

# Data--

## jungle\_chess\_2pcs\_endgame\_rat\_lion

### Exploratory data analysis :

This dataset is part of a collection datasets based on the game "Jungle Chess" (a.k.a. Dou Shou Qi).

It has 47 variables, 5880 records, and the corresponding class label. Out of the 47 variables, 16 of them are numerical and one is

categorical (Class) which has categories 'w', 'b', etc and remaining are nominal (object) variables. All the numerical columns are

analyzed using the describe() function. The describe() function gives us the count, mean, standard deviation, median, max, min, etc.

### Normalization:-

- Only the numerical columns are normalized in the range of 0 to 1.
- The numerical columns are extracted to a new dataframe.
- Imported **sklearn** library and from that used the **preprocessing** package.
- Using the **MinMaxScaler()** in preprocessing package, normalized the
- numerical data in the range of 0 to 1.
- The 'Class' variable is added back into this dataframe so as to perform one-hot encoding.

## One-hot encoding:-

- One-hot encoding is performed on the categorical variable which is 'class'.
- It can be only done on numerical data types. Since the categories in 'class' variable are numerical, one-hot encoding can be performed directly on it.
- From *sklearn.preprocessing* imported *OneHotEncoder*.
- After encoding, the category names become column names and new columns are formed. Since there 4 categories in 'class' variable, 4 new columns are formed.
- This is in the form of an array. So, it is converted into a dataframe.

Before using logistic regression or neural networks, the data has to be split into train & test data. To do this task, *train\_test\_split* package from *sklearn.model\_selection* is used. The test size is 20% of the data. The arguments for *train\_test\_split* are the numerical variables, target variable ('Class'), test size (0.2), *random\_state*, *stratify*.

Since the values in the 'Class' variable are imbalanced, *stratify* is used to take care of that problem.

## Logistic Regression

- Imported ***LogisticRegression*** package from ***sklearn.linear\_model*** library.
- Imported ***accuracy\_score*** package from ***sklearn.metrics*** library.
- Ran the logistic regression model.
- Accuracy was 0.964285. This model itself is sufficient and accurate.