Clustering and Classification in Astronomy

Jessi Cisewski
Carnegie Mellon University

Wednesday, June 4, 2014: 10:30 AM - 12 PM
Statistical Learning
Learning from data

1. Unsupervised learning
2. Supervised learning
3. ...
One schematic for addressing problems in machine learning...

classification

SVC Ensemble Classifiers
KNeighbors Classifier
SGD Classifier
Naive Bayes

Text Data
Linear SVC

<100K samples

kernel approximation

>50 samples

more data

regression

SGD Regressor
ElasticNet Lasso
SVR(kernel="rbf") Ensemble Regressors

<100K samples

few features should be important

clustering

Spectral Clustering GMM
KMeans
MiniBatch KMeans
MeanShift VBGMM

<10K samples

<10K samples

tough luck

predicting a category

predicting a quantity

number of categories known

not working

not working

not working

not working

not working

not working

not working

not working

not working

not working

dimensionality reduction

Randomized FCA
Isomap Spectral Embedding
LLE

<10K samples

kernel approximation

not working

not working

not working

not working

not working

Clustering
Find subtypes or groups that are not defined \textit{a priori} based on measurements
\quad \rightarrow \text{“Unsupervised learning” or “Learning without labels”}

Classification
Use \textit{a priori} group labels in analysis to assign new observations to a particular group or class
\quad \rightarrow \text{“Supervised learning” or “Learning with labels”}

* Some content and notation used throughout derived from notes by Rebecca Nugent (CMU), Ryan Tibshirani (CMU), and textbooks Hastie et al. (2009) and James et al. (2013).
Clustering and Classification

data1

data1 with labels
Sample Data

Here is the code for generating the data:

```
ng <- 50
set.seed(321)
g1 <- cbind(rnorm(ng,-3,.1), rnorm(ng,1,1))
g2 <- cbind(rnorm(ng,3,.1), rnorm(ng,1,1))
gtemp <- cbind(rnorm(ng*2,0,.75), rnorm(ng*2,0,.75))
rad1 <- sqrt(gtemp[,1]^2+gtemp[,2]^2)
g3 <- gtemp[order(rad1)[1:ng],]
g4 <- gtemp[order(rad1)[(ng+1):(2*ng)],]

g5.1<-seq(-2.75,2.75, length.out = ng)
g5.2 <- (g5.1/2)^2 - 4
g5 <- cbind(g5.1,g5.2 + rnorm(ng,0,.5))

data1<-rbind(g1,g2,g3,g4,g5)
labels1<-c(rep(1,ng),rep(2,ng),rep(3,ng),rep(4,ng),rep(5,ng))
```

This dataset will be used to illustrate clustering and classification methodologies throughout the lecture.
Good references

- **An Introduction to Statistical Learning** (Hastie et al. 2009)
  → good introduction to get started using methods/not technical
- **The Elements of Statistical Learning** (James et al. 2013)
  → very thorough and technical coverage of statistical learning
- **All of Statistics** (Wasserman 2004)
  → great overview of statistics
CLUSTERING

• Grouping of similar objects (unsupervised learning) → members of the same cluster are “close” in some sense
• Pattern recognition
• Data segmentation

Image: http://www.cfa.harvard.edu/research/hea/massive-cluster-surveys
Clustering in Astronomy

- **Gamma ray bursts**

  → Use properties of GRBs (e.g. location in the sky, arrival time, duration, fluence, spectral hardness) to find subtypes/classes of events

[Image: http://science.hq.nasa.gov Image: Mukherjee et al. (1998)]

**THREE TYPES OF GAMMA-RAY BURSTS**

Soma Mukherjee,1,2,3 Eric D. Feigelson,4 Gutti Jogesh Babu,5 Fionn Murtagh,6,7 Chris Fraley,8 and Adrian Raftery8

Received 1998 February 9; accepted 1998 June 25
Clustering set-up

Notation: given vectors $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\} \in \mathbb{R}^p$

$\rightarrow$ $n$ observations in $p$-dimensional space

$\rightarrow$ Variables/features/attributes indexed by $j = 1, \ldots, p$: $j$th variable is $\mathbf{X}_j$

$\rightarrow$ Observations indexed by $i = 1, \ldots, n$: $i$th variable is $\mathbf{X}_i$

- Want to learn properties about the joint distribution $P(\mathbf{X})$ of these vectors: organize, summarize, categorize, explain

- No direct measure of success (e.g. no notion of a misclassification rate) $\rightarrow$ Successful if true structure is captured...

Image: http://20102250.blogspot.com
General goals of clustering

Partition observations such that

1. Observations within a cluster are similar
   “Compactness Property”

2. Observations in different clusters are non similar
   “Closeness Property”

* Typically want compact clusters that are well-separated *
Dissimilarity Measure

- Characterizes degree of “closeness”

- Dissimilarity matrix $\mathbf{D} = \{d_{ii'}\}$ such that $d_{ii} = 0$ and

  \[ d_{ii'}^j = d(x_{ij}, x_{i'j}) \]

- Some examples of $d_{ii'}^j$ are $(x_{ij} - x_{i'j})^2$ or $|x_{ij} - x_{i'j}|$

- $D_{ii'} = D(X_i, X_{i'}) = \sum_{j=1}^{p} w_j \cdot d_{ii'}^j$, where $\sum_{j=1}^{p} w_j = 1$
Dissimilarity Measure: within cluster variation

\[
\text{Total cluster variability} = \frac{1}{2} \sum_{i=1}^{n} \sum_{i' = 1}^{n} D_{ii'}
\]

\[
= \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i) = k} \left( \sum_{C(i') = k} D_{ii'} + \sum_{C(i') \neq k} D_{ii'} \right)
\]

- where \( C(i) = k \) is the assignment of observation \( i \) to cluster \( k \)
- Total within cluster variability: \( \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i) = k} \sum_{C(i') = k} D_{ii'} \)
- Total between cluster variability: \( \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i) = k} \sum_{C(i') \neq k} D_{ii'} \)
“Specifying an appropriate dissimilarity measure is far more important in obtaining success with clustering than choice of clustering algorithm.” (Hastie et al. 2009)

- Less emphasized in literature because is often domain-specific
Clustering methods

1. Combinatorial algorithms
   - K-means clustering
   - Hierarchical clustering

2. Mixture modeling/Statistical clustering (parametric)

3. Mode seeking/“Bump Hunting”/Statistical clustering (nonparametric)
K-means Clustering

Image: http://www.holehouse.org/mlclass/13_Clustering.html
**Main idea:** partition observations in $K$ separate clusters that do not overlap
K-means clustering - procedure

**Goal:** minimize total within-cluster scatter using
\[ D_{ii'} = \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = \|X_i - X_{i'}\|^2. \]

Then the within-cluster scatter is written as
\[
\frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} \|X_i - X_{i'}\|^2 = \sum_{k=1}^{K} |C_k| \sum_{C(i)=k} \|X_i - \bar{X}_k\|^2
\]

- \(|C_k| = \text{number of observations in cluster } C_k\)
- \(\bar{X}_k = (\bar{X}_1^k, \ldots, \bar{X}_p^k)\)
K-means clustering - recipe

Pick K (number of clusters)
Select K centers
Alternate between the following:

1. Assign each observation to closest center
2. Recalculate centers

R: `kmeans(data, K, nstart = 20)`
set.seed(321)
g1 <- cbind(rnorm(50,0,1), rnorm(50,-1,.1))
g2 <- cbind(rnorm(50,0,1), rnorm(50,1,.1))
g3 <- cbind(rnorm(50,0,1), rnorm(50,3,.1))
data.points <- rbind(g1, g2, g3)
km.out <- kmeans(data.points,3, nstart = 20)
plot(data.points, pch = 16, xlab = "X1", ylab = "X2", col = km.out$cluster)
points(km.out$centers[,1], km.out$centers[,2],pch=4,lwd=4, col="blue", cex = 2)

→ Try moving the clusters closer together (in terms of X2) and see what happens with the K-means clusters.
K-means clustering - determining $K$

- Choose the $k$ that has the last “significant” reduction in the within groups sum of squares (i.e. find the “elbow”)

```r
set.seed(123); wss <- c(); number.k <- c(1:15)
for(ii in number.k){km0 <- kmeans(data1,ii,nstart = 20)
wss[ii] <- km0$tot.withinss}
plot(number.k, wss, xlab="K",ylab="WSS", type = "b", pch = 19)
```
K-means clustering - tips

- Can be unstable; solution depends on starting set of centers
  → finds local optima, but want global optima
  → start with different centers `kmeans(x, K, nstart = 20)`

- Cluster assignments are strict → no notion of degree or strength of cluster membership

- Possible lack of interpretability of centers
  → centers are averages:
    - Fine for clustering things like apartment prices based on price, square footage, distance to nearest grocery store
    - But what if observations are images of faces?

K-means clustering - tips, continued.

- Influenced by outliers
  -→ use medoids - the observation in the data set whose average distance to all other observations is minimal
  -→ medoids is more computationally intensive
  -→ centers are actual observations leading to better interpretability

R: `pam(data, K)` (‘‘partitioning around medoids’’) in cluster package
Hierarchical Clustering
Hierarchical vs. Flat Partitions

1. Flat partitioning (e.g. K-means clustering)
   - partitions data into \( K \) clusters; \( K \) determined by user
   - no sense of the relationships among the clusters

2. Hierarchical partitioning
   - Generates a hierarchy of partitions; user selects the partition
   - \( P_1 = 1 \) cluster, \ldots, \( P_n = n \) clusters (agglomerative clustering)
   - Partition \( P_i \) is the union of one or more clusters from Partition \( P_{i+1} \)
Hierarchical clustering - recipe

Define a dissimilarity $d_{kk'} = d(C_k, C_{k'})$ between clusters $C_k$ and $C_{k'}$ as a function of a distance between points in the clusters

1. Start with every observation in its own cluster
2. Find $\min d(C_k, C_{k'}) \rightarrow$ merge $C_k$ and $C_{k'}$
   (Minimum is across all pairs of clusters)
3. Repeat until only one cluster remains

R: hclust(distances)
Hierarchical clustering - distances

1. Friends-of-friends = Single-linkage clustering: intergroup distance is smallest possible distance

\[ d(C_k, C_{k'}) = \min_{x \in C_k, y \in C_{k'}} d(x, y) \]

2. Complete-linkage clustering: intergroup distance is largest possible distance

\[ d(C_k, C_{k'}) = \max_{x \in C_k, y \in C_{k'}} d(x, y) \]

3. Average-linkage clustering: average intergroup distance

\[ d(C_k, C_{k'}) = \text{Ave}_{x \in C_k, y \in C_{k'}} d(x, y) \]

4. Ward’s clustering

\[ d(C_k, C_{k'}) = \frac{2 \left( |C_k| \cdot |C_{k'}| \right)}{|C_k| + |C_{k'}|} \left\| \bar{X}_{C_k} - \bar{X}_{C_{k'}} \right\|^2 \]
Single-linkage clustering
data1 \( (K = 4) \)
Clustering Recap

1. **Algorithmic clustering** (no statistical assumptions)
   1. K-means
   2. Hierarchical linkage

2. **Statistical clustering**
   1. Parametric - associates a specific model with the density (e.g. Gaussian, Poisson)
      → parameters associated with each cluster
   2. Nonparametric - looks at contours of the density to find cluster information (e.g. kernel density estimate)
Mixture Modeling/Parametric Statistical Clustering

The alignment of molecular cloud magnetic fields with the spiral arms in M33

Hua-bai Li$^1$ & Thomas Henning$^1$

Image: Li and Henning (2011)
Model-based clustering (parametric)

Assumes that each subgroup/cluster/component in the population has its own density

\[ p(X) = \sum_{k=1}^{K} \pi_k p_k(X; \theta_k) \]

where \( \sum_{k=1}^{K} \pi_k = 1 \) and \( 0 \leq \pi_k \leq 1 \).

\[ p(x) = 0.5\phi(x; 4; 1) + 0.5\phi(x; 0; 1) \] (middle plot)
\[ p(x) = 0.75\phi(x; 4; 1) + 0.25\phi(x; 0; 1) \] (right plot)
Suppose there are $K$ clusters - each cluster is modeled by a particular distribution (e.g. a Gaussian distribution with parameters $\mu_k$, $\Sigma_k$)

- The density of each cluster $k$ is

$$p_k(X) = \phi(X \mid \mu_k, \Sigma_k)$$

$$= \frac{1}{\sqrt{(2\pi)^p|\Sigma_k|}} \exp \left( -\frac{(X - \mu_k)^T \Sigma_k^{-1} (X - \mu_k)}{2} \right)$$

Letting $\pi_k$ be the weight of cluster $k$, the mixture density is

$$p(X) = \sum_{k=1}^{K} \pi_k p_k(X) = \sum_{k=1}^{K} \pi_k \phi(X \mid \mu_k, \Sigma_k)$$
Model-based clustering: advantages

1. Well-studied statistical inference techniques available
2. Flexibility in choosing the component distributions
3. Obtain a density estimate for each cluster
4. *Soft* classification is available
Model-based clustering: fitting the model

Suppose there are $K$ clusters - each cluster is modeled by a particular distribution (e.g. a Gaussian distribution with parameters $\mu_k, \Sigma_k$).

**Expectation - Maximization (EM) algorithm**

Finds maximum likelihood estimates in incomplete data (e.g. missing cluster labels).

Alternates between expectation step and maximization steps:

- **E-step**: compute conditional expectation of the cluster labels
- **M-step**: maximize the likelihood and estimate parameters given the current labels; update parameter estimates
Model-based clustering: EM algorithm

- Observations \( \{X_1, X_2, \ldots, X_n\} \) are incomplete (i.e. no labels)
- Complete observations: \( \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)\} \)
  (where \( Y_i \) are the labels)
- The collections of parameters, \( \theta \) are \((\pi_k, \mu_k, \Sigma_k)\) for \( k = 1, \ldots, K \)
- The log-likelihood function is

\[
l(X_1, \ldots, X_n \mid \theta) = \sum_{i=1}^{n} \log \left( \sum_{k=1}^{K} \pi_k \phi(x_i \mid \mu_k, \Sigma_k) \right)
\]

- \( l(X_1, \ldots, X_n \mid \theta) \) is the objective function of the EM algorithm. Numerical difficulty comes from the sum inside the log.
Consider a two-component Gaussian mixture model as follows:

\[
X_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)
\]
\[
X_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)
\]
\[
X = (1 - \Delta) \cdot X_1 + \Delta \cdot X_2
\]

- \(\Delta = 0\) or \(1\), and \(P(\Delta = 1) = \pi\)

Then the density of \(X\) can be written as

\[
p(X) = (1 - \pi) \cdot \phi(X \mid \mu_1, \sigma_1^2) + \pi \cdot \phi(X \mid \mu_2, \sigma_2^2)
\]

with unknown parameters \((\pi, \mu_1, \sigma_1^2, \mu_2, \sigma_2^2)\)

* Example from §8.5.1 of Hastie et al. (2009)
Then the log-likelihood of the sample is

\[
l(X) = \sum_{i=1}^{n} \log \left( (1 - \pi) \cdot \phi(x_i \mid \mu_1, \sigma_1^2) + \pi \cdot \phi(x_i \mid \mu_2, \sigma_2^2) \right)
\]

The EM algorithm proceeds as follows:

1. Initialize \( \hat{\theta} = (\hat{\mu}_1, \hat{\sigma}_1^2, \hat{\mu}_2, \hat{\sigma}_2^2) \)
2. E Step: estimate \( \gamma_i = E(\Delta_i \mid \hat{\theta}) = P(\Delta_i = 1 \mid \hat{\theta}) \)
   \[
   \hat{\gamma}_i = \frac{\hat{\pi} \cdot \phi(x_i \mid \hat{\mu}_2, \hat{\sigma}_2^2)}{(1 - \hat{\pi}) \cdot \phi(x_i \mid \hat{\mu}_1, \hat{\sigma}_1^2) + \hat{\pi} \cdot \phi(x_i \mid \hat{\mu}_2, \hat{\sigma}_2^2)}
   \]
3. M Step: update the estimates of \( (\hat{\pi}, \hat{\mu}_1, \hat{\sigma}_1^2, \hat{\mu}_2, \hat{\sigma}_2^2) \) using \( \hat{\gamma}_i \):
   \[
   \hat{\pi} = \frac{1}{n} \sum_{i=1}^{n} \hat{\gamma}_i,
   \hat{\mu}_1 = \frac{\sum_{i=1}^{n} (1 - \hat{\gamma}_i) x_i}{\sum_{i=1}^{n} (1 - \hat{\gamma}_i)},
   \hat{\mu}_2 = \frac{\sum_{i=1}^{n} \hat{\gamma}_i x_i}{\sum_{i=1}^{n} \hat{\gamma}_i},
   \hat{\sigma}_1^2 = \frac{\sum_{i=1}^{n} (1 - \hat{\gamma}_i) (x_i - \hat{\mu}_1)^2}{\sum_{i=1}^{n} (1 - \hat{\gamma}_i)},
   \hat{\sigma}_2^2 = \frac{\sum_{i=1}^{n} \hat{\gamma}_i (x_i - \hat{\mu}_2)^2}{\sum_{i=1}^{n} \hat{\gamma}_i}
   \]
4. Iterate (2) and (3) until convergence.

* Example from §8.5.1 of Hastie et al. (2009)
Nonparametric Clustering

X-RAY STAR CLUSTERS IN THE CARINA COMPLEX

Eric D. Feigelson\textsuperscript{1}, Konstantin V. Getman\textsuperscript{1}, Leisa K. Townsley\textsuperscript{1}, Patrick S. Broos\textsuperscript{1}, Matthew S. Povich\textsuperscript{1,11}, Gordon P. Garmire\textsuperscript{1}, Robert R. King\textsuperscript{2}, Thierry Montmerle\textsuperscript{3}, Thomas Preibisch\textsuperscript{4}, Nathan Smith\textsuperscript{5,6}, Keivan G. Stassun\textsuperscript{7,8}, Junfeng Wang\textsuperscript{9}, Scott Wolk\textsuperscript{9}, and Hans Zinnecker\textsuperscript{10}

Image: Feigelson et al. (2011)
Nonparametric clustering

- Associate groups with high frequency areas → groups in the population correspond to modes of the density $p(x)$
- Goal: find the modes of the density $p(x)$, or $\hat{p}(x)$. Assign observations to the “domain of attraction” of a mode.
Nonparametric clustering

- Associate groups with high frequency areas $\rightarrow$ groups in the population correspond to modes of the density $p(x)$
- Goal: find the modes of the density $p(x)$, or $\hat{p}(x)$. Assign observations to the “domain of attraction” of a mode.
Nonparametric clustering

- Associate groups with high frequency areas → groups in the population correspond to modes of the density $p(x)$
- Goal: find the modes of the density $p(x)$, or $\hat{p}(x)$. Assign observations to the “domain of attraction” of a mode.

NP clustering is very dependent on the density estimate $\hat{p}(x)$
Final comment on selecting $K$

Background

Just How Many Clusters are there in the Galaxy Data?

- Galaxy Data from Postman et al. (1986): measurements of velocities in $10^3$ km/sec of 82 galaxies from a survey of the Corona Borealis region.

- Roeder (1990): at least 3, no more than 7 modes (Confidence set)

- Others are in consensus

<table>
<thead>
<tr>
<th>9172</th>
<th>9850</th>
<th>9483</th>
<th>9558</th>
<th>9775</th>
<th>10227</th>
</tr>
</thead>
<tbody>
<tr>
<td>10406</td>
<td>16084</td>
<td>16170</td>
<td>18419</td>
<td>18552</td>
<td>18600</td>
</tr>
<tr>
<td>18927</td>
<td>19052</td>
<td>19070</td>
<td>19330</td>
<td>19343</td>
<td>19349</td>
</tr>
<tr>
<td>19440</td>
<td>19473</td>
<td>19529</td>
<td>19541</td>
<td>19547</td>
<td>19663</td>
</tr>
<tr>
<td>19846</td>
<td>19856</td>
<td>19863</td>
<td>19914</td>
<td>19918</td>
<td>19973</td>
</tr>
<tr>
<td>19689</td>
<td>20166</td>
<td>20175</td>
<td>20179</td>
<td>20196</td>
<td>20215</td>
</tr>
<tr>
<td>20221</td>
<td>20415</td>
<td>20620</td>
<td>20795</td>
<td>20821</td>
<td>20846</td>
</tr>
<tr>
<td>20875</td>
<td>20886</td>
<td>21137</td>
<td>21402</td>
<td>21701</td>
<td>21814</td>
</tr>
<tr>
<td>21021</td>
<td>21060</td>
<td>21185</td>
<td>22200</td>
<td>22242</td>
<td>22249</td>
</tr>
<tr>
<td>22314</td>
<td>22374</td>
<td>22495</td>
<td>22746</td>
<td>22747</td>
<td>22888</td>
</tr>
<tr>
<td>22914</td>
<td>23206</td>
<td>23241</td>
<td>23263</td>
<td>23484</td>
<td>23538</td>
</tr>
<tr>
<td>23542</td>
<td>23666</td>
<td>23706</td>
<td>23711</td>
<td>24129</td>
<td>24285</td>
</tr>
<tr>
<td>24289</td>
<td>24366</td>
<td>24717</td>
<td>24990</td>
<td>25633</td>
<td>26960</td>
</tr>
<tr>
<td>26995</td>
<td>32065</td>
<td>32789</td>
<td>34279</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Histogram from Roeder and Wasserman (1997)
Final comment on selecting $K$

**Background**

**Modes/Clusters in the Galaxy Data: The Statistics All-Star Team**

<table>
<thead>
<tr>
<th>Author (Year)</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roeder (1990)</td>
<td>at least 3, no more than 7 (Confidence set)</td>
</tr>
<tr>
<td>Richardson &amp; Green (1997)</td>
<td>6 has highest posterior probability</td>
</tr>
<tr>
<td>Roeder &amp; Wasserman (1997)</td>
<td>The posterior clearly supports three groups</td>
</tr>
<tr>
<td>Lau &amp; Green (2007)</td>
<td>Optimal number of clusters is three</td>
</tr>
<tr>
<td>Wang &amp; Dunson (2011)</td>
<td>Five clusters</td>
</tr>
</tbody>
</table>

► Anyone want to bet that there are more than SEVEN??

*Slide from Ryan Tibshirani’s lecture, which was from George Cassella’s CMU seminar on 1/16/2011.*
CLASSIFICATION

• Build a model, classifier, etc. to separate data into known groups/classes (supervised learning)
• Response variable is not continuous \(\rightarrow\) want to predict labels

1. Bayes classifiers
2. K Nearest Neighbors (KNN) classifiers
3. Logistic regression
4. Linear Discriminant Analysis (LDA)
5. Support Vector Machines (SVM)
6. Tree classifiers
Classification in Astronomy

- Stars can be classified into OBFGKMLTY

<table>
<thead>
<tr>
<th>Class</th>
<th>Temperature</th>
<th>Apparent color</th>
<th>Hydrogen lines</th>
<th>Other noted spectral features</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>≥ 30,000 K</td>
<td>blue</td>
<td>Weak</td>
<td>ionized helium lines</td>
</tr>
<tr>
<td>B</td>
<td>10,000–30,000 K</td>
<td>blue white</td>
<td>Medium</td>
<td>neutral helium</td>
</tr>
<tr>
<td>A</td>
<td>7,500–10,000 K</td>
<td>white to blue white</td>
<td>Strong</td>
<td>ionized calcium (weak)</td>
</tr>
<tr>
<td>F</td>
<td>6,000–7,500 K</td>
<td>white</td>
<td>Medium</td>
<td>ionized calcium (weak)</td>
</tr>
<tr>
<td>G</td>
<td>5,200–6,000 K</td>
<td>yellowish white</td>
<td>Weak</td>
<td>ionized calcium (medium)</td>
</tr>
<tr>
<td>K</td>
<td>3,700–5,200 K</td>
<td>yellow orange</td>
<td>Very weak</td>
<td>ionized calcium (strong)</td>
</tr>
<tr>
<td>M</td>
<td>≤ 3,700 K</td>
<td>orange red</td>
<td>Very weak</td>
<td>Titanium oxide lines</td>
</tr>
</tbody>
</table>

- Classification of Active galactic nuclei (e.g. Starburst, Seyfert (I or II), Quasars, Blazars, BL Lac, OVV, Radio)

- Galaxies can be classified into Hubble morphological types (E/S0/S/Irr), clustering environments, and by star formation activity
Classification set-up

Notation: given vectors \( X = \{X_1, X_2, \ldots, X_n\} \in \mathbb{R}^p \) and class labels \( Y = \{y_1, y_2, \ldots, y_n\} \)

\( \rightarrow \) the \( y_i \)'s are qualitative

\( \rightarrow \) let \( \hat{y}_i \) be the predicted class label for observation \( i \)

\( \rightarrow \) main interest is \( P(Y \mid X) \)

- The classification **training error** rate is often estimated using a training dataset as
  \[
  \frac{1}{n} \sum_{i=1}^{n} I(y_i \neq \hat{y}_i)
  \]

  where \( I(\cdot) \) is the indicator function.

- The classification **test error** rate is often estimated using a test dataset, \((x_{test}, y_{test})\) as
  \[
  E(I(y_{test} \neq \hat{y}_{test}))
  \]

\( \rightarrow \) good classifiers have small test errors
Bayes Classifiers

Test error is minimized by assigning observations with predictors $x$ to the class that has the largest probability:

$$P(Y = j \mid X = x)$$

for classes $j = 1, \ldots, J$

- If there are two classes ($J = 2$), the Bayes decision rule given predictors $x$ is

$$\hat{y}_i = \begin{cases} 
\text{class 1} & \text{if } P(Y = 1 \mid X = x) > 0.50 \\
\text{class 2} & \text{if } P(Y = 2 \mid X = x) > 0.50 
\end{cases}$$

*In general, intractable because the distribution of $Y \mid X$ is unknown.*
K Nearest Neighbors

What is your generic term for a sweetened carbonated beverage?

Image: http://www4.ncsu.edu/~jakatz2/files/dialectposter.png

(KNN was used here, but not in the classification setting)
K Nearest Neighbors (KNN)

**Main idea**: An observation is classified based on the $K$ observations in the training set that are nearest to it.

- A probability of each class can be estimated by

$$P(Y = j \mid X = x) = K^{-1} \sum_{i \in N(x)} I(y_i = j)$$

where $j = 1, \ldots, \#\text{classes in training set}$, and $I = \text{indicator function}$.

- $K = 3$ nearest neighbors to the $X$ are within the circle.
- The predicted class of $X$ would be blue because there are more blue observations than green among the 3 NN.
R: knn(training.set, test.set, training.set.labels, K) in class package
KNN: data1 decision boundary

Using all observations of data1, $K = 1$
Linear Classifiers

- Decision boundary is linear

- Logistic regression
- Linear Discriminant Analysis

Image: http://fouryears.eu/2009/02/
Predicting two groups: binary labels

\[ \begin{align*}
Y &= \begin{cases} 
1 & \text{with probability } \pi \\
0 & \text{with probability } 1 - \pi
\end{cases} \\
\end{align*} \]

- \( E[Y] = P(Y = 1) = \pi \)

We assume the following model for logistic regression

\[ \text{logit}(\pi) = \log \left( \frac{\pi}{1 - \pi} \right) = \beta_0 + \beta^T X \]

where \( \beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p \)
Logistic Regression, continued.

\[
\text{logit}(\pi) = \log \left( \frac{\pi}{1 - \pi} \right) = \beta_0 + \beta^T X
\]

\[
\implies \pi = \frac{e^{\beta_0 + \beta^T X}}{1 + e^{\beta_0 + \beta^T X}}
\]

- Can fit \(\beta_0, \ldots, \beta_p\) via MLE
- \(l(\beta_0, \beta) = \sum_{i=1}^n \log \left( P(Y = y_i \mid X = x_i) \right) \implies \)

\[
(\hat{\beta}_0, \hat{\beta}) = \arg\max_{\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p} \sum_{i=1}^n \left[ y_i \cdot (\beta_0 + \beta^T x_i) - \log(1 + e^{\beta_0 + \beta^T x_i}) \right]
\]
Logistic Regression for data1

```r
logit.labels = matrix(0, nrow = length(labels1))
logit.labels[which(labels1 == 2), which(labels1 == 5)] <- 1
logit.fit <- glm(logit.labels ~ data1[,1]+data1[,2], family = binomial)
summary(logit.fit) #<---provides details of the fit model

logit.fit.probs <- predict(logit.fit, type = "response")
logit.class = matrix(0, nrow = length(logit.fit.probs))
logit.class[logit.fit.probs >= .5] = 1

par(mfrow = c(1,2))
plot(data1, xlim = c(-4,4), ylim = c(-5,4), xlab = "X1", ylab = "X2", col = logit.labels+1, pch = logit.labels+1, lwd = 3, main = "True classes")
plot(data1, xlim = c(-4,4), ylim = c(-5,4), xlab = "X1", ylab = "X2", col = logit.class+1, pch = logit.class+1, lwd = 3, main = "Predicted classes")
```

![True classes](image1)
![Predicted classes](image2)
Logistic Regression for data1: fit model

\[
\text{logit}(\hat{\pi}_*) = \hat{\beta}_0 + \hat{\beta}_1 \cdot x_{1*} + \hat{\beta}_2 \cdot x_{2*} = -1.94 + 1.56 \cdot x_{1*} - 1.50 \cdot x_{2*}
\]
What do we do if our response is not binary (i.e. $J > 2$)?

- There are extensions of logistic regression for multiple classes: Multiple-class logistic regression
- Linear Discriminant Analysis (LDA) is another option
Discriminant Analysis

Goal: estimate a decision boundary that gives a classification rule

- Basic idea: estimate the posterior probabilities of class membership

  If an observation is in a particular location, what is the probability it belongs to a particular class

Bayes’ Rule

\[ P(Y = j' \mid x) = \frac{\pi_{j'} \cdot p_{j'}(x)}{\sum_{j=1}^{J} \pi_j \cdot p_j(x)} \]

- \( \pi_j \) = prior probabilities of class \( j \)
- \( p_j(x) = P(X = x \mid Y = j) \)

  Need a way to estimate \( p_j(x) \) in order to do classification
Example using Bayes’ Rule

Consider $X = 1.9$, $P(Y = 1) =$

\[
\frac{\frac{1}{2} \cdot 0.065}{\frac{1}{2} \cdot 0.065 + \frac{1}{2} \cdot 0.219} \approx 0.229
\]

\[
\frac{\frac{2}{3} \cdot 0.065}{\frac{2}{3} \cdot 0.065 + \frac{1}{3} \cdot 0.172} \approx 0.274
\]
Linear Discriminant Analysis (LDA)

Multivariate Gaussian

\[ p_j(X) = \phi(X \mid \mu_j, \Sigma_j) = \frac{1}{\sqrt{(2\pi)^p|\Sigma_j|}} \exp \left( -\frac{(X - \mu_j)^T \Sigma_j^{-1} (X - \mu_j)}{2} \right) \]

- LDA assumes all covariance matrices are equal \((\Sigma_1 = \cdots = \Sigma_J)\)
A predictor $x$ is classified in the class $j = 1, \ldots, J$ according to the following:

$$\arg\max_j \{ P(Y = j \mid X = x) \} =$$

$$\arg\max_j \{ \phi(x \mid \mu_j, \Sigma) \pi_j \}$$

$$= \arg\max_j \{ \log (\phi(x \mid \mu_j, \Sigma) \pi_j) \}$$

$$\vdots$$

$$= \arg\max_j \left\{ -\frac{1}{2} (x - \mu_j)^T \Sigma^{-1} (x - \mu_j) + \log(\pi_j) \right\}$$

$$= \arg\max_j \left\{ x^T \Sigma^{-1} \mu_j - \frac{1}{2} \mu_j^T \Sigma^{-1} \mu_j + \log(\pi_j) \right\}$$
LDA for data

R: lda(y ~ x)
Quadratic Discriminant Analysis (QDA)

- Assuming a common covariance matrix, \( \Sigma \), is not always reasonable.
- Allowing for different covariance matrices, \( \Sigma_j \), a predictor \( x \) is classified in the class \( j = 1, \ldots, J \) according to the following:

\[
\arg\max_j \{ P(Y = j \mid X = x) \} = \\
\arg\max_j \{ \phi(x \mid \mu_j, \Sigma_j) \pi_j \} \\
= \arg\max_j \{ \log (\phi(x \mid \mu_j, \Sigma_j) \pi_j) \} \\
= \arg\max_j \left\{ -\frac{1}{2} (x - \mu_j)^T \Sigma_j^{-1} (x - \mu_j) + \log(\pi_j) \right\} \\

= \arg\max_j \left\{ -\frac{1}{2} x^T \Sigma_j^{-1} x + x^T \Sigma_j^{-1} \mu_j - \frac{1}{2} \mu_j^T \Sigma_j^{-1} \mu_j + \log(\pi_j) \right\}
\]
QDA for data1

R: qda(y ~ x)
Support Vector Machines and Kd-tree for Separating Quasars from Large Survey Databases

Dan Gao¹,², Yan-Xia Zhang¹* and Yong-Heng Zhao¹

¹National Astronomical Observatorium, Chinese Academy of Sciences, 20A Datun Road, Chaoyang District, 100012, Beijing, P.R.China
²Graduate University of Chinese Academy of Sciences, Beijing, P.R.China
Support Vector Machines (SVM)

- Goal: Find the hyperplane that “best” separates the two classes (i.e. maximize the margin between the classes)

Data: \(\{x_i, y_i\}, \ i = 1, \ldots, n\), and \(y_i = \{-1, 1\}\)

- A separating hyperplane does not always exist
  \[\rightarrow\] incorporate a parameter to penalize misclassifications
Support Vector Machines (SVM), continued.

\[(X_i \cdot w + b) y_i \geq 1 - \xi_i, \quad \xi_i \geq 0 \text{ for } i = 1, \cdots, n\]

\[
\text{minimize} \left( \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \right)
\]

- \(\xi_i\) is capturing the degree to which observation \(i\) is misclassified
- \(C\) is a misclassification penalty

Support Vector Machines (SVM): “kernel trick”

Image: http://ccforum.com/content/11/4/r83/figure/f1
SVM for data1 (linear) - A

R: `svm(x, y, kernel = 'linear', class.weights)` in e1071 package
SVM for data1: fit model (linear) - A
R: `svm(x, y, kernel = 'linear', class.weights)` in e1071 package
SVM for data1: fit model (linear) - B
SVM for data1 (radial basis)

R: svm(x,y,kernel = "radial", class.weights) in e1071 package
SVM for data1: fit model (radial basis)
Classification Trees

![Diagram of Classification Trees](http://astronomy.swin.edu.au)

![Diagram of Classification Trees](http://dame.dsf.unina.it/dame_td.html)
Classification trees

- Goal: determine which variables are “best” at separating observations into the labeled groups

  1. Predictor space is partitioned into hyper-rectangles
  2. Any observations in the hyper-rectangle would be predicted to have the same label
  3. Next split is chosen to maximize “purity” of hyper-rectangles

- Tree-based methods are not typically the best classification methods based on prediction accuracy, but they are often more easily interpreted (James et al. 2013)

- CART = “Classification and Regression Trees”
Classification Tree for data1
Recipe

- Start at the “top” of the tree and use recursive binary splitting to grow the tree
- At each split, determine the classification error rate for the new region (i.e. the number of observations that are not of the majority class in their region)
  → The Gini Index or cross-entropy are better for node purity.

R: tree(y ~ x) in tree package.
Classification Trees - remarks

- **Tree pruning** - the classification tree may be over fit, or too complex; pruning removes portions of the tree that are not useful for the classification goals of the tree.

- **Bootstrap aggregation** (aka “bagging”) - there is a high variance in classification trees, and bagging (averaging over many trees) provides a means for variance reduction. (Boosting is another approach, but grows trees sequentially rather than using a bootstrapped sample.)

- **Random forest** - similar idea to bagging except it incorporates a step that helps to decorrelate the trees.
Concluding Remarks

- Clustering - unsupervised/no labels → find structure
  1. K - means/ K - medics
  2. Agglomerative hierarchical clustering
  3. Parametric/Non-parametric

- Classification - supervised/labels → predict classes
  1. KNN
  2. Logistic regression
  3. LDA/QDA
  4. Support Vector Machines
  5. Tree classifiers

- Clustering and classification are useful tools, but need to be familiar with assumptions that go into the methods


