Classification and nearest neighbors

D.A. Forsyth UIUC
Main points

• Classification is important
  • what is it
    • takes a feature vector and makes a label
      • where does fv come from? later
    • examples: credit card, doctor, running new code
  • main types of classification
    • binary/multiclass
    • multiclass requires care
  • evaluation
• key issue in classification
  • want to know how well it works on data where you don’t know labels
  • cause that’s what matters
Main points

• There is no guarantee that classification will be perfect
  • for any given problem
  • example:
    • alien, male female from height

• Ideas
  • bayes risk
    • the very best a classifier can do with a given dataset
    • usually very hard to know
  • the decision boundary
How well does a classifier work?

• Binary classifier
  • (1/0; yes/no; sick/well; etc)
  • accuracy
    • fraction of examples that are classified correctly
  • error rate
    • fraction of examples that are classified incorrectly
  • acc>0.5; e<0.5
• other measures
  • false positive rate (rate at which 0->1)
  • false negative rate (rate at which 1->0)
  • true positive rate (rate at which 1->1)
  • true negative rate (rate at which 0->0)
Setup for classification methods

• dataset \((x, y)\)
  • \(x\) are feature vectors, \(y\) are labels
  • for the moment, feature vectors are vectors (i.e. real numbers in each component, same dimension, etc).

• query \(x\)
  • this is something we want to label

• We must:
  • make a classifier from this dataset
  • make an estimate of how well it will work *on future data*
    • where future data is “like” past data
    • there are some formal guarantees but they’re weak
Nearest neighbors

- **Classifier**
  - a query gets the label of the closest labelled example

- **Classifier Issues**
  - how to measure closest?
  - how to find closest?
  - how to improve?

- **Conceptual issues**
  - if we have enough data, and if we can find the nearest neighbor, could be very good
Nearest neighbors

- Practical issues
  - Reading data (surprisingly important nuisance)
    - errors in data
    - funny formats
    - missing data
  - How do we measure accuracy on future data?
    - split dataset into test and train
      - test data - we pretend we don’t know labels and predict
      - train data - these are the examples
  - key idea
    - if we don’t touch the data when we make the classifier we get an unbiased estimate of accuracy/error rate
• Notice - there are nans in this data, ? in file
  • for now, just drop those data items cause we can’t compute distances
  • but we’ll have to get back to this issue
  • indexing trick with goodflag
  • test - train split
eg - 2

- Notice - funny file format - we have to do some ducking and weaving
  - doesn’t work - what’s happening
    - test - train split
Simple cross-validation

• Issue
  • a simple test-train split fails
  • why not split randomly?

• Notice
  • different accuracies with different splits
  • a big test set gives a poor classifier; a small test set gives an inaccurate estimate
  • => average over different small random splits

• Easiest case
  • repeat
    • choose one data item at random; this is test set
    • evaluate; compute accuracy (0% or 100%)
  • average accuracies over many trials
• Notice:
  • leave one out cross-validation
  • the number of splits is a bit silly
• Notice:
  • data problems (two spaces)
  • look at data - different variables have different scales
    • this could be a real problem
K-NN

• Issue
  • why use only one neighbor? you could use many and vote
  • Advantage:
    • pooling data
  • Disadvantage
    • you have to find the neighbors
    • error rate may go up
• **Notice:**
  • data problems (two spaces)
  • is it better?
Scale and dataset

- Looking at seeds features, some have different scales
- what to do?
  - divide each feature by standard deviation (whitening)
  - can help, but not always
• Notice:
  • is it better?
• Notice:
  • you can whiten knn, too
  • is it better?
Question: which is better?

- The estimate of accuracy is not exact
  - it’s the value of a random variable
    - draw different examples in test split and get different numbers
  - this means
    - relying on one accuracy number is very dangerous
    - you need to know at least standard deviation of accuracy
Question: which is better?

- For most cases, the central limit theorem guarantees
  - estimate of accuracy is value of a normal random variable
  - mean of that normal random variable is true accuracy

- **Standard problem:**
  - given IID samples from each of two normal distributions A, B
  - how strong is the evidence that A has larger mean than B?
  - known as a two-sample z-test
Notice:

- we can get extent of improvement by measuring accuracy mean and std
- current code is a bit clumsy
• Notice:
  • we don’t actually have to do all the work by hand - there are packages
  • scikit-learn
An important feature of NN

- You can predict *any label*
  - or even a number
- Regression:
  - Predicting a number (rather than a label) from a feature vector
  - We’ll see a lot of this later
    - Yacht example