Unsupervised and supervised learning

In the Hawks data set, which is available as part of the Stat2Data library in R and is copied in the unit github, wing and weight measurements are made for three types of hawk:

CH=Cooper's, RT=Red-tailed, SS=Sharp-Shinned

with a pretty unequal distribution:

	CH	RT	SS	total
number	31	121	68	221

We can plot this data and the result is shown in Fig. **??**. Obviously supervised learning works in this case and this would be useful if we found a hawk and didn't know how to identify it; from the decision boundaries we can see that knowing the wing length and weight would allow us to identify the hawk.

In this case we are clearly benifitting from the expertise of the people who supplied the data. The machine learning algorithim isn't doing something new for us, it isn't working out how many different types of hawks there are, or discovering stuff we didn't know; instead it is allowing us to apply to new data points, new hawks, a classification we have already discovered. This seems a less important task than the *unsupervised* task, to classify without being told the classes. Say you just went out an measured some hawks and had the results in Fig. **??**, could you spot that there were species.

n

Unsupervised learning, studying unlabelled data, is about discovering structure in the data. Clearly this is useful and hard and in an obvious way its goal is discovering knowledge, rather than applying it. We should not take too seriously this stark division between supervised and unsupervised learning, for a start it relies on an obvious division between "label" and other properties of the data. These days it feels like training and learning approaches often combine supervised and unsupervised elements, along, indeed, with reinforcement learning. Here we will look at some unsupervised learning algorithms, they give a lot of insight.

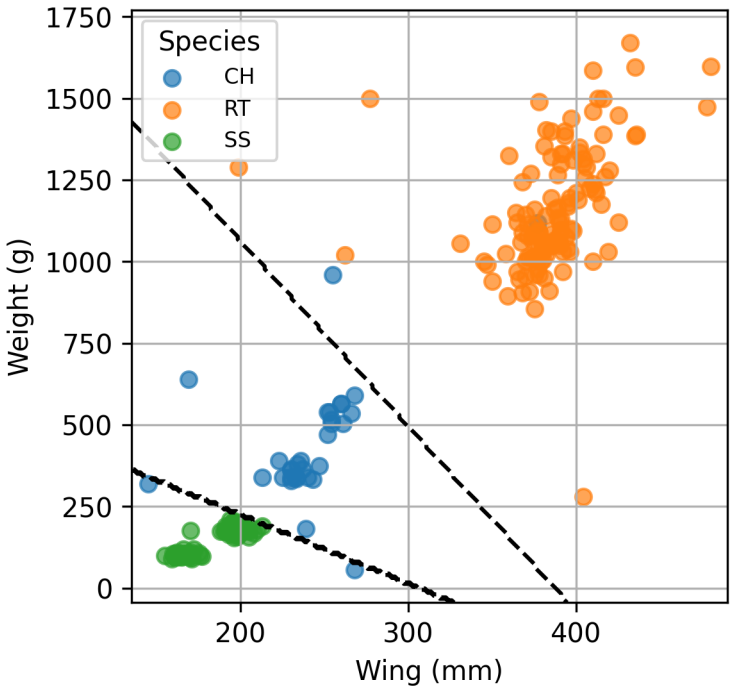

Figure 1: The hawk data is plotted wiht wing length and weight; the three species have been marked by colour. Clearly the three species correspond to different clusters and logistic regression has been used to find the two decision boundaries plotted as black dashed lines, these are pretty accurate.

k-means clustering

Probably the most famous and the most straight-forward approach to unsupervised learning is k-means; k-means only works in very specific situations but it is always the first thing to try. In k-means you decide how many clusters you think there should be, that doesn't sound very 'unsupervised', but in practice given how quickly the algorithm runs, you can try different values. k different points are picked at random, these are the *centroids* and to each k points the data points that are nearer to it than to any of the other centroids is given to that centroid. That gives k sets of points, one for each centroid. Now k new centroids are calculate, each is at the center of a cluster. This is then repeated until the centroids stop moving.

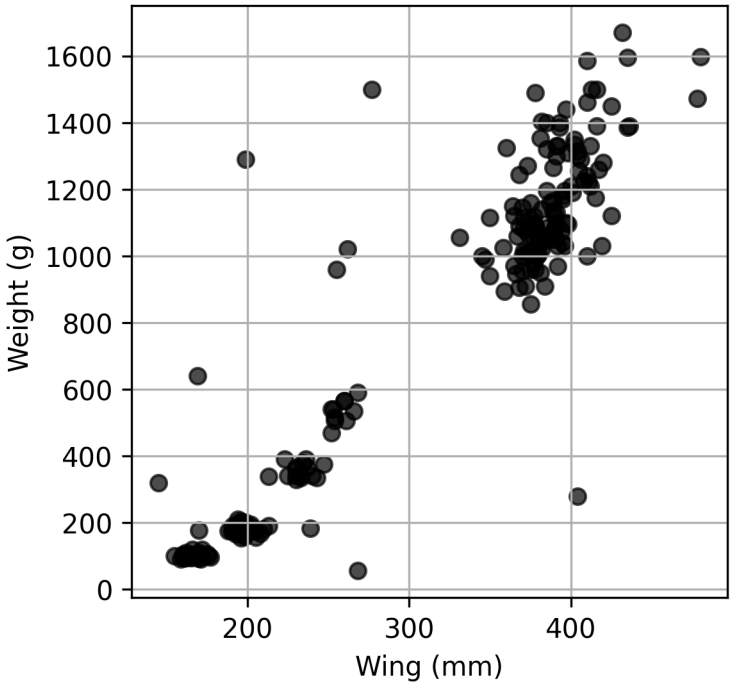

Figure 2: The hawk data is plotted wiht wing length and weight but without labels, it isn't so clear that there should be three clusters, it looks more like five or six, maybe the sex of the hawks also has an affect. Unsupervised learning is hard!

In mathematics here it is the algorithm again, let

$$\mathcal{D}\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \tag{1}$$

be the data and $\mathbf{y}_1$ up to $\mathbf{y}_k$ be the initial centroids. Now make the clusters

$$C_i = \{\mathbf{x}_j \in \mathcal{D} : d(\mathbf{x}_j, \mathbf{y}_i) < d(\mathbf{x}_j, \mathbf{y}_{i'}) \forall i' \neq i\} \tag{2}$$

where $d(\mathbf{x}, \mathbf{y})$ is the distance between $\mathbf{x}$ and $\mathbf{y}$; usually this would just be the Euclidean distance:

$$d(\mathbf{x}, \mathbf{y}) = |\mathbf{x} - \mathbf{y}| \tag{3}$$

and we will discuss the choice of distance later[1]. Now you make new centroids:

$$y_i \to \frac{1}{|C_i|} \sum_{\mathbf{x}_j \in C_i} \mathbf{x}_j \tag{4}$$

and repeat.

Lets try it with the hawk data; in Fig. 3. The algorithm converges quickly and gives three clusters, just not the clusters we might've expected. This is the thing with unsupervised learning, it learns from the data, not our intuition. If we were hoping to discover hawk species this way we would fail, it clusters together the CT and SS hawks and splits the RT into two. However, we also learn something, we learn that this is what the unsupervised algorithm sees and as data scientists we'd consider if we had the correct value of k. In Fig. reffig:khawksk6 we use $k = 6$ and get something more like we might expect. Hopefully this shows the advantages and the disadvantages of using k-means, we started off hoping to discover species using the clustering, in the end we learned that weight and wing length does not produce natural clusters corresponding to species and we had to use our intuition to suggest we need to look at other properties as well.

In the next note we will think a bit more about how to pick k and to assess the quality of our clustering once we have performed it. The main point is that k-means works best when the clusters are spherical and roughly equal; obviously the Hawks data has clusters that are neither very spherical nor of equal size, though making a smaller data set with the same number of each hawk doesn't help, unsupervised learning still doesn't give clusters corresponding to the three hawk types, though, and this is the important point, it does tell you something about the data.

[1] In this set up with the Euclidean distances there are unlikely to be draws, but if there are draws you need a procedure for dealing with them. This is usually just fiddly but not a problem!

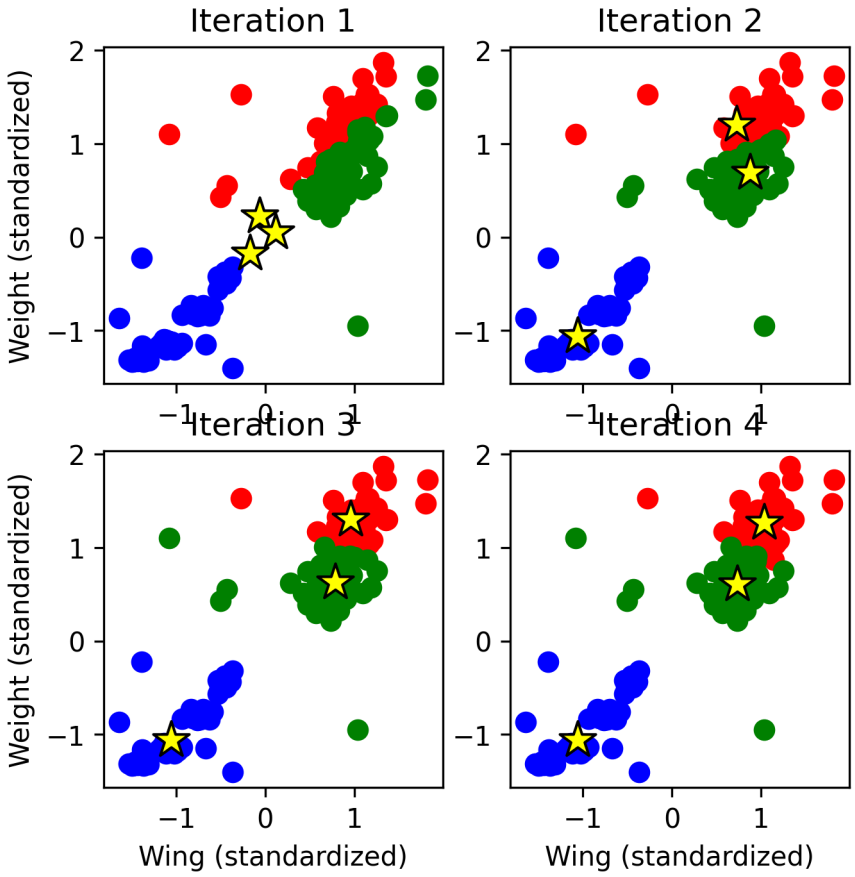

Figure 3: The k-means algorithm is run for the hawk data with k=3. This uses standarded values for the two component values, wing length and weight; because the two things aren't really comporable, one measured in milimeters, the other in grammes, it would be peculiar to just measure distances in the mixed gram, milimeter space. Instead we standardize first, take away the mean and divide by the standard deviation, now the two components have no units and have a similar spread of values. The first values are randomly chosen near the middle to make it easier to see what's happening, usually we just pick k of the points as the initial centroids.

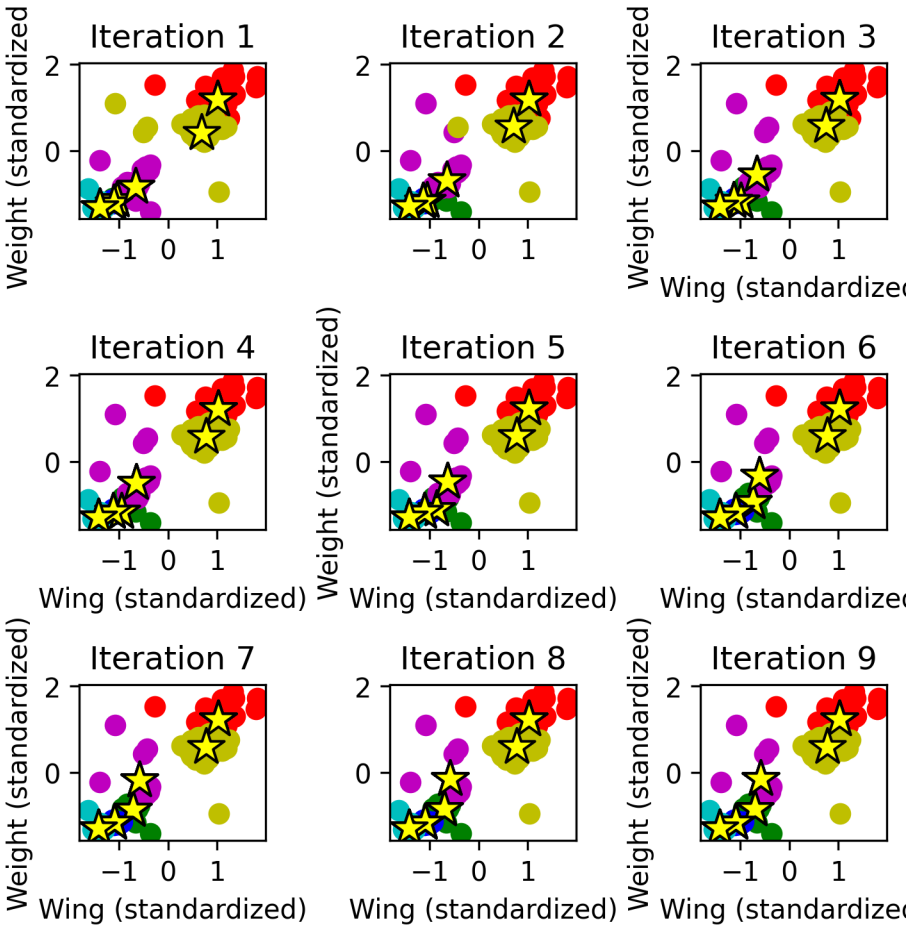

Figure 4: The k-means algorithm is run for the hawk data with k=6. This takes longer to converge, but iteration 9 is actually the same as iteration 8; I included it just to make up the grid. It has found six clusters, roughly two for each of the species we saw at the start, probably corresponding to each of two sexes..