Nonparametric Density Estimation

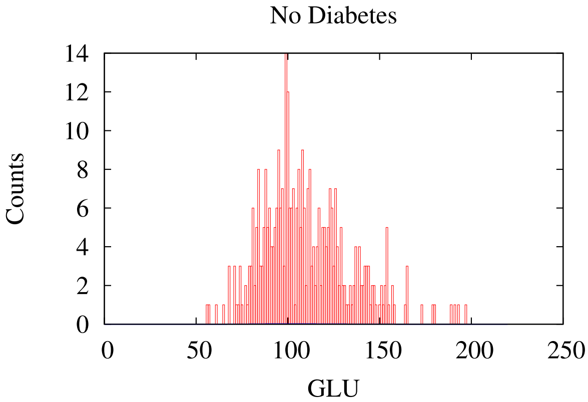
October 1, 2018

Introduction

- If we can't fit a distribution to our data, then we use nonparametric density estimation.
- Start with a histogram.
- But there are problems with using histrograms for density estimation.
- A better method is *kernel density estimation*.
- Let's consider an example in which we predict whether someone has diabetes based on their glucode concentration.
- We can also use kernel density estimation with naive Bayes or other probabilistic learners.

Introduction

- Plot of plasma glucose concentration (GLU) for a population of women who were at least 21 years old, of Pima Indian heritage and living near Phoenix, Arizona, with no evidence of diabetes:



Introduction

- Assume we want to determine if a person's GLU is abnormal.
- The population was tested for diabetes according to World Health Organization criteria.
- The data were collected by the US National Institute of Diabetes and Digestive and Kidney Diseases.
- First, are these data distributed normally?
- No, according to a χ^2 test of goodness of fit.

Histograms

- A histogram is a first (and rough) approximation to an unknown probability density function.
- We have a sample of n observations, $X_1, \ldots, X_i, \ldots, X_n$.
- An important parameter is the bin width, h.
- Effectively, it determines the width of each bar.
- We can have thick bars or thin bars, obviously.
- h determines how much we smooth the data.
- Another parameter is the origin, x_0.
- x_0 determines where we start binning data.
- This obviously effects the number of points in each bin.
- We can plot a histogram as
  - the number of items in each bin or
  - the proportion of the total for each bin

Histograms

- We define a bins or intervals as

$$[x_0 + mh, x_0 + (m+1)h] \text{ for } m \in \mathbb{Z}$$

(i.e., the positive and negative integers).

- But for our purposes, it's best to plot the relative frequency

$$\hat{f}(x) = \frac{1}{nh}(\text{number of } X_i \text{ in same bin as } x)$$

- Notice that this is the density estimate for x.

Problems with Histograms

- One program with using histograms as an estimate of the PDF is there can be discontinuities.
- For example, if we have a bin with no counts, then its probability is zero.
- This is also a problem "at the tails" of the distribution, the left and right side of the histogram.
- First off, with real PDFs, there are no impossible events (i.e., events with probability zero).
- There are only events with extremely small probabilities.
- The histogram is discrete, rather than continuous, so depending on the smoothing factor, there could be large jumps in the density with very small changes in x.
- And depending on the bin width, the density may not change at all with reasonably large changes to x.

Kernel Density Estimator: Motivation

- Research has shown that a kernel density estimator for continuous attributes improve the performance of naive Bayes over Gaussian distributions [John and Langley, 1995].
- KDE is more expensive in time and space than a Gaussian estimator, and the result is somewhat intuitive: If the data do not follow the distributional assumptions of your model, then performance can suffer.
- With KDE, we start with a histogram, but when we estimate the density of a value, we smooth the histogram using a kernel function.
- Again, start with the histogram.
- A generalization of the histogram method is to use a function to smooth the histogram.
- We get rid of discontinuities.
- If we do it right, we get a continuous estimate of the PDF.

Kernel Density Estimator

[McLachlan, 1992, Silverman, 1998]

- Given the sample X_i and the observation x

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right),$$

  where h is the *window width*, *smoothing parameter*, or *bandwidth*.
- K is a kernel function, such that

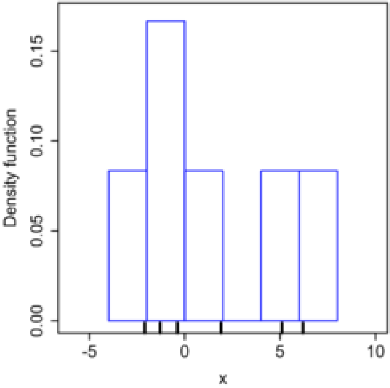
$$\int_{-\infty}^{\infty} K(x)\, dx = 1$$

- One popular choice for K is the Gaussian kernel

$$K(t) = \frac{1}{\sqrt{2\pi}} e^{-(1/2)t^2}.$$

- One of the most important decisions is the bandwidth (h).
- We can just pick a number based on what looks good.

Kernel Density Estimator



Source: https://en.wikipedia.org/wiki/Kernel_density_estimation

Algorithm for KDE

- Representation: The sample X_i for $i = 1, \ldots, n$.
- Learning: Add a new sample to the collection.
- Performance:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right),$$

  where h is the *window width*, *smoothing parameter*, or *bandwidth*, and K is a kernel function, such as the Gaussian kernel

$$K(t) = \frac{1}{\sqrt{2\pi}} e^{-(1/2)t^2}.$$

Kernel Density Estimator

```
public double getProbability( Number x ) {
  int n = this.X.size();
  double Pr = 0.0;
  for ( int i = 0; i < n; i++ ) {
    Pr += X.get(i) * Gaussian.pdf((x - X.get(i)) / this.h );
  } // for
  return Pr / ( n * this.h );
} // KDE::getProbability
```

Automatic Bandwidth Selection

- Ideally, we'd like to set h based on the data.
- This is called *automatic bandwidth selection*.
- Silverman's [1998] rule-of-thumb method estimates h as

$$\hat{h}_0 = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{1/5} \approx 1.06\hat{\sigma}n^{-1/5} \, ,$$

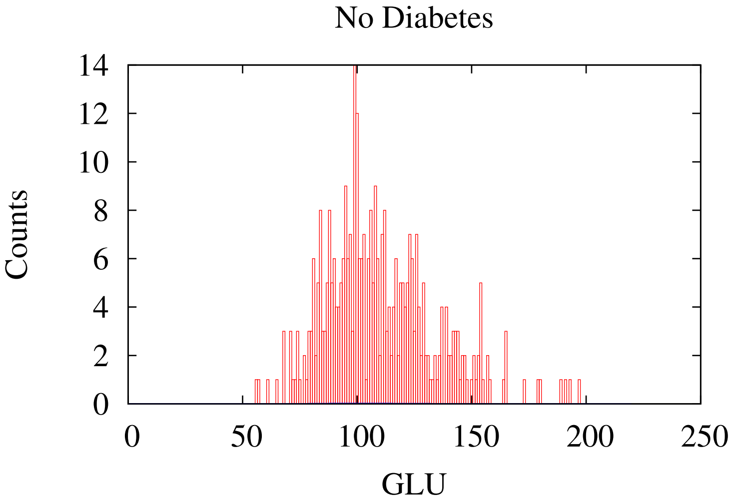
  where $\hat{\sigma}$ is the sample standard deviation and n is the number of samples.
- Silverman's rule of thumb assumes that the kernel is Gaussian and that the underlying distribution is normal.
- This latter assumption may not be true, but we get a simple expression that evaluates in constant time, and it seems to perform well.
- Evaluating in constant time doesn't include the time it takes to compute $\hat{\sigma}$, but we can compute $\hat{\sigma}$ as we read the samples.

Automatic Bandwidth Selection

- Sheather and Jones' [1991] solve-the-equation plug-in method is a bit more complicated.
- It's $O(n^2)$, and we have to solve numerically a set of equations, which could fail.
- It is regarded as theoretically and empirically, the best method we have.

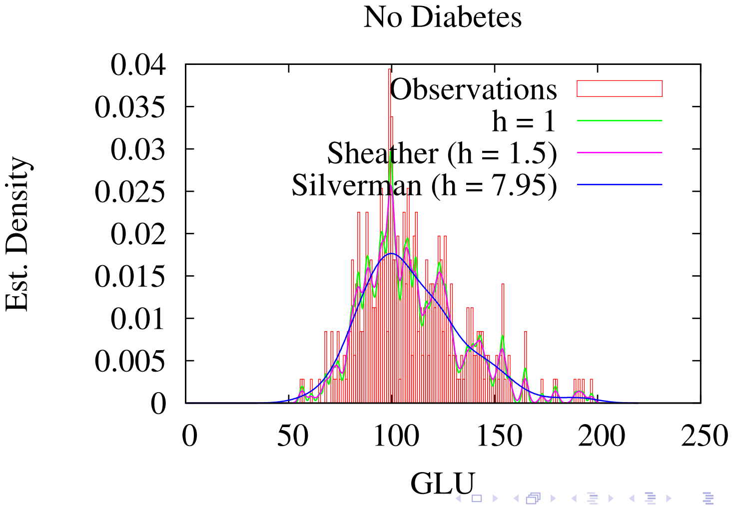
Simple KDE Example

- Determine if a person's GLU is abnormal.



Simple KDE Example

- Green line: Fixed value, $h = 1$
- Magenta line: Sheather and Jones' method, $h = 1.5$
- Blue line: Silverman's method, $h = 7.95$

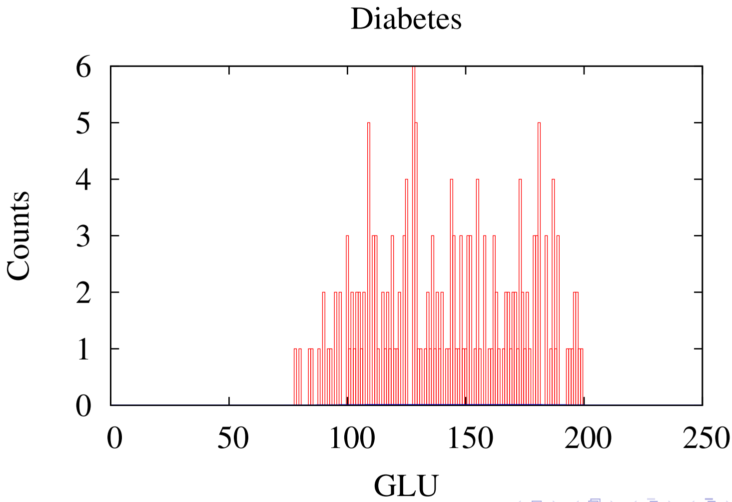


Simple KDE Example

- Assume $h = 7.95$
- $\hat{f}(100) = 0.018$
- $\hat{f}(250) = 3.3 \times 10^{-14}$
- $P(0 \leq x \leq 100) = \int_0^{100} \hat{f}(x)\, dx$
- $P(0 \leq x \leq 100) = \sum_0^{100} \hat{f}(x)\, dx$
- $P(0 \leq x \leq 100) \approx 0.393$

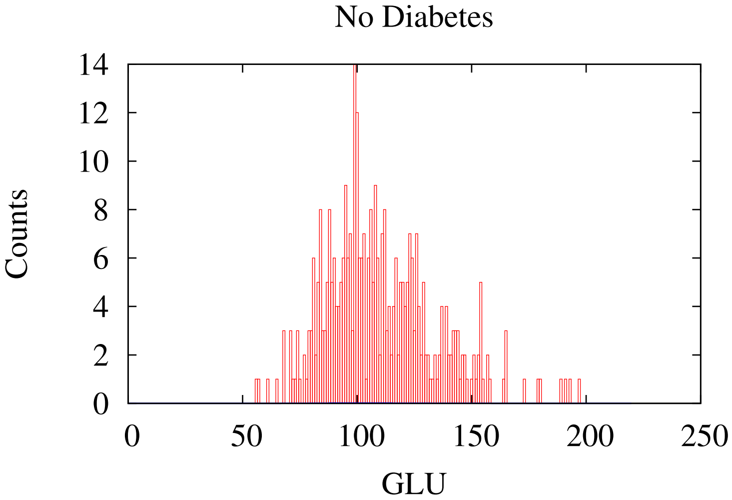
Naive Bayes with KDEs

- Assume we have GLU measurements for women with and without diabetes.
- Plot of women with diabetes:



Naive Bayes with KDEs

- Plot of women without:

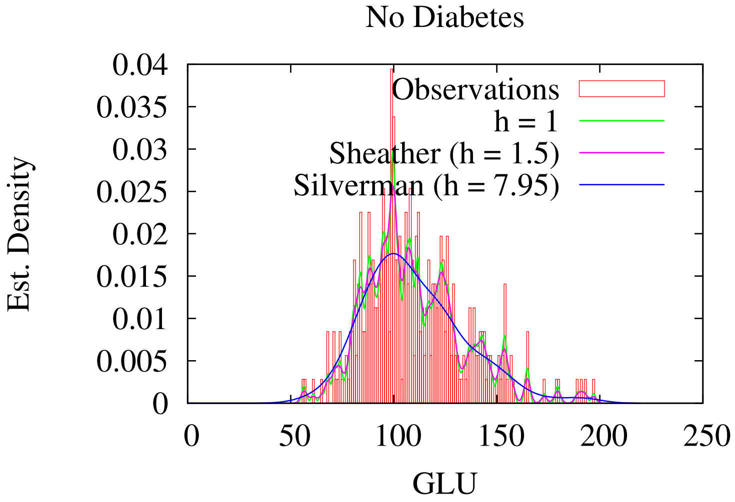


Naive Bayes with KDEs

- The task is to determine, given a woman's GLU measurement, if it is more likely that she has diabetes (or vice versa).
- For this, we can use Bayes' rule.
- Like before, we build a kernel density estimator for both sets of data.

Naive Bayes with KDEs

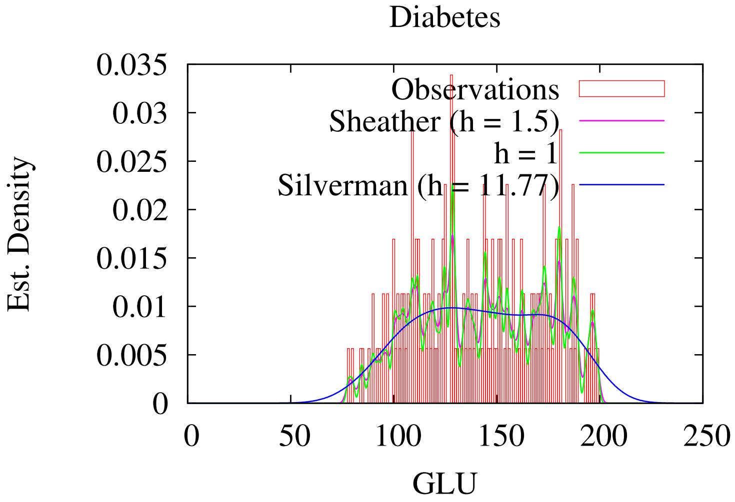
- Without diabetes:



- Silverman's rule of thumb gives $\hat{h}_0 = 7.95$

Naive Bayes with KDEs

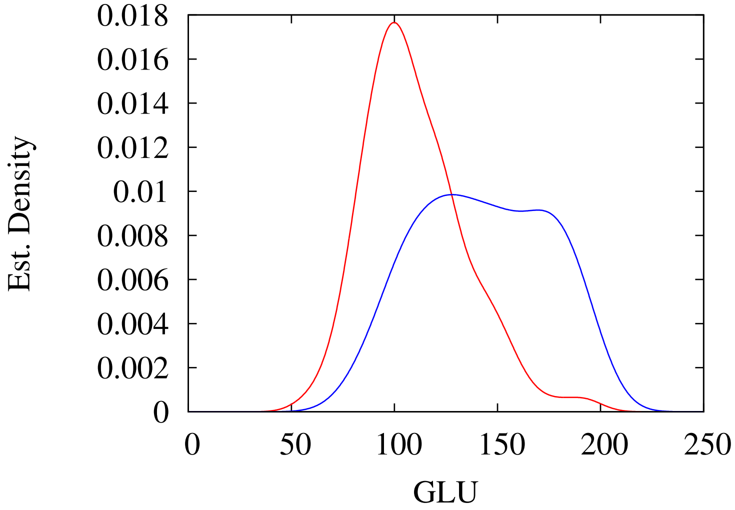
- With diabetes:



- Silverman's rule of thumb gives $\hat{h}_1 = 11.77$

Naive Bayes with KDEs

- All together:



Naive Bayes with KDEs

- Now that we've built these kernel density estimators, they give us $P(GLU|Diabetes = true)$ and $P(GLU|Diabetes = false)$.

Naive Bayes with KDEs

- We now need to calculate the *base rate* or the *prior probability* of each class.
- There are 355 samples of women without diabetes, and 177 samples of women with diabetes.
- Therefore,

$$P(Diabetes = \text{true}) = \frac{177}{177 + 355} = .332$$

- And,

$$P(Diabetes = \text{false}) = \frac{355}{177 + 355} = .668$$

- Or,

$$P(Diabetes = \text{false}) = 1 - P(Diabetes = \text{true}) = 1 - .332 = .668$$

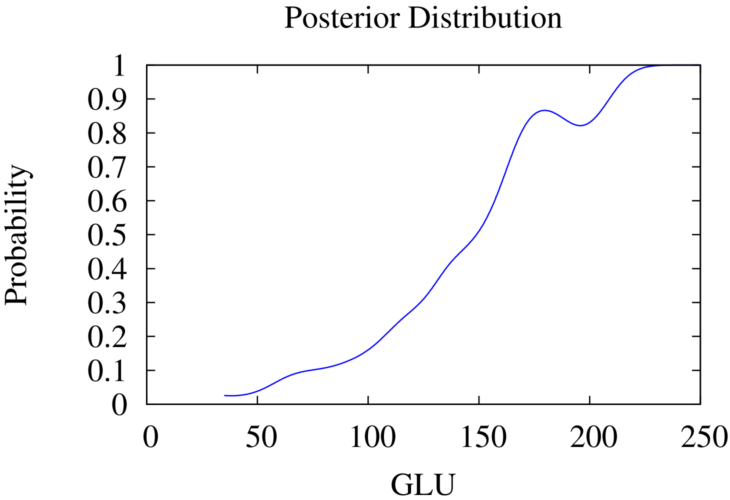
Naive Bayes with KDEs

- Bayes rule:

$$P(D|GLU) = \frac{P(D)P(GLU|D)}{P(D)P(GLU|D) + P(\neg D)P(GLU|\neg D)}$$

Naive Bayes with KDEs

- Plot of the posterior distribution:



Naive Bayes with KDEs

- $P(D|GLU = 50)$?

$$P(D|GLU = 50) = \frac{(.332)(2.73E - 5)}{(.332)(2.73E - 5) + (.668)(3.39E - 4)} = .0385$$

- $P(D|GLU = 175)$?

$$P(D|GLU = 175) = \frac{(.332)(.009)}{(.332)(.009) + (.668)(7.65E - 4)} = .854$$

References

G. H. John and P. Langley. Estimating continuous distributions in Bayesian classifiers. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Francisco, CA, 1995. Morgan Kaufmann.

G. J. McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*. John Wiley & Sons, New York, NY, 1992.

S. J. Sheather and M. C. Jones. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53(3):683–690, 1991.

B. W. Silverman. *Density estimation for statistics and data analysis*, volume 26 of *Monographs on statistics and applied probability*. Chapman & Hall/CRC, Boca Raton, FL, 1998.