CPSC 425: Computer Vision

Instructor: Jim Little
little@cs.ubc.ca

Department of Computer Science
University of British Columbia

Lecture Notes 2016/2017 Term 2
Menu March 21, 2017

Topics:
   Classification

Reading:
   Today: Forsyth & Ponce (2nd ed.) 15.1, 15.2
   Next: Forsyth & Ponce (2nd ed.) 16.1.3—16.1.4

Reminders:
   Assignment 6 due Tuesday, Mar. 28
   www: http://www.cs.ubc.ca/~little/cpsc425/
Mollusc sees with its shell

A marine mollusc has hundreds of eyes in its armour that can see images.
Christine Ortiz at the Massachusetts Institute of Technology in Cambridge and her colleagues studied the structural, optical and mechanical properties of the eyes of *Acanthopleura granulata* (pictured) using various experimental and computational techniques. Unlike in most animals, the microscopic lenses are not organic, but are made of the mineral aragonite. These minimize light scattering because they are made of large and aligned crystals. Projecting images through the lenses showed that they could resolve an image of a potential predator of around 20 centimetres in size from about 2 metres away.
The shells are much weaker at these points than elsewhere, but the organism has evolved ways to compensate for the structural weakness, the team found.

Lecture 18: Re-cap

- To use standard clustering techniques we must define an inter-cluster distance measure

- A dendrogram visualizes a hierarchical clustering process

- K-means is a clustering technique that iterates between
  1. Assume the cluster centers are known. Assign each point to the closest cluster center.
  2. Assume the assignment of points to clusters is known. Compute the best cluster center for each cluster (as the mean).

- K-means clustering is initialization dependent and converges to a local minimum
Framework for Today’s Topic

Problem:
Assign new observations into one of a fixed set of categories (classes)

Key Idea(s):
Build a model of data in a given category based on observations of instances in that category

Theory:
Supervised learning

“Gotchas:”
— generalization versus overfitting
A **classifier** is a procedure that accepts as input a set of features and outputs a class **label**.

Classifiers can be binary (face vs. not-face) or multi-class (cat, dog, horse, ...).

We build a classifier using a **training set** of labelled examples \( \{(\mathbf{x}_i, y_i)\} \), where each \( \mathbf{x}_i \) is a feature vector and each \( y_i \) is a class label.

Given a previously unseen observation, we use the classifier to predict its class label.
Example 1: A Classification Problem

- Categorize images of fish — “Atlantic salmon” vs “Pacific salmon”
- Use features such as length, width, lightness, fin shape & number, mouth position, etc.
- Given a previously unobserved image of a salmon, use the learned classifier to guess whether it is an Atlantic or Pacific salmon

Figure credit: Duda & Hart
Bayes Rule (Review and Definitions)

Let $i$ be the class label and let $x$ be the measurement (i.e., evidence)

\[
P(i \mid x) = \frac{P(x \mid i) \cdot P(i)}{P(x)}
\]

- class–conditional probability (aka likelihood)
- posterior probability
- unconditional probability (aka marginal likelihood)
- prior probability
Bayes’ Risk

Some errors may be inevitable: the minimum risk (shaded area) is called the Bayes’ risk

Forsyth & Ponce (2nd ed.) Figure 15.1
Discriminative vs Generative Models

Finding a decision boundary is not the same as modeling a conditional density — while a normal density here is a poor fit to $P(1|x)$, the quality of the classifier depends only on how well the boundary is positioned.

Forsyth & Ponce (2nd ed.) Figure 15.5
Loss Functions in Classifiers

- **Loss**
  - Some errors may be more expensive than others
    - Example: A fatal disease that is easily cured by a cheap medicine with no side-effects. Here, false positives in diagnosis are better than false negatives
  
  - We discuss two class classification:
    - $L(1 \rightarrow 2)$ is the loss caused by calling 1 a 2

- **Total risk** of using classifier $s$ is

  $$R(s) = Pr\{1 \rightarrow 2 \mid \text{using } s\} L(1 \rightarrow 2) + Pr\{2 \rightarrow 1 \mid \text{using } s\} L(2 \rightarrow 1)$$
Two Class Classification

- Generally, we should classify as 1 if the expected loss of classifying as 1 is less than for 2.

- Classify $x$ as

  $1 \quad \text{if} \quad p(1|x)L(1 \rightarrow 2) > p(2|x)L(2 \rightarrow 1)$

  $2 \quad \text{if} \quad p(1|x)L(1 \rightarrow 2) < p(2|x)L(2 \rightarrow 1)$

- Decision boundary: points where the loss is the same for either class.
Training Error, Testing Error, and Overfitting

- Training error is the error a classifier makes on the training set.

- We want to minimize the testing error – the error the classifier makes on an unseen testing set.

- Classifiers that have small training error may not necessarily have small testing error.

- The phenomenon that causes testing error to be worse than training error is called overfitting.
Training Error, Testing Error, and Overfitting

- Underfitting: model is too simple to represent all the relevant class characteristics
- Overfitting: model is too complex and fits irrelevant characteristics (noise) in the data
Cross-Validation

- We cannot reliably estimate the error rate of the classifier using the training set.
- An alternative is to split some training data to form a validation set, then train the classifier on the rest of the data and evaluate on the validation set.
Cross-Validation

- Cross-validation involves performing multiple splits and averaging the error over all splits
When evaluating a multi-class classifier, it may be useful to know how often certain classes are often misclassified as others.

A confusion matrix is a table whose \((i,j)\)th entry is the frequency (or proportion) an item of true class \(i\) was labelled as \(j\) by the classifier.

Forsyth & Ponce (2nd ed.) Figure 15.3. Original credit: H. Zhang et al., 2006.
Receiver Operating Characteristic (ROC)

ROC curves plot trade-off between false positives and false negatives

Forsyth & Ponce (2nd ed.) Figure 15.4

Figure from M. J. Jones and J. Rehg, “Statistical color models with application to skin detection,” Proc. CVPR, ©1999, IEEE
Classifier Strategies

- Classification strategies fall under two broad types: parametric and non-parametric.

- Parametric classifiers are model driven. The parameters of the model are learned from training examples. New data points are classified by the learned model.
  — fast, compact
  — flexibility and accuracy depend on model assumptions
Classifier Strategies

- Classification strategies fall under two broad types: parametric and non-parametric.

- Parametric classifiers are model driven. The parameters of the model are learned from training examples. New data points are classified by the learned model.
  - fast, compact
  - flexibility and accuracy depend on model assumptions

- Non-parametric classifiers are data driven. New data points are classified by comparing to the training examples directly. "The data is the model".
  - slow
  - highly flexible decision boundaries
Nearest Neighbour Classifier

Given a new data point, assign the label of nearest training example in feature space.

Voronoi partitioning of feature space for 2-category 2D and 3D data

Figure credit: Duda, Hart & Stork
k-Nearest Neighbour (kNN) Classifier

- We can gain some robustness to noise by voting over multiple neighbours.

- Given a new data point, find the $k$ nearest training examples. Assign the label by majority vote.

- Simple method that works well if the distance measure correctly weights the various dimensions.

- For large data sets, as $k$ increases kNN approaches optimality in terms of minimizing probability of error.
k-Nearest Neighbour (kNN) Classifier

kNN decision boundaries respond to local clusters where one class dominates

Figure credit: Hastie, Tibshirani & Friedman (2nd ed.)
Support Vector Machines

- Idea: Try to obtain the decision boundary directly

- The decision boundary is parameterized as a separating hyperplane in feature space.
  — e.g. a separating line in 2D
Support Vector Machines

- Idea: Try to obtain the decision boundary directly

- The decision boundary is parameterized as a separating hyperplane in feature space.
  — e.g. a separating line in 2D

- We choose the hyperplane that is as far as possible from each class - that maximizes the distance to the closest point from either class.
Support Vector Machines

Forsyth & Ponce (2nd ed.) Figure 15.6
Example 2: Pedestrian Detection with SVMs

Figure credit: Papageorgiou, Oren, and Poggio, 1998
Summary

- A **classifier** accepts as input a set of features and outputs (predicts) a class label.

- Classifiers need to take into account “loss” associated with each kind of classification error.

- A Receiver Operating Characteristic (ROC) curve plots the trade-off between false negatives and false positives.

- Parametric classifiers are model driven. The parameters of the model are learned from training examples — e.g. support vector machine, decision tree.

- Non-parametric classifiers are data driven. New data points are classified by comparing to the training examples directly — e.g. k-nearest neighbour.