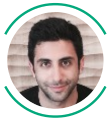
Inter-rater agreement Kappas



Amir Ziai Follow
Sep 24, 2017 · 4 min read

a.k.a. inter-rater reliability or concordance

> *In statistics, inter-rater reliability, inter-rater agreement, or concordance is the degree of agreement among raters. It gives a score of how much homogeneity, or consensus, there is in the ratings given by judges.*

The Kappas covered here are most appropriate for “**nominal**” data. The natural ordering in the data (if any exists) is ignored by these methods. If you’re going to use these metrics make sure you’re aware of the limitations.

Cohen's Kappa

$$\kappa = 1 - \frac{1 - p_o}{1 - p_e}$$

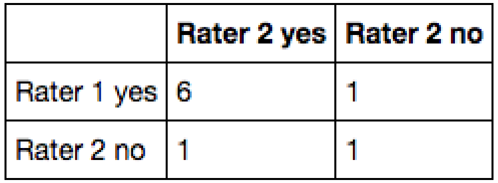
There's two parts to this:

1. Calculate observed agreement
2. Calculate agreement by chance

Let's say we're dealing with "yes" and "no" answers and 2 raters. Here are the ratings:

```
rater1 = ['yes', 'no', 'yes', 'yes', 'yes', 'yes', 'no', 'yes',
'yes']
rater2 = ['yes', 'no', 'no', 'yes', 'yes', 'yes', 'yes', 'yes',
'yes']
```

Turning these ratings into a confusion matrix:



	Rater 2 yes	Rater 2 no
Rater 1 yes	6	1
Rater 2 no	1	1

```
Observed agreement = (6 + 1) / 10 = 0.7
Chance agreement   = probability of randomly saying yes (P_yes) + 
probability of randomly saying no (P_no)
P_yes              = (6 + 1) / 10 * (6 + 1) / 10 = 0.49
P_no               = (1 + 1) / 10 * (1 + 1) / 10 = 0.04
Chance agreement   = 0.49 + 0.04 = 0.53
```

Since the observed agreement is larger than chance agreement we'll get a positive Kappa.

```
kappa = 1 - (1 - 0.7) / (1 - 0.53) = 0.36
```

Or just use `sklearn`'s implementation

```
from sklearn.metrics import cohen_kappa_score
```

```
cohen_kappa_score(rater1, rater2)
```

which returns 0.35714.

Interpretation of Kappa

Special cases

Less than chance agreement

```
rater1 = ['no', 'no', 'no', 'no', 'no', 'yes', 'no', 'no', 'no',
'no']
rater2 = ['yes', 'no', 'no', 'yes', 'yes', 'no', 'yes', 'yes', 'yes',
'yes']
cohen_kappa_score(rater1, rater2)
-0.2121
```

If all the ratings are the same and opposite

This case reliably produces a `kappa` of 0

```
rater1 = ['yes'] * 10
rater2 = ['no'] * 10
cohen_kappa_score(rater1, rater2)
0.0
```

Random ratings

For random ratings `Kappa` follows a normal distribution with a mean of about zero.

As the number of ratings increases there's less variability in the value of Kappa in the distribution.

10 random ratings for each rater (random sample of 1,000 inter-rater Kappa calculations)

100 random ratings for each rater (random sample of 1,000 inter-rater Kappa calculations)

You can find more details here

Note that Cohen’s Kappa only applied to 2 raters rating the exact same items.

. . .

Fleiss

Extends Cohen’s Kappa to more than 2 raters.

Interpretation

> It can be interpreted as expressing the extent to which the observed amount of agreement among raters exceeds what would be expected if all raters made their ratings completely randomly.

The raters can rate different items whereas for Cohen’s they need to rate the exact same items

> Fleiss’ kappa specifically allows that although there are a fixed number of raters (e.g., three), different items may be rated by different individuals

For example let’s say we have 10 raters, each doing a “yes” or “no” rating on 5 items:

For example let’s say we have 10 raters, each doing a “yes” or “no” rating on 5 items:

For example the first row (P_1):

```
P_1 = (10 ** 2 + 0 ** 2 - 10) / (10 * 9) = 1
```

And the first columns (p_1):

```
p_1 = 34 / (5 * 10) = 0.68
```

Go through the worked example here if this is not clear.

Now you can calculate Kappa:

```
P_bar = (1 / 5) * (1 + 0.64 + 0.8 + 1 + 0.53) = 0.794
P_bar_e = 0.68 ** 2 + 0.32 ** 2 = 0.5648
```

At this point we have everything we need and `kappa` is calculated just as we calculated Cohen's:

```
kappa = (0.794 - 0.5648) / (1 - 0.5648) = 0.53
```

You can find the Jupyter notebook accompanying this post here.

References

- https://www.wikiwand.com/en/Inter-rater_reliability
- https://www.wikiwand.com/en/Fleiss%27_kappa

Statistics Data Science

129 claps

WRITTEN BY

Amir Ziai

Engineer

Follow

Towards Data Science

A Medium publication sharing concepts, ideas, and codes.

Follow

See responses (2)

More From Medium

More from Towards Data Science

Bye-bye Python. Hello Julia!

Rhea Moutafis in Towards Data Science
May 1 · 8 min read ★

9.94K

More from Towards Data Science

Don’t Become a Data Scientist

Chris in Towards Data Science
May 4 · 6 min read ★

6.6K

More from Towards Data Science

Do Not Use “+” to Join Strings in Python

Christopher Tao in Towards Data Science
May 10 · 5 min read ★

1.4K

Discover Medium

Welcome to a place where words matter. On Medium, smart voices and original ideas take center stage - with no ads in sight. Watch

Make Medium yours

Follow all the topics you care about, and we'll deliver the best stories for you to your homepage and inbox. Explore

Become a member

Get unlimited access to the best stories on Medium — and support writers while you're at it. Just $5/month. Upgrade