# ASSIGNMENT 1

Pattern Recognition and Machine Learning

(模式识别与机器学习)

COMP130137.01

Name: ANON

Fudan-ID: XXX

10 april 2019

# 1 Overview

## 1.1 Introduction

In this assignment, we are going to use the three non-parametric density estimation algorithms (namely histogram method, kernel density estimate and the nearest neighbour method) to estimate the distribution of the given data set.

## 1.2 Structure

```
assignment-1/                                                    1
     -- 16349086036/          // Users report and sorce code     2
          -- report.pdf                                          3
          -- source.py                                           4
     -- handout/                                                 5
          -- __init__.py       // Provided base file            6
```

## 1.3 Usage

```
python3 source.py --{command} {value}                           1
```

See details in source code.

## 1.4 Packages (Beyond default python)

The following external packages were used in the experiment: numpy, matplotlib, pandas or scipy. Which can be installed easily via pip.

```
$ sudo pip install {PACKAGE_NAME}                               1
```

### 1.5 Requirements

In this report, there were in total four different requirements needed to satisfy during the estimation of the distribution. The list (The tasks in its original form) is provided below:

1. (10%) For all three algorithms, you should vary the number of data used, lets say you could use 100, 500, 1000, 10000 to see what will happen for your estimation, you don't have to report 3x 4 = 12 plots just make an empirical assertion about how does the number of data influence the quality of the estimation. In the rest part of the requirements, if not specified, use num_data = 200 for exploration (or other number you think is better for your exploration, please state clear and keep consistent if so).

2. (20%) For histogram estimation, you could vary the number of bins used to locate the data samples to see how this parameter affect the estimation. Please answer: how could you pick the best (or good) choice for this number of bins?

3. (30%) For kernel density estimation, you should try the Gaussian kernel

$$p(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{(2\pi h^2)^{1/2}.}\exp\{-\frac{||\mathbf{x}-\mathbf{x}_n||}{2h^2}\} \tag{1}$$

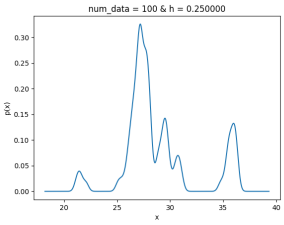
   Please also try to tune $h$ to see what will happen, answer if you have a clue of how to choose $h$, plot the best estimate you could achieve with num_data=100.

4. (40%) For nearest neighbour methods, you should vary $K$ in $p(\mathbf{x}) = \frac{K}{NV}$ to see the difference, you are encouraged to plot an illustration as Figure 2.26 in the text book, to plot the true distribution, see GaussianMixture1D.plot in the handout for more details. Additionally, please show that the nearest neighbour method does not always yield a valid distribution, i.e. it won't converge to 1 if you sum the probability mass over all the space (in our case $(-\infty, \infty)$), you could show this either empirically or theoretically.

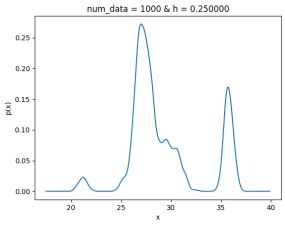
# 2 Experiments

# 3 Task 1

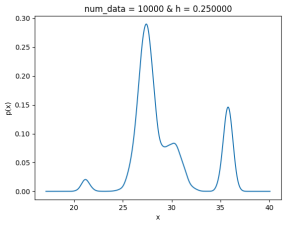
The first task is to plot the graphs of the three different algorithms and evaluate how the number of data used would influence the estimation. The data I will show in this report is 100, 1000 and 10000. Even though not required, I decided to plot out all of the combinations in this report. The result can be seen the following images:



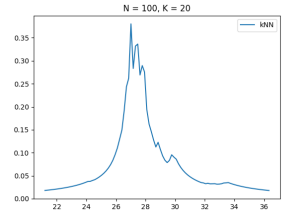

(a) Kernel n=100, h=0.25



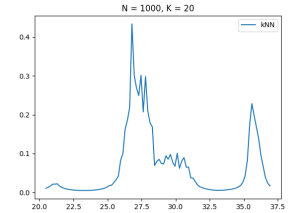

(b) Kernel n=1000, h=0.25



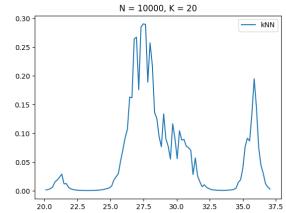

(c) Kernel n=10000, h=0.25



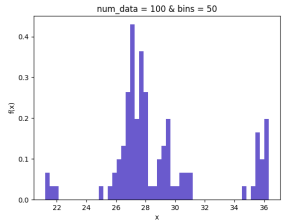

(d) knn n=100,k=20



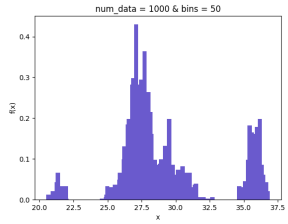

(e) knn n=1000,k=20



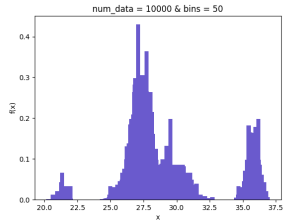

(f) knn n=10000,k=20




(g) hist n=100, h=50




(h) hist n=1000, h=50

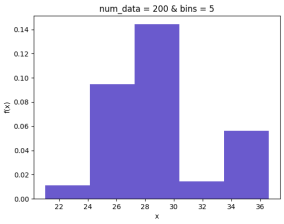



(i) hist n=10000, h=50

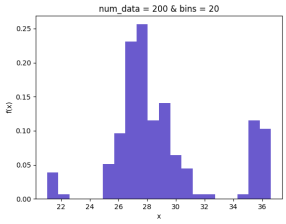
It is clear that when more data is used, our estimation will be more closer to the actual underling distribution.

# 4 Task 2

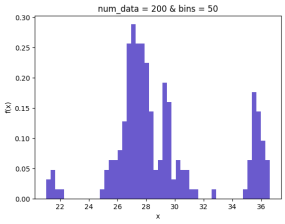
In this task, we will study the histogram estimation by vary the number of bins used to understand how this would influence the parameter and thus affect the estimation. We will also study how to find a good choice of the number of bins. Comparing Bins= $5, 20, 50$ graph shows that bins size selection is closely related to the estimation effect of close size selection and estimation effect. By plotting out different variations, we can see that after




(a) hist n=200, h=5




(b) hist n=200, h=20




(c) hist n=200, h=50

20 number of bins, the result is pretty much the same. By using a visual decision, I though that between 15-20 bins was a good number.

```
python3 source.py --histogram --num_bins 25 --num_samp 10000
```
1

# 5 Task 3

In this task, we will study the Kernel density estimation (Gaussian Kernel). More specificity, we will understand how find the best estimation of the variable $h$.

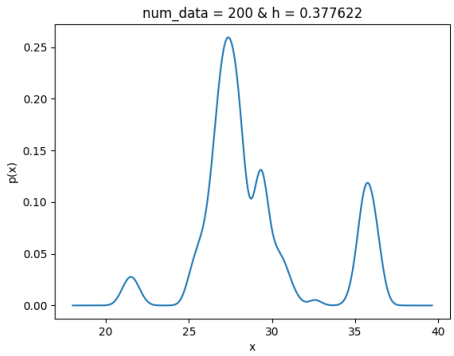
In this function, variable $h$ is a smoothing parameter called the bandwidth and is $h \geq 0$. We can start by experimenting what will happen when we variate the value of $h$ in the below graphs.

It is clear that when $h$ is to small, leads to a thinner, more peaked and when $h$ is to large the graph becomes over smoothed. We can use maximum likelihood $L(h)$ function to find the value

$$p(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{(2\pi h^2)^{1/2.}}\exp\{-\frac{||\mathbf{x}-\mathbf{x}_n||}{2h^2}\} \quad (2)$$

$$L(h) = \prod_{i=1}^{n} p(\mathbf{x}) \quad (3)$$

The result was $h = 0.3776223$ with the graph:



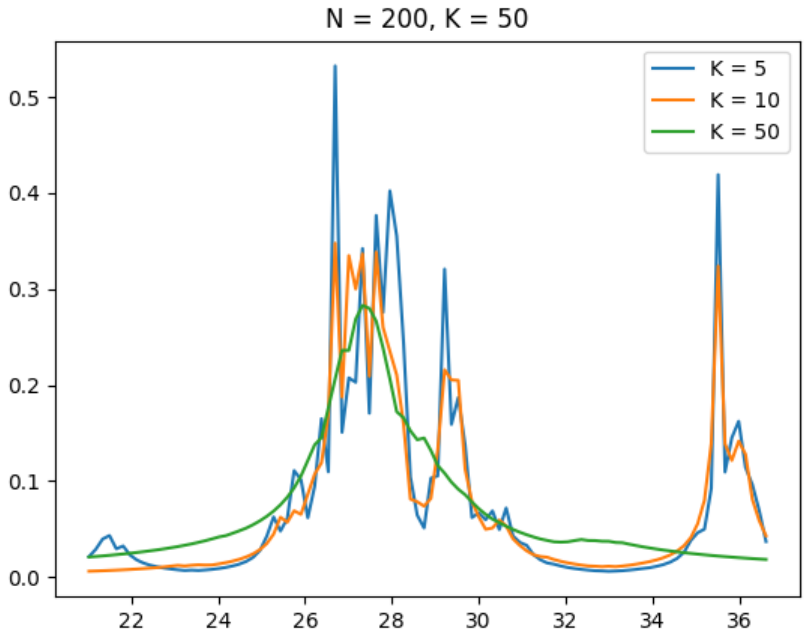

(a) Kernel n=200, h=0.377

```
$ python3 source.py --opt_h
```

1

# 6 Task 4

In this task we will explore the nearest neighbour method and show that the nearest neighbour hood does not always yield a valid distribution. Try $k = 5, 10, 50$, we plot the image of the k nearest neighbour estimate:
It can be seen that the smaller one can be seen that a smaller k=5 will result




(a) knn plot

in a larger noise density model, and a larger noise density model will result, resulting in a larger noise density model, while k=50. Will be so smooth that some of the features of the original distribution are ignored.

KNN method will not converge to 1 if we sum the probability mass over all the space. Here I will demonstrate it. First, we estimate each position density by such formula: $p(x) = (\frac{k}{N}) \cdot \frac{1}{V}$

Regarding the convergence of the k-nearest neighbor estimate, assume that

the maximum value of the sample data is m and the minimum value is n. Now determine the calculation of V in the formula. According to V, the definition of the sphere volume centered on x contains exactly k data points. In the one-dimensional case, V should be twice the distance from x to the kth data point e.g V=2(m-n).

We then have $p(x) = (\frac{k}{N}) \cdot \frac{1}{V} = (\frac{k}{N}) \cdot \frac{1}{2(m-n)}$

$$\int_{-\infty}^{\infty} p(x)dx \geq \int_{-r}^{\infty} p(x)dx \geq \int_{-r}^{\infty} \frac{K}{N \cdot 2(x-l)}dx = \frac{K}{2 \cdot N} \int_{r-1}^{\infty} \frac{1}{x}dx \to \infty \quad (4)$$

Thus we can see that the KKN metod does not converge to 1 if we sum the probability mass over all the space.