

COMP9517 Week 04 Tutorial

Yifan He
Irfan Dwiki Bhaswara

`https://github.com/hharryyf/COMP9517-24T2-tutoring`
`https://github.com/bhaswara/CVLab/`

Outline

- Image Classification Task
- K-Nearest Neighbour
- Decision Tree
- Stochastic Gradient Descent
- Demo

Image Classification

- Task: assign a label to each image in the dataset
- Preprocessing step: represent each image with a set of **features**
 - Example: 28x28 image, 784-d vector, 1 entry/pixel
- Image classifier
 - Calculate the probability of the image belonging to each label based on the **feature** space
 - Pick the label with the highest probability
 - K-Nearest Neighbour, Decision Tree, Stochastic Gradient Descent, Bayesian method, Deep Learning

K-Nearest Neighbour

Method

- 1 Read in the dataset
- 2 Map each training instance to the feature space
- 3 For each testing image, find K images that are the “closest” to it (e.g., Euclidian distance)
- 4 Predict the majority class among the K images

Pros and Cons

- Pro: easy to implement, no training is required
- Cons: hard to pick K, doesn't work well when the feature space is large, high time complexity

Decision Tree

Method

- Each node represents a feature, and each leaf is a label
- Each node has several branches, the instances would be partitioned into different branches of the tree
- Which features to split: largest information gain (Entropy difference between the parent and children)

Pros and Cons

- Pro: easy to implement, robust to outliers
- Cons: tends to overfit

Stochastic Gradient Descent

- Binary classification: create a line (hyperplane) that “best” separates the 2 classes
- SGD is the algorithm that computes the parameters of the line
- Also supports multiclass classification
 - Suppose there are N classes, train N binary classifiers
 - Each classifier would be a class vs. the remaining $N - 1$ classes
 - Each classifier is a K -d hyperplane (K is the number of features)
 - During classification: pick the class such that the hyperplane is the closest to the testing instance

Demo