

STAT 8020 R Session 2: Multiple Linear Regression I

Whitney Huang, Clemson University

Contents

Species diversity on the Galapagos Islands	1
Step 1: Load the Data	1
Explore Pairwise Relationships	2
Correlation Matrix	3
Model 1: Simple Linear Regression	3
Model 2: Multiple Linear Regression (Adding <i>Area</i>)	4
Model 3: Adding <i>Adjacent</i>	7
Full Model	8
Parameter Estimation	9
ANOVA	10
Simulation	10
Comparing R^2 and Adjusted R^2	10
Results: Behavior of R^2	11
Results: Behavior of Adjusted R^2	12

Species diversity on the Galapagos Islands

Step 1: Load the Data

```
# Install the package if not already installed (run once)
install.packages("faraway")

# Load the faraway package, which contains the dataset
library(faraway)

# Load the Galápagos Islands dataset
data(gala)

# Preview the first few rows to understand the structure
head(gala)
```

```
##           Species Endemics Area Elevation Nearest Scruz Adjacent
## Baltra      58      23 25.09      346      0.6  0.6      1.84
## Bartolome   31      21  1.24      109      0.6 26.3     572.33
## Caldwell     3       3  0.21      114      2.8 58.7      0.78
## Champion    25       9  0.10       46      1.9 47.4      0.18
## Coamano      2       1  0.05       77      1.9  1.9     903.82
## Daphne.Major 18      11  0.34      119      8.0  8.0      1.84
```

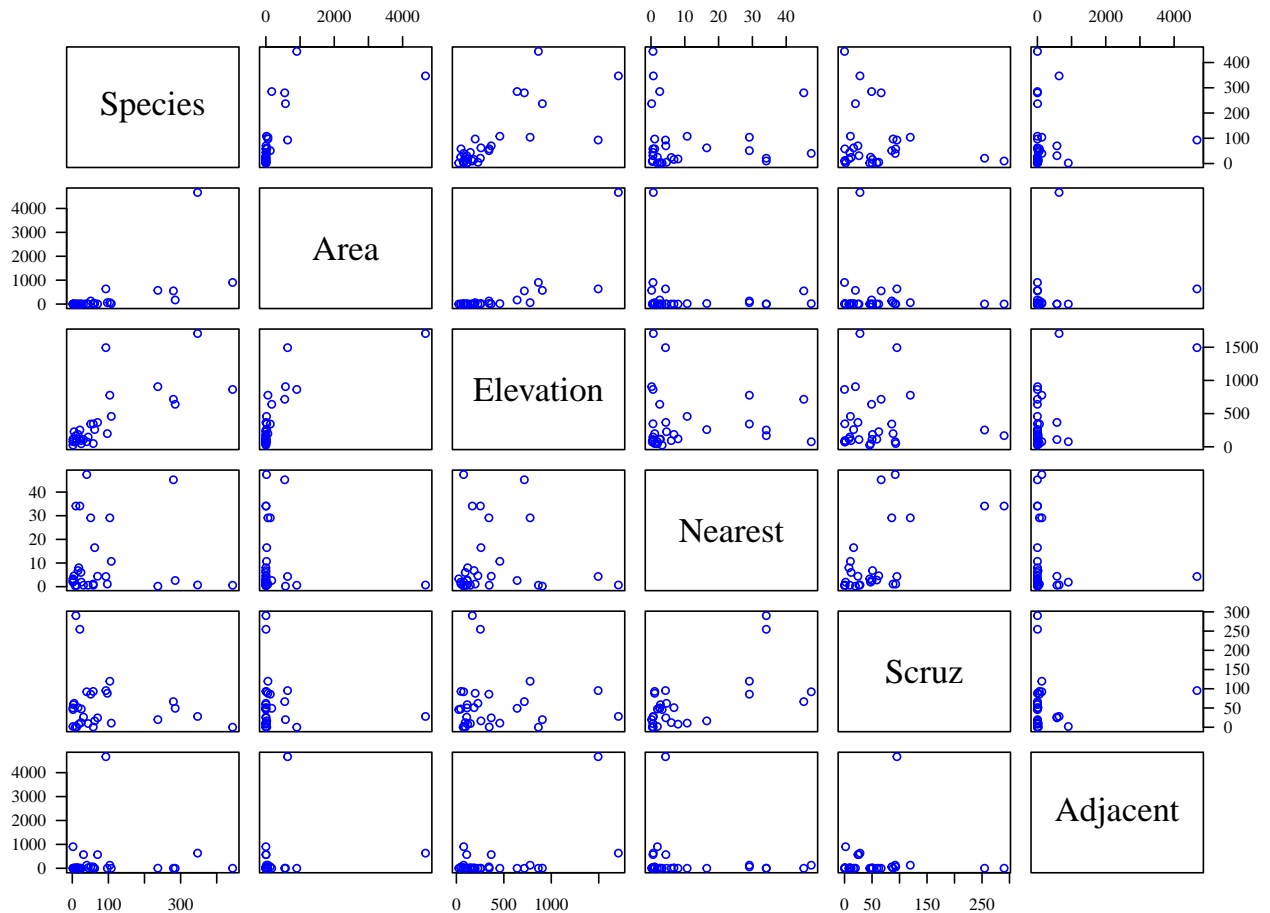
Notes:

- The `gala` dataset contains information on species diversity across islands
- Each row represents an island, with variables describing ecological and geographical characteristics
- Use `head()` to quickly inspect variable names and data format before analysis

Explore Pairwise Relationships

Visualize pairwise relationships between variables to identify trends, correlations, and potential multi-collinearity.

```
par(las = 1, mar = c(3.5, 3.5, 1, 0.5), mgp = c(2.5, 1, 0), family = "serif")
# Create pairwise scatterplots (excluding the 2nd column)
# Helps detect linear relationships and possible correlations
pairs(gala[, -2], cex = 0.95, col = "blue")
```



Correlation Matrix

Compute correlations to quantify the strength and direction of linear relationships between variables.

```
# Compute correlation matrix
cor(gala[, -2])
```

```
##           Species      Area  Elevation  Nearest  Scruz
## Species    1.0000000  0.6178431  0.73848666 -0.01409407 -0.17114244
## Area       0.61784307  1.0000000  0.75373492 -0.11110320 -0.10078493
## Elevation  0.73848666  0.7537349  1.00000000 -0.01107698 -0.01543829
## Nearest   -0.01409407 -0.1111032 -0.01107698  1.00000000  0.61541036
## Scruz     -0.17114244 -0.1007849 -0.01543829  0.61541036  1.00000000
## Adjacent  0.02616635  0.1800376  0.53645782 -0.11624788  0.05166066
##           Adjacent
## Species    0.02616635
## Area       0.18003759
## Elevation  0.53645782
## Nearest   -0.11624788
## Scruz     0.05166066
## Adjacent  1.00000000
```

Model 1: Simple Linear Regression

Fit a simple linear regression using *Elevation* as the predictor, as it shows the strongest correlation with *Species*.

```
# Fit simple linear regression model
M1 <- lm(Species ~ Elevation, data = gala)

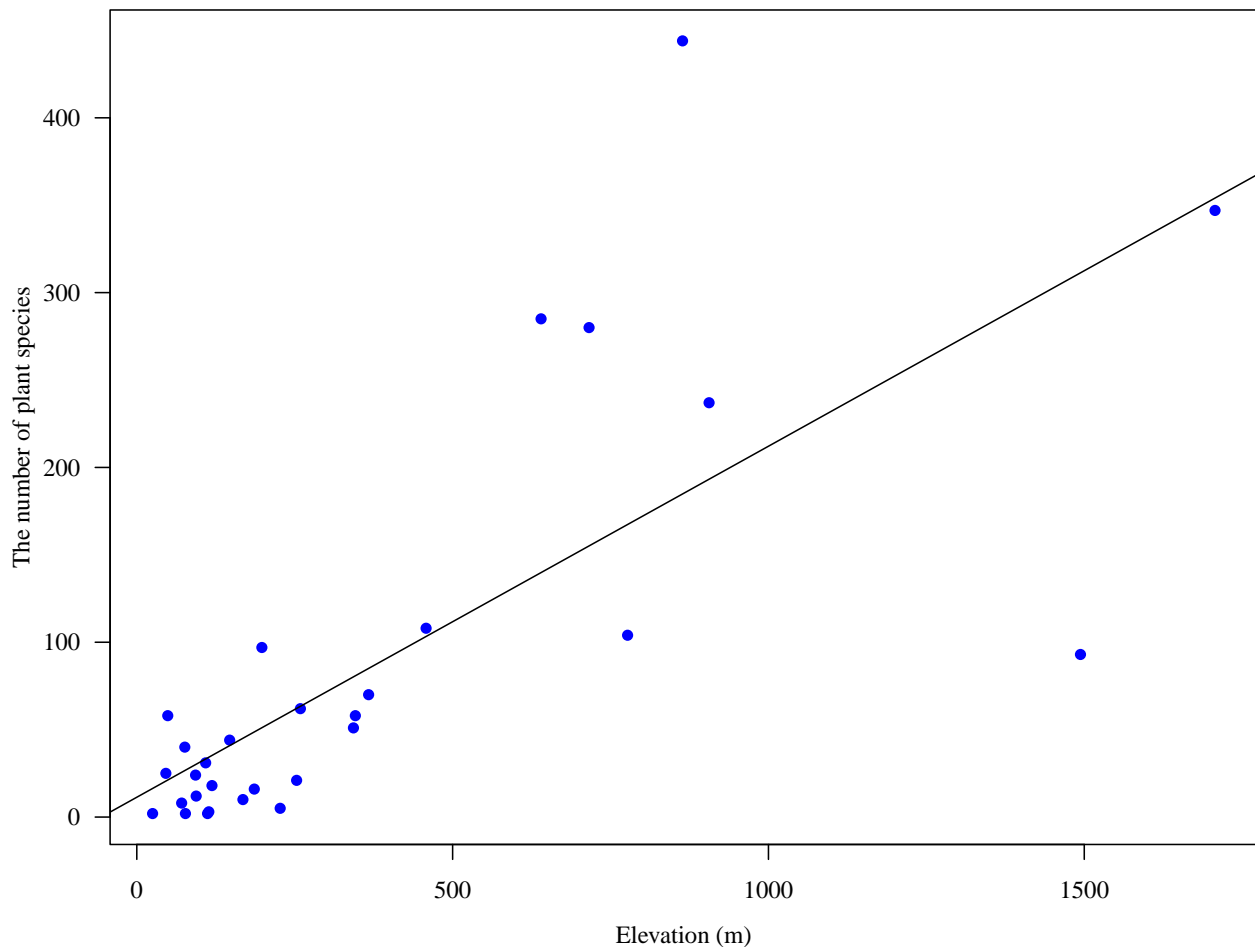
# Display model summary (coefficients, significance, R-squared)
summary(M1)
```

```
##
## Call:
## lm(formula = Species ~ Elevation, data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -218.319  -30.721  -14.690    4.634   259.180
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.33511   19.20529    0.590    0.56
## Elevation     0.20079    0.03465    5.795 3.18e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 78.66 on 28 degrees of freedom
## Multiple R-squared:  0.5454, Adjusted R-squared:  0.5291
## F-statistic: 33.59 on 1 and 28 DF,  p-value: 3.177e-06
```

```

par(las = 1, mar = c(3.5, 3.5, 1, 0.5), mgp = c(2.5, 1, 0), family = "serif")
plot(gala$Elevation, gala$Species, xlab = "Elevation (m)",
     ylab = "The number of plant species",
     pch = 16, col = "blue")
abline(M1)

```



Model 2: Multiple Linear Regression (Adding Area)

Extend the model by including *Area* to assess whether it improves prediction.

```

# Fit multiple linear regression model
M2 <- lm(Species ~ Elevation + Area, data = gala)
summary(M2)

```

```

##
## Call:
## lm(formula = Species ~ Elevation + Area, data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -192.619  -33.534  -19.199    7.541  261.514
##

```

```

## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept) 17.10519   20.94211   0.817  0.42120
## Elevation   0.17174    0.05317   3.230  0.00325 **
## Area        0.01880    0.02594   0.725  0.47478
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 79.34 on 27 degrees of freedom
## Multiple R-squared:  0.554, Adjusted R-squared:  0.521
## F-statistic: 16.77 on 2 and 27 DF,  p-value: 1.843e-05

# Visualize predicted response surface with original data points
library(fields)

# Create grids for Elevation and Area
Elevation_grid <- seq(0, 1750, 10)
Area_grid <- seq(0, 4700, 10)

temp <- expand.grid(Elevation = Elevation_grid, Area = Area_grid)
x_new <- data.frame(Elevation = temp$Elevation, Area = temp$Area)

# Predicted response surface
y_pred <- matrix(predict(M2, newdata = x_new),
  nrow = length(Elevation_grid),
  ncol = length(Area_grid))

# Define a common color scale for both surface and observed Species values
zlim_common <- range(c(y_pred, gala$Species), na.rm = TRUE)

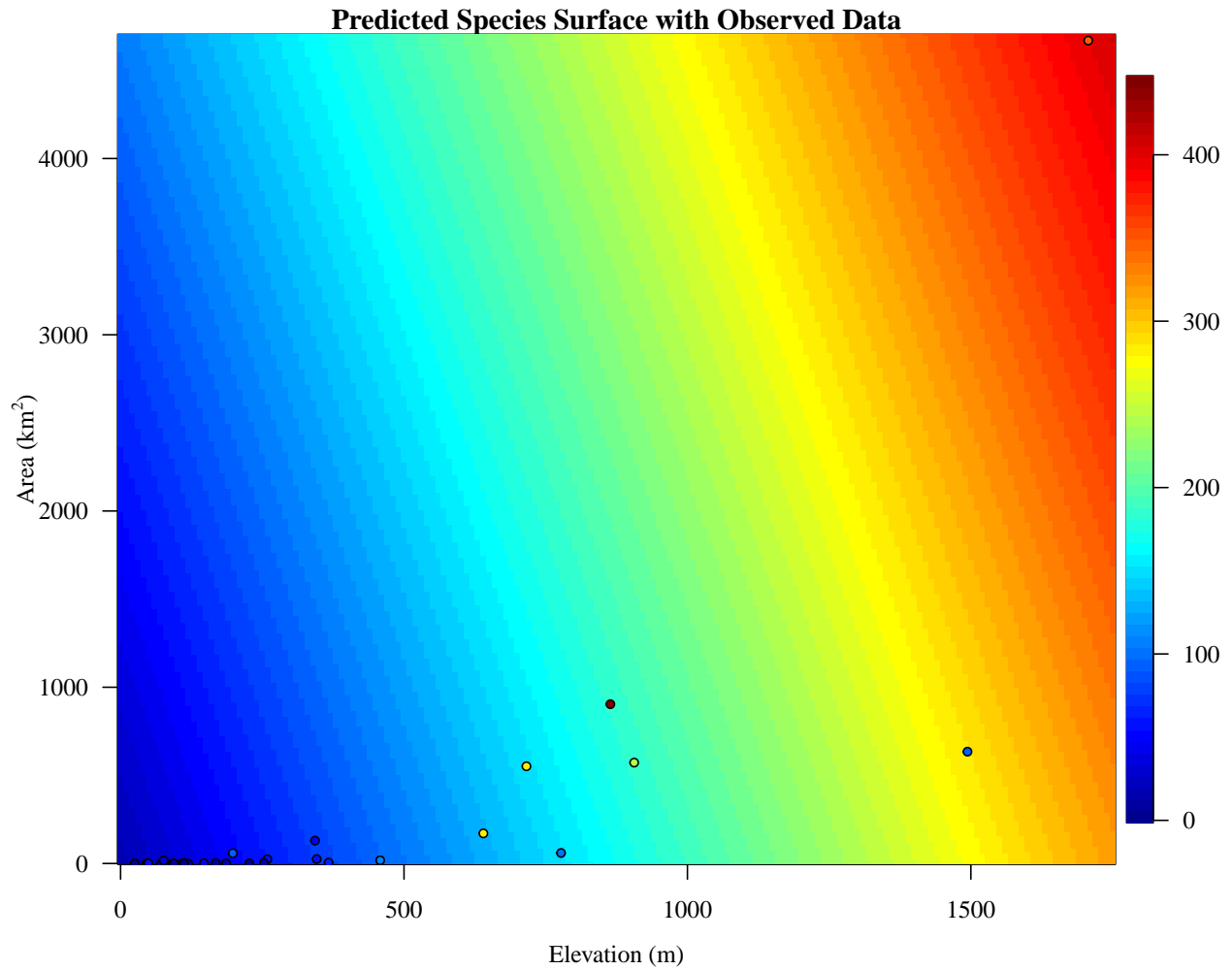
# Choose colors
cols <- tim.colors(64)

# Plot predicted surface
par(las = 1, mar = c(3.5, 3.75, 1, 0.5), mgp = c(2.5, 1, 0), family = "serif")
image.plot(Elevation_grid, Area_grid, y_pred,
  col = cols, zlim = zlim_common,
  xlab = "Elevation (m)",
  ylab = expression(paste("Area (", km^2, ")")),
  main = "Predicted Species Surface with Observed Data")

# Add observed data points using the same color scale
point_cols <- cols[
  cut(gala$Species, breaks = seq(zlim_common[1], zlim_common[2], length.out = length(cols) + 1),
    include.lowest = TRUE)]

points(gala$Elevation, gala$Area, pch = 21,
  bg = point_cols, col = "black", cex = 0.75)

```



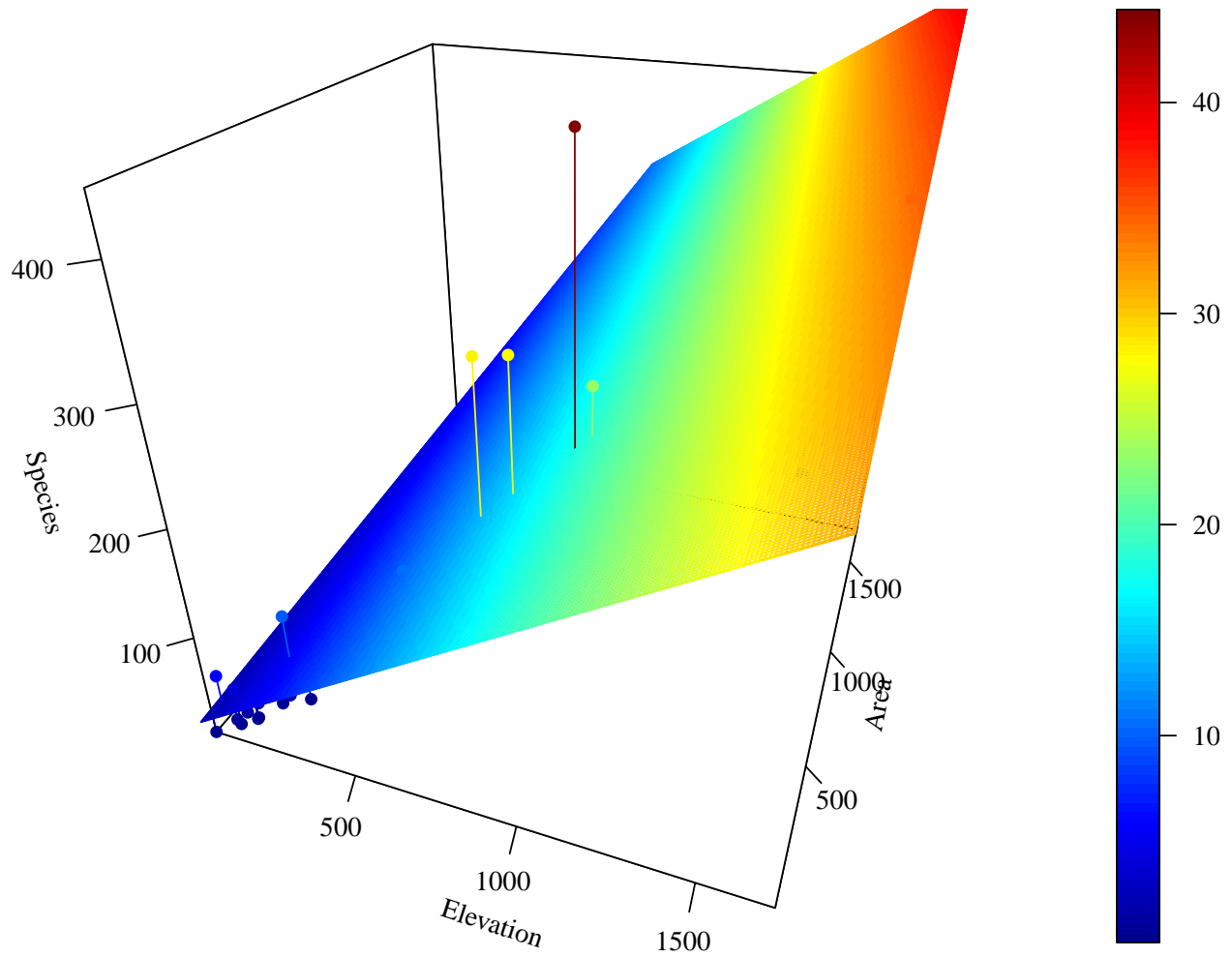
```

# 3D visualization of regression surface
library(plot3D)

# Predicted values at observed data points
fitpoints <- predict(M2)

# scatter plot with regression plane
par(las = 1, mar = c(3.5, 3.75, 1, 0.5), mgp = c(2, 1, 0), family = "serif")
scatter3D(gala$Elevation, gala$Elevation, gala$Species, pch = 16, cex = 1,
  theta = 20, phi = 30, ticktype = "detailed",
  xlab = "Elevation", ylab = "Area", zlab = "Species",
  surf = list(x = Elevation_grid, y = Area_grid, z = y_pred, facets = NA, fit = fitpoints))

```



Model 3: Adding *Adjacent*

Extend the model by including *Adjacent* to examine whether spatial context improves prediction of species diversity.

```
# Fit multiple linear regression model with an additional predictor (Adjacent)
M3 <- lm(Species ~ Elevation + Area + Adjacent, data = gala)
summary(M3)
```

```
##
## Call:
## lm(formula = Species ~ Elevation + Area + Adjacent, data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -124.064  -34.283   -8.733   27.972  195.973
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5.71893   16.90706  -0.338  0.73789
## Elevation     0.31498    0.05211   6.044  2.2e-06 ***
```

```
## Area      -0.02031    0.02181  -0.931  0.36034
## Adjacent  -0.07528    0.01698  -4.434  0.00015 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 61.01 on 26 degrees of freedom
## Multiple R-squared:  0.746, Adjusted R-squared:  0.7167
## F-statistic: 25.46 on 3 and 26 DF,  p-value: 6.683e-08
```

Full Model

Fit a full model including all available predictors to assess their combined effect on species diversity.

```
# Fit full multiple linear regression model
M4 <- lm(Species ~ Elevation + Area + Adjacent + Nearest + Scruz, data = gala)
summary(M4)
```

```
##
## Call:
## lm(formula = Species ~ Elevation + Area + Adjacent + Nearest +
##     Scruz, data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -111.679  -34.898   -7.862   33.460  182.584
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.068221  19.154198   0.369  0.715351
## Elevation    0.319465   0.053663   5.953 3.82e-06 ***
## Area        -0.023938   0.022422  -1.068 0.296318
## Adjacent    -0.074805   0.017700  -4.226 0.000297 ***
## Nearest     0.009144   1.054136   0.009 0.993151
## Scruz       -0.240524   0.215402  -1.117 0.275208
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.98 on 24 degrees of freedom
## Multiple R-squared:  0.7658, Adjusted R-squared:  0.7171
## F-statistic: 15.7 on 5 and 24 DF,  p-value: 6.838e-07
```

```
# Generate fitted (predicted) values for observed data
predict(M4)
```

```
##      Baltra      Bartolome      Caldwell      Champion      Coamano Daphne.Major
## 116.7259460 -7.2731544 29.3306594 10.3642660 -36.3839155 43.0877052
## Daphne.Minor      Darwin      Eden      Enderby      Espanola      Fernandina
## 33.9196678 -9.0189919 28.3142017 30.7859425 47.6564865 96.9895982
## Gardner1      Gardner2      Genovesa      Isabela      Marchena      Onslow
## -4.0332759 64.6337956 -0.4971756 386.4035578 88.6945404 4.0372328
## Pinta      Pinzon      Las.Plazas      Rabida SanCristobal      SanSalvador
## 215.6794862 150.4753750 35.0758066 75.5531221 206.9518779 277.6763183
```

```
##      SantaCruz      SantaFe      SantaMaria      Seymour      Tortuga      Wolf
## 261.4164131    85.3764857    195.6166286    49.8050946    52.9357316    26.7005735
```

```
# Provides uncertainty estimates for each parameter
confint(M4)
```

```
##              2.5 %      97.5 %
## (Intercept) -32.4641006 46.60054205
## Elevation    0.2087102  0.43021935
## Area         -0.0702158  0.02233912
## Adjacent     -0.1113362 -0.03827344
## Nearest      -2.1664857  2.18477363
## Scruz        -0.6850926  0.20404416
```

Parameter Estimation

Compute regression coefficients and fitted values using matrix algebra.
This demonstrates how the least squares estimator is obtained explicitly.

```
# Construct design matrix X (includes intercept and predictors)
X <- model.matrix(M4)
```

```
# Response variable
y <- gala$Species
```

```
# Estimate regression coefficients using normal equations:
# beta_hat = (X'X)^{-1} X'y
(beta_hat <- solve(t(X) %*% X) %*% t(X) %*% y)
```

```
##              [,1]
## (Intercept) 7.068220709
## Elevation   0.319464761
## Area       -0.023938338
## Adjacent   -0.074804832
## Nearest    0.009143961
## Scruz     -0.240524230
```

```
# Faster and numerically stable computation using crossprod()
# crossprod(X) = X'X, crossprod(X, y) = X'y
beta_hat_faster <- solve(crossprod(X), crossprod(X, y))
```

```
# Compute fitted values:
# y_hat = X * beta_hat
(y_hat <- X %*% solve(t(X) %*% X) %*% t(X) %*% y)
```

```
##              [,1]
## Baltra      116.7259460
## Bartolome   -7.2731544
## Caldwell    29.3306594
## Champion    10.3642660
## Coamano     -36.3839155
```

```
## Daphne.Major 43.0877052
## Daphne.Minor 33.9196678
## Darwin -9.0189919
## Eden 28.3142017
## Enderby 30.7859425
## Espanola 47.6564865
## Fernandina 96.9895982
## Gardner1 -4.0332759
## Gardner2 64.6337956
## Genovesa -0.4971756
## Isabela 386.4035578
## Marchena 88.6945404
## Onslow 4.0372328
## Pinta 215.6794862
## Pinzon 150.4753750
## Las.Plazas 35.0758066
## Rabida 75.5531221
## SanCristobal 206.9518779
## SanSalvador 277.6763183
## SantaCruz 261.4164131
## SantaFe 85.3764857
## SantaMaria 195.6166286
## Seymour 49.8050946
## Tortuga 52.9357316
## Wolf 26.7005735
```

ANOVA

Use ANOVA to assess the overall significance of the model and the contribution of each predictor.

```
# Perform analysis of variance (ANOVA) for the fitted model
# Tests whether predictors significantly improve the model fit
anova(M4)
```

```
## Analysis of Variance Table
##
## Response: Species
##          Df Sum Sq Mean Sq F value    Pr(>F)
## Elevation 1 207828 207828 55.8981 1.023e-07 ***
## Area      1   3307    3307  0.8895 0.3550197
## Adjacent  1  73171   73171 19.6804 0.0001742 ***
## Nearest   1   2909    2909  0.7823 0.3852165
## Scruz     1   4636    4636  1.2469 0.2752082
## Residuals 24  89231    3718
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Simulation

Comparing R^2 and Adjusted R^2

In this simulation, we investigate how R^2 and adjusted R^2 behave when adding an **irrelevant predictor**.

Key idea:

- R^2 always increases (or stays the same) when adding predictors
- Adjusted R^2 may decrease if the new predictor does not improve the model

We simulate many datasets to observe this behavior empirically.

```
# Set seed for reproducibility (ensures same results each time)
set.seed(123)

# Number of simulated datasets
N = 500

# Generate predictor variables:
# x1 → relevant predictor
# x2 → irrelevant (noise) predictor
x1 <- replicate(N, rnorm(30))
x2 <- replicate(N, rnorm(30))

# Generate response variable:
# y depends ONLY on x1 (x2 has no true effect)
y1 <- apply(x1, 2, function(x) 5 + 2 * x + rnorm(30, 0, 1))
```

```
# Store R^2 and adjusted R^2 values
R.sq <- array(dim = c(N, 4))

# Loop over simulations
for (i in 1:N){

  # Model 1: uses only relevant predictor x1
  fit1 <- lm(y1[, i] ~ x1[, i])

  # Model 2: adds irrelevant predictor x2
  fit2 <- lm(y1[, i] ~ x1[, i] + x2[, i])

  # Extract R^2 and adjusted R^2
  R.sq[i, 1] = summary(fit1)$r.squared
  R.sq[i, 2] = summary(fit1)$adj.r.squared
  R.sq[i, 3] = summary(fit2)$r.squared
  R.sq[i, 4] = summary(fit2)$adj.r.squared
}
```

Results: Behavior of R^2

```
par(las = 1, mfrow = c(1, 2), mar = c(3.5, 3.75, 1, 0.5), mgp = c(2, 1, 0), family = "serif")

# Compare R^2 values between models
plot(R.sq[, 1], R.sq[, 3],
     pch = 16, cex = 0.65, col = "blue",
     xlab = expression(paste("Model 1: ", R^2)),
     ylab = expression(paste("Model 2: ", R^2)))

# Reference line: equality
```

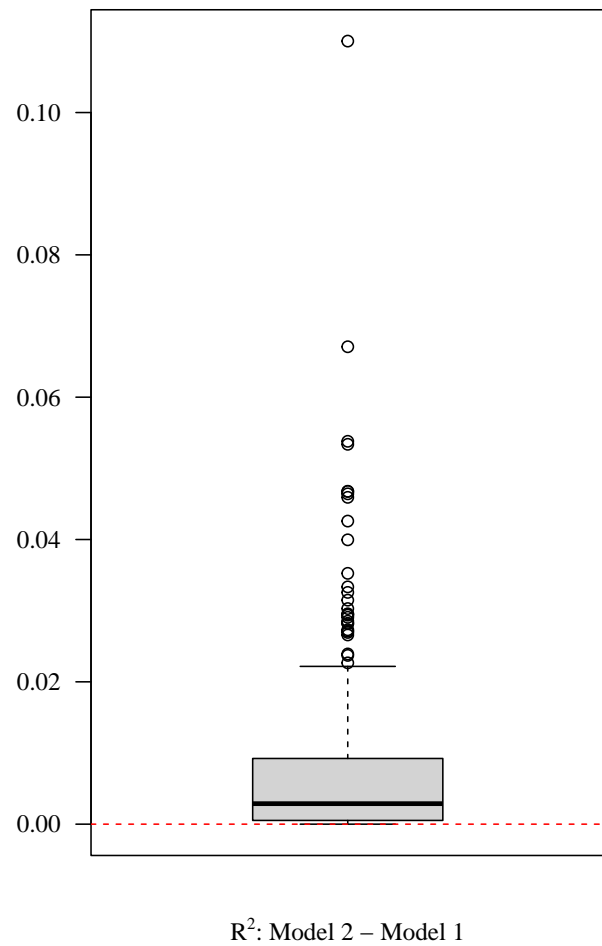
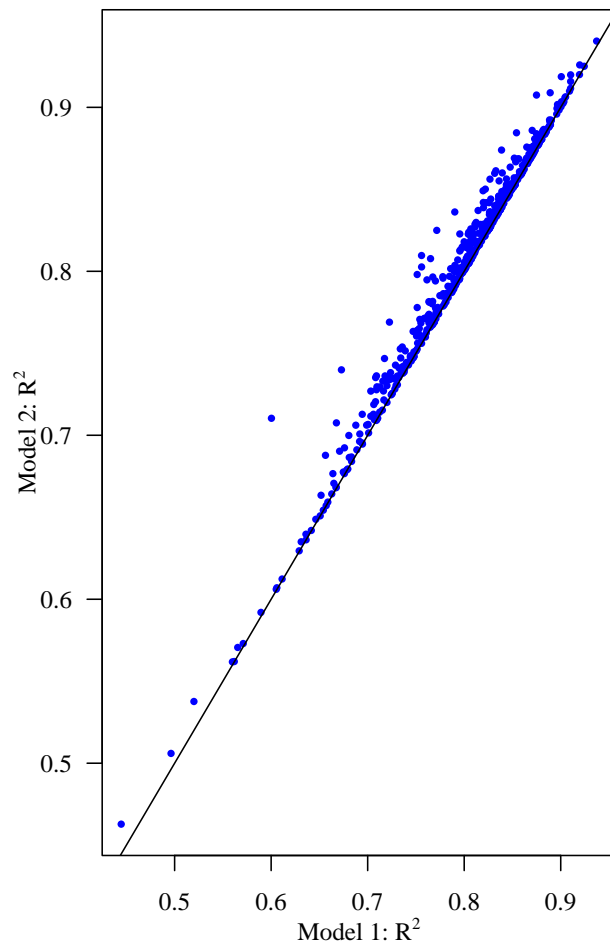
```

abline(0, 1)

# Difference in R^2 (Model 2 - Model 1)
boxplot(R.sq[, 3] - R.sq[, 1],
        las = 1,
        xlab = expression(paste(R^2, ": Model 2 - Model 1")))

abline(h = 0, lty = 2, col = "red")

```



Results: Behavior of Adjusted R^2

```

par(las = 1, mfrow = c(1, 2), mar = c(3.5, 3.75, 1, 0.5), mgp = c(2, 1, 0), family = "serif")

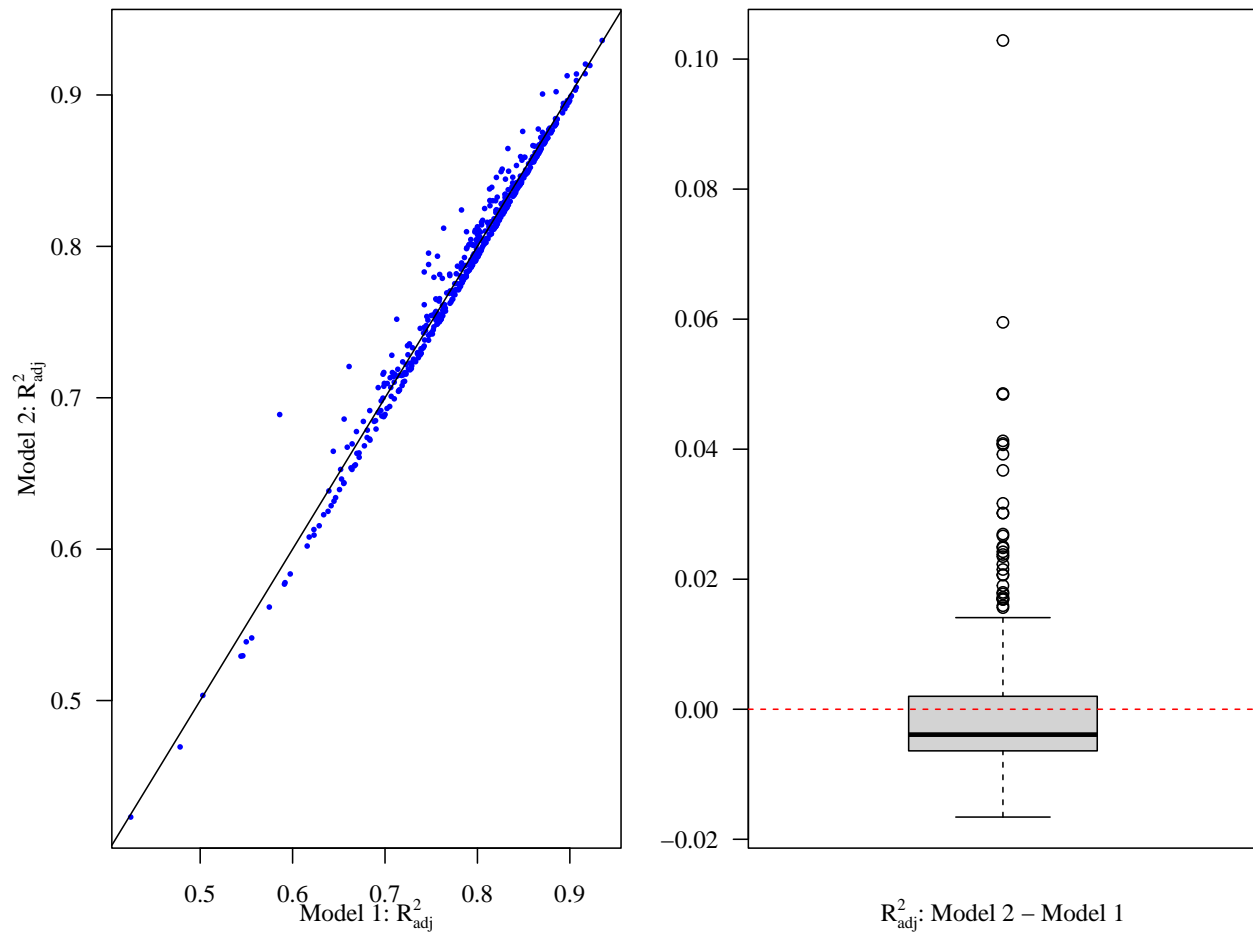
# Compare adjusted R^2 values
plot(R.sq[, 2], R.sq[, 4],
     pch = 16, cex = 0.5, col = "blue",
     xlab = expression(paste("Model 1: ", R[adj]^2)),
     ylab = expression(paste("Model 2: ", R[adj]^2)))

# Reference line
abline(0, 1)

```

```
# Difference in adjusted R^2
boxplot(R.sq[, 4] - R.sq[, 2],
       las = 1,
       xlab = expression(paste(R[adj]^2, ": Model 2 - Model 1")))

abline(h = 0, lty = 2, col = "red")
```



What we observe:

- R^2 :
 - Almost always increases when adding x_2
 - Even though x_2 is NOT useful
- Adjusted R^2 :
 - Often decreases when adding x_2
 - Penalizes unnecessary predictors

Conclusion:

- R^2 can be misleading for model selection
- Adjusted R^2 provides a better balance between fit and model complexity