Astro Data Science Seminar
Lab 7 - 2016 May 31
Machine Learning

In this final installment of the seminar, we'll briefly discuss the vast topic of "machine learning". Machine learning problems are broadly grouped in to 2 categories:

1) **Supervised learning** - you provide "labels" for your data, or previous examples of what you want the computer to learn from.
2) **Unsupervised learning** - you provide data and ask the computer to search for structure, over-densities, or patterns.

You should view Machine Learning as a tool, just like fitting a line (a very related problem, it turns out). Machine Learning is a means to an end, not a result itself. It is a powerful set of algorithms that can go wildly off base if you feed it bad data or use it poorly. It can recover noise instead of signal, and people may believe junk results because they use fancy techniques.

Machine learning can do 2 basic tasks (I'm over simplifying it a bit):

1) Classification - which bin or group does a data point fall in? How many bins are there in this data?
2) Regression - can this data predict an outcome? Can we fit a complex model to data?

The main package you'll use for machine learning in Python is called Scikit-learn. The best way to learn a new method from this important package is to go to the website and check out the Examples page!

http://scikit-learn.org/

Warning: Scikit-learn has some great examples, and some not-so-great examples. Always Google for more examples when possible!

Part 1
Update your Fork (again)

You've been doing this all quarter, just 1 more time now!

Part 2
Gaussian Mixture Models

We're going to focus on only 1 example of machine learning, but it is one that's easy to understand, use, and incorporate in your own work!

Gaussian Mixture Models (GMM) are exactly what they sound like: you are modeling a dataset using a mixture of many Gaussian functions. These "normal curves" can be in any number of dimensions, and can be covariant (tilted). This makes them *very powerful* for describing data!

The science example today is a classical classification problem: cluster members versus field stars. You can estimate cluster membership using many measurements: distance, radial velocity, position on sky, proper motions, location in color magnitude diagrams.

We're going to use GMM only on the Proper Motions for a single cluster: Messier 67

Follow the example code in the IPython notebook!
You will generate a model with 2 Gaussians, 1 that represents the proper motions of the cluster, 1 for the field stars. Using this we will classify each star in the dataset, and then make a color magnitude diagram of *only* the likely cluster members!