Classifiers!!!

BCH394P/364C Systems Biology / Bioinformatics
Edward Marcotte, Univ of Texas at Austin

Clustering = task of grouping a set of objects in such a way that objects in the same group (a cluster) are more similar (in some sense) to each other than to those in other groups (clusters).

VS.

Classification = task of categorizing a new observation, on the basis of a training set of data with observations (or instances) whose categories are known

Adapted from Wikipedia
Remember, for clustering, we had a matrix of data...

\[ M \text{ samples} \]

\[
\begin{array}{cccccccc}
\text{Gene 1, sample 1} & \ldots & \text{Gene 1, sample } j & \ldots & \text{Gene 1, sample } M \\
\text{Gene 2, sample 1} & \ldots & \text{Gene 2, sample } j & \ldots & \text{Gene 2, sample } M \\
\vdots & \ldots & \vdots & \ldots & \vdots \\
\text{Gene } i, \text{ sample 1} & \ldots & \text{Gene } i, \text{ sample } j & \ldots & \text{Gene } i, \text{ sample } M \\
\vdots & \ldots & \vdots & \ldots & \vdots \\
\text{Gene } N, \text{ sample 1} & \ldots & \text{Gene } N, \text{ sample } j & \ldots & \text{Gene } N, \text{ sample } M \\
\end{array}
\]

For yeast, \( N \sim 6,000 \)
For human, \( N \sim 22,000 \)

\[ i.e., \text{ a matrix of } N \times M \text{ numbers} \]

We discussed gene expression profiles. Here’s another example of gene features.

\[ M \text{ samples} \]

\[
\begin{array}{cccccccc}
\text{Gene 1, sample 1} & \ldots & \text{Gene 1, sample } j & \ldots & \text{Gene 1, sample } M \\
\text{Gene 2, sample 1} & \ldots & \text{Gene 2, sample } j & \ldots & \text{Gene 2, sample } M \\
\vdots & \ldots & \vdots & \ldots & \vdots \\
\text{Gene } i, \text{ sample 1} & \ldots & \text{Gene } i, \text{ sample } j & \ldots & \text{Gene } i, \text{ sample } M \\
\vdots & \ldots & \vdots & \ldots & \vdots \\
\text{Gene } N, \text{ sample 1} & \ldots & \text{Gene } N, \text{ sample } j & \ldots & \text{Gene } N, \text{ sample } M \\
\end{array}
\]

For yeast, \( N \sim 6,000 \)
For human, \( N \sim 22,000 \)

**Gene expression profiles:** each entry indicates an mRNA’s abundance in a different condition

**Phylogenetic profiles:** each entry indicates whether the gene has homologs in a different organism
This is useful because biological systems tend to be modular and often inherited intact across evolution.

(e.g. you tend to have a flagellum or not)

Many such features are possible...

\( M \) samples

\[ \begin{array}{cccc}
\text{Gene 1, sample 1} & \ldots & \text{Gene 1, sample} M & \text{Gene 1, sample} M \\
\text{Gene 2, sample 1} & \ldots & \text{Gene 2, sample} j & \text{Gene 2, sample} M \\
\text{Gene 3, sample 1} & \ldots & \text{Gene 3, sample} j & \text{Gene 3, sample} M \\
& \ldots & \ldots & \ldots \\
\text{Gene } i, \text{ sample 1} & \ldots & \text{Gene } i, \text{ sample } j & \text{Gene } i, \text{ sample } M \\
& \ldots & \ldots & \ldots \\
\text{Gene } N, \text{ sample 1} & \ldots & \text{Gene } N, \text{ sample } j & \text{Gene } N, \text{ sample } M \\
\end{array} \]

**i.e., a matrix of** \( N \times M \) **numbers**

For yeast, \( N \sim 6,000 \)

For human, \( N \sim 22,000 \)
We also needed a measure of the similarity between feature vectors. Here are a few (of many) common distance measures used in clustering.

<table>
<thead>
<tr>
<th>Names</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Euclidean distance</td>
<td>$|a - b|_2 = \sqrt{\sum_i (a_i - b_i)^2}$</td>
</tr>
<tr>
<td>Manhattan distance</td>
<td>$|a - b|_1 = \sum_i</td>
</tr>
<tr>
<td>cosine similarity</td>
<td>$\frac{a \cdot b}{|a| |b|}$</td>
</tr>
</tbody>
</table>

(credit: Wikipedia)
Clustering refresher: 2-D example

Clustering refresher: hierarchical
Clustering refresher: SOM

Clustering refresher: k-means
Clustering refresher: $k$-means

One of the simplest classifiers uses the same notion of decision boundaries.
One of the simplest classifiers uses this notion of decision boundaries.

Rather than first clustering, calculate the centroid (mean) of objects with each label.

New observations are classified as belonging to the group whose mean is nearest.

="minimum distance classifier"


For example....
Enzyme-based histochemical analyses were introduced in the 1960s to demonstrate that some leukemias were periodic acid-Schiff positive, whereas others were myeloperoxidase positive...

This provided the first basis for classification of acute leukemias into those arising from lymphoid precursors (acute lymphoblastic leukemia, ALL), or from myeloid precursors (acute myeloid leukemia, AML).

"Distinguishing ALL from AML is critical for successful treatment...

Chemotherapy regimens for ALL generally contain corticosteroids, vincristine, methotrexate, and L-asparaginase, whereas most AML regimens rely on a backbone of daunorubicin and cytarabine (8).

Although remissions can be achieved using ALL therapy for AML (and vice versa), cure rates are markedly diminished, and unwarranted toxicities are encountered."
Let’s look at a specific historic example:

\[
c = (1,1,1,1,1,0,0,0,0,0,0)
\]
\[
gene_1 = (e_1, e_2, e_3, \ldots, e_{12})
\]
\[
gene_2 = (e_1, e_2, e_3, \ldots, e_{12})
\]

Take labeled samples, find genes whose abundances separate the samples...

Let’s look at a specific historic example:

Calculate weighted average of indicator genes to assign class of an unknown
PS = \frac{(V_{win} - V_{lose})}{(V_{win} + V_{lose})}, where \( V_{win} \) and \( V_{lose} \) are the vote totals for the winning and losing classes.
Cross-validation

Withhold a sample, build a predictor based only on the remaining samples, and predict the class of the withheld sample.

Repeat this process for each sample, then calculate the cumulative or average error rate.

X-fold cross-validation

e.g. 3-fold or 10-fold

Can also withhold 1/X (e.g. 1/3 or 1/10) of sample, build a predictor based only on the remaining samples, and predict the class of the withheld samples.

Repeat this process X times for each withheld fraction of the sample, then calculate the cumulative or average error rate.
**Independent data**

Withhold an entire dataset, build a predictor based only on the remaining samples (the training data).

Test the trained classifier on the independent test data to give a fully independent measure of performance.

---

You already know how to measure how well these algorithms work (way back in our discussion of gene finding!)

<table>
<thead>
<tr>
<th>Algorithm predicts:</th>
<th>True answer:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>Negative</td>
</tr>
<tr>
<td>True positive</td>
<td>False positive</td>
</tr>
<tr>
<td>False negative</td>
<td>True negative</td>
</tr>
</tbody>
</table>

**Specificity** = $\frac{TP}{TP + FP}$

**Sensitivity** = $\frac{TP}{TP + FN}$
You already know how to measure how well these algorithms work (way back in our discussion of gene finding)...

Sort the data by their classifier score, then step from best to worst and plot the performance:

Precision = \( \frac{TP}{TP + FP} \)
also called positive predictive value (PPV)

Recall = \( \frac{TP}{TP + FN} \)
(= sensitivity)

Another good option:

Sort the data by their classifier score, then step from best to worst and plot the performance:

Sensitivity = \( \frac{TP}{TP + FN} \)
also called True Positive Rate (TPR)

1 - Specificity = \( \frac{FP}{FP + TN} \)
also called False Positive Rate (FPR)

First used in WWII to analyze radar signals (e.g., after attack on Pearl Harbor)
ROC curve, as you go from stronger to weaker predictions

Thanks to Dariya Sydykova (UT Austin), for her excellent visualizations, available here: https://github.com/dariasydykova/open_projects/tree/master/ROC_animation

ROC curve, as you go from stronger to weaker classifiers

Thanks to Dariya Sydykova (UT Austin), for her excellent visualizations, available here: https://github.com/dariasydykova/open_projects/tree/master/ROC_animation
ROC versus Recall/Precision

The 2 measures are related and both useful. They differ strongly in performance as proportions of positive and negative classes change.

Thanks to Dariya Sydykova (UT Austin), for her excellent visualizations, available here: https://github.com/dariyasydykova/open_projects/tree/master/ROC_animation

ROC versus Recall/Precision

- R/P depends strongly on relative rates of the 2 classes
- ROC performance is independent of their relative rates

(It may be important or not for your particular problem...)

Thanks to Dariya Sydykova (UT Austin), for her excellent visualizations, available here: https://github.com/dariyasydykova/open_projects/tree/master/ROC_animation
Back to our minimum distance classifier...

Would it work well for this data?

Back to our minimum distance classifier...

How about this data? What might?
Back to our minimum distance classifier...

How about this data? What might?

For each new object, calculate the $k$ closest data points. Let them vote on the label of the new object.

This is a great case for something called a **$k$-nearest neighbors classifier**:

This one is surrounded by X's and will probably be voted to be an X.

This is surrounded by O's and will probably be voted to be an O.
Back to leukemias. There was a follow-up study in 2010:

- Tested clinical use of expression profiling to subtype leukemias
- Meta-analysis of 11 labs, 3 continents, 3,334 patients
- Stage 1 (2,096 patients):
  - 92.2% classification accuracy for 18 leukemia classes (99.7% median specificity)
- Stage 2 (1,152 patients):
  - 95.6% median sensitivity and 99.8% median specificity for 14 subtypes of acute leukemia
- Microarrays outperformed routine diagnostics in 29 (57%) of 51 discrepant cases

Conclusion: “Gene expression profiling is a robust technology for the diagnosis of hematologic malignancies with high accuracy”

Current commercial breast cancer gene expression panels use this same strategy

<table>
<thead>
<tr>
<th>Gene Signature</th>
<th>Biomarker Sources</th>
<th>Analysis Type</th>
<th>Clinical Outcome</th>
<th>No. Genes</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oncotype DX Breast</td>
<td>Breast tumor tissue</td>
<td>mRNA</td>
<td>Survival, benefit of chemotherapy</td>
<td>21</td>
<td>2004 Paik [83]</td>
</tr>
<tr>
<td>Mammaprint</td>
<td>Breast tumor tissue</td>
<td>mRNA</td>
<td>Survival</td>
<td>70</td>
<td>2002 van’t Veer [83]</td>
</tr>
<tr>
<td>Endopredict</td>
<td>Breast tumor tissue</td>
<td>mRNA</td>
<td>Survival</td>
<td>12</td>
<td>2017 Warf [84]</td>
</tr>
<tr>
<td>Prosigna/PAM50</td>
<td>Breast tumor tissue</td>
<td>mRNA</td>
<td>Survival</td>
<td>50</td>
<td>2009 Parker [85]</td>
</tr>
<tr>
<td>Breast Cancer Index</td>
<td>Breast tumor tissue</td>
<td>mRNA</td>
<td>Survival, benefit of hormone therapy after 5 years</td>
<td>7</td>
<td>2008 Ma, 2013 [86,87]</td>
</tr>
</tbody>
</table>
In practice, if you want to explore classifiers, I also strongly recommend always testing these classifiers:

- **Random forests**
- **Support vector machines (SVM)**

These two are surprisingly often the best for many biological classification problems. Weka can do both of them.

→ Note that I didn’t say neural networks. Deep neural networks can be extremely powerful (e.g. AlphaFold) but are significantly more expert level and require extensive training examples. In general, you’ll often be better off starting off with the above classifiers for many problems, only moving to deep neural networks if you really need to and only when you have data to support it.

The two slide overview of **Random forest classifiers**:
(1) Construct many decision trees from random subsets of your features. Because the features vary across trees, trees tend to be weak but uncorrelated.
(2) All the trees “vote” on the answer, majority wins.

![Random Forest Diagram](https://www.globalsoftwaresupport.com/random-forest-classifier-bagging-machine-learning/)
The two slide overview of **Random forest classifiers**:  
(1) Construct many decision trees from random subsets of your features. Because the features vary across trees, trees tend to be weak but uncorrelated  
(2) All the trees “vote” on the answer, majority wins.

![Random forest classifiers](https://www.researchgate.net/figure/The-Random-forest-classifier-is-an-ensemble-of-decision-trees-where-the-single-trees-are_fig1_228540194)

The one slide overview of **Support vector machines**:  
(1) Goal: make a linear classifier, choosing a decision boundary that *maximizes* the *distance margin* between classes 

![Support vector machines](https://quantdare.com/svm-versus-a-monkey/)

(2) But what if the boundary is non-linear? Use **kernels** to implicitly map the data to higher dimension where a linear decision can be made  

![Maximum margin hyperplane](https://quantdare.com/svm-versus-a-monkey/)
In practice, if you want to explore classifiers, I strongly recommend the Weka package:

http://www.cs.waikato.ac.nz/ml/weka/

It’s free, and easy to install, use, & troubleshoot. It lets you quickly test many alternative (well-vetted) classifiers, all in a proper cross-validated/precision-recall framework.

Here’s a nice step-by-step intro for biologists:


There’s also a great book to walk you through the entire process. Highly recommended!!!

In Python, you can also use the scikit-learn library:
https://scikit-learn.org/stable/
Like Weka, it’s free, easy to install & use, and very powerful

I recommend combining it with the Pandas library for data analysis to make it easy to work with big, tabular datasets:
https://pandas.pydata.org/