

EVALUATING BOX COMPRESSION STRENGTH (BCS) USING AN ARTIFICIAL
NEURAL NETWORK (ANN)

By

Juan Gu

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Packaging – Doctor of Philosophy

2025

ABSTRACT

Though box compression strength (BCS) is commonly used as a performance criterion for shipping containers, the state-of-the-art BCS estimation produces results within a broad range of values. In this study we implemented a new approach, artificial neural networks (ANN), to explore how much data may be needed for an ANN to reasonably predict compression strength, and how the ANN approach performs while facing variation that adversely impacts other modeling methodologies. An ANN model can be built by comprehensively adjusting four modeling factors that interact with each other to influence model accuracy and can be optimized by minimizing model MSE. Using both data available from the literature and a “synthetic” data set using idealized data based on the McKee equation, we find that model estimation accuracy remains limited by the uncertainty or error in the input parameters combined with uncertainty from the ANN process itself, and we produce an estimate for this impact. The population size to build an ANN model that can reasonably estimate BCS has been identified based on different data sets in this study.

Packaging design plays a crucial role in ensuring the protective performance of packages. Various factors must be considered to ensure package strength during the packaging design process. Understanding the relative importance of each influencing factor or design feature provides valuable insights for optimizing packaging material utilization. However, current methods such as testing and finite element analysis have limitations in evaluating the relative significance of these parameters. In response to these challenges, in this research, we applied different methods to comprehensively evaluate the relative importance of different packaging design features on a given packaging property. Using BCS as a representative packaging property, the relative importance of up to six BCS features (Edge Crush Test (ECT), Perimeter, Thickness, Depth, and Flexural Stiffness in both the machine and cross-machine directions) were evaluated. Four distinct ANN methods

were employed - Connection weights method, Gradient-based method, Permutation method, and SHAP values. These techniques were applied to two datasets: one comprising "synthetic" data based on the McKee formula and the other representing real-world scenarios. The reliability of these methods was assessed. Different input feature importance (FI) scores obtained from the four methods have been calculated and compared with theoretical BCS FI derived from the McKee formula. The BCS feature ranking result given by the synthetic data is verified by the theoretical feature importance ranking indicated by the McKee formula. Although box depth is considered to have zero importance in the McKee formula, the BCS feature importance ranking from the real dataset highlights its significance, aligning with buckling theory. The study gives an insight into the BCS feature importance evaluation using ANN approach and guides packaging design material and cost saving.

The ultimate objective of this research is to develop a comprehensive ANN model for predicting Box Compression Strength (BCS). To achieve this, we utilized a dataset encompassing a wide range of box dimensions commonly encountered in industrial applications. After applying multiple optimization methods to determine the optimal number of hidden neurons and further identifying the key factor values influencing the model, a generalized ANN model was trained. The trained ANN model can predict the BCS commonly used in the industrial applicable level with an error of 9.51%. The primary factor contributing to the high BCS error is the presence of boundary data points and the small sample size of the current data set. One possible strategy to improve ANN prediction accuracy is to continually expand the current dataset sample size using available resources. In essence, this study serves as a roadmap for forthcoming research endeavors seeking to leverage ANN techniques to tackle challenges and provide solutions within the corrugated industry.

COPYRIGHT BY
JUAN GU
2025

Dedicated to my parents and my husband.
Thank you for always believing in me.

ACKNOWLEDGEMENTS

I wish to extend my sincere gratitude to several individuals and organizations whose contributions have been instrumental in the completion of this dissertation. Firstly, I want to express profound gratitude to Dr. Euihark Lee. His mentorship, encouragement, and willingness to explore unconventional avenues of inquiry have been invaluable to my growth and development as a researcher. Secondly, I am immensely grateful to the Packaging Corporation of America for their generous provision of resources and data that underpinned this research. Without their support, this work would not have been possible. I extend my heartfelt thanks to Packaging Corporation of America for their invaluable insights into corrugated board compression in the industry. Their guidance has greatly enriched the depth of this study. I also acknowledge with gratitude the support and advice of Dr. Amin Joodaky, Dr. Qiang Yang, and Dr. Yan, whose contributions have significantly enhanced the quality of this research.

Special thanks are due to the School of Packaging for allowing me to begin my PhD journey and for providing the necessary infrastructure and equipment for this study. I am indebted to my colleagues for their invaluable suggestions in machine learning and for the cherished memories we have shared. I express my deepest appreciation to the professors at the School of Packaging for their wisdom, guidance, and unwavering support throughout this endeavor. Thank them for their remarkable contributions and support throughout this journey.

TABLE OF CONTENTS

CHAPTER 1: BACKGROUND	1
1.1 BOX COMPRESSION STRENGTH (BCS)	1
1.2 APPROACHES FOR BCS ESTIMATION	8
1.3 ARTIFICIAL NEURAL NETWORK (ANN)	15
1.4 APPLICATIONS OF ANNS	24
1.5 RESEARCH OVERVIEW	28
CHAPTER 2: A COMPARATIVE ANALYSIS OF ARTIFICIAL NEURAL NETWORK (ANN) ARCHITECTURES FOR BOX COMPRESSION STRENGTH ESTIMATION ...	31
2.1 INTRODUCTION	31
2.2 DATA SETS APPLIED	32
2.3 ANN KEY FACTORS INITIALIZATION	33
2.4 ANN AND MCKEE DATA SET	37
2.5 ANN AND AN IDEALIZED DATA SET	43
2.6 ANN AND A DATA SET WITH VARIATION	50
2.7 CONCLUSION	53
CHAPTER 3: EVALUATION OF PACKAGING DESIGN RELATIVE FEATURES IMPORTANCE USING ANN	55
3.1 INTRODUCTION	55
3.2 CURRENT METHODS FOR EVALUATING BCS FEATURE IMPORTANCE	55
3.3 ANN APPROACH FOR EVALUATING FEATURE IMPORTANCE	57
3.4 FLOW OF FEATURE IMPORTANCE EVALUATION USING ANN	64
3.5 CASE STUDY FOR FEATURE IMPORTANCE ANALYSIS	73
3.6 CONCLUSION	84
CHAPTER 4: BUILDING A GENERALIZED ANN MODEL TO EVALUATE BCS	86
4.1 INTRODUCTION	86
4.2 EXTRACT REAL DATA SET TO COVER THE MAJORITY OF BCS IN THE INDUSTRY	87
4.3 DETERMINATION OF HIDDEN LAYER NEURON SETTING	88
4.4 TRAINING ANN MODEL TO EVALUATE BCS IN THE INDUSTRY	89
4.5 CONCLUSION	92
CHAPTER 5: RESEARCH SUMMARY AND FUTURE RESEARCH	94
5.1 RESEARCH SUMMARY	94
5.2 FUTURE RESEARCH	97
BIBLIOGRAPHY	99

CHAPTER 1: BACKGROUND

This chapter focuses on the Box Compression Strength (BCS) of corrugated packaging using the Artificial Neural Network (ANN) approach. This chapter covers four main sections: BCS, existing approaches for BCS evaluation, ANN, and the application of ANN. The BCS section discusses the application of corrugated packaging, the reasons for its failure, and the factors influencing BCS. The section on existing methods for evaluating BCS introduces the shortcomings of each method. The ANN section covers the architecture and components of ANN, its working principles, and its various types. The application of ANN section introduces different fields involving ANN applications, with a particular emphasis on packaging.

1.1 BOX COMPRESSION STRENGTH (BCS)

In the packaging industry, evaluating packaging properties is essential to ensure the reliability of a package's utilization. Among various types of packaging, corrugated packages have gained significant popularity in the modern market. Due to the unique properties of corrugated paperboard, evaluating the properties of these packages has become a critical research topic. Given the diverse demands of the market, estimating the strength of corrugated boxes has become increasingly important. Box Compression Strength (BCS) is one of the most crucial parameters to consider for corrugated packages. Over the past 130 years, the compression strength of corrugated boxes has been extensively studied due to failures occurring during the shipping, distribution, and storage processes of various [1].

1.1.1 Ubiquitous corrugated box

Corrugated boxes are one type of shipping container that is widely used in the market currently. Corrugated boxes are made from paper, and machine-shaped from corrugated box boards with hollow structures. Since the corrugated box was first accepted by legal freight classification

organizations as containers for freight transportation, the application history of corrugated boxes and studying corrugated box dynamics has been over more than 100 years. Corrugated boxes are widely applied in various fields [2] because of their lightweight, low cost, ease of assembly and disassembly, good sealing performance, certain cushioning and anti-vibration ability, and easy recovery and waste treatment.

The most commonly used corrugated box structure is the Regular Slotted Container (RSC) due to its simplicity in production, formation, and ease of use. With the development of the economy, e-commerce has become increasingly popular. As e-commerce advances, the types of corrugated boxes have diversified. Various box structures are now being utilized in the market, as shown in Figure 1 [3].



Figure 1 Different structures of corrugated boxes used in the market

The utilization of corrugated boxes has become widespread across various countries. According to the 2015 Global Corrugated Packaging Market Overview report, based on data from the United

Nations, each person in the world uses packaging worth over USD 110 annually, significantly contributing to the expansion of the packaging industry [4]. The corrugated packaging industry is witnessing an incredible growth due to the increasing demand in packaging for food & beverages, personal and household care products, medicines, and other products. The booming e-commerce industry is playing a vital role in the adoption of corrugated packaging for consumer goods. Around 85% of corrugated packaging is used for shipping boxes where high protection is required. Moreover, the increasing popularity of corrugated retail display stands which are used for effective highlighting of products in retail stores is likely to contribute to the expansion of the corrugated packaging business. These factors are increasing the global production of corrugated boards. In 2017, as per International Corrugated Case Association (ICCA) over 240 billion square meters of corrugated boards were produced, where North America occupies 30% revenue share of the total production [4].

Furthermore, innovative solutions provided to key vendors for the adoption of corrugated packaging are also contributing to the growth of the global corrugated packaging market. For instance, International Paper Co. uses cellulose fibers in corrugated packaging which is mainly used for packaging textiles, construction material, paints and coatings, and other non-durable goods [4].

The global corrugated packaging market is segmented on the following basis: Slotted boxes, Telescope boxes, Folder boxes, Self-Erecting Boxes / Auto-Bottom Boxes, Bliss / Rigid Boxes, and Others (Mailing Boxes, Bin Boxes, Slide Boxes). Based on End-User, the global corrugated packaging market is segmented into Food & Beverages, Electronic Goods, Personal and Home Care Goods, Glassware & Ceramics, Healthcare & Pharmaceuticals, and Others (Textile, Chemical, Paper products). Based on Geography, the global corrugated packaging market is segmented into North America (U.S. & Canada); Latin America (Brazil, Mexico & the rest of Latin America);

Europe (the U.K., Germany, France, Italy, Spain, Poland, Sweden & the rest of Europe); Asia-Pacific (China, India, Japan, Singapore, South Korea, Australia, New Zealand, the rest of Asia); Middle East & Africa (GCC, South Africa, North Africa, the rest of Middle East and Africa) and Rest of World. All these markets have a large market size measured in USD billions and a production quantity measured in tonnes. Almost 80% of the volume of paper packaging used in the United States are corrugated boxes. A similar proportion of goods are transported using corrugated boxes. The goods are not only the goods from the distribution process to the end user but also the parts of goods brought to their assembly locations using corrugated boxes as well. Corrugated boxes protect the products during almost all the phases of the distribution process [1]. Corrugated boxes are one of the main types of delivery packages in China as well. 9.9 billion corrugated boxes have been used in China as reported in 2015 [5].

Due to a huge spike in the e-commerce segment, the corrugated packaging market is growing rapidly, as a consequence the global corrugated market is growing faster at the rate of 5.62% annually and it has been predicted to achieve \$386 billion in 2026, as reported by the Indian Pulp and Paper Technical Association [6]. E-commerce retail sales are continuing to surge, with estimates of around 20% annual growth in e-commerce trade in Europe. This will have a profound impact on packaging demand, especially in the corrugated industry as it represents 80% of demand in e-commerce. The corrugated Packaging Market size was valued at USD 70 billion in 2022 and is poised to depict a 4% CAGR through 2023-2032, on account of the burgeoning e-commerce sales worldwide [7].

As sustainable development becomes a global priority, corrugated packages are gaining popularity in packaging, reflecting the growing emphasis on sustainability throughout the value chain. Corrugated packages are easy to recycle, and the pulp and paper industry has already adapted to

converting these into new generations of container boards. Consumers prefer corrugated protective formats over polymer-based alternatives, such as expanded polystyrene (EPS) foams [6].

1.1.2 Reason of failure of corrugated packages

The failure of corrugated boxes can be influenced by both distribution and material factors. The actual BCS of corrugated will decrease over time due to various environmental factors, such as stacking height, the mass of the filled box, the number of layers packed high, types of pallets used and overhang, unitizing practice, and the number of pallets high packed at the storage, storage, and distribution time, and transportation circumstances [8].

The overhang has a significant influence on the BCS of corrugated boxed during the storage and shipping process. The major strength of a container is mostly derived from the corners [9] as demonstrated in Figure 2 below.

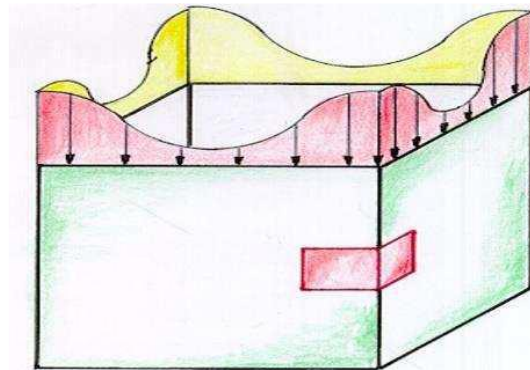


Figure 2 Load Distribution along the Perimeter of Corrugated Box (source: chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.ijltemas.in/DigitalLibrary/Vol.6Issue7/26-28.pdf)

Practice such as overhang should be avoided as it has been found that the deficit in BCS of packed boxes as an effect of overhang can range between 23–49% but vary on the extent and direction (length, width, or adjacent panel) of the overhang [10]. Another practice that should be avoided is misalignment of the boxes when packed on each other on a pallet as it plays a significant role in

decreasing the strength and the lifetime of the box, where the percentage decrease in BCS (lateral) value can be as much as 11% and 31% for 90% contact area and 80% contact area, respectively [11]. To ease the consequences of environmental factors, the end user must minimize the habit of particular methods that negatively affect the strength characteristics of the corners, including overhang on the pallet, packing on pallets with only a few slats, excessive shrink warp tension, and using “interlocking” stacking patterns [11].

Corrugated cardboard is a highly deformable material, the limit of its use may be delimited by the deformation of the box [12]. Corrugated boxes are made from a specialized material known as corrugated paperboard. It is necessary to accurately estimate the strength of corrugated boxes before applying them in real-world scenarios. This is due to their unique material composition, which allows their structure to be easily customized and strengthened to achieve high packaging performance but can also degrade over time due to prolonged use or environmental factors such as humidity. Paper material is orthotropic, exhibiting non-linear mechanical properties, which means that it possesses varying strength in different directions. For instance, the tensile strength of paper fibers in the machine direction can double compared to that in the cross-machine direction as strain increases [13]. Consequently, the orientation of corrugated board utilization becomes critical. However, even when the corrugated board is used in the correct direction, the weaker direction of this anisotropic material can lead to the failure of the corrugated box under specific conditions. For example, unexpected damage to the box's side panels can be induced by shock or piercing, significantly reducing the strength of the corrugated box.

1.1.3 Influencing factors of BCS

BCS is influenced by various factors, such as material properties, flute types, dimensions, and more. Each factor, or BCS feature, affects the BCS differently. The BCS features for corrugated packaging

can be systemized into three groups, the mechanical strengths of the raw paper material, the corrugated board, and the corrugated box itself. In the level of raw paper, the key factors involve liner type, weights of liner and a constant related to the fluting. The critical influencing factors for BCS involve the ring crush test (RCT), Concora liner test (CLT), Takeup factor, thickness, flexural stiffnesses in Machine Direction and Cross Machine Direction (MD and CD), Edge Crush Strength (ECT) and moisture content, etc, as shown in Figure 3. When it comes to the level of corrugated box, dimensions and perimeter of the box, design structure, applied load ratio, stacking time, and buckling ratio all have significant impact on the BCS value of a corrugated package. What's more, some other factors also make a difference in BCS, such as the presence of openings, ventilation holes and perforations, moisture content of the box, storage time, stacking conditions, etc [14].

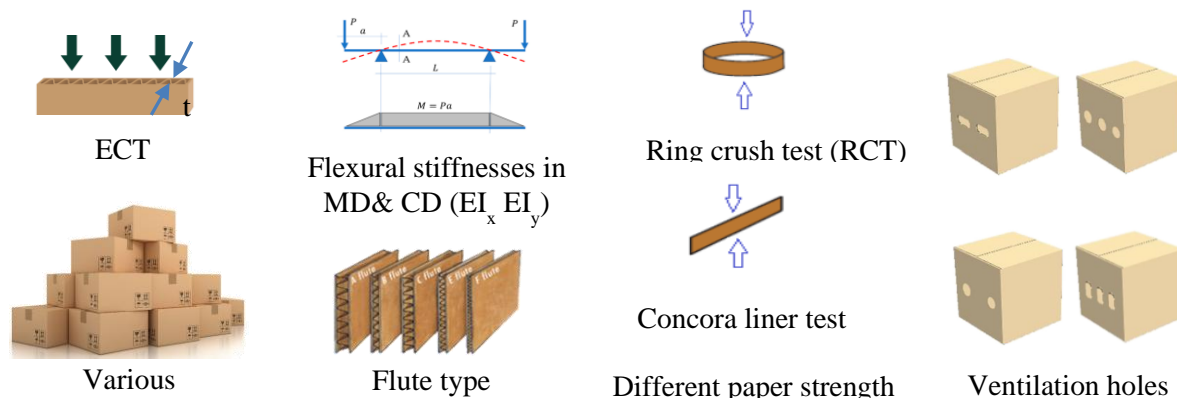


Figure 3 BCS Influencing Factors

ECT (Edge Crush Test) can be a vital indicator for the BCS of the corrugated packages. The BCS of a packaging container of the regular slotted container (RSC) design has been anticipated from the ECT value of the board [15]. The ECT & BCS, Stiffness & BCS, and Thickness & BCS links were proven to be solid and positive correlations.

The effect of box depth (which is not included in the McKee formula) is that the box becomes weaker as the height increases due to the wall buckling, where the compression strength dropped by as much as 62% from the 127mm to 1219mm box heights, which points out the weakness of

using McKee formula [9].

The stacking pattern has a significant impact on the BCS. Existing research has revealed that column stacking practice results in higher strength as compared to interlocking stacking patterns, and while stacking boxes, one needs to ensure that the four carton corners are placed in alignment [9], see Figure 4 below.

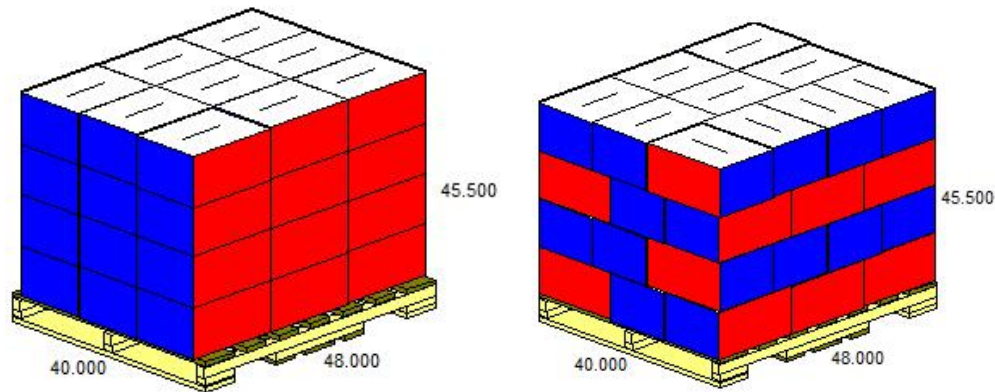


Figure 4 Column and interlocking stacking pattern of corrugated boxes

The mechanical properties are important design features because the function and performance of a product depend on its capacity to resist deformation under the stresses encountered in use, hence in design, the usual objective is for the product and its components to withstand these stresses without significant change in geometry [16]. Edge Crush Test (ECT) and Flat Crush Test (FCT) are the main two tests that determine if the mechanical properties of the corrugated board will meet the set or targeted performance of the box in the market [17].

1.2 APPROACHES FOR BCS ESTIMATION

The evaluation methods of BCS primarily include three approaches: compression tests (the most traditional method), mathematical models, and Finite Element Analysis (FEM). Each method has its drawbacks in BCS evaluation, and while researchers have made efforts to address these challenges, improving evaluation efficiency and accuracy still presents difficulties.

1.2.1 Compression Test

Compression test is one of the commonly used methods to test the corrugated box compression strength or stack load, to make sure the boxes do not fail when stacked over each other during the storage and distribution process. The Box Compression Test (BCT) is a standardized procedure designed to measure the maximum pressure or force that a material can withstand before rupturing. It is particularly relevant for assessing the strength of corrugated and paperboard materials commonly used in packaging applications. Different test standards are applied based on the requirements of corrugated packaging of different utilizations. The test standards involve ISO 12048 Packaging - Complete, Filled Transport Packages - Compression and Stacking Tests Using a Compression Tester, TAPPI T 804 Compression test of fiberboard shipping containers, ASTM D642 Standard Test Method for Determining Compressive Resistance of Shipping Containers, Components, and Unit Loads, JIS Z0212 Japanese Industrial Standard Method of Compression Test for Packaged Freights and Containers, ASTM D4169 Standard Practice for Performance Testing of Shipping Containers and Systems. The ASTM D642 is developed by the American Society for Testing and Materials (ASTM) to determine the compressive resistance of shipping containers, components, and unit loads. The key points include the test method which is to apply the compressive force to the package until failure and the result interpretation about how to determine the maximum compressive load. TAPPI T804 is provided by the Technical Association of the Pulp and Paper Industry (TAPPI) focusing on the determination of the compressive strength of fiberboard shipping containers by applying the compressive load until the failure of the packages. ISO 12048 is an international standard that defines the compressive and stacking tests for transport packages by applying compressive force to a package until failure or set load to simulate the stacking conditions in warehouses. JIS Z0212 focuses on compression test methods for corrugated

fiberboard boxes by applying compressive load until collapse and recording the force [18]. ASTM D4169 is a (Food and Drug Administration (FDA) - recognized consensus standard for conducting a transit simulation study for sterile barrier medical device packaging systems, it is the most common choice in the medical packaging industry [19]. Overall, the box compression Test is a fundamental assessment that evaluates the strength and resilience of packaging materials, and the compression test gives insight into the optimization of packaging design and material selection.

However, there are some limitations of the compression test. The package sample is limited by the laboratory facilities. The test conduction is limited by the laboratory setting and environment. Besides, a lot of other factors can reduce the accuracy of the compression test, including systematic errors, instrumental errors, environmental errors, procedure errors, and human errors [20]. Some instruments have limitations, which can cause consistent deviation from the real value. Laboratory temperature and humidity can change because of the unexpected failure of electronic devices. Human errors can cause measurement deviation, which cannot be eliminated in laboratory testing. In addition, the compression test process is very time-consuming and costly, starting with preconditioning of at least 24 hours in the required temperature and humidity environment, then setting up the sample, mounting it onto the testing apparatus, and ensuring that it is evenly aligned and free from any wrinkles or folds that could affect the test results, then applying the pressure steadily using the testing machine with a certain moving speed [21]. A testing machine for the compression test is shown in Figure 5. To minimize errors, multiple samples need to be tested. When it comes to various box structures, box dimensions, box materials, etc, the workload of physical testing increases dramatically. The current compression test can only test boxes one by one, which is very low efficiency. On top of repeated testing of one single sample, samples with different material properties, and different batches of product source can further increase the error

of compression test. Thus, these are the drawbacks of compression testing.

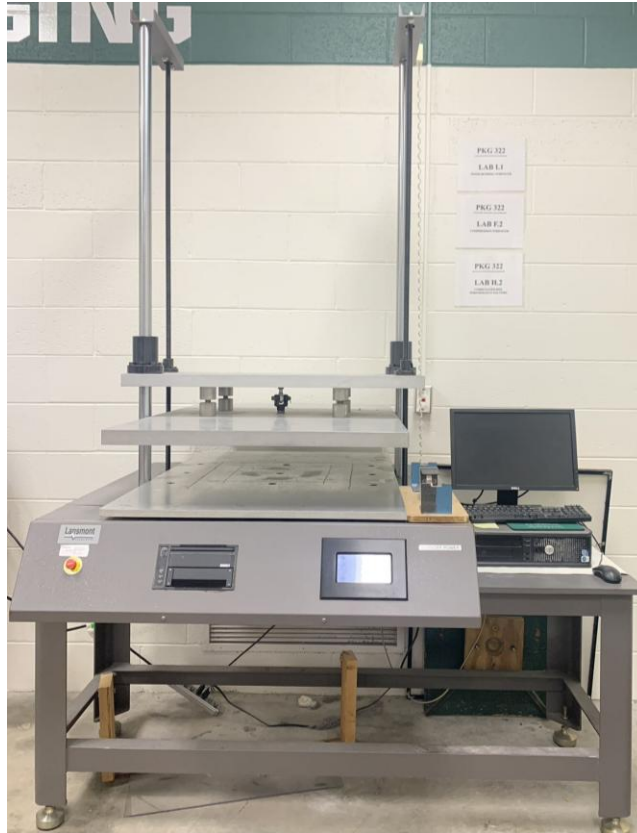


Figure 5 Testing machine for Box Compression Test

1.2.2 Mathematical models

Many mathematical models have been developed for BCS evaluation. The McKee formula is one of the most commonly used mathematical models used in industry for BCS estimation for corrugated packages, as shown in equation (1) [9]. The McKee formula was developed by McKee et al. [22] in the 1960s. McKee's formula estimates the BCS of corrugated boxes by employing three basic physical parameters of a box, including ECT, thickness, and box perimeter. However, the McKee formula is limited in its predictive accuracy by the uncertainty in the measurements of package properties [23]. The McKee formula is simple and accurate to predict the regular slotted container (RSC). The research by McKee et al. [22]. presented certain limitations due to the simplification of more general physical relationships. Fundamentally, these were linear regression

analyses based on specific data sets, typically limited by processing constraints.

$$BCS = 5.87 \times ECT \times \sqrt{Thickness \times P}$$

Where:

ECT – Edge Crush Strength (lb/in).

Thickness – Thickness of the corrugated board (in).

P – Perimeter of the box (in).

(1)

The McKee formula has limitations in that it must only be utilized when the length-to-width ratio or the height-to-length ratio of the box is not too large. Specifically, this assumes that the length is less than three times the width, and the perimeter is less than seven times the depth [14]. However, corrugated packaging patterns have become more and more diverse as the advancing of e-commerce, corrugated packages with various dimensions become more and more common. Plus, the McKee formula is not able to estimate the BCS of packages with various patterns available in the market, as it does not account for variations in material properties and box structures.

The limitation of the McKee formula is the limited consideration of only three physical parameters of a box. There are a lot of other physical parameters that influence the BCS, such as structural mechanics factors (flexural stiffness, torsional stiffness, diagonal stiffness), production factors (crush, scoring, slotting quality), use factors (the squareness of the box when erected, how the box is sealed). Many of these are difficult or impossible to capture in a closed-form mechanistic model of BCS. For example, torsional stiffness, also called shear stiffness, measures the torsional resistance of a corrugated board in the machine direction (MD). When a corrugated box undergoes compressive loading in the MD, the side walls tend to deform outwardly in a buckling response to the compression. This deformation is affected by the longitudinal shear stiffness. The shear stiffness can directly influence how well the box can protect its product [24]. However, MD torsional stiffness is a more sensitive predictor of corrugated board performance and there is no test standard for this parameter [25].

As mentioned above, buckling is another critical factor that influences the compression strength of corrugated boxes. Urbanik, T. J., & Frank, B. (2006) studied the impact of buckling on the box compression strength and formulated a mathematical equation to demonstrate the relationship between buckling and BCS [26], as shown in equation (2). However, this equation involves some parameters (such as the flexural stiffness in transverse, axial, and twisting directions) that are difficult to obtain after the corrugated board production process, causing the utilization restriction of this equation to estimate the BCS. Although researchers have attempted to develop other equations to estimate BCS more accurately, these equations still contain parameters that are difficult to access. For example, an improved version of the McKee formula after considering buckling is shown in equation (3), the flexural stiffnesses in the transverse, and axial directions are included in this equation, and they are usually not measured after the corrugated board production process.

For inelastic buckling $P_1 = P_f l = \alpha P_m l$

For elastic buckling

$$P_1 = P_f l = \alpha (4\pi^2)^\eta P_m^{(1-\eta)} (EI_x EI_y)^{-\eta} l^{(1-2\eta)} \left(\left(\frac{d}{l} \left(\frac{EI_x}{EI_y} \right)^{1/4} \right)^\tau \frac{2\hat{c}+M}{4(1-\nu^2)} \right)^\eta$$

Where:

P – compression.

P_m – ECT.

l – panel length.

d – depth.

EI_x, EI_y, EI_{xy} – Flexural stiffness per unit width in transverse, axial, and twisting directions.

ν – Geometric mean Poisson's ratio.

\hat{c} – Normalized in-plane shear modulus of elasticity derived in Urbanik (1992)

$$\hat{c} = \nu + 2(1 - \nu^2)(EI_{xy}/EI_x)\sqrt{EI_y/EI_x}.$$

τ – Empirical improvement.

$\eta = 1 - b$ (b is McKee formula constants).

(2)

$$BCS = 2.028 ECT^{0.746} \sqrt{EI_x \times EI_y}^{0.254} P^{0.492}$$

(3)

Where:

ECT - Edge Crush Strength (lb/in).

EI_x , EI_y - Flexural stiffness in the machine direction & cross-machine direction of the corrugated board (lb*in).

P - Perimeter of the box (in).

1.2.3 Finite Element Analysis (FEA)

Finite element analysis (FEA), a powerful technique often used for the simulation of engineering processes, is finding a home in the corrugated industry, and has been applied to evaluate the BCS of corrugated packages. FEA models generate predictions by leveraging fundamental physical mechanics across different length scales, stitching together functional relationships to estimate the effect of changes in very basic material properties (e.g., paper elasticity) on the larger final system (e.g., box strength). When the functional form is known, the propagation of parameters and their impacts produce a prediction of the result. Various studies have explored using an FEA approach to predict ECT 26-29. or BCS 30-39., allowing for a detailed examination of the impact of moisture, perforations, holes, openings, crushing, and more complex structures. Literature has grown so extensively that even review articles addressing the usefulness of FEA on broader topics have sections discussing corrugated paperboard packaging [27]. Each of these studies requires detailed information on the material parameters to input into the models, typically producing a reasonable agreement between the model and the limited number of physical samples evaluated. As such, they potentially contribute to our understanding of the impact of specific changes examined (i.e., hole size and placement.) [28]. However, very few of these studies address or investigate how well their models work with boxes made of different, varied, or unknown materials. They also don't often discuss how the varying physical and mechanical properties of paper or combined board can affect the accuracy of their predictions. Typically, the input parameters required for an FEA are not properties regularly measured in the papermaking or box-making process. Thus, existing (published) analyses cannot reasonably be used for a generalized assessment of a random box in the same way that we can use the McKee equation.

1.3 ARTIFICIAL NEURAL NETWORK (ANN)

ANN is a subset of AI, which serves as an intelligent tool with great advantages in data processing and estimation. Artificial neural network was introduced in 1956 [29]. Artificial neural networks are inspired by the human biological neural network. An ANN is an algorithm that can recognize the relationships of a set of data and use the computer to make decisions or predictions. The Artificial neural network (ANN) model involves computations and mathematics, which simulate the human–brain processes. The Artificial Neural Networks (ANNs) are a very different computing approach that can also be used to explore the underlying relationships in a set of data and generate predictive models.

ANNs have many advantages because they can strive to take whatever information we happen to know in terms of materials inputs and gather relationships to the outputs of interest. This inference process can take in a broader range of inputs, teasing out their connections (implicit or explicit) to “understand” their relationship to a given output. The goal of ANNs is to minimize the error of the predicted property. By mapping features in data, ANNs can substantially add to the power of exploratory data analysis [30]. Using ANNs can bring many benefits to scientific research [31], making more consistent decisions and shortening the decision-making process [32]. Given the fundamentally non-linear relationship between fiber characteristics and the mechanical properties of paper, combined board, and boxes, this alternate approach is beginning to garner interest among researchers [33, 34]. The prediction capability of ANN potentially allows us to incorporate a large number of input parameters into a single prediction model, limited only by the size of our data set. ANN research to date has focused on specific areas or factors influencing box strength [35].

1.3.1 Components and architectures of ANNs

An ANN consists of three types of layers, including the input layer, hidden layers, and output

layers. Each layer contains several neurons which are real numbers, and these neurons are connected by weights which are the influence strength between the two neurons connected. If the neuron in the last layer has a strong influence on the neuron in the next layer, the weight is a large number. If the influence of the neuron in the last layer is weak on the neuron in the next layer, the weight is a small number, a typical ANN schematic is shown in Figure 6. An activation function is involved to realize nonlinear patterns between the input and output. An ANN can have two or more hidden layers, and each layer can have several neurons. Therefore, all the connections or weights between neurons allow an ANN to have a high number of degrees of freedom. Sometimes, a bias is also added to the summation (or the weighted sum of all neurons) to allow an ANN to get activated above a certain value. The biases also increase the number of degrees of freedom of an ANN. As a result, an ANN can have a high flexibility and high capability to recognize the nonlinear pattern for a set of data and provide the best possible prediction through several iteration weights updating.

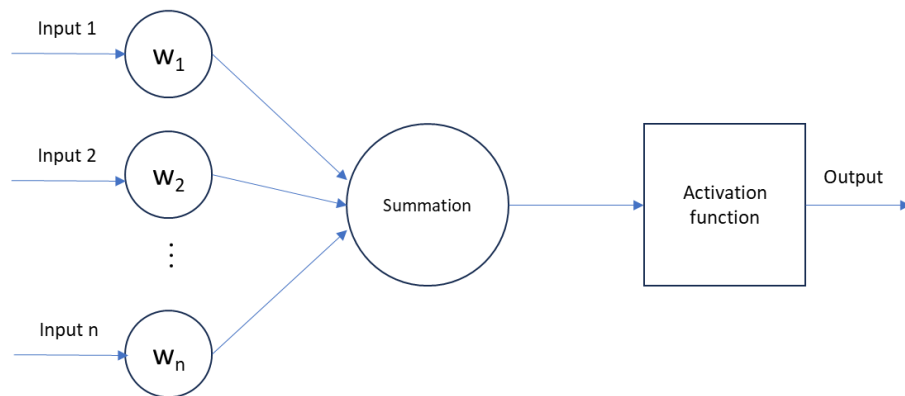


Figure 6 Schematic of a typical ANN

Input: Input data is usually labeled which is used for ANNs to learn and recognize the underlying patterns (or relationships) in the input data. Input data can be collected through physical tests, mathematical fabrication, and some other approaches. The values of input data will be the neurons'

values in the input layer.

Weights and biases: Weights and biases are the parameters through a whole neuronal network. The weights of the connections between neurons are the adjustable model parameters that govern how the model calculates the output from the given inputs. To some degree, weights can be also regarded as the coefficients of input data. By adjusting the weights, ANNs can reduce the influence of those not important inputs and increase the influence of those critical inputs. In this way, an ANN can nudge its output as close as possible to the real values. The adjusting of weights and biases is the key part of ANNs' learning.

Epoch: An epoch signifies the process of feeding a dataset into the model and the model's weights are adjusted to reduce the overall error. This iterative repetition of the process, known as multiple epochs, continues until the error reduction rate falls below the given criteria.

Activation function: Activation function is a series of functions that allows an ANN to realize non-linear operations. Considering the complex problems in real situations in our lives, most of those problems are complicated non-linear problems, such as the changeable temperature around a year or several years. To fit the non-linear problems using ANNs, activation functions are necessary. Without applying a non-linear activation function, an ANN will be just a linear combination of input values. There are different types of activation functions, both linear functions and non-linear functions. Although the linear function has very limited application, it can still be counted as a type of activation function. The non-linear activation functions are more commonly used. Figure 7 presents graphical representations of several commonly used activation functions, which play a crucial role in ANNs. These activation functions introduce non-linearity into the network, allowing it to model complex relationships within data that would otherwise be impossible for a purely linear system to capture. It is this non-linear behavior that enables ANNs to process intricate

patterns, make meaningful decisions, and effectively solve complicated real-world problems across various domains, such as image recognition, natural language processing, and predictive analytics.



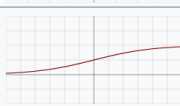
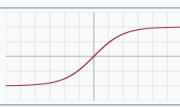
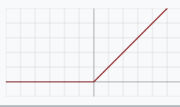
Name	Plot	Function, $f(x)$	Derivative of f , $f'(x)$	Range
Identity		x	1	$(-\infty, \infty)$
Binary step		$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$	$\begin{cases} 0 & \text{if } x \neq 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$	$\{0, 1\}$
Logistic, sigmoid, or soft step		$\sigma(x) = \frac{1}{1 + e^{-x}}$ ^[1]	$f(x)(1 - f(x))$	$(0, 1)$
tanh		$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	$1 - f(x)^2$	$(-1, 1)$
Rectified linear unit (ReLU) ^[11]		$\begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$ $= \max\{0, x\} = x \mathbf{1}_{x>0}$	$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x > 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$	$[0, \infty)$

Figure 7 Common activation functions (Source: https://en.wikipedia.org/wiki/Activation_function)

The working principle of ANNs is related to the weight updates during the training process. Weights are assigned randomly at the beginning of the ANN learning process. An ANN calculates the outputs by receiving the input neurons' values, computing the weighted sum of all neurons in the last layer, adding biases, and passing the weighted sum to an activation function. This process is called propagation. Since the weights are randomly assigned initially, there is usually a difference between the outputs and the real values. An ANN will minimize the difference by updating the weights and biases, which is called the backpropagation process. An ANN usually updates the weights and biases several times to give a best-predicted result. This is the working principle of ANNs or how ANNs learn from the data and make predictions [36, 37].

1.3.2 Cost Function

The cost function is the criteria of an ANN to adjust its weights. Generally, the cost function calculates the error between the output and actual values and chooses the output with minimum error. There are different types of cost functions used in an ANN depending on what problem an ANN is solving 50-52.. Generally, there are two common types of problems, regression problems and classification problems. Based on the problems that need to be solved, the commonly used cost function includes three types, which are Regression cost Functions, Binary Classification cost Functions, and Multi-class Classification cost Functions. If the problem that needs to be solved is a regression problem, then a regression cost function should be used. If the problem needs to be solved is a classification problem, then a binary classification problem or multiple-class classification cost functions should be chosen.

Regression cost function: The regression cost function deals with predicting a continuous value, for example, the weather during a day, or the mileage that a person drives. The regression cost function measures the average error or the average difference between the output and real value of data training over the entire data set. There are three different errors to calculate by using the regression cost function, including Mean Error (ME), Mean Squared Error (MSE), and Mean Absolute Error (MAE).

Mean Error (ME) is equal to the sum of the error between the output and the real value, as shown in equation (4). For each training data point, the error between the output and the real value can be either positive or negative, and they can cancel out each other when they are added up then giving zero error to the regression models. Due to this cancel-out problem, Mean Error (ME) is usually not used frequently.

$$ME = \frac{1}{n} \sum_{i=1}^n (Output - real\ value) \quad (4)$$

Where n is the number of samples in a trained dataset.

Mean Squared Error (MSE) is the average square difference between the output and real value, which is shown in equation (5). MSE doesn't have the drawback of cancelling out problems in Mean Error (ME), which is more commonly used for regression models. However, the disadvantage of MSE is that it is not very robust for the outliers in a dataset because the square can enlarge the error from the outlier data points.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Output - real\ value)^2 \quad (5)$$

Where n is the number of samples in a trained dataset.

Mean Absolute Error (MAE) is the average absolute difference between the output and real value, as shown in equation (6). MAE improves the shortcoming of Mean Error (ME) by using the absolute value of the error for each data point. MAE is very robust to the outliers in a dataset. So, if a dataset that an ANN needs to train has many noises or outliers, calculating MAE for the regression models is better.

$$MAE = \frac{1}{n} \sum_{i=1}^n |Output - real\ value| \quad (6)$$

Where n is the number of samples in a trained dataset.

1.3.3 Classification of ANNs

ANNs are classified into two types, feed-forward neural networks, feed-back neural networks, or recurrent neural networks [38].

The first type of ANN is called feed-forward neural networks [39]. In feed-forward neural networks, the connections between nodes don't form a cycle, which means the signals move only in one direction, from input to output. Figure 8 shows the schematic of a typical feed-forward propagation. The feed-forward ANN calculation process cycle includes a forward-step

computation of fitting input data into an ANN and a backward-step computation of calculating errors and updating the weights in the model. A single iteration of this computational process is termed an epoch within an ANN.

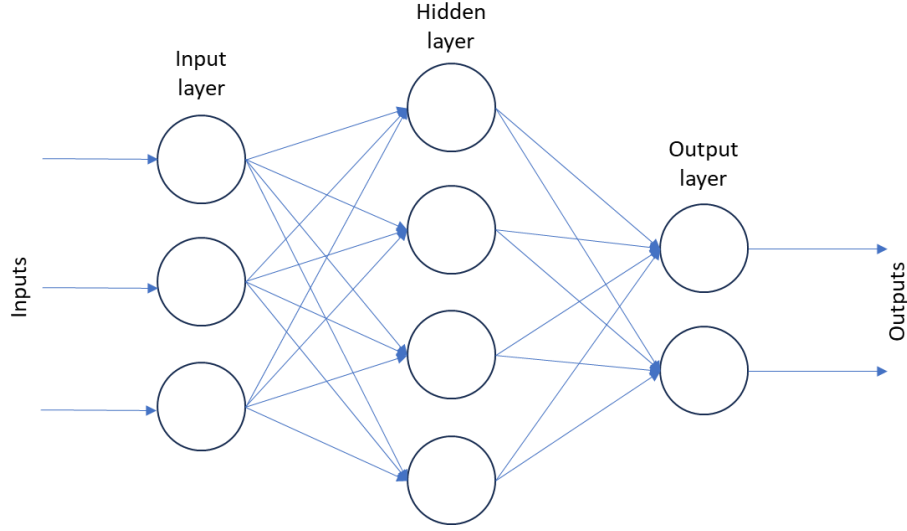


Figure 8 Schematic of a typical feed forward propagation

Training process of feed-forward ANN: While training a model using data, a feed-forward ANN approach segments a given set of known data into two uneven groups, training data and testing data. The former is used to build and refine the model, and the latter is used to evaluate the model accuracy.

Generally, to assess an ANN, 67% of data of a data set is split into training data, and the remaining 33% is split into testing data. Each node in the hidden layers could be defined based on a weighted sum of the parameters in the prior layer, as shown in equation (7).

$$h_j^i = f \sum_k (w_k^j \times x_k)$$

Where

h_j^i is the value of j^{th} neuron in i^{th} hidden layer. $j = 1, 2, 3, \dots, n_1$ when $i = 1$; $j = 1, 2, 3, \dots, n_2$ when $i = 2$.

x_k is the value of k^{th} neuron in previous layer, $k = 1, 2, 3, \dots, n$.

w_k^j is the weight from the k^{th} neuron in previous layer to j^{th} neuron.

(7)

f is the activation function.

One of the most popular feed-forward neural networks is the convolutional neural network (CNN). CNN is especially good for image recognition and classification because CNN can identify the patterns in an image. CNN can be used to recognize the content or numbers in an image. Figure 9 shows an example of CNN recognizing a handwritten digital number '2'.

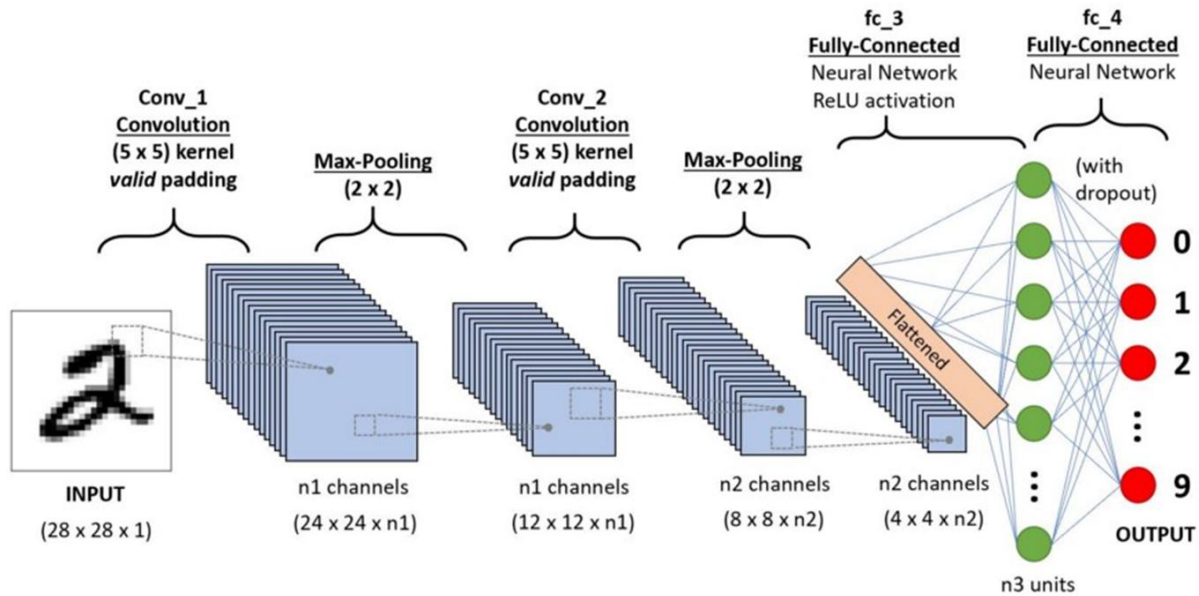


Figure 9 An Example CNN architecture for a handwritten digit recognition task

The second type of ANN is called feed-back neural networks also can be called recurrent neural networks (RNN) [40, 41]. Feedback neural networks allow signals to move in both directions, say input to output or output to input, which can form a loop traveling for signals. Feed-back neural networks are dynamic networks that keep changing until they reach an equilibrium point [42].

1.3.4 Learning strategies of ANNs

ANNs use different learning strategies, including supervised learning, unsupervised learning, and reinforced learning.

In supervised learning, ANNs learn the underlying relationships between input data and output data. ANNs recognize the governing function involving all input data and the output. It is like a fitting process that fits a function between the input data and output data. In supervised learning, ANNs need labeled input data and output data, which can teach the computer to learn the patterns between the input data and output data. Supervised learning ANNs are mainly used for classification and regression problems.

For unsupervised learning, ANNs don't need a labeled input data set to guide the computer to learn the underlying patterns between input and output data. Instead, unsupervised learning ANNs classify a set of elements according to some similar patterns between data and data. Unsupervised learning ANNs are mainly used for clustering and anomaly detection problems.

Reinforced learning neural networks are different from supervised learning neural networks [43]. Reinforced learning doesn't need labeled input data and output data. Figure 10 is the typical framing of a reinforced learning scenario.

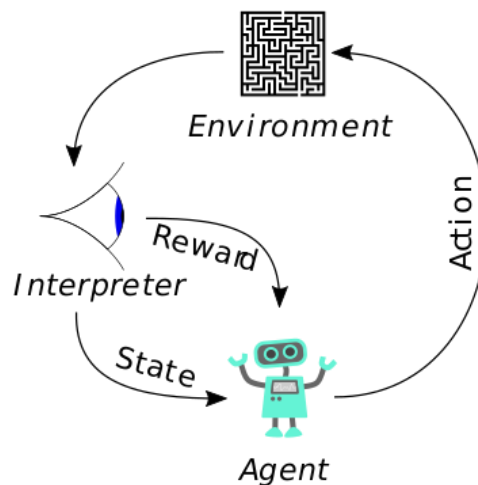


Figure 10 The typical framing of a reinforced learning scenario

An intelligent agent takes actions in an environment. The environment interprets the agent's action result as a reward and a representation of the state and gives feedback to the agent so that the agent

can adjust its action to maximize the accumulative reward. The environment of reinforced learning typically adopts the Markov decision process (MDP) [44], a mathematical framework that is good for modeling decision-making when the outcomes are partly random and partly under the control of a decision-maker. Many reinforced learning neural networks use dynamic programming techniques. Reinforced learning can be used for environmental learning.

1.4 APPLICATIONS OF ANNS

The ANN approach has been utilized in many different applications and various fields over the past few decades [45]. In recent years ANNs have drawn attention in the areas of Facial Recognition, Image Analysis, and Natural Language Processing (NLP). In the field of packaging, ANNs have also been utilized to solve certain problems, such as transport packaging cushioning property evaluation, polymer product characteristics prediction, and municipal solid waste (MSW) management. However, the application of ANNs in packaging strength estimation is very limited.

1.4.1 Applications of ANNs in Facial Recognition, Image Analysis, and NLP

ANNs have been applied in facial recognition area. An optimized ANN system using a harmony search algorithm was developed to improve the accuracy of face recognition, which can give a lower Mean Squared Error than the other hybrid ANN system called hybrid particle swarm optimization ANN [46]. Application of ANNs in predicting turbulent stock markets was discussed and studied [47]. A hybrid ANN model based on a genetic algorithm and simulated annealing was developed to predict the stock market with improved accuracy and a new set of input variables for ANN models was proposed [48]. ANNs with different algorithms (including Levenberg-Marquardt, Scaled Conjugate Gradient, and Bayesian Regularization) were studied to predict the Indian stock market, getting an accuracy of 99.9% using tick data [49].

In the field of image classification and regression, Deep Learning (DL), a subset of ANN, has also

been applied to characterize the symmetries of simulated measurements of samples. In ref. 216, Ziletti et al. (2018) obtained a large database of perfect crystal structures, introduced defects into the perfect lattices, and simulated diffraction patterns for each structure [50]. DL models were trained to identify the space group of each diffraction pattern. The model achieved high classification performance, even on crystals with significant numbers of defects, surpassing the performance of conventional algorithms for detecting symmetries from diffraction patterns. DL has also been applied to classify symmetries in simulated STM measurements of 2D material systems by Choudhary et al. (2021) [51].

In Natural language processing, one of the major uses of NLP methods is to extract datasets from the text in published studies. Cooper et al. (2019) demonstrated a “design-to-device approach” for designing dye-sensitized solar cells that are co-sensitized with two dyes [52]. Natural language processing can also directly make material predictions without intermediary models. shitoyan et al. (2019) reported that word embeddings (i.e., numerical vectors representing distinct words) trained on materials science literature could directly predict materials applications through a simple dot product between the trained embedding for a composition word (such as PbTe) and an application word (such as thermos electrics) [53].

1.4.2 Applications of ANNs in Packaging

ANNs have been applied in packaging since the 1990s, involving different fields in packaging. According to the reports recently published, ANNs’ applications have been explored in the various parts of packaging, from transport packaging, and cushioning packaging, to packaging design and manufacturing systems, PE product characteristics prediction, and municipal solid waste (MSW) management for classifying different packaging materials.

Applications of ANN in transport packaging: Bahrami et al. (1995) developed an intelligent

packaging system to retrieve a design from a standard set of chair designs that can satisfy the required needs using ANNs [54]. Siripong Malasri (2015) applied an artificial neural network in transportation packaging to measure the temperature of a wooden software pallet stringer under different temperatures at the time of the drop test, by building several temperature profiles from data collection with different starting temperatures. This application solved the problem of thermocouple cords interfering with the free-fall drop of a pallet sample [55].

Applications of ANN in cushioning packaging: Yanchun Liang & Xiaowei Yang et al. (1996) developed neural networks to identify the nonlinear characteristics in cushioning packaging to help reduce the shock and vibration during the transportation process [56].

Applications of ANN in design and manufacturing: Siripong Malasri et al. (2016) developed a neural network to estimate the temperature profile in a wooden Softwood Pallet Stringer during the time of drop test [34].

Applications of ANN in material product characteristics prediction: Polyethylene (PE) is one of the most widely used polymers in packaging materials. The ethylene index (EIX) is an important variable for PE product characteristics. However, EIX is hard to measure because it is affected by various factors, such as pressure, ethylene flow, hydrogen flow, catalyst flow, etc. To estimate EIX, different neural network models were developed by Akbar Maleki & Mostafa Safdari Shadloo et al (2020). Their result showed that the Multi-Layer Perceptron model could predict the production level of HDPE with a high Regression coefficient [57].

Applications of ANN in municipal solid waste (MSW) management: Municipal solid waste (MSW) is waste from rejected packages with different packaging materials. The sustainable management of MSW is a challenging task when it comes to packaging sustainability, because MSW involves all kinds of packaging materials, including plastics, paper, metal, glass, and wood, and the

characterization of different packaging materials is very expensive. That's how the modeling approaches comes. The classical models are less effective, and artificial intelligence models have drawn the attention of researchers. Adeleke & Akinlabi et al. (2021) explored the application of neural networks in predicting the physical composition of MSW. They optimized the network architecture, training algorithms and activation function of a neural network to predict the fraction of MSW streams from meteorological parameters with high accuracy. Multiple training algorithms, and activation functions are combined and compared to optimize the neural network to predict the percentage composition of four different maximum packaging materials streams based on the data of minimum temperatures, wind speed, and humidity in their case study. Their study result concluded that the complex physical composition of MSW can be predicted with a single hidden layer neural network, which provided the theoretical support for handling MSW and contributed to the academic community related to packaging sustainability [58]. Oliveira & Sousa et al. (2019) also studied a feedforward neural network to identify the varies (from the level of education of the population, the size and level of urbanization of the municipality to factors intrinsic to the waste collection service) influencing the amount of separately collected packaging waste. With a dataset of 42 municipalities in Portugal, their study result showed that the high-performance neural network gave a 34% higher coefficient of determination (R-value) than the traditional regression models [59].

Although the ANN approach has been explored in various aspects of the packaging field, there are limited studies evaluating the BCS of corrugated packages at an industry-applicable level using ANN models. The primary challenge lies in collecting a sufficiently large data set that encompasses the majority of BCS values used in the industry. Existing studies often rely on small datasets that do not adequately represent the broad range of commonly used box dimensions or BCS values,

making it difficult to build a generalized ANN model suitable for industry applications. This study aims to bridge this gap by developing a generalized ANN model for BCS evaluation, using a dataset that encompasses the majority of commonly used box dimensions in the industry.

1.5 RESEARCH OVERVIEW

This paper evaluates the Box Compression Strength (BCS) of corrugated boxes using an Artificial Neural Network (ANN). Chapter one describes the background of the research, including fundamental knowledge of BCS and ANN, as well as the motivation behind the study.

Chapter two details the training of the ANN model using available datasets for BCS evaluation, examining key modeling factors including the number of neurons in the hidden layers, epoch number, number of modeling cycles, and number of data points. By applying datasets from both the literature and synthetic data created using the McKee formula, the optimal values for these factors and the minimum data population needed were identified. This chapter provides insights into the ANN's performance in evaluating BCS values and demonstrates the feasibility of using ANN to estimate BCS.

Chapter three investigates the relative importance of packaging design features using the ANN approach. Using BCS as a representative packaging property, four different ANN algorithms (Connection Weights method, Gradient-based method, Permutation method, and SHAP values) were employed to determine the relative importance of different BCS features (Edge Crush Strength (ECT), box dimensions, thickness, and flexural stiffness). A synthetic dataset generated using the McKee formula was used to compute the theoretical relative importance of these BCS features. The ANN predicted BCS feature importance ranking aligns with the theoretical relative importance of studied BCS features. A real dataset from the industry was also used to estimate the relative importance of five BCS features. The ANN predicted feature importance ranking was also

consistent with theoretical relative importance calculated from the McKee formula. In addition, the ANN predicted BCS feature importance from the real data set proves the importance of depth was not zero, which aligns with the buckling theory and reveals the inaccuracy of the McKee formula. The result indicates the feasibility of applying the ANN approach to evaluate the relative importance of packaging design features, allowing designers to minimize the design effort by prioritizing changing the more impactful packaging design features. These findings guide for material and cost savings in packaging design.

Chapter four covers the development of an ANN using a real dataset that includes box dimensions representative of the majority of BCS values at an industry application level. An extracted dataset from the real data that covers the majority of BCS values used in the industry was applied to train a generalized ANN model. The values of key ANN modeling factors were determined based on the study of previous datasets and five optimization methods for optimizing the hidden layers neuron configuration, including Information Criteria using the AIC method, Hebb's rule, Information Criteria using the BIC method, Optimal Brain Damage rule, and Bayesian method. The optimal hidden neuron configuration was identified by striking a balance between the model prediction error minimization and computational efficiency maximization. The final ANN model prediction error for the test data was calculated for BCS prediction. The error was 9.52%. A possible solution for improving the ANN prediction accuracy is given in the end. Chapter Five summarizes the work of this research and highlights the research directions for future studies. This study provides a methodological guide for future research exploring the applicability of ANN approaches to address problems and answer questions in the packaging industry.

Objectives of this research:

- Objective 1: Study how the ANN performs in evaluating BCS and determine the amount of

data required for reliable ANN predictions.

- Objective 2: Validate the ANN capability for evaluating the relative importance of packaging design features using BCS as a representative packaging design property.
- Objective 3: Develop a generalized ANN model applicable at an industrial level using a real-world data set.

CHAPTER 2: A COMPARATIVE ANALYSIS OF ARTIFICIAL NEURAL NETWORK (ANN) ARCHITECTURES FOR BOX COMPRESSION STRENGTH ESTIMATION

2.1 INTRODUCTION

In this chapter, we investigate the data requirements for an artificial neural network (ANN) to estimate compressive strength and evaluate the ANN's ability to address input variation limitations in the papermaking and box manufacturing process. Supervised learning methods are applied.

Given the limited existing research, it remains to be seen whether an ANN can estimate BCS any more accurately than our historical, closed-form approaches. A properly structured ANN might be able to identify additional parameters that contribute to BCS with similar level of impact as known existing factors (e.g., Edge crush test (ECT) value) and thus improve current models over the known levels of inherent variation in the input data. In order to leverage those opportunities, we need to clearly identify the size of the data set required. Compared with many ANN applications which automatically create the underlying data to build a model, collecting data point for BCS estimation model is comparatively expensive, necessitating a series of off-line tests. For ANN modeling of corrugated packaging, the challenges required to generate sufficient data sets may well be the limiting factor on the capability of the model.

This research aims to apply ANN to the box compression strength (BCS) of corrugated boxes. Various datasets of BCS have been collected and used to train the ANN model for BCS evaluation. The training process is complex and influenced by multiple factors, including internal factors related to ANN architecture and external factors pertaining to the applied datasets.

Internal factors of ANN involve the number of input neurons, hidden neurons, hidden layers, output neurons, and epochs. A new concept called modeling cycle was introduced to mitigate the noise in ANN predictions. This concept aims to obtain results that accurately reflect the average

error level of ANN predictions, thereby enhancing the reliability of the model's output. In this study, the BCS features serve as the input neurons, and BCS values are the output neurons. Thus, the number of input neurons corresponds to the number of BCS features used during ANN model training, and the number of output neurons is one. The ANN training process involves determining the optimal number of hidden neurons, hidden layers, epochs and modeling cycles.

External factors include the number of data points needed to achieve reliable training results for the ANN model. A dataset that is too small cannot provide reliable results, while an excessively large dataset can unnecessarily increase ANN training time. Therefore, it is crucial to determine the minimum amount of data needed to avoid resource wastage while ensuring robust model performance.

2.2 DATA SETS APPLIED

In this study, three datasets were trained to build an ANN model for BCS estimation. The three data sets include the McKee data set, an idealized data set and a data set with variation. The McKee data set is from literature presented by McKee in 1963 [22], specifically compiled for BCS estimation. It consists of 63 data points derived from box compression testing. The idealized data set is a synthetic data set based on McKee equation [22]. This data set was generated by including the box dimensions, ECT values, and thicknesses of 3,009 boxes commonly used in commerce and substituting them into the McKee equation. The data set with variation was created by introducing random errors to the parameters of the idealized data set's boxes. BCS values were then calculated using the McKee equation [22]. This process was carried out to achieve a variation of $\pm 5.4\%$ for BCS. It contains an equivalent number of data points as the idealized data set. Detailed descriptions of these three datasets are provided in the ANN training section, delineating the specifics for each dataset.

2.3 ANN KEY FACTORS INITIALIZATION

To begin, we apply the ANN approach to the existing data from McKee's 1963 research. Although the McKee data set proves too small for a robust ANN study, its well-established nature enables us to define our ANN methodology. Moreover, it illustrates the process to readers who are familiar with box compression modeling but less acquainted with ANNs. Next, we employ the ANN approach to analyze a significantly larger "synthetic" data set, constructed using idealized data derived from the McKee equation. This dataset allows us to evaluate the potential accuracy of an ANN model when applied to an established large data set and physical relationship. Furthermore, we introduce variation to the input data of the idealized data set, enabling us to assess how this variation propagates through the ANN. This investigation addresses the fundamental question of data set size and evaluates whether the current data collection approaches in the corrugated industry are sufficiently advanced to support the application of ANN in assessing box performance. The conclusion has been appropriately presented at the end, encapsulating the main findings and providing a conclusive summary.

A general ANN is structurally composed of three fundamental types of layers: the input layer, the hidden layer(s), and the output layer. The input layer receives raw data and passes it forward, while the hidden layer(s) performs complex computations by applying activation functions to weighted inputs. The output layer then generates the final prediction or classification result based on the processed information. Each layer consists of multiple neurons, which are interconnected with neurons from adjacent layers, forming a network of weighted connections that facilitate learning and pattern recognition. In this study, the ANN model specifically designed for evaluating BCS follows this structural framework and is visually represented in Figure 11, illustrating the organization and connectivity of the network's layers. By leveraging this multi-layered structure,

the ANN can effectively capture non-linear relationships in the data, improving the accuracy and reliability of BCS evaluations.

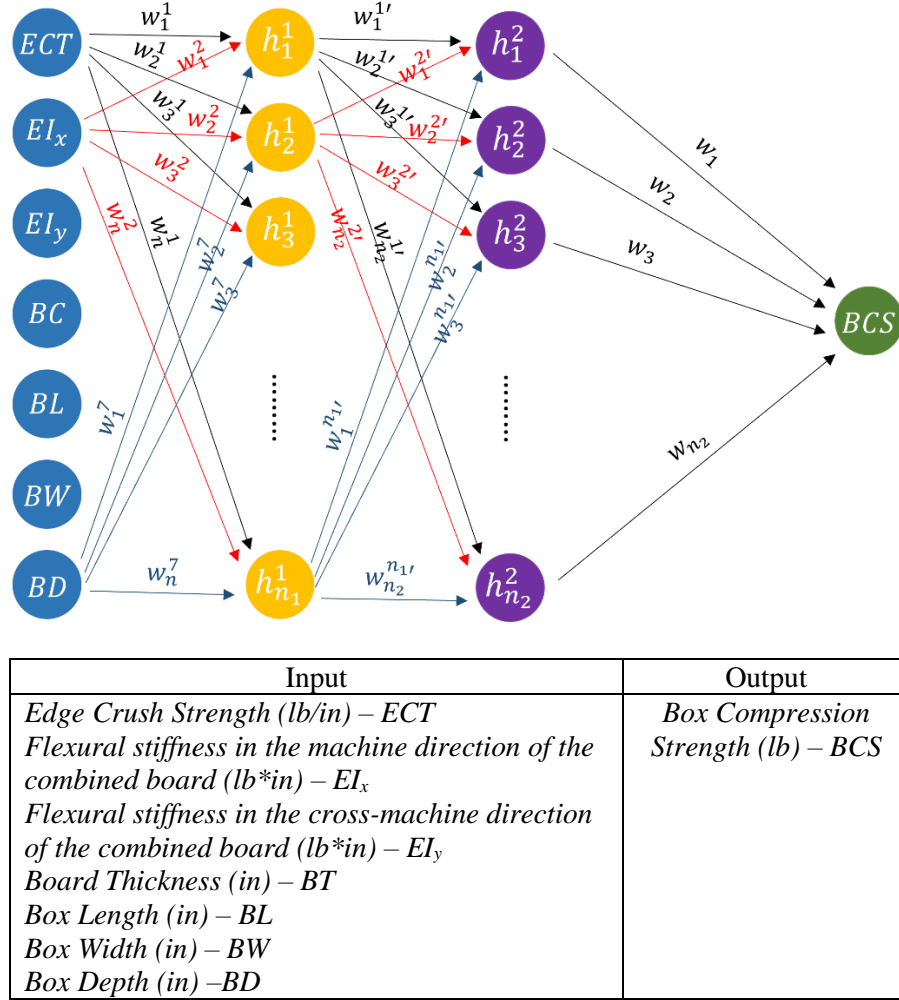


Figure 11 A model of an Artificial Neural Network (ANN) structure for predicting box compression strength (BCS) using inputs provided by the McKee data set

At the beginning of the ANN training process, all weights between nodes are randomly assigned. The squared difference between predicted BCS values from our training data and their actual BCS values is then calculated in equation (8), and the weights are adjusted via a backpropagation process.

$$MSE = \frac{1}{n} \sum_{i=1}^n (BCS_{predicted} - BCS_{actual})^2 \quad (8)$$

where n is the number of samples in a trained dataset, and MSE represents mean squared error

As mentioned above, an ANN approach segments a given set of known data into two uneven groups, training data and testing data. The former is used to build and refine the model and the latter is used to evaluate the model accuracy. To assess our ANN, 67% of each data set were split into training data and the remaining 33% were split into testing data. Each node in the hidden layers could be defined based on a weighted sum of the parameters in the prior layer, as shown in equation (9).

$$h_j^i = f \sum_k (w_k^j \times x_k)$$

h_j^i is the value of j^{th} neuron in i^{th} hidden layer. $j = 1, 2, 3, \dots, n_1$ when $i = 1$; $j = 1, 2, 3, \dots, n_2$ when $i = 2$. (9)

x_k is the value of k^{th} neuron in previous layer, $k = 1, 2, 3, \dots, n$.

w_k^j is the weight from the k^{th} neuron in previous layer to j^{th} neuron.

f is the activation function.

The choice of activation function is critical for ANN model prediction. To enhance efficiency, the Rectified Linear Unit (ReLU) is used as the activation function for hidden layers since it's the default activation function for hidden layers and perhaps the most common function used for hidden layers in Machine Learning studies [60]. Besides, since only a subset of neurons is activated at any given time, the ReLU activation function significantly mitigates the vanishing gradient problem, which often hampers deep neural network training by causing gradients to diminish as they propagate backward through layers. This property allows ReLU to enhance learning efficiency and contribute to faster convergence during model training. Since the output layer typically uses a different activation function from the hidden layers, a sigmoid function was selected. The sigmoid function is widely used in neural network research due to its smooth and S-shaped curve, which maps input values to a range between 0 and 1. This characteristic makes it

particularly suitable for binary classification tasks and probabilistic interpretation. Moreover, its first derivative is computationally convenient, facilitating gradient-based optimization methods [61]. The sigmoid function is an efficient way of producing an output $p \in [0, 1]$, which can be interpreted as a probability. Plots of the ReLU function and sigmoid function are shown in Figure 12 and Figure 13.

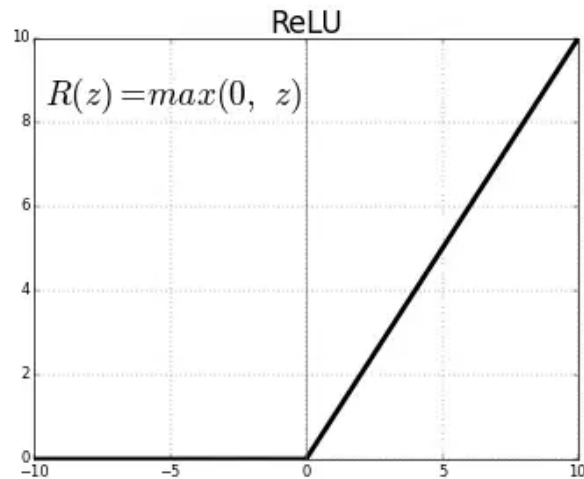


Figure 12 The curve of the ReLU function

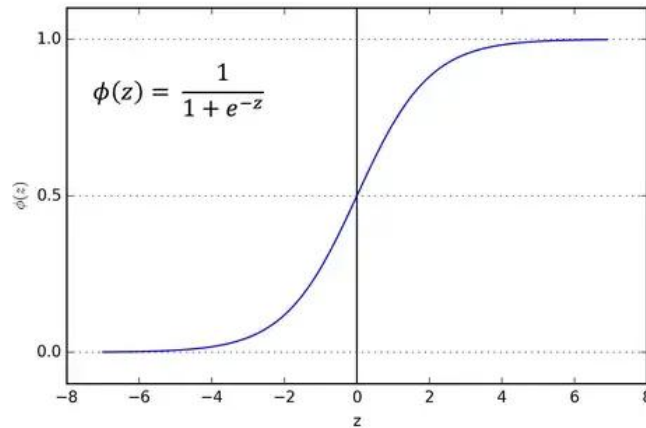


Figure 13 The curve of the sigmoid function

This study involved running programming tasks on an HP Laptop 15t-dy100 featuring an Intel(R) Core(TM) i5-1035G1 CPU, operating at a processing speed of 1.00GHz. The coding process to train the ANN model was conducted using Jupiter Notebook software, an integrated development

environment (IDE). Figure 14 illustrates the sequential steps in constructing an ANN model. The training duration varied depending on the dataset's size and characteristics, influenced by the combination of hardware and software. For instance, training a smaller data set of around 60 data points took approximately 3 minutes, whereas training a larger data set comprising approximately 3000 data points required about 30 minutes.

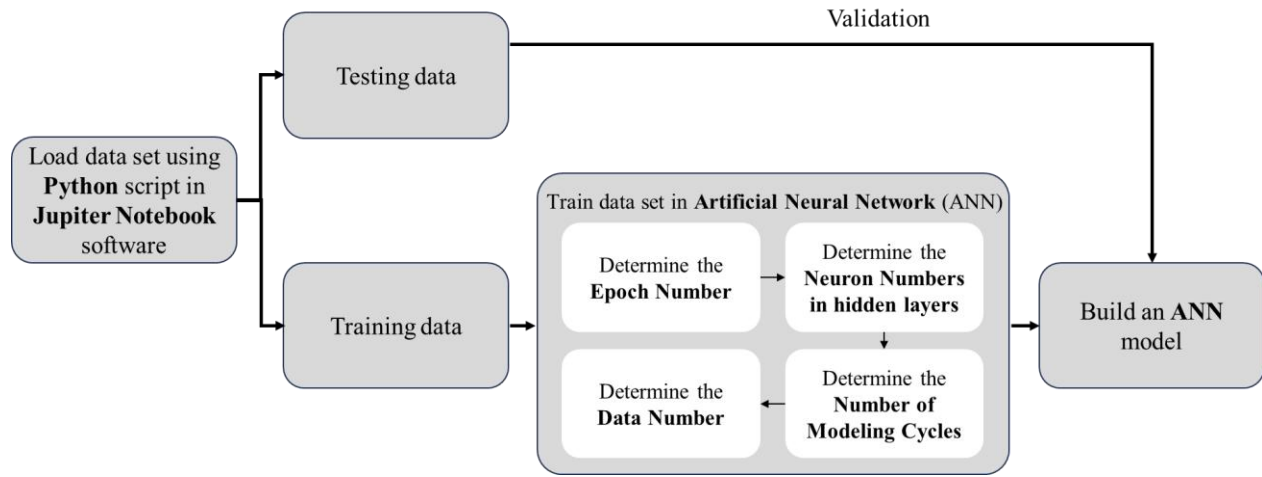


Figure 14 Flow for building an ANN model for BCS estimation

2.4 ANN AND MCKEE DATE SET

Like many of the modeling efforts in the industry, we begin our exploration of the applicability of ANNs on box compression estimation with the work of McKee et.al. Their model was built using 63 data points including A-, B-, and C-flute boxes. This data set captured information on ECT, flexural stiffness in the machine and cross-machine directions of the combined board (EI_x and EI_y), thickness of the corrugated board, and the length, width, and depth of the box. Those seven physical parameters serve as the input parameters for an ANN model with BCS as the output, as shown in Figure 11. Of note, these parameters are not independent - flexural stiffness depends in part on the thickness of the board. Including all the available parameters in the data allows the ANN to appropriately assess the relative importance of each parameter to BCS estimation.

To assess our ANN given the limited data presented by McKee et.al., the 63 data points were split into 42 training data points and 21 testing data points. Two hidden layers were implemented to generate the output value (BCS). We initially considered utilizing 200 epochs for conducting the calculations.

Model accuracy and consistency can be influenced by many modeling factors. The first task in developing an ANN model includes optimizing neuron number combinations in each of the hidden layers. We implemented an exhaustive search method [62] examining different neuron number combinations in the various layers, as shown in Figure 15: To assess the model accuracy, the neuron numbers for the first hidden layer were examined ranging from 80 to 184 (with an increment of 8). Similarly, for the second hidden layer, the neuron numbers were examined from 24 to 42 (with an increment of 3). The MSE was calculated for each combination to evaluate the model's accuracy. In each case a random selection of data points from the underlying data set was used, which has implications on the robustness of the minimum MSE. The minimum MSE occurred with 160 neurons in the first hidden layer and 36 in the second hidden layer. To confirm this result, the increments for both hidden layers were reduced. The increment of 8 in the first hidden layer decreased to 2, and the increment of 3 in the second hidden layer decreased to 1. Remarkably, even with these decreased increments, the minimum MSE still occurred with the same combination of neuron numbers. The same structural framework was maintained for analyzing the McKee data to ensure consistency and comparability in the evaluation process. Notably, this design choice allows for a significantly higher degree of freedom in the model compared to the amount of available data in the McKee dataset. This imbalance suggests that the model has the flexibility to capture a wide range of potential relationships and interactions within the data, which may not be fully constrained by the limited dataset. As a result, the excess degrees

of freedom could contribute to the observed variations across different parameter combinations, as illustrated in Figure 15.

Mean Square Error		Neurons' number (2nd hidden layer)						
		24	27	30	33	36	39	42
Neurons' number (1st hidden layer)	80	1559	1837	1842	1533	1675	1681	1672
	88	1643	1742	1357	1459	1468	1555	1579
	96	1718	1379	1334	1477	1691	1343	1414
	104	2091	1886	1688	1601	1452	1556	1254
	112	1545	1846	1447	1530	1343	1512	1423
	120	1335	1925	1587	1236	1275	1463	1201
	128	1431	2123	1446	1515	1619	1447	1322
	136	1850	1712	1429	1311	1383	1180	1485
	144	1402	1322	1573	1427	1379	1301	1315
	152	1471	1213	1139	1236	1304	1481	1314
	160	1605	1597	1484	1376	1069	1363	1393
	168	1390	1512	1377	1326	1295	1457	1403
	176	1396	1267	1477	1253	1223	1424	1251
	184	1282	1473	1385	1318	1210	1479	1171

Mean Square Error		Neurons' number (2nd hidden layer)		
		35	36	37
Neurons' number (1st hidden layer)	152	1301	1304	1313
	154	1260	1154	1296
	156	1441	1236	1208
	158	1391	1327	1387
	159	1502	1452	1381
	160	1482	1069	1213
	161	1292	1237	1331
	162	1395	1318	1305
	164	1434	1395	1352
	166	1442	1285	1555
	168	1266	1295	1192

Figure 15 Mean Squared Error (MSE) calculations for the model using McKee data with varying numbers of neurons in each of two hidden layers for the McKee data set. The error is minimized for 160 neurons in the first layer and 36 neurons in the second

To understand the computational load further, we explored how the number of epochs impacted model convergence. While again this calculation is not resource intensive for a small data set like the one provided by McKee et.al., it becomes critical to stop the process at convergence once the data set grows. Figure 16 illustrates that the MSE begins to converge in less than 50 epochs. As the number of epochs increases up to 200, the rate at which the MSE decreases gradually slows down, indicating that each additional epoch yields only a small incremental improvement. This phenomenon suggests that the model's performance is reaching a plateau, where further training offers minimal benefits relative to the computational cost. To address this challenge and strike an optimal balance between computational time and model accuracy, we implement a stopping criterion: the training process is halted once the MSE reduction rate falls below 3.0%. This threshold serves as a practical indicator that the model has achieved sufficient convergence, and that additional training is unlikely to significantly enhance performance. In our experiments, this

threshold is typically reached at approximately 100 epochs, ensuring that we maintain optimal computational efficiency while still achieving a robust level of accuracy. Consequently, this approach not only saves valuable computational resources but also minimizes the risk of overfitting by preventing unnecessary prolonged training.

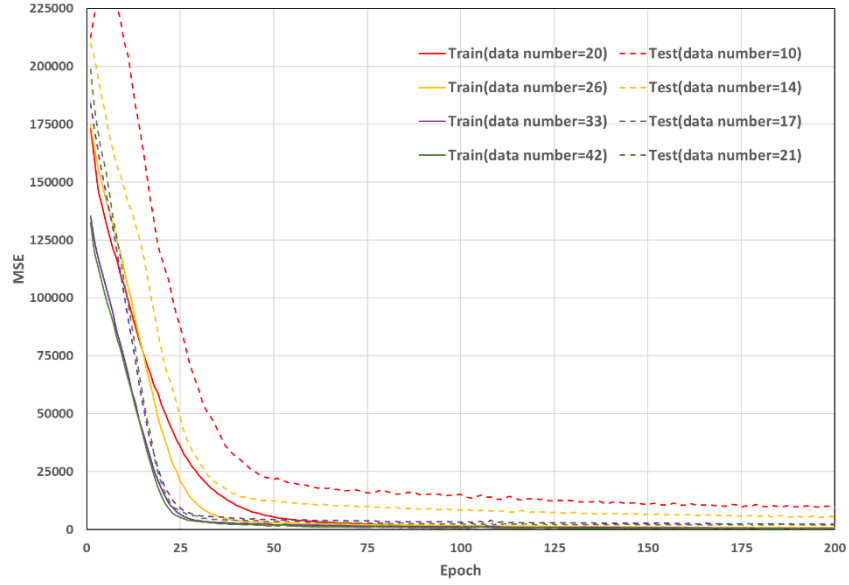


Figure 16 MSE versus epoch plot of different data numbers (McKee data set)

The number of data points plays a very important role in the ANN accuracy. Figure 17 displays how the number of data points influences the ANN accuracy. Exploring different total population sizes from 30 to the full data set of 63 points, the chosen population was randomly divided into training data ($2/3$ of the points) and testing data ($1/3$ of the points). In the modeling process, partitions of underlying data vary from modeling cycle to cycle. This can have a significant impact on the model accuracy due to some special data points may fall into training data in a given cycle and fall into testing data in the subsequent cycle. To assess the impact of variation in the input data on model results and predictions, the process of partitioning a data set into training and testing data was regularly repeated. Multiple modeling cycles were performed. For the McKee data set, a sufficient number of “modeling cycles” of 60 were performed in each population size from 30 data

points to the full data set. Overall, the calculation process reflects a confidence interval on the model output related to the breadth of variation in potential input data sets. The confidence intervals around the mean error reflected the ANN training accuracy, as shown in Figure 17. As expected, ANN accuracy increases with population size. For this small data set like the McKee data set, ANN accuracy of the whole data set is notably higher than that of the partial data set, which concludes that the whole data set is needed for the McKee data set to minimize the error.

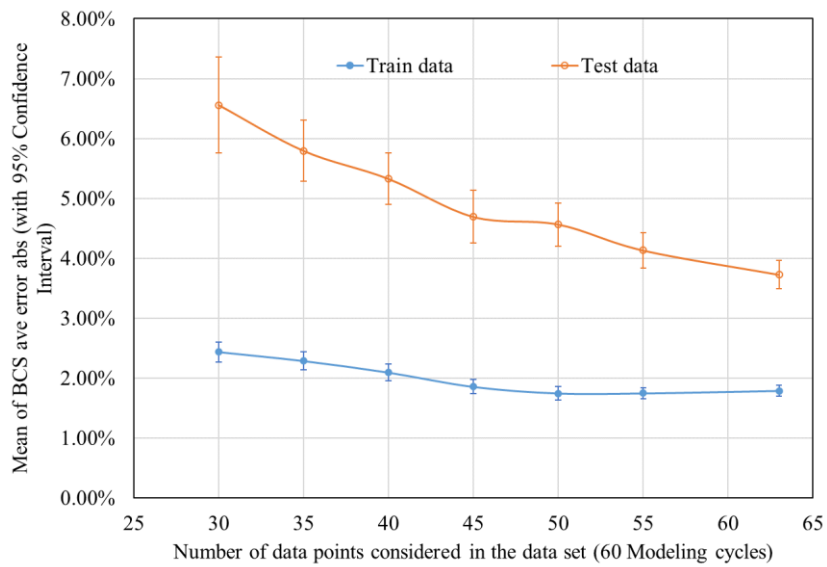


Figure 17 Average error in estimated box compression strength given different subsets of the data (McKee data set), each run through 60 modeling cycles. Note error bars indicate 95% confidence intervals on the mean values

Since ANN randomly splits the data into training and testing data in the modeling process, each modeling cycle can have different underlying data partitions. As a result, each modeling cycle can generate a unique model that optimally fits the training data provided but can produce very different values for the error when assessing the testing data. Therefore, it is important to understand how many modeling cycles are required to have results converge to a “typical” reliability. To investigate the impact of different underlying data partitions on the accuracy of the ANN, various numbers of modeling cycles were examined. Figure 18 shows that when we

partitioned the full McKee data set (all 63 points), the training data accuracy remained relatively consistent as the number of evaluation cycles increased. The average testing data accuracy converged after roughly 60 rounds of testing, very similar to the total number of data points in the database.

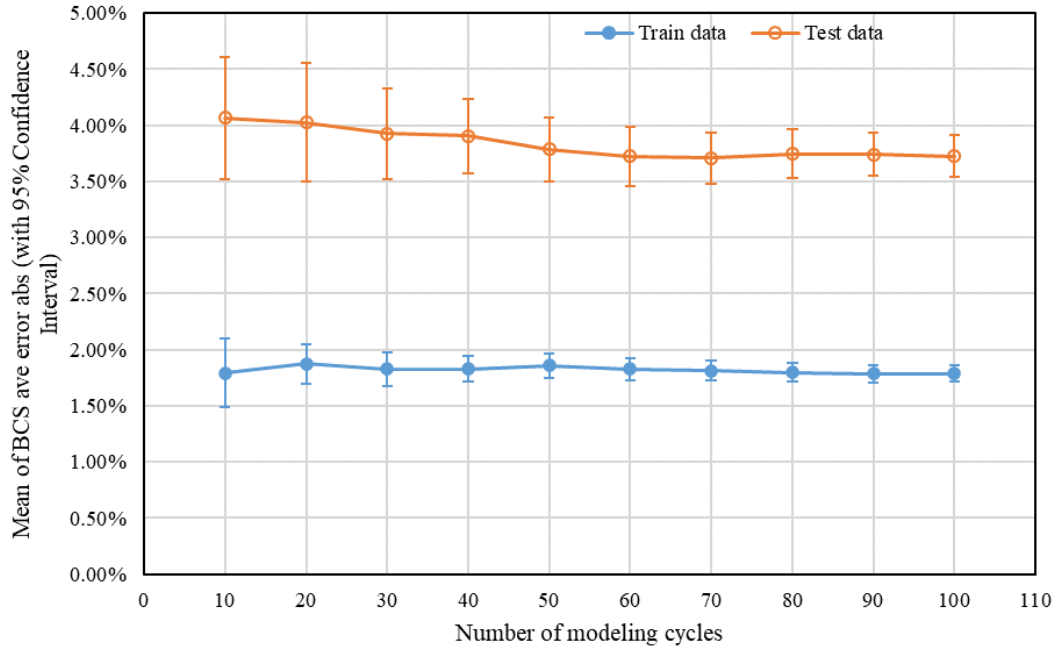


Figure 18 Mean of average error in estimated box compression strength given different numbers of modeling cycles. Note that the testing data values converge after 60 modeling cycles (McKee data set). Error bars indicate 95% confidence intervals on the mean values

We have explored four modeling factors common in the ANN process using the data presented by McKee: the combination of neuron numbers in hidden layers, the number of epochs, the number of modeling cycles, and the number of data points in a data set. An optimal combination of neuron numbers in hidden layers can minimize the MSE and increase the ANN accuracy for BCS estimation. As the epoch number increases, the MSE reduction rate becomes increasingly slow. To strike a balance between computational time and accuracy, a stopping point when the MSE reduction rate falls below 3.0% was selected to ensure optimal computational efficiency without

significantly compromising accuracy. Consistency for the ANN model is realized when the number of modeling cycles reaches a critical number for a given population size, and a minimum number of data points can be identified at which the MSE is minimized, and the ANN is most robust. We carry these observations forward into our analysis of larger, more generalizable data sets.

2.5 ANN AND AN IDEALIZED DATA SET

McKee et.al.'s simplified model for box compression strength can be used to explore the applicability of ANN to compression strength estimation. However, the limited size of their data set constrains the ANN approach as noted above. Therefore, a larger data set is desirable. Using the simplified McKee equation [22] as shown in equation (1), a synthetic data set could be generated.

In this way, the “idealized” data set was created with 3,009 data points. These data points represent boxes with ranges in length, width, aspect ratio, ECT, thickness and flute types (B- & C-flute) commonly used in North America (Table 1). Note that each “data point” discussed in this section is a specific set of information defining the physical properties of the box (lengths, width, thickness, and ECT) and the associated BCS calculated using equation (1).

Table 1 Minimum and maximum values of the data incorporated in the idealized data set

Property	Min	Max
Length (cm)	19.05	99.38
Width (cm)	12.70	76.96
L/W (aspect ratio)	1.00	4.00
Perimeter (cm)	71.12	346.71
Thickness (cm)	0.26	0.44
ECT (lbs/inch)	64.77	228.35

Given the “perfect” nature of the fabricated data set, it is obvious that a simple least-squares fit of equation (1) to the input parameters reproduces the BCS values with 100% accuracy and 0% error. With a data set this large, one might also hope to overcome the ANN challenges experienced in

fitting the much more limited data from McKee, and potentially reproduce the expected values in a test data subset perfectly, with close to no variation from the actual values.

To start this process, 67% of the data set (2,016 randomly selected samples) were used for the ANN training process and the rest were used for evaluation of the resulting model. With 200 epochs, the optimal neuron number combination in the hidden layers was again explored using an exhaustive search method. Figure 19 displays the examination of neuron numbers in the first hidden layer, ranging from 128 to 160 (with an increment of 8), as well as the examination of neuron numbers in the second hidden layer, ranging from 24 to 48 (with an increment of 3).

SUM MSE of train data and test data		Neurons' number in second hidden layer								
		24	27	30	33	36	39	42	45	48
Neurons' number in first hidden layer	128	17	18	16	15	16	16	16	16	14
	136	19	14	16	16	17	15	15	15	16
	144	16	14	17	17	16	14	13	12	15
	152	18	18	18	17	14	15	13	14	13
	160	16	14	16	14	17	16	13	14	15

SUM MSE of train data and test data		Neurons' number in second hidden layer						
		36	39	42	44	45	46	48
Neurons' number in first hidden layer	136	17	15	15	16	15	15	16
	138	14	13	15	14	17	15	18
	140	14	13	16	15	13	16	16
	141	14	15	15	17	16	14	15
	142	14	15	14	13	12	17	16
	143	14	14	14	17	14	15	16
	144	16	14	13	14	12	15	15

Figure 19 Mean Squared Error (MSE) calculations for the model using idealized data with varying numbers of neurons in each of two hidden layers for the idealized data set. The error is minimized for 142 neurons in the first layer and 45 neurons in the second

The MSE was calculated for each combination to assess the model's accuracy. The MSE was minimized with 144 neurons in the first hidden layer and 45 neurons in the second hidden layer. To validate this result, the increments for both hidden layers were reduced to 1. Notably, the minimum MSE was observed with 142 neurons in the first hidden layer and 45 neurons in the second hidden layer. As with our McKee analysis above, this number of neurons provides more degrees of freedom in our modeling space than data sets in our model population. Note that the MSE is much lower than for the McKee data set because the data is perfect. However, the values

are not zero indicating some residual uncertainty in the estimation of BCS even for this idealized data.

In exploring the idealized dataset, we employed an analytical approach similar to that used in the McKee dataset analysis within the context of ANNs. Specifically, we investigated how varying the number of training epochs impacted the model's MSE, with these findings initially illustrated in Figure 20-a. As the training progressed and the number of epochs increased to 200, we observed a general downward trend in the MSE, indicating improvements in model performance. However, this improvement was not entirely smooth; significant fluctuations were present in the MSE values, suggesting intermittent variability in the learning process. As the number of epochs continued to increase, the rate at which the MSE decreased began to slow down, highlighting a diminishing return in performance gains with additional training. To address the challenge posed by these fluctuations and to provide a clearer, more interpretable view of the overall trend, we applied the Moving Average technique [63]. This method effectively smoothed out short-term irregularities, resulting in a refined graphical representation of the MSE trend that is depicted in Figure 20-b. The smoothed graph not only offers a more consistent and reliable perspective on the model's performance improvements over time but also aids in the identification of the optimal training duration. By reducing the visual noise in the MSE curve, we can more accurately pinpoint the stage at which additional epochs no longer yield significant benefits, thereby enhancing our understanding of the model's learning dynamics and informing decisions regarding computational resource allocation. This comprehensive analysis ultimately provides valuable insights into the balance between training efficiency and model accuracy, further demonstrating the robustness of our approach when applied to both idealized and real-world datasets. Moreover, these findings underscore the importance of tailoring the training process to the specific characteristics of the

dataset at hand, paving the way for future research into adaptive training strategies and alternative smoothing techniques that could further optimize model performance.

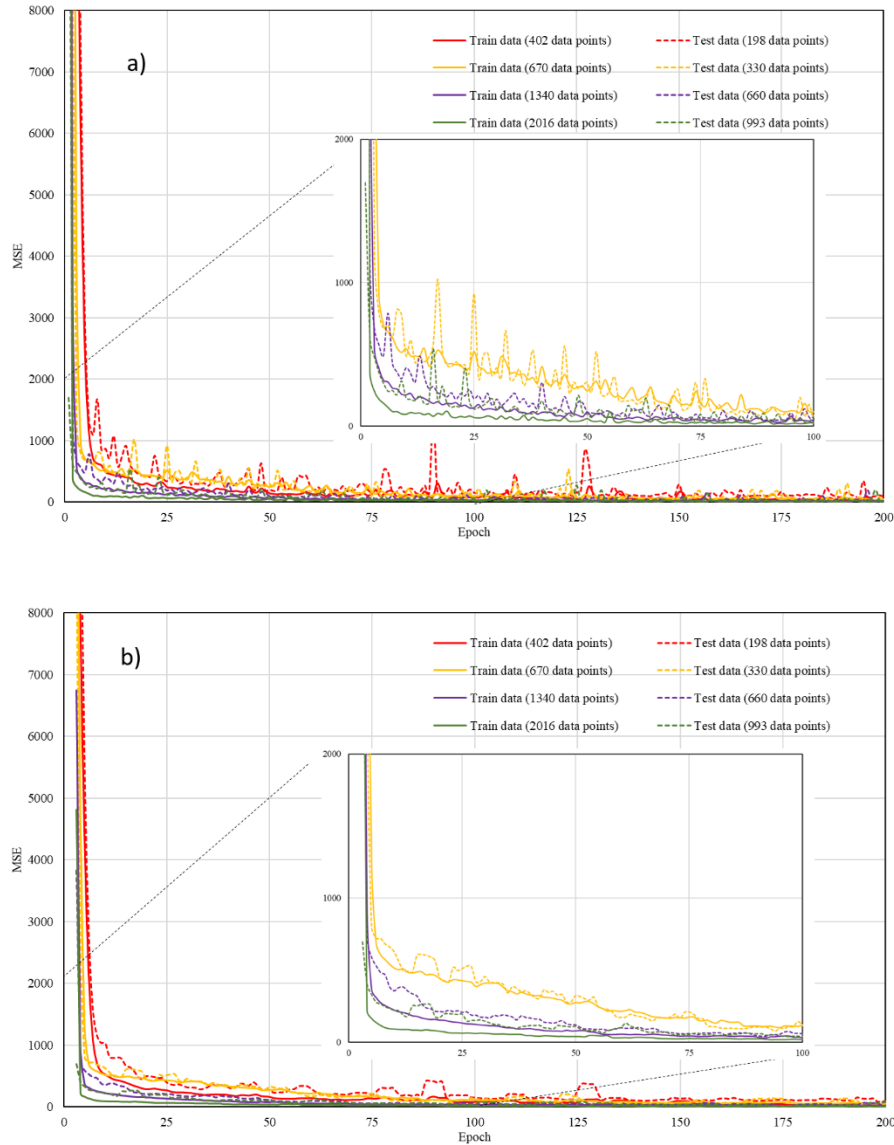


Figure 20 Mean Squared Error (MSE) of the fits as a function of epochs for different sized data subsets from the idealized data set. 6a displays the raw MSE calculated for each epoch, while 6b presents smoothed data, more clearly displaying the asymptotic nature of the functional relationships

This revealed that the MSE experienced a rapid decrease before reaching 50 epochs. From 50 to 200 epochs, the MSE decreased steadily, and the large fluctuations disappeared after 140 epochs.

Similar to the study of the McKee data set, to strike a balance between computational time and accuracy, a stopping point was selected when the MSE reduction rate falls below 3.0% after applying the Moving Average technique. This threshold is typically reached at approximately 140 epochs, ensuring optimal computational efficiency without significantly compromising accuracy. When examining the full data set of 3,009 data points, the ANN accuracy remained relatively consistent while the confidence interval around the mean error decreased as the number of modeling cycles increased (Figure 21).

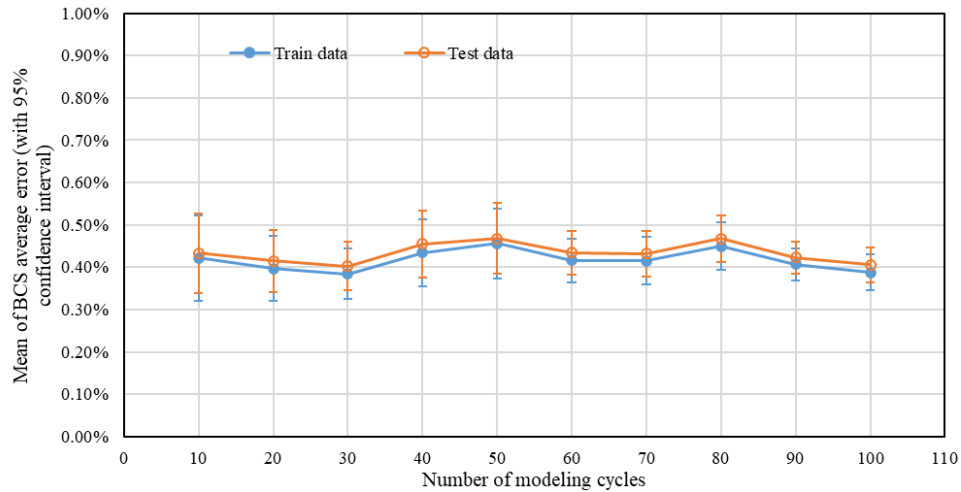


Figure 21 Mean of average error in estimated box compression strength given different numbers of modeling cycle (Idealized data set). Note error bars indicate 95% confidence intervals on the mean values

To better understand why the error in the model was not zero as might be expected for a model fitting “perfect” data, the specific results from each cycle were examined. It was observed that the BCS errors of four data points in particular always showed higher estimation errors than other data points. Those four data points are at the limits of the data set (or boundary data points). Figure 22 shows the frequency of the actual BCS values. As is typical for data at the end points of a distribution, these four points have excessive leverage in the modeling. Their impact on model accuracy in test data depends on what adjacent points happen to be in the training data. When the

boundary data points are randomly selected to be part of the testing data and thus do not appear in the training data, the result tends to show higher BCS average error. The average error across multiple cycles is impacted by this contribution.

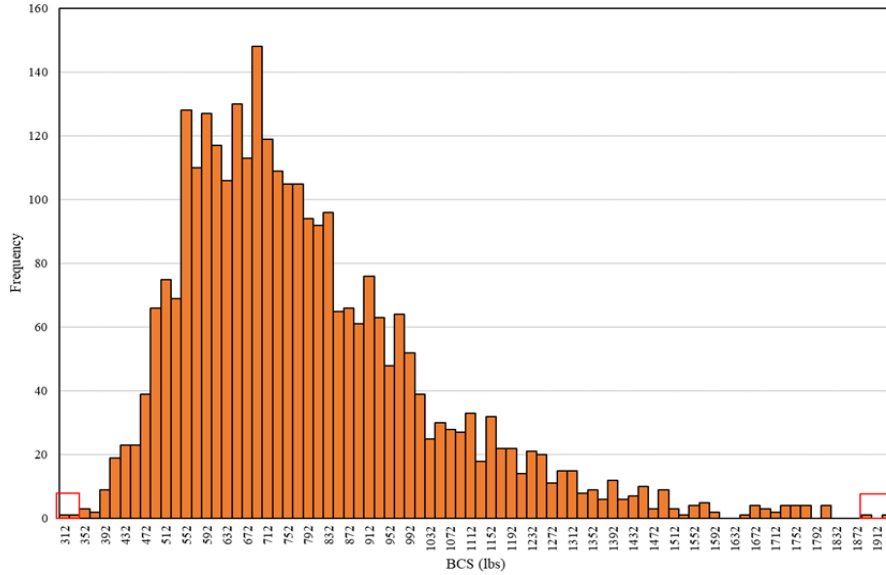


Figure 22 BCS distribution of 3,009 data points (Idealized data set)

To see the influence of population size on the ANN accuracy for the perfect model (similar to Figure 17 above), we examined populations from 600 to 3,009 data points using 10 modeling cycles. The results show that the mean of BCS average errors fluctuates notably when we consider a limited number of data subsets (Figure 23). Even for this larger population, the impact of limiting population size remains significant if the iterative process is not executed for a sufficient number of cycles. In our analysis, we observed that over the course of 50 modeling cycles, the mean BCS average error exhibited a steady decline as the volume of included data increased. However, this trend reached a plateau at approximately 1,500 data points, beyond which additional data did not yield noticeable improvements in accuracy. This finding suggests a crucial relationship between the number of modeling cycles and the population size, indicating that both factors must be carefully considered together to optimize model performance. If the population size is too small,

even a high number of iterations may not be sufficient to minimize error effectively. Conversely, if the process is not iterated enough times, the model may fail to fully leverage the benefits of a larger dataset. Therefore, achieving an optimal balance between these two parameters is essential to maximizing model accuracy and reliability.

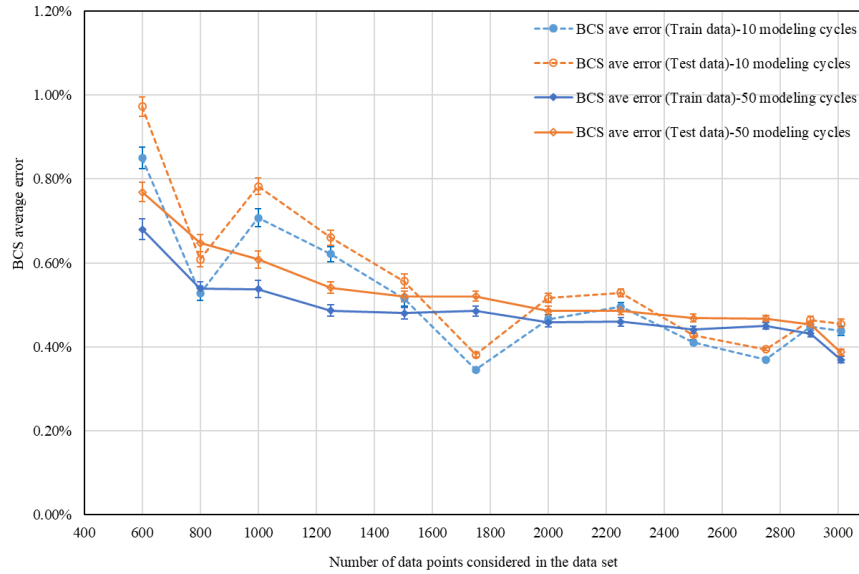


Figure 23 Average error in estimated box compression strength given different numbers with same modeling cycles of 10 and 50 (Idealized data set). Note error bars indicate 95% confidence intervals on the mean values

The combination of neuron numbers, the number of epochs, the number of modeling cycles, and the number of data points impacts the accuracy of the ANN prediction. Even when using the full data population (>3000 data points) and many modeling cycles on a perfect data set generated by a closed form equation, the average relative error of the BCS prediction is not zero. From Figure 23, in conjunction with Figure 21, this analysis identifies the error contribution of the ANN approach itself at around 0.4% when estimating BCS from this type of data and data sets of this size. This residual error is independent of any physical properties; rather, it arises from the ANN process itself. As such, we would expect it to be additive 10. to any other errors that may arise in

using a model for prediction, including measurement errors of the input parameters to the model as well as fundamental errors in the model functional form.

2.6 ANN AND A DATA SET WITH VARIATION

Variation occurs naturally in all processes. Typical variations in measurement of inputs associated with the performance of a corrugated box are on the order of 4-5% for measured quantities like ECT and BCS. To further study if and how the ANN model works while handling a data set incorporating variation, we modified the ideal set to represent boxes that might appear in commerce. We added fluctuations to each input value, using randomized, normally distributed values on the order of the variation observed in the different test methods. As with the idealized data set, a “data point” represents a specific set of information defining the physical properties of the box (lengths, width, thickness, and ECT) and the associated BCS calculated by equation (1). The average absolute difference between the new predicted BCS values for the 3,009 data points and the “actual” BCS of the idealized model was obtained by adding variations to the input parameters and calculated by equation (1). This process was carried out to achieve a variation of $\pm 5.4\%$ for BCS. We then followed the same process as for the idealized data set: 67% of the data set (2,016 randomly selected samples) were used for the ANN training process and the remaining were used for evaluation of the model. We used the same number of epochs and neuron numbers in hidden layers as in the idealized modeling.

To explore the impact of the number of epochs on the convergence behavior of the ANN model, we conducted experiments by running the model on different subsets of data for up to 250 epochs. In these experiments, we observed that the MSE decreased rapidly during the first several epochs across all data subsets, as illustrated in Figure 24. Notably, the largest dataset consistently achieved the lowest MSE for any given epoch, indicating that a greater volume of data can enhance the

learning efficiency and accuracy of the model. In alignment with our earlier modeling efforts, Figure 24-a displays the raw MSE values computed at each epoch, providing a detailed view of the initial rapid improvement followed by a gradual tapering in error reduction.

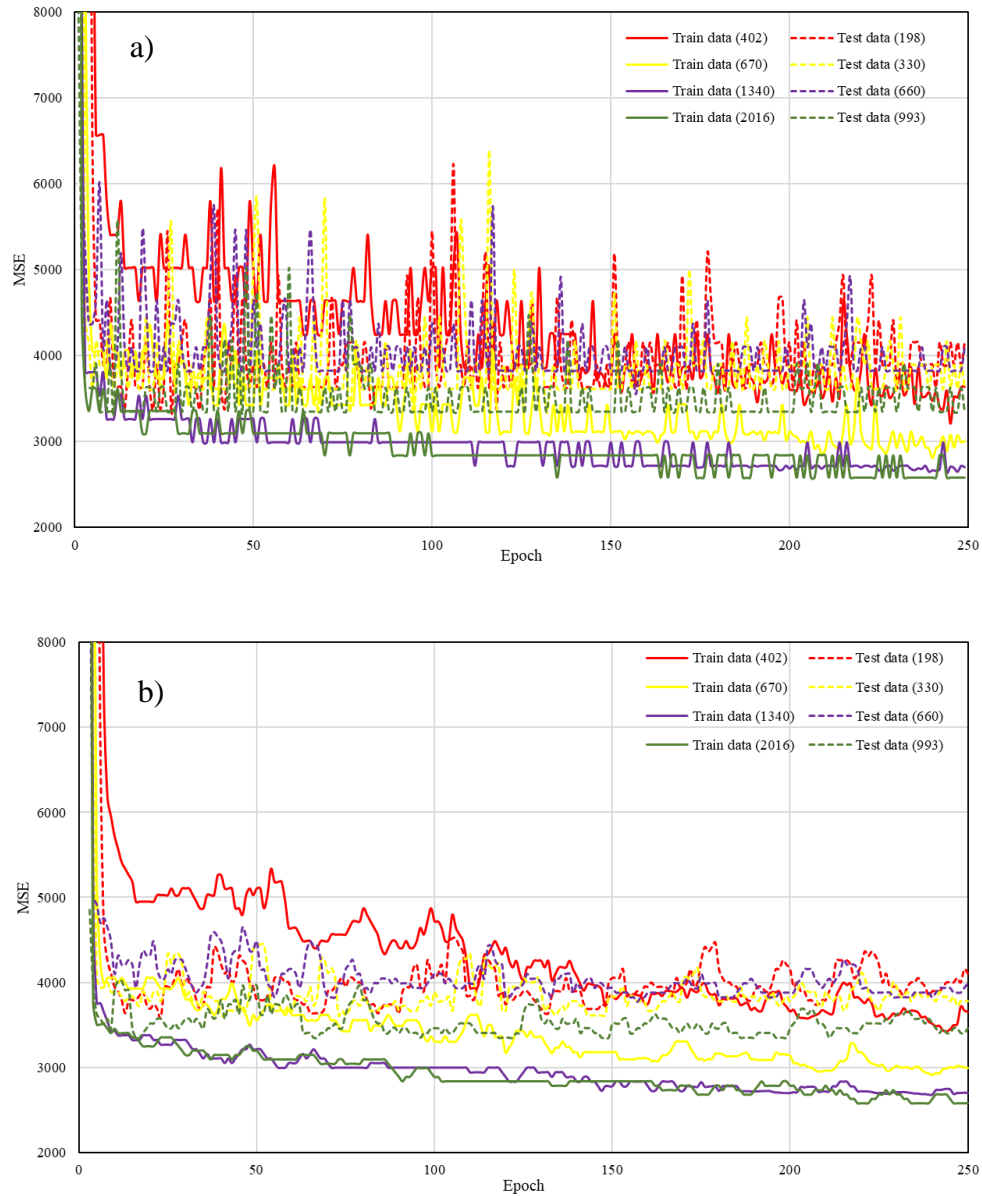


Figure 24 Mean Squared Error (MSE) as a function of epochs for different sized data subsets from the variation data set. 10a displays the raw MSE calculated for each epoch, while 10b presents smoothed data, more clearly displaying the asymptotic nature of the functional relationships

To further elucidate the long-term convergence behavior and to reduce the impact of short-term fluctuations, Figure 24-b presents a smoothed version of the data using an appropriate smoothing technique. This smoothed graph clearly highlights the asymptotic behavior of the MSE, allowing us to discern the point at which additional epochs yield diminishing returns. As expected, given the deliberate addition of variation to the input data, the MSE values observed in these experiments are considerably higher than those recorded for the idealized dataset shown in Figure 20.

This contrast underscores the challenges introduced by increased data variability and emphasizes the need for robust modeling strategies when working with more complex, real-world datasets.

We modeled different population sizes as above to again identify the influence of population size on ANN accuracy (Figure 25).

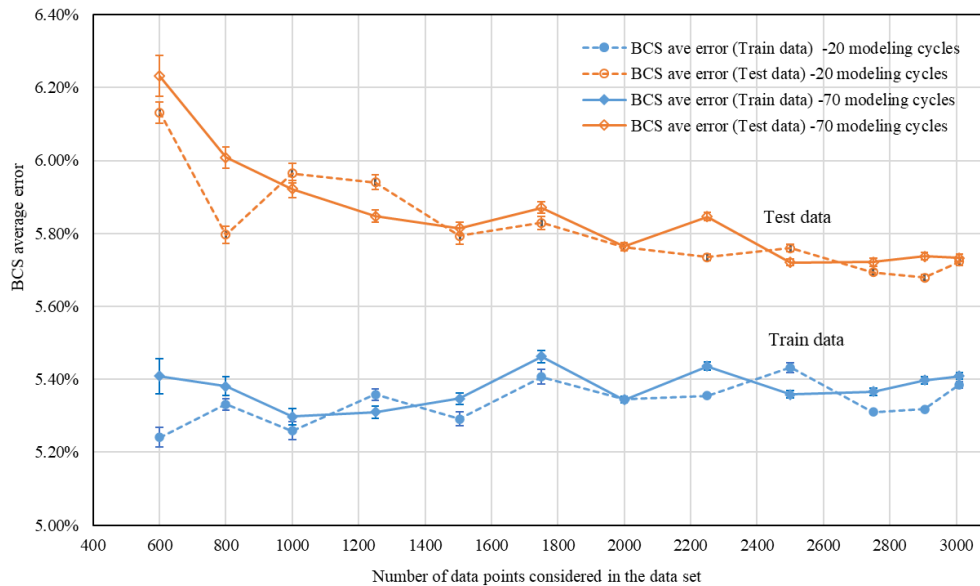


Figure 25 Average error in estimated box compression strength given different numbers with same modeling cycles of 10 and 50 (Data set with Variation). Note error bars indicate 95% confidence intervals on the mean values

The accuracy of both the training data and testing data remained relatively consistent as the number of modeling cycles increased. While the accuracy of the training data was in line with expectations

from the variation built into the data set (~5.4%), the influence of limiting population size can have a meaningful impact if we don't iterate the process sufficiently. Notably, the ANN approach was not working with any more information than the closed form equation itself, and so the prediction accuracy did not improve upon what we would get from the closed form equation. Test data accuracy didn't begin to converge until around 1500 data points when we used 20 modeling cycles, yet accuracy occurred slightly sooner (~1250 data points) when we used 70 modeling cycles. The BCS average error levels out at 2,500 data points, nearly the entire data set, at a value combining the inherent uncertainty in the input data and the uncertainty of the ANN process itself, identified above. This is notably larger than for the idealized data, because of the influence of variation in the input parameters. As with the idealized data above, the influence of modeling cycles and population size need to be considered together. The minimum data population size to get a robust result is also larger for the variation data set.

2.7 CONCLUSION

In this session of our study, we explored BCS estimation using Artificial Neural Networks across input data sets that included both actual data from the literature and data based on literature models. Partitioning each data set into test and training subsets and running multiple modeling cycles on different partitions provides an analysis of average model estimation accuracy that can be expected when the resulting models encounter new data. An ANN model with high accuracy and consistency can be built by adjusting four modeling factors: the combination of neuron numbers in hidden layers, the number of epochs, the number of modeling cycles, and the size of the data set. All four interact to influence model accuracy and can be optimized by minimizing model MSE. The combination of neuron numbers in the two hidden layers was determined as 160 and 36 for the McKee data set, and 142 and 45 for the idealized data set. Employing the same stopping

criterion, where the MSE reduction rate is required to be below 3.0%, the epoch numbers were established as 100 for the McKee data set and 140 for the idealized data set. To ensure a robust result with high consistency in the ANN, it was found that 60 modeling cycles are needed for the McKee data set, 50 modeling cycles are required for the idealized data set, and 70 modeling cycles are necessary for the data set with variation. The data size needed to get a robust result varies based on the input data variations and can be identified by minimizing average BCS error: For the McKee data set, 63 data points are not enough for an ANN to predict the BCS reasonably. The other two data sets (idealized data set and data set with variation) need at least 1000 data points to get a robust result for ANN prediction. The data size needed is significant and data collection can be expensive considering the physical testing required. Our ANN models had more degrees of freedom than the number of underlying data sets, which might lead us to expect that we could perfectly fit the underlying data and achieve BCS estimations very close to “measured” values. Instead, we found that model estimation accuracy remains limited by the uncertainty or error in the input parameters combined with uncertainty from the ANN process itself. The variation of input parameters had a positive correlation with an ANN process (high variation increases the training error and vice versa). By identifying the challenges of small data sets and the interrelationship between modeling parameters and the estimation error in the data space, this study provides a methodological guide for future research exploring the applicability of ANN approaches to address problems and answer questions in the corrugated industry.

CHAPTER 3: EVALUATION OF PACKAGING DESIGN RELATIVE FEATURES IMPORTANCE USING ANN

3.1 INTRODUCTION

This chapter focuses on leveraging artificial neural network (ANN) models to evaluate the relative importance of packaging design features. In this section, Box Compression strength (BCS) was used as a representative packaging property, and the relative importance of up to six packaging design features was assessed using four ANN-based methods (the weight connections method, the permutation method, the gradient-based method, and SHAP values) based on the reliability of feature importance rankings. Two datasets were utilized: one synthetic dataset generated using a widely used mathematical model (the McKee formula) and one real dataset [26]. These datasets were used to train ANNs to assess packaging design feature importance. Theoretical feature importance was calculated and compared with the feature importance from the four ANN-based methods. This result demonstrates the feasibility of applying the ANN approach to evaluating the relative importance of packaging design features.

3.2 CURRENT METHODS FOR EVALUATING BCS FEATURE IMPORTANCE

Packaging design plays a vital role in ensuring packaging performance. Effective design can reduce costs by minimizing material usage and waste while protecting products during transportation, storage, and handling. To improve packaging performance, the design process must consider various influencing factors, each of which impacts specific packaging properties differently. Understanding the relative importance of these influencing factors for packaging design, or packaging design features, is crucial for enhancing packaging performance and maximizing cost savings.

The evaluation of the relative importance of packaging design has traditionally relied on conventional physical testing, which involves numerous mechanical tests to obtain accurate measurements, making the process both resource-intensive and time-consuming. Consequently, several new methods for assessing the importance of packaging design features have emerged in recent years. Techniques such as the Analytical Hierarchy Process (AHP) [64, 65], and Finite Element Analysis (FEA) [27, 66, 67] have been developed to optimize different packaging systems. Alicia Pérez et al. (2020) applied AHP to optimize a company business strategy of corrugate cardboard boxes to support multicriteria decision-making, generating multiple improvements, such as the reduction of the overall cost, the optimal fill rate operations, and the articulation of the strategic and functional decisions in this organization [65]. Jongmin Park et al. (2020) investigated the edgewise compression behavior (load vs. displacement plot, ECT, and failure mechanism) of corrugated paperboard based on different types of testing standards and flute types using finite element analysis (FEA) and experimental analysis [67]. However, these methods are not consistently applied or fully integrated into industry practices. For instance, a primary disadvantage of AHP is expert subjectivity [68]. AHP relies on expert input for pairwise comparisons between options, where experts evaluate the relative importance or performance of one option over another. These judgments, being influenced by personal opinions, can introduce subjectivity [69]. On the other hand, FEA requires material properties that are challenging to obtain due to the anisotropic and non-linear mechanical behavior exhibited by paper fibrous material [70]. Additionally, there are limited applications of these methods for systematically evaluating the importance of packaging design features. Therefore, there is a substantial opportunity to develop a more efficient approach for assessing the feature importance for packaging design.

3.3 ANN APPROACH FOR EVALUATING FEATURE IMPORTANCE

Despite the limited advancement in methods for evaluating packaging design feature importance, researchers have extensively explored various methods for evaluating the relative importance of input variables. In the last decades, Artificial Neural Networks (ANNs) have gained growing interest in various engineering and multidisciplinary research fields, such as the tourism industry [71], the financial sector [72], and complex engineering applications [73]. However, in the field of packaging, Artificial Neural Networks (ANNs) have been applied in only a few areas, such as estimating edge crush resistance and evaluating other packaging properties [74, 75], with limited applications beyond these.

Tomasz Gajewski et al. (2024) predicted the crush resistance of corrugated packaging boxes with ventilation openings, packages with perforations, and typical flap boxes using different ANN models [76]. Siripong Malasri et al. (2016) trained a small data set of 74 box samples using ANN to predict the compression strength of cubical RSC single-wall corrugated boxes [34]. ANNs have been a focal point due to their capability to generalize complex non-linear problems.

ANN modes have been implemented to evaluate the relative importance of input variables using many methods. Some commonly used methods include the weight connections method [77], the sensitivity analysis method, the gradient-based method [78], and SHAP (Shapley Additive exPlanations) values [70]. Within the weight connections method, the measures of input variable importance rely on the connection strengths (weights) within a trained neural network [79]. In 1991, Garson et al. (1991) proposed a method to determine the relative impact of each input variable by calculating the percentage of output weight values associated with the contribution of a single input across the entire network [79, 80]. Yoon et al. (1994) provided a representation of the relative contribution of input i with respect to the overall behavior of the neural network [81].

Compared with Garson's method, Yoon's method considers the direction (positive/negative) of the contribution of an input. In 1999, Howes and Crook also proposed a formulation to determine the relative influence of input variables in neural networks [77, 79], which is similar to Yoon's method. However, it normalizes the effect of extreme weights connecting input and hidden nodes and is the only method measuring the importance of the variables in multiple hidden layers. Additionally, SH Tsaur et al. (2002) [82] defined the input importance scores by taking the sum of the weights connecting the input to the output layer. JD Olden et al. (2021) measured the relative importance of the input variables based on the product of the input-hidden and hidden-output connection weights and summing the products across all hidden neurons [77]. Within the gradient-based method, A Hill et al. (2020) [78] proposed a gradient-based method to identify the relative importance of influencing factors for robotic control by obtaining the gradient of the output with respect to any component of the neural network using the chain rule. As for the sensitivity method, one of the most commonly used techniques is the permutation method. H Mandler et al. (2023) used the permutation method to measure the sensitivity of a model with the presence or absence of a feature to determine the importance scores of input features in fluid dynamics using a neural network-based turbulence model. Regarding SHAP values [83], SM Lundberg et al. [70] presented a unified framework for interpreting predictions, assigning each feature an importance value for a particular prediction in a deep learning model.

These methods have found application in various domains for extracting the influence of inputs in machine learning models. For instance, they have been used to analyze design parameters' impact on complex engineering systems, guest loyalty to hotels in the tourist industry, the relative importance of textual indexes in predicting the future performance of banks, and geographical phenomena visualization, among others. NL da Costa et al. (2021) [77] utilized Garson's method

to evaluate the contribution of inputs to outputs in trained neural networks for both classification and regression problems. Goh, A.T. C. [84] employed Yoon's method to identify the relative importance of input factors influencing cone stresses and soil properties in an ANN model. HF Luoh et al. (2014) [85] applied Tsaur's method to identify moderating effects concerning tour leader age stereotypes, age in-group bias, and respondents' age on perceived roles played by tour leaders. J Iqbal et al. (2023) [86] utilized Olden's connection weights method to predict banks' future performance by identifying the relative importance of textual indexes representing management sentiments in banks' annual reports. K Fukumizu et al. (2012) [87] conducted dimension reduction for both feature extraction and variable selection based on the gradient-based method in supervised learning. A Altmann et al. (2010) [88] estimated the distribution of measured importance for each variable in a non-informative setting based on the permutation method in RandomForest (RF) models. Ziqi Li (2022) [89] applied SHAP values to extract spatial effects for interpreting and visualizing complex geographical phenomena and processes in machine learning models.

Based on the reliability of packaging feature importance rankings of the synthetic dataset generated using the widely used mathematical model, four ANN-based methods were selected to evaluate the importance of the target packaging design features in this study. These four methods are the Connection Weights method, the Gradient-based method, the Permutation method, and SHAP values. The principles underlying each method for extracting input feature importance are explained in this section.

3.3.1 Connection Weights method

The Connection Weights method is a valuable technique for interpreting ANN models. An ANN comprises three types of layers: input, hidden, and output layers, as depicted in Figure 26. Neurons

in each layer are connected by weights, which indicate the strength of the connections between the neurons. The Connection Weights method employs the weight matrix to determine the relative significance of each input feature in relation to the output [77]. The weights in the first layer, which connect the input neurons, can reflect the relative importance of each input feature. Thus, the Connection Weights method derives the relative importance of input features by extracting these weights from the first layer, as illustrated in equation (10) and Figure 26. The get-weights function was employed to extract the weights from the first layer of the ANN model.

$$I_i = \sigma_i \sum_{j=1}^{n_{hidden}} |w_{ij}|$$

Where:

σ_i - The standard deviation of the i th input.

(10)

I_i - The i th input's importance.

n_{hidden} - The number of hidden nodes in the first layer.

w_{ij} - The weight connecting the i th input to the j th hidden node in the first layer.

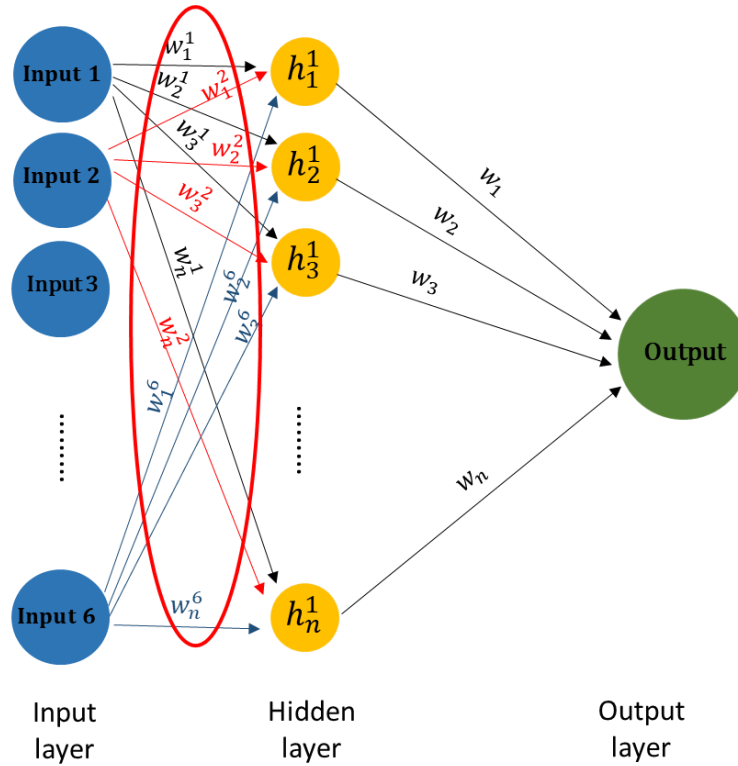


Figure 26 The ANN structure & Connection weights method extracts the weights in the first layer as the input feature importance

3.3.2 Gradient-based method

The gradient-based method is a key technique for evaluating how a model's outputs are influenced by its input features. It involves calculating the partial derivative of the output with respect to the input, which measures sensitivity. This approach is applicable in Deep Neural Networks (DNNs), a specialized type of ANN. [90]. Examining how variations in input features influence the predicted output can reveal insights into feature importance within ANN models. The gradient's magnitude indicates the extent of change in the predicted output due to an infinitesimal alteration in the input feature [91]. The gradient of output \hat{y} with respect to input X is represented in equation (11),

$$\nabla \hat{y} = \nabla F(X) = \left[\frac{\partial F(X)}{\partial x_1} \dots \frac{\partial F(X)}{\partial x_d} \right]^T \quad (11)$$

The differentiability of deep neural networks is determined by the activation function used. Activation functions like sigmoid, ReLU, and Tanh are differentiable almost everywhere. In this study, where the sigmoid function is utilized, a central difference method was applied to numerically approximate the gradient of $F(X)$ at X , as defined in equation (12),

$$\frac{\partial F(X)}{\partial x_k} \triangleq \frac{F(X^{(k+)}) - F(X^{(k-)})}{2\delta x} \quad (12)$$

Where:

$$X^{(k+)} \triangleq X + \delta x \cdot e_k \quad X^{(k-)} \triangleq X - \delta x \cdot e_k$$

$\delta x \in R$ is the step size, and for all $k = 1, \dots, d$, $e_k \in R^d$ is the standard basis vector. The terms $F(X^{(k+)})$ and $F(X^{(k-)})$ are obtained from two forward passes of the model. The importance of the k^{th} feature is then defined as the absolute value of the partial derivative with respect to x^k . This gradient vector provides the feature importance for a single test sample. To determine the global feature importance, the feature importances for all samples in the test set S_N were averaged, where N represents the number of samples in the test set [91], as outlined in equation (13),

$$\begin{aligned} \text{Sample gradient imp } (x_k) &\triangleq \left| \frac{\partial F(X)}{\partial x_k} \right|_X \\ \text{Global gradient imp } (x_k) &\triangleq \frac{1}{N} \sum_{X \in S_N} \left| \frac{\partial F(X)}{\partial x_k} \right|_X \end{aligned} \quad (13)$$

In this study, gradients were computed using `tf.GradientTape()`, a Python tool that allows for nesting to calculate higher-order derivatives.

3.3.3 Permutation method

The Permutation method measures a feature's importance by observing how model accuracy changes when the values of that feature are randomly shuffled while keeping other feature values intact. A feature with higher importance will have a greater effect on the model's accuracy when its values are shuffled [83]. As detailed in equation (14) and Table 2, to determine the feature importance of feature f_j , the column corresponding to f_j is randomly shuffled to generate a corrupted data set D_{kj} . The ANN model is then trained on D_{kj} , and its accuracy is compared with the accuracy of the original model s . The difference in accuracy scores indicates the importance of feature f_j .

$$i_j = s - \frac{1}{K} \sum_{k=1}^K s_{kj}$$

Where:

i_j - Importance for feature f_j .

(14)

s - Reference score of the model m on data set D .

K - Repetition times for randomly shuffling column j of the dataset D to generate a corrupted version of the data named D_{kj} .

s_{kj} - Score of the model on corrupted data D_{kj} .

Table 2 Principle of permutation feature importance

ECT	Caliper	l	w	d	BCT
lb/in	in	in	in	in	lb
46.8	0.197	29.75	26.25	47.5	1780
52	0.200	31.5	29.5	47	1720
...
46.83	0.206	12	6	8	541.5
44.32	0.215	12	6	15	549.8

In this study, the `np.random.permutation()` function from the NumPy library in Python is utilized to randomly shuffle the input features in the ANN.

3.3.4 SHAP values

SHAP (SHapley Additive exPlanations) values provide a method for interpreting the outputs of machine learning models. This approach uses principles from game theory to measure the contribution of each feature to the final prediction [92, 93]. It aims to fairly allocate the contributions of each feature towards achieving the overall result [94]. SHAP values can be applied in machine learning to measure the contribution of each feature to the model's prediction, providing insights into how each feature collectively influences the final outcome [95]. In evaluating feature importance, SHAP values compare the model's output with and without a specific feature for a given data point, accounting for all possible combinations of the other features. The average difference in the output is then computed. The SHAP value for feature X_j in a model is represented by equation (15):

$$Shapley(X_j) = \sum_{S \subseteq N \setminus \{j\}} \frac{K!(p-k-1)!}{p!} (f(S \cup \{j\}) - f(S))$$

Where:

p – The total number of features.

$N \setminus \{j\}$ – A set of all possible combinations of features excluding X_j .

S – A feature set in $N \setminus \{j\}$.

$f(S)$ – The model prediction with features in S .

$f(S \cup \{j\})$ – The model prediction with features in S plus feature X_j .

(15)

According to equation (15), the SHAP value of a feature represents its marginal contribution to the model's prediction, averaged across all possible models with varying combinations of features [95]. Equation (15) determines feature importance for a single data point. In our study, which involves multiple data points, this process is repeated for each data point. The average difference in output across all data points is then used as a metric to quantify the contribution and importance of a single feature in the model's predictions. This approach ensures that the impact of each feature is evaluated in a comprehensive manner, considering various interactions within the dataset. One practical example of how SHAP values are calculated for multiple data points in our dataset is

illustrated in Figure 27, where the contributions of individual features are systematically analyzed to provide a clear interpretation of their influence on the model's output.

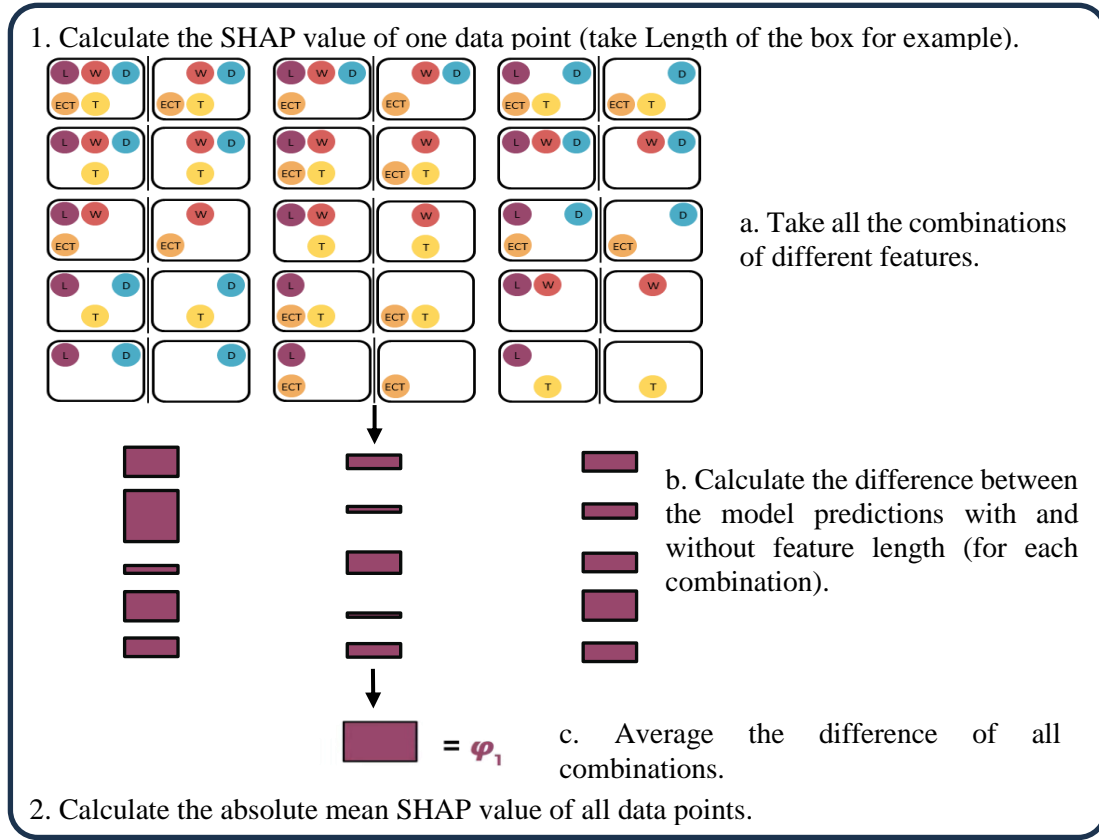


Figure 27 Calculation process of SHAP values with multiple data points in a data set

In this study, the `explainer.shap_values()` function from the SHAP library in Python is employed to compute the SHAP values for each BCS feature.

3.4 FLOW OF FEATURE IMPORTANCE EVALUATION USING ANN

To evaluate the input feature importance, the first step is to train an ANN model. In principle, training an ANN model involves building a function that recognizes the underlying relationships between input variables and output variables. By feeding the models with values of relevant input features and the corresponding output values of available data points, an ANN can be trained to learn the relationship between output(s) and their input features. Consequently, the trained ANN

model can forecast the output values for new data points based on their input feature values as they become available. In this section, the process began by constructing an ANN model through the training of available data sets, analogous to the procedure used for predicting output values. Then the four aforementioned methods (the weight connections method, the permutation method, the gradient-based method, and SHAP values) were applied to assess the relative importance of various packaging design features within the trained ANN model. The results of the packaging design feature importance assessment were validated by comparing them with the theoretical feature importance calculated using the well-established mathematical model. Figure 28 illustrates the sequential procedure for mapping input feature importance within an ANN model utilizing the four selected methods. The development and training of the ANN model were carried out using Jupyter Notebook, an integrated development environment (IDE).

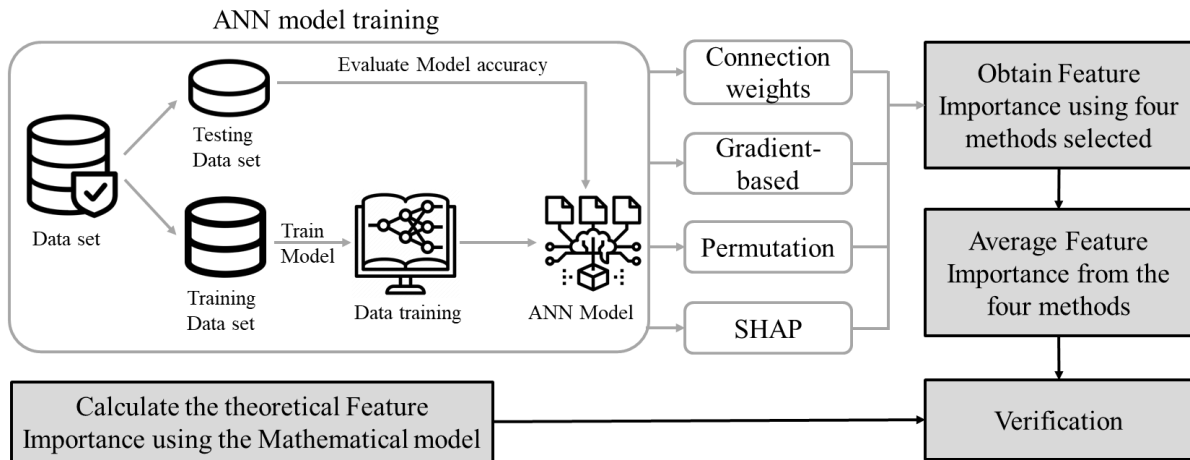


Figure 28 Flow of mapping feature importance using four ANN based methods

3.4.1 Methods for hidden layer neuron number optimization

When it comes to the methods for optimizing the hidden layer neuron numbers, the key is to balance the model accuracy and the computational efficiency. Recall Chapter 3, the exhaustive search method could identify the hidden layer neuron setting by locating the minimum error of ANN model prediction. However, the exhaustive search method is very time-consuming.

Therefore, researchers have investigated various computation techniques to achieve the output with minimum calculation while maintaining a high model accuracy.

The Akaike information criterion (AIC) was introduced by Akaike, Hirotogu in the 1998 [96], it originally was developed to identify an optimal model from a class of competing models [96] but has been adapted to the detection of outlier gene expression and model evaluation. By evaluating the model with different input features deleted, the input feature importance can be evaluated at the same. Hebb's rule, also known as Hebb's law or Hebbian learning, is a neuropsychological theory proposed by Canadian psychologist Donald Hebb in 1949 [97]. Hebb's Rule is based on the idea that the brain is capable of reorganizing itself in response to experience: when two neurons are activated simultaneously, the connection between them is strengthened. The Bayesian information criterion (BIC) or Schwarz information criterion (also SIC, SBC, SBIC) was developed by Gideon E. Schwarz and published in a 1978 paper [98]. The BIC is a criterion for model selection among a finite set of models in statistics, attempting to resolve this problem by introducing a penalty term for the number of parameters in the model [99]. The Optimal Brain Damage (OBD) rule was proposed by Yann Lecun et al. in 1989 by removing unimportant weights from a network to reduce the training examples required and improve the learning speed. OBD uses the second derivatives of the error function to determine which weights in the network are least important to the overall performance, making a trade-off between network complexity and training set error [100]. Bayesian Optimization is created by Jonas Mockus in the 1970s [101, 102]. Bayesian Optimization builds a probability model of the objective function and uses it to select a hyperparameter to evaluate the true objective function [103].

In this study, five computation methods, including Information Criteria using the AIC method, Hebb's rule, Information Criteria using the BIC method, Optimal Brain Damage rule, and Bayesian

Optimization method, were applied to optimize the hidden layer neuron number setting in order to reduce the computation time and achieve a high efficiency of ANN model training.

3.4.2.1 Bayesian optimization method

Bayesian Optimization is designed for black-box derivative-free global optimization [104].

Bayesian Optimization builds a probability model of the objective function and uses it to select hyperparameters to evaluate the true objective function. The true objective function is a fixed function, as shown in the dotted line in Figure 29. Generally, for a derivative-free function, what can be accessible are some data points (or observations), but not all, as shown as the black points in Figure 29. A surrogate model (surrogate function) can be built to approximate the true objective function. The surrogate function is represented as the black line in Figure 29.

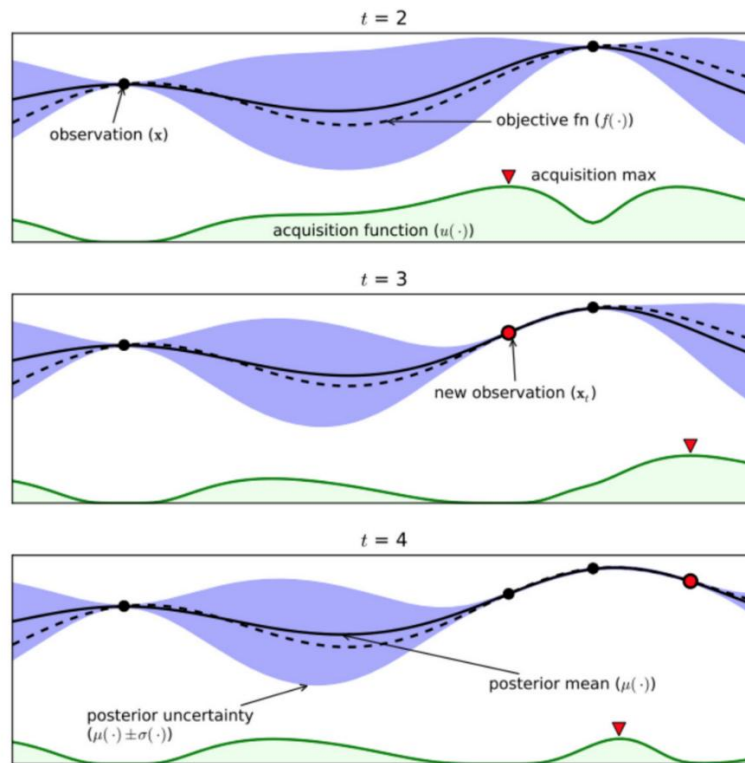


Figure 29 Schematic of Bayesian optimization process (source: <http://haikufactory.com/files/bayopt.pdf>)

The blue shade represents the deviation. A surrogate function by definition is “the probability representation of the objective function”, which is essentially a model trained on the

(hyperparameter, true objective function score) pairs. As some observations have been known, it is possible to find new observations by trying different parameters, and this is where an acquisition function needs to be built. An acquisition function can be generated using the surrogate function, which will be detailed in this chapter later. The way to identify the new observation is to locate the maximum point of the acquisition function and calculate the corresponding hyperparameter and its objective function. After the new observation is found, the surrogate function and acquisition function will be updated. This process is repeated till the surrogate function is as close as possible to the objective function. The schematic of the Bayesian Optimization process is shown above in Figure 29.

In this study, the Gaussian Process Model (GP) was used as the surrogate function. The Gaussian Process Model's acquisition function is the Expected Improvement function, as shown in equation (16),

$$EI(x) = \int_{-\infty}^{\infty} \max(f(x)^* - f(x), 0) pM(f(x)|x) df(x)$$

Where:

$pM(f(x)|x)$: the surrogate function. $f(x)$ is the true objective function score, x is the hyperparameter. (16)

$f(x)^*$: the minimum observed true objective function score so far.

$f(x)$: new scores.

Function BayesianOptimization() was used in Python to conduct the Bayesian Optimization for ANN hidden layer optimization.

3.4.2.2 Information Criteria using Akaike information criterion (AIC) method

The Akaike information criterion (AIC) is an estimator of prediction error and thereby relative quality of statistical models for a given set of data 126-128.. Given a collection of models for the data, AIC estimates the quality of each model, relative to each of the other models. Thus, AIC provides a means for model selection.

Let k be the number of estimated parameters in a statistical model of some data. Let \hat{L} be the maximized value of the likelihood function for the model. Then the AIC value of the model is the following, as shown in equation (17) [105] [106].

$$AIC = 2k - \ln \hat{L} \quad (17)$$

Where:

\hat{L} - The maximized value of the likelihood function for the model.

k - The number of estimated parameters in a statistical model.

This number estimates the amount of information that is lost when the model M is used to approximate reality. The model with the lowest AICM is considered the one best fitting the data [107].

In this study, the mean squared error (MSE) was calculated as the \hat{L} in the equation above. Given a set of candidate models for the data, the preferred model is the one with the minimum AIC value. Thus, AIC rewards goodness of fit (as assessed by the likelihood function), but it also includes a penalty that is an increasing function of the number of estimated parameters. The penalty discourages overfitting, which is desired because increasing the number of parameters in the model almost always improves the goodness of the fit [108].

3.4.2.3 Information Criteria using Bayesian information criterion (BIC) method

Bayesian information criterion (BIC) (Stone, 1979) is another criterion for model selection that measures the trade-off between model fit and complexity of the model. A lower AIC or BIC value indicates a better fit [109]. In statistics, the Bayesian information criterion (BIC) or Schwarz information criterion (also SIC, SBC, SBIC) is a criterion for model selection among a finite set of models; models with lower BIC are generally preferred. It is based, in part, on the likelihood function and it is closely related to the Akaike information criterion (AIC). When fitting models, it is possible to increase the maximum likelihood by adding parameters, but doing so may result

in overfitting. Both BIC and AIC attempt to resolve this problem by introducing a penalty term for the number of parameters in the model; the penalty term is larger in BIC than in AIC for sample sizes greater than 7 [110]. The BIC is formally defined as shown in equation (18) [111] [112],

$$BIC = k \ln(n) - 2 \ln \hat{L}$$

Where:

\hat{L} - The maximized value of the likelihood function of the model, i.e. $\hat{L} = p(x | \hat{\theta}, M)$ (18), where $\hat{\theta}$ are the parameter values that maximize the likelihood function and x is the observed data.

n - The number of data points in x , or the sample size.

k - The number of parameters estimated by the model.

In this study, the mean squared error (MSE) was calculated as the \hat{L} in the equation above.

3.4.2.4 Hebb's rule

According to Hebb's rule, in a network, the more often two neurons are activated together, the more efficient the connection between them becomes. When we learn new information or skills, the connections between neurons in our brain are modified to facilitate the formation of new neural pathways. Hebb's rule suggests that this process of neural reorganization is driven by the repeated co-activation of neurons.

From the point of view of artificial neurons and artificial neural networks, Hebb's principle can be described as a method of determining how to alter the weights between model neurons. The weight between two neurons increases if the two neurons activate simultaneously and reduces if they activate separately. Nodes that tend to be either both positive or both negative at the same time have strong positive weights, while those that tend to be opposite have strong negative weights.

The implementation of Hebb's rule is:

- a) Train the neural network using all of the neurons in the hidden layer.
- b) Use the weights learned during training to calculate the Hessian matrix.
- c) Use the Hessian matrix to calculate the sensitivity of the cost function with respect to

each neuron.

d) Prune the neurons with the smallest sensitivity values.

This process can be repeated until the desired level of network complexity is reached.

In this study, Hebb's rule used two empirical functions in the Python library.

3.4.2.5 Optimal Brain Damage (OBD) rule

The Optimal Brain Damage (OBD) rule is to use the second derivatives of the error function to determine which weights in the network are least important to the overall performance, making a trade-off between network complexity and training set error [100]. The OBD procedure can be carried out as follows:

- a) Choose a reasonable network architecture.
- b) Train the network until a reasonable solution is obtained.
- c) Compute the second derivatives for each parameter.
- d) Compute the saliencies for each parameter.
- e) Sort the parameters by saliency and delete some low-saliency parameters.
- f) Iterate to step b).

In this study, OBD is applied based on an empirical function in the Python library.

3.4.2 ANN neuron number determination in the hidden layer

To build an ANN model, the values of three key modeling factors need to be determined. These three modeling factors include the hidden neuron number setting, the epoch number, and the modeling cycle number. Dataset for training an ANN was randomly divided into two subsets: a training data set for training the model and a testing data set for assessing model accuracy. To construct the ANN model, the number of hidden layer(s), and neuron numbers for each layer were determined by balancing the model accuracy and computational efficiency. In this study, five

optimization methods were examined, each targeting the optimization of the hidden layer configuration. These methods included the Bayesian optimization method [113] [103], Information Criteria using the Akaike information criterion (AIC) method [96], Information Criteria using the Bayesian information criterion (BIC) method [98], Hebb's rule [114], and the Optimal Brain Damage (OBD) rule [100]. The final hidden layer configuration was determined based on minimizing error and maximizing computational efficiency through a comprehensive comparison of model errors across the five optimization methods mentioned above. Figure 30 presents a conceptual diagram of the ANN structure developed in this study.

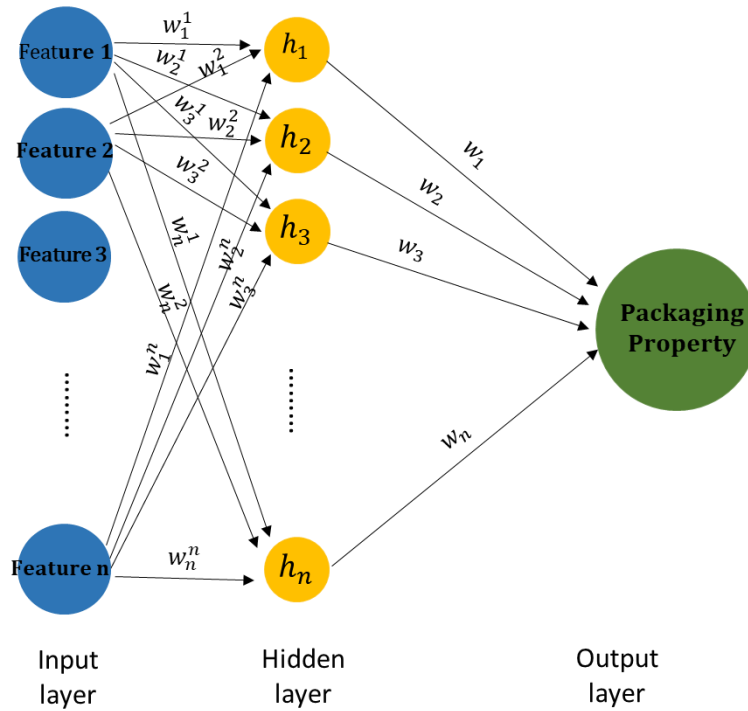


Figure 30 A conceptual diagram of the ANN structure used in this study

The number of epochs was determined based on the model loss (error) versus epoch plot. The epoch count was chosen when the model error decreased and reached a stable, plateaued level.

3.4.3 Final relative feature importance determination

With the optimal hidden neuron setting and the optimal epoch number, the studied packaging design features were incorporated into the ANN model. The relative importance of these packaging

design features was evaluated using the previously mentioned four methods. Each method provided importance scores for the studied packaging features, which were then normalized to a scale from 0 to 1 to account for differences in scaling. Referring to the mathematical model, The theoretical feature importance of 0 for features not included in the mathematical model was used to identify and eliminate unreliable methods. In practice, researchers and practitioners often average feature importance scores from various methods to obtain a more stable or general feature ranking. Therefore, in this study, to produce a comprehensive measurement, the feature importance results from the reliable methods were averaged to determine the final importance of different packaging design features. To confirm the validity of these results, the averaged packaging design feature importance was compared with the theoretical feature importance calculated using the well-established mathematical model.

3.5 CASE STUDY FOR FEATURE IMPORTANCE ANALYSIS

As a critical parameter in the evaluation of shipping containers, BCS is determined by various factors, such as material properties, flute types, dimensions, and more. Each factor, or BCS feature, affects the BCS differently. Understanding how each BCS feature influences the BCS value and identifying the most impactful ones are crucial for packaging design. This knowledge enables designers to strategically prioritize adjustments to the most influential features, ultimately reducing material consumption and costs [115]. However, few systematic methods to evaluate the BCS features have been developed yet. Current analytical methods for BCS prediction indicate the dominant BCS features but require plenty of various mechanical tests considering the various BCS features. Existing numerical models based on finite element analysis (FEA) face difficulties in obtaining relevant parameters and dealing with the anisotropic non-linear properties of paper materials. Assessing the BCS feature importance is a great challenge for the corrugated packaging

industry. Therefore, in this section, BCS was used as a representative packaging property to validate the capability of the ANN approach in assessing the relative importance of packaging design features. Two datasets — one synthetic and one real—were employed as case studies. Up to six BCS features —including box perimeter, depth, ECT, thickness, and bending stiffness in both machine (EI_x) and cross directions (EI_y) — were evaluated using the four selected ANN-based approaches. The average feature importance of these BCS features, as determined by the ANN approach, was calculated to provide a comprehensive result. These values were then compared with the theoretical feature importance values derived from the McKee formula to verify the ANN assessment.

3.5.1 Case study 1-Relative feature importance of the synthetic data set

The first data set used was a synthetic dataset which was created by inputting the box perimeters, depths, ECTs, and thicknesss of 3,009 commonly used commercial boxes [116] into the simplified McKee formula, as detailed in equation (19) [117] to compute the BCS values.

$$BCS = 5.87 \times ECT \times \sqrt{Caliper \times P}$$

Where:

ECT – Edge Crush Strength (lb/in).

Thickness – Thickness of the corrugated board (in).

P – Perimeter of the box (in).

(19)

Using the simplified McKee formula along with the concept of derivatives, we computed the theoretical relative importance of the four BCS features. The analysis, detailed in Figure 31, shows that the ECT feature has the highest relative importance, with a weight of 0.500, indicating that it is the most influential factor in the model. Both the perimeter and thickness features were found to have equal significance, each contributing a weight of 0.250, which underscores their moderate but noteworthy impact on the model's performance. In contrast, the depth feature was determined to have no influence in this analysis, receiving a weight of 0. This outcome, as illustrated in Figure

31, not only quantifies the contributions of each feature but also emphasizes the critical role of ECT in the model, while suggesting that the depth feature may be redundant or less relevant for this particular application.

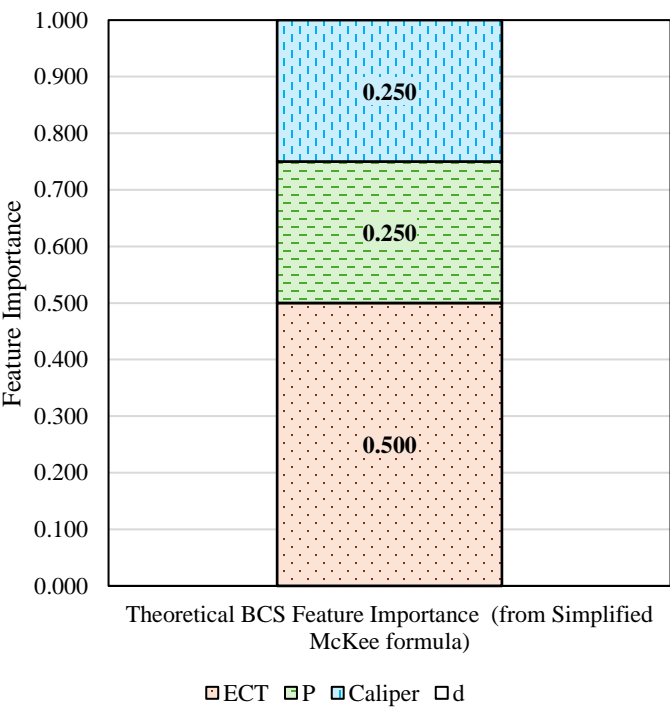


Figure 31 Theoretical BCS Feature Importance calculated using the Simplified McKee formula

3.5.1.1 ANN training using the synthetic data set

During the ANN model training process, the synthetic dataset with 3009 data points was randomly divided into two subsets: 70% of the data (2106 data points) for training the model and the remaining 30% (993 data points) for testing the model’s accuracy.

As previously mentioned, the hidden layer neuron number setting was determined by comparing the model errors obtained by the five optimization methods mentioned in section 2.1(including the Bayesian optimization method, AIC method, BIC method, Hebb's rule, and the OBD rule) to minimize the model error and maximize the computational efficiency. Figure 32 presents the optimal neuron settings for the hidden layer as determined by each method, along with their

corresponding model errors over 70 modeling cycles. The results showed that the AIC method achieved the lowest model errors with one hidden layer and 120 neurons. The BIC method produced the second-lowest model error, with one hidden layer and 34 neurons. The error difference between these two methods was no more than 0.0032, considering their 95% confidence intervals (0.0032 ± 0.0015 and 0.0026 ± 0.0011). This indicates that the more complex configuration suggested by the AIC method was not necessary. In contrast, the simple configuration of 34 provides nearly the same accuracy but is significantly more efficient in terms of computational resources. Consequently, the neuron configuration proposed by the BIC rule was applied throughout the study for this data set. Namely, the ANN model developed in this real data set includes a single hidden layer with 34 neurons.

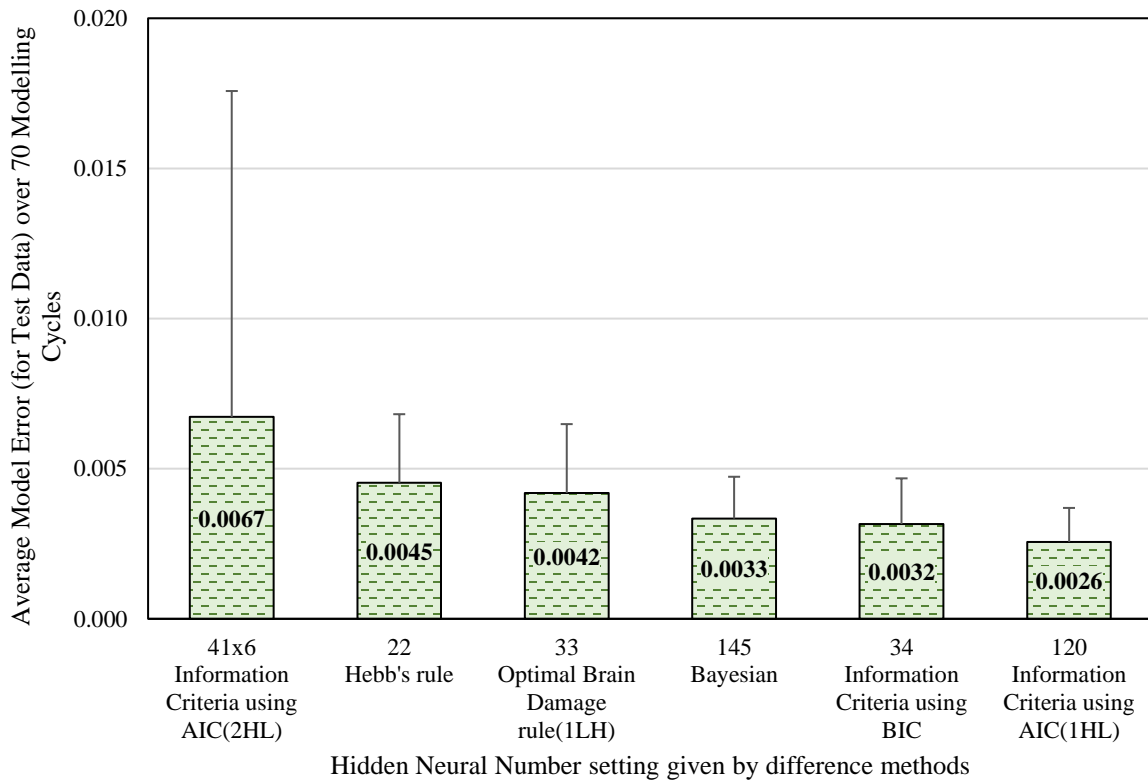


Figure 32 Optimal neuron numbers in the hidden layer(s) determined by different methods for the synthetic data set

The number of epochs was determined based on the model loss (error)-epoch plot (Figure 33),

which showed that model error reduction plateaued after 25 epochs. To maintain a conservative approach, the number of epochs was set to 35.

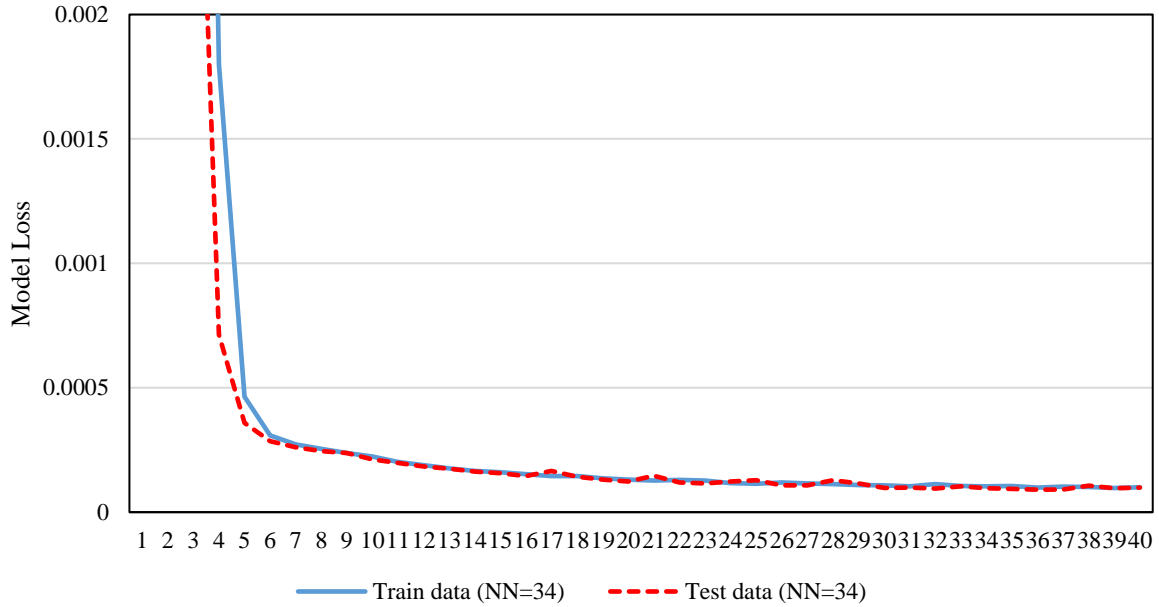


Figure 33 Model Loss (Error) versus epoch plot with 34 neurons in the hidden layer

The feature importance of 10 modeling cycles was averaged to obtain a reliable average feature importance.

3.5.1.2 Relative feature importance analysis of the synthetic data set

The importance of the BCS features in this dataset was assessed using the four methods previously discussed, each of which provided an independent measure of the relative significance of the four key BCS features: edge crush test (ECT), thickness of the corrugated board, box perimeter, and box depth. Since different methods may yield results on different scales, a normalization process was applied to ensure consistency and comparability across all approaches. Specifically, the feature importance values were scaled such that the sum of the importance scores for all four features equaled 1 within each method. This normalization allows for a direct comparison of how each method ranks the importance of individual features while eliminating potential discrepancies caused by variations in scale or magnitude. The results of this analysis, presented in Figure 34,

provide a comprehensive overview of how different methods evaluate feature importance and highlight any similarities or discrepancies in their assessments.

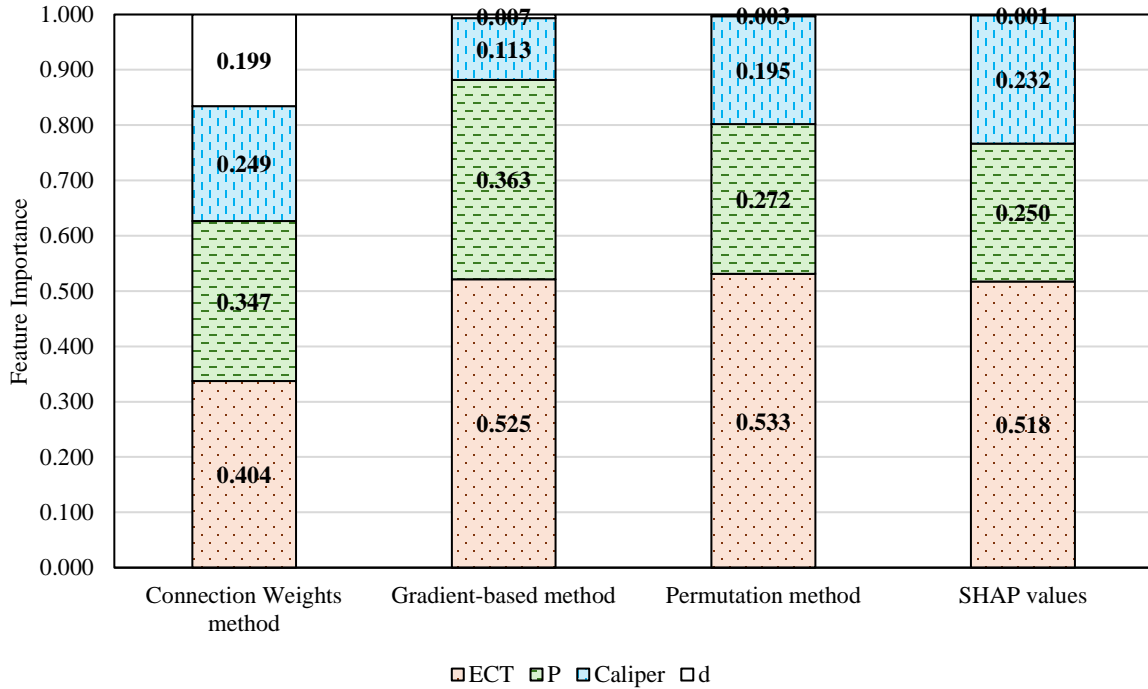


Figure 34 ANN evaluated BCS feature importance of the synthetic data set generated using the Simplified McKee formula

The ranking of BCS feature importance consistently identified by the four methods is ECT > Perimeter > Thickness > Depth. However, the connection weights method shows unusually high importance for the depth feature, which deviates significantly from the expected value of zero given the synthetic dataset's design. This discrepancy renders the results from the connection weights method unreliable. Therefore, the feature importance results of the other three approaches were averaged to provide a comprehensive estimate of BCS feature importance. The average BCS feature importance from these three methods shows that ECT weights 0.525, perimeter 0.295, thickness 0.180, and depth 0.004, as illustrated in Figure 35 (left). When compared to the theoretical BCS feature importance ranking calculated using the simplified McKee formula (depicted in Figure 35 (right)), the results from the three reliable methods are closely aligned. This

indicates that the ANN approach can be a potential tool for evaluating the relative importance of packaging design features.

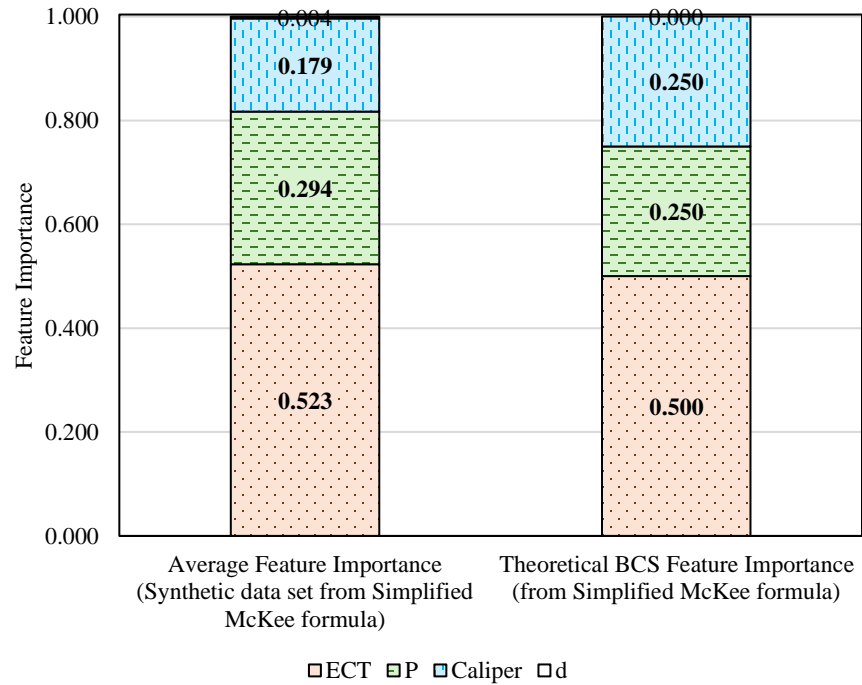


Figure 35 Comparison of the average feature importance assessed by the selected ANN-based methods and theoretical BCS feature importance calculated using Simplified McKee formula

3.5.2 Case study 2-Relative feature importance of the real data set

The second data set used was a real data set that comprises industry data on 429 commonly used commercial boxes, detailing five BCS features: box perimeters, depths, ECTs, and flexural stiffness in both the machine and cross-machine directions. These BCS values were obtained through actual testing, accurately reflecting real-world industry conditions. Since real-world data captures fluctuations in BCS feature values due to measurement inaccuracies and variations in material parameters, studying a real dataset from industry is very meaningful to the feature importance assessment.

As with the first synthetic dataset, the theoretical relative importance of these five BCS features was calculated using the improved McKee formula (equation (20)) [118], which is ECT weights

0.500, perimeter has 0.330, both stiffness EI_x and EI_y have 0.085, and depth has no importance (0), as shown in Figure 36.

$$BCS = 2.028ECT^{0.746} \sqrt{(EI_x \times EI_y)^{0.254}} P^{0.492}$$

Where:

(20)

ECT - Edge Crush Strength (lb/in).

EI_x, EI_y - Flexural stiffness in the machine direction & cross-machine direction of the corrugated board (lb*in).

P - Perimeter of the box (in).

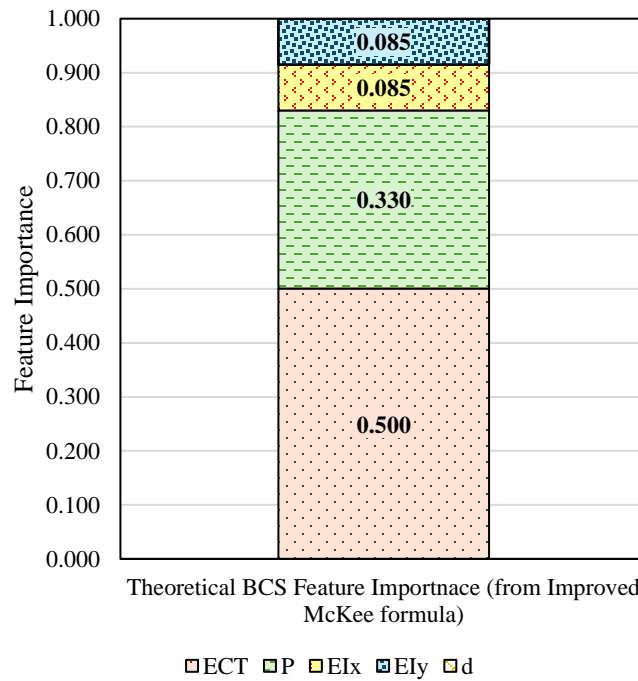


Figure 36 Theoretical BCS Feature Importance calculated using the Improved McKee formula

3.5.2.1 ANN training using the synthetic data set

During the ANN model training process, the real dataset with 429 data points was randomly divided into two subsets: 70% of the data (287 data points) for training the model and the remaining 30% (142 data points) for testing the model's accuracy.

As with the synthetic data set, the hidden layer neuron number setting was also determined by comparing the model errors given by the five optimization methods mentioned above (including

the Bayesian optimization method, AIC method, BIC method, Hebb's rule, and the OBD rule) to minimize the model error and maximize the computational efficiency. Figure 37 presents the optimal neuron settings for the hidden layer as determined by each method, along with their corresponding model errors over 70 modeling cycles.

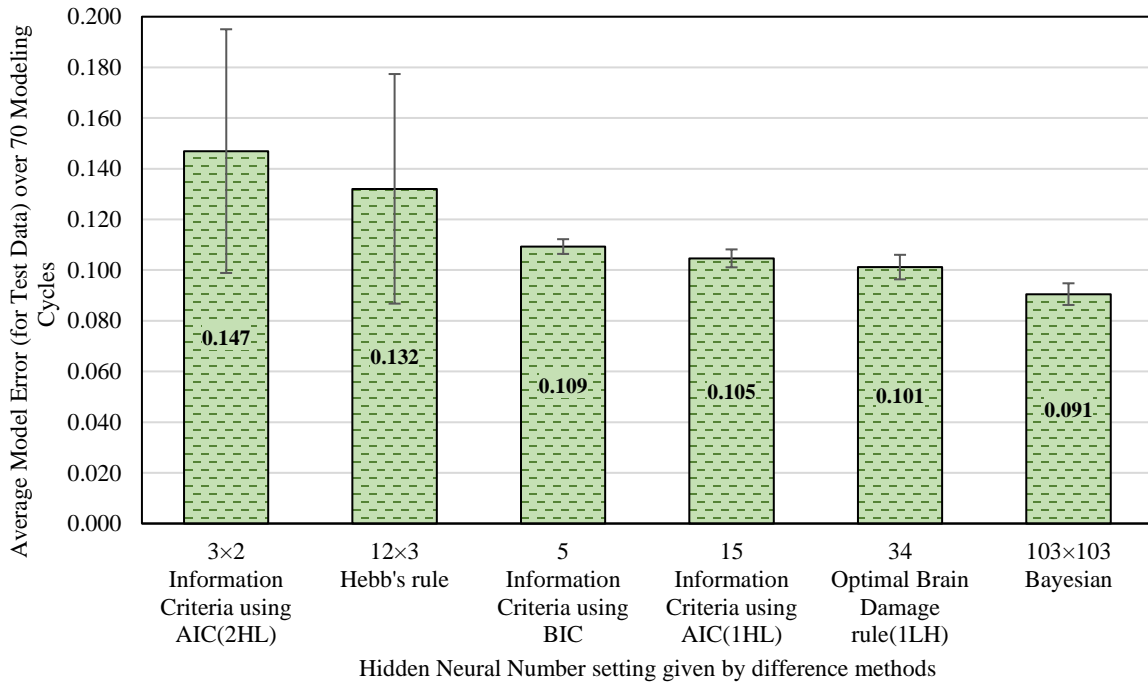


Figure 37 Optimal neuron numbers in the hidden layer(s) determined by different methods for the real data set

The results showed that Bayesian optimization achieved the lowest model errors with two hidden layers, each containing 103 neurons. The OBD rule produced the second-lowest model, with one hidden layer and 34 neurons. The error difference between these two methods was no more than 0.0191, considering their 95% confidence intervals (0.101 ± 0.0048 and 0.091 ± 0.0043). This indicates that the more complex configuration suggested by Bayesian optimization was not necessary. In contrast, the simple configuration of 34 provides nearly the same accuracy but is significantly more efficient in terms of computational resources. Consequently, the neuron configuration proposed by the OBD rule was applied throughout the study for this data set.

Namely, the ANN model developed in this real data set includes a single hidden layer with 34 neurons.

The number of epochs was determined based on the model loss (error)-epoch plot (Figure 38), which showed that model loss (error) reduction plateaued after 40 epochs. To ensure a conservative result, the number of epochs was set to 50.

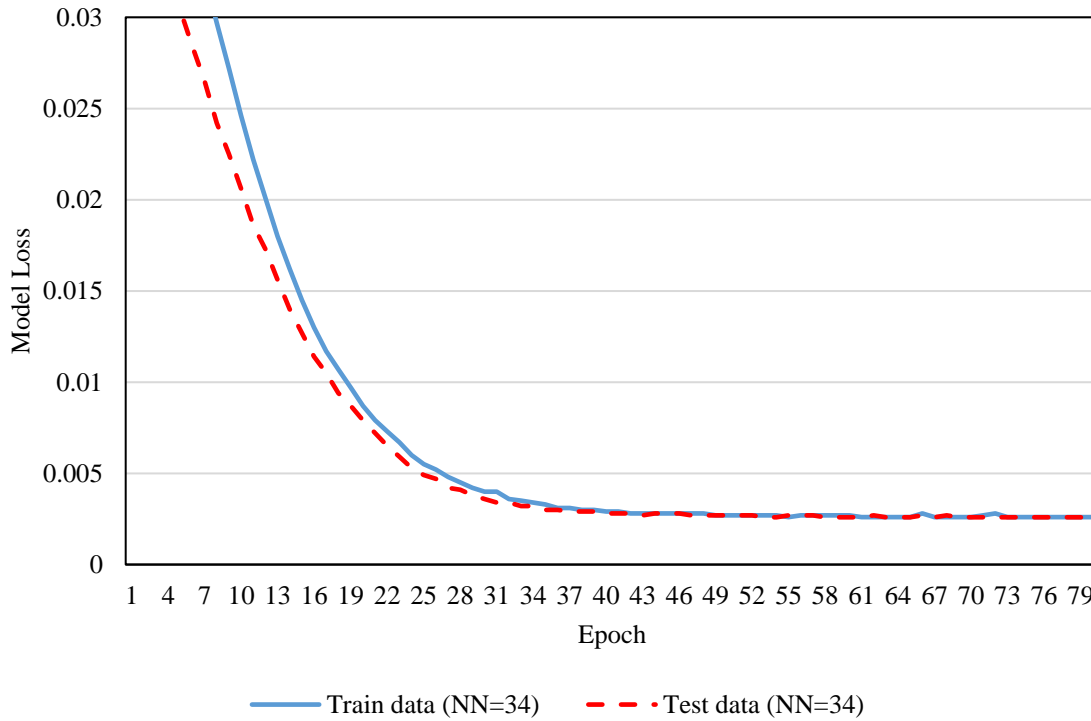


Figure 38 Model loss (error) versus epoch plot with 34 neurons in the hidden layer

The feature importance of 10 modeling cycles was averaged to achieve a reliable average feature importance.

3.5.2.2 Relative feature importance analysis of the real data set

Given the unreliability of the connection weights method for assessing BCS feature importance, particularly for the depth feature, it was excluded from the analysis of the real dataset. Instead, the BCS feature importance for the real data was evaluated using the remaining three methods. Following the same procedure applied to the synthetic data set, the BCS feature importances were normalized across these methods to ensure that the sum of the five BCS feature importances

equaled 1 for each method, as illustrated in Figure 39.

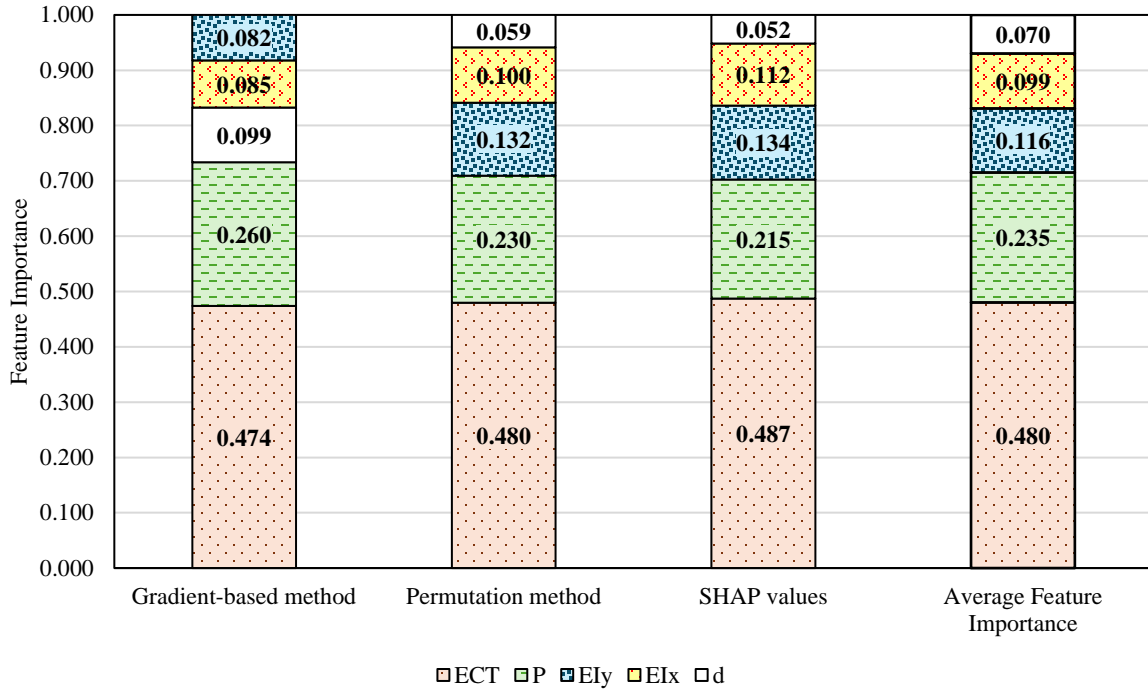


Figure 39 ANN evaluated BCS feature importance of the real data set

The results from the three methods demonstrate overall consistency and have been averaged to establish a comprehensive ranking of the five BCS feature importances. The average BCS feature importance is ranked as $ECT > Perimeter > EI_y > EI_x > Depth$, which aligns well with the theoretical BCS feature importance calculated using the mathematical model, as shown in Figure 40. The average BCS feature importance from these three methods shows that ECT weights 0.480, perimeter 0.235, EI_y 0.116, EI_x 0.099, and depth 0.070. The analysis of the real dataset reveals that, although Depth is ranked last, it still has a notable influence. In general, Depth remains an important factor in determining BCS (Buckling Compression Strength). As the depth value increases, buckling theory suggests that depth can significantly affect compression strength, making it a critical consideration. Despite the McKee equation [74] theoretically assigning a zero-importance value to depth, the real-world data demonstrates that the depth's effect should not be disregarded. Our results emphasize the influence of depth, validating that the proposed methods

for measuring feature importance accurately reflect practical applications. Furthermore, although the theoretical BCS feature importance ranks EI_x EI_y equally, the average importance ranking from the three methods shows a 0.017 difference between these two features. This small variation is understandable considering the fluctuations of measurement inaccuracies that mathematical models may not fully account for.

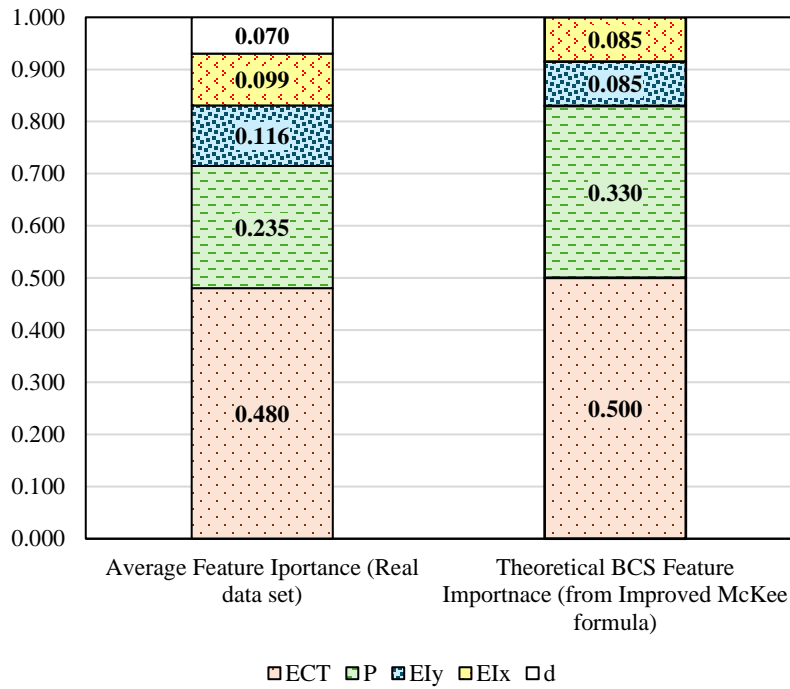


Figure 40 Comparison of the average ANN evaluated BCS feature importance of the real data set and theoretical BCS feature importance calculated using the Improved McKee formula

In summary, the BCS feature importance ranking for the real dataset is consistent with the findings from the theoretical feature importance. The evaluation of the real dataset provides a real-world context to the findings and further demonstrates the capability of the ANN approach in terms of feature importance evaluation for packaging design.

3.6 CONCLUSION

This study introduces a new method for evaluating the importance of packaging design features using four ANN-based approaches: the Connection Weights method, the Gradient-based method,

the Permutation method, and SHAP values. Using BCS as a representative packaging design property, the relative importance of up to six BCS features was assessed through these ANN-based approaches. One synthetic dataset derived from the well-established mathematical model (McKee formula) and one real dataset were used as two case studies for training the ANN model and obtaining the feature importance influencing BCS. The feature importance rankings provided by the ANN approaches were consistent with the theoretical feature importance calculated using the mathematical model across both datasets. This result highlights the effectiveness of the ANN approach in evaluating feature importance in packaging design, allowing for a more efficient assessment of the relative impact of various design features. This allows designers to prioritize adjustments to the most influential features, ultimately reducing material consumption and costs. For instance, to increase BCS, designers can first consider increasing the box dimensions for a minimal design effort and reduced material waste, rather than modifying the thickness or flexural stiffness, which would require changes to the materials or production process. Overall, this study offers a novel approach to assessing packaging design feature importance through ANN techniques, providing practical insights for improving material efficiency and cost-effectiveness. This method can be easily applied to evaluate the relative importance of other packaging properties beyond BCS, offering valuable insights for addressing various challenges in the packaging industry using ANN approaches.

CHAPTER 4: BUILDING A GENERALIZED ANN MODEL TO EVALUATE BCS

4.1 INTRODUCTION

The goal of this chapter is to build a generalized artificial neural network (ANN) model for box BCS evaluation. Based on the available data, a dataset extracted from a real data set containing the majority of BCS values used in the industry was utilized to train the model.

The ANN modeling factors include the number of epochs, modeling cycles, and the hidden layer neuron setting. The number of epochs and modeling cycles were set up based on conservative results from the dataset with variation. Specifically, the number of epochs was set to 140, and the number of modeling cycles was initially set to 70 to get a conservative result. The hidden layer neuron setting was optimized using the same five optimization methods as Chapter 4 while balancing the model accuracy and computational efficiency. Namely, the five optimization methods used in this chapter are the Information Criteria using the Akaike information criterion (AIC) method, Hebb's rule, the Information Criteria using the Bayesian information criterion (BIC) method, the Optimal Brain Damage rule, and the Bayesian Optimization method.

To evaluate the performance of the ANN model, the model prediction error of the test data was calculated and compared. After comparing the model error given by each optimization method, the optimal hidden neuron configuration, determined by the Optimal Brain Damage rule, consists of a single hidden layer with 35 neurons. This configuration was selected for its ability to best balance model accuracy and computational efficiency. This configuration resulted in a model error of 9.51% when evaluated on the test dataset, indicating that the model achieves a strong balance between accuracy and generalizability for practical industry applications. The observed error can primarily be attributed to the presence of boundary data points, which introduce variability and potential inconsistencies in the predictions, as well as the limited size of the dataset, which restricts

the model's ability to learn from a broader range of patterns. Despite these challenges, the model demonstrates reliable performance, making it suitable for real-world implementation. Further refinements, such as expanding the dataset or employing advanced regularization techniques, could potentially enhance accuracy and reduce error margins. The overall structure and logical progression of this chapter are visually outlined Figure 41, providing a clear roadmap of the analysis and methodology employed.

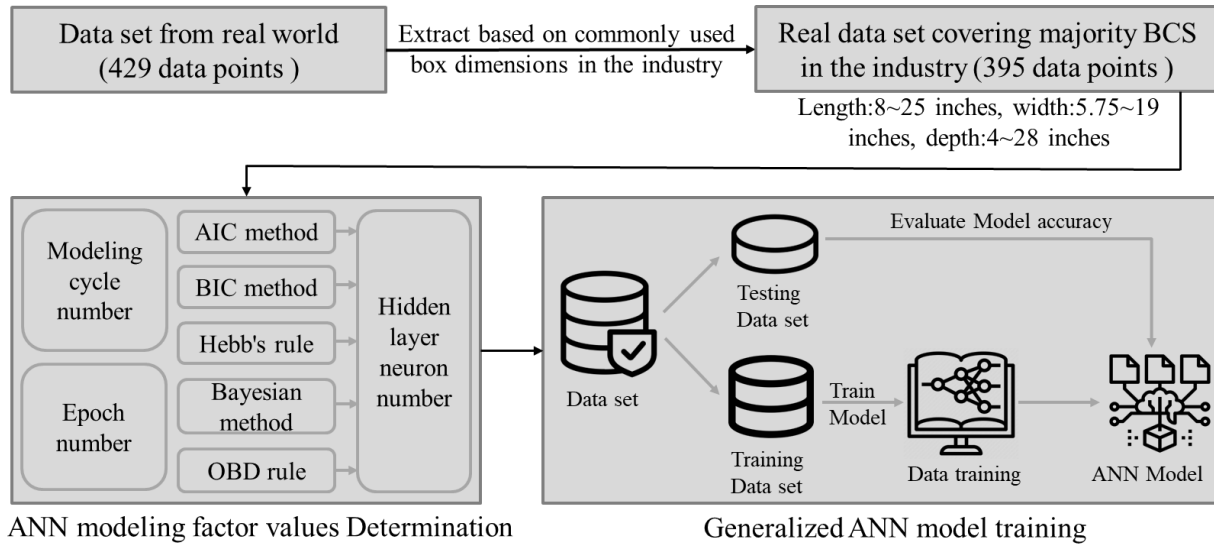


Figure 41 Flow of building a generalized ANN model for BCS prediction

4.2 EXTRACT REAL DATA SET TO COVER THE MAJORITY OF BCS IN THE INDUSTRY

To build a generalized ANN model for BCS evaluation, we trained a data set extracted from real-world data containing the most commonly used box dimensions. Based on an investigation of box dimensions used in the industry, provided by Packaging Corporation of America (PCA) company, we determined that 90% of commonly used boxes in the industry have the following dimensions: length between 8 and 25 inches, width between 5.75 and 19 inches, and depth between 4 and 28 inches. Therefore, our extracted dataset includes box dimensions within these ranges. The dataset comprises 395 data points in total, with BCS values ranging from 347 to 2172 lbs.

4.3 DETERMINATION OF HIDDEN LAYER NEURON SETTING

Five methods mentioned above (including the Information Criteria using the AIC method, Hebb's rule, the Information Criteria using the BIC method, the Optimal Brain Damage rule, and the Bayesian Optimization method) for optimizing the ANN hidden neuron number setting were applied.

The model error of test data obtained by each method has been calculated and compared, as shown in Figure 42.

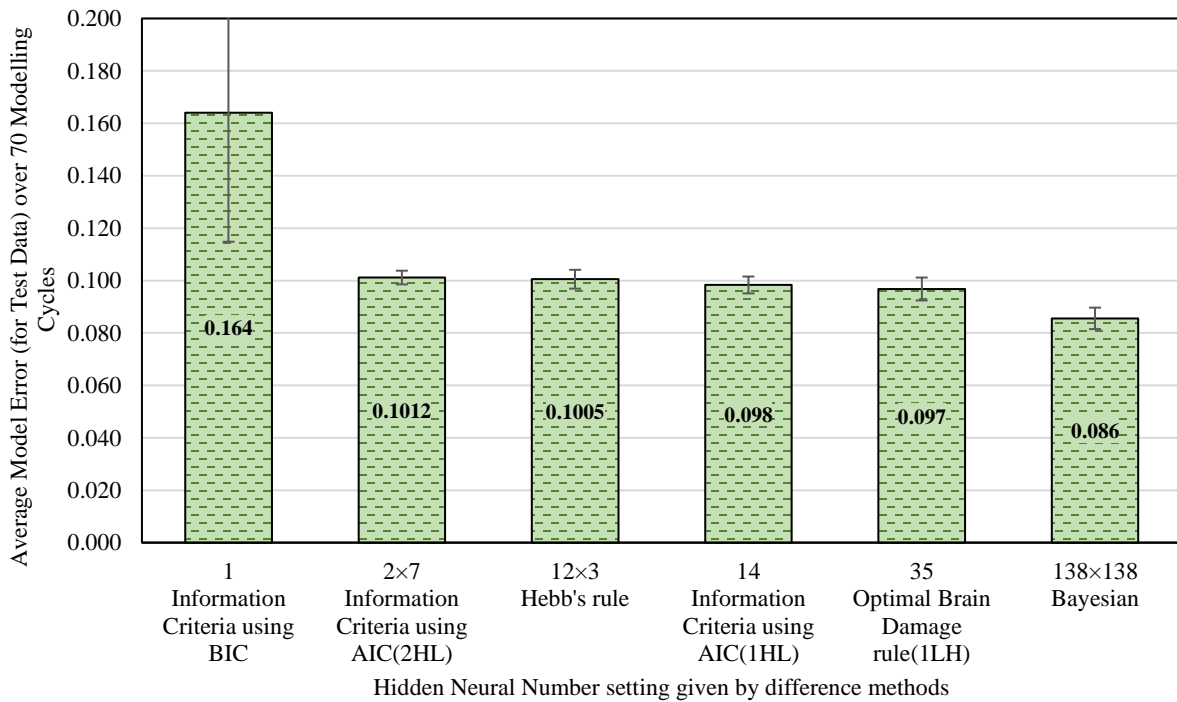


Figure 42 ANN model error with their optimized hidden neuron numbers using the five selected methods

For the Information Criteria using the AIC method, both one hidden layer and two hidden layers were tested. The result shows that the optimal hidden neuron setting from the five methods are 1 with one single hidden layer for the BIC method, 2 and 7 in the first and second hidden layer for the AIC method (2 hidden layers), 12 and 3 in the first and second hidden layer for the Hebb's rule, 14 with one single hidden layer for the AIC method (1 hidden layer), 35 with one hidden layer for

the OBD rule, and 138 in both the first and second hidden layer for the Bayesian Optimization method. Overall, the model error decreases as the hidden neuron number increases. The model errors of the test data using the Bayesian method and the OBD rule are the lowest and second lowest. However, the error reduction rate is not significantly improved when the number of hidden neurons increases to 138 across two hidden layers (as determined by the Bayesian method), compared to 35 neurons in a single hidden layer (as determined by the OBD method). Therefore, 35 neurons with one single hidden layer obtained by the OBD rule were chosen as the optimal hidden neuron number setting, reducing the training computation time while maintaining a good performance for the ANN prediction.

4.4 TRAINING ANN MODEL TO EVALUATE BCS IN THE INDUSTRY

With the optimal neuron number of 35 in the hidden layer, 140 epochs, 70 modeling cycles, and 395 data points from the real world, the ANN model was trained and the model error from 10 to 70 modeling cycles with have been calculated with the 95% confidence interval, as shown in Figure 43.

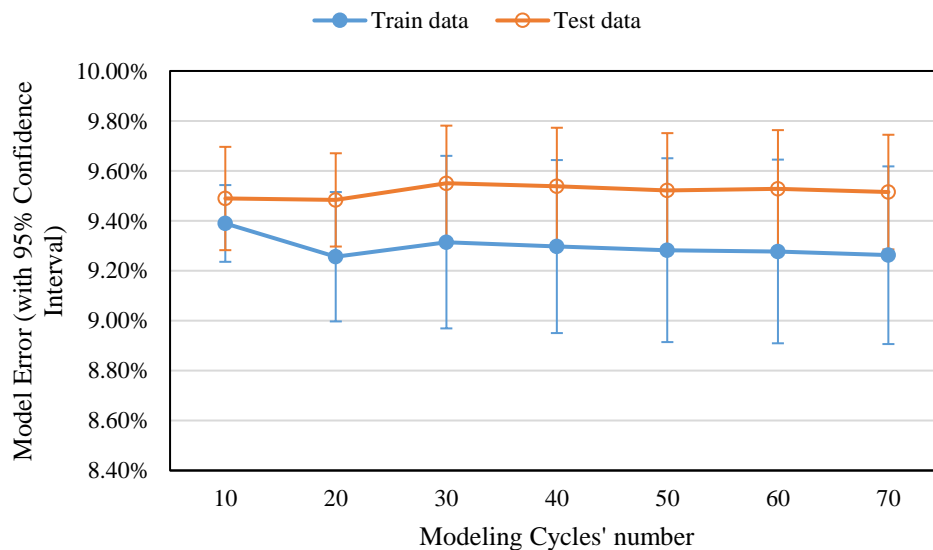


Figure 43 ANN Model error of the samples covering the 90% BCS values with commonly used dimensions of boxes in the industry

Overall, the model error of both train and test data converged at 30 modeling cycles. The ANN prediction error for BCS is below 9.60%. The average BCS error of the train and test data across 70 modeling cycles is 9.26% and 9.51%, respectively, as shown in Figure 44.

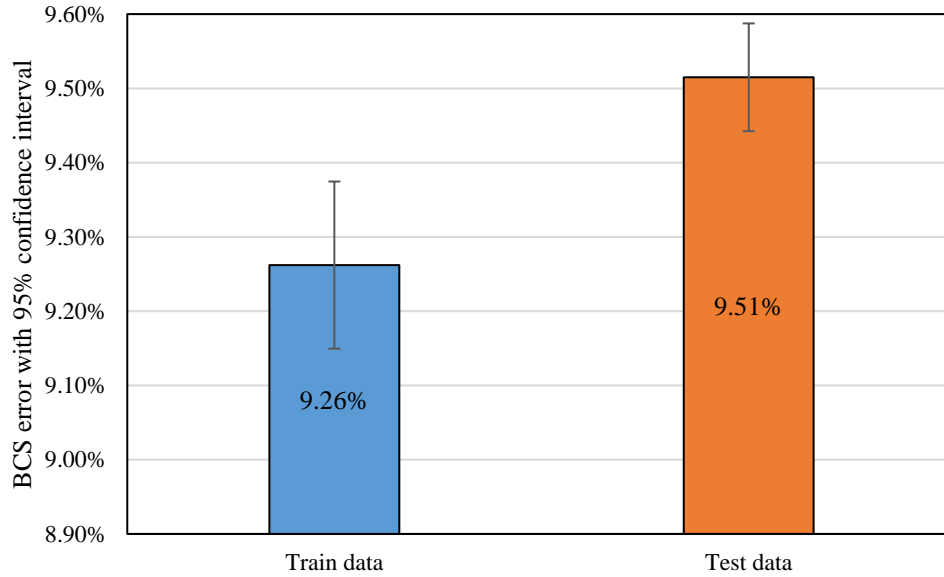


Figure 44 Average BCS error in the train and test data cross the 70 modeling cycles

To investigate the reason that causes the BCS error for the ANN model prediction, the BCS distribution of two random distinctive modeling cycles was also studied and plotted, as shown in Figure 45. The actual BCS distribution is represented by the blue columns, while the predicted BCS distributions for two randomly selected modeling cycles are shown in orange and green. When comparing the actual and predicted BCS distributions, the predicted distributions for these cycles indicate that data points with BCS values between 347 lbs and 450 lbs, as well as values greater than 1997 lbs, cannot be accurately predicted and consistently exhibit higher errors than other data points. This suggests that these boundary data points contribute significantly to the high prediction error of the ANN model for BCS evaluation in the current dataset. Another reason could be that the dataset size is not large enough, as the minimum number of data points for an idealized dataset, as mentioned in Chapter 3, was around 1500. In comparison, the extracted data set has a

smaller sample size than the minimum sample size required for achieving reliable ANN prediction accuracy.

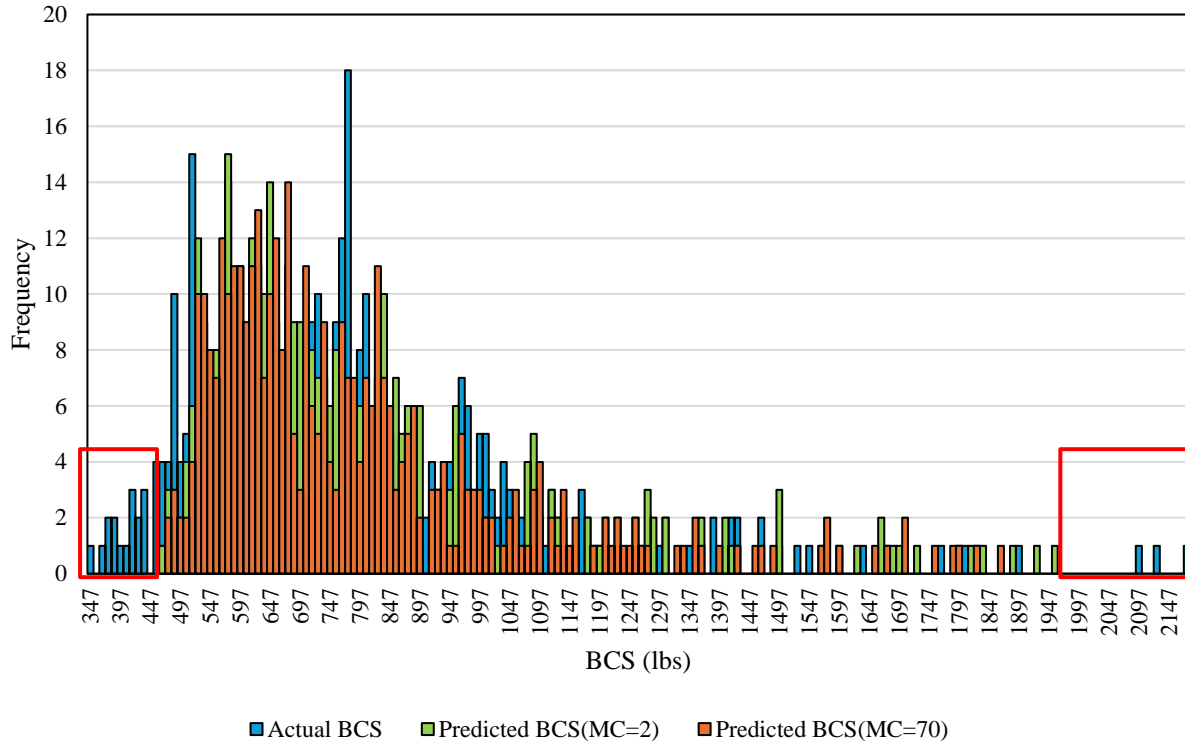


Figure 45 BCS distribution of ANN model prediction for the extracted real data set

The structure of the generalized ANN model built for BCS prediction is shown in Figure 46.

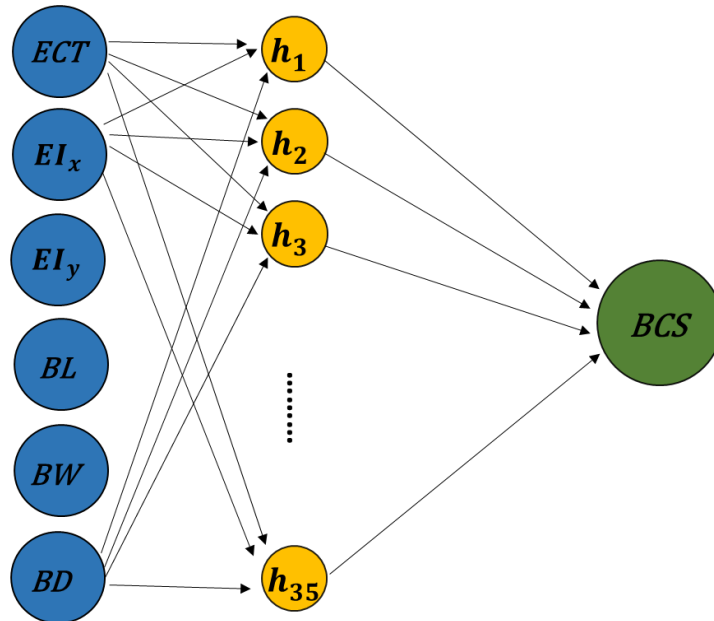


Figure 46 The structure of the generalized ANN model built by the real data set

The structure of the generalized ANN model contains six BCS features as the inputs, 1 hidden layer with 35 neurons. The optimal epoch number is 50 and the optimal modeling cycle number is 30.

4.5 CONCLUSION

In this chapter, a generalized ANN model for BCS evaluation targeting industrial applications has been built. A generalized dataset derived from real-world data was used to cover the 90% BCS values with commonly used dimensions of boxes in the industry. Drawing from the training results of the previous data sets (data set with variations), ANN modeling factor values for epoch number and modeling cycles' number were set to 140 and 70, respectively, to ensure a conservative outcome. Five methods (the Information Criteria using the AIC method, Hebb's rule, the Information Criteria using the BIC method, the Optimal Brain Damage rule, and the Bayesian Optimization method) for optimizing the hidden layer neuron setting were investigated and their training results with the corresponding model errors were compared. The optimal neuron number in the hidden layer was determined to be 35 to strike a balance between minimizing the model error of the test data and saving computational training time. Throughout 70 modeling cycles, the average BCS error for the test data, accounting for the corresponding neuron count in the hidden layer, was computed at 9.51%. The BCS value distribution revealed that the data points whose BCS values fell between 347 lbs and 450 lbs, as well as those exceeding 1997 lbs, exhibited higher errors in the ANN prediction. This observation suggests that the primary factor contributing to the high BCS error is the presence of boundary data points. These data points, situated at the edges of the dataset range, pose challenges for the ANN model in accurately predicting their corresponding BCS values. The small sample size of the extracted real dataset is another limiting factor that hinders achieving higher ANN prediction accuracy. In conclusion, the current ANN model can

predict the BCS of commonly used box dimensions in the industrial applicable level with an error of 9.51%. One possible strategy to improve ANN prediction accuracy is to continually expand the current dataset sample size using available resources. In summary, this study provides valuable insights utilizing the ANN approach to evaluate BCS of corrugated packages and solve the problems in the corrugated industry.

CHAPTER 5: RESEARCH SUMMARY AND FUTURE RESEARCH

5.1 RESEARCH SUMMARY

This dissertation research explores the feasibility of using ANN to evaluate the BCS of corrugated packaging. The results demonstrate that employing ANN for BCS prediction is both feasible and meaningful, offering substantial advantages over traditional evaluation methods. ANNs can effectively address several challenges inherent in current BCS evaluation methods, including enhancing efficiency, reducing costs, and ensuring the validity of model construction, among others. The intelligent and robust analytical capabilities of ANN, grounded in data and mathematical methodologies, hold significant potential for enhancing efficiency, cost-effectiveness, and reliability in BCS evaluation. This study contributes to the exploration of ANN's potential in predicting BCS and its application in addressing complex challenges within the corrugated industry.

To optimize the key modeling parameters of ANN for BCS evaluation with a reliable result (Chapter 3), both data sets from literature with small data population and synthetic data sets with large data populations have been trained to interpret the performance of ANN for BCS estimation. Four key modeling parameters (the combination of neuron numbers in hidden layers, the number of epochs, the number of modeling cycles, and the size of the data set) can significantly influence the ANN prediction accuracy and can be optimized based on the ANN model error reduction. These four ANN modeling parameters have been identified for the data set from literature and two synthetic data sets. The result shows that these ANN modeling factors' values vary as the data sets internal noises change. The small data set with 63 data points needs a relatively larger hidden neuron number setting of 160 and 36 in the first and second hidden layer with 100 epochs and 60 modeling cycles; For a large data set with 3009 data points with variations of $\pm 0.4\%$ and $\pm 5.4\%$,

neuron number setting in the hidden layer needed was 45 and 142 in the first and second hidden layer, epoch needed was 140, modeling cycles needed were 50 and 70, and the minimum data points number required to achieve a reliable ANN prediction were around 1500 and 2500, respectively. The results highlighted that the optimal values of these ANN modeling factors varied depending on the characteristics and size of the dataset, particularly in response to internal noise levels. This variability underscores the importance of carefully tuning these parameters to achieve robust and accurate BCS predictions across different data scenarios. The optimization scenario is to strike a balance between model error minimization and model complexity, as well as the training efficiency maximization.

To explore the feasibility of applying the ANN approach for evaluating the relative importance of packaging design features, BCS was used as a representative packaging property and up to six BCS features' relative importance have been evaluated in ANN to guide packaging design cost and material saving (Chapter 4). A synthetic dataset (generated using the McKee formula) and a real dataset (from industry) were used to determine the relative feature importance influencing BCS. Four methods—Connection Weights, Gradient-based, Permutation, and SHAP values—were employed in this analysis. This analysis identified the relative importance ranking of six BCS features (ECT, thickness, flexural stiffness in both machine (EI_x) and cross-machine (EI_y) directions of the corrugated board, perimeter, and depth of the box). The result shows that the ANN estimated BCS relative importance ranking aligns with the theoretical relative feature importance ranking calculated using the McKee formula. Notably, the analysis of the real dataset reveals that, although Depth is ranked last, it still has a notable influence. In general, Depth remains an important factor in determining BCS (Buckling Compression Strength). As the depth value increases, buckling theory suggests that depth can significantly affect compression strength,

making it a critical consideration. Despite being theoretically assigned an importance value of zero in the McKee equation [74], implying it may not be a key factor in certain models or calculations, the real-world data demonstrates that its effect should not be disregarded. This result indicated that the ANN predicted BCS feature importance is more reflective to the real-world cases compared with the analytical method. This study demonstrates the capability of the ANN approach in terms of feature importance evaluation for packaging design, helping designers prioritize adjustments to the most influential features, ultimately reducing material consumption and costs. This method can be easily applied to evaluate the relative importance of other packaging properties beyond BCS, offering valuable insights for addressing various challenges in the packaging industry using ANN approaches.

Based on the study conducted above, finally, a generalized ANN model for BCS evaluation has been built using a data set derived from real data (Chapter 5). The ANN modeling factors of epoch and modeling cycles were conservatively set to 140 and 70 based on the training of the data set with variation. The hidden layer neuron number setting was optimized using the same five optimization methods as Chapter 4 (including the Information Criteria using the AIC method, Hebb's rule, the Information Criteria using the BIC method, the Optimal Brain Damage rule, and the Bayesian Optimization method). The optimized hidden neuron setting was identified to be 35 given by the Optimal Brain Damage rule by achieving a balance of model error minimization and model training time-saving. The epoch number and modeling cycle number were determined to 50 and 30 when the training error reduction reached a plateau. With the corresponding ANN modeling parameters and 395 data points, a generalized ANN model was trained and achieved an accuracy of BCS error of 9.51% for the industrial applicable level.

5.2 FUTURE RESEARCH

Great effort should be continuously put into the development of the ANN model improvement as it can bring renovation for corrugated packaging design and optimization, achieving efficiency, sustainability, and cost-effectiveness in the corrugated board industry. First, BCS data obtained from real testing is the most reliable source. Therefore, physical testing can be used to validate the ANN-predicted BCS values and assess the importance of features such as flexural stiffness in the machine and cross-machine directions. Second, additional parameters influencing BCS, such as the corrugated board layer and flute type, can be incorporated into this study. The current research focuses solely on single-wall boxes; however, double-wall boxes and C-flute are in high demand in the U.S. market. Third, other criteria for evaluating the accuracy of the ANN model, beyond MSE, can be considered to provide a more comprehensive understanding of its predictive performance. Additionally, the Finite Element Method (FEM) tool could be utilized to generate BCS data [119], replacing synthetic data derived from the McKee formula. Besides, the ANN model prediction accuracy can be further improved by trying some other techniques that were not involved in this study, such as data transformation [120] to modify the distribution of input variables so that they can better match outputs, data augmentation [121, 122] to boost robust accuracy of the ANN model, weight decay and dropout to improve the generalization performance of ANN model and further improve the model accuracy [123, 124].

Last but not the least, the current data set can be expanded as much as possible to cover more BCS data existing in the industry so that the generalization of the ANN model can be improved to a better level. The more BCS data reflecting industry application can be collected, the more accurate the ANN model prediction fitting to the actual BCS values. Although the current data set has been able to cover 90% of the corrugated boxes commonly used in industry, it is still important to

expand it to cover the remaining 10% of corrugated boxes, which is critical for the final generalization of the ANN model. Further, it is critical to keep up to date the data set to cover the large majority of dimensions of the corrugated boxes used in the industry considering the changing needs in the market, so that the developed ANN model can keep up pace with the needs of customers in modern life.

BIBLIOGRAPHY

1. B. Frank, "Corrugated box compression-A literature survey," *Packaging Technology and science*, vol. 27, no. 2, pp. 105-128, 2014.
2. J. Y.-l. Z. a. J. S. Chen, "An overview of the reducing principle of design of corrugated box used in goods packaging," *Procedia Environmental Sciences*, vol. 10, pp. 992-998, 2011.
3. "Explore Custom Box Types," Online.. Available: <https://customboxesnow.com/box-styles/>.
4. Global Corrugated Packaging Market Outlook: Global Opportunity And Market Segmentation Based On Packaging Type, Based On End-User & By Region With Forecast 2017-2030, 2020.
5. W. Z. W. R. & S. Q. Yi Y., "Life cycle assessment of delivery packages in China," *Energy Procedia*, vol. 105, pp. 3711-3719, 2017.
6. B. Sharma, "Corrugation Trends," *Quarterly Journal of Indian Pulp and Paper Technical Association*, vol. 33, no. E2, pp. 51-54, 2021.
7. "Corrugated Packaging Market - By Product (Corrugated Box, Folding boxboard), By Printing Technique (Lithography, Flexography, Digital Printing), By End-use Industry (Food & beverages, Medical, Agriculture, Industrial, Paper & Carton), & Forecast, 2023 – 2," Global Market Insights Inc., Nov 2022. Online.. Available: <https://www.gminsights.com/industry-analysis/corrugated-packaging-market#:~:text=Corrugated%20Packaging%20Market%20size%20was%20valued,of%20the%20burgeoning%20e-commerce%20sales%20worldwide.&text=Corrugated%20Packaging%20Market%20size,burgeoning%20e-comm>.
8. A. & B. J. Clayton, "Investigation of the Effect of Corrugated Boxes on the Distribution of Compression Stresses on the Top Surface of Wooden Pallets," *Correia, M. (n.d.). Creasing Training Manual*, vol. 5, pp. 5-6, 2018.
9. K. R. N. M. & M. B. T. Ramdass, "Determining the Root Causes of Boxes Stacking Strength Failure and Find Possible Solutions".
10. S. P. S. J. & S. K. Singh, "Effect of palletized box offset on compression strength of unitized and stacked empty corrugated fiberboard boxes," *Journal of Applied Packaging Research*, vol. 5, no. 3, p. 157, 2011.
11. G. T. T. & Ö. S. Meng, "Stacking misalignment of corrugated boxes-a preliminary study," in *In 23rd IAPRI Symposium on Packaging*, Windsor, UK, 2007.
12. J. C. F. A. E. & G. A. Gallo, "Mechanical behavior modeling of containers and octabins made of corrugated cardboard subjected to vertical stacking loads," *Materials*, vol. 14, no. 9, p. 2392, 2021.
13. "PROPERTIES OF PAPER," Online.. Available:

<https://www.paperonweb.com/paperpro.htm>.

14. D. G. T. & K.-P. A. Mrówczyński, “Estimation of the compressive strength of corrugated board boxes with shifted creases on the flaps,” *Materials*, vol. 14, no. 18, p. 5181, 2021.
15. S. Manoj, “Performance And Appearance Of Packaging Grades Of Paper – Study On Quality Measurement Methods,” *The Official International Journal of the Indian Pulp and Paper Technical Association*, vol. 27, no. 4, pp. 29-39, 2015.
16. K. T. & E. S. D. Ulrich, *Product Design and Development* (6th ed.), McGraw-Hill Education, 2016.
17. I. Chalmers, “Evaluating Corrugated Box Performance,” *Corrugator Today*, pp. 1-10, 2019.
18. “Evaluating Box Compression Strength with Compression Testing,” Online.. Available: <https://www.pacorr.com/blog/evaluating-box-compression-strength-with-compression-testing/>.
19. “ASTM D4169 Transit Simulation,” Online.. Available: https://pkgcompliance.com/test/astm-d4169-transit-distribution-simulation/?matchtype=p&network=o&device=c&keyword=ASTM%20D4169&campaign=519511179&adgroup=1316117729722451&msclkid=b7d0db76040214553689bd2308b4ad02&utm_source=bing&utm_medium=cpc&utm_campaign.
20. “Practices of Science: Scientific Error,” Online.. Available: <https://manoa.hawaii.edu/exploringourfluidearth/physical/world-ocean/map-distortion/practices-science-scientific-error#:~:text=Human%20error%20is%20due%20to,recorded%20or%20written%20down%20incorrectly>.
21. P. Group, “The Box compression Test Procedure: A Comprehensive Overview,” 12 April 2024. Online.. Available: <https://medium.com/@prestogrouponline/the-box-compression-test-procedure-a-comprehensive-overview-99e0f0330b22>.
22. R. C. G. J. W. & W. J. R. McKee, “Compression strength formula for corrugated boxes,” *Paperboard Packag*, vol. 48, no. 8, pp. 149-159, 1963.
23. B. a. K. K. Frank, “Assessing variation in package modeling,” *TAPPI JOURNAL*, vol. 20, no. 4, pp. 231-238, 2021.
24. “Chalmers DST MD Torsional Stiffness,” Online.. Available: <http://www.rdmtest.com/p/Chalmers-DST-MD-Torsional-Stiffness/#:~:text=MD%20Torsional%20Stiffness%20measures%20how,in%20real%20world%20service%20environments>.
25. I. R. Chalmers, “The use of MD shear stiffness by the torsional stiffness technique to predict corrugated board properties and box performance,” *Appita: Technology, Innovation, Manufacturing, Environment*, vol. 60, no. 5, pp. 357-361, 2007.

26. T. J. a. B. F. Urbanik, "Box compression analysis of world-wide data spanning 46 years," *Wood and fiber science*, pp. 399-416, 2006.
27. T. C. C. J. B. T. M. A. A. & O. U. L. Fadiji, "The efficacy of finite element analysis (FEA) as a design tool for food packaging: A review," *Biosystems Engineering*, vol. 174, pp. 20-40, 2018.
28. M. A. C. I. G. B. & L. E. Jiménez-Caballero, "Design of different types of corrugated board packages using finite element tools," in *SIMULIA customer conference*, 2009.
29. R. Anyoha, "The History of Artificial Intelligence," Online.. Available: <https://sitn.hms.harvard.edu/flash/2017/history-artificial-intelligence/#:~:text=It%E2%80%99s%20considered%20by%20many%20to%20be%20the%20first,by%20John%20McCarthy%20and%20Marvin%20Minsky%20in%201956..>
30. P. Lisboa, "A review of evidence of health benefit from artificial neural networks in medical intervention," *Neural networks*, vol. 15, no. 1, pp. 11-39, 2002.
31. J. Smith, "Advances in neural networks and potential for their application to steel metallurgy," *Materials Science and Technology*, vol. 36, no. 17, pp. 1805-1819, 2020.
32. A. F. S. a. Z. H. D. Sheikhtaheri, "Developing and using expert systems and neural networks in medicine: a review on benefits and challenges," *Journal of medical systems*, vol. 38, pp. 1-6, 2014.
33. S. K. A. R. E. & B. D. Adamopoulos, "Predicting the properties of corrugated base papers using multiple linear regression and artificial neural networks," *Drewno: prace naukowe, doniesienia, komunikaty*, vol. 59, 2016.
34. S. P. R. a. D. K. Malasri, "Predicting Corrugated Box Compression Strength Using an Artificial Neural Network," *International Journal*, vol. 4, no. 1, pp. 169-176, 2016.
35. T. C. R. S. J. & J. T. Archaviboonyobul, "An analysis of the influence of hand hole and ventilation hole design on compressive strength of corrugated fiberboard boxes by an artificial neural network model," *Packaging Technology and Science*, vol. 33, no. 4-5, pp. 171-181, 2020.
36. A. Krogh, "What are artificial neural networks?," *Nature biotechnology*, vol. 26, no. 2, pp. 195-197., 2008.
37. N. C. Steven Walczak, "Artificial Neural Networks," in *Encyclopedia of Physical Science and Technology (Third Edition)*, 2003.
38. A. Lheureux, "Feed-forward vs feedback neural networks," Online.. Available: <https://blog.paperspace.com/feed-forward-vs-feedback-neural-networks/>.
39. "Feed Forward Neural Network," Online.. Available: <https://deepai.org/machine-learning-glossary-and-terms/feed-forward-neural-network>.
40. "Recurrent Neural Network," Online.. Available: <https://deepai.org/machine-learning-glossary-and-terms/recurrent-neural-network>.

41. A. Singh, "Artificial Neural Network | Types | Feed Forward | Feedback | Structure | Perceptron | Machine Learning | Applications," 21 May 2018. Online.. Available: <https://msatechnosoft.in/blog/artificial-neural-network-types-feed-forward-feedback-structure-perceptron-machine-learning-applications/#:~:text=Feedback%20neural%20networks%20are%20dynamic,equilibrium%20needs%20to%20be%20found..>
42. N. R. T. & B. W. Shahid, "Applications of artificial neural networks in health care organizational decision-making: A scoping review," *PloS one*, vol. 14, no. 2, p. e0212356., 2019.
43. "Reinforcement learning," Wikimedia Foundation, Inc., Online.. Available: https://en.wikipedia.org/wiki/Reinforcement_learning.
44. "Markov decision process," Wikimedia Foundation, Inc., Online.. Available: https://en.wikipedia.org/wiki/Markov_decision_process.
45. O. I. J. A. O. A. E. D. K. V. M. N. A. & A. H. Abiodun, "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, no. 11, 2018.
46. E.-S. M. A. I. S. M. M. E. a. S. E. H. El-Kenawy, "Novel feature selection and voting classifier algorithms for COVID-19 classification in CT images," *IEEE access*, vol. 8, pp. 179317-179335, 2020.
47. P. M. S. a. A. K. Chhajer, "The applications of artificial neural networks, support vector machines, and long–short term memory for stock market prediction.,” *Decision Analytics Journal*, vol. 2, p. 100015, 2022.
48. M. a. Y. S. Qiu, "Predicting the direction of stock market index movement using an optimized artificial neural network model,” *PloS one*, vol. 5, p. e0155133, 2016.
49. D. V. K. a. A. M. Selvamuthu, "Indian stock market prediction using artificial neural networks on tick data,” *Financial Innovation*, vol. 5, no. 1, pp. 1-12, 2019.
50. A. D. K. M. S. a. L. M. G. Ziletti, "Insightful classification of crystal structures using deep learning,” *Nature communications*, vol. 9, no. 1, p. 2775, 2018.
51. K. K. F. G. C. C. S. V. K. R. V. M. Z. a. F. T. Choudhary, "Computational scanning tunneling microscope image database,” *Scientific data*, vol. 8, no. 1, p. 57, 2021.
52. C. B. E. V. Á. S. L. S. G. N. D. V. J. T. T. J. J. B. G. a. C. S. Cooper, "Design-to-device approach affords panchromatic co-sensitized solar cells,” *Advanced Energy Materials*, vol. 9, no. 5, p. 1802820, 2019.
53. V. J. D. L. W. A. D. Z. R. O. K. K. A. P. G. C. a. A. J. Tshitoyan, "Unsupervised word embeddings capture latent knowledge from materials science literature.,” *Nature*, vol. 571, no. 7763, pp. 95-98, 2019.
54. A. M. L. a. C. H. D. Bahrami, "Intelligent design retrieval and packaging system: application of neural networks in design and manufacturing,” *THE INTERNATIONAL*

JOURNAL OF PRODUCTION RESEARCH, vol. 33, no. 2, pp. 405-426, 1995.

55. S. Malasri, "Applications of Neural Networks in Transport Packaging," in *PACKCON 2015*, Online, 2015.
56. Y. X. Y. C. Z. a. Z. W. Liang, "Application of neural networks to identification of nonlinear characteristics in cushioning packaging," *Mechanics research communications*, vol. 23, no. 6, pp. 607-613, 1996.
57. A. M. S. S. Maleki, "Application of artificial neural networks for producing an estimation of high-density polyethylene," *Polymers*, vol. 12, no. 10, p. 2319, 2020.
58. O. S. A. A. T.-C. J. a. I. D. Adeleke, "Application of artificial neural networks for predicting the physical composition of municipal solid waste: An assessment of the impact of seasonal variation," *Waste Management & Research*, vol. 39, no. 8, pp. 1058-1068, 2021.
59. V. V. S. a. C. D.-F. Oliveira, "Artificial neural network modelling of the amount of separately-collected household packaging waste," *Journal of cleaner production*, vol. 210, pp. 401-409, 2019.
60. Y. B. A. C. Ian Goodfellow, *Deep Learning (Adaptive Computation and Machine Learning series, Cambridge, MA: The MIT Press (Illustrated edition), 2016.*
61. M. a. M. K. Kubat, "Artificial neural networks," *An introduction to machine learning*, pp. 117-143, 2021.
62. H. a. Y. C. Hang, "Motion Estimation for Image Sequence Compression," in *Handbook of Visual Communications*, H.-M. Hang and J.W. Woods, Editors, vol. 17, San Diego, Academic Press, 1995, pp. 147-188.
63. R. J. Hyndman, "Moving Averages," in *International Encyclopedia of Statistical Science. 2nd ed.*, Lovric, Miodrag. , Springer, 2011, pp. 866-869.
64. L. J. L. G. C. a. H. M. Zhang, "Research on Packaging Evaluation System of Fast Moving Consumer Goods Based on Analytical Hierarchy Process Method," in *Advanced Graphic Communications and Media Technologies*, Springer Singapore, 2017, pp. 711-718.
65. A. J. V. D. M. A. L. a. M. C. Pérez, "An Analytical Hierarchy Approach Applied in the Packaging Supply Chain," in *Supply Chain Management and Logistics in Emerging Markets*, Emerald Publishing Limited, 2020, pp. 89-104.
66. B. J. G. M. a. D. S. Hicks, "A finite element-based approach for whole-system simulation of packaging systems for their improved design and operation," *Packaging Technology and Science*, vol. 22, no. 4, 2009.
67. J. P. M. C. D. S. J. H. M. & H. S. W. Park, "Finite element-based simulation for edgewise compression behavior of corrugated paperboard for packaging of agricultural products," *Applied Sciences*, vol. 10, no. 19, p. 6716, 2020.
68. H. A. E. A. S. C. G. a. Z. A. Nefeslioglu, "A modified analytical hierarchy process (M-

- AHP) approach for decision support systems in natural hazard assessments,” *Computers & Geosciences*, pp. 1-8, 2013.
69. L. J. L. G. C. a. H. M. Zhang, “Research on Packaging Evaluation System of Fast Moving Consumer Goods Based on Analytical Hierarchy Process Method,” in *Advanced Graphic Communications and Media Technologies*, Singapore, Springer, 2017, pp. 711-718.
 70. S. a. L. S. Lundberg, “A unified approach to interpreting model predictions,” *Advances in neural information processing systems*, vol. 30, 2017.
 71. S.-H. Y.-C. C. a. C.-H. H. Tsaur, “Determinants of guest loyalty to international tourist hotels—a neural network approach,” *Tourism Management*, vol. 23, no. 4, pp. 397-405, 2002.
 72. J. A. S. a. R. A. K. Iqbal, “The relative importance of textual indexes in predicting the future performance of banks: A connection weight approach,” *Borsa Istanbul Review*, vol. 23, no. 1, pp. 240-253, 2023.
 73. A. T. Goh, “Back-propagation neural networks for modeling complex systems,” *Artificial intelligence in engineering*, vol. 9, no. 3, pp. 143-151, 1995.
 74. T. J. K. G. A. C. a. T. G. Gajewski, “On the use of artificial intelligence in predicting the compressive strength of various cardboard packaging,” *Packaging Technology and Science*, vol. 37, no. 2, pp. 97-105, 2024.
 75. S. P. R. a. D. K. Malasri, “Predicting corrugated box compression strength using an artificial neural network,” *International Journal*, vol. 4, no. 1, pp. 169-176, 2016.
 76. T. J. K. G. A. C. a. T. G. Gajewski, “On the use of artificial intelligence in predicting the compressive strength of various cardboard packaging,” *Packaging Technology and Science*, vol. 37, no. 2, pp. 97-105, 2024.
 77. N. d. L. M. a. B. R. 4. da Costa, “Evaluation of feature selection methods based on artificial neural network weights,” *Expert Systems with Applications*, vol. 168, p. 114312, 2021.
 78. A. L. E. a. L. R. Hill, “A Novel Gradient Feature Importance Method for Neural Networks: An Application to Controller Gain Tuning for Mobile Robots,” in *In International Conference on Informatics in Control, Automation and Robotics*. pp. 124-141, Cham: Springer International Publishing, 2020.
 79. C. W. a. D. F. C. Zobel, “Evaluation of neural network variable influence measures for process control,” *Engineering applications of artificial intelligence*, vol. 24, no. 5, pp. 803-812, 2011.
 80. D. G. Garson, Interpreting neural network connection weights, pp. 47-51, 1991, pp. 47-51.
 81. Y. T. G. a. G. S. Yoon, “Integrating artificial neural networks with rule-based expert systems,” *Decision Support Systems*, vol. 11, no. 5, pp. 497-507, 1994.
 82. S.-H. Y.-C. C. a. C.-H. H. Tsaur, “Determinants of guest loyalty to international tourist hotels—a neural network approach,” *Tourism Management*, vol. 23, no. 4, pp. 397-405,

2002.

83. W. B. Mandler H, "Feature importance in neural networks as a means of interpretation for data-driven turbulence models," *Computers & Fluids*, vol. 265, p. 105993, 2023.
84. A. T. Goh, "Back-propagation neural networks for modeling complex systems," *Artificial intelligence in engineering*, vol. 9, no. 3, pp. 143-151, 1995.
85. H.-F. a. S.-H. T. Luoh, "The effects of age stereotypes on tour leader roles," *Journal of Travel Research*, vol. 53, no. 1, pp. 111-123, 2014.
86. J. A. S. a. R. A. K. Iqbal, "The relative importance of textual indexes in predicting the future performance of banks: A connection weight approach," *Borsa Istanbul Review*, vol. 23, no. 1, pp. 240-253, 2023.
87. K. a. L. C. Fukumizu, "Gradient-based kernel method for feature extraction and variable selection," *Advances in neural information processing systems*, vol. 25, 2012.
88. A. L. T. O. S. a. T. L. Altmann, "Permutation importance: a corrected feature importance measure," *Bioinformatics*, vol. 26, no. 10, pp. 1340-1347, 2010.
89. Z. Li, "Extracting spatial effects from machine learning model using local interpretation method: An example of SHAP and XGBoost," *Computers, Environment and Urban Systems*, vol. 96, p. 101845, 2022.
90. G. H. J. a. J. C. Jeon, "Distilled gradient aggregation: Purify features for input attribution in the deep neural network," *Advances in Neural Information Processing Systems*, vol. 35, pp. 26478-26491, 2022.
91. M. a. A. M. A. Azmat, "Feature Importance Estimation Using Gradient Based Method for Multimodal Fused Neural Networks," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, 2022.
92. H. a. B. W. Mandler, "Feature importance in neural networks as a means of interpretation for data-driven turbulence models," *Computers & Fluids*, vol. 265, p. 105993, 2023.
93. G. A. L. M. S. a. D. S. Van den Broeck, "On the tractability of SHAP explanations," *Journal of Artificial Intelligence Research*, vol. 74, pp. 851-86, 2022.
94. Z. Li, "Extracting spatial effects from machine learning model using local interpretation method: An example of SHAP and XGBoost," *Computers, Environment and Urban Systems*, vol. 96, p. 101845, 2022.
95. E. a. I. K. Štrumbelj, "Explaining prediction models and individual predictions with feature contributions," *Knowledge and information systems*, vol. 41, pp. 647-665, 2014.
96. H. Akaike, "Information theory and an extension of the maximum likelihood principle," Springer New York, New York, NY, 1998.
97. "Hebb's rule," Online.. Available: <https://search.brave.com/search?q=Hebb%27s+rule&source=desktop&summary=1&sum>

mary_og=7726c1f501903b41a133a1.

98. G. Schwarz, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, no. 2, pp. 461-464, 1978.
99. G. Schwarz, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, no. 2, pp. 461-464, 1978.
100. Y. J. D. a. S. S. LeCun, "Optimal brain damage," *Advances in neural information processing systems*, vol. 2, 1989.
101. J. Moćkus, "On Bayesian methods for seeking the extremum," *In Optimization techniques IFIP technical conference: Novosibirsk, Springer Berlin Heidelberg*, pp. 400-404, 1974.
102. J. a. J. M. Mockus, "The Bayesian approach to local optimization," *Springer Netherlands*, 1989.
103. W. Koehrsen, "A Conceptual Explanation of Bayesian Hyperparameter Optimization for Machine Learning," Medium, 24 June 2018. Online.. Available: <https://towardsdatascience.com/a-conceptual-explanation-of-bayesian-model-based-hyperparameter-optimization-for-machine-learning-b8172278050f>.
104. P. I. Frazier, "A tutorial on Bayesian optimization," *arXiv preprint arXiv*, vol. 1807.02811, 2018.
105. E. E. v. d. H. a. J. R. Wit, "All models are wrong...': an introduction to model uncertainty," *Statistica Neerlandica*, vol. 66, no. 3, pp. 217-236, 2012.
106. G. a. N. L. H. Claeskens, *Model selection and model averaging*, Cambridge books, 2008.
107. Michael Weiß and Markus Goßner, "The Yeasts (Fifth Edition)," in *Molecular Phylogenetic Reconstruction*, Elsevier Science, 2011, pp. 159-174.
108. "Akaike information criterion," Wikimedia Foundation, Inc., 25 May 2024. Online.. Available: https://en.wikipedia.org/wiki/Akaike_information_criterion.
109. C. N. B. H. F. Emad A. Mohammed, "Emerging Business Intelligence Framework for a Clinical Laboratory Through Big Data Analytics," in *Emerging Trends in Computational Biology, Bioinformatics, and Systems Biology*, 2015, pp. 577-602.
110. P. a. Y. S. Stoica, "Model-order selection: a review of information criterion rules," *IEEE Signal Processing Magazine*, vol. 21, no. 4, pp. 36-47, 2004.
111. K. P. a. D. R. A. e. Burnham, *Model selection and multimodel inference: a practical information-theoretic approach*, New York, NY: Springer New York (2nd edition), 2002.
112. H. Akaike, "A new look at the statistical model identification," *IEEE transactions on automatic control*, vol. 19, no. 6, pp. 716-723, 1974.
113. P. I. Frazier, "A tutorial on Bayesian optimization," *arXiv preprint arXiv:1807.02811*., 2018.

114. "Hebbian theory," Wikimedia Foundation, Inc., 16 May 2024. Online.. Available: https://en.wikipedia.org/wiki/Hebbian_theory.
115. Y. E. K. L. G. K. V. a. I. V. 2. ". m. f. p. c. s. o. p. c. Pyr'yev, *Wood and Fiber Science*, vol. 54, no. 1.
116. F. B. L. E. Gu J, "A Comparative Analysis of Artificial Neural Network (ANN) Architectures for Box Compression Strength Estimation," *Korean J Packag Sci Technol*, vol. 29, no. 3, pp. 163-74, 2023.
117. R. C. J. W. G. a. J. R. W. McKee, "Compression strength formula for corrugated boxes," *Paperboard Packag* , vol. 48, no. 8, pp. 149-159, 1963.
118. Y. E. K. L. G. K. V. a. I. V. Pyr'yev, "Empirical models for prediction compression strength of paperboard carton," *Wood and Fiber Science*, vol. 54, no. 1, 2022.
119. T. A. A. C. C. B. T. a. O. U. Fadiji, "Application of finite element analysis to predict the mechanical strength of ventilated corrugated paperboard packaging for handling fresh produce," *Biosystems Engineering*, vol. 174, pp. 260-281, 2018.
120. J. J. Shi, "Reducing prediction error by transforming input data for neural networks," *Journal of computing in civil engineering*, vol. 14, no. 2, pp. 109-116, 2000.
121. S.-A. S. G. D. A. C. F. S. O. W. a. T. A. M. Rebuffi, "Data augmentation can improve robustness," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29935-29948, 2021.
122. L. Z. J. P. N. C. L. D. M. A. N. a. N. S. Khan, "Data augmentation to improve performance of neural networks for failure management in optical networks," *Journal of Optical Communications and Networking*, vol. 15, pp. 57-67, 2023.
123. A. a. J. H. Krogh, "A simple weight decay can improve generalization," *Advances in neural information processing systems*, vol. 4, 1991.
124. N. Srivastava, "Improving neural networks with dropout," University of Toronto, Toronto, 2013.
125. J. S. C. a. H. J. Park, "Numerical prediction of equivalent mechanical properties of corrugated paperboard by 3D finite element analysis," *Applied Sciences*, vol. 10, no. 22, p. 7973, 2020.
126. J.-M. T.-Y. P. a. H.-M. J. Park, "Prediction of Deflection Due to Multistage Loading of a Corrugated Package," *Applied Sciences*, vol. 13, no. 7, p. 4236, 2023.
127. E. a. L. H. Molina, "Development of a Gaussian Process Model as a Surrogate to Study Load Bridging Performance in Racked Pallets," *Applied Sciences*, vol. 11, no. 24, p. 11865, 2021.
128. R. C. J. W. B. S. P. R. & S. M. Haj-Ali, "Refined nonlinear finite element models for corrugated fiberboards," *Composite Structures*, vol. 87, no. 4, pp. 321-333, 2009.

129. G. J. S. Y. Z. D. & X. Y. Hua, "Experimental and numerical analysis of the edge effect for corrugated and honeycomb fiberboard," *Strength of Materials*, vol. 49, no. 1, pp. 188-197, 2017.
130. T. T. G. a. J. G. Garbowski, "Estimation of the compressive strength of corrugated cardboard boxes with various perforations," *Energies*, vol. 14, no. 4, p. 1095, 2021.
131. T. T. G. a. J. G. Garbowski, "Estimation of the compressive strength of corrugated cardboard boxes with various openings," *Energies*, vol. 14, no. 1, p. 155, 2020.
132. M. a. C. B. Biancolini, "Numerical and experimental investigation of the strength of corrugated board packages," *Packaging Technology and Science*, vol. 16, no. 2, pp. 47-60, 2003.
133. J. a. J. P. Han, "Finite element analysis of vent/hand hole designs for corrugated fibreboard boxes," *Packaging Technology and Science*, vol. 20, no. 1, pp. 39-47, 2007.
134. T. G. T. M. D. & J. R. Garbowski, "Crushing of single-walled corrugated board during converting: Experimental and numerical study," *Energies*, vol. 14, no. 11, p. 3203, 2021.
135. G. S. P. N. M. & Ö. S. Marin, "Experimental and finite element simulated box compression tests on paperboard packages at different moisture levels," *Packaging Technology and Science*, vol. 34, no. 4, pp. 229-243, 2021.
136. T. Kobayashi, "Numerical Simulation for Compressive Strength of Corrugated Fiberboard Box," *JAPAN TAPPI JOURNAL*, vol. 73, no. 8, pp. 793-800, 2019.
137. S. Kleene, "Representation of events in nerve nets and finite automata," *Automata studies*, vol. 34, pp. 3-41, 1956.
138. V. Mitić, "Benefits of artificial intelligence and machine learning in marketing," in *Sinteza 2019-International scientific conference on information technology and data related research*, Singidunum University, 2019.
139. A. a. B. S. Sun, "How can Big Data and machine learning benefit environment and water management: a survey of methods, applications, and future directions," *Environmental Research Letters*, vol. 14, no. 7, p. 073001, 2019.
140. Y. Pi, "Machine learning in governments: Benefits, challenges and future directions," *JeDEM-eJournal of eDemocracy and Open Government*, vol. 13, no. 1, pp. 203-219, 2021.
141. S. Kalogirou, "Artificial neural networks in renewable energy systems applications: a review," *Renewable and sustainable energy reviews*, vol. 5, no. 4, pp. 373-401, 2001.
142. B. L. İ. M. O. O. S. G. A. N. S. M. & S. B. Aylak, "Application of machine learning methods for pallet loading problem," *Applied Sciences*, vol. 11, no. 18, p. 8304, 2021.
143. K.-P. A. G. J. Garbowski T, "Estimation of the Edge Crush Resistance of Corrugated Board Using Artificial Intelligence," *Materials*, vol. 16, p. 1631, 2023.
144. T. Jacob, "Vanishing Gradient Problem: Causes, Consequences, and Solutions," 15 June

2023. Online.. Available: <https://www.kdnuggets.com/2022/02/vanishing-gradient-problem.html#:~:text=When%20there%20are%20more%20layers,this%20the%20vanishing%20gradient%20problem..>
145. “Tanh Activation,” Online.. Available: <https://paperswithcode.com/method/tanh-activation#:~:text=Tanh%20Activation%20is%20an%20activation,for%20multi%2Dlayer%20neural%20networks.>
 146. S. Sharma, “Activation Function in Neural Networks,” Towards Data Science, 6 September 2017. Online.. Available: <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>.
 147. “Vanishing gradient problem,” Wikimedia Foundation, Inc, Online.. Available: https://en.wikipedia.org/wiki/Vanishing_gradient_problem.
 148. M. J. G. S. a. A. W. Roodschild, “A new approach for the vanishing gradient problem on sigmoid activation,” *Progress in Artificial Intelligence*, vol. 9, no. 4, pp. 351-360, 2020.
 149. “Multi-Class Neural Networks: Softmax,” Online.. Available: <https://developers.google.com/machine-learning/crash-course/multi-class-neural-networks/softmax>.
 150. B. P. C, “Softmax Activation Function: Everything You Need to Know,” 30 June 2023. Online.. Available: <https://www.pinecone.io/learn/softmax-activation/>.
 151. T. Wood, “Softmax Function,” Online.. Available: <https://deepai.org/machine-learning-glossary-and-terms/softmax-layer>.
 152. S. Polamuri, “DIFFERENCE BETWEEN SOFTMAX FUNCTION AND SIGMOID FUNCTION,” Online.. Available: <https://dataaspirant.com/difference-between-softmax-function-and-sigmoid-function/>.
 153. S. Shah, “Cost Function is No Rocket Science!,” 20 March 2024. Online.. Available: <https://www.analyticsvidhya.com/blog/2021/02/cost-function-is-no-rocket-science/>.
 154. “Cost Function in Machine Learning,” Online.. Available: <https://www.javatpoint.com/cost-function-in-machine-learning>.
 155. MILK, “Dummies guide to Cost Functions in Machine Learning with Animation.,” 14 November 2020. Online.. Available: <https://machinelearningknowledge.ai/cost-functions-in-machine-learning/>.
 156. A. A. & N. L. A. Faisal, “Simulation of ammonia nitrogen removal from simulated wastewater by sorption onto waste foundry sand using artificial neural network,” *Association of Arab Universities Journal of Engineering Sciences*, vol. 26, no. 1, pp. 28-34, 2019.
 157. N.-T. a. K.-U. D. Vu, “Prediction of ammonium removal by biochar produced from agricultural wastes using artificial neural networks: Prospects and bottlenecks,” *Soft Computing Techniques in Solid Waste and Wastewater Management*, pp. 455-467, 2021.

158. Y.-S. a. S. L. Park, "Artificial neural networks: Multilayer perceptron for ecological modeling.," *Developments in environmental modelling*, vol. 28, pp. 123-140, 2016.
159. E. V. A. K. S. A. K. V. S. S. V. P. K. K. T. B. & P. A. Antunes, "Application of biochar for emerging contaminant mitigation," *Advances in Chemical Pollution, Environmental Management and Protection*, vol. 7, pp. 65-91, 2021.
160. H. Y. R. R. A. & N. P. A. Kang, "Artificial neural network modeling of phytoplankton blooms and its application to sampling sites within the same estuary," *Elsevier*, pp. 161-172, 2011.
161. M. M. S. H. B. I. M. a. M. I. H. S. Hussain, "Application of different artificial neural network for streamflow forecasting," *Advances in streamflow forecasting*, pp. 149-170, 2021.
162. M. Z. F. B. D. P. N. a. K. K. Mohseni-Dargah, "Machine learning in surface plasmon resonance for environmental monitoring," *In Artificial intelligence and data science in environmental sensing*, pp. 269-298, 2022.
163. Z. R. H. a. W. H. Zhang, "Application of Artificial Neural Network Algorithm in Facial Biological Image Information Scanning and Recognition," *Contrast Media & Molecular Imaging*, 2022.
164. M. R. a. R. Z. Jabłońska, "Artificial neural networks for predicting social comparison effects among female Instagram users," *PloS one*, vol. 15, no. 2, 2020.
165. L. S. N. P. a. P. L. N. M. Berke, "Optimum design of aerospace structural components using neural networks.," *Computers & structures*, vol. 48, no. 6, pp. 1001-1010, 1993.
166. S. K. K. D. J. R. D. V. B. G. a. T. R. N. Paul, "Application of artificial neural networks in aircraft maintenance, repair and overhaul solutions," *arXiv preprint arXiv*, vol. 1001, no. 3741, 2010.
167. Y. Y. L. a. Y. W. Li, "New Algorithm of Traditional Chinese Medicine and Protection of Intangible Cultural Heritage Based on Big Data Deep Learning," *BioMed Research International*, 2022.
168. G. G. S. V. N. V. K. a. T. S. Gopichand, "Digital signature verification using artificial neural networks," *Int J Recent Technol Eng (IJRTE) Blue Eyes Intell Eng*, vol. 7, p. 552, 2019.
169. M. Kumar, "Signature verification using neural network," *International Journal on Computer Science and Engineering*, vol. 4, no. 9, p. 1498, 2012.
170. A. B. D. a. S. B. Karouni, "Offline signature recognition using neural networks approach," *Procedia Computer Science*, vol. 3, pp. 155-161, 2011.
171. K. M. P. S. S. G. a. A. A. Abhishek, "Weather forecasting model using artificial neural network," *Procedia Technology*, vol. 4, pp. 311-318, 2012.
172. D. N. a. D. K. S. Fente, "Weather forecasting using artificial neural network," *second*

- international conference on inventive communication and computational technologies (ICICCT)*, pp. 1757-1761, 2018.
173. M. a. Z. M. Hayati, "Application of artificial neural networks for temperature forecasting," *International Journal of Electrical and Computer Engineering*, vol. 1, no. 4, pp. 662-666, 2007.
 174. K. D. B. C. C. J. A. T. F. C. R. P. C. C. A. A. B. S. a. H. E. Choudhary, "Recent advances and applications of deep learning methods in materials science," *npj Computational Materials*, vol. 8, no. 1, p. 59, 2022.
 175. T. a. J. C. G. Xie, "Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties," *Physical review letters*, vol. 120, no. 14, p. 145301, 2018.
 176. T. a. J. C. G. Xie, "Hierarchical visualization of materials space with graph convolutional neural networks," *chemical physics*, vol. 149, no. 17, 2018.
 177. L. A. A. A. C. a. C. W. Ward, "A general-purpose machine learning framework for predicting properties of inorganic materials," *npj Computational Materials*, vol. 2, no. 1, pp. 1-7, 2016.
 178. W. B. J. C. J. J. K. S. S. P. S. M. P. N. S. a. K.-S. S. Park, "Classification of crystal structure using a convolutional neural network," *IUCrJ*, vol. 4, no. 4, pp. 486-494, 2017.
 179. M. Hellenbrandt, "The inorganic crystal structure database (ICSD)—present and future," *Crystallography Reviews*, vol. 10, no. 1, pp. 17-22, 2004.
 180. V. G. H. P. G. a. B. G. S. Fung, "Machine learned features from density of states for accurate adsorption energy prediction," *Nature communications*, vol. 12, no. 1, p. 88, 2021.
 181. Y. E. K. L. G. K. V. a. I. V. Pyr'yev, "Empirical models for prediction compression strength of paperboard carton," *Wood and Fiber Science*, vol. 54, no. 1, 2022.
 182. P. R. B. P. Fehér L, "Compression strength estimation of corrugated board boxes for a reduction in sidewall surface cutouts—experimental and numerical approaches," *Materials*, vol. 16, no. 2, p. 597, 2023.
 183. S. P. R. a. D. K. Malasri, "Predicting corrugated box compression strength using an artificial neural network," *International Journal*, vol. 1, pp. 169-176, 2016.
 184. S. a. A. C. Chakravorty, "Hidden layer optimization of neural network using computational technique," in *the International Conference on Advances in Computing, Communication and Control*, 2009.
 185. "Bayesian optimization," Wikimedia Foundation, Inc., 13 May 2024. Online.. Available: https://en.wikipedia.org/wiki/Bayesian_optimization.
 186. "Hebbian theory," Wikimedia Foundation, Inc., 16 May 2024. Online.. Available: https://en.wikipedia.org/wiki/Hebbian_theory.

187. Y. J. D. a. S. S. LeCun, "Optimal brain damage," *Advances in neural information processing systems* 2, vol. 2, 1989.
188. P. a. Y. S. Stoica, "Model-order selection: a review of information criterion rules," *IEEE Signal Processing Magazine*, vol. 21, no. 4, pp. 36-47, 2004.
189. R. McElreath, *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*, CRC Press, 2016, p. 189.
190. M. Taddy, in *Business Data Science: Combining Machine Learning and Economics to Optimize, Automate, and Accelerate Business Decisions*, New York, McGraw-Hill, 2019, p. 90.
191. H. Akaike, "Information theory and an extension of the maximum likelihood principle," New York, NY, Springer New York, 1998, pp. 199-213.
192. A. H., "Information theory and an extension of the maximum likelihood principle," *Proc: 2nd Int symp information theory. Budapest*, p. 267-81, 1973.
193. J. Močkus, "On bayesian methods for seeking the extremum," *Optimization Techniques IFIP Technical Conference Novosibirsk*, vol. 27, p. 400-404, 1975.
194. W. Koehrsen, "A Conceptual Explanation of Bayesian Hyperparameter Optimization for Machine Learning," Medium, 24 June 2018. Online.. Available: <https://towardsdatascience.com/a-conceptual-explanation-of-bayesian-model-based-hyperparameter-optimization-for-machine-learning-b8172278050f>.
195. W. a. D. R. N. Li, "Marker selection by Akaike information criterion and Bayesian information criterion," *Genetic Epidemiology*, vol. 21, no. S1, pp. S272-S277, 2001.
196. H. a. M. H. Wang, "Supervised Hebb rule based feature selection for text classification," *Information Processing & Management*, vol. 56, no. 1, pp. 167-191, 2019.
197. D. X. X. a. F. P. Lin, "Bayesian Information Criterion Based Feature Filtering for the Fusion of Multiple Features in High-Spatial-Resolution Satellite Scene Classification," *Journal of Sensors* 2015, p. 142612, 2015.
198. Y. a. N. S. Mate, "Hybrid feature selection and Bayesian optimization with machine learning for breast cancer prediction," *In 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, pp. 612-619, 2021.
199. C. Z. Z. a. D. W. Liu, "Pruning deep neural networks by optimal brain damage," *Interspeech*, pp. 1092-1095, 2014.
200. T. a. R. A. S. Maheshwari, "Study of the Effect of Squareness of the Corrugated Box on its Box Compression Strength," *Int. J. Latest Technol. Eng. Manag. Appl. Sci*, vol. 6, pp. 26-28, 2017.
201. L. J. L. G. C. a. H. M. Zhang, "Research on Packaging Evaluation System of Fast Moving Consumer Goods Based on Analytical Hierarchy Process Method," in *Advanced Graphic Communications and Media Technologies*, Springer Singapore, 2017, pp. 711-718.

202. S. S. S. M. R. M. S. H. a. S. S. I. Beg, "Application of design of experiments (DoE) in pharmaceutical product and process optimization," *Pharmaceutical quality by design*, pp. 43-64, 2019.
203. J. a. T. M. Hron, "Application of design of experiments to welding process of food packaging," *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis*, vol. 61, no. 4, pp. 909-915, 2013.
204. T. C. J. C. T. M. B. A. A. a. U. L. O. Fadiji, "The efficacy of finite element analysis (FEA) as a design tool for food packaging: A review," *Biosystems Engineering*, vol. 174, pp. 20-40, 2018.
205. B. J. G. M. a. D. S. Hicks, "A finite element-based approach for whole-system simulation of packaging systems for their improved design and operation," *Packaging Technology and Science*, vol. 22, no. 4, 2009.
206. J. M. P. D. S. C. H. M. J. a. S. W. H. Park, "Finite element-based simulation for edgewise compression behavior of corrugated paperboard for packaging of agricultural products," *Applied Sciences*, vol. 10, no. 9, p. 6716, 2020.
207. H. N. J. M. T. J. Nygård M, "A finite element model for simulations of creasing and folding of paperboard," in *Abaqus Users*, 2005.
208. N. L. M. D. d. L. a. R. B. da Costa, "Evaluation of feature selection methods based on artificial neural network weights," *Expert Systems with Applications*, vol. 168, p. 114312, 2021.
209. B. a. A. S. Iooss, "Introduction to sensitivity analysis," in *Handbook of uncertainty quantification*, 2017, pp. 1103-1122.
210. A. E. L. a. R. L. Hill, "A Novel Gradient Feature Importance Method for Neural Networks: An Application to Controller Gain Tuning for Mobile Robots," in *International Conference on Informatics in Control, Automation and Robotics*, 2020.
211. S. M. a. S.-I. L. Lundberg, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.
212. C. W. a. D. F. C. Zobel, "Evaluation of neural network variable influence measures for process control," *Engineering applications of artificial intelligence*, vol. 24, no. 5, pp. 803-812, 2011.
213. D. G. Garson, "Interpreting neural network connection weights," pp. 47-51, 1991.
214. K. a. C. L. Fukumizu, "Gradient-based kernel method for feature extraction and variable selection," *Advances in neural information processing systems*, vol. 25, 2012.
215. A. L. T. O. S. a. T. L. Altmann, "Permutation importance: a corrected feature importance measure," *Bioinformatics*, vol. 26, no. 10, pp. 1340-1347, 2010.
216. T. a. F. B. Urbanik, "Box compression analysis of world-wide data spanning 46 years," *Wood and fiber science*, pp. 399-416, 2006.

217. S. P. R. a. D. K. Malasri, "Predicting corrugated box compression strength using an artificial neural network," *International Journal*, vol. 4, no. 1, pp. 169-176, 2016.
218. A. J. V. D. M. A. L. a. M. C. Pérez, "An Analytical Hierarchy Approach Applied in the Packaging Supply Chain," in *upply Chain Management and Logistics in Emerging Markets*, Emerald Publishing Limited, 2020, pp. 89-104.
219. S. S. S. M. R. M. S. H. a. S. S. I. Beg, "Application of design of experiments (DoE) in pharmaceutical product and process optimization," *Pharmaceutical quality by design*, Vols. 43-64, 2019.
220. J. a. T. M. Hron, "Application of design of experiments to welding process of food packaging," *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis*, vol. 61, no. 4, pp. 909-915, 2013.