**Lesson 02**

**CSC357 Machine Learning**

**16 January 2020**

**Data Transforming**

**Instructor:** **Leon Tabak**

**Authors:** **Alex Heisdorffer, Chenxi Liu, Yudong Zhang, Linda Wang**

**Problems:**

* When we exploring the dataset which has a lot of missing values? How to deal with this problem?
* For a dataset, there will be some numeric attributes, categorical attributes. How to handle text and categorical attribute of a dataset before feeding a model?

**Filling the missing values:**

**Intro:** We are able to use the imputer class to fill the miss values with the median, mean, mode of each numeric attribute by changing the parameter of strategy to “median”, “mean”, “most\_frequent”.

 **Example:**

 **from sklearn.preprocessing import Imputer**

 **imputer = Imputer ( strategy = “median” )**

 **imputer.fit ( housing\_num )**

 **filled\_array = imputer.transform ( housing\_num )**

 **filled\_housing\_num = pd.DataFrame ( filled\_array,**

**columns = housing\_num.columns)**

**Handling the text and categorical attributes:**

**Intro:** we map different classes of the categorical attribute to number (0, 1, 2, 3), or we can directly map different classes to one-hot vectors.

 **Example:**

 **from sklearn.preprocessing import OrdinalEncoder**

**encoder = OrdinalEncoder ()**

**housing\_cat\_encoded = encoder.fit\_transform ( housing[[ “ocean\_proximity” ]] )**

 **from sklearn.preprocessing import OneHotEncoder**

 **encoder2 = OneHotEncoder ( sparse = False )**

 **housing\_cat\_one\_hot = encoder2.fit\_transform (**

**housing[[ “ocean\_proximity” ]] )**

**References:**

* One hot encoding:

 <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OneHotEncoder.html>

* Label encoding:

<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html>