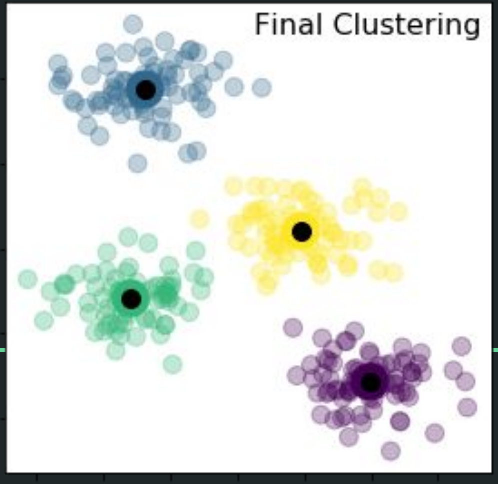
Clustering & Classification

BIPN 162



By the end of today you’ll be able to:

- Identify use cases for clustering and/or classification algorithms
- Describe the process of K-means expectation-maximization
- Implement and assess a logistic regression

What neuroscientists classify & why

- Decoding neural activity
  - **Decoding** refers to a class of methods that aim to predict or reconstruct some variable of interest (e.g., stimulus presented, movement) from neural activity patterns.
  - If we’re using linear decoding, we assume there's a linear relationship between the neural activity and the variable we're trying to decode. As in a typical linear regression, this means we can express the variable as a weighted sum of the neural activity, plus some noise or error term.
- Diagnostic classification
  - E.g., is it a cancerous tumor, or not?
- Determining cell types (or other groupings of neuro-stuff)

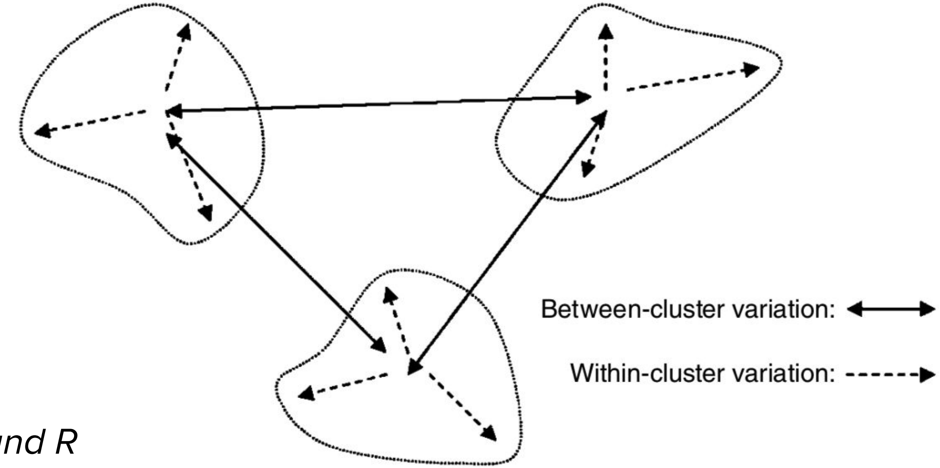
Clustering algorithms seek to learn, from the properties of the data, an optimal division or discrete labeling of groups of points.

Classification algorithms try to predict the value of a target variable

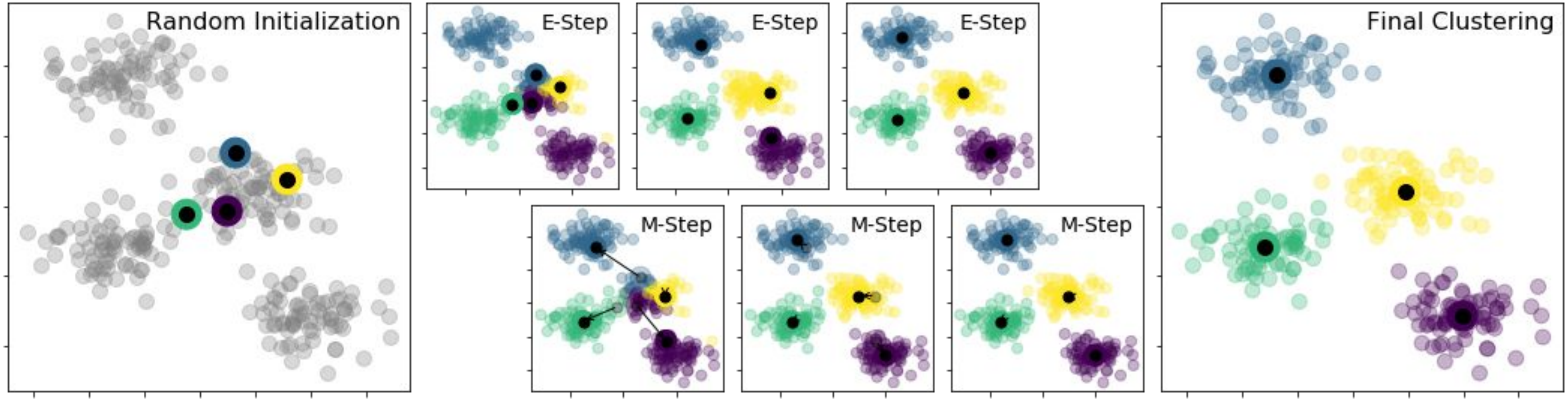
The **K-means algorithm** searches for a predetermined number of clusters within an unlabeled multidimensional dataset; relies on the idea of “optimal clustering”

1. The "cluster center" is the arithmetic mean of all the points belonging to the cluster.
2. Each point is closer to its own cluster center than to other cluster centers.

These clusters could be genes, neurons, etc.



From *Data Science in Python and R*



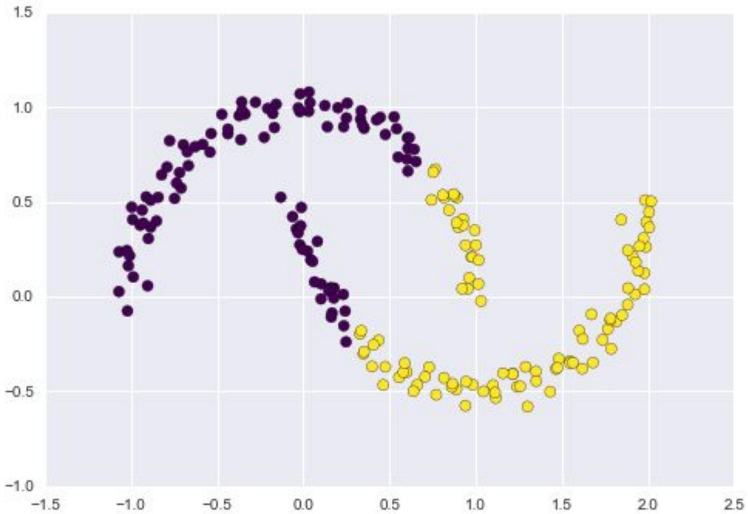
See the code for this figure here:
https://jakevdp.github.io/PythonDataScienceHandbook/06.00-figure-code.html#Expectation-Maximization

The expectation-maximization (E-M) algorithm

1. **Initialization**: user decides on # of clusters; the centers of the clusters are randomly chosen
2. **Expectation**: each data point is assigned to the closest cluster center (usually using Euclidean distance)
3. **Maximization** : cluster centers are re-computed (they are the center of mass of the colored points)
4. **Expectation-Maximization** steps 2 & 3 are repeated until convergence (ie the cluster centers do not move anymore)

Notes about K-means clustering

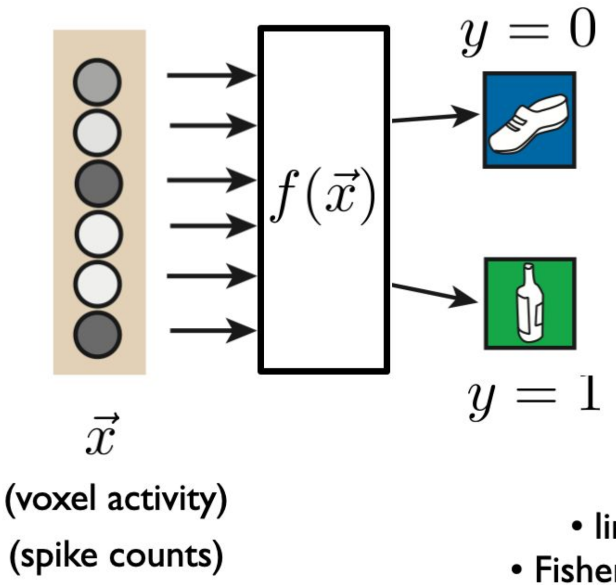
- The user chooses the expected number of clusters — so there is some subjectivity involved
- The E-M procedure maximizes each step, but may not provide an optimal *global* solution
- K-means is limited to *linear* cluster boundaries
  - For non-linear, try **support vector machines**
- Because it is iterative, it can be slow for large sample sizes

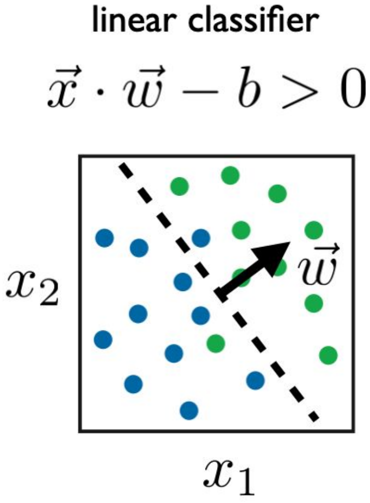


See https://towardsdatascience.com/want-clusters-how-many-will-you-have-8737f4ba9bf2 for more details.

Classification

Classification: mapping from vector input to discrete category





- linear perceptron
- Fisher linear discriminant
- support vector machine (SVM)

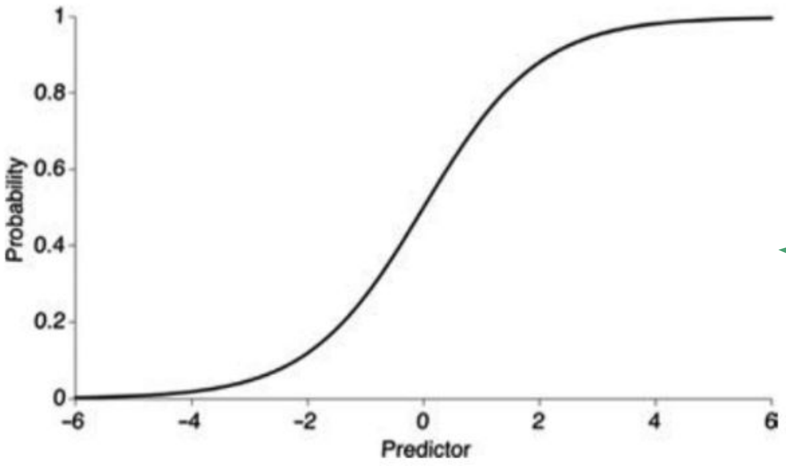
Slide: Mikio Aoi / Marcus Benna

Logistic regression

- *Generalized* linear model to predict binary (categorical) outcomes
- Instead of fitting a line, we fit a *sigmoidal* function
- Logistic regression estimates the weights that best link predictors to outcomes using maximum likelihood estimation (MLE)

"Log odds"

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X$$



Solve for p...

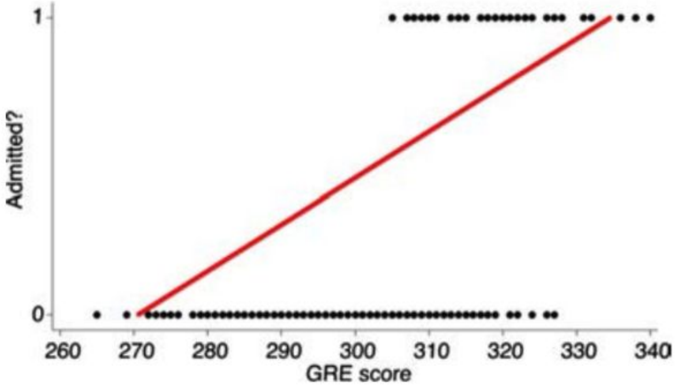
$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$$

(The logistic function)

Logistic regression notebook example:

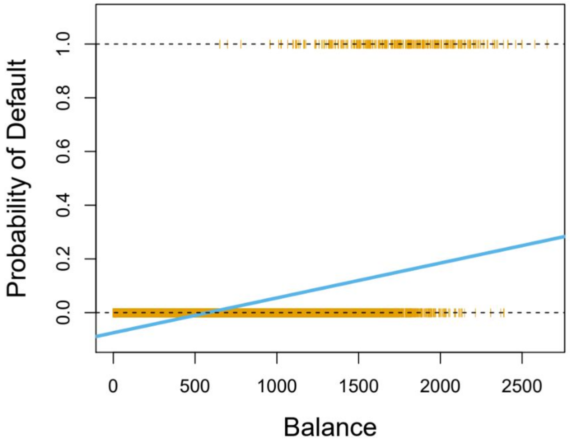
Could we predict likelihood of being admitted to graduate school based on GRE scores?

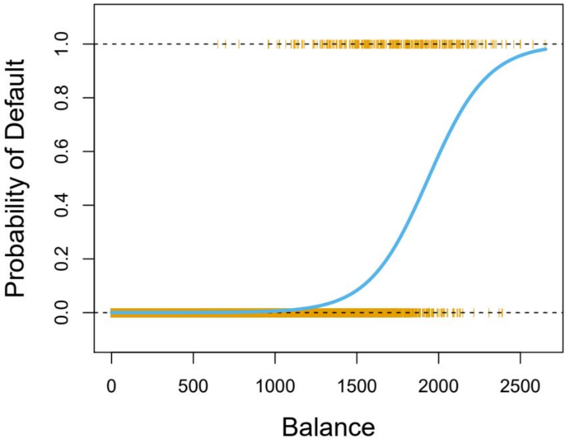
Applicant	1	2	3	4	5	...	496	497	498	499	500
GRE score	304	279	338	296	299	...	312	290	319	300	293
Admitted	0	0	1	0	0	...	1	0	0	0	0



We should not use a linear regression with categorical data because:

1. Linear regression assumes that the effect of an unit change in the independent variable on the dependent variable is constant across all values
2. Errors are not normally distributed (an assumption of linear regression)
3. A linear regression predicts things beyond the possible bounds of 0 and 1!





[An Introduction to Statistical Learning]()

Resources

An Introduction to Statistical Learning Chapter 4

In-Depth K-Means

Classification and Clustering, *Neural Data Science* Chapter 9
https://www.sciencedirect.com/science/article/pii/B978012804043000009X

The 5 Clustering Algorithms Data Scientists Need to Know

If you’d like to learn about more of these topics in depth…

DSE 80: The Practice and Application of Data Science

COGS 109: Modeling & Data Analysis

COGS 118A. Supervised Machine Learning Algorithms

COGS 118B. Introduction to Machine Learning II

CSE 291. Unsupervised Learning