Train In Data

Chi-square test

Considerations

Chi-square test - Asumptions

Observations are independent: each observation can only be assigned to 1 cell in the contingency table.

Chi-square test - Asumptions

Expected frequencies should be greater than 5.

	Cats	Dogs
Died	210.8	49.19
Surived	89.19	20.81

(Some argue that in big contingency tables up to 20% of cells can be smaller than 5, and it is sort of ok.)

With rare labels, it might be common to have small frequencies ➔ consider grouping categories first.

Fisher exact test

- Chi-square has an approximate $\chi 2$ distribution.
- In large samples, the approximation is good enough.
- In small samples, it is better to use Fisher's exact test.
- Fisher's test is normally used in 2x2 contingency tables.

Sample size effect

When the the sample size is big, even tiny differences in the frequency become significant.

- We will conclude that there is an association between the categorical variable and the target, when perhaps there isn't.
- To rank features it is not important.

Continuous variables

What to do if we also have continuous variables?

- Discretise the continuous variable, and then proceed as if it was categorical
- Beware: the way the intervals are constructed may affect the statistic.

Sklearn's chi test is not the right test!

sklearn.feature_selection.chi2

`sklearn.feature_selection.chi2(X, y)` [source]

Compute chi-squared stats between each non-negative feature and class.

This score can be used to select the n_features features with the highest values for the test chi-squared statistic from X, which must contain only non-negative features such as booleans or frequencies (e.g., term counts in document classification), relative to the classes.

Recall that the chi-square test measures dependence between stochastic variables, so using this function "weeds out" the features that are the most likely to be independent of class and therefore irrelevant for classification.

Read more in the User Guide.

Parameters:	**X :** ***{array-like, sparse matrix} of shape (n_samples, n_features)*** Sample vectors. **y :** ***array-like of shape (n_samples,)*** Target vector (class labels).
Returns:	**chi2 :** ***ndarray of shape (n_features,)*** Chi2 statistics for each feature. **p_values :** ***ndarray of shape (n_features,)*** P-values for each feature.

Train In Data

THANK YOU

www.trainindata.com