September 12, 2017 Lecture 1, Sep. 6, 2017

Introduction and Overview

Lecturer: Justin Domke *Scribe: Boya Ren*

1 Summary

In the first lecture we will briefly introduce the major contents of this course and some administrative information. Reading will be Chapter 1 and Chapter 2.1-2.6 of "The Elements of Statistical Learning" by T. Hastie, R. Tibshirani, and J. H. Friedman. Link to the book can be found on the course homepage: https://people.cs.umass.edu/ domke/courses/compsci589/. It should be noted that the goal of this course is to use machine learning methods, not to derive new algorithms.

2 Definition of Machine Learning

A classical definition of machine learning (ML) is given by Mitchell (1997). "A computer program is said to learn from experience E with respect to task T and performance measure P, if its performance at T as measured by P improves with E".

However, I would define ML as the following: ML is what the ML community does. It doesn't seem to give any information, but that is how ML grows and goes into various fields in the past decades.

3 Five Major Models in the Course

Here we concisely describe the five types of models we will cover in this course, i.e., regression, classification, kernel methods, Bayesian methods, and unsupervised learning.

3.1 Regression

First let's look at an example. We have some data points with input $x = (math, python)$ indicating a student's math and python skills, and output $y = grade$ indication his final grade in machine learning. We have nine points as shown in Fig. 3.1.

The goal of regression is to find some function $f(math, python)$ such that for new students, we have $grade \approx f(math, python)$. So how to do this?

The simplest way is linear regression, which aims to find β_0, β_1 and β_2 such that $f(math, python) = \beta_0 + \beta_1 math + \beta_2 python$. Another basic but widely used model is nearest neighbors. Let $f(math, python) = y_i$, where i is the closest data point. Also trees, neural networks etc. can also be applied to the regression task.

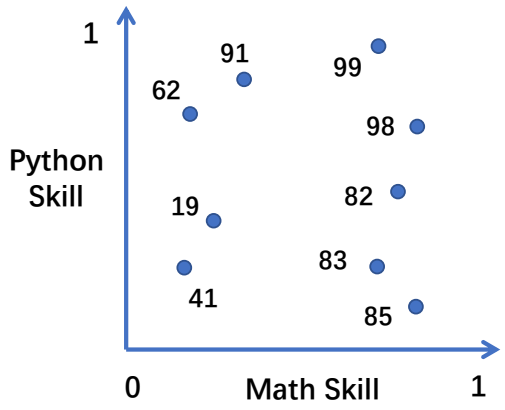


Figure 3.1: An Example of Regression

3.2 classification

Classification is similar to regression in many ways, except that it tries to predict a class label instead of a number. Take the following example as shown in Fig. 3.2. Still we input $x = (math, python)$, but the output becomes the handedness of a student, i.e. $y \in \{R, L\}$ standing for right and left handedness.

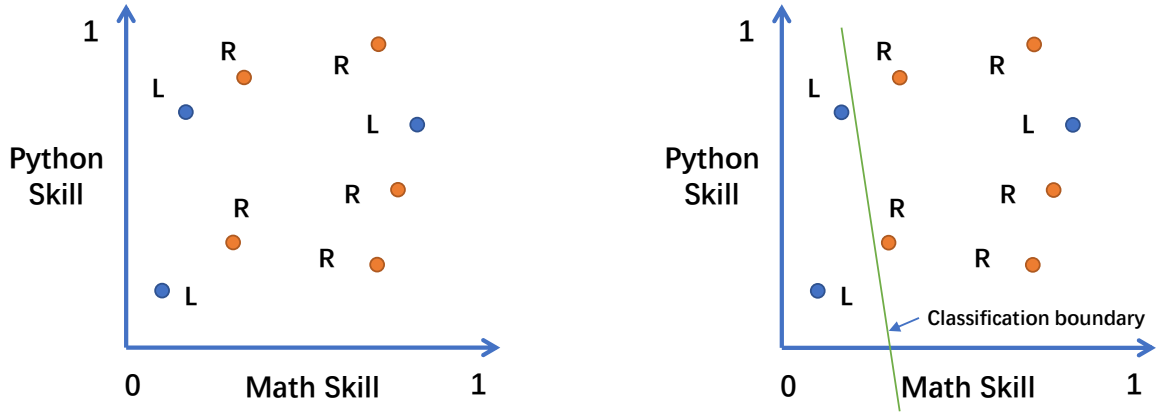


Figure 3.2: An Example of Classification

Our goal of classification is to find some function $f(math, python)$ such that for new students when $f > 0$, $hand \approx R$, and when $f \leq 0$, $hand \approx L$.

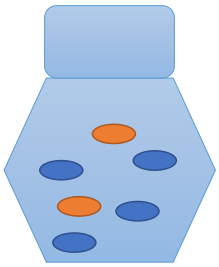
Usually, most models for regression can be slightly adjusted and be applied to classification. First we can use linear classification by constructing the model as $f(math, python) = \beta_0 + \beta_1 math + \beta_2 python$. As shown in Fig. 3.2, the two classes can be decided by the green line. It should be noted that one sample is misclassified. Also nearest neighbors classification can be applied as $f(math, python) = y_i$ where i is the closest point.

3.3 Kernel Methods

Nearest Neighbors method is based on how "close" the points are, while linear method is based on inner product. Kernel methods, however, unifies these two. More details will be covered in the subsequent lectures.

3.4 Bayesian Methods

We will explain the idea of Bayesian methods with the following example. Suppose in a jar we have 100 coins in total. 90 of the coins are of type A, which has a 90% chance of head at each flip, while the rest 10 are of type B, which has 10% chance to be head at each flip. Suppose we drew a coin and flipped, getting head. What's the probability that the coin is type A?



As we observed the coin got a head, we aim to compute the probability that the coin is type A given head. It can be computed using the Bayesian rule as below.

$$\begin{aligned} p(A|H) &= \frac{p(A)p(H|A)}{p(H)} = \frac{p(A)p(H|A)}{p(A)p(H|A)+p(B)p(H|B)} \\ &= \frac{0.9\times 0.9}{0.9\times 0.9+0.1\times 0.1} = \frac{81}{82} \end{aligned}$$

The philosophy/recipe of Bayesian method can be concluded as below.

- Assume "priors" over possible worlds
- Assume probabilities of observations given any world
- Observe data, and use Bayes to get a "posterior" over the true world

3.5 Unsupervised Learning

Unsupervised learning is learning through data with only features but no label. For example, we still have input $x = (math, python)$ but output is unknown, i.e., $y =$. The data points are shown in Fig. 3.3.

The goal of unsupervised learning is to better "understand" the data. More specifically, we want to find cluster m_1, m_2, and m_3, such that each x_i is close m_j. Intuitively, the data points can be clustered as in Fig. 3.3.

A specially interesting topic in unsupervised learning is dimensionality reduction, which aims to preserve certain structures of data with fewer dimensions. There are several benefits to do so, including computation, visualization, and some statistical benefits, such as preventing overfitting.

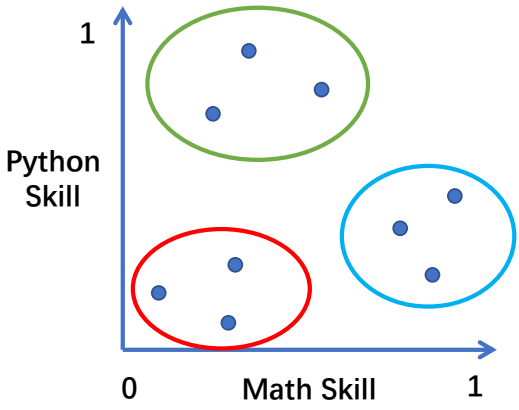


Figure 3.3: An Example of Unsupervised Learning

4 Administrative Information

4.1 Grades

The final grade consists of 50% homework, 30% final, 15% quiz and 5% participation. Participation is mostly evaluated through question-answering on Piazza. There will be 5 assignments, each including both programming and math. In total you have 5 free late days for the assignments. Also there will be Kaggle competitions. To finish the programming tasks, python in a standard environment is required.

4.2 Required Background for the Course

The basic knowledge and skills required for this course are listed as below.

- Linear algebra
- Probability
- Basic calculus
- Python

Next Monday, Sep 11, there will be a quiz with questions in the following aspects to test your basic knowledge

- Matrix computation: orthogonal, inverse, eigenvalue, etc.
- Probability: expected value, covariance, Bayes, etc.
- Calculus: derivatives, integrals, etc.

4.3 Some Advice

- Don't wait till the last minute to start assignments
  - Read the assignments
  - If any question, ask
  - Start work and go to step 2 if you have trouble
- Use office hours well
- Get a physical copy of the textbook
- Ask questions in class

5 Loss Functions

Continuing the examples described above, with regression, we aim to find a model satisfying $grade \approx f(math, python)$. However, it is not clear whether we want $|grade - f(math, python)|$ or $[grade - f(math, python)]^2$ to be small. This is the motivation that we define a loss function $L(Y, f(x))$ as our optimization target. We will go into more details in the next lecture.