ANALYSIS OF DOUBLE GENERALIZED ADDITIVE MODELS

NATAKON NAWARATANA

A Thesis Submitted in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy in Applied Mathematics

Suranaree University of Technology

Academic Year 2025

การวิเคราะห์ตัวแบบเชิงบวกนัยทั่วไปคู่

นายณฐกร นวรตน

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรดุษฎีบัณฑิต

สาขาวิชาคณิตศาสตร์ประยุกต์

มหาวิทยาลัยเทคโนโลยีสุรนารี

ปีการศึกษา 2568

ANALYSIS OF DOUBLE GENERALIZED ADDITIVE MODELS

Suranaree University of Technology has approved this thesis submitted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy.

Thesis Examining Committee

(Asst. Prof. Dr. Prapart Pue-on)
Chairperson

(Asst. Prof. Dr. Jessada Tanthanuch)
Member (Thesis Advisor)

(Assoc Prof. Dr. Supaporn Suksern)
Member

(Asst. Prof. Dr. Tidarut Areerak)
Member

(Dr. Akanat Wetayawanich)
Member

(Assoc. Prof. Dr. Yupaporn Ruksakulpiwat)
Acting Vice Rector for Academic Affairs
and Quality Assurance

(Prof. Dr. Santi Maensiri)
Acting Dean of Institute of Science

ณฐกร นวรตน : การวิเคราะห์ตัวแบบเชิงบวกนัยทั่วไปคู่ (ANALYSIS OF DOUBLE GENERALIZED ADDITIVE MODELS) อาจารย์ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร.เจษฎา ตัณฑนุช, 128 หน้า.

คำสำคัญ: ตัวแบบเชิงบวกนัยทั่วไปคู่/การวิเคราะห์เชิงทำนาย/การเรียนรู้ของเครื่อง

วิทยานิพนธ์นี้ต้องการพัฒนาและขยายแนวคิดการสร้างตัวแบบทางคณิตศาสตร์เพื่อใช้ในการทำนายไปสู่ตัวแบบเชิงบวกนัยทั่วไปคู่ เนื้อหาวิทยานิพนธ์ได้แสดงให้เห็นถึงตัวแบบเชิงทำนายต่าง ๆ ได้แก่ ตัวแบบเชิงเส้น ตัวแบบเชิงเส้นนัยทั่วไป ตัวแบบเชิงเส้นนัยทั่วไปสองชั้น รวมไปถึงตัวแบบเชิงบวกนัยทั่วไป และ การประมาณค่าพารามิเตอร์ต่าง ๆ ในการสร้างตัวแบบ ผลของการศึกษาแสดงให้เห็นถึงโครงสร้างทางคณิตศาสตร์ของตัวแบบเชิงบวกนัยทั่วไปคู่ ได้แก่ ส่วนประกอบสุ่ม โครงสร้างค่าเฉลี่ย โครงสร้างค่าการกระจาย ฟังก์ชันเชื่อมโยง และการประมาณค่าพารามิเตอร์ของตัวแบบดังกล่าวด้วยวิธี เสมือนพหุนามเชิงแบบฉบับ และการประมาณโดยใช้เทคนิคการเรียนรู้ของเครื่องอีก 4 แบบ ได้แก่ เอ็กซ์เพลนเนเบิลบูสติงแมชชีน เอ็กซ์ตรีมแกรเดียนต์บูสติง ป่าสุ่ม และซัพพอร์ตเวกเตอร์รีเกรสชัน พบว่าตัวแบบเชิงบวกนัยทั่วไปคู่แตกต่างจากตัวแบบเชิงบวกนัยทั่วไปโดยไม่เพียงแต่จะเป็นการสร้างตัวแบบของค่าเฉลี่ย แต่มีการสร้างตัวแบบของค่าการกระจายอีกด้วย เพื่อให้เห็นถึงประสิทธิภาพการทำนายของตัวแบบที่พัฒนาขึ้น ได้นำไปทดสอบการสร้างตัวแบบกับชุดข้อมูล 4 ชุด ได้แก่ Fetal Health Monitoring Dataset, Life Style Data, Health Checkup Result Dataset และ Life Expectancy (WHO) Fixed Dataset ในแต่ละข้อมูลทำการแบ่งเป็นชุดเรียนรู้ ชุดปรับพารามิเตอร์ และ ชุดทดสอบ 3 กลุ่มได้แก่ 60:20:20, 70:15:15 และ 80:10:10 และทำการเปรียบเทียบประสิทธิภาพกับตัวแบบเชิงบวกนัยทั่วไปด้วยเครื่องวัดต่าง ๆ ได้แก่ เอไอซี ค่าควรจะเป็นลบ ค่าความคลาดเคลื่อนสมบูรณ์เฉลี่ย รากของค่าคลาดเคลื่อนกำลังสองเฉลี่ย สัมประสิทธิ์การกำหนด ซีอาร์พีเอส และพีไอซีพี95 โดยแต่ละชุดข้อมูลจะให้ทำการทำนายตัวแปรลักษณะต่าง ๆ กัน 4 ตัวแปร ทำให้ได้ตัวแบบเชิงบวกนัยทั่วไปทั้งหมด 240 ตัวแบบ และตัวแบบเชิงบวกนัยทั่วไปคู่ทั้งหมด 240 ตัวแบบ ผลการดำเนินการพบว่าเมื่อวัดประสิทธิภาพในเชิงควรจะเป็นและในเชิงความน่าจะเป็น เห็นได้ชัดว่าตัวแบบเชิงบวกนัยทั่วไปคู่มีประสิทธิภาพในการทำนายดีกว่าอย่างชัดเจน

สาขาวิชาคณิตศาสตร์และภูมิสารสนเทศ	ลายมือชื่อนักศึกษา ณฐกร
ปีการศึกษา 2568	ลายมือชื่ออาจารย์ที่ปรึกษา J. Tanthanuch
	ลายมือชื่ออาจารย์ที่ปรึกษาร่วม Oลรรัตน์

NATAKON NAWARATANA : ANALYSIS OF DOUBLE GENERALIZED ADDITIVE MODELS. THESIS ADVISOR : ASST. PROF. JESSADA TANTHANUCH, Ph.D. 128 PP.

Keyword: DOUBLE GENERALIZED ADDITIVE MODEL, PREDICTIVE ANALYSIS, MACHINE LEARNING

This thesis aims to develop and extend the concept of mathematical modeling for prediction toward the Double Generalized Additive Model (DGAM) framework. The thesis presents a range of predictive models, including linear models, generalized linear models, double generalized linear models, and generalized additive models (GAM), together with methods for parameter estimation in model construction.

The results demonstrate the mathematical structure of the DGAM, which comprises the random component, mean structure, dispersion structure, link functions, and parameter estimation procedures. Parameter estimation is carried out using classical spline-based techniques as well as four machine learning approaches: Explainable Boosting Machine, Extreme Gradient Boosting, Random Forest, and Support Vector Regression. The findings indicate that the DGAM differs fundamentally from the GAM in that it does not solely model the mean structure but additionally incorporates a dispersion model.

To evaluate the predictive performance of the proposed models, experiments are conducted on four datasets: the Fetal Health Monitoring Dataset, Life Style Data, Health Checkup Result Dataset, and the Life Expectancy (WHO) Fixed Dataset. For each dataset, the data are partitioned into training, validation, and testing sets using three split ratios: 60:20:20, 70:15:15, and 80:10:10. The predictive performance of the proposed models is compared with that of the generalized additive model using multiple evaluation metrics, including the Akaike Information Criterion (AIC), negative log-likelihood, mean absolute error, root mean squared error, coefficient of determination (R^2), Continuous Ranked Probability Score (CRPS), and 95% Prediction Interval Coverage Probability (PICP95). For each dataset, four different feature variables are predicted, resulting in a total of 240 GAM models and 240 DGAM models.

The calculation results clearly show that, when evaluated using likelihood-based and probabilistic performance measures, the Double Generalized Additive Model consistently outperforms the Generalized Additive Model, demonstrating substantially superior predictive performance.

School of Mathematical Sciences	Student's Signature Natakon
and Geoinformatics	Advisor's Signature J.Tanthanuch
Academic Year 2025	Co-Advisor's Signature Amornrat.

ACKNOWLEDGEMENTS

I wish to express my deepest gratitude to all who have contributed to the completion of this dissertation.

First and foremost, I am profoundly grateful to my advisor, Asst. Prof. Dr. Jessada Tanthanuch, for his invaluable guidance, profound mathematical insights, and dedication to teaching the fundamental principles of mathematics and research methodology. His patience, encouragement, and expertise have been instrumental in shaping my academic development.

I extend my heartfelt thanks to my co-advisor, Asst. Prof. Dr. Amornrat Suriyawichitseranee, for her meticulous reviews, comprehensive feedback, and ability to illuminate new perspectives that significantly enhanced the quality of this research. Her constructive criticism and thoughtful suggestions have been invaluable.

I am deeply indebted to Prof. Dr. Hong Zhang for the opportunity to conduct a research visit at Changzhou Institute of Technology, China. His mentorship provided invaluable international experience and broadened my perspective on practical applications of statistical modeling.

I gratefully acknowledge the financial support from the 84th Birthday Anniversary of His Majesty King Bhumibol Adulyadej Scholarship and the Institute for the Promotion of Teaching Science and Technology (IPST) Scholarship.

My sincere appreciation extends to the faculty and staff of the School of Mathematics, Suranaree University of Technology, for their support throughout my studies.

Finally, I am forever grateful to my family and friends for their unconditional love, patience, and unwavering belief in me.

Natakon Nawaratana

CONTENTS

CONTENTS (Continued)

CONTENTS (Continued)

CONTENTS (Continued)

CONTENTS (Continued)

CONTENTS (Continued)

LIST OF TABLES

LIST OF TABLES (Continued)

LIST OF TABLES (Continued)

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The importance of mathematical and statistical models in forecasting cannot be overstated, as they empower us to uncover patterns within historical data and make predictions about the future. The utilization of these models brings forth a multitude of benefits, including:

- Unveiling Hidden Trends: These models are capable of recognizing trends in historical data that may not be visible to the unaided eye.
- Forecasting a Various Set of Variables: By utilizing these models, we are able to forecast a wide range of variables, like those that present difficulties for conventional modeling methods.
- Accurate Future Predictions: Through mathematical and statistical models, we are able to predict the future with some degree of precision.
- Evaluation of Forecast Accuracy and Identification of Risks: These models can be employed to evaluate forecast accuracy and find potential risks related to anticipated outcomes.

There exist various types of mathematical and statistical models that can be utilized for forecasting, encompassing time series models, econometric models, and machine learning models, among others. It is important to note, however, that these models are not flawless and may occasionally produce inaccurate predictions. Nevertheless, they remain a valuable tool for forecasting future trends and facilitating well-informed decision-making processes (De Jong and Heller, 2008).

Generalized linear models (GLMs) are statistical models specifically designed to handle data that deviates from a normal distribution. This characteristic renders them highly valuable in the field forecasting since numerous real-world variables exhibit non-normal distributions. Take, for instance, sales data, which often displays a skew towards higher values. GLMs effectively capture such patterns and enable accurate forecasts for this type of data. Although GLMs are effective tools for modeling non-normal data, it is important to keep in mind that they may have limitations when dealing with data that does not follow a particular distribution or exhibit non-linear relationships. GLMs might not be able to fully capture the complexity of such data patterns, despite their advantages. Generalized additive models (GAMs) serve as an extension of GLMs, offering increased flexibility in capturing the intricate relationship between predictors and the response variable. Consequently, they prove to be an invaluable tool for forecasting since many real-world variables defy specific distribution assumptions and exhibit non-linear relationships (Zanin and Marra, 2012). The following are some advantages of extending GLMs to GAMs:

- Flexibility: GAMs allow for greater adaptability in modeling the complex relationships between predictors and the response variable. This is particularly advantageous when dealing with variables that do not adhere to specific distribution assumptions and showcase non-linear patterns.

- Accuracy: GAMs often yield more accurate forecasts compared to GLMs, especially when handling data that does not follow specific distributions or exhibits non-linear relationships. The flexibility of GAMs enables them to capture more nuanced patterns and variations, resulting in improved forecasting accuracy.

- Interpretability: GAMs can provide better interpretability compared to GLMs, as they can uncover and depict non-linear relationships between predictors and the response variable. This offers valuable insights into the underlying factors influencing the response variable, aiding in informed decision-making for future planning.

Here are some instances of applying GAMs to forecasting:

- Sales Forecasting: GAMs can be employed to forecast sales of products or services, enabling businesses to plan production, inventory management, and targeted marketing strategies more effectively.

- Demand Forecasting: GAMs can predict the demand for products or services, assisting businesses in optimizing resource allocation, staffing levels, pricing strategies, and inventory management.

- Risk Forecasting: GAMs can be utilized to forecast the likelihood of events such as defaults, bankruptcies, or natural disasters. This empowers businesses to assess potential financial exposures and make well-informed risk management decisions.

- Healthcare Forecasting: GAMs can forecast healthcare service demand, including hospital admissions and doctor visits. Healthcare providers can utilize these forecasts to optimize resource allocation, staffing, and financial projections.

Extending generalized linear models (GLMs) to double generalized linear models (DGLMs) can be important for forecasting in cases where the variance of the response variable is not constant. In GLMs, the variance of the response variable is assumed to be constant, but this assumption may not hold true in all cases. For example, the variance of sales data may be higher for products that are more popular. DGLM is a model that extends the GLM by allowing the variance of the response variable to be modeled as a function of the mean. This is in contrast to GLMs, which assume that the variance of the response variable is constant. In this case, extending GLMs to DGLMs can be a valuable tool for forecasting. DGLMs are more flexible, accurate, and interpretable than GLMs, making them a better choice for modeling data that does not follow a specific distribution and has non-linear relationships.

Croux, Gijbels, and Prosdocimi (2012) developed robust estimation methods for mean and dispersion functions within the framework of extended generalized additive models. Their approach utilized P-splines and quasi-likelihood methods to model both the mean and dispersion components, demonstrating improved performance in the presence of outliers. Gijbels and Prosdocimi (2011) further illustrated these concepts through applications to Italian abortion data, showing how flexible modeling of both mean and dispersion can reveal important patterns that would be missed by standard GLMs or GAMs with fixed variance structures. In 2022, Chang, Caruana, and Goldenberg extended this research by employing several techniques, namely Explainable Boosting Machine (EBM), splines, nodes, eXtreme Gradient Boosting (XGBoost), and Random Forest (RF), to find the parameters of GAMs models. The results of their study demonstrated that these models were capable of identifying interesting patterns within the data.While GAMs provide excellent interpretability of nonlinear relationships, their performance can be enhanced through integration with other machine learning techniques. Support Vector Regression (SVR), based on Vapnik's statistical learning theory (1995), has emerged as a complementary approach that excels in handling high-volatility data regions where GAMs may struggle. Hybrid GAM-SVR frameworks capitalize on the interpretability of GAMs for elucidating fundamental variable relationships while leveraging SVR's robust error minimization capabilities in regions characterized by extreme fluctuations (López et al., 2025). This synergistic combination enables predictive models that maintain both transparency and superior accuracy across diverse data conditions.

In this thesis, we develop and evaluate advanced parameter estimation techniques for Double Generalized Additive Models (DGAMs) that integrate traditional statistical approaches with modern machine learning methods, including Smoothing Splines, Support Vector Regression, Explainable Boosting Machine, eXtreme Gradient Boosting, and Random Forest. The research systematically compares multiple estimation frameworks to identify optimal strategies for different forecasting contexts and data characteristics. Comprehensive validation will be conducted using standard benchmark datasets and real-world health data encompassing critical life-stage outcomes including birth, aging, morbidity, and mortality patterns. Model evaluation will assess predictive accuracy, interpretability, computational efficiency, and robustness through rigorous cross-validation, sensitivity analyses, and practical case studies in healthcare forecasting applications.

1.1 Research Objectives

1. To develop techniques for determining model parameters in Double Generalized Additive Models (DGAMs) using Smoothing Splines, Explainable Boosting Machine, eXtreme Gradient Boosting, Random Forest, and Support Vector Regression. This will be accomplished by employing all proposed techniques to create forecasting models across multiple datasets, each containing multiple variables examined at various proportion levels.

2. To evaluate and compare the performance of all proposed methods for the obtained forecasts based on predictive accuracy, interpretability, computational efficiency, and robustness across different dataset configurations, variable selections, and proportion specifications.

1.2 Scope and Limitations

In this study, a range of techniques has been utilized to address the challenge of solving the double generalized additive models (DGAMs) problem. These techniques encompass spline, Explainable Boosting Machine (EBM), eXtreme Gradient Boosting (XGBoost), Random Forest (RF), and Support Vector Regression (SVR). Each technique brings its unique strengths and capabilities to the table, providing diverse approaches for effectively tackling the problem at hand. In addition, the proposed theoretical developments are empirically evaluated using four publicly available healthcare datasets obtained from Kaggle:

1. Fetal Health Monitoring Dataset;

2. Life Style Data;

3. Health Checkup Result Dataset;

4. Life Expectancy (WHO) Fixed Dataset.

1.3 Research Procedure

The research work will proceed as follows:

1. Study background techniques for estimating the mean and dispersion/variance in relation to covariates is critical for focusing on the flexibility of Generalized Additive Models.

2. Study and prepare the knowledge of mathematics essential to comprehend the techniques that the relationship uses to estimate model parameters.

3. Modify the source code of the statistical computing program to improve the prediction of the Double Generalized Additive Models.

4. Apply the improved code in conjunction with real-world data to generate relevant predictive or problem-solving abilities models.

5. Evaluate the predictive performance of the model by measuring its accuracy and analyze the experimental results to identify areas for potential future research advancement.

1.4 Expected Result

A performance analysis of DGAMs obtained using various techniques in parameter estimation will be presented.

CHAPTER II

LITERATURE REVIEW

This chapter presents the mathematical and statistical modeling background, along with the statistical measures essential for evaluating mathematical and statistical models. In addition, relevant prior studies are reviewed to inform the research process and support the achievement of more robust and effective results.

2.1 History and Development of Linear Modeling

The development of modern statistical modeling follows a smooth progression from Gauss's original concept of simple least squares to contemporary generalized linear modeling frameworks. This evolution, as outlined by de Jong and Heller (2008), encompasses several key stages that progressively increase in mathematical sophistication and practical applicability.

2.1.1 Simple Linear Modeling

The foundational approach aims to explain an observed response variable y through a single explanatory variable x. In the statistical literature, the response variable y is alternatively referred to as the dependent variable, outcome variable, or (in econometrics) endogenous variable. Similarly, the explanatory variable x has various designations including covariate, independent variable, predictor, driver, risk factor, exogenous variable, regressor, or simply the "x" variable. When x represents categorical data, it is termed a factor.

The simple linear model takes the form:

$$y_i = \beta_0 + \beta_1 x_{i1} + \epsilon_i, \quad i = 1, 2, \ldots, n,$$

where β_0 and β_1 are parameters to be estimated, x_{i1} represents the single explanatory variable for observation i, and ϵ_i represents the random error term.

2.1.2 Multiple Linear Modeling

This framework extends simple least squares by incorporating multiple explanatory variables. Here, $\mathsf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ik})^T$ contains k explanatory variables for each observation i, with their combination serving to explain the response y_i (de Jong and Heller, 2008).

The multiple linear model is expressed as:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \epsilon_i,$$

or equivalently:

$$y_i = \mathsf{x}_i^T \boldsymbol{\beta} + \epsilon_i,$$

where $\mathsf{x}_i = (1, x_{i1}, x_{i2}, \ldots, x_{ik})^T$ is the $(k+1) \times 1$ covariate vector for observation i, and $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_k)^T$ is the $(k+1) \times 1$ parameter vector.

In matrix notation for all n observations:

$$\mathbf{y} = \mathsf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)^T$ is the $n \times 1$ response vector, X is the $n \times (k+1)$ design matrix with the ith row being $\mathsf{x}_i^T = (1, x_{i1}, x_{i2}, \ldots, x_{ik})$, and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)^T$ is the $n \times 1$ error vector.

2.1.3 Response Transformation

A natural extension involves replacing the response variable y with a transformation $g(y)$. In this approach, the aim is to model the transformed response $g(y_i)$ in terms of the observed explanatory variables in $\mathsf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ik})^T$:

$$g(y_i) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \epsilon_i,$$

or equivalently:

$$g(y_i) = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i.$$

Common transformations include the logarithm $g(y) = \log(y)$ and logit $g(y) = \log\left(\frac{y}{1-y}\right)$ functions. For interpretability and mathematical tractability, the transformation function g is constrained to be monotonic.

2.1.4 Classical Linear Modeling

A more conceptually sophisticated development replaces the response y_i with its expected value $\mathbb{E}[y_i]$, or more precisely, $\mathbb{E}[y_i|\mathbf{x}_i]$. This paradigm shift enables modeling of the statistical average of y in terms of the predictors $\mathbf{x}_i$:

$$\mathbb{E}[y_i|\mathbf{x}_i] = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik},$$

or equivalently:

$$\mathbb{E}[y_i|\mathbf{x}_i] = \mathbf{x}_i^T \boldsymbol{\beta}.$$

This modeling perspective focuses on the conditional expectation of the response rather than individual observations, thereby accounting for the inherent variability present in real data (de Jong and Heller, 2008).

However, the classical linear model relies on the assumptions of normality and constant variance, which may be restrictive when dealing with non-Gaussian outcomes such as binary responses, counts, or proportions. These limitations motivate a broader modeling framework capable of accommodating different distributional structures while preserving interpretability and analytical coherence.

The formal development of this extended framework is presented in the following section.

2.2 Generalized Linear Model (GLM)

A generalized linear model (GLM) is a flexible statistical framework that extends linear regression to accommodate various types of response variables and their associated probability distributions. It provides a way to model the relationship between a set of predictors and a response variable while accounting for non-normal and non-continuous data. GLMs allow for the use of different link functions to relate the predictors to the response, making them suitable for analyzing a wide range of data types, including binary, count, and categorical outcomes (Wood, 2017). The flexibility of GLMs makes them widely used in fields such as medicine, social sciences, and economics, where complex data structures and diverse response variables are encountered

The generalized linear models (GLMs) differ from traditional linear models in that they do not assume the normal distribution of the response variable Y_i and do not strictly require $\mathbb{E}[Y] = X\beta$. Instead, GLMs offer more flexibility by allowing the choice of distribution for Y and a corresponding link function g such that $g(\mathbb{E}[Y]) = \eta = X\beta$. These generalizations are based on three main components:

2.2.1 Random Component

The distribution of the response variable for each of the n independently sampled observations is determined by a random component. This random component defines the conditional distribution based on the values of the explanatory variables included in the model. The distribution is selected from the exponential family and can belong to various families, including Gaussian (normal), Binomial, Poisson, Gamma, or Inverse-Gaussian distributions.

The exponential dispersion family represents a class of probability density functions characterized by the following form:

$$f_y(y;\theta,\phi) = \text{exp}\left(\frac{y\theta - b(\theta)}{a(\phi)} + c(y;\phi)\right). \tag{2.1}$$

In this expression, $\phi > 0$ denotes the dispersion parameter, $\theta \in \Theta$ represents the parameter specific to the distribution, and $\Theta \subset \mathbb{R}$ is an open set that includes θ. The function a allows for the incorporation of weights in the distribution, although generally $a(\phi) = \phi$. Meanwhile, $b : \Theta \to \mathbb{R}$ corresponds to the cumulant function, and c is a normalization factor that does not depend on θ.

2.2.2 Systematic Component

A linear predictor of a GLM is a linear function of the i-th regressors:

$$\eta_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_k X_{ik},$$

where the β_j are the model parameters associated with the regressors X_{ij} to be estimated.

For responses in the exponential family with cumulant function $b(\theta)$,

$$\mu_i = \mathbb{E}[Y_i] = b'(\theta_i), \qquad \text{Var}(Y_i) = \phi\, b''(\theta_i) = \phi V(\mu_i).$$

2.2.3 The Link Function

The GLM incorporates a smooth and invertible link function denoted by $g(\cdot)$. This function relates the mean $\mu_i = \mathbb{E}[Y_i]$ to the linear predictor:

$$g(\mu_i) = \eta_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_k X_{ik}. \tag{2.2}$$

For example, we might require that the mean must be strictly positive (as in claim counts and severity). In such cases, link functions that only produce positive values might be more suitable. The following link functions are commonly considered:
where $\Phi(\cdot)$ is the standard normal cumulative distribution function.

Due to the invertibility of the link function, an alternative expression can be used:

$$\mu_i = g^{-1}(\eta_i) = g^{-1}(\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_k X_{ik}), i = 1, 2, \ldots, n.$$

This representation allows GLMs to be considered either as a linear model for a transfor-

Table 2.1 Common link functions in generalized linear models.

Link function name	Link function formula
Identity	$g(\mu) = \mu$
Reciprocal	$g(\mu) = \mu^{-1}$
Log	$g(\mu) = \log(\mu)$
Complementary log-log	$g(\mu) = \log(-\log(1-\mu))$
Logit	$g(\mu) = \log(\mu/(1-\mu))$
Probit	$g(\mu) = \Phi^{-1}(\mu)$

mation of the expected response or as a nonlinear regression model for the response. The inverse function $g^{-1}(\cdot)$ is also referred to as the mean function. It is worth noting that the identity link function simply returns its argument unchanged, so $\eta_i = g(\mu_i) = \mu_i$, resulting in $\eta_i = g^{-1}(\mu_i) = \mu_i$.

GLMs are applied to data using the maximum likelihood method, which allows for the estimation of both the regression coefficients and the estimated asymptotic standard errors of the coefficients.

2.2.4 Parameter Estimation

The parameters of a generalized linear model are typically estimated using the maximum likelihood estimation (MLE) method. Given n independent observations $(y_1, y_2, \ldots, y_n)$ with corresponding covariate vectors $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ik})^T$ for $i = 1, 2, \ldots, n$, the likelihood function is constructed based on the exponential dispersion family:

$$L(\boldsymbol{\beta}, \phi; \mathbf{y}) = \prod_{i=1}^{n} f_{y_i}(y_i; \theta_i, \phi) = \prod_{i=1}^{n} \exp\left(\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i; \phi)\right).$$

The log-likelihood function is then given by:

$$\ell(\boldsymbol{\beta}, \phi; \mathbf{y}) = \sum_{i=1}^{n} \left[\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i; \phi)\right],$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_k)^T$ is the vector of regression parameters.

Since $\mu_i = \mathbb{E}[Y_i] = b'(\theta_i)$ and $\eta_i = g(\mu_i) = \mathbf{x}_i^T\boldsymbol{\beta}$, the relationship between θ_i

and $\boldsymbol{\beta}$ is established through the link function. The maximum likelihood estimates $\hat{\boldsymbol{\beta}}$ are obtained by solving the score equations:

$$\mathsf{U}(\boldsymbol{\beta}) = \frac{\partial \ell(\boldsymbol{\beta}, \phi; \mathsf{y})}{\partial \boldsymbol{\beta}} = \mathsf{0},$$

which can be expressed as:

$$\mathsf{U}(\boldsymbol{\beta}) = \sum_{i=1}^{n} \frac{(y_i - \mu_i)}{a(\phi)V(\mu_i)} \cdot \frac{\partial \mu_i}{\partial \eta_i} \cdot \mathsf{x}_i = \mathsf{0},$$

where $V(\mu_i) = b''(\theta_i)$ is the variance function and $\frac{\partial \mu_i}{\partial \eta_i} = \frac{1}{g'(\mu_i)}$ is the derivative of the inverse link function.

Since closed-form solutions generally do not exist for GLMs, the maximum likelihood estimates are computed iteratively using the Fisher scoring algorithm or the iteratively reweighted least squares (IRLS) method. The Fisher scoring algorithm updates the parameter estimates according to:

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + [\mathcal{I}(\boldsymbol{\beta}^{(t)})]^{-1}\mathsf{U}(\boldsymbol{\beta}^{(t)}),$$

where $\mathcal{I}(\boldsymbol{\beta})$ is the Fisher information matrix defined as:

$$\mathcal{I}(\boldsymbol{\beta}) = \mathbb{E}\left[-\frac{\partial^2 \ell(\boldsymbol{\beta}, \phi; \mathsf{y})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T}\right] = \sum_{i=1}^{n} \frac{1}{a(\phi)V(\mu_i)} \left(\frac{\partial \mu_i}{\partial \eta_i}\right)^2 \mathsf{x}_i \mathsf{x}_i^T.$$

Equivalently, the IRLS algorithm can be expressed as a weighted least squares problem at each iteration:

$$\boldsymbol{\beta}^{(t+1)} = (\mathsf{X}^T \mathsf{W}^{(t)} \mathsf{X})^{-1} \mathsf{X}^T \mathsf{W}^{(t)} \mathsf{z}^{(t)},$$

where X is the $n \times (k+1)$ design matrix, $\mathsf{W}^{(t)}$ is a diagonal weight matrix with elements:

$$w_i^{(t)} = \frac{1}{a(\phi)V(\mu_i^{(t)})} \left(\frac{\partial \mu_i^{(t)}}{\partial \eta_i^{(t)}}\right)^2,$$

and $\mathbf{z}^{(t)}$ is the adjusted dependent variable (working response) with elements:

$$z_i^{(t)} = \eta_i^{(t)} + (y_i - \mu_i^{(t)}) \frac{\partial \eta_i^{(t)}}{\partial \mu_i^{(t)}}.$$

The iteration continues until convergence is achieved, typically when $||\boldsymbol{\beta}^{(t+1)} - \boldsymbol{\beta}^{(t)}|| < \epsilon$ for some predetermined tolerance level $\epsilon > 0$.

The asymptotic distribution of the maximum likelihood estimator is:

$$\hat{\boldsymbol{\beta}} \sim \mathcal{N}\left(\boldsymbol{\beta}, [\mathcal{I}(\boldsymbol{\beta})]^{-1}\right),$$

which provides the basis for constructing confidence intervals and hypothesis tests for the model parameters.

2.3 Double Generalized Linear Model

The double generalized linear model (DGLM) is a comprehensive and computationally efficient approach that enables the detection of variance effects while accounting for mean effects. It comprises two components: a generalized linear model for the mean and a generalized linear sub-model for the dispersion (residual variance). Unlike a Generalized Linear Model (GLM), the DGLM allows the dispersion to be dependent on its own linear predictor. The dispersion model can incorporate fixed effects for genotype and relevant covariates associated with the trait. Consequently, a DGLM facilitates the simultaneous estimation of the genotype's impact on both the mean value and the residual variance of a quantitative trait. By including genotype as a fixed effect in the model for residual variance, the DGLM enables the assessment of whether genotype significantly influences variability around the mean trait value (Paula, 2013).

To achieve a simultaneous estimation of mean and variance effects, the DGLM employs an iterative process, iteratively refining the models until convergence. This approach incorporates adjustments for estimation uncertainty at each iteration. The DGLMs are defined by

$$g(\mu_i) = \eta_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_k X_{ik}$$

and

$$h(\phi_i) = \lambda_i = \gamma_0 + \gamma_1 Z_{i1} + \gamma_2 Z_{i2} + \ldots + \gamma_m Z_{im},$$

where β_i and γ_i are the model parameters to be estimated,X_{ij} and Z_{ij}, $j = 1, 2, \ldots, k$, contain values of explanatory variables, ϕ_i is the dispersion parameter, and $g(\cdot)$ and $h(\cdot)$ are the link functions. Various methods, such as maximum likelihood estimation and restricted maximum likelihood estimation, can be employed to fit the DGLMs.

2.3.1 Parameter Estimation for Double Generalized Linear Models

The estimation of parameters in Double Generalized Linear Models (DGLMs) involves simultaneously modeling both the mean and dispersion structures. Given the coupled nature of these two components, specialized estimation procedures are required to obtain consistent and efficient parameter estimates.

Maximum Likelihood Estimation For DGLMs, the log-likelihood function based on the exponential dispersion family can be written as:

$$\ell(\boldsymbol{\beta}, \boldsymbol{\gamma}; \mathbf{y}) = \sum_{i=1}^{n} \left[\frac{y_i \theta_i - b(\theta_i)}{a(\phi_i)} + c(y_i; \phi_i) \right],$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_k)^T$ are the mean model parameters and $\boldsymbol{\gamma} = (\gamma_0, \gamma_1, \ldots, \gamma_k)^T$ are the dispersion model parameters. The relationships $\mu_i = b'(\theta_i)$, $\eta_i = g(\mu_i) = \mathbf{x}_i^T \boldsymbol{\beta}$, and $\lambda_i = h(\phi_i) = \mathbf{z}_i^T \boldsymbol{\gamma}$ establish the connections between the canonical parameters and the linear predictors. The maximum likelihood estimates $(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})$ are obtained by solving the system of score equations:

$$\frac{\partial \ell}{\partial \boldsymbol{\beta}} = \mathbf{0} \quad \text{and} \quad \frac{\partial \ell}{\partial \boldsymbol{\gamma}} = \mathbf{0}.$$

However, direct maximization of the joint likelihood presents computational challenges due to the interdependence between $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. An iterative estimation procedure is employed, alternating between updating the mean and dispersion parameters until convergence (Smyth, 1989; Lee and Nelder, 1996).

Iterative Fitting Algorithm The estimation procedure follows an iterative scheme:

1. Initialize $\phi_i^{(0)}$ for all $i = 1, 2, \ldots, n$ (typically $\phi_i^{(0)} = 1$).

2. At iteration t:

   (a) Fix $\phi_i = \phi_i^{(t-1)}$ and estimate $\boldsymbol{\beta}^{(t)}$ by solving:

   $$\sum_{i=1}^{n} \frac{(y_i - \mu_i)}{a(\phi_i^{(t-1)})V(\mu_i)} \cdot \frac{\partial \mu_i}{\partial \eta_i} \cdot \mathbf{x}_i = \mathbf{0}. \tag{2.3}$$

   (b) Using $\boldsymbol{\beta}^{(t)}$, compute adjusted responses:

   $$d_i^{(t)} = \frac{(y_i - \mu_i^{(t)})^2}{V(\mu_i^{(t)})^2} + \frac{V'(\mu_i^{(t)})}{V(\mu_i^{(t)})}(y_i - \mu_i^{(t)}), \tag{2.4}$$

   where $V'(\mu_i) = \frac{\partial V(\mu_i)}{\partial \mu_i}$.

   (c) Estimate $\boldsymbol{\gamma}^{(t)}$ by fitting a GLM with response $d_i^{(t)}$ and linear predictor $\lambda_i = \mathbf{z}_i^T \boldsymbol{\gamma}$:

   $$\sum_{i=1}^{n} w_i^{(t)} (d_i^{(t)} - a(\phi_i^{(t)})) \frac{\partial \phi_i^{(t)}}{\partial \lambda_i^{(t)}} \mathbf{z}_i = \mathbf{0}, \tag{2.5}$$

   where $w_i^{(t)}$ are appropriate weights.

3. Repeat step 2 until convergence: $\|\boldsymbol{\beta}^{(t)} - \boldsymbol{\beta}^{(t-1)}\| < \epsilon_1$ and $\|\boldsymbol{\gamma}^{(t)} - \boldsymbol{\gamma}^{(t-1)}\| < \epsilon_2$.

Restricted Maximum Likelihood Estimation A critical limitation of maximum likelihood estimation in DGLMs is that the dispersion parameter estimates can be biased, particularly in small samples. This bias arises because ML estimation does not account for the loss of degrees of freedom due to estimating the mean parameters (Patterson and Thompson, 1971; Harville, 1977). Restricted Maximum Likelihood (REML) addresses this bias by partitioning the likelihood into two components: one that depends on $\boldsymbol{\beta}$ and one that does not. The REML estimator maximizes only the portion of the likelihood that is invariant to $\boldsymbol{\beta}$, thereby providing less biased estimates of the dispersion parameters. The REML log-likelihood for the dispersion parameters can be expressed as

$$\ell_R(\boldsymbol{\gamma}; \mathbf{y}) = \ell(\hat{\boldsymbol{\beta}}, \boldsymbol{\gamma}; \mathbf{y}) - \frac{1}{2} \log |\mathbf{X}^T \mathbf{W} \mathbf{X}|, \tag{2.6}$$

where $\hat{\boldsymbol{\beta}}$ is the ML estimate conditional on $\boldsymbol{\gamma}$, $\mathbf{W}$ is the weight matrix from the mean model, and $|\cdot|$ denotes the determinant. The second term adjusts for the uncertainty in estimating $\boldsymbol{\beta}$. The REML estimates $\hat{\boldsymbol{\gamma}}_R$ are obtained by maximizing $\ell_R(\boldsymbol{\gamma}; \mathbf{y})$:

$$\frac{\partial \ell_R(\boldsymbol{\gamma}; \mathbf{y})}{\partial \boldsymbol{\gamma}} = \mathbf{0}. \tag{2.7}$$

For DGLMs, the REML procedure modifies the iterative algorithm by incorporating the adjustment term in the dispersion model estimation step. Specifically, the adjusted responses in Eq. (2.4) are replaced by REML-adjusted responses that account for the degrees of freedom correction (Smyth and Verbyla, 1999):

$$d_i^{R,(t)} = d_i^{(t)} + \text{tr}\left[(\mathbf{X}^T \mathbf{W}^{(t)} \mathbf{X})^{-1} \frac{\partial (\mathbf{X}^T \mathbf{W}^{(t)} \mathbf{X})}{\partial \phi_i} \right], \tag{2.8}$$

where $\text{tr}(\cdot)$ denotes the trace operator.

Asymptotic Properties Under regularity conditions, the REML estimators have desirable asymptotic properties. As $n \to \infty$:

$$\hat{\boldsymbol{\gamma}}_R \sim \mathcal{N}\left(\boldsymbol{\gamma}, [\mathcal{I}_R(\boldsymbol{\gamma})]^{-1}\right),$$

where $\mathcal{I}_R(\boldsymbol{\gamma})$ is the Fisher information matrix for the dispersion parameters under REML. The REML approach generally provides estimates with smaller mean squared error than ML, particularly for variance components, making it the preferred method for fitting DGLMs in practice (Smyth and Verbyla, 1999; Lee et al., 2006).

The choice between ML and REML estimation depends on the specific application and sample size. While REML is preferred for dispersion parameter estimation, ML may be more appropriate when comparing models with different fixed effects structures, as REML estimates are not directly comparable across models with different mean structures.

2.4 Generalized Additive Model

A generalized additive model (GAM) is a statistical modeling technique that extends the concept of generalized linear models (GLMs) by allowing for nonlinear relationships between predictors and the response variable. Unlike GLMs, which assume linear relationships, GAMs can capture complex and nonlinear patterns by incorporating smooth functions of the predictors (Wood, 2017). In a GAM, the response variable is still related to the predictors through a specified probability distribution and a link function, similar to GLMs. However, instead of assuming a linear relationship, GAMs employ smoothing functions, such as splines or wavelets, to model nonlinearities and interactions. By allowing for flexible and nonparametric modeling, GAMs can handle complex data structures and capture intricate relationships between predictors and the response. They are particularly useful when dealing with data that exhibit nonlinear patterns, such as time series, spatial data, and interactions between variables. Instead of using the linear form $\Sigma\beta_i X_i$, GAM replaces it with a sum of smooth functions $\Sigma(X_i)$ (Hastie and Tibshirani, 1986). The general structure of the model can be described as follows:

$$g(\mu_i) = A_i\theta + f_1(x_{1i}) + f_2(x_{2i}) + \ldots + f_k(x_{ki}),$$

where $g(\mu_i) = \mathbb{E}(Y_i)$ and response variable Y_i follows a conditional distribution from the exponential family distribution with mean μ_i and scale parameterϕ. A_i is an explanatory variable vector for strictly parametric components, is the equivalent parameter vector, and f_i are smooth functions of non-parametric model variables, x_i for $i = 1, 2, \ldots, k$.

2.4.1 Penalized Estimation

The estimation of smooth functions in Generalized Additive Models requires balancing two competing objectives: achieving good fit to the data while maintaining appropriate smoothness to avoid overfitting. This trade-off is formalized through penalized likelihood estimation, which combines a measure of fit with a penalty for excessive roughness (Wood, 2017).

Penalized Likelihood Framework For a GAM with response variable Y_i following an exponential family distribution, the penalized log-likelihood is given by:

$$\ell_p(\boldsymbol{\theta}, \mathsf{f}) = \ell(\boldsymbol{\theta}, \mathsf{f}) - \frac{1}{2}\sum_{j=1}^{k} \lambda_j \int [f_j''(x)]^2 dx,$$

where $\ell(\boldsymbol{\theta}, \mathsf{f})$ is the log-likelihood function:

$$\ell(\boldsymbol{\theta}, \mathsf{f}) = \sum_{i=1}^{n} \left[\frac{y_i \eta_i - b(\eta_i)}{\phi} + c(y_i; \phi) \right],$$

with $\eta_i = A_i\boldsymbol{\theta} + \sum_{j=1}^{k} f_j(x_{ji})$, and the penalty term $\int [f_j''(x)]^2 dx$ measures the roughness of the j-th smooth function through its second derivative. The parameters $\lambda_j \geq 0$ are smoothing parameters that control the trade-off between fit and smoothness for each smooth function f_j.

Basis Function Representation In practice, each smooth function f_j is represented using a basis function expansion:

$$f_j(x) = \sum_{m=1}^{M_j} b_{jm}(x)\beta_{jm} = \mathsf{b}_j(x)^T \boldsymbol{\beta}_j,$$

where $\{b_{j1}(x), b_{j2}(x), \ldots, b_{jM_j}(x)\}$ are basis functions (typically B-splines or thin plate splines), $\boldsymbol{\beta}_j = (\beta_{j1}, \beta_{j2}, \ldots, \beta_{jM_j})^T$ are the corresponding coefficients, and M_j is the dimension of the basis for the j-th smooth term.

The roughness penalty can then be expressed in matrix form as:

$$\int [f_j''(x)]^2 dx = \boldsymbol{\beta}_j^T \mathsf{S}_j \boldsymbol{\beta}_j,$$

where S_j is a positive semi-definite penalty matrix with elements:

$$[\mathsf{S}_j]_{mn} = \int b_{jm}''(x) b_{jn}''(x) dx.$$

The complete penalized log-likelihood becomes:

$$\ell_p(\boldsymbol{\theta}, \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_k) = \ell(\boldsymbol{\theta}, \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_k) - \frac{1}{2} \sum_{j=1}^{k} \lambda_j \boldsymbol{\beta}_j^T \mathsf{S}_j \boldsymbol{\beta}_j. \quad (2.9)$$

Penalized Iteratively Reweighted Least Squares The penalized maximum likelihood estimates are obtained by maximizing Eq. (2.9). Let $\boldsymbol{\beta} = (\boldsymbol{\theta}^T, \boldsymbol{\beta}_1^T, \ldots, \boldsymbol{\beta}_k^T)^T$ denote the complete parameter vector. The estimation is performed using a penalized iteratively reweighted least squares (P-IRLS) algorithm (Wood, 2017).

At iteration t, given $\boldsymbol{\beta}^{(t)}$, compute $\boldsymbol{b}^{(t)}$ by

$$\boldsymbol{\beta}^{(t+1)} = (\mathsf{X}^T \mathsf{W}^{(t)} \mathsf{X} + \mathsf{S}_\lambda)^{-1} \mathsf{X}^T \mathsf{W}^{(t)} \mathsf{z}^{(t)},$$

where:

- X is the model matrix with columns corresponding to the parametric terms and basis function evaluations,
- $\mathsf{W}^{(t)}$ is the diagonal weight matrix with elements $w_i^{(t)} = \frac{1}{\phi V(\mu_i^{(t)})} [g'(\mu_i^{(t)})]^{-2}$,
- $\mathsf{z}^{(t)}$ is the working response vector with elements $z_i^{(t)} = \eta_i^{(t)} + (y_i - \mu_i^{(t)}) g'(\mu_i^{(t)})$,
- $\mathsf{S}_\lambda = \text{diag}(\mathbf{0}, \lambda_1 \mathsf{S}_1, \lambda_2 \mathsf{S}_2, \ldots, \lambda_k \mathsf{S}_k)$ is the block-diagonal penalty matrix.

Smoothing Parameter Selection The choice of smoothing parameters $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_k)$ is crucial for achieving the optimal balance between fit and smoothness. Generalized Cross-Validation (GCV) is commonly employed for automatic selection (Wood, 2006):

$$\mathsf{GCV}(\boldsymbol{\lambda}) = \frac{n\mathcal{D}(\boldsymbol{\lambda})}{(n - \mathsf{tr}(\mathsf{A}(\boldsymbol{\lambda})))^2},$$

where $\mathcal{D}(\boldsymbol{\lambda})$ is the deviance and $\mathsf{A}(\boldsymbol{\lambda}) = \mathsf{X}(\mathsf{X}^T\mathsf{W}\mathsf{X} + \mathsf{S}_\lambda)^{-1}\mathsf{X}^T\mathsf{W}$ is the influence matrix. The term $\mathsf{tr}(\mathsf{A}(\boldsymbol{\lambda}))$ represents the effective degrees of freedom.

Alternatively, Restricted Maximum Likelihood (REML) treats the smoothing parameters as variance components (Wood, 2011):

$$\mathsf{REML}(\boldsymbol{\lambda}) = -\frac{1}{2}\left[\log|\mathsf{X}^T\mathsf{W}\mathsf{X} + \mathsf{S}_\lambda| + \log|\mathsf{S}_\lambda^-| + \mathbf{z}^T(\mathsf{W} - \mathsf{W}\mathsf{X}\mathsf{C}\mathsf{X}^T\mathsf{W})\mathbf{z}\right],$$

where $\mathsf{C} = (\mathsf{X}^T\mathsf{W}\mathsf{X} + \mathsf{S}_\lambda)^{-1}$ and S_λ^- denotes the generalized inverse.

The optimal smoothing parameters are obtained by minimizing GCV or maximizing REML:

$$\hat{\boldsymbol{\lambda}} = \arg\min_{\boldsymbol{\lambda}} \mathsf{GCV}(\boldsymbol{\lambda}) \quad \text{or} \quad \hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \mathsf{REML}(\boldsymbol{\lambda}).$$

Effective Degrees of Freedom The effective degrees of freedom (EDF) provides a measure of the complexity of each smooth term (Hastie and Tibshirani, 1990). For the j-th smooth function:

$$\mathsf{EDF}_j = \mathsf{tr}(\mathsf{F}_j),$$

where F_j is the submatrix of $\mathsf{A}(\boldsymbol{\lambda})$ corresponding to the j-th smooth term. When $\lambda_j = 0$ (no penalty), $\mathsf{EDF}_j = M_j$ (fully flexible), and as $\lambda_j \to \infty$, $\mathsf{EDF}_j \to 0$ (linear or constant term).

The penalized estimation framework thus provides a principled approach to controlling model complexity, ensuring that GAMs achieve an optimal balance between capturing genuine patterns in the data and avoiding overfitting.

2.5 Methods for Parameter Estimation

In the construction of statistical models, it is often necessary to employ efficient methods for estimating model parameters. This section presents parameter estimation using classical approaches, as well as more recent methods derived from machine learning techniques.

2.5.1 Smoothing Splines

Chouldechova and Hastie (2015) demonstrate for instance the univariate regression condition where we observe data on a single predictor x and a response variable y. On the real line, one assumes that x has values that occur in some compact interval. A choice of $\lambda \geq 0$ and observations $(y_1, x_1), \ldots, (y_n, x_n)$, the smoothing spline estimator, $\hat{f}_\lambda$, is defined by,

$$\hat{f} = \underset{f \in C^2}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \int [f''(x)]^2 dx$$

Although the uncountably infinite parameter space C^2 may make this optimization problem seem challenging at first, it can be demonstrated that the problem can be reduced to a much simpler form. A natural cubic spline with knots at the specific x values is the answer. The smoothing spline objective can be expressed as a finite dimensional problem because it can be demonstrated that the space of such splines is n-dimensional.

2.5.2 Explainable Boosting Machine

The Explainable Boosting Machine (EBM), another interpretability algorithm, is an element of the InterpretML framework. EBM is a transparent model with an emphasis on interpretability and explainability with the intent to achieve accuracy comparable to that of more sophisticated machine learning methods like Random Forest and Boosted Trees. It adopts a Generalized Additive Model (GAM) structure and is represented by the form:

$$g(E[y]) = \beta_0 + \sum f_j(x_j),$$

The link function $g(\cdot)$ adapts the Generalized Additive Model (GAM) to different contexts, such as regression or classification. The Explainable Boosting Machine (EBM) incorporates several notable advancements compared to traditional GAMs. Firstly, EBM utilizes modern machine learning techniques like bagging and gradient boosting to learn each feature function f_j. The boosting procedure is carefully constrained to train on one feature at a time in a round-robin fashion with a very low learning rate.

This approach ensures that the order of features does not impact the results. By cycling through features in a round-robin manner, EBM mitigates the effects of co-linearity and learns the optimal feature function f_j for each feature, revealing how each feature contributes to the model's prediction. Secondly, EBM has the capability to automatically identify and incorporate pairwise interaction terms. These terms capture interactions between pairs of features and are expressed in the following form:

$$g(\mathbb{E}[y]) = \beta_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{1 \leq i < j \leq p} f_{ij}(x_i, x_j).$$

The most significant of the primary benefits of EBM is its ease of comprehension. By plotting the corresponding feature function f_j, the contribution of each feature to the final prediction can be visualized and easily understood. The additive model structure of EBM ensures that each feature contributes to the predictions in a modular manner, making it easier to interpret the impact of each feature on the overall prediction.

2.5.3 XGBoosT

Extreme gradient boosting (XGBoost) is an algorithm that combines weak learners, typically regression trees, to create a strong predictive model (Chen and Guestrin, 2016). Boosting is the sequential ensemble of trees, where new trees are built based on residuals from previous trees, focusing on the unexplained variance in the training set. The algorithm adds new trees until reaching the maximum number or when they do not significantly improve predictive power. At the beginning of the algorithm, each observation is predicted as the mean of all response variables with equal weights. These weights are adjusted throughout each iteration, increasing for observations that are poorly predicted to force the algorithm's attention to them.

The fitting process of XGBoost involves sequentially tuning a series of model parameters to achieve the most accurate model, which sets the maximum number of trees, is tested for multiple values in each tuning iteration. The algorithm returns the optimal combination of input parameters by minimizing the Loss function,

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k),$$

where

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2$$

denoted as $\mathcal{L}(\phi)$is the estimated predictive loss (l), provides a second-degree approximation of the residuals. It quantifies the discrepancy between the predicted and actual values. To prevent overfitting and promote simpler models, a penalty function $\Omega(f)$ is applied. This penalty function aims to control the complexity of the model. One specific aspect that is penalized is the number of leaf nodes (T) in a tree, and this is achieved through the Gamma (γ) parameter (Harvri, 2019).

2.5.4 Random Forest

The random forest algorithm, introduced by Breiman (2001), is a powerful ensemble method that improves classification accuracy by combining multiple decision trees. A random forest consists of a collection of tree-based classifiers, denoted as$f(\cdot, \Theta_k) : k \geq 1$, where Θ_k represents independent and identically distributed random vectors. The classification process involves a majority voting scheme, where each tree casts a vote, $f(x, \Theta_k)$, for a test point x, and the class with the most votes is chosen. Here is a step-by-step outline of the random forest algorithm:

1. Input: The algorithm takes a dataset $S = \{(Y^1, X^1), \ldots, (Y^n, X^n)\}$, where Y belongs to the set $\{1, \ldots, J\}$ and $X \in \mathbb{R}^{|}$ is in p-dimensional space. Additionally, specify the number of trees K, positive integers d (much smaller than p and x).

2. For each k from 1 to K:

(a) Create a bootstrap sample S_k by randomly selecting data points from the training set S.

(b) At each node of the tree, randomly select d input variables from $X_1, \ldots, X_p$. Based on these variables and S_k, determine the best split to partition the data.

(c) Grow the tree until each terminal node contains no more than ℓ training samples. No pruning is performed. Let $f(\cdot, S_k)$ represent the constructed tree.

3. Output: The random forest classifier, denoted as $f_{\text{RandFrst}}(x, S)$, is defined as the class that has the highest sum of votes over K trees, i.e.

$$\underset{1 \leq k \leq K}{\text{argmax}} \sum_{K} I[f(x, S_k)].$$

In summary, the random forest algorithm combines the predictions of multiple decision trees to make accurate classifications. Each tree is trained on a bootstrap sample and selects random subsets of input variables, resulting in a diverse ensemble of trees. The final classification is determined by majority voting among the individual tree predictions (Shim and Lee, 2011).

2.5.5 Support Vector Regression

Support Vector Regression (SVR), introduced by Vapnik (1995), is a powerful machine learning algorithm based on statistical learning theory that extends the principles of Support Vector Machines to regression problems. SVR aims to find a function that approximates the relationship between input features and continuous output values while maintaining a balance between model complexity and prediction accuracy (Smola and Schölkopf, 2004).

Unlike traditional regression methods that minimize the sum of squared errors, SVR employs an ε-insensitive loss function that ignores errors smaller than a threshold ε. This approach creates a margin of tolerance around the regression function, focusing computational effort on predictions that deviate significantly from observed values. The algorithm identifies support vectors data points that lie outside or on the boundary of this ε-tube which are critical in determining the regression function.

The SVR optimization problem can be formulated as follows. Given a training dataset $\{(x_i, y_i)\}_{i=1}^n$ where $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$, SVR seeks to find a function:

$$f(x) = \langle w, \phi(x) \rangle + b,$$

where w is the weight vector, $\phi(x)$ maps input features to a higher-dimensional feature space, and b is the bias term. The optimization objective is:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*),$$

subject to the constraints:

$$\begin{aligned} y_i - \langle w, \phi(x_i) \rangle - b &\leq \varepsilon + \xi_i, \\ \langle w, \phi(x_i) \rangle + b - y_i &\leq \varepsilon + \xi_i^*, \\ \xi_i, \xi_i^* &\geq 0, \end{aligned}$$

where $C > 0$ is a regularization parameter that controls the trade-off between model flatness and tolerance for deviations larger than ε, and ξ_i, ξ_i^* are slack variables that allow some points to fall outside the ε-tube.

The dual formulation of this optimization problem involves Lagrange multipliers α_i and α_i^*, leading to the prediction function:

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K(x_i, x) + b,$$

where $K(x_i, x) = \langle \phi(x_i), \phi(x) \rangle$ is a kernel function. Common kernel choices include:

- Linear kernel: $K(x_i, x_j) = \langle x_i, x_j \rangle$
- Polynomial kernel: $K(x_i, x_j) = (\gamma\langle x_i, x_j \rangle + r)^d$ (degree d, i.e., the polynomial order)
- Radial Basis Function (RBF) kernel: $K(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|^2)$
- Sigmoid kernel: $K(x_i, x_j) = \tanh(\gamma\langle x_i, x_j \rangle + r)$.

The kernel function allows SVR to capture nonlinear relationships by implicitly mapping data to higher-dimensional spaces without explicitly computing the transformation $\phi(x)$, a technique known as the "kernel trick."

The fitting process of SVR involves tuning several hyperparameters to achieve optimal performance. The key hyperparameters include:

- C: The regularization parameter that balances model complexity and training error
- ε: The width of the ε-insensitive tube
- Kernel-specific parameters: γ (for RBF and polynomial kernels), d (degree for polynomial kernel), r (coefficient for polynomial and sigmoid kernels)

2.6 Model Evaluation

Model evaluation is a critical component in the development and validation of statistical and machine learning models. For Generalized Additive Models (GAM) and Double GAM (DGAM), which estimate both the conditional mean and variance, comprehensive assessment requires metrics that evaluate point prediction accuracy, model complexity, and uncertainty quantification. The following metrics provide a robust framework for evaluating model performance from multiple perspectives.

2.6.1 Point Prediction

Several metrics are employed to evaluate point prediction performance. For all metrics, y_i denotes the observed value, $\hat{y}_i$ denotes the predicted value, and N denotes the total number of observations.

Mean Absolute Error

Mean Absolute Error (MAE) provides a straightforward measure of average prediction error by calculating the mean of absolute differences between predicted and observed values:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|.$$

Root Mean Squared Error

Root Mean Squared Error ($RMSE$) measures prediction accuracy by taking the square root of the average squared error:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}.$$

A smaller value for either metric signifies more accurate predictions. $RMSE$ is particularly sensitive to large errors, making it well suited for evaluating mean prediction performance in GAM and DGAM models (James et al., 2013).

In contrast, the Mean Absolute Error (MAE) assigns equal weight to all prediction errors regardless of their magnitude, providing a more robust and interpretable measure of average prediction accuracy, especially in the presence of outliers.

Coefficient of Determination

The Coefficient of Determination, or R^2, is a statistical measure that represents the proportion of variance in the dependent variable that is explained by the independent variables in the regression model. It is calculated as:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2},$$

where SS_{res} is the residual sum of squares, SS_{tot} is the total sum of squares, y_i is the observed value, $\hat{y}_i$ is the predicted value, and $\bar{y}$ is the mean of observed values. The R^2 value ranges from 0 to 1, where a value closer to 1 indicates that the model explains a larger proportion of the variance. For GAM and DGAM, R^2 provides an intuitive measure of how well the smooth functions capture the underlying relationship in the data (James et al., 2013).

2.6.2 Likelihood Based

Akaike Information Criterion

The Akaike Information Criterion, or AIC, is a model selection criterion that balances the goodness of fit with model complexity. It estimates the relative amount of information lost by a given model by dealing with the trade-off between overfitting and underfitting. The AIC is calculated using the equation:

$$AIC = -2\ln(\hat{L}) + 2k,$$

where $\hat{L}$ is the maximum value of the likelihood function for the model, and k is the number of estimated parameters. A smaller AIC value indicates a better model, as it suggests less information loss. This criterion is particularly important for GAM and DGAM, where the selection of smooth functions and the effective degrees of freedom directly affect model complexity (Akaike, 1974).

Negative Log-Likelihood

The Negative Log-Likelihood, or NLL, is a fundamental loss function derived from maximum likelihood estimation that quantifies how well a probabilistic model fits the observed data. For a dataset with N observations, the NLL is defined as:

$$NLL = -\sum_{i=1}^{N} \log p(y_i \mid x_i; \theta),$$

where $p(y_i \mid x_i; \theta)$ represents the predicted probability density of the observed value y_i given the input x_i and model parameters θ. Minimizing NLL is equivalent to maximizing the likelihood of observing the data under the model. For distributional regression models such as DGAM, which estimate both location and scale parameters, NLL serves as the primary training objective and evaluation metric (Hastie and Tibshirani, 1990).

In this thesis, we focus on the Gaussian negative log-likelihood, which corresponds to the NLL obtained under the assumption that the response variable follows a normal distribution.

2.6.3 Probabilistic Forecasts

Continuous Ranked Probability Score

The Continuous Ranked Probability Score, or $CRPS$, is a strictly proper scoring rule used to evaluate probabilistic forecasts. It measures the integrated squared difference between the predicted cumulative distribution function and the empirical cumulative distribution function of the observation. The $CRPS$ is defined as:

$$CRPS(F, y) = \int_{-\infty}^{\infty} [F(x) - \mathbf{1}(x \geq y)]^2 \, dx,$$

where F is the predicted cumulative distribution function, y is the observed value, and $\mathbf{1}(x \geq y)$ is the indicator function that equals 1 if $x \geq y$ and 0 otherwise. A smaller $CRPS$ value indicates a better probabilistic forecast. The $CRPS$ generalizes the Mean Absolute Error to probabilistic predictions and is particularly important for DGAM, which outputs a full predictive distribution rather than a single point estimate (Gneiting and Raftery, 2007).

Prediction Interval Coverage Probability at 95%

The Prediction Interval Coverage Probability at 95%, or $PICP_{95}$, measures the reliability of uncertainty estimates by calculating the proportion of observed values that fall within the 95% prediction interval. It is defined as:

$$PICP_{95} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}(L_i \leq y_i \leq U_i),$$

where L_i and U_i are the lower and upper bounds of the 95% prediction interval for observation i, y_i is the observed value, and $\mathbf{1}(\cdot)$ is the indicator function. For a well-calibrated model, $PICP_{95}$ should be close to 0.95. This metric is essential for evaluating DGAM, as it directly assesses whether the estimated variance component produces reliable prediction intervals. A $PICP_{95}$ significantly below 0.95 indicates overconfident predictions, while a value significantly above 0.95 suggests overly conservative uncertainty estimates (Kuleshov et al., 2018).

2.7 Related Researches

Gijbels and Prosdocimi (2012) address the problem of real data showing a larger or smaller variability than what is assumed in exponential family models, which are the basis of Generalized Linear Models (GLMs) and additive models, in their research. They address this issue by introducing extended generalized additive models with P-spline smooth estimation techniques for the mean and the dispersion function. Using this methodology as a foundation, the authors broaden their analysis by allowing some covariates to be modeled parametrically while keeping others nonparametric. The article's main contribution is a simulation study that looks at how well the P-spline estimation method performs in these extended models. This study includes comparisons with a standard generalized additive modeling approach and a hierarchical modeling approach, providing insights into the finite-sample behavior of the proposed technique.

Croux, Gijbels, and Prosdocimi (2012) examined how Generalized Linear Models (GLMs) are frequently used to generate parametric estimates for the mean function in their study. They introduce the idea of generalized additive models (GAMs) to increase the adaptability of the relationship between the mean function and the covariates. They discover, however, that the fixed variance structure imposed by GLMs can frequently be overly restrictive. The researchers suggest using the Extended Quasilikelihood (EQL) framework, which enables the estimation of both the mean and the dispersion (variance) as functions of the covariates, in order to overcome this limitation. However, they point out that EQL estimates are susceptible to the impact of outliers, just like other maximum likelihood techniques. Therefore, there is a need for robust estimation methods that can produce reliable estimates for both the mean and the dispersion functions.

Lou, Caruana, and Gehrke (2013) investigated common Generalized Additive Models (GAMs) that are frequently used to model the dependent variable as a sum of univariate models in their study. Although earlier studies have shown that these common GAMs can be understood, they are less accurate than more intricate models that take interactions into account. By incorporating specific terms of interacting pairs of features, the authors of this paper suggest improving standard GAMs. The resulting models, referred to as GA2M-models (Generalized Additive Models plus Interactions), include univariate terms as well as a select few pairwise interaction terms. GA2M models can be visualized and understood by users, improving their interpretability, by limiting the model components to one- and two-dimensional components.

Harvei, Jørgensen, and Ådland (2019) studied comparing XGBoost and GAM in modeling complex relationships among multiple variables, aligning with prior research in maritime economics that highlights the limitations of linear models for vessel valuation. The study identifies vessel age at sale, timecharter rates, and fuel efficiency index as the key variables in the XGBoost model. Additionally, predicting the prices of vessels priced over 20 million poses significant challenges, potentially due to limited data, unaccounted vessel characteristics, or the influence of the financial super cycle between 2003 and 2009. Overall, the XGBoost algorithm enables accurate predictions of vessel prices based on vessel characteristics and market conditions, serving as a valuable machine learning framework for desktop valuation. Its flexibility makes it highly applicable for investors, ship owners, and other stakeholders in the maritime industry.

Chang, Tan, Lengerich, Goldenberg, and Caruana (2021) ask whether Generalized Additive Models (GAMs) are truly interpretable and trustworthy. While GAMs have grown in popularity as model class for interpretable machine learning, the existence of multiple algorithms for training GAMs presents a challenge: these algorithms can learn different, even contradictory models while achieving the same level of accuracy. As a result, determining which GAM should be trusted becomes critical. The authors extensively investigate various GAM algorithms using quantitative and qualitative analyses of real and simulated datasets. They find that GAMs with high feature sparsity, utilizing only a few variables for predictions, can miss important patterns and exhibit unfairness towards rare subpopulations. The study highlights the crucial role of inductive bias in shaping interpretable models and concludes that tree-based GAMs achieve the best balance of sparsity, fidelity, and accuracy. Thus, tree-based GAMs are deemed the most trustworthy models, offering a reliable solution for interpretable machine learning tasks.

Chang, Caruana, and Goldenberg (2022) describe Support Vector Regression (SVR) as a supervised learning technique grounded in statistical learning theory that has been widely applied across various domains, including time series prediction, financial forecasting, and healthcare applications. Unlike traditional regression models that minimize empirical risk, SVR seeks to minimize an upper bound on the generalization error by employing a loss function that defines an ϵ-insensitive tube around the regression function, within which deviations are ignored. This formulation, together with kernel functions that map input data into high-dimensional feature spaces, enables SVR to effectively capture complex nonlinear relationships while maintaining strong generalization performance. However, SVR models are often regarded as black-box methods due to their limited interpretability, which can hinder their deployment in high-risk or decision-critical environments where transparency and explainability are essential. In contrast, Generalized Additive Models (GAMs) provide an interpretable alternative by extending generalized linear models through the replacement of the linear predictor with a sum of smooth, nonparametric functions. GAMs retain the ability to learn nonlinear transformations for individual features while combining these effects additively, thereby offering modular interpretability that allows each feature's contribution to be examined independently. Despite their long history of successful application, traditional GAMs typically rely on spline-based smoothers and backfitting algorithms, whereas modern extensions such as NODE-GAM integrate the interpretability advantages of GAMs with the flexibility and scalability of deep learning approaches.

CHAPTER III

RESEARCH METHODOLOGY

This chapter presents the systematic research methodology developed to achieve the study objectives. The methodology is structured into two principal phases: **Phase I: Theoretical and Algorithmic Development**, which establishes the mathematical foundations and computational algorithms for Double Generalized Additive Models (DGAMs) enhanced through integration with advanced machine learning techniques, and **Phase II: Computational Implementation and Empirical Validation**, which translates these theoretical developments into executable code and rigorously evaluates model performance using real-world health data spanning critical life-stage transitions.

The research is grounded in a fundamental conceptual framework rooted in Buddhist philosophy and universal human experience: birth, aging, illness, and death. This framework motivates the selection of health datasets representing these four life stages, providing a comprehensive testbed for evaluating DGAM performance across diverse demographic and clinical contexts.

3.1 Phase I: Theoretical and Algorithmic Development

This phase focuses on the mathematical formulation and algorithmic specification of Double Generalized Additive Models (DGAMs) integrated with five machine learning techniques: Smoothing Splines, Explainable Boosting Machines (EBM), Extreme Gradient Boosting (XGBoost), Random Forests (RF), and Support Vector Regression (SVR).

3.1.1 Conceptual Foundation: From Classical GAMs to DGAMs

Limitations of Classical GAMs

In the context of Generalized Linear Models (GLMs) and Generalized Additive Models (GAMs), the response variable Y, conditional on covariates $X_d = x_d$ (where

$X_d = (X_1, \ldots, X_d)^\top$ denotes the d-dimensional covariate vector and x_d is its realized value for a given observation), is typically assumed to follow a distribution from the one-parameter exponential family. Under this framework, the relationship between the mean μ and variance of Y is fully specified:

$$\mathrm{Var}(Y \mid X_d = x_d) = \phi\, b''\big(\theta(x_d)\big), \tag{3.1}$$

where ϕ denotes a constant dispersion parameter and $b(\cdot)$ is the cumulant function of the exponential family distribution as defined in equation (2.1). While mathematically convenient, this assumption can be overly restrictive in practice. Real-world data frequently exhibit several departures from this idealized structure:

- **Overdispersion and Underdispersion:** Count or proportion data often display variance that is larger (overdispersion) or smaller (underdispersion) than that implied by the theoretical distribution.
- **Covariate-Dependent Dispersion:** The degree of variability may itself vary systematically as a function of covariates, a phenomenon not accommodated by constant ϕ.
- **Heteroscedasticity:** Continuous data modeled under normality assumptions may display non-constant variance. For the Gaussian distribution where $b(\theta) = \theta^2/2$, $b''(\theta) = 1$, and $\phi = \sigma^2$, the assumption of constant variance may inadequately reflect the variability observed in real data (Rigby and Stasinopoulos, 2005).

Traditional approaches to addressing these limitations include adopting alternative distributions with more flexible variance structures (e.g., negative binomial or gamma-Poisson models) or applying variance-stabilizing transformations. However, these approaches often require strong distributional assumptions or sacrifice interpretability.

The DGAM Framework

The Double Generalized Additive Model (DGAM) provides a principled extension of classical GAMs by allowing flexible estimation of both mean and dispersion functions. The response variable Y is assumed to follow a distribution from the exponential dispersion family (equation (2.1)). Unlike classical GAMs that assume constant dispersion, DGAMs model both the location parameter (mean, μ) and scale parameter (dispersion, ϕ) as smooth additive functions of covariates:

$$g(\mu_i) = \eta_{\mu,i} = f_\mu(\mathbf{x}_i) = \beta_0 + \sum_{j=1}^{p} f_{\mu,j}(x_{ij}),$$
$$h(\phi_i) = \eta_{\phi,i} = f_\phi(\mathbf{z}_i) = \gamma_0 + \sum_{j=1}^{q} f_{\phi,j}(z_{ij}),$$

where $g(\cdot)$ and $h(\cdot)$ are link functions, $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$ and $\mathbf{z}_i = (z_{i1}, z_{i2}, \ldots, z_{iq})^T$ are covariate vectors (which may be identical or different), and $f_{\mu,j}(\cdot)$ and $f_{\phi,j}(\cdot)$ are smooth functions estimated from the data.

For Gaussian responses with identity link for the mean and log link for the dispersion, this framework takes the form:

$$Y_i \sim \mathcal{N}(\mu_i, \sigma_i^2),$$

with:

$$\mu_i = \beta_0 + \sum_{j=1}^{p} f_{\mu,j}(x_{ij}),$$
$$\log(\sigma_i^2) = \gamma_0 + \sum_{j=1}^{q} f_{\phi,j}(z_{ij}).$$

This dual-structure enables comprehensive characterization of how both central tendency and variability change with predictor variables, making DGAMs particularly powerful for modeling heteroscedastic data where variance depends systematically on covariates.

3.1.2 Mathematical Formulations and Algorithmic Specifications

The theoretical foundations for parameter estimation in DGAMs using each of the five machine learning techniques have been detailed in the preceding sections of this chapter. These include:

1. **Smoothing Splines:** Penalized likelihood estimation with basis function representations and coupled Penalized Iteratively Reweighted Least Squares (PIRLS) algorithm (Algorithm 1).

2. **Explainable Boosting Machine (EBM):** Alternating boosting on coupled negative log-likelihood gradients with round-robin feature selection (Algorithm 2).

3. **Extreme Gradient Boosting (XGBoost):** Alternating second-order boosting using gradient and Hessian information with regularized tree construction (Algorithm 3).

4. **Random Forest:** Alternating residual forests for mean and dispersion with bootstrap aggregation (Algorithm 4).

5. **Support Vector Regression (SVR):** Alternating optimization with ε-insensitive loss functions and kernel methods (Algorithm 5).

Each technique has been mathematically formulated to accommodate the dual structure of DGAMs, where separate but interdependent models are estimated for the mean μ and dispersion σ^2 functions.

3.1.3 Theoretical Properties and Convergence

The formal mathematical proofs of theoretical properties, including consistency, asymptotic normality, convergence guarantees, and model identifiability conditions for the proposed DGAM estimators, are presented in Chapter 4 alongside empirical results. These theoretical developments establish the statistical foundations supporting the computational implementations developed in Phase II.

3.1.4 Modeling Framework Comparison

Two distinct modeling paradigms are investigated to isolate the impact of dispersion modeling:

GAM with Constant Dispersion

The standard GAM assumes homoscedastic errors:

$$Y_i = \mu_i + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2), \tag{3.2}$$

where $\mu_i = \mu(x_i)$ is the mean function for observation i modeled using one of five techniques (Smoothing Splines, EBM, XGBoost, RF, or SVR), and σ^2 is a constant dispersion parameter (identical for all observations) estimated as:

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{\mu}_i)^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{\mu}(x_i))^2,$$

where $\hat{\mu}_i = \hat{\mu}(x_i)$ denotes the fitted mean value for observation i obtained from the trained model. This framework yields 5 distinct GAM configurations, each employing a different technique for mean estimation while maintaining constant variance across all observations.

DGAM with Flexible Dispersion

The Double Generalized Additive Model extends the GAM framework by explicitly modeling heteroscedastic errors:

$$Y_i = \mu_i + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma_i^2),$$

where both $\mu_i = \mu(x_i)$ and $\sigma_i^2 = \sigma^2(x_i)$ are modeled as flexible functions of covariates. Note that the variance σ_i^2 now varies across observations (subscript i), unlike the constant σ^2 in equation (3.2).

Following the GAM structure, the mean and dispersion functions are expressed as sums of smooth component functions:

$$\mu_i = \mu(x_i) = \beta_0 + \sum_{j=1}^{p} f_{\mu,j}(x_{ij}),$$
$$\log(\sigma_i^2) = f_\phi(x_i) = \gamma_0 + \sum_{j=1}^{q} f_{\phi,j}(z_{ij}),$$

where the dispersion is modeled on the log scale to ensure positivity ($\sigma_i^2 > 0$ for all i), and $f_{\mu,j}(\cdot)$ and $f_{\phi,j}(\cdot)$ are smooth functions estimated from the data.

The DGAM framework allows independent selection of techniques for mean and dispersion modeling, yielding $5 \times 5 = 25$ distinct configurations. This factorial design enables systematic investigation of how different technique combinations affect prediction performance and uncertainty quantification.

3.1.5 Summary of Phase I

Phase I establishes the complete theoretical and algorithmic foundation for DGAMs, including:

- Mathematical formulations for dual mean-dispersion modeling
- Five distinct parameter estimation algorithms
- Theoretical properties and convergence guarantees
- Framework for comparing GAM and DGAM approaches

These theoretical developments provide the foundation for the computational implementation and empirical validation conducted in Phase II.

3.2 Phase II: Computational Implementation and Empirical Validation

This phase translates the theoretical developments from Phase I into executable Python code and conducts comprehensive empirical evaluation using real-world health datasets. The implementation follows a systematic seven-step procedure encompassing data acquisition, preprocessing, model training, and evaluation.

3.2.1 Step 1: Data Acquisition and Preparation

Dataset Selection and Conceptual Framework

Guided by the universal life-cycle framework (birth, aging, illness, death), four health-related datasets were selected to represent distinct stages of human existence:

1. **Birth Stage Dataset:** Perinatal and neonatal health indicators
   `https://www.kaggle.com/datasets/yaminh/`
   `fetal-health-monitoring-dataset`
2. **Aging Stage Dataset:** Age-related physiological and functional measurements
   `https://www.kaggle.com/datasets/jockeroika/life-style-data`
3. **Illness Stage Dataset:** Disease progression and treatment response variables
   `https://www.kaggle.com/datasets/hongseoi/health-checkup-result`
4. **Death Stage Dataset:** Mortality risk factors and end-of-life health metrics
   `https://www.kaggle.com/datasets/lashagoch/`
   `life-expectancy-who-updated`

Each dataset contains multiple target variables, yielding a total of 16 distinct prediction tasks (4 datasets $\times$ 4 target variables per dataset). This design ensures comprehensive evaluation across diverse data characteristics and prediction contexts.

Datasets were obtained from Kaggle using the Kaggle API to ensure reproducibility and standardized access protocols. All data acquisition procedures were implemented in Python, with code archived for independent verification.

Data Quality Assessment and Preprocessing

Initial data exploration and quality assessment were performed to identify and address data integrity issues. All preprocessing steps were implemented as reusable Python functions:

Missing Value Treatment:

- **Continuous variables:** Median imputation was employed to preserve distributional properties while minimizing sensitivity to outliers.
- **Categorical variables:** Mode imputation was applied to maintain the most prevalent category representation.

Outlier Detection and Treatment: Extreme values were identified using the Interquartile Range (IQR) method:

$$\text{Outlier if: } x_i < Q_1 - 1.5 \times IQR \text{ or } x_i > Q_3 + 1.5 \times IQR,$$

where Q_1 and Q_3 represent the first and third quartiles, respectively. Identified outliers were retained but flagged for sensitivity analysis.

Categorical Variable Encoding: Categorical variables were transformed using one-hot encoding to generate binary indicator variables suitable for continuous regression modeling, implemented using `scikit-learn`'s `OneHotEncoder`.

3.2.2 Step 2: Data Partitioning Strategy

To assess model stability across different training set sizes, each of the 16 prediction tasks was subjected to three distinct data splitting configurations. All splits employed a fixed random seed (`RANDOM_STATE = 42`) to ensure reproducibility across all computational experiments.

Table 3.1 Data Splitting Configurations

Split Configuration	Training	Validation	Test
Split A	60%	20%	20%
Split B	70%	15%	15%
Split C	80%	10%	10%

This multi-split design, implemented using `scikit-learn`'s `train_test_split` function, enables evaluation of:

- Model performance sensitivity to training set size
- Robustness of conclusions across different data availability scenarios
- Overfitting tendencies as training data increases

The validation set is used for hyperparameter selection (when applicable) and early stopping criteria, while the test set provides unbiased performance estimates for final model comparison.

3.2.3 Step 3: Feature Preprocessing

To ensure numerical stability and facilitate convergence, all continuous features were standardized using z-score normalization, implemented as a custom Python class:

$$z_i = \frac{x_i - \bar{x}_{\text{train}}}{\sigma_{\text{train}}},$$

where $\bar{x}_{\text{train}}$ and σ_{train} are the mean and standard deviation computed from the training set only. Crucially, the same transformation parameters derived from the training data are applied to validation and test sets to prevent information leakage.

Preprocessing parameters are stored as Python objects (using `pickle` or `joblib`) for consistent application during model deployment and to enable proper interpretation of model coefficients and feature importance measures.

3.2.4 Step 4: Model Training Procedures

All model training procedures were implemented as Python classes following object-oriented programming principles, ensuring modularity and reusability.

GAM Training Protocol

For each of the 5 GAM configurations (one per mean modeling technique), training proceeds as follows:

1. Train the mean function $\hat{\mu}(x)$ using the specified technique (Spline, EBM, XGBoost, RF, or SVR) on the training data $\{(x_i, y_i)\}_{i=1}^{n_{\text{train}}}$.
2. Compute residuals:

$$r_i = y_i - \hat{\mu}(x_i), \quad i = 1, \ldots, n_{\text{train}}.$$

3. Estimate constant dispersion via maximum likelihood:

$$\hat{\sigma}^2 = \frac{1}{n_{\text{train}}} \sum_{i=1}^{n_{\text{train}}} r_i^2.$$

This procedure yields a predictive distribution for new observations:

$$\hat{p}(y \mid x) = \mathcal{N}(y; \hat{\mu}(x), \hat{\sigma}^2).$$

DGAM Training Protocol

For each of the 25 DGAM configurations (5 mean techniques $\times$ 5 dispersion techniques), training proceeds through alternating optimization implemented in Python:

1. **Mean Model Training:** Train the mean function $\hat{\mu}(x)$ using the specified mean modeling technique.

2. **Residual Computation:** Calculate squared residuals:

$$r_i^2 = (y_i - \hat{\mu}(x_i))^2, \quad i = 1, \ldots, n_{\text{train}}.$$

3. **Dispersion Model Training:** Train the dispersion function $\log(\hat{\sigma}^2(x))$ on the log-transformed squared residuals:

$$z_i = \log(r_i^2 + \delta),$$

where $\delta = 0.01$ is a small constant ensuring numerical stability.

4. **Iterative Refinement (if applicable):** For techniques supporting iterative updates (EBM, XGBoost), alternate between refining mean and dispersion models until convergence.

The resulting predictive distribution is:

$$\hat{p}(y \mid x) = \mathcal{N}(y; \hat{\mu}(x), \hat{\sigma}^2(x)).$$

Table 3.2 Complete Model Configuration Matrix.

Framework	Mean Technique	Dispersion Modeling Technique					
		Const	Spline	EBM	XGBoost	RF	SVR
GAM	Spline	✓	—	—	—	—	—
	EBM	✓	—	—	—	—	—
	XGBoost	✓	—	—	—	—	—
	RF	✓	—	—	—	—	—
	SVR	✓	—	—	—	—	—
DGAM	Spline	—	✓	✓	✓	✓	✓
	EBM	—	✓	✓	✓	✓	✓
	XGBoost	—	✓	✓	✓	✓	✓
	RF	—	✓	✓	✓	✓	✓
	SVR	—	✓	✓	✓	✓	✓
Total Configurations		5 GAM + 25 DGAM = 30 models per prediction task					

3.2.5 Step 5: Hyperparameter Configuration

To ensure fair comparison and reproducibility, hyperparameters were fixed at empirically stable values rather than extensively tuned for each model configuration. This design decision reflects the study's primary objective: to isolate and quantify the benefit of flexible dispersion modeling within the DGAM framework.

All hyperparameter values were hard-coded in Python configuration files, ensuring consistency across all experiments. The complete hyperparameter settings are summarized in Table 3.3.

Table 3.3 Fixed Hyperparameter Settings for All Experiments.

Model Component	Hyperparameter	Value
Smoothing Splines	Number of knots	10
	Spline degree	3 (cubic)
	Ridge penalty (α)	1.0
	Feature standardization	Enabled (zero mean, unit variance)
EBM	Implementation	`ExplainableBoostingRegressor`
	Random seed	42
	Fallback model	Spline Ridge GAM (if unavailable)
XGBoost	Number of trees	1200
	Maximum tree depth	4
	Learning rate	0.03
	Subsample ratio	0.9
	Column subsample ratio	0.9
	L2 regularization (λ)	1.0
Random Forest (Mean)	Number of trees	800
	Minimum samples per leaf	3
	Bootstrap sampling	Enabled
Random Forest (Disp.)	Number of trees	600
	Minimum samples per leaf	3
	Bootstrap sampling	Enabled
SVR	Kernel	RBF (Radial Basis Function)
	Regularization (C)	10
	Kernel parameter (γ)	`scale`

Rationale for Fixed Hyperparameters

The decision to employ fixed hyperparameters across all configurations is supported by three methodological considerations:

1. **Reproducibility:** Fixed settings ensure exact replication of results across different computational environments and runs, facilitating independent verification.

2. **Fair Comparison:** All model configurations operate under equivalent conditions,

enabling direct attribution of performance differences to the mean-dispersion modeling framework.

3. **Research Focus:** The primary scientific question concerns the value of dispersion modeling in DGAMs, not hyperparameter optimization.

3.2.6 Step 6: Model Evaluation Framework

Each evaluation metric was implemented as a standalone Python function to ensure modularity and clarity in the evaluation pipeline. This structure supports independent testing, easier maintenance, and seamless extension for future metrics.

Point Prediction Metrics

Mean Absolute Error (MAE) MAE provides a straightforward measure of average prediction error:

$$\text{MAE} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} |y_i - \hat{\mu}(x_i)|.$$

Root Mean Squared Error (RMSE) RMSE measures prediction accuracy with emphasis on larger errors:

$$\text{RMSE} = \sqrt{\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} (y_i - \hat{\mu}(x_i))^2}.$$

Coefficient of Determination (R^2) R^2 quantifies the proportion of variance explained by the model:

$$R^2 = 1 - \frac{\sum_{i=1}^{n_{\text{test}}} (y_i - \hat{\mu}(x_i))^2}{\sum_{i=1}^{n_{\text{test}}} (y_i - \bar{y})^2},$$

where $\bar{y} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} y_i$ is the mean of observed values. Values closer to 1 indicate better model fit.

Likelihood-Based Metrics

For all metrics in this section, let n_{test} denote the number of observations in the test set, y_i the observed value, $\hat{\mu}(x_i)$ the predicted mean, and $\hat{\sigma}(x_i)$ the predicted standard deviation for observation i.

Akaike Information Criterion (AIC) The AIC balances model fit and complexity:

$$\text{AIC} = -2\ln(\hat{L}) + 2k,$$

where $\hat{L}$ is the maximized likelihood and k is the effective number of parameters. Lower AIC values indicate better models.

Negative Log-Likelihood (NLL) The NLL measures the goodness of fit:

$$\text{NLL} = -\sum_{i=1}^{n_{\text{test}}} \log p(y_i \mid x_i; \theta),$$

where $p(y_i \mid x_i; \theta)$ is the probability density function evaluated at y_i given covariates x_i and parameters θ. Lower NLL indicates better predictive performance.

Probabilistic Calibration Metrics

Continuous Ranked Probability Score (CRPS) CRPS evaluates the calibration of probabilistic forecasts:

$$\text{CRPS}(F, y) = \int_{-\infty}^{\infty} [F(x) - \mathbf{1}(x \geq y)]^2 \, dx,$$

where $F(\cdot)$ is the predicted cumulative distribution function and $\mathbf{1}(\cdot)$ is the indicator function. The average CRPS over the test set is:

$$\text{CRPS} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \text{CRPS}(F_i, y_i).$$

Lower CRPS values indicate better calibrated probabilistic predictions.

Prediction Interval Coverage Probability (PICP_{95}) PICP_{95} measures the empirical coverage of 95% prediction intervals:

$$\text{PICP}_{95} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \mathbf{1}(L_i \leq y_i \leq U_i),$$

Model performance is assessed using a comprehensive suite of seven evaluation metrics, all implemented as Python functions, spanning three categories: likelihood-based measures, point prediction accuracy, and probabilistic calibration.

Table 3.4 Comprehensive Evaluation Metrics.

Category	Metric	Notation	Optimal Direction
Point Prediction	Mean Absolute Error	MAE	Minimize (↓)
	Root Mean Squared Error	RMSE	Minimize (↓)
	Coefficient of Determination	R^2	Maximize (↑)
Likelihood-Based	Akaike Information Criterion	AIC	Minimize (↓)
	Negative Log-Likelihood	NLL	Minimize (↓)
Probabilistic	Continuous Ranked Probability Score	CRPS	Minimize (↓)
	Prediction Interval Coverage (95%)	PICP_{95}	≈ 0.95

3.2.7 Step 7: Statistical Analysis and Result Compilation

All evaluation metrics were computed for each of the 1,440 trained models (30 models × 16 tasks × 3 splits) and stored in structured data formats (CSV files and Python DataFrames) for subsequent statistical analysis. Python scripts were developed to:

- Aggregate results across splits and tasks
- Compute summary statistics (mean, median, standard deviation)
- Generate comparative visualizations
- Perform statistical significance tests
- Export results for presentation in Chapter 4

3.2.8 Implementation and Computational Infrastructure

All code was implemented in Python 3.9+ using standard scientific computing libraries:

- **Data manipulation:** `pandas`, `numpy`
- **Smoothing Splines:** `scikit-learn`, `scipy.interpolate`
- **EBM:** `interpret` package (InterpretML framework)
- **XGBoost:** `xgboost` library
- **Random Forest:** `scikit-learn.ensemble`
- **SVR:** `scikit-learn.svm`
- **Visualization:** `matplotlib`, `seaborn`
- **Statistical analysis:** `scipy.stats`, `statsmodels`

Complete source code, including data preprocessing pipelines, model training scripts, evaluation procedures, and result compilation scripts, is archived and version-controlled using Git, ensuring full reproducibility and independent verification of results.

Computational experiments were conducted with typical model training times ranging from seconds (for Smoothing Splines and SVR) to minutes (for XGBoost and Random Forests) per configuration, enabling completion of all experiments within reasonable computational budgets.

3.3 Research Framework Summary

To achieve the research objectives, this study adopts a structured two-phase research framework that links theoretical development with computational implementation and empirical evaluation. The framework clarifies how mathematical modeling advances are translated into algorithmic procedures and subsequently validated through systematic experiments.

The overall research design is illustrated in Figures 3.1 and 3.2, showing the progression from theoretical foundations to model implementation and performance assessment.

Comprehensive Two-Phase Research Framework

Phase I: Theoretical & Algorithmic Development

Mathematical Formulation of DGAMs

↓

Algorithm Development (Five Techniques)

Splines, EBM, XGBoost, RF, SVR

↓

Theoretical Properties and Convergence Analysis

Figure 3.1 Overall Phase I Research Framework: Theoretical and Algorithmic Development.

Comprehensive Phase II Framework for Model Implementation

Phase II: Computational Implementation and Validation

Conceptual Foundation

Universal Life Cycle: Birth → Aging → Illness → Death

↓

Data Structure

4 Health Datasets × 4 Target Variables = 16 Prediction Tasks

↓

Data Partitioning (Python Implementation)

Three Splitting Configurations: (60/20/20), (70/15/15), (80/10/10)

↓

Model Architecture (Python Classes)

GAM Framework: 5 Mean Techniques × Constant Dispersion = 5 Models

DGAM Framework: 5 Mean Techniques × 5 Dispersion Techniques = 25 Models

Total: 30 Models per Task × 16 Tasks × 3 Splits = 1,440 Trained Models

↓

Evaluation Metrics (Python Functions)

Point Prediction: RMSE, MAE, R^2

Likelihood-based: AIC, NLL

Probabilistic: CRPS, PICP_{95}

↓

Statistical Analysis (Python Scripts)

Results Compilation → Visualization → Interpretation

Figure 3.2 Overall Phase II Research Framework: Model Implementation.

This two-phase methodology ensures a rigorous and coherent pathway for the development and evaluation of Double Generalized Additive Models (DGAMs). Phase I establishes the mathematical and algorithmic foundations, including formulation, estimation strategies, and theoretical properties. Phase II operationalizes these advances through structured computational implementation and systematic empirical validation. Together, the phases provide comprehensive evidence regarding the benefits of flexible dispersion modeling in enhancing predictive performance and uncertainty quantification within statistical health analytics.

CHAPTER IV

RESULTS

This chapter presents the theoretical and algorithmic contributions of this research, establishing the mathematical foundations for Double Generalized Additive Models (DGAMs) enhanced through integration with advanced machine learning techniques. The chapter systematically demonstrates the theoretical feasibility, algorithmic specifications, and mathematical rigor underlying the proposed methodologies.

4.1 Overview and Structure

The primary objective of this chapter is to provide rigorous mathematical proofs and algorithmic formulations that establish the theoretical validity of extending classical Generalized Additive Models to simultaneously model both mean and dispersion functions using five distinct machine learning techniques: Smoothing Splines, Explainable Boosting Machines (EBM), Extreme Gradient Boosting (XGBoost), Random Forests (RF), and Support Vector Regression (SVR).

This chapter addresses the first major research objective articulated in Phase I of the methodology: *to develop theoretical concepts and computational algorithms for flexible generalized additive models of the mean and dispersion functions*. The mathematical developments presented here establish the theoretical foundation necessary for subsequent computational implementation and empirical validation.

4.1.1 Structure of This Chapter

The chapter is structured as follows:

Section 4.1: Conceptual Framework We begin by formally defining the Double Generalized Additive Model framework, establishing the mathematical foundations that extend classical GAMs to accommodate flexible dispersion modeling. This section provides the formal specification of the DGAM structure for Gaussian responses, probabilistic foundations from the exponential dispersion family, dual link function formulation for mean and dispersion, and mathematical conditions ensuring model identifiability and estimability.

Sections 4.2–4.7: Technique-Specific Developments For each of the five machine learning techniques, we provide:

- **Mathematical Formulation** with rigorous derivation of how each technique is adapted to the DGAM framework, including objective functions, penalization mechanisms, and coupling between mean and dispersion estimation;
- **Theoretical Properties** with formal proofs establishing existence and uniqueness of solutions, consistency and asymptotic normality of estimators, convergence guarantees, and identifiability conditions; and
- **Algorithmic Specification** with detailed computational algorithms, alternating optimization strategies, convergence criteria, and complexity analysis.

Each technique-specific section demonstrates the *mathematical feasibility* of integrating that particular method with the DGAM framework, providing the theoretical justification necessary for subsequent computational implementation.

4.1.2 Purpose and Contribution

The mathematical and algorithmic developments in this chapter serve three critical purposes:

1. Theoretical Validation : The formal mathematical proofs establish that the proposed DGAM extensions are theoretically sound, with well-defined optimization objectives, identifiable parameters, and desirable asymptotic properties.

2. Algorithmic Specification : The detailed algorithms provide precise computational recipes that can be directly translated into executable code, bridging the gap between abstract mathematical theory and concrete computational implementation.

3. Methodological Innovation : By systematically demonstrating how five distinct machine learning techniques can be adapted to the DGAM framework, this chapter establishes a general methodological template for extending flexible dispersion modeling to other advanced methods.

4.1.3 Relationship to Empirical Validation

The mathematical proofs and algorithms presented in this chapter establish *what is theoretically possible*—demonstrating that DGAMs with flexible machine learning components are mathematically well-defined and computationally tractable. However, theoretical feasibility alone does not guarantee practical utility.

The computational implementations based on these algorithms, along with their empirical evaluation on real-world health data, are presented in the Appendix (Appendix A), which contains complete Python code implementations of all algorithms, detailed experimental results across 16 prediction tasks, comprehensive performance comparisons between GAM and DGAM frameworks, statistical significance tests and model selection analyses, and visualization of fitted mean and dispersion functions.

This organizational structure—theoretical development in the main chapter, computational implementation in the Appendix—reflects the logical progression from mathematical possibility to practical realization.

4.1.4 Key Contributions

The primary contributions of this chapter include:

1. **Mathematical Framework:** Rigorous formalization of DGAMs for Gaussian responses with identity link for mean and log link for dispersion.

2. **Algorithmic Integration:** Development of five distinct computational algorithms (Algorithms 1–5) that integrate Smoothing Splines, EBM, XGBoost, RF, and SVR with the DGAM framework.

3. **Theoretical Properties:** Formal proofs of consistency, convergence, and identifiability for DGAM estimators under each technique.

4. **Coupled Estimation Procedures:** Novel coupled Penalized Iteratively Reweighted Least Squares (PIRLS) and alternating boosting/bagging algorithms that properly account for the interdependence between mean and dispersion parameters.

5. **Computational Specifications:** Complete algorithmic specifications with precise stopping criteria, hyperparameter settings, and complexity analysis.

4.1.5 Reading Guide

Readers primarily interested in theoretical foundations should focus on the mathematical formulations and theoretical property proofs within each technique-specific section. Those interested in computational implementation should pay particular attention to the algorithmic specifications and detailed algorithm steps, which can be directly translated into code. Readers seeking empirical evidence of model performance should proceed directly to the Appendix (Appendix **??**), where the Python implementations are applied to real-world health datasets.

The remainder of this chapter presents the detailed mathematical and algorithmic developments for DGAMs, beginning with the foundational framework definition and proceeding through each of the five machine learning techniques.

4.2 Double Generalized Additive Models: Mathematical Framework

A Double Generalized Additive Model (DGAM) extends Generalized Additive Models (GAMs) by simultaneously modeling both the mean and dispersion (variance) of the response variable as smooth functions of covariates. While traditional GAMs focus solely on modeling the mean response, DGAMs recognize that variability in data may also change systematically with predictor variables, making them particularly valuable for analyzing heteroscedastic data where variance is not constant (Rigby and Stasinopoulos, 2005; Wood, 2017).

DGAMs are especially useful in healthcare, environmental science, and finance, where both the central tendency and the spread of outcomes may depend on explanatory variables. For instance, in medical studies, not only might the average health outcome vary with age and treatment, but the variability in individual responses may also change across different patient subgroups.

The Double Generalized Additive Model specifies separate additive predictor structures for both location (mean) and scale (dispersion) parameters. Instead of assuming a constant dispersion parameter ϕ as in GLMs and GAMs, DGAMs allow ϕ_i to vary across observations as a function of covariates.

4.2.1 Random Component

The distribution of the response variable Y_i for each of the n independently sampled observations follows a distribution from the exponential dispersion family:

$$f_{y_i}(y_i; \theta_i, \phi_i) = \exp\left(\frac{y_i\theta_i - b(\theta_i)}{a(\phi_i)} + c(y_i; \phi_i)\right),$$

where $\theta_i \in \Theta$ is the canonical parameter, $\phi_i > 0$ is the observation-specific dispersion parameter, $a(\cdot)$ is a known function (typically $a(\phi_i) = \phi_i$), $b(\cdot)$ is the cumulant function, and $c(\cdot)$ is a normalization constant. Common distributions include Gaussian, Gamma, and Inverse Gaussian.

As in GLMs and GAMs, the mean and variance relationships are:

$$\mathbb{E}[Y_i] = b'(\theta_i) = \mu_i,$$
$$\text{Var}[Y_i] = a(\phi_i)b''(\theta_i) = \phi_i V(\mu_i),$$

where $V(\mu_i)$ is the variance function depending on the chosen distribution family.

4.2.2 Mean Structure (Location Model)

The mean model replaces the linear predictor with an additive predictor composed of smooth functions:

$$g(\mu_i) = \eta_i = A_i\boldsymbol{\theta} + f_1(x_{1i}) + f_2(x_{2i}) + \ldots + f_p(x_{pi}),$$

where $g(\cdot)$ is a monotonic link function (e.g., identity, log, logit), A_i is a $1 \times r$

row vector of covariates for strictly parametric terms, $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_r)^T$ is the $r \times 1$ vector of parametric coefficients (ensuring $A_i\boldsymbol{\theta}$ is a scalar), $f_j(\cdot)$ are smooth, potentially nonlinear functions of covariates x_{ji} for $j = 1, 2, \ldots, p$, and η_i is the linear predictor for the mean structure.

The smooth functions f_j are typically estimated using penalized splines, smoothing splines, or other basis function representations.

4.2.3 Dispersion Structure (Scale Model)

A key innovation of DGAMs is the explicit modeling of the dispersion parameter as a function of covariates:

$$h(\phi_i) = \lambda_i = B_i\boldsymbol{\gamma} + s_1(z_{1i}) + s_2(z_{2i}) + \ldots + s_q(z_{qi}),$$

where $h(\cdot)$ is a monotonic link function for the dispersion (commonly log to ensure $\phi_i > 0$), B_i is a row vector of covariates for parametric terms in the dispersion model, $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_m)^T$ is the vector of parametric coefficients for dispersion, $s_j(\cdot)$ are smooth functions of covariates z_{ji} for $j = 1, 2, \ldots, q$, and λ_i is the linear predictor for the dispersion structure.

The covariates in the dispersion model may be the same as or different from those in the mean model, allowing investigation of whether certain variables affect only the mean, only the variability, or both.

4.2.4 Link Functions

DGAMs employ two separate smooth and invertible link functions to connect the parameters of interest to their respective additive predictors.

Mean Link Function for Gaussian Response The function $g(\cdot)$ relates the mean μ_i to the additive predictor η_i:

$$g(\mu_i) = \eta_i = A_i\boldsymbol{\theta} + f_1(x_{1i}) + f_2(x_{2i}) + \ldots + f_p(x_{pi}).$$

For Gaussian (normal) distributed response variables, the canonical link function is the identity link:

Link function name	Link function formula	Inverse link
Identity	$g(\mu) = \mu$	$g^{-1}(\eta) = \eta$

The identity link is natural for Gaussian responses as it imposes no restrictions on the range of μ_i, which is appropriate since the mean of a normal distribution can take any real value.

Dispersion Link Function for Gaussian Response The function $h(\cdot)$ relates the dispersion parameter ϕ_i (representing the variance σ_i^2) to the additive predictor λ_i:

$$h(\phi_i) = \lambda_i = B_i\boldsymbol{\gamma} + s_1(z_{1i}) + s_2(z_{2i}) + \ldots + s_q(z_{qi}).$$

For the dispersion parameter, which must be strictly positive ($\phi_i > 0$), the most commonly used link function is:

Link function name	Link function formula	Inverse link
Log	$h(\phi) = \log(\phi)$	$h^{-1}(\lambda) = \exp(\lambda)$

The log link naturally ensures $\phi_i > 0$ for all values of λ_i.

Inverse Link Functions The inverse link functions $g^{-1}(\cdot)$ and $h^{-1}(\cdot)$ transform the fitted additive predictors back to the original scale:

$$\mu_i = g^{-1}(\eta_i) = g^{-1}\left(A_i\boldsymbol{\theta} + \sum_{j=1}^{p} f_j(x_{ji})\right) = A_i\boldsymbol{\theta} + \sum_{j=1}^{p} f_j(x_{ji}),$$

$$\phi_i = h^{-1}(\lambda_i) = h^{-1}\left(B_i\boldsymbol{\gamma} + \sum_{j=1}^{q} s_j(z_{ji})\right) = \exp\left(B_i\boldsymbol{\gamma} + \sum_{j=1}^{q} s_j(z_{ji})\right),$$

for $i = 1, 2, \ldots, n$.

4.2.5 Model Specification Summary

The complete Gaussian DGAM specification can be summarized as:

$$g(\mu_i) = \mu_i = A_i\boldsymbol{\theta} + \sum_{j=1}^{p} f_j(x_{ji}),$$
$$h(\phi_i) = \log(\phi_i) = B_i\boldsymbol{\gamma} + \sum_{j=1}^{q} s_j(z_{ji}),$$

with $Y_i \sim \mathcal{N}(\mu_i, \phi_i)$ for $i = 1, 2, \dots, n$, where $\phi_i = \sigma_i^2$ represents the variance parameter.

Equivalently, using inverse link functions:

$$\mu_i = A_i\boldsymbol{\theta} + \sum_{j=1}^{p} f_j(x_{ji}),$$
$$\sigma_i^2 = \exp\left(B_i\boldsymbol{\gamma} + \sum_{j=1}^{q} s_j(z_{ji})\right).$$

This dual-structure framework enables Gaussian DGAMs to provide comprehensive characterization of how both central tendency (mean) and variability (variance) in the response variable change with predictor variables, making them particularly powerful for modeling heteroscedastic data.

4.3 Smoothing Splines

Smoothing splines provide a flexible nonparametric approach to estimating smooth functions by balancing data fit and smoothness through penalized optimization. Given observations $(y_1, x_1), \dots, (y_n, x_n)$ and a choice of $\lambda \geq 0$, the smoothing spline estimator $\hat{f}_\lambda$ is defined by:

$$\hat{f}_\lambda = \underset{f \in C^2}{\text{argmin}} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \int [f''(t)]^2 dt,$$

where C^2 denotes the space of twice-differentiable functions. The solution is a natural cubic spline with knots at the observed x values, making this optimization problem finite-dimensional (Chouldechova and Hastie, 2015).

4.3.1 Basis Representation and Matrix Formulation

In practice, the smooth function $f(x)$ is represented using a basis expansion. Let $\{b_1(x), b_2(x), \ldots, b_M(x)\}$ denote a set of basis functions (e.g., B-splines or natural cubic splines):

$$f(x) = \sum_{m=1}^{M} b_m(x)\beta_m = \mathbf{b}(x)^T \boldsymbol{\beta},$$

where $\mathbf{b}(x) = (b_1(x), b_2(x), \ldots, b_M(x))^T$ is the basis function vector and $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_M)^T$ are the coefficients to be estimated.

Let B denote the $n \times M$ design matrix with elements $B_{im} = b_m(x_i)$. The fitted values are $\hat{\mathbf{y}} = B\boldsymbol{\beta}$. The roughness penalty can be expressed as:

$$\int [f''(t)]^2 dt = \boldsymbol{\beta}^T S \boldsymbol{\beta},$$

where S is the $M \times M$ penalty matrix with elements $S_{mn} = \int b''_m(x) b''_n(x) dx$.

The penalized least squares objective becomes:

$$\hat{\boldsymbol{\beta}}_\lambda = \underset{\boldsymbol{\beta}}{\text{argmin}} \|\mathbf{y} - B\boldsymbol{\beta}\|^2 + \lambda \boldsymbol{\beta}^T S \boldsymbol{\beta},$$

with closed-form solution:

$$\hat{\boldsymbol{\beta}}_\lambda = (B^T B + \lambda S)^{-1} B^T \mathbf{y}.$$

4.3.2 Parameter Estimation for DGAM with Smoothing Splines

For DGAMs, smoothing splines represent both mean and dispersion functions:

$$\eta_{\mu,i} = (B_\mu)_i \boldsymbol{\beta}_\mu = \sum_{m=1}^{M_\mu} b_{\mu,m}(x_i)\beta_{\mu,m},$$
$$\eta_{\phi,i} = (B_\phi)_i \boldsymbol{\beta}_\phi = \sum_{m=1}^{M_\phi} b_{\phi,m}(x_i)\beta_{\phi,m},$$

where B_μ is the $n \times M_\mu$ design matrix for the mean model, B_ϕ is the $n \times M_\phi$ design matrix for the dispersion model, $\boldsymbol{\beta}_\mu$ and $\boldsymbol{\beta}_\phi$ are the coefficient vectors.

The mean and dispersion are obtained via inverse link functions:

$$\mu_i = g_\mu^{-1}(\eta_{\mu,i}) = g_\mu^{-1}((B_\mu)_i \boldsymbol{\beta}_\mu),$$
$$\phi_i = g_\phi^{-1}(\eta_{\phi,i}) = g_\phi^{-1}((B_\phi)_i \boldsymbol{\beta}_\phi).$$

The penalized log-likelihood for the DGAM with smoothing splines is:

$$\ell_p(\boldsymbol{\beta}_\mu, \boldsymbol{\beta}_\phi) = \sum_{i=1}^{n} \log p(y_i \mid \mu_i, \phi_i) - \frac{1}{2}\lambda_\mu \boldsymbol{\beta}_\mu^T S_\mu \boldsymbol{\beta}_\mu - \frac{1}{2}\lambda_\phi \boldsymbol{\beta}_\phi^T S_\phi \boldsymbol{\beta}_\phi,$$

where $p(y_i \mid \mu_i, \phi_i)$ is the probability density, S_μ and S_ϕ are the penalty matrices, and λ_μ and λ_ϕ are smoothing parameters.

4.3.3 Algorithm: Classical Spline DGAM via Coupled PIRLS

The estimation proceeds via coupled Penalized Iteratively Reweighted Least Squares (PIRLS):

Algorithm 1 Classical Spline DGAM via Coupled PIRLS

Input: $\{(y_i, x_i)\}_{i=1}^{n}$, link functions g_μ, g_ϕ, distribution $p(y \mid \mu, \phi)$

Hyperparameters: Spline bases B_μ, B_ϕ, penalties S_μ, S_ϕ, smoothing $\lambda_\mu, \lambda_\phi$

1: **Trainable parameters:** $\boldsymbol{\beta}_\mu, \boldsymbol{\beta}_\phi$

2: Construct design matrices $X_\mu = B_\mu(X)$, $X_\phi = B_\phi(X)$

3: Initialize $\boldsymbol{\beta}_\mu^{(0)}, \boldsymbol{\beta}_\phi^{(0)}$

4: **repeat**

5: Compute $\boldsymbol{\eta}_\mu = X_\mu \boldsymbol{\beta}_\mu$, $\boldsymbol{\eta}_\phi = X_\phi \boldsymbol{\beta}_\phi$

6: Update $\boldsymbol{\mu} = g_\mu^{-1}(\boldsymbol{\eta}_\mu)$, $\boldsymbol{\phi} = g_\phi^{-1}(\boldsymbol{\eta}_\phi)$

7: Evaluate penalized log-likelihood $\ell_p(\boldsymbol{\beta}_\mu, \boldsymbol{\beta}_\phi)$

8: Compute coupled score vectors U_μ, U_ϕ

9: Compute observed information blocks $I_{\mu\mu}, I_{\mu\phi}, I_{\phi\phi}$

10: Solve joint Newton update for $(\boldsymbol{\beta}_\mu, \boldsymbol{\beta}_\phi)$:

11: $$\begin{bmatrix} I_{\mu\mu} & I_{\mu\phi} \\ I_{\mu\phi}^T & I_{\phi\phi} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\beta}_\mu \\ \Delta\boldsymbol{\beta}_\phi \end{bmatrix} = \begin{bmatrix} U_\mu \\ U_\phi \end{bmatrix}$$

12: Update $\boldsymbol{\beta}_\mu \leftarrow \boldsymbol{\beta}_\mu + \Delta\boldsymbol{\beta}_\mu$, $\boldsymbol{\beta}_\phi \leftarrow \boldsymbol{\beta}_\phi + \Delta\boldsymbol{\beta}_\phi$

13: **until** convergence: $\|\Delta\boldsymbol{\beta}_\mu\| < \epsilon_\mu$ and $\|\Delta\boldsymbol{\beta}_\phi\| < \epsilon_\phi$

14: **Output:** $\hat{\mu}(x)$ and $\hat{\phi}(x)$

Detailed Algorithm Steps:

Step 1: Initialization (Line 3) Initialize $\boldsymbol{\beta}_\mu^{(0)}$ and $\boldsymbol{\beta}_\phi^{(0)}$ using GLM estimates with constant dispersion.

Step 2: Compute Score Vectors (Lines 8-9) At iteration t, compute:

$$U_\mu = B_\mu^T W_\mu (\mathbf{y} - \boldsymbol{\mu}) - \lambda_\mu S_\mu \boldsymbol{\beta}_\mu,$$
$$U_\phi = B_\phi^T W_\phi \mathbf{d} - \lambda_\phi S_\phi \boldsymbol{\beta}_\phi,$$

where W_μ and W_ϕ are weight matrices, and $\mathbf{d}$ is the vector of adjusted responses for dispersion.

Step 3: Compute Information Matrix Blocks (Line 9)

$$
\begin{aligned}
I_{\mu\mu} &= B_\mu^T W_\mu B_\mu + \lambda_\mu S_\mu, \\
I_{\phi\phi} &= B_\phi^T W_\phi B_\phi + \lambda_\phi S_\phi, \\
I_{\mu\phi} &= B_\mu^T W_{\mu\phi} B_\phi,
\end{aligned}
$$

where $W_{\mu\phi}$ captures coupling between mean and dispersion.

Step 4: Final Predictions (Line 13) Compute fitted functions:

$$
\begin{aligned}
\hat{\mu}(x) &= g_\mu^{-1}(\mathbf{b}_\mu(x)^T \hat{\boldsymbol{\beta}}_\mu), \\
\hat{\phi}(x) &= g_\phi^{-1}(\mathbf{b}_\phi(x)^T \hat{\boldsymbol{\beta}}_\phi).
\end{aligned}
$$

This coupled PIRLS algorithm ensures proper accounting of interdependence between mean and dispersion during estimation.

4.4 Explainable Boosting Machine

The Explainable Boosting Machine (EBM) is an interpretable machine learning algorithm within the InterpretML framework that combines the transparency of Generalized Additive Models with modern boosting techniques (Nori et al., 2019; Lou et al., 2012). EBM maintains interpretability while achieving prediction accuracy comparable to sophisticated ensemble methods.

4.4.1 EBM Model Structure

EBM adopts a Generalized Additive Model structure:

$$
g(\mathbb{E}[Y]) = \beta_0 + \sum_{j=1}^{p} f_j(x_j),
$$

where $g(\cdot)$ is a link function, β_0 is an intercept term, and $f_j(\cdot)$ are univariate shape functions capturing each feature's contribution.

For models requiring interaction terms:

$$g(\mathbb{E}[Y]) = \beta_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{i<j} f_{ij}(x_i, x_j),$$

where $f_{ij}(x_i, x_j)$ represents pairwise interactions.

4.4.2 Key Features of EBM

Modern Learning Techniques: Unlike traditional GAMs using splines, EBM employs gradient boosting to learn each feature function f_j with carefully constrained training on one feature at a time in round-robin fashion with very low learning rate.

Round-Robin Feature Selection: By cycling through features sequentially rather than selecting features based on gradient magnitude, EBM learns optimal feature functions independently, revealing unbiased feature contributions.

Automatic Interaction Detection: EBM can automatically detect and incorporate significant pairwise interaction terms, capturing non-additive effects while maintaining interpretability.

4.4.3 Parameter Estimation for DGAM with EBM

For DGAMs, EBM is applied separately to model both mean and dispersion:

$$\eta_{\mu,i} = f_\mu(x_i) = \beta_{0,\mu} + \sum_{j=1}^{p} f_{\mu,j}(x_{ji}),$$

$$\eta_{\phi,i} = f_\phi(x_i) = \beta_{0,\phi} + \sum_{j=1}^{q} f_{\phi,j}(z_{ji}),$$

where $f_\mu(\cdot)$ and $f_\phi(\cdot)$ are additive predictors learned through boosting.

The mean and dispersion parameters are obtained via inverse link functions:

$$\mu_i = g_\mu^{-1}(\eta_{\mu,i}) = g_\mu^{-1}(f_\mu(x_i)),$$
$$\phi_i = g_\phi^{-1}(\eta_{\phi,i}) = g_\phi^{-1}(f_\phi(x_i)).$$

4.4.4 Algorithm: EBM-DGAM via Alternating Boosting

The estimation proceeds through alternating boosting on coupled negative log-likelihood gradients:

Algorithm 2 EBM-DGAM via Alternating Boosting for Mean and Dispersion

Input: $\{(y_i, x_i)\}_{i=1}^n$, link functions g_μ, g_ϕ, distribution $p(y \mid \mu, \phi)$
Hyperparameters: Boosting rounds T, learning rates ν_μ, ν_ϕ, EBM constraints (bins/interactions), early stopping tolerance ε
Trainable parameters: Additive predictors $f_\mu(\cdot), f_\phi(\cdot)$ with $\eta_\mu = f_\mu(x)$, $\eta_\phi = f_\phi(x)$
Initialize $f_\mu^{(0)}(x) \equiv c_\mu$ and $f_\phi^{(0)}(x) \equiv c_\phi$
for $t = 0, 1, \ldots, T-1$ **do**
    Compute linear predictors $\eta_{\mu,i}^{(t)} = f_\mu^{(t)}(x_i)$ and $\eta_{\phi,i}^{(t)} = f_\phi^{(t)}(x_i)$
    Update distributional parameters $\mu_i^{(t)} = g_\mu^{-1}(\eta_{\mu,i}^{(t)})$, $\phi_i^{(t)} = g_\phi^{-1}(\eta_{\phi,i}^{(t)})$
    Evaluate negative log-likelihood $L^{(t)} = \sum_{i=1}^n -\log p(y_i \mid \mu_i^{(t)}, \phi_i^{(t)})$
    Compute mean pseudo-residuals:
        $r_{\mu,i}^{(t)} = -\frac{\partial}{\partial \eta_{\mu,i}}\left[-\log p(y_i \mid \mu_i^{(t)}, \phi_i^{(t)})\right]$
    Fit EBM base-learner $h_\mu^{(t)}$ to $\{(x_i, r_{\mu,i}^{(t)})\}_{i=1}^n$
    Update $f_\mu^{(t+1)} \leftarrow f_\mu^{(t)} + \nu_\mu h_\mu^{(t)}$
    Recompute $\mu_i^{(t+1)}$ using $f_\mu^{(t+1)}$ while holding $f_\phi^{(t)}$ fixed
    Compute dispersion pseudo-residuals:
        $r_{\phi,i}^{(t)} = -\frac{\partial}{\partial \eta_{\phi,i}}\left[-\log p(y_i \mid \mu_i^{(t+1)}, \phi_i^{(t)})\right]$
    Fit EBM base-learner $h_\phi^{(t)}$ to $\{(x_i, r_{\phi,i}^{(t)})\}_{i=1}^n$
    Update $f_\phi^{(t+1)} \leftarrow f_\phi^{(t)} + \nu_\phi h_\phi^{(t)}$
    if early stopping criterion satisfied (e.g., $|L^{(t+1)} - L^{(t)}| < \varepsilon$) **then**
        break
    end if
end for
Output: $\hat{\mu}(x) = g_\mu^{-1}(f_\mu(x))$ and $\hat{\phi}(x) = g_\phi^{-1}(f_\phi(x))$

Detailed Algorithm Steps:

Step 1: Initialization (Line 4) Initialize mean and dispersion functions as constants: $f_\mu^{(0)}(x) = c_\mu$ and $f_\phi^{(0)}(x) = c_\phi$, typically based on overall mean and variance of the response.

Step 2: Mean Boosting (Lines 9-12) Compute the gradient of negative log-likelihood with respect to mean linear predictor. For Gaussian distribution with identity link for mean and log link for dispersion:

$$r_{\mu,i}^{(t)} = \frac{y_i - \mu_i^{(t)}}{\phi_i^{(t)}}.$$

Fit EBM base-learner $h_\mu^{(t)}$ to pseudo-residuals and update mean function with learning rate ν_μ.

Step 3: Dispersion Boosting (Lines 13-16) After updating mean, compute gradient with respect to dispersion. For Gaussian distribution with log link:

$$r_{\phi,i}^{(t)} = \frac{(y_i - \mu_i^{(t+1)})^2}{\phi_i^{(t)}} - 1.$$

Fit EBM base-learner $h_\phi^{(t)}$ to dispersion pseudo-residuals and update with learning rate ν_ϕ.

Step 4: Convergence Check (Lines 17-18) Monitor change in negative log-likelihood. If $|L^{(t+1)} - L^{(t)}| < \varepsilon$, terminate to prevent overfitting.

This alternating boosting approach allows EBM to learn complex mean and dispersion functions while maintaining interpretability through additive structure and enabling detailed examination of individual feature contributions through partial dependence plots.

4.5 XGBoost

Extreme Gradient Boosting (XGBoost) is a scalable and efficient machine learning algorithm that combines weak learners, typically regression trees, to create a powerful predictive ensemble model (Chen and Guestrin, 2016). XGBoost employs gradient boosting, a sequential ensemble technique where new trees are built to model the residuals (prediction errors) from previous trees, progressively focusing on the unexplained variance in the training set.

4.5.1 XGBoost Model Structure

The XGBoost algorithm constructs an additive ensemble of regression trees. For a dataset with n observations, the prediction is given by:

$$\hat{y}_i = \phi(x_i) = \sum_{k=1}^{K} f_k(x_i),$$

where K is the number of trees, f_k represents the k-th tree function, and $\phi(x_i)$ is the ensemble prediction for observation i.

At the beginning of the algorithm, each observation is predicted using a constant initialization (typically the mean of the response variable). The algorithm then sequentially adds trees, with each new tree fit to approximate the negative gradient (first derivative) and curvature (second derivative) of the loss function with respect to current predictions.

4.5.2 Objective Function

The XGBoost training objective combines a loss function with a regularization term to control model complexity:

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(\hat{y}_i, y_i) + \sum_{k=1}^{K} \Omega(f_k),$$

where:

- $l(\hat{y}_i, y_i)$ is the differentiable convex loss function measuring prediction error,
- $\Omega(f_k)$ is the regularization term that penalizes model complexity.

The regularization term is defined as:

$$\Omega(f_k) = \gamma T_k + \frac{1}{2}\lambda \|w_k\|^2,$$

where T_k is the number of leaf nodes in tree k, w_k are the leaf weights, γ controls the penalty for the number of leaves, and λ is the L2 regularization parameter on leaf weights. This regularization prevents overfitting by promoting simpler tree structures.

4.5.3 Second-Order Taylor Approximation

XGBoost uses a second-order Taylor expansion of the loss function to efficiently determine optimal tree structures. At iteration t, the objective becomes:

$$\mathcal{L}^{(t)} \approx \sum_{i=1}^{n} \left[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t),$$

where:

$$g_i = \frac{\partial l(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}},$$
$$h_i = \frac{\partial^2 l(y_i, \hat{y}_i^{(t-1)})}{\partial (\hat{y}_i^{(t-1)})^2}.$$

These first and second derivatives (gradient and Hessian) guide the tree-building process, allowing XGBoost to efficiently find optimal split points and leaf values.

4.5.4 Parameter Estimation for DGAM with XGBoost

For Double Generalized Additive Models, XGBoost is applied in an alternating fashion to model both the mean and dispersion structures. The mean and dispersion are represented as boosted ensembles:

$$\eta_{\mu,i} = F_\mu(x_i) = \sum_{k=1}^{M_\mu} f_{\mu,k}(x_i),$$
$$\eta_{\phi,i} = F_\phi(x_i) = \sum_{k=1}^{M_\phi} f_{\phi,k}(x_i),$$

where $F_\mu(\cdot)$ and $F_\phi(\cdot)$ are ensembles of M_μ and M_ϕ trees, respectively, and $f_{\mu,k}$ and $f_{\phi,k}$ are individual tree functions.

The mean and dispersion parameters are obtained via inverse link functions:

$$\mu_i = g_\mu^{-1}(\eta_{\mu,i}) = g_\mu^{-1}(F_\mu(x_i)),$$
$$\phi_i = g_\phi^{-1}(\eta_{\phi,i}) = g_\phi^{-1}(F_\phi(x_i)).$$

4.5.5 Algorithm: XGBoost-DGAM via Alternating Second-Order Boosting

The estimation of DGAM parameters using XGBoost proceeds through alternating second-order boosting for location and dispersion, as described in Algorithm 3.

Algorithm 3 XGBoost-DGAM via Alternating Second-Order Boosting for Location and Dispersion

Input: $\{(y_i, x_i)\}_{i=1}^{n}$, link functions g_μ, g_ϕ, distribution $p(y \mid \mu, \phi)$

Hyperparameters: Mean-stage trees M_μ, dispersion-stage trees M_ϕ, tree depth, subsampling rates, regularization parameters (γ, λ), learning rates ν_μ, ν_ϕ, stopping tolerance ε

Trainable parameters: Boosted ensembles $F_\mu(\cdot), F_\phi(\cdot)$ with $\eta_\mu = F_\mu(x)$, $\eta_\phi = F_\phi(x)$

Initialize $F_\mu^{(0)}(x) \equiv c_\mu$ and $F_\phi^{(0)}(x) \equiv c_\phi$

repeat

  Compute $\eta_{\mu,i} = F_\mu(x_i)$, $\eta_{\phi,i} = F_\phi(x_i)$

  Update $\mu_i = g_\mu^{-1}(\eta_{\mu,i})$, $\phi_i = g_\phi^{-1}(\eta_{\phi,i})$

  Mean update: Compute first and second derivatives

  $g_{\mu,i} = \frac{\partial}{\partial \eta_{\mu,i}} \left[-\log p(y_i \mid \mu_i, \phi_i)\right]$

  $h_{\mu,i} = \frac{\partial^2}{\partial \eta_{\mu,i}^2} \left[-\log p(y_i \mid \mu_i, \phi_i)\right]$

  Fit next mean tree t_μ by minimizing the quadratic approximation:

    $\mathcal{L}_\mu^{(t)} = \sum_{i=1}^{n} \left[g_{\mu,i} t_\mu(x_i) + \frac{1}{2} h_{\mu,i} t_\mu^2(x_i)\right] + \Omega(t_\mu)$

  Update $F_\mu \leftarrow F_\mu + \nu_\mu t_\mu$ and recompute μ_i

  Dispersion update: Compute first and second derivatives conditional on updated mean

  $g_{\phi,i} = \frac{\partial}{\partial \eta_{\phi,i}} \left[-\log p(y_i \mid \mu_i, \phi_i)\right]$

  $h_{\phi,i} = \frac{\partial^2}{\partial \eta_{\phi,i}^2} \left[-\log p(y_i \mid \mu_i, \phi_i)\right]$

  Fit next dispersion tree t_ϕ by minimizing the quadratic approximation:

    $\mathcal{L}_\phi^{(t)} = \sum_{i=1}^{n} \left[g_{\phi,i} t_\phi(x_i) + \frac{1}{2} h_{\phi,i} t_\phi^2(x_i)\right] + \Omega(t_\phi)$

  Update $F_\phi \leftarrow F_\phi + \nu_\phi t_\phi$

  if improvement in held-out NLL $< \varepsilon$ (or max trees reached) **then**

    break

  end if

until convergence

Output: $\hat{\mu}(x) = g_\mu^{-1}(F_\mu(x))$ and $\hat{\phi}(x) = g_\phi^{-1}(F_\phi(x))$

Detailed Algorithm Steps:

Step 1: Initialization (Line 4) Initialize the mean and dispersion ensembles as constants: $F_\mu^{(0)}(x) = c_\mu$ and $F_\phi^{(0)}(x) = c_\phi$. For Gaussian responses, c_μ is typically set to $\bar{y}$ (sample mean) and c_ϕ to $\log(s^2)$ where s^2 is the sample variance.

Step 2: Mean Boosting (Lines 8-12) For Gaussian distribution with identity link for mean:

$$g_{\mu,i} = -\frac{y_i - \mu_i}{\phi_i},$$
$$h_{\mu,i} = \frac{1}{\phi_i}.$$

Build a regression tree t_μ that minimizes the second-order approximation of the loss, using $(g_{\mu,i}, h_{\mu,i})$ to determine optimal splits and leaf values. The optimal leaf weight for a node is:

$$w^* = -\frac{\sum_{i \in \text{leaf}} g_{\mu,i}}{\sum_{i \in \text{leaf}} h_{\mu,i} + \lambda}.$$

Step 3: Dispersion Boosting (Lines 13-17) After updating the mean, compute gradients for the dispersion. For Gaussian distribution with log link for dispersion:

$$g_{\phi,i} = -\frac{1}{2}\left[\frac{(y_i - \mu_i)^2}{\phi_i} - 1\right],$$
$$h_{\phi,i} = \frac{1}{2}.$$

Build a regression tree t_ϕ for the dispersion update using these gradients.

Step 4: Regularization and Early Stopping (Lines 18-19) The regularization term $\Omega(f) = \gamma T + \frac{1}{2}\lambda\|w\|^2$ penalizes tree complexity. Monitor performance on a validation set and stop when improvement falls below tolerance ε to prevent overfitting.

This alternating second-order boosting approach allows XGBoost to efficiently learn complex nonlinear relationships in both mean and dispersion while maintaining computational efficiency through the use of gradient and Hessian information.

4.6 Random Forest

Random Forest (RF), introduced by Breiman (2001), is a powerful ensemble learning method that improves prediction accuracy and robustness by combining multiple decision trees. Unlike single decision trees that can suffer from high variance and overfitting, Random Forest creates a collection of diverse trees through bootstrap sampling and random feature selection, then aggregates their predictions to produce stable and accurate results.

4.6.1 Random Forest Model Structure

A Random Forest consists of a collection of tree-based predictors, denoted as $\{f(\cdot, \Theta_k) : k = 1, 2, \ldots, K\}$, where Θ_k represents independent and identically distributed random vectors that govern the construction of each tree. For regression problems, the Random Forest prediction is obtained by averaging the predictions of all trees:

$$\hat{y}(x) = \frac{1}{K} \sum_{k=1}^{K} f(x, \Theta_k),$$

where K is the number of trees in the forest and $f(x, \Theta_k)$ is the prediction from the k-th tree.

For classification problems, the Random Forest employs a majority voting scheme, where each tree casts a vote and the class with the most votes is selected:

$$\hat{y}_{\text{class}}(x) = \underset{j \in \{1, \ldots, J\}}{\arg\max} \sum_{k=1}^{K} \mathbb{I}[f(x, \Theta_k) = j],$$

where J is the number of classes and $\mathbb{I}[\cdot]$ is the indicator function.

4.6.2 Random Forest Algorithm

The Random Forest algorithm constructs an ensemble of decision trees through the following process:

Input: A dataset $S = \{(y^1, x^1), \ldots, (y^n, x^n)\}$, where for classification $y \in \{1, \ldots, J\}$ or for regression $y \in \mathbb{R}$, and $x \in \mathbb{R}^p$ is in p-dimensional feature space. Hyperparameters include the number of trees K, the number of randomly selected features at each split d (typically $d \approx \sqrt{p}$ for classification or $d \approx p/3$ for regression), and the minimum leaf size ℓ.

Tree Construction (for $k = 1, 2, \ldots, K$):

1. Create a bootstrap sample S_k by randomly selecting n observations from the training set S with replacement. Approximately 63% of unique observations will be included, with the remaining 37% forming the out-of-bag (OOB) sample.

2. At each node of the tree:

   - Randomly select d input variables from $\{X_1, \ldots, X_p\}$
   - Based on these d variables and the data S_k at this node, determine the best split to partition the data (e.g., minimizing mean squared error for regression or Gini impurity for classification)

3. Grow the tree recursively until each terminal (leaf) node contains no more than ℓ training samples. No pruning is performed.

4. Let $f(\cdot, S_k)$ represent the constructed tree.

Output: For regression, the Random Forest predictor is:

$$f_{\text{RF}}(x, S) = \frac{1}{K} \sum_{k=1}^{K} f(x, S_k).$$

For classification, the Random Forest classifier is:

$$f_{\text{RF}}(x, S) = \operatorname*{arg\,max}_{j \in \{1, \ldots, J\}} \sum_{k=1}^{K} \mathbb{I}[f(x, S_k) = j].$$

The diversity among trees, created through bootstrap sampling and random feature selection, reduces variance while maintaining low bias, resulting in excellent predictive performance (Breiman, 2001; Hastie et al., 2009).

4.6.3 Parameter Estimation for DGAM with Random Forest

For Double Generalized Additive Models, Random Forest is applied in an alternating fashion to model both the mean and dispersion structures. The mean and dispersion are represented as forest predictors:

$$\eta_{\mu,i} = R_\mu(x_i) = \frac{1}{N_\mu}\sum_{k=1}^{N_\mu} f_{\mu,k}(x_i),$$
$$\eta_{\phi,i} = R_\phi(x_i) = \frac{1}{N_\phi}\sum_{k=1}^{N_\phi} f_{\phi,k}(x_i),$$

where $R_\mu(\cdot)$ and $R_\phi(\cdot)$ are Random Forest ensembles with N_μ and N_ϕ trees, respectively, and $f_{\mu,k}$ and $f_{\phi,k}$ are individual regression trees.

The mean and dispersion parameters are obtained via inverse link functions:

$$\mu_i = g_\mu^{-1}(\eta_{\mu,i}) = g_\mu^{-1}(R_\mu(x_i)),$$
$$\phi_i = g_\phi^{-1}(\eta_{\phi,i}) = g_\phi^{-1}(R_\phi(x_i)).$$

4.6.4 Algorithm: Random Forest DGAM via Alternating Residual Forests

The estimation of DGAM parameters using Random Forest proceeds through alternating residual forests for mean and dispersion, as described in Algorithm 4.

Algorithm 4 Random Forest DGAM via Alternating Residual Forests for Mean and Dispersion

1: **Input:** $\{(y_i, x_i)\}_{i=1}^n$, link functions g_μ, g_ϕ, distribution $p(y \mid \mu, \phi)$
2: **Hyperparameters:** Forest sizes (N_μ, N_ϕ), max depth, min leaf size, bootstrap scheme, stability constant $\delta > 0$, stopping tolerance ε
3: **Trainable parameters:** Forest predictors $R_\mu(\cdot), R_\phi(\cdot)$ with $\eta_\mu = R_\mu(x)$, $\eta_\phi = R_\phi(x)$
4: Fit initial mean forest $R_\mu^{(0)}$ on $\{(x_i, y_i)\}_{i=1}^n$ and set $\eta_{\mu,i}^{(0)} \leftarrow R_\mu^{(0)}(x_i)$
5: Initialize dispersion predictor $R_\phi^{(0)}(x) \equiv c_\phi$
6: **repeat**
7:     Update $\mu_i \leftarrow g_\mu^{-1}(\eta_{\mu,i})$ and $\phi_i \leftarrow g_\phi^{-1}(R_\phi(x_i))$
8:     Compute mean residuals $r_{\mu,i} \leftarrow y_i - \mu_i$ (or NLL pseudo-residuals)
9:     Refit/update mean forest R_μ on $\{(x_i, r_{\mu,i})\}_{i=1}^n$
10:     Update $\eta_{\mu,i} \leftarrow \eta_{\mu,i} + R_\mu(x_i)$
11:     Recompute μ_i using the updated mean predictor
12:     Construct dispersion targets $z_{\phi,i} \leftarrow \log\left((y_i - \mu_i)^2 + \delta\right)$
13:     Fit/update dispersion forest R_ϕ on $\{(x_i, z_{\phi,i})\}_{i=1}^n$
14:     **if** parameter changes (or held-out NLL change) $< \varepsilon$ **then**
15:         **break**
16:     **end if**
17: **until** convergence
18: **Output:** $\hat{\mu}(x) = g_\mu^{-1}(\eta_\mu(x))$ and $\hat{\phi}(x) = g_\phi^{-1}(R_\phi(x))$

Detailed Algorithm Steps:

Step 1: Initialization (Lines 4-5) Fit an initial Random Forest $R_\mu^{(0)}$ directly to the response variable y_i to obtain initial mean predictions. For Gaussian responses with identity link, this provides $\eta_{\mu,i}^{(0)} = R_\mu^{(0)}(x_i)$.

Initialize the dispersion predictor as a constant: $R_\phi^{(0)}(x) = c_\phi$, where $c_\phi = \log(s^2)$ and s^2 is the sample variance of residuals from the initial mean forest.

Step 2: Mean Forest Update (Lines 7-10) Compute residuals from the current mean predictions:

$$r_{\mu,i} = y_i - \mu_i.$$

For Gaussian DGAM, these are the simple residuals. Alternatively, negative log-likelihood pseudo-residuals can be used:

$$r_{\mu,i} = \frac{y_i - \mu_i}{\phi_i}.$$

Fit a new Random Forest to predict these residuals, then update the mean predictor by adding the residual predictions. This iterative residual fitting helps refine the mean model.

Step 3: Dispersion Forest Update (Lines 11-12) After updating the mean, construct dispersion targets based on squared residuals:

$$z_{\phi,i} = \log\left((y_i - \mu_i)^2 + \delta\right),$$

where the small constant $\delta > 0$ (e.g., $\delta = 0.01$) ensures numerical stability when residuals are very small. The logarithm is used because the dispersion model employs a log link.

Fit a Random Forest R_ϕ to predict these log-squared-residuals, which provides estimates of $\log(\phi_i)$ that can vary with covariates.

Step 4: Convergence Check (Lines 13-14) Monitor convergence by tracking changes in parameter estimates or the negative log-likelihood on a held-out validation set. Stop when changes fall below tolerance ε to prevent overfitting.

4.7 Support Vector Regression

Support Vector Regression (SVR), introduced by Vapnik (1995), is a powerful machine learning algorithm based on statistical learning theory that extends the principles of Support Vector Machines to regression problems. SVR aims to find a function that approximates the relationship between input features and continuous output values while maintaining a balance between model complexity and prediction accuracy (Smola and Schölkopf, 2004).

4.7.1 SVR Formulation

Unlike traditional regression methods that minimize the sum of squared errors, SVR employs an ε-insensitive loss function that ignores errors smaller than a threshold ε. This approach creates a margin of tolerance around the regression function, focusing computational effort on predictions that deviate significantly from observed values. The algorithm identifies support vectors—data points that lie outside or on the boundary of this ε-tube—which are critical in determining the regression function.

The SVR optimization problem can be formulated as follows. Given a training dataset $\{(x_i, y_i)\}_{i=1}^n$ where $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$, SVR seeks to find a function:

$$f(x) = \langle w, \phi(x) \rangle + b,$$

where w is the weight vector, $\phi(x)$ maps input features to a higher-dimensional feature space, and b is the bias term. The optimization objective is:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*),$$

subject to the constraints:

$$y_i - \langle w, \phi(x_i) \rangle - b \leq \varepsilon + \xi_i,$$

$$\langle w, \phi(x_i) \rangle + b - y_i \leq \varepsilon + \xi_i^*,$$

$$\xi_i, \xi_i^* \geq 0,$$

where $C > 0$ is a regularization parameter that controls the trade-off between model flatness and tolerance for deviations larger than ε, and ξ_i, ξ_i^* are slack variables that allow some points to fall outside the ε-tube.

4.7.2 Kernel Methods and Dual Formulation

The dual formulation of this optimization problem involves Lagrange multipliers α_i and α_i^*, leading to the prediction function:

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K(x_i, x) + b,$$

where $K(x_i, x) = \langle \phi(x_i), \phi(x) \rangle$ is a kernel function. Common kernel choices include:

- Linear kernel: $K(x_i, x_j) = \langle x_i, x_j \rangle$
- Polynomial kernel: $K(x_i, x_j) = (\gamma\langle x_i, x_j \rangle + r)^d$
- Radial Basis Function (RBF) kernel: $K(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|^2)$
- Sigmoid kernel: $K(x_i, x_j) = \tanh(\gamma\langle x_i, x_j \rangle + r)$

The kernel function allows SVR to capture nonlinear relationships by implicitly mapping data to higher-dimensional spaces without explicitly computing the transformation $\phi(x)$, a technique known as the "kernel trick."

4.7.3 SVR Model Structure

For a given input x, the SVR prediction is obtained as a weighted sum of kernel evaluations with the support vectors:

$$\hat{y}(x) = \sum_{i \in SV} (\alpha_i - \alpha_i^*) K(x_i, x) + b,$$

where SV denotes the set of support vectors (observations for which $\alpha_i \neq 0$ or $\alpha_i^* \neq 0$). The sparsity of the solution—only support vectors contribute to predictions—is a key computational advantage of SVR.

4.7.4 Parameter Estimation for DGAM with SVR

For Double Generalized Additive Models, SVR is applied separately to model both the mean and dispersion structures. The mean and dispersion are represented as:

$$\eta_{\mu,i} = f_\mu^{SVR}(x_i) = \sum_{j \in SV_\mu} (\alpha_{\mu,j} - \alpha_{\mu,j}^*) K_\mu(x_j, x_i) + b_\mu,$$

$$\eta_{\phi,i} = f_\phi^{SVR}(x_i) = \sum_{j \in SV_\phi} (\alpha_{\phi,j} - \alpha_{\phi,j}^*) K_\phi(x_j, x_i) + b_\phi,$$

where $f_\mu^{SVR}(\cdot)$ and $f_\phi^{SVR}(\cdot)$ are SVR models with potentially different kernel functions and hyperparameters, SV_μ and SV_ϕ are the sets of support vectors for mean and dispersion models, respectively.

The mean and dispersion parameters are obtained via inverse link functions:

$$\mu_i = g_\mu^{-1}(\eta_{\mu,i}) = g_\mu^{-1}(f_\mu^{SVR}(x_i)),$$

$$\phi_i = g_\phi^{-1}(\eta_{\phi,i}) = g_\phi^{-1}(f_\phi^{SVR}(x_i)).$$

4.7.5 Algorithm: SVR-DGAM via Alternating Optimization

The estimation of DGAM parameters using SVR proceeds through alternating optimization of mean and dispersion models, as described in Algorithm 5.

Algorithm 5 SVR-DGAM via Alternating Optimization for Mean and Dispersion

---

Input: $\{(y_i, x_i)\}_{i=1}^n$, link functions g_μ, g_ϕ, distribution $p(y \mid \mu, \phi)$
Hyperparameters: Regularization parameters (C_μ, C_ϕ), ε-tube widths $(\varepsilon_\mu, \varepsilon_\phi)$, kernel functions (K_μ, K_ϕ) with parameters $(\gamma_\mu, \gamma_\phi)$, convergence tolerance δ
Trainable parameters: SVR predictors $f_\mu^{SVR}(\cdot), f_\phi^{SVR}(\cdot)$ with $\eta_\mu = f_\mu^{SVR}(x), \eta_\phi = f_\phi^{SVR}(x)$
Fit initial mean SVR model $f_\mu^{SVR,(0)}$ on $\{(x_i, y_i)\}_{i=1}^n$ and compute $\eta_{\mu,i}^{(0)} \leftarrow f_\mu^{SVR,(0)}(x_i)$
Initialize dispersion: compute initial residuals $e_i = y_i - g_\mu^{-1}(\eta_{\mu,i}^{(0)})$
Set initial dispersion targets $z_{\phi,i} \leftarrow \log(e_i^2 + 0.01)$
Fit initial dispersion SVR model $f_\phi^{SVR,(0)}$ on $\{(x_i, z_{\phi,i})\}_{i=1}^n$
repeat
  Compute $\eta_{\mu,i} \leftarrow f_\mu^{SVR}(x_i)$ and $\eta_{\phi,i} \leftarrow f_\phi^{SVR}(x_i)$
  Update $\mu_i \leftarrow g_\mu^{-1}(\eta_{\mu,i})$ and $\phi_i \leftarrow g_\phi^{-1}(\eta_{\phi,i})$
  Mean update: Compute weighted residuals
    $r_{\mu,i} \leftarrow \frac{y_i - \mu_i}{\sqrt{\phi_i}}$ (weighted by inverse dispersion)
  Fit SVR model f_μ^{SVR} on $\{(x_i, r_{\mu,i})\}_{i=1}^n$ with hyperparameters $(C_\mu, \varepsilon_\mu, K_\mu)$
  Solve: $\min_{w_\mu, b_\mu} \frac{1}{2}\|w_\mu\|^2 + C_\mu \sum_{i=1}^n (\xi_{\mu,i} + \xi_{\mu,i}^*)$
    subject to SVR constraints with ε_μ-insensitive loss
  Update mean predictor and recompute μ_i
  Dispersion update: Construct dispersion targets
    $z_{\phi,i} \leftarrow \log\left((y_i - \mu_i)^2 + 0.01\right)$
  Fit SVR model f_ϕ^{SVR} on $\{(x_i, z_{\phi,i})\}_{i=1}^n$ with hyperparameters $(C_\phi, \varepsilon_\phi, K_\phi)$
  Solve: $\min_{w_\phi, b_\phi} \frac{1}{2}\|w_\phi\|^2 + C_\phi \sum_{i=1}^n (\xi_{\phi,i} + \xi_{\phi,i}^*)$
    subject to SVR constraints with ε_ϕ-insensitive loss
  Evaluate negative log-likelihood $L = \sum_{i=1}^n - \log p(y_i \mid \mu_i, \phi_i)$
  if change in $L < \delta$ or max iterations reached **then**
    break
  end if
until convergence
Output: $\hat{\mu}(x) = g_\mu^{-1}(f_\mu^{SVR}(x))$ and $\hat{\phi}(x) = g_\phi^{-1}(f_\phi^{SVR}(x))$

Detailed Algorithm Steps:

Step 1: Initialization (Lines 4-7) Fit an initial SVR model for the mean directly to the response variable y_i. For Gaussian responses with identity link, this provides initial predictions $\eta_{\mu,i}^{(0)} = f_\mu^{SVR,(0)}(x_i)$.

Compute initial residuals $e_i = y_i - g_\mu^{-1}(\eta_{\mu,i}^{(0)})$ and construct log-squared-residuals as initial dispersion targets:

$$z_{\phi,i} = \log(e_i^2 + 0.01),$$

where the small constant 0.01 ensures numerical stability.

Fit an initial SVR model $f_\phi^{SVR,(0)}$ to predict these log-squared-residuals, providing initial dispersion estimates.

Step 2: Mean SVR Update (Lines 11-15) Compute weighted residuals that account for heteroscedasticity:

$$r_{\mu,i} = \frac{y_i - \mu_i}{\sqrt{\phi_i}}.$$

For Gaussian DGAM with identity link for mean, the weighting by $1/\sqrt{\phi_i}$ gives more importance to observations with smaller variance, improving efficiency.

Fit an SVR model by solving the primal optimization problem:

$$\min_{w_\mu, b_\mu, \xi_\mu, \xi_\mu^*} \frac{1}{2}\|w_\mu\|^2 + C_\mu \sum_{i=1}^{n} (\xi_{\mu,i} + \xi_{\mu,i}^*),$$

subject to:

$$r_{\mu,i} - (\langle w_\mu, \phi(x_i) \rangle + b_\mu) \leq \varepsilon_\mu + \xi_{\mu,i},$$
$$(\langle w_\mu, \phi(x_i) \rangle + b_\mu) - r_{\mu,i} \leq \varepsilon_\mu + \xi_{\mu,i}^*,$$
$$\xi_{\mu,i}, \xi_{\mu,i}^* \geq 0.$$

The dual formulation yields:

$$f_\mu^{SVR}(x) = \sum_{j \in SV_\mu} (\alpha_{\mu,j} - \alpha_{\mu,j}^*) K_\mu(x_j, x) + b_\mu,$$

where $\alpha_{\mu,j}$ and $\alpha_{\mu,j}^*$ are Lagrange multipliers obtained via quadratic programming.

Step 3: Dispersion SVR Update (Lines 16-20) After updating the mean, construct new dispersion targets based on current residuals:

$$z_{\phi,i} = \log\left((y_i - \mu_i)^2 + 0.01\right).$$

Fit an SVR model to predict these log-squared-residuals by solving:

$$\min_{w_\phi, b_\phi, \xi_\phi, \xi_\phi^*} \frac{1}{2}\|w_\phi\|^2 + C_\phi \sum_{i=1}^{n} (\xi_{\phi,i} + \xi_{\phi,i}^*),$$

subject to analogous ε_ϕ-insensitive constraints.

The resulting dispersion predictor is:

$$f_\phi^{SVR}(x) = \sum_{j \in SV_\phi} (\alpha_{\phi,j} - \alpha_{\phi,j}^*) K_\phi(x_j, x) + b_\phi.$$

For Gaussian DGAM with log link for dispersion, this directly provides $\log(\phi_i)$ estimates.

Step 4: Kernel Function Selection

The choice of kernel function $K(\cdot,\cdot)$ is crucial for SVR performance. For both mean and dispersion models, common choices include:

Radial Basis Function (RBF) kernel:

$$K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right),$$

where $\gamma > 0$ controls the kernel width. RBF kernels are effective for capturing smooth nonlinear relationships and are often the default choice.

Polynomial kernel:

$$K(x_i, x_j) = (\gamma \langle x_i, x_j \rangle + r)^d,$$

where d is the polynomial degree. This is useful when relationships are believed to be polynomial in nature.

Step 5: Hyperparameter Tuning

The performance of SVR-DGAM depends critically on hyperparameter selection:

- **Regularization parameters** (C_μ, C_ϕ): Larger values allow more training errors but may lead to overfitting; smaller values enforce stronger regularization.
- **ε-tube widths** $(\varepsilon_\mu, \varepsilon_\phi)$: Larger values create wider tubes, ignoring more small errors and producing sparser solutions.
- **Kernel parameters** $(\gamma_\mu, \gamma_\phi)$: For RBF kernels, smaller γ creates smoother functions; larger γ allows more local variation.

These hyperparameters are typically optimized using cross-validation or grid search procedures.

Step 6: Convergence Check (Lines 21-23) Monitor the negative log-likelihood:

$$L = \sum_{i=1}^{n} -\log p(y_i \mid \mu_i, \phi_i).$$

For Gaussian distribution:

$$L = \sum_{i=1}^{n} \left[\frac{1}{2} \log(2\pi\phi_i) + \frac{(y_i - \mu_i)^2}{2\phi_i} \right].$$

Terminate when the change in L falls below tolerance δ or maximum iterations are reached.

4.8 Results (Phase II): Model Implementation and Empirical Validation

Phase II operationalized the proposed Double Generalized Additive Model (DGAM) algorithms within a unified computational framework and evaluated them against the conventional Generalized Additive Model (GAM) across all experimental tables and train–validation–test splits. The primary objective of Phase II was not only to assess point prediction accuracy, but also to verify whether explicitly modeling dispersion improves probabilistic fit and uncertainty quantification in settings where heteroscedasticity is expected.

Dataset heterogeneity and variable structure. This study focuses on four publicly available healthcare datasets from Kaggle:

1. **Fetal Health Monitoring Dataset:** 2,126 cardiotocography records with 21 features for fetal health assessment during pregnancy and labor.

2. **Life Style Data:** Lifestyle and behavioral information including physical activities, dietary patterns, and physiological measurements.

3. **Health Checkup Result Dataset:** Clinical examination results from approximately 19,000,000 individuals in South Korea (2002–2020), with a stratified sample used for computational feasibility.

4. **Life Expectancy (WHO) Fixed Dataset:** Country-level health indicators for 193 countries (2000–2015) with 22 features including immunization, mortality, and socioeconomic factors.

4.8.1 Target Variables

Sixteen target variables were selected across four datasets (four per dataset):

Table 4.1 Target Variables by Dataset and Life Cycle Stage.

Stage	ID	Variable Name	Description
Birth	1.1	Abnormal_STV	Percentage of time with abnormal short-term variability
	1.2	Abnormal_LTV	Percentage of time with abnormal long-term variability
	1.3	Mean_STV	Mean value of short-term variability
	1.4	Mean_LTV	Mean value of short-term variability
Aging	2.1	Calories_Burned	Total calories burned during activities
	2.2	Fat_Percentage	Body fat percentage
	2.3	Avg_BPM	Average heart rate (beats per minute)
	2.4	Max_BPM	Maximum heart rate (beats per minute)
Illness	3.1	Creatinine	Serum creatinine (kidney function)
	3.2	GGT	Gamma-glutamyl transferase (liver enzyme)
	3.3	SGOT	Serum glutamic-oxaloacetic transaminase (AST)
	3.4	SGPT	Serum glutamic-pyruvic transaminase (ALT)
Death	4.1	Life_expectancy	Average life expectancy in years
	4.2	Adult_mortality	Adult mortality rate per 1,000 population
	4.3	Under_five_deaths	Deaths under age five per 1,000 births
	4.4	Infant_deaths	Infant deaths per 1,000 live births

4.8.2 Modeling Techniques

Five techniques were employed for modeling the mean and dispersion functions:

1. **Spline:** Smooth spline with ridge regularization
2. **EBM:** Explainable Boosting Machine with cyclic gradient boosting
3. **XGB:** Extreme Gradient Boosting with tree-based learners
4. **RF:** Random Forest ensemble method
5. **SVR:** Support Vector Regression with RBF kernel

4.8.3 Data Splitting Ratios

Three splitting ratios were investigated:

- Split A: Training (60%), Validation (20%), Test (20%)
- Split B: Training (70%), Validation (15%), Test (15%)
- Split C: Training (80%), Validation (10%), Test (10%)

Linking Phase I to Phase II. Phase I motivated the Double Generalized Additive Model (DGAM) by arguing that a mean-only additive predictor can be insufficient when dispersion varies systematically with covariates, i.e., under heteroscedasticity. Phase II operationalizes these theoretical claims through an end-to-end Python implementation and evaluates DGAM empirically across multiple healthcare datasets and experimental splits. Consequently, the Phase II results are interpreted as a direct test of the Phase I hypothesis: explicitly modeling dispersion should improve distributional fit and uncertainty quantification, even when improvements in point prediction accuracy are not guaranteed.

4.8.4 Experimental Grid and Paired-Comparison Protocol

The Phase II benchmark comprises 16 prediction tasks (four targets per dataset across four datasets), each evaluated under three train–validation–test splits (60/20/20, 70/15/15, 80/10/10) and five mean-function learners. This design yields $16 \times 3 \times 5 = 240$ matched configurations. For each configuration, GAM is fitted under a constant-dispersion assumption. DGAM is fitted by jointly learning (i) a mean submodel and (ii) a dispersion submodel. To ensure a fair one-to-one comparison, DGAM is represented by the dispersion learner with minimum negative log-likelihood (NLL) within each matched configuration. Win rates are reported as the proportion of matched configurations in which DGAM outperforms GAM on the corresponding metric.

4.8.5 Overall Comparative Performance: DGAM versus GAM

Table 4.2 reports overall win rates across all $n = 240$ paired comparisons. The most salient result is that DGAM substantially improves probabilistic fit. DGAM outperforms GAM on NLL in 90.0% of comparisons (216/240) and on CRPS in 90.8% (218/240), providing strong empirical evidence that explicit dispersion modeling yields a systematically better predictive distribution in the evaluated settings.

Calibration and coverage improve to a moderate extent. DGAM attains better empirical 95% prediction interval coverage (PICP95) in 56.7% of comparisons (136/240). This indicates that DGAM more frequently produces uncertainty intervals whose empirical coverage is closer to the nominal level than GAM, although not uniformly so across all tasks.

In contrast, point-prediction accuracy is not improved in the majority of cases. DGAM improves MAE in 6.2% of comparisons (15/240), RMSE in 7.1% (17/240), and R^2 in 7.9% (19/240). This divergence between probabilistic and point metrics is consistent with the methodological intent of DGAM: the principal benefit of explicitly learning dispersion is expected to manifest in likelihood-based criteria and proper scoring rules rather than in uniformly lower point-error metrics.

Finally, DGAM improves AIC in 75.0% of comparisons (180/240). Since AIC penalizes model complexity, this result suggests that DGAM's likelihood gains typically outweigh the additional flexibility introduced by the dispersion component.

Table 4.2 Overall win-rate of DGAM relative to GAM across all paired comparisons ($n = 240$). For AIC, NLL, CRPS, MAE, and RMSE, lower is better. For R^2 and PICP95, higher is better.

Metric	DGAM better	Win rate (%)
AIC ↓	180/240	75.0
NLL ↓	216/240	90.0
MAE ↓	15/240	6.2
RMSE ↓	17/240	7.1
R^2 ↑	19/240	7.9
CRPS ↓	218/240	90.8
PICP95 ↑	136/240	56.7

4.8.6 Robustness Across Splitting Ratios

To assess whether conclusions depend on the train–validation–test ratio, Table 4.3 reports win rates stratified by split (each split contributes $n = 80$ comparisons). The probabilistic advantage of DGAM is stable across splits. DGAM wins on NLL in 87.5–95.0% of comparisons and on CRPS in 88.8–95.0%. Coverage improvements are also consistent, with PICP95 win rates ranging from 52.5% to 58.8%.

By contrast, point-metric improvements remain low under all splits (typically below 12%), reinforcing that the empirical benefit of DGAM is primarily distributional rather than purely point-accuracy driven. Overall, the split-stratified analysis supports robustness: the key Phase II conclusion does not appear to be an artifact of a particular partitioning regime.

Table 4.3 DGAM win-rate (%) relative to GAM by train–validation–test split (each split has $n = 80$ paired comparisons).

Split	AIC ↓	NLL ↓	MAE ↓	RMSE ↓	R^2 ↑	CRPS ↓	PICP95 ↑
60/20/20	77.5	87.5	10.0	11.2	11.2	88.8	52.5
70/15/15	78.8	95.0	5.0	5.0	6.2	95.0	58.8
80/10/10	68.8	87.5	3.8	5.0	6.2	88.8	58.8

4.8.7 Heterogeneity Across Datasets

Table 4.4 reports win rates stratified by dataset (each dataset contributes $n = 60$ comparisons). DGAM improves NLL and CRPS in the majority of comparisons across all datasets (NLL: 83.3–96.7%; CRPS: 80.0–100.0%), suggesting that the benefit of explicit dispersion modeling generalizes across distinct healthcare prediction settings.

However, improvements in point-prediction metrics concentrate primarily in Dataset 1, whereas Datasets 2–4 show limited to no systematic advantage in MAE/RMSE/R^2. This result supports a nuanced interpretation: DGAM primarily enhances uncertainty modeling and distributional fit, while gains in mean accuracy are context dependent and may emerge only in subsets of tasks where improved dispersion modeling indirectly stabilizes mean estimation.

Table 4.4 DGAM win-rate (%) relative to GAM by dataset (each dataset has $n = 60$ paired comparisons).

Dataset	AIC ↓	NLL ↓	MAE ↓	RMSE ↓	R^2 ↑	CRPS ↓	PICP95 ↑
1	80.0	96.7	25.0	28.3	31.7	95.0	85.0
2	43.3	86.7	0.0	0.0	0.0	88.3	75.0
3	83.3	83.3	0.0	0.0	0.0	80.0	0.0
4	93.3	93.3	0.0	0.0	0.0	100.0	66.7

4.8.8 Most Frequently Selected Dispersion Technique Under DGAM

To interpret the empirical behavior of the dispersion component, Table 4.5 summarizes which dispersion learner is selected most frequently when DGAM is chosen by minimum NLL within each matched configuration. Random Forest (RF) and Support Vector Regression (SVR) dominate the selections (28.3% and 27.5%, respectively), followed by Explainable Boosting Machines (EBM, 21.7%). Spline-based and XGBoost-based dispersion models are selected less frequently (11.7% and 10.8%).

This selection pattern is consistent with the hypothesis that heteroscedasticity in real-world healthcare datasets may involve nonlinear and interaction-driven variance structures; flexible dispersion learners may therefore be better suited to capture such patterns and improve likelihood-based fit.

Table 4.5 Frequency with which each dispersion technique is selected as the best DGAM dispersion learner (selection by minimum NLL).

Dispersion technique	Count	Share (%)
Spline	28	11.7
EBM	52	21.7
XGB	26	10.8
RF	68	28.3
SVR	66	27.5

4.8.9 Magnitude of Improvements and Representative Examples

While win rates quantify how often DGAM outperforms GAM, they do not convey the magnitude of the improvements. To complement the win-rate analysis, Table 4.6 summarizes effect sizes of the paired differences $\Delta =$ DGAM $-$ GAM across all $n = 240$ matched comparisons. For likelihood-based fit, DGAM exhibits sizable gains: the mean paired difference is ΔNLL $= -232.950$ and the median is ΔNLL $= -88.698$, indicating that improvements are not driven solely by a small number of extreme cases. Similarly, the mean and median differences are negative for CRPS (mean ΔCRPS $= -0.479$, median ΔCRPS $= -0.172$) and AIC (mean ΔAIC $= -424.708$, median ΔAIC $= -138.718$), consistent with the overall conclusion that explicit dispersion learning primarily improves distributional fit.

In contrast, point-accuracy differences concentrate near zero. The median differences for MAE and RMSE are exactly zero, and the interquartile ranges also collapse at zero for these metrics, reflecting that DGAM frequently leaves point-prediction performance essentially unchanged relative to GAM. For interval calibration, the average change in 95% coverage is close to zero (mean ΔPICP95 ≈ 0; median ΔPICP95 $= 0.005$), aligning with the earlier win-rate finding that coverage improvements occur more often than not but are not universal.

To illustrate where the distributional improvements are most pronounced, Table 4.7 reports representative high-impact configurations selected by the largest improvements in NLL (most negative ΔNLL). These examples demonstrate that DGAM can substantially reduce NLL, often accompanied by meaningful reductions in CRPS, while leaving RMSE unchanged. This empirical pattern is coherent with the DGAM design objective: improving the predictive distribution via an explicit dispersion submodel, rather than re-optimizing the conditional mean.

For completeness and methodological transparency, Table 4.8 reports representative failure-mode cases where DGAM degrades NLL relative to GAM (largest positive ΔNLL). Even in these instances, CRPS may still improve, suggesting that distributional scoring rules can disagree in particular configurations and that dispersion learning can be sensitive to model selection and data characteristics. These failure cases motivate cautious deployment practices, such as monitoring likelihood-based diagnostics and using validation-based selection of the dispersion learner, which is the strategy adopted in this study.

Overall, the effect-size summaries and representative examples provide convergent evidence for the main Phase II claim. GAM implicitly assumes constant dispersion and therefore cannot adapt when residual variability changes systematically with covariates. When heteroscedasticity is present, a mean-only model can remain competitive in point-prediction metrics, yet its predictive distribution becomes mis-specified. This mis-specification is directly penalized by NLL and AIC and is reflected by proper scoring rules such as CRPS. By explicitly learning a dispersion function, DGAM produces predictive distributions that are better fitting and more informative for uncertainty-aware decision making, which is particularly relevant in health analytics settings where downstream decisions often depend on reliable uncertainty quantification rather than point estimates alone.

Table 4.6 Effect-size summary for DGAM−GAM across all paired comparisons ($n = 240$). Negative values indicate improvements for metrics where lower is better.

Metric	n	Mean	SD	Median	Q1	Q3
AIC ↓	240	-424.708	966.985	-138.718	-810.202	-0.097
NLL ↓	240	-232.950	476.934	-88.698	-432.607	-32.191
CRPS ↓	240	-0.479	1.160	-0.172	-0.648	-0.011
PICP95 ↑ (coverage)	240	-0.000	0.065	0.005	-0.018	0.021
MAE ↓	240	-0.111	0.740	0.000	0.000	0.000
RMSE ↓	240	-0.109	0.721	0.000	0.000	0.000
R^2 ↑	240	0.010	0.056	0.000	0.000	0.000

Table 4.7 Representative high-impact examples (largest improvements in NLL; most negative ΔNLL).

Table	Dataset	Target	Split	Mean	Dispersion	NLL (GAM)	NLL (DGAM)	ΔNLL	CRPS (GAM)	CRPS (DGAM)	ΔCRPS	RMSE (GAM)	RMSE (DGAM)	ΔRMSE
9	3	var1	70/15/15	ebm	ebm	4700.368	1926.798	-2773.570	0.152	0.145	-0.007	0.805	0.805	0.000
11	3	var3	60/20/20	ebm	svr	7068.459	5722.778	-1345.681	6.636	4.486	-2.150	20.062	20.062	0.000
11	3	var3	60/20/20	xgb	svr	7034.438	5695.333	-1339.105	6.509	4.412	-2.097	19.640	19.640	0.000
15	4	var3	80/10/10	xgb	xgb	1761.777	481.344	-1280.433	1.169	1.008	-0.161	4.378	4.378	0.000
15	4	var3	80/10/10	spline	svr	1797.276	611.172	-1186.104	1.792	1.554	-0.238	5.594	5.594	0.000

Table 4.8 Representative failure-mode examples (worst ΔNLL; DGAM degrades NLL relative to GAM).

Table	Dataset	Target	Split	Mean	Dispersion	NLL (GAM)	NLL (DGAM)	ΔNLL	CRPS (GAM)	CRPS (DGAM)	ΔCRPS	RMSE (GAM)	RMSE (DGAM)	ΔRMSE
9	3	var1	60/20/20	ebm	ebm	1417.619	3975.411	2557.792	0.135	0.128	-0.007	0.483	0.483	0.000
9	3	var1	70/15/15	svr	xgb	949.700	4647.266	3697.566	0.216	0.123	-0.093	0.362	0.362	0.000

CHAPTER V

DISCUSSION AND CONCLUSION

5.1 Discussion

This research was undertaken with the primary objective *to develop techniques for determining model parameters in Double Generalized Additive Models (DGAMs)* using Smoothing Splines, Explainable Boosting Machine (EBM), Extreme Gradient Boosting (XGBoost), Random Forest (RF), and Support Vector Regression (SVR). In DGAMs, model development must address both the conditional mean and the conditional dispersion. Therefore, the ability to learn nonlinear effects while accommodating heteroscedasticity is central to method selection and to the reliability of uncertainty quantification.

Overall, the empirical findings in Phase II support the motivation of DGAM for the present problem setting. In many experimental conditions, DGAM improved distribution-focused criteria such as negative log-likelihood (NLL), continuous ranked probability score (CRPS), and Akaike information criterion (AIC). At the same time, point-prediction metrics such as MAE and RMSE were often comparable to GAM. This pattern is consistent with the DGAM design: the primary benefit is improved modeling of the predictive distribution via a dispersion submodel, rather than guaranteed improvement in point estimates.

5.1.1 Smoothing Splines for DGAM

Smoothing splines provide a principled baseline for DGAMs because they estimate smooth nonlinear functions without requiring a prespecified parametric form. Within the DGAM framework, splines naturally support separate smooth components for the mean and dispersion submodels, allowing the analyst to directly interpret how predictors influence both central tendency and variability. A key advantage is the explicit control of model complexity through smoothing parameters, which makes the bias–variance trade-off transparent and supports statistically grounded selection via cross-validation or information criteria.

In addition, spline-based DGAMs retain strong inferential tooling, including approximate standard errors, confidence bands for smooth effects, and hypothesis tests for functional terms. This remains particularly valuable in scientific and health analytics applications where interpretability is as important as predictive accuracy and where communicating uncertainty requires clear model explanations.

5.1.2 Explainable Boosting Machine (EBM) for DGAM

EBM offers an attractive compromise between interpretability and predictive strength by learning an additive structure composed of data-driven shape functions (and, when needed, limited interactions). In a DGAM setting, this additive decomposition maps well to distributional regression because both the mean and dispersion components can be modeled as sums of transparent effects. This supports a practical workflow in which the analyst can examine which predictors drive the mean response versus which predictors primarily drive predictive uncertainty.

A further advantage is that EBM can capture nonlinear patterns without requiring extensive manual feature engineering. However, as with any flexible learner, it is important to validate performance using both probabilistic metrics (e.g., NLL and CRPS) and calibration diagnostics, particularly when EBM is used for the dispersion component.

5.1.3 XGBoost for DGAM

XGBoost is a powerful learner for capturing complex nonlinearities and feature interactions. When integrated into DGAM, XGBoost can be especially effective for the dispersion submodel in settings where the variance structure changes in a non-smooth or interaction-driven manner. This flexibility can translate into strong likelihood-based improvements when heteroscedasticity is pronounced and cannot be adequately captured by simpler smoothers.

Nevertheless, the main practical challenge with XGBoost lies in hyperparameter tuning and the risk of overfitting, particularly in the dispersion model. Overfitting the dispersion component can degrade out-of-sample calibration even if in-sample fit appears improved. For this reason, XGBoost-based DGAM models should be accompanied by careful validation across splits and, ideally, by regularization strategies that prioritize stable uncertainty estimates.

5.1.4 Random Forest (RF) for DGAM

Random Forest provides a robust nonparametric approach that can learn nonlinear relationships and interactions with relatively limited tuning. In DGAM, RF can serve as an effective dispersion learner because it can adapt to complex variance patterns that arise in real-world healthcare and operational data. A practical strength of RF is its stability: compared with boosting approaches, RF tends to be less sensitive to hyperparameter choices and less prone to overly sharp variance estimates.

However, RF interpretability is typically lower than spline-based or EBM-based approaches. Therefore, RF is often most suitable when the primary objective is improving distributional fit and calibration rather than maximizing interpretability of individual functional effects.

5.1.5 Support Vector Regression (SVR) for DGAM

SVR is a competitive technique when the underlying signal is nonlinear but relatively smooth, and when regularization is important for generalization. In the DGAM framework, SVR can be used as a dispersion learner to capture systematic changes in residual variability driven by covariates. Its kernel-based formulation provides flexibility while retaining explicit control over model complexity through regularization parameters.

The main limitation of SVR is computational scalability and sensitivity to kernel and parameter settings, particularly for larger datasets. Despite this limitation, SVR remains a useful option in DGAM, especially when the dataset size is moderate and the dispersion structure does not require highly complex interaction modeling.

5.1.6 Recommendations

Based on the empirical findings, DGAM is recommended over a mean-only GAM when heteroscedasticity is present and when reliable uncertainty quantification is required. In such cases, performance should be assessed using probabilistic fit metrics such as NLL and CRPS, as well as calibration diagnostics, rather than relying solely on point-prediction errors. When interpretability is a priority, smoothing splines and EBM provide transparent and communicable models. When the primary goal is flexible capture of variance patterns, RF and XGBoost are strong candidates, with SVR serving as an effective alternative in moderate-sized settings.

Implementation Considerations

- **Model selection should be distribution-aware.** When DGAM is used, selecting models solely by MAE/RMSE can miss the core advantage of dispersion learning. Prefer NLL/CRPS as primary selection criteria and always report calibration (e.g., PICP95).
- **Regularize the dispersion learner explicitly.** Dispersion models can overfit even when the mean model is well-behaved. Use early stopping and/or conservative tree depth for boosting methods, and prefer stable defaults (e.g., RF) when data are limited.
- **Check calibration stability across splits.** Validate that gains in NLL/CRPS persist across train/validation/test ratios and are not driven by a single split configuration.
- **Monitor numerical stability and constraints.** Ensure the dispersion parameterization enforces positivity (e.g., log-link for variance/scale) and verify that predicted scales are within plausible ranges to avoid unstable likelihood values.
- **Interpretability depends on the learner choice.** If interpretability is required, prefer spline- or EBM-based components and use their effect plots for communicating how predictors influence both mean and uncertainty.

Future Research Directions

- Develop automated, principled hyperparameter selection to make DGAM training consistent across learners.
- Improve uncertainty quantification for ML-based DGAMs, including calibrated predictive intervals and reliability diagnostics.
- Extend the DGAM framework to additional exponential-family responses and link functions beyond the Gaussian case.
- Compare DGAM with GAMLSS (location–scale–shape modeling) to capture skewness and heavy tails when dispersion alone is insufficient.
- Integrate deep learning components for mean and dispersion to model high-dimensional, nonlinear effects more flexibly.
- Provide robust, well-documented software packages implementing these methods in widely used statistical environments.
- Conduct large-scale simulation studies to characterize finite-sample behavior, calibration, and failure modes under controlled settings.

5.2 Conclusion

This research has successfully accomplished its primary objective of developing comprehensive techniques for parameter determination in Double Generalized Additive Models using five distinct machine learning approaches. The theoretical foundations, algorithmic specifications, and evaluation across multiple datasets provide a solid basis for both immediate practical application and future methodological development. The integration of Smoothing Splines, Support Vector Regression, Explainable Boosting Machine, Extreme Gradient Boosting, and Random Forest within the DGAM framework demonstrates that diverse machine learning techniques can be systematically adapted to distributional regression while preserving their individual strengths. This unification of classical statistical modeling with modern machine learning represents a promising direction for the continued evolution of statistical methodology.

As data complexity continues to increase across scientific disciplines, the need for flexible methods that can model both central tendency and variability will only grow. The DGAM framework and associated techniques developed here provide researchers and practitioners with powerful tools to meet this challenge, enabling more accurate predictions, deeper insights, and better-informed decisions in applications ranging from health sciences to environmental modeling and beyond. The journey from theoretical concept through algorithmic development to practical implementation demonstrates the value of rigorous mathematical foundations combined with computational innovation. This research establishes a comprehensive foundation for DGAM methodology that we hope will stimulate further developments and widespread adoption in statistical practice.

REFERENCES

REFERENCES

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *19*(6), 716–723.

Basak, D., Pal, S., and Patranabis, D. C. (2007). Support vector regression. *Neural Information Processing–Letters and Reviews*, *11*(10), 203–224.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Chang, C.-H., Tan, S., Lengerich, B., Goldenberg, A., and Caruana, R. (2021). How interpretable and trustworthy are GAMs? In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 95–105.

Chang, C.-H., Caruana, R., and Goldenberg, A. (2022). NODE-GAM: Neural generalized additive model for interpretable deep learning. In *International Conference on Learning Representations (ICLR)*.

Chen, T., and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794.

Chouldechova, A., and Hastie, T. (2015). Generalized additive model selection. *arXiv preprint arXiv:1506.03850*.

Croux, C., Gijbels, I., and Prosdocimi, I. (2012). Robust estimation of mean and dispersion functions in extended generalized additive models. *Biometrics*, *68*(1), 31–44.

de Jong, P., and Heller, G. Z. (2008). *Generalized Linear Models for Insurance Data*. Cambridge University Press, Cambridge, UK.

Drucker, H., Burges, C. J., Kaufman, L., Smola, A., and Vapnik, V. (1997). Support vector regression machines. In *Advances in Neural Information Processing Systems 9*, pp. 155–161.

Gijbels, I., and Prosdocimi, I. (2011). Smoothness-adaptive local linear estimation of the mean and dispersion functions in additive models. *Computational Statistics and Data Analysis*, *55*(1), 580–594.

Gijbels, I., and Prosdocimi, I. (2012). Flexible mean and dispersion function estimation in extended generalized additive models. *Communications in Statistics - Theory and Methods*, *41*(16–17), 3259–3277.

Gneiting, T., and Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, *102*(477), 359–378.

Harville, D. A. (1977). Maximum likelihood approaches to variance component estimation and to related problems. *Journal of the American Statistical Association*, *72*(358), 320–338.

Harvei, G. (2019). *Predicting Vessel Prices Using Machine Learning: A Comparison of XGBoost and Generalized Additive Models*. Master's thesis, Norwegian School of Economics, Bergen, Norway.

Hastie, T., and Tibshirani, R. (1986). Generalized additive models. *Statistical Science*, *1*(3), 297–310.

Hastie, T., and Tibshirani, R. (1990). *Generalized Additive Models*. Chapman and Hall/CRC, London, UK.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer, New York, NY.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning with Applications in R*. Springer, New York, NY.

Kuleshov, V., Fenner, N., and Ermon, S. (2018). Accurate uncertainties for deep learning using calibrated regression. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 2796–2804.

Lee, Y., and Nelder, J. A. (1996). Hierarchical generalized linear models (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *58*(4), 619–678.

Lee, Y., Nelder, J. A., and Pawitan, Y. (2006). *Generalized Linear Models with Random Effects: Unified Analysis via H-likelihood*. Chapman and Hall/CRC, Boca Raton, FL.

López, M. T., Martínez-García, M., Berenguer-Vidal, R., and García-Hernández, J. (2025). Hybrid GAM-SVR approach for accurate solar irradiance forecasting under extreme weather conditions. *Renewable Energy*, *220*, 119–132.

Lou, Y., Caruana, R., and Gehrke, J. (2012). Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 150–158.

Lou, Y., Caruana, R., Gehrke, J., and Hooker, G. (2013). Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 623–631.

Nori, H., Jenkins, S., Koch, P., and Caruana, R. (2019). InterpretML: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*.

Patterson, H. D., and Thompson, R. (1971). Recovery of inter-block information when block sizes are unequal. *Biometrika*, *58*(3), 545–554.

Paula, G. A. (2013). *Modelos de Regressão com Apoio Computacional*. Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, Brazil.

Rigby, R. A., and Stasinopoulos, D. M. (2005). Generalized additive models for location, scale and shape. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *54*(3), 507–554.

Shim, J., and Lee, S. (2011). Random forests for medical data analysis. *Journal of the Korean Statistical Society*, *40*(3), 357–375.

Smola, A. J., and Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, *14*(3), 199–222.

Smyth, G. K. (1989). Generalized linear models with varying dispersion. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *51*(1), 47–60.

Smyth, G. K., and Verbyla, A. P. (1999). Adjusted likelihood methods for modelling dispersion in generalized linear models. *Environmetrics*, *10*(6), 695–709.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, NY.

Wood, S. N. (2006). *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, Boca Raton, FL.

Wood, S. N. (2011). Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *73*(1), 3–36.

Wood, S. N. (2017). *Generalized Additive Models: An Introduction with R* (2nd ed.). Chapman and Hall/CRC, Boca Raton, FL.

Zanin, L., and Marra, G. (2012). Rolling regression versus time-varying coefficient modelling: An empirical investigation of the stability of regression coefficients over time. *Econometric Reviews*, *31*(6), 703–722.

APPENDICES

APPENDIX A

EXAMPLE PYTHON CODE FOR DGAM IMPLEMENTATIONS

This appendix provides representative Python code snippets corresponding to the DGAM training procedures described in Algorithms 1 (Random Forest DGAM). The intent is to document the implementation logic and the fixed hyperparameter settings used across all experiments (Table 3.3). The snippets are simplified for readability and mirror the alternating mean/dispersion fitting pattern used throughout the experimental pipeline.

A.1 Fixed Hyperparameter Configuration (Used by All Snippets)

```
RANDOM_SEED = 42

SPLINE_CONFIG = {
    "n_knots": 10,
    "degree": 3,
    "alpha": 1.0,
    "standardize": True,
}

EBM_CONFIG = {
    "estimator": "ExplainableBoostingRegressor",
    "random_state": RANDOM_SEED,
    "fallback": "SplineRidgeGAM",
}

XGB_CONFIG = {
    "n_estimators": 1200,
    "max_depth": 4,
    "learning_rate": 0.03,
    "subsample": 0.9,
    "colsample_bytree": 0.9,
    "reg_lambda": 1.0,
    "random_state": RANDOM_SEED,
}

RF_MEAN_CONFIG = {
    "n_estimators": 800,
    "min_samples_leaf": 3,
    "bootstrap": True,
    "random_state": RANDOM_SEED,
}

RF_DISP_CONFIG = {
    "n_estimators": 600,
    "min_samples_leaf": 3,
    "bootstrap": True,
    "random_state": RANDOM_SEED,
```

```
}

SVR_CONFIG = {
    "kernel": "rbf",
    "C": 10,
    "gamma": "scale",
}
```

A.2 Classical Spline DGAM via coupled PIRLS (Algorithm 1)

Note: The key idea is to alternately update a mean model and a dispersion model until both parameter updates are small. In practice, the exact basis construction may differ (e.g., B-splines), but the coupled update pattern remains the same.

```
import numpy as np
from sklearn.preprocessing import StandardScaler


def _rbf_spline_design_1d(x_vals, n_knots=10, degree=3):
    """Lightweight 1D spline-like basis via fixed radial basis knots.

    This is a compact, dependency-free stand-in for a full
    smoothing-spline/GAM basis.
    For the thesis appendix, the key point is the coupled mean/dispersion
    PIRLS structure.
    """
    x_arr = np.asarray(x_vals).reshape(-1, 1)
    knot_vals = np.quantile(x_arr[:, 0], np.linspace(0.0, 1.0, n_knots))
    knot_vals = np.unique(knot_vals)
    widths = np.std(x_arr[:, 0])
    widths = 1.0 if widths == 0 else widths
    phi = []
    for kv in knot_vals:
        phi.append(np.exp(-0.5 * ((x_arr[:, 0] - kv) / widths) ** 2))
    phi_mat = np.column_stack(phi)
    intercept = np.ones((x_arr.shape[0], 1))
    return np.hstack([intercept, phi_mat])


def fit_spline_dgam_pirls(X, y, max_iter=50, tol_mu=1e-6, tol_phi=1e-6,
    delta=1e-8):
    # Alternating updates: mean (mu) then dispersion (phi), repeated until
    convergence
    """Coupled PIRLS-like alternating updates for mean and log-variance.

    Mean: ridge regression on spline basis.
```

```
Dispersion: ridge regression on log((y - mu)^2 + delta).

Hyperparameters are taken from SPLINE_CONFIG in Table tab:hyperparam.
"""
x_vec = np.asarray(X).reshape(-1)
y_vec = np.asarray(y).reshape(-1)

if SPLINE_CONFIG["standardize"]:
    scaler = StandardScaler()
    x_std = scaler.fit_transform(x_vec.reshape(-1, 1)).reshape(-1)
else:
    scaler = None
    x_std = x_vec

X_mu = _rbf_spline_design_1d(x_std, n_knots=SPLINE_CONFIG["n_knots"],
degree=SPLINE_CONFIG["degree"])
X_phi = X_mu.copy()

alpha = float(SPLINE_CONFIG["alpha"])
p_dim = X_mu.shape[1]
I_mat = np.eye(p_dim)

beta_mu = np.zeros(p_dim)
beta_phi = np.zeros(p_dim)

for _ in range(max_iter):
    eta_mu = X_mu @ beta_mu
    mu_hat = eta_mu

    resid = y_vec - mu_hat
    z_phi = np.log(resid ** 2 + delta)

    # Mean update (ridge regression on spline basis)
    beta_mu_new = np.linalg.solve((X_mu.T @ X_mu) + alpha * I_mat,
X_mu.T @ y_vec)
    # Dispersion update on log(residual^2) target
    beta_phi_new = np.linalg.solve((X_phi.T @ X_phi) + alpha * I_mat,
X_phi.T @ z_phi)

    d_mu = np.linalg.norm(beta_mu_new - beta_mu)
    d_phi = np.linalg.norm(beta_phi_new - beta_phi)

    beta_mu = beta_mu_new
    beta_phi = beta_phi_new

    if (d_mu < tol_mu) and (d_phi < tol_phi):
        break

def predict(x_new):
    x_new_vec = np.asarray(x_new).reshape(-1)
```

```
        if scaler is not None:
            x_new_std = scaler.transform(x_new_vec.reshape(-1,
    1)).reshape(-1)
        else:
            x_new_std = x_new_vec
        X_new = _rbf_spline_design_1d(x_new_std,
    n_knots=SPLINE_CONFIG["n_knots"], degree=SPLINE_CONFIG["degree"])
        mu_pred = X_new @ beta_mu
        log_var_pred = X_new @ beta_phi
        phi_pred = np.exp(log_var_pred)
        return mu_pred, phi_pred

    return {"beta_mu": beta_mu, "beta_phi": beta_phi, "predict": predict}
```

A.3 EBM-DGAM via Alternating Boosting for Mean and Dispersion

Note: This is a minimal, readable stand-in for an EBM implementation. The appendix focuses on the alternating mean/dispersion fitting loop; in the experiments, this block corresponds to extttExplainableBoostingRegressor with the fixed random seed.

```
import numpy as np


def fit_ebm_dgam_alternating(X, y, max_rounds=200, early_stop=1e-4,
    delta=1e-8):
    """Alternating additive-model fitting (EBM-like) for mean and dispersion.

    If interpret is installed, you can replace the placeholders with
    ExplainableBoostingRegressor.
    This snippet keeps the alternating update structure explicit.
    """
    rng = np.random.default_rng(EBM_CONFIG["random_state"])

    X_arr = np.asarray(X)
    y_vec = np.asarray(y).reshape(-1)

    mu_pred = np.repeat(np.mean(y_vec), y_vec.shape[0])
    log_var_pred = np.repeat(np.log(np.var(y_vec) + delta), y_vec.shape[0])

    mean_models = []
    disp_models = []

    for _ in range(max_rounds):
        resid_mu = y_vec - mu_pred
```

```
    # Placeholder weak learner: single-feature linear fit chosen at
random
    j_idx = int(rng.integers(0, X_arr.shape[1]))
    xj = X_arr[:, j_idx]
    coef = np.cov(xj, resid_mu, bias=True)[0, 1] / (np.var(xj) + delta)
    intercept = np.mean(resid_mu) - coef * np.mean(xj)
    step_pred = intercept + coef * xj

    mu_new = mu_pred + 0.1 * step_pred

    resid = y_vec - mu_new
    z_phi = np.log(resid ** 2 + delta)

    # Dispersion weak learner: same idea
    j_idx2 = int(rng.integers(0, X_arr.shape[1]))
    xk = X_arr[:, j_idx2]
    coef2 = np.cov(xk, z_phi - log_var_pred, bias=True)[0, 1] /
(np.var(xk) + delta)
    intercept2 = np.mean(z_phi - log_var_pred) - coef2 * np.mean(xk)
    step_pred2 = intercept2 + coef2 * xk

    log_var_new = log_var_pred + 0.1 * step_pred2

    if (np.mean(np.abs(mu_new - mu_pred)) < early_stop) and
(np.mean(np.abs(log_var_new - log_var_pred)) < early_stop):
        mu_pred = mu_new
        log_var_pred = log_var_new
        break

    mu_pred = mu_new
    log_var_pred = log_var_new

    mean_models.append((j_idx, intercept, coef))
    disp_models.append((j_idx2, intercept2, coef2))

def predict(X_new):
    Xn = np.asarray(X_new)
    mu_out = np.repeat(np.mean(y_vec), Xn.shape[0])
    for j_idx, intercept, coef in mean_models:
        mu_out = mu_out + 0.1 * (intercept + coef * Xn[:, j_idx])

    log_var_out = np.repeat(np.log(np.var(y_vec) + delta), Xn.shape[0])
    for j_idx2, intercept2, coef2 in disp_models:
        log_var_out = log_var_out + 0.1 * (intercept2 + coef2 * Xn[:,
j_idx2])

    return mu_out, np.exp(log_var_out)

return {"mean_models": mean_models, "disp_models": disp_models,
"predict": predict}
```

A.4 Random Forest DGAM via Alternating Residual Forests

Note: Two forests are trained: one for the conditional mean and one for a log-variance target derived from residuals. Hyperparameters match Table 3.3.

```
import numpy as np
from sklearn.ensemble import RandomForestRegressor


def fit_rf_dgam_alternating(X, y, max_iter=25, tol=1e-4, delta=1e-8):
    # Loop: fit mean forest -> compute residual-based dispersion target ->
    fit dispersion forest
    """Alternating RF fits for mean and dispersion.

    Mean model: RandomForestRegressor with RF_MEAN_CONFIG.
    Dispersion model: RandomForestRegressor with RF_DISP_CONFIG fit on
    log((y - mu)^2 + delta).
    """
    X_arr = np.asarray(X)
    y_vec = np.asarray(y).reshape(-1)

    rf_mu = RandomForestRegressor(
        n_estimators=RF_MEAN_CONFIG["n_estimators"],
        min_samples_leaf=RF_MEAN_CONFIG["min_samples_leaf"],
        bootstrap=RF_MEAN_CONFIG["bootstrap"],
        random_state=RF_MEAN_CONFIG["random_state"],
        n_jobs=-1,
    )
    rf_phi = RandomForestRegressor(
        n_estimators=RF_DISP_CONFIG["n_estimators"],
        min_samples_leaf=RF_DISP_CONFIG["min_samples_leaf"],
        bootstrap=RF_DISP_CONFIG["bootstrap"],
        random_state=RF_DISP_CONFIG["random_state"],
        n_jobs=-1,
    )

    mu_prev = np.repeat(np.mean(y_vec), y_vec.shape[0])
    for _ in range(max_iter):
        rf_mu.fit(X_arr, y_vec)
        mu_hat = rf_mu.predict(X_arr)

        z_phi = np.log((y_vec - mu_hat) ** 2 + delta)
        rf_phi.fit(X_arr, z_phi)

        if np.mean(np.abs(mu_hat - mu_prev)) < tol:
            break
        mu_prev = mu_hat
```

```
    def predict(X_new):
        Xn = np.asarray(X_new)
        mu_out = rf_mu.predict(Xn)
        log_var_out = rf_phi.predict(Xn)
        return mu_out, np.exp(log_var_out)

    return {"rf_mu": rf_mu, "rf_phi": rf_phi, "predict": predict}
```

APPENDIX B

MODEL PERFORMANCE EVALUATION

Table B.1 Predictive performance of GAM, DGAM, and machine learning models for abnormal short-term variability (STV) across data splitting strategies using the Fetal Health Monitoring Dataset.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	3311.890	1591.945	7.876	10.427	0.663	5.782	0.927
2	60/20/20	GAM	ebm	const	3046.544	1501.272	6.165	8.416	0.780	4.570	0.920
3	60/20/20	GAM	xgb	const	2891.566	1423.783	4.747	6.953	0.850	3.712	0.953
4	60/20/20	GAM	rf	const	2856.122	1406.061	4.336	6.717	0.860	3.458	0.939
5	60/20/20	GAM	svr	const	3488.274	1722.137	12.360	14.187	0.376	8.244	0.986
6	60/20/20	DGAM	spline	spline	3395.162	1571.581	7.876	10.427	0.663	5.622	0.934
7	60/20/20	DGAM	spline	ebm	3252.034	1542.017	7.876	10.427	0.663	5.434	0.962
8	60/20/20	DGAM	spline	xgb	4642.215	2237.108	7.876	10.427	0.663	5.956	0.967
9	60/20/20	DGAM	spline	rf	3296.069	1564.034	7.876	10.427	0.663	5.449	0.962
10	60/20/20	DGAM	spline	svr	3356.961	1594.48	7.876	10.427	0.663	5.740	0.965
11	60/20/20	DGAM	ebm	spline	3144.435	1488.218	6.165	8.416	0.780	4.597	0.953
12	60/20/20	DGAM	ebm	ebm	2955.550	1435.775	6.165	8.416	0.780	4.339	0.929
13	60/20/20	DGAM	ebm	xgb	3502.427	1709.214	6.165	8.416	0.780	4.976	0.976
14	60/20/20	DGAM	ebm	rf	2918.302	1417.151	6.165	8.416	0.780	4.238	0.972
15	60/20/20	DGAM	ebm	svr	3047.202	1481.601	6.165	8.416	0.780	4.525	0.948
16	60/20/20	DGAM	xgb	spline	3233.007	1532.504	4.747	6.953	0.850	4.086	0.972
17	60/20/20	DGAM	xgb	ebm	2972.206	1444.103	4.747	6.953	0.850	3.717	0.967
18	60/20/20	DGAM	xgb	xgb	3584.554	1750.277	4.747	6.953	0.850	4.474	0.979
19	60/20/20	DGAM	xgb	rf	3132.036	1524.018	4.747	6.953	0.850	3.957	0.974
20	60/20/20	DGAM	xgb	svr	2973.294	1444.647	4.747	6.953	0.850	3.850	0.965
21	60/20/20	DGAM	rf	spline	2988.397	1410.198	4.336	6.717	0.860	3.510	0.936
22	60/20/20	DGAM	rf	ebm	2786.926	1351.463	4.336	6.717	0.860	3.303	0.941
23	60/20/20	DGAM	rf	xgb	2896.298	1406.149	4.336	6.717	0.860	3.452	0.936
24	60/20/20	DGAM	rf	rf	2849.183	1382.592	4.336	6.717	0.860	3.299	0.953
25	60/20/20	DGAM	rf	svr	2903.813	1409.906	4.336	6.717	0.860	3.511	0.950
26	60/20/20	DGAM	svr	spline	3632.206	1732.103	12.360	14.187	0.376	8.310	0.976
27	60/20/20	DGAM	svr	ebm	3544.013	1730.006	12.360	14.187	0.376	8.039	0.955
28	60/20/20	DGAM	svr	xgb	3792.210	1854.105	12.360	14.187	0.376	9.307	0.981
29	60/20/20	DGAM	svr	rf	3476.816	1696.408	12.360	14.187	0.376	7.865	0.974
30	60/20/20	DGAM	svr	svr	3557.916	1736.958	12.360	14.187	0.376	8.227	0.983
31	70/15/15	GAM	spline	const	2496.865	1184.432	7.517	9.987	0.686	5.555	0.940
32	70/15/15	GAM	ebm	const	2254.692	1105.346	5.565	7.738	0.811	4.198	0.959
33	70/15/15	GAM	xgb	const	2140.053	1048.026	4.334	6.377	0.872	3.414	0.962
34	70/15/15	GAM	rf	const	2129.612	1042.806	4.022	6.368	0.872	3.310	0.950
35	70/15/15	GAM	svr	const	2624.765	1290.382	11.962	13.996	0.383	8.079	0.978
36	70/15/15	DGAM	spline	spline	2581.625	1164.812	7.517	9.987	0.686	5.413	0.940
37	70/15/15	DGAM	spline	ebm	2418.637	1125.318	7.517	9.987	0.686	5.186	0.994
38	70/15/15	DGAM	spline	xgb	2479.548	1155.774	7.517	9.987	0.686	5.327	0.972
39	70/15/15	DGAM	spline	rf	2400.930	1116.465	7.517	9.987	0.686	5.087	0.978
40	70/15/15	DGAM	spline	svr	2504.949	1168.474	7.517	9.987	0.686	5.432	0.975
41	70/15/15	DGAM	ebm	spline	2359.287	1095.644	5.565	7.738	0.811	4.264	0.959
42	70/15/15	DGAM	ebm	ebm	2181.170	1048.585	5.565	7.738	0.811	3.952	0.975
43	70/15/15	DGAM	ebm	xgb	2381.107	1148.554	5.565	7.738	0.811	3.947	0.969
44	70/15/15	DGAM	ebm	rf	2158.714	1037.357	5.565	7.738	0.811	3.826	0.984
45	70/15/15	DGAM	ebm	svr	2226.301	1071.15	5.565	7.738	0.811	4.058	0.962
46	70/15/15	DGAM	xgb	spline	2585.728	1208.864	4.334	6.377	0.872	5.045	0.994
47	70/15/15	DGAM	xgb	ebm	2166.856	1041.428	4.334	6.377	0.872	3.388	0.962
48	70/15/15	DGAM	xgb	xgb	3747.356	1831.678	4.334	6.377	0.872	4.020	0.987
49	70/15/15	DGAM	xgb	rf	2211.537	1063.768	4.334	6.377	0.872	3.592	0.965
50	70/15/15	DGAM	xgb	svr	2152.371	1034.186	4.334	6.377	0.872	3.399	0.975
51	70/15/15	DGAM	rf	spline	2276.795	1054.398	4.022	6.368	0.872	3.459	0.962
52	70/15/15	DGAM	rf	ebm	2048.164	982.082	4.022	6.368	0.872	3.097	0.969
53	70/15/15	DGAM	rf	xgb	2190.993	1053.496	4.022	6.368	0.872	3.760	0.981
54	70/15/15	DGAM	rf	rf	2059.222	987.611	4.022	6.368	0.872	3.221	0.972
55	70/15/15	DGAM	rf	svr	2167.804	1041.902	4.022	6.368	0.872	3.376	0.969
56	70/15/15	DGAM	svr	spline	2747.936	1289.968	11.962	13.996	0.383	7.980	0.950
57	70/15/15	DGAM	svr	ebm	2625.347	1270.674	11.962	13.996	0.383	7.803	0.962
58	70/15/15	DGAM	svr	xgb	2836.950	1376.475	11.962	13.996	0.383	7.835	0.965
59	70/15/15	DGAM	svr	rf	2590.689	1253.344	11.962	13.996	0.383	7.594	0.972
60	70/15/15	DGAM	svr	svr	2689.454	1302.727	11.962	13.996	0.383	8.140	0.991
61	80/10/10	GAM	spline	const	1704.872	788.436	7.370	9.971	0.692	5.505	0.929
62	80/10/10	GAM	ebm	const	1532.917	744.458	5.920	8.080	0.798	4.400	0.892
63	80/10/10	GAM	xgb	const	1408.169	682.084	4.169	6.011	0.888	3.208	0.948
64	80/10/10	GAM	rf	const	1404.466	680.233	3.972	5.966	0.890	3.087	0.929
65	80/10/10	GAM	svr	const	1769.559	862.78	12.043	14.165	0.378	8.142	0.967
66	80/10/10	DGAM	spline	spline	1853.678	800.839	7.370	9.971	0.692	5.578	0.958
67	80/10/10	DGAM	spline	ebm	1671.322	751.661	7.370	9.971	0.692	5.155	0.943
68	80/10/10	DGAM	spline	xgb	1948.621	890.31	7.370	9.971	0.692	8.342	0.995
69	80/10/10	DGAM	spline	rf	1674.000	753.0	7.370	9.971	0.692	5.163	0.981
70	80/10/10	DGAM	spline	svr	1727.598	779.799	7.370	9.971	0.692	5.390	0.958
71	80/10/10	DGAM	ebm	spline	1629.619	730.81	5.920	8.080	0.798	4.256	0.892
72	80/10/10	DGAM	ebm	ebm	1485.841	700.92	5.920	8.080	0.798	4.127	0.929
73	80/10/10	DGAM	ebm	xgb	1446.792	681.396	5.920	8.080	0.798	4.021	0.981
74	80/10/10	DGAM	ebm	rf	1425.680	670.84	5.920	8.080	0.798	3.900	0.986
75	80/10/10	DGAM	ebm	svr	1549.047	732.524	5.920	8.080	0.798	4.300	0.892
76	80/10/10	DGAM	xgb	spline	1530.128	681.064	4.169	6.011	0.888	3.303	0.967
77	80/10/10	DGAM	xgb	ebm	1426.977	671.488	4.169	6.011	0.888	3.200	0.986
78	80/10/10	DGAM	xgb	xgb	1480.247	698.124	4.169	6.011	0.888	3.469	0.981
79	80/10/10	DGAM	xgb	rf	1409.009	662.504	4.169	6.011	0.888	3.157	0.972
80	80/10/10	DGAM	xgb	svr	1426.906	671.453	4.169	6.011	0.888	3.188	0.948
81	80/10/10	DGAM	rf	spline	1543.710	687.855	3.972	5.966	0.890	3.058	0.920
82	80/10/10	DGAM	rf	ebm	1403.575	659.788	3.972	5.966	0.890	2.989	0.929
83	80/10/10	DGAM	rf	xgb	1498.621	707.31	3.972	5.966	0.890	3.722	0.981
84	80/10/10	DGAM	rf	rf	1366.376	641.188	3.972	5.966	0.890	3.000	0.958
85	80/10/10	DGAM	rf	svr	1449.567	682.784	3.972	5.966	0.890	3.049	0.901
86	80/10/10	DGAM	svr	spline	1879.556	855.778	12.043	14.165	0.378	7.971	0.953
87	80/10/10	DGAM	svr	ebm	1783.398	849.699	12.043	14.165	0.378	7.911	0.953
88	80/10/10	DGAM	svr	xgb	1770.484	843.242	12.043	14.165	0.378	7.785	0.991
89	80/10/10	DGAM	svr	rf	1748.241	832.12	12.043	14.165	0.378	7.618	0.976
90	80/10/10	DGAM	svr	svr	1809.895	862.948	12.043	14.165	0.378	8.071	0.967

Table B.2 Predictive performance of GAM, DGAM, and machine learning models for percentage of time with abnormal long-term variability across data splitting strategies using the Fetal Health Monitoring Dataset.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	3267.919	1610.96	7.801	10.337	0.660	5.768	0.929
2	60/20/20	GAM	ebm	const	3056.443	1506.222	5.977	8.288	0.779	4.612	0.913
3	60/20/20	GAM	xgb	const	2889.949	1414.974	4.730	6.964	0.834	3.727	0.948
4	60/20/20	GAM	rf	const	2833.332	1405.666	4.333	6.728	0.855	3.335	0.938
5	60/20/20	GAM	svr	const	3437.324	1714.662	12.240	14.193	0.385	8.434	0.981
6	60/20/20	DGAM	spline	spline	3048.649	1502.324	5.931	8.236	0.789	4.582	0.948
7	60/20/20	DGAM	spline	ebm	3048.649	1502.324	5.931	8.236	0.789	4.582	0.958
8	60/20/20	DGAM	spline	xgb	3048.649	1502.324	5.931	8.236	0.789	4.582	0.957
9	60/20/20	DGAM	spline	rf	3048.649	1502.324	5.931	8.236	0.789	4.582	0.964
10	60/20/20	DGAM	spline	svr	3048.649	1502.324	5.931	8.236	0.789	4.582	0.957
11	60/20/20	DGAM	ebm	spline	3048.649	1502.324	5.931	8.265	0.789	4.591	0.941
12	60/20/20	DGAM	ebm	ebm	2941.744	1434.872	5.931	8.201	0.789	4.291	0.948
13	60/20/20	DGAM	ebm	xgb	3048.649	1502.324	5.931	8.236	0.789	4.582	0.974
14	60/20/20	DGAM	ebm	rf	2890.186	1424.093	5.931	8.236	0.789	4.232	0.975
15	60/20/20	DGAM	ebm	svr	3048.649	1488.324	5.931	8.236	0.784	4.359	0.951
16	60/20/20	DGAM	xgb	spline	3048.649	1502.324	4.789	6.901	0.850	4.041	0.964
17	60/20/20	DGAM	xgb	ebm	2976.138	1462.069	4.618	7.014	0.842	3.781	0.966
18	60/20/20	DGAM	xgb	xgb	3048.649	1502.324	4.872	6.888	0.849	4.482	0.983
19	60/20/20	DGAM	xgb	rf	3048.649	1502.324	4.774	6.821	0.868	4.033	0.978
20	60/20/20	DGAM	xgb	svr	2968.853	1423.426	4.654	6.836	0.856	3.953	0.977
21	60/20/20	DGAM	rf	spline	2977.346	1416.673	4.211	6.635	0.849	3.464	0.943
22	60/20/20	DGAM	rf	ebm	2786.208	1359.104	4.406	6.673	0.858	3.390	0.941
23	60/20/20	DGAM	rf	xgb	2883.117	1398.558	4.563	6.583	0.866	3.424	0.940
24	60/20/20	DGAM	rf	rf	2832.694	1371.347	4.207	6.780	0.863	3.435	0.953
25	60/20/20	DGAM	rf	svr	2906.165	1402.082	4.299	6.982	0.868	3.484	0.955
26	60/20/20	DGAM	svr	spline	3048.649	1502.324	5.931	8.236	0.789	4.582	0.979
27	60/20/20	DGAM	svr	ebm	3048.649	1502.324	5.931	8.236	0.789	4.582	0.958
28	60/20/20	DGAM	svr	xgb	3048.649	1502.324	5.931	8.236	0.789	4.582	0.991
29	60/20/20	DGAM	svr	rf	3048.649	1502.324	5.931	8.236	0.789	4.582	0.976
30	60/20/20	DGAM	svr	svr	3048.649	1502.324	5.931	8.236	0.789	4.582	0.984
31	70/15/15	GAM	spline	const	2467.462	1199.731	7.463	9.940	0.684	5.524	0.941
32	70/15/15	GAM	ebm	const	2237.037	1116.518	5.582	7.631	0.795	4.189	0.959
33	70/15/15	GAM	xgb	const	2085.197	1053.598	4.641	6.195	0.866	3.292	0.958
34	70/15/15	GAM	rf	const	2126.696	1042.348	3.920	6.379	0.870	3.183	0.934
35	70/15/15	GAM	svr	const	2605.615	1295.808	12.041	13.959	0.375	8.186	0.974
36	70/15/15	DGAM	spline	spline	2230.905	1114.452	5.535	7.578	0.805	4.160	0.968
37	70/15/15	DGAM	spline	ebm	2230.905	1114.452	5.535	7.578	0.805	4.160	0.996
38	70/15/15	DGAM	spline	xgb	2230.905	1114.452	5.535	7.578	0.805	4.160	0.965
39	70/15/15	DGAM	spline	rf	2230.905	1106.452	5.535	7.578	0.805	4.160	0.970
40	70/15/15	DGAM	spline	svr	2230.905	1114.452	5.535	7.578	0.805	4.160	0.975
41	70/15/15	DGAM	ebm	spline	2230.905	1107.452	5.535	7.578	0.806	4.160	0.968
42	70/15/15	DGAM	ebm	ebm	2196.034	1059.017	5.566	7.578	0.813	3.899	0.970
43	70/15/15	DGAM	ebm	xgb	2230.905	1114.452	5.535	7.578	0.820	3.854	0.959
44	70/15/15	DGAM	ebm	rf	2200.991	1048.496	5.535	7.578	0.812	3.840	0.987
45	70/15/15	DGAM	ebm	svr	2230.905	1056.452	5.535	7.578	0.816	4.027	0.968
46	70/15/15	DGAM	xgb	spline	2230.905	1114.452	4.369	6.481	0.874	4.160	0.988
47	70/15/15	DGAM	xgb	ebm	2149.060	1032.53	4.274	6.440	0.876	3.263	0.968
48	70/15/15	DGAM	xgb	xgb	2230.905	1114.452	4.300	6.237	0.873	4.105	0.987
49	70/15/15	DGAM	xgb	rf	2209.013	1055.506	4.396	6.351	0.883	3.526	0.962
50	70/15/15	DGAM	xgb	svr	2155.177	1038.589	4.510	6.342	0.885	3.362	0.974
51	70/15/15	DGAM	rf	spline	2230.905	1051.452	3.914	6.222	0.869	3.314	0.972
52	70/15/15	DGAM	rf	ebm	2011.391	990.696	4.015	6.358	0.881	3.046	0.964
53	70/15/15	DGAM	rf	xgb	2192.604	1050.302	4.017	6.563	0.862	3.805	0.969
54	70/15/15	DGAM	rf	rf	2084.931	997.466	3.957	6.396	0.874	3.165	0.982
55	70/15/15	DGAM	rf	svr	2174.043	1025.022	4.010	6.462	0.891	3.422	0.977
56	70/15/15	DGAM	svr	spline	2230.905	1114.452	5.535	7.578	0.805	4.160	0.968
57	70/15/15	DGAM	svr	ebm	2230.905	1114.452	5.535	7.578	0.805	4.160	0.968
58	70/15/15	DGAM	svr	xgb	2230.905	1114.452	5.535	7.578	0.805	4.160	0.971
59	70/15/15	DGAM	svr	rf	2230.905	1114.452	5.535	7.578	0.805	4.160	0.974
60	70/15/15	DGAM	svr	svr	2230.905	1114.452	5.535	7.578	0.805	4.160	1.000
61	80/10/10	GAM	spline	const	1711.235	791.618	7.243	9.913	0.687	5.607	0.921
62	80/10/10	GAM	ebm	const	1515.446	745.723	5.739	8.137	0.800	4.243	0.886
63	80/10/10	GAM	xgb	const	1421.923	675.962	4.087	6.066	0.882	3.145	0.949
64	80/10/10	GAM	rf	const	1399.868	681.934	3.908	6.151	0.877	3.081	0.932
65	80/10/10	GAM	svr	const	1791.671	865.836	12.000	13.972	0.371	8.176	0.967
66	80/10/10	DGAM	spline	spline	1510.833	743.416	5.692	8.078	0.810	4.208	0.958
67	80/10/10	DGAM	spline	ebm	1510.833	743.416	5.692	8.078	0.810	4.208	0.942
68	80/10/10	DGAM	spline	xgb	1510.833	743.416	5.692	8.078	0.810	4.208	0.998
69	80/10/10	DGAM	spline	rf	1510.833	742.416	5.692	8.078	0.810	4.208	0.993
70	80/10/10	DGAM	spline	svr	1510.833	743.416	5.692	8.078	0.810	4.208	0.959
71	80/10/10	DGAM	ebm	spline	1510.833	728.416	5.692	8.078	0.810	4.208	0.942
72	80/10/10	DGAM	ebm	ebm	1459.281	699.64	5.692	8.078	0.810	4.192	0.933
73	80/10/10	DGAM	ebm	xgb	1445.975	674.988	5.692	8.078	0.810	4.027	0.985
74	80/10/10	DGAM	ebm	rf	1414.897	652.448	5.692	8.078	0.809	3.910	0.981
75	80/10/10	DGAM	ebm	svr	1510.833	735.416	5.692	8.046	0.815	4.208	0.942
76	80/10/10	DGAM	xgb	spline	1510.833	693.416	4.092	5.900	0.865	3.285	0.956
77	80/10/10	DGAM	xgb	ebm	1408.989	669.494	4.146	6.252	0.890	3.219	0.986
78	80/10/10	DGAM	xgb	xgb	1486.843	697.422	4.100	6.368	0.895	3.630	0.990
79	80/10/10	DGAM	xgb	rf	1398.513	668.256	4.176	5.930	0.886	3.167	0.968
80	80/10/10	DGAM	xgb	svr	1428.230	667.115	4.141	6.186	0.885	3.072	0.960
81	80/10/10	DGAM	rf	spline	1510.833	690.416	3.913	6.072	0.880	2.970	0.942
82	80/10/10	DGAM	rf	ebm	1432.229	652.114	4.046	6.015	0.887	2.907	0.942
83	80/10/10	DGAM	rf	xgb	1470.769	703.384	4.057	5.890	0.890	3.880	0.979
84	80/10/10	DGAM	rf	rf	1357.647	641.824	3.984	5.957	0.896	3.079	0.970
85	80/10/10	DGAM	rf	svr	1448.652	679.326	3.940	5.980	0.888	3.388	0.942
86	80/10/10	DGAM	svr	spline	1510.833	743.416	5.692	8.078	0.810	4.208	0.939
87	80/10/10	DGAM	svr	ebm	1510.833	743.416	5.692	8.078	0.810	4.208	0.947
88	80/10/10	DGAM	svr	xgb	1510.833	743.416	5.692	8.078	0.810	4.208	0.995
89	80/10/10	DGAM	svr	rf	1510.833	743.416	5.692	8.078	0.810	4.208	0.983
90	80/10/10	DGAM	svr	svr	1510.833	743.416	5.692	8.078	0.810	4.208	0.973

Table B.3 Predictive performance of GAM, DGAM, and machine learning models for mean short-term variability of fetal heart rate across data splitting strategies using the Fetal Health Monitoring Dataset.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	429.387	194.694	0.314	0.439	0.709	0.238	0.962
2	60/20/20	GAM	ebm	const	884.449	392.224	0.172	0.281	0.880	0.139	0.789
3	60/20/20	GAM	xgb	const	883.552	391.776	0.171	0.281	0.881	0.139	0.792
4	60/20/20	GAM	rf	const	883.552	391.776	0.171	0.281	0.881	0.139	0.792
5	60/20/20	GAM	svr	const	359.363	155.682	0.244	0.384	0.777	0.205	0.956
6	60/20/20	DGAM	spline	spline	201.625	60.812	0.326	0.528	0.626	0.218	0.903
7	60/20/20	DGAM	spline	ebm	-68.009	110.996	0.195	0.339	0.830	0.139	0.882
8	60/20/20	DGAM	spline	xgb	739.886	316.943	0.197	0.334	0.806	0.161	0.854
9	60/20/20	DGAM	spline	rf	963.862	-82.069	0.187	0.354	0.809	0.138	0.915
10	60/20/20	DGAM	spline	svr	430.498	175.249	0.314	0.439	0.709	0.222	0.861
11	60/20/20	DGAM	ebm	spline	-15.776	-77.888	0.176	0.330	0.864	0.118	0.698
12	60/20/20	DGAM	ebm	ebm	-356.743	-248.372	0.176	0.330	0.864	0.109	0.995
13	60/20/20	DGAM	ebm	xgb	-356.750	-248.375	0.176	0.330	0.864	0.109	0.995
14	60/20/20	DGAM	ebm	rf	-356.750	-248.375	0.176	0.330	0.864	0.109	0.995
15	60/20/20	DGAM	ebm	svr	6098.399	2979.2	0.172	0.281	0.880	0.149	0.445
16	60/20/20	DGAM	xgb	spline	-16.026	-78.013	0.175	0.326	0.868	0.117	0.693
17	60/20/20	DGAM	xgb	ebm	-354.474	-247.237	0.175	0.326	0.868	0.108	0.995
18	60/20/20	DGAM	xgb	xgb	-354.539	-247.27	0.175	0.326	0.868	0.108	0.995
19	60/20/20	DGAM	xgb	rf	-354.539	-247.27	0.175	0.326	0.868	0.108	0.995
20	60/20/20	DGAM	xgb	svr	6264.281	3062.14	0.171	0.281	0.881	0.149	0.429
21	60/20/20	DGAM	rf	spline	-16.026	-78.013	0.175	0.326	0.868	0.117	0.693
22	60/20/20	DGAM	rf	ebm	-354.440	-247.22	0.175	0.326	0.868	0.108	0.995
23	60/20/20	DGAM	rf	xgb	-354.467	-247.234	0.175	0.326	0.868	0.108	0.995
24	60/20/20	DGAM	rf	rf	-354.467	-247.234	0.175	0.326	0.868	0.108	0.995
25	60/20/20	DGAM	rf	svr	6264.801	3062.4	0.171	0.281	0.881	0.149	0.429
26	60/20/20	DGAM	svr	spline	34.008	-26.996	0.244	0.384	0.777	0.167	0.801
27	60/20/20	DGAM	svr	ebm	538.121	-64.94	0.164	0.304	0.895	0.131	0.868
28	60/20/20	DGAM	svr	xgb	-388.918	164.541	0.172	0.303	0.845	0.131	0.920
29	60/20/20	DGAM	svr	rf	648.793	166.396	0.201	0.283	0.897	0.157	0.808
30	60/20/20	DGAM	svr	svr	854.059	383.03	0.244	0.384	0.777	0.186	0.773
31	70/15/15	GAM	spline	const	2496.865	1184.432	7.517	9.987	0.686	5.555	0.940
32	70/15/15	GAM	ebm	const	2254.692	1105.346	5.565	7.738	0.811	4.198	0.959
33	70/15/15	GAM	xgb	const	2140.053	1048.026	4.334	6.377	0.872	3.414	0.962
34	70/15/15	GAM	rf	const	2129.612	1042.806	4.022	6.368	0.872	3.310	0.950
35	70/15/15	GAM	svr	const	2624.765	1290.382	11.962	13.996	0.383	8.079	0.978
36	70/15/15	DGAM	spline	spline	2581.625	1164.812	7.517	9.987	0.686	5.413	0.940
37	70/15/15	DGAM	spline	ebm	2418.637	1125.318	7.517	9.987	0.686	5.186	0.994
38	70/15/15	DGAM	spline	xgb	2479.548	1155.774	7.517	9.987	0.686	5.327	0.972
39	70/15/15	DGAM	spline	rf	2400.930	1116.465	7.517	9.987	0.686	5.087	0.978
40	70/15/15	DGAM	spline	svr	2504.949	1168.474	7.517	9.987	0.686	5.432	0.975
41	70/15/15	DGAM	ebm	spline	2359.287	1095.644	5.565	7.738	0.811	4.264	0.959
42	70/15/15	DGAM	ebm	ebm	2181.170	1048.585	5.565	7.738	0.811	3.952	0.975
43	70/15/15	DGAM	ebm	xgb	2381.107	1148.554	5.565	7.738	0.811	3.947	0.969
44	70/15/15	DGAM	ebm	rf	2158.714	1037.357	5.565	7.738	0.811	3.826	0.984
45	70/15/15	DGAM	ebm	svr	2226.301	1071.15	5.565	7.738	0.811	4.058	0.962
46	70/15/15	DGAM	xgb	spline	2585.728	1208.864	4.334	6.377	0.872	5.045	0.994
47	70/15/15	DGAM	xgb	ebm	2166.856	1041.428	4.334	6.377	0.872	3.388	0.962
48	70/15/15	DGAM	xgb	xgb	3747.356	1831.678	4.334	6.377	0.872	4.020	0.987
49	70/15/15	DGAM	xgb	rf	2211.537	1063.768	4.334	6.377	0.872	3.592	0.965
50	70/15/15	DGAM	xgb	svr	2152.371	1034.186	4.334	6.377	0.872	3.399	0.975
51	70/15/15	DGAM	rf	spline	2276.795	1054.398	4.022	6.368	0.872	3.459	0.962
52	70/15/15	DGAM	rf	ebm	2048.164	982.082	4.022	6.368	0.872	3.097	0.969
53	70/15/15	DGAM	rf	xgb	2190.993	1053.496	4.022	6.368	0.872	3.760	0.981
54	70/15/15	DGAM	rf	rf	2059.222	987.611	4.022	6.368	0.872	3.221	0.972
55	70/15/15	DGAM	rf	svr	2167.804	1041.902	4.022	6.368	0.872	3.376	0.969
56	70/15/15	DGAM	svr	spline	2747.936	1289.968	11.962	13.996	0.383	7.980	0.950
57	70/15/15	DGAM	svr	ebm	2625.347	1270.674	11.962	13.996	0.383	7.803	0.962
58	70/15/15	DGAM	svr	xgb	2836.950	1376.475	11.962	13.996	0.383	7.835	0.965
59	70/15/15	DGAM	svr	rf	2590.689	1253.344	11.962	13.996	0.383	7.594	0.972
60	70/15/15	DGAM	svr	svr	2689.454	1302.727	11.962	13.996	0.383	8.140	0.991
61	80/10/10	GAM	spline	const	1704.872	788.436	7.370	9.971	0.692	5.505	0.929
62	80/10/10	GAM	ebm	const	1532.917	744.458	5.920	8.080	0.798	4.400	0.892
63	80/10/10	GAM	xgb	const	1408.169	682.084	4.169	6.011	0.888	3.208	0.948
64	80/10/10	GAM	rf	const	1404.466	680.233	3.972	5.966	0.890	3.087	0.929
65	80/10/10	GAM	svr	const	1769.559	862.78	12.043	14.165	0.378	8.142	0.967
66	80/10/10	DGAM	spline	spline	1853.678	800.839	7.370	9.971	0.692	5.578	0.958
67	80/10/10	DGAM	spline	ebm	1671.322	751.661	7.370	9.971	0.692	5.155	0.943
68	80/10/10	DGAM	spline	xgb	1948.621	890.31	7.370	9.971	0.692	8.342	0.995
69	80/10/10	DGAM	spline	rf	1674.000	753.0	7.370	9.971	0.692	5.163	0.981
70	80/10/10	DGAM	spline	svr	1727.598	779.799	7.370	9.971	0.692	5.390	0.958
71	80/10/10	DGAM	ebm	spline	1629.619	730.81	5.920	8.080	0.798	4.256	0.892
72	80/10/10	DGAM	ebm	ebm	1485.841	700.92	5.920	8.080	0.798	4.127	0.929
73	80/10/10	DGAM	ebm	xgb	1446.792	681.396	5.920	8.080	0.798	4.021	0.981
74	80/10/10	DGAM	ebm	rf	1425.680	670.84	5.920	8.080	0.798	3.900	0.986
75	80/10/10	DGAM	ebm	svr	1549.047	732.524	5.920	8.080	0.798	4.300	0.892
76	80/10/10	DGAM	xgb	spline	1530.128	681.064	4.169	6.011	0.888	3.303	0.967
77	80/10/10	DGAM	xgb	ebm	1426.977	671.488	4.169	6.011	0.888	3.200	0.986
78	80/10/10	DGAM	xgb	xgb	1480.247	698.124	4.169	6.011	0.888	3.469	0.981
79	80/10/10	DGAM	xgb	rf	1409.009	662.504	4.169	6.011	0.888	3.157	0.972
80	80/10/10	DGAM	xgb	svr	1426.906	671.453	4.169	6.011	0.888	3.188	0.948
81	80/10/10	DGAM	rf	spline	1543.710	687.855	3.972	5.966	0.890	3.058	0.920
82	80/10/10	DGAM	rf	ebm	1403.575	659.788	3.972	5.966	0.890	2.989	0.929
83	80/10/10	DGAM	rf	xgb	1498.621	707.31	3.972	5.966	0.890	3.722	0.981
84	80/10/10	DGAM	rf	rf	1366.376	641.188	3.972	5.966	0.890	3.000	0.958
85	80/10/10	DGAM	rf	svr	1449.567	682.784	3.972	5.966	0.890	3.049	0.901
86	80/10/10	DGAM	svr	spline	1879.556	855.778	12.043	14.165	0.378	7.971	0.953
87	80/10/10	DGAM	svr	ebm	1783.398	849.699	12.043	14.165	0.378	7.911	0.953
88	80/10/10	DGAM	svr	xgb	1770.484	843.242	12.043	14.165	0.378	7.785	0.991
89	80/10/10	DGAM	svr	rf	1748.241	832.12	12.043	14.165	0.378	7.618	0.976
90	80/10/10	DGAM	svr	svr	1809.895	862.948	12.043	14.165	0.378	8.071	0.967

Table B.4 Predictive performance of GAM, DGAM, and machine learning models for mean long-term variability of fetal heart rate across data splitting strategies using the Fetal Health Monitoring Dataset.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	1839.504	900.752	2.933	4.134	0.481	2.186	0.927
2	60/20/20	GAM	ebm	const	2092.379	996.19	2.214	3.274	0.660	1.803	0.689
3	60/20/20	GAM	xgb	const	2089.966	994.983	2.213	3.265	0.661	1.802	0.684
4	60/20/20	GAM	rf	const	2089.966	994.983	2.213	3.265	0.661	1.802	0.684
5	60/20/20	GAM	svr	const	1965.145	959.572	3.542	4.970	0.250	2.646	0.943
6	60/20/20	DGAM	spline	spline	1202.809	574.404	2.201	3.252	0.664	1.595	0.811
7	60/20/20	DGAM	spline	ebm	1868.301	899.15	2.436	3.754	0.582	1.628	0.987
8	60/20/20	DGAM	spline	xgb	2010.788	951.394	2.436	3.754	0.582	1.694	0.976
9	60/20/20	DGAM	spline	rf	1127.670	469.835	2.297	3.514	0.626	1.448	0.976
10	60/20/20	DGAM	spline	svr	1383.022	595.511	2.297	3.514	0.629	1.494	0.989
11	60/20/20	DGAM	ebm	spline	1405.479	632.74	2.304	3.524	0.623	1.545	0.868
12	60/20/20	DGAM	ebm	ebm	1155.440	507.72	2.304	3.524	0.623	1.422	0.997
13	60/20/20	DGAM	ebm	xgb	1155.026	507.513	2.304	3.524	0.623	1.421	0.997
14	60/20/20	DGAM	ebm	rf	1155.026	507.513	2.304	3.524	0.623	1.421	0.997
15	60/20/20	DGAM	ebm	svr	2378.709	1119.354	2.304	3.524	0.623	1.737	0.486
16	60/20/20	DGAM	xgb	spline	1413.495	636.748	2.307	3.520	0.624	1.548	0.871
17	60/20/20	DGAM	xgb	ebm	1160.043	510.021	2.307	3.520	0.624	1.427	0.991
18	60/20/20	DGAM	xgb	xgb	1159.938	509.969	2.307	3.520	0.624	1.426	0.991
19	60/20/20	DGAM	xgb	rf	1159.938	509.969	2.307	3.520	0.624	1.426	0.991
20	60/20/20	DGAM	xgb	svr	2365.612	1112.806	2.307	3.520	0.624	1.735	0.492
21	60/20/20	DGAM	rf	spline	1413.495	636.748	2.307	3.520	0.624	1.548	0.871
22	60/20/20	DGAM	rf	ebm	1159.534	509.767	2.307	3.520	0.624	1.426	0.991
23	60/20/20	DGAM	rf	xgb	1159.563	509.782	2.307	3.520	0.624	1.426	0.991
24	60/20/20	DGAM	rf	rf	1159.563	509.782	2.307	3.520	0.624	1.426	0.991
25	60/20/20	DGAM	rf	svr	2365.629	1112.814	2.307	3.520	0.624	1.735	0.492
26	60/20/20	DGAM	svr	spline	1656.915	785.458	3.542	4.970	0.250	2.371	0.968
27	60/20/20	DGAM	svr	ebm	1340.264	444.132	2.297	3.514	0.624	1.488	0.982
28	60/20/20	DGAM	svr	xgb	1347.486	572.743	2.297	3.514	0.624	1.559	0.999
29	60/20/20	DGAM	svr	rf	1438.051	610.026	2.297	3.514	0.629	1.546	0.994
30	60/20/20	DGAM	svr	svr	1117.520	515.76	3.591	5.036	0.195	2.293	0.991
31	70/15/15	GAM	spline	const	2496.865	1184.432	7.517	9.987	0.686	5.555	0.940
32	70/15/15	GAM	ebm	const	2254.692	1105.346	5.565	7.738	0.811	4.198	0.959
33	70/15/15	GAM	xgb	const	2140.053	1048.026	4.334	6.377	0.872	3.414	0.962
34	70/15/15	GAM	rf	const	2129.612	1042.806	4.022	6.368	0.872	3.310	0.950
35	70/15/15	GAM	svr	const	2624.765	1290.382	11.962	13.996	0.383	8.079	0.978
36	70/15/15	DGAM	spline	spline	2581.625	1164.812	7.517	9.987	0.686	5.413	0.940
37	70/15/15	DGAM	spline	ebm	2418.637	1125.318	7.517	9.987	0.686	5.186	0.994
38	70/15/15	DGAM	spline	xgb	2479.548	1155.774	7.517	9.987	0.686	5.327	0.972
39	70/15/15	DGAM	spline	rf	2400.930	1116.465	7.517	9.987	0.686	5.087	0.978
40	70/15/15	DGAM	spline	svr	2504.949	1168.474	7.517	9.987	0.686	5.432	0.975
41	70/15/15	DGAM	ebm	spline	2359.287	1095.644	5.565	7.738	0.811	4.264	0.959
42	70/15/15	DGAM	ebm	ebm	2181.170	1048.585	5.565	7.738	0.811	3.952	0.975
43	70/15/15	DGAM	ebm	xgb	2381.107	1148.554	5.565	7.738	0.811	3.947	0.969
44	70/15/15	DGAM	ebm	rf	2158.714	1037.357	5.565	7.738	0.811	3.826	0.984
45	70/15/15	DGAM	ebm	svr	2226.301	1071.15	5.565	7.738	0.811	4.058	0.962
46	70/15/15	DGAM	xgb	spline	2585.728	1208.864	4.334	6.377	0.872	5.045	0.994
47	70/15/15	DGAM	xgb	ebm	2166.856	1041.428	4.334	6.377	0.872	3.388	0.962
48	70/15/15	DGAM	xgb	xgb	3747.356	1831.678	4.334	6.377	0.872	4.020	0.987
49	70/15/15	DGAM	xgb	rf	2211.537	1063.768	4.334	6.377	0.872	3.592	0.965
50	70/15/15	DGAM	xgb	svr	2152.371	1034.186	4.334	6.377	0.872	3.399	0.975
51	70/15/15	DGAM	rf	spline	2276.795	1054.398	4.022	6.368	0.872	3.459	0.962
52	70/15/15	DGAM	rf	ebm	2048.164	982.082	4.022	6.368	0.872	3.097	0.969
53	70/15/15	DGAM	rf	xgb	2190.993	1053.496	4.022	6.368	0.872	3.760	0.981
54	70/15/15	DGAM	rf	rf	2059.222	987.611	4.022	6.368	0.872	3.221	0.972
55	70/15/15	DGAM	rf	svr	2167.804	1041.902	4.022	6.368	0.872	3.376	0.969
56	70/15/15	DGAM	svr	spline	2747.936	1289.968	11.962	13.996	0.383	7.980	0.950
57	70/15/15	DGAM	svr	ebm	2625.347	1270.674	11.962	13.996	0.383	7.803	0.962
58	70/15/15	DGAM	svr	xgb	2836.950	1376.475	11.962	13.996	0.383	7.835	0.965
59	70/15/15	DGAM	svr	rf	2590.689	1253.344	11.962	13.996	0.383	7.594	0.972
60	70/15/15	DGAM	svr	svr	2689.454	1302.727	11.962	13.996	0.383	8.140	0.991
61	80/10/10	GAM	spline	const	1704.872	788.436	7.370	9.971	0.692	5.505	0.929
62	80/10/10	GAM	ebm	const	1532.917	744.458	5.920	8.080	0.798	4.400	0.892
63	80/10/10	GAM	xgb	const	1408.169	682.084	4.169	6.011	0.888	3.208	0.948
64	80/10/10	GAM	rf	const	1404.466	680.233	3.972	5.966	0.890	3.087	0.929
65	80/10/10	GAM	svr	const	1769.559	862.78	12.043	14.165	0.378	8.142	0.967
66	80/10/10	DGAM	spline	spline	1853.678	800.839	7.370	9.971	0.692	5.578	0.958
67	80/10/10	DGAM	spline	ebm	1671.322	751.661	7.370	9.971	0.692	5.155	0.943
68	80/10/10	DGAM	spline	xgb	1948.621	890.31	7.370	9.971	0.692	8.342	0.995
69	80/10/10	DGAM	spline	rf	1674.000	753.0	7.370	9.971	0.692	5.163	0.981
70	80/10/10	DGAM	spline	svr	1727.598	779.799	7.370	9.971	0.692	5.390	0.958
71	80/10/10	DGAM	ebm	spline	1629.619	730.81	5.920	8.080	0.798	4.256	0.892
72	80/10/10	DGAM	ebm	ebm	1485.841	700.92	5.920	8.080	0.798	4.127	0.929
73	80/10/10	DGAM	ebm	xgb	1446.792	681.396	5.920	8.080	0.798	4.021	0.981
74	80/10/10	DGAM	ebm	rf	1425.680	670.84	5.920	8.080	0.798	3.900	0.986
75	80/10/10	DGAM	ebm	svr	1549.047	732.524	5.920	8.080	0.798	4.300	0.892
76	80/10/10	DGAM	xgb	spline	1530.128	681.064	4.169	6.011	0.888	3.303	0.967
77	80/10/10	DGAM	xgb	ebm	1426.977	671.488	4.169	6.011	0.888	3.200	0.986
78	80/10/10	DGAM	xgb	xgb	1480.247	698.124	4.169	6.011	0.888	3.469	0.981
79	80/10/10	DGAM	xgb	rf	1409.009	662.504	4.169	6.011	0.888	3.157	0.972
80	80/10/10	DGAM	xgb	svr	1426.906	671.453	4.169	6.011	0.888	3.188	0.948
81	80/10/10	DGAM	rf	spline	1543.710	687.855	3.972	5.966	0.890	3.058	0.920
82	80/10/10	DGAM	rf	ebm	1403.575	659.788	3.972	5.966	0.890	2.989	0.929
83	80/10/10	DGAM	rf	xgb	1498.621	707.31	3.972	5.966	0.890	3.722	0.981
84	80/10/10	DGAM	rf	rf	1366.376	641.188	3.972	5.966	0.890	3.000	0.958
85	80/10/10	DGAM	rf	svr	1449.567	682.784	3.972	5.966	0.890	3.049	0.901
86	80/10/10	DGAM	svr	spline	1879.556	855.778	12.043	14.165	0.378	7.971	0.953
87	80/10/10	DGAM	svr	ebm	1783.398	849.699	12.043	14.165	0.378	7.911	0.953
88	80/10/10	DGAM	svr	xgb	1770.484	843.242	12.043	14.165	0.378	7.785	0.991
89	80/10/10	DGAM	svr	rf	1748.241	832.12	12.043	14.165	0.378	7.618	0.976
90	80/10/10	DGAM	svr	svr	1809.895	862.948	12.043	14.165	0.378	8.071	0.967

Table B.5 Predictive performance of GAM, DGAM, and machine learning models for Calories burned across data splitting strategies using lifestyle data collected for health recommendation systems and mobile applications.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	8411.371	4165.685	30.485	44.085	0.993	22.769	0.961
2	60/20/20	GAM	ebm	const	8622.156	4271.078	37.174	50.362	0.990	27.330	0.946
3	60/20/20	GAM	xgb	const	8618.049	4269.024	36.877	50.221	0.991	27.162	0.936
4	60/20/20	GAM	rf	const	9447.018	4683.509	60.954	84.389	0.973	45.840	0.931
5	60/20/20	GAM	svr	const	11612.206	5766.103	253.501	325.784	0.600	179.575	0.956
6	60/20/20	DGAM	spline	spline	8254.178	4047.089	30.485	44.085	0.993	22.234	0.946
7	60/20/20	DGAM	spline	ebm	8197.797	4018.899	30.485	44.085	0.993	21.830	0.955
8	60/20/20	DGAM	spline	xgb	8318.774	4079.387	30.485	44.085	0.993	22.160	0.958
9	60/20/20	DGAM	spline	rf	8150.506	3995.253	30.485	44.085	0.993	21.385	0.966
10	60/20/20	DGAM	spline	svr	8789.904	4314.952	30.485	44.085	0.993	24.532	0.946
11	60/20/20	DGAM	ebm	spline	8613.657	4226.828	37.174	50.362	0.990	26.961	0.948
12	60/20/20	DGAM	ebm	ebm	8633.716	4236.858	37.174	50.362	0.990	26.880	0.946
13	60/20/20	DGAM	ebm	xgb	8769.097	4304.548	37.174	50.362	0.990	27.515	0.948
14	60/20/20	DGAM	ebm	rf	8685.588	4262.794	37.174	50.362	0.990	27.067	0.945
15	60/20/20	DGAM	ebm	svr	9739.753	4789.876	37.174	50.362	0.990	32.250	0.966
16	60/20/20	DGAM	xgb	spline	8725.598	4282.799	36.877	50.221	0.991	27.087	0.936
17	60/20/20	DGAM	xgb	ebm	8806.771	4323.386	36.877	50.221	0.991	27.268	0.934
18	60/20/20	DGAM	xgb	xgb	9120.320	4480.16	36.877	50.221	0.991	28.067	0.940
19	60/20/20	DGAM	xgb	rf	8971.336	4405.668	36.877	50.221	0.991	27.272	0.948
20	60/20/20	DGAM	xgb	svr	9516.291	4678.146	36.877	50.221	0.991	36.118	0.975
21	60/20/20	DGAM	rf	spline	9361.288	4600.644	60.954	84.389	0.973	44.538	0.936
22	60/20/20	DGAM	rf	ebm	9312.134	4576.067	60.954	84.389	0.973	44.181	0.959
23	60/20/20	DGAM	rf	xgb	9418.330	4629.165	60.954	84.389	0.973	45.227	0.955
24	60/20/20	DGAM	rf	rf	9452.632	4646.316	60.954	84.389	0.973	46.499	0.980
25	60/20/20	DGAM	rf	svr	9875.906	4857.953	60.954	84.389	0.973	54.102	0.964
26	60/20/20	DGAM	svr	spline	11610.293	5725.146	253.501	325.784	0.600	175.678	0.958
27	60/20/20	DGAM	svr	ebm	11561.176	5700.588	253.501	325.784	0.600	173.706	0.966
28	60/20/20	DGAM	svr	xgb	11791.312	5815.656	253.501	325.784	0.600	181.128	0.970
29	60/20/20	DGAM	svr	rf	11572.177	5706.088	253.501	325.784	0.600	178.588	0.991
30	60/20/20	DGAM	svr	svr	12398.627	6119.314	253.501	325.784	0.600	262.974	0.996
31	70/15/15	GAM	spline	const	6248.752	3084.376	27.275	40.740	0.994	20.716	0.970
32	70/15/15	GAM	ebm	const	6484.452	3202.226	36.761	49.540	0.991	26.924	0.917
33	70/15/15	GAM	xgb	const	6421.265	3170.632	33.657	47.152	0.992	25.138	0.935
34	70/15/15	GAM	rf	const	7069.366	3494.683	55.759	80.139	0.977	42.626	0.910
35	70/15/15	GAM	svr	const	8720.490	4320.245	246.234	318.227	0.635	174.396	0.950
36	70/15/15	DGAM	spline	spline	6109.670	2974.835	27.275	40.740	0.994	19.794	0.957
37	70/15/15	DGAM	spline	ebm	6117.603	2978.802	27.275	40.740	0.994	19.488	0.980
38	70/15/15	DGAM	spline	xgb	6285.886	3062.943	27.275	40.740	0.994	19.787	0.982
39	70/15/15	DGAM	spline	rf	6389.384	3114.692	27.275	40.740	0.994	19.035	0.960
40	70/15/15	DGAM	spline	svr	7206.466	3523.233	27.275	40.740	0.994	21.770	0.960
41	70/15/15	DGAM	ebm	spline	6463.348	3151.674	36.761	49.540	0.991	26.434	0.943
42	70/15/15	DGAM	ebm	ebm	6460.458	3150.229	36.761	49.540	0.991	26.362	0.947
43	70/15/15	DGAM	ebm	xgb	6520.230	3180.115	36.761	49.540	0.991	26.751	0.947
44	70/15/15	DGAM	ebm	rf	6509.093	3174.546	36.761	49.540	0.991	26.583	0.943
45	70/15/15	DGAM	ebm	svr	7037.827	3438.914	36.761	49.540	0.991	37.037	0.990
46	70/15/15	DGAM	xgb	spline	6443.266	3141.633	33.657	47.152	0.992	25.229	0.945
47	70/15/15	DGAM	xgb	ebm	6467.254	3153.627	33.657	47.152	0.992	25.163	0.940
48	70/15/15	DGAM	xgb	xgb	6568.385	3204.192	33.657	47.152	0.992	25.621	0.953
49	70/15/15	DGAM	xgb	rf	6495.199	3167.6	33.657	47.152	0.992	25.134	0.937
50	70/15/15	DGAM	xgb	svr	7120.970	3480.485	33.657	47.152	0.992	28.969	0.955
51	70/15/15	DGAM	rf	spline	6982.959	3411.48	55.759	80.139	0.977	41.151	0.928
52	70/15/15	DGAM	rf	ebm	6886.869	3363.434	55.759	80.139	0.977	40.466	0.937
53	70/15/15	DGAM	rf	xgb	6895.973	3367.986	55.759	80.139	0.977	40.949	0.937
54	70/15/15	DGAM	rf	rf	7085.357	3462.678	55.759	80.139	0.977	44.696	0.982
55	70/15/15	DGAM	rf	svr	7338.822	3589.411	55.759	80.139	0.977	48.557	0.928
56	70/15/15	DGAM	svr	spline	8725.940	4282.97	246.234	318.227	0.635	170.807	0.948
57	70/15/15	DGAM	svr	ebm	8639.194	4239.597	246.234	318.227	0.635	167.686	0.978
58	70/15/15	DGAM	svr	xgb	8768.423	4304.212	246.234	318.227	0.635	178.988	0.987
59	70/15/15	DGAM	svr	rf	11528.895	5684.448	246.234	318.227	0.635	1418.371	0.998
60	70/15/15	DGAM	svr	svr	10176.090	5008.045	246.234	318.227	0.635	496.559	0.998
61	80/10/10	GAM	spline	const	4196.348	2058.174	26.125	41.501	0.993	20.368	0.965
62	80/10/10	GAM	ebm	const	4289.470	2104.735	34.171	46.638	0.992	25.302	0.938
63	80/10/10	GAM	xgb	const	4257.622	2088.811	31.686	44.763	0.992	23.792	0.948
64	80/10/10	GAM	rf	const	4630.675	2275.338	48.654	71.291	0.981	38.098	0.940
65	80/10/10	GAM	svr	const	5745.447	2832.724	220.132	287.565	0.683	158.231	0.958
66	80/10/10	DGAM	spline	spline	4052.725	1946.362	26.125	41.501	0.993	19.048	0.943
67	80/10/10	DGAM	spline	ebm	4025.693	1932.846	26.125	41.501	0.993	18.352	0.955
68	80/10/10	DGAM	spline	xgb	4032.774	1936.387	26.125	41.501	0.993	18.392	0.955
69	80/10/10	DGAM	spline	rf	3951.076	1895.538	26.125	41.501	0.993	18.097	0.945
70	80/10/10	DGAM	spline	svr	4834.291	2337.146	26.125	41.501	0.993	21.426	0.958
71	80/10/10	DGAM	ebm	spline	4327.073	2083.537	34.171	46.638	0.992	24.879	0.945
72	80/10/10	DGAM	ebm	ebm	4317.782	2078.891	34.171	46.638	0.992	24.774	0.945
73	80/10/10	DGAM	ebm	xgb	4349.972	2094.986	34.171	46.638	0.992	25.030	0.963
74	80/10/10	DGAM	ebm	rf	4329.629	2084.814	34.171	46.638	0.992	24.804	0.948
75	80/10/10	DGAM	ebm	svr	4557.805	2198.902	34.171	46.638	0.992	27.365	0.945
76	80/10/10	DGAM	xgb	spline	4345.269	2092.634	31.686	44.763	0.992	23.683	0.940
77	80/10/10	DGAM	xgb	ebm	4348.858	2094.429	31.686	44.763	0.992	23.460	0.935
78	80/10/10	DGAM	xgb	xgb	4433.294	2136.647	31.686	44.763	0.992	23.651	0.932
79	80/10/10	DGAM	xgb	rf	4525.056	2182.528	31.686	44.763	0.992	23.574	0.930
80	80/10/10	DGAM	xgb	svr	4871.777	2355.888	31.686	44.763	0.992	26.400	0.935
81	80/10/10	DGAM	rf	spline	4510.990	2175.495	48.654	71.291	0.981	35.240	0.958
82	80/10/10	DGAM	rf	ebm	4492.321	2166.16	48.654	71.291	0.981	35.164	0.963
83	80/10/10	DGAM	rf	xgb	4513.689	2176.844	48.654	71.291	0.981	35.763	0.963
84	80/10/10	DGAM	rf	rf	4576.140	2208.07	48.654	71.291	0.981	36.062	0.955
85	80/10/10	DGAM	rf	svr	4796.696	2318.348	48.654	71.291	0.981	42.796	0.943
86	80/10/10	DGAM	svr	spline	5781.403	2810.702	220.132	287.565	0.683	157.454	0.948
87	80/10/10	DGAM	svr	ebm	5748.430	2794.215	220.132	287.565	0.683	154.104	0.970
88	80/10/10	DGAM	svr	xgb	5856.856	2848.428	220.132	287.565	0.683	170.435	0.990
89	80/10/10	DGAM	svr	rf	5749.565	2794.782	220.132	287.565	0.683	153.022	0.983
90	80/10/10	DGAM	svr	svr	6842.646	3341.323	220.132	287.565	0.683	510.945	0.998

Table B.6 Predictive performance of GAM, DGAM, and machine learning models for body fat percentage across data splitting strategies using lifestyle data collected for health recommendation systems and mobile applications.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	2594.624	1258.312	0.907	1.166	0.946	0.648	0.949
2	60/20/20	GAM	ebm	const	2446.513	1184.256	0.801	1.063	0.955	0.587	0.944
3	60/20/20	GAM	xgb	const	2094.140	1008.07	0.631	0.851	0.971	0.465	0.931
4	60/20/20	GAM	rf	const	3287.063	1604.532	1.395	1.796	0.871	1.007	0.938
5	60/20/20	GAM	svr	const	2027.739	974.87	0.560	0.818	0.973	0.434	0.931
6	60/20/20	DGAM	spline	spline	2628.127	1236.064	0.907	1.166	0.946	0.644	0.946
7	60/20/20	DGAM	spline	ebm	2629.966	1236.983	0.907	1.166	0.946	0.643	0.946
8	60/20/20	DGAM	spline	xgb	2798.862	1321.431	0.907	1.166	0.946	0.659	0.945
9	60/20/20	DGAM	spline	rf	2707.971	1275.986	0.907	1.166	0.946	0.644	0.940
10	60/20/20	DGAM	spline	svr	6085.228	2964.614	0.907	1.166	0.946	0.934	0.948
11	60/20/20	DGAM	ebm	spline	2568.083	1206.042	0.801	1.063	0.955	0.589	0.943
12	60/20/20	DGAM	ebm	ebm	2511.544	1177.772	0.801	1.063	0.955	0.584	0.939
13	60/20/20	DGAM	ebm	xgb	2656.350	1250.175	0.801	1.063	0.955	0.595	0.936
14	60/20/20	DGAM	ebm	rf	2512.788	1178.394	0.801	1.063	0.955	0.582	0.927
15	60/20/20	DGAM	ebm	svr	4899.529	2371.764	0.801	1.063	0.955	0.788	0.950
16	60/20/20	DGAM	xgb	spline	2141.650	992.825	0.631	0.851	0.971	0.462	0.920
17	60/20/20	DGAM	xgb	ebm	2133.906	988.953	0.631	0.851	0.971	0.463	0.932
18	60/20/20	DGAM	xgb	xgb	2289.049	1066.524	0.631	0.851	0.971	0.485	0.940
19	60/20/20	DGAM	xgb	rf	2148.559	996.28	0.631	0.851	0.971	0.462	0.932
20	60/20/20	DGAM	xgb	svr	3552.065	1698.032	0.631	0.851	0.971	0.973	0.980
21	60/20/20	DGAM	rf	spline	3308.463	1576.232	1.395	1.796	0.871	1.007	0.941
22	60/20/20	DGAM	rf	ebm	3241.693	1542.846	1.395	1.796	0.871	0.999	0.946
23	60/20/20	DGAM	rf	xgb	3388.753	1616.376	1.395	1.796	0.871	1.029	0.955
24	60/20/20	DGAM	rf	rf	3288.920	1566.46	1.395	1.796	0.871	0.997	0.949
25	60/20/20	DGAM	rf	svr	9087.774	4465.887	1.395	1.796	0.871	1.460	0.968
26	60/20/20	DGAM	svr	spline	1953.712	898.856	0.560	0.818	0.973	0.418	0.941
27	60/20/20	DGAM	svr	ebm	1946.840	895.42	0.560	0.818	0.973	0.419	0.943
28	60/20/20	DGAM	svr	xgb	2126.798	985.399	0.560	0.818	0.973	0.433	0.946
29	60/20/20	DGAM	svr	rf	1863.700	853.85	0.560	0.818	0.973	0.417	0.940
30	60/20/20	DGAM	svr	svr	2360.795	1102.398	0.560	0.818	0.973	0.512	0.944
31	70/15/15	GAM	spline	const	1909.681	915.84	0.859	1.102	0.952	0.613	0.928
32	70/15/15	GAM	ebm	const	1829.281	875.64	0.792	1.034	0.957	0.571	0.923
33	70/15/15	GAM	xgb	const	1498.157	710.078	0.569	0.782	0.976	0.420	0.928
34	70/15/15	GAM	rf	const	2484.687	1203.344	1.400	1.775	0.874	1.002	0.918
35	70/15/15	GAM	svr	const	1550.545	736.272	0.556	0.815	0.973	0.430	0.908
36	70/15/15	DGAM	spline	spline	1942.820	893.41	0.859	1.102	0.952	0.607	0.938
37	70/15/15	DGAM	spline	ebm	1956.380	900.19	0.859	1.102	0.952	0.605	0.937
38	70/15/15	DGAM	spline	xgb	2121.127	982.564	0.859	1.102	0.952	0.620	0.930
39	70/15/15	DGAM	spline	rf	2030.797	937.398	0.859	1.102	0.952	0.608	0.930
40	70/15/15	DGAM	spline	svr	3330.326	1587.163	0.859	1.102	0.952	1.517	0.988
41	70/15/15	DGAM	ebm	spline	1909.057	876.528	0.792	1.034	0.957	0.571	0.925
42	70/15/15	DGAM	ebm	ebm	1875.119	859.56	0.792	1.034	0.957	0.568	0.917
43	70/15/15	DGAM	ebm	xgb	1968.509	906.254	0.792	1.034	0.957	0.576	0.918
44	70/15/15	DGAM	ebm	rf	1925.216	884.608	0.792	1.034	0.957	0.568	0.915
45	70/15/15	DGAM	ebm	svr	2677.317	1260.658	0.792	1.034	0.957	0.699	0.923
46	70/15/15	DGAM	xgb	spline	1524.136	684.068	0.569	0.782	0.976	0.419	0.933
47	70/15/15	DGAM	xgb	ebm	1501.614	672.807	0.569	0.782	0.976	0.418	0.932
48	70/15/15	DGAM	xgb	xgb	1605.969	724.984	0.569	0.782	0.976	0.430	0.943
49	70/15/15	DGAM	xgb	rf	1508.680	676.34	0.569	0.782	0.976	0.419	0.943
50	70/15/15	DGAM	xgb	svr	2936.315	1390.158	0.569	0.782	0.976	0.506	0.935
51	70/15/15	DGAM	rf	spline	2530.822	1187.411	1.400	1.775	0.874	1.002	0.943
52	70/15/15	DGAM	rf	ebm	2482.858	1163.429	1.400	1.775	0.874	0.996	0.938
53	70/15/15	DGAM	rf	xgb	2559.406	1201.703	1.400	1.775	0.874	1.008	0.953
54	70/15/15	DGAM	rf	rf	2527.553	1185.776	1.400	1.775	0.874	0.995	0.962
55	70/15/15	DGAM	rf	svr	3273.768	1558.884	1.400	1.775	0.874	1.419	0.968
56	70/15/15	DGAM	svr	spline	1561.048	702.524	0.556	0.815	0.973	0.417	0.923
57	70/15/15	DGAM	svr	ebm	1608.783	726.392	0.556	0.815	0.973	0.418	0.920
58	70/15/15	DGAM	svr	xgb	1759.418	801.709	0.556	0.815	0.973	0.427	0.922
59	70/15/15	DGAM	svr	rf	1540.341	692.17	0.556	0.815	0.973	0.418	0.912
60	70/15/15	DGAM	svr	svr	1867.693	855.846	0.556	0.815	0.973	0.509	0.920
61	80/10/10	GAM	spline	const	1209.124	565.562	0.741	0.995	0.962	0.542	0.943
62	80/10/10	GAM	ebm	const	1163.264	542.632	0.711	0.937	0.966	0.520	0.943
63	80/10/10	GAM	xgb	const	916.503	419.252	0.503	0.690	0.982	0.374	0.943
64	80/10/10	GAM	rf	const	1605.861	763.93	1.262	1.633	0.898	0.914	0.940
65	80/10/10	GAM	svr	const	972.425	447.212	0.477	0.739	0.979	0.380	0.932
66	80/10/10	DGAM	spline	spline	1216.784	530.392	0.741	0.995	0.962	0.533	0.965
67	80/10/10	DGAM	spline	ebm	1220.494	532.247	0.741	0.995	0.962	0.534	0.950
68	80/10/10	DGAM	spline	xgb	1310.541	577.27	0.741	0.995	0.962	0.545	0.943
69	80/10/10	DGAM	spline	rf	1254.623	549.312	0.741	0.995	0.962	0.537	0.963
70	80/10/10	DGAM	spline	svr	2909.728	1376.864	0.741	0.995	0.962	0.701	0.950
71	80/10/10	DGAM	ebm	spline	1220.607	532.304	0.711	0.937	0.966	0.514	0.935
72	80/10/10	DGAM	ebm	ebm	1235.641	539.82	0.711	0.937	0.966	0.515	0.945
73	80/10/10	DGAM	ebm	xgb	1313.799	578.9	0.711	0.937	0.966	0.523	0.940
74	80/10/10	DGAM	ebm	rf	1259.003	551.502	0.711	0.937	0.966	0.516	0.940
75	80/10/10	DGAM	ebm	svr	1844.015	844.008	0.711	0.937	0.966	0.749	0.958
76	80/10/10	DGAM	xgb	spline	950.326	397.163	0.503	0.690	0.982	0.368	0.935
77	80/10/10	DGAM	xgb	ebm	943.334	393.667	0.503	0.690	0.982	0.368	0.950
78	80/10/10	DGAM	xgb	xgb	984.797	414.398	0.503	0.690	0.982	0.377	0.950
79	80/10/10	DGAM	xgb	rf	955.577	399.788	0.503	0.690	0.982	0.369	0.938
80	80/10/10	DGAM	xgb	svr	1500.027	672.014	0.503	0.690	0.982	0.445	0.940
81	80/10/10	DGAM	rf	spline	1653.365	748.682	1.262	1.633	0.898	0.915	0.943
82	80/10/10	DGAM	rf	ebm	1592.224	718.112	1.262	1.633	0.898	0.906	0.945
83	80/10/10	DGAM	rf	xgb	1655.104	749.552	1.262	1.633	0.898	0.932	0.932
84	80/10/10	DGAM	rf	rf	1609.519	726.76	1.262	1.633	0.898	0.907	0.958
85	80/10/10	DGAM	rf	svr	2532.294	1188.147	1.262	1.633	0.898	1.127	0.905
86	80/10/10	DGAM	svr	spline	961.329	402.664	0.477	0.739	0.979	0.361	0.938
87	80/10/10	DGAM	svr	ebm	951.421	397.71	0.477	0.739	0.979	0.364	0.932
88	80/10/10	DGAM	svr	xgb	1017.781	430.89	0.477	0.739	0.979	0.371	0.925
89	80/10/10	DGAM	svr	rf	945.498	394.749	0.477	0.739	0.979	0.363	0.935
90	80/10/10	DGAM	svr	svr	1139.264	491.632	0.477	0.739	0.979	0.424	0.945

Table B.7 Predictive performance of GAM, DGAM, and machine learning models for average heart rate maintained during the session across data splitting strategies using lifestyle data collected for health recommendation systems and mobile applications.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	1795.287	858.644	0.505	0.708	0.998	0.376	0.945
2	60/20/20	GAM	ebm	const	1718.634	820.317	0.523	0.674	0.998	0.377	0.954
3	60/20/20	GAM	xgb	const	1251.272	586.636	0.370	0.503	0.999	0.275	0.948
4	60/20/20	GAM	rf	const	1547.864	734.932	0.408	0.603	0.998	0.318	0.949
5	60/20/20	GAM	svr	const	2347.043	1134.522	0.641	0.999	0.995	0.501	0.949
6	60/20/20	DGAM	spline	spline	1603.987	723.994	0.505	0.708	0.998	0.360	0.956
7	60/20/20	DGAM	spline	ebm	1390.773	617.386	0.505	0.708	0.998	0.343	0.964
8	60/20/20	DGAM	spline	xgb	1271.266	557.633	0.505	0.708	0.998	0.332	0.966
9	60/20/20	DGAM	spline	rf	1215.683	529.842	0.505	0.708	0.998	0.329	0.974
10	60/20/20	DGAM	spline	svr	1801.754	822.877	0.505	0.708	0.998	0.334	0.966
11	60/20/20	DGAM	ebm	spline	1824.895	834.448	0.523	0.674	0.998	0.378	0.970
12	60/20/20	DGAM	ebm	ebm	1778.108	811.054	0.523	0.674	0.998	0.374	0.953
13	60/20/20	DGAM	ebm	xgb	1886.781	865.39	0.523	0.674	0.998	0.382	0.966
14	60/20/20	DGAM	ebm	rf	1752.492	798.246	0.523	0.674	0.998	0.372	0.965
15	60/20/20	DGAM	ebm	svr	2296.387	1070.194	0.523	0.674	0.998	0.482	0.974
16	60/20/20	DGAM	xgb	spline	1508.472	676.236	0.370	0.503	0.999	0.281	0.951
17	60/20/20	DGAM	xgb	ebm	1440.294	642.147	0.370	0.503	0.999	0.280	0.949
18	60/20/20	DGAM	xgb	xgb	1527.101	685.55	0.370	0.503	0.999	0.289	0.940
19	60/20/20	DGAM	xgb	rf	1351.235	597.618	0.370	0.503	0.999	0.276	0.946
20	60/20/20	DGAM	xgb	svr	2890.002	1367.001	0.370	0.503	0.999	0.471	0.971
21	60/20/20	DGAM	rf	spline	1668.678	756.339	0.408	0.603	0.998	0.322	0.951
22	60/20/20	DGAM	rf	ebm	1619.194	731.597	0.408	0.603	0.998	0.317	0.955
23	60/20/20	DGAM	rf	xgb	1667.397	755.698	0.408	0.603	0.998	0.322	0.954
24	60/20/20	DGAM	rf	rf	1529.050	686.525	0.408	0.603	0.998	0.312	0.949
25	60/20/20	DGAM	rf	svr	2345.364	1094.682	0.408	0.603	0.998	0.403	0.960
26	60/20/20	DGAM	svr	spline	2300.326	1072.163	0.641	0.999	0.995	0.495	0.955
27	60/20/20	DGAM	svr	ebm	2306.124	1075.062	0.641	0.999	0.995	0.498	0.953
28	60/20/20	DGAM	svr	xgb	2294.837	1069.418	0.641	0.999	0.995	0.499	0.956
29	60/20/20	DGAM	svr	rf	2441.492	1142.746	0.641	0.999	0.995	0.515	0.960
30	60/20/20	DGAM	svr	svr	3608.668	1726.334	0.641	0.999	0.995	0.953	0.996
31	70/15/15	GAM	spline	const	1367.021	644.51	0.510	0.705	0.998	0.375	0.937
32	70/15/15	GAM	ebm	const	1282.558	602.279	0.512	0.659	0.998	0.369	0.953
33	70/15/15	GAM	xgb	const	885.033	403.516	0.350	0.473	0.999	0.258	0.935
34	70/15/15	GAM	rf	const	1226.943	574.472	0.407	0.626	0.998	0.318	0.930
35	70/15/15	GAM	svr	const	1852.874	887.437	0.666	1.042	0.995	0.516	0.922
36	70/15/15	DGAM	spline	spline	1252.256	548.128	0.510	0.705	0.998	0.359	0.953
37	70/15/15	DGAM	spline	ebm	1131.563	487.782	0.510	0.705	0.998	0.346	0.932
38	70/15/15	DGAM	spline	xgb	1045.651	444.826	0.510	0.705	0.998	0.337	0.942
39	70/15/15	DGAM	spline	rf	1095.780	469.89	0.510	0.705	0.998	0.336	0.947
40	70/15/15	DGAM	spline	svr	1123.111	483.556	0.510	0.705	0.998	0.331	0.958
41	70/15/15	DGAM	ebm	spline	1355.571	599.786	0.512	0.659	0.998	0.366	0.957
42	70/15/15	DGAM	ebm	ebm	1310.168	577.084	0.512	0.659	0.998	0.362	0.957
43	70/15/15	DGAM	ebm	xgb	1349.672	596.836	0.512	0.659	0.998	0.364	0.963
44	70/15/15	DGAM	ebm	rf	1295.712	569.856	0.512	0.659	0.998	0.359	0.970
45	70/15/15	DGAM	ebm	svr	1799.450	821.725	0.512	0.659	0.998	0.469	0.973
46	70/15/15	DGAM	xgb	spline	1048.248	446.124	0.350	0.473	0.999	0.264	0.932
47	70/15/15	DGAM	xgb	ebm	1021.984	432.992	0.350	0.473	0.999	0.264	0.945
48	70/15/15	DGAM	xgb	xgb	1088.196	466.098	0.350	0.473	0.999	0.273	0.962
49	70/15/15	DGAM	xgb	rf	1012.003	428.002	0.350	0.473	0.999	0.260	0.935
50	70/15/15	DGAM	xgb	svr	2167.098	1005.549	0.350	0.473	0.999	0.343	0.947
51	70/15/15	DGAM	rf	spline	1283.862	563.931	0.407	0.626	0.998	0.321	0.950
52	70/15/15	DGAM	rf	ebm	1284.883	564.442	0.407	0.626	0.998	0.320	0.957
53	70/15/15	DGAM	rf	xgb	1280.350	562.175	0.407	0.626	0.998	0.324	0.957
54	70/15/15	DGAM	rf	rf	1286.028	565.014	0.407	0.626	0.998	0.311	0.950
55	70/15/15	DGAM	rf	svr	1678.211	761.106	0.407	0.626	0.998	0.478	0.977
56	70/15/15	DGAM	svr	spline	1852.465	848.232	0.666	1.042	0.995	0.509	0.938
57	70/15/15	DGAM	svr	ebm	1907.323	875.662	0.666	1.042	0.995	0.509	0.935
58	70/15/15	DGAM	svr	xgb	1976.364	910.182	0.666	1.042	0.995	0.512	0.937
59	70/15/15	DGAM	svr	rf	1975.423	909.712	0.666	1.042	0.995	0.518	0.927
60	70/15/15	DGAM	svr	svr	1875.415	859.708	0.666	1.042	0.995	0.511	0.945
61	80/10/10	GAM	spline	const	907.869	414.934	0.501	0.682	0.998	0.367	0.935
62	80/10/10	GAM	ebm	const	854.455	388.228	0.509	0.638	0.998	0.361	0.955
63	80/10/10	GAM	xgb	const	562.857	242.428	0.341	0.443	0.999	0.246	0.950
64	80/10/10	GAM	rf	const	674.388	298.194	0.334	0.502	0.999	0.262	0.953
65	80/10/10	GAM	svr	const	1120.858	521.429	0.563	0.880	0.996	0.444	0.965
66	80/10/10	DGAM	spline	spline	836.267	340.134	0.501	0.682	0.998	0.349	0.958
67	80/10/10	DGAM	spline	ebm	767.507	305.754	0.501	0.682	0.998	0.337	0.953
68	80/10/10	DGAM	spline	xgb	704.744	274.372	0.501	0.682	0.998	0.326	0.953
69	80/10/10	DGAM	spline	rf	674.652	259.326	0.501	0.682	0.998	0.324	0.963
70	80/10/10	DGAM	spline	svr	640.991	242.496	0.501	0.682	0.998	0.318	0.965
71	80/10/10	DGAM	ebm	spline	923.064	383.532	0.509	0.638	0.998	0.358	0.970
72	80/10/10	DGAM	ebm	ebm	909.647	376.824	0.509	0.638	0.998	0.353	0.963
73	80/10/10	DGAM	ebm	xgb	955.688	399.844	0.509	0.638	0.998	0.353	0.968
74	80/10/10	DGAM	ebm	rf	942.044	393.022	0.509	0.638	0.998	0.351	0.965
75	80/10/10	DGAM	ebm	svr	1087.474	465.737	0.509	0.638	0.998	0.392	0.945
76	80/10/10	DGAM	xgb	spline	721.191	282.596	0.341	0.443	0.999	0.259	0.978
77	80/10/10	DGAM	xgb	ebm	698.265	271.132	0.341	0.443	0.999	0.255	0.968
78	80/10/10	DGAM	xgb	xgb	738.816	291.408	0.341	0.443	0.999	0.259	0.955
79	80/10/10	DGAM	xgb	rf	695.205	269.602	0.341	0.443	0.999	0.252	0.955
80	80/10/10	DGAM	xgb	svr	1459.239	651.62	0.341	0.443	0.999	0.310	0.938
81	80/10/10	DGAM	rf	spline	719.829	281.914	0.334	0.502	0.999	0.264	0.945
82	80/10/10	DGAM	rf	ebm	761.942	302.971	0.334	0.502	0.999	0.264	0.970
83	80/10/10	DGAM	rf	xgb	767.309	305.654	0.334	0.502	0.999	0.271	0.968
84	80/10/10	DGAM	rf	rf	718.938	281.469	0.334	0.502	0.999	0.258	0.960
85	80/10/10	DGAM	rf	svr	981.659	412.83	0.334	0.502	0.999	0.353	0.968
86	80/10/10	DGAM	svr	spline	1130.894	487.447	0.563	0.880	0.996	0.433	0.970
87	80/10/10	DGAM	svr	ebm	1145.603	494.802	0.563	0.880	0.996	0.435	0.970
88	80/10/10	DGAM	svr	xgb	1138.073	491.037	0.563	0.880	0.996	0.436	0.968
89	80/10/10	DGAM	svr	rf	1211.229	527.614	0.563	0.880	0.996	0.445	0.965
90	80/10/10	DGAM	svr	svr	2178.210	1011.105	0.563	0.880	0.996	0.449	0.960

Table B.8 Predictive performance of GAM, DGAM, and machine learning models for Maximum heart rate recorded during a workout session across data splitting strategies using lifestyle data collected for health recommendation systems and mobile applications.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	2252.490	1087.245	0.655	0.942	0.993	0.489	0.955
2	60/20/20	GAM	ebm	const	1919.574	920.787	0.577	0.758	0.996	0.422	0.961
3	60/20/20	GAM	xgb	const	1723.045	822.522	0.483	0.669	0.997	0.364	0.955
4	60/20/20	GAM	rf	const	1749.401	835.7	0.474	0.686	0.996	0.364	0.936
5	60/20/20	GAM	svr	const	2590.679	1256.34	0.725	1.146	0.990	0.581	0.961
6	60/20/20	DGAM	spline	spline	1987.937	915.968	0.655	0.942	0.993	0.461	0.956
7	60/20/20	DGAM	spline	ebm	1855.423	849.712	0.655	0.942	0.993	0.444	0.959
8	60/20/20	DGAM	spline	xgb	1711.789	777.894	0.655	0.942	0.993	0.430	0.973
9	60/20/20	DGAM	spline	rf	1719.104	781.552	0.655	0.942	0.993	0.434	0.961
10	60/20/20	DGAM	spline	svr	1754.349	799.174	0.655	0.942	0.993	0.429	0.963
11	60/20/20	DGAM	ebm	spline	2032.447	938.224	0.577	0.758	0.996	0.427	0.959
12	60/20/20	DGAM	ebm	ebm	1968.176	906.088	0.577	0.758	0.996	0.418	0.961
13	60/20/20	DGAM	ebm	xgb	2079.059	961.53	0.577	0.758	0.996	0.429	0.966
14	60/20/20	DGAM	ebm	rf	2091.657	967.829	0.577	0.758	0.996	0.419	0.968
15	60/20/20	DGAM	ebm	svr	2775.135	1309.568	0.577	0.758	0.996	0.597	0.976
16	60/20/20	DGAM	xgb	spline	1945.012	894.506	0.483	0.669	0.997	0.376	0.955
17	60/20/20	DGAM	xgb	ebm	1960.307	902.154	0.483	0.669	0.997	0.377	0.963
18	60/20/20	DGAM	xgb	xgb	2165.249	1004.624	0.483	0.669	0.997	0.403	0.963
19	60/20/20	DGAM	xgb	rf	1952.260	898.13	0.483	0.669	0.997	0.372	0.959
20	60/20/20	DGAM	xgb	svr	3547.795	1695.898	0.483	0.669	0.997	0.836	0.986
21	60/20/20	DGAM	rf	spline	1868.973	856.486	0.474	0.686	0.996	0.369	0.945
22	60/20/20	DGAM	rf	ebm	1791.076	817.538	0.474	0.686	0.996	0.362	0.944
23	60/20/20	DGAM	rf	xgb	1829.071	836.536	0.474	0.686	0.996	0.364	0.961
24	60/20/20	DGAM	rf	rf	1754.448	799.224	0.474	0.686	0.996	0.362	0.954
25	60/20/20	DGAM	rf	svr	2762.507	1303.254	0.474	0.686	0.996	0.473	0.968
26	60/20/20	DGAM	svr	spline	2561.149	1202.574	0.725	1.146	0.990	0.576	0.966
27	60/20/20	DGAM	svr	ebm	2592.177	1218.088	0.725	1.146	0.990	0.578	0.966
28	60/20/20	DGAM	svr	xgb	2572.772	1208.386	0.725	1.146	0.990	0.581	0.969
29	60/20/20	DGAM	svr	rf	2661.747	1252.874	0.725	1.146	0.990	0.584	0.965
30	60/20/20	DGAM	svr	svr	2769.878	1306.939	0.725	1.146	0.990	0.587	0.951
31	70/15/15	GAM	spline	const	1651.897	786.948	0.625	0.890	0.994	0.463	0.933
32	70/15/15	GAM	ebm	const	1468.735	695.368	0.599	0.768	0.996	0.429	0.952
33	70/15/15	GAM	xgb	const	1324.085	623.042	0.483	0.678	0.997	0.363	0.930
34	70/15/15	GAM	rf	const	1261.950	591.975	0.449	0.648	0.997	0.340	0.933
35	70/15/15	GAM	svr	const	2039.886	980.943	0.731	1.211	0.989	0.573	0.932
36	70/15/15	DGAM	spline	spline	1412.504	628.252	0.625	0.890	0.994	0.430	0.947
37	70/15/15	DGAM	spline	ebm	1293.873	568.936	0.625	0.890	0.994	0.418	0.957
38	70/15/15	DGAM	spline	xgb	1254.016	549.008	0.625	0.890	0.994	0.408	0.960
39	70/15/15	DGAM	spline	rf	1348.249	596.124	0.625	0.890	0.994	0.412	0.965
40	70/15/15	DGAM	spline	svr	1387.815	615.908	0.625	0.890	0.994	0.413	0.955
41	70/15/15	DGAM	ebm	spline	1551.506	697.753	0.599	0.768	0.996	0.430	0.957
42	70/15/15	DGAM	ebm	ebm	1530.170	687.085	0.599	0.768	0.996	0.424	0.955
43	70/15/15	DGAM	ebm	xgb	1619.038	731.519	0.599	0.768	0.996	0.430	0.970
44	70/15/15	DGAM	ebm	rf	1551.386	697.693	0.599	0.768	0.996	0.423	0.953
45	70/15/15	DGAM	ebm	svr	2246.134	1045.067	0.599	0.768	0.996	0.526	0.958
46	70/15/15	DGAM	xgb	spline	1463.389	653.694	0.483	0.678	0.997	0.368	0.927
47	70/15/15	DGAM	xgb	ebm	1487.930	665.965	0.483	0.678	0.997	0.367	0.945
48	70/15/15	DGAM	xgb	xgb	1660.268	752.134	0.483	0.678	0.997	0.377	0.945
49	70/15/15	DGAM	xgb	rf	1466.709	655.354	0.483	0.678	0.997	0.365	0.950
50	70/15/15	DGAM	xgb	svr	2540.083	1192.042	0.483	0.678	0.997	0.517	0.955
51	70/15/15	DGAM	rf	spline	1416.671	630.336	0.449	0.648	0.997	0.343	0.945
52	70/15/15	DGAM	rf	ebm	1354.150	599.075	0.449	0.648	0.997	0.342	0.950
53	70/15/15	DGAM	rf	xgb	1368.576	606.288	0.449	0.648	0.997	0.346	0.950
54	70/15/15	DGAM	rf	rf	1278.902	561.451	0.449	0.648	0.997	0.335	0.950
55	70/15/15	DGAM	rf	svr	1800.232	822.116	0.449	0.648	0.997	0.500	0.980
56	70/15/15	DGAM	svr	spline	2068.350	956.175	0.731	1.211	0.989	0.569	0.943
57	70/15/15	DGAM	svr	ebm	2087.973	965.986	0.731	1.211	0.989	0.566	0.945
58	70/15/15	DGAM	svr	xgb	2122.064	983.032	0.731	1.211	0.989	0.569	0.948
59	70/15/15	DGAM	svr	rf	2051.011	947.506	0.731	1.211	0.989	0.572	0.932
60	70/15/15	DGAM	svr	svr	2182.897	1013.448	0.731	1.211	0.989	0.588	0.935
61	80/10/10	GAM	spline	const	1038.693	480.346	0.569	0.804	0.995	0.421	0.958
62	80/10/10	GAM	ebm	const	952.769	437.384	0.560	0.721	0.996	0.403	0.945
63	80/10/10	GAM	xgb	const	809.237	365.618	0.434	0.593	0.997	0.325	0.955
64	80/10/10	GAM	rf	const	777.650	349.825	0.385	0.573	0.997	0.301	0.953
65	80/10/10	GAM	svr	const	1167.219	544.61	0.626	0.932	0.993	0.484	0.963
66	80/10/10	DGAM	spline	spline	947.033	395.516	0.569	0.804	0.995	0.397	0.963
67	80/10/10	DGAM	spline	ebm	844.682	344.341	0.569	0.804	0.995	0.375	0.980
68	80/10/10	DGAM	spline	xgb	783.275	313.638	0.569	0.804	0.995	0.366	0.970
69	80/10/10	DGAM	spline	rf	829.984	336.992	0.569	0.804	0.995	0.375	0.970
70	80/10/10	DGAM	spline	svr	827.008	335.504	0.569	0.804	0.995	0.371	0.983
71	80/10/10	DGAM	ebm	spline	1071.922	457.961	0.560	0.721	0.996	0.405	0.953
72	80/10/10	DGAM	ebm	ebm	1048.969	446.484	0.560	0.721	0.996	0.399	0.953
73	80/10/10	DGAM	ebm	xgb	1107.839	475.92	0.560	0.721	0.996	0.406	0.955
74	80/10/10	DGAM	ebm	rf	1077.795	460.898	0.560	0.721	0.996	0.399	0.955
75	80/10/10	DGAM	ebm	svr	1253.316	548.658	0.560	0.721	0.996	0.469	0.978
76	80/10/10	DGAM	xgb	spline	966.833	405.416	0.434	0.593	0.997	0.334	0.975
77	80/10/10	DGAM	xgb	ebm	863.420	353.71	0.434	0.593	0.997	0.322	0.978
78	80/10/10	DGAM	xgb	xgb	865.522	354.761	0.434	0.593	0.997	0.323	0.970
79	80/10/10	DGAM	xgb	rf	893.191	368.596	0.434	0.593	0.997	0.323	0.973
80	80/10/10	DGAM	xgb	svr	1463.689	653.844	0.434	0.593	0.997	0.408	0.955
81	80/10/10	DGAM	rf	spline	943.220	393.61	0.385	0.573	0.997	0.308	0.955
82	80/10/10	DGAM	rf	ebm	831.118	337.559	0.385	0.573	0.997	0.298	0.963
83	80/10/10	DGAM	rf	xgb	855.113	349.556	0.385	0.573	0.997	0.304	0.950
84	80/10/10	DGAM	rf	rf	812.810	328.405	0.385	0.573	0.997	0.295	0.955
85	80/10/10	DGAM	rf	svr	1401.854	622.927	0.385	0.573	0.997	0.585	0.978
86	80/10/10	DGAM	svr	spline	1187.464	515.732	0.626	0.932	0.993	0.476	0.965
87	80/10/10	DGAM	svr	ebm	1202.462	523.231	0.626	0.932	0.993	0.479	0.958
88	80/10/10	DGAM	svr	xgb	1211.796	527.898	0.626	0.932	0.993	0.483	0.965
89	80/10/10	DGAM	svr	rf	1218.539	531.27	0.626	0.932	0.993	0.480	0.948
90	80/10/10	DGAM	svr	svr	1325.794	584.897	0.626	0.932	0.993	0.511	0.963

Table B.9 Predictive performance of GAM, DGAM, and machine learning models for serum creatinine levels across data splitting strategies using the National Health Checkup dataset of South Korea (2002–2020).

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	1874.861	912.43	0.143	0.227	0.258	0.211	0.998
2	60/20/20	GAM	ebm	const	2885.238	1417.619	0.165	0.483	-2.351	0.135	0.981
3	60/20/20	GAM	xgb	const	1492.962	721.481	0.155	0.267	-0.024	0.188	0.995
4	60/20/20	GAM	rf	const	1779.620	864.81	0.138	0.228	0.256	0.204	0.998
5	60/20/20	GAM	svr	const	1814.820	882.41	0.157	0.238	0.190	0.209	0.999
6	60/20/20	DGAM	spline	spline	68.800	-15.6	0.143	0.227	0.258	0.105	0.806
7	60/20/20	DGAM	spline	ebm	902.599	401.3	0.143	0.227	0.258	0.106	0.791
8	60/20/20	DGAM	spline	xgb	188.330	44.165	0.143	0.227	0.258	0.106	0.782
9	60/20/20	DGAM	spline	rf	189.887	44.944	0.143	0.227	0.258	0.106	0.794
10	60/20/20	DGAM	spline	svr	1721.429	810.714	0.143	0.227	0.258	0.107	0.764
11	60/20/20	DGAM	ebm	spline	14559.680	7229.84	0.165	0.483	-2.351	0.129	0.839
12	60/20/20	DGAM	ebm	ebm	8050.822	3975.411	0.165	0.483	-2.351	0.128	0.842
13	60/20/20	DGAM	ebm	xgb	8525.684	4212.842	0.165	0.483	-2.351	0.128	0.836
14	60/20/20	DGAM	ebm	rf	14841.409	7370.704	0.165	0.483	-2.351	0.130	0.812
15	60/20/20	DGAM	ebm	svr	21804.440	10852.22	0.165	0.483	-2.351	0.132	0.776
16	60/20/20	DGAM	xgb	spline	1690.694	795.347	0.155	0.267	-0.024	0.118	0.813
17	60/20/20	DGAM	xgb	ebm	152.009	26.004	0.155	0.267	-0.024	0.115	0.822
18	60/20/20	DGAM	xgb	xgb	120.967	10.484	0.155	0.267	-0.024	0.115	0.825
19	60/20/20	DGAM	xgb	rf	693.887	296.944	0.155	0.267	-0.024	0.117	0.819
20	60/20/20	DGAM	xgb	svr	3298.115	1599.058	0.155	0.267	-0.024	0.119	0.768
21	60/20/20	DGAM	rf	spline	-38.519	-69.26	0.138	0.228	0.256	0.102	0.815
22	60/20/20	DGAM	rf	ebm	1072.767	486.384	0.138	0.228	0.256	0.101	0.822
23	60/20/20	DGAM	rf	xgb	-333.072	-216.536	0.138	0.228	0.256	0.100	0.827
24	60/20/20	DGAM	rf	rf	232.167	66.084	0.138	0.228	0.256	0.102	0.801
25	60/20/20	DGAM	rf	svr	1142.828	521.414	0.138	0.228	0.256	0.103	0.788
26	60/20/20	DGAM	svr	spline	2498.411	1199.206	0.157	0.238	0.190	0.118	0.776
27	60/20/20	DGAM	svr	ebm	3157.988	1528.994	0.157	0.238	0.190	0.119	0.768
28	60/20/20	DGAM	svr	xgb	2686.043	1293.022	0.157	0.238	0.190	0.119	0.768
29	60/20/20	DGAM	svr	rf	3315.885	1607.942	0.157	0.238	0.190	0.119	0.758
30	60/20/20	DGAM	svr	svr	4437.309	2168.654	0.157	0.238	0.190	0.120	0.744
31	70/15/15	GAM	spline	const	1976.524	963.262	0.147	0.355	0.133	0.216	0.998
32	70/15/15	GAM	ebm	const	9450.735	4700.368	0.182	0.805	-3.457	0.152	0.981
33	70/15/15	GAM	xgb	const	1715.678	832.839	0.160	0.403	-0.116	0.196	0.995
34	70/15/15	GAM	rf	const	1914.556	932.278	0.149	0.361	0.106	0.212	0.997
35	70/15/15	GAM	svr	const	1949.401	949.7	0.160	0.362	0.100	0.216	0.998
36	70/15/15	DGAM	spline	spline	1304.781	602.39	0.147	0.355	0.133	0.111	0.846
37	70/15/15	DGAM	spline	ebm	468.743	184.372	0.147	0.355	0.133	0.111	0.832
38	70/15/15	DGAM	spline	xgb	835.824	367.912	0.147	0.355	0.133	0.111	0.835
39	70/15/15	DGAM	spline	rf	2184.257	1042.128	0.147	0.355	0.133	0.111	0.837
40	70/15/15	DGAM	spline	svr	5641.743	2770.872	0.147	0.355	0.133	0.113	0.805
41	70/15/15	DGAM	ebm	spline	22454.403	11177.202	0.182	0.805	-3.457	0.146	0.845
42	70/15/15	DGAM	ebm	ebm	3953.597	1926.798	0.182	0.805	-3.457	0.145	0.836
43	70/15/15	DGAM	ebm	xgb	4508.632	2204.316	0.182	0.805	-3.457	0.146	0.831
44	70/15/15	DGAM	ebm	rf	8600.314	4250.157	0.182	0.805	-3.457	0.147	0.837
45	70/15/15	DGAM	ebm	svr	30928.858	15414.429	0.182	0.805	-3.457	0.149	0.778
46	70/15/15	DGAM	xgb	spline	2647.643	1273.822	0.160	0.403	-0.116	0.124	0.822
47	70/15/15	DGAM	xgb	ebm	487.386	193.693	0.160	0.403	-0.116	0.120	0.843
48	70/15/15	DGAM	xgb	xgb	739.683	319.842	0.160	0.403	-0.116	0.122	0.830
49	70/15/15	DGAM	xgb	rf	2501.843	1200.922	0.160	0.403	-0.116	0.123	0.828
50	70/15/15	DGAM	xgb	svr	5298.785	2599.392	0.160	0.403	-0.116	0.126	0.792
51	70/15/15	DGAM	rf	spline	815.753	357.876	0.149	0.361	0.106	0.114	0.818
52	70/15/15	DGAM	rf	ebm	2220.543	1060.272	0.149	0.361	0.106	0.111	0.832
53	70/15/15	DGAM	rf	xgb	1903.093	901.546	0.149	0.361	0.106	0.111	0.830
54	70/15/15	DGAM	rf	rf	2313.467	1106.734	0.149	0.361	0.106	0.112	0.832
55	70/15/15	DGAM	rf	svr	5958.630	2929.315	0.149	0.361	0.106	0.116	0.762
56	70/15/15	DGAM	svr	spline	10230.713	5065.356	0.160	0.362	0.100	0.122	0.797
57	70/15/15	DGAM	svr	ebm	15586.403	7743.202	0.160	0.362	0.100	0.123	0.785
58	70/15/15	DGAM	svr	xgb	9394.531	4647.266	0.160	0.362	0.100	0.123	0.788
59	70/15/15	DGAM	svr	rf	10547.136	5223.568	0.160	0.362	0.100	0.123	0.794
60	70/15/15	DGAM	svr	svr	10971.630	5435.815	0.160	0.362	0.100	0.124	0.784
61	80/10/10	GAM	spline	const	-553.581	-301.79	0.131	0.178	0.325	0.098	0.986
62	80/10/10	GAM	ebm	const	-674.088	-362.044	0.128	0.176	0.343	0.093	0.939
63	80/10/10	GAM	xgb	const	-717.342	-383.671	0.129	0.173	0.363	0.094	0.972
64	80/10/10	GAM	rf	const	-684.130	-367.065	0.128	0.174	0.356	0.094	0.981
65	80/10/10	GAM	svr	const	393.004	171.502	0.148	0.201	0.147	0.109	0.824
66	80/10/10	DGAM	spline	spline	-260.778	-180.389	0.131	0.178	0.325	0.096	0.837
67	80/10/10	DGAM	spline	ebm	-237.290	-168.645	0.131	0.178	0.325	0.096	0.829
68	80/10/10	DGAM	spline	xgb	-171.029	-135.514	0.131	0.178	0.325	0.096	0.822
69	80/10/10	DGAM	spline	rf	-255.576	-177.788	0.131	0.178	0.325	0.096	0.836
70	80/10/10	DGAM	spline	svr	940.475	420.238	0.131	0.178	0.325	0.098	0.792
71	80/10/10	DGAM	ebm	spline	33.332	-33.334	0.128	0.176	0.343	0.094	0.828
72	80/10/10	DGAM	ebm	ebm	116.917	8.458	0.128	0.176	0.343	0.094	0.823
73	80/10/10	DGAM	ebm	xgb	-12.745	-56.372	0.128	0.176	0.343	0.094	0.828
74	80/10/10	DGAM	ebm	rf	200.933	50.466	0.128	0.176	0.343	0.095	0.826
75	80/10/10	DGAM	ebm	svr	1477.513	688.756	0.128	0.176	0.343	0.096	0.797
76	80/10/10	DGAM	xgb	spline	-163.016	-131.508	0.129	0.173	0.363	0.094	0.822
77	80/10/10	DGAM	xgb	ebm	-181.842	-140.921	0.129	0.173	0.363	0.094	0.825
78	80/10/10	DGAM	xgb	xgb	-183.613	-141.806	0.129	0.173	0.363	0.094	0.828
79	80/10/10	DGAM	xgb	rf	-31.204	-65.602	0.129	0.173	0.363	0.095	0.813
80	80/10/10	DGAM	xgb	svr	817.571	358.786	0.129	0.173	0.363	0.096	0.798
81	80/10/10	DGAM	rf	spline	-122.025	-111.012	0.128	0.174	0.356	0.094	0.826
82	80/10/10	DGAM	rf	ebm	-97.944	-98.972	0.128	0.174	0.356	0.094	0.822
83	80/10/10	DGAM	rf	xgb	-113.522	-106.761	0.128	0.174	0.356	0.094	0.818
84	80/10/10	DGAM	rf	rf	-172.512	-136.256	0.128	0.174	0.356	0.094	0.825
85	80/10/10	DGAM	rf	svr	889.791	394.896	0.128	0.174	0.356	0.096	0.792
86	80/10/10	DGAM	svr	spline	499.284	199.642	0.148	0.201	0.147	0.109	0.819
87	80/10/10	DGAM	svr	ebm	924.083	412.042	0.148	0.201	0.147	0.110	0.822
88	80/10/10	DGAM	svr	xgb	648.066	274.033	0.148	0.201	0.147	0.110	0.822
89	80/10/10	DGAM	svr	rf	677.490	288.745	0.148	0.201	0.147	0.110	0.824
90	80/10/10	DGAM	svr	svr	1072.583	486.292	0.148	0.201	0.147	0.110	0.806

Table B.10 Predictive performance of GAM, DGAM, and machine learning models for serum gamma-glutamyl transferase (GGT) levels across data splitting strategies using the National Health Checkup dataset of South Korea (2002–2020).

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	15791.648	7893.824	17.889	33.605	0.173	15.120	0.961
2	60/20/20	GAM	ebm	const	15539.362	7767.681	16.233	31.058	0.294	14.045	0.958
3	60/20/20	GAM	xgb	const	15551.493	7773.746	16.088	31.176	0.289	14.068	0.954
4	60/20/20	GAM	rf	const	15513.037	7754.518	15.949	30.804	0.305	13.634	0.962
5	60/20/20	GAM	svr	const	15678.884	7837.442	14.254	32.426	0.230	13.596	0.962
6	60/20/20	DGAM	spline	spline	15408.757	7702.378	17.889	33.605	0.173	13.321	0.931
7	60/20/20	DGAM	spline	ebm	15006.586	7501.293	17.889	33.605	0.173	12.899	0.926
8	60/20/20	DGAM	spline	xgb	15050.150	7523.075	17.889	33.605	0.173	12.910	0.925
9	60/20/20	DGAM	spline	rf	14970.323	7483.162	17.889	33.605	0.173	13.101	0.921
10	60/20/20	DGAM	spline	svr	14648.233	7322.116	17.889	33.605	0.173	19.856	0.946
11	60/20/20	DGAM	ebm	spline	14759.317	7377.658	16.233	31.058	0.294	12.397	0.918
12	60/20/20	DGAM	ebm	ebm	14568.316	7282.158	16.233	31.058	0.294	11.698	0.926
13	60/20/20	DGAM	ebm	xgb	14658.191	7327.096	16.233	31.058	0.294	11.717	0.928
14	60/20/20	DGAM	ebm	rf	14987.311	7491.656	16.233	31.058	0.294	11.888	0.911
15	60/20/20	DGAM	ebm	svr	14120.813	7058.406	16.233	31.058	0.294	16.135	0.943
16	60/20/20	DGAM	xgb	spline	14935.573	7465.786	16.088	31.176	0.289	12.273	0.911
17	60/20/20	DGAM	xgb	ebm	14765.727	7380.864	16.088	31.176	0.289	11.633	0.918
18	60/20/20	DGAM	xgb	xgb	14708.266	7352.133	16.088	31.176	0.289	11.671	0.922
19	60/20/20	DGAM	xgb	rf	15059.675	7527.838	16.088	31.176	0.289	11.886	0.906
20	60/20/20	DGAM	xgb	svr	14190.469	7093.234	16.088	31.176	0.289	11.126	0.936
21	60/20/20	DGAM	rf	spline	14526.659	7261.33	15.949	30.804	0.305	12.024	0.938
22	60/20/20	DGAM	rf	ebm	14453.549	7224.774	15.949	30.804	0.305	11.570	0.938
23	60/20/20	DGAM	rf	xgb	14430.387	7213.194	15.949	30.804	0.305	11.582	0.938
24	60/20/20	DGAM	rf	rf	14834.149	7415.074	15.949	30.804	0.305	11.655	0.932
25	60/20/20	DGAM	rf	svr	14160.648	7078.324	15.949	30.804	0.305	18.237	0.954
26	60/20/20	DGAM	svr	spline	16659.291	8327.646	14.254	32.426	0.230	11.342	0.874
27	60/20/20	DGAM	svr	ebm	16454.648	8225.324	14.254	32.426	0.230	11.097	0.878
28	60/20/20	DGAM	svr	xgb	16472.873	8234.436	14.254	32.426	0.230	11.097	0.878
29	60/20/20	DGAM	svr	rf	16940.479	8468.24	14.254	32.426	0.230	11.223	0.874
30	60/20/20	DGAM	svr	svr	15705.759	7850.88	14.254	32.426	0.230	16.799	0.892
31	70/15/15	GAM	spline	const	12977.888	6486.944	19.825	53.643	0.217	19.772	0.978
32	70/15/15	GAM	ebm	const	12972.272	6484.136	18.262	53.518	0.221	19.205	0.982
33	70/15/15	GAM	xgb	const	12980.903	6488.452	18.381	53.710	0.215	19.347	0.978
34	70/15/15	GAM	rf	const	13032.182	6514.091	18.059	54.869	0.181	19.235	0.983
35	70/15/15	GAM	svr	const	13120.285	6558.142	16.869	56.908	0.119	19.837	0.979
36	70/15/15	DGAM	spline	spline	11886.587	5941.294	19.825	53.643	0.217	14.755	0.914
37	70/15/15	DGAM	spline	ebm	11821.606	5908.803	19.825	53.643	0.217	14.766	0.909
38	70/15/15	DGAM	spline	xgb	11867.758	5931.879	19.825	53.643	0.217	14.769	0.910
39	70/15/15	DGAM	spline	rf	12241.753	6118.876	19.825	53.643	0.217	14.992	0.905
40	70/15/15	DGAM	spline	svr	11245.788	5620.894	19.825	53.643	0.217	17.650	0.935
41	70/15/15	DGAM	ebm	spline	11593.384	5794.692	18.262	53.518	0.221	13.853	0.914
42	70/15/15	DGAM	ebm	ebm	11812.764	5904.382	18.262	53.518	0.221	13.608	0.919
43	70/15/15	DGAM	ebm	xgb	11591.768	5793.884	18.262	53.518	0.221	13.575	0.916
44	70/15/15	DGAM	ebm	rf	12637.027	6316.514	18.262	53.518	0.221	13.845	0.904
45	70/15/15	DGAM	ebm	svr	11136.051	5566.026	18.262	53.518	0.221	16.845	0.937
46	70/15/15	DGAM	xgb	spline	11807.104	5901.552	18.381	53.710	0.215	13.826	0.905
47	70/15/15	DGAM	xgb	ebm	11997.402	5996.701	18.381	53.710	0.215	13.724	0.910
48	70/15/15	DGAM	xgb	xgb	12018.811	6007.406	18.381	53.710	0.215	13.702	0.911
49	70/15/15	DGAM	xgb	rf	12791.330	6393.665	18.381	53.710	0.215	13.988	0.897
50	70/15/15	DGAM	xgb	svr	11186.916	5591.458	18.381	53.710	0.215	17.199	0.931
51	70/15/15	DGAM	rf	spline	12011.302	6003.651	18.059	54.869	0.181	13.858	0.911
52	70/15/15	DGAM	rf	ebm	12231.459	6113.73	18.059	54.869	0.181	13.681	0.920
53	70/15/15	DGAM	rf	xgb	12271.681	6133.84	18.059	54.869	0.181	13.673	0.921
54	70/15/15	DGAM	rf	rf	13386.057	6691.028	18.059	54.869	0.181	13.750	0.921
55	70/15/15	DGAM	rf	svr	11345.213	5670.606	18.059	54.869	0.181	16.010	0.938
56	70/15/15	DGAM	svr	spline	13788.059	6892.03	16.869	56.908	0.119	13.695	0.864
57	70/15/15	DGAM	svr	ebm	14401.426	7198.713	16.869	56.908	0.119	13.567	0.871
58	70/15/15	DGAM	svr	xgb	14056.604	7026.302	16.869	56.908	0.119	13.513	0.875
59	70/15/15	DGAM	svr	rf	15594.452	7795.226	16.869	56.908	0.119	13.677	0.866
60	70/15/15	DGAM	svr	svr	12857.973	6426.986	16.869	56.908	0.119	14.051	0.888
61	80/10/10	GAM	spline	const	8025.171	4010.586	18.997	36.390	0.263	16.444	0.959
62	80/10/10	GAM	ebm	const	7970.071	3983.036	17.601	35.158	0.312	15.722	0.959
63	80/10/10	GAM	xgb	const	7948.141	3972.07	17.373	34.680	0.331	15.582	0.959
64	80/10/10	GAM	rf	const	7951.111	3973.556	17.229	34.744	0.328	15.281	0.962
65	80/10/10	GAM	svr	const	8115.222	4055.611	16.623	38.449	0.178	16.204	0.961
66	80/10/10	DGAM	spline	spline	8099.435	4047.718	18.997	36.390	0.263	14.281	0.900
67	80/10/10	DGAM	spline	ebm	7818.036	3907.018	18.997	36.390	0.263	13.977	0.905
68	80/10/10	DGAM	spline	xgb	7823.413	3909.706	18.997	36.390	0.263	13.975	0.904
69	80/10/10	DGAM	spline	rf	7917.811	3956.906	18.997	36.390	0.263	14.129	0.905
70	80/10/10	DGAM	spline	svr	7546.286	3771.143	18.997	36.390	0.263	17.107	0.938
71	80/10/10	DGAM	ebm	spline	7936.420	3966.21	17.601	35.158	0.312	13.358	0.904
72	80/10/10	DGAM	ebm	ebm	7986.340	3991.17	17.601	35.158	0.312	13.024	0.909
73	80/10/10	DGAM	ebm	xgb	8224.462	4110.231	17.601	35.158	0.312	13.077	0.904
74	80/10/10	DGAM	ebm	rf	8160.027	4078.014	17.601	35.158	0.312	13.164	0.892
75	80/10/10	DGAM	ebm	svr	7456.919	3726.46	17.601	35.158	0.312	16.760	0.924
76	80/10/10	DGAM	xgb	spline	8049.102	4022.551	17.373	34.680	0.331	13.161	0.895
77	80/10/10	DGAM	xgb	ebm	8147.542	4071.771	17.373	34.680	0.331	12.857	0.904
78	80/10/10	DGAM	xgb	xgb	8149.663	4072.832	17.373	34.680	0.331	12.838	0.904
79	80/10/10	DGAM	xgb	rf	8190.674	4093.337	17.373	34.680	0.331	13.018	0.891
80	80/10/10	DGAM	xgb	svr	7451.439	3723.72	17.373	34.680	0.331	16.431	0.919
81	80/10/10	DGAM	rf	spline	8046.098	4021.049	17.229	34.744	0.328	13.115	0.908
82	80/10/10	DGAM	rf	ebm	8063.922	4029.961	17.229	34.744	0.328	12.837	0.910
83	80/10/10	DGAM	rf	xgb	8006.777	4001.388	17.229	34.744	0.328	12.843	0.908
84	80/10/10	DGAM	rf	rf	8047.189	4021.594	17.229	34.744	0.328	12.889	0.909
85	80/10/10	DGAM	rf	svr	7503.745	3749.872	17.229	34.744	0.328	16.035	0.926
86	80/10/10	DGAM	svr	spline	9473.683	4734.842	16.623	38.449	0.178	13.361	0.850
87	80/10/10	DGAM	svr	ebm	9450.052	4723.026	16.623	38.449	0.178	13.164	0.851
88	80/10/10	DGAM	svr	xgb	9842.460	4919.23	16.623	38.449	0.178	13.107	0.859
89	80/10/10	DGAM	svr	rf	9658.349	4827.174	16.623	38.449	0.178	13.326	0.851
90	80/10/10	DGAM	svr	svr	8701.247	4348.624	16.623	38.449	0.178	13.786	0.868

Table B.11 Predictive performance of GAM, DGAM, and machine learning models for serum aspartate aminotransferase (AST/SGOT) levels across data splitting strategies using the National Health Checkup dataset of South Korea (2002–2020).

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	13835.414	6915.707	5.757	18.236	0.458	6.192	0.985
2	60/20/20	GAM	ebm	const	14140.918	7068.459	5.833	20.062	0.344	6.636	0.987
3	60/20/20	GAM	xgb	const	14072.875	7034.438	5.761	19.640	0.371	6.509	0.986
4	60/20/20	GAM	rf	const	14394.409	7195.204	6.394	21.716	0.231	7.230	0.987
5	60/20/20	GAM	svr	const	13936.223	6966.112	5.611	18.818	0.423	6.289	0.986
6	60/20/20	DGAM	spline	spline	12762.596	6379.298	5.757	18.236	0.458	4.446	0.898
7	60/20/20	DGAM	spline	ebm	13183.365	6589.682	5.757	18.236	0.458	4.455	0.897
8	60/20/20	DGAM	spline	xgb	12990.863	6493.432	5.757	18.236	0.458	4.453	0.892
9	60/20/20	DGAM	spline	rf	15278.087	7637.044	5.757	18.236	0.458	4.487	0.894
10	60/20/20	DGAM	spline	svr	11865.115	5930.558	5.757	18.236	0.458	4.422	0.944
11	60/20/20	DGAM	ebm	spline	12155.615	6075.808	5.833	20.062	0.344	4.509	0.897
12	60/20/20	DGAM	ebm	ebm	12628.607	6312.304	5.833	20.062	0.344	4.503	0.901
13	60/20/20	DGAM	ebm	xgb	12545.774	6270.887	5.833	20.062	0.344	4.500	0.899
14	60/20/20	DGAM	ebm	rf	15210.199	7603.1	5.833	20.062	0.344	4.548	0.900
15	60/20/20	DGAM	ebm	svr	11449.557	5722.778	5.833	20.062	0.344	4.486	0.939
16	60/20/20	DGAM	xgb	spline	12087.630	6041.815	5.761	19.640	0.371	4.439	0.892
17	60/20/20	DGAM	xgb	ebm	12474.089	6235.044	5.761	19.640	0.371	4.441	0.891
18	60/20/20	DGAM	xgb	xgb	12231.194	6113.597	5.761	19.640	0.371	4.428	0.889
19	60/20/20	DGAM	xgb	rf	15325.153	7660.576	5.761	19.640	0.371	4.486	0.896
20	60/20/20	DGAM	xgb	svr	11394.666	5695.333	5.761	19.640	0.371	4.412	0.936
21	60/20/20	DGAM	rf	spline	12796.580	6396.29	6.394	21.716	0.231	4.920	0.893
22	60/20/20	DGAM	rf	ebm	12959.105	6477.552	6.394	21.716	0.231	4.917	0.895
23	60/20/20	DGAM	rf	xgb	12791.309	6393.654	6.394	21.716	0.231	4.899	0.896
24	60/20/20	DGAM	rf	rf	16237.070	8116.535	6.394	21.716	0.231	4.971	0.900
25	60/20/20	DGAM	rf	svr	12366.222	6181.111	6.394	21.716	0.231	4.904	0.939
26	60/20/20	DGAM	svr	spline	12560.004	6278.002	5.611	18.818	0.423	4.363	0.882
27	60/20/20	DGAM	svr	ebm	14016.959	7006.48	5.611	18.818	0.423	4.382	0.881
28	60/20/20	DGAM	svr	xgb	13896.925	6946.462	5.611	18.818	0.423	4.388	0.878
29	60/20/20	DGAM	svr	rf	16050.522	8023.261	5.611	18.818	0.423	4.412	0.886
30	60/20/20	DGAM	svr	svr	11885.910	5940.955	5.611	18.818	0.423	4.337	0.922
31	70/15/15	GAM	spline	const	9217.019	4606.51	5.713	11.208	0.577	4.849	0.972
32	70/15/15	GAM	ebm	const	9326.724	4661.362	5.624	11.732	0.536	4.930	0.972
33	70/15/15	GAM	xgb	const	9306.339	4651.17	5.625	11.633	0.544	4.902	0.973
34	70/15/15	GAM	rf	const	9589.612	4792.806	6.220	13.089	0.423	5.407	0.978
35	70/15/15	GAM	svr	const	9233.072	4614.536	5.442	11.282	0.571	4.773	0.973
36	70/15/15	DGAM	spline	spline	9227.593	4611.796	5.713	11.208	0.577	4.288	0.901
37	70/15/15	DGAM	spline	ebm	9038.135	4517.068	5.713	11.208	0.577	4.295	0.899
38	70/15/15	DGAM	spline	xgb	9078.611	4537.306	5.713	11.208	0.577	4.297	0.895
39	70/15/15	DGAM	spline	rf	9292.685	4644.342	5.713	11.208	0.577	4.326	0.900
40	70/15/15	DGAM	spline	svr	8560.258	4278.129	5.713	11.208	0.577	4.253	0.940
41	70/15/15	DGAM	ebm	spline	8894.992	4445.496	5.624	11.732	0.536	4.235	0.896
42	70/15/15	DGAM	ebm	ebm	8638.466	4317.233	5.624	11.732	0.536	4.221	0.899
43	70/15/15	DGAM	ebm	xgb	8619.647	4307.824	5.624	11.732	0.536	4.221	0.899
44	70/15/15	DGAM	ebm	rf	9239.403	4617.702	5.624	11.732	0.536	4.278	0.899
45	70/15/15	DGAM	ebm	svr	8337.293	4166.646	5.624	11.732	0.536	4.206	0.938
46	70/15/15	DGAM	xgb	spline	8838.372	4417.186	5.625	11.633	0.544	4.233	0.888
47	70/15/15	DGAM	xgb	ebm	8675.077	4335.538	5.625	11.633	0.544	4.210	0.892
48	70/15/15	DGAM	xgb	xgb	8742.776	4369.388	5.625	11.633	0.544	4.217	0.894
49	70/15/15	DGAM	xgb	rf	9109.145	4552.572	5.625	11.633	0.544	4.272	0.885
50	70/15/15	DGAM	xgb	svr	8334.967	4165.484	5.625	11.633	0.544	4.202	0.929
51	70/15/15	DGAM	rf	spline	9324.410	4660.205	6.220	13.089	0.423	4.697	0.873
52	70/15/15	DGAM	rf	ebm	9126.300	4561.15	6.220	13.089	0.423	4.669	0.895
53	70/15/15	DGAM	rf	xgb	8978.628	4487.314	6.220	13.089	0.423	4.655	0.890
54	70/15/15	DGAM	rf	rf	9747.771	4871.886	6.220	13.089	0.423	4.717	0.891
55	70/15/15	DGAM	rf	svr	8703.357	4349.678	6.220	13.089	0.423	4.651	0.935
56	70/15/15	DGAM	svr	spline	9212.044	4604.022	5.442	11.282	0.571	4.142	0.878
57	70/15/15	DGAM	svr	ebm	9118.109	4557.054	5.442	11.282	0.571	4.141	0.883
58	70/15/15	DGAM	svr	xgb	9101.826	4548.913	5.442	11.282	0.571	4.142	0.881
59	70/15/15	DGAM	svr	rf	9605.328	4800.664	5.442	11.282	0.571	4.177	0.888
60	70/15/15	DGAM	svr	svr	8626.062	4311.031	5.442	11.282	0.571	4.106	0.921
61	80/10/10	GAM	spline	const	6008.428	3002.214	5.630	10.317	0.649	4.717	0.964
62	80/10/10	GAM	ebm	const	6166.269	3081.134	5.646	11.387	0.573	4.949	0.964
63	80/10/10	GAM	xgb	const	6109.876	3052.938	5.452	10.993	0.602	4.755	0.964
64	80/10/10	GAM	rf	const	6330.946	3163.473	6.157	12.622	0.475	5.384	0.971
65	80/10/10	GAM	svr	const	6102.298	3049.149	5.339	10.939	0.606	4.751	0.966
66	80/10/10	DGAM	spline	spline	5908.388	2952.194	5.630	10.317	0.649	4.219	0.896
67	80/10/10	DGAM	spline	ebm	5673.118	2834.559	5.630	10.317	0.649	4.213	0.892
68	80/10/10	DGAM	spline	xgb	5673.266	2834.633	5.630	10.317	0.649	4.218	0.891
69	80/10/10	DGAM	spline	rf	5794.486	2895.243	5.630	10.317	0.649	4.266	0.896
70	80/10/10	DGAM	spline	svr	5525.835	2760.918	5.630	10.317	0.649	4.190	0.939
71	80/10/10	DGAM	ebm	spline	5704.400	2850.2	5.646	11.387	0.573	4.243	0.905
72	80/10/10	DGAM	ebm	ebm	5551.057	2773.528	5.646	11.387	0.573	4.210	0.900
73	80/10/10	DGAM	ebm	xgb	5512.564	2754.282	5.646	11.387	0.573	4.201	0.905
74	80/10/10	DGAM	ebm	rf	5795.664	2895.832	5.646	11.387	0.573	4.304	0.892
75	80/10/10	DGAM	ebm	svr	5409.430	2702.715	5.646	11.387	0.573	4.220	0.938
76	80/10/10	DGAM	xgb	spline	5724.380	2860.19	5.452	10.993	0.602	4.118	0.901
77	80/10/10	DGAM	xgb	ebm	5520.193	2758.096	5.452	10.993	0.602	4.092	0.902
78	80/10/10	DGAM	xgb	xgb	5504.466	2750.233	5.452	10.993	0.602	4.087	0.900
79	80/10/10	DGAM	xgb	rf	5761.147	2878.574	5.452	10.993	0.602	4.151	0.898
80	80/10/10	DGAM	xgb	svr	5407.826	2701.913	5.452	10.993	0.602	4.095	0.936
81	80/10/10	DGAM	rf	spline	6298.413	3147.206	6.157	12.622	0.475	4.658	0.889
82	80/10/10	DGAM	rf	ebm	5872.269	2934.134	6.157	12.622	0.475	4.619	0.894
83	80/10/10	DGAM	rf	xgb	5830.692	2913.346	6.157	12.622	0.475	4.603	0.886
84	80/10/10	DGAM	rf	rf	6065.846	3030.923	6.157	12.622	0.475	4.694	0.890
85	80/10/10	DGAM	rf	svr	5786.031	2891.016	6.157	12.622	0.475	4.630	0.936
86	80/10/10	DGAM	svr	spline	5992.659	2994.33	5.339	10.939	0.606	4.088	0.890
87	80/10/10	DGAM	svr	ebm	5719.944	2857.972	5.339	10.939	0.606	4.070	0.894
88	80/10/10	DGAM	svr	xgb	5787.985	2891.992	5.339	10.939	0.606	4.081	0.894
89	80/10/10	DGAM	svr	rf	6068.732	3032.366	5.339	10.939	0.606	4.134	0.886
90	80/10/10	DGAM	svr	svr	5608.916	2802.458	5.339	10.939	0.606	4.059	0.928

Table B.12 Predictive performance of GAM, DGAM, and machine learning models for serum alanine aminotransferase (SGPT) levels across data splitting strategies using the National Health Checkup dataset of South Korea (2002–2020).

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	13504.592	6750.296	7.614	16.445	0.361	6.697	0.970
2	60/20/20	GAM	ebm	const	12772.284	6384.142	7.140	13.081	0.596	5.987	0.956
3	60/20/20	GAM	xgb	const	12731.077	6363.538	6.925	12.913	0.606	5.843	0.962
4	60/20/20	GAM	rf	const	13059.004	6527.502	7.542	14.307	0.516	6.323	0.966
5	60/20/20	GAM	svr	const	13284.730	6640.365	7.017	15.353	0.443	6.272	0.966
6	60/20/20	DGAM	spline	spline	11734.046	5865.023	7.614	16.445	0.361	5.540	0.888
7	60/20/20	DGAM	spline	ebm	11659.592	5827.796	7.614	16.445	0.361	5.560	0.890
8	60/20/20	DGAM	spline	xgb	11723.334	5859.667	7.614	16.445	0.361	5.564	0.894
9	60/20/20	DGAM	spline	rf	12284.249	6140.124	7.614	16.445	0.361	5.684	0.886
10	60/20/20	DGAM	spline	svr	11387.512	5691.756	7.614	16.445	0.361	5.502	0.938
11	60/20/20	DGAM	ebm	spline	11459.349	5727.674	7.140	13.081	0.596	5.195	0.868
12	60/20/20	DGAM	ebm	ebm	11414.564	5705.282	7.140	13.081	0.596	5.182	0.883
13	60/20/20	DGAM	ebm	xgb	11447.497	5721.748	7.140	13.081	0.596	5.180	0.879
14	60/20/20	DGAM	ebm	rf	11726.833	5861.416	7.140	13.081	0.596	5.316	0.876
15	60/20/20	DGAM	ebm	svr	11053.681	5524.84	7.140	13.081	0.596	5.145	0.930
16	60/20/20	DGAM	xgb	spline	11281.805	5638.902	6.925	12.913	0.606	5.032	0.885
17	60/20/20	DGAM	xgb	ebm	11248.687	5622.344	6.925	12.913	0.606	5.024	0.892
18	60/20/20	DGAM	xgb	xgb	11310.810	5653.405	6.925	12.913	0.606	5.025	0.888
19	60/20/20	DGAM	xgb	rf	11616.859	5806.43	6.925	12.913	0.606	5.156	0.881
20	60/20/20	DGAM	xgb	svr	10923.821	5459.91	6.925	12.913	0.606	4.992	0.936
21	60/20/20	DGAM	rf	spline	11807.606	5901.803	7.542	14.307	0.516	5.512	0.868
22	60/20/20	DGAM	rf	ebm	11662.965	5829.482	7.542	14.307	0.516	5.485	0.882
23	60/20/20	DGAM	rf	xgb	11678.922	5837.461	7.542	14.307	0.516	5.489	0.881
24	60/20/20	DGAM	rf	rf	12341.650	6168.825	7.542	14.307	0.516	5.617	0.879
25	60/20/20	DGAM	rf	svr	11345.378	5670.689	7.542	14.307	0.516	5.459	0.931
26	60/20/20	DGAM	svr	spline	11391.393	5693.696	7.017	15.353	0.443	5.140	0.882
27	60/20/20	DGAM	svr	ebm	11485.789	5740.894	7.017	15.353	0.443	5.155	0.879
28	60/20/20	DGAM	svr	xgb	11564.888	5780.444	7.017	15.353	0.443	5.169	0.872
29	60/20/20	DGAM	svr	rf	12161.240	6078.62	7.017	15.353	0.443	5.277	0.871
30	60/20/20	DGAM	svr	svr	11036.112	5516.056	7.017	15.353	0.443	5.094	0.931
31	70/15/15	GAM	spline	const	9512.116	4754.058	7.577	12.673	0.581	6.175	0.949
32	70/15/15	GAM	ebm	const	9538.458	4767.229	7.294	12.813	0.572	6.046	0.953
33	70/15/15	GAM	xgb	const	9536.708	4766.354	7.166	12.803	0.572	5.979	0.958
34	70/15/15	GAM	rf	const	9600.650	4798.325	7.614	13.149	0.549	6.173	0.953
35	70/15/15	GAM	svr	const	9528.903	4762.452	7.171	12.762	0.575	6.027	0.948
36	70/15/15	DGAM	spline	spline	8685.184	4340.592	7.577	12.673	0.581	5.484	0.882
37	70/15/15	DGAM	spline	ebm	8661.734	4328.867	7.577	12.673	0.581	5.483	0.887
38	70/15/15	DGAM	spline	xgb	8674.050	4335.025	7.577	12.673	0.581	5.477	0.888
39	70/15/15	DGAM	spline	rf	8791.132	4393.566	7.577	12.673	0.581	5.606	0.878
40	70/15/15	DGAM	spline	svr	8510.812	4253.406	7.577	12.673	0.581	5.452	0.926
41	70/15/15	DGAM	ebm	spline	8743.261	4369.63	7.294	12.813	0.572	5.347	0.863
42	70/15/15	DGAM	ebm	ebm	8603.919	4299.96	7.294	12.813	0.572	5.312	0.883
43	70/15/15	DGAM	ebm	xgb	8611.364	4303.682	7.294	12.813	0.572	5.314	0.885
44	70/15/15	DGAM	ebm	rf	8752.368	4374.184	7.294	12.813	0.572	5.424	0.874
45	70/15/15	DGAM	ebm	svr	8370.613	4183.306	7.294	12.813	0.572	5.292	0.922
46	70/15/15	DGAM	xgb	spline	8652.698	4324.349	7.166	12.803	0.572	5.250	0.863
47	70/15/15	DGAM	xgb	ebm	8518.878	4257.439	7.166	12.803	0.572	5.220	0.877
48	70/15/15	DGAM	xgb	xgb	8521.134	4258.567	7.166	12.803	0.572	5.224	0.879
49	70/15/15	DGAM	xgb	rf	8784.419	4390.21	7.166	12.803	0.572	5.348	0.863
50	70/15/15	DGAM	xgb	svr	8312.668	4154.334	7.166	12.803	0.572	5.199	0.921
51	70/15/15	DGAM	rf	spline	8956.052	4476.026	7.614	13.149	0.549	5.577	0.848
52	70/15/15	DGAM	rf	ebm	8732.568	4364.284	7.614	13.149	0.549	5.504	0.881
53	70/15/15	DGAM	rf	xgb	8723.588	4359.794	7.614	13.149	0.549	5.498	0.886
54	70/15/15	DGAM	rf	rf	8963.345	4479.672	7.614	13.149	0.549	5.624	0.870
55	70/15/15	DGAM	rf	svr	8604.855	4300.428	7.614	13.149	0.549	5.518	0.916
56	70/15/15	DGAM	svr	spline	8678.601	4337.3	7.171	12.762	0.575	5.279	0.859
57	70/15/15	DGAM	svr	ebm	8647.258	4321.629	7.171	12.762	0.575	5.256	0.864
58	70/15/15	DGAM	svr	xgb	8659.444	4327.722	7.171	12.762	0.575	5.257	0.863
59	70/15/15	DGAM	svr	rf	8855.360	4425.68	7.171	12.762	0.575	5.394	0.853
60	70/15/15	DGAM	svr	svr	8434.751	4215.376	7.171	12.762	0.575	5.223	0.916
61	80/10/10	GAM	spline	const	6221.361	3108.68	7.240	11.786	0.668	5.808	0.948
62	80/10/10	GAM	ebm	const	6247.463	3121.732	6.948	11.980	0.657	5.724	0.946
63	80/10/10	GAM	xgb	const	6283.803	3139.902	6.838	12.255	0.641	5.754	0.949
64	80/10/10	GAM	rf	const	6407.192	3201.596	7.461	13.238	0.581	6.097	0.954
65	80/10/10	GAM	svr	const	6201.332	3098.666	6.794	11.639	0.676	5.631	0.946
66	80/10/10	DGAM	spline	spline	5807.542	2901.771	7.240	11.786	0.668	5.159	0.891
67	80/10/10	DGAM	spline	ebm	5787.269	2891.634	7.240	11.786	0.668	5.166	0.899
68	80/10/10	DGAM	spline	xgb	5779.918	2887.959	7.240	11.786	0.668	5.159	0.901
69	80/10/10	DGAM	spline	rf	5852.918	2924.459	7.240	11.786	0.668	5.256	0.896
70	80/10/10	DGAM	spline	svr	5620.947	2808.474	7.240	11.786	0.668	5.134	0.941
71	80/10/10	DGAM	ebm	spline	5634.973	2815.486	6.948	11.980	0.657	5.022	0.890
72	80/10/10	DGAM	ebm	ebm	5616.425	2806.212	6.948	11.980	0.657	5.002	0.901
73	80/10/10	DGAM	ebm	xgb	5603.432	2799.716	6.948	11.980	0.657	5.011	0.896
74	80/10/10	DGAM	ebm	rf	5764.295	2880.148	6.948	11.980	0.657	5.110	0.884
75	80/10/10	DGAM	ebm	svr	5527.647	2761.824	6.948	11.980	0.657	5.004	0.938
76	80/10/10	DGAM	xgb	spline	5643.215	2819.608	6.838	12.255	0.641	4.960	0.891
77	80/10/10	DGAM	xgb	ebm	5603.444	2799.722	6.838	12.255	0.641	4.956	0.895
78	80/10/10	DGAM	xgb	xgb	5615.337	2805.668	6.838	12.255	0.641	4.949	0.895
79	80/10/10	DGAM	xgb	rf	5847.619	2921.81	6.838	12.255	0.641	5.090	0.884
80	80/10/10	DGAM	xgb	svr	5483.047	2739.524	6.838	12.255	0.641	4.949	0.941
81	80/10/10	DGAM	rf	spline	5896.374	2946.187	7.461	13.238	0.581	5.357	0.886
82	80/10/10	DGAM	rf	ebm	5930.659	2963.33	7.461	13.238	0.581	5.346	0.896
83	80/10/10	DGAM	rf	xgb	5950.114	2973.057	7.461	13.238	0.581	5.341	0.896
84	80/10/10	DGAM	rf	rf	6088.728	3042.364	7.461	13.238	0.581	5.472	0.886
85	80/10/10	DGAM	rf	svr	5710.616	2853.308	7.461	13.238	0.581	5.345	0.935
86	80/10/10	DGAM	svr	spline	5665.037	2830.518	6.794	11.639	0.676	4.936	0.895
87	80/10/10	DGAM	svr	ebm	5696.457	2846.228	6.794	11.639	0.676	4.906	0.892
88	80/10/10	DGAM	svr	xgb	5704.597	2850.298	6.794	11.639	0.676	4.911	0.898
89	80/10/10	DGAM	svr	rf	5773.775	2884.888	6.794	11.639	0.676	5.015	0.884
90	80/10/10	DGAM	svr	svr	5541.368	2768.684	6.794	11.639	0.676	4.924	0.930

Table B.13 Predictive performance of GAM, DGAM, and machine learning models for life expectancy across data splitting strategies using the World Health Organization life expectancy dataset.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	1758.911	877.456	0.817	1.035	0.987	0.584	0.894
2	60/20/20	GAM	ebm	const	1383.912	689.956	0.594	0.765	0.993	0.426	0.895
3	60/20/20	GAM	xgb	const	1338.570	667.285	0.565	0.731	0.994	0.407	0.897
4	60/20/20	GAM	rf	const	1113.128	554.564	0.453	0.608	0.996	0.332	0.895
5	60/20/20	GAM	svr	const	1458.449	727.224	0.571	0.816	0.992	0.430	0.906
6	60/20/20	DGAM	spline	spline	1753.461	872.73	0.817	1.035	0.987	0.568	0.888
7	60/20/20	DGAM	spline	ebm	1538.195	765.098	0.817	1.035	0.987	0.557	0.906
8	60/20/20	DGAM	spline	xgb	1525.483	758.742	0.817	1.035	0.987	0.554	0.901
9	60/20/20	DGAM	spline	rf	1432.242	712.121	0.817	1.035	0.987	0.534	0.916
10	60/20/20	DGAM	spline	svr	1659.454	825.727	0.817	1.035	0.987	0.560	0.871
11	60/20/20	DGAM	ebm	spline	1458.393	725.196	0.594	0.765	0.993	0.425	0.885
12	60/20/20	DGAM	ebm	ebm	1363.238	677.619	0.594	0.765	0.993	0.423	0.897
13	60/20/20	DGAM	ebm	xgb	1361.090	676.545	0.594	0.765	0.993	0.422	0.897
14	60/20/20	DGAM	ebm	rf	1317.781	654.89	0.594	0.765	0.993	0.410	0.892
15	60/20/20	DGAM	ebm	svr	1452.762	722.381	0.594	0.765	0.993	0.422	0.876
16	60/20/20	DGAM	xgb	spline	1416.392	704.196	0.565	0.731	0.994	0.406	0.883
17	60/20/20	DGAM	xgb	ebm	1369.822	680.911	0.565	0.731	0.994	0.405	0.890
18	60/20/20	DGAM	xgb	xgb	1379.159	685.58	0.565	0.731	0.994	0.405	0.888
19	60/20/20	DGAM	xgb	rf	1342.840	667.42	0.565	0.731	0.994	0.394	0.887
20	60/20/20	DGAM	xgb	svr	1529.204	760.602	0.565	0.731	0.994	0.405	0.874
21	60/20/20	DGAM	rf	spline	1176.466	584.233	0.453	0.608	0.996	0.335	0.883
22	60/20/20	DGAM	rf	ebm	1054.135	523.068	0.453	0.608	0.996	0.326	0.892
23	60/20/20	DGAM	rf	xgb	1062.173	527.086	0.453	0.608	0.996	0.325	0.902
24	60/20/20	DGAM	rf	rf	1192.407	592.204	0.453	0.608	0.996	0.318	0.871
25	60/20/20	DGAM	rf	svr	1183.739	587.87	0.453	0.608	0.996	0.327	0.885
26	60/20/20	DGAM	svr	spline	1782.067	887.034	0.571	0.816	0.992	0.428	0.909
27	60/20/20	DGAM	svr	ebm	1472.530	732.265	0.571	0.816	0.992	0.412	0.920
28	60/20/20	DGAM	svr	xgb	1547.507	769.754	0.571	0.816	0.992	0.410	0.918
29	60/20/20	DGAM	svr	rf	1765.921	878.96	0.571	0.816	0.992	0.386	0.921
30	60/20/20	DGAM	svr	svr	2959.736	1475.868	0.571	0.816	0.992	0.422	0.932
31	70/15/15	GAM	spline	const	1263.026	629.513	0.770	1.002	0.988	0.553	0.909
32	70/15/15	GAM	ebm	const	958.231	477.116	0.554	0.725	0.994	0.400	0.930
33	70/15/15	GAM	xgb	const	909.305	452.652	0.520	0.685	0.994	0.376	0.928
34	70/15/15	GAM	rf	const	877.526	436.763	0.426	0.640	0.995	0.322	0.930
35	70/15/15	GAM	svr	const	1008.481	502.24	0.511	0.743	0.993	0.391	0.891
36	70/15/15	DGAM	spline	spline	1233.829	612.914	0.770	1.002	0.988	0.538	0.895
37	70/15/15	DGAM	spline	ebm	1204.595	598.298	0.770	1.002	0.988	0.531	0.923
38	70/15/15	DGAM	spline	xgb	1220.454	606.227	0.770	1.002	0.988	0.528	0.921
39	70/15/15	DGAM	spline	rf	1433.846	712.923	0.770	1.002	0.988	0.516	0.905
40	70/15/15	DGAM	spline	svr	1307.794	649.897	0.770	1.002	0.988	0.533	0.879
41	70/15/15	DGAM	ebm	spline	972.463	482.232	0.554	0.725	0.994	0.396	0.935
42	70/15/15	DGAM	ebm	ebm	940.608	466.304	0.554	0.725	0.994	0.394	0.914
43	70/15/15	DGAM	ebm	xgb	961.423	476.712	0.554	0.725	0.994	0.394	0.902
44	70/15/15	DGAM	ebm	rf	929.227	460.614	0.554	0.725	0.994	0.381	0.914
45	70/15/15	DGAM	ebm	svr	938.450	465.225	0.554	0.725	0.994	0.391	0.912
46	70/15/15	DGAM	xgb	spline	939.755	465.878	0.520	0.685	0.994	0.374	0.928
47	70/15/15	DGAM	xgb	ebm	898.940	445.47	0.520	0.685	0.994	0.371	0.919
48	70/15/15	DGAM	xgb	xgb	909.353	450.676	0.520	0.685	0.994	0.371	0.912
49	70/15/15	DGAM	xgb	rf	880.545	436.272	0.520	0.685	0.994	0.359	0.914
50	70/15/15	DGAM	xgb	svr	863.602	427.801	0.520	0.685	0.994	0.366	0.935
51	70/15/15	DGAM	rf	spline	807.376	399.688	0.426	0.640	0.995	0.320	0.914
52	70/15/15	DGAM	rf	ebm	766.467	379.234	0.426	0.640	0.995	0.313	0.919
53	70/15/15	DGAM	rf	xgb	779.354	385.677	0.426	0.640	0.995	0.313	0.921
54	70/15/15	DGAM	rf	rf	637.777	314.888	0.426	0.640	0.995	0.299	0.914
55	70/15/15	DGAM	rf	svr	1048.966	520.483	0.426	0.640	0.995	0.315	0.900
56	70/15/15	DGAM	svr	spline	2863.300	1427.65	0.511	0.743	0.993	0.387	0.881
57	70/15/15	DGAM	svr	ebm	1330.164	661.082	0.511	0.743	0.993	0.378	0.898
58	70/15/15	DGAM	svr	xgb	1358.327	675.164	0.511	0.743	0.993	0.377	0.900
59	70/15/15	DGAM	svr	rf	1656.085	824.042	0.511	0.743	0.993	0.356	0.895
60	70/15/15	DGAM	svr	svr	2243.133	1117.566	0.511	0.743	0.993	0.383	0.898
61	80/10/10	GAM	spline	const	864.933	430.466	0.740	1.057	0.987	0.536	0.948
62	80/10/10	GAM	ebm	const	751.587	373.794	0.557	0.811	0.993	0.410	0.927
63	80/10/10	GAM	xgb	const	746.519	371.26	0.527	0.782	0.993	0.389	0.927
64	80/10/10	GAM	rf	const	631.082	313.541	0.403	0.600	0.996	0.301	0.913
65	80/10/10	GAM	svr	const	625.153	310.576	0.482	0.705	0.994	0.365	0.934
66	80/10/10	DGAM	spline	spline	828.783	410.392	0.740	1.057	0.987	0.526	0.941
67	80/10/10	DGAM	spline	ebm	852.271	422.136	0.740	1.057	0.987	0.520	0.927
68	80/10/10	DGAM	spline	xgb	843.070	417.535	0.740	1.057	0.987	0.517	0.927
69	80/10/10	DGAM	spline	rf	1425.662	708.831	0.740	1.057	0.987	0.496	0.909
70	80/10/10	DGAM	spline	svr	761.653	376.826	0.740	1.057	0.987	0.509	0.937
71	80/10/10	DGAM	ebm	spline	811.638	401.819	0.557	0.811	0.993	0.407	0.889
72	80/10/10	DGAM	ebm	ebm	762.929	377.464	0.557	0.811	0.993	0.405	0.889
73	80/10/10	DGAM	ebm	xgb	775.991	383.996	0.557	0.811	0.993	0.404	0.889
74	80/10/10	DGAM	ebm	rf	913.966	452.983	0.557	0.811	0.993	0.392	0.906
75	80/10/10	DGAM	ebm	svr	659.074	325.537	0.557	0.811	0.993	0.401	0.913
76	80/10/10	DGAM	xgb	spline	886.572	439.286	0.527	0.782	0.993	0.387	0.895
77	80/10/10	DGAM	xgb	ebm	755.950	373.975	0.527	0.782	0.993	0.386	0.895
78	80/10/10	DGAM	xgb	xgb	780.839	386.42	0.527	0.782	0.993	0.386	0.899
79	80/10/10	DGAM	xgb	rf	992.176	492.088	0.527	0.782	0.993	0.374	0.909
80	80/10/10	DGAM	xgb	svr	662.472	327.236	0.527	0.782	0.993	0.383	0.902
81	80/10/10	DGAM	rf	spline	598.808	295.404	0.403	0.600	0.996	0.297	0.916
82	80/10/10	DGAM	rf	ebm	491.594	241.797	0.403	0.600	0.996	0.293	0.923
83	80/10/10	DGAM	rf	xgb	493.272	242.636	0.403	0.600	0.996	0.293	0.909
84	80/10/10	DGAM	rf	rf	553.902	272.951	0.403	0.600	0.996	0.283	0.909
85	80/10/10	DGAM	rf	svr	484.815	238.408	0.403	0.600	0.996	0.292	0.920
86	80/10/10	DGAM	svr	spline	943.323	467.662	0.482	0.705	0.994	0.360	0.916
87	80/10/10	DGAM	svr	ebm	730.057	361.028	0.482	0.705	0.994	0.352	0.895
88	80/10/10	DGAM	svr	xgb	725.726	358.863	0.482	0.705	0.994	0.351	0.902
89	80/10/10	DGAM	svr	rf	680.815	336.408	0.482	0.705	0.994	0.334	0.906
90	80/10/10	DGAM	svr	svr	1153.843	572.922	0.482	0.705	0.994	0.355	0.909

Table B.14 Predictive performance of GAM, DGAM, and machine learning models for adult mortality across data splitting strategies using the World Health Organization life expectancy dataset.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	4952.701	2474.35	13.709	17.630	0.974	9.788	0.918
2	60/20/20	GAM	ebm	const	4727.988	2361.994	9.951	14.319	0.983	7.516	0.899
3	60/20/20	GAM	xgb	const	4682.986	2339.493	9.433	13.667	0.984	7.143	0.897
4	60/20/20	GAM	rf	const	4461.487	2228.744	7.355	11.471	0.989	5.832	0.914
5	60/20/20	GAM	svr	const	5474.471	2735.236	17.453	28.372	0.933	14.131	0.920
6	60/20/20	DGAM	spline	spline	4991.578	2491.789	13.709	17.630	0.974	9.657	0.885
7	60/20/20	DGAM	spline	ebm	4809.775	2400.888	13.709	17.630	0.974	9.379	0.895
8	60/20/20	DGAM	spline	xgb	4791.217	2391.608	13.709	17.630	0.974	9.316	0.899
9	60/20/20	DGAM	spline	rf	4886.951	2439.476	13.709	17.630	0.974	9.012	0.914
10	60/20/20	DGAM	spline	svr	4892.949	2442.474	13.709	17.630	0.974	9.363	0.890
11	60/20/20	DGAM	ebm	spline	4781.274	2386.637	9.951	14.319	0.983	7.238	0.920
12	60/20/20	DGAM	ebm	ebm	4506.919	2249.46	9.951	14.319	0.983	7.148	0.908
13	60/20/20	DGAM	ebm	xgb	4508.454	2250.227	9.951	14.319	0.983	7.130	0.908
14	60/20/20	DGAM	ebm	rf	4395.845	2193.922	9.951	14.319	0.983	6.802	0.921
15	60/20/20	DGAM	ebm	svr	4516.052	2254.026	9.951	14.319	0.983	7.066	0.895
16	60/20/20	DGAM	xgb	spline	4753.265	2372.632	9.433	13.667	0.984	6.905	0.923
17	60/20/20	DGAM	xgb	ebm	4518.609	2255.304	9.433	13.667	0.984	6.846	0.916
18	60/20/20	DGAM	xgb	xgb	4495.926	2243.963	9.433	13.667	0.984	6.831	0.916
19	60/20/20	DGAM	xgb	rf	4369.424	2180.712	9.433	13.667	0.984	6.497	0.916
20	60/20/20	DGAM	xgb	svr	4455.480	2223.74	9.433	13.667	0.984	6.771	0.899
21	60/20/20	DGAM	rf	spline	4196.403	2094.202	7.355	11.471	0.989	5.484	0.901
22	60/20/20	DGAM	rf	ebm	4038.371	2015.186	7.355	11.471	0.989	5.309	0.928
23	60/20/20	DGAM	rf	xgb	4030.773	2011.386	7.355	11.471	0.989	5.295	0.923
24	60/20/20	DGAM	rf	rf	3838.761	1915.38	7.355	11.471	0.989	5.048	0.925
25	60/20/20	DGAM	rf	svr	4168.848	2080.424	7.355	11.471	0.989	5.362	0.908
26	60/20/20	DGAM	svr	spline	5377.758	2684.879	17.453	28.372	0.933	13.190	0.925
27	60/20/20	DGAM	svr	ebm	4956.499	2474.25	17.453	28.372	0.933	12.333	0.923
28	60/20/20	DGAM	svr	xgb	4952.876	2472.438	17.453	28.372	0.933	12.281	0.923
29	60/20/20	DGAM	svr	rf	5003.577	2497.788	17.453	28.372	0.933	11.542	0.883
30	60/20/20	DGAM	svr	svr	4931.304	2461.652	17.453	28.372	0.933	11.982	0.901
31	70/15/15	GAM	spline	const	3753.867	1874.934	13.146	18.288	0.972	9.722	0.909
32	70/15/15	GAM	ebm	const	3562.891	1779.446	10.293	14.220	0.983	7.716	0.881
33	70/15/15	GAM	xgb	const	3515.671	1755.836	9.734	13.451	0.985	7.297	0.872
34	70/15/15	GAM	rf	const	3396.739	1696.37	7.445	11.973	0.988	5.965	0.907
35	70/15/15	GAM	svr	const	4208.267	2102.134	18.461	30.198	0.924	14.858	0.895
36	70/15/15	DGAM	spline	spline	3703.809	1847.904	13.146	18.288	0.972	9.355	0.900
37	70/15/15	DGAM	spline	ebm	3574.602	1783.301	13.146	18.288	0.972	9.302	0.905
38	70/15/15	DGAM	spline	xgb	3567.613	1779.806	13.146	18.288	0.972	9.268	0.907
39	70/15/15	DGAM	spline	rf	3397.175	1694.588	13.146	18.288	0.972	8.861	0.900
40	70/15/15	DGAM	spline	svr	3757.351	1874.676	13.146	18.288	0.972	9.154	0.891
41	70/15/15	DGAM	ebm	spline	3625.472	1808.736	10.293	14.220	0.983	7.553	0.879
42	70/15/15	DGAM	ebm	ebm	3424.466	1708.233	10.293	14.220	0.983	7.371	0.870
43	70/15/15	DGAM	ebm	xgb	3418.821	1705.41	10.293	14.220	0.983	7.354	0.881
44	70/15/15	DGAM	ebm	rf	3376.959	1684.48	10.293	14.220	0.983	7.053	0.886
45	70/15/15	DGAM	ebm	svr	3595.235	1793.618	10.293	14.220	0.983	7.272	0.916
46	70/15/15	DGAM	xgb	spline	3577.973	1784.986	9.734	13.451	0.985	7.205	0.874
47	70/15/15	DGAM	xgb	ebm	3419.587	1705.794	9.734	13.451	0.985	6.994	0.874
48	70/15/15	DGAM	xgb	xgb	3408.985	1700.492	9.734	13.451	0.985	6.965	0.884
49	70/15/15	DGAM	xgb	rf	3418.330	1705.165	9.734	13.451	0.985	6.716	0.881
50	70/15/15	DGAM	xgb	svr	3596.944	1794.472	9.734	13.451	0.985	6.956	0.895
51	70/15/15	DGAM	rf	spline	3268.739	1630.37	7.445	11.973	0.988	5.638	0.926
52	70/15/15	DGAM	rf	ebm	3036.043	1514.022	7.445	11.973	0.988	5.393	0.937
53	70/15/15	DGAM	rf	xgb	3021.838	1506.919	7.445	11.973	0.988	5.374	0.928
54	70/15/15	DGAM	rf	rf	2959.858	1475.929	7.445	11.973	0.988	5.210	0.921
55	70/15/15	DGAM	rf	svr	3139.163	1565.582	7.445	11.973	0.988	5.355	0.926
56	70/15/15	DGAM	svr	spline	3926.665	1959.332	18.461	30.198	0.924	13.602	0.895
57	70/15/15	DGAM	svr	ebm	3825.381	1908.69	18.461	30.198	0.924	13.135	0.888
58	70/15/15	DGAM	svr	xgb	3830.433	1911.216	18.461	30.198	0.924	13.044	0.893
59	70/15/15	DGAM	svr	rf	3435.670	1713.835	18.461	30.198	0.924	11.866	0.919
60	70/15/15	DGAM	svr	svr	3788.763	1890.382	18.461	30.198	0.924	12.450	0.916
61	80/10/10	GAM	spline	const	2521.336	1258.668	12.613	18.376	0.973	9.189	0.930
62	80/10/10	GAM	ebm	const	2284.343	1140.172	9.116	12.496	0.987	6.669	0.909
63	80/10/10	GAM	xgb	const	2249.489	1122.744	8.573	11.762	0.989	6.269	0.916
64	80/10/10	GAM	rf	const	2176.573	1086.286	6.036	9.618	0.993	4.756	0.892
65	80/10/10	GAM	svr	const	2736.364	1366.182	16.554	26.880	0.942	13.359	0.899
66	80/10/10	DGAM	spline	spline	2476.685	1234.342	12.613	18.376	0.973	8.958	0.916
67	80/10/10	DGAM	spline	ebm	2656.794	1324.397	12.613	18.376	0.973	8.805	0.913
68	80/10/10	DGAM	spline	xgb	2708.595	1350.298	12.613	18.376	0.973	8.776	0.920
69	80/10/10	DGAM	spline	rf	5498.988	2745.494	12.613	18.376	0.973	8.511	0.895
70	80/10/10	DGAM	spline	svr	5493.467	2742.734	12.613	18.376	0.973	8.764	0.895
71	80/10/10	DGAM	ebm	spline	2337.213	1164.606	9.116	12.496	0.987	6.546	0.899
72	80/10/10	DGAM	ebm	ebm	2208.710	1100.355	9.116	12.496	0.987	6.449	0.899
73	80/10/10	DGAM	ebm	xgb	2195.599	1093.8	9.116	12.496	0.987	6.417	0.902
74	80/10/10	DGAM	ebm	rf	2211.991	1101.996	9.116	12.496	0.987	6.252	0.868
75	80/10/10	DGAM	ebm	svr	2257.490	1124.745	9.116	12.496	0.987	6.392	0.906
76	80/10/10	DGAM	xgb	spline	2358.548	1175.274	8.573	11.762	0.989	6.198	0.882
77	80/10/10	DGAM	xgb	ebm	2182.336	1087.168	8.573	11.762	0.989	6.067	0.913
78	80/10/10	DGAM	xgb	xgb	2194.137	1093.068	8.573	11.762	0.989	6.064	0.909
79	80/10/10	DGAM	xgb	rf	2216.745	1104.372	8.573	11.762	0.989	5.930	0.857
80	80/10/10	DGAM	xgb	svr	2289.814	1140.907	8.573	11.762	0.989	6.064	0.878
81	80/10/10	DGAM	rf	spline	2020.548	1006.274	6.036	9.618	0.993	4.565	0.920
82	80/10/10	DGAM	rf	ebm	2068.668	1030.334	6.036	9.618	0.993	4.367	0.916
83	80/10/10	DGAM	rf	xgb	2061.554	1026.777	6.036	9.618	0.993	4.347	0.920
84	80/10/10	DGAM	rf	rf	2061.528	1026.764	6.036	9.618	0.993	4.199	0.909
85	80/10/10	DGAM	rf	svr	2148.658	1070.329	6.036	9.618	0.993	4.344	0.923
86	80/10/10	DGAM	svr	spline	2731.448	1361.724	16.554	26.880	0.942	12.319	0.889
87	80/10/10	DGAM	svr	ebm	2413.295	1202.648	16.554	26.880	0.942	11.592	0.916
88	80/10/10	DGAM	svr	xgb	2396.847	1194.424	16.554	26.880	0.942	11.530	0.902
89	80/10/10	DGAM	svr	rf	2371.905	1181.952	16.554	26.880	0.942	10.871	0.885
90	80/10/10	DGAM	svr	svr	2488.799	1240.4	16.554	26.880	0.942	11.125	0.895

Table B.15 Predictive performance of GAM, DGAM, and machine learning models for under-five deaths across data splitting strategies using the World Health Organization life expectancy dataset.

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	3437.321	1716.66	2.191	3.750	0.993	1.800	0.866
2	60/20/20	GAM	ebm	const	2910.804	1453.402	1.545	2.784	0.996	1.344	0.873
3	60/20/20	GAM	xgb	const	2837.446	1416.723	1.479	2.659	0.996	1.288	0.881
4	60/20/20	GAM	rf	const	3113.444	1554.722	1.523	3.195	0.995	1.419	0.908
5	60/20/20	GAM	svr	const	4616.909	2306.454	2.716	6.743	0.977	2.538	0.890
6	60/20/20	DGAM	spline	spline	2780.976	1386.488	2.191	3.750	0.993	1.611	0.874
7	60/20/20	DGAM	spline	ebm	2362.998	1177.499	2.191	3.750	0.993	1.552	0.906
8	60/20/20	DGAM	spline	xgb	2362.659	1177.33	2.191	3.750	0.993	1.544	0.908
9	60/20/20	DGAM	spline	rf	2402.191	1197.096	2.191	3.750	0.993	1.497	0.885
10	60/20/20	DGAM	spline	svr	2560.919	1276.46	2.191	3.750	0.993	1.530	0.892
11	60/20/20	DGAM	ebm	spline	2376.443	1184.222	1.545	2.784	0.996	1.144	0.890
12	60/20/20	DGAM	ebm	ebm	1881.140	936.57	1.545	2.784	0.996	1.123	0.913
13	60/20/20	DGAM	ebm	xgb	1892.486	942.243	1.545	2.784	0.996	1.126	0.909
14	60/20/20	DGAM	ebm	rf	1904.258	948.129	1.545	2.784	0.996	1.106	0.908
15	60/20/20	DGAM	ebm	svr	2013.542	1002.771	1.545	2.784	0.996	1.119	0.876
16	60/20/20	DGAM	xgb	spline	2316.163	1154.082	1.479	2.659	0.996	1.099	0.892
17	60/20/20	DGAM	xgb	ebm	1881.238	936.619	1.479	2.659	0.996	1.080	0.902
18	60/20/20	DGAM	xgb	xgb	1896.450	944.225	1.479	2.659	0.996	1.081	0.906
19	60/20/20	DGAM	xgb	rf	2035.453	1013.726	1.479	2.659	0.996	1.066	0.911
20	60/20/20	DGAM	xgb	svr	1939.654	965.827	1.479	2.659	0.996	1.066	0.885
21	60/20/20	DGAM	rf	spline	1745.922	868.961	1.523	3.195	0.995	1.120	0.899
22	60/20/20	DGAM	rf	ebm	1619.822	805.911	1.523	3.195	0.995	1.083	0.921
23	60/20/20	DGAM	rf	xgb	1651.146	821.573	1.523	3.195	0.995	1.082	0.911
24	60/20/20	DGAM	rf	rf	1452.971	722.486	1.523	3.195	0.995	1.033	0.901
25	60/20/20	DGAM	rf	svr	1515.705	753.852	1.523	3.195	0.995	1.069	0.925
26	60/20/20	DGAM	svr	spline	3288.376	1640.188	2.716	6.743	0.977	2.087	0.909
27	60/20/20	DGAM	svr	ebm	3624.148	1808.074	2.716	6.743	0.977	2.011	0.885
28	60/20/20	DGAM	svr	xgb	3197.927	1594.964	2.716	6.743	0.977	2.008	0.888
29	60/20/20	DGAM	svr	rf	3252.776	1622.388	2.716	6.743	0.977	1.876	0.902
30	60/20/20	DGAM	svr	svr	2779.168	1385.584	2.716	6.743	0.977	1.913	0.908
31	70/15/15	GAM	spline	const	2477.539	1236.77	1.896	3.465	0.993	1.567	0.905
32	70/15/15	GAM	ebm	const	2125.280	1060.64	1.288	2.469	0.997	1.118	0.886
33	70/15/15	GAM	xgb	const	2108.340	1052.17	1.228	2.370	0.997	1.063	0.888
34	70/15/15	GAM	rf	const	2495.072	1245.536	1.606	3.303	0.994	1.472	0.879
35	70/15/15	GAM	svr	const	2798.048	1397.024	2.336	5.157	0.985	2.174	0.919
36	70/15/15	DGAM	spline	spline	1753.764	872.882	1.896	3.465	0.993	1.401	0.919
37	70/15/15	DGAM	spline	ebm	1647.494	819.747	1.896	3.465	0.993	1.350	0.914
38	70/15/15	DGAM	spline	xgb	1639.017	815.508	1.896	3.465	0.993	1.344	0.912
39	70/15/15	DGAM	spline	rf	1575.152	783.576	1.896	3.465	0.993	1.288	0.912
40	70/15/15	DGAM	spline	svr	1772.290	882.145	1.896	3.465	0.993	1.352	0.860
41	70/15/15	DGAM	ebm	spline	1442.332	717.166	1.288	2.469	0.997	0.954	0.928
42	70/15/15	DGAM	ebm	ebm	1387.255	689.628	1.288	2.469	0.997	0.936	0.928
43	70/15/15	DGAM	ebm	xgb	1375.566	683.783	1.288	2.469	0.997	0.936	0.930
44	70/15/15	DGAM	ebm	rf	1571.260	781.63	1.288	2.469	0.997	0.927	0.900
45	70/15/15	DGAM	ebm	svr	1844.576	918.288	1.288	2.469	0.997	0.942	0.884
46	70/15/15	DGAM	xgb	spline	1412.354	702.177	1.228	2.370	0.997	0.917	0.926
47	70/15/15	DGAM	xgb	ebm	1355.679	673.84	1.228	2.370	0.997	0.896	0.921
48	70/15/15	DGAM	xgb	xgb	1358.410	675.205	1.228	2.370	0.997	0.897	0.919
49	70/15/15	DGAM	xgb	rf	1558.114	775.057	1.228	2.370	0.997	0.893	0.881
50	70/15/15	DGAM	xgb	svr	1850.741	921.37	1.228	2.370	0.997	0.902	0.888
51	70/15/15	DGAM	rf	spline	1561.278	776.639	1.606	3.303	0.994	1.198	0.881
52	70/15/15	DGAM	rf	ebm	1345.798	668.899	1.606	3.303	0.994	1.144	0.891
53	70/15/15	DGAM	rf	xgb	1351.773	671.886	1.606	3.303	0.994	1.135	0.900
54	70/15/15	DGAM	rf	rf	1230.447	611.224	1.606	3.303	0.994	1.087	0.912
55	70/15/15	DGAM	rf	svr	1295.769	643.884	1.606	3.303	0.994	1.114	0.926
56	70/15/15	DGAM	svr	spline	2715.634	1353.817	2.336	5.157	0.985	1.728	0.886
57	70/15/15	DGAM	svr	ebm	2223.108	1107.554	2.336	5.157	0.985	1.699	0.872
58	70/15/15	DGAM	svr	xgb	2180.999	1086.5	2.336	5.157	0.985	1.691	0.870
59	70/15/15	DGAM	svr	rf	2339.331	1165.666	2.336	5.157	0.985	1.618	0.914
60	70/15/15	DGAM	svr	svr	1898.004	945.002	2.336	5.157	0.985	1.662	0.902
61	80/10/10	GAM	spline	const	3598.553	1797.276	2.117	5.594	0.985	1.792	0.850
62	80/10/10	GAM	ebm	const	2962.546	1479.273	1.365	4.422	0.990	1.212	0.875
63	80/10/10	GAM	xgb	const	3527.554	1761.777	1.325	4.378	0.991	1.169	0.857
64	80/10/10	GAM	rf	const	1813.003	904.502	1.266	3.474	0.994	1.189	0.902
65	80/10/10	GAM	svr	const	2423.514	1209.757	2.168	6.367	0.980	2.072	0.906
66	80/10/10	DGAM	spline	spline	1954.216	973.108	2.117	5.594	0.985	1.616	0.906
67	80/10/10	DGAM	spline	ebm	1916.307	954.154	2.117	5.594	0.985	1.586	0.916
68	80/10/10	DGAM	spline	xgb	2016.915	1004.458	2.117	5.594	0.985	1.584	0.909
69	80/10/10	DGAM	spline	rf	10498.524	5245.262	2.117	5.594	0.985	1.500	0.909
70	80/10/10	DGAM	spline	svr	1230.343	611.172	2.117	5.594	0.985	1.554	0.899
71	80/10/10	DGAM	ebm	spline	1119.034	555.517	1.365	4.422	0.990	1.047	0.906
72	80/10/10	DGAM	ebm	ebm	976.862	484.431	1.365	4.422	0.990	1.039	0.895
73	80/10/10	DGAM	ebm	xgb	941.692	466.846	1.365	4.422	0.990	1.035	0.895
74	80/10/10	DGAM	ebm	rf	1172.893	582.446	1.365	4.422	0.990	1.018	0.902
75	80/10/10	DGAM	ebm	svr	949.875	470.938	1.365	4.422	0.990	1.037	0.909
76	80/10/10	DGAM	xgb	spline	1640.710	816.355	1.325	4.378	0.991	1.026	0.927
77	80/10/10	DGAM	xgb	ebm	1003.626	497.813	1.325	4.378	0.991	1.010	0.930
78	80/10/10	DGAM	xgb	xgb	970.687	481.344	1.325	4.378	0.991	1.008	0.920
79	80/10/10	DGAM	xgb	rf	1340.131	666.066	1.325	4.378	0.991	0.995	0.902
80	80/10/10	DGAM	xgb	svr	1092.580	542.29	1.325	4.378	0.991	1.011	0.916
81	80/10/10	DGAM	rf	spline	713.948	352.974	1.266	3.474	0.994	0.957	0.937
82	80/10/10	DGAM	rf	ebm	667.474	329.737	1.266	3.474	0.994	0.917	0.923
83	80/10/10	DGAM	rf	xgb	629.460	310.73	1.266	3.474	0.994	0.911	0.916
84	80/10/10	DGAM	rf	rf	811.554	401.777	1.266	3.474	0.994	0.879	0.920
85	80/10/10	DGAM	rf	svr	587.832	289.916	1.266	3.474	0.994	0.887	0.923
86	80/10/10	DGAM	svr	spline	1427.922	709.961	2.168	6.367	0.980	1.679	0.906
87	80/10/10	DGAM	svr	ebm	1111.621	551.81	2.168	6.367	0.980	1.593	0.916
88	80/10/10	DGAM	svr	xgb	1135.304	563.652	2.168	6.367	0.980	1.589	0.906
89	80/10/10	DGAM	svr	rf	1245.776	618.888	2.168	6.367	0.980	1.516	0.923
90	80/10/10	DGAM	svr	svr	961.842	476.921	2.168	6.367	0.980	1.495	0.916

Table B.16 Predictive performance of GAM, DGAM, and machine learning models for infant mortality outcomes across data splitting strategies using the National Health Checkup dataset of South Korea (2002–2020).

Order	Split	Group	MeanModel	DispModel	AIC	NLL	MAE	RMSE	R^2	CRPS	PICP95
1	60/20/20	GAM	spline	const	2872.778	1434.389	1.502	2.557	0.991	1.217	0.894
2	60/20/20	GAM	ebm	const	2285.655	1140.828	0.928	1.677	0.996	0.801	0.901
3	60/20/20	GAM	xgb	const	2241.159	1118.58	0.909	1.625	0.996	0.780	0.901
4	60/20/20	GAM	rf	const	2337.880	1166.94	0.957	1.822	0.996	0.887	0.892
5	60/20/20	GAM	svr	const	3123.363	1559.682	1.387	2.905	0.989	1.225	0.914
6	60/20/20	DGAM	spline	spline	2578.345	1285.172	1.502	2.557	0.991	1.091	0.867
7	60/20/20	DGAM	spline	ebm	2444.463	1218.232	1.502	2.557	0.991	1.066	0.899
8	60/20/20	DGAM	spline	xgb	2500.969	1246.484	1.502	2.557	0.991	1.061	0.902
9	60/20/20	DGAM	spline	rf	2600.744	1296.372	1.502	2.557	0.991	1.021	0.895
10	60/20/20	DGAM	spline	svr	2691.162	1341.581	1.502	2.557	0.991	1.031	0.890
11	60/20/20	DGAM	ebm	spline	1735.746	863.873	0.928	1.677	0.996	0.685	0.892
12	60/20/20	DGAM	ebm	ebm	1454.092	723.046	0.928	1.677	0.996	0.677	0.895
13	60/20/20	DGAM	ebm	xgb	1447.116	719.558	0.928	1.677	0.996	0.675	0.909
14	60/20/20	DGAM	ebm	rf	1762.847	877.424	0.928	1.677	0.996	0.665	0.881
15	60/20/20	DGAM	ebm	svr	1541.837	766.918	0.928	1.677	0.996	0.668	0.902
16	60/20/20	DGAM	xgb	spline	1605.153	798.576	0.909	1.625	0.996	0.671	0.894
17	60/20/20	DGAM	xgb	ebm	1379.907	685.954	0.909	1.625	0.996	0.661	0.913
18	60/20/20	DGAM	xgb	xgb	1369.806	680.903	0.909	1.625	0.996	0.659	0.916
19	60/20/20	DGAM	xgb	rf	1584.185	788.092	0.909	1.625	0.996	0.652	0.895
20	60/20/20	DGAM	xgb	svr	1630.882	811.441	0.909	1.625	0.996	0.656	0.906
21	60/20/20	DGAM	rf	spline	1210.810	601.405	0.957	1.822	0.996	0.720	0.897
22	60/20/20	DGAM	rf	ebm	1341.265	666.632	0.957	1.822	0.996	0.686	0.911
23	60/20/20	DGAM	rf	xgb	1325.546	658.773	0.957	1.822	0.996	0.687	0.906
24	60/20/20	DGAM	rf	rf	1265.473	628.736	0.957	1.822	0.996	0.670	0.895
25	60/20/20	DGAM	rf	svr	1242.799	617.4	0.957	1.822	0.996	0.696	0.894
26	60/20/20	DGAM	svr	spline	17401.039	8696.52	1.387	2.905	0.989	1.052	0.881
27	60/20/20	DGAM	svr	ebm	2054.994	1023.497	1.387	2.905	0.989	1.022	0.904
28	60/20/20	DGAM	svr	xgb	2042.423	1017.212	1.387	2.905	0.989	1.019	0.904
29	60/20/20	DGAM	svr	rf	2290.646	1141.323	1.387	2.905	0.989	0.967	0.899
30	60/20/20	DGAM	svr	svr	2770.981	1381.49	1.387	2.905	0.989	1.000	0.878
31	70/15/15	GAM	spline	const	2251.463	1123.732	1.448	2.625	0.990	1.202	0.898
32	70/15/15	GAM	ebm	const	2126.931	1061.466	0.992	1.966	0.995	0.853	0.870
33	70/15/15	GAM	xgb	const	2137.925	1066.962	0.940	1.888	0.995	0.809	0.867
34	70/15/15	GAM	rf	const	2177.379	1086.69	1.118	2.293	0.993	1.028	0.867
35	70/15/15	GAM	svr	const	2782.050	1389.025	1.429	3.044	0.987	1.249	0.900
36	70/15/15	DGAM	spline	spline	1545.667	768.834	1.448	2.625	0.990	1.045	0.914
37	70/15/15	DGAM	spline	ebm	1761.069	876.534	1.448	2.625	0.990	1.037	0.888
38	70/15/15	DGAM	spline	xgb	1813.154	902.577	1.448	2.625	0.990	1.033	0.895
39	70/15/15	DGAM	spline	rf	1694.079	843.04	1.448	2.625	0.990	0.971	0.923
40	70/15/15	DGAM	spline	svr	1471.285	731.642	1.448	2.625	0.990	0.983	0.909
41	70/15/15	DGAM	ebm	spline	1462.689	727.344	0.992	1.966	0.995	0.751	0.898
42	70/15/15	DGAM	ebm	ebm	1246.989	619.494	0.992	1.966	0.995	0.742	0.902
43	70/15/15	DGAM	ebm	xgb	1196.339	594.17	0.992	1.966	0.995	0.740	0.898
44	70/15/15	DGAM	ebm	rf	1092.384	542.192	0.992	1.966	0.995	0.720	0.898
45	70/15/15	DGAM	ebm	svr	1274.648	633.324	0.992	1.966	0.995	0.737	0.900
46	70/15/15	DGAM	xgb	spline	1448.413	720.206	0.940	1.888	0.995	0.717	0.898
47	70/15/15	DGAM	xgb	ebm	1282.909	637.454	0.940	1.888	0.995	0.715	0.886
48	70/15/15	DGAM	xgb	xgb	1296.462	644.231	0.940	1.888	0.995	0.716	0.874
49	70/15/15	DGAM	xgb	rf	1152.407	572.204	0.940	1.888	0.995	0.693	0.877
50	70/15/15	DGAM	xgb	svr	1327.590	659.795	0.940	1.888	0.995	0.709	0.886
51	70/15/15	DGAM	rf	spline	1089.224	540.612	1.118	2.293	0.993	0.831	0.923
52	70/15/15	DGAM	rf	ebm	937.491	464.746	1.118	2.293	0.993	0.801	0.907
53	70/15/15	DGAM	rf	xgb	902.424	447.212	1.118	2.293	0.993	0.799	0.905
54	70/15/15	DGAM	rf	rf	839.973	415.986	1.118	2.293	0.993	0.756	0.919
55	70/15/15	DGAM	rf	svr	768.647	380.324	1.118	2.293	0.993	0.779	0.923
56	70/15/15	DGAM	svr	spline	2763.862	1377.931	1.429	3.044	0.987	1.093	0.895
57	70/15/15	DGAM	svr	ebm	1664.658	828.329	1.429	3.044	0.987	1.076	0.900
58	70/15/15	DGAM	svr	xgb	1710.509	851.254	1.429	3.044	0.987	1.072	0.905
59	70/15/15	DGAM	svr	rf	4140.008	2066.004	1.429	3.044	0.987	1.002	0.870
60	70/15/15	DGAM	svr	svr	1490.187	741.094	1.429	3.044	0.987	1.029	0.909
61	80/10/10	GAM	spline	const	1615.606	805.803	1.474	2.663	0.991	1.208	0.861
62	80/10/10	GAM	ebm	const	1186.453	591.226	0.855	1.651	0.996	0.732	0.913
63	80/10/10	GAM	xgb	const	1200.326	598.163	0.829	1.634	0.997	0.712	0.913
64	80/10/10	GAM	rf	const	1133.442	564.721	0.913	1.730	0.996	0.844	0.927
65	80/10/10	GAM	svr	const	1273.432	634.716	0.953	1.906	0.995	0.830	0.906
66	80/10/10	DGAM	spline	spline	1228.387	610.194	1.474	2.663	0.991	1.072	0.878
67	80/10/10	DGAM	spline	ebm	1209.029	600.514	1.474	2.663	0.991	1.056	0.899
68	80/10/10	DGAM	spline	xgb	1248.550	620.275	1.474	2.663	0.991	1.052	0.892
69	80/10/10	DGAM	spline	rf	3773.961	1882.98	1.474	2.663	0.991	1.014	0.864
70	80/10/10	DGAM	spline	svr	1610.994	801.497	1.474	2.663	0.991	1.011	0.906
71	80/10/10	DGAM	ebm	spline	742.753	367.376	0.855	1.651	0.996	0.641	0.909
72	80/10/10	DGAM	ebm	ebm	627.089	309.544	0.855	1.651	0.996	0.627	0.923
73	80/10/10	DGAM	ebm	xgb	629.761	310.88	0.855	1.651	0.996	0.627	0.913
74	80/10/10	DGAM	ebm	rf	805.722	398.861	0.855	1.651	0.996	0.620	0.895
75	80/10/10	DGAM	ebm	svr	639.519	315.76	0.855	1.651	0.996	0.616	0.937
76	80/10/10	DGAM	xgb	spline	769.375	380.688	0.829	1.634	0.997	0.628	0.889
77	80/10/10	DGAM	xgb	ebm	640.043	316.022	0.829	1.634	0.997	0.616	0.920
78	80/10/10	DGAM	xgb	xgb	644.684	318.342	0.829	1.634	0.997	0.616	0.913
79	80/10/10	DGAM	xgb	rf	712.594	352.297	0.829	1.634	0.997	0.606	0.909
80	80/10/10	DGAM	xgb	svr	661.220	326.61	0.829	1.634	0.997	0.602	0.927
81	80/10/10	DGAM	rf	spline	538.520	265.26	0.913	1.730	0.996	0.664	0.909
82	80/10/10	DGAM	rf	ebm	659.948	325.974	0.913	1.730	0.996	0.638	0.895
83	80/10/10	DGAM	rf	xgb	668.544	330.272	0.913	1.730	0.996	0.638	0.878
84	80/10/10	DGAM	rf	rf	782.825	387.412	0.913	1.730	0.996	0.625	0.878
85	80/10/10	DGAM	rf	svr	1106.156	549.078	0.913	1.730	0.996	0.635	0.892
86	80/10/10	DGAM	svr	spline	1111.583	551.792	0.953	1.906	0.995	0.772	0.868
87	80/10/10	DGAM	svr	ebm	839.100	415.55	0.953	1.906	0.995	0.712	0.916
88	80/10/10	DGAM	svr	xgb	833.781	412.89	0.953	1.906	0.995	0.708	0.920
89	80/10/10	DGAM	svr	rf	863.239	427.62	0.953	1.906	0.995	0.667	0.864
90	80/10/10	DGAM	svr	svr	707.131	349.566	0.953	1.906	0.995	0.698	0.927

CURRICULUM VITAE

NAME : Natakon Nawaratana **GENDER :** Male

EDUCATION BACKGROUND:

- Master of Science (Applied Mathematics), Suranaree University of Technology, Thailand, 2019

SCHOLARSHIP:

- Institute for the Promotion of Teaching Science and Technology (IPST) Scholarship

PUBLICATIONS AND CONFERENCE PROCEEDINGS:

- Nawaratana, N., Suriyawichitseranee, A., and Tanthanuch, J. (2023). A Comparative Study between Generalized Linear Models and Generalized Additive Models in the Modeling of Health Biological Signal Data. In *Proceedings of the 20th International and National Conference on Applied Computer Technology and Information System (ACTIS)*, August 25, 2023, pp. 61–67.

- Ngekkoy, R., Nawaratana, N., Boonmunewai, J., and Tanthanuch, J. (2022). Mathematical Modelling for The Evaluation of Body Fat Percentage by Data of Body Components. In *Proceedings of The International Conference on Applied Statistics 2022 (ICAS2022)*, November 3–4, 2022, Department of Applied Statistics, Faculty of Applied Science, King Mongkut's University of Technology North Bangkok.

RESEARCH EXPERIENCE:

- Short-term research visit on "Analysis of Double Generalized Additive Models" at Changzhou Institute of Technology, People's Republic of China, May 1, 2024 – March 31, 2025 (11 months), under the supervision of Prof. Dr. Hong Zhang.