Machine Learning
Lecture 1

Lecture by Dr. Shai Fine
Typeset by Steven Karas

2017-10-22
Last edited 20:55:02 2017-10-22

Disclaimer   These lecture notes are based on the lecture for the course Machine Learning, taught by Dr. Shai Fine at IDC Herzliyah in the fall semester of 2017/2018. Sections may be based on the lecture slides written by Dr. Shai Fine.

Agenda

- Admin
- Course Overview
- First Homework

1   Admin

1.1   Homework

5 homeworks. All must be submitted. Only the first exercise is to be submitted individually. Practical exercises can be submitted in pairs.

Practical exercises include a data set, and will be implemented in python 2.7. Homeworks are usually due one week after being assigned.

Each exercise will typically include 10 bonus points.

50% of final grade.

1.2   Exam

50% of final grade.

2   What is Learning?

Many definitions. Conversion of data into knowledge.

Formal definition of machine learning: A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E[1].

For most machine learning tasks, we take in raw data and output a model.

Most tasks are either used for value prediction (regression) or type prediction (classification).

Another task is recommendations, where the model is applied to each user individually and outputs recommended items in the corpus.

---

[1]Tom Mitchell (1999)

3 Concept Learning

An instance is a representation of a particular entity from the application domain. Also referred to as a feature vector, as a vector of numeric or ordinal values in the application domain (also called the feature space).

A concept is a set of instances. Alternatively, a predicate function over U that defines membership.

A hypothesis is a mapping from feature vectors to labels as a conjunction of constraints on features.

A formal definition of a consistent hypothesis was given verbally, but I don't recall it here.

A training set is a set of labeled examples: a set of feature vectors and their associated label values.

The version space is the set of all hypothetical concepts that are consistent with the training set.

Concrete Example Let *EnjoySport* be the set of days on which my friend Aldo enjoys his favorite water sport.

Our task is to predict the value of *EnjoySport* for an arbitrary day.

Training set:

Sky	Air Temp	Humidity	Wind	Water	Forecast	EnjoySport
Sunny	Warm	Normal	Strong	Warm	Same	yes
Sunny	Warm	High	Strong	Warm	Same	yes
Rainy	Cold	High	Strong	Warm	Change	no
Sunny	Warm	High	Strong	Cold	Change	yes

Assuming that the value of Sky can be either Sunny, Rainy, or Cloudy, whereas all the other features are binary, our feature space has cardinality $3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 = 96$.

This gives us a hypothesis space of $1 + (4 \cdot 3 \cdot 3 \cdot 3 \cdot 3 \cdot 3) = 973$, including the null hypothesis of all negative instances, and the apathy operator ? for each feature value.

3.1 Find-S

Finds the maximally specific hypothesis h.[2]

Algorithm Start with the most specific hypothesis in H. Generalize this hypothesis each time it fails to include a positive example.

Correctness Assuming that H contains a hypothesis describing the true concept c, and the training set does not contain any errors; Find-S outputs a consistent hypothesis.

Note that Find-S simply ignores every negative training example. There is a formal proof that this does not affect correctness on slide 43.

3.2 List-Then-Eliminate

Checks all possible hypotheses.

Algorithm Initialize the hypothesis space to H. Eliminate every hypothesis that is not consistent with any training example. Output a list of hypotheses in the version space.

3.3 Candidate-Elimination

Finds the version space that efficiently contains all consistent hypotheses, by exploiting the partial order of the hypothesis space.

Compact representation Rather than enumerating all the possible hypotheses, we can start from the most specific and most generic hypotheses, and define the version space as all the hypotheses in between the two.

---

[2]He claimed that the ordering of training examples affects the intermediate values, and may affect the final hypothesis

Algorithm Initialize G to be the set of most generic hypotheses in H. Initialize S to be the set of most specific hypotheses in H.

For each positive training vector d, generalize S. For each hypothesis $g \in G$ that is inconsistent with d, remove g from G, and add to G all the minimal specializations h of g such that h is consistent with d and some member of S is more specific than h. Remove from G any hypothesis that is more specific than any other h in G. Remove from S any hypothesis that is inconsistent with d.

For each negative training vector d, specialize G. For each hypothesis $s \in S$ that is inconsistent with d, remove s from S, and add to S all the minimal generalizations h of s such that h is consistent with d and some member of G is more generic than h. Remove from S any hypothesis that is more generic than any other h in S. Remove from G any hypothesis that is inconsistent with d.

Properties Assuming there are no errors in the training set, and there is some hypothesis h that correctly describes the target concept c; Candidate-Elimination will converge on the correct hypothesis.

The learned version space is independent of the order of training examples.

Active Learning The convergence speed is affected by the ordering of training examples. Optimal speed is provably $O(\log(|VS|))$ by selecting training examples that effectively halve the version space in each iteration.

The algorithm can construct a training example to request such that convergence is optimal.

4   Next week

1. Python Overview

2. Statistical Learning