# DENSITY ESTIMATION
# PARZEN WINDOW
# K-NEAREST-NEIGHBOR ESTIMATION

# Nonparametric Methods

- In parametric methods, whether for density estimation, classification, or regression, we assume a model valid over the whole input space.
- In regression, for example, when we assume a linear model, we assume that for any input, the output is the same linear function of the input.
- In classification when we assume a normal density, we assume that all examples of the class are drawn from this same density.
- The advantage of a parametric method is that it reduces the problem of estimating a probability density function, discriminant, or regression function to estimating the values of a small number of parameters.
- Its disadvantage is that this assumption does not always hold and we may incur a large error if it does not.

- If we cannot make such assumptions and cannot come up with a parametric model.
- In *nonparametric estimation, all we assume is that similar inputs have similar outputs.*
- This is a reasonable assumption: the world is smooth and functions, whether they are densities, discriminants, or regression functions, change slowly.
- Similar instances mean similar things.
- We all love our neighbors because they are so much like us.
- Therefore, our algorithm is composed of finding the similar past instances from the training set using a suitable distance measure and interpolating from them to find the right output.

- Different nonparametric methods differ in the way they define similarity or interpolate from the similar training instances.
- In a parametric model, all of the training instances affect the final global estimate.
- Whereas in the nonparametric case, there is no single global model;
- Local models are estimated as they are needed, affected only by the nearby training instances.
- Nonparametric methods do not assume any a priori parametric form for the underlying densities;
- A nonparametric model is not fixed but its complexity depends on the size of the training set, or rather, the complexity of the problem inherent in the data.

- In machine learning literature, nonparametric methods are also called *instance-based or instance-based memory-based learning algorithms.*
- Such methods are also called *lazy learning algorithms, because* unlike the *eager parametric models.*
- *They do not compute a model when* they are given the training set but postpone the computation of the model until they are given a test instance.

# Nonparametric Density Estimation

- As usual in density estimation, we assume that the sampleX ={*xt}Nt*=1 is drawn independently from some unknown probability density *p(·).*
- ˆ*p(·)* is our estimator of *p(·).*
- The nonparametric estimator for the cumulative distribution function, *F(x), at point x is the proportion of sample points that are less than or* equal to *x.*

$$F(x) = \#\{xt \leq x\}/N$$

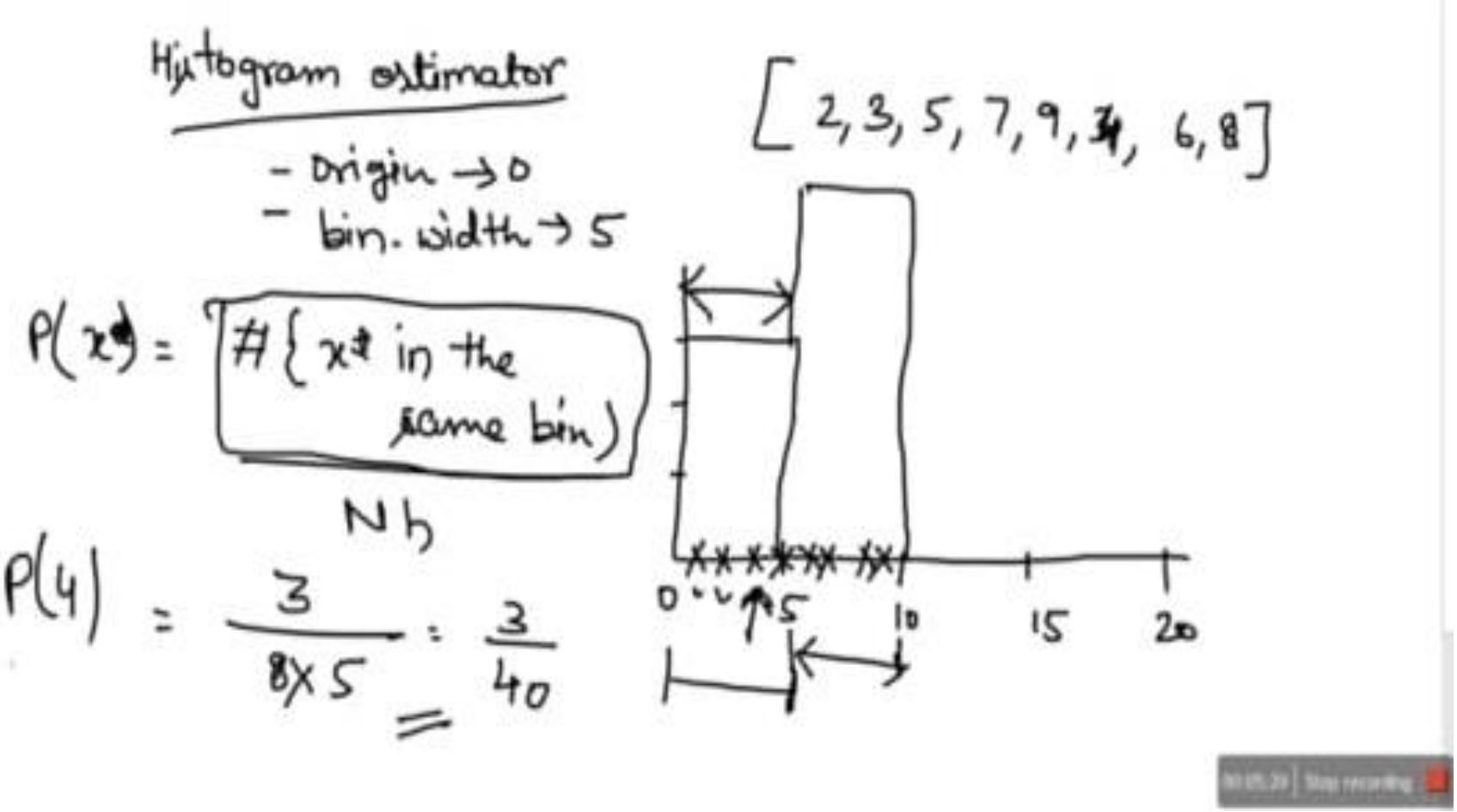
- where #{*xt ≤ x} denotes the number of training instances whose xt is* less than or equal to *x.*

- Similarly, the nonparametric estimate for the density function can be calculated a

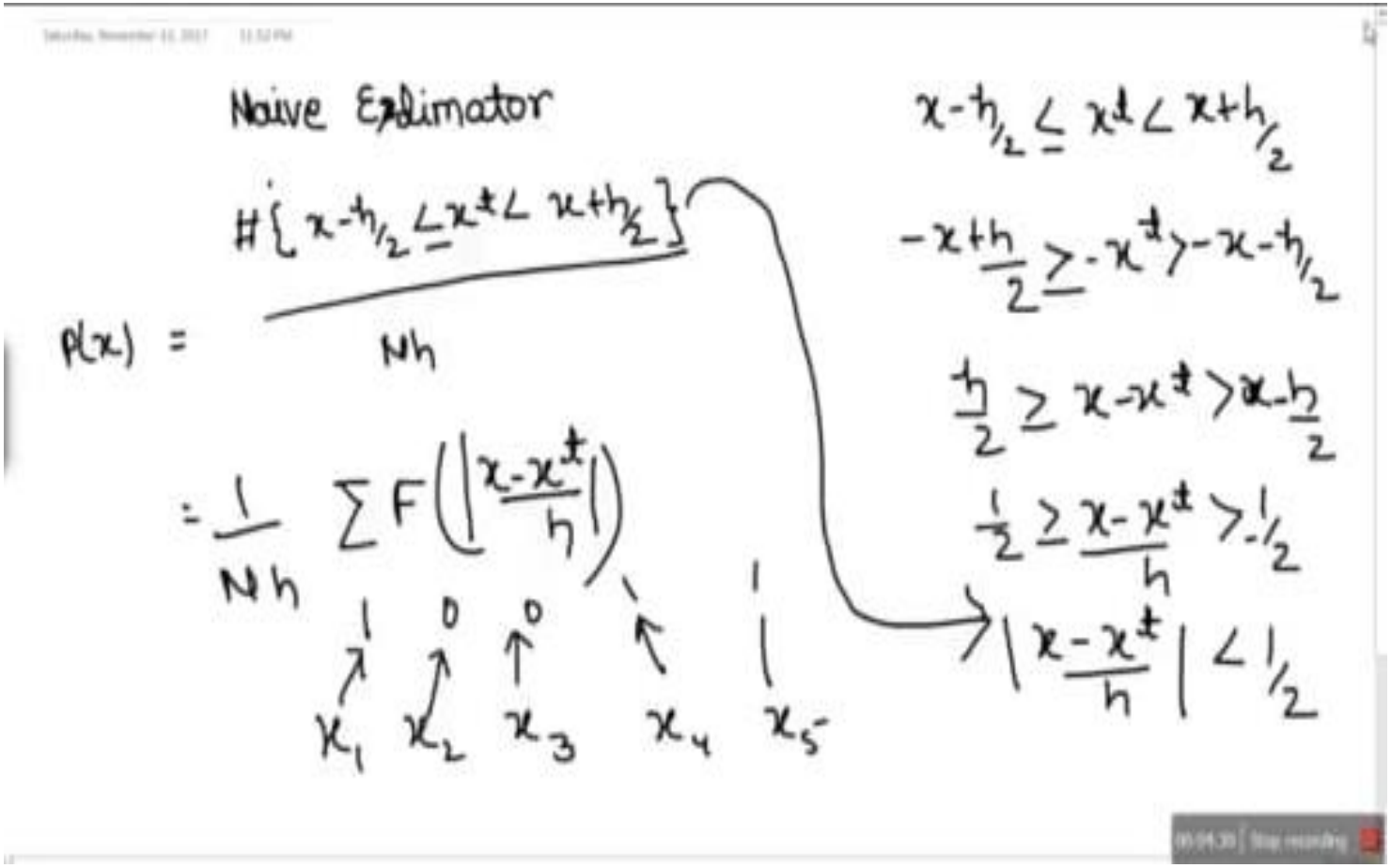
$$\hat{p}(x) = \frac{1}{h}\left[\frac{\#\{x^t \le x + h\} - \#\{x^t \le x\}}{N}\right]$$

- *Where h is the length of the interval and instances xt that fall in this interval* are assumed to be "close enough."
- Different types of nonparametric estimation methods are:-
  - Histogram estimation
  - Naive estimation
  - Kernel Estimation

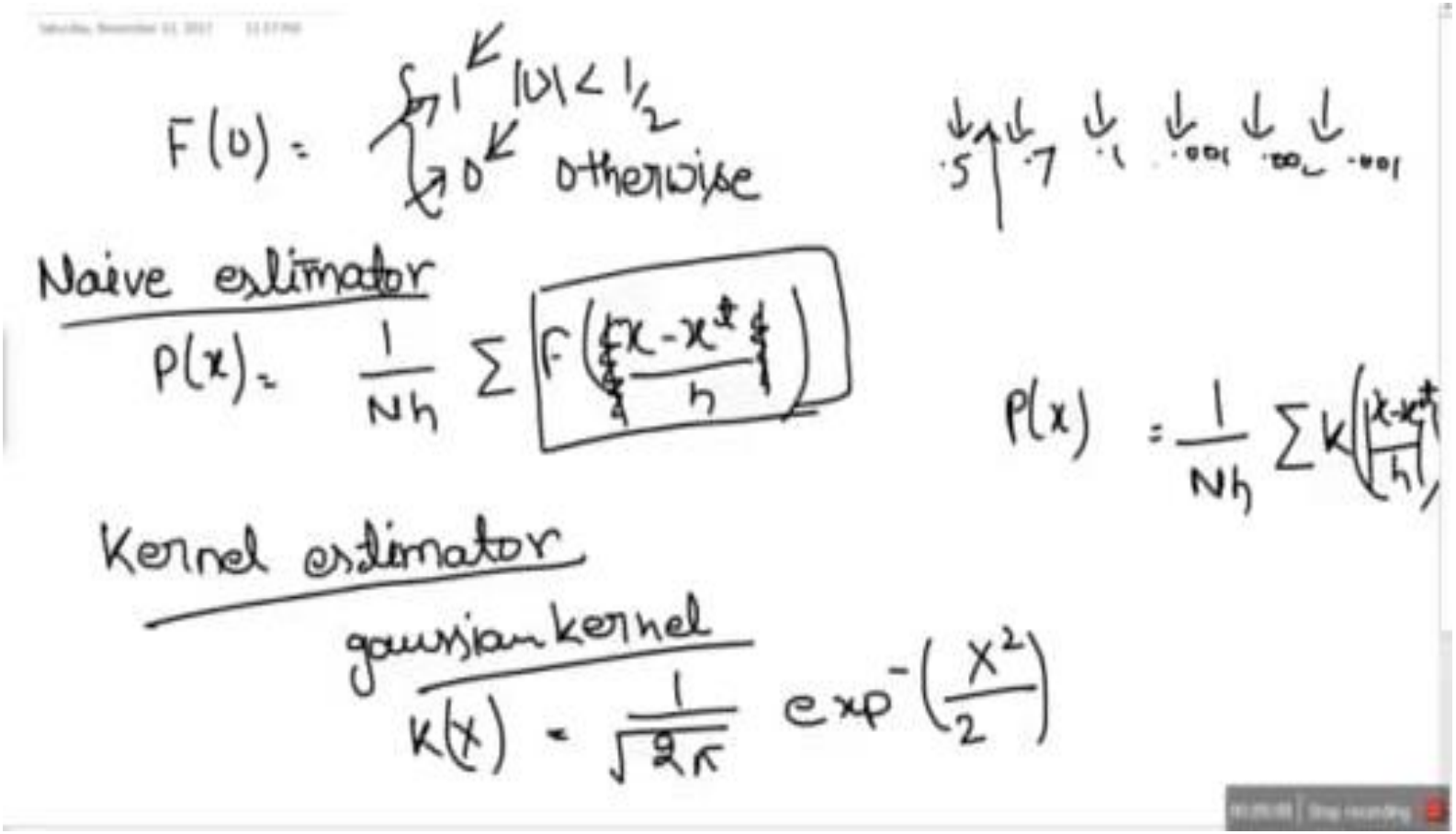
# Histogram Estimator



# Naive Estimator



# Kernel Estimator



# Non-Parametric Techniques

- **All Parametric densities are unimodal (have a single local maximum), whereas many practical problems involve multi-modal densities.**

- ***Nonparametric procedures* can be used with *arbitrary distributions* and without the assumption that the forms of the underlying densities are known**

Neither probability distribution nor discriminant function is known

- Happens quite often

All we have is labeled data


Estimate the probability distribution from the labeled data

# Probability density function (pdf)

- The mathematical definition of a continuous probability function, p(x), satisfies the following properties:
  - The probability that x is between two points a and b.

$$P(a < x < b) = \int_a^b p(x)dx$$

  - It is non-negative for all real x.
  - The integral of the probability function is one, that is

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

- The most commonly used probability function is Gaussian function (also known as Normal distribution).

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-c)^2}{2\sigma^2}\right)$$

- where c is the mean, $\sigma^2$ is the variance and is the $\sigma$ standard deviation.

# Density estimation

- Given a set of n data samples x1,..., xn, we can estimate the density function p(x), so that we can output p(x) for any new sample x. This is called density estimation.
- The basic ideas behind many of the methods of estimating an unknown probability density function are very simple. The most fundamental techniques rely on the fact that the probability P that a vector falls in a region R is given by

$$P = \int_{\mathcal{R}} p(\mathbf{x})\, d\mathbf{x}$$

- If we now assume that R is so small that p(x) does not vary much within it, we can write

$$P = \int_{\mathcal{R}} p(\mathbf{x})\, d\mathbf{x} \approx p(\mathbf{x}) \int_{\mathcal{R}} d\mathbf{x} = p(\mathbf{x})V$$

- where V is the “volume” of R.

- On the other hand, suppose that n samples x1,..., xn are independently drawn according to the probability density function p(x), and there are k out of n samples falling within the region R, we have

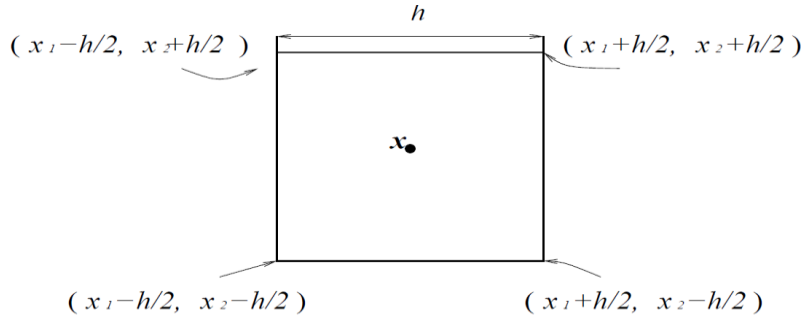
$$P = k/n$$

- Thus we arrive at the following obvious estimate for p(x),

$$p(\mathbf{x}) = \frac{k/n}{V}$$

# Parzen window density estimation

- Consider that R is a hypercube centered at x (think about a 2-D square). Let h be the length of the edge of the hypercube, then $V = h^2$ for a 2-D square, and $V = h^3$ for a 3-D cube.




Introduce

$$\phi(\frac{\mathbf{x}_i - \mathbf{x}}{h}) = \begin{cases} 1 & \frac{|x_{ik} - x_k|}{h} \leq 1/2, \quad k = 1, 2 \\ 0 & otherwise \end{cases}$$

which indicates whether $\mathbf{x}_i$ is inside the square (centered at $\mathbf{x}$, width $h$) or not.

The total number $k$ samples falling within the region $\mathcal{R}$, out of $n$, is given by

$$k = \sum_{i=1}^{n} \phi(\frac{\mathbf{x}_i - \mathbf{x}}{h})$$

The Parzen probability density estimation formula (for 2-D) is given by

$$p(\mathbf{x}) = \frac{k/n}{V}$$

$$= \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h^2} \phi(\frac{\mathbf{x}_i - \mathbf{x}}{h})$$

$\phi(\frac{\mathbf{x}_i - \mathbf{x}}{h})$ is called a window function.

We can generalize the idea and allow the use of other window functions so as to yield other Parzen window density estimation methods. For example, if Gaussian function is used, then (for 1-D) we have

$$p(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x_i - x)^2}{2\sigma^2}\right)$$

This is simply the average of $n$ Gaussian functions with each data point as a center. $\sigma$ needs to be predetermined.

# Parzen Windows

The Parzen-window approach to estimating densities can be introduced by temporarily assuming that the region $\mathcal{R}_n$ is a $d$-dimensional hypercube. If $h_n$ is the length of an edge of that hypercube, then its volume is given by

$$V_n = h_n^d. \tag{8}$$

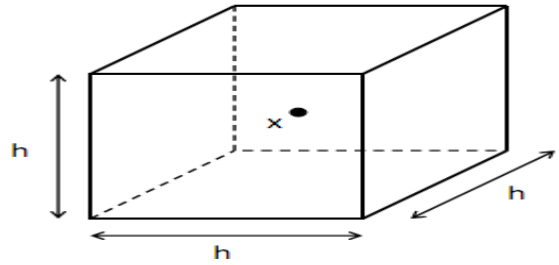
We can obtain an analytic expression for $k_n$, the number of samples falling in the hypercube, by defining the following *window function*:

$$\varphi(\mathbf{u}) = \begin{cases} 1 & |u_j| \leq 1/2 \qquad j = 1, ..., d \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Thus, $\varphi(\mathbf{u})$ defines a unit hypercube centered at the origin. It follows that $\varphi((\mathbf{x} - \mathbf{x}_i)/h_n)$ is equal to unity if $\mathbf{x}_i$ falls within the hypercube of volume $V_n$ centered at $\mathbf{x}$, and is zero otherwise. The number of samples in this hypercube is therefore given by

$$k_n = \sum_{i=1}^{n} \varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right), \tag{10}$$

$$p_n(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{V_n} \varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right).$$




window function itself be a density function. To be more precise, if we require that

$$\varphi(\mathbf{x}) \geq 0 \tag{12}$$

and

$$\int \varphi(\mathbf{u})\, d\mathbf{u} = 1, \tag{13}$$

and if we maintain the relation $V_n = h_n^d$, then it follows at once that $p_n(\mathbf{x})$ also satisfies these conditions.

Let us examine the effect that the *window width* $h_n$ has on $p_n(\mathbf{x})$. If we define the function $\delta_n(\mathbf{x})$ by

$$\delta_n(\mathbf{x}) = \frac{1}{V_n}\varphi\left(\frac{\mathbf{x}}{h_n}\right), \tag{14}$$

then we can write $p_n(\mathbf{x})$ as the average

$$\boxed{p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\delta_n(\mathbf{x}-\mathbf{x}_i).} \tag{15}$$

Since $V_n = h_n^d$, $h_n$ clearly affects both the amplitude and the width of $\delta_n(\mathbf{x})$ (Fig. 4.3).

Example: Given a set of five data points $x_1 = 2$, $x_2 = 2.5$, $x_3 = 3$, $x_4 = 1$ and $x_5 = 6$, find Parzen probability density function (pdf) estimates at $x = 3$, using the Gaussian function with $\sigma = 1$ as window function.

$$\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_1 - x)^2}{2}\right)$$

$$= \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(2-3)^2}{2}\right) = 0.2420$$

$$\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_2 - x)^2}{2}\right)$$

$$= \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(2.5-3)^2}{2}\right) = 0.3521$$

$$\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_3 - x)^2}{2}\right) = 0.3989$$

$$\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_4 - x)^2}{2}\right) = 0.0540$$

$$\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_5 - x)^2}{2}\right) = 0.0044$$

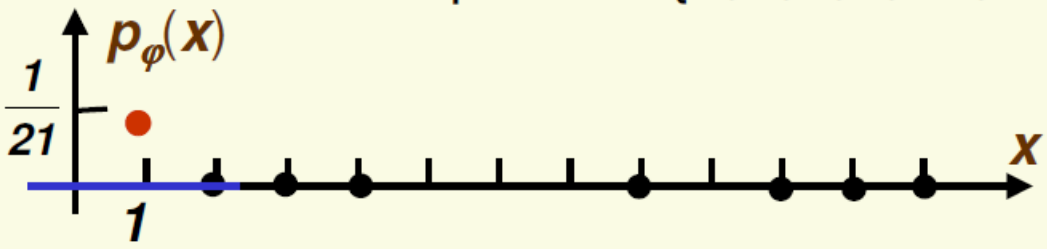
$$p(x = 3) = (0.2420 + 0.3521 + 0.3989$$

$$+0.0540 + 0.0044)/5 \quad = 0.2103$$

# Parzen Windows: Example in 1D

$$p_\varphi(x) = \frac{1}{n}\sum_{i=1}^{i=n} \frac{1}{h^d}\,\varphi\left(\frac{x - x_i}{h}\right)$$

- Suppose we have 7 samples $\boldsymbol{D}$={2,3,4,8,10,11,12}




- Let window width $\boldsymbol{h}$=3, estimate density at x=1

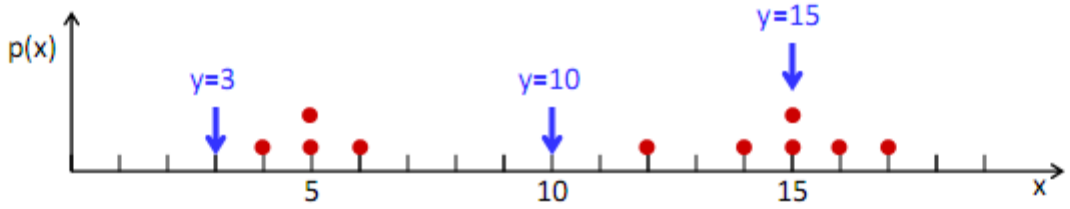
$$p_\varphi(1) = \frac{1}{7}\sum_{i=1}^{i=7} \frac{1}{3}\,\varphi\left(\frac{1 - x_i}{3}\right) = \frac{1}{21}\left[\varphi\left(\frac{1-2}{3}\right) + \varphi\left(\frac{1-3}{3}\right) + \varphi\left(\frac{1-4}{3}\right) + \ldots + \varphi\left(\frac{1-12}{3}\right)\right]$$

$$\left|-\frac{1}{3}\right| \le 1/2 \qquad \left|-\frac{2}{3}\right| > 1/2 \qquad |-1| > 1/2 \qquad \left|-\frac{11}{3}\right| > 1/2$$

$$p_\varphi(1) = \frac{1}{7}\sum_{i=1}^{i=7} \frac{1}{3}\,\varphi\left(\frac{1 - x_i}{3}\right) = \frac{1}{21}[1 + 0 + 0 + \ldots + 0] = \frac{1}{21}$$

## Exercise

- Given dataset $X = \{4, 5, 5, 6, 12, 14, 15, 15, 16, 17\}$, use Parzen windows to estimate the density $p(x)$ at $y = 3,10,15$; use $h = 4$
- Solution
  - Let's first draw the dataset to get an idea of the data




  - Let's now estimate $p(y = 3)$

$$p(y=3) = \frac{1}{Nh^D}\sum_{n=1}^{N} K\left(\frac{x - x^{(n}}{h}\right) = \frac{1}{10 \times 4^1}\left[K\left(\frac{3-4}{4}\right) + K\left(\frac{3-5}{4}\right) + \cdots K\left(\frac{3-17}{4}\right)\right] = 0.0025$$

  - Similarly

$$p(y=10) = \frac{1}{10\times 4^1}[0+0+0+0+0+0+0+0+0+0] = 0$$

$$p(y=15) = \frac{1}{10\times 4^1}[0+0+0+0+0+1+1+1+1+0] = 0.1$$

# *k-Nearest Neighbors*

- Recall the generic expression for density estimation

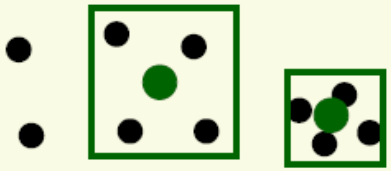
$$p(x) \approx \frac{k/n}{V}$$

- In Parzen windows estimation, we fix $V$ and that determines $k$, the number of points inside $V$

- In k-nearest neighbor approach we fix $k$, and find $V$ that contains $k$ points inside

# *k*-Nearest Neighbors

- kNN approach seems a good solution for the problem of the "best" window size
  - Let the cell volume be a function of the training data
  - Center a cell about x and let it grows until it captures ***k*** samples
  - ***k*** are called the ***k*** nearest-neighbors of ***x***

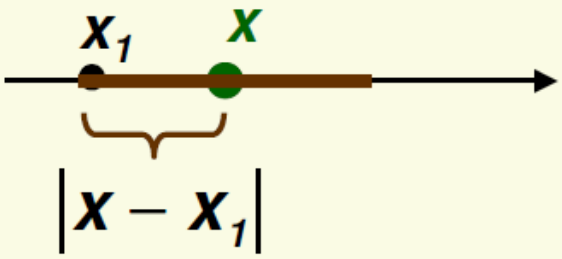


- 2 possibilities can occur:
  - Density is high near *x*; therefore the cell will be small which provides a good resolution
  - Density is low; therefore the cell will grow large and stop until higher density regions are reached

# k-Nearest Neighbor

- Of course, now we have a new question
  - How to choose $k$?
- A good "rule of thumb" is $k = \sqrt{n}$
  - Can prove convergence if **n** goes to infinity
  - Not too useful in practice, however
- Let's look at 1-D example
  - we have one sample, i.e. $n = 1$

$$p(x) \approx \frac{k/n}{V} = \frac{1}{2|x - x_1|}$$

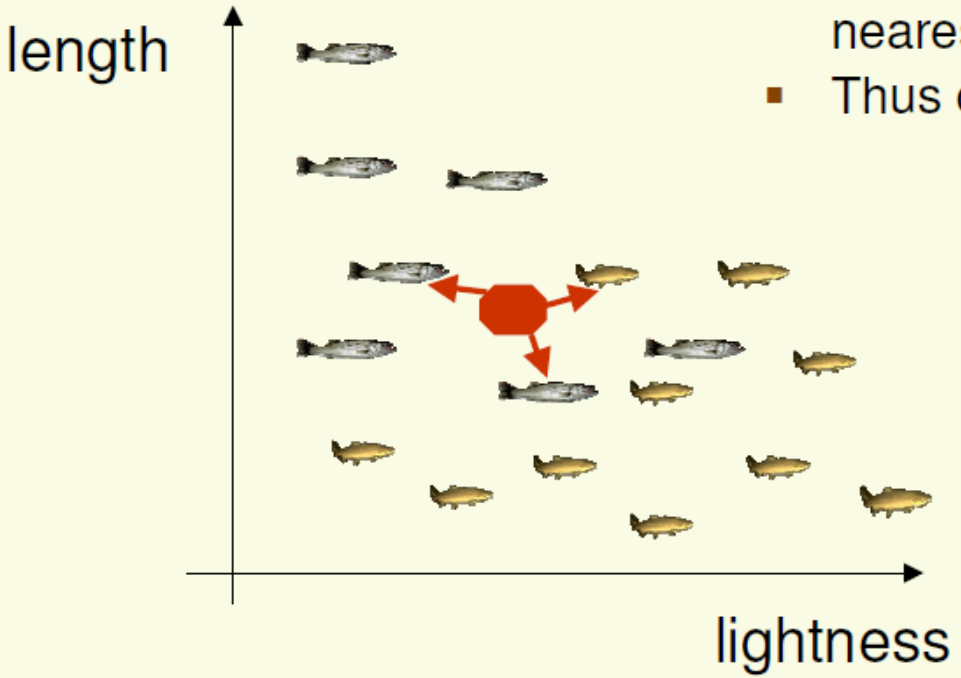



- But the estimated $p(x)$ is not even close to a density function:

$$\int_{-\infty}^{\infty} \frac{1}{2|x - x_1|} dx = \infty \neq 1$$

# *k*-Nearest Neighbor: Example

- Back to fish sorting
  - Suppose we have 2 features, and collected sample points as in the picture
  - Let ***k*** = 3




- 2 sea bass, 1 salmon are the 3 nearest neighbors
- Thus classify as sea bass

**Example**

We have data from the questionnaires survey (to ask people opinion) and objective testing with two attributes (acid durability and strength) to classify whether a special paper tissue is good or not. Here is four training samples

| X1 = Acid Durability (seconds) | X2 = Strength (kg/square meter) | Y = Classification |
|---|---|---|
| 7 | 7 | Bad |
| 7 | 4 | Bad |
| 3 | 4 | Good |
| 1 | 4 | Good |

Now the factory produces a new paper tissue that pass laboratory test with X1 = 3 and X2 = 7. Without another expensive survey, can we guess what the classification of this new tissue is?

*1. Determine parameter K = number of nearest neighbors*

# Exercise

**Table 5.2**

The SPEED and AGILITY ratings for 20 college athletes and whether they were drafted by a professional team.

| ID | SPEED | AGILITY | DRAFT | ID | SPEED | AGILITY | DRAFT |
|---|---|---|---|---|---|---|---|
| 1 | 2.50 | 6.00 | no | 11 | 2.00 | 2.00 | no |
| 2 | 3.75 | 8.00 | no | 12 | 5.00 | 2.50 | no |
| 3 | 2.25 | 5.50 | no | 13 | 8.25 | 8.50 | no |
| 4 | 3.25 | 8.25 | no | 14 | 5.75 | 8.75 | yes |
| 5 | 2.75 | 7.50 | no | 15 | 4.75 | 6.25 | yes |
| 6 | 4.50 | 5.00 | no | 16 | 5.50 | 6.75 | yes |
| 7 | 3.50 | 5.25 | no | 17 | 5.25 | 9.50 | yes |
| 8 | 3.00 | 3.25 | no | 18 | 7.00 | 4.25 | yes |
| 9 | 4.00 | 4.00 | no | 19 | 7.50 | 8.00 | yes |
| 10 | 4.25 | 3.75 | no | 20 | 7.25 | 5.75 | yes |

- Consider K= 4, Determine Draft

# *kNN: How Well Does it Work?*

- kNN rule is certainly simple and intuitive, but does it work?
- Pretend that we can get an unlimited number of samples
- By definition, the best possible error rate is the Bayes rate ***E****
- Even for ***k*** =1, the nearest-neighbor rule leads to an error rate greater than ***E****
- But as $n \to \infty$, it can be shown that nearest neighbor rule error rate is smaller than ***2E****
- If we have a lot of samples, the kNN rule will do very well !

# *kNN: How to Choose k?*

- In theory, when the infinite number of samples is available, the larger the ***k***, the better is classification (error rate gets closer to the optimal Bayes error rate)
- But the caveat is that all ***k*** neighbors have to be close to ***x***
  - Possible when infinite # samples available
  - Impossible in practice since # samples is finite

# *kNN Summary*

- Advantages
  - Can be applied to the data from any distribution
  - Very simple and intuitive
  - Good classification if the number of samples is large enough
- Disadvantages
  - Choosing best ***k*** may be difficult
  - Computationally heavy, but improvements possible
  - Need large number of samples for accuracy
    - Can never fix this without assuming parametric distribution

Thank You !!!