Classwork 2: Regression Trees and Model Validation

Machine Learning with Tree-Based Models in R

Prof. Asc. Endri Raco, Ph.D.

November 2025

Learning Objectives

By completing this classwork, you will:

- Build regression tree models for continuous outcome prediction
- Implement and apply cross-validation for robust model assessment
- Calculate and interpret regression metrics (RMSE, MAE, R-squared)
- Understand the bias-variance tradeoff through model complexity manipulation
- Compare model performance across different validation strategies

Time Allocation: 30 minutes

Setup (5 min) | Regression Modeling (10 min) | Cross-Validation (10 min) | Analysis (5 min)

Dataset: Chocolate Bar Ratings

We analyze chocolate bar ratings to predict quality scores based on chocolate characteristics:

- `final_grade`: Quality rating (1.0 to 5.0) - **TARGET VARIABLE**
- `review_date`: Year of review
- `cocoa_percent`: Percentage of cocoa content
- `company_location`: Country where company is located
- `bean_type`: Type of cocoa bean used
- `broad_bean_origin`: Country where beans were grown

Task 1: Regression Tree Fundamentals (5 minutes)

Load the chocolate dataset and prepare it for regression modeling.

```
# Load required libraries
library(tidymodels)
library(dplyr)

# Load chocolate dataset
data(chocolate, package = "modeldata")

# Task 1a: Examine the structure and summary of the dataset


# Task 1b: Check the distribution of the target variable (final_grade)
```

```
# Create a histogram or summary statistics


# Task 1c: Create a train-test split (80% training)
# Set seed to 2025 for reproducibility


# Task 1d: Extract training and testing datasets
```

Question 1.1: What is the range and mean of the final_grade variable? Does it appear approximately normally distributed?

Your Answer:

Question 1.2: How is regression different from classification in terms of the outcome variable?

Your Answer:

Task 2: Build and Evaluate Regression Tree (10 minutes)

Create a regression tree model to predict chocolate quality ratings.

```
# Task 2a: Create a regression tree model specification
# Use decision_tree() with mode = "regression" and engine = "rpart"


# Task 2b: Fit the model predicting final_grade from cocoa_percent only
# Formula: final_grade ~ cocoa_percent


# Task 2c: Generate predictions on the test set


# Task 2d: Combine predictions with actual values
# Create a data frame with .pred and the actual final_grade


# Task 2e: Calculate regression metrics
# Use rmse(), mae(), and rsq() from yardstick
```

Question 2.1: Report the three metrics you calculated. What does each metric tell you about model performance?

Your Answer:

- RMSE:
- MAE:
- R-squared:

Question 2.2: Which metric is most interpretable in the context of chocolate ratings (1-5 scale)? Why?

Your Answer:

Task 3: Cross-Validation Implementation (10 minutes)

Implement k-fold cross-validation to get more robust performance estimates.

```
# Task 3a: Create 5-fold cross-validation splits from training data
# Use vfold_cv() with v = 5


# Task 3b: Define a more complex model using multiple predictors
# Include: cocoa_percent, review_date, company_location


# Task 3c: Create a workflow combining the model spec and formula
# Use workflow() %>% add_model() %>% add_formula()


# Task 3d: Fit the model to all CV folds
# Use fit_resamples() with your workflow and CV folds


# Task 3e: Collect and examine the cross-validated metrics
# Use collect_metrics() to see average performance
```

Question 3.1: What are the cross-validated RMSE and R-squared values? How do they compare to your single train-test split results from Task 2?

Your Answer:

Question 3.2: Why is cross-validation preferable to a single train-test split? What advantage does it provide?

Your Answer:

Task 4: Bias-Variance Tradeoff (10 minutes)

Explore how model complexity affects performance by manipulating tree depth.

```
# Task 4a: Create a shallow tree (max_depth = 2)
# This represents a high-bias, low-variance model
tree_shallow <- decision_tree(mode = "regression",
                              engine = "rpart",
                              tree_depth = 2) %>%
  fit(final_grade ~ cocoa_percent + review_date,
      data = chocolate_train)

# Task 4b: Create a deep tree (max_depth = 10)
# This represents a low-bias, high-variance model
tree_deep <- decision_tree(mode = "regression",
                           engine = "rpart",
                           tree_depth = 10) %>%
  fit(final_grade ~ cocoa_percent + review_date,
      data = chocolate_train)

# Task 4c: Calculate training RMSE for both models
```

```
# Task 4d: Calculate test RMSE for both models


# Task 4e: Compare training vs test performance
```

Question 4.1: Fill in the performance comparison table:

Model	Training RMSE	Test RMSE	Difference
Shallow (depth=2)			
Deep (depth=10)			

Question 4.2: Which model shows signs of overfitting? How can you tell from the training vs test RMSE comparison?

Your Answer:

Question 4.3: Explain the bias-variance tradeoff in your own words, using these two models as examples.

Your Answer:

Task 5: Model Selection Strategy (5 minutes)

Question 5.1: Based on your cross-validation results and bias-variance analysis, which model would you recommend for predicting chocolate ratings? Justify your choice considering both performance and generalization.

Your Answer:

Question 5.2: You are tasked with deploying a chocolate rating prediction system for a quality control application. What additional validation steps would you take before deploying the model to production?

Your Answer:

Question 5.3: The lecture mentioned that tree-based models require no normalization of numeric features. Why is this an advantage compared to other algorithms like linear regression or neural networks?

Your Answer:

Submission Instructions

1. Complete all code chunks and verify they execute without errors
2. Answer all questions with clear, concise explanations
3. Knit this document to PDF
4. Submit the PDF through the course management system

Grading Rubric (100 points total)

Component	Points	Criteria
Task 1: Data Preparation	15	Correct data loading, exploration, and splitting

Component	Points	Criteria
Task 2: Regression Modeling	20	Proper regression tree implementation and metric calculation
Task 3: Cross-Validation	25	Correct CV implementation and interpretation
Task 4: Bias-Variance Analysis	25	Accurate comparison of shallow vs deep trees with overfitting detection
Task 5: Strategic Thinking	15	Thoughtful model selection and deployment considerations
Total	**100**	

Expected Outputs Summary

By the end of this classwork, you should have:

- A trained regression tree predicting chocolate ratings
- RMSE, MAE, and R-squared metrics from both train-test split and cross-validation
- Two models (shallow and deep) demonstrating the bias-variance tradeoff
- Performance comparisons showing overfitting patterns
- Written analysis of model selection and validation strategies

---

Key Concepts Reinforced

This classwork reinforces critical concepts from the second quarter of the lecture:

- **Regression trees** handle continuous outcomes by predicting mean values in leaf nodes
- **Cross-validation** provides more reliable performance estimates than single splits
- **Bias-variance tradeoff** is fundamental to understanding model complexity
- **RMSE and MAE** measure prediction error magnitude in original units
- **Overfitting** occurs when models memorize training data rather than learning generalizable patterns

---

This classwork covers approximately 25% of the lecture content, focusing on regression trees, cross-validation, and the bias-variance tradeoff.