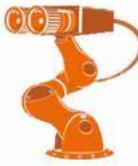


# Computer vision

by Ahmed Ibrahim

# Mean Shift Segmentation



# Mean Shift Segmentation

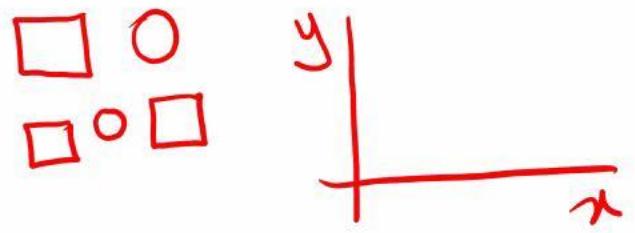


- Mean-Shift Algorithm is a feature-space technique that locates the maxima of a feature-density-function.
- **Question 1:** What is a **Feature**?

# Mean Shift Segmentation



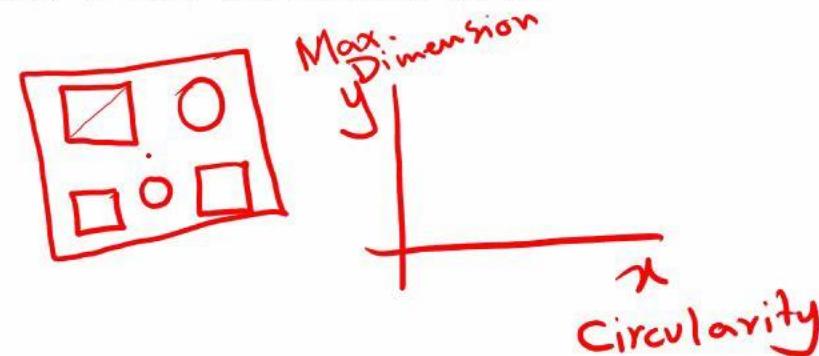
- Mean-Shift Algorithm is a feature-space technique that locates the maxima of a feature-density-function.
- **Question 1:** What is a **Feature**?
- **Question 2:** What is **Feature-Space**?



# Mean Shift Segmentation



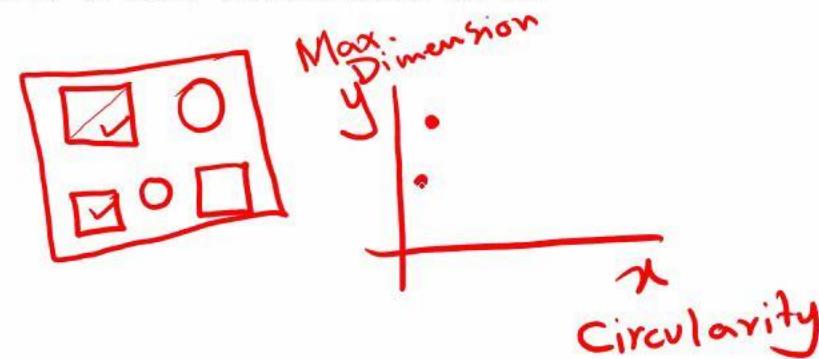
- Mean-Shift Algorithm is a feature-space technique that locates the maxima of a feature-density-function.
- **Question 1:** What is a **Feature**?
- **Question 2:** What is **Feature-Space**?



# Mean Shift Segmentation



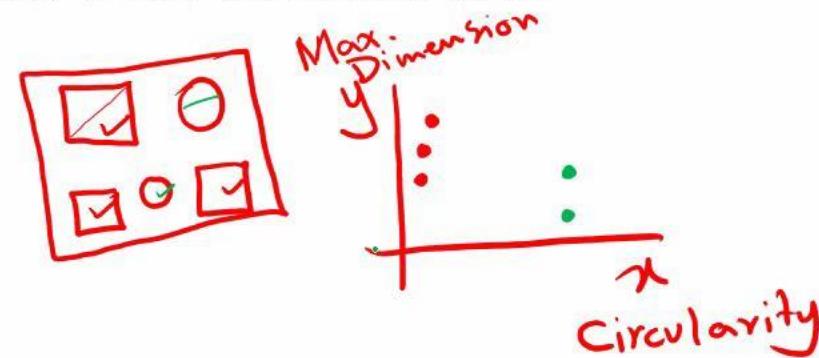
- Mean-Shift Algorithm is a feature-space technique that locates the maxima of a feature-density-function.
- **Question 1:** What is a **Feature**?
- **Question 2:** What is **Feature-Space**?



# Mean Shift Segmentation



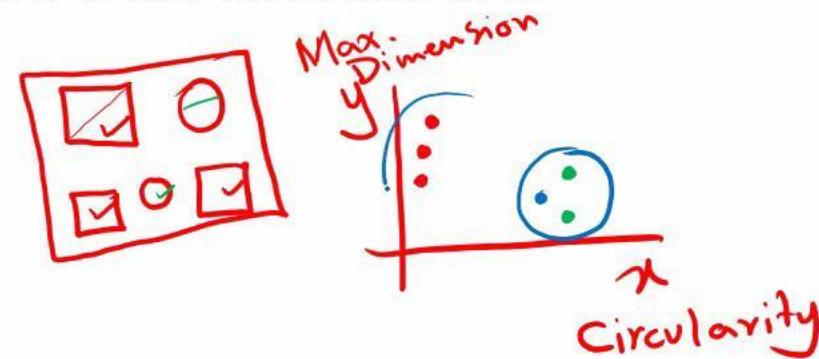
- Mean-Shift Algorithm is a feature-space technique that locates the maxima of a feature-density-function.
- **Question 1:** What is a **Feature**?
- **Question 2:** What is **Feature-Space**?



# Mean Shift Segmentation



- Mean-Shift Algorithm is a feature-space technique that locates the maxima of a feature-density-function.
- **Question 1:** What is a **Feature**?
- **Question 2:** What is **Feature-Space**?



# Mean Shift Segmentation

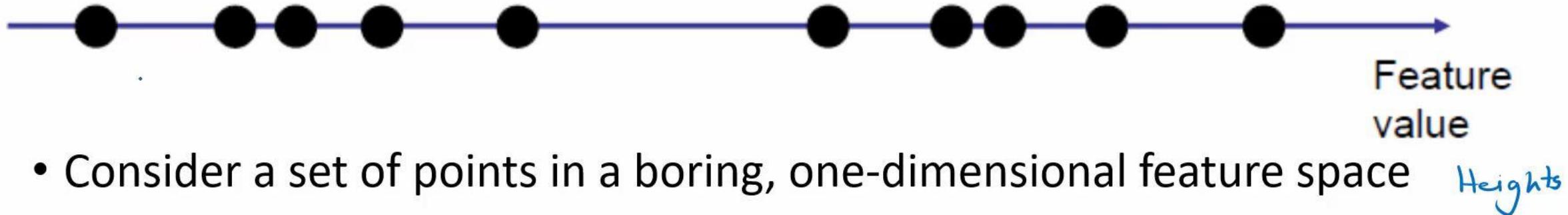


- Consider 1-D Example

# Mean Shift Segmentation



- Consider 1-D Example



# Mean Shift Segmentation

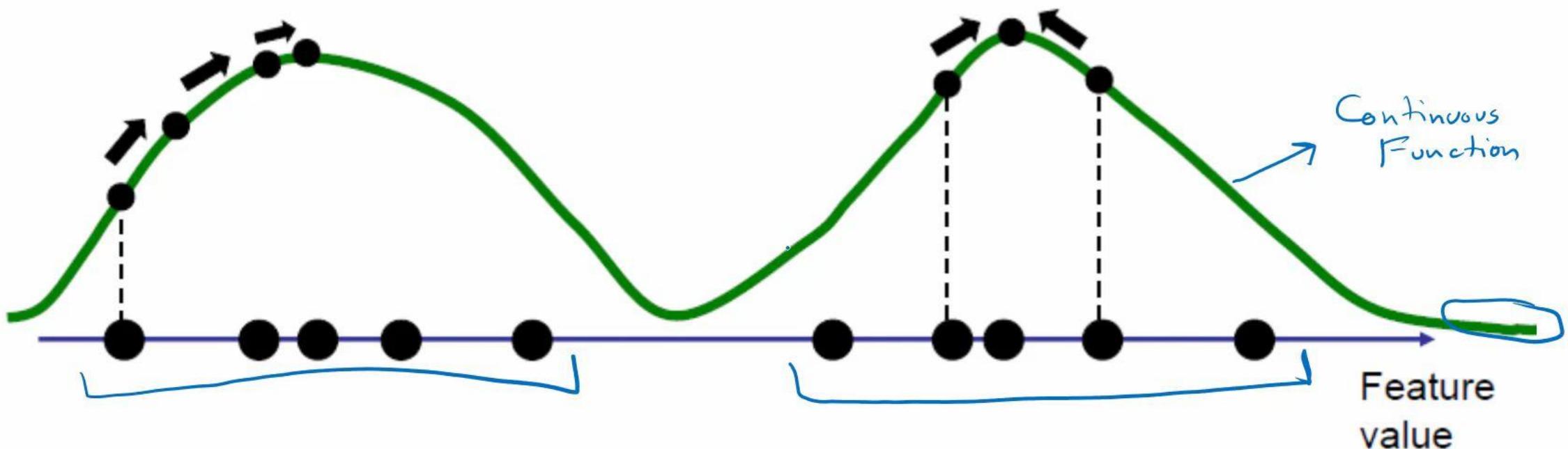


- How can we convert our set of points into a continuous function?

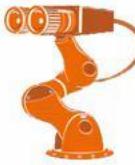
# Mean Shift Segmentation



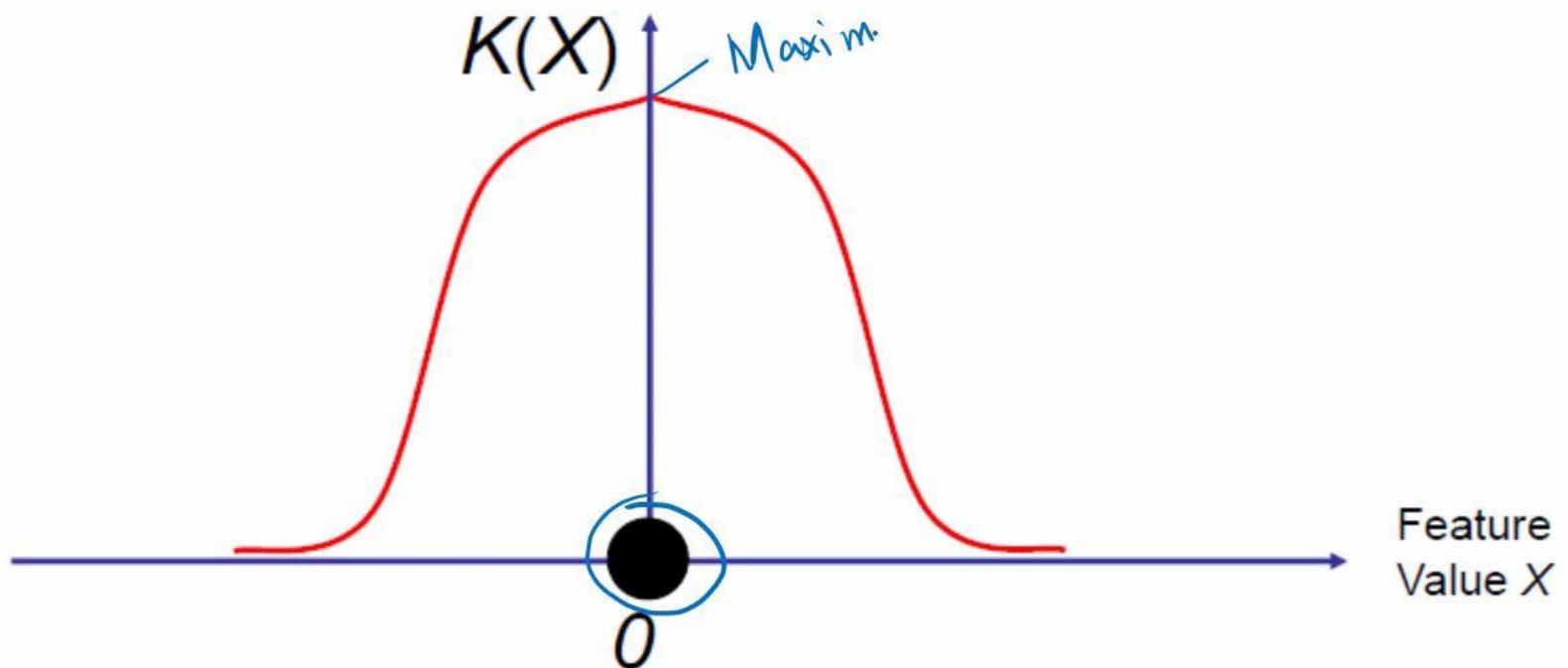
- How can we convert our set of points into a continuous function?



# Mean Shift Segmentation



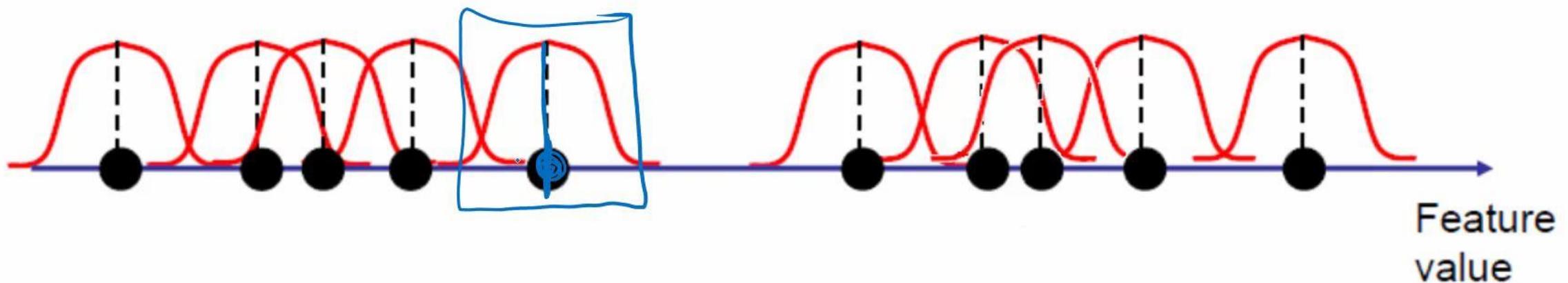
- Consider a kernel function  $K(x)$



# Mean Shift Segmentation



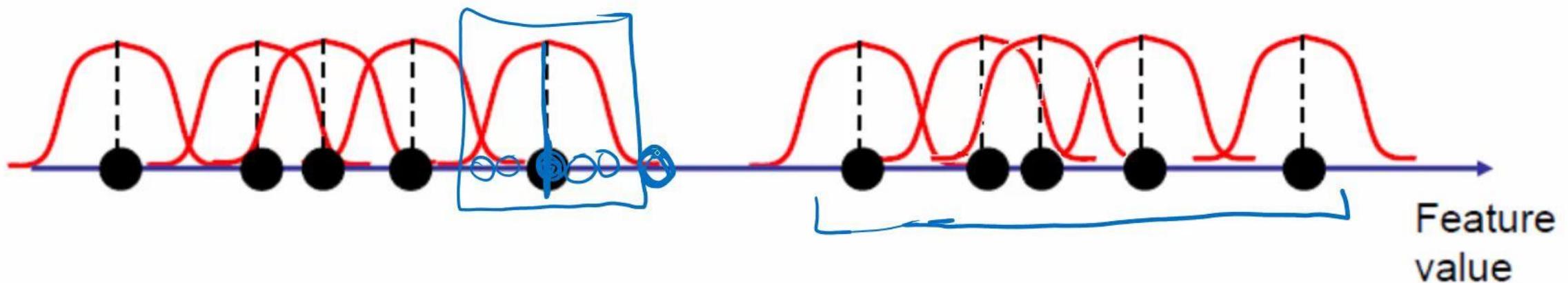
- Application of  $K(x)$  on each data point will yield the following result



# Mean Shift Segmentation



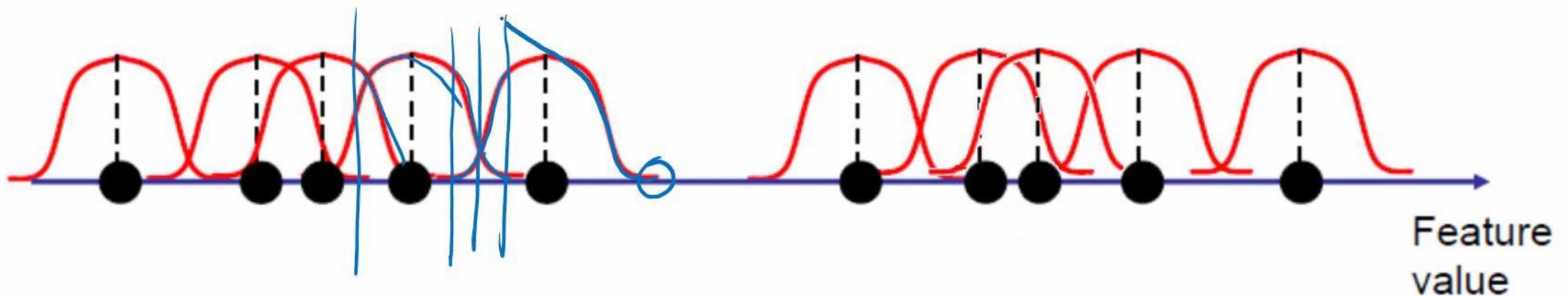
- Application of  $K(x)$  on each data point will yield the following result



# Mean Shift Segmentation



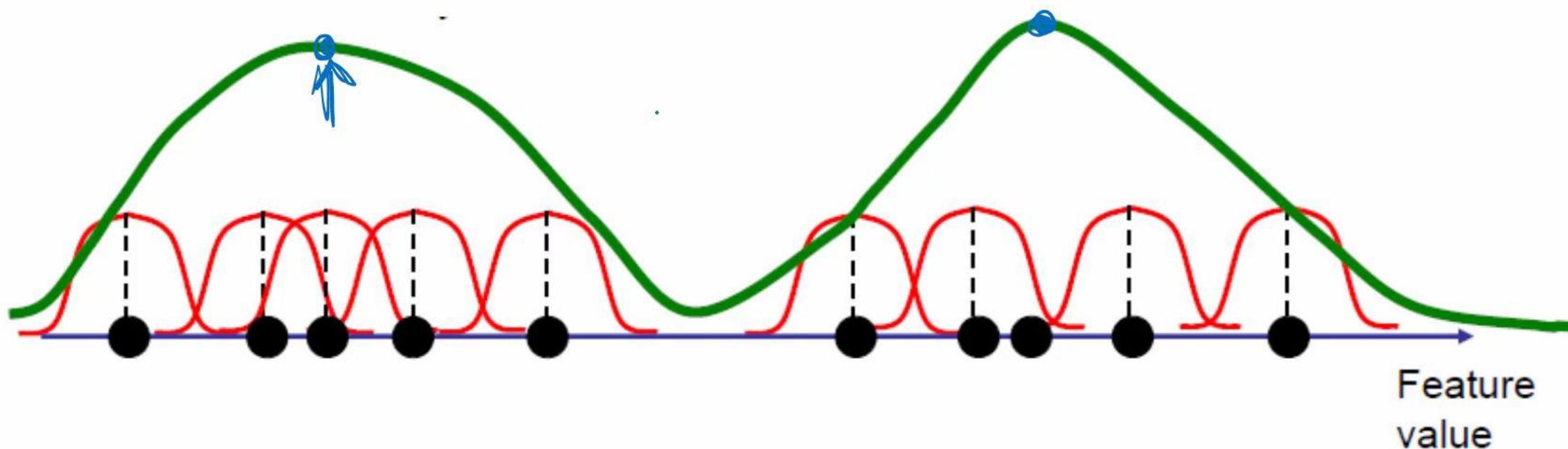
- Application of  $K(x)$  on each data point will yield the following result



# Mean Shift Segmentation



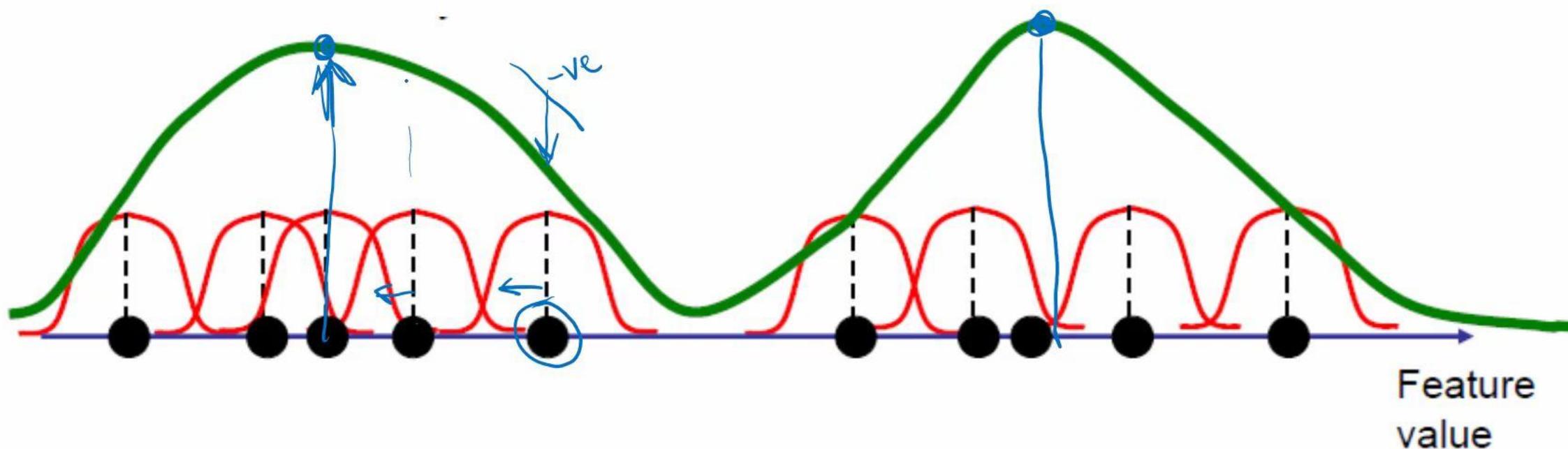
- Green curve  $f(x)$  approximates the probability that feature X is observed given the data points
- The maxima of  $f$  (the modes of the pdf) correspond to the clusters in the data



# Mean Shift Segmentation



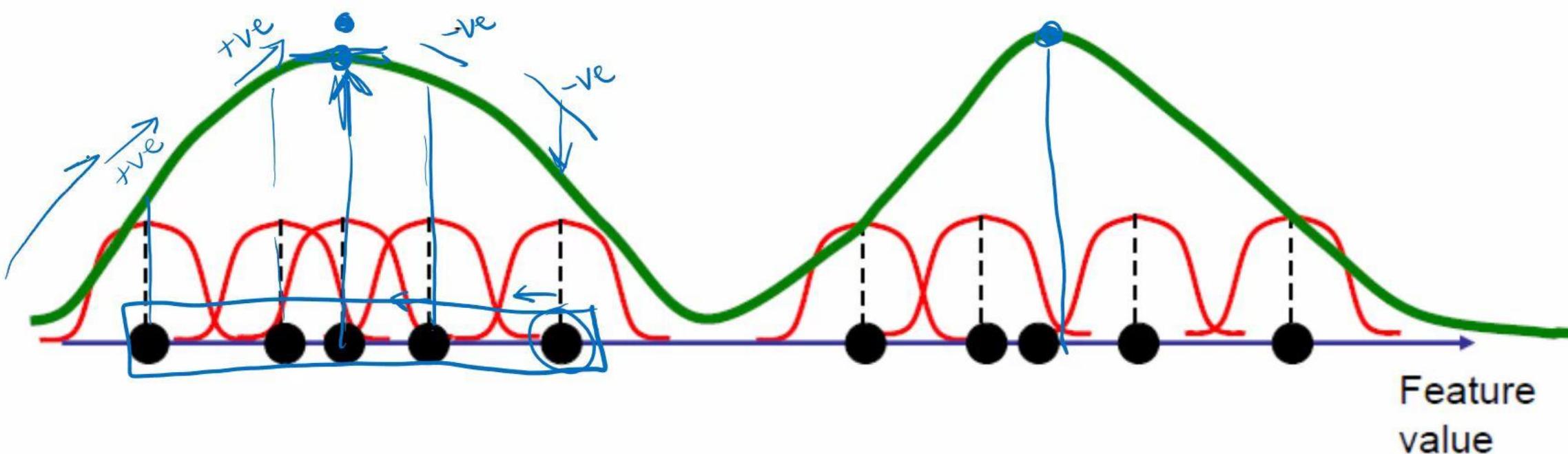
- Green curve  $f(x)$  approximates the probability that feature X is observed given the data points
- The maxima of  $f$  (the modes of the pdf) correspond to the clusters in the data



# Mean Shift Segmentation



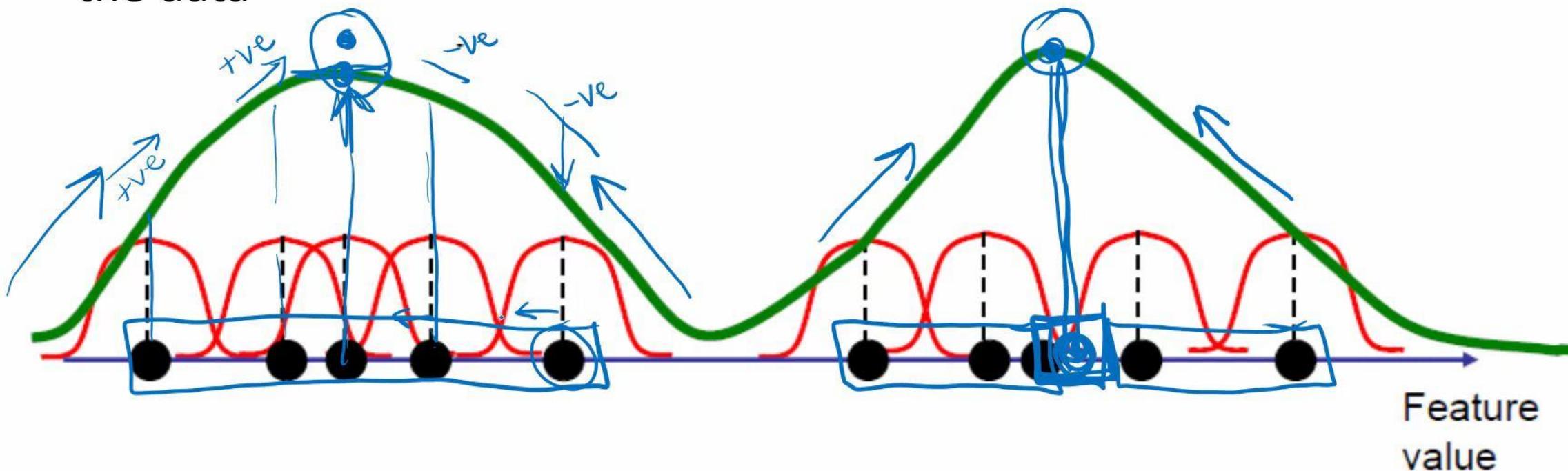
- Green curve  $f(x)$  approximates the probability that feature X is observed given the data points
- The maxima of  $f$  (the modes of the pdf) correspond to the clusters in the data



# Mean Shift Segmentation



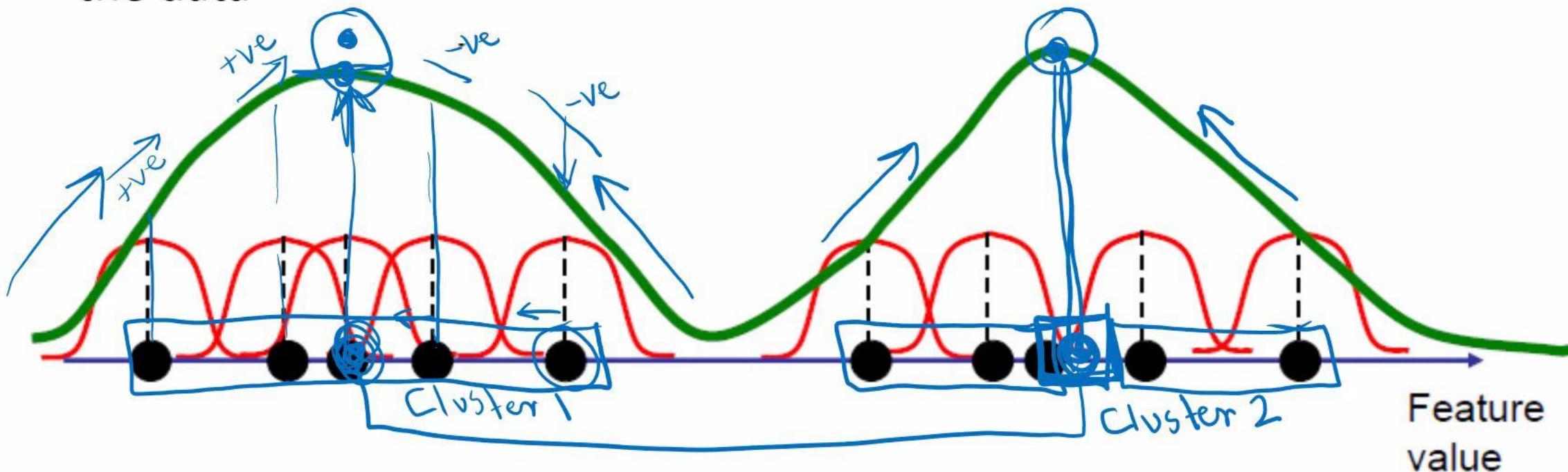
- Green curve  $f(x)$  approximates the probability that feature X is observed given the data points
- The maxima of  $f$  (the modes of the pdf) correspond to the clusters in the data



# Mean Shift Segmentation



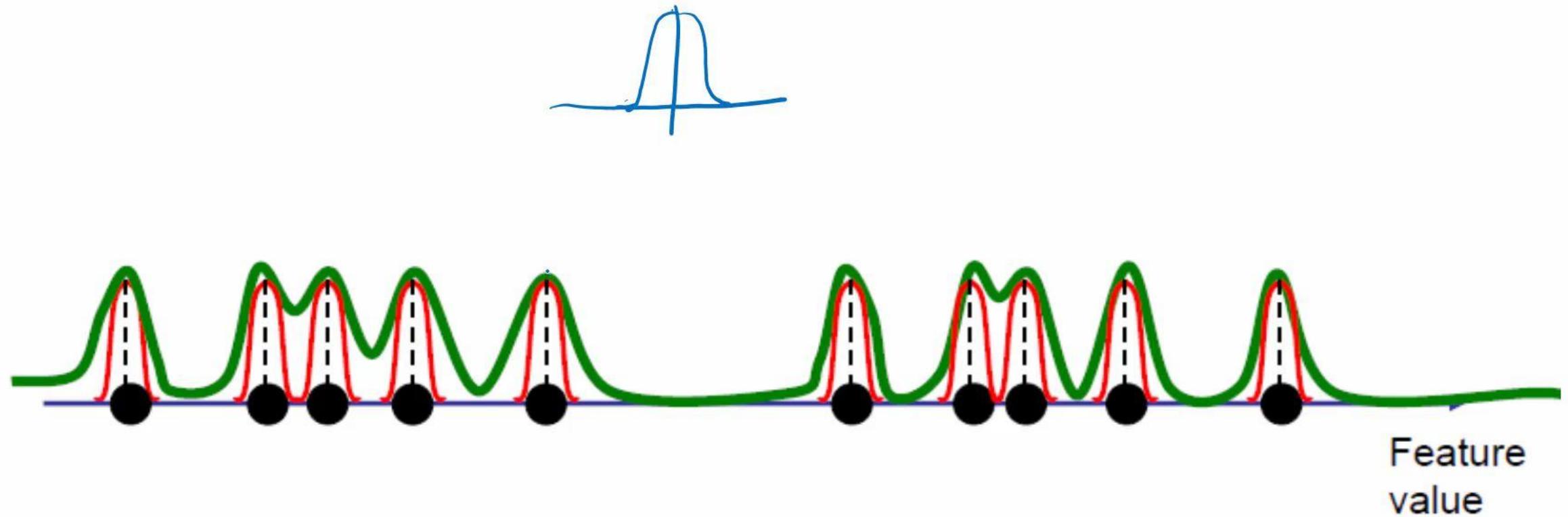
- Green curve  $f(x)$  approximates the probability that feature X is observed given the data points
- The maxima of  $f$  (the modes of the pdf) correspond to the clusters in the data



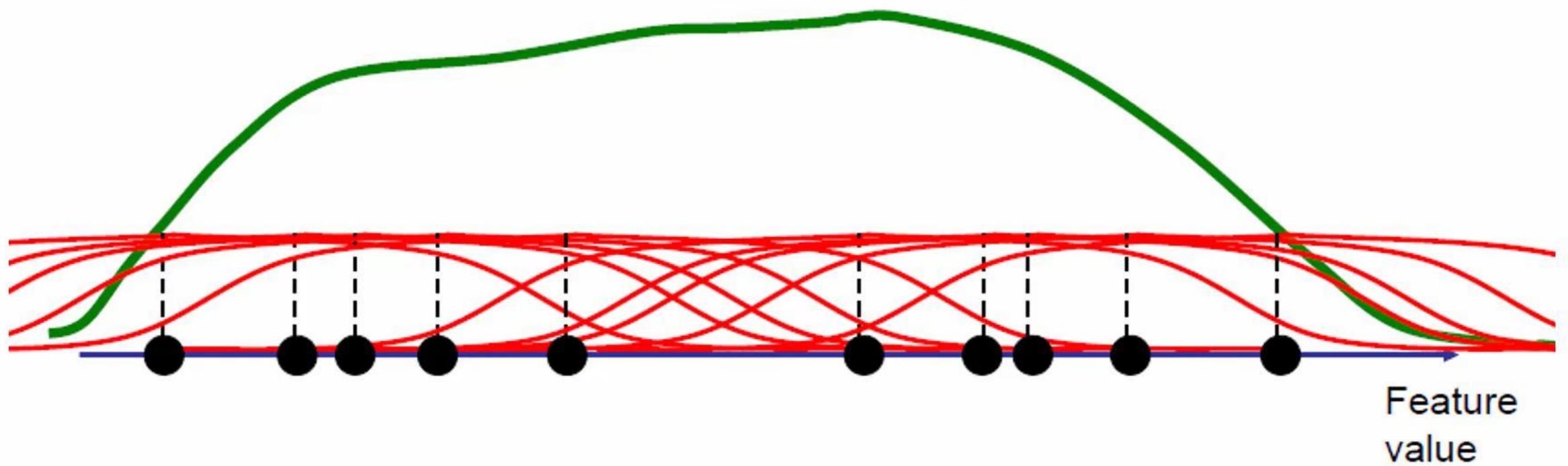
# Mean Shift Segmentation



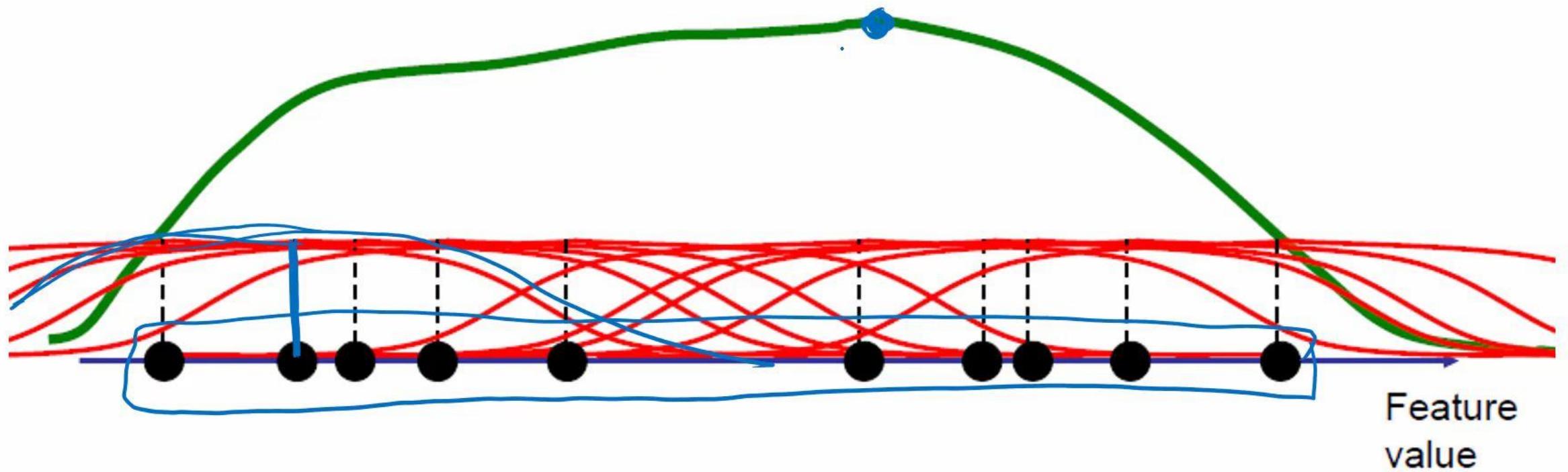
- This is what will happen if we'll choose a **small bandwidth** of the kernel.



# Mean Shift Segmentation



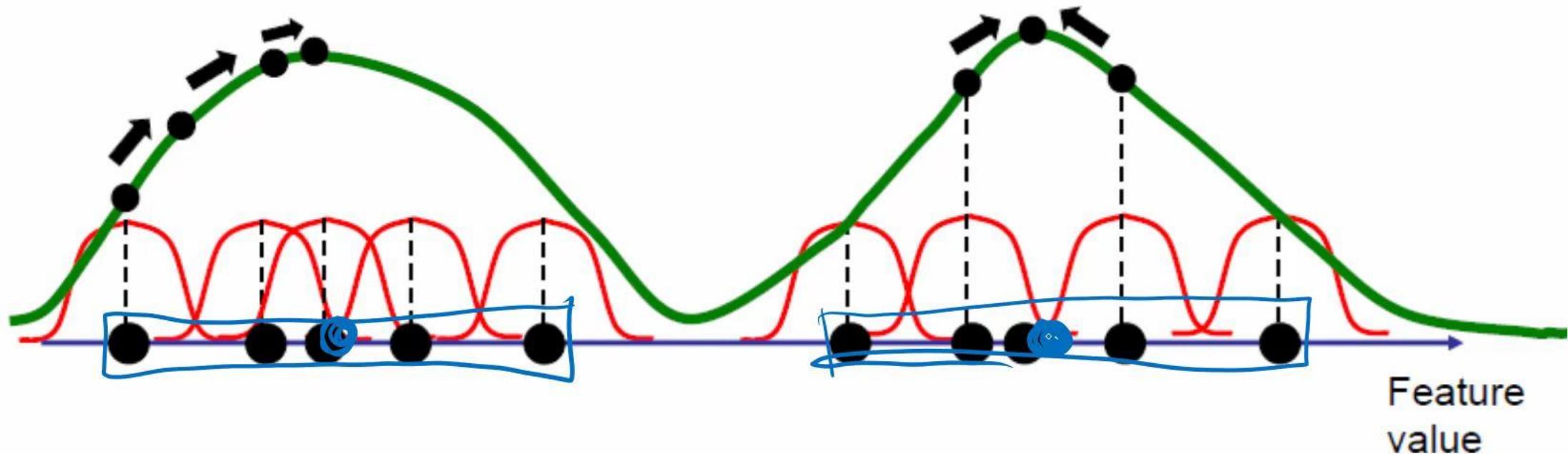
# Mean Shift Segmentation



# Mean Shift Segmentation



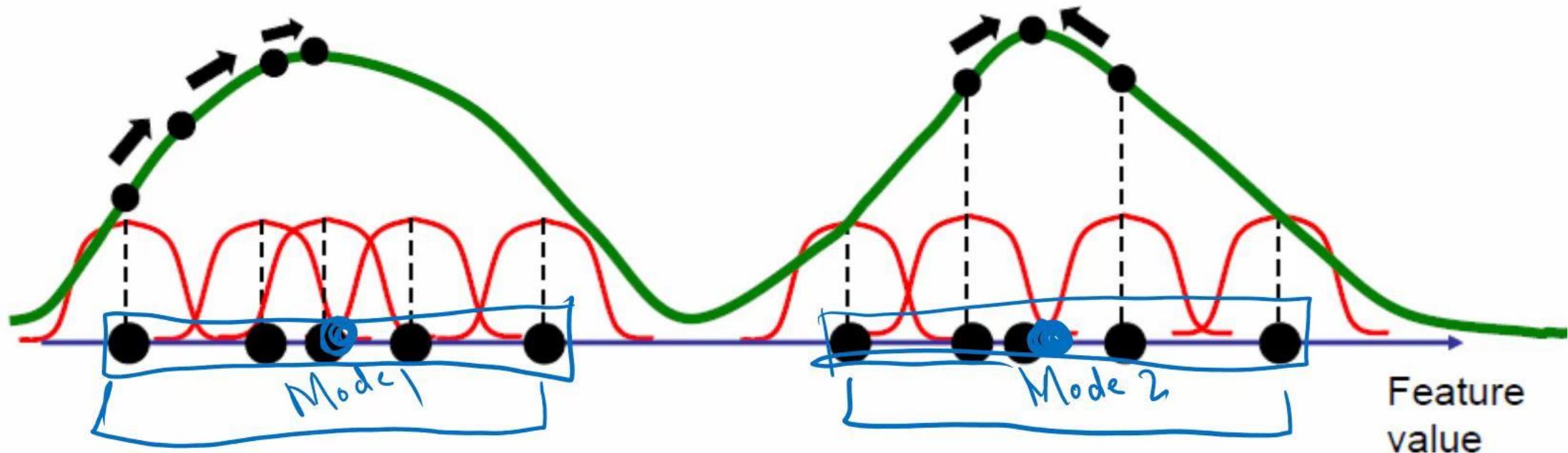
- If we move each point in the direction of the gradient, we will converge to the closest mode



# Mean Shift Segmentation



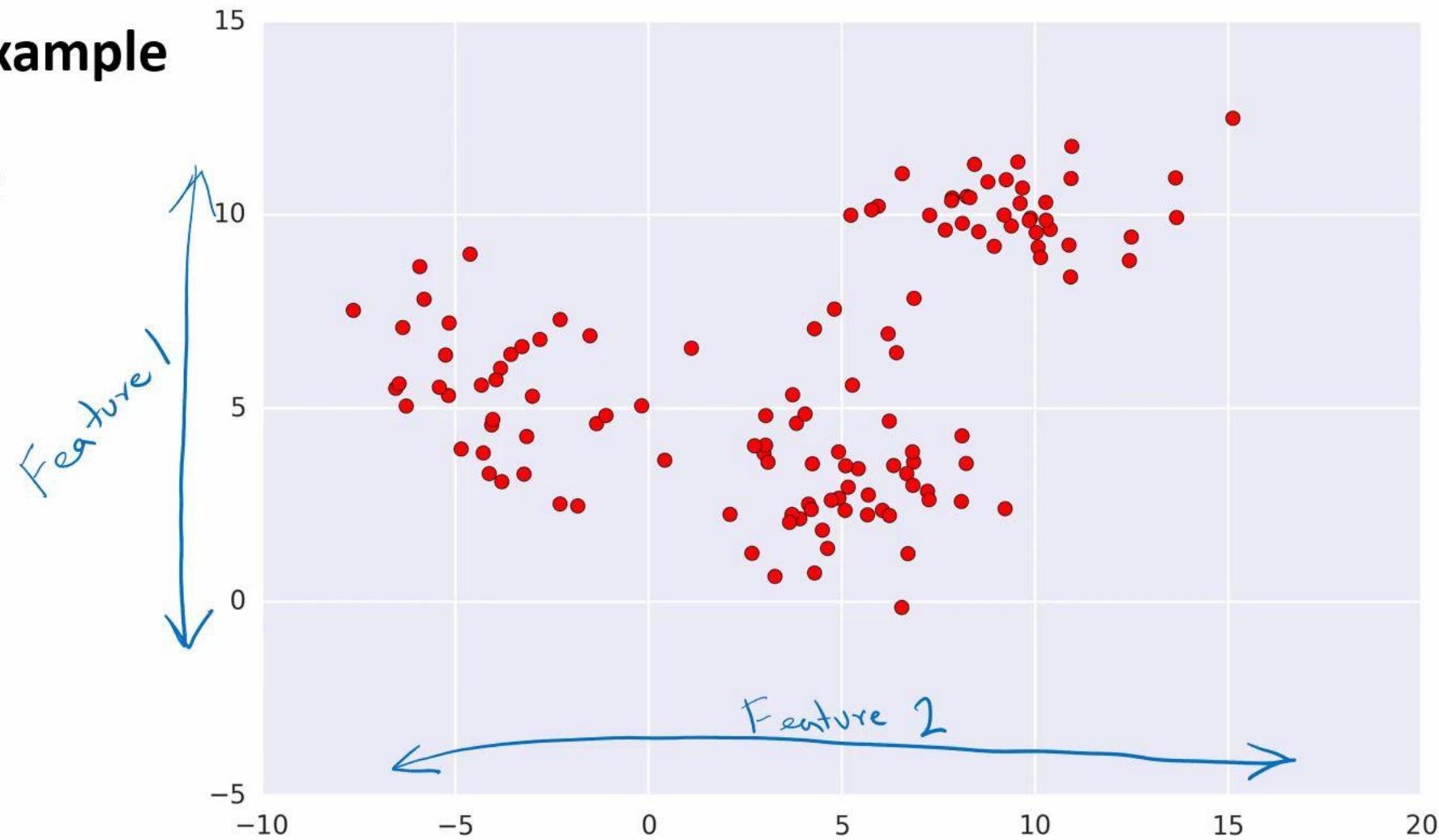
- If we move each point in the direction of the gradient, we will converge to the closest mode



# Mean Shift Segmentation



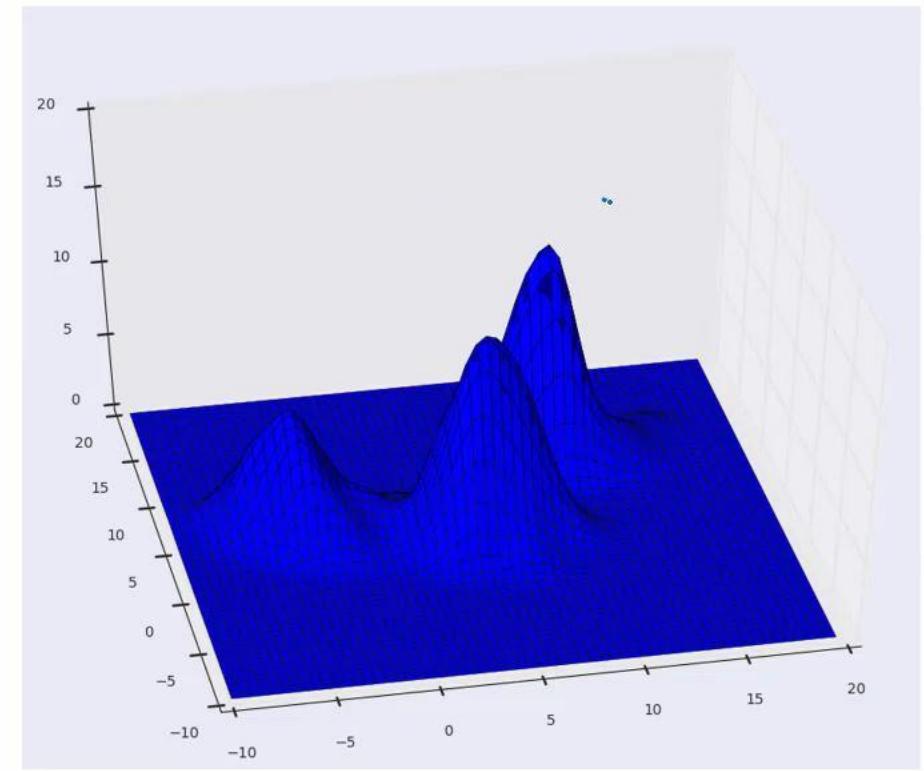
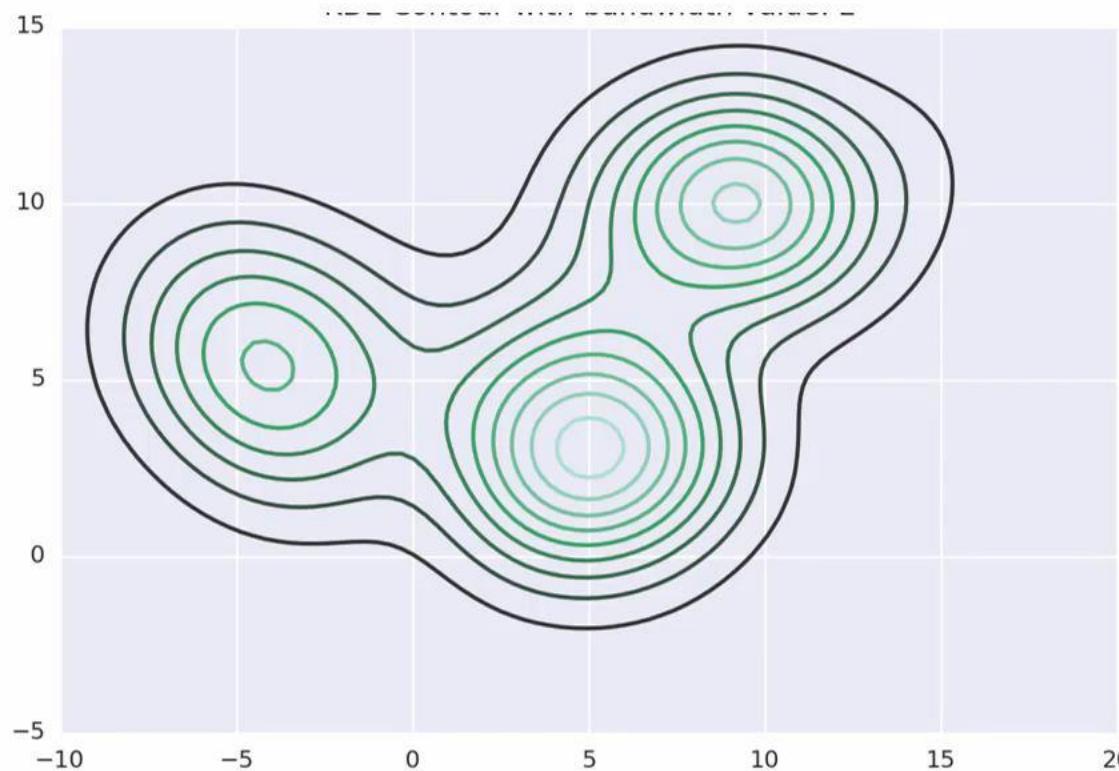
- Consider a **2-D Example** whose Feature-Space is given as:



# Mean Shift Segmentation



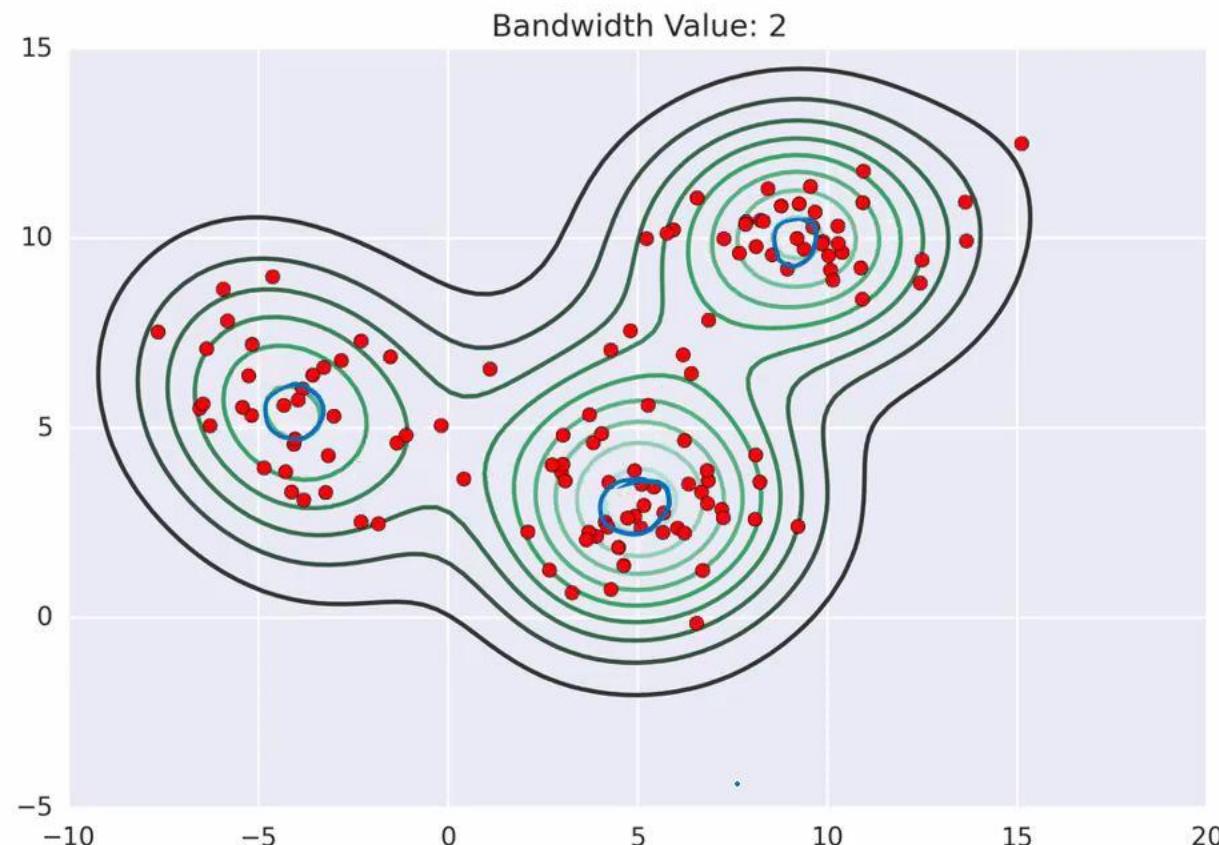
- Using a simple 2-D Gaussian function **Kernel Density Estimation** surface or contour can be generated as:



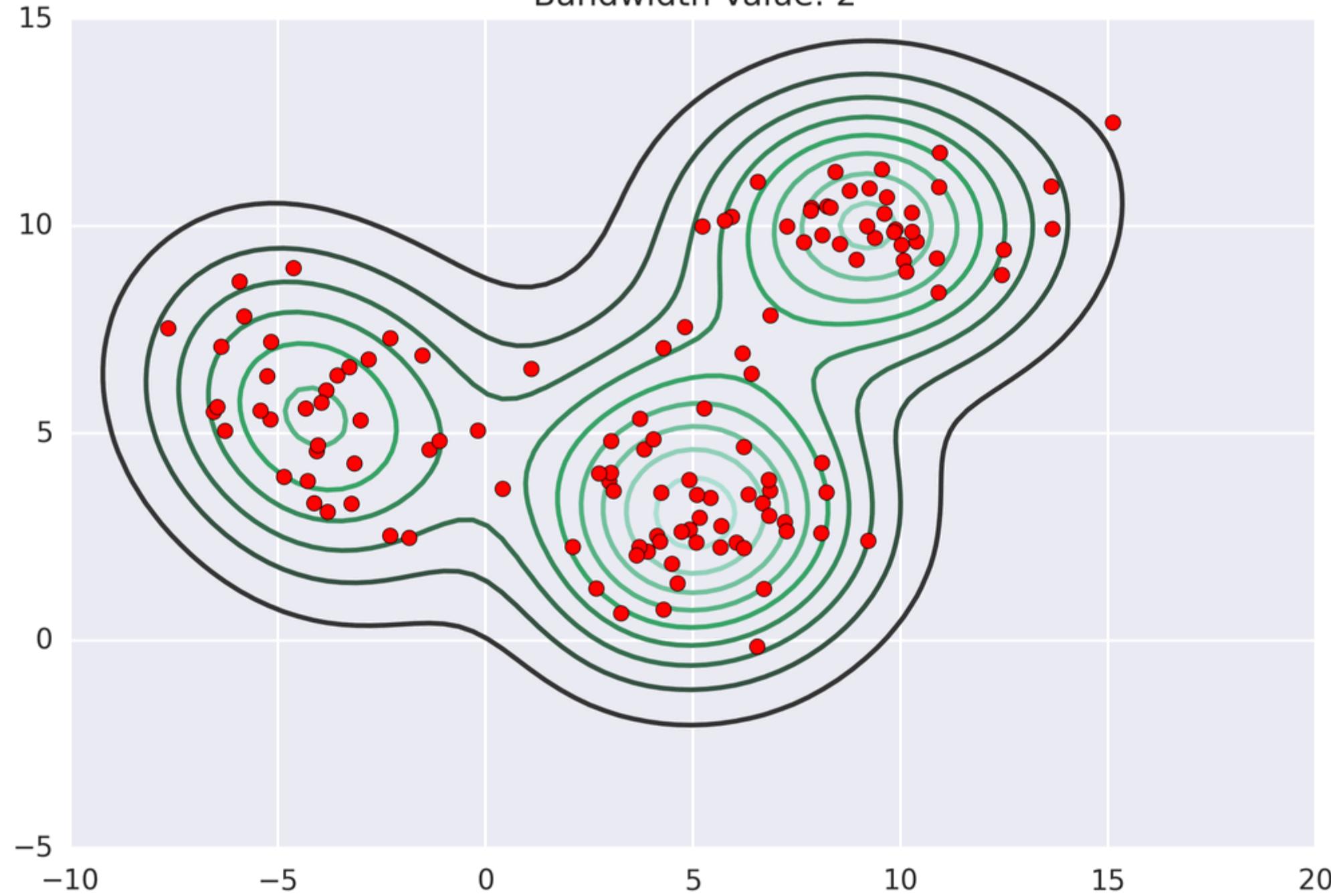
# Mean Shift Segmentation



- Application of Mean-Shifting will look like the following animation



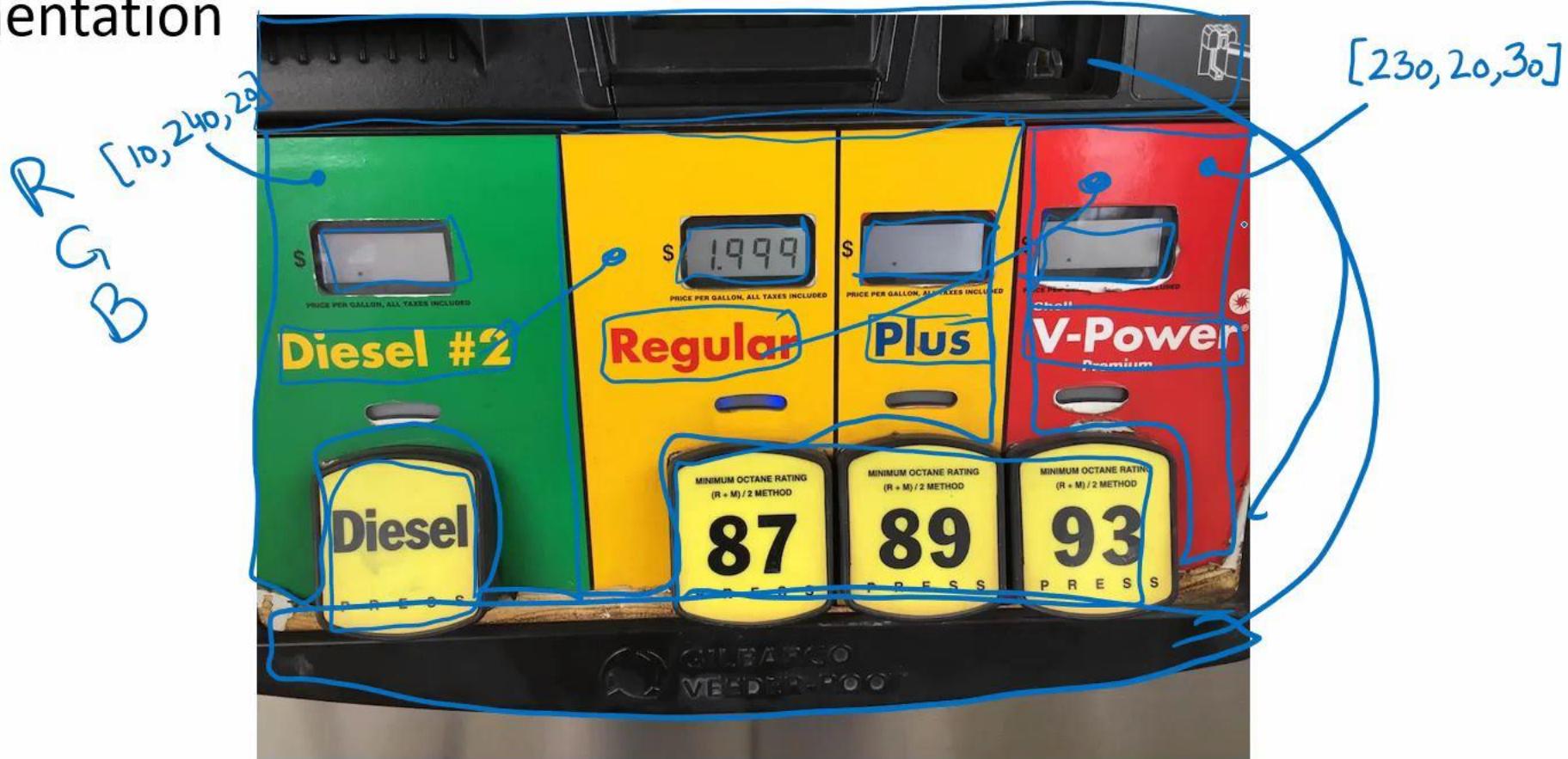
Bandwidth Value: 2



# Real Life Example



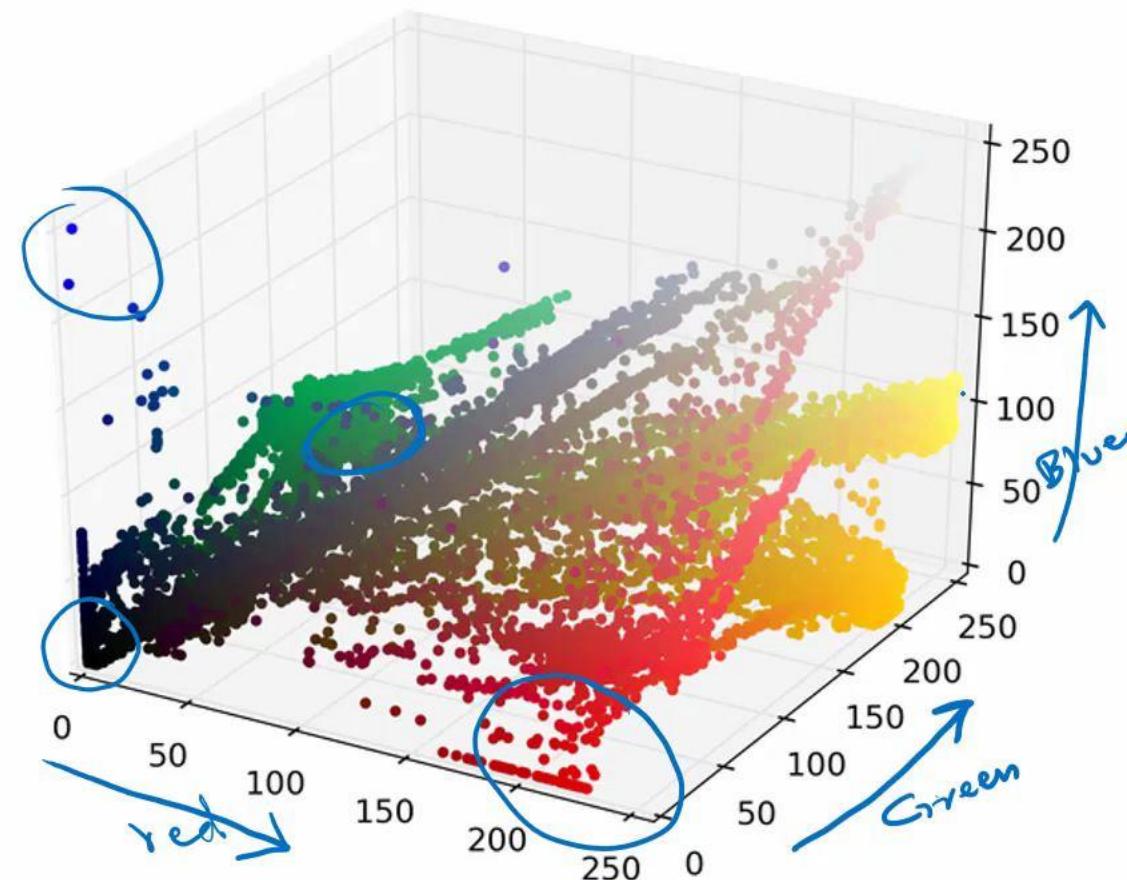
- We want to segment the following image using Mean-Shift Segmentation



# Real Life Example



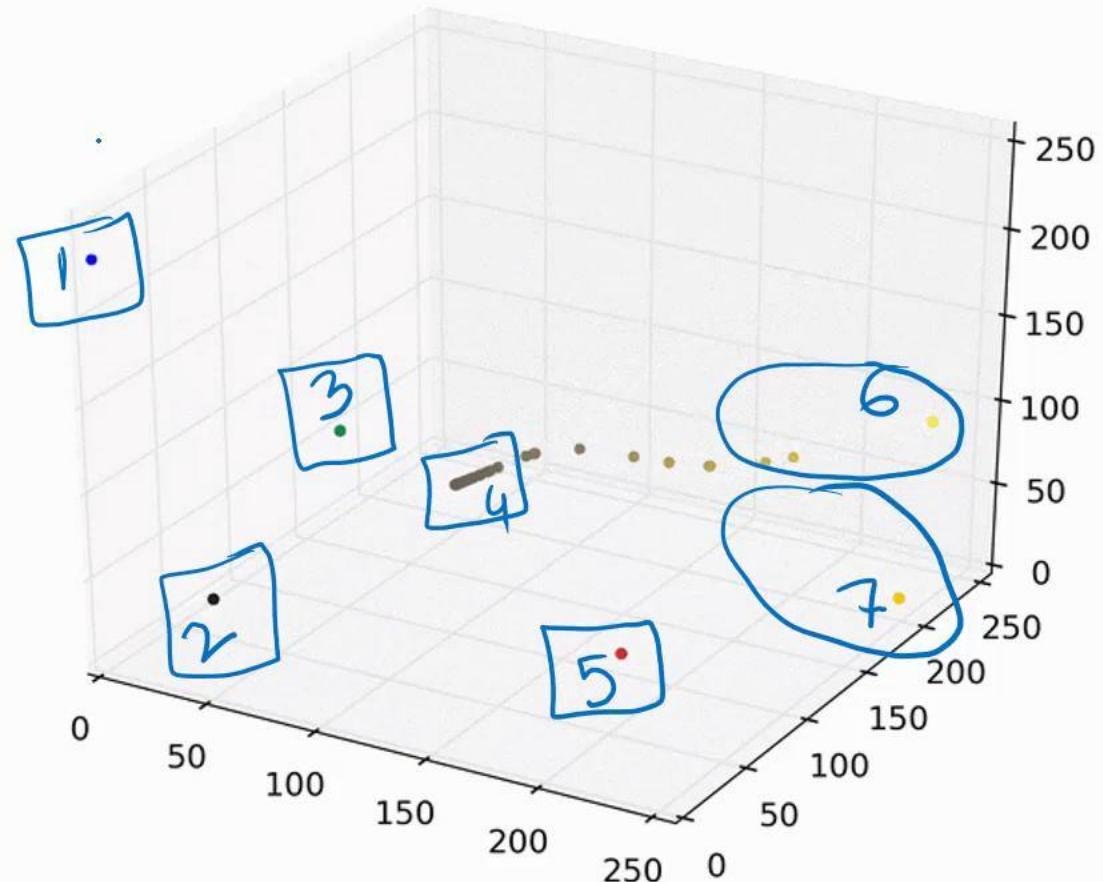
- Feature Space consisting of simple RGB values will look like



# Real Life Example



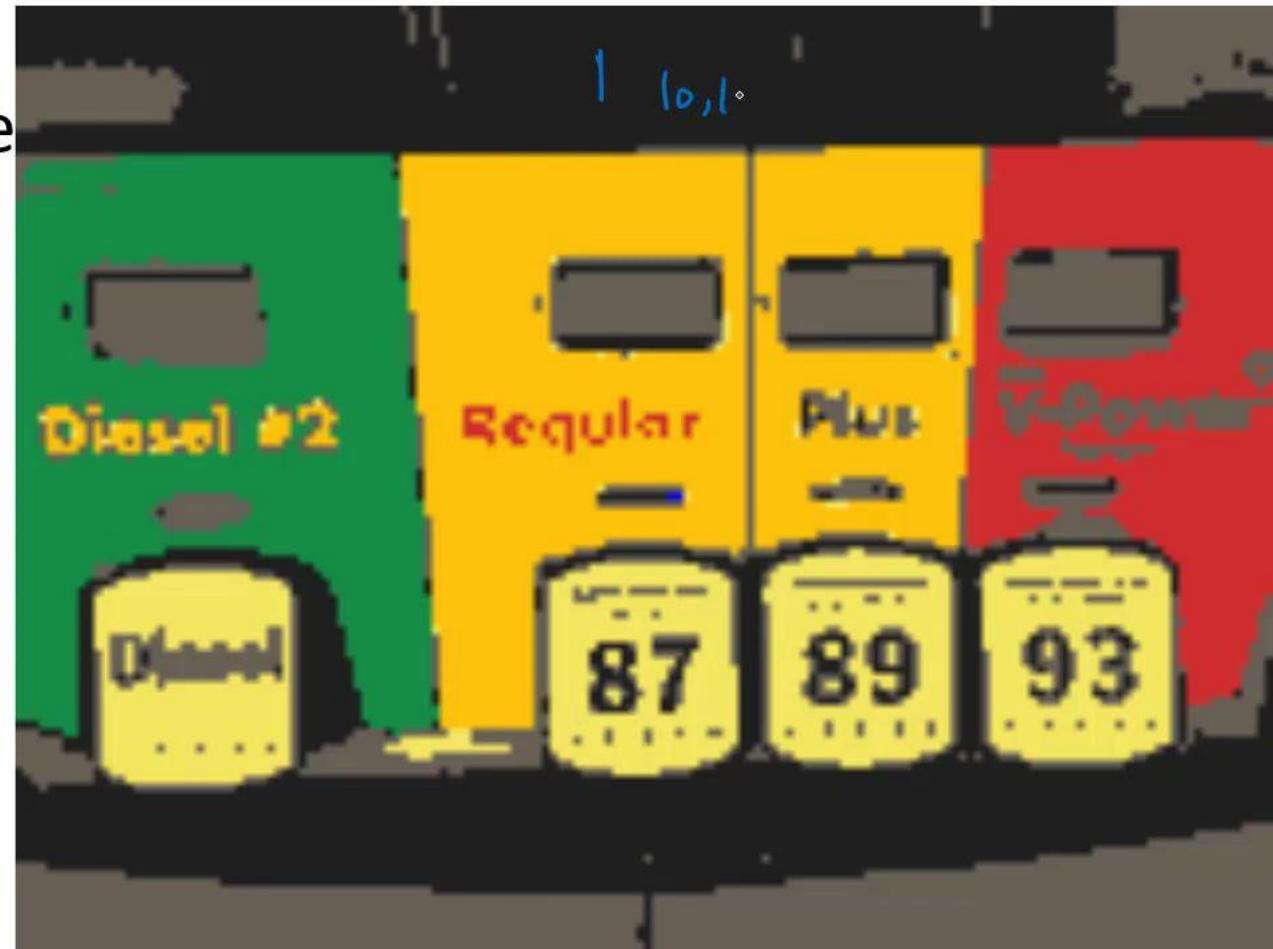
- A **3-D Kernel Density Estimation** surface may be formed and Mean-Shifting can then be applied on it.



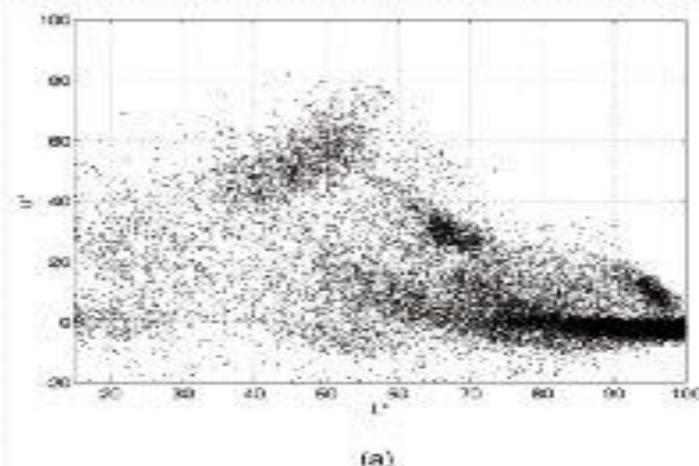
# Real Life Example



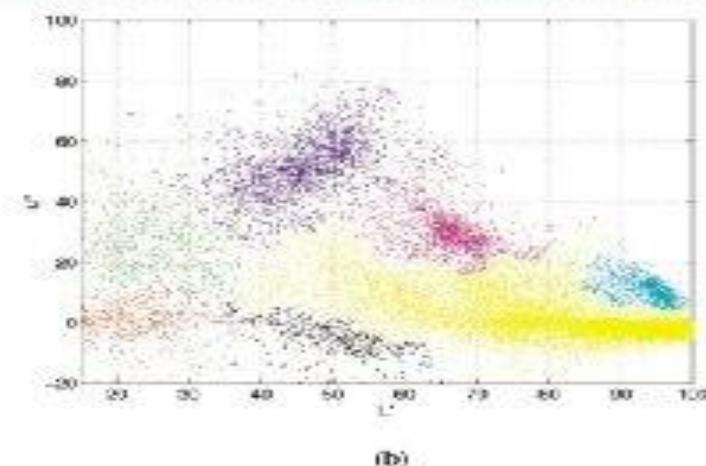
- All data points will converge on one of the peaks.
- Displaying the image using colors of the 7 modes/peaks will look like



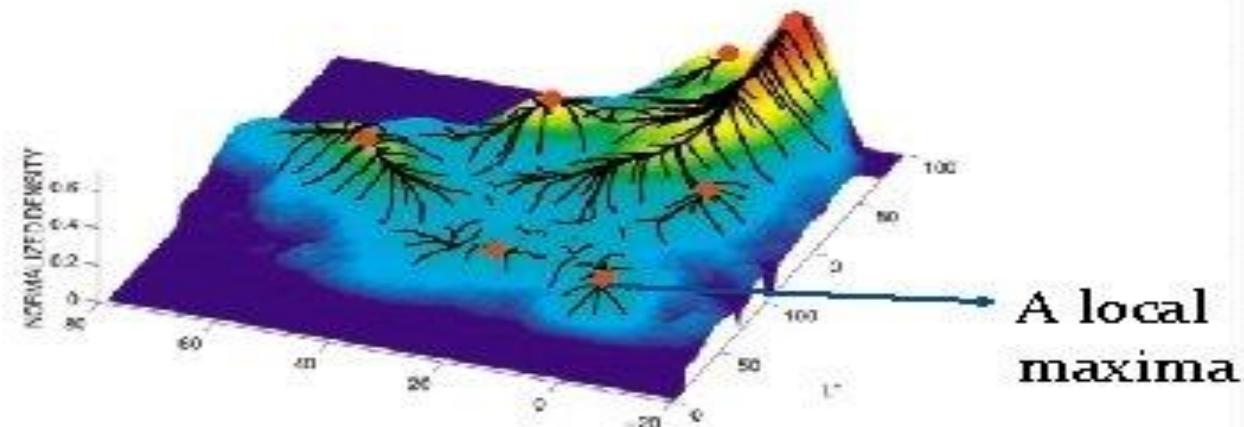
## Examples for Mean shift clustering



(a)



(b)

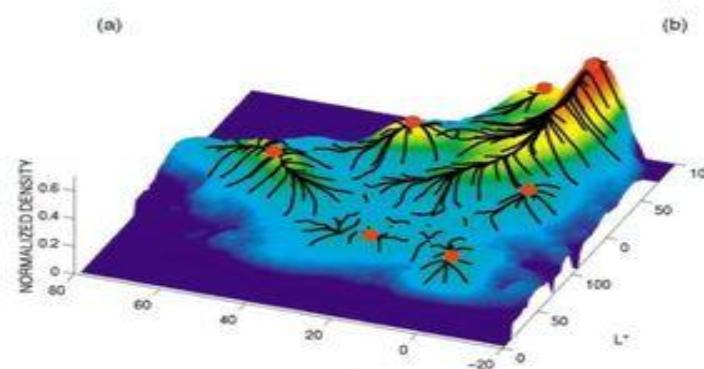
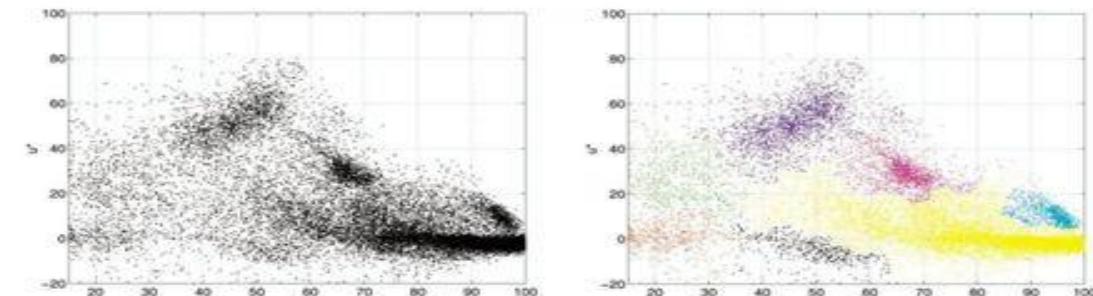
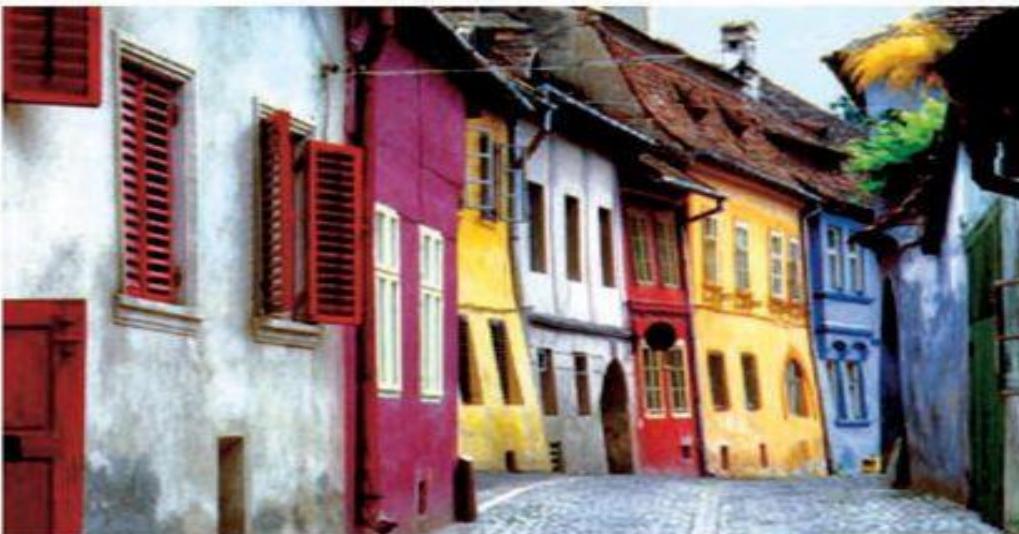


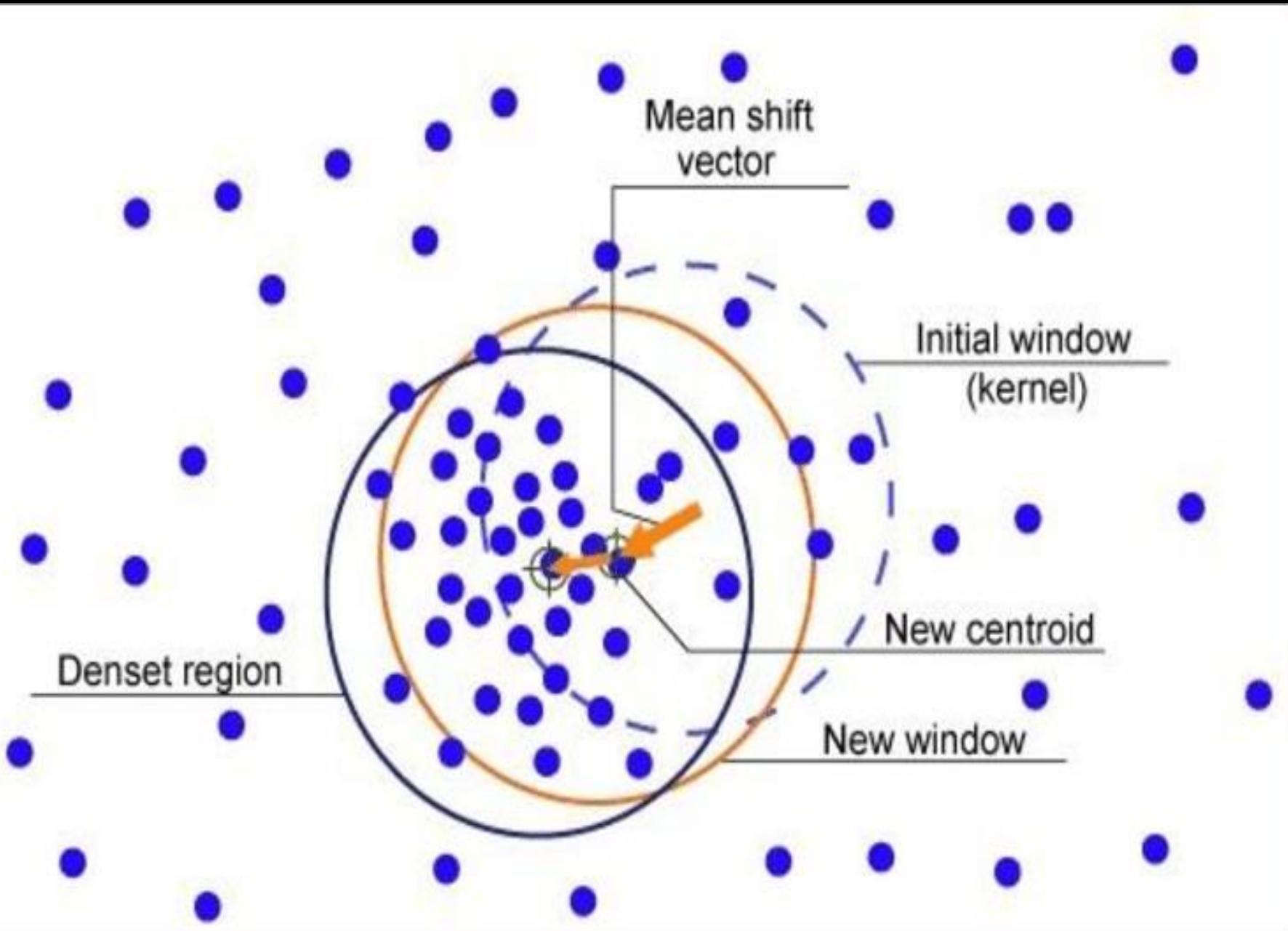
A local maxima

Note how points are moving towards the local maximas (modes) that represents the center of gravity for those data points.  
In this example we have ended up with 7 segments (modes).

# Mean shift clustering/segmentation

- Find features (color, gradients, texture, etc).
- Initialize windows at individual feature points.
- Perform mean shift for each window until convergence.
- Merge windows that end up near the same “peak” or mode.





$$M_h(\mathbf{y}) = \left[ \frac{1}{n_x} \sum_{i=1}^{n_x} \mathbf{x}_{k_i} \right] - \mathbf{y}_0$$

# Other Real Life Example



# Other Real Life Example

