In [61]:

```julia
using Distributions
```

Nonparametric Inference

Nonparametric estimation is also referred as curve estimation or smoothing. We do not make any functional assumptions.

The Bias-Variance Tradeoff

Let g denote an unknown function such as a density function or a regression function. Let $\hat{g_n}$ denote an estimator of g. Bear in mind that $\hat{g_n}$ is a random function evaluated at a point x. The estimator is random because it depends on the data.

alt text

A loss function, we will use integrated squared error (ISE):

$$L(g, \hat{g_n}) = \int (g(u) - \hat{g_n}(u))^2$$

The **risk** or mean integrated squared error (MISE) with respect to squared error loss is

$$R(f, \hat{f}) = E(L(g, \hat{g}))$$

Then, the risk can be written as

$$R(f, \hat{f}) = \int b^2(x)dx + \int v(x)dx$$

where

$$b(x) = E(\hat{g_n}) - g(x)$$

is the bias of $\hat{g_n}(x)$ at a fixed x and

$$v(x) = Var(\hat{g_n}(x)) = E([\hat{g_n} - E(\hat{g_n})]^2)$$

is the variance of $\hat{g_n}(x)$ at a fixed x.

In summary,

$$\text{RISK} = \text{BIAS}^2 + \text{VARIANCE}$$

When the data are oversmoothed, the bias term is large and the variance is small. When the data are undersmoothed the opposite is true.

This is called the bias-variance tradeoff.

alt text

Kernel Regression Estimators

Nadaraya-Watson Kernel Regression

Kernel regression estimation is an example of fully nonparametric estimation (others are splines, nearest neighbors, etc.). We'll consider the Nadaraya-Watson kernel regression estimator in a simple case.

Suppose we have an iid sample from the joint density $f(x, y)$, where x is k -dimensional. The model is

$$y_t = g(x_t) + \varepsilon_t,$$

where

$$E(\varepsilon_t | x_t) = 0.$$

The conditional expectation of y given x is $g(x)$. By definition of the conditional expectation, we have

$$\begin{aligned} g(x) &= \int y \frac{f(x, y)}{h(x)} dy \\ &= \frac{1}{h(x)} \int y f(x, y) dy, \end{aligned}$$

where $h(x)$ is the marginal density of x :

$$h(x) = \int f(x, y) dy.$$

This suggests that we could estimate $g(x)$ by estimating $h(x)$ and $\int y f(x, y) dy$.

Estimation of the denominator

A kernel estimator for $h(x)$ has the form

$$\hat{h}(x) = \frac{1}{n} \sum_{t=1}^{n} \frac{K\left[\left(x - x_t\right) / \gamma_n\right]}{\gamma_n^k},$$

where n is the sample size and k is the dimension of x.

The function $K(\cdot)$ (the kernel) is absolutely integrable:

$$\int |K(x)| dx < \infty,$$

and $K(\cdot)$ integrates to 1 :

$$\int K(x) dx = 1.$$

In this respect, $K(\cdot)$ is like a density function, but we do not necessarily restrict $K(\cdot)$ to be nonnegative.

The window width parameter, γ_n is a sequence of positive numbers that satisfies

$$\lim_{n\to\infty} \gamma_n = 0$$
$$\lim_{n\to\infty} n\gamma_n^k = \infty$$

So, the window width must tend to zero, but not too quickly.

To show **pointwise consistency** of $\hat{h}(x)$ for $h(x)$, first consider the expectation of the estimator (because the estimator is an average of iid terms, we only need to consider the expectation of a representative term):

$$E\left[\hat{h}(x)\right] = \int \gamma_n^{-k} K\left[(x-z)/\gamma_n\right] h(z) dz.$$

Change variables as $z^* = (x-z)/\gamma_n$, so $z = x - \gamma_n z^*$ and $|\frac{dz}{dz^{*\prime}}| = \gamma_n^k$, we obtain

$$\begin{aligned} E\left[\hat{h}(x)\right] &= \int \gamma_n^{-k} K\left(z^*\right) h(x-\gamma_n z^*) \gamma_n^k dz^* \\ &= \int K\left(z^*\right) h(x-\gamma_n z^*) dz^*. \end{aligned}$$

Now, asymptotically,

$$\begin{aligned} \lim_{n\to\infty} E\left[\hat{h}(x)\right] &= \lim_{n\to\infty} \int K\left(z^*\right) h(x-\gamma_n z^*) dz^* \\ &= \int \lim_{n\to\infty} K\left(z^*\right) h(x-\gamma_n z^*) dz^* \\ &= \int K\left(z^*\right) h(x) dz^* \\ &= h(x) \int K\left(z^*\right) dz^* \\ &= h(x), \end{aligned}$$

since $\gamma_n \to 0$ and $\int K\left(z^*\right) dz^* = 1$ by assumption. (Note:\ that we can pass the limit through the integral is a result of the dominated convergence theorem. For this to hold we need that $h(\cdot)$ be dominated by an absolutely integrable function.)

Next, considering the **variance** of $\hat{h}(x)$, we have, due to the iid assumption

$$\begin{aligned} n\gamma_n^k V\left[\hat{h}(x)\right] &= n\gamma_n^k \frac{1}{n^2} \sum_{t=1}^n V\left\{\frac{K\left[(x-x_t)/\gamma_n\right]}{\gamma_n^k}\right\} \\ &= \gamma_n^{-k} \frac{1}{n} \sum_{t=1}^n V\left\{K\left[(x-x_t)/\gamma_n\right]\right\} \end{aligned}$$

By the representative term argument, this is

$$n\gamma_n^k V\left[\hat{h}(x)\right] = \gamma_n^{-k} V\left\{K\left[(x-z)/\gamma_n\right]\right\}$$

\item Also, since $V(x) = E(x^2) - E(x)^2$ we have

$$\begin{aligned}
n\gamma_n^k V\left[\hat{h}(x)\right] &= \gamma_n^{-k} E\left\{\left(K\left[(x-z)/\gamma_n\right]\right)^2\right\} - \gamma_n^{-k}\{E\left(K\left[(x-z)/\gamma_n\right]\right)\}^2 \\
&= \int \gamma_n^{-k} K[(x-z)/\gamma_n]^2 h(z)dz - \gamma_n^k \left\{\int \gamma_n^{-k} K\left[(x-z)/\gamma_n\right] h(z)dz\right\}^2 \\
&= \int \gamma_n^{-k} K[(x-z)/\gamma_n]^2 h(z)dz - \gamma_n^k E\left[\widehat{h}(x)\right]^2
\end{aligned}$$

The second term converges to zero:

$$\gamma_n^k E\left[\widehat{h}(x)\right]^2 \to 0,$$

by the previous result regarding the expectation and the fact that $\gamma_n \to 0$. Therefore,

$$\lim_{n\to\infty} n\gamma_n^k V\left[\hat{h}(x)\right] = \lim_{n\to\infty} \int \gamma_n^{-k} K[(x-z)/\gamma_n]^2 h(z)dz.$$

Using exactly the same change of variables as before, this can be shown to be

$$\lim_{n\to\infty} n\gamma_n^k V\left[\hat{h}(x)\right] = h(x)\int \left[K(z^*)\right]^2 dz^*.$$

Since both $\int \left[K(z^*)\right]^2 dz^*$ and $h(x)$ are bounded, the RHS is bounded, and since $n\gamma_n^k \to \infty$ by assumption, we have that

$$V\left[\hat{h}(x)\right] \to 0.$$

Since the bias and the variance both go to zero, we have **pointwise consistency** (convergence in quadratic mean implies convergence in probability).

Estimation of the numerator

To estimate $\int yf(x,y)dy$, we need an estimator of $f(x,y)$. The estimator has the same form as the estimator for $h(x)$, only with one dimension more:

$$\hat{f}(x,y) = \frac{1}{n}\sum_{t=1}^{n} \frac{K_*\left[(y-y_t)/\gamma_n, (x-x_t)/\gamma_n\right]}{\gamma_n^{k+1}}$$

The kernel $K_*(\cdot)$ is required to have mean zero:

$$\int yK_*(y,x)\,dy = 0$$

and to marginalize to the previous kernel for $h(x)$:

$$\int K_*(y,x)\,dy = K(x).$$

With this kernel, we have (not obviously, see Li and Racine, Ch. 2, section 2.1)

$$\int y\hat{f}(y,x)dy = \frac{1}{n}\sum_{t=1}^{n} y_t \frac{K\left[(x-x_t)/\gamma_n\right]}{\gamma_n^k}$$

by marginalization of the kernel, so we obtain

$$\begin{aligned}\hat{g}(x) &:= \frac{1}{\hat{h}(x)}\int y\hat{f}(y,x)dy \\ &= \frac{\frac{1}{n}\sum_{t=1}^{n} y_t \frac{K[(x-x_t)/\gamma_n]}{\gamma_n^k}}{\frac{1}{n}\sum_{t=1}^{n} \frac{K[(x-x_t)/\gamma_n]}{\gamma_n^k}} \\ &= \frac{\sum_{t=1}^{n} y_t K\left[(x-x_t)/\gamma_n\right]}{\sum_{t=1}^{n} K\left[(x-x_t)/\gamma_n\right]}\end{aligned}$$

This is the Nadaraya-Watson kernel regression estimator.

Kernel Regression

Defining:

$$w_t = \frac{K\left[(x-x_t)/\gamma_n\right]}{\sum_{t=1}^{n} K\left[(x-x_t)/\gamma_n\right]},$$

the kernel regression estimator for $g(x_t)$ can be written as

$$\hat{g}(x) = \sum_{t=1}^{n} y_t w_t,$$

a weighted average of the $y_j,\ j=1,2,\dots,n$, where higher weights are associated with points that are closer to x_t.

- The window width parameter γ_n imposes smoothness. The estimator is increasingly flat as $\gamma_n \to \infty$, since in this case each weight tends to $1/n$.
- A large window width reduces the variance (strong imposition of flatness), but increases the bias.
- A small window width reduces the bias, but makes very little use of information except points that are in a small neighborhood of x_t. Since relatively little information is used, the variance is large when the window width is small.
- The standard normal density is a popular choice for $K(.)$ and $K_*(y,x)$, though there are possibly better alternatives.

One can choose the window width using Cross-validation.

Example: Nadaraya-Watson Estimator with Guassian Kernel and Silverman's Rule of Thumb Window

In [65]:

```
n=1000
k=2
```

```julia
ydata = rand(Bernoulli(0.5),n)
xdata = randn(n,k);
```

In [77]:

```julia
function nw_ap(xeval,xdata,ydata)
    h = 1.06.*std(xdata,dims=1).*(size(xdata,1)^-0.2) #Silverman's rule of thumb
    input = (repeat(randn(2)',size(xdata,1)) .- xdata) ./ repeat(h,size(xdata,1))
    phi = exp.(-0.5.*(input).^2)./sqrt(2*pi)
    phiprod = prod(phi,dims=2).*(1/(prod(h)*size(xdata,1)))

    return sum(phiprod.*ydata) / sum(phiprod)

end
```

Out[77]:

```
nw_ap (generic function with 1 method)
```

In [86]:

```julia
nw_ap(randn(2),xdata,ydata)
```

Out[86]:

```
0.47560938381448953
```