Selecting bandwidth for kernel estimator:

eqn (30) of Wasserman, CMU lecture 36-708 Statistical Methods for Machine Learning

https://www.stat.cmu.edu/~larry/=sml/densityestimation.pdf

see Chap 20 of Wasserman's "All of Statistics, A Concise Course in Statistical Inference" for various risk estimators and bandwidth selectors.

* caveat: it's missing details like data splitting that are part of the equations, but, that he has in his lecture notes:

https://www.stat.cmu.edu/~larry/=sml/densityestimation.pdf

36-708 Statistical Methods for Machine Learning by Larry Wasserman, CMU

* regarding loss functions in context of estimating kernel densities:

Kullback-Leibler is not a good loss function to use for nonparametric density estimation because it is completely dominated by the tails of the densities. The use of an assumed gaussian for the unknown distribution can also be problematic for non-symmetric distributions, including multi-modal.

* end of chapter 20 footnote: For large data sets the kernel density estimator, and (20.25) can be computed quickly using the fast Fourier transform.

see Chap 6 of Bishop's "Pattern Recognition and Machine Language"

see Section 5 of "A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation"
S. J. Sheather,M. C. Jones, 1991, J.R. Statistic Society B, Volume 53, Issue 3, 1991, Pages 683-690

A fast implementation using FFT

https://kdepy.readthedocs.io/en/latest/introduction.html#Selecting-a-suitable-bandwidth

see wikipedia for a single mode data, gaussian estimate:

https://en.m.wikipedia.org/wiki/Kernel_density_estimation#A_rule-of-thumb_bandwidth_estimator

Cross-validation methods can be used for choosing the bandwidth h.

https://www.stat.cmu.edu/~larry/=sml/densityestimation.pdf

36-708 Statistical Methods for Machine Learning by Larry Wasserman, CMU

the kernel density estimator is defined to be

$$\widehat{p}(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h^d}K\left(\frac{\|x - X_i\|}{h}\right). \tag{10}$$

More generally, we define

$$\widehat{p}_H(x) = \frac{1}{n}\sum_{i=1}^{n} K_H(x - X_i)$$

where H is a positive definite bandwidth matrix and $K_H(x) = |H|^{-1/2}K(H^{-1/2}x)$. For simplicity, we will take $H = h^2 I$ and we get back the previous formula.

Sometimes we write the estimator as $\widehat{p}_h$ to emphasize the dependence on h. In the multivariate case the coordinates of X_i should be standardized so that each has the same variance, since the norm $\|x - X_i\|$ treats all coordinates as if they are on the same scale.

Data splitting:
k-fold cross-validation (a.k.a. V-fold C.V.)
https://rsample.tidymodels.org/reference/vfold_cv.html

randomly splits the data into V groups of roughly equal size (called "folds"). A resample of the analysis data consists of V-1 of the folds while the assessment set contains the final fold.
In basic V-fold cross-validation (i.e. no repeats), the number of resamples is equal to V.

With more than one repeat, the basic V-fold cross-validation is conducted each time. For example, if three repeats are used with v = 10, there are a total of 30 splits: three groups of 10 that are generated separately.

based on bandwidth h. For simplicity, assume the sample size is even and denote the sample size by $2n$. Randomly split the data $X = (X_1, \ldots, X_{2n})$ into two sets of size n. Denote these by $Y = (Y_1, \ldots, Y_n)$ and $Z = (Z_1, \ldots, Z_n)$.[1] Let $\mathcal{H} = \{h_1, \ldots, h_N\}$ be a finite grid of bandwidths. Let

$$\widehat{p}_j(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h_j^d}K\left(\frac{\|x - Y_i\|}{h}\right).$$

Thus we have a set $\mathcal{P} = \{\widehat{p}_1, \ldots, \widehat{p}\}$ of density estimators.

We would like to minimize $L(p, \widehat{p}_j) = \int \widehat{p}_j^2(x) - 2\int \widehat{p}_j(x)p(x)dx$. Define the estimated risk

$$\widehat{L}_j \equiv \widehat{L}(p, \widehat{p}_j) = \int \widehat{p}_j^2(x) - \frac{2}{n}\sum_{i=1}^{n}\widehat{p}_{j(i)}(Z_i). \tag{30}$$

Let $\widehat{p} = \text{argmin}_{g\in\mathcal{P}}\widehat{L}(p, g)$. Schematically:

$$X = (X_1, \ldots, X_{2n}) \xrightarrow{\text{split}} \begin{array}{l} Y \to \{\widehat{p}_1, \ldots, \widehat{p}_N\} = \mathcal{P} \\ Z \to \{\widehat{L}_1, \ldots, \widehat{L}_N\} \end{array}$$