3-2021

Data- Driven Asset Degradation Modeling: An Enterprise-wide Roof System Case Study

Kurt R. Lamm

DATA-DRIVEN ASSET DEGRADATION MODELING:
AN ENTERPRISE-WIDE ROOF SYSTEM CASE STUDY

THESIS

Kurt R. Lamm, R.A., Civilian, USAF

AFIT-ENV-MS-21-M-242

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENV-MS-21-M-242

DATA- DRIVEN ASSET DEGRADATION MODELING:
AN ENTERPRISE-WIDE ROOF SYSTEM CASE STUDY

THESIS

Presented to the Faculty

Department of Systems Engineering and Management

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Engineering Management.

Kurt R. Lamm, BS Architecture, R.A.

Civilian, USAF

March 2021

AFIT-ENV- MS-21-M-242

DATA-DRIVEN ASSET DEGRADATION MODELING:

AN ENTERPRISE-WIDE ROOF SYSTEM CASE STUDY

Kurt R. Lamm, BS Architecture, R.A.

Civilian, USAF

Committee Membership:

Lt Col Steven J. Schuldt, PhD, P.E.

Chair

Maj Justin D. Delorit, PhD, P.E.

Member

Dr. Michael N. Grussing, PhD, P.E.

Member

Dr. Louis B. Bartels, PhD, P.E.

Member

AFIT-ENV- MS-21-M-242

Abstract

Organizations with large facility and infrastructure portfolios have used asset management databases for over ten years to collect and standardize asset condition data. Decision makers use this data to predict asset degradation and expected service-life, enabling prioritized maintenance, repair, and renovation actions that reduce asset life-cycle costs and achieve organizational objectives. However, these asset condition forecasts are calculated using standardized, self-correcting distribution models that rely on poorly-fit, continuous functions. This research presents four stepwise asset condition forecast models that utilize historical asset inspection data to improve prediction accuracy: (1) Slope, (2) Weighted Slope, (3) Condition-intelligent Weighted Slope, and (4) Nearest Neighbor. Model performance was evaluated against BUILDER SMS, the industry-standard asset management database, using data for five roof types on 8,549 facilities across 61 U.S. military bases within the Contiguous United States. The stepwise Weighted-slope model predicted asset degradation more accurately than BUILDER SMS 92% of the time. These results suggest that using historical condition data, alongside or in-place of manufacturer expected service-life, may increase degradation and failure prediction accuracy. Additionally, the developed models are expected to improve prediction skills as data quantity increases over time. These results are expected to enable decision makers to achieve more accurate enterprise management and reduce infrastructure budget shortfalls.

Acknowledgments

My pursuit of education at AFIT has been supported by many individuals, and much thanks will be mentioned here. First, I would like to thank my wife and children for their support. COVID-19 has played a large role in shaping my education, and I could not have been this successful without my family's support. Secondly, I would like to express my sincere appreciation for the faculty advisership of Lt Col Steven Schuldt and Maj Justin Delorit for their guidance and support throughout the course of this thesis effort. The insight, experience, and independent studies were greatly appreciated. The hours of education and revision you provided to guide my research and writing efforts were an essential propulsion that got me to this point. I would, also like to thank both Dr. Mike Grussing and Dr. Louis Bartels, from the US Army Construction Engineering Research Laboratory (CERL) for their support and expertise provided to me in this endeavor. Last but not least, I would like to thank two students, Capt Sarah Brown and Capt Evan Fortney, for their collaboration, ideas, and work on tangential topics during our independent study courses.

Kurt R. Lamm

Table of Contents

List of Figures

Page

List of Tables

Page

DATA-DRIVEN ASSET DEGRADATION MODELING:

AN ENTERPRISE-WIDE ROOF SYSTEM CASE STUDY

I. Introduction

Background

Since 1998, the U.S. Army Corps of Engineers (USACE) Engineer Research Development Center (ERDC) has been helping solve our Nation's most challenging problems in civil and military engineering, geospatial sciences, water resources, and environmental sciences for the Army, Department of Defense (DoD), civilian agencies, and our Nation's public good (*ERDC Mission* 2019). Part of this initiative focuses on maximizing the environmental sustainability and improving the life-cycle management of DoD's installation and infrastructure assets. In 2012, the U.S. Army possessed over 165,000 buildings totaling more than 1.1 billion square feet and spent 55% of its real property budget maintaining and repairing these facilities (U.S. Army 2012). The U.S. Government Accountability Office (GAO) requires that taxpayer dollars are spent in a fashion that best suits national interests, and GAO saw this spending as a black box process for spending federal funds (GAO 2011). Accordingly, ERDC developed the BUILDER Sustainment Management System (SMS) to inventory, assess, and proactively manage the condition of all Army assets. Since its development and implementation by the U.S Army, the U.S. Marine Corps (2010), the U.S. Navy (2011), and U.S. Air Force (2014) have also adopted BUILDER SMS as their primary asset management system (U.S. Army 2012).

BUILDER SMS is the current foundation for planning the life-cycle management of nearly all DoD infrastructure and facility assets. BUILDER SMS is more than a database; it utilizes built-in technology to convert data input into condition scores that are projected over time, allowing for more intelligent planning. However, the Air Force Civil Engineer Center (AFCEC) Operations Directorate realizes the limitations of BUILDER when it says this about the SMS data, “Technology is never the complete solution. There is an immediate need to provide guidance to the field to achieve the mission of standardizing, collecting, analyzing, validating and accurate horizontal and vertical infrastructure data to support resource allocation and operational decisions.” (Somers and Bates 2019). The use of an Enterprise Asset Management (EAM) system, such as BUILDER SMS, allows institutions to plan projects for repair, replacement and divestiture with far greater purpose and quantifiable justification. When compared to the reactionary alternative, this is an improvement, but BUILDER SMS projections currently operate with a margin of error that can be improved.

Degradation is a given in asset management, and it is usually a very clear focus of most life-cycle management practices. Large entities use EAM systems such as BUILDER SMS to account for their assets and plan the reoccurring expenditures required to maintain those assets. Even with a robust tool like BUILDER SMS, facility condition risk exposure still manifests itself in budgetary draws that are both surprising and crippling to planning. With such high fidelity in asset information, BUILDER SMS should be able to increase planning accuracy to limit risk.

BUILDER SMS employs a degradation function to produce future life-cycle anticipations of the asset over time (Grussing et al. 2006). At first glance, this predictive model seems intuitive, but predictive degradation is not simple to forecast accurately. Asset degradation does not often assimilate to a static degradation model (Alley et al. 2017), where all assets behave like the population average. Assets more often vary both temporally and spatially in a manner that is a bit more complex, which is why stepwise predictive models have been proposed for research. Stepwise models selectively compare a target asset with population assets by using inspection data to make tailored predictions. This means that sample data are divided into subsets based on similar characteristics of age, condition, rate of degradation, or other criteria before using them as input to forecast models.

Problem Statement

This paper aims to investigate stepwise methods for producing asset degradation models to make more skillful forecast predictions than the current BUILDER SMS Weibull model. Much like the current BUILDER SMS model, the observed condition information recorded at discrete age timesteps and gathered during condition assessments will be used to predict future asset condition values. The USAF has selected BUILDER SMS as its Enterprise Asset Management (EAM) system, and over ten years of data have been logged into the system at this time. Improving BUILDER SMS degradation curve calculations will increase planning accuracy and could eliminate future funding issues due to deferred actions. In FY18 alone, the Department of Defense (DoD) requested $12.8 billion (DoD Real Property Portfolio Office 2018) for facility sustainment,

restoration, and modernization (SRM). The budget pales in comparison to the value of DoD facility assets, which was estimated at $1.046 (DoD Real Property Portfolio Office 2018). While basing the cost for sustaining these 568,383 DoD facilities (DoD Real Property Portfolio Office 2018) on a standard degradation formula is helpful, it may lead to non-optimal investment decisions. Enhancement of the BUILDER SMS predictive degradation formula will not solve all budget fidelity issues. However, it should improve degradation prediction accuracy and reduce current variations from reality. Additionally, there will be opportunities to employ singular or combinations of models as ensembles to target specific goals and timelines of different decision makers.

Several areas of study need to be explored to support create stepwise asset degradation modeling and forecast development: 1. Asset degradation models; 2. Factors that influence roofing asset degradation; 3. Forecasting and data projections.

Research Objectives

A case study of roofing data has been selected to investigate and utilize data to develop these stepwise models. Using a single asset system type, roofing, removes service-life variabilities between systems *(think HVAC vs Roofing)* and focuses on subset asset behaviors to expedites results. For example, interior systems are sheltered from the weather, but exterior systems such as roof and wall assemblies are not. As a result, roofing and wall systems experience degradation as a function of both time and weather. This acute difference in weather exposure suggests that these systems are much more at risk to weather, and their predictive degradation should reflect this. However, weather is not the only factor that makes systems degrade at different rates. These complex

relationships between different building systems and their resultant service-life differences suggest that inter-system comparison is valid while intra-system comparison is not. Furthermore, while average roof systems (B30) have an expected life-cycle of 20-30 years, wall systems (B20) have an expected life-cycle that is typically three times longer. One can infer that roof systems are much less resilient than wall systems since they degrade three times as fast. However, climatic differences can drastically affect these tendencies.

The proposed research focuses on forecasting the degradation of five major commercial and residential roofing category types ("Roofing Systems" 2016): Built-Up Roofing (BUR), Modified Bitumen Roofing (MOD), Single-Ply (SP), Shingle (SH), and Standing Seam Metal (SSM). Roofing data is used as a case study to create improved degradation methodologies that can be applied to various asset types, not just roofing. The current BUILDER SMS projection model and the stepwise models are compared to observed conditions as a way to rank performance in terms of the Delta Condition Index (DCI), which is the Observed Condition Index (OCI) minus each model's prediction, called the Expected Condition Index (ECI). Computationally, the models produce both specific service-life predictions for individual assets and average population service-life predictions. This study analyzes individual model predictions and combinations of those models as ensemble forecasts to evaluate roof service life from a decision-maker perspective. United States Air Force (USAF) B30 Roofing System data from BUILDER SMS are used to analyze the real-world performance of the five roofing types mentioned previously at 61 geographically unique base locations. This sample was selected to

provide a representative sample of variations in manufacturer, climate, maintenance, and other conditions present across the enterprise.

While the USAF employs standardized maintenance plans, routine inspections, and uniform condition metrics, data quality and consistency vary across locations based on the subjectivity of technician ratings of the assets and projects that improve an asset's condition. Stringent pre-processing and filtering of the data have been employed to eliminate inconsistencies. Additionally, data quantity increases with the number of locations included in the study and as inspections occur over time. The USAF data provides a unique opportunity to maximize both the quantity and uniform quality of asset data simultaneously.

This analysis hypothesizes stepwise degradation forecast models as an improvement to the way BUILDER SMS data is employed. It also creates a discussion forum regarding model utilization. Assuming stepwise methodologies outperform current continuous statistical models, better asset performance predictions will improve infrastructure planning, budgeting, and enterprise management. However, a discussion about how these models can be used singularly or in concert to empower different levels of decision making is also warranted to fully understand the benefits of multiple models with varying skill. For example, some models may produce more accurate short-term forecasts, while a different model may more skillfully produce long-term forecasts. This thesis will discuss decision making and how it relates to forecast models to better connect these models with practical application and field use.

Thesis Organization

This thesis follows a scholarly format in which chapters 3 and 4 each serve as stand-alone academic journal publications. Chapter 2 follows a traditional literature review format where existing research is investigated, and detailed application to this research is developed as a foundation that grounds the new research conducted. This overview of the body of knowledge covers current asset degradation models, factors that influence roofing asset degradation, and forecasting and data projections. Although Chapter 2 was not published, it connects the new research to existing findings. These roots are the impetus for the following two publications.

Chapter 3, "Data-Driven Asset Degradation Modeling: An Enterprise-wide Roof System Case Study," provides an in-depth technical understanding and application of the methods developed in this research for a scholarly, non-DoD audience. The target audience, both public and private, is shielded from confusing government and military-bureaucratic regulations, frameworks, jargon, and initiatives while scholarly content is preserved. This article presents the four stepwise asset condition forecast models developed: (1) Slope, (2) Weighted Slope, (3) Condition-intelligent Weighted Slope, and (4) Nearest Neighbor. Model performance is evaluated against BUILDER SMS forecasts, which uses the industry standard's continuous self-correcting prediction model. The results suggest that using historical condition data, alongside or in-place of manufacturer expected service life, may increase the accuracy of degradation and failure prediction models. Additionally, the resulting improvements in forecast skill are discussed as a way to enable decision makers to manage facility assets more proactively and achieve better

returns on facility investments. The target journal for this paper is the *Journal of Building Engineering,* an international peer-reviewed journal published by Elsevier with an impact rating of 3.379 (*Journal of Building Engineering* 2021).

Chapter 4, “Data-Driven Asset Condition Models: An Air Force Roof System Case Study,” provides a general non-technical understanding and application of data-driven models in plain English format for a DoD-wide audience. The target audience, both technical and non-technical, is presented with the research concepts and how they apply to specific government and military objectives and initiatives while emphasizing the role and value of individuals who interface with the BUILDER SMS database. The payback for individuals who have developed the inventory and condition data that the BUILDER database houses are not often afforded a comprehensive understanding of their contribution to the overall progress of DoD asset management objectives. This paper aims to enlighten that audience. This paper's target journal is *The Military Engineer* (TME), a well-known DoD and A/E/C industry-partnered journal published by the Society of American Military Engineers (SAME). TME is circulated to over 30,000 people quarterly via both print and digital mediums (SAME 2021).

II. Literature Review

Chapter Overview

The purpose of this chapter is to summarize the current body of literature that surrounds and connects data-driven asset degradation modeling. After highlighting the existing bounds of research, a clear gap in research can be seen, and the focus of this research becomes precise. This section of the paper is sub-divided into four parts: (1) asset management and degradation models; (2) roofing degradation factors; (3) forecasting and prediction models; and (4) area of contribution. Part one focuses on existing asset management practices, including the necessity for asset management, data collection, and existing asset management products. Part two develops an understanding of the real-world factors that cause roofs (and other assets) to degrade in condition over time. Part three explains current forecasting method types and how they are used to predict future conditions. Finally, chapter four discusses the contributions of each of these three research areas and how they are individually limited, but using them together creates a unique gap of opportunity for research and contribution. Figure 1 depicts the literature categories that will be discussed and the research gap that exists when these areas are combined. This research gap is called the “Area of Contribution.”

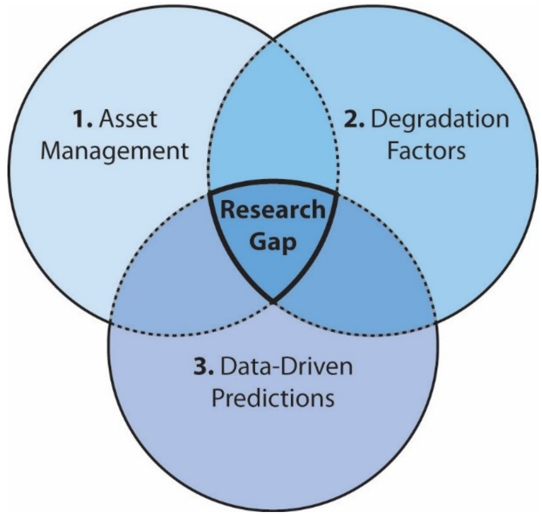


Figure 1: Literature Areas

Roofing Degradation Factors

The second area of study that will need to be explored is the degradation of roofing system conditions. Current degradation models suggest that roofing systems age with time, or age-based obsolescence (Grussing 2014), which in and of itself is not incorrect. However, weather exposure suggests that roofing systems are at risk to weather factors that accumulate over time and not merely time itself. Additionally, roof design, material, and maintenance are essential to understanding roofing systems' varying degradation rates. As a result, predictive degradation models should show this. While industry currently views roofing assets from a life-cycle perspective, understanding the unique factors that contribute to roofing degradation provides insight into the degradation of assets as a varying annual component that is not captured in overall service-life projections. This section of the paper aims to explain roofing degradation by exploring the factors that cause the degradation.

Many industries have vested interests in roofing system performance, but climate and weather are typically a sub-conscious, minor focus compared to their bottom line. Unlike most roofing owners, roofing manufacturers are interested in providing minimum performance requirements. These roof manufacturers provide material and system warranties for roofing systems that are typically specified to last 20 years or more. Still, when measured, the actual variation in life-cycle performance is much harder to predict (Grant et al. 2014). Roofing systems have been shown to have varying service lives depending on heat aging, roof traffic, roof slope, and annual maintenance (Hodges 1999).

Additionally, time seems to be an important factor inherent to all roofing service lives (Cash 2006). Life-cycle analysis is often a cornerstone to the justification that supports roofing project decisions. However, when an analysis of five service-life software was conducted, the variation of predictions for the service life of three different roof systems (Built-up, Thermoplastic or Single-ply, and Vegetated) within these models was extreme (Grant et al. 2014). Since a broad life-cycle view of roofing systems results in large prediction variations, an understanding of climate and weather impact on roof systems' performance is suggested to provide a more granular explanation of the degradation inherent to all roof systems.

The Government Accountability Office (GAO) recently spoke on the importance of understanding how the climate may change in the future and what impacts it would have on DoD facilities and infrastructure worldwide by stating, "installations' master plans and related installation planning documents did not (1) identify a range of possible extreme weather events and climate change effects that could affect the installation, (2)

assess the likelihood of each event occurring and the possible effect on the installation, and (3) identify potential responses to these events" (*Climate Resilience* 2019). While large catastrophic events like hurricanes, flooding, wildfires, and rising sea-level capture most audiences' attention, the small, aggregated effects of extreme temperatures, increased precipitation, and higher-speed winds over time typically receive much less consideration, even though they are equally important. Major Justin Delorit (USAF) highlights the importance of forecasting energy usage as a function of climate factors to allow informed understanding and decision making for heating and cooling practices across the United States, which are expected to dynamically change in the next century (Delorit et al. 2020). While roofs perform in a much different capacity than heating, cooling, and ventilation systems, the aggregated effects of climate shift over time are likely to have a noticeable impact on roof system performance as well. This aggregation of weather acting on roofing systems over time is the type of understanding needed to predict roof degradation due to weather factors compounded over time.

The insurance industry is chiefly concerned with the area of weather extremes, or acute degradation, and how they actively degrade facilities and infrastructure at a rapid rate because the conditions experienced are outside the designed range of system performance (Karl et al. 2013). Degradation from weather extremes may look like heat stress from high-temperature extremes and solar radiance or hail damage that may result from severe thunderstorms (Harkness and Hassanain 2001). Since weather extremes are likely to contribute to roofing system degradation at a much higher rate, the bounds (max/min) of weather factors are frequently more important than averages (Karl et al. 2013). While current research helps identify factors that contribute to roofing degradation

over time, it does not focus on climate or weather as individual factors for predicting future conditions, leaving the importance and magnitudes of weather still a mystery. Interestingly, recent research focuses more on the impact roofing has on the environment and not the converse. This research has shown through factor analysis that different roofing types contribute to global warming in an influential fashion (Grant et al. 2016), suggesting energy efficiency tailored roof system design and selection. While the correlation between roofing types and climate change is being drawn, the gap is quite broad when trying to attribute specific climate factors to roofing degradation.

While much of the roofing industry has focused on providing energy-efficient roofing materials, practices, and system designs (Habibi et al. 2020), there is little focus on installing roof types in climates to maximize service life. This optimization approach would focus on on unique material-based performance characteristics and not a factor analysis. Passive design principles focus on building placement, materials used, and design details that passively maximize architecture. In contrast, active design of a system ignores natural phenomen and uses energy input to maintain operational parameters. Simply put, this means that passive design focuses on using systems, materials, and design solutions in ways that minimize the overall system complexity and maximizing the use of simpler systems that already match design parameters. For heating and cooling, this means heat protection, heat modulation, and heat dissipation are the top priority (Bhamare et al. 2019). Using a passive approach in roofing means that regional climate characteristics guide roofing system decisions and minimize the adverse effects weather has on the selected roof system's service life. Still, the granular performance of systems over time is not addressed. For example, temperature’s effect on low-slope roofing

system service life has been researched (Cash 1997), but it has not been conducted at a scale that analyzes multiple or exhaustive weather factors.

Additionally, clay tile roofing degradation has shown a significant correlation to manufacturing variances in porosity, making it subject to freeze-thaw (Ducman et al. 2011). However, the effects of freeze-thaw on other roofing system types are generally undefined. Similarly, cementitious roofing tile degradation due to manufacturing density and porosity (Tonoli et al. 2011) has shown a correlation to its service life, but this again is only related to one roof system type. At this time, more data and research are required to understand how different roofing system type behaviors respond to individual weather factors, but regional climatic trends are generally understood. Asset management databases may provide ample opportunities to analyze weather relationships in the future, but for now, these relationships are not fully understood.

Asset Management and Degradation Models

Asset management methodologies have been in place for decades for several infrastructure domains, including roads and pavements, railroads, bridges, and distribution pipelines (Grussing 2014). These domains are primarily linear systems, and a failure in one segment of the infrastructure would almost guarantee a significant and disproportionate failure in that system/sub-system overall. For example, if a stormwater pipe would collapse, this would result in a near-complete failure in the pipe to convey stormwater. Similar results would be observed for failures in railways, pavements, or bridges. As the technological coupling of asset management principles expands to facilities and beyond linear/horizontal infrastructure, the framework for accurately

modeling these systems and systems of systems (SOS) morphs. The metrics used to assess infrastructure systems have changed over time and still vary based on the type of infrastructure being assessed and the municipality's objectives (Shahata 2013). Small municipalities with small linear infrastructure systems may use Microsoft Excel spreadsheets or Microsoft Access databases (Vanier and Danylo 1998). However, large entities such as the DoD or Metropolitan cities employ state-of-the-art asset management systems that employ much larger databases (Grant et al. 2014), which is where Sustainment Management Systems begin to vary. Two of these systems are BUILDER SMS (Grussing and Liu 2014) and BELCAM (Lounis et al. 1999). While the software differs somewhat technically, both concept models use temporally-sensitive asset conditions as the input for calculations for outputting service-life expectations. Furthermore, the service-life data are then used to prioritize, plan, and estimate the work required to maintain the system. Sustainment Management Systems target specific systems and their long-term performances in fashions that more realistically represent their actual performance in terms of an aggregation of complex sub-system reliabilities, where each sub-system plays an integral part of the overall system performance in a less-linear fashion.

Since 1998 the U.S. Army Corps of Engineers (USACE) Engineer Research Development Center (ERDC) has been helping solve our Nation's most challenging problems in civil and military engineering, geospatial sciences, water resources, and environmental sciences for the Army, Department of Defense (DoD), civilian agencies, and our Nation's public good (*ERDC Mission* 2019). Part of this initiative has focused directly on maximizing the environmental sustainability and life-cycle management of

the DoD's installation and infrastructure assets. In 2012, the U.S. Army possessed over 165,000 buildings totaling more than 1.1 billion square feet and spent 55% of its real property budget maintaining and repairing these facilities ("BUILDER Fact Sheet" 2012). The U.S. Government Accountability Office (GAO) requires that taxpayer dollars are spent in a fashion that best suits national interests, and GAO saw this spending as a black box process for spending federal funds (Defense Infrastructure 2008). Accordingly, ERDC developed the BUILDER SMS to inventory, assess, and proactively manage the condition of all Army Assets. Since its development and implementation by the U.S Army, the U.S. Marine Corps (2010), the U.S. Navy (2011), and U.S. Air Force (2014) have also adopted it as their primary asset management system ("BUILDER Fact Sheet" 2012).

BUILDER SMS is the current foundation for planning the life-cycle management of nearly all DoD infrastructure and facility assets. BUILDER SMS is more than a database; it utilizes built-in technology to convert data input into condition scores that are projected over time, allowing for more intelligent planning. However, the Air Force Civil Engineer Center (AFCEC) Operations Directorate realizes the limitations of BUILDER when it says this about the SMS data, "Technology is never the complete solution. There is an immediate need to provide guidance to the field to achieve the mission of standardizing, collecting, analyzing, validating and accurate horizontal and vertical infrastructure data to support resource allocation and operational decisions." (Somers et al. 2019). The use of an EAM system, such as BUILDER SMS, allows institutions to plan projects for repair, replacement, and divestiture with far greater purpose and

quantifiable justification compared to the reactionary alternative, but it currently does this with a margin of error that can be improved.

Degradation is inherent to asset management, and it is normally a very clear focus of most life-cycle management practices. EAM systems such as BUILDER are a technological tool that large entities use to comprehend how many assets they possess and plan the reoccurring expenditures required to maintain those assets. Even with a robust tool like BUILDER SMS, facility condition risk exposure still manifests itself in budgetary draws that are both surprising and crippling to planning (Climate Change Adaptation 2011). With such high fidelity in asset information, BUILDER SMS should be able to increase planning accuracy to limit risk. Using the information gathered in databases like BUILDER SMS to predict future asset conditions is called data-driven modeling.

Data-Driven Predictions

BUILDER SMS employs time-based condition inspection data and a degradation function to produce future life-cycle anticipations of individual assets over time (Grussing et al. 2006). At first glance, this predictive model seems intuitive, but predictive degradation is more complicated because asset degradation does not often assimilate a static time-based degradation model (Alley et al. 2017). Assets more often vary both temporally and spatially in a complex or even stochastic manner (Grant et al. 2014). As previously discussed, this degradation behavior results from climate, material performance, and other hidden factors. With surface-level research, these generic responses to known climatic conditions are easily discoverable. However, significant

statistical research is required for this logic to be applied to enhance predictive degradation in BUILDER SMS and resultant asset management planning (Grussing and Liu 2014). However, condition data alone is a powerful tool to create data-driven asset prediction models.

This data-driven approach to asset management has been increasing in popularity. It is also increasing in use as a management tool as the amount of data collected for facilities and infrastructure continues to grow. Converting these existing data sets into prediction models to forecast future asset conditions requires overcoming quantity, quality, and management decision threshold hurdles. In contrast to early Gompertz, Richard, or Morgan-Mercer-Flodin models (Sjostrom 2004), current models use statistical methods like the Weibull probability distribution function (Grussing et al. 2006) and the Discrete Markov process (Grussing et al. 2016) to fit a continuous function to asset data and make condition predictions as a function of age. These approaches focus on population life-cycle expectations to make future probabilistic life-cycle predictions of individual assets. The standard industry practice of viewing assets in terms of service-life ranges or life cycles (Hodges 1999) results in large prediction ranges, thus labeling the performance of individual assets from year to year a stochastic phenomenon (Grant et al. 2014). A holistic data-driven approach could instead be applied to predict asset-specific conditions throughout its life instead of just focusing on an end-of-life expectation for the population overall. The BUILDER SMS assessment process discussed above records the condition of individual assets in quantitative form as a Condition Index (CI) score (Uzarski 1995), allowing an asset's behavior over time to be tracked. This quantitative

indexing of asset conditions allows decision makers to manage asset portfolios in a prioritized fashion (Sitzabee and Harnly 2013).

Coupling the understanding of degradation factors from climate and material factors with this time-based condition data reveals system-level trends. For example, interior systems are sheltered from the weather, but exterior systems such as roof and wall assemblies are not. As a result, roofing and wall systems experience quicker degradation as a function of both time and weather than interior systems (Grant et al. 2016). This acute difference in weather exposure suggests that these systems are much more at risk to weather, and their predictive degradation does show this in terms of shorter life cycles. While average roof systems have an expected life cycle of 20-40 years, wall systems have an expected life cycle that is typically 2-3 times longer (Grant et al. 2016). It can be inferred that roof systems are much less resilient than wall systems since they degrade three times as fast (Hodges 1999). However, climatic variations can drastically affect these tendencies.

This same method of indexing asset-specific conditions over time can be coupled with other attribute data, such as asset age, from the asset management database to develop a precise, data-driven stepwise method. By extracting groups of assets with similar performance behaviors at times of inspection, the degradation characteristics of those groups can be used to make future condition predictions. Leveraging this comprehensive data set as a tool to improve both short and long-term prediction models enable better management decisions, reduces the risk of premature asset failures and financially crippling expenses (GAO 2011).

Area of Contribution

Despite the significant contributions of the aforementioned studies and EAM systems, current asset prediction methods still only produce broad life-cycle expectations from population data instead of asset-specific condition expectations. Industry tends to view asset condition prediction from an end-of-life perspective, which is meant to inform replacement planning. However, this leaves large gaps in understanding an asset's performance over its lifespan, which translates to poor maintenance and repair management planning.

For this reason, research into different predictive model types and their strengths and weaknesses is imperative to provide managers skillful predictions at all points along the asset life cycle. New data-driven forecast types can be developed to fill this gap. New model types using stepwise methods will be created and compared to conventional models as ways to convert asset data into asset-calibrated degradation predictions. The methods for creating each of the model types will be explained, and the prediction strengths of each type will be discussed along with insights on how to employ them singularly or as ensemble tools for making management decisions.

This research will use data and methods to understand asset-specific degradation rates of several roof types due to broad variances in climate and material by analyzing asset groupings that behave similarly. However, this research will not uncover nuanced manufacturing-specific vulnerabilities due to individual weather factors. Due to current research gaps, leaving a factor-based analysis of degradation to adopt a broader understanding of overall asset degradation rates or slopes between condition assessments

is a necessary alternative. This type of slope-based understanding supports a stepwise approach to forecasting future conditions that employ the past condition history and behaviors of temporally similar assets to forecast degradation expectations.

As an illustration of each model approach's efficacy, this research uses Air Force roof data from 61 unique US locations. The stepwise degradation models compete with the state-of-the-art degradation model used by BUILDER SMS (Uzarski et al. 2019) to determine whether and which model approaches offer improvements in degradation prediction. Roofing systems were selected over other assets because the average expected life cycle is 20-30 years, as opposed to other building systems, which have an expected life cycle that is typically 2-3 times longer (Grant et al. 2016). Selecting assets with a shorter life cycle requires a smaller temporal data range for calibration and validation. Given that BUILDER condition data has only been collected for 11 years, results for longer-lived assets would be speculative. Stated alternatively, the sheer number of facilities operated by the Air Force means that the number of roofs tracked across various segments of their life cycle will provide a statistically significant sample with which to perform this analysis.

III. Scholarly Article 1: Improving Data-driven Infrastructure Degradation Forecast Skill With Stepwise Asset Condition Prediction Models

Abstract

Organizations with large facility and infrastructure portfolios have used asset management databases for over ten years to collect and standardize asset condition data. Decision makers use these data to predict asset degradation and expected service life, enabling prioritized maintenance, repair, and renovation actions that reduce asset life-cycle costs and achieve organizational objectives. However, these asset condition forecasts are calculated using standardized, self-correcting distribution models that rely on poorly-fit, continuous functions. This research presents four stepwise asset condition forecast models that utilize historical asset inspection data to improve prediction accuracy: (1) Slope, (2) Weighted Slope, (3) Condition-intelligent Weighted Slope, and (4) Nearest Neighbor. Model performance was evaluated against BUILDER SMS, the industry-standard asset management database, using data for five roof types on 8,549 facilities across 61 U.S. military bases within the United States. The stepwise Weighted Slope model more accurately predicted asset degradation 92% of the time, as compared to the industry standard's continuous self-correcting prediction model. These results suggest that using historical condition data, alongside or in-place of manufacturer expected service life, may increase the accuracy of degradation and failure prediction models. Additionally, as data quantity increases over time, the models presented are expected to improve prediction skills. The resulting improvements in forecasting enable decision makers to manage facility assets more proactively and achieve better returns on facility investments.

Introduction

Asset Management is the method by which facility managers plan and care for built infrastructure and facilities. Both public and private entities are responsible for managing asset portfolios over their life cycle. This is a challenging task, especially for large agencies, like universities, hospitals, supply-chain companies, and municipalities. Ultimately, all organizations with built infrastructure portfolios face the same asset management problem (Vanier 2001), with America's infrastructure rated a D+ (ASCE 2017).

Whether accounted for in facility conditions or dollars, deferred maintenance is growing in attention because it has been growing in deferment in the US since the 1930s (Stupak 2018). For example, the DoD was authorized $26.7 billion in fiscal year 2020 to construct, repair, alter, maintain, and modernize its 585,000 facilities and associated infrastructure (DoD Comptroller 2019). Despite this considerable funding that results from the DoD's annual budget of 1.2% of these assets' replacement value (DoD Real Property Portfolio Office 2018), there remains an estimated $116 billion maintenance project backlog (Moon-Cronk 2018). Unfortunately, the DOD is not an anomaly when it comes to foregone maintenance (Stupak 2018).

Asset management requires the creation of a comprehensive infrastructure inventory, which makes prioritizing essential repairs and replacement projects, in addition to planning a long-term capital budget, efficient for policymakers and asset owners (ASCE 2020). Since the condition of assets is not static, plans must be routinely updated to ensure asset strategies and management decisions are in sync with

degradation. Current degradation models suggest that infrastructure assets age with time, or age-based obsolescence (Grussing 2014), but several distinct factors cause degradation. Degradation directly results from exogenous influences acting on infrastructure or assets, and roofing systems are among the most exposed built assets. Research shows that heat aging, roof traffic, roof slope, and annual maintenance (Hodges 1999) are significant degradation factors in addition to extreme weather events (Karl et al. 2013), such as hail damage or heat stress from high-temperature extremes and solar radiance (Harkness and Hassanain 2001).

Additionally, time appears to influence roof service life (Cash 2006). While the correlation between roofing types and specific degradation factors is being drawn, the research gap is still quite broad when trying to use these factors to predict roofing degradation. For this reason, life-cycle analysis is typically the most impactful justification to support roofing research and project decisions (Grant et al. 2016). However, when an analysis of five service-life software was conducted, the variation of predictions for the service life of three different roof systems (Built-up, Thermoplastic or Single-ply, and Vegetated) within these models was extreme (Grant et al. 2014). The tension between using broad life-cycle predictions and factor-specific degradation models leads current research to employ data gathered by asset management databases.

Asset management methodologies have been in place for decades for several infrastructure domains, including roads and pavements, railroads, bridges, and distribution pipelines (Grussing and Liu 2014). Over the past ten years, industry leaders have also begun to use Enterprise Asset Management (EAM) systems to collect and

standardize asset condition data across their diverse portfolio of facility assets, such as roofs (Grussing 2014). Two of these systems are the BUILDER Sustainment Management System (SMS) (Grussing and Liu 2014) and BELCAM (Lounis et al. 1999). While the software differs somewhat technically, both concept models start with population trends and adjust those trends using observed condition inspection data. This approach results in shaping or scaling population expectations instead of a tailored prediction for assets with a similar inspection history. Decision makers use these systems' data to predict asset degradation and expected service life, enabling prioritized maintenance, repair, and renovation actions to reduce asset life-cycle costs and achieve organizational objectives.

This data-driven approach to asset management has been increasing in popularity, and it is also growing in use as a management tool as the amount of data collected for facilities and infrastructure continues to grow. Converting these existing data sets into prediction models to forecast future asset conditions requires overcoming quantity, quality, and management decision threshold hurdles. In contrast to early Gompertz, Richard, or Morgan-Mercer-Flodin models (Sjostrom 2004), current models use statistical methods like the Weibull probability distribution function (Grussing et al. 2006) to fit a continuous function to asset data and make condition predictions as a function of age or the Discrete Markov process (Grussing et al. 2016) to predict the probability of a component being in a particular condition state at a particular time step. These approaches focus on population life-cycle expectations to make future probabilistic life-cycle predictions of individual assets. The standard industry practice of viewing assets in terms of service-life ranges or life cycles (Hodges 1999) results in large

prediction ranges, thus labeling the performance of individual assets from year to year a stochastic phenomenon (Grant et al. 2014).

A holistic data-driven approach could instead be applied to predict asset-specific conditions throughout its life instead of just focusing on an end-of-life expectation for the population overall. The BUILDER SMS assessment process discussed above records the condition of individual assets in quantitative form as a Condition Index (CI) score (Uzarski 1995), enabling asset performance to be tracked over time. This quantitative indexing of asset conditions equips decision makers to manage asset portfolios in a prioritized fashion (Sitzabee and Harnly 2013). This same method of indexing asset-specific conditions over time can be coupled with other attribute data in the asset management database to develop a precise, data-driven stepwise method by extracting groups of assets with similar performance behaviors at times of inspection and using the characteristics of those groups to make future condition predictions. Leveraging this comprehensive data set as a tool to improve both short and long-term prediction models enable better management decisions, reduces the risk of premature asset failures and financially crippling expenses (GAO 2011).

Despite the significant contributions of the aforementioned studies and SMS, current asset prediction methods still only produce broad life-cycle expectations from population data or failure probabilities instead of asset-specific condition expectations. Industry tends to view asset condition prediction from an end-of-life perspective, which is meant to inform replacement planning. However, this leaves large gaps in understanding an asset's performance over its lifespan, which translates to poor

maintenance and repair management planning. For this reason, research into different predictive model types and their strengths and weaknesses is imperative to provide managers skillful predictions at all points along the asset life cycle. Data-driven forecasts can be developed to fill this gap. Four new model types will be discussed and compared as ways to convert asset data into degradation predictions using (1) Slope, (2) Weighted Slope, (3) Condition-intelligent Weighted Slope, & (4) Nearest Neighbor approaches. The methods for creating each of the model types will be explained, and the prediction strengths of each type will be discussed along with insights on how to employ them singularly or as ensemble tools for making management decisions.

As an illustration of each model approach's efficacy, this research uses Air Force roof data from 61 unique US locations. Both model prediction values and BUILDER SMS prediction values (Uzarski et al. 2019) are compared with observation data to quantify degradation prediction improvements for each model. Roofing systems were selected over other assets because the average expected life cycle is 20-30 years, as opposed to other building systems, which have an expected life cycle that is typically 2-3 times longer (Grant et al. 2016). Selecting assets with a shorter life cycle requires a smaller temporal data range for calibration and validation. Given that BUILDER data has only been collected for 11 years, results for longer-lived assets would be speculative. Stated alternatively, the sheer number of facilities operated by the Air Force means that the number of roofs tracked across various segments of their life cycle will provide a statistically significant sample with which to perform this analysis.

Data and Case Study

BUILDER SMS inspection data was gathered from 61 unique, contiguous United States Air Force (USAF) installations and used in this analysis to provide a representative sample of variations in manufacturer, climate, maintenance, and other conditions present across the enterprise. The data includes time-based Condition Index (CI) records for assets installed between 1985 - July 2020. Roof system data was selected for this case study because roofing subtypes have a range of service-life expectancies between 20-50 years, which helps prove this research's applicability to assets of differing service-life expectancies. Roofing (B30) data were collected utilizing BUILDER SMS reports titled AF QC 06, which give a comprehensive catalog of assets down to the system sub-type level (Charette and Marshall 1999). At the system sub-type level, an individual asset has multiple unique inspections over its service life. These inspection values are used to filter the data for quality purposes before employing the data.

Data Quality: SMS data quality and quantity must first be assessed to create a tailored model. While the USAF employs standardized maintenance plans, routine inspections, and uniform condition metrics, data quality and consistency vary across locations based on the subjectivity of technician ratings of the assets and projects that improve an asset’s condition. This is why stringent pre-processing and filtering of the data has been employed. Additionally, data quantity increases with the number of locations included in the study and as inspections occur over time. The USAF data provides a unique opportunity to maximize both the quantity and uniform quality of asset data simultaneously.

Filtering Hierarchy

The data was filtered to remove all roof subtypes other than Built-Up Roofing (BUR), Modified Bitumen Roofing (MOD), Single-Ply (SP), Shingle (SH), and Standing Seam Metal (SSM) roof-types. The roofing service life of these five roofing types are known to be different, so they were selected for comparison. All other roofing types were not analyzed in this study.

Cleaning Hierarchy

The data cleaning process employed is listed below and quantified in Table 1.

1. Remove repaired assets: If the Observed Condition Index (OCI) of the asset improved from one inspection to the next (OCI2 – OCI1 ≥ +1), this asset was removed. Note: Component Section Condition Index (CSCI) was used, but for simplified communication, these values will be referred to as "CI" in this paper.

2. Remove replaced assets: If the original construction date changed from one rating period to the next, this asset was removed.

3. If an asset had less than a perfect score (100 = CI) at age zero, this asset was removed because assets not in perfect condition when installed contain install defects.

4. If an asset had a score of zero (0=CI), the asset was removed.

5. Component Specifics

    a. The data fields retained for analysis of the assets are Unique Asset Identifier, System Sub-type, Asset Install/Construction Year, Asset Age at time of Inspection, Year of Inspection, and Condition at Inspection.

    b. Roofs: 870 Built-Up Roofing (BUR), 461 Modified Bitumen Roofing (MOD), 525 Single-Ply (SP), 476 Shingle (SH), and 1179 Standing Seam Metal (SSM) roof-types were selected as the components for comparison. The roofing service life of these five roofing types are known to be different, so they have been analyzed separately. All other roofing types were not analyzed in this study.

Table 1. Data Description

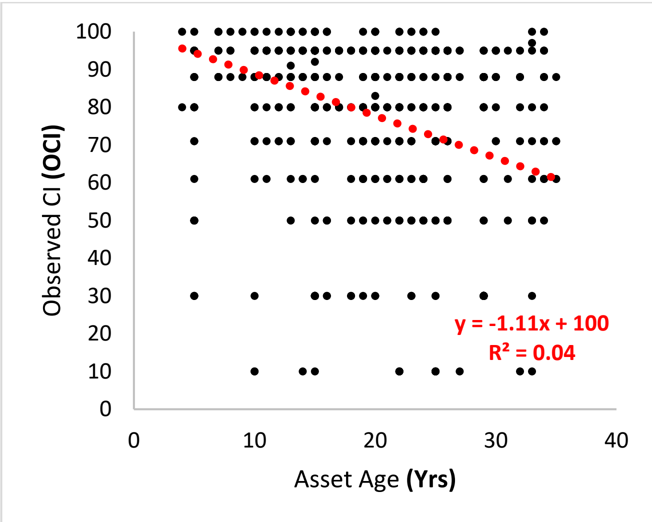
	Initial QC-06 Inspections	Initial Unique QC-06 Assets	Filtered QC-06 Inspections	Filtered Unique QC-06 Assets	Cleaned QC-06 Inspections	Cleaned Unique QC-06 Assets	% of Final Inspections (Retained / Original)	% of Final Assets (Retained / Original)
All 61 Bases	166,163	80,696	90,076	40,327	18,817	8,549	11%	11%

Note: At the end of cleaning, 11% of the original data remains. This approach ensures data used to predict service lives only captures assets without improvements, resulting in natural degradation data. Specific location data is included in the appendix.

Development of new variables/data explained: Initial analysis of the data revealed that minor post-processing is required to utilize the data for model building purposes. These data processing steps are described below.

Calculating Age: The temporal scale provided was converted from a relative date to an absolute asset age, allowing assets of the same age with different installation dates to be compared.

Delta Condition Versus Condition: Raw data from BUILDER is captured in OCI, while BUILDER predictions are assigned an Expected CI (ECI). The calculation for BUILDER SMS ECI values is an output of the age-dependent Weibull function it employs. When looking at the correlation between age and OCI (Figure 2a), the R^2 value is very low, suggesting that they are not inherently related. However, an analysis of the residuals or Delta Condition Index (DCI), calculated by subtracting Expected CI (ECI) from the Observed CI (OCI), reveals a strong relationship in the data. Although the signal is strong, plotting the age versus DCI (Figure 2b) shows that the data range is widely spread across the possible outcomes. Notice the increase in the R^2 values between the two figures, although the spread, or range, of the data remains at around 100 CI points in both figures.



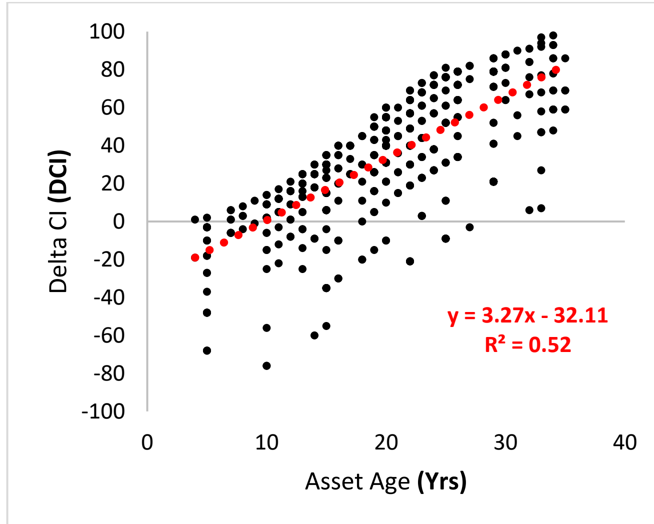


Figure 2a: Age versus Observed Condition Index (OCI)

Figure 3b: Age versus Delta Condition Index (DCI)

Limitations: The filtering process ultimately reduced data quantity while maximizing data quality remains the same. Only 11% of the original data was retained. This is likely because as assets age, they are more likely to have a repair, replacement, or maintenance action, which ultimately removed those assets from inclusion in the analysis. The quantity trade-off is one that should increase confidence in the results of this research. However, as the number of data subsets that are used increases, each subset's size decreases.

For this reason, more data is always more powerful and will produce different results. While this research's methods are applicable to multiple data samples, the results and discussion are applicable to this specific sample only. Another limitation of the data is that USAF BUILDER guidance requires each asset be inspected at least once every five years, although more frequent inspections are encouraged. Inspection intervals, inspector, and other intangible factors vary across the assets. Additionally,

while older assets are required to have more inspections, many assets have annual or semi-annual inspections performed for warranty purposes. The frequency of inspections ultimately results in differing data resolution between assets. While this research aims to synthesize these differences by increasing data quantity, these differences were not studied in depth.

Methodology

An iterative, data-driven methodology resulted in the production of four asset degradation prediction models. The following methodology will explain the models that build from the most simplistic to the most rigorous. There are several reasons to develop multiple models instead of relying on a singular model. First, researchers should seek to create the least complicated tool that provides the level of service necessary to make the decisions they want. In this case, predictions need to be accurate throughout the asset's life cycle to make better maintenance and repair decisions. Secondly, the iterative approach creates models that could be useful for other data and assets. Even if a particular model is not useful in this study, alternative conditions could prove the model more useful. Finally, the creation of more than one model allows for trade-offs and ensembles, which often provide better results than a single model can achieve on its own. Ultimately, more than one model can be coupled to provide the best results. The iterative methodology presented below provides a robust use of the data to satisfy both short and long-term decision needs.

There are several commonalities between the model types, such as initial Search space and stepwise computation. Search space constraints limit the initial data

that the model explores to obtain input variables before applying mathematical computation, and it can be categorized by Age (x) or Condition (y). Model types are developed using different initial search spaces and mathematical treatment of the data once selected as an input variable (Table 2). Stepwise computation is used to convert discrete condition and age outcomes into a complete model by selectively interpolating data based on groups of similar assets. This process is different from fitting a continuous function to a data set because the focus of stepwise computation is incrementally slope-based, which results in the data and model being much closer aligned. All model iterations employ stepwise computation and analysis of the case study data.

Table 2: Model Overview

Model	Search Space	*Description*
Slope (SM)	Condition (y)	*Created by using age-specific (stepwise) average sample slope to predict a 1-year forecast.*
Weighted Slope (W-SM)	Condition (y)	*Created by using age-specific (stepwise) proximity-weighted 4-year average slope matrix to predict 1-year forecast.*
Condition-intelligent Weighted Slope (CI-W-SM)	Condition (y)	*Created using age-specific (stepwise) proximity-weighted 4-year average slope & condition-bound matrix to predict 1-year forecast.*
Nearest Neighbor (KNN)	Age (x)	*Created by an expanding age search to fill sample quota (K), then predicts a 1-year forecast.*

The Model Overview shows the search space, input variable, and general description of the mathematical operation(s) applied to convert the input data into a prediction value.

1. **Slope Model (SM)**

   Methods: The first-generation model is the Slope Model (SM). The prediction at any age (x) is the median value of all asset inspections (OCI) at that discrete time step.

2. Weighted-slope Model (W-SM)

Methods: The second-generation model is the Weighted-slope Model (W-SM), which focuses on individual asset performance over a four-year period instead of performance at a single discrete age. The W-SM uses a four-year, forward-looking search of the data set to calculate a weighted average ECI for a single target asset at age (t) to predict the next year's (CI_{t+1}) condition as shown in Equation (1 and Equation (2.

$$W\text{-}SM_{Prediction} = CI_{t+1} = CI_t - \frac{\sum_{t=0}^{T} w_t \times \bar{y}_t}{t} \tag{1}$$

$$w_t = \frac{\mathrm{T} - (\mathrm{t} - 1)}{T!} \tag{2}$$

Where:

CI_{t+1} = the Expected Condition Index (ECI) produced by the model for the next year;

CI_t = the Observed Condition Index (OCI) of the asset in question at its last inspection;

T = the total number of years past the current inspection;

t = the out-year index between zero and T;

w_t = the proximity weighted value assigned to each out-year, where the weight assigned is greater than or equal to zero, decreases as the out-year increases, and all weight values sum to one; and

$\bar{y}_t$ = the average change in condition of assets from each out-year.

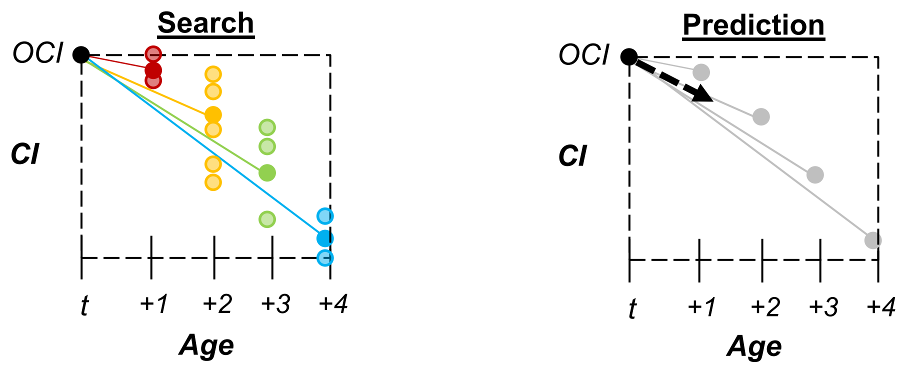


Figure 3: The search graphic shows the 4-year search space and age (t) input variable, with colors representing asset groups. The prediction graphic shows the consolidation of each out-year average into a 1-year prediction value/vector.

The model is a proximal weighted average of the collective assets' condition averages at $t + n$ years past the observed condition of the asset in question. For example, if $n = 4$, which is used in the research, the search space is 4 years past the inspection of an asset at t. Weight values are $w_1 = .4$, $w_1 = .3$, $w_1 = .2$, & $w_1 = .1$ respectively for out-years 1, 2, 3, & 4. So, any asset at age = t is expected to degrade in condition at the same rate, or slope, as the model at age = t (Figure 3).

3. **Condition-intelligent Weighted-slope Model (CI-W-SM)**

Methods: The third-generation model is the Condition-intelligent Weighted-slope Model (CI-W-SM), which adds condition thresholds to the W-SM and constrains asset selection to a condition performance category. This fine-tunes the model focus on assets with similar performance paths to make better predictions. This improvement allows the model to filter out assets performing

better or worse than the asset in question, thus producing a more accurate degradation prediction as long as sufficient data is available. Within BUILDER SMS, performance categories of Good (100-81 Green), Repair (80-61 Amber), and Replace (60-0 Red) are used as general guides for managers. Here, BUILDER's categories are used to subset the data before calculating an expected condition value. Decision makers should set these performance category thresholds to target their maintenance, repair, and replacement actions appropriately.

The CI-W-SM uses the same four-year proximity search of the data set to calculate a weighted average ECI (y) for an asset at age (x) to predict the next year's condition, as shown in Equation (3.

$$W\text{-}SM_{Prediction} = CI_{t+1} = CI_t - \frac{\sum_{t=0}^{T} w_t \times \bar{y}_t}{t} \tag{3}$$

Where:

$\forall y_t$ = the current condition of all asset at each timestep;

CI_{t+1} = the Expected Condition Index (ECI) produced by the model for the next year;

CI_t = the Observed Condition Index (OCI) of the asset in question at its last inspection; n is the number of years past the current inspection

t = the age of the asset(s)

w_n = the weighted value assigned to each out-year

$\bar{y}_{G_{t+n}}$ = the average condition of *Good* category assets from each out-year

$\bar{y}_{R_{t+n}}$ = the average condition of *Repair* category assets from each out-year

$\bar{y}_{Rp_{t+n}}$ = the average condition of *Replace* category assets from each out-year.

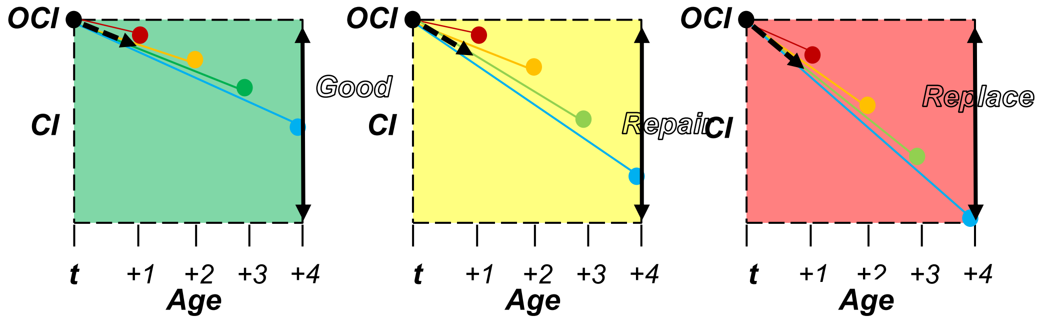


Figure 4: CI-W-SM Search Graphic

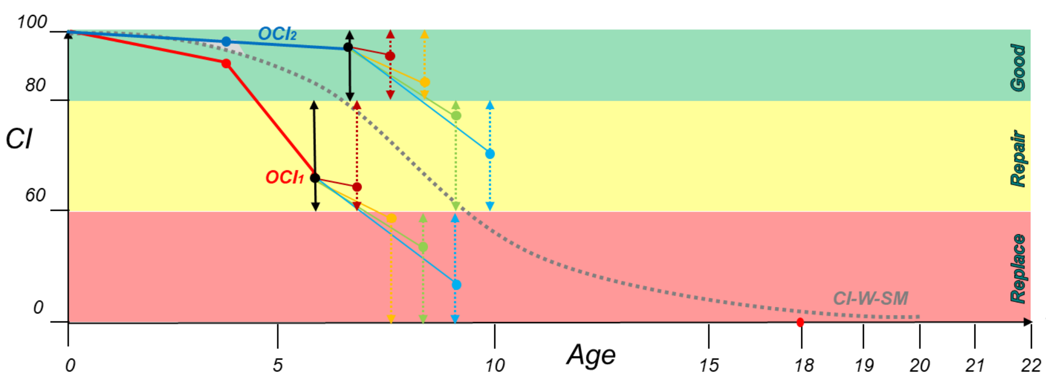


Figure 5: The CI-W-SM plot shows the 4-year search space and age (t) input variable for each of the (3) separate performance categories (Good, Repair, & Replace), while the dashed arrow shows the consolidation into a 1-year prediction value/vector.

As shown in Figure 4 and Figure 5, each of the bins has its own weighted-slope values for each age index. Finally, when predicting the target asset's forecast value, the model first checks the last inspected OCI (CI_t) before using the corresponding bin(s) to make a 1-year prediction. Additionally, when bootstrapping consecutive predictions past 1-year, the model adjusts the bins it uses for prediction to match the condition of the asset in question. So, when the CI-W-SM model makes a prediction that crosses the Good/Repair

condition boundary of 81-80 CI, it stops using data in the "Good" bin and uses data from the "Repair" bin to run the next year's prediction calculations.

4. **Nearest Neighbor Model (KNN)**

Methods: The fourth model uses a Nearest Neighbor (KNN) approach. This model differs from the others as it employs a radiating search space for neighboring assets starting at the target asset age (t), as shown in Figure 6 and Equation (4. The search radiates outward by $\pm p$ year increments until it fills a minimum asset quota (K). Once K is satisfied, each asset's condition slope is calculated; this slope represents the change in condition between the time at which the asset is retained, and its next assessment. The average of the K condition slopes is averaged to make a $t + n$ prediction for the asset in question. The radiating search is unnecessary if the number of assets with condition assessments at age (t) is greater than or equal to K.

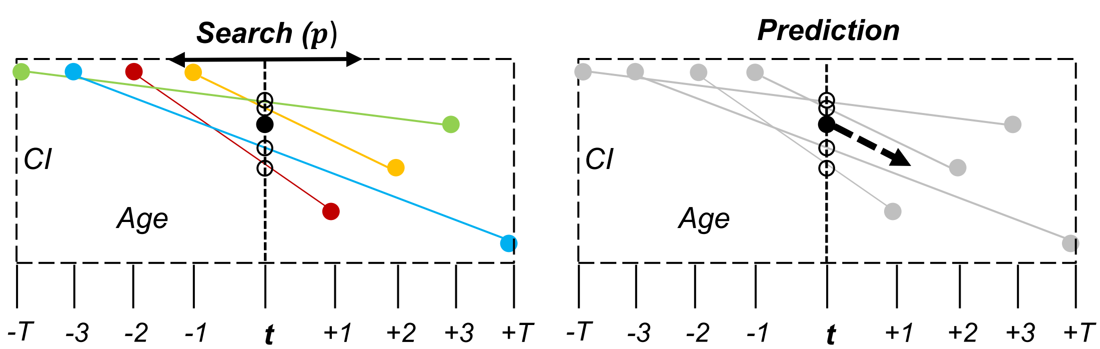


Figure 6: The KNN graphic shows the age ($\boldsymbol{t}$) input variable and radiating search space value ($\boldsymbol{p}$) required to fill the minimum asset quota ($\boldsymbol{K}$), while the dashed arrow shows the consolidation into a 1-year prediction value/vector. Colors are used to represent individual asset inspections.

Furthermore, in this case, all assets with condition assessments are used to make a prediction, as not to limit the model's data unnecessarily. This model assigns equal weight to all assets included in the search quota so that assets further away from the target asset in age are not penalized for their age difference. *Note: This model's outcomes vary based on the size set for minimum asset quota (k) because this directly changes the minimum size of the sample required to make predictions. A K value of 6 is used in this paper because it achieves satisfactory results when validated against known inspection data.*

$$kNN_{Prediction} = C_{t+1} = \frac{\left[\sum_{k=1}^{K}\left[\frac{C_{k,(t-p)}-C_{k,(t+q)}}{q-p}\right]\right]}{K} ; \forall q \neq p, p \geq t \tag{4}$$

Where:

CI_{t+1} = the Expected Condition Index (ECI) produced by the model for the next year;

$C_{k,(t-p)}$ = the first inspection condition (OCI) of each asset filling the quota (K);

$C_{k,(t+q)}$ = the second inspection condition (OCI) of each asset filling the quota (K);

q = the number of years past the current inspection;

t = the age of the asset in question;

p = the number of years before t;

K = the minimum number of assets in the quota; and

k = each asset in the quota.

5. Nearest Neighbor Model (KNN)

A framework is developed to compete the models using DCI as the validation metric. Simply put, DCI is the difference between the observed and forecast

values. In this framework, a "win" is categorized by the model with the lowest DCI for an individual age within the service life so that the quantity of possible wins between the models is equal to the service life predicted by the W-SM. The individual results for the five researched roof system types are reported as well as a collective performance value for each model. The model value shows the overall win percentage for the model across all roof types.

Results

The four models are discussed individually in this section. Then, model validation will be addressed collectively at the end of this section to show how the models compare to BUILDER SMS and each other.

1. **Slope Model (SM)**

   Results: While BUILDER data directly drive this modeling approach, the simplified single-year median produces ECI values that occasionally increase between predictions. This means that as the population data increases in age, it does not always decrease in condition, which causes large variations in the data distribution between years. Although an increase in average condition between asset ages is an accurate depiction of the data when taking single-year population medians, individual assets cannot behave this way because assets that increased between inspections were removed during data filtering. A non-positivity constraint has been employed to combat the average condition increases between years. Unfortunately, after using the non-positivity constraint for this model, the degradation plateaus significantly due to the

number of data points removed. As discussed in the data section, age is not highly correlated with condition the data. This model re-illustrates the limitations of directly correlating age and condition.

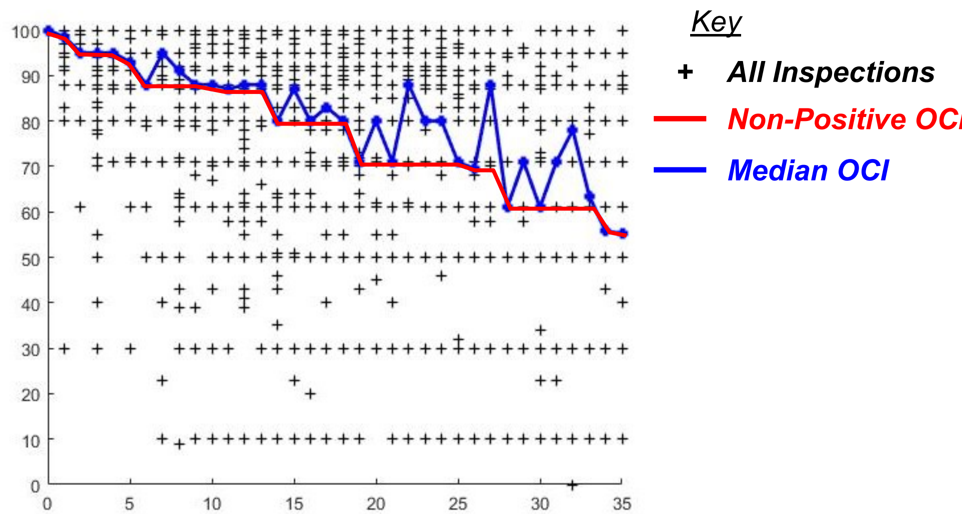


Figure 7: Slope Model plot shows the median observed condition for BUR asset inspections at each discrete age. If used to predict an asset's future condition, this model requires a non-positivity constraint to eliminate erroneous improvements.

2. **Weighted-slope Model (W-SM)**

Results: The four-year proximity-weighted averaging eliminates ECI value increases between predictions. The model only uses the data of assets that have inspections at the same age as the target asset and have an additional inspection at 1, 2, 3, or 4 years immediately after. In order to use the model values to predict future values of individual assets at different initial inspection conditions, the slope values are extracted from the weighted condition values

by taking the difference of expected values and indexed by age. The plotted result of this model is shown in Figure 8. Validation of this model is included at the end of this section.

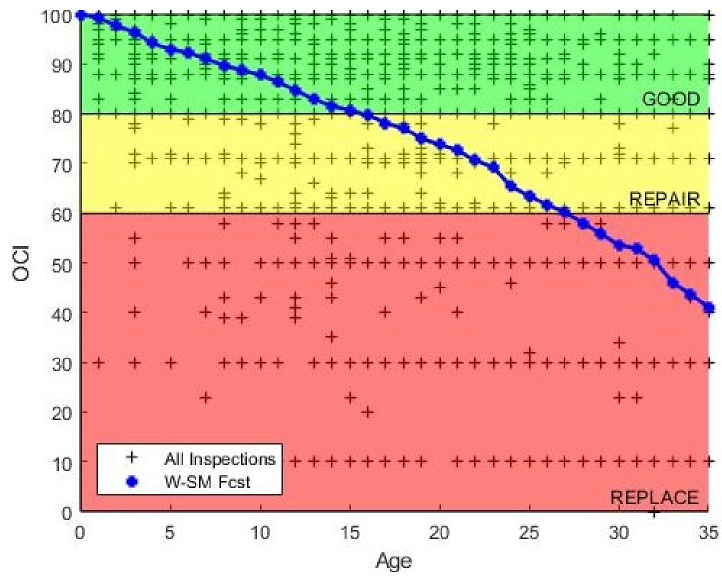


Figure 8: W-SM projection for a brand-new asphalt shingle roof. Critical decision points occur as prediction approaches condition thresholds. The forecast expects an average SH roof to last between 21-22 years; however, individual asset performance will vary. The discrete model slopes are indexed to each age and are unique to this data set.

3. **Condition-intelligent Weighted-slope Model (CI-W-SM)**

   Results: The four-year proximity-weighted averaging, like the W-SM, eliminates ECI value increases between predictions but only uses the data of assets that pass through both the same age and condition category of the target asset. Because of this, the model becomes more optimistic, as it ignores assets outside the target asset's condition bin (Green = Good, Amber = Repair, and Red = Replace). As discussed in the Data section, the categorical subdivision

of the data reduces the number of assets in each prediction sample. While this approach should produce more realistic predictions, it does reduce the statistical significance of each prediction by reducing the sample size used to make the prediction. In years where there is not enough data to compute a prediction, this results in a prediction slope of zero, or no change from the previous year.

This model requires the highest data quantity, and data quantity must be sustained across the entire life cycle of the asset. In this specific data set, metal roofing (SSM) had the highest quantity of data and the longest life cycle, which produced the least no-change predictions (Figure 9). Single-ply membrane (SP) roofing had the second-lowest quantity of data and a significantly shorter life-cycle expectation, which resulted in the most no-change predictions (Figure 10). These results suggest that data quantity is imperative for making service-life predictions using this model. Validation of this model is included at the end of this section.

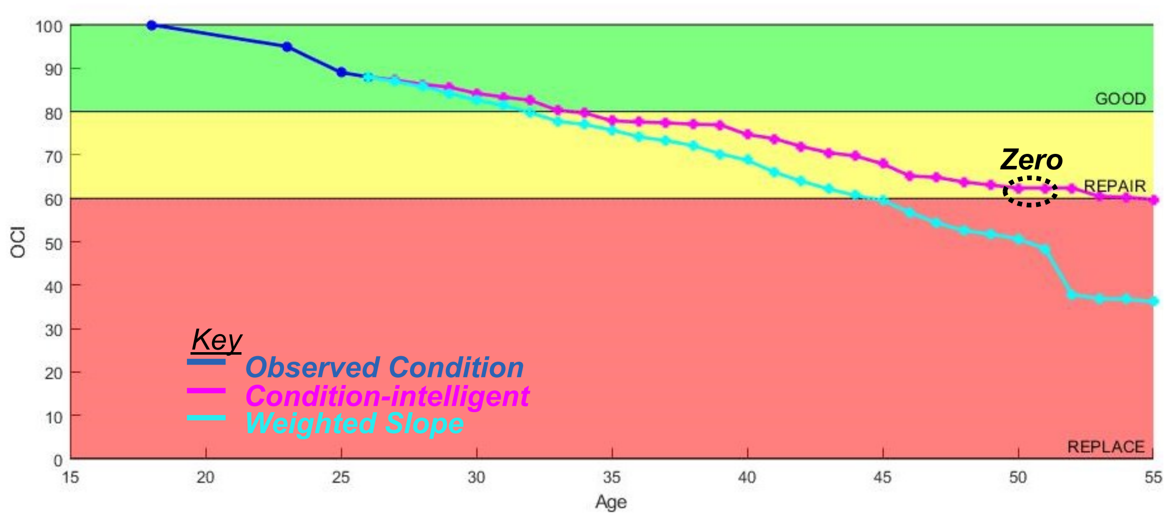


Figure 9: The plot shows the service-life condition forecast of a single SSM roof asset using both the W-SM and the CI-W-SM. The CI-W-SM has several timesteps, including the time interval between 51-52yrs, where the slope appears to be zero. This zero-slope outcome results from insufficient data quantity to make a prediction when only using the assets in the repair bin with an inspection recorded at both age 52 and another inspection at age 53, 54, 55, or 56 as required by the methodology for this model.

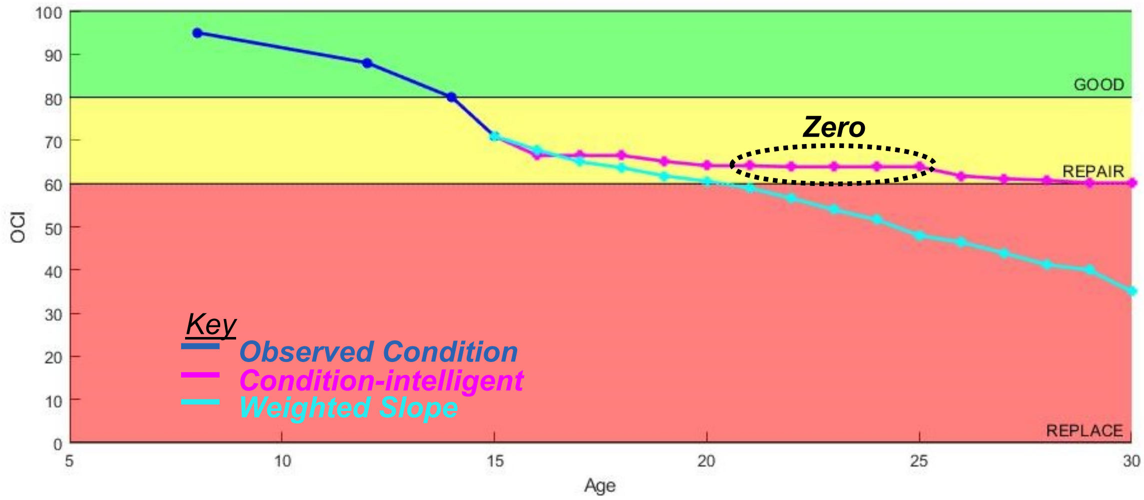


Figure 10: The plot shows the service-life-condition forecast of a single SP roof asset using both the W-SM and the CI-W-SM. The CI-W-SM has several timesteps, including the time interval between 21 and 25 years, where the slope appears to be zero. This zero-slope outcome results from insufficient data quantity to predict when only using the assets in the repair bin. The plot shows how the lack of data quantity can result in erroneous over-projections of service life.

4. **Nearest Neighbor Model (KNN)**

Results: This model makes highly skillful 1-year lead predictions (Figure 11). Notably, almost all 1-year prediction values produced by this model are within five CI points or less of the actual condition, which is very good. One example of the 1-year prediction accuracy is shown in Figure 11, where the model predicts the value of the last recorded inspection with zero error (both points are on top of one another). In order to make long-term predictions of service life using this model, bootstrapping of the data is required. However, when bootstrapping is used, it quickly results in a compounded underprediction of the assets' actual condition. The most likely reason for this is that assets with

catastrophic failures (or rapid degradation) are increasingly more likely as assets age. Since this model uses a varying number of years instead of a four-year average to make predictions, these rapidly failing assets have the potential to account for a significant weight in the average depending on the quota (k) size selected.

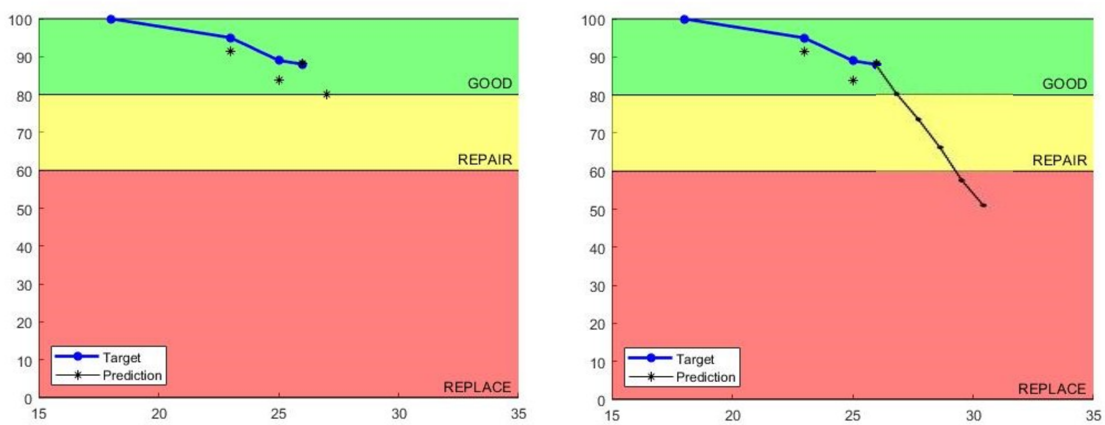


Figure 11: The plot on the left shows the Observed Condition Index (OCI) compared to the KNN model forecast for the same inspection year. Additionally, there is a 1-year forecast at the end of the inspection data to project the asset's condition one year later. The plot on the right shows consecutive out-year inspections utilizing bootstrapping to make predictions, which deteriorates quicker than is reasonably expected.

Additionally, this model requires a balance of the trade-off between increasing the quota (k) size and limiting the search radius. A small (k) means that rapidly failing assets can easily result in pessimistic predictions. While increasing (k) means the search area will likely increase, making the predictions more optimistic. This research has found a (k) value of six

provides accurate 1-year predictions, but longer-term predictions from this model result in low-accuracy. For this reason, validation beyond 1-year from the last recorded inspection was not completed for this model, foregoing comparison to BUILDER SMS.

5. Model Validation

Three of the models ultimately competed against the predictions of BUILDER SMS. The Slope Model (SM), Weighted-slope Model (W-SM), and the Condition-intelligent Weighted-slope Model (CI-W-SM) are reported because they all show strong graphical performance when initially plotted. The prediction plots for each roof type produced results consisitent with industry service-life. The results of these three models and BUILDER SMS predictions are compared to observed asset conditions to validate prediction skill. When comparing the SM results to the W-SM results, it becomes apparent that both have similar win percentages and similar shapes, but their y-intercepts vary. Ultimately, the rapid deterioration predictions that resulted from bootstrapping with the Nearest Neighbor (KNN) model made it unbeneficial for long-term service-life comparison.

The service life of each roof system type is determined by the number of years the W-SM outcomes remain in the Good/Repair bins. While manufacturers guarantee specific performance ranges for roofing products and systems, these data-driven results show the actual average service-life ranges for each system installed at the 61 Air Force locations in this study. Initial

validation of the W-SM shows that service-life predictions for the five roof types researched are similar to that of manufacturer specifications, suggesting validation of the W-SM as a service-life forecast method. The CI-W-SM is not used to determine service-life values for the roof types because all roof types contain years without data, which is a direct result of insufficient data quantity due to the additional subdivision of the data, as discussed in the results section.

The DCI validation metric discussed in the Methods section of this paper is used as the framework to compete the models. In this framework, a “win” is categorized by the model with the lowest DCI (Figure 12, Figure 13, and Figure 14.) for an individual age within the service-life range. The lower the DCI value, the better the model is at predicting observed conditions. The individual results for the five researched roof system types are reported in Table 3, as well as a collective model performance value. Model values capture the overall win percentage for that model across all roof types. The W-SM outperforms the BUILDER SMS prediction an average of 92% of the time. The CI-W-SM beat the BUILDER SMS prediction an average of 69% of the time. For BUR, the W-SM resulted in an R^2 value of 0.38, while the CI-W-SM produced an R^2 value of 0.39, and the BUILDER SMS R^2 value is 0.06. Additionally, the root mean square error (RMSE) values for the W-SM, CI-W-SM, and BUILDER SMS are 32.81, 39.26, and 49.83, respectively. Individual results for the five roof sub-types are shown in Table 3.

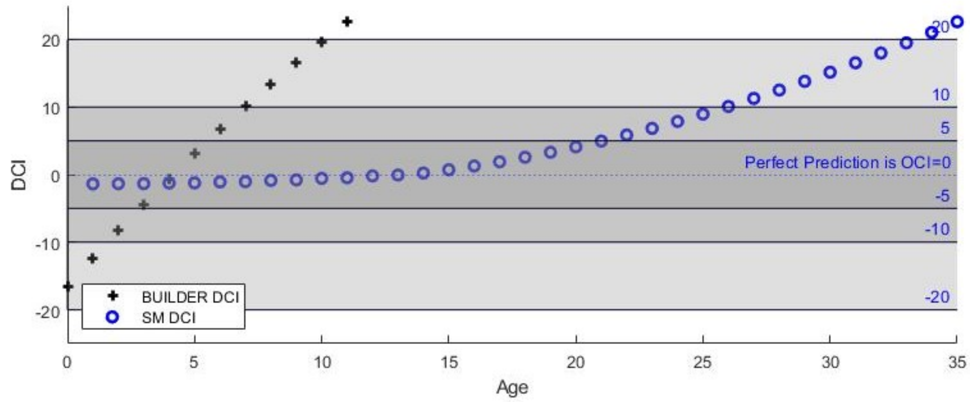


Figure 12: DCI plot for SM using BUR data.

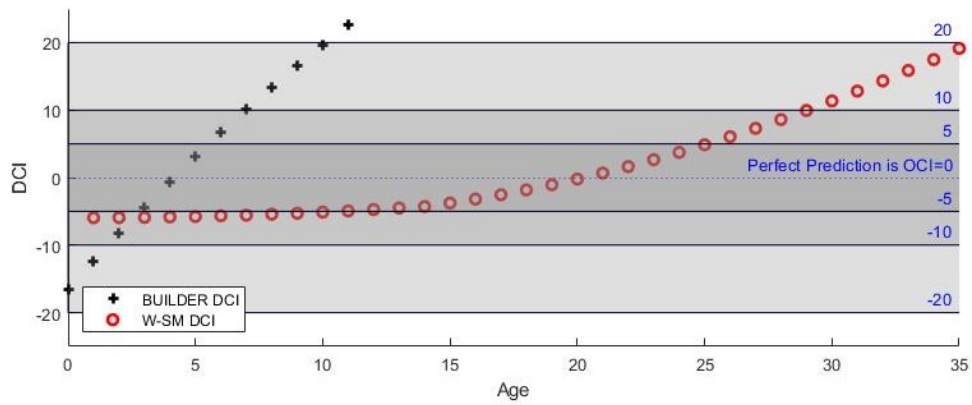


Figure 13: DCI plot for W-SM using BUR data.

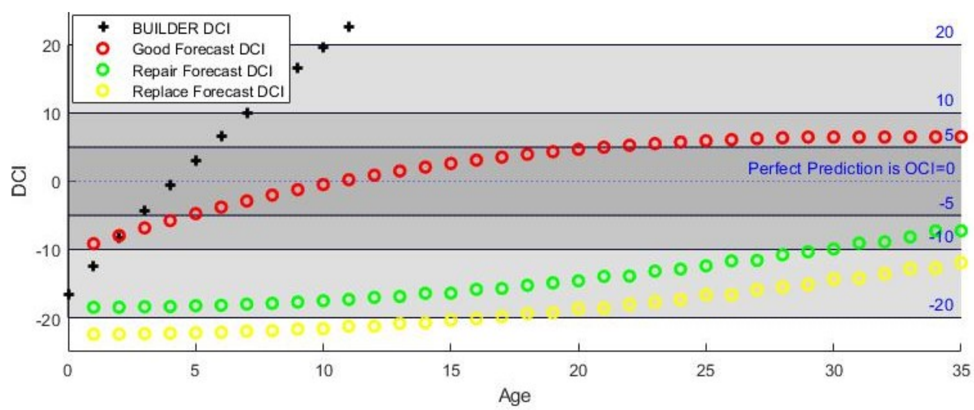


Figure 14: DCI plot for CI-W-SM using BUR data.

Table 3: Competition Outcomes

Metric	SH	BUR	SSM	MOD	SP	Total
	Wins by Roof System Type					
BUILDER	4	1	1	6	5	17
Slope Model (SM)	17	26	37	21	19	120
Service Life (Yrs)	21	27	38	27	24	137
% Win	81%	96%	97%	78%	79%	**88%**
BUILDER	1	1	5	3	1	11
Weighted-slope Model (W-SM)	20	26	33	24	23	126
Service Life (Yrs)	21	27	38	27	24	137
% Win	95%	96%	87%	89%	96%	**92%**
BUILDER	8	4	20	4	7	43
CI-W-SM: Good	13	23	5	23	17	81
CI-W-SM: Repair	0	0	13	0	0	13
CI-W-SM: Replace	0	0	0	0	0	0
Service Life (Yrs)	21	27	38	27	24	137
% Win	62%	85%	47%	85%	71%	**69%**

Discussion

Stepwise data-driven modeling techniques can be used to calibrate degradation forecasts based on observed conditions and improve the correlation between asset age and condition. As asset data continues to grow in quantity, the results of these models are likely to change. A discussion of the models and their response to increased inspection data quantity over time is detailed below to explain the models in more depth, as summarized in the model wrap-up shown in Table 4.

Table 4: Model Wrap-Up

	Model Wrap-Up				
	Model	Search Space	Pros	Cons	Description
1	Slope (SM)	Conditon (y)	Data Driven Time-step "best fit" Boot-strapping	Erroneous positive slopes Not asset specific	*Created by using age-specific (step-wise) average sample slope to predict 1yr forecast.*
2	Weighted Slope (W-SM)	Condition (y)	Data Driven Time-Step "Fit" Asset Specific Negative Slope Long-term Pessimistic	Uses Assets in other conditions	*Created by using age-specific (step-wise) Proximity-weighted 4yr average slope matrix to predict 1yr forecast.*
3	Condition Intelligent Weighted Slope (CI-W-SM)	Conditon (y)	Data Driven Time-Step "Fit" Asset Specific Negative Slope Long-term Optimistic	Requires more data	*Created using age-specific (step-wise) Proximity-weighted 4yr average slope & Condition-bound matrix to predict 1yr fcst.*
4	Nearest Neighbor (kNN)	Age (x)	Requires less data Short-tem Forecst (+1yr)	No Boot-strapping Long-term Inacurate	*Created by an expanding age search to fill sample quota (K), then predicts 1yr forecast.*

1. **Slope Model (SM)**

Thus far, the simplicity of the SM has been discussed as a shortfall. However, as data quantity grows, this model type's performance should improve due to the central limit theorem. Since the number of inspections at each age will increase with time, the median values across each discrete age step should assimilate a natural degradation between years as a result. The poor direct correlation of age and condition suggests that positive outcome variability is likely. This means the non-positivity constraint may be required even as data quantity increases over time. While using a non-positivity constraint forces the model to degrade over time, the artificial plateaus leave much room for improvement. Although this model will likely improve with time, the amount of time this will take and the magnitude of improvement is unknown.

2. **Weighted-slope Model (W-SM)**

While the W-SM approach eliminates positive changes in condition, it can be improved because prediction calculations incorporate asset data regardless of their condition relative to the target asset's condition. As discussed in the

results, this produces a pessimistic prediction that under-values top-performing assets in the good and repair condition categories. Over time, this theoretically results in a series of slightly pessimistic predictions compared to the actual life cycle of well-performing assets. However, because of the quantity of inspection data currently available, this model outperforms all others. As inspection history increases, this model will likely move into second place behind the CI-W-SM because it utilizes prediction bins.

3. **Condition-intelligent Weighted-slope Model (CI-W-SM)**

Since the CI-W-SM employs an additional level of condition filtering before selecting assets to make predictions, it is expected to make the most accurate service-life predictions of all the models presented. However, current inspection data history is only 10-15 years for most assets in the inventory. This makes covering the entire service life of an asset with the data quantity required for these predictions harder to achieve. As data history grows, the quantity of inspection data will also increase, which will aid the CI-W-SM model in achieving the data thresholds necessary to make predictions covering an asset's entire service life.

4. **Nearest Neighbor (KNN)**

The KNN model lacks the long-term prediction capability of the W-SM and the CI-W-SM, but it has a strong prediction capability for short-term forecasting. The trade-off of this model may provide significant benefits for decision makers who are more nearsighted, and this model has an additional level of variability due to the quota size used to make predictions. The k-

variable provides quantified asset prediction minimums and an alternate search space. Instead of limiting its asset search space by condition, as is done in the CI-W-SM, the KNN model limits search space by age before selecting assets used to make predictions. Quantity of asset inspection data again plays a role in the performance of this model over time. As inspection quantity increases, the KNN model will not need to look as far to the left or right of the inspection year to fill the quota minimum. This means that the data used in prediction calculations should gravitate towards the year immediately following the last recorded inspection. Assuming this theoretical prediction is accurate, this tool's condition prediction will begin to assimilate the SM prediction because the quota will increasingly be satisfied by assets from a single-year average that approaches the same value calculated by the SM.

Decision Making: The four models discussed in this research demonstrate that while some methodologies are beneficial for short-term predictions, those same models may not be skilled at predicting an asset's service life. For this reason, the models created have been categorized into short-term or long-term categories based on their unique skill. Short-term models are those that make skilled near-future predictions, such as the KNN model discussed in this paper. The KNN model makes strong 1-year forecasts, but it lacks the skill to make predictions further into the future. This type of model helps analyze assets close to decision points, deciding whether an asset will likely need a repair or replacement project in the coming year, or whether it will remain relatively stable. Long-term models

are characterized by their skill in forecasting the service-life degradation of assets, which may be years or decades away. While current service-life models traditionally blanket-apply singular population averages to every asset uniformly, the other three models discussed in this paper show that a stepwise, data-driven approach is more accurate than continuous statistical functions because stepwise methods look at rates of degradation instead of targeting a single service-life age. This makes the SM, W-SM, and CI-W-SM great planning tools for enterprise-wide asset management efforts like those frequently drawn from BUILDER SMS. As asset management progresses, aligning model types to decision-maker priorities should be as much a focus of the industry as building accurate degradation forecast models.

Ensembles: In reality, decision makers typically exist at all levels of agencies, and their priorities vary based on their level of authority. For example, an enterprise-level decision maker may set corporate budgets for facilities maintenance and repair, while a program manager may hold the responsibility for selecting individual projects and assets to utilize funds as they become available. These differences make it difficult to justify the use of a single forecast model. This is why understanding the goal of decision makers should inform the types of models used to analyze data. While it may be more complex, combining each model's benefits into an ensemble may be more informative and skillful for making holistic asset predictions. This type of approach may be able to inform and satisfy both types of decision makers simultaneously, and the effort of

combining stepwise models becomes quite easy to automate through the use of index values like is done in the CI-W-SM for assets when they transition from one condition bin to another. The versatility of stepwise modeling to include the aggregation of multiple models via indexing is another potential advantage of the proposed framework.

Conclusion

Although asset management methodologies have been in place for decades, the methodologies used for employing asset management data to predict future conditions are still evolving as new data become available. Existing prediction models produce broad life-cycle expectations from population averages instead of data-driven, asset-specific condition expectations. This research employed roofing data from 61 unique US Air Force locations to show that stepwise methodologies can be superior to the industry-leading continuous methodologies employed by BUILDER SMS in service-life prediction accuracy and decision-making versatility as ensembles. Notably, the data used to train the models created was also used to test them; however, the stepwise fashion employed by the models does exclude future conditions of target assets when making predictions. This means that bias in these models should be minimal if at all present. These methodologies should be employed using an alternate data set to validate and compare the results.

Future research is suggested to refine and validate the findings of this case study. Roofing systems were analyzed in this research due to their variety of service-life durations. While the data used for this research is specific to roofing assets, using the same methodologies to analyze all BUILDER SMS assets is the broader intent. For this reason, it is recommended that other asset types such as exterior wall systems, mechanical equipment, structural elements, and other facility component types be analyzed using the proposed stepwise approach to study the assumptions and any necessary adjustments for other asset types. While results are anticipated to be similar, the K-value in the KNN model and the number of years (n) used for proximity weighting of the W-SM and the CI-W-SM are relatively new, and further research and statistical analysis of these variables may offer opportunities for optimization as future improvement opportunities. The concept of the Delta Condition Index (DCI) provides a consistent metric for comparing future model results in a uniform metric, and additional research into statistically fitting the DCI of each asset subtype as a continuous function may provide breakthroughs into rapid improvement to the current BUILDER SMS degradation formula.

Nevertheless, the asset management industry must move away from just focusing on when a component will fail and consider the strategic points throughout a component's life, when targeted maintenance or repair may be beneficial. Moving towards more intelligent stepwise models is one way to increase the understanding of an asset's middle-life. This transition will also

enable decision makers at operational levels to make stronger predictions of short-term asset performance, thus capitalizing on right-time planning for asset-specific repair and replacement projects. The four new models discussed in this research can be used as short-term, long-term, or ensemble forecast tools that elevate the prediction power of asset managers of all levels even as data quantity expands. While individual models may be best suited for some decision makers, ensembles that employ the indexing power of the stepwise methodologies developed in this research are likely to provide the most comprehensive asset overview yet published.

IV. Scholarly Article 2: Data-Driven Asset Condition Models: An Air Force Roof System Case Study

Summary

The Department of Defense (DoD) employs its ever-growing repository of facility condition data to predict service lives and plan maintenance, repair, and renovation actions. These BUILDER forecasts are founded on manufacturer expectations. Research conducted at the Air Force Institute of Technology (AFIT) supports stepwise asset condition forecast models as a superior alternative.

Background

Facility managers use asset management principles to plan and care for built infrastructure and facilities. Public and private entities are similarly responsible for managing asset portfolios throughout their life cycle. This can be a monumental task, especially for large organizations such as universities, hospitals, and municipalities. Whether accounted for in facility conditions or dollars, maintenance deferment in the US has been growing in since the 1930s, and this risk is gaining attention. The DoD was authorized $26.7 billion in fiscal year 2020 to construct, sustain, restore, and modernize its 585,000 facilities and infrastructure. While annual DoD funds are budgeted at 1.2% of the replacement value for these assets, an estimated $116 billion maintenance project backlog remains. Inevitably, all agencies with facility portfolios face the same asset management problem, degrading infrastructure, with America's infrastructure currently rated a D+. Asset management databases have been created to close this gap.

Asset Management Databases

Asset management techniques were first used to manage infrastructure such as roads and pavements, railroads, bridges, and distribution pipelines. Over the las decade though, industry has begun to harvest and catalog facility condition data in Enterprise Asset Management (EAM) systems. The creation of a comprehensive infrastructure inventory enables policymakers and asset owners to efficiently prioritize projects and plan long-term capital budgets. A variety of decision makers use this data to predict asset degradation, expected service life, reduce life-cycle costs and achieve organizational objectives. Unlike original asset management techniques, EAM requires software, initial asset inventory, and ongoing condition input.

BUILDER SMS and BELCAM are two examples of EAM software. Although the software have different technical approaches, they both use time-based condition inspection data to modify population service-life expectations, but this results in a simple scaling of a prediction curve instead of a tailored condition prediction based on assets with similar historical behavior. Current degradation models suggest that infrastructure assets age with time, but several exogenous factors cause degradation, including weather and maintenance, some of which are stochastic. Due to degradation, the condition of assets is constantly changing, and databases must be updated on a routine basis to maintain accurate asset strategies and management decisions.

Data-Driven Predictions

Data-driven approaches to EAM have been increasing in popularity, and they are also increasingly being adopted for use as a management tool. Using data as a fuel to

power condition prediction models requires overcoming quantity and quality hurdles. Current models use statistical functions like the Weibull distribution to capture population trends and make condition predictions as a function of age. Using life-cycle expectations of a population to make condition predictions of individual assets results in large prediction ranges since individual assets can behave quite differently than a population average. Viewing individual assets in terms of average population service-life ranges or life cycles may be the industry standard, but it results in seeing the performance of an individual asset as stochastic. This creates large gaps in understanding an asset's performance over its lifespan, which translates to weaker facility sustainment, restoration, and modernization (FSRM) management planning. Data-driven forecasts can be developed to fill this gap.

Enhancing Prediction Skill

A holistic EAM approach should instead use data-driven models to predict individual conditions of an asset throughout its service life instead of treating end-of-life expectations as the foundation for all predictions. EAM programs were created to track individual asset conditions over time in quantitative form as a Condition Index (CI) score. Decision makers currently use this quantitative index to prioritize asset portfolios in an individualized fashion. However, the condition history of assets over time can be combined with additional EAM data to create a data-driven stepwise method for identifying groups of assets that degrade at similar rates between inspections. Once these groups are identified, these groups can be used to make future condition predictions. Using EAM data in this fashion leverages data quality and quantity as a tool to develop

both short and long-term prediction models for a variety of decision maker objectives. Since objectives vary, developing multiple model types with different prediction strengths and weaknesses is the only way to provide managers skillful predictions at all points throughout an asset life cycle.

Roof Degradation ***(this section is not included in publication due to length limitations)***

Degradation is a direct result of exogenous influences acting on infrastructure or assets, and roofing systems are among the most exposed built assets. Research shows that heat aging, roof traffic, roof slope, and annual maintenance are significant degradation factors in addition to extreme weather events, such as heat stress from high-temperature extremes and solar radiance, or hail damage. Additionally, time appears to influence roof service life. While the correlation between roofing types and specific degradation factors is being drawn, the research gap is still quite broad when trying to use these factors to predict roofing degradation. For this reason, life-cycle analysis is typically the backbone justification to support roofing research and project decisions. However, when an analysis of five service-life software was conducted, the variation of predictions for the service life of three different roof systems (Built-up, Thermoplastic or Single-ply, and Vegetated) within these models was extreme. The tension between using broad life-cycle predictions and factor-specific degradation models leads current research to employ data gathered by asset management databases.

Data and Case Study

Research conducted at the Air Force Institute of Technology suggests stepwise forecasts may outperform current models. Three models were developed and tested using Air Force roof data for five roof types from 61 unique US locations. Each model was then compared with the state-of-the-art degradation model used by BUILDER SMS to determine how each model approach improves degradation predictions. Roofing systems were selected over other assets because their shorter average expected life cycle of 20-30 years is best covered by the data. However, the methods these models employ to convert data into predictions can be used for assets of all BUILDER System types.

Model Methods

There are several shared characteristics between the model types. Search space is a constraint that limits the population data that the model searches through to obtain input variables before applying mathematical computation, and it can be categorized by either Age (x) or Condition (y). Different initial search spaces and mathematical computation are used to create the different model types. The stepwise computation is incrementally slope-based, which results in unique asset groupings being used to compute predictions at every time step. Discrete condition and age outcomes are then translated into a complete model by using stepwise computation and intelligent interpolation of predictions. The three new models are the Slope (SM), Weighted Slope (W-SM), and Condition-intelligent Weighted Slope (CI-W-SM). A deeper dive into the model methodology is available in the publication titled *Improving data-driven infrastructure degradation forecast skill with step-wise asset condition prediction models*.

Results

Validation of the models is discussed to show how they compare to both BUILDER SMS and one another. The Delta Condition Index (DCI) is a validation metric that captures the difference between the observed and forecast values. All model predictions are compared to observed conditions, and a "win" is awarded to the model if its DCI is lower than that of the BUILDER SMS prediction. The quantity of possible wins between the model and BUILDER SMS is equal to the service life. Since there were five roof types in this case study, the individual results for each roof system type are reported in Table 5 as well as a collective model performance value, which is the overall "win" percentage for the model across all roof types. The W-SM outperforms the BUILDER SMS prediction an average of 92% of the time, while the CI-W-SM beat the BUILDER SMS prediction an average of 69% of the time. For BUR, the W-SM and CI-W-SM accounted for over six-times as much variation in outcomes as BUILDER SMS. These results show the stepwise models produced can outperform current predictions.

Conclusions

While current service-life models use population averages to make predictions, the three models discussed in this paper illustrate the magnitude of improvement possible by stepwise, data-driven model predictions. Expecting all assets to approximate the Weibull curve or population service-life is less accurate than stepwise methods that look at similar asset behaviors. The SM, W-SM, and CI-W-SM are better tools for translating BUILDER SMS data into enterprise-wide asset management plans. As EAM systems

progress, taking time to align or create model types that target decision-maker priorities, as shown in Table 6 and Figure 15, should be an industry priority.

In reality, data-driven predictions are only as good as the data they employ. For this, huge thanks are due to the folks who develop EAM databases and those who collect inventory and assessment data for the enterprise. The quality of information they enter into the EAM database, whether good or bad, is the foundation on which all forecasts must rely. The DoD has devoted itself to EAM to steward resources, and improving short and long-term forecast skill is one fruit of that labor.

Table 5. Competition Outcome: Results show the win percentage for each model when compared to BUILDER SMS.

	Metric	Wins by Roof System Type: SH	BUR	SSM	MOD	SP	Total
Slope Model (SM)	BUILDER	4	1	1	6	5	**17**
	Model	17	26	37	21	19	**120**
	Service Life	21	27	38	27	24	**137**
	% Win	***81%***	***96%***	***97%***	***78%***	***79%***	***88%***
Weighted Slope (W-SM)	BUILDER	1	1	5	3	1	**11**
	Model	20	26	33	24	23	**126**
	Service Life	21	27	38	27	24	**137**
	% Win	***95%***	***96%***	***87%***	***89%***	***96%***	***92%***
Condition-intelligent (CI-W-SM)	BUILDER	8	4	20	4	7	**43**
	Model - Good	13	23	5	23	17	**81**
	Model - Repair	0	0	13	0	0	**13**
	Model - Replace	0	0	0	0	0	**0**
	Service Life	21	27	38	27	24	**137**
	% Win	***62%***	***85%***	***47%***	***85%***	***71%***	***69%***

Table 6. Model Wrap-up: shows the search space, input variable, pros/cons, and general description of the mathematical operation(s) used to convert input data into a prediction value.

	Model Wrap-Up: Model	Search Space	Pros	Cons	Description
1	Slope (SM)	Conditon (y)	Data Driven Time-step "best fit" Boot-strapping	Erroneous positive slopes Not asset specific	*Created by using age-specific (step-wise) average sample slope to predict 1yr forecast.*
2	Weighted Slope (W-SM)	Condition (y)	Data Driven Time-Step "Fit" Asset Specific Negative Slope Long-term Pessimistic	Uses Assets in other conditions	*Created by using age-specific (step-wise) Proximity-weighted 4yr average slope matrix to predict 1yr forecast.*
3	Condition Intelligent Weighted Slope (CI-W-SM)	Conditon (y)	Data Driven Time-Step "Fit" Asset Specific Negative Slope Long-term Optimistic	Requires more data	*Created using age-specific (step-wise) Proximity-weighted 4yr average slope & Condition-bound matrix to predict 1yr fcst.*

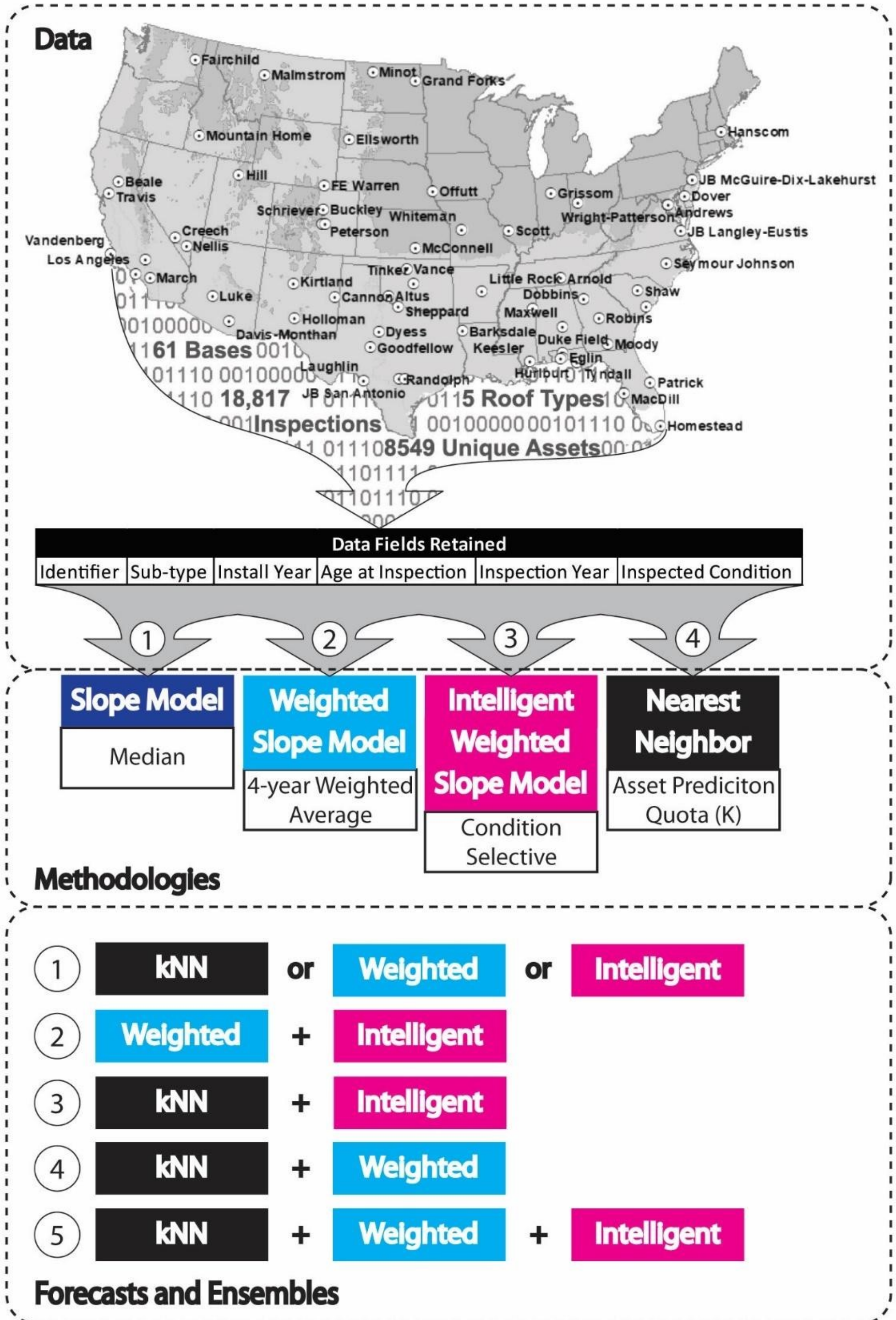


Figure 15. Summary Infographic: The figure shows the transformation process starting with data at 61 base locations, the development of (4) four iterative methodologies, and the (5) five ways that decision makers can use the models as single-forecasts or combined ensemble tools.

V. Conclusions and Recommendations

Research Conclusions

This thesis focused on creating state-of-the-art, data-driven stepwise methodologies that predict future facility asset conditions and creating stepwise asset degradation models. Three research objectives were explored to support modeling and forecast development:

1. Conduct a comprehensive review of literature surrounding the body of knowledge in three areas: roofing degradation factors, asset degradation models, and methods used for forecasting and data projections.
2. A case study of roofing data is used to investigate and utilize data to develop these stepwise models. United States Air Force (USAF) B30 Roofing System data from BUILDER SMS is used to analyze the real-world performance of five roofing types at 61 geographically unique base locations.
3. Decision making and how it relates to forecast model use are discussed in this paper to better relate these models to practical application and field use. Singular and ensemble combinations are explored to fully understand the benefits of multiple models with varying skill.

First, the comprehensive literature review covering roofing degradation factors, asset degradation models, and methods for producing forecasts and data projects was completed in Chapter 2. In the chapter, roofing degradation factors are discussed in terms of material properties and weather factors. The discovery that

degradation conclusively occurs at different rates for different roofing material types allows us to rely on service life to understand projections, but more data and more research are required to understand how different roofing system type behaviors respond to individual weather factors. This guides research to a data-driven forecasting approach that employs data being gathered by existing EAM systems, like BUILDER SMS. This data is currently being used to create statistical models that fit functions based on population trends, which often vary from individual asset behaviors. Several existing modeling types are discussed, and stepwise methods are introduced to better employ this data because they could be used to select assets with similar behaviors as a target asset to make more accurate predictions.

The second and third research objectives were addressed in an in-depth technical fashion in Chapter 3, "Scholarly Article 1: Data-Driven Asset Degradation Modeling: An Enterprise-wide Roof System Case Study." This journal article presented the use of USAF BUILDER SMS roofing data from 8,549 unique facilities to develop four novel stepwise modeling methodologies: (1) Slope, (2) Weighted Slope, (3) Condition-intelligent Weighted Slope, and (4) Nearest Neighbor. Each stepwise model is explained in detail with equations, figures, and rationale so that it can be comprehensively understood. The models are developed in an iterative fashion, which helps understand how the models relate to one another even though they employ unique methods to translate data into future condition predictions. The discussion section of the paper provides an

in-depth explanation of each model's strengths and weaknesses in terms of short-term and long-term prediction skill.

Additionally, a section of this paper is entirely devoted to discussing the use of these models by different decision makers. These decision makers' objectives are used to determine which model or models should be used to more accurately meet organizational and program goals. The goal is to publish this paper during calendar year 2021 in the Elsevier's Journal of Building Engineering, an international, peer-reviewed publication with a 2019 impact factor of 3.379.

The second and third research objectives are again addressed in a more conceptual fashion that focuses on their field-use and resulting benefits instead of technical attributes in Chapter 4, "Scholarly Article 2: Data-Driven Asset Condition Models: An Air Force Roof System Case Study." This journal article presents a general non-technical understanding and application of data-driven models in plain English format for a DoD-wide audience. Concepts of the research and how they apply to specific government and military objectives and initiatives are discussed while emphasizing the role and value of individuals who interface with the BUILDER SMS database. DoD-wide payback is highlighted in this paper to enlighten the audience of the magnitude and contribution to the organization as a whole. The target journal for this paper is The Military Engineer (TME), a well-known DoD and industry partnered journal published by the Society of American Military Engineers (SAME), which is circulated to

government employees, active-duty members, contractors, and firms that focus on architecture, engineering and construction (A/E/C) sectors of work.

Research Contributions

The primary research contributions of this thesis include the development of:

1. Four data-driven, stepwise forecasting methodologies that utilized existing BUILDER SMS data gathered by the DoD. In this paper's results, three of these models are shown to outperform the current industry-standard model used by BUILDER SMS.
2. A novel metric, Delta Condition Index (DCI), that is used to quantitatively compare and evaluate competing models on equal grounds. Methods are developed to use this metric to evaluate the skill of individual asset predictions and population aggregations for investigating performance at short-term and long-term scales.
3. Ensemble forecasts that combine the benefits of different models to make better-informed decisions and target tradeoffs between short and long-term skill.
4. Decision-maker informed forecast model selections. With the creation of more than one forecast model, a discussion can begin to fit prediction models to decision maker’s unique objectives.

Research Impact

The aforementioned research contributions are expected to significantly impact current facilities sustainment, restoration, and modernization (FSRM) planning and budgeting practices. This thesis is the first effort to improve facility asset condition forecasts by developing four stepwise condition prediction methods. These models have the potential to enhance the accuracy of short and long-term facilities asset management practices and reduce variation between plans and reality. Decision makers at different levels of ownership in the asset management chain will be empowered by forecast models that more accurately depict varying out-year prediction lead times that match their organizational objectives. This thesis has laid the groundwork for follow-on research efforts to improve BUILDER SMS forecasts that are currently underway at the US Army's Construction Engineering Research Lab (CERL). Furthermore, this thesis culminated in the development of two publishable journal papers, one presentation for SAME's Kittyhawk Post, and one poster exhibition at AFCEC's 2020 virtual Design and Construction Symposium. This research has undoubtedly enhanced the academic and military community's awareness and knowledge of the present subject matter.

Recommendations for Future Research

Current condition prediction models make life-cycle expectations that are founded on population averages instead of asset-specific condition information. Alternatively, the case study conducted as part of this research employed roofing data from 61 unique US Air Force locations to show that stepwise methodologies can be superior to the industry-leading continuous methodologies employed by BUILDER SMS both in service-life

prediction accuracy and short-term decision-making versatility. Notably, the data used to develop the four models created was also used to test them. If continuous statistical models were developed this way, bias would occur due to the overfitting of models to the sample data set. However, the models' stepwise fashion does exclude future conditions of any target asset when making predictions, which means that bias in the developed models should be minimal if at all present. To test this theory, it is recommended that these methodologies be employed using an alternate data set to validate and compare the findings. Data from the U.S. Army or U.S. Navy BUILDER database is recommended for this research due to formatting and low-likelihood of introducing unforeseen biases. Further testing of the methods employed in this research will provide more conclusive findings and eliminate bias if present.

Additional future research is suggested to refine and validate the findings of this case study. Only roofing system data were analyzed in this research due to their variety of service-life durations. While the data used for this research is specific to roofing assets, the same methodologies can be used to analyze and predict future conditions for all BUILDER SMS assets. For this reason, it is recommended that assets from other BUILDER SMS categories such as exterior wall systems, mechanical equipment, and structural elements be analyzed using the proposed stepwise approach to study the assumptions and any necessary adjustments to variables for predicting condition values of other asset types. Some model-specific improvements are suggested. For the SM, a non-positivity constraint has been used to remove positive improvements, but a multi-year running average slope calculation could also correct this condition. For the CI-W-SM, interpolation or use of a previous slope value instead of a zero-slope forecast could

improve the model predictions. Furthermore, the condition bins/bands used for categorizing the CI-W-SM assets have the potential for optimization to better account for decision maker preferences. While results should theoretically be similar, the K-value in the KNN model and the number of years (n) used for proximity weighting of the W-SM and the CI-W-SM are relatively new, and more research that includes statistical analysis of these variables may offer opportunities to optimize these model components as improvements. The concept of the Delta Condition Index (DCI) provides a common metric for comparing future model results in a single uniform metric for individual assets or population service lives. Additional research into statistically fitting the DCI of each asset subtype as a continuous function may provide low-cost breakthroughs and rapid improvement that can be applied to the current BUILDER SMS degradation formula. Lastly, a factor-driven condition prediction model that relates weather data with asset condition degradation could be incredibly beneficial, providing more insight into the individual root causes and magnitude of degradation by explaining correlations. In lieu of a complete factor analysis, location-based weighting modifiers could be developed and applied to models as a way to account for varying rates of degradation that occur due to differences of weather unique to specific climate zones.

The most significant suggestion for future research is to shift the asset management industry's focus from when a component is going to fail and to focus instead on the opportunities throughout an asset’s life cycle that are most beneficial for repair and maintenance actions. Using more intelligent stepwise models is one way to increase this understanding of an asset’s “middle-life.” This transition will enable decision makers at operational levels to make better predictions of short-term asset performance, thus

capitalizing on right-time actions for asset-specific repair and replacement projects. Additionally, strategic-level planning initiatives will improve based on aggregated effects of increased accuracy in individual asset predictions. The four novel models discussed in this research can be used as short-term, long-term, and ensemble forecast tools that increase the prediction skill of asset managers of all levels even as data quantity expands. These models will likely improve in accuracy as the data they use to make predictions increases in quantity. While individual models may be best suited for some decision makers, ensembles that employ the indexing power of the stepwise methodologies developed in this research are likely to provide the most comprehensive facility asset condition prediction overview yet published.

Appendix A

Detailed Data Description:

		Initial QC-06 Lines	Unique Assets in Initial QC-06	Filtering*	Filtered QC-06 Lines	Unique Assets in Filtered QC-06	Cleaning**	After QC-06 Lines	Unique Assets in After QC-06	Statistics	% of Original Report Lines Retained	% of Original Unique Assets Retained
	Grand Forks	618	342		530	266		118	58		19%	17%
	Minot	2,070	716		1,910	625		547	230		26%	32%
	Mt Home	1,148	675		582	347		86	42		7%	6%
	USAF Acadey	5,903	4,258		1,299	641		194	95		3%	2%
	Buckley	909	496		467	258		46	22		5%	4%
	Cannon	1,633	722		748	357		86	41		5%	6%
	Eielson	2,030	1,012		977	419		201	94		10%	9%
	Fairchild	1,814	782		1,235	482		295	135		16%	17%
	FE Warren	2,403	938		1,021	403		248	114		10%	12%
	Hill	5,312	4,286		1,545	1,171		288	143		5%	3%
	Kirtland	2,646	1,483		1,289	775		424	195		16%	13%
	Malmstrom	1,513	877		585	344		101	49		7%	6%
	Peterson	1,776	1,043		1,424	851		483	227		27%	22%
	Schriever	952	462		434	209		143	61		15%	13%
	Freeze/Dry Tota	**30,727**	**18,092**		**14,046**	**7,148**		**3,260**	**1,506**		**11%**	**8%**
B30 - Roofing	WPAFB	5,115	3,585	Filtering*	2,108	1,374	Cleaning**	446	211	Statistics	9%	6%
	McGuire	5,685	3,749		2,820	1,820		882	430		16%	11%
	Altus	604	251		565	215		91	39		15%	16%
	JB Anacostia-B	526	526		311	311		-	-		0%	0%
	Andrews	2,516	1,225		1,233	620		202	92		8%	8%
	Arnold	1,369	878		780	494		66	33		5%	4%
	Dover	2,495	879		1,802	562		175	73		7%	8%
	JB Elmendorf-R	2,382	1,087		2,167	980		132	64		6%	6%
	Hanscom	2,186	533		1,111	270		109	47		5%	9%
	JB Langley-Eus	4,468	2,700		2,212	1,342		390	176		9%	7%
	Little Rock	1,435	957		710	479		139	68		10%	7%
	McConnell	1,168	471		654	275		185	86		16%	18%
	Offutt	1,179	644		797	453		263	125		22%	19%
	Scott	1,864	1,011		813	426		169	78		9%	8%
	Tinker	4,799	1,754		4,410	1,401		741	331		15%	19%
	Vance	587	533		230	195		36	18		6%	3%
	Whiteman	2,860	1,193		1,154	506		159	73		6%	6%
	Freeze/Wet Tot	**41,238**	**21,976**		**23,877**	**11,723**		**4,185**	**1,944**		**10%**	**9%**
	Luke	1,878	920		1,363	679		645	314		34%	34%
	Holloman	1,460	877		1,270	707		321	149		22%	17%
	Nellis	2,870	1,182		1,885	803		226	100		8%	8%
	Creech	445	333		213	157		22	11		5%	3%
	Davis Monthan	2,247	1,138		1,171	617		187	91		8%	8%
	Edwards	2,842	1,266		1,902	841		452	214		16%	17%
	Ellsworth	945	472		673	331		90	41		10%	9%
	Laughlin	1,163	562		507	269		72	35		6%	6%
	LosAngeles	342	158		171	75		30	75		9%	47%
	No Freeze/Dry T	**14,192**	**6,908**		**9,155**	**4,479**		**2,045**	**1,030**		**14%**	**15%**
	MacDill	2,262	1,073		831	412		177	81		8%	8%
	Hurlburt	2,785	1,349		1,154	578		481	235		17%	17%
	Eglin	13,777	3,974		9,361	2,307		1,359	608		10%	15%
B30 - Roofing	Moody	1,728	591	Filtering*	1,463	495	Cleaning**	379	170	Statistics	22%	29%
	Robins	15,568	3,665		7,520	1,552		2,462	908		16%	25%
	Seymour Johns	1,636	865		805	452		199	92		12%	11%
	Barksdale	2,046	1,402		1,188	720		260	123		13%	9%
	Beale	1,241	773		643	401		173	85		14%	11%
	JB Charleston	4,154	2,325		2,063	1,176		698	332		17%	14%
	Columbus	754	472		387	239		183	89		24%	19%
	Dyess	1,213	506		804	344		142	68		12%	13%
	Keesler	1,169	747		595	379		147	73		13%	10%
	Maxwell	4,638	2,198		1,412	604		118	57		3%	3%
	Patrick	6,707	2,769		3,123	1,167		466	202		7%	7%
	JB SanAntonio	8,400	5,204		4,453	2,955		715	340		9%	7%
	Shaw	1,586	836		857	464		225	104		14%	12%
	Sheppard	3,868	1,443		3,075	954		615	248		16%	17%
	Travis	2,877	1,249		1,227	463		204	93		7%	7%
	Tyndall	1,725	1,053		1,009	598		150	74		9%	7%
	Vandenberg	1,872	1,226		1,028	717		174	87		9%	7%
	No Freeze/Wet	**80,006**	**33,720**		**42,998**	**16,977**		**9,327**	**4,069**		**12%**	**12%**

Total QC-06 Lines	Total Unique Assets
18,817	8,549

Figure 16: Detailed Data Description. Data Description showing initial data for asset and inspection quantities by location, and the resulting data, which was used in this research, after the filtering logic was employed.

Average Condition Value Table:

Model Slope Values (t = inspection CI, & t+1 = all assets with inspections at t & t+1)		Condition (OCI) at Inspection				
	Age	t	t+1	t+2	t+3	t+4
	0	100	94	95	94	NaN
	1	98.5	80	83	88	72
	2	95	50	92	77	79.5
	3	95	95	79.5	61	10
	4	95	95	80	88	91
	5	93	84	94	83.5	59
	6	88	84	75.5	85.75	80
	7	95	91	83	88	88
	8	91	79	83	71	71
	9	88	88	71	83	65
	10	88	45.5	76	71	91.5
	11	87	75.5	88	55	61
	12	88	88	71	71	80
	13	88	61	80	80	86
	14	80	97	70	84	88
	15	87	71	71	66	75.5
	16	80	66	77.75	71	61
	17	83	61	83	66.5	38.75
	18	80	71	61	71	30
	19	71	71	71	61	80
	20	80	61	85	75.5	71
	21	71	71	71	61	75.5
	22	88	80	71	83	61
	23	80	71	60	61	70.5
	24	80	71	45	66	61
	25	71	80	50	60.5	80
	26	69.5	54	71	61	30
	27	88	30	30	61	59
	28	61	71	50	57.5	88
	29	71	61	61	71	61
	30	61	61	88	30	46.5

Table 7: Average Condition Value Table. Average Condition Value Table. The Average Condition Value table for Built-Up Roofing (BUR) shows the OCI at age (x_t) and the median OCI of all assets with out-year inspections at 1, 2, 3, or 4 years immediately after.

Model Slopes Value Table:

		Condition (OCI) at Inspection				
Model Slope Values (t = inspection CI, & t+1 = all assets with inspections at t & t+1)	**Age**	**Weighted** *(100%)*	**t+1** *(40%)*	**t+2** *(30%)*	**t+3** *(20%)*	**t+4** *(10%)*
	0	-0.75	-6	-2.5	-2	NaN
	1	-1.56875	-8	-8.5	-1.66667	-5.875
	2	-1.233333333	-37.5	-4	-6.33333	-4.5
	3	-2.258333333	-5	-10.25	-11.3333	-22.5
	4	-1.233333333	-5	-7.5	-2.33333	-2.25
	5	-0.675	-8.5	-2.5	-3	-1.625
	6	-1.204166667	-7.5	-6.25	-2.66667	-3.75
	7	-1.433333333	-9	-8	-2.33333	-2.375
	8	-0.858333333	-12.5	-3.5	-3.33333	-4.25
	9	-0.975	-7	-4.5	-3	-2.375
	10	-1.408333333	-24.5	-3.5	-8.83333	-0.875
	11	-1.758333333	-10	-3.5	-12.3333	-4.875
	12	-1.7	-7	-8	-5	-2.25
	13	-1.391666667	-10	-7.5	-2.66667	-2.25
	14	-0.833333333	-3	-5	-2.33333	-2.5
	15	-0.991666667	-14	-3.5	-4.66667	-2.875
	16	-1.604166667	-19.5	-6.25	-6.66667	-6.75
	17	-1.0375	-19	-3.5	-5	-10.25
	18	-2.05	-24	-9.5	-5.66667	-12.5
	19	-1.175	-9	-4.5	-5	-2
	20	-1.233333333	-19	-5	-4.83333	-1.25
	21	-2.025	-9	-7.5	-9	-5.5
	22	-1.358333333	-8	-7.5	-2.33333	-3.75
	23	-3.929166667	-17	-19.75	-9.66667	-2.5
	24	-1.820833333	-18	-9.25	-4.33333	-4.25
	25	-1.741666667	-9	-7.5	-6.16667	-2
	26	-1.608333333	-19.5	-8.5	-3.33333	-5.875
	27	-2.133333333	-58	-10	-6.33333	-3.375
	28	-2.116666667	-9	-10	-6.16667	-1.75
	29	-2.3	-10	-17	-3	-5.5
	30	-0.775	-10.5	-3.5	-1.33333	-5

Table 8: Model Slope Values Table. The Model Slope Values table shows weighted slope degradation values expected at any given age and the un-weighted condition degradation values for out-year asset inspections. All values in the Model Slope table are negative. This approach is a significant improvement over SM.

Appendix B

Code for Slope Models:

```
%% Slope-Based Asset Degradation Model (Median-value based)
% temporal/slope-based kNN search

clear, clc

%Base = 'ALL'
%Tknn = readtable([Base, '_B30.csv'], 'HeaderLines',0);  %Import
filtered BUILDER data
Tknn = readtable('Roofing_Only.csv', 'HeaderLines',0);  %Import
filtered BUILDER data

%% Select Roof Type(s) from Tknn Section_Subtype

AssetTypes = unique(Tknn.Section_Subtype)   %lists all Asset Type
OPTIONS

%% Perform simple kNN (non-exclusive OCI at each Age)
AssetType = 'Built-Up' %Use (1) of the OPTIONS in line above
    Tknn = Tknn((Tknn.Section_Subtype == string(AssetType)), :);
%filter out asset types that are NOT desired
% MultiAsset = 'Metal Roof'   %    "Formed Metal" "Formed Metal - Metal
Standing Seam" "Preformed Metal - Metal Panel" "Preformed Metal"
%     Tknn = Tknn((Tknn.Section_Subtype == "Formed Metal"   |
Tknn.Section_Subtype == "Formed Metal - Metal Standing Seam"   |
Tknn.Section_Subtype == "Preformed Metal - Metal Panel"   |
Tknn.Section_Subtype == "Preformed Metal"), :);  %include multiple
asset types


age = unique(Tknn.AgeOffset); % Use age as horizontal axis timescale
variable
x = [Tknn.AgeOffset Tknn.Comp_Rating];   %   Create matrix of yr and
some indicator (example, CI)

CI_fcst = [];
% loop finds kNN based on indicator variable selected
for i = min(age):max(age);
    y = (Tknn.AgeOffset == i) ; %   Select year to find kNN
    z = x((y),:);   %   Create row-vector with age and CI value

    zmed = median(z(:,2));
    zmean = mean(z(:,2));
    temp = [i zmed zmean];

    CI_fcst = vertcat(CI_fcst, temp);
end
```

```
% BUILDER SMS Forecast Values - Original
EXPx = Tknn.AgeOffset;
EXPx(isnan(EXPx)) = 0;
EXPy = Tknn.Expected_Rating;
%EXPy(isnan(EXPy)) = 0;
EXP = [EXPx, EXPy];
EXP = sortrows(EXP, 1);
EXP = unique(EXP, 'rows');
EXP2 = []
i2 = []
      for i = 0:(max(unique(EXP))-1);
          j = EXP(:,1)==i;
          j = max(EXP(:,2).*j);
          EXP2 = vertcat(EXP2,j);
          i2 = vertcat(i2,i);
      end
EXP2 = horzcat(i2, EXP2);


%% Perform 5 Year kNN (exclusive OCI at each Age 1-5 from initial
inspection)

%Add asset unique ID
Tknn.Site_Name = string(Tknn.Site_Name);
Tknn.SEC_ID = string(Tknn.SEC_ID);
ID = Tknn.Site_Name + '-' + Tknn.SEC_ID;
Tknn = addvars(Tknn, ID);

% Convert data format into HztConcat ID, Age 1, CI 1, Age 2, CI 2,Age
2, CI 2, etc.
Tknn2 = table(Tknn.ID, Tknn.Section_Subtype, Tknn.AgeOffset,
Tknn.Comp_Rating);
    Tknn100 = Tknn2.Var4 > 80 ;  %find ONLY assets w/CI's  >60
    Tknn100 = Tknn2.Var4.*Tknn100;
    Tknn100(Tknn100(:,1) == 0) = NaN;
    Tknn100 = table(Tknn2.Var1, Tknn2.Var2, Tknn2.Var3, Tknn100);
%find ONLY assets w/CI's of 81 to 100

    Tknn80 = Tknn2.Var4 > 60 & Tknn2.Var4 <81;  %find ONLY assets
w/CI's  >60
    Tknn80 = Tknn2.Var4.*Tknn80;
    Tknn80(Tknn80(:,1) == 0) = NaN;
    Tknn80 = table(Tknn2.Var1, Tknn2.Var2, Tknn2.Var3, Tknn80);  %find
ONLY assets w/CI's of 81 to 100

    Tknn60 = Tknn2.Var4 <61;  %find ONLY assets w/CI's  >60
    Tknn60 = Tknn2.Var4.*Tknn60;
    Tknn60(Tknn60(:,1) == 0) = NaN;
    Tknn60 = table(Tknn2.Var1, Tknn2.Var2, Tknn2.Var3, Tknn60);  %find
ONLY assets w/CI's of 81 to 100

Tknn2.Properties.VariableNames = {'ID' 'Section_Subtype' 'AgeOffset'
'Comp_Rating'};
```

```
Tknn2 = unstack(Tknn2, 'Comp_Rating', 'AgeOffset',
'AggregationFunction', @mean);
Tknn100.Properties.VariableNames = {'ID' 'Section_Subtype' 'AgeOffset'
'Comp_Rating'};
Tknn100 = unstack(Tknn100, 'Comp_Rating', 'AgeOffset',
'AggregationFunction', @mean);
Tknn80.Properties.VariableNames = {'ID' 'Section_Subtype' 'AgeOffset'
'Comp_Rating'};
Tknn80 = unstack(Tknn80, 'Comp_Rating', 'AgeOffset',
'AggregationFunction', @mean);
Tknn60.Properties.VariableNames = {'ID' 'Section_Subtype' 'AgeOffset'
'Comp_Rating'};
Tknn60 = unstack(Tknn60, 'Comp_Rating', 'AgeOffset',
'AggregationFunction', @mean);


%id = rmmissing(unique(Tknn2.ID)); % Uniquie asset ID's
%x2 = [Tknn.AgeOffset Tknn.Comp_Rating];   %   Create matrix of yr and
some indicator (example, CI) w/metadata for ID & Roof Type

CI2_fcst = [];
CI2_slope = [];
for ii = 0:(width(Tknn2)-2);
    y1 = Tknn2(: , string('x' + string(ii)));   %   Select assets at
age ii to find kNN at first inspection
    y1i = table2array(y1); % Create index vector for first inspection
    y1i(y1i >= 0)=1;     % Create index vector for first inspection
    y1i(isnan(y1i))=0;  % Create index vector for first inspection

    % Find assets with inspections at both first age and each age after
for next 4yrs = 5yrs total
    y2 = y1i.*(table2array(Tknn2(: , string('x' + string(ii+1))))); %
Select assets at age ii+1 to find kNN at second inspection
    y2(y2==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y3 = y1i.*(table2array(Tknn2(: , string('x' + string(ii+2))))); %
Select assets at age ii+2 to find kNN at second+ inspection
    y3(y3==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y4 = y1i.*(table2array(Tknn2(: , string('x' + string(ii+3))))); %
Select assets at age ii+3 to find kNN at second+ inspection
    y4(y4==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y5 = y1i.*(table2array(Tknn2(: , string('x' + string(ii+4))))); %
Select assets at age ii+4 to find kNN at second+ inspection
    y5(y5==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below

    % Find mean values of inspections at each year (5yrs total)
    y1_val = nanmedian(table2array(y1)); %Mean value of all assets with
values at ii
    y2_val = nanmedian(y2); %Mean value of all assets with values at ii
& ii+1
```

```
    y3_val = nanmedian(y3); %Mean value of all assets with values at ii
& ii+2
    y4_val = nanmedian(y4); %Mean value of all assets with values at ii
& ii+3
    y5_val = nanmedian(y5); %Mean value of all assets with values at ii
& ii+4

    % Find slope values between inspections (4 values total)
    y1_slope = y2 - (table2array(y1));
    y1_slope = nanmedian(y1_slope);
    y2_slope = y3 - (table2array(y1));
    y2_slope = nanmedian(y2_slope)/2;
    y3_slope = y4 - (table2array(y1));
    y3_slope = nanmedian(y3_slope)/3;
    y4_slope = y5 - (table2array(y1));
    y4_slope = nanmedian(y4_slope)/4;
    mean_slope = nanmedian([y1_slope y2_slope y3_slope y4_slope].*[4 3
2 1])/(4+3+2+1); %proximity weight the slopes for each year beyond
inspection

    %z1 = x2((Tknn2),:)    %   Create row-vector with age and CI value

    temp = [ii y1_val y2_val y3_val y4_val y5_val];
    temp2 = [ii mean_slope y1_slope y2_slope y3_slope y4_slope];

    CI2_fcst = vertcat(CI2_fcst, temp);
    CI2_slope = vertcat(CI2_slope, temp2);
end

%CI2_fcst(:,1)


CI2_fcst100 = [];
CI2_slope100 = [];
for ii = 0:(width(Tknn100)-2);
    y1 = Tknn100(: , string('x' + string(ii)));    %   Select assets at
age ii to find kNN at first inspection
    y1i = table2array(y1); % Create index vector for first inspection
    y1i(y1i >= 0)=1;     % Create index vector for first inspection
    y1i(isnan(y1i))=0;  % Create index vector for first inspection

    % Find assets with inspections at both first age and each age after
for next 4yrs = 5yrs total
    y2 = y1i.*(table2array(Tknn100(: , string('x' + string(ii+1))))); %
Select assets at age ii+1 to find kNN at second inspection
    y2(y2==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y3 = y1i.*(table2array(Tknn100(: , string('x' + string(ii+2))))); %
Select assets at age ii+2 to find kNN at second+ inspection
    y3(y3==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y4 = y1i.*(table2array(Tknn100(: , string('x' + string(ii+3))))); %
Select assets at age ii+3 to find kNN at second+ inspection
```

```
    y4(y4==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y5 = y1i.*(table2array(Tknn100(: , string('x' + string(ii+4))))); %
Select assets at age ii+4 to find kNN at second+ inspection
    y5(y5==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below

    % Find mean values of inspections at each year (5yrs total)
    y1_val = nanmedian(table2array(y1)); %Mean value of all assets with
values at ii
    y2_val = nanmedian(y2); %Mean value of all assets with values at ii
& ii+1
    y3_val = nanmedian(y3); %Mean value of all assets with values at ii
& ii+2
    y4_val = nanmedian(y4); %Mean value of all assets with values at ii
& ii+3
    y5_val = nanmedian(y5); %Mean value of all assets with values at ii
& ii+4

    % Find slope values between inspections (4 values total)
    y1_slope = y2 - (table2array(y1));
    y1_slope = nanmedian(y1_slope);
    y2_slope = y3 - (table2array(y1));
    y2_slope = nanmedian(y2_slope)/2;
    y3_slope = y4 - (table2array(y1));
    y3_slope = nanmedian(y3_slope)/3;
    y4_slope = y5 - (table2array(y1));
    y4_slope = nanmedian(y4_slope)/4;
    mean_slope = nanmedian([y1_slope y2_slope y3_slope y4_slope].*[4 3
2 1])/(4+3+2+1); %proximity weight the slopes for each year beyond
inspection

    %z1 = x2((Tknn2),:)    %   Create row-vector with age and CI value

    temp = [ii y1_val y2_val y3_val y4_val y5_val];
    temp2 = [ii mean_slope y1_slope y2_slope y3_slope y4_slope];

    CI2_fcst100 = vertcat(CI2_fcst100, temp);
    CI2_slope100 = vertcat(CI2_slope100, temp2);
end

CI2_fcst80 = [];
CI2_slope80 = [];
for ii = 0:(width(Tknn80)-2);
    y1 = Tknn80(: , string('x' + string(ii)));   %   Select assets at
age ii to find kNN at first inspection
    y1i = table2array(y1); % Create index vector for first inspection
    y1i(y1i >= 0)=1;     % Create index vector for first inspection
    y1i(isnan(y1i))=0;   % Create index vector for first inspection

    % Find assets with inspections at both first age and each age after
for next 4yrs = 5yrs total
    y2 = y1i.*(table2array(Tknn80(: , string('x' + string(ii+1))))); %
Select assets at age ii+1 to find kNN at second inspection
```

```
    y2(y2==0) = NaN;    %   Convert zeros to NaN to erase from mean
calc below
    y3 = y1i.*(table2array(Tknn80(: , string('x' + string(ii+2))))); %
Select assets at age ii+2 to find kNN at second+ inspection
    y3(y3==0) = NaN;    %   Convert zeros to NaN to erase from mean
calc below
    y4 = y1i.*(table2array(Tknn80(: , string('x' + string(ii+3))))); %
Select assets at age ii+3 to find kNN at second+ inspection
    y4(y4==0) = NaN;    %   Convert zeros to NaN to erase from mean
calc below
    y5 = y1i.*(table2array(Tknn80(: , string('x' + string(ii+4))))); %
Select assets at age ii+4 to find kNN at second+ inspection
    y5(y5==0) = NaN;    %   Convert zeros to NaN to erase from mean
calc below

    % Find mean values of inspections at each year (5yrs total)
    y1_val = nanmedian(table2array(y1)); %Mean value of all assets with
values at ii
    y2_val = nanmedian(y2); %Mean value of all assets with values at ii
& ii+1
    y3_val = nanmedian(y3); %Mean value of all assets with values at ii
& ii+2
    y4_val = nanmedian(y4); %Mean value of all assets with values at ii
& ii+3
    y5_val = nanmedian(y5); %Mean value of all assets with values at ii
& ii+4

    % Find slope values between inspections (4 values total)
    y1_slope = y2 - (table2array(y1));
    y1_slope = nanmedian(y1_slope);
    y2_slope = y3 - (table2array(y1));
    y2_slope = nanmedian(y2_slope)/2;
    y3_slope = y4 - (table2array(y1));
    y3_slope = nanmedian(y3_slope)/3;
    y4_slope = y5 - (table2array(y1));
    y4_slope = nanmedian(y4_slope)/4;
    mean_slope = nanmedian([y1_slope y2_slope y3_slope y4_slope].*[4 3
2 1])/(4+3+2+1); %proximity weight the slopes for each year beyond
inspection

    %z1 = x2((Tknn2),:)    %   Create row-vector with age and CI value

    temp = [ii y1_val y2_val y3_val y4_val y5_val];
    temp2 = [ii mean_slope y1_slope y2_slope y3_slope y4_slope];

    CI2_fcst80 = vertcat(CI2_fcst80, temp);
    CI2_slope80 = vertcat(CI2_slope80, temp2);
end

CI2_fcst60 = [];
CI2_slope60 = [];
for ii = 0:(width(Tknn60)-2);
    y1 = Tknn60(: , string('x' + string(ii)));    %   Select assets at
age ii to find kNN at first inspection
```

```
    y1i = table2array(y1); % Create index vector for first inspection
    y1i(y1i >= 0)=1;     % Create index vector for first inspection
    y1i(isnan(y1i))=0;  % Create index vector for first inspection

    % Find assets with inspections at both first age and each age after
for next 4yrs = 5yrs total
    y2 = y1i.*(table2array(Tknn60(: , string('x' + string(ii+1))))); %
Select assets at age ii+1 to find kNN at second inspection
    y2(y2==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y3 = y1i.*(table2array(Tknn60(: , string('x' + string(ii+2))))); %
Select assets at age ii+2 to find kNN at second+ inspection
    y3(y3==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y4 = y1i.*(table2array(Tknn60(: , string('x' + string(ii+3))))); %
Select assets at age ii+3 to find kNN at second+ inspection
    y4(y4==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below
    y5 = y1i.*(table2array(Tknn60(: , string('x' + string(ii+4))))); %
Select assets at age ii+4 to find kNN at second+ inspection
    y5(y5==0) = NaN;     %   Convert zeros to NaN to erase from mean
calc below

    % Find mean values of inspections at each year (5yrs total)
    y1_val = nanmedian(table2array(y1)); %Mean value of all assets with
values at ii
    y2_val = nanmedian(y2); %Mean value of all assets with values at ii
& ii+1
    y3_val = nanmedian(y3); %Mean value of all assets with values at ii
& ii+2
    y4_val = nanmedian(y4); %Mean value of all assets with values at ii
& ii+3
    y5_val = nanmedian(y5); %Mean value of all assets with values at ii
& ii+4

    % Find slope values between inspections (4 values total)
    y1_slope = y2 - (table2array(y1));
    y1_slope = nanmedian(y1_slope);
    y2_slope = y3 - (table2array(y1));
    y2_slope = nanmedian(y2_slope)/2;
    y3_slope = y4 - (table2array(y1));
    y3_slope = nanmedian(y3_slope)/3;
    y4_slope = y5 - (table2array(y1));
    y4_slope = nanmedian(y4_slope)/4;
    mean_slope = nanmedian([y1_slope y2_slope y3_slope y4_slope].*[4 3
2 1])/(4+3+2+1); %proximity weight the slopes for each year beyond
inspection

    %z1 = x2((Tknn2),:)    %   Create row-vector with age and CI value

    temp = [ii y1_val y2_val y3_val y4_val y5_val];
    temp2 = [ii mean_slope y1_slope y2_slope y3_slope y4_slope];

    CI2_fcst60 = vertcat(CI2_fcst60, temp);
```

```
    CI2_slope60 = vertcat(CI2_slope60, temp2);
end

%% Plot original OCIs, Mean OCI, & Fcst CI @ each Age-step (+1, +2, +3,
+4)

figure1 = figure('color', [1,1,1])
OCI = scatter(Tknn.AgeOffset, Tknn.Comp_Rating,'k+','LineWidth',1)
hold on
CImean = plot(CI_fcst(1: length(CI_fcst),1), CI_fcst(1:
length(CI_fcst),3),'b+','LineWidth',3)
a1 = plot(CI2_fcst(:,1), CI2_fcst(:,2),'k+','LineWidth',3) %Mean OCI
(:,2) at inspection Age (:,1)

a2 = plot(CI2_fcst(:,1)+1, CI2_fcst(:,3),'bo','LineWidth',1) %Mean OCI
at 1yr later
a3 = plot(CI2_fcst(:,1)+2, CI2_fcst(:,4),'go','LineWidth',1) %Mean OCI
at 2yrs later
a4 = plot(CI2_fcst(:,1)+3, CI2_fcst(:,5),'ro','LineWidth',1) %Mean OCI
at 3yrs later
a5 = plot(CI2_fcst(:,1)+4, CI2_fcst(:,6),'o','LineWidth',1) %Mean OCI
at 4yrs later

for iii = 1:(length(CI_fcst)-1)
    plot([CI2_fcst(iii,1); CI2_fcst(iii,1)+1],[CI2_fcst(iii,2);
CI2_fcst(iii,3)],'b','LineWidth',1)  %Connects Mean OCIs for each iii+1
    plot([CI2_fcst(iii,1); CI2_fcst(iii,1)+2],[CI2_fcst(iii,2);
CI2_fcst(iii,4)],'g','LineWidth',1)  %Connects Mean OCIs for each iii+2
    plot([CI2_fcst(iii,1); CI2_fcst(iii,1)+3],[CI2_fcst(iii,2);
CI2_fcst(iii,5)],'r','LineWidth',1)  %Connects Mean OCIs for each iii+3
    plot([CI2_fcst(iii,1); CI2_fcst(iii,1)+4],[CI2_fcst(iii,2);
CI2_fcst(iii,6)],'o','LineWidth',1)  %Connects Mean OCIs for each iii+4
end

xlim([0 length(CI2_fcst)]) %This automatically sets x-axis limits
ylim([0 100]) %This automatically sets x-axis limits
set(gca,'XTick',[0:5:length(CI2_fcst)]) %This automatically sets x-axis
ticks
xlabel('Age')
ylabel('OCI')
title('Nearest Neighbor Forecast - ' + string(AssetType)) %Single Asset
Type
%title('Nearest Neighbor Forecast - ' + string(MultiAsset)) %MultiAsset
"Formed Metal" "Formed Metal - Metal Standing Seam" "Preformed Metal -
Metal Panel" "Preformed Metal"

legend([OCI CImean a1 a2 a3 a4 a5], {'All Inspections', 'Median OCI',
'Forecast', 'Forecast +1', 'Forecast +2', 'Forecast +3', 'Forecast
+4'}, 'Location', 'southwest')
hold off

%% Plot original OCIs, Mean OCI, & Fcst CI @ each Age-step

% figure1 = figure('color', [1,1,1])
```

```
OCI = scatter(Tknn.AgeOffset, Tknn.Comp_Rating,'k+','LineWidth',1)
hold on
CImean = scatter(CI_fcst(:,1), CI_fcst(:,2),'r+','LineWidth',3) %MEDIAN
forecast
    for iiii = 1:(length(CI_fcst)-1)
        plot([CI_fcst(iiii,1) CI_fcst(iiii+1,1)], [CI_fcst(iiii,2)
CI_fcst(iiii+1,2)],'r','LineWidth',2)
    end

CI2mean = scatter(CI2_fcst(:,1), CI2_fcst(:,2),'b+','LineWidth',3)
%MEAN forecast
    for iiii = 1:(length(CI2_fcst)-1)
        plot([CI2_fcst(iiii,1) CI2_fcst(iiii+1,1)], [CI2_fcst(iiii,2)
CI2_fcst(iiii+1,2)],'b','LineWidth',2)
    end

% ECI = scatter(Tknn.AgeOffset,
Tknn.Expected_Rating,'k+','LineWidth',3) %BUILDER's forecast (Expected)
%     for iiii = 1:(height(Tknn)-1)
%         plot([Tknn.AgeOffset(iiii,1) Tknn.AgeOffset(iiii+1,1)],
[Tknn.Expected_Rating(iiii)
Tknn.Expected_Rating(iiii+1)],'r','LineWidth',2)
%     end

CI2_fcstSlope = [min(CI2_slope(:,1)) : max(CI2_slope(:,1))]'; %Slope-
base model: starts at 100=CI then subtracts slope values as age
increases
CI2_fcstSlope = horzcat(CI2_fcstSlope, CI2_slope(:,2));
CI2_fcstSlope(isnan(CI2_fcstSlope)) = 0;
pred = []
    for j = 0 : length(CI2_slope(:,1))-1;
        p = 100 + sum(CI2_fcstSlope(1:j, 2));
        temp3 = [j p];
        pred = vertcat(pred, temp3);
    end
CIslope = scatter(pred(:,1), pred(:,2),'c+','LineWidth',3) %MEDIAN
forecast
    for iiii = 1:(length(pred)-1)
        plot([pred(iiii,1) pred(iiii+1,1)], [pred(iiii,2)
pred(iiii+1,2)],'c','LineWidth',2)
    end

% xlim([0 length(CI2_fcst)]) %This automatically sets x-axis limits
xlim([0 35]) %This automatically sets x-axis limits
ylim([0 100])
set(gca,'XTick',[0:5:length(CI2_fcst)]) %This automatically sets x-axis
ticks
xlabel('Age')
ylabel('OCI')
title('Slope Forecast - ' + string(AssetType)) %Single Asset Type
% title('Nearest Neighbor Forecast - ' + string(MultiAsset))
%MultiAsset    "Formed Metal" "Formed Metal - Metal Standing Seam"
"Preformed Metal - Metal Panel" "Preformed Metal"
```

```
legend([OCI CI2mean], {'All Inspections', 'Mean OCI'}, 'Location',
'southwest')
legend([OCI CImean CI2mean CIslope], {'All Inspections', 'Median OCI',
'Forecast', 'Slope Fcst'}, 'Location', 'southwest')
hold off

%% Weighted Slope Model & BUILDER  Residual Comparison

figure1 = figure('Color', [1 1 1]);      %create a figure with white
background color.
subplot(2,1,1)                           %subplot(row,columns,position)

%% CI Slope Model w/BUILDER Ranges

OCI = scatter(Tknn.AgeOffset, Tknn.Comp_Rating,'k+','LineWidth',1)
hold on

 v = [0 80; 0 100; length(CI2_fcst) 100; length(CI2_fcst) 80];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[0 1 0], 'FaceAlpha',(.5))
    yline(80, ":k",'GOOD') %Plus 20 CI bound
 v = [0 60; 0 80; length(CI2_fcst) 80; length(CI2_fcst) 60];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[1 1 0], 'FaceAlpha',(.5))
    yline(60, ":k",'REPAIR') %Plus 20 CI bound
 v = [0 0; 0 60; length(CI2_fcst) 60; length(CI2_fcst) 0];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[1 0 0], 'FaceAlpha',(.5))
    yline(0, ":k",'REPLACE') %Plus 20 CI bound

    CI2_fcstSlope = [min(CI2_slope(:,1)) : max(CI2_slope(:,1))]';
%Slope-base model: starts at 100=CI then subtracts slope values as age
increases
CI2_fcstSlope = horzcat(CI2_fcstSlope, CI2_slope(:,2));
CI2_fcstSlope(isnan(CI2_fcstSlope)) = 0;
pred = []
    for j = 0 : length(CI2_slope(:,1))-1;
        p = 100 + sum(CI2_fcstSlope(1:j, 2));
        temp3 = [j p];
        pred = vertcat(pred, temp3);
    end

    CIslope = scatter(pred(:,1), pred(:,2),'b+','LineWidth',3) %MEDIAN
forecast
    for iiii = 1:(length(pred)-1)
        plot([pred(iiii,1) pred(iiii+1,1)], [pred(iiii,2)
pred(iiii+1,2)],'b','LineWidth',2)
    end

xlim([0 length(CI2_fcst)]) %This automatically sets x-axis limits
%xlim([0 35]) %Sets x-axis limits for Shingle/BUR/ModBit
ylim([0 100])
set(gca,'XTick',[0:5:length(CI2_fcst)]) %This automatically sets x-axis
ticks
```

```
xlabel('Age')
%xlabel('Age (Range =Asset Service Life)')
ylabel('OCI')
title('Weighted Slope Model - ' + string(AssetType)) %Single Asset Type
% title('Weighted Slope Model - ' + string(MultiAsset)) %MultiAsset
"Formed Metal" "Formed Metal - Metal Standing Seam" "Preformed Metal -
Metal Panel" "Preformed Metal"
%title('Condition Intelligent Weighted Slope Model - ' +
string(AssetType)) %Single Asset Type
% title('Condition Intelligent Weighted Slope Model - ' +
string(MultiAsset)) %MultiAsset    "Formed Metal" "Formed Metal - Metal
Standing Seam" "Preformed Metal - Metal Panel" "Preformed Metal"


legend([OCI CIslope], {'All Inspections', 'W-SM Fcst'}, 'Location',
'southwest')

%% RESIDUALS - Delta CI BUILDER vs Delta CI Slope Model
subplot(2,1,2)
hold on

%figure1 = figure('Color', [1 1 1]);     %create a figure with white
background color.


%DCI Original
DCIx = Tknn.AgeOffset;
DCIx(isnan(DCIx)) = 0;
DCIy = Tknn.DeltaCI;
DCIy(isnan(DCIy)) = 0;
%DCI = [DCIx, DCIy];
DCI = polyfit(DCIx, DCIy, 3);
DCI = [DCIx polyval(DCI, DCIx)];
DCI = sort(DCI, 1);
DCI = unique(DCI, 'rows');
    for iiii = 1:(length(DCI)-1)
            plot1 = plot([DCI(iiii,1) DCI(iiii+1,1)], [DCI(iiii,2)
DCI(iiii+1,2)],'k+','LineWidth',2) %BUILDER's forecast residuals(Delta
CI = OCI - ECI)
    end

    hold on

%DCI Weighted Slope Model (WSM)
DCI2x = [1:(length(pred))]';
DCI2x(isnan(DCI2x)) = 0;
DCI2y = CI_fcst(1:length(pred), 3) - pred(:,2); %Calculate the newDCI
value using Slope Forecast (Delta CI2 = OCI - FCI)
DCI2y(isnan(DCI2y)) = 0;
DCI2 = polyfit(DCI2x, DCI2y, 3);
DCI2 = [DCI2x polyval(DCI2, DCI2x)];
DCI2 = sort(DCI2, 1);
    for iiii = 1:(length(DCI2x)-1);
```

```
            plot2 = plot([DCI2(iiii,1) DCI2(iiii+1,1)], [DCI2(iiii,2)
DCI2(iiii+1,2)],'ro','LineWidth',2) %Slope Forecast residuals(Delta CI
= OCI - FCI)
    end

%DCI Slope Model (SM)
DCI3x = [1:(length(pred))]';
DCI3x(isnan(DCI3x)) = 0;
DCI3y = CI_fcst(1:length(pred), 2);
%insert non-positive values

DCI3y = DCI3y - pred(:,2); %Calculate the newDCI value using Slope
Forecast (Delta CI3 = OCI - FCI)
DCI3y(isnan(DCI3y)) = 0;
DCI3 = polyfit(DCI3x, DCI3y, 3);
DCI3 = [DCI3x polyval(DCI3, DCI3x)];
DCI3 = sort(DCI3, 1);
    for iiii = 1:(length(DCI3x)-1);
            plot23 = plot([DCI3(iiii,1) DCI3(iiii+1,1)], [DCI3(iiii,2)
DCI3(iiii+1,2)],'bo','LineWidth',2) %Slope Forecast residuals(Delta CI
= OCI - FCI)
    end



xlim([0 length(CI2_fcst)]) %This automatically sets x-axis limits
%xlim([0 35]) %Sets x-axis limits for Shingle/BUR/ModBit
ylim([-25 25])
        v = [0 -20; 0 20; length(CI2_fcst) 20; length(CI2_fcst) -20];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[.5 .5 .5],
'FaceAlpha',(.25))
    plus20 = yline(20, ":b",+20) %Plus 20 CI bound
    minus20 = yline(-20, ":b", -20) %Minus 20 CI bound
        v = [0 -10; 0 10; length(CI2_fcst) 10; length(CI2_fcst) -10];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[.5 .5 .5],
'FaceAlpha',(.25))
    plus10 = yline(10, ":b",+10) %Plus 10 CI bound
    minus10 = yline(-10, ":b", -10) %Minus 10 CI bound
        v = [0 -5; 0 5; length(CI2_fcst) 5; length(CI2_fcst) -5];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[.5 .5 .5],
'FaceAlpha',(.25))
    plus10 = yline(5, ":b",+5) %Plus 5 CI bound
    minus10 = yline(-5, ":b", -5) %Minus 5 CI bound
    OCI0 = yline(0, ":b", 'Perfect Prediction is OCI=0') %OCI Target
Residual = 0

set(gca,'XTick',[0:5:length(CI2_fcst)]) %This automatically sets x-axis
ticks
xlabel('Age')
%xlabel('Age (Range =Asset Service Life)')
ylabel('DCI')
```

```
title('Weighted Slope Model - (Comparison of Residuals = DCI) - ' +
string(AssetType)) %Single Asset Type
% title('Weighted Slope Model - (Comparison of Residuals = DCI) - ' +
string(MultiAsset)) %MultiAsset    "Formed Metal" "Formed Metal - Metal
Standing Seam" "Preformed Metal - Metal Panel" "Preformed Metal"
%title('Condition Intelligent Weighted Slope Model - (Comparison of
Residuals = DCI) - ' + string(AssetType)) %Single Asset Type
% title('Condition Intelligent Weighted Slope Model - (Comparison of
Residuals = DCI) - ' + string(MultiAsset)) %MultiAsset    "Formed
Metal" "Formed Metal - Metal Standing Seam" "Preformed Metal - Metal
Panel" "Preformed Metal"


legend([plot1 plot2], {'BUILDER DCI', 'W-SM DCI'}, 'Location',
'southwest' )
%legend([plot1 plot23], {'BUILDER DCI', 'SM DCI'}, 'Location',
'southwest' )
legend([plot1 plot2 plot23], {'BUILDER DCI', 'W-SM DCI', 'SM DCI'},
'Location', 'southwest' )
%legend([plot1 plot21 plot22 plot23], {'BUILDER DCI', 'Good Forecast
DCI', 'Repair Forecast DCI', 'Replace Forecast DCI'}, 'Location',
'northwest' )
hold off
%% Calculate fit parameters


CI_fcst(isnan(CI_fcst)) = 0; %NAN's to zeros for correlation calc
R2_fcst = corr(pred(:,2),CI_fcst(1:length(pred), 3))^2
%R2_fcst = corr(pred(1:26,2),CI_fcst(1:26,3))^2
RMSE_fcst = sqrt(immse(pred(:,2),CI_fcst(1:length(pred),3)))
%RMSE_fcst = sqrt(immse(pred(1:26,2),CI_fcst(1:26,3)))

Tknn.Expected_Rating(isnan(Tknn.Expected_Rating)) = 0;
Tknn.Comp_Rating(isnan(Tknn.Comp_Rating)) = 0;
R2_bldr = corr(Tknn.Expected_Rating, Tknn.Comp_Rating)^2
%R2_bldr = corr(Tknn.Expected_Rating(1:26), Tknn.Comp_Rating(1:26))^2
RMSE_bldr = sqrt(immse(Tknn.Expected_Rating(:),Tknn.Comp_Rating(:)))
%RMSE_bldr =
sqrt(immse(Tknn.Expected_Rating(1:26),Tknn.Comp_Rating(1:26)))
%rmse=sqrt(mean((y(:)-yhat(:)).^2));
%rmse = sqrt(immse(scores, dates));


%% Output Slope Model to .mat file for use in ensemble

% writematrix(pred, [AssetType, '_SlopeFcst.csv'], 'Delimiter', ',');
%output Slope Forecast values to a csv file
writematrix(CI2_slope, [AssetType, '_SlopeFcstS.csv'], 'Delimiter',
',');    %output Slope Forecast values to a csv file
writematrix(CI2_slope100, [AssetType, '_SlopeFcstS_100.csv'],
'Delimiter', ',');    %output Slope Forecast values to a csv file
writematrix(CI2_slope80, [AssetType, '_SlopeFcstS_80.csv'],
'Delimiter', ',');    %output Slope Forecast values to a csv file
```

```
writematrix(CI2_slope60, [AssetType, '_SlopeFcstS_60.csv'],
'Delimiter', ',');    %output Slope Forecast values to a csv file

%Multi-Asset (Metal Roofing) File Export
% writematrix(CI2_slope, [MultiAsset, '_SlopeFcstS.csv'], 'Delimiter',
',');    %output Slope Forecast values to a csv file
% writematrix(CI2_slope100, [MultiAsset, '_SlopeFcstS_100.csv'],
'Delimiter', ',');    %output Slope Forecast values to a csv file
% writematrix(CI2_slope80, [MultiAsset, '_SlopeFcstS_80.csv'],
'Delimiter', ',');    %output Slope Forecast values to a csv file
% writematrix(CI2_slope60, [MultiAsset, '_SlopeFcstS_60.csv'],
'Delimiter', ',');    %output Slope Forecast values to a csv file


%% CI-W-SM
%% CONDITION INTELLIGENT PLOTs

CI3_fcstSlope = [CI2_slope100(:,1:2) CI2_slope80(:,2)
CI2_slope60(:,2)];
CI3_fcstSlope(isnan(CI3_fcstSlope)) = 0;

    pred3 = [0 100]
        for j = 1 : length(CI3_fcstSlope(:,1))-1;
            if pred3(j,2) > 80
                p = CI3_fcstSlope(j, 2)+ pred3(j, 2);
                temp4 = [j p];
            elseif pred3(j,2) > 60
                p = CI3_fcstSlope(j, 3)+ pred3(j, 2);
                temp4 = [j p];
            else
                p = CI3_fcstSlope(j, 4)+ pred3(j, 2);
                temp4 = [j p];
            end
            pred3 = vertcat(pred3, temp4);
        end

% THIS CODE IS NOT NEEDED...IT WAS USED TO BUILD THE ABOVE FOR LOOP
% pred100 = []
%     for j = 0 : length(CI3_fcstSlope(:,1))-1;
%         p = 100 + sum(CI3_fcstSlope(1:j, 2));
%         temp4 = [j p];
%         pred100 = vertcat(pred100, temp4);
%     end
%     pred80 = []
%     for j = 0 : length(CI3_fcstSlope(:,1))-1;
%         p = 100 + sum(CI3_fcstSlope(1:j, 3));
%         temp4 = [j p];
%         pred80 = vertcat(pred80, temp4);
%     end
%     pred60 = []
%     for j = 0 : length(CI3_fcstSlope(:,1))-1;
%         p = 100 + sum(CI3_fcstSlope(1:j, 4));
%         temp4 = [j p];
%         pred60 = vertcat(pred60, temp4);
```

```
%     end
%



%% STOP!!!!!!!!!!!!!!!!!!!

%% CONDITION INTELLIGENT Weighted Slope Model & BUILDER  Residual
Comparison

figure1 = figure('Color', [1 1 1]);     %create a figure with white
background color.
subplot(2,1,1)                          %subplot(row,columns,position)

%% CI Slope Model w/BUILDER Ranges

OCI = scatter(Tknn.AgeOffset, Tknn.Comp_Rating,'k+','LineWidth',1)
hold on

 v = [0 80; 0 100; length(CI2_fcst) 100; length(CI2_fcst) 80];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[0 1 0], 'FaceAlpha',(.5))
    yline(80, ":k",'GOOD') %Plus 20 CI bound
 v = [0 60; 0 80; length(CI2_fcst) 80; length(CI2_fcst) 60];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[1 1 0], 'FaceAlpha',(.5))
    yline(60, ":k",'REPAIR') %Plus 20 CI bound
 v = [0 0; 0 60; length(CI2_fcst) 60; length(CI2_fcst) 0];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[1 0 0], 'FaceAlpha',(.5))
    yline(0, ":k",'REPLACE') %Plus 20 CI bound


    for iiii = 1:(length(pred3)-1)
        plot3 = plot([pred3(iiii,1) pred3(iiii+1,1)], [pred3(iiii,2)
pred3(iiii+1,2)],'c','LineWidth',2)
    end

xlim([0 length(CI3_fcstSlope)]) %This automatically sets x-axis limits
%xlim([0 35]) %Sets x-axis limits for Shingle/BUR/ModBit
ylim([0 100])
set(gca,'XTick',[0:5:length(CI2_fcst)]) %This automatically sets x-axis
ticks
xlabel('Age')
%xlabel('Age (Range =Asset Service Life)')
ylabel('DCI')
title('Condition Intelligent Weighted Slope Model - ' +
string(AssetType)) %Single Asset Type
% title('Condition Intelligent Weighted Slope Model - ' +
string(MultiAsset)) %MultiAsset    "Formed Metal" "Formed Metal - Metal
Standing Seam" "Preformed Metal - Metal Panel" "Preformed Metal"
```

```
legend([OCI plot3], {'All Inspections', 'Condition Intelligent Fcst'},
'Location', 'southwest')

%% RESIDUALS - Delta CI BUILDER vs Delta CI Slope Model
subplot(2,1,2)
hold on

        v = [0 -20; 0 20; length(CI2_fcst) 20; length(CI2_fcst) -20];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[.5 .5 .5],
'FaceAlpha',(.25))
    plus20 = yline(20, ":b",+20) %Plus 20 CI bound
    minus20 = yline(-20, ":b", -20) %Minus 20 CI bound
        v = [0 -10; 0 10; length(CI2_fcst) 10; length(CI2_fcst) -10];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[.5 .5 .5],
'FaceAlpha',(.25))
    plus10 = yline(10, ":b",+10) %Plus 10 CI bound
    minus10 = yline(-10, ":b", -10) %Minus 10 CI bound
        v = [0 -5; 0 5; length(CI2_fcst) 5; length(CI2_fcst) -5];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[.5 .5 .5],
'FaceAlpha',(.25))
    plus10 = yline(5, ":b",+5) %Plus 5 CI bound
    minus10 = yline(-5, ":b", -5) %Minus 5 CI bound
    OCI0 = yline(0, ":b", 'Perfect Prediction is OCI=0') %OCI Target
Residual = 0

%DCI Original
DCIx = Tknn.AgeOffset;
DCIx(isnan(DCIx)) = 0;
DCIy = Tknn.DeltaCI;
DCIy(isnan(DCIy)) = 0;
DCI = polyfit(DCIx, DCIy, 3);
DCI = [DCIx polyval(DCI, DCIx)];
DCI = sort(DCI, 1);
DCI = unique(DCI, 'rows');
    for iiii = 1:(length(DCI)-1)
            plot1 = plot([DCI(iiii,1) DCI(iiii+1,1)], [DCI(iiii,2)
DCI(iiii+1,2)],'k+','LineWidth',2) %BUILDER's forecast residuals(Delta
CI = OCI - ECI)
    end

    hold on


%DCI100 Slope Model
DCI100x = [1:(length(pred3))]';
DCI100x(isnan(DCI100x)) = 0;
DCI100y = CI2_fcst100(1:length(pred3), 3) - pred3(:,2); %Calculate the
newDCI value using Slope Forecast (Delta CI2 = OCI - FCI)
DCI100y(isnan(DCI100y)) = 0;
DCI100 = polyfit(DCI100x, DCI100y, 3);
DCI100 = [DCI100x polyval(DCI100, DCI100x)];
```

```
DCI100 = sort(DCI100, 1);
    for iiii = 1:(length(DCI100)-1);
            plot21 = plot([DCI100(iiii,1) DCI100(iiii+1,1)],
[DCI100(iiii,2) DCI100(iiii+1,2)],'ro','LineWidth',2) %Slope Forecast
residuals(Delta CI = OCI - FCI)
    end
%DCI80 Slope Model
DCI80x = [1:(length(pred3))]';
DCI80x(isnan(DCI80x)) = 0;
DCI80y = CI2_fcst80(1:length(pred3), 3) - pred3(:,2); %Calculate the
newDCI value using Slope Forecast (Delta CI2 = OCI - FCI)
DCI80y(isnan(DCI80y)) = 0;
DCI80 = polyfit(DCI80x, DCI80y, 3);
DCI80 = [DCI80x polyval(DCI80, DCI80x)];
DCI80 = sort(DCI80, 1);
    for iiii = 1:(length(DCI80)-1);
            plot22 = plot([DCI80(iiii,1) DCI80(iiii+1,1)],
[DCI80(iiii,2) DCI80(iiii+1,2)],'go','LineWidth',2) %Slope Forecast
residuals(Delta CI = OCI - FCI)
    end
%DCI60 Slope Model
DCI60x = [1:(length(pred3))]';
DCI60x(isnan(DCI60x)) = 0;
DCI60y = CI2_fcst60(1:length(pred3), 3) - pred3(:,2); %Calculate the
newDCI value using Slope Forecast (Delta CI2 = OCI - FCI)
DCI60y(isnan(DCI60y)) = 0;
DCI60 = polyfit(DCI60x, DCI60y, 3);
DCI60 = [DCI60x polyval(DCI60, DCI60x)];
DCI60 = sort(DCI60, 1);
    for iiii = 1:(length(DCI60)-1);
            plot23 = plot([DCI60(iiii,1) DCI60(iiii+1,1)],
[DCI60(iiii,2) DCI60(iiii+1,2)],'yo','LineWidth',2) %Slope Forecast
residuals(Delta CI = OCI - FCI)
    end


%     %DCICombined Slope Model
%     %DCIc = [1:(length(pred)) ]';
%     %DCIc = [DCIc DCI100(:,2)+DCI80(:,2)+DCI60(:,2)];
%         for iiii = 1:(length(DCIc)-1);
%                 plot24 = plot([DCIc(iiii,1) DCIc(iiii+1,1)],
[DCIc(iiii,2) DCIc(iiii+1,2)],'bo','LineWidth',2) %Slope Forecast
residuals(Delta CI = OCI - FCI)
%         end

xlim([0 length(CI2_fcst)]) %This automatically sets x-axis limits
%xlim([0 35]) %Sets x-axis limits for Shingle/BUR/ModBit
ylim([-25 25])

set(gca,'XTick',[0:5:length(CI2_fcst)]) %This automatically sets x-axis
ticks
xlabel('Age')
%xlabel('Age (Range =Asset Service Life)')
ylabel('DCI')
```

```
title('Condition Intelligent Weighted Slope Model - (Comparison of
Residuals = DCI) - ' + string(AssetType)) %Single Asset Type
% title('Condition Intelligent Weighted Slope Model - (Comparison of
Residuals = DCI) - ' + string(MultiAsset)) %MultiAsset    "Formed
Metal" "Formed Metal - Metal Standing Seam" "Preformed Metal - Metal
Panel" "Preformed Metal"


legend([plot1 plot21 plot22 plot23], {'BUILDER DCI', 'Good Forecast
DCI', 'Repair Forecast DCI', 'Replace Forecast DCI'}, 'Location',
'northwest' )
hold off
%% Calculate fit parameters


CI_fcst(isnan(CI_fcst)) = 0; %NAN's to zeros for correlation calc
R2_fcst2 = corr(pred3(:,2),CI_fcst(1:length(pred3), 3))^2
%R2_fcst2 = corr(pred3(1:26,2),CI_fcst(1:26,3))^2
RMSE_fcst2 = sqrt(immse(pred3(:,2),CI_fcst(1:length(pred3),3)))
%RMSE_fcst2 = sqrt(immse(pred3(1:26,2),CI_fcst(1:26,3)))

Tknn.Expected_Rating(isnan(Tknn.Expected_Rating)) = 0;
Tknn.Comp_Rating(isnan(Tknn.Comp_Rating)) = 0;
R2_bldr = corr(Tknn.Expected_Rating, Tknn.Comp_Rating)^2
%R2_bldr = corr(Tknn.Expected_Rating(1:26), Tknn.Comp_Rating(1:26))^2
RMSE_bldr = sqrt(immse(Tknn.Expected_Rating(:),Tknn.Comp_Rating(:)))
%RMSE_bldr =
sqrt(immse(Tknn.Expected_Rating(1:26),Tknn.Comp_Rating(1:26)))
%rmse=sqrt(mean((y(:)-yhat(:)).^2));
%rmse = sqrt(immse(scores, dates));
```

Appendix C

Code for KNN and Ensembles:

```
%% Ensemble Forecast: Asset Degradation Model Slope + KNN
% Slope = Average slope-based degradation for each time-step (year/age)
% KNN = Nearest neighbor search for assets with

clear, clc

%% Load data
%Base = 'ALL'
%Tknn = readtable([Base, '_B30.csv'], 'HeaderLines',0);  %Import
filtered BUILDER data
Tknn = readtable('Roofing_Only.csv', 'HeaderLines',0);  %Import
filtered BUILDER data

%% Select Roof Type(s) from Tknn Section_Subtype

AssetTypes = unique(Tknn.Section_Subtype)   %lists all Asset Type
OPTIONS

%% Perform simple kNN (non-exclusive OCI at each Age)
AssetType = 'Single Ply Membrane' %Use (1) of the OPTIONS in line above
    Tknn = Tknn((Tknn.Section_Subtype == string(AssetType)), :);
%filter out asset types that are NOT desired
% MultiAsset = 'Metal Roof'   %    "Formed Metal" "Formed Metal - Metal
Standing Seam" "Preformed Metal - Metal Panel" "Preformed Metal"
%     Tknn = Tknn((Tknn.Section_Subtype == "Formed Metal"   |
Tknn.Section_Subtype == "Formed Metal - Metal Standing Seam"   |
Tknn.Section_Subtype == "Preformed Metal - Metal Panel"   |
Tknn.Section_Subtype == "Preformed Metal"), :);  %include multiple
asset types

%%     NOT USED
%      if AssetType == 'Single Ply Membrane'
%          SVC = 30
%          elseif AssetType == 'Asphalt Shingles'
%          SVC = 25
%          elseif AssetType == 'Built-Up'
%          SVC = 30
%          elseif AssetType == 'Modified Bitumen'
%          SVC = 30
%      else MultiAsset == 'Metal Roof'
%          SVC = 50
%      end

ct=unique(Tknn.SEC_ID);
length(ct)
%% Create occ4S - Count asset entries & take highest #of
entries/inspections for occ4s
%Add asset unique ID
Tknn.Site_Name = string(Tknn.Site_Name);
```

```
Tknn.SEC_ID = string(Tknn.SEC_ID);
ID = Tknn.Site_Name + '-' + Tknn.SEC_ID;
Tknn = addvars(Tknn, ID);

% Convert data format into HztConcat ID, Age 1, CI 1, Age 2, CI 2,Age
2, CI 2, etc.
Tknn3 = table(Tknn.ID, Tknn.Section_Subtype, Tknn.AgeOffset,
Tknn.Comp_Rating);
Tknn3.Properties.VariableNames = {'ID' 'Section_Subtype' 'AgeOffset'
'Comp_Rating'};
 Tknn4 = table2array(Tknn3 (: , 3:width(Tknn3))); % Create index vector
for first inspection
    Tknn4(isnan(Tknn4))=-999;  % Create index vector for first
inspection
Tknn3 = table(Tknn3.ID, Tknn3.Section_Subtype, Tknn4(:,1), Tknn4(:,2));
Tknn3.Properties.VariableNames = {'ID' 'Section_Subtype' 'AgeOffset'
'Comp_Rating'};

% Count Entries/Inspections
unq = unique(Tknn3.ID,'stable');
occ = cellfun(@(x) sum(ismember(Tknn3.ID,x)),unq,'un',0);
% occ(isnan(occ)) = 0;
occ = [unq occ];
occ2 = str2double(occ (:, 2));
mx = max(occ2(:,:))                    %I ADDED THIS LINE TO INCREASE
NUMBER OF INSPECTIONS!!!!!!!

test = occ2 == mx; %(CHANGE THIS VALUE FOR MAX # OF INSPECTIONS)
occ4 = occ((test), :);


Tknn3a = unstack(Tknn3, 'Comp_Rating', 'AgeOffset',
'AggregationFunction', @mean);
 Tknn3a = table2array(Tknn3a (: , 3:width(Tknn3a))); % Create index
vector for first inspection
    Tknn3a(isnan(Tknn3a))=0;  % Create index vector for first
inspection

Test2 = [] %width = #inspections x2 & length = #Unique assets (Tknn3a
length)
for i = 1:size(Tknn3a, 1) ;
   [o, temp] = find(Tknn3a(i,:) ~=0) ; %find "o" column value = Age
   temp2 = nonzeros(Tknn3a(i,:))'; %grab CI @ assessment

  temp = temp' ;
  temp2 = temp2' ;
    j = size(temp, 2)-size(temp2, 2) ;
   x = [temp(:) temp2(:)]' ;
   x=x(:)' ;
   x = [x zeros(1, (2*mx)-size(x,2))] ; %change this line to (2*mx) in
order to increase number of inspections

   Test2 = vertcat(Test2, x);
end
```

```
    %sum(Test2 (:,9:10)) %Check for inspections in last columns to
ensure inspection values are present...should be >0
% Test2 = Test2(:,1:8); %Uncomment this line in order to increase
number of inspections


%% Create Test 3-n (n = # of inspections)
Test3 = Test2(:, 8) > 0; %select ONLY assets with (4)
inspections...=column #8 (must change this if more/less inspections
exist)
Test3 = Test2(Test3,:);
Test3 (Test3 ==0) = nan ;
target = Test3;

Test4 = [Test2(:,1:4); Test2(:,3:6); Test2(:,5:8)];
Test4a = Test4(:, 4) > 0; %select ONLY assets with inspections
Test4 = Test4(Test4a,:);

Test5 = [Test3(:,1:4); Test3(:,3:6); Test3(:,5:8)];
Test5a = Test5(:, 4) > 0; %select ONLY assets with (4) inspections
Test5 = Test5(Test5a,:);

Test6 = []
if size(Test2,2)/2 == 4
    Test6 = vertcat(Test6, Test2(:,1:4), Test2(:,3:6), Test2(:,5:8));
    elseif size(Test2,2)/2 == 5
    Test6 = vertcat(Test6, Test2(:,1:4), Test2(:,3:6), Test2(:,5:8),
Test2(:,7:10));
    elseif size(Test2,2)/2 == 6
    Test6 = vertcat(Test6, Test2(:,1:4), Test2(:,3:6), Test2(:,5:8),
Test2(:,7:10), Test2(:,9:12));
    else size(Test2,2)/2 == 7
    Test6 = vertcat(Test6, Test2(:,1:4), Test2(:,3:6), Test2(:,5:8),
Test2(:,7:10), Test2(:,9:12), Test2(:,11:14));
end


Test2 (Test2 ==0) = nan ; %Change zeros to NAN
inputs = Test2;
Test4 (Test4 ==0) = nan ;
%inputs = Test4;
Test5 (Test5 ==0) = nan ;
%target = Test5;

Test6 (Test6 ==-999) = 0 ;
%     Test6(isnan(Test6)) = 0 ;
%     Test6a = Test6(:,4 ) ~= 0 ;
%     Test6 = Test6(Test6a, :)
%Test6 (Test6 ==0) = nan ;

%% Simple, two rule structure to define some number of NN that
satisfies a defined K
```

```
%  1)  If between two assessments, e.g. 5 and 10 years, there are
enough
%  exact matches, use as many as you have.  So, if we have a occ4 that
%  has assessments at age 5 and 10, and K=4, then there need to be at
least
%  4 samples that have assessments at age 5 and 10 to avoid moving to
rule
%  2.

% 2) If there are not enough samples to satify K, then the knnsearch is
% used to achieve a suitable sample size.

% Could add another kNN search after the first "else" to account for
% another assessment.
K = 6 % Number of nearest neighbors, 1 greater than we want, because
MATLAB finds the year we're forecasting
z = 4 % Select the Target Asset that you wish to forecast values for by
entering its row# from the 'target' matrix
SlopeVals = readmatrix([AssetType, '_SlopeFcsts.csv'],
'HeaderLines',0);  %Import filtered BUILDER data
SlopeVals100 = readmatrix([AssetType, '_SlopeFcsts_100.csv'],
'HeaderLines',0);  %Import filtered BUILDER data
SlopeVals80 = readmatrix([AssetType, '_SlopeFcsts_80.csv'],
'HeaderLines',0);  %Import filtered BUILDER data
SlopeVals60 = readmatrix([AssetType, '_SlopeFcsts_60.csv'],
'HeaderLines',0);  %Import filtered BUILDER data

% Multi-Asset (Metal Roof) File Import
% SlopeVals = readmatrix([MultiAsset, '_SlopeFcsts.csv'],
'HeaderLines',0);  %Import filtered BUILDER data
% SlopeVals100 = readmatrix([MultiAsset, '_SlopeFcsts_100.csv'],
'HeaderLines',0);  %Import filtered BUILDER data
% SlopeVals80 = readmatrix([MultiAsset, '_SlopeFcsts_80.csv'],
'HeaderLines',0);  %Import filtered BUILDER data
% SlopeVals60 = readmatrix([MultiAsset, '_SlopeFcsts_60.csv'],
'HeaderLines',0);  %Import filtered BUILDER data
%

    target1 = target(z,:);
if   sum(inputs(:,1) == target1(1,1) & inputs(:,3) == target1(1,3))>= K
     q = inputs(inputs(:,1) == target1(1,1) & inputs(:,3) ==
target1(1,3),:);
else q = inputs(knnsearch(inputs(:,3),target1(:,3),'K',K),:);
end



%% Calculated the slope and make a prediction

% Slope y2-y1/x2-x1...accounts for assesment age difference
slope = (q(:,4)-q(:,2))./(q(:,3)-q(:,1));
slope = mean(slope);
prediction = slope*(target1(1,3)-target1(1,1))+target1(1,2);
```

```
figure()
 v = [0 80; 0 100; length(SlopeVals) 100; length(SlopeVals) 80];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[0 1 0], 'FaceAlpha',(.5))
    yline(80, ":k",'GOOD') %Plus 20 CI bound
 v = [0 60; 0 80; length(SlopeVals) 80; length(SlopeVals) 60];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[1 1 0], 'FaceAlpha',(.5))
    yline(60, ":k",'REPAIR') %Plus 20 CI bound
 v = [0 0; 0 60; length(SlopeVals) 60; length(SlopeVals) 0];
    f = [1 2 3 4];
    patch('Faces',f,'Vertices',v,'FaceColor',[1 0 0], 'FaceAlpha',(.5))
    yline(0, ":k",'REPLACE') %Plus 20 CI bound

 hold on

    % for i =1:size(q,1)    %This for-loop plots the (K or more number
of) assets used from the search space
    % plot([q(i,1) q(i,3)],[q(i,2) q(i,4)],'r--*')
    % hold on
    % end

plot([target1(1,1) target1(1,3)],[target1(1,2) target1(1,4)],'b-
*','LineWidth',2)
plot(target1(1,3),prediction,'k-*')
xlim([min(15) max(35)]) %This automatically sets x-axis limits
% xlim([0 max(q(:,3))]) %This automatically sets x-axis limits
ylim([0 100]) %This automatically sets x-axis limits

%% Error
error = target1(1,4)-prediction


%% prediction of second assessment

if   sum(inputs(:,3) == target1(1,3) & inputs(:,5) == target1(1,5))>= K
     q2 = inputs(inputs(:,3) == target1(1,3) & inputs(:,5) ==
target1(1,5),:);
else q2 = inputs(knnsearch(inputs(:,5),target1(:,5),'K',K),:);
end

%% Calculated the slope and make a prediction

% Slope y2-y1/x2-x1...accounts for assesment age difference
slope2 = (q2(:,6)-q2(:,4))./(q2(:,5)-q2(:,3));
slope2 = mean(slope2);
prediction2 = slope2*(target1(1,5)-target1(1,3))+target1(1,4);

    % for i =1:size(q2,1)    %This for-loop plots the (K or more number
of) assets used from the search space
    % plot([q2(i,3) q2(i,5)],[q2(i,4) q2(i,6)],'r--*')
    % hold on
```

```
    % end

plot([target1(1,3) target1(1,5)],[target1(1,4) target1(1,6)],'b-
*','LineWidth',2)
plot(target1(1,5),prediction2,'k-*')
xlim([0 max(q2(:,5))]) %This automatically sets x-axis limits
ylim([0 100]) %This automatically sets x-axis limits

%% Error
error2 = target1(1,6)-prediction2

%% prediction of third assessment

if   sum(inputs(:,5) == target1(1,5) & inputs(:,7) == target1(1,7))>= K
     q3 = inputs(inputs(:,5) == target1(1,5) & inputs(:,7) ==
target1(1,7),:);
else q3 = inputs(knnsearch(inputs(:,7),target1(:,7),'K',K),:);
end

%% Calculated the slope and make a prediction

% Slope y2-y1/x2-x1...accounts for assesment age difference
slope3 = (q3(:,8)-q3(:,6))./(q3(:,7)-q3(:,5));
slope3 = mean(slope3);
prediction3 = slope3*(target1(1,7)-target1(1,5))+target1(1,6);

    % for i =1:size(q3,1)    %This for-loop plots the (K or more number
of) assets used from the search space
    % plot1 = plot([q3(i,5) q3(i,7)],[q3(i,6) q3(i,8)],'r--*')
    % hold on
    % end

plot2 = plot([target1(1,5) target1(1,7)],[target1(1,6)
target1(1,8)],'b-*','LineWidth',2)
plot3 = plot(target1(1,7),prediction3,'k-*')
xlim([0 max(q3(:,7))+6]) %This automatically sets x-axis limits
% xlim([0 55]) %This automatically sets x-axis limits
ylim([0 100]) %This automatically sets x-axis limits

%title('Nearest Neighbor Forecast - ' + string(AssetType)) %Single
Asset Type
%title('Nearest Neighbor Forecast - ' + string(MultiAsset)) %MultiAsset
"Formed Metal" "Formed Metal - Metal Standing Seam" "Preformed Metal -
Metal Panel" "Preformed Metal"

legend([plot2 plot3], {'Target', 'Prediction'}, 'Location',
'southwest')
%legend([plot1 plot2 plot3], {'OCI', 'Target', 'Prediction'},
'Location',% 'southwest')  %this legend provides red markers for K# of
assets used from the search space


%% Error
```

```
error3 = target1(1,8)-prediction3



%% Slope Forecast Ensemble Starts HERE

%% prediction of fourth assessment (+1)

if   sum(Test6(:,1) == target1(1,7)+1)>= K % sum(Test6(:,1) ==
target1(1,7)+1 & Test6(:,3) < target1(1,8))>= K ... Search for only
inspections in the same year and CI's of lower value then previous
inspection
     q4 = Test6(Test6(:,1) == target1(1,7)+1 , :);
else q4 = Test6(knnsearch(Test6(:,1),Test6(:,3),'K',K),:);
end

q4(isnan(q4)) = 0 ;
q4a = q4(:,4 ) ~= 0 ;
q4 = q4(q4a, :);


%% Calculated the slope and make a prediction

% Slope y2-y1/x2-x1...accounts for assesment age difference
slope4 = (q4(:,4)-q4(:,2))./(q4(:,3)-q4(:,1));
slope4 = mean(slope4);
prediction4 = target1(1,8) + (slope4*(1));
plot3 = plot(target1(1,7)+1, prediction4,'k-*')

%% prediction of fifth assessment (+2)

if   sum(Test6(:,1) == target1(1,7)+2)>= K
     q5 = Test6(Test6(:,1) == target1(1,7)+2 , :);
else q5 = Test6(knnsearch(Test6(:,1),Test6(:,3),'K',K),:);
end

q5(isnan(q5)) = 0 ;
q5a = q5(:,4 ) ~= 0 ;
q5 = q5(q5a, :);


%% Calculated the slope and make a prediction

% Slope y2-y1/x2-x1...accounts for assesment age difference
slope5 = (q5(:,4)-q5(:,2))./(q5(:,3)-q5(:,1));
slope5 = mean(slope5);
prediction5 = prediction4 + (slope5*(1));
plot3 = plot(target1(1,7)+2, prediction5,'k-*')

%% prediction of Sixth assessment (+3)

if   sum(Test6(:,1) == target1(1,7)+3)>= K
```

```
     q6 = Test6(Test6(:,1) == target1(1,7)+3 , :);
else q6 = Test6(knnsearch(Test6(:,1),Test6(:,3),'K',K),:);
end

q6(isnan(q6)) = 0 ;
q6a = q6(:,4 ) ~= 0 ;
q6 = q6(q6a, :);

%% Calculated the slope and make a prediction

% Slope y2-y1/x2-x1...accounts for assesment age difference
slope6 = (q6(:,4)-q6(:,2))./(q6(:,3)-q6(:,1));
slope6 = mean(slope6);
prediction6 = prediction5 + (slope6*(1));
plot3 = plot(target1(1,7)+3, prediction6,'k-*')

%% prediction of Seventh assessment (+4)

if   sum(Test6(:,1) == target1(1,7)+4)>= K
     q7 = Test6(Test6(:,1) == target1(1,7)+4 , :);
else q7 = Test6(knnsearch(Test6(:,1),Test6(:,3),'K',K),:);
end

q7(isnan(q7)) = 0 ;
q7a = q7(:,4 ) ~= 0 ;
q7 = q7(q7a, :);

%% Calculated the slope and make a prediction

% Slope y2-y1/x2-x1...accounts for assesment age difference
slope7 = (q7(:,4)-q7(:,2))./(q7(:,3)-q7(:,1));
slope7 = mean(slope7);
prediction7 = prediction6 + (slope7*(1));
plot3 = plot(target1(1,7)+4, prediction7,'k-*')

%% prediction of Eigth assessment (+5)

    if   sum(Test6(:,1) == target1(1,7)+5)>= K
         q8 = Test6(Test6(:,1) == target1(1,7)+5 , :);
    else q8 = Test6(knnsearch(Test6(:,1),Test6(:,3),'K',K),:);
    end

    q8(isnan(q8)) = 0 ;
    q8a = q8(:,4 ) ~= 0 ;
    q8 = q8(q8a, :)

%% Calculated the slope and make a prediction

%Slope y2-y1/x2-x1...accounts for assesment age difference
    slope8 = (q8(:,4)-q8(:,2))./(q8(:,3)-q8(:,1));
    slope8 = mean(slope8);
    prediction8 = prediction7 + (slope8*(1));
    plot3 = plot(target1(1,7)+5, prediction8,'k-*')
```

```
%% Add Slope Forecast to Out-year Forecast Predictions

age0 = target1(1,7); % prediction4
age1 = target1(1,7)+1; % prediction4
age2 = target1(1,7)+2; % prediction5
age3 = target1(1,7)+3; % prediction6
age4 = target1(1,7)+4; % prediction7
age5 = target1(1,7)+5; % prediction8

%% Adjust Slope and add to Predicted values of KNN+ 1-5

hold on
CI2_fcstSlope1 = [min(SlopeVals(:,1)) : max(SlopeVals(:,1))]'; %Slope-
base model: starts at 100=CI then subtracts slope values as age
increases
CI2_fcstSlope1 = horzcat(CI2_fcstSlope1, SlopeVals(:,2),
SlopeVals100(:,2), SlopeVals80(:,2), SlopeVals60(:,2));
CI2_fcstSlope1(isnan(CI2_fcstSlope1)) = 0;

pred = []
    for j = age0 : length(SlopeVals(:,1))-1;
        if target1(1,8) > 80
            p = target1(1,8) + sum(CI2_fcstSlope1(target1(1,7)+1:j,
3));
            temp3 = [j p];
        elseif target1(1,8) > 60
            p = target1(1,8) + sum(CI2_fcstSlope1(target1(1,7)+1:j,
4));
            temp3 = [j p];
        else
            p = target1(1,8) + sum(CI2_fcstSlope1(target1(1,7)+1:j,
5));
            temp3 = [j p];
        end
        pred = vertcat(pred, temp3);
    end
CIslope = scatter(pred(:,1), pred(:,2),'m+','LineWidth',3) %MEDIAN
forecast
    for iiii = 1:(length(pred)-1)
        plot4 = plot([pred(iiii,1) pred(iiii+1,1)], [pred(iiii,2)
pred(iiii+1,2)],'m','LineWidth',2)
    end

    %This section of code provides "Condition Intelligent" forecasts
for each KNN forecast
    % pred2 = []
    %     for j = age1 : length(SlopeVals(:,1))-1;
    %         if pred > 80
    %             p = prediction4 +
sum(CI2_fcstSlope1(target1(1,7)+2:j, 3));
    %             temp3 = [j p];
    %         elseif pred > 60
```

```
    %             p = prediction4 +
sum(CI2_fcstSlope1(target1(1,7)+2:j, 4));
    %             temp3 = [j p];
    %         else
    %             p = prediction4 +
sum(CI2_fcstSlope1(target1(1,7)+2:j, 5));
    %             temp3 = [j p];
    %         end
    %         pred2 = vertcat(pred2, temp3);
    %     end
    % CIslope = scatter(pred2(:,1), pred2(:,2),'m+','LineWidth',3)
%MEDIAN forecast
    %     for iiii = 1:(length(pred2)-1)
    %         plot([pred2(iiii,1) pred2(iiii+1,1)], [pred2(iiii,2)
pred2(iiii+1,2)],'m','LineWidth',2)
    %     end
    %
    % pred3 = []
    %     for j = age2 : length(SlopeVals(:,1))-1;
    %         if pred2 > 80
    %             p = prediction5 +
sum(CI2_fcstSlope1(target1(1,7)+3:j, 3));
    %             temp3 = [j p];
    %         elseif pred2 > 60
    %             p = prediction5 +
sum(CI2_fcstSlope1(target1(1,7)+3:j, 4));
    %             temp3 = [j p];
    %         else
    %             p = prediction5 +
sum(CI2_fcstSlope1(target1(1,7)+3:j, 5));
    %             temp3 = [j p];
    %         end
    %         pred3 = vertcat(pred3, temp3);
    %     end
    % CIslope = scatter(pred3(:,1), pred3(:,2),'m+','LineWidth',3)
%MEDIAN forecast
    %     for iiii = 1:(length(pred3)-1)
    %         plot([pred3(iiii,1) pred3(iiii+1,1)], [pred3(iiii,2)
pred3(iiii+1,2)],'m','LineWidth',2)
    %     end
    %
    % pred4 = []
    %     for j = age3 : length(SlopeVals(:,1))-1;
    %         if pred3 > 80
    %             p = prediction6 +
sum(CI2_fcstSlope1(target1(1,7)+4:j, 3));
    %             temp3 = [j p];
    %         elseif pred3 > 60
    %             p = prediction6 +
sum(CI2_fcstSlope1(target1(1,7)+4:j, 4));
    %             temp3 = [j p];
    %         else
    %             p = prediction6 +
sum(CI2_fcstSlope1(target1(1,7)+4:j, 5));
    %             temp3 = [j p];
```

```
    %         end
    %         pred4 = vertcat(pred4, temp3);
    %     end
    % CIslope = scatter(pred4(:,1), pred4(:,2),'m+','LineWidth',3)
%MEDIAN forecast
    %     for iiii = 1:(length(pred4)-1)
    %         plot([pred4(iiii,1) pred4(iiii+1,1)], [pred4(iiii,2)
pred4(iiii+1,2)],'m','LineWidth',2)
    %     end
    %
    %
    % pred5 = []
    %     for j = age4 : length(SlopeVals(:,1))-1;
    %         if pred4 > 80
    %             p = prediction7 +
sum(CI2_fcstSlope1(target1(1,7)+5:j, 3));
    %             temp3 = [j p];
    %         elseif pred4 > 60
    %             p = prediction7 +
sum(CI2_fcstSlope1(target1(1,7)+5:j, 4));
    %             temp3 = [j p];
    %         else
    %             p = prediction7 +
sum(CI2_fcstSlope1(target1(1,7)+5:j, 5));
    %             temp3 = [j p];
    %         end
    %         pred5 = vertcat(pred5, temp3);
    %     end
    % CIslope = scatter(pred5(:,1), pred5(:,2),'m+','LineWidth',3)
%MEDIAN forecast
    %     for iiii = 1:(length(pred5)-1)
    %         plot([pred5(iiii,1) pred5(iiii+1,1)], [pred5(iiii,2)
pred5(iiii+1,2)],'m','LineWidth',2)
    %     end
    %
    %

%% STOP!!!!!!!!!!!!!!!!!!!




pred = []
    for j = age0 : length(SlopeVals(:,1))-1;
        p = target1(1,8) + sum(CI2_fcstSlope1(target1(1,7)+1:j, 2));
        temp3 = [j p];
        pred = vertcat(pred, temp3);
    end
CIslope = scatter(pred(:,1), pred(:,2),'c+','LineWidth',3) %MEDIAN
forecast
    for iiii = 1:(length(pred)-1)
        plot5 = plot([pred(iiii,1) pred(iiii+1,1)], [pred(iiii,2)
pred(iiii+1,2)],'c','LineWidth',2)
```

```
    end

    %This section of code provides "Slope-weighted" forecasts for each
KNN forecast
    % pred2 = []
    %     for j = age1 : length(SlopeVals(:,1))-1;
    %         p = prediction4 + sum(CI2_fcstSlope1(target1(1,7)+2:j,
2));
    %         temp3 = [j p];
    %         pred2 = vertcat(pred2, temp3);
    %     end
    % CIslope = scatter(pred2(:,1), pred2(:,2),'c+','LineWidth',3)
%MEDIAN forecast
    %     for iiii = 1:(length(pred2)-1)
    %         plot([pred2(iiii,1) pred2(iiii+1,1)], [pred2(iiii,2)
pred2(iiii+1,2)],'c','LineWidth',2)
    %     end
    %
    %
    % pred3 = []
    %     for j = age2 : length(SlopeVals(:,1))-1;
    %         p = prediction5 + sum(CI2_fcstSlope1(target1(1,7)+3:j,
2));
    %         temp3 = [j p];
    %         pred3 = vertcat(pred3, temp3);
    %     end
    % CIslope = scatter(pred3(:,1), pred3(:,2),'c+','LineWidth',3)
%MEDIAN forecast
    %     for iiii = 1:(length(pred3)-1)
    %         plot([pred3(iiii,1) pred3(iiii+1,1)], [pred3(iiii,2)
pred3(iiii+1,2)],'c','LineWidth',2)
    %     end
    %
    %     pred4 = []
    %     for j = age3 : length(SlopeVals(:,1))-1;
    %         p = prediction6 + sum(CI2_fcstSlope1(target1(1,7)+4:j,
2));
    %         temp3 = [j p];
    %         pred4 = vertcat(pred4, temp3);
    %     end
    % CIslope = scatter(pred4(:,1), pred4(:,2),'c+','LineWidth',3)
%MEDIAN forecast
    %     for iiii = 1:(length(pred4)-1)
    %         plot([pred4(iiii,1) pred4(iiii+1,1)], [pred4(iiii,2)
pred4(iiii+1,2)],'c','LineWidth',2)
    %     end
    %
    %      pred5 = []
    %     for j = age4 : length(SlopeVals(:,1))-1;
    %         p = prediction7 + sum(CI2_fcstSlope1(target1(1,7)+5:j,
2));
    %         temp3 = [j p];
    %         pred5 = vertcat(pred5, temp3);
    %     end
```

```
    % CIslope = scatter(pred5(:,1), pred5(:,2),'c+','LineWidth',3)
%MEDIAN forecast
    %     for iiii = 1:(length(pred5)-1)
    %        plot4 = plot([pred5(iiii,1) pred5(iiii+1,1)],
[pred5(iiii,2) pred5(iiii+1,2)],'c','LineWidth',2)
    %     end

% plot6 = plot([target1(1,7) target1(1,7)+1 target1(1,7)+2
target1(1,7)+3 target1(1,7)+4 target1(1,7)+5], [target1(1,8)
prediction4 prediction5 prediction6 prediction7 prediction8],'k-
*','LineWidth',2)
plot6 = plot([target1(1,7) target1(1,7)+1], [target1(1,8) prediction4
],'k-*','LineWidth',2)

hold on
        %  v = [0 80; 0 100; length(SlopeVals) 100; length(SlopeVals)
80];
        %     f = [1 2 3 4];
        %     patch('Faces',f,'Vertices',v,'FaceColor',[0 1 0],
'FaceAlpha',(.5))
        %     yline(80, ":k",'GOOD') %Plus 20 CI bound

% x2 = pred(1:6,1);
% curve1 = [target1(1,8) prediction4 prediction5 prediction6
prediction7 prediction8]';
% curve2 = pred(1:6,2);,
% plot(x2, curve1, 'k', 'LineWidth', 2);
% hold on;
% plot(x2, curve2, 'c', 'LineWidth', 2);
% x2 = [x2, fliplr(x2)];
% inBetween = [curve1, fliplr(curve2)];
% fill(x2, inBetween, 'k', 'FaceAlpha',(.25));
%
xlim([0 length(SlopeVals)]) %This automatically sets x-axis limits
%xlim([0 55]) %This automatically sets x-axis limits
%xlim([min(15) max(55)]) %This automatically sets x-axis limits
%xlim([min(5) max(30)]) %This automatically sets x-axis limits
ylim([0 100])
set(gca,'XTick',[0:5:length(SlopeVals)]) %This automatically sets x-
axis ticks
xlabel('Age')
ylabel('OCI')
title('Nearest Neighbor Forecast - ' + string(AssetType)) %Single Asset
Type
%title('Condition Intelligent Forecast - ' + string(AssetType)) %Single
Asset Type
% title('Nearest Neighbor Forecast - ' + string(MultiAsset))
%MultiAsset    "Formed Metal" "Formed Metal - Metal Standing Seam"
"Preformed Metal - Metal Panel" "Preformed Metal"

legend([plot2 plot3 plot4 plot5 plot6], {'Target', 'Prediction',
'Condition-Slope FCST', 'Weighted Slope FCST', 'KNN Bootstrap'},
'Location', 'southwest')
```

```
%legend([plot2 plot4 plot5 plot6], {'Target', 'Condition-Slope FCST',
'Weighted Slope FCST', 'KNN Bootstrap'}, 'Location', 'southwest')
%legend([plot2 plot6], {'Target', 'KNN Bootstrap'}, 'Location',
'southwest')
%legend([plot1 plot2 plot3 plot4], {'OCI', 'Target', 'Prediction',
'Slope FCST'}, 'Location', 'southwest') %Includes red markers for K# of
assets used
legend([plot2 plot6 plot4 plot5], {'Inspection', 'Nearest Neighbor',
'Condition-Slope FCST', 'Weighted Slope FCST'}, 'Location',
'southwest')
hold off



%% Output Slope Model to .mat file for use in ensemble

% writematrix(pred, [AssetType, '_EnsembleFcst.csv'], 'Delimiter',
',');    %output KNN Forecast values to a csv file
```

Bibliography

Alley, S. L., Valencia, V. V., Thal, A. E., and White, E. D. (2017). "Probabilistic Assessment of Failure for United States Air Force Building Systems." *Journal of Performance of Constructed Facilities*, 31(5), 04017088.

ASCE. (2017). "ASCE's 2017 American Infrastructure Report Card | GPA: D+." *ASCE's 2017 Infrastructure Report Card*, <https://www.infrastructurereportcard.org> (Dec. 8, 2020).

ASCE. (2020). *Changing Infrastructure Equation: Using Asset Management to Optimize Investments*. Changing Infrastructure Equation, ASCE.

Bhamare, D. K., Rathod, M. K., and Banerjee, J. (2019). "Passive cooling techniques for building and their applicability in different climatic zones—The state of art." *Energy and Buildings*, 198, 467–490.

"BUILDER Sustainment Management System: Fact Sheet." (2012). U.S. Army.

Cash, C. G. (1997). "The Relative Durability of Low-Slope Roofing." Roof Consultant Institute.

Cash, C. G. (2006). "2005 Roofing Industry Durability and Cost Survey." Roof Consultant Institute.

Charette, R. P., and Marshall, H. E. (1999). *UNIFORMAT II elemental classification for building specifications, cost estimating, and cost analysis*. National Institute of Standards and Technology, Gaithersburg, MD, NIST IR 6389.

Delorit, J. D., Schuldt, S. J., and Chini, C. M. (2020). "Evaluating an Adaptive Management Strategy for Organizational Energy Use under Climate Uncertainty." 35.

DoD Comptroller. (2019). "Department of Defense Budget Fiscal Year 2020." DoD.

DOD Needs to Assess Risk and Provide Guidance on Use of Climate Projections in Installation Master Plans and Facilities Designs Climate Resilience. (2019). Climate Resilience, Report to GAO.

DoD Real Property Portfolio Office. (2018). *Operations and Maintenance Overview Fiscal Year 2019 Budget Estimates*. Office of the Under Secretary of Defense (Comptroller)/Chief FInancial Officer.

Ducman, V., Škapin, A. S., Radeka, M., and Ranogajec, J. (2011). "Frost resistance of clay roofing tiles: Case study." *Ceramics International*, 37(1), 85–91.

Elsevier. (2021). "Journal of Building Engineering." *Journal of Building Engineering*, <https://www.journals.elsevier.com/journal-of-building-engineering> (Feb. 8, 2021).

ERDC Mission Statement. (n.d.). U.S. Army.

GAO. (2011). *Federal Efforts to Provide Information Could Help Government Decision Making*. CLIMATE CHANGE ADAPTATION, Report to GAO.

Grant, A., Ries, R., and Kibert, C. (2014). “Life Cycle Assessment and Service Life Prediction: A Case Study of Building Envelope Materials.” *Journal of Industrial Ecology*, 18(2), 187–200.

Grant, A., Ries, R., and Thompson, C. (2016). “Quantitative approaches in life cycle assessment—part 1—descriptive statistics and factor analysis.” *The International Journal of Life Cycle Assessment*, 21(6), 903–911.

Grussing, M. N. (2014). “Life Cycle Asset Management Methodologies for Buildings.” *Journal of Infrastructure Systems*, 20(1), 04013007.

Grussing, M. N., and Liu, L. Y. (2014). “Knowledge-Based Optimization of Building Maintenance, Repair, and Renovation Activities to Improve Facility Life Cycle Investments.” *Journal of Performance of Constructed Facilities*, 28(3), 539–548.

Grussing, M. N., Liu, L. Y., Uzarski, D. R., El-Rayes, K., and El-Gohary, N. (2016). “Discrete Markov Approach for Building Component Condition, Reliability, and Service-Life Prediction Modeling.” *Journal of Performance of Constructed Facilities*, 30(5), 04016015.

Grussing, M. N., Uzarski, D. R., and Marrano, L. R. (2006). “Condition and Reliability Prediction Models Using the Weibull Probability Distribution.” *Applications of Advanced Technology in Transportation*, American Society of Civil Engineers, Chicago, Illinois, United States, 19–24.

Habibi, S., Obonyo, E. A., and Memari, A. M. (2020). “Design and development of energy efficient re-roofing solutions.” *Renewable Energy*, 151, 1209–1219.

Harkness, E. L., and Hassanain, M. A. (2001). “Hailstorm & Heat Stress Damage of Membrane Roofing in Saudi Arabia, North America & Australia: Guidelines for Service life.” *Architectural Science Review*, Taylor & Francis, 44(3), 211–220.

Hodges, C. P. (1999). “EFFECTIVE ROOF MANAGEMENT - UNDERSTANDING THE LIFE CYCLE OF YOUR ROOF SYSTEMS.” 10.

Karl, T. R., Nicholls, N., and Ghazi, A. (2013). *Weather and Climate Extremes: Changes, Variations and a Perspective from the Insurance Industry*. Springer Science & Business Media.

Lounis, Z., Vanier, D. J., Lacasse, M. A., and Kyle, B. (1999). “DECISION-SUPPORT SYSTEM FOR SERVICE LIFE ASSET MANAGEMENT: THE BELCAM PROJECT.” 12.

Moon-Cronk, T. (2018). “DoD’s Underfunded Maintenance Backlog Exceeds $116 Billion, Official Says.” *U.S. DEPARTMENT OF DEFENSE*, DoD News,

<https://www.defense.gov/Explore/News/Article/Article/1497582/dods-underfunded-maintenance-backlog-exceeds-116-billion-official-says/> (Dec. 8, 2020).

"Roofing Systems | WBDG - Whole Building Design Guide." (2016). <https://www.wbdg.org/guides-specifications/building-envelope-design-guide/roofing-systems> (Dec. 12, 2019).

SAME. (2021). "About SAME - The Society of American Military Engineers." *About SAME*, <https://www.same.org/discover-same/about-same> (Mar. 1, 2021).

Shahata, K. F. (2013). "Decision-Support Framework for Integrated Asset Management of Major Municipal Infrastructure." 286.

Sitzabee, W., and Harnly, M. (2013). "A Strategic Assessment of Infrastructure Asset-Management Modeling." *Air & Space Power Journal*, (November-December 2013), 24.

Sjostrom, C. (2004). *Durability of Building Materials and Components 7: Proceedings of the Seventh International Conference*. Routledge.

Somers, W., and Bates, D. I. (2019). "Sustainment Management System enables proactive asset management of CE built infrastructure." *Air Force Civil Engineer Center Operations Directorate Asset Visibility Division*, 4.

Stupak, J. M. (2018). "Economic Impact of Infrastructure Investment." 19.

Tonoli, G. H. D., Santos, S. F., Savastano, H., Delvasto, S., Mejía de Gutiérrez, R., and Lopez de Murphy, M. del M. (2011). "Effects of natural weathering on microstructure and mineral composition of cementitious roofing tiles reinforced with fique fibre." *Cement and Concrete Composites*, 33(2), 225–232.

Uzarski, D., Grussing, M., and Mehnert, B. (2019). *Knowledge-based condition assessment reference manual for building component-sections : for use with BUILDER™ and BuilderRED™ (v. 3 series)*. Engineer Research and Development Center (U.S.).

Uzarski, D. R. (1995). *Development of Condition Indexes for Building Exteriors*.

Vanier, D. (2001). "Why Industry Needs Asset Management Tools." *Journal of Computing in Civil Engineering*, 15.

Vanier, D. J., and Danylo, N. H. (1998). "Municipal Infrastructure Investment Planning: Asset Management." *Asset Management*, 15.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 074-0188*

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
25-03-2021	Graduate Thesis	Sept 2019 - March 2021

TITLE AND SUBTITLE

Data-Driven Asset Degradation Modeling:

An Enterprise-Wide Roof System Case Study

5a. CONTRACT NUMBER

5b. GRANT NUMBER AFIT2019-157

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Lamm, Kurt R, GS-13, USAF
Schuldt, Steven J, Lt Col, USAF
Delorit, Justin D, Maj, USAF

5d. PROJECT NUMBER 21V219

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/ENV)
2950 Hobson Way, Building 640
WPAFB OH 45433-8865

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENV-MS-21-242

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Mr. John (Dale) Clark
Air Force Civil Engineer Center (AFCEC/DS)
2261 Hughes Ave
JBSA Lackland, TX 78236-9853
john.clark.7@us.af.mil)

10. SPONSOR/MONITOR'S ACRONYM(S)

AFCEC

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

N/A

12. DISTRIBUTION/AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Organizations with large facility and infrastructure portfolios have used asset management databases for over ten years to collect and standardize asset condition data. Decision makers use this data to predict asset degradation and expected service life, enabling prioritized maintenance, repair, and renovation actions that reduce asset life-cycle costs and achieve organizational objectives. However, these asset condition forecasts are calculated using standardized, self-correcting distribution models that rely on continuous functions. This research presents four stepwise asset condition forecast models that utilize historical asset inspection data to improve prediction accuracy: (1) Slope, (2) Weighted Slope, (3) Condition-intelligent Weighted Slope, and (4) Nearest Neighbor. Model performance was evaluated against BUILDER SMS, the industry-standard asset management database, using data for five roof types on 8,549 facilities across 61 U.S. military bases within the Contiguous United States. The stepwise Weighted-slope model predicted asset degradation more accurately than BUILDER SMS 92% of the time. These results suggest that using historical condition data, alongside or in-place of manufacturer expected service life, may increase degradation and failure prediction accuracy. Additionally, the developed models are expected to improve prediction skills as data quantity increases over time. These results are expected to enable decision makers to achieve more accurate construction management and infrastructure investment objectives.

15. SUBJECT TERMS

Asset Management, BUILDER SMS, Roofing, Forecast Model, Stepwise

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Lt Col Steven J. Schuldt
U	U	U	UU	124	**19b. TELEPHONE NUMBER** *(Include area code)* (937) 255-6565 x4645 steven.schuldt@afit.edu