ML Practices

Dr. Kalidas Y., IIT Tirupati

By the end of this lecture you will understand 10 key concepts in ML practices

135) key phrase… “Pre-processing”

- Programs to “clean the data”
- Programs to “prepare data”

- Before actually creating data set for machine learning

136) key phrase… “Post-processing”

- Programs to take the output and generate refined outputs
- Can take outputs of several classifiers and apply rules
- Programs that are important after classifier has done its job
- Data reorganization, Preparation for further steps

137) key phrase… “Noise”

- **Noise: Unwanted Pattern**
- Conceptual extension to noise in signal processing domain…

- How to handle this?

- Pre-processing, hand crafted rules to get rid of noise.

- What is a rule?

138) key phrase… “Data Cleaning”

- Eliminating Noise in Data

139) key phrase… “Noise Elimination Rules”

- What is a hand crafted rule?
- It is a pre-processing methodology

- Noise Elimination Rule
  - Define Positive (data that you don’t want!)
  - Define Negative (data that you want)
  - These are formulated by a team upon mutual agreement!

- A rule can be thought of as a Predictor with 100% Precision!

- Whatever the rule marks as noise, that is it! Remove those points. You trust it 100%.

- “Get rid of noise” as much as possibly by “Elimination rules” and “clean data set”

140) key phrase… “Label Corruption”

- Target labels are wrong in a classification data set

141) key phrase… “Missing Values”

- Delete rows having missing values

OR

- Impute

142) key phrase… “Missing Value Imputation”

- Simple imputation
  - Mean value (for numeric data)
  - Mode value (highly frequent category e.g. ‘cat’)
  - Constant value (e.g. fill 0)

- Multi variate imputation – Build a classifier *to predict missing value* for each set of the missing values

- Nearest Neighbour imputation – Determine *neighbours* based on distance metric (Kd-tree) and apply *rules* to impute

143) key phrase… “Class Imbalance”

- Positive class is very less
  - If a classifier blindly predicts everything as negative, accuracy = almost 100%
  - Recall will be 0%
- Negative class is very less
  - If a classifier blindly predicts everything as positive, accuracy = almost 100%
  - False Positive Rate will be 100%

- Methods to overcome
  - Over sample minority class
  - Down sample majority class
  - Generate synthetic data points – Hand engineered rules

144) key phrase… “Cross Validated Grid Search”

- K Fold Cross Validation (CV)
  - STEP 1: Shuffle Data set
  - STEP 2: Split Data set → Training Set and Test Set
  - STEP 3: Use Training Set to perform Cross Validation
    - Split Training Data Set into K chunks
    - FOREACH Chunk as Validation set, Use other K-1 Chunks as Learning Set
      - Calculate Metric of interest (e.g. AUC, Accuracy, RMSD, R2 score etc.)
    - Compute Average Metric
  - *Boot strapped sampling* – ensure each of the K subsets, has equal proportion of points from each of the classes (in a classification setting)
- Leave One Out (LOO) Cross Validation
  - Same as above, however validation is just 1 point
  - Do model building N times as the number of data points in the Training Set
- Grid Search CV
  - Parameter Grid
  - For each value of parameter (e.g. tree depth, kernel function to use, alpha value etc.)
    - Perform Cross Validation → Compute Average Metric of interest
  - Determine best parameters and report

145) key phrase… “Feature Engineering”

- Human observations
- Convert those observations to Rules and code
- Add the output of those codes as features

- E.g. presence of bright light in a photograph may indicate it is a sunny day, write a program to do that!
- E.g. presence of sharp noise in an audio file, may indicate location as outdoor
- E.g. presence of high rate of activities on a credit card in a short time

146) key phrase… “Model Maintenance”