Package 'fairness'

September 27, 2019

Title Algorithmic Fairness Metrics

Version 1.0.1

Maintainer Nikita Kozodoi <nikita.kozodoi@hu-berlin.de>

Description

Offers various metrics of algorithmic fairness. Fairness in machine learning is an emerging topic with the overarching aim to critically assess algorithms (predictive and classification models) whether
their results reinforce existing social biases. While unfair algorithms can propagate such biases and offer
prediction or classification results with a disparate impact on various sensitive subgroups of populations (defined
by sex, gender, ethnicity, religion, income, socioeconomic status, physical or mental disabilities), fair algorithms possess
the underlying foundation that these groups should be treated similarly / should have similar outcomes. The fairness
R package offers the calculation and comparisons of commonly and less commonly used fairness metrics in population
subgroups. These methods are described by Calders and Verwer (2010) <doi:10.1007/s10618-010-0190-x>, Chouldechova
(2017) <doi:10.1089/big.2016.0047>, Feldman et al. (2015) <doi:10.1145/2783258.2783311> , Friedler et al.
(2018) <doi:10.1145/3287560.3287589> and Zafar et al. (2017) <doi:10.1145/3038912.3052660>. The package also
offers convenient visualizations to help understand fairness metrics.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

BugReports https://github.com/kozodoi/Fairness/issues

Depends R (>= 3.5.0)

Imports caret, devtools, e1071, ggplot2, pROC

Suggests testthat, knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Nikita Kozodoi [aut, cre],
Tibor V. Varga [aut] (<https://orcid.org/0000-0002-2383-699X>)

Repository CRAN

Date/Publication 2019-09-27 09:00:18 UTC

R topics documented:

`acc_parity` *Accuracy parity*

Description

This function computes the Accuracy parity metric

Usage

```
acc_parity(data, outcome, group, probs = NULL, preds = NULL,
  outcome_levels = NULL, cutoff = 0.5, base = NULL)
```

Arguments

`data`	The dataframe that contains the necessary columns.
`outcome`	The column name of the actual outcomes.
`group`	Sensitive group to examine.
`probs`	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.

preds	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
outcome_levels	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
cutoff	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
base	Base level for sensitive group comparison

Details

This function computes the Accuracy parity metric as described by Friedler et al., 2018. Accuracy metrics are calculated by the division of correctly predicted observations (the sum of all true positives and true negatives) with the number of all predictions. In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their accuracies are lower or higher compared to the reference group. Lower accuracies will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean WORSE prediction for the subgroup.

Value

Metric	Raw accuracy metrics for all groups and metrics standardized for the base group (accuracy parity metric). Lower values compared to the reference group mean lower accuracies in the selected subgroups
Metric_plot	Bar plot of Accuracy parity metric
Probability_plot	Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
acc_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', preds = NULL, outcome_levels = c('no', 'yes'),
cutoff = 0.4, base = 'Caucasian')
acc_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = NULL, preds = 'predicted', outcome_levels = c('no', 'yes'),
cutoff = 0.5, base = 'Hispanic')
```

---

compas *Modified COMPAS dataset*

Description

compas is a landmark dataset to study algorithmic (un)fairness. This data was used to predict recidivism (whether a criminal will reoffend or not) in the USA. The tool was meant to overcome human biases and offer an algorithmic, fair solution to predict recidivism in a diverse population. However, the algorithm ended up propagating existing social biases and thus, offered an unfair algorithmic solution to the problem. In this dataset, a model to predict recidivism has already been fit and predicted probabilities and predicted status (yes/no) for recidivism have been concatenated to the original data.

Usage

```
compas
```

Format

A data frame with 6172 rows and 9 variables:

Two_yr_Recidivism factor, yes/no for recidivism or no recidivism. This is the outcome or target in this dataset

Number_of_Priors numeric, number of priors, normalized to mean = 0 and standard deviation = 1

Age_Above_FourtyFive factor, yes/no for age above 45 years or not

Age_Below_TwentyFive factor, yes/no for age below 25 years or not

Female factor, female/male for gender

Misdemeanor factor, yes/no for having recorded misdemeanor(s) or not

ethnicity factor, Caucasian, African American, Asian, Hispanic, Native American or Other

probability numeric, predicted probabilities for recidivism, ranges from 0 to 1

predicted numeric, predicted values for recidivism, 0/1 for no/yes

Source

The dataset is downloaded from Kaggle https://www.kaggle.com/danofer/compass and has undergone modifications (e.g. ethnicity was originally encoded using one-hot encoding, number or priors have been normalized, variables have been renamed, prediction model was fit and predicted probabilities and predicted status were concatenated to the original dataset).

---

dem_parity *Demographic parity*

Description

This function computes the Demographic parity metric

Usage

```
dem_parity(data, group, probs = NULL, preds = NULL, cutoff = 0.5,
  base = NULL)
```

Arguments

data	The dataframe that contains the necessary columns.
group	Sensitive group to examine.
probs	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.
preds	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
cutoff	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
base	Base level for sensitive group comparison

Details

This function computes the Demographic parity metric (also known as Statistical Parity, Equal Parity, Equal Acceptance Rate or Independence) as described by Calders and Verwer 2010. Demographic parity is calculated based on the comparison of the absolute number of all positively classified individuals in all subgroups of the data. In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their proportion of positively predicted observations are lower or higher compared to the reference group. Lower proportions will be reflected in numbers lower than 1 in the returned named vector.

Value

Metric	Absolute number of positive classifications for all groups and metrics standardized for the base group (demographic parity metric). Lower values compared to the reference group mean lower number of positively predicted observations in the selected subgroups
Metric_plot	Bar plot of Demographic parity metric
Probability_plot	Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
dem_parity(data = compas, group = 'ethnicity',
probs = 'probability', preds = NULL,
cutoff = 0.4, base = 'Caucasian')
dem_parity(data = compas, group = 'ethnicity',
probs = NULL, preds = 'predicted',
cutoff = 0.5, base = 'Hispanic')
```

`equal_odds` *Equalized Odds*

Description

This function computes the Equalized Odds metric

Usage

```
equal_odds(data, outcome, group, probs = NULL, preds = NULL,
  outcome_levels = NULL, cutoff = 0.5, base = NULL)
```

Arguments

`data`	The dataframe that contains the necessary columns.
`outcome`	The column name of the actual outcomes.
`group`	Sensitive group to examine.
`probs`	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.
`preds`	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
`outcome_levels`	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
`cutoff`	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
`base`	Base level for sensitive group comparison

Details

This function computes the Equalized Odds metric (also known as Equal Opportunity, Positive Rate Parity or Separation). Equalized Odds are calculated by the division of true positives with all positives (irrespective of predicted values). This metrics equals to what is traditionally known as sensitivity. In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their sensitivities are lower or higher compared to the reference group. Lower sensitivities will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean WORSE prediction for the subgroup.

Value

`Metric`	Raw sensitivities for all groups and metrics standardized for the base group (equalized odds parity metric). Lower values compared to the reference group mean lower sensitivities in the selected subgroups
`Metric_plot`	Bar plot of Equalized Odds metric
`Probability_plot`	Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
equal_odds(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', preds = NULL, outcome_levels = c('no', 'yes'),
cutoff = 0.4, base = 'Caucasian')
equal_odds(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = NULL, preds = 'predicted', outcome_levels = c('no', 'yes'),
cutoff = 0.5, base = 'Hispanic')
```

`fairness` *fairness: Algorithmic Fairness Metrics*

Description

The **fairness** package offers various metrics of algorithmic fairness. Fairness in machine learning is an emerging topic with the overarching aim to critically assess algorithms (predictive and classification models) whether their results reinforce existing social biases. While unfair algorithms can propagate such biases and offer prediction or classification results with a disparate impact on various sensitive subgroups of populations (defined by sex, gender, ethnicity, religion, income, socioeconomic status, physical or mental disabilities), fair algorithms possess the underlying foundation that these groups should be treated similarly / should have similar outcomes. The fairness R package offers the calculation and comparisons of commonly and less commonly used fairness metrics in population subgroups. The package also offers convenient visualizations to help understand fairness metrics.

Details

Package:	fairness
Depends:	R (>= 3.5.0)
Type:	Package
Version:	1.0.1
Date:	2019-09-19
License:	MIT
LazyLoad:	Yes

Author(s)

- Nikita Kozodoi <nikita.kozodoi@hu-berlin.de>
- Tibor V. Varga <tirgit@hotmail.com>

See Also

https://github.com/kozodoi/Fairness

fnr_parity *False Negative Rate parity*

Description

This function computes the False Negative Rate (FNR) parity metric

Usage

```
fnr_parity(data, outcome, group, probs = NULL, preds = NULL,
  outcome_levels = NULL, cutoff = 0.5, base = NULL)
```

Arguments

data	The dataframe that contains the necessary columns.
outcome	The column name of the actual outcomes.
group	Sensitive group to examine.
probs	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.
preds	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
outcome_levels	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
cutoff	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
base	Base level for sensitive group comparison

Details

This function computes the False Negative Rate (FNR) parity metric as described by Chouldechova 2017. False negative rates are calculated by the division of false negatives with all positives (irrespective of predicted values). In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their false negative rates are lower or higher compared to the reference group. Lower false negative error rates will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean BETTER prediction for the subgroup.

Value

Metric	Raw false negative rates for all groups and metrics standardized for the base group (false negative rate parity metric). Lower values compared to the reference group mean lower false negative error rates in the selected subgroups
Metric_plot	Bar plot of False Negative Rate parity metric
Probability_plot	Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
fnr_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', preds = NULL, outcome_levels = c('no', 'yes'),
cutoff = 0.4, base = 'Caucasian')
fnr_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = NULL, preds = 'predicted', outcome_levels = c('no', 'yes'),
cutoff = 0.5, base = 'Hispanic')
```

---

fpr_parity *False Positive Rate parity*

Description

This function computes the False Positive Rate (FPR) parity metric

Usage

```
fpr_parity(data, outcome, group, probs = NULL, preds = NULL,
  outcome_levels = NULL, cutoff = 0.5, base = NULL)
```

Arguments

data	The dataframe that contains the necessary columns.
outcome	The column name of the actual outcomes.
group	Sensitive group to examine.
probs	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.
preds	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
outcome_levels	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
cutoff	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
base	Base level for sensitive group comparison

Details

This function computes the False Positive Rate (FPR) parity metric as described by Chouldechova 2017. False positive rates are calculated by the division of false positives with all negatives (irrespective of predicted values). In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their false positive rates are lower or higher compared to the reference group. Lower false positives error rates will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean BETTER prediction for the subgroup.

Value

`Metric` Raw false positive rates for all groups and metrics standardized for the base group (false positive rate parity metric). Lower values compared to the reference group mean lower false positive error rates in the selected subgroups

`Metric_plot` Bar plot of False Positives Rate metric

`Probability_plot`
Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
fpr_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', preds = NULL, outcome_levels = c('no', 'yes'),
cutoff = 0.4, base = 'Caucasian')
fpr_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = NULL, preds = 'predicted', outcome_levels = c('no', 'yes'),
cutoff = 0.5, base = 'Hispanic')
```

---

germancredit *Modified german credit dataset*

Description

germancredit is a credit scoring data set that can be used to study algorithmic (un)fairness. This data was used to predict defaults on consumer loans in the German market. In this dataset, a model to predict default has already been fit and predicted probabilities and predicted status (yes/no) for default have been concatenated to the original data.

Usage

```
germancredit
```

Format

A data frame with 1000 rows and 23 variables:

Account_status factor, status of existing checking account

Duration numeric, loan duration in month

Credit_history factor, previous credit history

Purpose factor, loan purpose

Amount numeric, credit amount

Savings factor, savings account/bonds

Employment factor, present employment since

Installment_rate numeric, installment rate in percentage of disposable income

Guarantors factor, other debtors / guarantors

Resident_since factor, present residence since

Property factor, property

Age numeric, age in years

Other_plans factor, other installment plans

Housing factor, housing

Num_credits numeric, Number of existing credits at this bank

Job factor, job

People_maintenance numeric, number of people being liable to provide maintenance for

Phone factor, telephone

Foreign factor, foreign worker

BAD factor, GOOD/BAD for whether a customer has defaulted on a loan. This is the outcome or target in this dataset

Female factor, female/male for gender

probability numeric, predicted probabilities for default, ranges from 0 to 1

predicted numeric, predicted values for default, 0/1 for no/yes

Source

The dataset has undergone modifications (e.g. categorical variables were encoded, prediction model was fit and predicted probabilities and predicted status were concatenated to the original dataset).

`mcc_parity` *Matthews Correlation Coefficient parity*

Description

This function computes the Matthews Correlation Coefficient (MCC) parity metric

Usage

```
mcc_parity(data, outcome, group, probs = NULL, preds = NULL,
  outcome_levels = NULL, cutoff = 0.5, base = NULL)
```

Arguments

`data`	The dataframe that contains the necessary columns.
`outcome`	The column name of the actual outcomes.
`group`	Sensitive group to examine.
`probs`	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.

preds	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
outcome_levels	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
cutoff	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
base	Base level for sensitive group comparison

Details

This function computes the Matthews Correlation Coefficient (MCC) parity metric. In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their Matthews Correlation Coefficients are lower or higher compared to the reference group. Lower Matthews Correlation Coefficients rates will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean WORSE prediction for the subgroup.

Value

Metric	Raw Matthews Correlation Coefficient metrics for all groups and metrics standardized for the base group (parity metric). Lower values compared to the reference group mean Matthews Correlation Coefficients in the selected subgroups
Metric_plot	Bar plot of Matthews Correlation Coefficient metric
Probability_plot	Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
mcc_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', preds = NULL, outcome_levels = c('no', 'yes'),
cutoff = 0.4, base = 'Caucasian')
mcc_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = NULL, preds = 'predicted', outcome_levels = c('no', 'yes'),
cutoff = 0.5, base = 'Hispanic')
```

---

npv_parity *Negative Predictive Value parity*

Description

This function computes the Negative Predictive Value (NPV) parity metric

Usage

```
npv_parity(data, outcome, group, probs = NULL, preds = NULL,
  outcome_levels = NULL, cutoff = 0.5, base = NULL)
```

Arguments

data	The dataframe that contains the necessary columns.
outcome	The column name of the actual outcomes.
group	Sensitive group to examine.
probs	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.
preds	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
outcome_levels	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
cutoff	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
base	Base level for sensitive group comparison

Details

This function computes the Negative Predictive Value (NPV) parity metric as described by the Aequitas bias toolkit. Negative Predictive Values are calculated by the division of true negatives with all predicted negatives. In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their negative predictive values are lower or higher compared to the reference group. Lower negative predictive values will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean WORSE prediction for the subgroup.

Value

Metric	Raw negative predictive values for all groups and metrics standardized for the base group (negative predictive value parity metric). Lower values compared to the reference group mean lower negative predictive values in the selected subgroups
Metric_plot	Bar plot of Negative Predictive Value metric
Probability_plot	Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
npv_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', preds = NULL, outcome_levels = c('no', 'yes'),
cutoff = 0.4, base = 'Caucasian')
```

```
npv_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = NULL, preds = 'predicted', outcome_levels = c('no', 'yes'),
cutoff = 0.5, base = 'Hispanic')
```

`pred_rate_parity` *Predictive Rate Parity*

Description

This function computes the Predictive Rate Parity metric

Usage

```
pred_rate_parity(data, outcome, group, probs = NULL, preds = NULL,
  outcome_levels = NULL, cutoff = 0.5, base = NULL)
```

Arguments

`data`	The dataframe that contains the necessary columns.
`outcome`	The column name of the actual outcomes.
`group`	Sensitive group to examine.
`probs`	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.
`preds`	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
`outcome_levels`	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
`cutoff`	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
`base`	Base level for sensitive group comparison

Details

This function computes the Predictive Rate Parity metric (also known as Sufficiency) as described by Zafar et al., 2017. Predictive rate parity is calculated by the division of true positives with all observations predicted positives. This metrics equals to what is traditionally known as precision or positive predictive value. In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their precisions are lower or higher compared to the reference group. Lower precisions will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean WORSE prediction for the subgroup.

Value

Metric	Raw precision metrics for all groups and metrics standardized for the base group (predictive rate parity metric). Lower values compared to the reference group mean lower precisions in the selected subgroups
Metric_plot	Bar plot of Predictive Rate Parity metric
Probability_plot	Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
pred_rate_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', preds = NULL, outcome_levels = c('no', 'yes'),
cutoff = 0.4, base = 'Caucasian')
pred_rate_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = NULL, preds = 'predicted', outcome_levels = c('no', 'yes'),
cutoff = 0.5, base = 'Hispanic')
```

---

prop_parity *Proportional parity*

Description

This function computes the Proportional parity metric

Usage

```
prop_parity(data, group, probs = NULL, preds = NULL, cutoff = 0.5,
  base = NULL)
```

Arguments

data	The dataframe that contains the necessary columns.
group	Sensitive group to examine.
probs	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.
preds	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
cutoff	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
base	Base level for sensitive group comparison

Details

This function computes the Proportional parity metric (also known as Impact Parity or Minimizing Disparate Impact) as described by Calders and Verwer 2010. Proportional parity is calculated based on the comparison of the proportion of all positively classified individuals in all subgroups of the data. In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their proportion of positively predicted observations are lower or higher compared to the reference group. Lower proportions will be reflected in numbers lower than 1 in the returned named vector.

Value

`Metric` Raw proportions for all groups and metrics standardized for the base group (proportional parity metric). Lower values compared to the reference group mean lower proportion of positively predicted observations in the selected subgroups

`Metric_plot` Bar plot of Proportional parity metric

`Probability_plot` Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
prop_parity(data = compas, group = 'ethnicity',
probs = 'probability', preds = NULL,
cutoff = 0.4, base = 'Caucasian')
prop_parity(data = compas, group = 'ethnicity',
probs = NULL, preds = 'predicted',
cutoff = 0.5, base = 'Hispanic')
```

---

roc_parity *ROC AUC parity*

Description

This function computes the ROC AUC parity metric

Usage

```
roc_parity(data, outcome, group, probs, outcome_levels = NULL,
  base = NULL)
```

Arguments

data	The dataframe that contains the necessary columns.
outcome	The column name of the actual outcomes.
group	Sensitive group to examine.
probs	The column name or vector of the predicted probabilities (numeric between 0 - 1).
outcome_levels	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
base	Base level for sensitive group comparison

Details

This function computes the ROC AUC values for each subgroup. In the returned table, the reference group will be assigned 1, while all other groups will be assigned values according to whether their ROC AUC values are lower or higher compared to the reference group. Lower ROC AUC will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean WORSE prediction for the subgroup.

Value

Metric	Raw ROC AUC metrics for all groups and metrics standardized for the base group (parity metric). Lower values compared to the reference group mean lower ROC AUC values in the selected subgroups
Metric_plot	Bar plot of ROC AUC metric
Probability_plot	Density plot of predicted probabilities per subgroup
ROCAUC_plot	ROC plots for all subgroups

Examples

```
data(compas)
roc_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', outcome_levels = c('no', 'yes'), base = 'Caucasian')
roc_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', outcome_levels = c('no', 'yes'), base = 'African_American')
```

---

spec_parity *Specificity parity*

Description

This function computes the Specificity parity metric

Usage

```
spec_parity(data, outcome, group, probs = NULL, preds = NULL,
  outcome_levels = NULL, cutoff = 0.5, base = NULL)
```

Arguments

data	The dataframe that contains the necessary columns.
outcome	The column name of the actual outcomes.
group	Sensitive group to examine.
probs	The column name or vector of the predicted probabilities (numeric between 0 - 1). If not defined, argument preds needs to be defined.
preds	The column name or vector of the predicted outcome (categorical outcome). If not defined, argument probs needs to be defined.
outcome_levels	The desired levels of the predicted outcome (categorical outcome). If not defined, all unique values of outcome are used.
cutoff	Cutoff to generate predicted outcomes from predicted probabilities. Default set to 0.5.
base	Base level for sensitive group comparison

Details

This function computes the Specificity parity metric. Specificities are calculated by the division of true negatives with all negatives (irrespective of predicted values). In the returned named vector, the reference group will be assigned 1, while all other groups will be assigned values according to whether their specificities are lower or higher compared to the reference group. Lower specificities will be reflected in numbers lower than 1 in the returned named vector, thus numbers lower than 1 mean WORSE prediction for the subgroup.

Value

Metric	Raw specificity metrics for all groups and metrics standardized for the base group (specificity parity metric). Lower values compared to the reference group mean lower specificities in the selected subgroups
Metric_plot	Bar plot of Specificity parity metric
Probability_plot	Density plot of predicted probabilities per subgroup. Only plotted if probabilities are defined

Examples

```
data(compas)
spec_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = 'probability', preds = NULL, outcome_levels = c('no', 'yes'),
cutoff = 0.4, base = 'Caucasian')
spec_parity(data = compas, outcome = 'Two_yr_Recidivism', group = 'ethnicity',
probs = NULL, preds = 'predicted', outcome_levels = c('no', 'yes'),
cutoff = 0.5, base = 'Hispanic')
```

Index