Unsupervised Learning - Density Estimation

MInDS @ Mines

In this lecture we cover density estimation and show how some density estimation methods can be also applied to clustering. We take a look at the various distributions we can use to estimate the density of data then discuss Gaussian mixture models, MeanShift, and DBSCAN as clustering methods.

Density Estimation

Density estimation is the estimation of a probability distribution function of provided data. The goal is to create a generalized model about the data that explains the probability of its creation or existence in a particular way. Density estimation has a variety of applications including clustering which we will discuss in further detail.

Histograms

A basic tool often used for examining the distribution of data is the histogram. Histograms group data into *bins*, usually equally sized, for the possible range of values for a particular feature. Histograms plot the bins on the x-axis and the y-axis represents the frequency of that data. Histograms can be really useful to get a general understanding of our data but it is somewhat difficult to select the number of bins or the bin width in the plot and sometimes the plots can be deceiving. There are several approaches to selecting the number of bins in a histogram. Two simple approaches to selecting the bin width are Scott's rule[1] and Freedman and Diaconis' rule[2]. Both these approaches assume that the data follows a normal distribution and focus on minimizing the difference between the histogram and the underlying distribution. Scott's rule states the the bin width, w, can be formulated with respect to the standard deviation, σ, and the number of samples, n, as,

It is common to generate a normalized histogram where the y-axis is normalized so the area under the graph is 1. This more closely resembles the probability density function of the data.

[1] David W. Scott. On optimal and data-based histograms. *Biometrika*, 66(3):605–610, 1979

[2] Alan Julian Izenman. Review papers: Recent developments in nonparametric density estimation. *Journal of the American Statistical Association*, 86(413):205–224, 1991

$$w = 3.49\sigma n^{\frac{-1}{3}}. \tag{1}$$

Freedman and Diaconis propose a more robust approach that incorporates the interquartile range, d, or the difference between the 25th and 75th percentile of the data, instead of the standard deviation and formulate the bin width as,

$$w = 2dn^{\frac{-1}{3}}. \tag{2}$$

Both these methods use descriptive statistics about the data to determine the bin width, whether it be the standard deviation or

the interquartile range. Both these methods are also quite dated and newer methods have been developed that are more effective however they are also more computationally intensive.

Knuth [3] introduces a statistical method of determining the bin width by investigating the relationship between the data and assuming a piece-wise constant density model. Bayesian Block Representations[4] also provide an even more sophisticated approach to bin width calculation based on the probabilistic relationship between data and provide variable sized bins.

[3] Kevin H Knuth. Optimal Data-Based Binning for Histograms. 2006

[4] I M Aug, Jeffrey D Scargle, Jay P Norris, Brad Jackson, and James Chiang. Bayesian blocks representations. (2008):1–82, 2012

Despite the sophistication of these methods, sometimes the results are still not great and we need to find other methods than the basic hsitogram for density estimation. Nick Strayer has a good demo on how the number of bins can change the histograms drastically available at `http://nickstrayer.me/histogram_bins/`.

Kernel Density Estimation

A kernel, $K(x_i)$, with respect to density estimation is a non-negative function that integrates to one. You can think of this as an example distribution. Some example kernels are gaussian, uniform, exponential, and linear. A scaled kernel $K_h(x_i)$ scaled by a parameter h, is

$$K_h(x_i) = \frac{1}{h} K(\frac{x_i}{h}). \tag{3}$$

With respect to density estimation, the hyperparameter h, is called the *bandwidth*, and it controls the tradeoff between bias and variance in the kernel density estimator.

Mathew Conlen has a great demo on Kernel Desnity Estimation that shows how the bandwidth works as a smoothing parameter.

Formulations for some example kernels are,

Gaussian	$K_h(x) \propto \exp \frac{-x^2}{2h^2}$,	(4)
Uniform / Tophat	$K_h(x) \propto 1$ if $x < h$,	(5)
Exponential	$K_h(x) \propto \exp \frac{-x}{h}$,	(6)
Linear	$K_h(x) \propto 1 - \frac{x}{h}$,	(7)
Epanechnikov	$K_h(x) \propto 1 - \frac{x^2}{h^2}$,	(8)
Cosine	$K_h(x) \propto \cos \frac{\pi x}{2h}$.	(9)

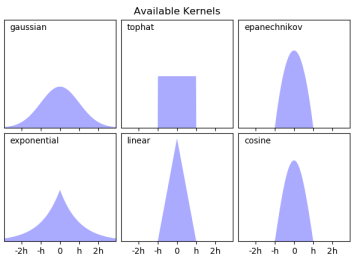

Figure 1: Examples of kernel distributions.

The kernel density estimator of an unknown density f is,

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K(\frac{x - x_i}{h}). \tag{10}$$

The general idea here is that we determine the parameters for a given distribution when fit to various portions of the data. We do not

assume that the data is uni-modal. With these ideas in mind, let's look at some clustering approaches that use density estimation.

Gaussian Mixture Models

Gaussian mixture models for clustering assume that each cluster follows a Gaussian distribution. Based on that, the method determines the distributions' parameters that fit each cluster.

MeanShift

The MeanShift algorithm focuses on determining centroids for each cluster. It is an iterative algorithm where the centroids' locations are updated by a "mean shift" vector which points in the direction of maximum increase in cluster density.

The mean shift vector, $\mathbf{m}_i$ for centroid, $\mathbf{x}_i$, is,

$$\mathbf{m}_i = \frac{\sum_{j \in N(\mathbf{x}_i)} K(\mathbf{x}_i - \mathbf{x}_j)\mathbf{x}_j}{\sum_{j \in N(\mathbf{x}_i)} K(\mathbf{x}_i - \mathbf{x}_j)} \tag{11}$$

where the set $N(\mathbf{x}_i)$ is the set of points within a specified distance of $\mathbf{x}_i$ called its *neighborhood*, and K is a kernel.

Based on the mean shift vector, the centroid $\mathbf{x}_i$ is updated at iteration $t+1$ as,

$$\mathbf{x}_{i,t+1} = \mathbf{m}_{i,t} \tag{12}$$

DBSCAN

DBSCAN determines clusters by looking at the variation in densities of the data. It assigns every point a status of being a core point or a border point. Core points are those that are in an area of high density whereas border points are those in areas of low density. DBSCAN has two hyperparameters, the minimum number of points to consider a high density area, and the distance to count the number of points within. Based on that, a core point is one that has the minimum number of points within the specified distance. Here, we can also determine the metric we wish to utilize for distance. When we've determined our core and border points, we can determine the clusters as those with connected core points.

Key benefits of DBSCAN are that clusters do not need to form as convex shapes, and that we do not need to pre-determine the number of clusters we want the model to detect.

References

[1] I M Aug, Jeffrey D Scargle, Jay P Norris, Brad Jackson, and James Chiang. Bayesian blocks representations. (2008):1–82, 2012.

[2] Alan Julian Izenman. Review papers: Recent developments in nonparametric density estimation. *Journal of the American Statistical Association*, 86(413):205–224, 1991.

[3] Kevin H Knuth. Optimal Data-Based Binning for Histograms. 2006.

[4] David W. Scott. On optimal and data-based histograms. *Biometrika*, 66(3):605–610, 1979.