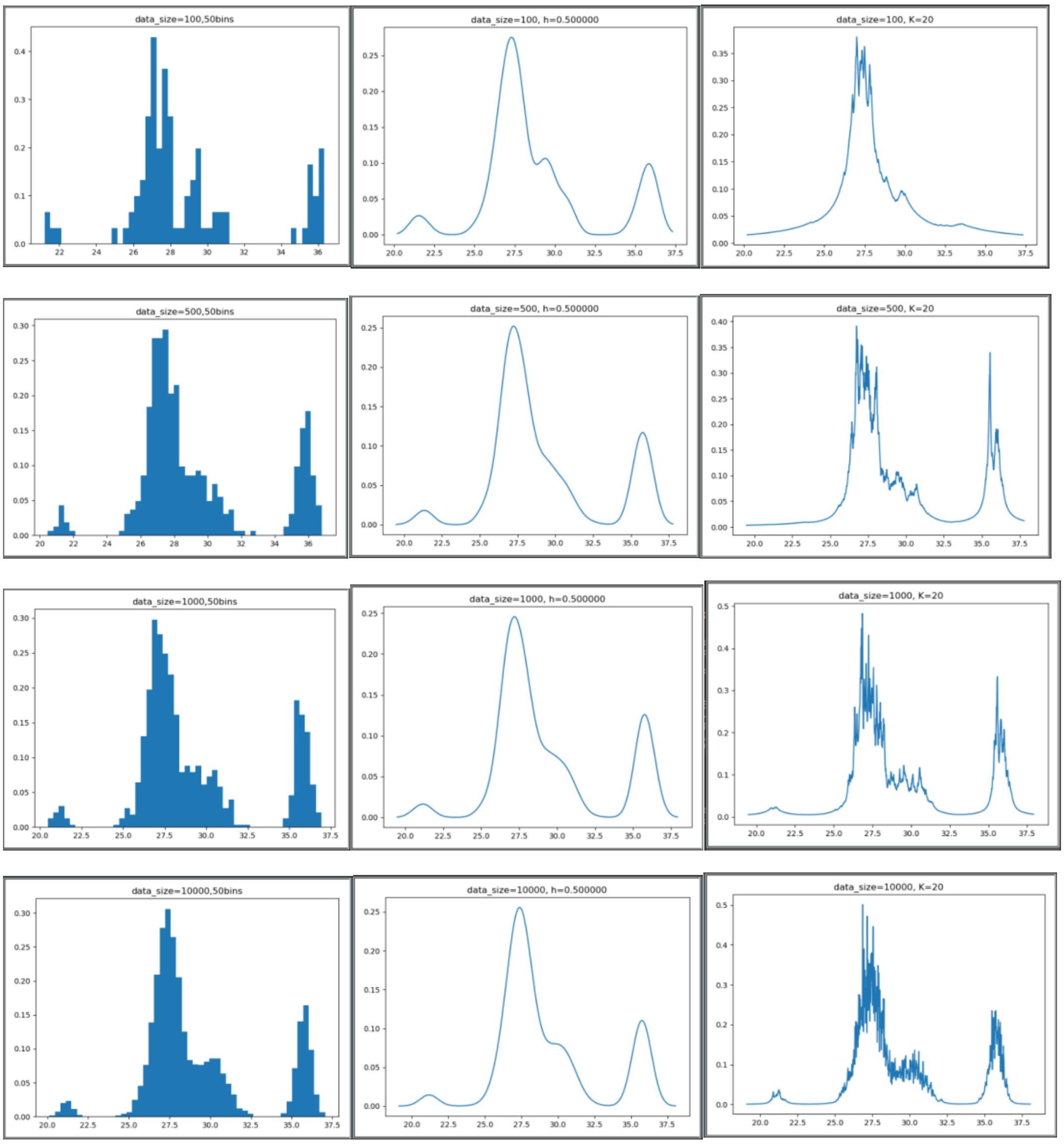
Assignment 1 report

- Readme

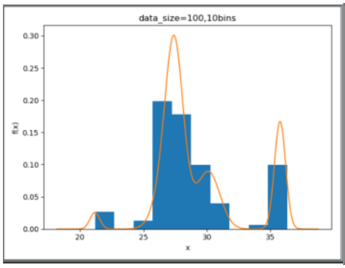
python source.py <histogram/gaussian/KNN> <model parameter> <sample_data_size>
eg. python source.py histogram 50 100

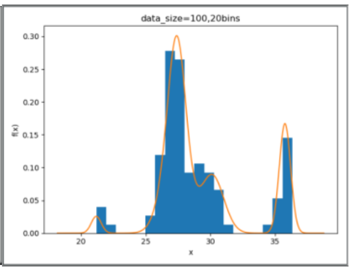
- How does the number of data influence the quality of the estimation?

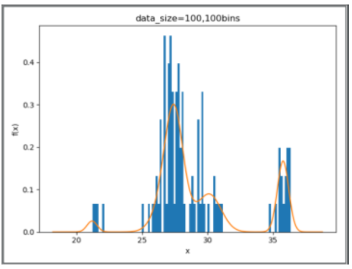


Generally, as the number of data increases, the quality of the estimation gets better, as the law of large numbers goes. However, for KNN, the model is largely dependent on K, and even a large number of training data can't make the curve smooth.

- How does the number of bins affect the estimation? Please answer: how could you pick the best (or good) choice for this number of bins?



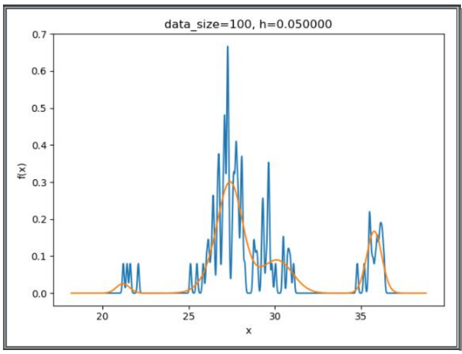


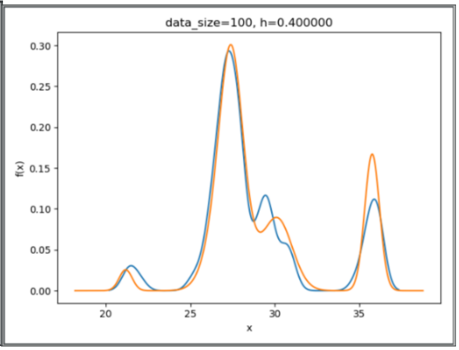


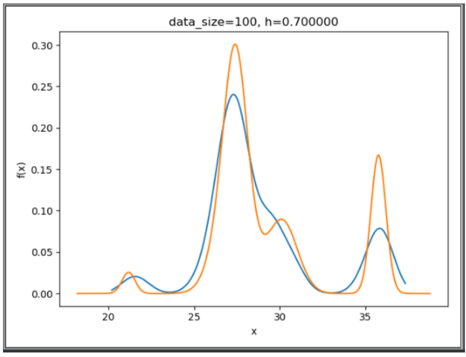
If there are **too many bins**, the resulting density model is too **spiky**, with a lot of structure that is not present in the underlying distribution that generated the data set. If there are **too few bins**, the result is a model that is **too smooth** and that consequently **fails to capture all the features of the true distribution.**

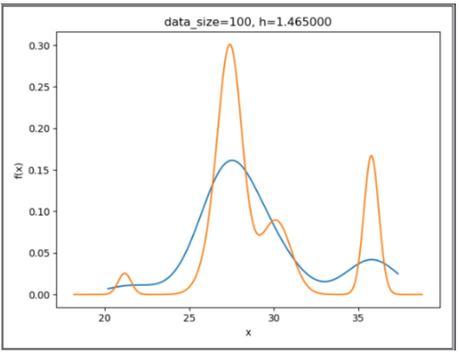
The best results are obtained for some intermediate value of the number of bins. For example, about **1/5 to 1/4 of the size of the sample data**.

- Tune h to see what will happen, answer if you have a clue of how to choose h , plot the best estimate you could achieve with num_data=100.









h acts as a **smoothing parameter**. If it is set too small, the result is a very noisy density model. If it is set too large, the underlying features of the true distribution is washed out.

Define an error function to represent the difference between our estimation and the true distribution, then minimize the error function. Choose the **mean integrated squared error**

$$\mathbf{MISE(h) = E\left[\int \left(f_h(x) - f(x)\right)^2 dx\right]}$$

Let $\boldsymbol{\frac{\partial}{\partial h} MISE(h) = 0}$, and we get the best choice of h.

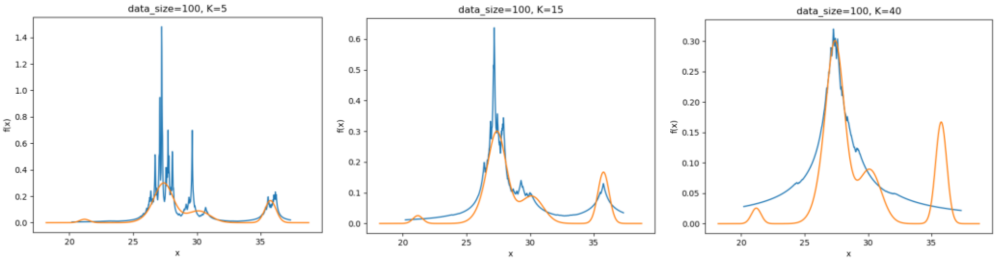
According to *Silverman, B.W. (1986). Density Estimation for Statistics and Data Analysis. London: Chapman & Hall/CRC. p. 48. ISBN 0-412-24620-1,* the **empirical choice of h for gaussian kernel**

$\mathbf{h} = \left(\frac{4\sigma^5}{3n}\right)^{\frac{1}{5}} \approx \mathbf{1.06}\boldsymbol{\sigma n}^{-\frac{1}{5}}$, where σ is the std of sample data. **The std of the 100 sampled data is 3.473, therefore h is 1.465. However, this just can't fit the true distribution well, and it seems more like a gaussian distribution with a single peak.**

Or we can use **variable kernel density estimation**. In a balloon estimator, the kernel width is varied depending on the location of the test point. In a pointwise estimator, the kernel width is varied depending on the location of the sample(from wiki). Another effective optimization method may be cross-validation.

In this case, I try to **find the best h through binary search**, and evaluate the performance of the kernel function empirically. The best estimation is when h=0.4, as shown above in figure 2.

- Vary k to see the difference, you are encouraged to plot an illustration and the true distribution. show that the nearest neighbor method does not always yield a valid distribution, the sum of probabilities doesn't equal to 1.



The model produced by K nearest neighbors is not a true density model because **the integral over all space diverges.**

$p(x) = \frac{K}{NV(\rho)}$ ρ is the distance from x to its K^{th} nearest neighbor in the data set.

$V(\rho)$ is the volume of a D-dimensional hypersphere with radius ρ.

Therefore, in polar coordinates, $p(x) \propto r^{-D}$

Consider the integral of p(x) over all space, $\int p(x)dx \propto \int r^{-D}r^{D-1}dr \propto \int r^{-1}dr = \infty$