Naive Bayes Classifier: Design

Lê Hồng Phương
phuonglh@gmail.com
Hanoi University of Science
Outline

• General design:
  – interface and abstract classes
  – utilities

• Your work
Datum

- **Datum** represents an example in the data \((x, y)\), where
  - \(x = (x_1, ..., x_D)\), \(x_j\) is a real value;
  - \(y\) is the label (or class) of \(x\).

- **SparseVector** is a data structure which maps a collection of features (integers) to their corresponding values, e.g.:
  - \(1 \rightarrow x_1\)
  - \(2 \rightarrow x_2\)
  - \(D \rightarrow x_D\)
AbstractClassifier

• An abstract class representing a general classifier.
• Main methods:
  – `scoresOf(x)`: computes the scores of labels, that is
    • `scoresOf(x) \sim P(y | x)`, for all `y`.
  – `classOf(x)`: finds the most probable label of a datum `x`;
    • `classOf(x) = \arg\max_y P(y | x)`
  – `computeAccuracy(Iterator<Datum>)` computes the accuracy of the classifier on a data set.
  – `print(PrintStream)` prints out the parameters of the classifier
    • should be implemented by concrete classifiers since different classifiers have different parameters.
IClassifierFactory

- **IClassifierFactory** creates classifiers from training data.
  - Training data is a list of **Datum** objects.
  - `create(List<Datum>)` returns an **AbstractClassifier** object.
Your Work

• Implement the naïve Bayes classifier:
  – Add the class `NaiveBayesClassifierFactory` which implements `IClassifierFactory`.
    • Implement the method `create(double[][] data,...)` using the algorithm for training a naïve Bayes classifier.
  – Add the class `NaiveBayesClassifier` which extends `AbstractClassifier`
    • Implement the method `scoresOf(Datum x)` of the naïve Bayes classifier. (Note that `x` is a binary vector)
Your Work

• Write a class **NaiveBayesTester** which
  – Train a naïve Bayes classifier on a training set
  – Test the classifier on a test set and report its accuracy

• Provided utility classes for ease of implementation:
  – **NaiveBayesIO**: input/output a NB classifier.
  – **RealValuedDataReader**: reads data set containing real-valued features into Datum objects