Combined Distance Method for Species Delimitation

Complete Methods Guide with Implementation Instructions

Jyhreh Johnson - University of Texas at Austin

2026-02-22

Contents

1 INTRODUCTION

1.1 Overview

This document provides complete instructions for implementing the combined Mahalanobis-Gower distance method for species delimitation in fossil hominins. The method combines:

- **Mahalanobis distance** (continuous measurements with covariance structure)
- **Gower distance** (discrete morphological characters)
- **Hierarchical variance partitioning** (temporal and geographic structure)

1.2 Why This Method?

Traditional distance metrics have limitations for paleontological data:

- **Euclidean distance**: Ignores correlations between measurements
- **Mahalanobis alone**: Cannot incorporate discrete traits
- **Standard Gower**: Ignores covariance structure

Our combined approach addresses all three limitations.

1.3 Key Innovation

The method establishes **empirically calibrated thresholds** from simulations:

- **Species threshold**: Mahalanobis $D^2 > 4.0$, Accuracy $> 80\%$
- **Temporal variance threshold**: $<30\%$ indicates chronospecies
- **Geographic variance threshold**: $<15\%$ indicates single species

2 WORKFLOW OVERVIEW

2.1 Complete Analysis Pipeline

```
STEP 1: PREPARE YOUR DATA
- Continuous measurements (dental dimensions)
- Discrete characters (cusp patterns, morphology)
- Temporal/geographic metadata

                        ↓

STEP 2: CALCULATE COMBINED DISTANCE
- Mahalanobis distance (continuous)
- Gower distance (discrete)
- Combine: D_combined =  ×D_mahal + (1- )×D_gower

                        ↓

STEP 3: CLASSIFY & CLUSTER
- k-NN classification with cross-validation
- PAM clustering with silhouette scores
- Hierarchical clustering for validation

                        ↓

STEP 4: VARIANCE PARTITIONING
- Temporal variance (chronospecies test)
- Geographic variance (single species test)
- Compare to species threshold (30%)

                        ↓

STEP 5: TAXONOMIC DECISION
- Apply decision criteria
- Generate formal revision
- Create identification key
```

3 DATA PREPARATION

3.1 Required Data Structure

3.1.1 Your Data Should Look Like This:

```
# Example data format
specimen_data <- data.frame(
  specimen_id = c("AL_288-1", "AL_333-1", "Sts_5"),
  taxon = c("Au_afarensis", "Au_afarensis", "Au_africanus"),

  # Continuous measurements (mm)
  M1_BL = c(12.3, 12.8, 11.5),
  M1_MD = c(11.2, 11.5, 10.8),
  M2_BL = c(13.1, 13.4, 12.2),
  M2_MD = c(11.8, 12.0, 11.3),
  P4_BL = c(10.2, 10.5, 9.8),

  # Discrete characters (factors)
  cusp_pattern = c("Y5", "Y5", "Y4"),
  hypocone_size = c("large", "medium", "medium"),
  cingulum = c("strong", "weak", "weak"),

  # Optional metadata
  site = c("Hadar", "Hadar", "Sterkfontein"),
  age_ma = c(3.2, 3.2, 2.5),

  stringsAsFactors = TRUE
)
```

3.1.2 Key Requirements:

1. **Specimen ID column**: Unique identifier for each fossil
2. **Taxon column**: Current taxonomic assignment
3. **Continuous measurements**: At least 3-5 dental dimensions
4. **Discrete characters**: At least 2-3 morphological traits
5. **Factor encoding**: Discrete characters must be factors, not strings

3.1.3 Save Your Data:

```
# Save as CSV
write.csv(specimen_data, "data/australopithecus_data.csv",
          row.names = FALSE)
```

```r
# Or save as R object
saveRDS(specimen_data, "data/australopithecus_data.rds")
```

4 KEY FUNCTIONS EXPLAINED

4.1 Function 1: Calculate Combined Distance

4.1.1 What It Does:

Combines Mahalanobis distance (accounts for covariance) with Gower distance (handles discrete traits).

4.1.2 How It Works:

```
# STEP 1: Calculate Mahalanobis distance
# - Centers continuous data
# - Computes covariance matrix S
# - Calculates D² = (x -x ) S ¹ (x -x )
# - Result: distances accounting for correlations

# STEP 2: Standardize to 0-1 range
# - Divide by maximum distance
# - Makes comparable to Gower distance

# STEP 3: Calculate Gower distance for discrete traits
# - For each trait: similarity = 1 if match, 0 if different
# - Average across all traits
# - Result: 0 (identical) to 1 (completely different)

# STEP 4: Combine distances
# D_combined =   × D_mahalanobis + (1- ) × D_gower
# where    0.65 (optimal from simulations)
```

4.1.3 The Code:

```
calc_combined_distance <- function(data, continuous_vars, discrete_vars,
                                   alpha = 0.65, robust = TRUE) {

  n <- nrow(data)

  # 1. Mahalanobis for continuous
  continuous_data <- data[, continuous_vars, drop = FALSE]
  centered <- scale(continuous_data, center = TRUE, scale = FALSE)
  S <- cov(centered)

  # Handle near-singular covariance (add small ridge)
  if (robust && (det(S) < 1e-10)) {
```

```
    S <- S + diag(0.001, ncol(S))
  }

  S_inv <- solve(S)

  # Calculate pairwise Mahalanobis distances
  D_mahal <- matrix(0, n, n)
  for (i in 1:(n-1)) {
    for (j in (i+1):n) {
      diff <- as.numeric(continuous_data[i,] - continuous_data[j,])
      D_mahal[i,j] <- sqrt(t(diff) %*% S_inv %*% diff)
      D_mahal[j,i] <- D_mahal[i,j]
    }
  }

  # Standardize to 0-1
  D_mahal_scaled <- D_mahal / max(D_mahal[D_mahal < Inf])

  # 2. Gower for discrete
  if (length(discrete_vars) > 0) {
    discrete_data <- data[, discrete_vars, drop = FALSE]
    D_gower <- as.matrix(daisy(discrete_data, metric = "gower"))
  } else {
    D_gower <- matrix(0, n, n)
  }

  # 3. Combine
  D_combined <- alpha * D_mahal_scaled + (1 - alpha) * D_gower

  return(list(
    distance = as.dist(D_combined),
    distance_matrix = D_combined,
    mahalanobis_matrix = D_mahal_scaled,
    gower_matrix = D_gower,
    alpha = alpha
  ))
}
```

4.1.4 Key Parameters:

- **alpha**: Weight for continuous data (default 0.65 = 65% continuous, 35% discrete)
- **robust**: If TRUE, adds small value to diagonal of singular covariance matrices
- **Returns**: Complete distance breakdown for analysis

4.2 Function 2: Classification with Cross-Validation

4.2.1 What It Does:

Tests how well specimens can be correctly assigned to species using k-nearest neighbors.

4.2.2 Why This Matters:

- **High accuracy (>80%)** = species are distinct
- **Low accuracy (<70%)** = taxa are not distinguishable
- **Borderline (70-80%)** = uncertain, need more data

4.2.3 The Code:

```
classify_with_distance <- function(distance_matrix, taxa,
                                   k_folds = 5, k_neighbors = 5) {

  dist_mat <- as.matrix(distance_matrix)
  n <- nrow(dist_mat)

  # Set up k-fold cross-validation
  fold_ids <- sample(rep(1:k_folds, length.out = n))

  predictions <- rep(NA, n)
  posterior_probs <- matrix(NA, nrow = n, ncol = length(unique(taxa)))
  colnames(posterior_probs) <- sort(unique(taxa))

  # For each fold
  for (fold in 1:k_folds) {
    test_idx <- which(fold_ids == fold)
    train_idx <- which(fold_ids != fold)

    # For each test specimen
    for (i in test_idx) {
      # Find k nearest neighbors in training set
      dists_to_train <- dist_mat[i, train_idx]
      nearest_idx <- order(dists_to_train)[1:k_neighbors]
      nearest_taxa <- taxa[train_idx[nearest_idx]]

      # Posterior probabilities (proportion of neighbors)
      taxon_counts <- table(nearest_taxa)
      for (tx in names(taxon_counts)) {
        posterior_probs[i, tx] <- taxon_counts[tx] / k_neighbors
      }

      # Prediction: majority vote
```

```
      predictions[i] <- names(sort(taxon_counts, decreasing = TRUE))[1]
    }
  }

  # Calculate metrics
  accuracy <- mean(predictions == taxa, na.rm = TRUE)
  conf_mat <- table(Predicted = predictions, True = taxa)
  mean_confidence <- mean(apply(posterior_probs, 1, max, na.rm = TRUE))

  return(list(
    accuracy = accuracy,
    predictions = predictions,
    posterior_probs = posterior_probs,
    confusion_matrix = conf_mat,
    mean_confidence = mean_confidence
  ))
}
```

4.2.4 Interpretation:

```
# Example results:
# accuracy = 0.873 (87.3%)
# mean_confidence = 0.812 (81.2%)
#
# Interpretation:
# - 87.3% of specimens correctly classified
# - Average 81% confidence in predictions
# - Both above 80% threshold → species are distinct
```

4.3 Function 3: Clustering Analysis

4.3.1 What It Does:

Finds natural groups in the data and estimates the number of species (k).

4.3.2 Key Metrics:

- **Optimal k**: Number of clusters detected
- **Silhouette score**: How well-separated clusters are
  - 0.70: Strong separation
  - 0.50-0.70: Moderate separation
  - <0.40: Weak separation (possible oversplit)

4.3.3 The Code:

```
cluster_with_distance <- function(distance_matrix, true_k = NULL,
                                  max_k = 6) {

  dist_obj <- as.dist(distance_matrix)

  # Try different numbers of clusters
  silhouette_scores <- numeric(max_k - 1)
  pam_results <- list()

  for (k in 2:max_k) {
    pam_fit <- pam(dist_obj, k = k, diss = TRUE)
    pam_results[[k]] <- pam_fit
    silhouette_scores[k - 1] <- pam_fit$silinfo$avg.width
  }

  # Select k with highest silhouette
  optimal_k <- which.max(silhouette_scores) + 1
  best_pam <- pam_results[[optimal_k]]

  # Hierarchical clustering for comparison
  hc <- hclust(dist_obj, method = "ward.D2")

  return(list(
    optimal_k = optimal_k,
    true_k = true_k,
    silhouette_scores = silhouette_scores,
    best_silhouette = silhouette_scores[optimal_k - 1],
    pam_clusters = best_pam$clustering,
    hc_model = hc
```

```
  ))
}
```

4.3.4 Decision Rules:

```
# If optimal_k = true_k AND silhouette > 0.6:
#   → Species are well-separated, maintain taxonomy

# If optimal_k  true_k OR silhouette < 0.4:
#   → Weak separation, investigate oversplitting

# If optimal_k > true_k:
#   → May indicate cryptic species OR temporal/geographic structure
```

4.4 Function 4: Variance Partitioning

4.4.1 What It Does:

Separates morphological variation into components: - **Temporal variance**: Change over time (chronospecies?) - **Geographic variance**: Regional differences (single species?) - **Species variance**: Differences between species

4.4.2 Key Thresholds (from simulations):

- **Inter-specific variance**: ~32% (species threshold)
- **Temporal variance <30%**: Indicates chronospecies
- **Geographic variance <15%**: Indicates single widespread species

4.4.3 The Code:

```
analyze_temporal_variation <- function(data, continuous_vars,
                                       time_var = "age_ma") {

  library(lme4)

  results <- list()

  for (var in continuous_vars) {
    # Hierarchical model: morphology ~ time + (1|taxon)
    formula <- as.formula(paste(var, "~", time_var, "+ (1|taxon)"))
    model <- lmer(formula, data = data, REML = TRUE)

    # Extract variance components
    var_comps <- as.data.frame(VarCorr(model))

    # Calculate proportions
    total_var <- sum(var_comps$vcov)
    temporal_var_prop <- var_comps$vcov[var_comps$grp == "taxon"] / total_var

    results[[var]] <- list(
      model = model,
      temporal_var_prop = temporal_var_prop
    )
  }

  # Average across variables
  mean_temporal_var <- mean(sapply(results, function(x) x$temporal_var_prop))

  return(list(
    continuous = results,
```

```
    mean_temporal_variance = mean_temporal_var
  ))
}
```

4.4.4 Interpretation Example:

```
# Results:
# mean_temporal_variance = 0.184 (18.4%)
# species_threshold = 0.321 (32.1%)
#
# Interpretation:
# 18.4% < 32.1% → Temporal variance below species threshold
# Conclusion: This is ONE species evolving through time (chronospecies)
# Recommendation: Synonymize early and late forms
```

5 DECISION CRITERIA

5.1 Empirically Calibrated Thresholds

These thresholds were established from simulations with known species boundaries:

5.1.1 Strong Evidence for Distinct Species:

```
 Mahalanobis D² > 4.0
 Classification accuracy > 80%
 Silhouette score > 0.60
 Mean posterior confidence > 0.85
 Between/within distance ratio > 2.0

Action: RECOGNIZE AS DISTINCT SPECIES
```

5.1.2 Weak Evidence (Should Lump):

```
 Mahalanobis D² < 2.5
 Classification accuracy < 70%
 Silhouette score < 0.40
 Mean posterior confidence < 0.70
 Between/within distance ratio < 1.5

Action: SYNONYMIZE (lump into single species)
```

5.1.3 Borderline (Uncertain):

```
~ Mahalanobis D² = 2.5-4.0
~ Classification accuracy = 70-80%
~ Silhouette score = 0.40-0.60
~ Sample size may be insufficient

Action: FLAG AS UNCERTAIN, collect more data
```

5.2 Variance Partitioning Criteria:

5.2.1 Chronospecies (Temporal):

```
IF temporal variance < 30% (species threshold)
AND significant linear trend (p < 0.01)
AND no morphological discontinuities
THEN: Single evolving lineage (synonymize)
```

5.2.2 Geographic Variation:

```
IF geographic variance < 15%
AND no significant ANOVA (p > 0.05)
AND morphospace overlap > 40%
THEN: Single widespread species (synonymize)
```

6 IMPLEMENTING WITH REAL DATA

6.1 Step-by-Step Guide

6.1.1 STEP 1: Install Required Packages

```
# Run this once
required_packages <- c(
  "MASS",         # Mahalanobis distance
  "cluster",      # Gower distance and PAM
  "lme4",         # Hierarchical models
  "ggplot2",      # Plotting
  "tidyverse"     # Data manipulation
)

install.packages(required_packages)
```

6.1.2 STEP 2: Prepare Your Data

```
# Load your data
australopith_data <- read.csv("data/australopithecus_data.csv",
                              stringsAsFactors = TRUE)

# Define variable names
continuous_vars <- c("M1_BL", "M1_MD", "M2_BL", "M2_MD", "P4_BL")
discrete_vars <- c("cusp_pattern", "hypocone_size", "cingulum")

# Check for missing data
summary(australopith_data[, continuous_vars])
summary(australopith_data[, discrete_vars])

# Handle missing data (if <30% missing)
# Option 1: Remove specimens with >50% missing
australopith_data <- australopith_data[
  rowSums(is.na(australopith_data[, continuous_vars])) < 3,
]

# Option 2: Impute missing values (use with caution)
# library(mice)
# imputed <- mice(australopith_data[, continuous_vars], m=5)
# australopith_data[, continuous_vars] <- complete(imputed)
```

6.1.3 STEP 3: Calculate Combined Distance

```r
# Source the functions
source("combined_distance_functions.R")

# Calculate distances
combined_results <- calc_combined_distance(
  data = australopith_data,
  continuous_vars = continuous_vars,
  discrete_vars = discrete_vars,
  alpha = 0.65,  # Use optimized value from simulations
  robust = TRUE
)

# Examine results
print(combined_results$diagnostics)
```

6.1.4 STEP 4: Run Classification

```r
# Classify specimens
classification <- classify_with_distance(
  distance_matrix = combined_results$distance,
  taxa = australopith_data$taxon,
  k_folds = 5,
  k_neighbors = 5
)

# Print results
cat("Classification Accuracy:",
    round(classification$accuracy * 100, 1), "%\n")
cat("Mean Confidence:",
    round(classification$mean_confidence * 100, 1), "%\n")

# View confusion matrix
print(classification$confusion_matrix)
```

6.1.5 STEP 5: Run Clustering

```r
# Cluster analysis
clustering <- cluster_with_distance(
  distance_matrix = combined_results$distance,
  true_k = length(unique(australopith_data$taxon)),
  max_k = 6
```

```
)

# Print results
cat("Optimal k:", clustering$optimal_k, "\n")
cat("True k:", clustering$true_k, "\n")
cat("Silhouette score:",
    round(clustering$best_silhouette, 3), "\n")
```

6.1.6 STEP 6: Variance Partitioning (If Applicable)

```
# For chronospecies analysis
if ("age_ma" %in% names(australopith_data)) {
  temporal_analysis <- analyze_temporal_variation(
    data = australopith_data,
    continuous_vars = continuous_vars,
    time_var = "age_ma"
  )

  cat("Mean temporal variance:",
      round(temporal_analysis$mean_temporal_variance * 100, 1), "%\n")
  cat("Species threshold: 30%\n")

  if (temporal_analysis$mean_temporal_variance < 0.30) {
    cat("INTERPRETATION: Chronospecies (synonymize)\n")
  } else {
    cat("INTERPRETATION: Multiple species\n")
  }
}
```

6.2 Making Taxonomic Decisions

6.2.1 Example Decision Process:

```r
# Compare Au. afarensis vs. Au. africanus

# Extract specimens
afarensis_idx <- which(australopith_data$taxon == "Au_afarensis")
africanus_idx <- which(australopith_data$taxon == "Au_africanus")

# Calculate between-taxon distance
dist_mat <- combined_results$distance_matrix
between_dist <- mean(dist_mat[afarensis_idx, africanus_idx])

# Get classification accuracy for this pair
# (subset data to just these two species and re-run)

# Decision rules:
if (between_dist > 0.4 && classification$accuracy > 0.80) {
  decision <- "RECOGNIZE AS DISTINCT"
} else if (between_dist < 0.25 && classification$accuracy < 0.70) {
  decision <- "SYNONYMIZE"
} else {
  decision <- "UNCERTAIN - collect more data"
}

cat("Decision:", decision, "\n")
cat("Supporting evidence:\n")
cat("  Mean distance:", round(between_dist, 3), "\n")
cat("  Accuracy:", round(classification$accuracy * 100, 1), "%\n")
cat("  Silhouette:", round(clustering$best_silhouette, 3), "\n")
```

7 INTERPRETATION GUIDE

7.1 How to Read Your Results

7.1.1 Scenario 1: Clear Distinct Species

```
Results:
  D² = 5.2
  Accuracy = 89%
  Silhouette = 0.72
  Confidence = 87%

Interpretation:
  All metrics above thresholds
  Species are morphologically distinct
  Can be reliably diagnosed
Action: MAINTAIN AS SEPARATE SPECIES
```

7.1.2 Scenario 2: Chronospecies

```
Results:
  D² = 2.8 (below 4.0 threshold)
  Accuracy = 72% (below 80%)
  Temporal variance = 18% (below 30%)
  Linear trend: R² = 0.81, p < 0.001

Interpretation:
  Weak separation
  Strong temporal pattern
  Gradual morphological change
Action: SYNONYMIZE (single evolving lineage)
```

7.1.3 Scenario 3: Oversplit Taxa

```
Results:
  D² = 1.6 (well below threshold)
  Accuracy = 65% (near random)
  Silhouette = 0.28 (weak)
  Geographic variance = 11% (below 15%)

Interpretation:
  Very weak separation
  Classification barely better than chance
  Geographic variation explains differences
Action: SYNONYMIZE (geographic variant)
```

7.1.4 Scenario 4: Uncertain Case

```
Results:
  D² = 3.2 (borderline)
  Accuracy = 76% (borderline)
  Silhouette = 0.52 (moderate)
  Sample size: n = 8 (small)

Interpretation:
~ Metrics in gray zone
~ Insufficient statistical power
~ Need more specimens
Action: TENTATIVELY MAINTAIN pending more data
```

8 TROUBLESHOOTING

8.1 Common Issues and Solutions

8.1.1 Issue 1: Singular Covariance Matrix

Error: `Error in solve.default(S): system is computationally singular`

Cause: Variables are too highly correlated or sample size too small

Solution:

```
# Option 1: Use robust = TRUE (already default)
combined_results <- calc_combined_distance(..., robust = TRUE)

# Option 2: Remove redundant variables
cor_matrix <- cor(data[, continuous_vars], use = "complete.obs")
# Remove variables with correlation > 0.95

# Option 3: Use PCA first
pca <- prcomp(data[, continuous_vars], scale = TRUE)
pc_scores <- pca$x[, 1:3]  # Use first 3 PCs
```

8.1.2 Issue 2: Low Classification Accuracy for All Comparisons

Problem: Even well-separated species show <70% accuracy

Possible Causes: 1. Sample size too small (n < 10 per species) 2. Too much missing data (>40%) 3. High measurement error 4. Wrong alpha value

Solutions:

```
# Check sample sizes
table(data$taxon)  # Need n  15 per species

# Check missing data
missing_prop <- rowSums(is.na(data[, continuous_vars])) /
                length(continuous_vars)
table(missing_prop > 0.3)  # Should be mostly FALSE

# Try different alpha
alpha_test <- seq(0.4, 1.0, 0.1)
results <- lapply(alpha_test, function(a) {
  calc_combined_distance(..., alpha = a)
  # Run classification...
})
```

8.1.3 Issue 3: Clustering Finds Wrong Number of Species

Problem: Optimal k expected number of species

Interpretation:

```
# If optimal_k > true_k:
#   → May indicate:
#      1. Temporal/geographic structure (check variance partitioning)
#      2. Cryptic species (check if clusters are geographically separated)
#      3. Sampling artifacts (check if clusters = sites)

# If optimal_k < true_k:
#   → May indicate:
#      1. Species are not morphologically distinct
#      2. Oversplit taxonomy (some species should be synonymized)
#      3. Insufficient morphological data
```

8.1.4 Issue 4: Very High Temporal Variance (>40%)

Problem: Temporal variance exceeds species threshold

Interpretation:

```
# This suggests:
# 1. Multiple species present (NOT a chronospecies)
# 2. Discontinuous evolution (check for morphological jumps)
# 3. Mixed samples (check taxonomic assignments)

# Investigate:
# - Plot morphology vs. time (look for jumps)
# - Check if "temporal" variance is actually taxonomic
# - Verify age estimates are correct
```

9 VALIDATION CHECKLIST

Before finalizing taxonomic decisions, verify:

9.1 Data Quality Checks

- ☐ Sample sizes: n 15 per species for well-supported taxa
- ☐ Missing data: <30% per specimen, <40% per variable
- ☐ Outliers identified and investigated
- ☐ Measurement error estimated (should be <0.5mm SD)
- ☐ Taxonomic assignments verified against literature

9.2 Statistical Validation

- ☐ Cross-validation performed (k-fold, not just training accuracy)
- ☐ Multiple distance metrics compared (combined should outperform)
- ☐ Clustering validated with multiple methods (PAM + hierarchical agree)
- ☐ Posterior probabilities examined (not just point predictions)

9.3 Biological Plausibility

- ☐ Temporal ranges do not violate decisions
  - Sympatric species MUST have $D^2 > 4.0$
  - Chronospecies should show temporal continuity
- ☐ Geographic distributions make sense
  - Allopatric species may have lower D^2
  - Sympatric species require higher D^2
- ☐ Functional morphology consistent with taxonomy
  - Different species should show ecological differences

9.4 Decision Consistency

- ☐ Multiple criteria converge on same decision
- ☐ Borderline cases flagged as uncertain
- ☐ Small-sample taxa not over-interpreted
- ☐ Synonymies supported by variance partitioning

10 REPORTING RESULTS

10.1 Elements to Include in Your Publication

10.1.1 Methods Section:

"Species delimitation was performed using a combined distance metric combining Mahalanobis distance (accounting for covariance among continuous measurements) with Gowers coefficient (incorporating discrete morphological characters). The combined distance (D_combined = × D_Mahalanobis + (1- all specimen pairs. Taxonomic decisions were based on empirically calibrated thresholds established from simulation studies: species were recognized as distinct if Mahalanobis $D^2 > 4.0$, classification accuracy > 80%, and silhouette score > 0.60; taxa were synonymized if $D^2 < 2.5$ and accuracy < 70%. Hierarchical mixed-effects models were used to partition temporal and geographic variance, with temporal variance < 30% indicating chronospecies and geographic variance < 15% indicating a single widespread species."

10.1.2 Results Section:

Present for each species comparison: - Mahalanobis D^2 (scaled) - Classification accuracy with 95% CI - Silhouette score - Mean posterior confidence - Decision and justification

10.1.3 Table Format:

Table X: Pairwise Species Delimitation Results

Comparison	D^2	Acc	Sil	Decision	Evidence
Au. afarensis-africanus	5.2	89%	0.72	Distinct	Strong (4/4)
Au. anamensis-afarensis	2.8	72%	0.48	Synonymize (chr)	Temporal <30%
Au. africanus-sediba	3.2	76%	0.51	Uncertain	Small n (n=2)

Abbreviations: D^2 = Mahalanobis distance (scaled); Acc = classification accuracy; Sil = silhouette score; chr = chronospecies

11 REFERENCES

11.1 Key Citations for Methods

combined Distance Approach: - Gower JC (1971) A general coefficient of similarity. *Biometrics* 27:857-871 - Mahalanobis PC (1936) On the generalized distance in statistics. *Proceedings of the National Institute of Sciences of India* 2:49-55

Species Delimitation: - Wood B, Lieberman DE (2001) Craniodental variation in *Paranthropus boisei*. *American Journal of Physical Anthropology* 116:13-25 - Kimbel WH, Rak Y (1993) The importance of species taxa in paleoanthropology. *American Journal of Physical Anthropology* 91:315-327

Statistical Methods: - Kaufman L, Rousseeuw PJ (1990) *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York - Hastie T, Tibshirani R, Friedman J (2009) *The Elements of Statistical Learning*. Springer, New York

Morphometric Analysis: - Wood BA (1991) *Koobi Fora Research Project, Volume 4: Hominid Cranial Remains*. Oxford University Press - Spoor F et al. (2015) Reconstructed Homo habilis type OH 7. *Nature* 519:83-86

12 APPENDIX: COMPLETE FUNCTION LIBRARY

12.1 All Functions in One Place

For your convenience, here are all the main functions together:

```
# [Include complete code for all functions here]
# calc_combined_distance()
# classify_with_distance()
# cluster_with_distance()
# analyze_temporal_variation()
# analyze_geographic_variation()
# make_taxonomic_decision()
```

---

13 SUPPORT

13.1 Getting Help

If you encounter issues:

1. Check the Troubleshooting section (page XX)
2. Verify your data format matches examples
3. Ensure all packages are installed and loaded
4. Check that sample sizes are adequate (n 15)

13.2 Contact

For questions about implementation: - Email: your.email@institution.edu - GitHub: github.com/yourusername/co
distance

Document Version: 1.0
Last Updated: 2026-02-22
Software: R version 4.5.2