partial_cm

*Compute the global confusion matrix from the FPR and TPR obtained from each node*

**Description**
Compute the global confusion matrix from the FPR and TPR obtained from each node.

**Usage**
```r
partial_cm(
  fpr,
  tpr,
  thresholds,
  negative_count,
  total_count,
  descending = FALSE
)
```

**Arguments**
- `fpr`: list - False positive rates for each individual ROC
- `tpr`: list - True positive rates for each individual ROC
- `thresholds`: list - Thresholds used to compute the fpr and tpr
- `negative_count`: list - Total number of samples corresponding to the negative case
- `total_count`: list - Total number of samples
- `descending`: thresholds in descending order?

**Value**
global confusion matrix and thresholds

---

precision_recall_curve

*Compute the precision recall curve*

**Description**
Compute the precision recall curve.

**Usage**
```r
precision_recall_curve(fpr, tpr, thresholds, negative_count, total_count)
```
**roc_curve**

Compute Receiver operating characteristic (ROC)

**Usage**

`roc_curve(fpr, tpr, thresholds, negative_count, total_count)`

**Arguments**

- **fpr**  
  list - False positive rates for each individual ROC.

- **tpr**  
  list - True positive rates for each individual ROC.

- **thresholds**  
  list - Thresholds used to compute the fpr and tpr.

- **negative_count**  
  vector - Total number of samples corresponding to the negative case.

- **total_count**  
  vector - Total number of samples.

**Value**

list with the global precision, recall, and thresholds (increasing)
shift_vector  

*Shift a vector left or right according to the value provided*

**Description**

Shift a vector left or right according to the value provided

**Usage**

```r
shift_vector(x, n)
```

**Arguments**

- `x`: the vector
- `n`: shift

**Value**

the vector shifted

**Examples**

```r
shift_vector(c(1,2,3,4), 1)
shift_vector(c(1,2,3,4), -1)
```
Index

partial_cm, 2
precision_recall_curve, 2
roc_curve, 3
shift_vector, 4