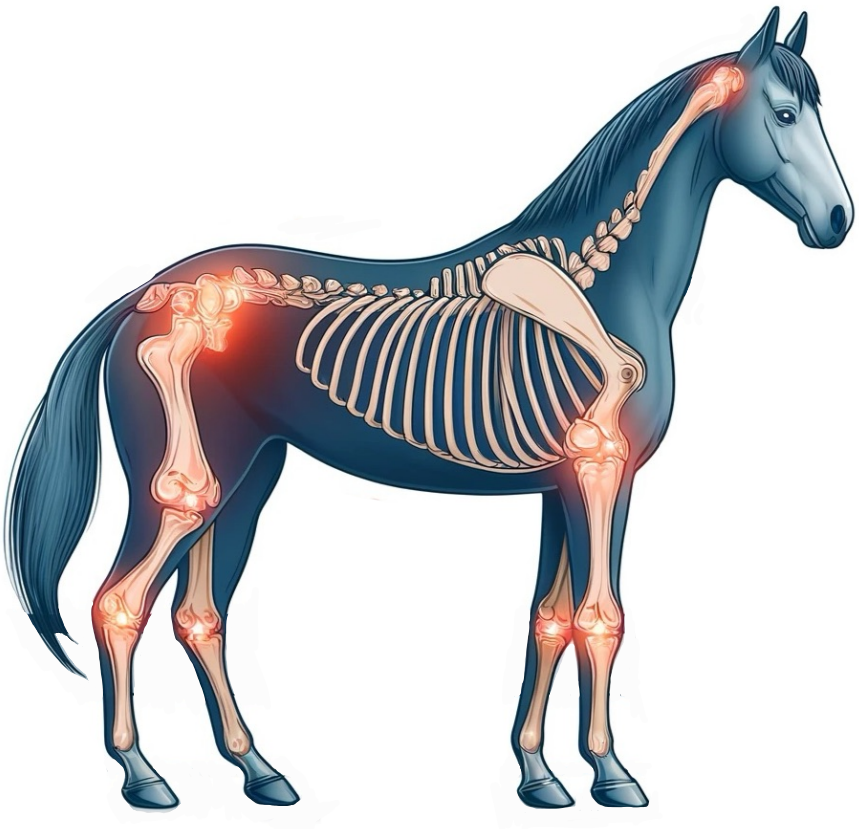
CHALMERS
UNIVERSITY OF TECHNOLOGY



Deep Learning in Early Osteoarthritis Detection via Biomarkers

Deep Learning for Differential Diagnosis in Equine Osteoarthritis: Exploring Synovial Fluid Biomarkers

Master's thesis in Complex Adaptive Systems

Sirada Kaewchino

DEPARTMENT OF PHYSICS
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Deep Learning in Early Osteoarthritis Detection via Biomarkers

Deep Learning for Differential Diagnosis in Equine Osteoarthritis: Exploring Synovial Fluid Biomarkers

SIRADA KAEWCHINO

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Physics
Division of Material Physics
Chalmers University of Technology
Gothenburg, Sweden 2024

Deep Learning in Early Osteoarthritis Detection via Biomarkers
Deep Learning for Differential Diagnosis in Equine Osteoarthritis:
Exploring Synovial Fluid Biomarkers
SIRADA KAEWCHINO

Supervisor/Examiner: Magnus Karlsteen, Department of Physics, Chalmers
Advisor: Eva Skiöldebrand, Department of Biomedical Sciences and Veterinary Public Health, SLU

Master's Thesis 2024
Department of Physics
Division of Material Physics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Deep Learning in Early Osteoarthritis Detection via Biomarkers
Deep Learning for Differential Diagnosis in Equine Osteoarthritis: Exploring Synovial Fluid Biomarkers
SIRADA KAEWCHINO
Department of Physics
Chalmers University of Technology

Abstract

In the past, radiographic techniques have been used to diagnose osteoarthritis (OA), a common joint disease leading to lameness in racehorses. However, these methods typically only reveal the disease once irreversible damage has occurred. In this thesis, the focus shifts from imaging to biological biomarkers for earlier detection. Using synovial fluid biomarkers, this work explores the potential for deep machine learning models to accurately classify early stages of OA, thereby enabling timely interventions to prevent disease progression. In addition, a user-friendly web application was developed to assist practitioners in making real-time diagnoses. Based on preliminary results, deep learning approaches, particularly those involving neural networks, can effectively differentiate between stages of OA, offering a promising tool for diagnosis and treatment. These findings suggest significant potential for improving diagnostic accuracy and, consequently, treatment outcomes in veterinary medicine.

Keywords: Deep Machine learning, engineering, biomarkers, osteoarthritis, diagnosis, treatment

Acknowledgements

My first thanks go to my supervisor and examiner Magnus Karlsteen for giving me the opportunity to combine two of my great passions in life in this thesis work. I would like to thank you for allowing me to take a deep dive into a subject that I'm extremely passionate about. As well as expressing gratitude to Eva Skiöldebrand for her kind words and extensive knowledge about horses and Osteoarthritis, I would also like to send a special thank you to her! The discovery of your research has been a pleasure, and I look forward to the future when I will be able to see where it may lead in the not too distant future.

I would also like to thank Francesco, who has been a great help with this thesis and motivated me throughout. In addition, I would like to thank Samuel and Jonas for making these five years at Chalmers full of happy memories and laughter. Last but not least, I would like to extend my thanks to all of my friends and family who have been supportive of my work and have provided me with great feedback and cheers along the way.

Sirada Kaewchino, Gothenburg, June 2024

Contents

List of Figures

List of Tables

1
Introduction

This thesis presents a study conducted in collaboration with Chalmers University of Technology, the Swedish University of Agricultural Sciences, the University of Gothenburg, and Sahlgrenska University Hospital. The focus is on early detection of osteoarthritis (OA) using deep learning techniques. The objective is to develop a decision support system for practitioners, enabling early identification of structural changes in joints. This is done through a deep machine learning model applied to biomarkers. This system aims to detect cartilage degradation long before clinical symptoms appear. This will allow for earlier intervention and improve our understanding of OA's complex nature.

1.1 Problem Specification

The research aims to utilize deep learning techniques to accurately classify OA stages in horses, facilitating an effective decision-support system for veterinary practitioners. The significance of this approach lies in detecting early cartilage degradation before clinical symptoms manifest, allowing for predictions of disease progression. Early detection and intervention are crucial for improving health and welfare by halting or reversing disease progression.

In this study, the dataset consists of synovial fluid samples collected from horses, categorized into four classifications: Healthy, Mild Osteoarthritis, Moderate Osteoarthritis, and Severe Osteoarthritis. The analysis focuses on four specific biomarkers: NGF, COMP664, COMP156, and BGN262, selected for their relevance in diagnosing and understanding OA progression.

1.2 Research

Recent advancements in OA treatment have focused on developing disease-modifying osteoarthritic drugs (DMOADs) that alleviate symptoms and slow joint damage progression. A recent randomized, triple-blinded controlled clinical study highlighted a novel treatment combination (TC) involving sildenafil, mepivacaine, and glucose, which significantly reduced osteoarthritic symptoms and markers of extracellular matrix degradation in horses. This underscores the importance of developing updated DMOADs and utilizing biomarkers for precise diagnosis and treatment monitoring. This will enhance early intervention and improve OA management in both veterinary and human medicine [2].

1.3 Hypothesis

The central hypothesis of this research posits that OA progresses through four distinct stages. It also posits that early diagnosis before clinical symptoms emerge is feasible through specific biomarker identification. This thesis assumes that a deep machine learning model can accurately categorize newly collected data into one of the four classifications. The investigation aims to explore and validate a methodology that, with more comprehensive datasets in the future, could enhance diagnostic accuracy and improve equine health outcomes.

1.4 Limitations

One of the principal challenges is the necessity to apply selection criteria to ensure dataset relevance and specificity to OA diagnostic markers. This filtration process, essential for isolating samples exhibiting the targeted biomarkers indicative of OA, leads to a substantial reduction in dataset size. Such a contraction poses potential risks to the study's robustness and applicability. The dataset may not fully capture the diversity and complexity of OA's manifestations across different stages and individuals. This may affect the generalizability of the predictive model developed from this research.

Furthermore, relying on a limited set of proteins as diagnostic parameters might overlook other biomarkers that could offer additional insights into the disease's dynamics. While the chosen proteins are based on current scientific understanding and believed to play critical roles in OA. However, the evolving nature of biomedical research could reveal new markers that provide more comprehensive diagnostic capabilities.

Acknowledging these limitations, this study emphasizes the importance of continuous data collection and expanding datasets to include a wider range of biomarkers. It also includes more extensive samples reflecting OA's diverse manifestations. Such advancements are crucial for refining predictive models, enhancing diagnostic accuracy and reliability. They also contribute more effectively to the early detection and management of osteoarthritis in horses.

2
Background

Osteoarthritis (OA) is a form of joint disease and the most prevalent type of arthritis. It frequently affects human athletes and is the leading cause of lameness and suboptimal performance in animal athletes, including racehorses [3]. Early detection of OA allows for timely interventions, which prevent chronic and painful joint tissue deterioration. This chapter will offer a concise overview of the research project, outlining the fundamental aspects of the disease and the relevant biomarkers.

2.1 Osteoarthritis

In horses, high-intensity activities and repetitive joint loading are significant contributors to OA. Horses with OA typically present with lameness, joint swelling, stiffness, and reduced range of motion. Clinical signs vary depending on the stage of the disease and the specific joints affected. Diagnosis of OA involves a combination of clinical examination, imaging techniques (such as radiography, ultrasound, MRI, and CT), and analysis of synovial fluid. Early diagnosis is crucial for effective management and slow disease progression. OA management in horses includes both non-pharmacological and pharmaceutical approaches. Non-pharmacological treatments involve rest, controlled exercise, weight management, and physical therapy. Pharmacological treatments include non-steroidal anti-inflammatory drugs (NSAIDs), corticosteroids, hyaluronic acid, and disease-modifying osteoarthritis drugs (DMOADs). Intraarticular injections and regenerative therapies, such as platelet-rich plasma (PRP) and stem cell therapy, are also used[4].

2.2 Anatomy and Function of Synovial Joints

The joint is a functional unit formed by anatomical structures and is shown in Figure 2.1. In the musculoskeletal system, synovial joints are also known as diarthrodial joints or movable joints. Fibrous capsules are enclosed in a synovial membrane. This membrane holds synovial fluid and connects adjacent bones covered by articular cartilage [5][6][7]. Synovial joints facilitate bone movement and absorb impacts shocks. Their function and structure classify them into high motion joints, prone to osteochondrosis, and low motion joints, often associated with osteoarthritis. While they provide strength and impact resistance, they have limited flexibility and repair capacity [8].

Joint cartilage is tissue without blood vessels, nerves, or lymphatic vessels. It is

made up of cells called chondrocytes and a substance known as the extracellular matrix (ECM) [9][6]. The ECM contains key components that help it function properly. Biglycan (BGN), a type of protein, contributes to forming the ECM structure. Another protein, called Cartilage Oligomeric Matrix Protein (COMP), stabilizes the collagen network by binding to collagen molecules. ECM proteoglycans attract water, which helps cartilage resist compression. Collagen fibers give cartilage strength. Calcified cartilage connects the cartilage to the bone [6]. Joint pain is usually caused by the innervated bone and other surrounding components.

Synovial fluid (SF) is primarily an ultra filtrate of blood plasma [6]. Due to the vascularized joint capsule and the lack of a basement membrane, blood can freely pass through the synovial cavity. In order to facilitate nutrient delivery and waste removal, hydrostatic pressure and colloid osmotic pressure differences are used during loading. In addition to maintaining joint homeostasis, SF also coordinates communication between articular tissues and distributes pressure when loading is applied [9].

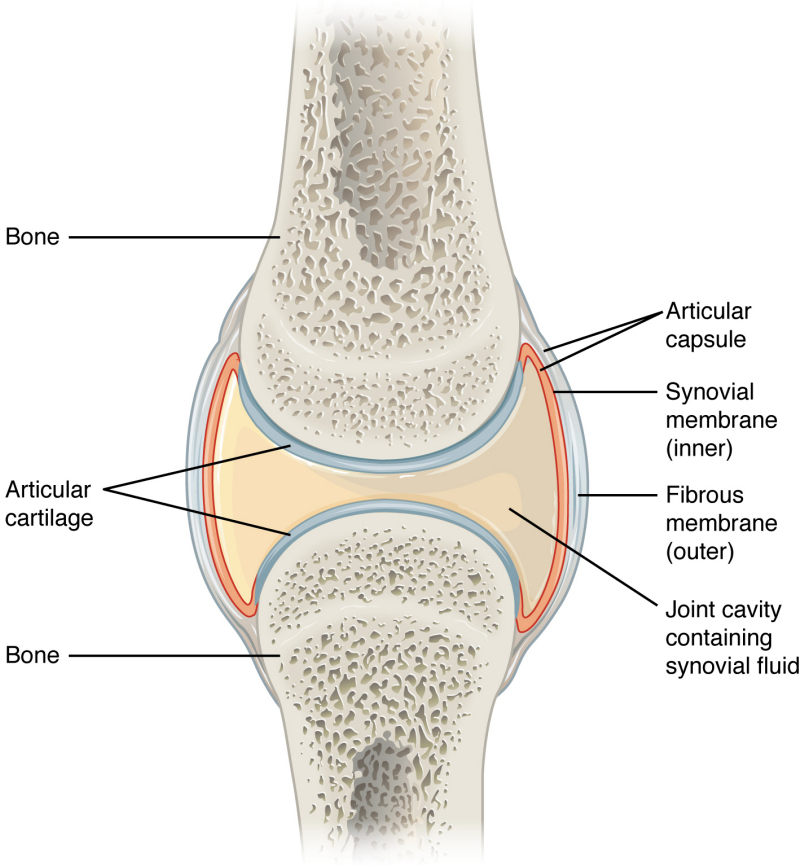


Figure 2.1: A cross section of a healthy synovial joint [1]

2.3 Biomarkers

In order to diagnose the disease earlier the focus has been shifted to biological markers, also known as biomarkers. Biomarkers can be measured in a biological system, and can be chemical, physical or biological [10]. Biomarkers can be used as a diagnostic tool when examining a biological process, such as the destruction of joint tissues in OA. Biomarkers are often measured in synovial fluid and serum, as these fluids reflect the joint's metabolic state. This study focuses on a variety of biomarkers reflecting cartilage degradation, bone turnover, and inflammation in joints. Key biomarkers include COMP, BGN, and since OA leads to severe pain, a relevant biomarker is the Nerve Growth Factor (NGF) [11].

2.4 Cartilage Oligomeric Matrix Protein COMP

The COMP protein, known as thrombospondin 5, is composed of five subunits, each containing 755 amino acids [12]. A damaged collagen network in joints leads to irreversible osteoarthritis (OA) [12]. In the early stages of OA, cartilage COMP levels increase. Inhibition ELISAs are used to detect COMP fragments in synovial fluid and serum, which serve as indicators of joint damage and monitor disease progression. Further details on ELISA methodology can be found in Section 2.4. As an indicator of early joint changes and OA in horses, COMP neoepitope levels in serum are useful for monitoring age, training intensity, and acute lameness impact on joint health. This supports its use for early OA detection and assessing therapeutic interventions [13].

2.4.1 Cartilage Oligomeric Matrix Protien (COMP156)

SF COMP156 levels in horses treated with Celestone® bifas® (CB) were significantly elevated, indicating severe cartilage damage, in a study by Skiöldebrand et al. (2023). This suggests the treatment might harm cartilage. Normally, effective OA treatments should reduce markers of cartilage damage like COMP156. The increase suggests that CB might negatively affect cartilage health, making COMP156 an appropriate biomarker for assessing OA progression and treatment effects. Monitoring COMP156 can help evaluate OA treatments' effectiveness and safety by tracking cartilage health. This makes it a valuable tool for predicting OA progression and adjusting treatments accordingly [2].

2.4.2 Cartilage Oligomeric Matrix Protien (COMP664)

In a study by Arrhult et al. (2024), the relationship between exercise, joint health, and OA biomarkers in horses is explored. The findings establish a groundwork for larger-scale investigations into the roles of these biomarkers in diagnosing and monitoring OA and joint health in equine athletes. The study highlights that COMP664 levels in saliva can serve as indicators of joint response to exercise, potentially distinguishing between physiological and pathological responses. These biomarkers hold

promise for early detection and management of OA in horses, paving the way for further research and potential therapeutic interventions [14]

2.5 Nerve Growth Factor, NGF

NGF plays a role in nerve cell development and survival, as well as pain and inflammation. NGF levels have been associated with OA in humans and other animals. The correlation between NGF levels and OA clinical signs, such as lameness and joint inflammation, was assessed. Horses with OA had significantly higher NGF concentrations in their synovial fluid than healthy controls. This suggests that NGF is upregulated in response to joint degeneration and inflammation. Higher NGF concentrations were associated with more severe lameness and a greater inflammatory response. The findings support the potential use of NGF as a biomarker for diagnosing and monitoring OA in horses [11].

2.6 Biglycan, BGN

BGN is a small leucine-rich proteoglycan (SLRP) in the ECM of various tissues, consisting of a protein core with leucine-rich repeats and two chondroitin sulfate/dermatan sulfate chains. It regulates collagen assembly and influences cellular growth, differentiation, and injury response [15]. In bone and cartilage dynamics, biglycan affects osteoblasts and chondrocytes. Elevated levels indicate disorders such as osteoarthritis and osteoporosis, often found in damaged bone and cartilage areas. BGN262 is a specific neo-epitope derived from biglycan, which is a predictive biomarker for early osteoarthritis (OA) changes in subchondral bone, correlating with OA severity in synovial fluid, making it useful for early diagnosis and monitoring OA progression. Combined with COMP156, it provides a comprehensive assessment of joint health and treatment efficacy [16].

2.7 ELISA

From various Swedish veterinary clinics, synovial fluid was collected from horses. Research team developed an inhibition ELISA to quantify biomarker concentrations in samples and to convert raw data into data. An ELISA, or enzyme-linked immunosorbent assay, is a plate-based technique used to detect specific antigens, antibodies, proteins, or hormones in samples. The process involves immobilizing an antigen (or antibody) on a plate, which is then exposed to and attached to an antibody (or antigen). This linked antibody or antigen is attached to an enzyme that, when combined with a substrate, produces a measurable color change. Various forms of ELISA exist depending on the antigen-antibody combination, and the research team developed and used an inhibition/competitive ELISA for their study. This technique is highly sensitive and suitable for high-throughput studies [17]

3
Theory

This chapter provides an introduction to machine learning fundamentals and explains how the performance of the algorithms used in this thesis is evaluated. Additionally, it covers the synthetic data generation.

3.1 Machine Learning

Machine learning has revolutionized industries, including healthcare, finance, and marketing. As a result of its ability to analyze large amounts of data and uncover hidden patterns, it has enabled more accurate predictions and faster decision-making. Automated processes, fraud detection, personalization of customer experiences, and other uses of machine learning are all possible through machine learning models. Computational algorithms enable computers to learn from data and make predictions or decisions based on it. Machine learning is based on these models, which allow systems to improve their tasks performance without being explicitly programmed. In order to understand data, they identify patterns, which they use to build mathematical models [18].

3.2 Neural networks

Neural networks are a type of machine learning model inspired by human brain's structure and functions. They consist of interconnected nodes, or "neurons," that work together to process and analyze data. Neural networks are particularly powerful in tasks such as image and speech recognition, natural language processing, and deep learning. They can learn complex patterns and relationships in data, making them a key component in advancing the capabilities of machine learning [19]. Deep learning can be used to analyze complex biological data and diagnose diseases such as osteoarthritis in horses. This includes images of joints or patterns in synovial fluid biomarkers. This can identify subtle signs of disease that may not be visible to the human eye or detectable by traditional diagnostic methods. Despite these advancements, deep learning is not without its challenges. One of the primary limitations is the dependency on large volumes of labeled data, which can be scarce or expensive in specialized fields. Furthermore, models can inadvertently learn and perpetuate biases present in the training data, leading to skewed or unfair outcomes. Additionally, the "black box" nature of artificial neural networks, where the decision-making process is not transparent, poses challenges for interpretability and trust in AI-driven decisions [20].

3.3 Layers and Neurons

Deep neural networks consist of an input layer, multiple hidden layers, and an output layer. Each layer contains units or neurons, where each neuron in one layer connects to neurons in the next layer. Neurons are the fundamental processing units of a neural network. Each neuron receives inputs from either the dataset or the previous layer's outputs. Every neuron applies weights to its inputs, sums them, and optionally adds a bias term. This sum is then passed through a non-linear activation function, which is crucial for learning and approximating complex functions [19].

As shown in figure 3.1, each node receives an input from some set of nodes, processes it, and sends out an output to another set of nodes. The input layer is the initial point of data entry into the neural network. The neuron receives the raw features of the dataset with one neuron for each feature in the data. For example, in an image recognition task, each neuron might correspond to a different pixel value. This layer does not perform any computation. It serves to pass the feature data to the hidden layers. Hidden layers comprise the bulk of a neural network. These layers are called "hidden" because they do not directly interact with the input or output data but work internally to process the inputs they receive. The different layers are shown in 3.1. The output layer produces the final predictions of the network. For regression tasks, this might be a single neuron that outputs continuous values. In classification tasks, each neuron corresponds to a class label. The network typically outputs either logits or probabilities (after applying a softmax function) indicating the likelihood of each class.

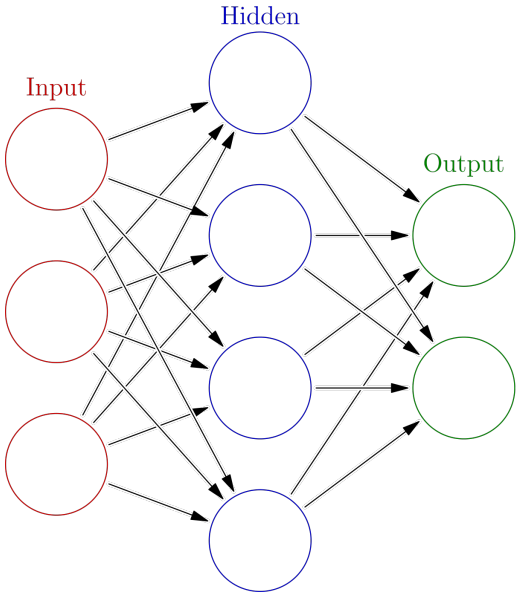


Figure 3.1: Neural network

Connections between neurons in a neural network are denoted by weights, which are crucial adjustable parameters. During training, these weights are tuned to minimize prediction error. Each neuron computes a weighted sum of its inputs, incorporating

a bias before passing the sum through an activation function. This weighted sum can be expressed mathematically as follows:

$$y = \sum_{i=1}^{n} w_i x_i + b$$

where w_i are the weights, x_i are the inputs, b is the bias, and n is the number of inputs to the neuron. Each neuron in these layers performs a weighted sum of its inputs followed by a non-linear transformation through an activation function. This process allows the network to learn complex patterns from the data. The number of hidden layers and the number of neurons in each layer define architecture depth and capacity. More layers and neurons typically allow the network to capture more complex relationships but can also lead to overfitting and increased computational cost [21].

3.4 Backpropagation

Backpropagation is the algorithm used for training neural networks, where the loss function's gradient with respect to each weight is calculated. This process involves two main phases:

Forward Pass: The network computes predictions and the resulting loss, propagating the data forward from inputs to outputs [21].

Backward Pass: The loss function gradient is computed with respect to each weight by applying the chain rule of calculus. This gradient tells us how the weights should be adjusted to reduce the loss [21].

Mathematically, the gradient with respect to a weight w can be expressed as:

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial a} \cdot \frac{\partial a}{\partial w}$$

where L is the loss, and a is the activation that the weight influences [22]. The weight adjustments are typically performed using an optimization algorithm, such as Gradient Descent. In this algorithm weights are updated in a way that minimally decreases the loss, according to:

$$w = w - \eta \frac{\partial L}{\partial w}$$

where η represents the learning rate, a small positive scalar determining the size of the steps taken in the gradient direction during each iteration [22].

The iterative process of forward and backward propagation, with repeated adjustments to the weights, enables the network to gradually improve its predictions. This minimizes the overall loss across training epochs.

3.5 Activation Functions

As the network learns complex patterns in its data stream, activation functions play a pivotal role by introducing non-linearity, which allows the network to operate in a nonlinear way. Without these functions, a neural network, regardless of its depth, could only learn linear relationships. Activation functions allow the network to transition between different layers of the network, which allows it to model more complex relationships. Without these functions, the network would not be able to learn complex relationships, as it would be limited to linear relationships.

3.5.1 Rectified Linear Unit (ReLU)

The Rectified Linear Unit (ReLU) is widely used in deep learning, particularly in hidden layers, due to its ability to introduce non-linearity efficiently. It is defined as:

$$f(x) = \max(0, x)$$

Graphically, ReLU resembles a ramp function, activating a neuron only if the input is positive; otherwise, the output is zero. The mathematical representation is:

$$f(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$$

The derivative of ReLU, important for the backpropagation, is:

$$f'(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ 1 & \text{if } x > 0 \end{cases}$$

Advantages of ReLU include a reduced likelihood of the vanishing gradient problem and the creation of sparse activations. This is where only a subset of neurons activates at any time, enhancing the network's efficiency and effectiveness.

3.5.2 Sigmoid activation function

The sigmoid activation function, often symbolized by σ, is a classical activation function used predominantly in neural networks, particularly for binary classification problems. It is mathematically defined as follows:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

The domain of the sigmoid function is all real numbers ($-\infty < x < \infty$), and its range is between 0 and 1 $(0, 1)$, making the function ideal for probability interpretations [23]. The derivative of the sigmoid function is notable for being expressible in terms of the function itself:

$$\sigma'(x) = \sigma(x) \cdot (1 - \sigma(x))$$

This derivative is essential for the backpropagation in neural networks, facilitating the gradient calculation of the loss function with respect to the weights [23].

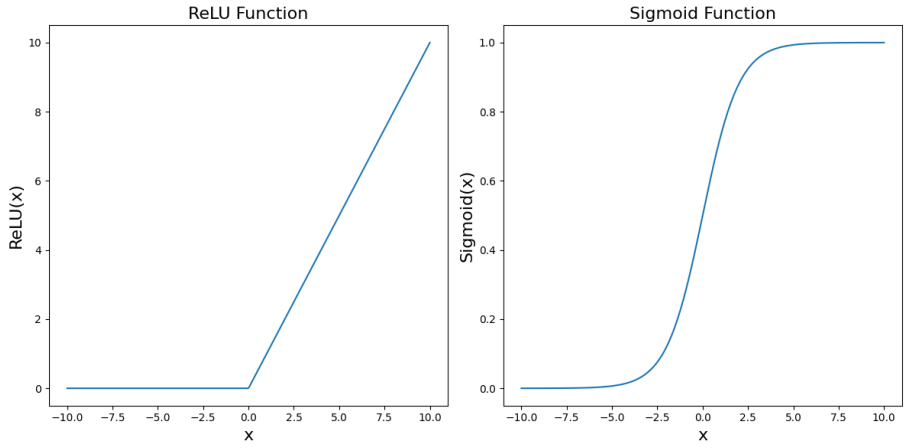


Figure 3.2: a) Relu fucntion b) Sigmoid function

This sequential processing from the input layer to the output layer constitutes forward propagation. It is during this phase that the neural network makes initial predictions based on the current weights and biases. These predictions are subsequently refined through backpropagation, where the model learns from the errors in its predictions and adjusts its weights and biases accordingly.

3.6 Loss Function

Neural networks are quantitatively assessed using a loss function, also known as a cost function. This function measures the discrepancy between the network's predicted outputs and the actual target labels provided during training. The choice of loss function depends on the specific type of learning task—common. In classification tasks within neural networks, the softmax activation function and cross-entropy loss are pivotal for model accuracy. These components are especially crucial in multi-class classification scenarios.

3.6.1 Softmax Activation Function

The softmax function converts logits—the raw predictions from the neural network's final linear layer—into probabilities. It computes the exponentials of each output and normalizes these values by dividing them by the sum of all exponentials. This ensures that the outputs are interpretable as probabilities that sum to one [24].
Mathematically, the softmax function is defined for a class i in a multi-class system as follows:

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

where z_i represents the logit or raw output for class i, and the denominator is the sum of the exponentials of all raw outputs across all classes j [24].

3.6.2 Cross-Entropy Loss

Cross-entropy loss, or log loss, quantifies the difference between predicted probabilities and the actual distribution. If the predicted probabilities are close to the actual labels, then the probability is minimal, and for perfect predictions, the probability is zero. For a classification model with C classes, the cross-entropy loss is defined as:

$$L = -\sum_{i=1}^{C} y_i \log(p_i)$$

where y_i is a binary indicator (0 or 1) if class label i is the correct classification for the observation, and p_i is the predicted probability of the observation belonging to class i [25].

Combination of Softmax and cross-entropy are typically used together in the output layer of neural network models for multi-class classification tasks. The softmax function converts logits to probabilities, and the cross-entropy loss measures the error in these probabilities compared with actual labels. This arrangement is conducive to training the model to minimize errors efficiently by adjusting the model weights through backpropagation.

3.7 Optimization Algorithms

Optimization algorithms play a crucial role in the training of neural networks by determining how to adjust the model's weights based on the gradients computed during backpropagation. The objective is to minimize the loss function iteratively over many epochs—each epoch representing a full pass through the training dataset.

3.7.1 Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD) is one of the most basic yet effective optimization methods. Unlike traditional gradient descent, which uses the entire dataset to update the weights once, SGD updates the weights incrementally for each training example:

$$w = w - \eta \nabla_w L(x_i, y_i, w)$$

where η is the learning rate, w denotes the weights, $\nabla_w L$ is the gradient of the loss function L with respect to the weights, and (x_i, y_i) is a single training example. This method is particularly effective for large datasets, offering faster convergence by frequently updating the weights [26].

3.7.2 Adaptive Moment Estimation (Adam)

Adam is an optimization algorithm that computes adaptive learning rates for each parameter. In addition to storing an exponentially decaying average of past squared gradients like RMSprop, Adam also keeps an exponentially decaying average of past

gradients, similar to momentum:

$$w = w - \frac{\eta}{\sqrt{\hat{v}} + \epsilon}\hat{m}$$

where $\hat{m}$ and $\hat{v}$ are estimates of the first moment (the mean) and the second moment (the uncentered variance) of the gradients, respectively, and ϵ is a small scalar used to prevent division by zero. This method is well-suited for problems with large datasets or parameters with very noisy/stochastic gradients [26].

3.8 Multiclass classification task

Multiclass classification is a type of classification problem where the objective is to assign inputs into one of three or more classes. Unlike binary classification, which deals with two classes, multiclass classification handles problems where each instance belongs to one of multiple categories as shown in figure 3.3. This overview provides a theoretical foundation for understanding multiclass classification models, their metrics, and evaluation techniques.In multiclass classification, the model's objective is to assign an input feature vector to one of several classes. The model learns from a training dataset, which includes examples of input vectors and their corresponding class labels. The training process adjusts the model parameters to minimize the error in predicting the class labels [27].

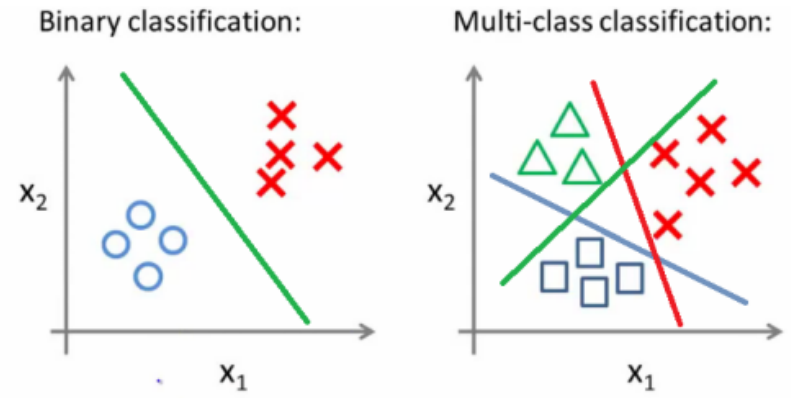


Figure 3.3: Binary classifation vs Multiclassification

Evaluating the performance of multiclass classification models requires different metrics compared to binary classification.

3.8.1 Confusion Matrix

A confusion matrix for multiclass classification is an extension of the binary confusion matrix. It is a square matrix of size $k \times k$, where k is the number of classes. Each entry C_{ij} represents the number of instances of class i predicted as class j [27]. The confusion matrix provides a comprehensive overview of the performance of a classification model, allowing for the identification of errors made by the classifier.

3.8.2 Accuracy

Accuracy is the ratio of correctly predicted instances to the total instances. It is calculated as:

$$\text{Accuracy} = \frac{\sum_{i=1}^{k} C_{ii}}{\sum_{i=1}^{k} \sum_{j=1}^{k} C_{ij}}$$

Accuracy returns an overall measure of how much the model is correctly predicting on the entire set of data. The basic element of the metric are the single individuals in the dataset: each unit has the same weight and they contribute equally to the Accuracy value [27].

3.8.3 Precision

These metrics can be computed for each class and then averaged. Precision for class i is the ratio of true positives to the sum of true positives and false positives:

$$\text{Precision}_i = \frac{C_{ii}}{\sum_{j=1}^{k} C_{ji}}$$

Precision is a measure of the accuracy of the positive predictions. It indicates the proportion of instances predicted as positive that are actually positive. High precision means that the classifier makes few false positive errors [27].

3.9 K-nearest neighbours (KNN)

The k-Nearest Neighbors (KNN) algorithm is a simple, yet powerful machine learning algorithm used for both classification and regression tasks. However, it is most commonly used for classification purposes. The KNN algorithm is a type of instance-based or lazy learning, where the function is only approximated locally and all computation is deferred until classification [28].

The KNN algorithm operates on a very simple principle: it classifies a data point based on how its neighbors are classified. This is achieved by identifying the k nearest neighbors to the point in question, where k is a user-defined constant, and a distance metric measures the closeness of the points.

The choice of distance metrics is crucial to the KNN algorithm's performance. The most commonly used metrics are Euclidean Distance: The standard distance metric for continuous variables, defined as:

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

where x and y are two points in Euclidean n-space, and $d(x, y)$ is the distance between them [29].

3.10 Logistic Regression

Logistic regression is a statistical model commonly used for binary classification tasks, though it can also be adapted for multiclass classification. It operates on the principle of probability prediction. Logistic regression models the probability that a given input belongs to a particular class (e.g., success or failure) based on the input features.

The logistic regression model utilizes the logistic function, also known as the sigmoid function, which is defined as:

$$\sigma(t) = \frac{1}{1 + e^{-t}}$$

where t is a linear combination of the input features, and e is the base of the natural logarithm [30]. The logistic regression equation is:

$$P(Y = 1 \mid X = x) = \sigma(w^T x + b)$$

where $X = x$ are the input features, w is the weight vector, b is the bias or intercept, $w^T x + b$ is the linear combination of inputs and $\sigma(\cdot)$ denotes the sigmoid function[30]. Parameters in logistic regression (weights w and bias b) are typically estimated using maximum likelihood estimation (MLE). The likelihood function L to be maximized is:

$$L(w, b) = \prod_{i=1}^{n} P(Y = y_i \mid X = x_i)^{y_i} (1 - P(Y = y_i \mid X = x_i))^{1-y_i}$$

where n is the number of training samples, y_i is the observed class label, and $P(Y = y_i \mid X = x_i)$ is the predicted probability [30].

While logistic regression is naturally suited for binary classification, it can be extended to multiclass classification using the one-vs-all (OvA) approach. In the OvA approach, a separate binary classifier is trained for each class. Each classifier predicts the probability that a sample belongs to its respective class versus all other classes. For a multiclass problem with K classes, the one-vs-all approach involves training K logistic regression models. Each model k (where $k \in \{1, 2, \ldots, K\}$) is trained to distinguish class k from the rest of the classes. The logistic regression equation for class k is:

$$P(Y = k \mid X = x) = \sigma(w_k^T x + b_k)$$

where w_k and b_k are the weight vector and bias for the classifier corresponding to class k.
During prediction, the class with the highest predicted probability is chosen as the final class label:

$$\hat{Y} = \arg \max_{k \in \{1,2,\ldots,K\}} P(Y = k \mid X = x)$$

This approach allows logistic regression to handle multiclass classification problems effectively by leveraging multiple binary classifiers to make a final decision [31].

3.11 Random Forest Algorithm

Random Forest is an ensemble learning technique that constructs a multitude of decision trees at training time and outputs the class that is the majority vote of the classes (classification) or mean prediction (regression) of the individual trees. Random Forests correct for decision trees' habit of overfitting to their training set. Random Forest builds on the concept of bagging (bootstrap aggregating) and the decision tree algorithms. By building multiple trees and merging their outputs, Random Forest mitigates the overfitting problem typical of decision trees and boosts the overall accuracy [32]. Random Forest starts by creating multiple bootstrap samples from the original dataset. This means that each bootstrap sample is created by randomly selecting observations with replacement. For each bootstrap sample, a decision tree is grown. When splitting a node during the construction of the tree, a random subset of the features is chosen as split candidates from the full set of features. This randomness helps in making the ensemble model more robust than a single decision tree. After many trees are built, Random Forest aggregates their predictions. For classification tasks, this aggregation is typically the mode of the classes predicted by individual trees (majority voting). For regression tasks, it is typically the average of the predictions [32].

The effectiveness of Random Forests often requires an understanding of the statistics used in aggregating predictions:

In classification, the prediction of the ensemble is given by:

$$\hat{y} = \text{mode}\{y_1, y_2, \ldots, y_n\}$$

where y_i is the prediction of the i-th decision tree. In regression, the prediction is the average across all individual trees' predictions:

$$\hat{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$$

where y_i is the prediction from the i-th tree and n is the number of trees [32].

3.12 Data distribution

In statistical analysis, fitting distributions to data is a fundamental step to understand the underlying probabilistic characteristics of the data. This process involves estimating the parameters of a theoretical distribution that best matches the observed data. The choice of distribution depends on the nature of the data and the specific analysis goals. We focus on three widely used distributions: the normal distribution, the log-normal distribution, and the gamma distribution.

By fitting these distributions to data, analysts can perform a range of tasks from probabilistic assessments to predictive modeling. This process not only aids in understanding the data's behavior but also in applying statistical methods to derive insights and make decisions based on the modeled distributions.

3.12.1 Normal Distribution

The normal (or Gaussian) distribution is characterized by its bell-shaped curve and is defined by two parameters: the mean (μ) and the standard deviation (σ). The probability density function (PDF) of a normal distribution is given by:

$$f(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

The mean (μ) represents the central tendency of the data, while the standard deviation (σ) quantifies the dispersion around the mean [33].

3.12.2 Log-normal Distribution

A log-normal distribution is applicable to data that, after a logarithmic transformation, follows a normal distribution. This distribution is particularly useful for data that cannot take negative values. The PDF of a log-normal distribution is:

$$f(x|\mu,\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln x-\mu)^2}{2\sigma^2}}$$

for $x > 0$, and where μ and σ are the mean and standard deviation of the distribution's logarithm [33].

3.12.3 Gamma Distribution

The gamma distribution is another choice for modeling positively skewed data. It is defined by a shape parameter (α), a scale parameter (θ), and optionally, a location parameter. The PDF is given by:

$$f(x|\alpha,\theta) = \frac{x^{\alpha-1}e^{-x/\theta}}{\theta^{\alpha}\Gamma(\alpha)}$$

for $x > 0$. The fitting function estimates α, θ, and fixes the location parameter at 0, making it suitable for data that begin from a minimum of 0. The shape parameter influences the distribution's general form, while the scale parameter adjusts its width [33].

3.13 Evaluate Distributions

The selection of the most appropriate probability distribution to model empirical data is a critical step in statistical analysis. This process involves comparing the empirical cumulative distribution function (ECDF) of the observed data against the cumulative distribution functions (CDFs) of candidate theoretical distributions. One of the most effective methods for this comparison is the Kolmogorov-Smirnov (K-S) test, which provides a quantitative measure of the fit between empirical and theoretical distributions.

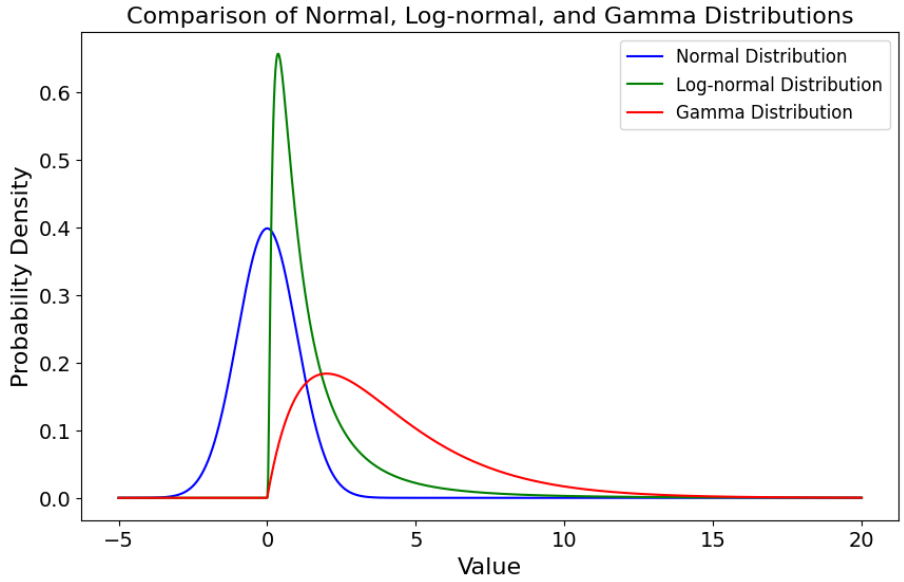


Figure 3.4: Comparison of Normal, Log-normal and Gamma Distributions

3.13.1 Kolmogorov-Smirnov Test

The Kolmogorov-Smirnov test, as detailed by Massey (1951) and further discussed in numerical recipes by Press et al. (1988), is a nonparametric test that quantifies the discrepancy between the ECDF of an empirical dataset and the CDF of a theoretical distribution. The K-S statistic, D, is defined as the maximum absolute difference between these two functions over the range of data:

$$D = \max_x |F_{\text{emp}}(x) - F_{\text{theo}}(x)|$$

where $F_{\text{emp}}(x)$ denotes the ECDF of the empirical data, and $F_{\text{theo}}(x)$ denotes the CDF of the theoretical distribution.

3.13.2 Implementation and Evaluation

The computational implementation of this method involves visualizing both the empirical and theoretical CDFs to qualitatively assess the fit. Quantitatively, the K-S statistic serves as a clear criterion for selecting the distribution that best represents the underlying data structure. Incorporating the K-S test into distribution fitting provides a robust approach to model selection, ensuring that the chosen model accurately reflects the distributional properties of the empirical data. In generating synthetic data, we employ uniform random sampling within specified bounds, determined by the minimum and maximum values observed for each principal variable in our dataset. This method relies on the uniform distribution, which is fundamental for synthesizing data that adheres to the observed range constraints.

4

Methods

This study is based on a dataset of synovial fluid samples collected from horses diagnosed with osteoarthritis at various stages of their condition and healthy horses. A total of four categories of samples were created on the basis of these observations: Healthy, Mild Osteoarthritis, Moderate Osteoarthritis, and Severe Osteoarthritis. The amount of the different classes in the original data is shown in table 4.1. Veterinarian experts evaluated the clinical data and classified the animals based on diagnostic imaging results. By doing so, it was possible to guarantee the relevance and specificity of the data with respect to the diagnostic markers of osteoarthritis.

In this study, four specific proteins were analyzed as biomarkers: $COMP664$, $BGN262$, $COMP156$, and NGF. As a result of their roles in cartilage synthesis and degradation, these proteins were selected based on the literature that identifies them as potential indicators of Osteoarthritis, due to their roles in cartilage synthesis and degradation.

Table 4.1: Distribution of Diagnosis Categories: 0 represents healthy horses, 1 indicates horses with mild OA, 2 denotes horses with moderate OA, and 3 signifies horses diagnosed with severe OA.

Diagnosis	0	1	2	3
Frequency	68	64	33	33

4.1 Statistical Analysis of the Dataset

The generation of synthetic data was preceded by a comprehensive statistical analysis aimed at fully understanding the inherent properties of the dataset. This was before synthetic data development could start. In order to ensure the validity and relevance of our research findings, it was important to conduct this analysis. This was to verify that the synthetic data closely mirrored the real dataset's statistical behavior. The data was analyzed to identify key patterns, outliers, and correlations. The synthetic data was also checked for statistical significance to ensure accuracy and reliability.

4.1.1 Distribution Analysis

The initial phase of our statistical analysis involved the computation of descriptive statistics for each key variable: $COMP664$, $BGN262$, $COMP156$, and NGF. This included measures of mean, median, standard deviation and variance. A baseline was established for understanding the dataset's characteristics based on these metrics. This provided insights into the distribution, spread, and shape of our data. A combination of visual aids such as histograms and line charts, along with assessments of statistical normality, was used to explore how well the data corresponded to various theoretical distributions. Examining distribution models in the generation of synthetic data was essential for identifying the most appropriate distribution models to apply to the generation of synthetic data.

Based on comprehensive assessments comparing various theoretical distributions including Normal, Log-Normal, and Gamma distributions across different OA diagnosis classes, it was observed that the log-normal distribution provided the best fit for each variable at different OA stages. Appendix 1 visually demonstrates these results, highlighting the normal distribution's superiority. Consequently, the log-normal distribution was chosen for synthetic data. This decision ensures that synthetic data preserves the statistical characteristics of the original dataset while enhancing future analyses' robustness and reliability.

4.1.2 Application of Findings to Synthetic Data Generation

Statistical analysis directly influenced how we generated synthetic data. The statistical parameters enabled us to ensure that synthetic data accurately replicated the original dataset's central tendencies and variability. In addition, it maintained its correlation structure and distribution features. Replicating the original dataset's central tendencies and variability is crucial for ensuring that the synthetic data accurately represents the original data characteristics.

4.2 Generation of Synthetic Data

The flowchart presented in figure 4.1 outlines the systematic approach taken to generate synthetic data from the original dataset. This method ensures that the synthetic data retains the statistical properties of the original dataset, enabling robust analysis and machine learning applications. In order to create a synthetic dataset that reproduces the statistical properties of our original dataset, an approach that leverages Python's Pandas and NumPy libraries was used to achieve this. By using this method, it ensured that the synthetic data contained the underlying distributions and relationships between variables observed in the original data.

The dataset was loaded from an data file. The columns `COMP664`, `BGN262`, `NGF`, and `COMP156` were converted to numeric values, replacing commas with dots to handle decimal values properly. The `Diagnosis` column was also converted to numeric values to ensure consistency. The `Diagnosis` contains the different classes, Healthy,

Mild Osteoarthritis, Moderate Osteoarthritis, and Severe Osteoarthritis.

For each unique `Diagnosis` class in the `Diagnosis` column, the data was filtered based on the current `Diagnosis` class and non-negative values for each selected column. Synthetic data was generated using a log norm and a normal distribution with the mean and standard deviation calculated from the filtered data. If no data was available for a column and duration class, an array of `NaN` values was generated.

To better understand the distribution of the data, it was grouped into histograms. This process involved categorizing the data into bins, which allowed for a visual representation of the frequency distribution for each `Diagnosis` and duration class. By grouping the data in this manner, we can more easily observe patterns and trends within the dataset.

The synthetic data for each `Diagnosis` class was stored in a temporary dictionary and then converted to a DataFrame. All DataFrames containing the synthetic data for each duration class were concatenated into a single DataFrame. This combined DataFrame was saved as a CSV file.The dataset was generated in two sizes, one with 80000 data rows and the other with 8000 data rows.

Log-normal distritubtion

Once the filtered data is prepared, the code applies a logarithmic transformation, taking the natural logarithm of each data point. This transformation helps normalize skewed data, making it more suitable for log-normal distributed data. Next, the code calculates the mean (mu) and standard deviation (sigma) of the log-transformed data, representing the average value and the variation, respectively. Using these parameters, synthetic data points are generated from a log-normal distribution, ensuring a large dataset for analysis.

Normal distribution

After the code filters the dataset to isolate data points relevant to a specific diagnosis class and column. This ensures that only valid, non-missing data is used. For each column in the filtered dataset, the mean (mu) and standard deviation (sigma) of the data points are calculated. The mean represents the average value of the data, while the standard deviation measures the amount of variation or dispersion from the mean. Using these parameters, synthetic data points are generated from a normal distribution. Here, the mean of the distribution, scale sets the standard deviation, and size specifies the number of synthetic data points generated. This method ensures that the generated synthetic data mirrors the statistical properties of the original data. This provides a large dataset for robust analysis and machine learning applications.

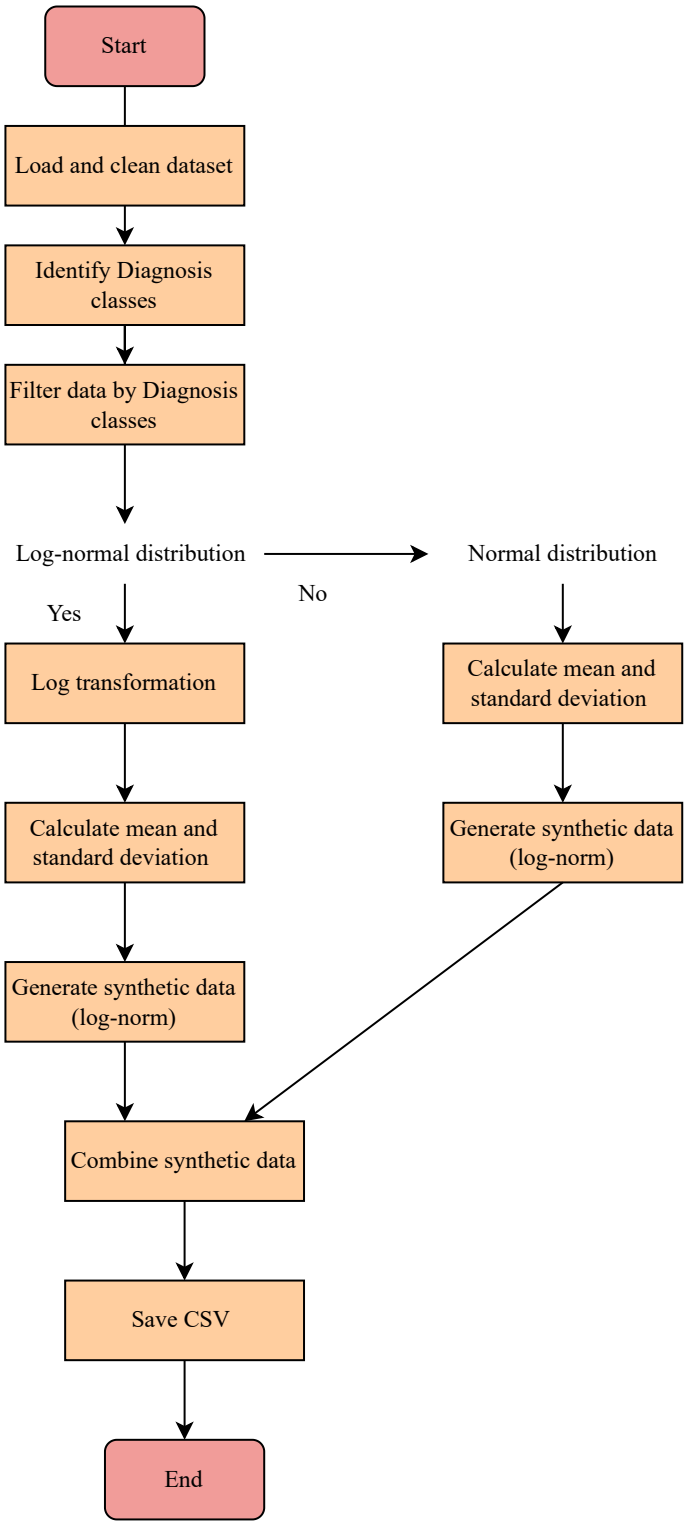


Figure 4.1: Flowchart illustrating the process of generating synthetic data using log-normal and normal distributions.

4.2.1 Application in Distribution Fitting

The procedure for employing the Kolmogorov-Smirnov (K-S) test in distribution fitting begins with calculating the empirical cumulative distribution function (ECDF), $F_{\text{emp}}(x)$, from the observed data. This step function increases by $1/n$ at each data point, where n is the total number of observations. Next, the theoretical cumulative distribution functions (CDFs), $F_{\text{theo}}(x)$, are computed by fitting the parameters of candidate theoretical distributions (e.g., normal, log-normal, and gamma) to empirical data. The K-S statistic, D, is then calculated for each candidate distribution, measuring the maximum distance between $F_{\text{emp}}(x)$ and $F_{\text{theo}}(x)$. The distribution with the smallest K-S statistic is considered the best fit, as it has the minimum deviation across the range of values. The K-S statistic quantifies the distance between the empirical distribution function of the sample data and the cumulative distribution function of the reference distribution. A lower K-S statistic indicates that the CDF of the synthetic data is closer to the CDF of the actual data, implying a better fit of the distribution to the data.

4.3 Neural network

The full process of the Deep Neural Network that was used in this thesis had this overall setup seen in figure 4.2. This diagram provides a comprehensive overview of the entire process, from data preprocessing to the final evaluation of the model.

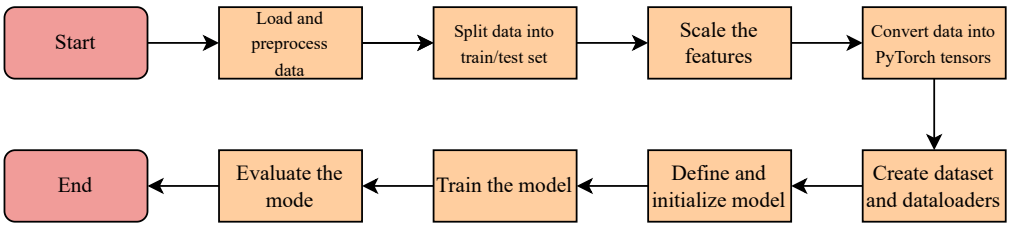


Figure 4.2: Flowchart over the full deep machine learning process

The process begins with loading the data file. Once the data is loaded, rows with missing values are dropped to ensure the dataset is clean. The next step involves separating the dataset into features (X) and target (y). The target variable (y) is then encoded to prepare it for machine learning algorithms. The preprocessing steps included:

- **Scaling:** Feature values were standardized using the `StandardScaler` to normalize the data, ensuring that the model's input features have a mean of zero and a standard deviation of one.
- **Encoding:** The target variable ('Diagnosis' indicating the stage of Osteoarthritis) was encoded into discrete class indices using a `LabelEncoder` to facilitate model training and evaluation.

After encoding, the dataset is split into training and test sets to facilitate model training and evaluation. The features are scaled to standardize the data, which improves the model's performance and convergence. The scaled data is then converted into PyTorch tensors, which are the primary data structure used in PyTorch.

With the data prepared, PyTorch datasets and dataloaders are created to handle batch processing during training and testing. The PyTorch model is then defined, specifying the neural network architecture. Following this, the model is initialized along with the loss function, optimizer, and learning rate scheduler.

Training the model involves multiple epochs. For each epoch, the training process includes several steps: zeroing the gradients to prevent accumulation, performing a forward pass to compute the model outputs, computing the loss, performing a backward pass to compute gradients, updating the model parameters using the optimizer, and finally, computing the training accuracy and loss.

The neural network model used in this study is a multi-class classification model designed to predict Osteoarthritis stages based on four selected biomarkers: NGF, BGN262, COMP664, and COMP156. The model architecture consists of several layers. It starts with an input layer that accepts the four biomarker values. The first hidden layer contains 128 neurons and incorporates batch normalization to stabilize and accelerate training. This is followed by the ReLU activation function to introduce non-linearity into the model.

The second hidden layer consists of 64 neurons and includes batch normalization and ReLU activation. The third hidden layer has 32 neurons, again using the ReLU activation function to enhance the model's ability to capture complex patterns in the data. The output layer is a fully connected layer with a number of neurons equal to the number of Osteoarthritis stages (classes), which in this case, are encoded into numerical values.

The output from the final layer provides raw logits, which are used by the CrossEntropyLoss function during training and evaluation. The model is trained using the Adam optimizer with a learning rate of 0.001. The learning rate is adjusted during training by a step learning rate scheduler. This architecture is designed to effectively classify the stages of osteoarthritis by processing and learning from the patterns in the biomarker data. This is done through multiple layers of transformation and non-linear activation functions.

After training at each epoch, the model is evaluated on the test set. This involves performing a forward pass to compute the model outputs, computing the loss, and then computing the test accuracy and loss. Upon completing all epochs, a final evaluation is conducted to calculate the final accuracy of the PyTorch model. The process concludes with reporting the final model accuracy.

4.4 Model Evaluation

Model performance was evaluated on a separate test set not used during the training phase. The primary metric for evaluation was accuracy, calculated as the percentage of correct predictions. Additionally, the loss on the test set provided insight into the model's generalization capabilities.

To assess the statistical significance of the findings, results from the model were compared using standard statistical methods suited to classification tasks, such as confusion matrices. These analyses helped determine the model's ability to distinguish between Osteoarthritis stages.

4.5 Machine Learning

The flowchart detailing the process of training and comparing multiple models is illustrated in Figure 4.3. The next steps involve training four different models: a deep neural network (DNN), k-nearest neighbors (KNN), random forest, and linear regression. Each model undergoes a training phase where it learns from the training data. After training, each model's performance is evaluated on the testing set, and accuracy metrics are recorded. Finally, the accuracy of each model is compared to determine which model performs best on the given dataset. This comparison helps in selecting the most suitable model for the classification task. The flowchart thus encapsulates the entire process from data preprocessing to model training, evaluation, and comparison.

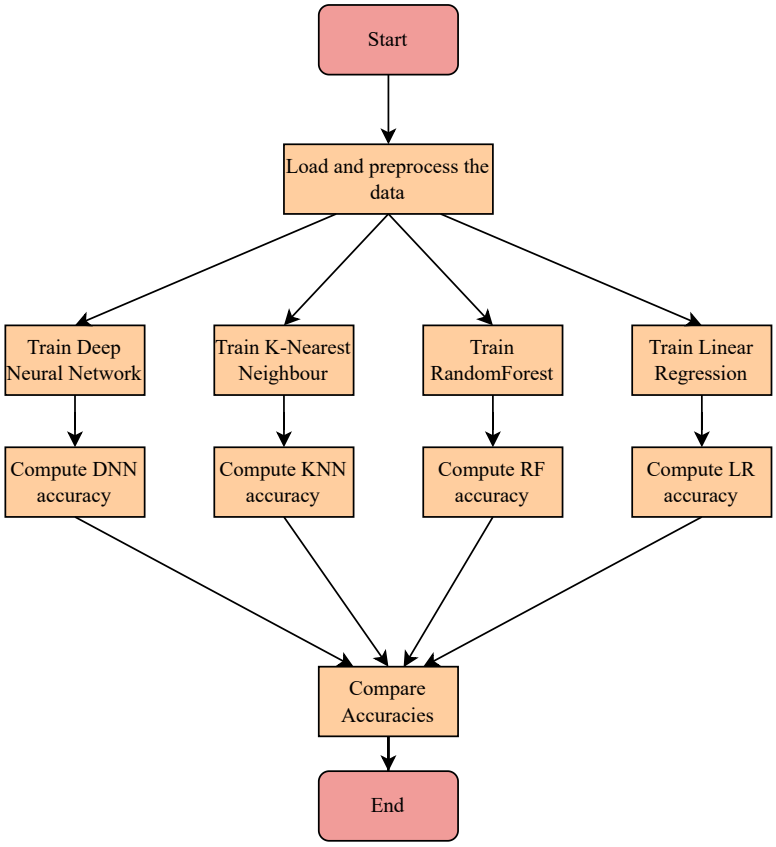


Figure 4.3: Flowchart illustrating the process of training and comparing multiple models: deep neural network (DNN), k-nearest neighbors (KNN), random forest, and logistic regression.

This comparison aims to underscore the deep neural network's capabilities in handling the intricacies of the dataset, potentially outperforming traditional algorithms in specific scenarios pertinent to our research objectives.

4.6 Web Application

To facilitate the practical application of the predictive model in veterinary medicine, a web application was developed. This application is specifically designed to predict different classes of OA, enhancing decision-making processes in clinical settings.

The web app, built using modern web technologies, offers a user-friendly interface that allows veterinary professionals to input patient data and receive immediate predictive insights regarding OA. The application is designed to run locally on a computer, ensuring data privacy and speed by eliminating external servers dependency.

Key features of the app include data input forms tailored to veterinary use, interactive elements for displaying predictive results, and a backend designed for rapid calculations. This setup not only makes it accessible to non-technical users but also ensures its seamless integration into existing veterinary practice workflows.

By providing a tool that simplifies the prediction of OA classes, the web application stands to significantly improve the efficiency and accuracy of OA diagnostics in veterinary medicine.

5
Results

This chapter will provide a presentation and visualization of the results gained from the implemented machine learning models.

5.1 Synthetic Dataset

The analysis in Appendix A indicates that the log-normal and normal distributions provided the best fit for the data. Consequently, the gamma distribution was discarded. By employing the log-normal and normal distribution models, we assessed their performance in producing synthetic datasets that closely resembled the distributional patterns of the original data. The distributions for each biomarker can be found in Appendix B.

5.1.1 Log-norm Distribution

The utilization of the log-normal distribution model resulted in synthetic datasets that closely mirrored the distributional characteristics of the original data. As shown in figure 5.1, the synthetic data generated using the log-normal distribution aligns well with the original data, preserving key statistical properties such as the mean, variance, and skewness. The log-normal distribution model was particularly effective in handling the biomarkers with high variability and non-normality, which are common in biological datasets.

5.1.2 Norm distribution

In contrast, the utilization of the normal distribution model resulted in synthetic datasets that did not mirror the distributional characteristics of the original data as effectively as the log-normal distribution. As shown in figure 5.2, the synthetic data generated from the normal distribution exhibited less alignment with the original data, particularly in cases where the original data was skewed. The normal distribution assumes symmetry and constant variance, which may not be applicable for all biomarkers in the dataset.The synthetic datasets generated using the normal distribution were less effective in capturing the tails of the distribution, leading to a loss of critical information about the variability and extremes present in the original data. This limitation can impact the reliability of the synthetic data for predictive modeling and other analyses that rely on accurate representation of the data's statistical properties.

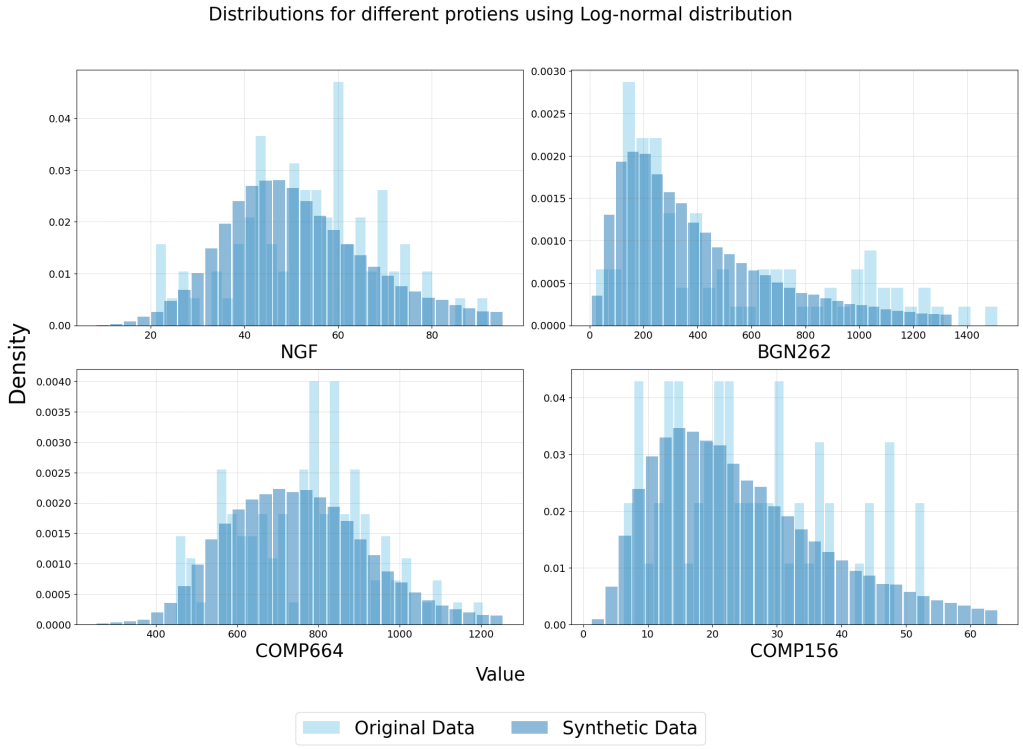


Figure 5.1: Comparison between original data and synthetic data from the log-normal distribution

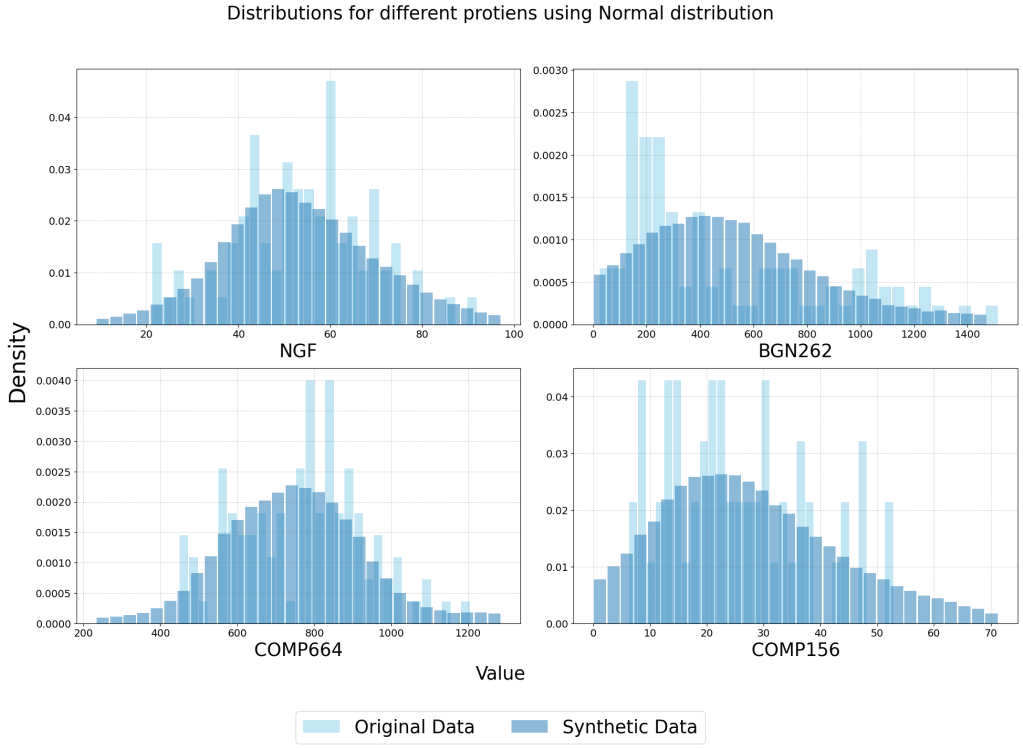


Figure 5.2: Comparison between original data and synthetic data from the normal distribution

5.2 Machine learning

This section analyzes the performance of various machine learning models trained on both log-normal and normal distribution datasets. The models were evaluated based on their training and test accuracy, which provide insights into how well the models can generalize to new, unseen data.

5.2.1 Accuracy and loss

The dataset that was synthetic data with two different sizes. The synthetic data generation process yielded valuable insights into the efficacy of different statistical methodologies in capturing the essential characteristics of the original dataset. The following table illustrates the training and test accuracy for models trained on both log-norm and norm distributions. Notably, while the training accuracy is reasonably high, the test accuracy, particularly for the log-norm dataset, is significantly lower. In contrast, the norm dataset shows a more consistent performance between training and test phases, suggesting better model stability.

Table 5.1: Training and Test Accuracy of Various Models Using Log-norm and Norm Datasets with Different Dataset Sizes (A = 80000 data points, B = 8000 data points). All models were trained for 50 epochs

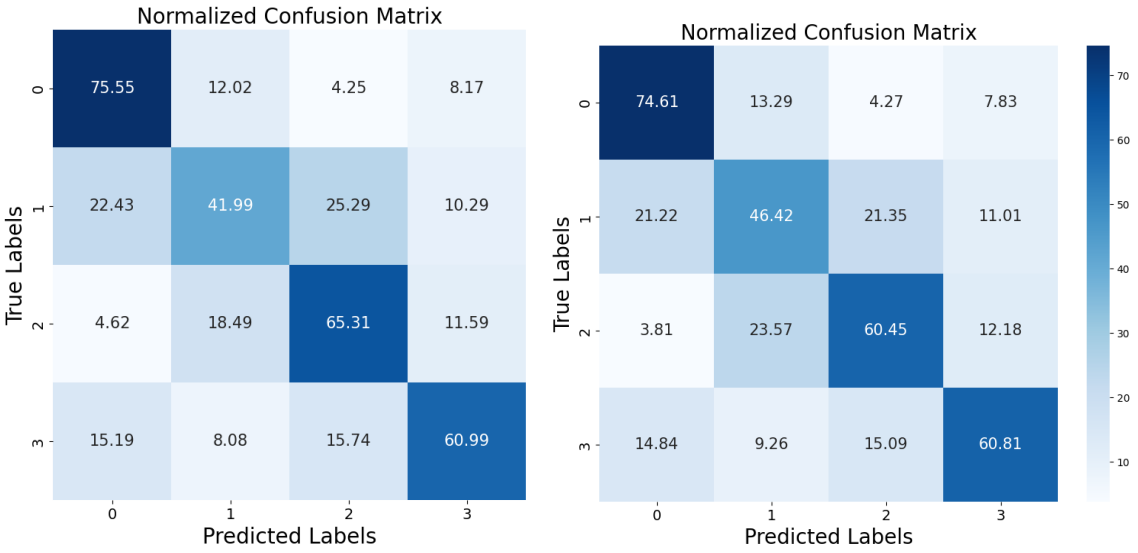
Model	Dataset Size	Training Accuracy	Test Accuracy
Log-norm dataset	A	0.6027	0.6111
Log-norm dataset	B	0.6006	0.5944
Norm dataset	A	0.7084	0.7093
Norm dataset	B	0.7055	0.6737

5.2.2 Log-norm distribution

For the log-norm distribution dataset two normalized confusion matrixes was made. One trained with 50 epoch and one with 20. Table 5.2 outlines the test accuracy's achieved by each model on the log-norm distribution dataset. Notably, the Deep Neural Network (DNN) model exhibited the highest accuracy. These results underscore the efficacy of these models in accurately predicting target classes using synthetic data from the log-normal distribution. Next, let's explore the results obtained after 20 epochs of training. Notably, the Multiclass Classification (DNN) model performs similar to 50 epochs.
Figure 5.3a illustrates the normalized confusion matrix derived after 50 epochs of training. This matrix offers a comprehensive view of the model's accuracy across different classes, shedding light on both correct and incorrect predictions. Furthermore, Figure 5.3b portrays the normalized confusion matrix resulting from 20 epochs of training.

Table 5.2: Test Accuracy of Various Models for Data points on Log-normal distribution dataset

Model	Test Accuracy	
	Epoch 50	**Epoch 20**
Multiclass Classification (DNN)	0.6111	0.6091
K-nearest neighbour (KNN)	0.5525	0.5525
Random forest	0.5851	0.5851
Logistic Regression	0.5428	0.6032



(a) Epoch 50 **(b)** Epoch 20

Figure 5.3: Normalized confusion matrix after training the model for 50 and 20 epochs on data generated from a log-normal distribution.

5.2.3 Norm distribution

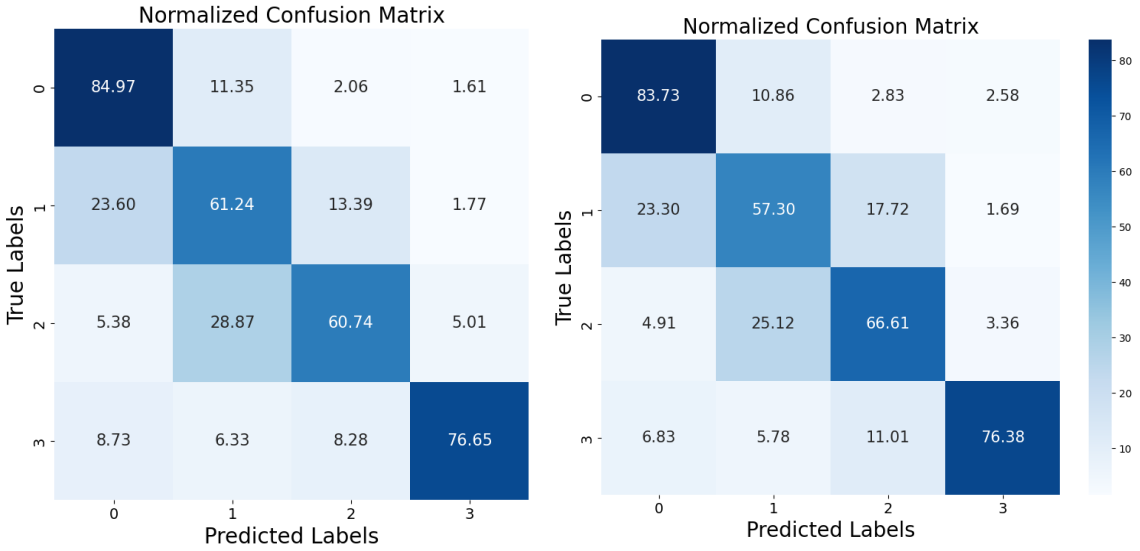
For the dataset generated from the normal distribution, confusion matrices were also generated for different numbers of training epochs.

Table 5.3: Test Accuracy of Various Models for Data points on the Normal distributed dataset

Model	Test Accuracy	
	Epoch 50	**Epoch 20**
Multiclass Classification (DNN)	0.7093	0.7108
K-nearest neighbour (KNN)	0.6663	0.6663
Random forest	0.6887	0.6887
Logistic regression	0.5428	0.5428

Figure 5.4a illustrates the normalized confusion matrix obtained after training the

model for 50 epochs. This matrix provides insights into the model's performance when trained on data generated from a normal distribution. Similarly, Figure 5.4b displays the normalized confusion matrix resulting from training the model for 20 epochs.



(a) Epoch 50 **(b)** Epoch 20

Figure 5.4: Normalized confusion matrix after training the model for 50 and 20 epochs on data generated from a normal distribution.

5.3 Analysis of each biomarker

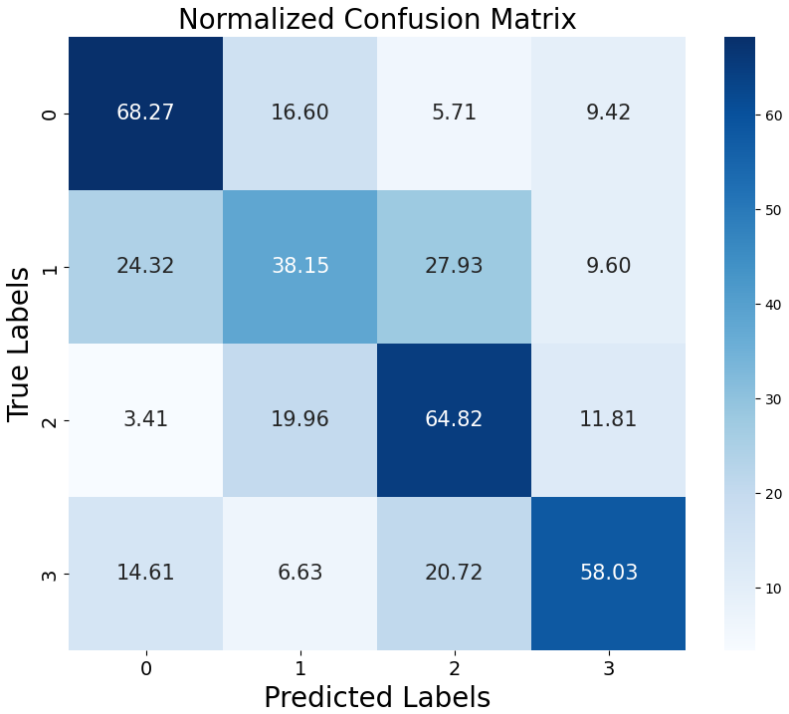
This section delineates the results of a systematic investigation into the impact of individual biomarkers on the predictive performance of our multi-classification models and other machine learning models. Each biomarker was sequentially omitted from the analysis, allowing for a detailed assessment of its influence on model accuracy. The dataset used for this part is the log-norm dataset. Due to little change in accuracy for the different sizes of datasets, this part used the bigger dataset for its analysis.

5.3.1 NGF

The exclusion of the Nerve Growth Factor (NGF) biomarker decreased the models' accuracy across all algorithms, as shown in the table 5.4a and figure 5.4b .

Model	Test Accuracy
Multiclass Classification (DNN)	0.5748
K-nearest neighbour (KNN)	0.5139
Random forest	0.5388
Logistic Regression	0.5329

(a) Test Accuracy for all models when NGF was removed



(b) Confusion matrix for the DNN when NGF was removed

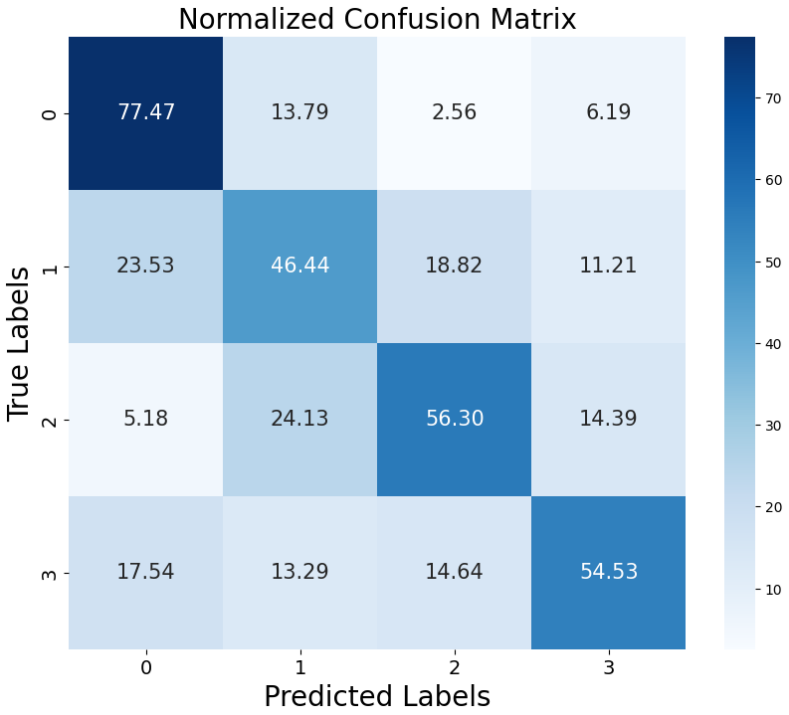
Table 5.4: Confusion matrix for the DNN when NGF was removed

5.3.2 BGN

The removal of the BGN biomarker resulted in a decrease in accuracy for all models as well. This biomarker appears integral to capturing the underlying pathology of Osteoarthritis, potentially due to its direct involvement in cartilage degradation processes.

Model	Test Accuracy
Multiclass Classification (DNN)	0.5877
K-nearest neighbour (KNN)	0.5284
Random forest	0.5483
Logistic Regression	0.5189

(a) Test Accuracy for all models when BGN was removed



(b) Confusion matrix for the DNN when BGN was removed

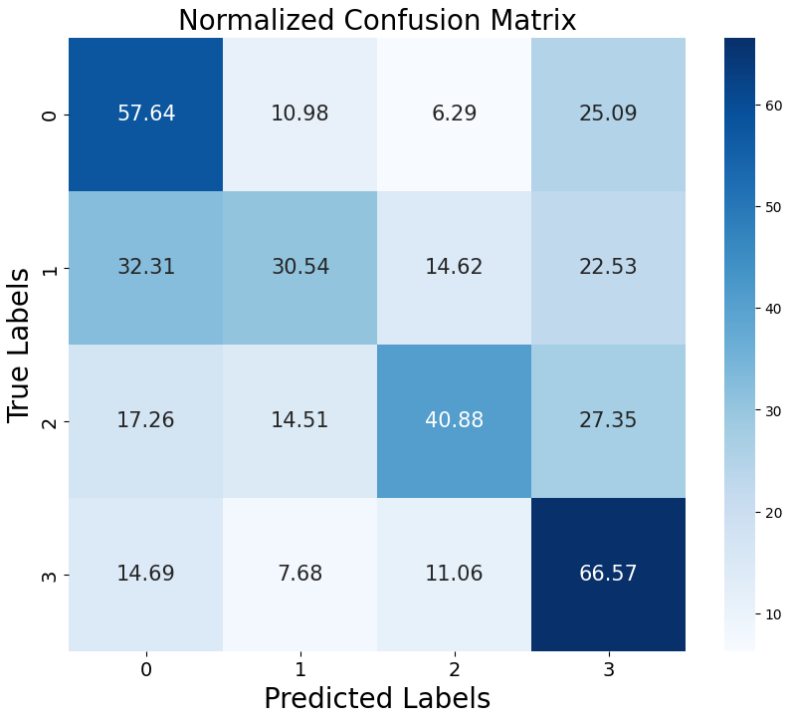
Table 5.5: Confusion matrix for the DNN when BGN was removed

5.3.3 COMP664

The removal of the COMP664 biomarker demonstrated a significant reduction in model accuracy, reinforcing its pivotal role in diagnosing osteoarthritis (OA).

Model	Test Accuracy
Multiclass Classification (DNN)	0.4899
K-nearest neighbour (KNN)	0.4283
Random forest	0.4521
Logistic regression	0.4349

(a) Test Accuracy for all models when COMP664 was removed



(b) Confusion matrix for the DNN when COMP664 was removed

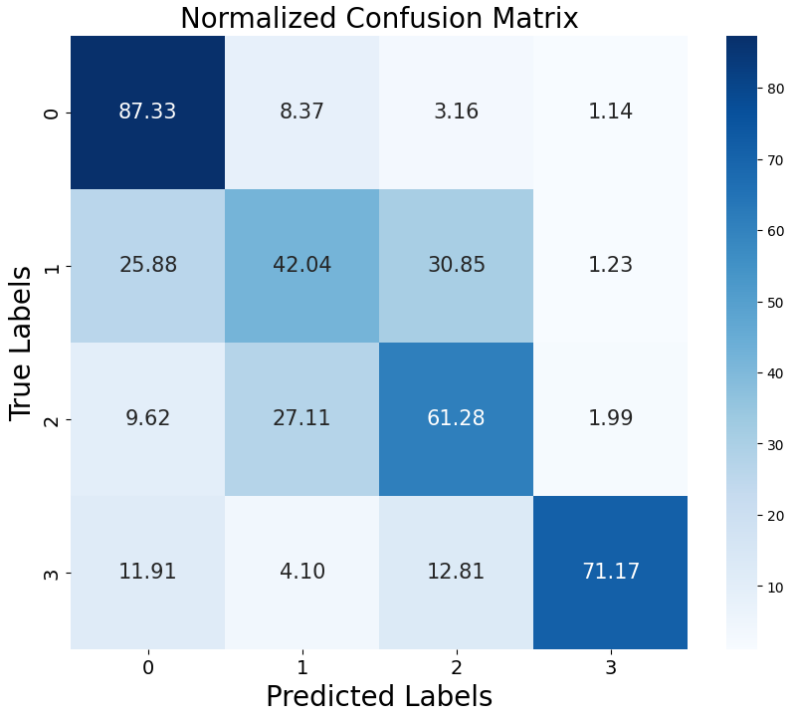
Table 5.6: Confusion matrix for the DNN when COMP664 was removed

5.3.4 COMP156

Similarly, the removal of COMP156 resulted in diminished test accuracies, highlighting its critical role in the predictive framework.

Model	Test Accuracy
Multiclass Classification (DNN)	0.5720
K-nearest neighbour (KNN)	0.5130
Random forest	0.5412
Logistic Regression	0.5148

(a) Test Accuracy for all models when COMP156 was removed



(b) Confusion matrix for the DNN when COMP156 was removed

Table 5.7: Confusion matrix for the DNN when COMP156 was removed

5.4 Webb Application

The application is straightforward and serves as an excellent tool for veterinarians, especially when the model demonstrates high accuracy. The simplicity of the interface ensures ease of use, making it accessible even for those with limited technical expertise. Additionally, the application can be enhanced by incorporating more diagrams and other useful features to further streamline the user experience and provide valuable insights.

To provide readers with a more comprehensive understanding of the application's interface and functionality, a screenshot showcasing its layout and features is included below 5.5. This visual representation highlights the user-friendly design and the various tools available within the application, illustrating how it can be seamlessly integrated into daily veterinary practice.

By adopting this innovative technology, veterinarians can leverage the power of machine learning to enhance their diagnostic capabilities, ultimately leading to better outcomes for the animals under their care. This development is a testament to our commitment to advancing veterinary medicine through cutting-edge technology and practical solutions.

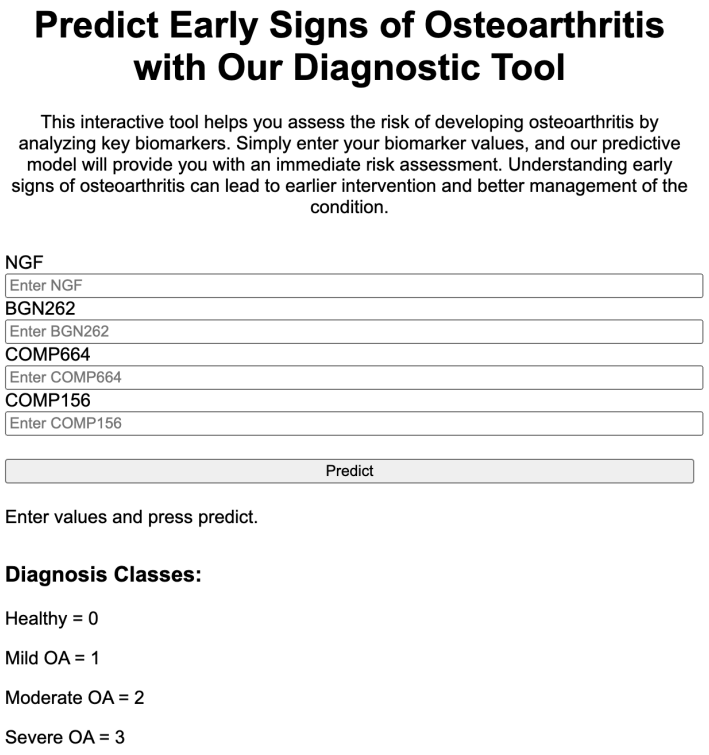


Figure 5.5: Screenshot of the webbapplication

6
Conclusion

6.1 Dataset

Both synthetic datasets showed a significant difference in accuracy across all models. As shown in Table 5.1, the normal distribution dataset exhibited better accuracy compared to the log-normal dataset. This result is surprising, given that Figures 5.1 and 5.2 both indicate that the log-normal dataset fits the original data better than the normal distribution.

One possible explanation for the discrepancy between accuracy and data fit could be the presence of outliers in the log-normal dataset. Outliers can have a substantial impact on model performance, potentially leading to lower accuracy despite a better fit to the overall data distribution. These outliers may skew the results, causing the models to perform less effectively even though the general data pattern aligns more closely with the original dataset.

6.2 Machine Learning

Table 5.1 illustrates the impact of dataset size on model performance. In this case, the dataset containing 80,000 data rows performs slightly better than the dataset with 8,000 data rows. This result suggests that larger datasets are more effective for this type of analysis.

The improved performance of the larger dataset underscores the importance of dataset size in predictive modeling. Larger datasets capture a more comprehensive range of variability and patterns within the data, leading to more accurate and reliable models. This finding highlights the necessity of using extensive datasets to achieve robust and valid data analysis results.

6.2.1 Log-norm distribution

This approach proved effective in capturing the skewed nature of the biomarker distributions present in the dataset. By leveraging the log-normal distribution's ability to model positively skewed data, we achieved synthetic datasets that exhibited similar patterns of variability and concentration around central tendencies as observed in the original dataset.

The log-normal distribution also demonstrated robustness in capturing the tails of the distributions, ensuring that extreme values were adequately represented in the synthetic datasets. This fidelity to the original data distribution is essential for maintaining the integrity of subsequent analyses and modeling tasks, as it preserves the underlying statistical properties inherent in the dataset.

In the lognorm dataset, the test accuracy for the multiclassification model was 0.6, while the test accuracy for the other machine learning models was lower. Considering that we want the accuracy to be as close as possible to 1, that is a low accuracy. Due to our four classes, if the model made a random prediction we would have a 0.25 accuracy score, so the 0.6 is a good score.

The multiclassification model has a higher accuracy than the other machine learning model, but the difference is not that great. Perhaps the model does not capture the complexity of the dataset because of the data structure. Figure 5.1 shows that when we train the model for more epochs, the model seems to prefer class 2 over class 3. There is a possibility that this could be the result of the fact that class 2 and 3 have a large overlap when it comes to the concentrations of the different biomarkers. Another observation from figure 5.3a is that the model tends to misclassify the closest classes most frequently, which is actually beneficial. This is because we prefer the model to avoid misclassifying a healthy horse as severely ill, or vice versa.

6.2.2 Normal distribution

In spite of the fact that the normal distribution is a widely used statistical model in statistical analysis, it is inherently limited in its ability to capture skewed or non-normally distributed data. The synthetic datasets generated using the normal distribution exhibited deviations from the original data distribution, particularly in the tails. Synthetic datasets may not adequately represent the variability and patterns present in the original datasets, leading to biases or inaccuracies in downstream analyses and modeling efforts. In the normal distribution dataset, we can see that the accuracy for the multiclassification model is higher than the log-norm, but also that the accuracy for the other models is higher than the lognorm as well. Observing the confusion matrix, we can see that the model seems to be making a more accurate prediction for the healthy or severe categories regardless of whether it was trained for 50 or 20 epochs.

6.3 Analysis of each biomarker

If the accuracy of a model decreases after the removal of a biomarker, this indicates that the biomarker was having a significant impact on the model's ability to predict the outcome. A conclusion that can be drawn from this observation is that each biomarker plays a very important role in the predictive framework. A biomarker that is evidently contributing valuable data that enhances the model's ability to make precise predictions, when it is excluded, causes a noticeable drop in accuracy when excluded from the model. For this reason, retaining such biomarkers is essential for

maintaining the robustness and effectiveness of the predictive model.

6.3.1 NGF

As shown in Table 5.4a, accuracy decreased across all models when NGF was removed. The confusion matrix 5.4b also reveal a tendency for the model to predict class 2 as opposed to class 3. This observation may be explained by NGF being a biomarker for pain. A key role played by NGF as a pain biomarker is crucial to the accurate diagnosis of severe osteoarthritis, where pain is a significant symptom. In the absence of NGF, the model will be unable to incorporate this essential information, which will result in reduced accuracy and incorrect predictions since NGF cannot be incorporated into the model.

6.3.2 BGN

The removal of the biomarker biglycan (BGN) from the model resulted in a decrease in accuracy, as illustrated in Table 5.5a. Despite the accuracy drop, the confusion matrix in figure 5.5b clearly shows that the model's effectiveness diminished, particularly in predicting severe and moderate stages of OA. This may be due to BGN's crucial role in collagen fibril assembly, where it controls collagen fiber diameter and spacing. Regulation is vital for the proper formation and maintenance of these structural components. Functionality and durability of joints are directly influenced by collagen integrity.

In addition, BGN affects the growth and differentiation of osteoblasts and chondrocytes in conjunction with other cellular components. These cells are essential to maintaining bone and cartilage health. In the aftermath of injury or degradation, this is particularly important. The absence of BGN data complicates the assessment of tissue degradation. Consequently, the model's reduced performance in accurately predicting severe and moderate OA likely stems from the omission of these key biological data. Therefore, including BGN in the biomarker panel will enhance diagnostic accuracy and ensure effective monitoring of OA progression.

6.3.3 COMP664

The removal of COMP664 from the predictive model leads to a significant reduction in accuracy, as detailed in table 5.6a and figure 5.6b. Interestingly, this decline in accuracy surpasses that observed when other biomarkers or the original dataset are excluded, supporting COMP664's pivotal role in the model. It serves as a sign of cartilage degradation prior to onset of other clinical symptoms as COMP664 is a neo-epitope of an enzyme called Cartilage Oligomeric Matrix Protein. Biological fluids such as synovial fluid and saliva contain this biomarker, which correlates with cartilage matrix breakdown.

The COMP664 responds to mechanical loads on joints, making it an effective indicator of how joints respond to strain or physical activity. By monitoring changes in COMP664 concentrations, clinicians and researchers can determine whether physical

activities adversely affect joint health, an assessment crucial for managing conditions like osteoarthritis in athletes or working animals. It is also invaluable for predicting future joint conditions due to its dynamic changes in response to joint stress and damage. A predictive capability like this can significantly influence the trajectory of diseases like osteoarthritis and potentially slow their progression.

The biomarker could be the first sign of osteoarthritis (OA) because when the bones begins to breakdown, they can release the biomarker. This early release is why COMP664 has a significant effect on machine learning models used for predicting OA. Its unique ability to capture complex biological interactions within the joint that other biomarkers might miss has a significant impact on model accuracy following its exclusion. As a result, COMP664 is critically important for effective joint diagnosis and ongoing monitoring as part of comprehensive biomarker panels.

6.3.4 COMP156

The exclusion of COMP156 from the predictive model, as shown in table 5.7a and figure 5.7b, results in a noticeable decrease in overall accuracy. Even without COMP156, the model still predicts severe cases of osteoarthritis (OA) well. COMP156 is particularly important in accurately predicting the intermediate stages of OA, as demonstrated by this observation.

COMP156, a fragment of Cartilage Oligomeric Matrix Protein (COMP), acts as an early biomarker of cartilage degradation. Detection of this protein in biological fluids such as synovial fluid or serum is directly related to the breakdown of key cartilage matrix components, as occurs in osteoarthritis. Monitoring COMP156 levels regularly allows early detection and intervention of cartilage deterioration. When COMP156 levels decrease following treatment, it may indicate cartilage degradation has been reduced or the joint has been stabilized. Conversely, an increase in COMP156 levels may indicate that damage persists despite treatment efforts, signaling a need to adjust therapy.

COMP156 plays a critical role in enhancing predictive models' accuracy. As a result, the model's ability to predict joint disease progression and evaluate treatment responses is significantly enhanced. This makes COMP156 an essential component of comprehensive biomarker panels for diagnosing and monitoring joint health.

6.4 App

The development of our application marks a significant milestone in bridging the gap between machine learning technology and veterinary practice. Designed for local deployment, this application represents an important first step towards enhancing the disease prediction processes for veterinarians. By integrating advanced machine learning algorithms, the application aims to provide more accurate and timely diagnostics, ultimately improving animal health care outcomes.

The application has been tailored to meet the specific needs of veterinary professionals, featuring an intuitive interface that ensures ease of use even for those with limited technical expertise. Its robust functionality includes real-time data analysis, predictive modeling, and customizable reporting features. These capabilities enable veterinarians to make more informed decisions, streamline their workflow, and allocate resources more efficiently.

Furthermore, the application is built with scalability in mind, allowing for future updates and enhancements as machine learning technologies evolve. This adaptability ensures that the application will remain relevant and effective in addressing emerging veterinary challenges.

6.5 Future work

A significant step for future work is to collect more data for the biomarkers. Veterinarians often face challenges in classifying different stages of the disease due to the complex relationships between biomarkers. Therefore, it is crucial to gather extensive data and perform statistical analyses to understand how the different biomarkers influence each other.

Additionally, utilizing deep neural networks is a promising approach for the future discovery of biomarkers that may have a substantial impact on OA. Should a new biomarker be identified in the future, machine learning can be effectively employed to analyze its influence on the disease, providing deeper insights and improving diagnostic accuracy. For future work, it would be advantageous to use regression tasks instead of classification tasks. This approach could provide more nuanced insights by predicting the continuous levels of biomarkers, allowing for a more detailed understanding of their impact on joint health and the progression of osteoarthritis.

Bibliography

[1] Peter DeSaix, Gordon J Betts, Eddie Johnson, Jody E Johnson, Korol Oksana, Dean H Kruse, Brandon Poe, James A Wise, and Kelly A Young. Anatomy & physiology (openstax), 2013.

[2] Eva Skiöldebrand, Saritha Adepu, Claudia Lützelschwab, S Nyström, A Lindahl, K Abrahamsson-Aurell, and E Hansson. A randomized, triple-blinded controlled clinical study with a novel disease-modifying drug combination in equine lameness-associated osteoarthritis. *Osteoarthritis and Cartilage Open*, 5(3):100381, 2023.

[3] E Skiöldebrand, S Ekman, Lillemor Mattsson Hultén, E Svala, Karin Björkman, A Lindahl, A Lundqvist, P Önnerfjord, Carina Sihlbom, and U Rüetschi. Cartilage oligomeric matrix protein neoepitope in the synovial fluid of horses with acute lameness: A new biomarker for the early stages of osteoarthritis. *Equine veterinary journal*, 49(5):662–667, 2017.

[4] Michael W Ross and Sue J Dyson. *Diagnosis and Management of Lameness in the Horse*. Elsevier Health Sciences, 2010.

[5] E.J. Olson and C.S. Carlson. Bones joints tendons and ligaments. In J.F. Zachary, editor, *Pathologic Basis of Veterinary Disease*, pages 954–1008.e2. Mosby, 6th edition, 2017.

[6] P.R. van Weeren. General anatomy and physiology of joints. In C.W. McIlwraith, D.D. Frisbie, C.E. Kawcak, and P.R. van Weeren, editors, *Joint Disease in the Horse*, pages 1–24. Elsevier, 2nd edition, 2016.

[7] E. Skiöldebrand. *Studies of Articular Cartilage Macromolecules in the Equine Middle Carpal Joint in Joint Pathology and Training*. PhD thesis, Swedish University of Agricultural Sciences, Uppsala, 2004.

[8] A. Wilson and R. Weller. The biomechanics of the equine limb and its effect on lameness. In M.W. Ross and S.J. Dyson, editors, *Diagnosis and Management of Lameness in the Horse*, pages 270–281. W.B. Saunders, 2nd edition, 2011.

[9] J.P. Caron. Osteoarthritis. In M.W. Ross and S.J. Dyson, editors, *Diagnosis and Management of Lameness in the Horse*, pages 655–668. W.B. Saunders,

2nd edition, 2011.

[10] C Wayne McIlwraith, Christopher E Kawcak, David D Frisbie, Christopher B Little, Peter D Clegg, Mandy J Peffers, Morten A Karsdal, Stina Ekman, Sheila Laverty, Richard A Slayden, et al. Biomarkers for equine joint injury and osteoarthritis. *Journal of Orthopaedic Research®*, 36(3):823–831, 2018.

[11] A Kendall, S Nyström, S Ekman, LM Hultén, A Lindahl, E Hansson, and E Skiöldebrand. Nerve growth factor in the equine joint. *The Veterinary Journal*, 267:105579, 2021.

[12] Susan Tseng, A Hari Reddi, and Paul E Di Cesare. Cartilage oligomeric matrix protein (comp): a biomarker of arthritis. *Biomarker insights*, 4:BMI–S645, 2009.

[13] Stina Ekman, A Lindahl, U Rüetschi, A Jansson, K Björkman, K Abrahamsson-Aurell, S Björnsdóttir, M Löfgren, L Mattsson Hultén, and E Skiöldebrand. Effect of circadian rhythm, age, training and acute lameness on serum concentrations of cartilage oligomeric matrix protein (comp) neo-epitope in horses. *Equine veterinary journal*, 51(5):674–680, 2019.

[14] Ebba Arrhult. Effect of training and feeding on saliva concentrations of cartilage oligomeric matrix protein neo-epitope (comp664) and neuropeptide substance p in horses. Master's thesis, Swedish University of Agricultural Sciences, Faculty of Veterinary Medicine and Animal Science, 2024.

[15] Madalina V Nastase, Marian F Young, and Liliana Schaefer. Biglycan: a multivalent proteoglycan providing structure and signals. *Journal of Histochemistry & Cytochemistry*, 60(12):963–975, 2012.

[16] Saritha Adepu, Stina Ekman, Jakob Leth, Ulrika Johansson, A Lindahl, and Eva Skiöldebrand. Biglycan neo-epitope (bgn262), a novel biomarker for screening early changes in equine osteoarthritic subchondral bone. *Osteoarthritis and Cartilage*, 30(10):1328–1336, 2022.

[17] John R Crowther. *The ELISA guidebook*, volume 149. Springer Science & Business Media, 2008.

[18] F DIXON MATTHEW. *Machine learning in finance: From theory to practice*. Springer, 2021.

[19] James A Anderson. *An introduction to neural networks*. MIT press, 1995.

[20] Cheng-Hsiung Weng, Tony Cheng-Kui Huang, and Ruo-Ping Han. Disease prediction with different types of neural network classifiers. *Telematics and Informatics*, 33(2):277–292, 2016.

[21] Bernhard Mehlig. *Machine Learning with Neural Networks*. Department of Physics, University of Gothenburg, Göteborg, Sweden, 2021.

[22] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[23] Towards AI. The sigmoid function: A key building block in neural networks. `https://towardsai.net`, 2023. Retrieved from Towards AI.

[24] Analytics Vidhya. Softmax activation function: How it works. `https://www.analyticsvidhya.com`, 2023. Retrieved from Analytics Vidhya.

[25] DataCamp. Cross-entropy loss function in machine learning: Enhancing model accuracy. `https://www.datacamp.com`, 2023. Retrieved from DataCamp.

[26] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[27] Margherita Grandini, Enrico Bagli, and Giorgio Visani. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*, 2020.

[28] Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, 2009.

[29] Andreas C. Müller and Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O'Reilly Media, 2016.

[30] Joseph M. Hilbe. *Logistic Regression Models*. Chapman and Hall/CRC, 2009.

[31] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[32] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, 2019.

[33] Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, and Keying E. Ye. *Probability and Statistics for Engineers and Scientists*. Pearson, 2011.

A

Appendix 1

In this appendix, the series of distribution figures that serve as a important information to our analysis on the progression timeline of Osteoarthritis (OA). Each figure showcases the original dataset corresponding to a distinct class within our study, overlaid with fitted models of Normal, Log-Normal, and Gamma distributions.

These graphical representations are designed to illustrate how each distribution aligns with the empirical data across different duration classes, thereby providing a visual examination of the statistical properties and variability inherent in the OA progression timeline. The implications of these distribution fits are further explored upon within the the document, underscoring their relevance to our analytics on OA.

Table A.1: K-S Statistics and Best Fitting Distributions by Diagnosis

Diagnosis	**Variable**	**Normal**	**Log-Normal**	**Gamma**
0	NGF	**0.1253**	0.1339	0.1264
0	BGN262	**0.1551**	0.2037	0.1702
0	COMP664	0.1436	**0.1210**	0.1293
0	COMP156	0.2964	**0.1938**	0.2448
1	NGF	**0.0770**	0.1486	0.1181
1	BGN262	0.2546	**0.1487**	0.1626
1	COMP664	**0.1291**	0.1612	0.1488
1	COMP156	0.1404	**0.1224**	0.1297
2	NGF	**0.2351**	0.2753	0.2553
2	BGN262	0.2381	**0.1501**	0.1759
2	COMP664	0.0843	0.0897	**0.0821**
2	COMP156	0.1908	0.1490	**0.1093**
3	NGF	0.0761	0.0913	**0.0746**
3	BGN262	0.1534	0.1887	**0.1425**
3	COMP664	0.2454	**0.1053**	0.1600
3	COMP156	0.1850	**0.1741**	0.1860

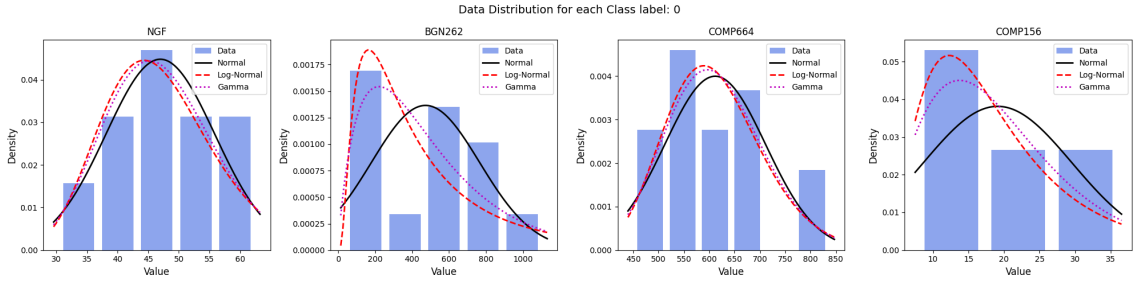


Figure A.1: Distribution Analysis for all variable within diagnosis class Healthy.

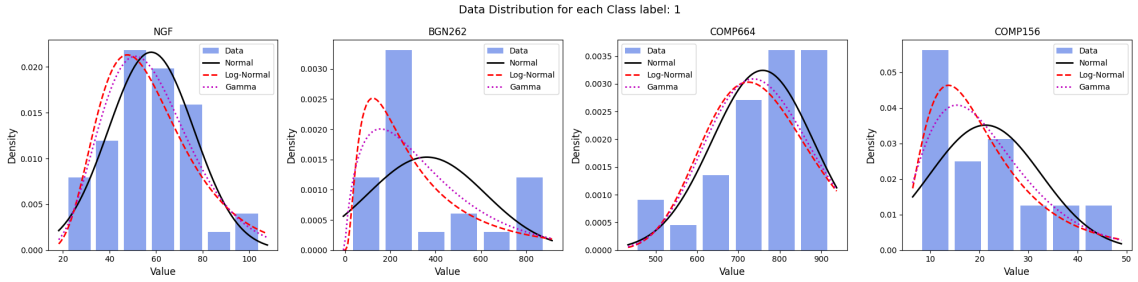


Figure A.2: Distribution Analysis for all variable within diagnosis class Mild OA.

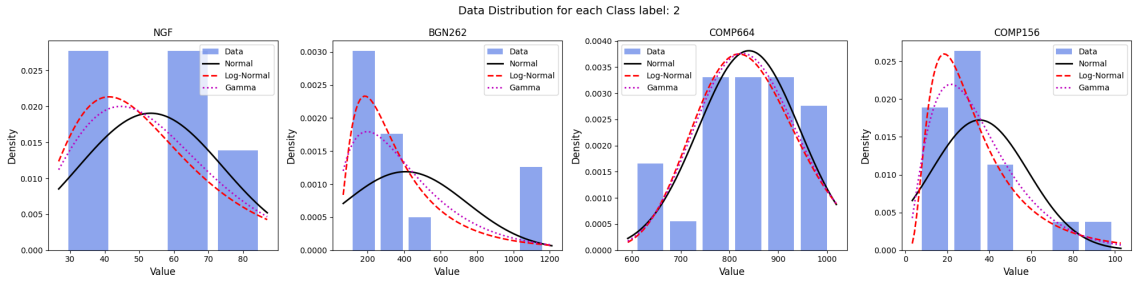


Figure A.3: Distribution Analysis for all variable within diagnosis class Moderate OA

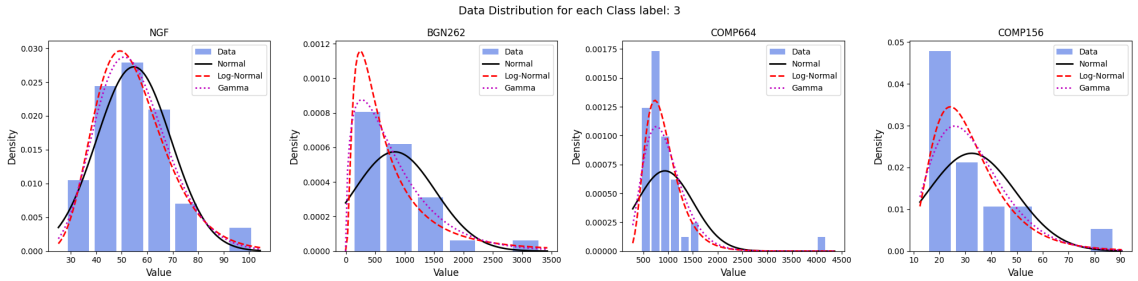


Figure A.4: Distribution Analysis for all variable within diagnosis class Servere OA

B

Appendix 2

In this section, a comparative analysis of the original and synthetic data for each biomarker used in the study. The purpose of this comparison is to illustrate the similarity between the original dataset and the synthetic dataset generated for the research. Each biomarker's data is plotted to visually demonstrate the distribution and characteristics of the data points in both the original and synthetic datasets.

Figure B.1 shows the distribution of NGF, BGN262, COMP664, and COMP156 biomarkers in the healthy class for both original and synthetic datasets. For the mild OA class, Figure B.2 provides a comparison of the same biomarkers between the original and synthetic datasets. In the case of the moderate OA class, Figure B.3 illustrates the distribution comparison for these biomarkers between the original and synthetic datasets. Finally, Figure B.4 depicts the distribution of NGF, BGN262, COMP664, and COMP156 biomarkers in the severe OA class for both original and synthetic datasets.

The synthetic data closely mirrors the distribution of the original data across all classes, indicating that the synthetic data generation process has effectively captured the key characteristics of the biomarkers. This validation ensures that the analysis and modeling conducted in this study are based on robust and reliable data.

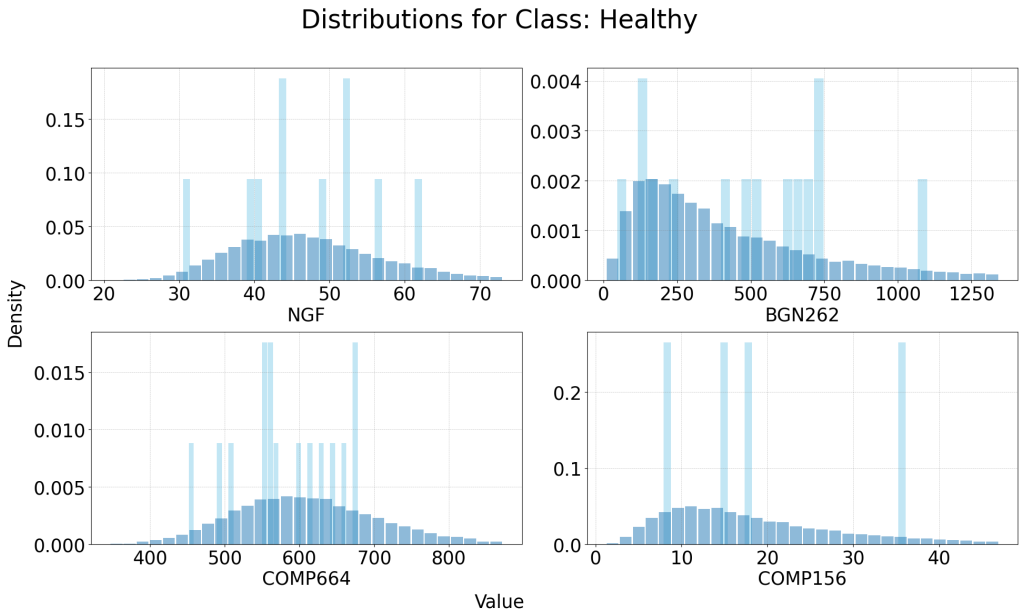


Figure B.1: Distribution of biomarkers in original and synthetic datasets for healthy horses.

Figure B.2: Distribution of biomarkers in original and synthetic datasets for Mild OA horses.

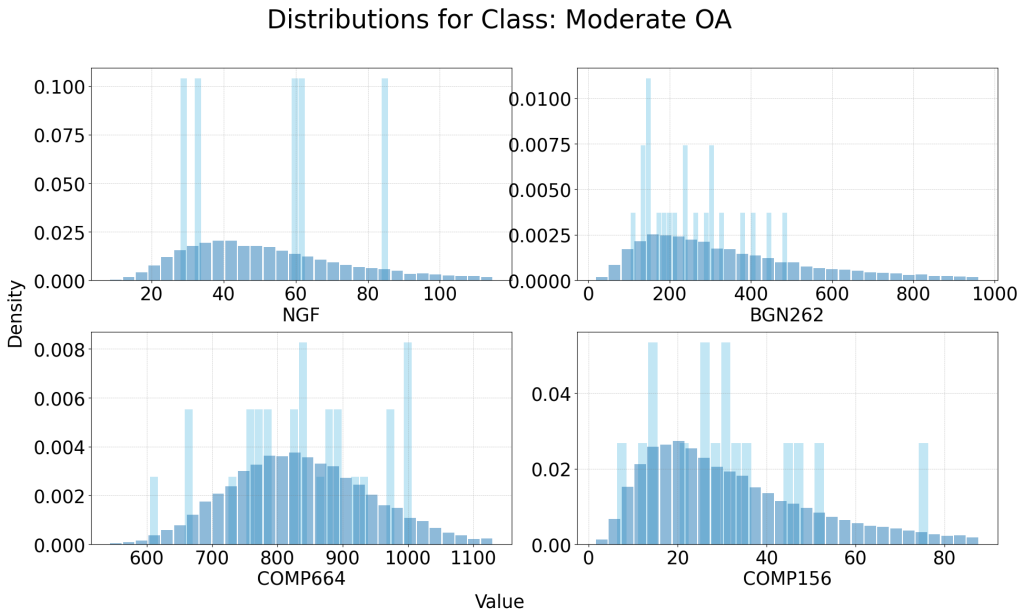


Figure B.3: Distribution of biomarkers in original and synthetic datasets for Moderate OA horses.

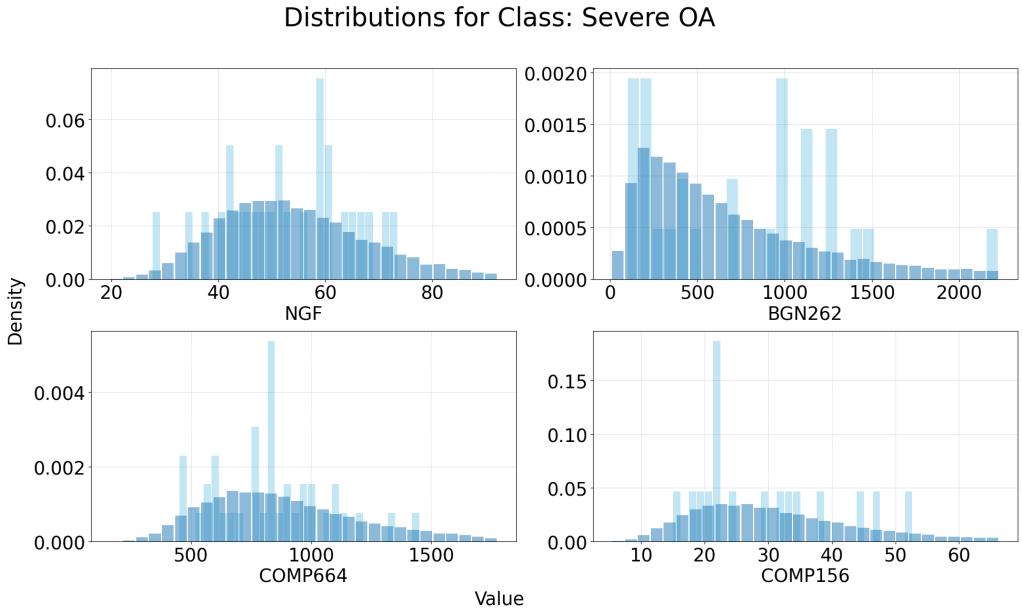


Figure B.4: Distribution of biomarkers in original and synthetic datasets for Severe OA horses.

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY