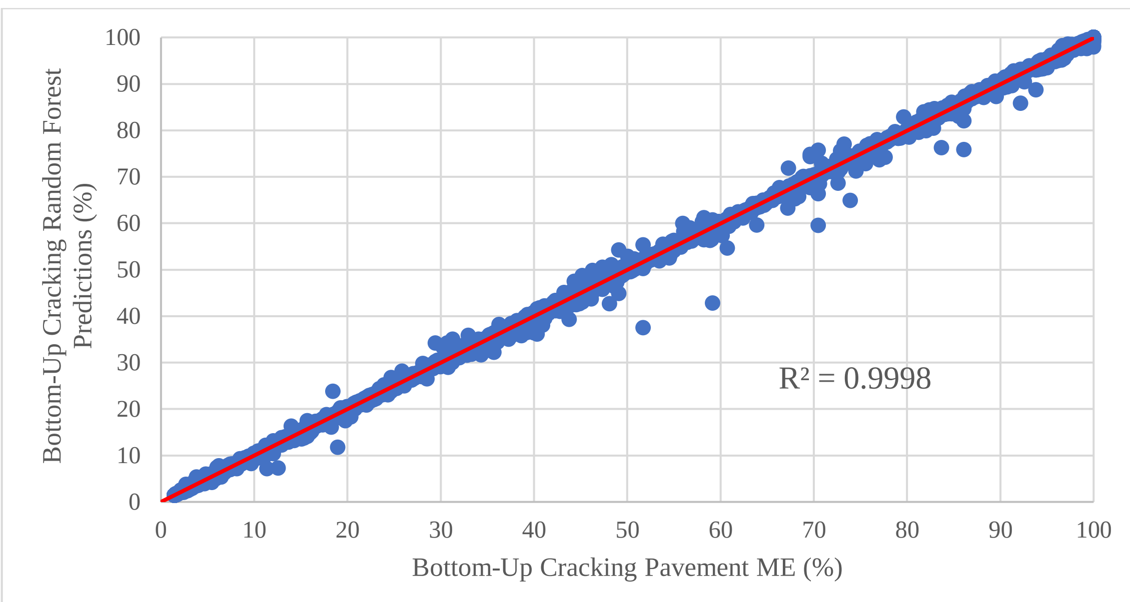

Figure 4.13 Random Forests Predictions vs. Pavement ME Bottom-Up Fatigue Cracking

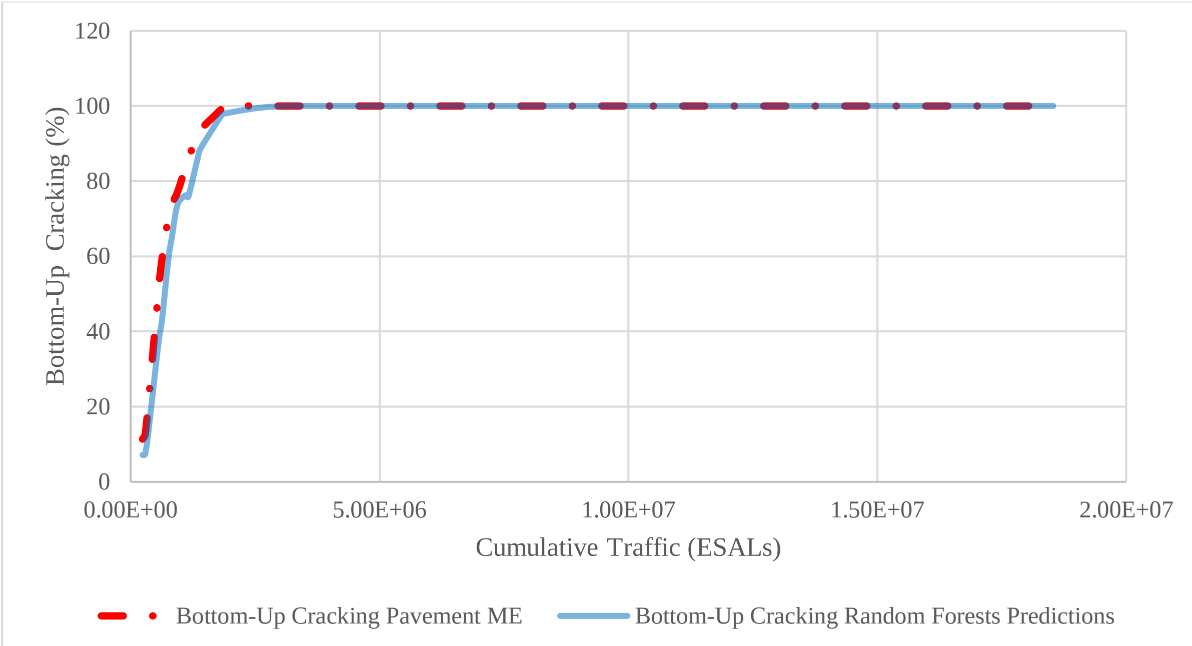

Figure 4.14 Bottom-Up Fatigue Cracking Decay Curve

The random forests regression models were able to accurately understand and predict cracking for this dataset based on these results. This theoretical dataset may not have fully captured the nuances of field data. However, both models (bottom-up cracking and top-down cracking) were able to identify the point of rapid increase in cracking for the test sections, which indicates that they had a great understanding of the cause of these rapid increases. This suggests that the model would be able to identify this point in field sections as well, and it would likely also have a high degree of accuracy when trained and tested on sufficient field data. All random forests regression models used for the cracking predictions are summarized and presented in Appendix C.

4.4 Long-Term Pavement Performance Models

This section discusses the AI models developed to predict the performance of asphalt pavements using field data. As described in the previous section, models were developed to predict rutting, IRI, and cracking. The data used were obtained from the LTPP database as discussed in Chapter 3. The LTPP data used for this study comprised 90 pavement sections from the Pacific Northwest (Idaho, Alaska, Montana, Wyoming, Washington, Oregon). Using this dataset, the total deformation, IRI, fatigue cracking, longitudinal cracking, and transverse cracking were predicted with random forests regression models. For all of these models, 60 percent of the data were used in the training set and the remaining 40 percent were used in the testing set. The dataset contained 373 usable data points for the total deformation prediction model, 581 usable data points for the IRI prediction model, 433 usable data points for the fatigue cracking prediction model, 431 usable data points for the transverse cracking prediction model, and 433 usable data points for both of the longitudinal cracking prediction models. The sections chosen for each data set were randomized in Python to reduce bias. These models used the thickness of each layer of the pavement structure, modulus of each layer, climatic conditions, and the cumulative traffic on the sections to make predictions. Figures representing the correlations between the models' predictions and the LTPP field data were generated. Note that these sections did not include frequent data points with measured performance without application of maintenance or rehabilitation treatments. Therefore, detailed performance decay curves were not available; instead, discrete data points were available and used in model predictions. These models considered major rehabilitation treatments on pavement sections by treating the sections as new after the treatment occurred. This process still considered the cumulative traffic on the sections before any major rehabilitation treatment, but it "reset" the pavements' age to zero years after the treatment occurred. This was done in an attempt to capture the effect that these major rehabilitation treatments may have had on performance.

4.4.1 Total Deformation Prediction Results

The random forests regression model developed to predict rutting of the LTPP sections had an R^2 of 0.9578 for the training set and an R^2 of 0.8123 for the testing set. The high R^2 for the training set indicated that the model had a good understanding of the training set, and the decrease in R^2 indicated that the data the model was trained on did not fully represent the data the model was tested on. The R^2 of the testing set could have been improved by training the model on a larger number of samples. This could have been done by changing the ratio of the train-test split; however, given the relatively small number of test sections in this dataset, changing this ratio did not make a significant difference. It is also important to have a large enough test set to evaluate the accuracy of the model. The model could have been improved by including a larger number of overall sections in the dataset and giving the model more examples of sections to learn from. However, the accuracy with the test set was still relatively high, and the decrease in accuracy from the training set was not large enough to raise concerns about the abilities of the model. Figure 4.15 shows the correlation between the random forests regression model and the LTPP dataset for total rutting or deformation for the testing set.

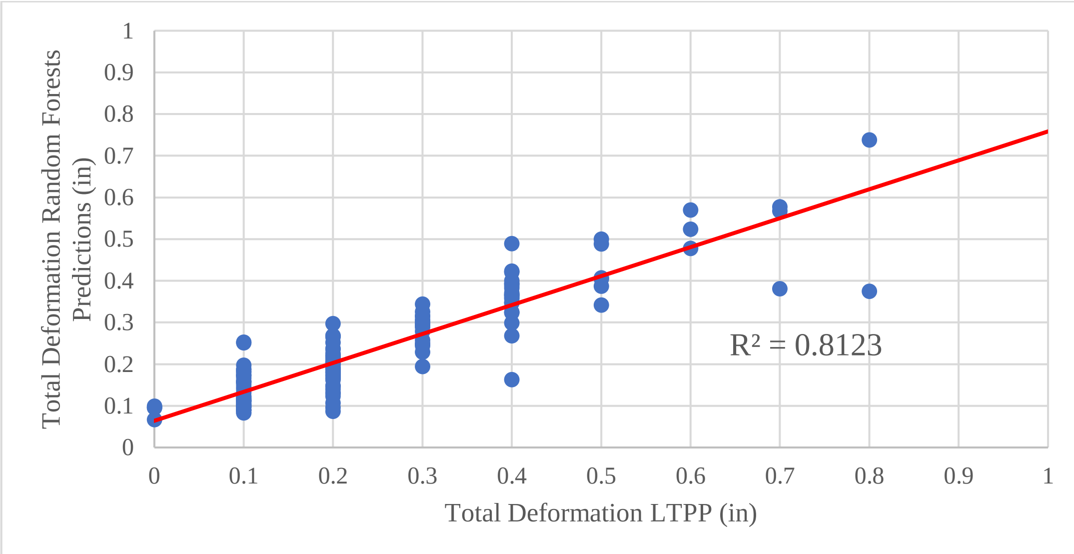

Figure 4.15 Random Forests Predictions vs. LTPP Total Deformation

In Figure 4.15, some rutting data are grouped at the vertical axis on every tenth of an inch. The reason is that the LTPP field rutting data were reported to the nearest tenth of an inch but the prediction model didn't round the predicted values. This means that on the vertical axis representing the LTPP data, there is nothing measured between each tenth of an inch. While not all of the predictions perfectly represented the actual measures found in the LTPP data, the majority of predictions fell within one tenth of an inch of the true value. This indicated that the incorrect predictions were still somewhat understood by the model. The incorrect predictions included both overestimations and underestimations, so there was not a clear trend for these misses. Meanwhile, the results demonstrated that the model was still able to predict total deformation with a higher R^2. This prediction accuracy could have been improved by increasing the number of samples in the dataset. All random forests regression models used for the rutting predictions are summarized and presented in Appendix C.

4.4.2 IRI Prediction Results

The random forests regression model developed to predict IRI had an R^2 of 0.9498 for the training set and an R^2 of 0.6844 for the testing set. This was a significant decrease from the training to the testing set and may have indicated that the model was overfit. Overfitting is when the model is trained on a small number of samples that is not representative of all possibilities. Because the dataset was randomly split into the training and testing groups, it was unlikely that the samples were entirely unrepresentative of the whole. However, the number of samples was limited and therefore could not fully encompass all the possibilities. The high R^2 for the training set indicated that the model had the ability to understand and predict IRI for field data, but it needed more samples to fully master predicting this measure. Training models on field data differs from training models on theoretical data because of the complexities within field data. These complexities (such as variability/inconsistencies during data collection) in field data make them less predictable. Figure 4.16 shows the correlation between the model predictions and the LTPP dataset for IRI in the testing set.

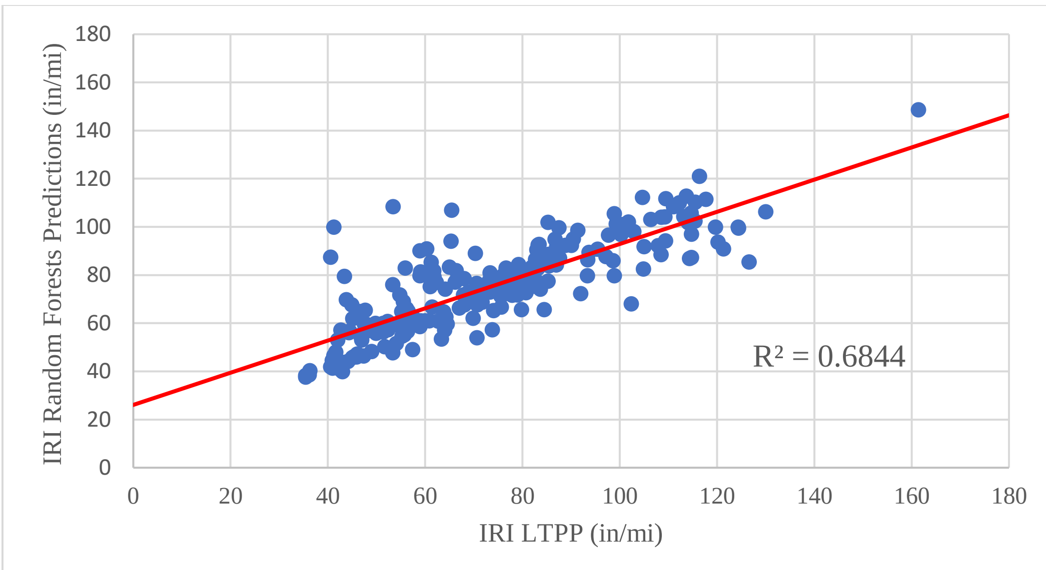

Figure 4.16 Random Forests Predictions vs. LTPP IRI

It is also important to consider that predicting IRI to an exact value is difficult. The model could have been improved by grouping IRI into separate categories (such as excellent, good, moderate, poor, extremely poor) and then having the model predict which category the road section would fall into. The models could have been further improved by incorporating all maintenance activities. Our models only considered major rehabilitation treatments. By factoring minor treatments into the model, the model would have been able to better capture the changes in IRI caused by these treatments. The models could also have been improved by adding more sections to the dataset. All random forests regression models used for the IRI predictions are summarized and presented in Appendix C.

4.4.3 Cracking Prediction Results

Random forests regression models were developed to predict fatigue cracking, transverse cracking, wheel path longitudinal cracking, and non-wheel path longitudinal cracking with the LTPP dataset. The model used to predict fatigue cracking had an R^2 of 0.9021 for the training set and an R^2 of 0.5815 for the testing set. This significant decrease from the training set to the testing set was likely caused by issues similar to those of the IRI model. Figure 4.17 shows the correlation between the model predictions and the LTPP data for the fatigue cracking testing set. The dataset was weighted heavily by 0 ft^2 of fatigue cracking, which likely affected the accuracy of the results. Finding more sections that contained fatigue cracking would have improved the accuracy. The models could also have been further improved by incorporating all maintenance activities. Our models only considered major rehabilitation treatments. By factoring minor treatments into the model, it would have been able to better capture the changes in cracking caused by these treatments. The random forests regression models did demonstrate the ability to predict these measures more accurately based on the high accuracy scored with the training set of these data.

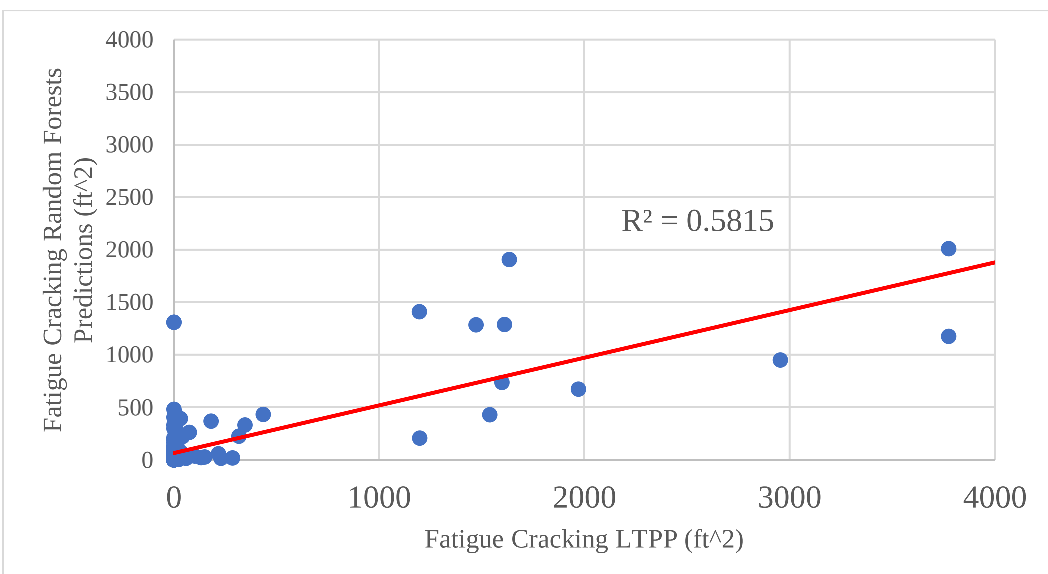

Figure 4.17 Random Forests Predictions vs. LTPP Fatigue Cracking

The model developed to predict transverse cracking had an R^2 of 0.9525 for the training set and an R^2 of 0.6859 for the testing set. This decrease in accuracy from the training to the testing set was likely due to the number of samples in the dataset, as in the other models that used this dataset. However, the decrease for transverse cracking was not as large as the decrease for fatigue cracking. This was likely because there was a better distribution of values in the transverse cracking measurements. However, the values were still largely distributed on the lower end of possibilities. This was likely due to the fact that field sections often receive treatments when conditions become too poor, so there was not a large representation of pavements in extremely poor condition. This was likely true for all the performance measures in the LTPP dataset because they would all be affected by construction on the sections. The correlation between the model predictions and the LTPP data in the testing set for transverse cracking is shown in Figure 4.18.

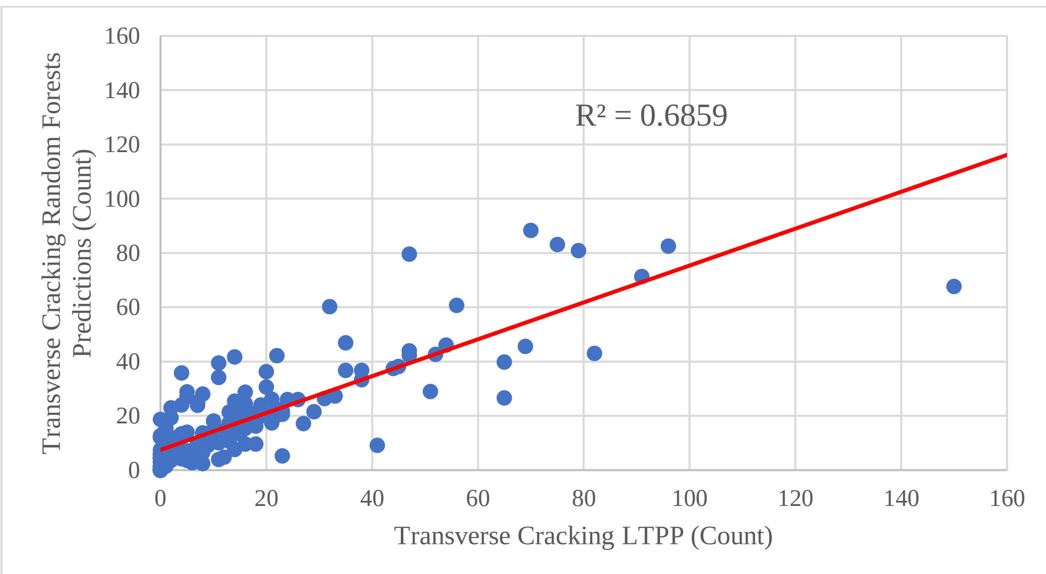

Figure 4.18 Random Forests Predictions vs. LTPP Transverse Cracking

Random forests regression models were also developed to predict both wheel path and non-wheel path longitudinal cracking. For non-wheel path longitudinal cracking, the model had an R^2 of 0.9483 for the training set and an R^2 of 0.5402 for the testing set. In predicting wheel path longitudinal cracking, the model had an R^2 of 0.8970 for the training set and an R^2 of 0.4857 for the testing set. The decreases in accuracy from the training sets to the testing sets for both were most likely due to the small number of samples in the dataset. However, the high accuracies for both of the training sets indicated that the models were able to understand the data enough to predict longitudinal cracking accurately but needed a larger representation of samples to produce higher accuracy for the testing set as well. Figure 4.19 shows the correlation between the model predictions and the LTPP data in the testing set for non-wheel path longitudinal cracking, and Figure 4.20 shows the same for wheel path longitudinal cracking.

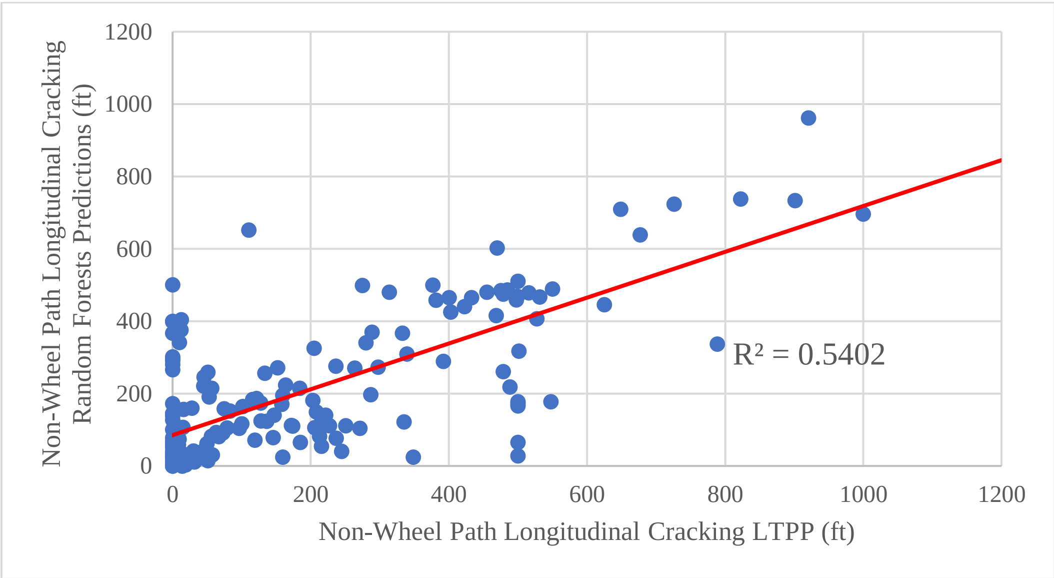

Figure 4.19 Random Forests Predictions vs. LTPP Non-Wheel Path Longitudinal Cracking

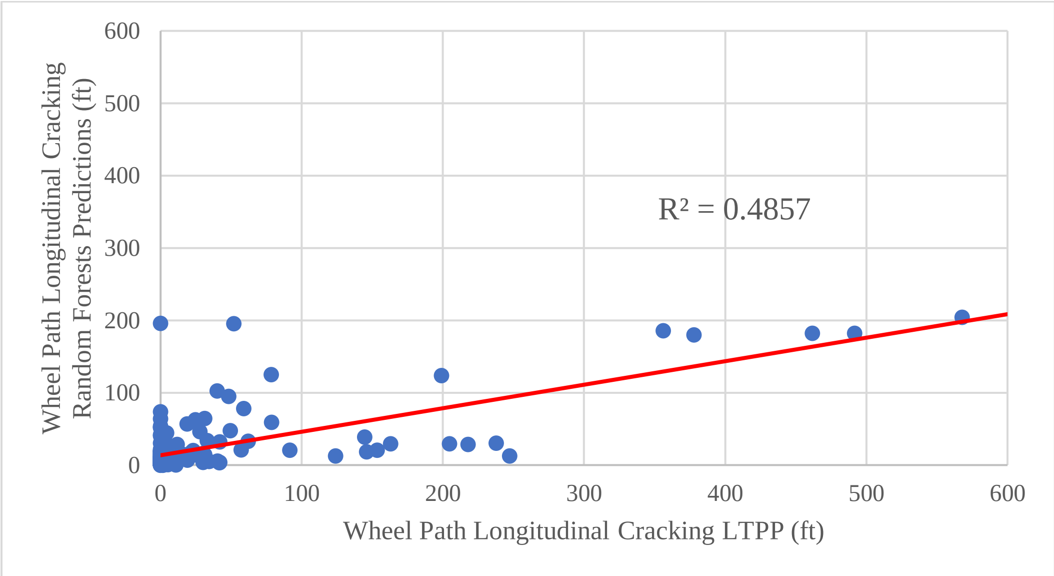

Figure 4.20 Random Forests Predictions vs. LTPP Data Wheel Path Longitudinal Cracking

The results obtained from all the models using the LTPP data showed potential. The coefficients of regression for the training sets for all performance measures indicated that the models can predict these performance measures with high accuracy. However, all performance measures experienced a decrease in accuracy for the testing sets. This demonstrated that the models needed to be trained on more samples to obtain a larger representation of all potential data so that the models could more accurately predict field performance.

The smallest decrease in accuracy from the training set to the testing set among all the LTPP trained models was with the model developed to predict total rutting. This was likely because rutting is the issue least affected by minor maintenance activities such as crack sealing or seal coats. It may also have been simply because the range of possible values in predicting total deformation was the smallest. The largest decrease in accuracy from the training set to the testing set among all the LTPP trained models was from the models predicting cracking. This was expected because cracking is a more difficult relationship to capture because of its complexity. The dataset also had the largest range of values in measuring cracking, which perhaps also could have contributed to this large decrease in accuracy. All random forests regression models used for the cracking predictions are summarized and presented in Appendix C.

Chapter 5 Conclusions and Recommendations

This study aimed to develop artificial intelligence models to predict pavement performance indicators using material properties, layer thickness, and traffic data. Several artificial intelligence models were investigated to determine the best method to use with the data. Through this investigation, random forests regression models demonstrated the highest correlation between predicted and theoretical performance measures, and the predictions produced the most accurate performance decay curves of all the tested models.

Next, random forests regression models were developed with a theoretical dataset generated by the Pavement ME software to predict rutting, IRI, and cracking. These models produced high correlations between the predicted and theoretical values for all performance indicators. The predicted performance indicators from these models were also used to create performance decay curves. Similar models were developed using a field dataset obtained from the LTPP program. Those models were also developed to predict rutting, IRI, and cracking. The models also produced good correlations between some of the measured and predicted performance indicators.

The main findings of this research study can be summarized as follows:

- Models developed with the theoretical dataset were able to achieve strong correlations ($R^2 = 0.99$) between all predicted and measured performance indicators (IRI, rutting, cracking).
- Predictions made by models developed with the theoretical dataset were used to develop performance decay curves. The predicted performance decay curves mimicked the measured decay curves for all investigated performance indicators.
- Models developed with the field dataset were able to achieve strong correlations between the predicted and measured performance indicators for some of the indicators. The prediction models had R^2 values of 0.81 for IRI, 0.68 for total deformation (rutting), 0.58 for fatigue cracking, 0.69 for transverse cracking, 0.54 for non-wheel path longitudinal cracking, and 0.48 for wheel path longitudinal cracking.
- Models developed with the field dataset did not perform as well as the models developed with the theoretical dataset. This was most likely due to the theoretical dataset being larger than the field dataset and the exclusion of maintenance activities in the theoretical models. Models developed with larger datasets gain greater

adaptability toward testing sections and therefore can obtain higher correlations with measured data. The theoretical dataset was also designed so that there were no maintenance activities on the sections throughout their service lives, whereas the field sections had more variability and included many maintenance activities during their service lives. Major maintenance activities were accounted for in the models, but minor treatments on the sections could not be included in the models. Therefore, their effect on pavement condition was not captured, which decreased the accuracy of the models based on field data.

The recommendations for future research can be summarized as follows:

- Additional field pavement sections can be added to the dataset to increase accuracy and further validate the models.
- Development of method(s) to accurately represent construction events on pavement sections to incorporate minor and major treatments into the models will improve their applicability.
- Investigation of the relationships between different variables and performance could indicate the best parameters to include in the dataset to achieve the highest model accuracy.
- Development of AI models based on larger quantities of test sections can improve their accuracy because their adaptability is dependent on the number of sections on which they are trained.

References

Arshadi, A., and Kachwalla, A. (2021). *Development of a Machine Learning-Based Quality Control Approach for Automated Pavement Condition Data*. Reading.

Artificial Intelligence (AI) vs. Machine Learning, 2022

Bayomy, F., A. Muftah, and E. Kassem. *Calibration of the AASHTOWare Pavement ME Design Performance Models for Flexible Pavements in Idaho*. Publication FHWA-ID-18-235, ITD Report RP 235, Idaho Transportation Department, 2018.

Bianchini, A., and Bandini, P. (2010). Prediction of pavement performance through neuro-Fuzzy Reasoning. *Computer-Aided Civil and Infrastructure Engineering*, *25*(1), 39–54. https://doi.org/10.1111/j.1467-8667.2009.00615.x

Bishop, C. M. (1994). Neural networks and their applications. *Review of Scientific Instruments*, *65*(6), 1803–1832. https://doi.org/10.1063/1.1144830

Chen, X., Dong, Q., Zhu, H., and Huang, B. (2016). Development of distress condition index of asphalt pavements using LTPP data through structural equation modeling. *Transportation Research Part C: Emerging Technologies*, *68*, 58–69. https://doi.org/10.1016/j.trc.2016.03.011

Columbia Engineering. (2022, March 3). *Artificial Intelligence (AI) vs. Machine Learning*. CU-CAI. Retrieved November 8, 2022, from https://ai.engineering.columbia.edu/ai-vs-machine-learning/

Deng, Y., Luo, X., Zhang, Y., Cai, S., Huang, K., Shi, X., and Lytton, R. L. (2020, November 22). *Determination of flexible pavement deterioration conditions using long-term Pavement performance database and artificial intelligence-based finite element model updating*. Wiley Online Library. Retrieved October 12, 2021, from https://onlinelibrary.wiley.com/doi/full/10.1002/stc.2671

Dick, S. (2019). Artificial Intelligence. *Issue 1*. https://doi.org/10.1162/99608f92.92fe150c

Dimeter, S., Rukavina, T., Domitrovic, J., and Dragovan, J. H. (2018). Application of an artificial neural network in Pavement Management System. *Tehnicki Vjesnik - Technical Gazette*, *25*(Supplement 2). https://doi.org/10.17559/tv-20150608121810

Ebid, A. M. (2020). 35 years of (AI) in Geotechnical Engineering: State of the art. *Geotechnical and Geological Engineering*, *39*(2), 637–690. https://doi.org/10.1007/s10706-020-01536-7

Elastic Modulus. Pavement Interactive. (2006). Retrieved November 9, 2022, from https://pavementinteractive.org/reference-desk/design/design-parameters/elastic-modulus/

Elhadidy, A. A., Elbeltagi, E. E., and Ammar, M. A. (2015). Optimum analysis of pavement maintenance using multi-objective genetic algorithms. *HBRC Journal*, *11*(1), 107–113. https://doi.org/10.1016/j.hbrcj.2014.02.008

Federal Highway Administration. (n.d.). *LTPP Program Brochure*. (2009). McLean, VA; Federal Highway Administration.

Gong, H., Sun, Y., Shu, X., and Huang, B. (2018). Use of random forests regression for predicting IRI of asphalt pavements. *Construction and Building Materials*, *189*, 890–897. https://doi.org/10.1016/j.conbuildmat.2018.09.017

Guo, R., Fu, D., and Sollazzo, G. (2021). An ensemble learning model for asphalt pavement performance prediction based on Gradient Boosting Decision tree. *International Journal of Pavement Engineering*, 1–14. https://doi.org/10.1080/10298436.2021.1910825

Hasan Ziari, M. M. (2009). *Prediction of pavement performance: Application of support vector regression with different kernels - Hasan Ziari, Mojtaba Maghrebi, Jalal Ayoubinejad, S. Travis Waller, 2016*. SAGE Journals. Retrieved November 9, 2021, from https://journals.sagepub.com/doi/10.3141/2589-15

Hossain, M. I., Gopisetti, L. S., and Miah, M. S. (2017). Prediction of international roughness index of flexible pavements from climate and traffic data using artificial neural network modeling. *Airfield and Highway Pavements 2017*. https://doi.org/10.1061/9780784480922.023

Hosseini, S. A. (2020). Data-driven framework for modeling deterioration of pavements in the state of Iowa. https://doi.org/10.31274/etd-20200624-235

Hosseini, S. A., Alhasan, A., and Smadi, O. (2020). Use of deep learning to study modelling deterioration of pavements a case study in Iowa. https://doi.org/10.31224/osf.io/edhvy

Ismail, N., Ismail, A., and Atiq, R. (2009). An Overview of Expert Systems in Pavement Management. *European Journal of Scientific Research*, *30*(1), 99–111.

Issa, A., Samaneh, H., and Ghanim, M. (2021, May 14). *Predicting pavement condition index using artificial neural networks approach*. Ain Shams Engineering Journal. Retrieved September 29, 2021, from https://www.sciencedirect.com/science/article/pii/S2090447921002264

Justo-Silva, R., Ferreira, A., and Flintsch, G. (2021). Review on machine learning techniques for developing pavement performance prediction models. *Sustainability*, *13*(9), 5248. https://doi.org/10.3390/su13095248

Kassem, E., and Sufian, A. A. (2022). Simplified Analysis Technique of Flexible Pavements using TSD and FWD Data at the Network Level. Moscow; University of Idaho.

Kaur, P. D., and Pulugurta, H. (2008). *Comparitive Analysis of Fuzzy Decision Tree and Logistic Regression Methods for Pavement Treatment Prediction*. eng.utoledo.edu. Retrieved October 12, 2021, from http://www.eng.utoledo.edu/~dkaur/publications.html

Kumar, R., Suman, S. K., and Prakash, G. (2021). Evaluation of pavement condition index using artificial neural network approach. *Transportation in Developing Economies*, *7*(2). https://doi.org/10.1007/s40890-021-00130-7

Lin, J.-D., Yau, J.-T., and Hsiao, L.-H. (2003). *Correlation Analysis Between international roughness index* ... ReserachGate. Retrieved December 18, 2021, from https://www.researchgate.net/profile/Jyh-Dong-Lin/publication/228848218_Correlation_analysis_between_international_roughness_index_IRI_and_pavement_distress_by_neural_network/links/02e7e52f385af7c205000000/Correlation-analysis-between-international-roughness-index-IRI-and-pavement-distress-by-neural-network.pdf

Liu and Scullion (2001) MODULUS 6.0 FOR WINDOWS: USER'S MANUAL, Report No. FHWA/TX-02/1869-3 , Texas Transportation Institute, https://pavementdesign.tamu.edu/manuals/Modulus6.pdf

Lou, Z., Gunaratne, M., Lu, J. J., and Dietrich, B. (2001). Application of neural network model to forecast short-term Pavement crack condition: Florida case study. *Journal of Infrastructure Systems*, *7*(4), 166–171. https://doi.org/10.1061/(asce)1076-0342(2001)7:4(166)

Lu, P., Chen, S., and Zheng, Y. (2012). Artificial intelligence in civil engineering. *Mathematical Problems in Engineering*, *2012*, 1–22. https://doi.org/10.1155/2012/145974

ME Design Guide (2008) Mechanistic-empirical pavement design guide, a manual of practice, July 2008.

Pavement Distresses. Pavement Interactive. (2006). Retrieved November 9, 2022, from https://pavementinteractive.org/reference-desk/pavement-management/pavement-distresses/

Piryonesi, M. S., and El-Diraby, T. E. (2021). Using machine learning to examine impact of type of performance indicator on flexible pavement deterioration modeling. *Journal of Infrastructure Systems*, *27*(2), 04021005. https://doi.org/10.1061/(asce)is.1943-555x.0000602

Piryonesi, S. M., and El-Diraby, T. (2021, September 21). *Climate change impact on infrastructure: A machine learning solution for predicting pavement condition index*. Construction and Building Materials. Retrieved December 6, 2021, from https://www.sciencedirect.com/science/article/pii/S0950061821026568

Rajagopal, A. (2006, June). *Developing pavement performance prediction models and decision trees for the city of Cincinnati*. PDF Free Download. Retrieved November 19, 2021, from https://businessdocbox.com/Green_Solutions/78215817-Developing-pavement-performance-prediction-models-and-decision-trees-for-the-city-of-cincinnati.html

Rokach, L., and Maimon, O. (2005). Decision trees. *Data Mining and Knowledge Discovery Handbook*, 165–192. https://doi.org/10.1007/0-387-25465-x_9

Salini, R., Xu, B., and Lenngren, C. A. (2015). Application of Artificial Intelligence for Optimization in Pavement Management. *International Journal of Engineering and Technology Innovation*, *5*(3), 189–197.

Shaheen, A. A., Fayek, A. R., and AbouRizk, S. M. (2009). Methodology for integrating fuzzy expert systems and discrete event simulation in Construction Engineering. *Canadian Journal of Civil Engineering*, *36*(9), 1478–1490. https://doi.org/10.1139/l09-091

Shahin, M. A., Jaksa, M. B., and Maier, H. R. (2009, November 15). *Recent advances and future challenges for Artificial Neural Systems in geotechnical engineering applications*. Advances in Artificial Neural Systems. Retrieved October 9, 2021, from https://www.hindawi.com/journals/aans/2009/308239/

Shahnazari, H., Tutunchian, M. A., Mashayekhi, M., and Amini, A. A. (2012). Application of soft computing for prediction of pavement condition index. *Journal of Transportation Engineering*, *138*(12), 1495–1506. https://doi.org/10.1061/(asce)te.1943-5436.0000454

Sollazzo, G., Fwa, T. F., and Bosurgi, G. (2017). An ann model to correlate roughness and structural performance in asphalt pavements. *Construction and Building Materials*, *134*, 684–693. https://doi.org/10.1016/j.conbuildmat.2016.12.186

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (1970, January 1). *Dropout: A simple way to prevent neural networks from overfitting*. Journal of Machine Learning Research. Retrieved November 3, 2021, from https://jmlr.org/papers/v15/srivastava14a.html

Terzi, S. (2007). Modeling the pavement serviceability ratio of flexible highway pavements by artificial neural networks. *Construction and Building Materials*, *21*(3), 590–593. https://doi.org/10.1016/j.conbuildmat.2005.11.001

Tohidi, M., Khayat, N., and Telvari, A. (2022). The use of intelligent search algorithms in the cost optimization of road pavement thickness design. *Ain Shams Engineering Journal*, *13*(3), 101596. https://doi.org/10.1016/j.asej.2021.09.023

UNR Releases 3D-Move Analysis Software. *Asphalt Research Consortium (ARC) Newsletter*. University of Nevada, Reno, 2010. http://www.arc.unr.edu/Newsletter/Aug10/UNRReleases3DMoveAnalysis.pdf. Accessed July 29, 2021.

Zhang, C., Nateghinia, E., Miranda-Moreno, L. F., and Sun, L. (2021). Pavement distress detection using Convolutional Neural Network (CNN): A case study in Montreal, Canada. *International Journal of Transportation Science and Technology*. https://doi.org/10.1016/j.ijtst.2021.04.008

Ziari, H., Sobhani, J., Ayoubinejad, J., and Hartmann, T. (2015). Prediction of IRI in short and long terms for flexible pavements: Ann and gmdh methods. *International Journal of Pavement Engineering*, *17*(9), 776–788. https://doi.org/10.1080/10298436.2015.1019498

Appendix A Long-Term Pavement Performance Data

This section contains 30 sample data points for the LTPP dataset formatted as they were when directly imported into the models.

Table A.1 Extended Sample Data LTPP Dataset

Section_Age_Years	**Subgrade_Layer Type**	**Asphalt_Total_Thickness**	**KESAL_CUMULATIVE**	**Asphalt_Majority_Material**	**Base1_Type**	**Base_Total_Thickness**	**Climatic_Zone**	**Field IRI**
10	145	8.4	3555	1	321	36.6	2	79.77
9	267	9.4	2389	1	308	18.7	2	114.36
10	267	7.5	2832	1	308	18.1	2	46.76
23	282	7.5	958	1	304	19.4	1	82.75
6	215	7.6	395	1	304	6.8	1	60.83
5	141	4.5	250	1	304	20.5	1	42.58
25	214	10.3	2577	1	304	5.4	2	85.35
5	214	11.5	3008	1	304	5.4	2	76.22
19	267	9.2	887.5	1	304	3.7	3	61.02
13	267	2.8	714	1	304	23.1	2	68.56
18	282	7.5	726	1	304	19.4	1	68.87
10	267	9.9	2832	1	308	19.1	2	68.05
4	215	7.6	294	1	304	6.8	1	45.11
22	282	7.5	902	1	304	19.4	1	83.64
15	267	9.8	5224	1	308	18	2	58.92
30	266	6	1462	1	304	19.2	2	82.94

Section_Age_Years	Subgrade_Layer Type	Asphalt_Total_Thickness	KESAL_CUMULATIVE	Asphalt_Majority_Material	Base1_Type	Base_Total_Thickness	Climatic_Zone	Field IRI
16	255	4.3	1599	1	308	6.9	2	113.67
6	265	7	780	1	331	42	1	95.48
27	217	10.6	13596	1	308	18.2	3	76.79
28	214	10.3	3284	1	304	5.4	2	99.29
1	255	3.9	552	1	304	12.6	2	65.01
3	114	4.1	19	1	303	46.4	3	59.05
26	214	10.9	2824	1	304	5.4	2	97.7
0	145	8.4	911	1	321	36.6	2	49.8
17	267	16.1	346	1	0	0	1	121.33
14	257	5.4	1435	1	304	12.9	2	83.38
10	267	6	2832	13	308	18.8	2	51.38
7	254	5.3	2022	1	304	5.3	2	83.13
16	254	5.1	1693	1	308	9.2	2	58.92
7	145	6.4	649	1	304	23.3	2	119.69

Appendix B Pavement ME Data

This section contains 30 sample data points for the Pavement ME dataset formatted as they were when directly imported into the models.

Table B.1 Extended Sample Data for Pavement ME Dataset

AC_ Moduli	**AC_ Thickness**	**Base_ Moduli**	**Cumulative_ Heavy_Trucks**	**Base_ Thickness**	**Subgrade_ Moduli**	**Total Deformation Pavement ME**
200	2.5	15	238723	8	7	0.366
200	2.5	15	291087	8	7	0.369
200	2.5	15	343460	8	7	0.393
200	2.5	15	564462	8	7	0.489
200	2.5	15	629118	8	7	0.524
200	2.5	15	759631	8	7	0.546
200	2.5	15	826818	8	7	0.549
200	2.5	15	939859	8	7	0.551
200	2.5	15	1102780	8	7	0.56
200	2.5	15	1159200	8	7	0.563
200	2.5	15	1341970	8	7	0.609
200	2.5	15	1408770	8	7	0.62
200	2.5	15	1829840	8	7	0.633
200	2.5	15	1887950	8	7	0.638
200	2.5	15	2520320	8	7	0.68
200	2.5	15	3111740	8	7	0.717
200	2.5	15	3833920	8	7	0.742
200	2.5	15	3961150	8	7	0.743
200	2.5	15	4274080	8	7	0.748
200	2.5	15	4565030	8	7	0.76
200	2.5	15	5096280	8	7	0.766
200	2.5	15	5240120	8	7	0.769
200	2.5	15	5317560	8	7	0.772
200	2.5	15	5476180	8	7	0.773
200	2.5	15	5545290	8	7	0.774

AC_ Moduli	AC_ Thickness	Base_ Moduli	Cumulative_ Heavy_Trucks	Base_ Thickness	Subgrade_ Moduli	Total Deformation Pavement ME
200	2.5	15	5611160	8	7	0.775
200	2.5	15	5675560	8	7	0.776
200	2.5	15	6011770	8	7	0.782
200	2.5	15	6473470	8	7	0.804
200	2.5	15	6743220	8	7	0.807

Appendix C Python Code Material Properties Based Models

C.1 Random Forests Asphalt Layer Only Deformation Prediction Model Using Pavement ME Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
df =
pd.read_excel('/content/ABUDATA_RANDOMFOREST.xlsx',sheet_name='Input3_ACDeform'
)
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
target_column = ['Permanent_Deformation_AConly_inches']
traffic_column = ['Cumulative_Heavy_Trucks']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
```

```
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.40, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
y_pred = rfr.predict(X_test)
x_ax = range(len(y_test))
plt.plot(x_ax, y_test, linewidth=1, label="original")
plt.plot(x_ax, y_pred, linewidth=1.1, label="predicted")
plt.title("y-test and y-predicted data")
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend(loc='best',fancybox=True, shadow=True)
plt.grid(True)
plt.show()
# Creating heat map for correlation study which will give us idea about study variables
and their inter relationships
#color is correlation
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
        xticklabels=corr.columns.values,
        yticklabels=corr.columns.values)
plt.figure(figsize=(10,10))
```

```
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('True Values', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#Need to make sure labels are correct above for this to work...fix them before running the
next block
dfr3 = pd.DataFrame(X_test)
dfr3.columns = ['Subgrade_Moduli',  'AC_Moduli',  'Base_Moduli','AC_Thickness',
'Base_Thickness'        ,'Cumulative_Heavy_Trucks']
print(dfr3)
dfr3[predictors] = dfr3[predictors]*unnormal[predictors].max()
dfr3["ACDeform Real"] = y_test
dfr3["ACDeform Predicted"] = y_pred
print(dfr3)
dfr3.to_csv('AbuRF_ACDeform_Export.csv')
```

C.2 Random Forests Total Deformation Prediction Model Using Pavement ME Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
```

```
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
df =
pd.read_excel('/content/ABUDATA_RANDOMFOREST.xlsx',sheet_name='Input2_TotalDefor
m')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
target_column = ['Permanent_Deformation_total_inches']
traffic_column = ['Cumulative_Heavy_Trucks']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.40, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
```

```
y_pred = rfr.predict(X_test)
x_ax = range(len(y_test))
plt.plot(x_ax, y_test, linewidth=1, label="original")
plt.plot(x_ax, y_pred, linewidth=1.1, label="predicted")
plt.title("y-test and y-predicted data")
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend(loc='best',fancybox=True, shadow=True)
plt.grid(True)
plt.show()
# Creating heat map for correlation study which will give us idea about study variables and their inter relationships
#color is correlation
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
        xticklabels=corr.columns.values,
        yticklabels=corr.columns.values)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('True Values', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#Need to make sure labels are correct above for this to work...fix them before running the next block
```

```
dfr3 = pd.DataFrame(X_test)
dfr3.columns = ['AC_Moduli',      'AC_Thickness',     'Base_Moduli',
'Cumulative_Heavy_Trucks', 'Base_Thickness'     ,'Subgrade_Moduli']
print(dfr3)
dfr3[predictors] = dfr3[predictors]*unnormal[predictors].max()
dfr3["TotalDeform Real"] = y_test
dfr3["TotalDeform Predicted"] = y_pred
print(dfr3)
dfr3.to_csv('AbuRF_TotalDeform_Export.csv')
```

C.3 Random Forests IRI Prediction Model Using Pavement ME Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
df =
pd.read_excel('/content/ABUDATA_RANDOMFOREST.xlsx',sheet_name='Input1_IRI')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
target_column = ['IRI']
```

```
traffic_column = ['Cumulative_Heavy_Trucks']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.40, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
y_pred = rfr.predict(X_test)
x_ax = range(len(y_test))
plt.plot(x_ax, y_test, linewidth=1, label="original")
plt.plot(x_ax, y_pred, linewidth=1.1, label="predicted")
plt.title("y-test and y-predicted data")
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend(loc='best',fancybox=True, shadow=True)
plt.grid(True)
plt.show()
# Creating heat map for correlation study which will give us idea about study variables and their inter relationships
#color is correlation
```

```
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
    xticklabels=corr.columns.values,
    yticklabels=corr.columns.values)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('True Values', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#Need to make sure labels are correct above for this to work...fix them before running the next block
dfr3 = pd.DataFrame(X_test)
dfr3.columns = ['AC_Thickness', 'Base_Thickness', 'Cumulative_Heavy_Trucks', 'Subgrade_Moduli', 'Base_Moduli' ,'AC_Moduli']
print(dfr3)
dfr3[predictors] = dfr3[predictors]*unnormal[predictors].max()
dfr3["IRI Real"] = y_test
dfr3["IRI Predicted"] = y_pred
print(dfr3)
dfr3.to_csv('AbuRF_IRI_Export.csv')
```

C.4 Random Forests Top-Down Fatigue Cracking Prediction Model Using Pavement ME Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

```
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
df =
pd.read_excel('/content/ABUDATA_RANDOMFOREST.xlsx',sheet_name='Input4_ACTop-
DownFatigueCrack')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
target_column = ['ACTop-DownFatigueCracking_%LaneArea']
traffic_column = ['Cumulative_Heavy_Trucks']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.40, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
```

```
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
y_pred = rfr.predict(X_test)
x_ax = range(len(y_test))
plt.plot(x_ax, y_test, linewidth=1, label="original")
plt.plot(x_ax, y_pred, linewidth=1.1, label="predicted")
plt.title("y-test and y-predicted data")
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend(loc='best',fancybox=True, shadow=True)
plt.grid(True)
plt.show()
# Creating heat map for correlation study which will give us idea about study variables
and their inter relationships
#color is correlation
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
        xticklabels=corr.columns.values,
        yticklabels=corr.columns.values)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
```

```
plt.xlabel('True Values', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#Need to make sure labels are correct above for this to work...fix them before running the
next block
dfr3 = pd.DataFrame(X_test)
dfr3.columns = ['AC_Thickness',     'Base_Thickness',     'Base_Moduli',
'Subgrade_Moduli',    'AC_Moduli'   ,'Cumulative_Heavy_Trucks']
print(dfr3)
dfr3[predictors] = dfr3[predictors]*unnormal[predictors].max()
dfr3["ACTop-Down Real"] = y_test
dfr3["ACTop-Down Predicted"] = y_pred
print(dfr3)
dfr3.to_csv('AbuRF_Top-Down_Export.csv')
```

C.5 Random Forests Bottom-Up Cracking Prediction Model Using Pavement ME Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
```

```
df =
pd.read_excel('/content/ABUDATA_RANDOMFOREST.xlsx',sheet_name='Input5_Bottom-
Up_Crack')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
target_column = ['Bottom-UpCracking_%']
traffic_column = ['Cumulative_Heavy_Trucks']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.40, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
y_pred = rfr.predict(X_test)
x_ax = range(len(y_test))
plt.plot(x_ax, y_test, linewidth=1, label="original")
plt.plot(x_ax, y_pred, linewidth=1.1, label="predicted")
plt.title("y-test and y-predicted data")
```

```
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend(loc='best',fancybox=True, shadow=True)
plt.grid(True)
plt.show()
# Creating heat map for correlation study which will give us idea about study variables and their inter relationships
#color is correlation
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
    xticklabels=corr.columns.values,
    yticklabels=corr.columns.values)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('True Values', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#Need to make sure labels are correct above for this to work...fix them before running the next block
dfr3 = pd.DataFrame(X_test)
dfr3.columns = ['Subgrade_Moduli',  'Base_Moduli','AC_Thickness',
'Cumulative_Heavy_Trucks', 'AC_Moduli'    ,'Base_Thickness']
print(dfr3)
dfr3[predictors] = dfr3[predictors]*unnormal[predictors].max()
```

```
dfr3["Bottom-Up Crack Real"] = y_test
dfr3["Bottom-Up Crack Predicted"] = y_pred
print(dfr3)
dfr3.to_csv('AbuRF_Bottom-Up_Export.csv')
```

C.6 Random Forests Total Deformation Prediction Model Using LTPP Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
df =
pd.read_excel('/content/LTPP_REGRESSION_RUTTING_10.20.22.xlsx',sheet_name='INPUT1
_RUTTING')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
df.dtypes
df.isnull().values.any()
#if true, make sure excel file has no formulas (copy as values)
target_column = ['Rutting_Inches']
predictors = list(set(list(df.columns))-set(target_column))
```

```
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)

rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
# Creating heat map for correlation study which will give us idea about study variables
and their inter relationships
#color is correlation
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
    xticklabels=corr.columns.values,
    yticklabels=corr.columns.values)
y_pred = rfr.predict(X_test)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
```

```
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('Pavement ME', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#make sure labels match with columns...should be same as initial randomization but double check by multiplying a number by the max value of that column in excel and see if it exists
dfr = pd.DataFrame(X_test)
dfr.columns = ['Asphalt_Total_Thickness', 'KESAL_CUMULATIVE', 'Section_Age_Years', 'Asphalt_Majority_Material', 'Base_Total_Thickness', 'Climatic_Zone', 'Subgrade_LayerType', 'Base1_Type']
print(dfr)
dfr[predictors] = dfr[predictors]*unnormal[predictors].max()
dfr["Field Rutting"] = y_test
dfr["Rutting RF Predicted"] = y_pred
print(dfr)
dfr.to_csv('LTPP_REGRESSION_RUTTING1_EXPORT.csv')
```

C.7 Random Forests IRI Prediction Model Using LTPP Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
```

```
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
df =
pd.read_excel('/content/LTPP_REGRESSION_RUTTING_10.20.22.xlsx',sheet_name='INPUT1
_IRI')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
df.dtypes
df.isnull().values.any()
#if true, make sure excel file has no formulas (copy as values)
target_column = ['IRI']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
```

```
# Creating heat map for correlation study which will give us idea about study variables and their inter relationships
#color is correlation
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
        xticklabels=corr.columns.values,
        yticklabels=corr.columns.values)
y_pred = rfr.predict(X_test)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('Pavement ME', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#make sure labels match with columns...should be same as initial randomization but double check by multiplying a number by the max value of that column in excel and see if it exists
dfr = pd.DataFrame(X_test)
dfr.columns = ['Section_Age_Years', 'Subgrade_LayerType', 'Asphalt_Total_Thickness', 'KESAL_CUMULATIVE', 'Asphalt_Majority_Material', 'Base1_Type', 'Base_Total_Thickness', 'Climatic_Zone']
print(dfr)
dfr[predictors] = dfr[predictors]*unnormal[predictors].max()
dfr["Field IRI"] = y_test
dfr["IRI RF Predicted"] = y_pred
```

```
print(dfr)
dfr.to_csv('LTPP_REGRESSION_IRI1_EXPORT.csv')
```

C.8 Random Forests Fatigue Cracking Prediction Model Using LTPP Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
df =
pd.read_excel('/content/LTPP_REGRESSION_RUTTING_10.20.22.xlsx',sheet_name='INPUT1
_FATIGUECRACKING')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
df.dtypes
df.isnull().values.any()
#if true, make sure excel file has no formulas (copy as values)
target_column = ['Fatigue Cracking']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
```

```
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
# Creating heat map for correlation study which will give us idea about study variables and their inter relationships
#color is correlation
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
        xticklabels=corr.columns.values,
        yticklabels=corr.columns.values)
y_pred = rfr.predict(X_test)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('Pavement ME', fontsize=15)
```

```
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#make sure labels match with columns...should be same as initial randomization but double check by multiplying a number by the max value of that column in excel and see if it exists
dfr = pd.DataFrame(X_test)
dfr.columns = ['Subgrade_LayerType', 'Climatic_Zone', 'Section_Age_Years', 'Base_Total_Thickness', 'Asphalt_Majority_Material', 'Base1_Type', 'Asphalt_Total_Thickness', 'KESAL_CUMULATIVE']
print(dfr)
dfr[predictors] = dfr[predictors]*unnormal[predictors].max()
dfr["Field Fatigue Cracking"] = y_test
dfr["Fatigue Cracking RF Predicted"] = y_pred
print(dfr)
dfr.to_csv('LTPP_REGRESSION_fatiguecracking1_EXPORT.csv')
```

C.9 Random Forests Transverse Cracking Prediction Model Using LTPP Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
```

```
df =
pd.read_excel('/content/LTPP_REGRESSION_RUTTING_10.20.22.xlsx',sheet_name='INPUT1
_TRANSVERSECRACKING')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
df.dtypes
df.isnull().values.any()
#if true, make sure excel file has no formulas (copy as values)
target_column = ['Transverse Cracking']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
# Creating heat map for correlation study which will give us idea about study variables
and their inter relationships
#color is correlation
```

```
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
        xticklabels=corr.columns.values,
        yticklabels=corr.columns.values)
y_pred = rfr.predict(X_test)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('Pavement ME', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#make sure labels match with columns...should be same as initial randomization but double check by multiplying a number by the max value of that column in excel and see if it exists
dfr = pd.DataFrame(X_test)
dfr.columns = ['KESAL_CUMULATIVE', 'Asphalt_Majority_Material', 'Subgrade_LayerType', 'Section_Age_Years', 'Base_Total_Thickness', 'Asphalt_Total_Thickness', 'Climatic_Zone', 'Base1_Type']
print(dfr)
dfr[predictors] = dfr[predictors]*unnormal[predictors].max()
dfr["Field Transverse Cracking"] = y_test
dfr["Transverse Cracking RF Predicted"] = y_pred
print(dfr)
dfr.to_csv('LTPP_REGRESSION_TransverseCracking1_EXPORT.csv')
```

C.10 Random Forests Non-Wheel Path Cracking Prediction Model Using LTPP Data

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import sklearn
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import load_boston
from sklearn.datasets import make_regression
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
from sklearn import set_config
from google.colab import files
uploaded = files.upload()
df =
pd.read_excel('/content/LTPP_REGRESSION_RUTTING_10.20.22.xlsx',sheet_name='INPUT_
LONGITUDINAL CRACKINF_NWP')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
df.dtypes
df.isnull().values.any()
#if true, make sure excel file has no formulas (copy as values)
target_column = ['Longitudinal_Cracking_NWP']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
```

```
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
# Creating heat map for correlation study which will give us idea about study variables
and their inter relationships
#color is correlation
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
        xticklabels=corr.columns.values,
        yticklabels=corr.columns.values)
y_pred = rfr.predict(X_test)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('Pavement ME', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
```

plt.show()

#make sure labels match with columns...should be same as initial randomization but double check by multiplying a number by the max value of that column in excel and see if it exists

dfr = pd.DataFrame(X_test)

dfr.columns = ['Section_Age_Years', 'Asphalt_Majority_Material', 'Base1_Type', 'Base_Total_Thickness', 'Climatic_Zone', 'KESAL_CUMULATIVE', 'Subgrade_LayerType', 'Asphalt_Total_Thickness']

print(dfr)

dfr[predictors] = dfr[predictors]*unnormal[predictors].max()

dfr["Field NWP Longitudinal Cracking"] = y_test

dfr["NWP Longitudinal Cracking RF Predicted"] = y_pred

print(dfr)

dfr.to_csv('LTPP_REGRESSION_NWP_LongitudinalCRacking1_EXPORT.csv')

C.11 Random Forests Wheel Path Cracking Prediction Model Using LTPP Data

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

import sklearn

from sklearn.ensemble import RandomForestRegressor

from sklearn.datasets import load_boston

from sklearn.datasets import make_regression

from sklearn.metrics import mean_squared_error

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import scale

import matplotlib.pyplot as plt

from sklearn import set_config

from google.colab import files

uploaded = files.upload()

```
df =
pd.read_excel('/content/LTPP_REGRESSION_RUTTING_10.20.22.xlsx',sheet_name='INPUT1
_LONGITUDINALCRACKING_WP')
#Importing the data
#If you edit the file and reupload make sure to erase the original file from the library
print(df.shape)
print(df)
df.dtypes
df.isnull().values.any()
#if true, make sure excel file has no formulas (copy as values)
target_column = ['Longitudinal_Cracking_WP']
predictors = list(set(list(df.columns))-set(target_column))
unnormal = df.copy()
print(df[predictors])
df[predictors] = df[predictors]/df[predictors].max()
print(df[predictors])
#df.describe()
X = df[predictors].values
y = df[target_column].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=40)
print(X_test) #added this in
print(X_train.shape); print(X_test.shape)
set_config(print_changed_only=False)
 rfr = RandomForestRegressor()
print(rfr)
rfr.fit(X_train, y_train)
score = rfr.score(X_train, y_train)
print("R-squared:", score)
# Creating heat map for correlation study which will give us idea about study variables
and their inter relationships
#color is correlation
```

```
import seaborn as sns
corr = df.corr()
sns.heatmap(corr,
        xticklabels=corr.columns.values,
        yticklabels=corr.columns.values)
y_pred = rfr.predict(X_test)
plt.figure(figsize=(10,10))
plt.scatter(y_test, y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')
p1 = max(max(y_pred), max(y_test))
p2 = min(min(y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('Pavement ME', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
#make sure labels match with columns...should be same as initial randomization but double check by multiplying a number by the max value of that column in excel and see if it exists
dfr = pd.DataFrame(X_test)
dfr.columns = ['Asphalt_Majority_Material', 'Subgrade_LayerType', 'KESAL_CUMULATIVE', 'Base1_Type', 'Base_Total_Thickness', 'Asphalt_Total_Thickness', 'Climatic_Zone', 'Section_Age_Years']
print(dfr)
dfr[predictors] = dfr[predictors]*unnormal[predictors].max()
dfr["Field WP Longitudinal Cracking"] = y_test
dfr["WP Longitudinal Cracking RF Predicted"] = y_pred
print(dfr)
dfr.to_csv('LTPP_REGRESSION_WP_LongitudinalCRacking1_EXPORT.csv')
```