Package ‘micer’

July 22, 2025

Type Package

Title Map Image Classification Efficacy

Version 0.2.0

Date 2025-01-05

Maintainer Aaron Maxwell <Aaron.Maxwell@mail.wvu.edu>

Description Map image classification efficacy (MICE) adjusts the accuracy rate relative to a random classification baseline (Shao et al. (2021)<doi:10.1109/ACCESS.2021.3116526> and Tang et al. (2024)<doi:10.1109/TGRS.2024.3446950>). portions from the reference labels are considered, as opposed to the proportions from the reference and predictions, as is the case for the Kappa statistic. This package offers means to calculate MICE and adjusted versions of class-level user's accuracy (i.e., precision) and producer's accuracy (i.e., recall) and F1-scores. Class-level metrics are aggregated using macro-averaging. Functions are also made available to estimate confidence intervals using bootstrapping and statistically compare two classification results.

Depends R (>= 2.10)

Imports dplyr (>= 1.1.3),

License GPL (>= 3)

URL https://github.com/maxwell-geospatial/micer

BugReports https://github.com/maxwell-geospatial/micer/issues

NeedsCompilation no

Encoding UTF-8

RoxygenNote 7.3.2

Suggests knitr, rmarkdown

VignetteBuilder knitr

Author Aaron Maxwell [aut, cre, cph],
Sarah Farhadpour [aut]

Repository CRAN

Date/Publication 2025-01-20 21:00:01 UTC

Contents

`biData` *Example binary classification dataset*

Description

Example binary classification dataset. "Mine" is the positive case and "Not Mine" is the background class. There are 178 samples from the "Mine" class and 4,822 samples from the "Not Mine" class. Counts are relative to reference labels. Class proportions are based on landscape proportions. There are a total of 5,000 samples.

Format

ref reference label

pred predicted label

References

Maxwell, A.E., Bester, M.S., Guillen, L.A., Ramezan, C.A., Carpinello, D.J., Fan, Y., Hartley, F.M., Maynard, S.M. and Pyron, J.L., 2020. Semantic segmentation deep learning for extracting surface mine extents from historic topographic maps. Remote Sensing, 12(24), p.4145.

`compareData` *Data for multiclass classification comparison*

Description

Example multiclass classification dataset with the following wetland-related classes: "PFO", "PEM", "RLP", and "Not". PFO = Palustrine Forested; PEM = Palustrine Emergent; RLP = River, Lake, Pond; Not = Not Wetland. There are 600 examples from each class relative to the reference labels.

Format

ref correct label

rfRred random forest prediction

dfRred single decision tree prediction

References

These data are unpublished

---

mcData *Example multiclass classification dataset*

Description

Example multiclass classification dataset with the following classes (counts relative to reference labels): "Barren" (n=163), "Forest" (n=20,807), "Impervious" (n=426), "Low Vegetation" (n=3,182), "Mixed Dev" (n=520), and "Water" (n=200). There are a total of 25,298 samples.

Format

ref reference label

pred predicted label

References

Maxwell, A.E., Strager, M.P., Warner, T.A., Ramezan, C.A., Morgan, A.N. and Pauley, C.E., 2019. Large-area, high spatial resolution land cover mapping using random forests, GEOBIA, and NAIP orthophotography: Findings and recommendations. Remote Sensing, 11(12), p.1409.

---

mice *mice*

Description

Calculate map image classification efficacy (MICE) and other metrics using columns/vectors of reference and predicted classes

Usage

```
mice(
  reference,
  prediction,
  mappings = levels(as.factor(reference)),
  multiclass = TRUE,
  positiveIndex = 1
)
```

Arguments

`reference`	column/vector of reference labels as factor data type.
`prediction`	column/vector of predicted labels as factor data type.
`mappings`	names of classes (if not provided, factor levels are used).
`multiclass`	TRUE or FALSE. If TRUE, treats classification as multiclass. If FALSE, treats classification as binary. Default is TRUE.
`positiveIndex`	index for positive case for binary classification. Ignored for multiclass classification. Default is 1 or first factor level.

Details

For multiclass classification, returns a list object with the following items: $Mappings = class names; $confusionMatrix = confusion matrix where columns represent the reference data and rows represent the classification result; $referenceCounts = count of samples in each reference class; $predictionCounts = count of predictions in each class; $overallAccuracy = overall accuracy; $MICE = map image classification efficacy; $usersAccuracies = class-level user's accuracies (1 - commission error); $CTBICEs = classification-total-based image classification efficacies (adjusted user's accuracies); $producersAccuracies = class-level producer's accuracies (1 - omission error); $RTBICEs = reference-total-based image classification efficacies (adjusted producer's accuracies); $F1Scores = class-level harmonic mean of user's and producer's accuracies; $F1Efficacies = F1-score efficacies; $macroPA = class-aggregated, macro-averaged producer's accuracy; $macroRTBICE = class-aggregated, macro-averaged reference-total-based image classification efficacy; $macroUA = class-aggregated, macro-averaged user's accuracy; $macroCTBICE = class-aggregated, macro-averaged classification-total-based image classification efficacy; $macroF1 = class-aggregated, macro-averaged F1-score; $macroF1Efficacy = class-aggregated, macro-averaged F1 efficacy;

For binary classification, returns a list object with the following items: $Mappings = class names; $confusionMatrix = confusion matrix where columns represent the reference data and rows represent the classification result; $referenceCounts = count of samples in each reference class; $predictionCounts = count of predictions in each class; $postiveCase = name or mapping for the positive case; $overallAccuracy = overall accuracy; $MICE = map image classification efficacy; $Precision = precision (1 - commission error relative to positive case); $precisionEfficacy = precision efficacy; $NPV = negative predictive value (1 - commission error relative to negative case); $npvEfficacy = negative predictive value efficacy; $Recall = recall (1 - omission error relative to positive case); $recallEfficacy = recall efficacy; $specificity = specificity (1 - omission error relative to negative case); $specificityEfficacy = specificity efficacy; $f1Score = harmonic mean of precision and recall; $f1Efficacy = F1-score efficacy;

Value

multiclass or binary assessment metrics in a list object. See details for description of generated metrics.

Examples

```
#Multiclass example
data(mcData)
```

```
mice(mcData$ref,
mcData$pred,
mappings=c("Barren", "Forest", "Impervious", "Low
Vegetation", "Mixed Dev", "Water"),
multiclass=TRUE)

#Binary example
data(biData)
mice(biData$ref,
biData$pred,
mappings = c("Mined", "Not Mined"),
multiclass=FALSE,
positiveIndex=1)
```

`miceCI` *miceCI*

Description

Calculate confidence intervals (CIs) for MICE and associated metrics using bootstrap sampling and the percentile method.

Usage

```
miceCI(
  reps = 200,
  frac = 0.7,
  lowPercentile,
  highPercentile,
  reference,
  prediction,
  mappings = levels(as.factor(reference)),
  multiclass = TRUE,
  positiveIndex = 1
)
```

Arguments

`reps`	number of bootstrap replicates to use. Default is 200.
`frac`	proportion of samples to include in each bootstrap sample. Default is 0.7.
`lowPercentile`	lower percentile for confidence interval. Default is 0.025 for a 95% CI.
`highPercentile`	upper percentile for confidence interval. Default is 0.975 for a 95% CI.
`reference`	column of reference labels as factor data type.
`prediction`	column of predicted labels as factor data type.
`mappings`	names of classes (if not provided, factor levels are used).
`multiclass`	TRUE or FALSE. If TRUE, treats classification as multiclass. If FALSE, treats classification as binary. Default is TRUE.

`positiveIndex` index for positive case for binary classification. Ignored for multiclass classification. Default is 1 or first factor level

Details

Confidence intervals are estimated for overall accuracy, MICE, and all class-aggregated, macro-averaged metrics produced by mice() or miceCM(). Returns metric name, mean metric value, median metric value, lower confidence interval bounds (low.ci), and upper confidence interval bounds (upper.ci) as a dataframe object.

Value

dataframe object of metric name and estimated mean value, median value, and lower and upper CIs.

Examples

```
#Multiclass example
data(mcData)
ciResultsMC <- miceCI(rep=100,
frac=.7,
mcData$ref,
mcData$pred,
lowPercentile=0.025,
highPercentile=0.975,
mappings=c("Barren", "Forest", "Impervious", "Low Vegetation", "Mixed Dev", "Water"),
multiclass=TRUE)

print(ciResultsMC)

#Binary example
data(biData)
ciResultsBi <- miceCI(rep=100,
frac=.7,
biData$ref,
biData$pred,
lowPercentile=0.025,
highPercentile=0.975,
mappings = c("Mined", "Not Mined"),
multiclass=FALSE,
positiveIndex=1)

print(ciResultsBi)
```

`miceCM` *miceCM*

Description

Calculate map image classification efficacy (MICE) and other metrics using confusion matrix

Usage

```
miceCM(
  cm,
  mappings = levels(as.factor(row.names(cm))),
  multiclass = TRUE,
  positiveIndex = 1
)
```

Arguments

`cm`	confusion matrix as table object where rows define predictions and columns define reference labels.
`mappings`	names of classes (if not provided, factor levels are used).
`multiclass`	TRUE or FALSE. If TRUE, treats classification as multiclass. If FALSE, treats classification as binary. Default is TRUE.
`positiveIndex`	index for positive case for binary classification. Ignored for multiclass classification. Default is 1 or first factor level.

Details

For multiclass classification, returns a list object with the following items: $Mappings = class names; $confusionMatrix = confusion matrix where columns represent the reference data and rows represent the classification result; $referenceCounts = count of samples in each reference class; $predictionCounts = count of predictions in each class; $overallAccuracy = overall accuracy; $MICE = map image classification efficacy; $usersAccuracies = class-level user's accuracies (1 - commission error); $CTBICEs = classification-total-based image classification efficacies (adjusted user's accuracies); $producersAccuracies = class-level producer's accuracies (1 - omission error); $RTBICEs = reference-total-based image classification efficacies (adjusted producer's accuracies); $F1Scores = class-level harmonic mean of user's and producer's accuracies; $F1Efficacies = F1-score efficacies; $macroPA = class-aggregated, macro-averaged producer's accuracy; $macroRTBICE = class-aggregated, macro-averaged reference-total-based image classification efficacy; $macroUA = class-aggregated, macro-averaged user's accuracy; $macroCTBICE = class-aggregated, macro-averaged classification-total-based image classification efficacy; $macroF1 = class-aggregated, macro-averaged F1-score; $macroF1Efficacy = class-aggregated, macro-averaged F1 efficacy;

For binary classification, returns a list object with the following items: $Mappings = class names; $confusionMatrix = confusion matrix where columns represent the reference data and rows represent the classification result; $referenceCounts = count of samples in each reference class; $predictionCounts = count of predictions in each class; $postiveCase = name or mapping for the positive case; $overallAccuracy = overall accuracy; $MICE = map image classification efficacy; $Precision = precision (1 - commission error relative to positive case); $precisionEfficacy = precision efficacy; $NPV = negative predictive value (1 - commission error relative to negative case); $npvEfficacy = negative predictive value efficacy; $Recall = recall (1 - omission error relative to positive case); $recallEfficacy = recall efficacy; $specificity = specificity (1 - omission error relative to negative case); $specificityEfficacy = specificity efficacy; $f1Score = harmonic mean of precision and recall; $f1Efficacy = F1-score efficacy;

Value

multiclass or binary assessment metrics in a list object. See details for description of generated metrics.

Examples

```
#Multiclass example
data(mcData)
cmMC <- table(mcData$pred, mcData$ref)
miceCM(cmMC,
mappings=c("Barren", "Forest", "Impervious", "Low Vegetation", "Mixed Dev", "Water"),
multiclass=TRUE)

#Binary example
data(biData)
cmB <- table(biData$pred, biData$ref)
miceMCResult <- miceCM(cmB,
mappings=c("Mined", "Not Mined"),
multiclass=FALSE,
positiveIndex=1)
print(miceMCResult)
```

---

miceCompare *miceCompare*

Description

Statistically compare two models using a paired t-test and bootstrap samples of the assessment results

Usage

```
miceCompare(ref, result1, result2, reps, frac)
```

Arguments

`ref`	column of reference labels as factor data type.
`result1`	column of predicted labels as factor data type (first result to compare).
`result2`	column of predicted labels as factor data type (second result to compare).
`reps`	number of bootstrap replicates to use. Default is 200.
`frac`	proportion of samples to include in each bootstrap sample. Default is 0.7.

Value

paired t-test results including t-statistic, degrees of freedom, p-value, 95% confidence interval, and mean difference

Examples

```
data(compareData)
compareResult <- miceCompare(ref=compareData$ref,
result1=compareData$rfPred,
result2=compareData$dtPred,
reps=100,
frac=.7)
print(compareResult)
```

Index