IRDS: Data Mining Process

Charles Sutton
University of Edinburgh

(many figures used from Murphy. *Machine Learning: A Probabilistic Perspective.*)

"Data Science"

- Our working definition
  - Data science is the study of the computational principles, methods, and systems for extracting knowledge from data.
- A relatively new term. A lot of current hype…
  - "If you have to put 'science' in the name…"
- Component areas have a long history
  - machine learning
  - databases
  - statistics
  - optimization
  - natural language processing
  - computer vision
  - speech processing
  - applications to science, business, health….
- Difficult to find another term for this intersection

The term "data mining"

Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarise the data in novel ways that are both understandable and useful to the data owner. — Hand, Mannila, Smyth, 2001

The term "data mining"

Data mining is the analysis of (often large) **observational data sets** to find unsuspected relationships and to summarise the data in novel ways that are both understandable and useful to the data owner. — Hand, Mannila, Smyth, 2001

not collected for the purpose of your analysis

The term “data mining”

Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarise the data **in novel ways** that are both understandable and useful to the data owner. — Hand, Mannila, Smyth, 2001

Many “easy” patterns already known

e.g., pregnant example from association rule mining

The term “data mining”

Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarise the data in novel ways that are both **understandable and useful** to the data owner. — Hand, Mannila, Smyth, 2001

Tradeoff between
- predictive performance
- human interpretability
  Ex: neural networks vs decision trees

Before I get too far ahead of myself…

What problem am I trying to solve?

Problem Types

- Visualization
- Prediction: Learn a map **x** —> y
  - Classification: Predict categorical value
  - Regression: Predict a real value
  - Others
    - Collaborative filtering
    - Learning to rank
    - Structured prediction
- Description
  - Clustering
  - Dimensionality reduction
  - Density estimation
  - Finding patterns
    - Association rule mining
    - Detecting anomalies / outliers

supervised learning

unsupervised learning

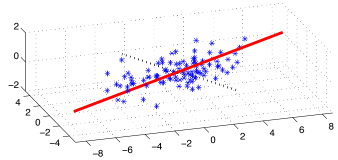
Prediction Examples

- Classification
  - Advertising
    - Ex: Given the text of an online advertisement and a search engine query, predict whether a user will click on the ad
  - Document classification
    - Ex: Spam filtering
  - Object detection
    - Ex: Given an image patch, dose it contain a face?
- Regression
  - Predict the final vote in an election (or referendum) from polls
  - Predict the temperature tomorrow given the previous few days
- Sometimes augmented with other structure / information
  - Structured prediction
    - Spatial data, Time series data
    - Ex: Predicting coding regions in DNA
  - Collaborative filtering (Amazon, Netflix)
  - Semi-supervised learning

Description Examples

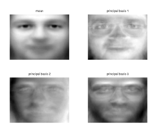
- Clustering
  - Assign data into groups with high intra-group similarity
    - (like classification, except without examples of “correct” group assignments)
  - Ex: Cluster users into groups, based on behaviour
    - Social network analysis
  - Autoclass system (Cheeseman et al. 1988) discovered a new type of star,
- Dimensionality reduction
  - Eigenfaces
  - Topic modelling
- Discovering graph structure
  - Ex: Transcription networks
  - Ex: JamBayes for Seattle traffic jams
- Association rule mining
  - Market basket data
  - Computer security

Dimensionality reduction
General problem

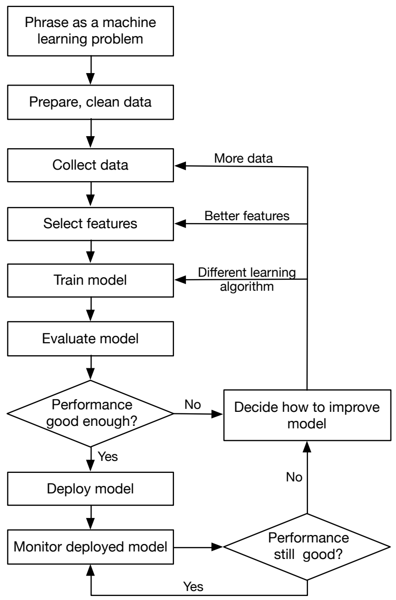


Application to Images
Data

Basis



Data Analysis Process



Inspired by Wagstaff, 2012.
"Machine Learning that Matters"

For another more industrial process, see CRISP-DM.

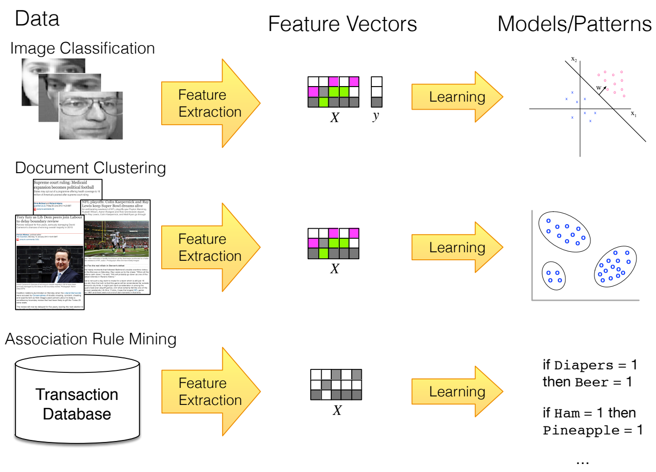
Roadmap

In the next few weeks, we'll talk about

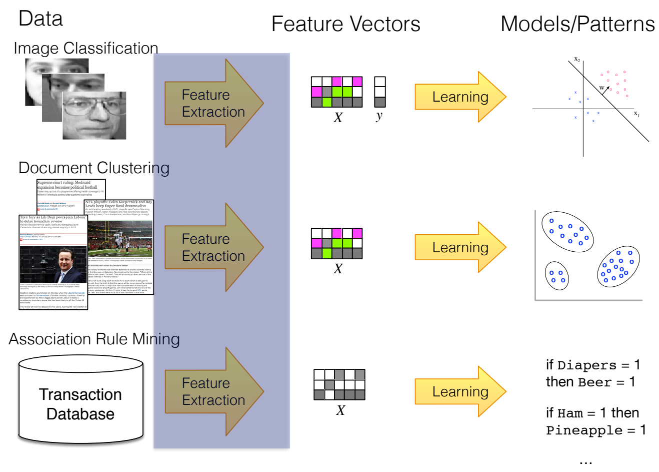
- Visualization
- Feature extraction
- Evaluation and debugging

But to talk about these, we still need to understand **representation** behind the algorithms

Two Representation Problems

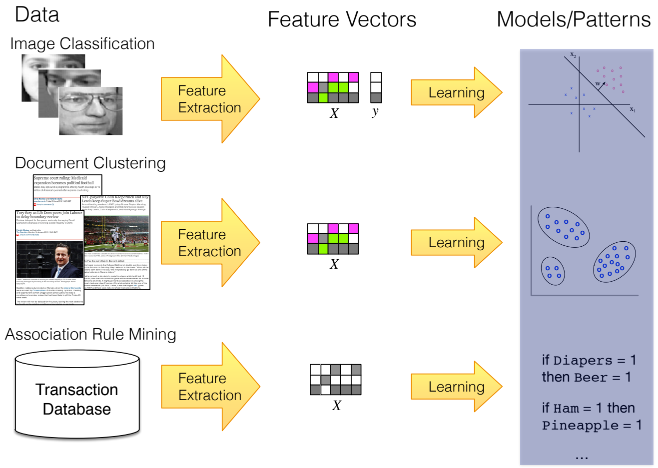


Two Representation Problems



1. Given input, what goes in the feature vector?

Two Representation Problems



2. What is the set of possible models?

Two Representation Problems

1. What features to use
2. What is the space of possible models

- In this course, we discuss features.
- Model —> IAML, PMR, MLPR
- But: To pick features, must understand model.
- So: Whirlwind tour of models, leaving out learning algorithms

Linear regression

Let $\mathbf{x} \in \mathbb{R}^d$ denote the feature vector. Trying to predict $y \in \mathbb{R}$

Simplest choice a linear function. Define parameters $\mathbf{w} \in \mathbb{R}^d$

$$\hat{y} = f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^\top \mathbf{x} = \sum_{j=1}^{d} w_j x_j$$

(to keep notation simple assume that always $x_d = 1$)

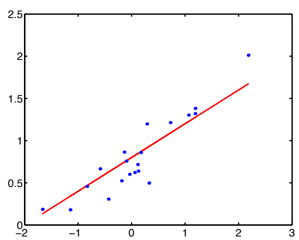
Given a data set

$$\mathbf{x}^{(1)} \dots \mathbf{x}^{(N)}, y^{(1)}, \dots, y^{(N)}$$

find the best parameters

$$\min_{\mathbf{w}} \sum_{i=1}^{N} \left(y^{(i)} - \mathbf{w}^\top \mathbf{x}^{(i)} \right)^2$$

which can be solved easily
(but I won't say how)



Nonlinear regression

What if we want to learn a nonlinear function?

Trick: Define new features, e.g., for scalar x, define $\phi(x) = (1, x, x^2)^\top$

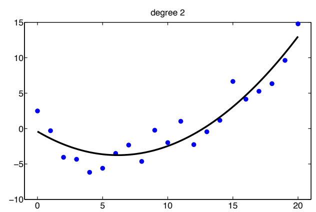
$$\hat{y} = f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^\top \phi(\mathbf{x})$$

this is still linear in $\mathbf{w}$

To find parameters,
the minimisation problem is now

$$\min_{\mathbf{w}} \sum_{i=1}^{N} \left(y^{(i)} - \mathbf{w}^\top \phi(\mathbf{x}^{(i)}) \right)^2$$

exactly the same form as before
(because **x** is fixed)
so still just as easy



Logistic regression

(a classification method, despite the name)

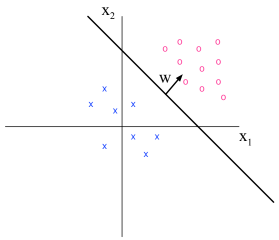
Linear regression was easy.
Can we do linear classification too?

Define a discriminant function

$$f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^\top \mathbf{x}$$

Then predict using

$$y = \begin{cases} 1 & \text{if } f(\mathbf{x}, \mathbf{w}) \geq 0 \\ 0 & \text{otherwise} \end{cases}$$



yields linear decision boundary

Can get class probabilities from this idea, using *logistic regression:*

$$p(y=1|\mathbf{x}) = \frac{1}{1+\exp\{-\mathbf{w}^\top \mathbf{x}\}}$$

(to show decision boundaries same, compute log odds $\log \frac{p(y=1|\mathbf{x})}{p(y=0|\mathbf{x})}$

K-Nearest Neighbour

simple method for classification or regression

Define a distance function between feature vectors $D(\mathbf{x}, \mathbf{x}')$

To classify a new feature vector $\mathbf{x}$

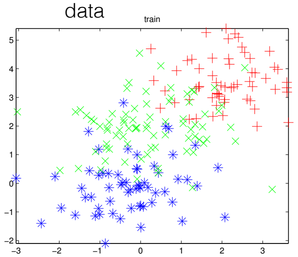
1. Look through your training set. Find the K closest points. Call them $N_K(\mathbf{x})$
   (this is **memory-based** learning.)
2. Return the majority vote.
3. If you want a probability, take the proportion

$$p(y=c|\mathbf{x}) = \frac{1}{K} \sum_{(y', \mathbf{x}') \in N_K(\mathbf{x})} \mathbb{I}\{y' = c\}$$

(the running time of this algorithm is terrible. See IAML for better indexing.)

K-Nearest Neighbour

data

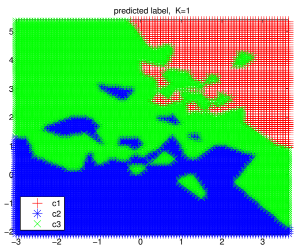


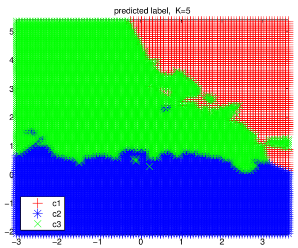
Decision boundaries can be highly nonlinear

The bigger the K, the smoother the boundary

This is **nonparametric**: the complexity of the boundary varies depending on the amount of training data

k=1





k=5

K-means clustering

To split the data into *k* clusters, iterate:

Initialize cluster centroids randomly: $\mu_1 \ldots \mu_K$

Repeat

For each data point *i*

Assign to closest cluster

$$k_i \leftarrow \arg\max_{k \in \{1 \ldots K\}} D(\mu_k, \mathbf{x}^{(i)})$$

Move cluster centroids (average all points currently in cluster)

For all *k* in 1,2,…*K*

$$\mu_k \leftarrow \frac{\sum_{i=1}^N \mathbf{x}^{(i)} \mathbb{I}\{k_i = k\}}{\sum_{i=1}^N \mathbb{I}\{k_i = k\}}$$

Until converged

Results in “convex” clusters

Summary

- Different types of model structures
  1. Linear boundaries (for classification and regression)
  2. Nonlinear boundaries (but linear in a set of features)
  3. “Wavy” boundaries (nonparametric, piecewise linear)
  4. Convex boundaries (with respect to Euclidean distance)
- This will affect feature construction, soon.