**AFRICAN INSTITUTE FOR MATHEMATICAL SCIENCES**

**(AIMS RWANDA, KIGALI)**

---

Name: Jean de Dieu NGIRINSHUTI
Assignment Number: 1
Course: Statistical Machine Learning
Date: December 9, 2024

---

# Part 1: Dataset Exploration

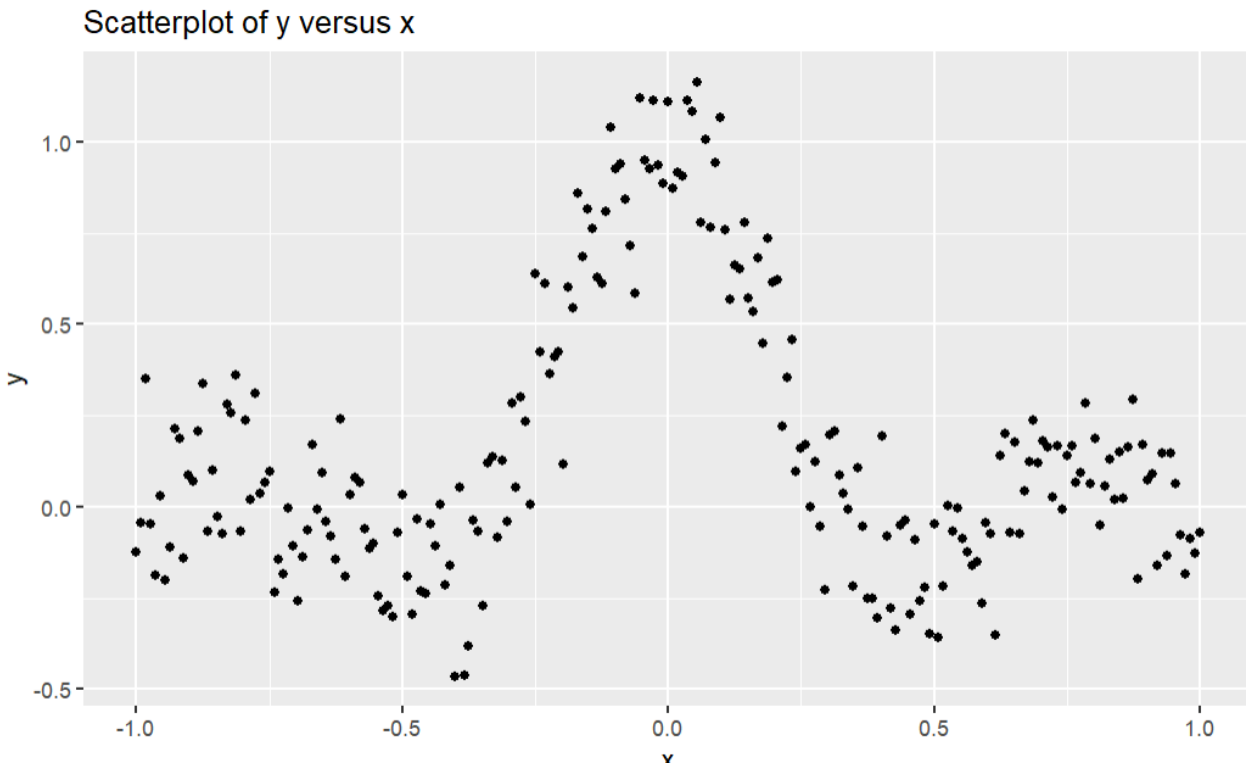
```
# Load the dataset
data <- read.csv('aims-sml-2024-2025-data.csv')
head(data)
```

```
##            x           y
## 1 -1.0000000 -0.12567108
## 2 -0.9910714 -0.04545708
## 3 -0.9821429  0.34967134
## 4 -0.9732143 -0.04689389
## 5 -0.9642857 -0.18649697
## 6 -0.9553571  0.02786734
```

```
# Determine the size of the dataset
n <- nrow(data)
cat("The size of the dataset (n) is:", n)
```

```
## The size of the dataset (n) is: 225
```

```
# Create a scatterplot of y versus x
ggplot(data, aes(x = x, y = y)) +
  geom_point() +
  labs(title = "Scatterplot of y versus x", x = "x", y = "y")
```




**The scatter plot shows a clear, periodic, non-linear relationship between $x$ and $y$, with data points forming distinct peaks and troughs. There are regions where data points are more densely packed, indicating clusters, but no significant outliers are observed. This pattern suggests that a polynomial or other non-linear model may be suitable for capturing the underlying relationship between these variables.**

```
# Check the structure of the dataset
str(data)
```

```
## 'data.frame':    225 obs. of  2 variables:
##  $ x: num  -1 -0.991 -0.982 -0.973 -0.964 ...
##  $ y: num  -0.1257 -0.0455 0.3497 -0.0469 -0.1865 ...
```

```
# Check the unique values in the response variable
unique(data$y)
```

```
##   [1] -0.125671077 -0.045457076  0.349671341 -0.046893887 -0.186496969
##   [6]  0.027867337 -0.201542542 -0.110648110  0.214021842  0.185395162
```

## 4. Determine whether this is a classification or regression task, and justify your answer.

```
## Based on the scatterplot and the characteristics of the response variable
##     `y`, this task is a REGRESSION Task because the target variable `y` is
##     continuous.
```

# Part 2: Theoretical Framework

## 1. Function Space

The function space $H$ is defined as:

$$H = \left\{ f(x) = \sum_{j=0}^{p} \beta_j x^j \mid \beta_j \in \mathbb{R},\, j = 0, 1, \ldots, p \right\}.$$

Here, $p$ is the degree of the polynomial, and $\beta_j$ are the coefficients. This is a set of all polynomial functions up to degree $p$.

## 2. Loss Function

The squared loss for a single observation $(x, y)$ is given by:

$$\text{loss}(y, f(x)) = (y - f(x))^2.$$

For the entire dataset of $n$ observations:

$$L_{\text{total}} = \sum_{i=1}^{n} (y_i - f(x_i))^2.$$

This loss is chosen because:

. It penalizes larger errors more heavily.

. It is convex, ensuring easier optimization.

. It aligns well with regression models.

## 3. Theoretical Risk

The theoretical risk $R(f)$ for a candidate function $f(x) \in H$ is the expected value of the squared loss:

$$R(f) = \mathbb{E}[(Y - f(X))^2],$$

where $(X, Y)$ follows the true joint distribution $P(X, Y)$.

Expanding this:

$$R(f) = \mathbb{E}[Y^2] - 2\mathbb{E}[Y f(X)] + \mathbb{E}[f(X)^2].$$

Here:

- $\mathbb{E}[Y^2]$: the expected squared value of the output.
- $2\mathbb{E}[Y f(X)]$: the expected cross-term of $Y$ and $f(X)$.
- $\mathbb{E}[f(X)^2]$: the expected squared value of the function $f(X)$.

The goal is to find $f(x)$ that minimizes $R(f)$.

## 4. Bayes Learning Machine

The Bayes optimal function $f^*(x)$ minimizes $R(f)$ and is given by:

$$f^*(x) = \mathbb{E}[Y \mid X = x].$$

Substituting $f^*(x)$ into $R(f)$:

$$R(f^*) = \mathbb{E}[(Y - \mathbb{E}[Y \mid X])^2],$$

which is the irreducible error.

### 5. Empirical Risk

The empirical risk $\widehat{R}(f)$ is the sample-based approximation of the theoretical risk:

$$\widehat{R}(f) = \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2.$$

For a polynomial function $f(x) = \sum_{j=0}^{p}\beta_j x^j$, the empirical risk becomes:

$$\widehat{R}(f) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \sum_{j=0}^{p}\beta_j x_i^j\right)^2.$$

## Part 3: Estimation and Model Complexity

### 1. Derivation of $\widehat{\beta}$: OLS Estimator

We aim to minimize the empirical risk:

$$\widehat{\beta} = \arg\min_{\beta}\widehat{R}(f),$$

where

$$\widehat{R}(f) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \mathbf{x}_i^\top\boldsymbol{\beta}\right)^2,$$

and $\mathbf{x}_i = [1, x_i, x_i^2, \ldots, x_i^p]^\top$.

The objective is:

$$L(\boldsymbol{\beta}) = \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2,$$

where:

- $\mathbf{y} = [y_1, y_2, \ldots, y_n]^\top$,
- $\mathbf{X}$ is the design matrix with rows $\mathbf{x}_i^\top$,
- $\boldsymbol{\beta} = [\beta_0, \beta_1, \ldots, \beta_p]^\top$.

Expanding $L(\boldsymbol{\beta})$:

$$L(\boldsymbol{\beta}) = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\top(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}).$$
$$L(\boldsymbol{\beta}) = \mathbf{y}^\top\mathbf{y} - 2\mathbf{y}^\top\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\beta}^\top\mathbf{X}^\top\mathbf{X}\boldsymbol{\beta}.$$

Taking the gradient with respect to $\boldsymbol{\beta}$:

$$\frac{\partial L(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}} = -2\mathbf{X}^\top\mathbf{y} + 2\mathbf{X}^\top\mathbf{X}\boldsymbol{\beta}.$$

Setting the gradient to zero:

$$-2\mathbf{X}^\top\mathbf{y} + 2\mathbf{X}^\top\mathbf{X}\boldsymbol{\beta} = 0.$$
$$\mathbf{X}^\top\mathbf{X}\boldsymbol{\beta} = \mathbf{X}^\top\mathbf{y}.$$

If ($\mathbf{X}^\top\mathbf{X}$ ) is invertible matrix, the solution is:

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}.$$

The estimated function is:

$$\widehat{f}(x) = \mathbf{X}^\top\widehat{\boldsymbol{\beta}},$$

where $\mathbf{X} = [1, x, x^2, \ldots, x^p]^\top$.

## 2. Properties of $\widehat{f}(x)$

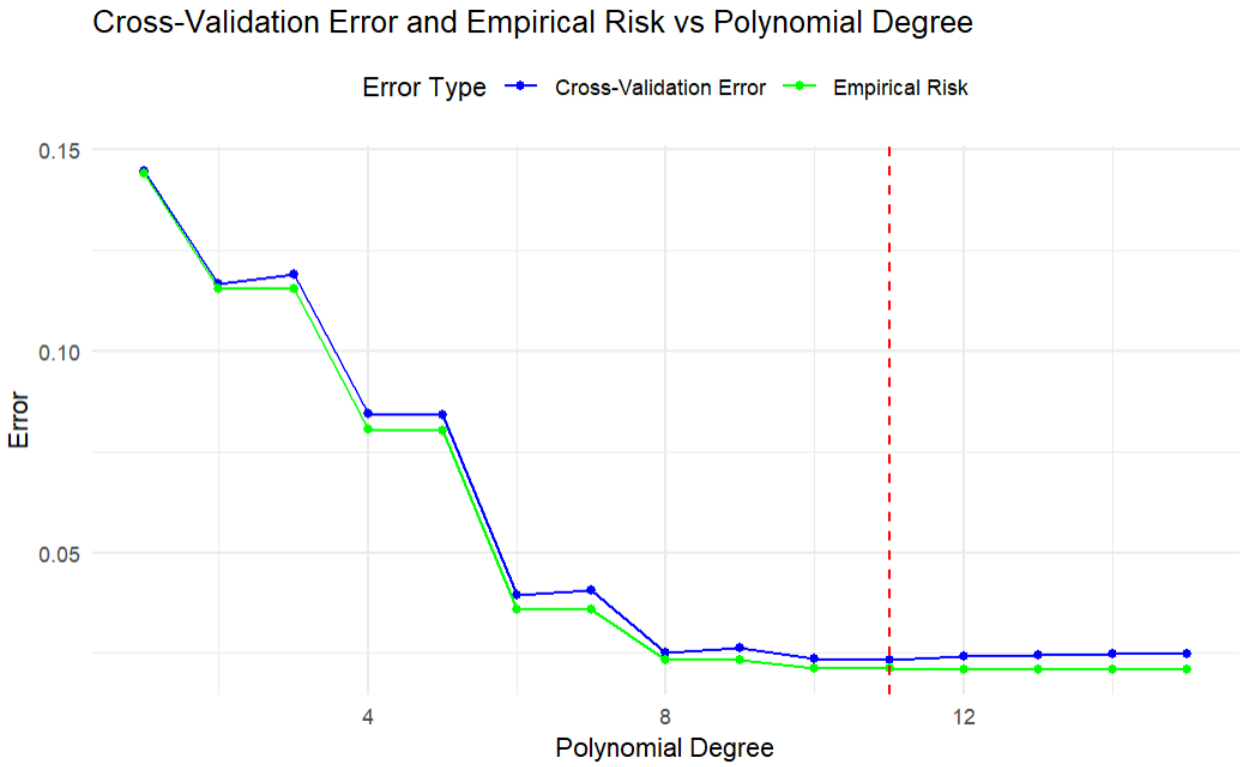
- **Linearity:** Predictions are linear combinations of the predictors.
- **Unbiasedness:** $\mathbb{E}[\widehat{\beta}] = \beta$ under the assumption of a correctly specified model.
- **Efficiency** Minimizes variance among unbiased estimators
- **Consistenc:** Increasing $p$ allows better approximation but risks overfitting.

## 3.Determining Optimal Complexity

```
## Optimal degree p: 11
```

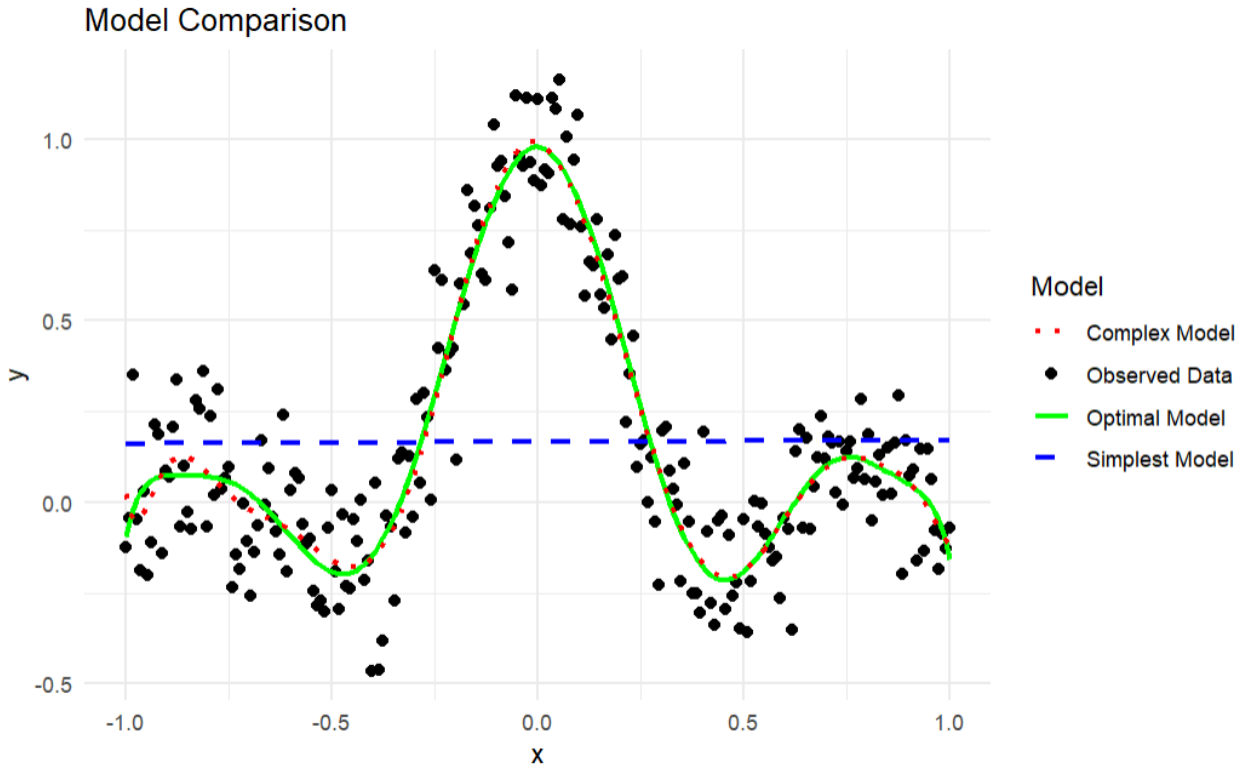
**Optimal complexity refers to the ideal balance between model simplicity and its ability to capture patterns in the data. It minimizes the cross-validation error, indicating a model that generalizes well to unseen data without overfitting or underfitting. Overly simple models (low complexity) fail to capture the data's underlying structure, resulting in underfitting, while overly complex models (high complexity) capture noise in the data, leading to overfitting. Optimal complexity is achieved when the model exhibits minimal cross-validation error, balanced bias and variance, and provides robust performance across both training and test datasets.**

## 4.The cross-validation error and empirical risk as functions of p




**The plot compares the cross-validation error and empirical risk as functions of polynomial degree, illustrating the trade-off between model complexity and error. The empirical risk (green line) decreases steadily as the polynomial degree increases, reflecting improved training data fit. Conversely, the cross-validation error (blue line) decreases initially and stabilizes around degree 11, beyond which higher complexity offers no significant improvement in generalization. The red dashed line indicates the optimal polynomial degree (11), where cross-validation error is minimized, achieving a balance between underfitting and overfitting.**

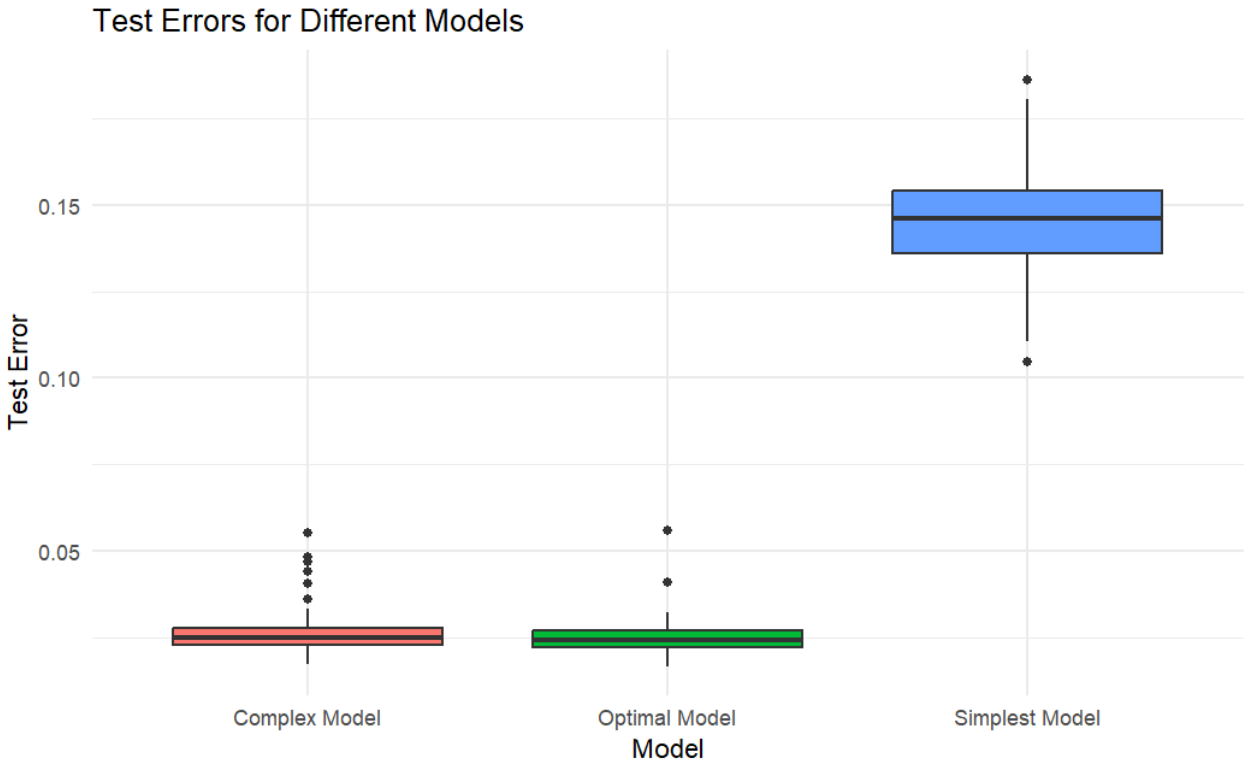
## Part 4: Model Comparison and Evaluation




**The plot illustrates a comparison among three models: the simplest model (blue dashed line), the optimal model (green solid line), and the complex model (red dotted line) alongside the observed data points (black dots). The simplest model underfits the data, failing to capture the underlying trend. Conversely, the complex model overfits the data, capturing noise instead of the true pattern, which compromises its generalization ability. In contrast, the optimal model achieves a balance between simplicity and accuracy, aligning well with the data while avoiding overfitting. This highlights the importance of selecting a model with appropriate complexity for robust generalization.**

## 2. Perform Stochastic Hold-Out Validation

**1.Fit and plot the models**




**2.Model Performance Summary**

∗. **Simplest Model:**

– Underfits the data, resulting in high test error and significant variability, indicating it is too simple to capture the underlying relationship.

∗. **Optimal Model:**

Achieves the lowest and most consistent test error, making it the best-performing model due to its balance between complexity and generalization.

∗. **Complex Model:**

Performs similarly to the optimal model in terms of median test error but has higher variability, indicating overfitting and reduced reliability.

# Part 5: Further Analysis

## 1.Perform ANOVA on Test Errors

```
# Perform ANOVA on test errors
anova_results <- aov(Error ~ Model, data = test_errors)

# Display the ANOVA table
summary(anova_results)
```

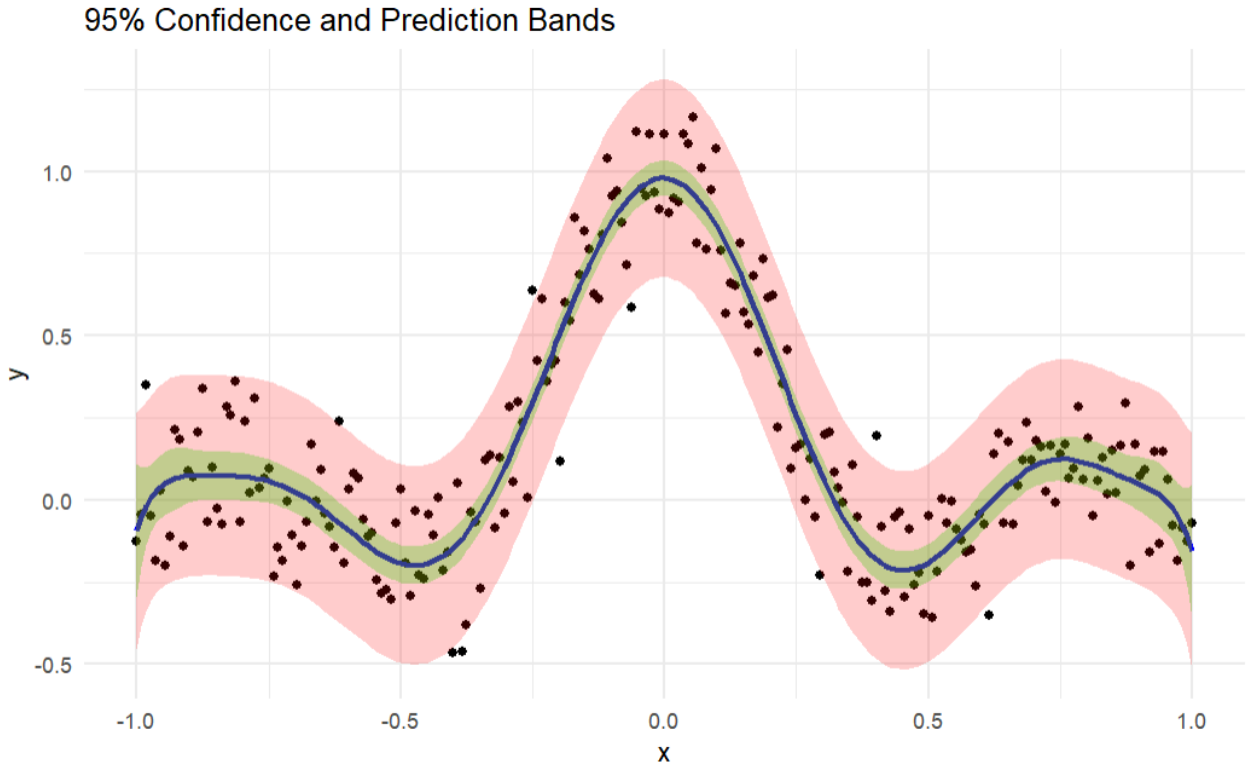
```
##              Df Sum Sq Mean Sq F value Pr(>F)
```

```
## Model         2 0.9527  0.4763    4986 <2e-16 ***
## Residuals   297 0.0284  0.0001
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

**Comment**

1. **ANOVA Results:** The ANOVA results show a highly significant effect of the model on test error (F = 4986, p < 2e-16).
2. **Significance:** This indicates that the mean test errors differ significantly across models.
3. **Variance Explanation:** The model explains a substantial portion of the variation in test errors (Sum Sq = 0.9927), while the residuals contribute minimally (Sum Sq = 0.0284).
4. **Fit Quality:** The small residual mean square (0.0001) suggests the models fit the data well.

## 2. Obtain and plot the 95% confidence and prediction bands for the dataset Dn.




**The plot shows the 95% confidence (green) and prediction (red) bands for the dataset $D_n$. The confidence band reflects uncertainty in estimating the regression line, being narrower and more precise in regions with dense data. The prediction band, which is wider, accounts for both the regression line's uncertainty and the variability of individual observations. This additional width, especially in sparse data regions, highlights greater uncertainty in predicting new observations.**

## 3.Mathematical Expressions for Bands

The 95% confidence band for the true regression line is:

$$\hat{y}_i \pm t_{\alpha/2,n-p-1} \cdot \sqrt{\text{Var}(\hat{y}_i)},$$

The 95% prediction band for a new observation is:

$$\hat{y}_i \pm t_{\alpha/2,n-p-1} \cdot \sqrt{\text{Var}(\hat{y}_i) + \sigma^2},$$

where $\sigma^2$ : Residual variance of the model.

### 4.Comments on the Confidence and Prediction Bands

**Confidence Band:** It represents the uncertainty in estimating the true regression line.

– Narrower bands indicate higher confidence in the model fit.

– Wider bands occur where data is sparse, reflecting greater uncertainty.

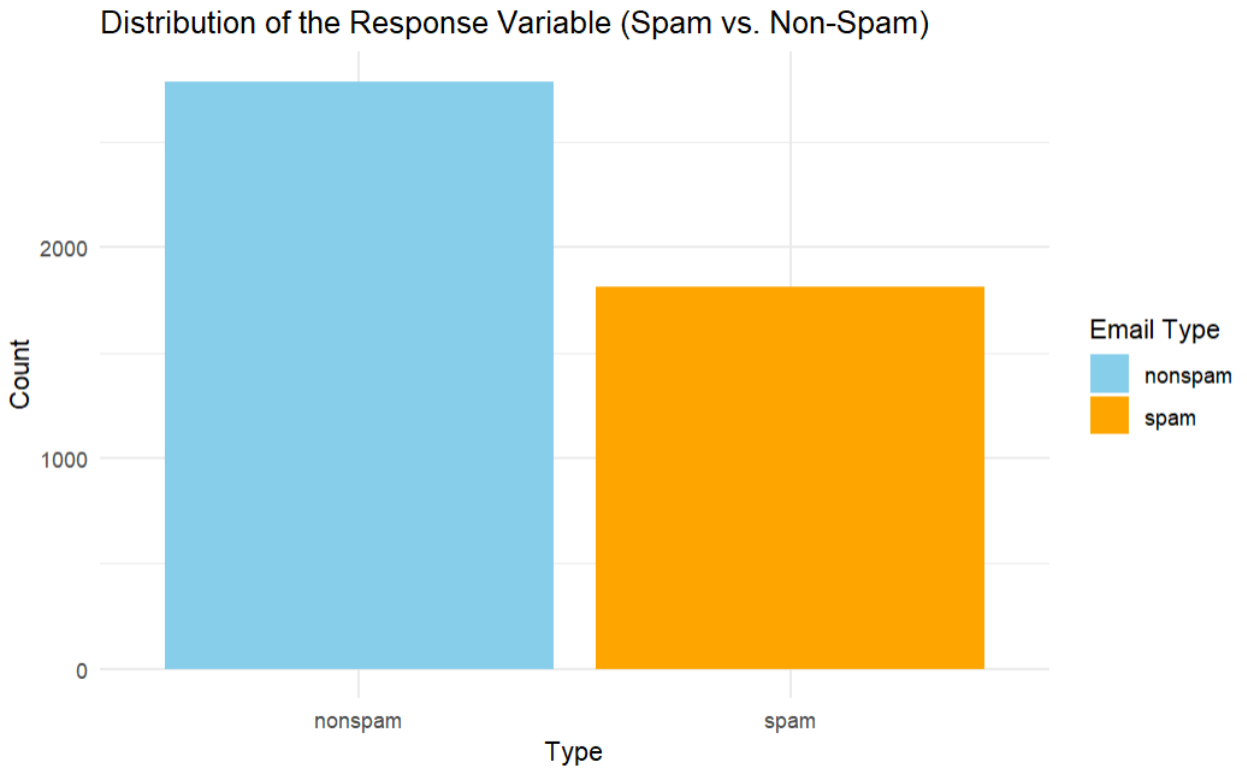
**Prediction Band:**It reflects the uncertainty in predicting new observations.

– Always wider than the confidence band as it includes regression uncertainty and observation variability.

– Wider in sparse data regions due to greater uncertainty.

## Exercise 2: Spam Dataset Analysis

### 1. Plot the Distribution of the Response




```
# Comment:
# The response variable `type` is binary, representing spam and non-spam emails.
# The plot shows that the classes are reasonably balanced, which is ideal for classification tasks.
```

```
# Display the shape of the dataset
cat("Number of observations:", nrow(spam), "\n")
```

```
## Number of observations: 4601
```

```r
cat("Number of features:", ncol(spam) - 1, "\n")
```

```
## Number of features: 57
```

## 2.Comment on the shape

∗. **Response Variable:** The dataset has 4601 observations and 57 features.

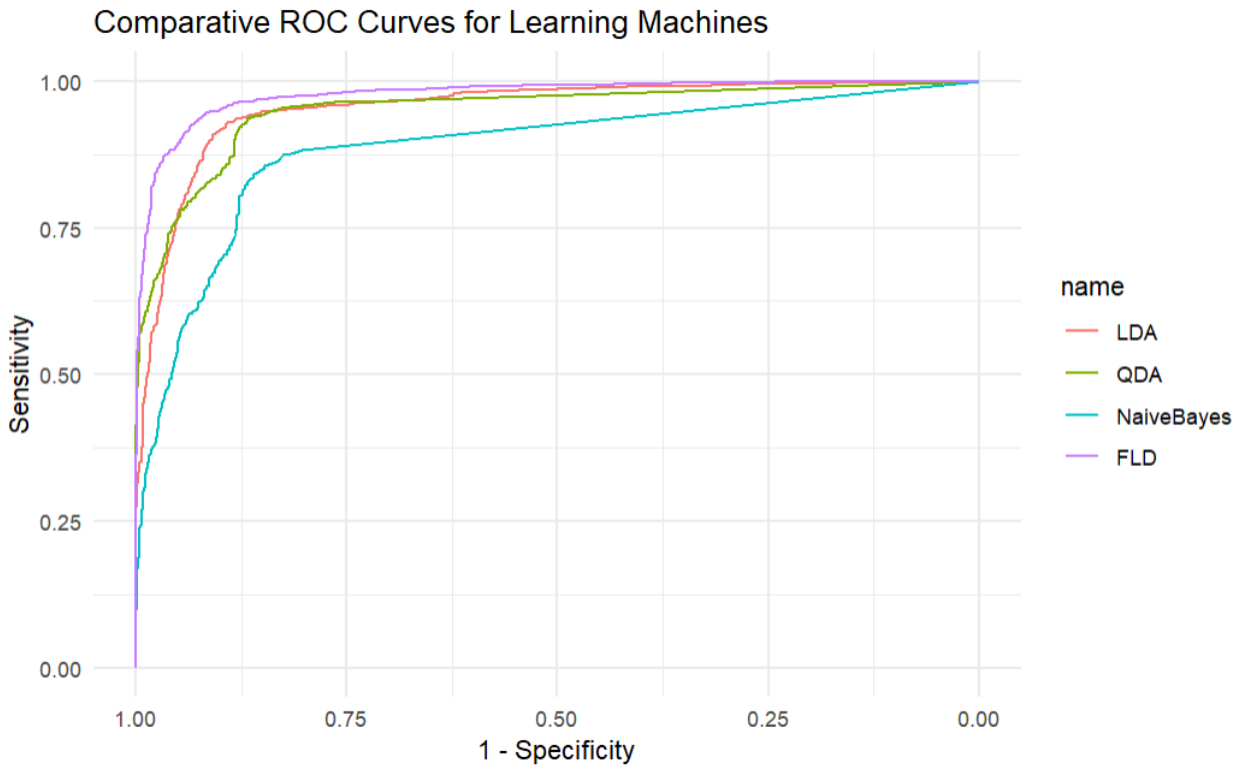
∗. The dimensionality is high relative to the sample size, which may pose challenges for some learning algorithms due to the curse of dimensionality.

## 3.Statistical Perspective on the Input Space

∗. **Input Space:** – The input space consists of numerical features representing email characteristics.
– These include word frequencies, character frequencies, and other email attributes.

∗. **Statistical Perspective:**
– Many features may be correlated, which could affect the performance of linear models.
– Dimensionality reduction techniques might be beneficial for better generalization.

## 4.Build Models and Plot Comparative ROC Curves




## 5.Comment on ROC Curves

1.**LDA (Red Curve):**

Performs the best, with the curve close to the top-left corner, indicating high accuracy.

2.**QDA (Green Curve):**

Slightly less accurate than LDA but still performs well.

3.**Naive Bayes (Cyan Curve):**

Performs the worst, with the curve further from the top-left corner, indicating lower accuracy.

4.**FLD (Purple Curve):**

Similar to LDA, showing good performance.

**Theoretical Insights:**

– LDA and FLD perform well as linear classifiers on linearly separable data.

– QDA handles complex boundaries but may overfit when unnecessary.

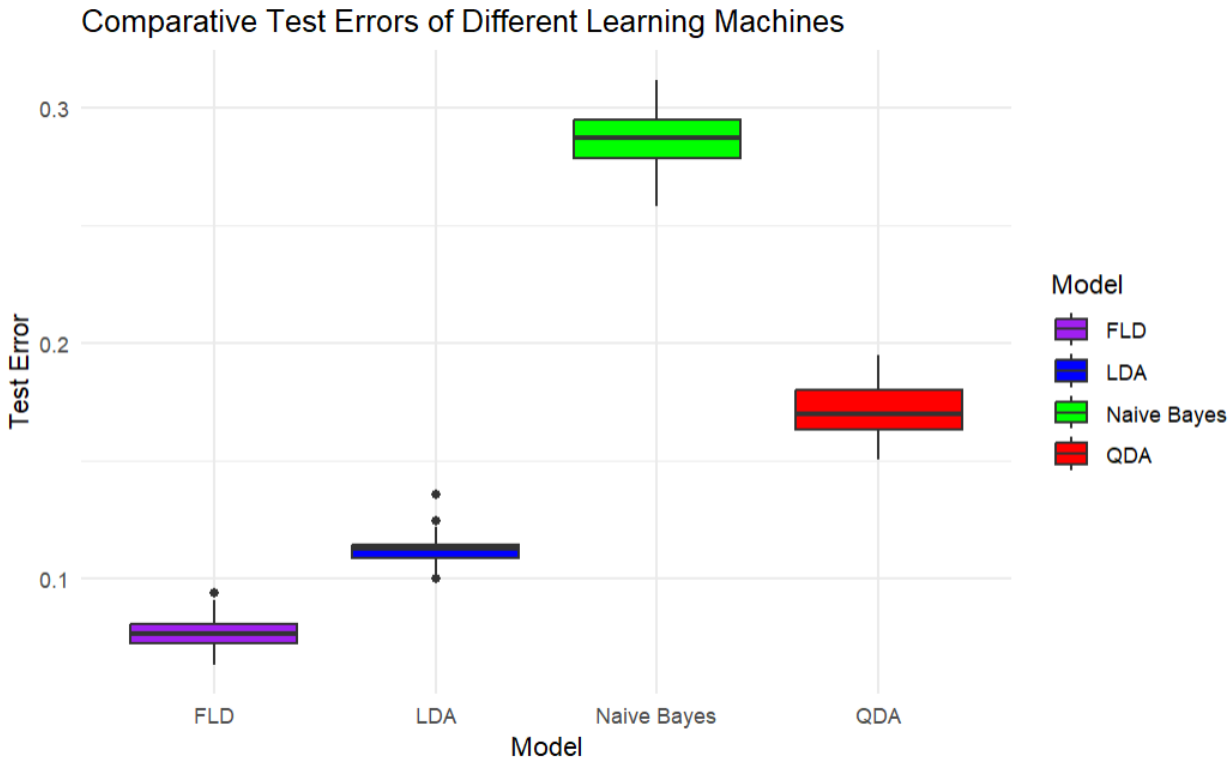
– Naive Bayes assumes feature independence, which likely doesn't hold here, leading to lower performance.

## 6.Stratified Stochastic Hold-Out Validation

```
# Display head of errors
head(errors)
```

```
##         Model      Error
## 1         LDA 0.09980431
## 2         QDA 0.16699282
## 3 Naive Bayes 0.27853881
## 4         FLD 0.06914547
## 5         LDA 0.11350294
## 6         QDA 0.16894977
```

## 7. Comparative Boxplots




## 8.Comment on the distribution of the test error

∗. The test error distribution for FLD, LDA, Naive Bayes, and QDA shows a clear pattern: as models increase in complexity, test error and variability also increase.

∗. **FLD:** The simplest model with the lowest test error and least variability, performing consistently well.

∗. **LDA:** Slightly more complex than FLD, with slightly higher test error and variability.

∗. **Naive Bayes:** Assumes feature independence, resulting in higher test error and variability, reflecting its limitations.

∗. **QDA:** The most complex model, with the highest test error and variability, likely due to overfitting.

∗. **Theoretical Insights:** Simpler models tend to generalize better, while more complex models risk overfitting, leading to higher test errors.

# YouTube Video Link

You can watch the full video presentation of this project on YouTube: **Exploring Statistical Machine Learning: Regression and Spam Classification**