Palmer Penguins Part 5: Random Forest versus Linear Models

Settling the interpretability-performance tradeoff in ecological modelling

Ronald 'Ryy' G. Thomas

2025-01-05

Table of contents

Figure 1: Penguins on rocky Antarctic terrain

Palmer Penguins series, Part 5 (finale). Photo used under open licence.

> **Palmer Penguins Data Analysis Series**
>
> This is **Part 5** (finale) of a 5-part series exploring penguin morphometrics:
>
> 1. Part 1: EDA and Simple Regression
> 2. Part 2: Multiple Regression and Species Effects
> 3. Part 3: Cross-Validation and Model Comparison
> 4. Part 4: Model Diagnostics and Interpretation
> 5. **Part 5: Random Forest versus Linear Models** (this post)

1 Introduction

Whether a random forest meaningfully outperforms a linear model remains unclear until a careful head-to- head comparison is conducted. Throughout this series, the species-aware linear model has consistently delivered an R-squared of approximately 0.86, survived cross-validation with minimal degradation, and passed every diagnostic check devised. The natural question for the finale was whether a more flexible algorithm could do substantially better.

The answer is somewhat unexpected. The random forest improved R-squared by roughly two percentage points, from about 0.86 to about 0.88, despite having access to additional predictors and making no assumptions about linearity. That narrow gap prompts careful reflection on when additional model complexity is worth the cost in interpretability, reproducibility, and communication.

This post presents the full comparison, examines feature importance from both perspectives, and concludes the series with practical guidance for choosing between linear and ensemble methods in ecological research.

1.1 Motivations

- To settle the question of whether machine learning methods meaningfully outperform linear regression on this ecological dataset.
- To establish a principled framework for evaluating the interpretability-performance tradeoff, not just an intuitive judgement.
- To determine whether the random forest identifies the same important predictors as the linear model or reveals hidden structure.
- To provide a concrete example of a case where the simpler model is the better choice, as a counterpoint to the assumption that more complex methods are always superior.
- To summarise the entire five-part journey into a set of practical recommendations.
- To produce a reusable template for model comparison applicable to future datasets.

1.2 Objectives

1. Train a random forest with all available predictors and compare its cross-validated RMSE and R-squared against the species-aware linear model under identical evaluation conditions.
2. Extract and compare feature importance rankings from both models, and assess whether they align with biological expectations.
3. Develop a decision framework that maps project requirements (interpretability, sample size, regulatory context) to an appropriate model class.
4. Summarise the key findings and methodological lessons from all five parts of the series.

Errors and better approaches are welcome; see the Feedback section at the end.

Visual interlude before the technical content.

2 Prerequisites and Setup

This analysis loads the same core packages used throughout the series, with `randomForest` and `caret` providing the ensemble learning and cross-validation infrastructure.

```
library(palmerpenguins)
library(tidyverse)
library(broom)
library(caret)
library(randomForest)
library(knitr)
library(patchwork)

theme_set(theme_minimal(base_size = 12))
penguin_colors <- c(
  "Adelie" = "#FF6B6B",
  "Chinstrap" = "#4ECDC4",
  "Gentoo" = "#45B7D1"
)

data(penguins)
penguins_clean <- penguins |> drop_na()

set.seed(42)
```

We use the same random seed and cross-validation configuration as Part 3 to ensure that the results are directly comparable.

3 What is the Interpretability-Performance Tradeoff?

The interpretability-performance tradeoff is the observation that, in statistical modelling, gains in predictive accuracy often come at the expense of the analyst's ability to explain how the model arrives at its predictions. A linear regression produces a simple equation: body mass equals a weighted sum of morphometric measurements plus species offsets. Every coefficient has a direct biological interpretation, and confidence intervals quantify uncertainty in each estimate.

A random forest, by contrast, aggregates predictions from hundreds of decision trees, each built on a bootstrap sample with a random subset of predictors. The resulting predictions are often more accurate, but there is no single equation to inspect. Feature importance scores indicate which variables matter most, but they do not tell you the direction or magnitude of each effect in the way that regression coefficients do.

The tradeoff is not purely technical. In regulatory settings, grant applications, and peer-reviewed publications, reviewers and stakeholders often require models whose logic can be stated explicitly. A two-percentage-point improvement in R-squared may not justify the loss of that transparency.

4 Getting Started

4.1 Cross-Validation Configuration

We establish a single cross-validation scheme that both models will share, eliminating any advantage from different fold assignments.

```
train_control <- trainControl(
  method = "cv",
  number = 10,
  savePredictions = "final",
  verboseIter = FALSE
)
```

4.2 Training the Linear Champion

The species-aware linear model uses the same formula that has been our workhorse since Part 2.

```
cv_linear <- train(
  body_mass_g ~ bill_length_mm + bill_depth_mm +
    flipper_length_mm + species,
  data = penguins_clean,
  method = "lm",
  trControl = train_control
)
```

4.3 Training the Random Forest Challenger

The random forest receives all available predictors, including sex and island, which were not used in the linear model. This gives the ensemble every opportunity to outperform.

```
set.seed(123)

cv_rf <- train(
  body_mass_g ~ bill_length_mm + bill_depth_mm +
    flipper_length_mm + species + sex + island,
  data = penguins_clean,
  method = "rf",
  trControl = train_control,
  ntree = 500,
  importance = TRUE
)
```

The `importance = TRUE` flag instructs the algorithm to compute permutation-based variable importance, which we examine in the feature importance section below.

Figure 2: Penguins gathered near the shore

Midpoint pause before the comparison results.

5 Head-to-Head Comparison

5.1 Cross-Validated Performance

```
comparison_table <- tibble(
  Model = c("Linear (species)", "Random Forest"),
  RMSE = c(
    cv_linear$results$RMSE,
    min(cv_rf$results$RMSE)
  ),
  R_squared = c(
    cv_linear$results$Rsquared,
    max(cv_rf$results$Rsquared)
  ),
  MAE = c(
    cv_linear$results$MAE,
    min(cv_rf$results$MAE)
  )
)

kable(
  comparison_table,
  digits = 3,
  col.names = c(
    "Model", "RMSE (g)", "R-squared", "MAE (g)"
  ),
  caption = "Cross-validated performance metrics
    for the linear model and random forest."
)
```

The random forest achieves a modestly higher R-squared and lower RMSE, but the differences are small. In absolute terms, the RMSE improvement amounts to a reduction of roughly 20-30 grams in prediction error, a quantity that is likely within the range of field measurement uncertainty.

5.2 Fold-by-Fold Variability

```
fold_results <- bind_rows(
  cv_linear$resample |>
    mutate(Model = "Linear"),
  cv_rf$resample |>
    mutate(Model = "Random Forest")
)
```

```
ggplot(
  fold_results,
  aes(x = Model, y = Rsquared, fill = Model)
) +
  geom_boxplot(alpha = 0.7) +
  scale_fill_manual(
    values = c(
      "Linear" = "#FF6B6B",
      "Random Forest" = "#45B7D1"
    )
  ) +
  labs(
    title = "R-squared Distribution Across Folds",
    x = NULL,
    y = "R-squared"
  ) +
  theme_minimal() +
  theme(legend.position = "none")
```

The boxplots show substantial overlap between the two models' per-fold R-squared distributions, confirming that the performance difference is not statistically compelling for this sample size.

6 Feature Importance Analysis

6.1 Random Forest Importance

```
rf_imp <- varImp(cv_rf)$importance |>
  rownames_to_column("Variable") |>
  arrange(desc(Overall)) |>
  head(6)

kable(
  rf_imp,
  digits = 1,
  col.names = c("Variable", "Importance"),
  caption = "Variable importance scores from the
    random forest (permutation-based)."
)
```

6.2 Linear Model Coefficients

```
linear_coefs <- tidy(
  cv_linear$finalModel, conf.int = TRUE
) |>
  filter(term != "(Intercept)") |>
  mutate(
    abs_effect = abs(estimate)
  ) |>
  arrange(desc(abs_effect)) |>
  select(term, estimate, conf.low, conf.high)

kable(
  linear_coefs,
  digits = 1,
  col.names = c(
    "Term", "Estimate",
    "95% CI Lower", "95% CI Upper"
  ),
  caption = "Linear model coefficients ranked by
    absolute effect size."
)
```

Both models agree that flipper length and species are the most important determinants of body mass. The linear model additionally provides the direction and magnitude of each effect (e.g., each millimetre of flipper length adds approximately 15-20 grams), which the random forest importance scores alone cannot convey.

7 Model Selection Guidelines

The following framework maps common project requirements to a recommended model class.

```
guidelines <- tibble(
  Requirement = c(
    "Regulatory or peer review",
    "Small sample (n < 500)",
    "Coefficient interpretation needed",
    "Maximum predictive accuracy",
    "Complex feature interactions",
    "Quick prototyping"
  ),
  Recommendation = c(
    "Linear model",
    "Linear model",
    "Linear model",
```

```
    "Random forest",
    "Random forest",
    "Linear model"
  ),
  Rationale = c(
    "Transparent, auditable logic",
    "Fewer parameters, lower variance",
    "Direct effect estimates with CIs",
    "Flexible, captures non-linearity",
    "Automatic interaction detection",
    "Fast fitting, minimal tuning"
  )
)

kable(
  guidelines,
  col.names = c(
    "Requirement", "Recommendation",
    "Rationale"
  ),
  caption = "Model selection guidelines based on
    project requirements."
)
```

For the Palmer Penguins dataset, the linear model satisfies the first three criteria and falls only marginally short on predictive accuracy. The random forest is preferred only when the application demands maximum predictive performance and the analyst is willing to forgo direct coefficient interpretation.

8 Practical Prediction Comparison

```
new_penguin <- data.frame(
  species = "Gentoo",
  flipper_length_mm = 220,
  bill_length_mm = 47,
  bill_depth_mm = 15,
  sex = "male",
  island = "Biscoe"
)

linear_pred <- predict(
  cv_linear, newdata = new_penguin
)
rf_pred <- predict(
  cv_rf, newdata = new_penguin
```

```
)

pred_comparison <- tibble(
  Model = c("Linear", "Random Forest"),
  Predicted_mass_g = c(linear_pred, rf_pred),
  Difference_g = c(
    NA, rf_pred - linear_pred
  )
)

kable(
  pred_comparison,
  digits = 0,
  col.names = c(
    "Model", "Predicted Mass (g)", "Difference (g)"
  ),
  caption = "Predicted body mass for a hypothetical
    male Gentoo penguin from Biscoe Island."
)
```

The two models produce predictions that differ by a modest amount, well within the typical measurement uncertainty for field-weighed penguins.

9 Series Summary

This five-part series traced the full modelling workflow from exploratory data analysis to model selection.

```
series_summary <- tibble(
  Part = paste("Part", 1:5),
  Topic = c(
    "EDA and Simple Regression",
    "Multiple Regression and Species",
    "Cross-Validation and Model Comparison",
    "Model Diagnostics and Interpretation",
    "Random Forest versus Linear Models"
  ),
  Key_result = c(
    "R-squared = 0.759 (flipper only)",
    "R-squared = 0.863 (added species)",
    "CV confirms robust generalisation",
    "All four assumptions satisfied",
    "RF gains only ~2 pp over linear"
  )
)
```

```r
kable(
  series_summary,
  col.names = c("Part", "Topic", "Key Result"),
  caption = "Summary of findings across all five
    parts of the Palmer Penguins series."
)
```

The central narrative is one of diminishing returns from complexity. The largest performance gain came from incorporating biological context (species) in Part 2. Every subsequent addition (polynomial terms, random forests, additional predictors) yielded only marginal improvements.

9.1 Things to Watch Out For

1. **Unfair comparisons.** Giving one model access to additional predictors that the other lacks introduces a confound. We did this deliberately to give the random forest every advantage, but in a formal comparison both models should use the same predictor set.
2. **Overstating small differences.** A two- percentage-point improvement in R-squared may be statistically indistinguishable from zero given the sample size. Always check fold-by-fold variability before drawing conclusions.
3. **Feature importance is not causation.** The random forest's variable importance ranking reflects predictive utility, not causal effect. Flipper length predicts body mass well, but lengthening a penguin's flippers will not make it heavier.
4. **Reproducibility of random forests.** Different random seeds produce different forests and therefore slightly different importance rankings. Always set a seed and report it.
5. **Communication burden.** A linear model can be explained in a single equation; a random forest requires diagrams, importance plots, and partial dependence plots. Consider the audience before choosing the model.

Figure 3: Antarctic landscape with penguins in the distance

Closing visual before the summary sections.

9.2 Lessons Learnt

9.2.1 Conceptual Understanding

- The random forest outperformed the linear model by approximately two percentage points of R-squared, a difference that is modest relative to the loss of interpretability.
- Both models identified flipper length and species as the dominant predictors, confirming the biological plausibility of the linear model's structure.
- The largest single improvement in the entire series came from adding species to the model in Part 2 (R-squared from 0.76 to 0.86), not from adopting a more complex algorithm.
- For well-behaved ecological data with moderate sample sizes, linear models with appropriate biological covariates can match or closely approximate ensemble methods.

9.2.2 Technical Skills

- The `caret` package enables fair model comparison by enforcing identical cross-validation folds across different algorithms via `trainControl`.
- `varImp()` provides a unified interface for extracting variable importance from both linear and ensemble models, facilitating direct comparison.
- Setting `importance = TRUE` in `randomForest` computes permutation-based importance, which is more reliable than impurity-based measures for correlated predictors.
- The `broom` package's `tidy()` and `glance()` functions make it straightforward to extract publication-ready coefficient tables from linear models.

9.2.3 Gotchas and Pitfalls

- The `caret` package's `train()` function defaults to tuning the `mtry` hyperparameter for random forests, which can produce confusing output if the analyst expects a single set of results.
- Random forest predictions from `caret` objects require that the new data include all predictor columns, including factor levels present in the training data.
- Large `ntree` values (e.g., 500 or 1000) improve stability but increase computation time; 500 trees is generally sufficient for datasets of this size.
- The `caret` package loads many dependencies; in a reproducible pipeline, pin all package versions with `renv`.

9.3 Limitations

- The random forest was given two additional predictors (sex and island) that the linear model did not use, making the comparison deliberately generous to the ensemble method.
- The dataset contains only 333 complete observations from a single geographic location; results may not generalise to larger or more diverse penguin populations.
- The temporal window (2007-2009) is narrow; both models assume stationary relationships that may shift under changing environmental conditions.

- We did not tune random forest hyperparameters beyond the default `mtry` grid, which means the ensemble's performance ceiling may be slightly higher.
- Partial dependence plots and interaction effects were not examined; doing so would provide a richer comparison of the two model classes.
- The analysis does not account for potential measurement error in the morphometric variables.

9.4 Opportunities for Improvement

1. Conduct a fair comparison in which both models use the same predictor set (including sex and island in the linear model).
2. Tune random forest hyperparameters (mtry, ntree, nodesize) using nested cross-validation.
3. Add gradient-boosted trees (e.g., `xgboost`) as a third competitor to broaden the comparison.
4. Generate partial dependence plots for the random forest to visualise the shape of each variable's marginal effect.
5. Fit a mixed-effects model with species as a random effect to account for population-level variation more formally.
6. Extend the comparison to classification tasks (e.g., predicting species from morphometrics) to test whether the tradeoff differs for categorical outcomes.

10 Wrapping Up

This series began with a simple scatter plot of flipper length versus body mass and ended with a head-to-head comparison between a linear model and a random forest. The central finding is that, for the Palmer Penguins dataset, the species-aware linear model provides an excellent balance of performance and interpretability. The random forest gains only about two percentage points of R-squared while sacrificing the transparency that makes linear models so valuable in scientific communication.

The principal lesson from this exercise is that model selection is not purely a technical decision. The right model depends on the audience, the regulatory context, and the relative importance of explanation versus prediction. For peer-reviewed ecological research, a well-validated linear model with clear coefficient interpretations will almost always be preferred over an opaque ensemble that offers marginal predictive gains.

For anyone beginning a modelling journey, three pieces of advice are offered. First, start simple and add complexity only when the data demand it. Second, always cross-validate before claiming that a model generalises. Third, run diagnostics to confirm that the model's assumptions are satisfied; good predictions from a misspecified model are a coincidence, not a guarantee.

Main takeaways:

- The random forest outperformed the linear model by only about two percentage points of R-squared.
- Both models agree on the importance of flipper length and species.
- The largest gain came from biological context (species), not algorithmic sophistication.
- For ecological data of this type, the linear model is the recommended default.

11 See Also

Series posts:

- Part 1: EDA and Simple Regression
- Part 2: Multiple Regression and Species Effects
- Part 3: Cross-Validation and Model Comparison
- Part 4: Model Diagnostics and Interpretation

Key resources:

- Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning* (2nd ed.). Springer. https://www.statlearning.com
- Horst, A. M., Hill, A. P., & Gorman, K. B. (2020). palmerpenguins: Palmer Archipelago (Antarctica) penguin data. R package. https://allisonhorst.github.io/palmerpenguins/
- Kuhn, M. (2008). Building predictive models in R using the caret package. *Journal of Statistical Software*, 28(5).
- Molnar, C. (2022). *Interpretable Machine Learning* (2nd ed.). https://christophm.github.io/interpretable-ml-book/

12 Reproducibility

Data source: `palmerpenguins` R package (built-in dataset, no external download required).

Pipeline commands:

```
Rscript analysis/scripts/01_prepare_data.R
Rscript analysis/scripts/02_fit_models.R
Rscript analysis/scripts/03_generate_figures.R
quarto render index.qmd
```

Session information:

13 Feedback

Corrections, suggestions, and questions are welcome. Please open an issue or pull request on the GitHub repository or send an email to user@example.com.