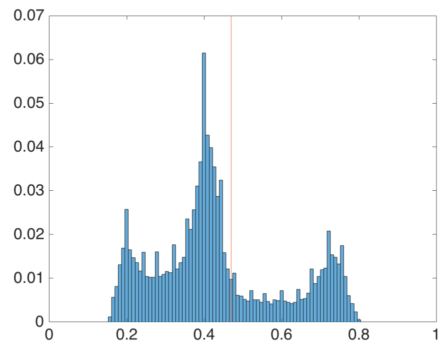
Statistical Image Analysis

Lecture 7: Image features

David Bolin
University of Gothenburg

Gothenburg
April 23, 2018

Intensity-based thresholding



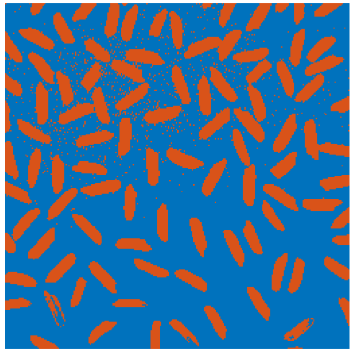
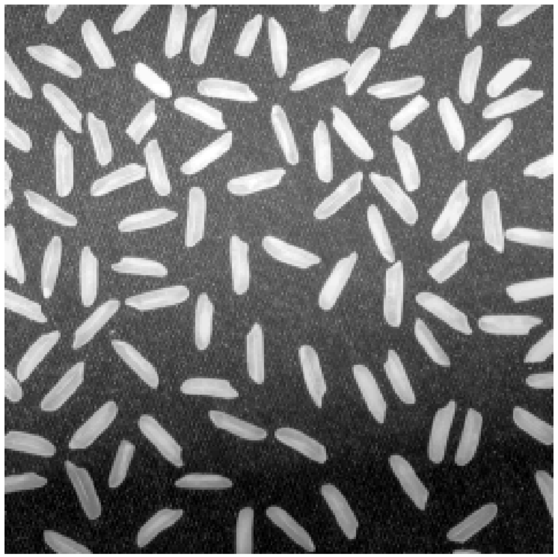


Image classification



Gaussian mixture models

- Hierarchical model for pixel values given classes:

$$\pi(\mathbf{Y}_i|z_i = k) \sim \mathsf{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

$$\pi(z_i) = \begin{cases} \pi_1 & \text{if } z_i = 1 \\ \pi_2 & \text{if } z_i = 2 \\ \vdots & \\ \pi_K & \text{if } z_i = K \\ 0 & \text{otherwise} \end{cases}$$

- Unconditional density:

$$\pi(\mathbf{Y}_i) = \sum_{k=1}^{K} \pi_k \pi_G(\mathbf{Y}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Classification using GMMs

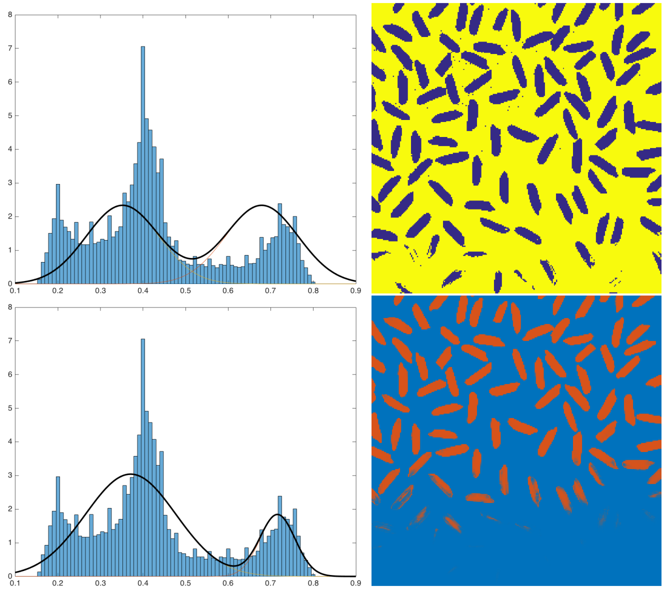
- Posterior class probabilities

$$\mathsf{P}(z_i = k|\mathbf{Y}_i) = \frac{\pi_k \pi_G(\mathbf{Y}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^K \pi_l \pi_G(\mathbf{Y}_i; \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}$$

- Maximum aposteriori-classification:

$$class_i = \arg\max_k \mathsf{P}(z_i = k|\mathbf{Y}_i)$$

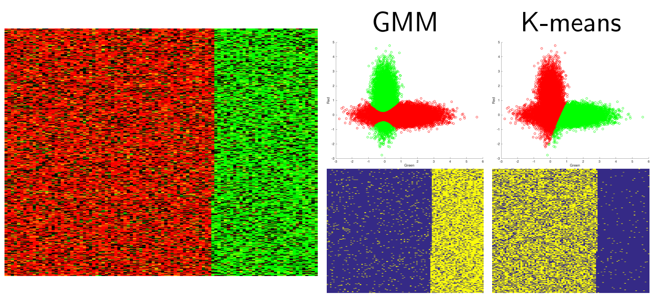
- This is also known as quadratic discriminant analysis. If all $\boldsymbol{\Sigma}_k$ are equal, we get linear discriminant analysis.

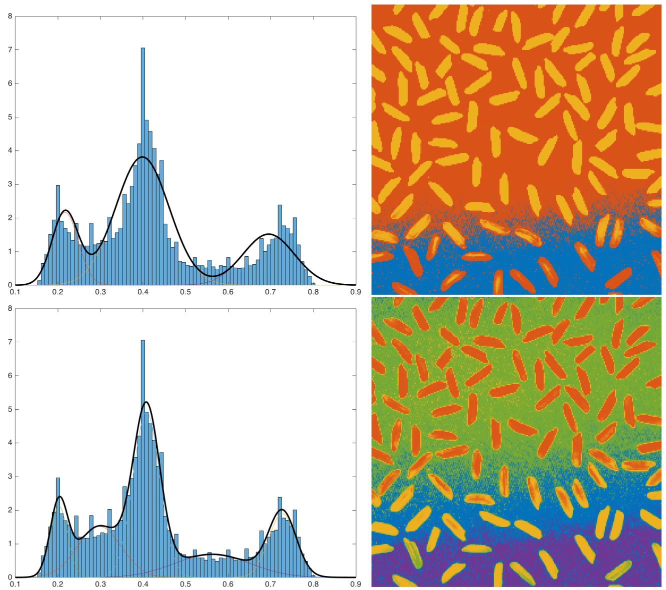


The K-means algorithm

1. Randomly select K observations as cluster centers.
2. Assign each observation to the nearest cluster center.
3. Compute the mean for each cluster and assign these as new cluster centers.
4. Repeat from Step 2.

In the K-means algorithm, we assume $\pi_k = 1/K$ and $\boldsymbol{\Sigma}_k = \sigma^2\mathbf{I}$.





Supervised learning

- We have a set of pixels values $\{\mathbf{Y}_1, \ldots, \mathbf{Y}_n\} \in \mathbb{R}^d$ with known classes $\{z_1, \ldots, z_n\}$.
- Base parameter estimates on these:

$$\hat{\pi}_k = \frac{n_k}{n} \quad \text{where } n_k = \sum_{i=1}^n 1(z_i = k)$$

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i=1}^n 1(z_i = k)\mathbf{Y}_i$$

$$\hat{\Sigma}_k = \frac{1}{n_k - d} \sum_{i=1}^n 1(z_i = k)(\mathbf{Y}_i - \boldsymbol{\mu}_k)(\mathbf{Y}_i - \boldsymbol{\mu}_k)^T$$

Classification using colors

Unsupervised learning

- Let $\boldsymbol{\theta}$ denote all model parameters. The gradient of the likelihood can be written as

$$\nabla \log \pi(\mathbf{Y}; \boldsymbol{\theta}) = \sum_{i=1}^N \sum_{k=1}^K \mathsf{P}(z_i = k | \mathbf{Y}_i, \boldsymbol{\theta})(\nabla \log \pi_G(\mathbf{Y}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \nabla \log \pi_k)$$

- Gradient descent optimization: Choose $\boldsymbol{\theta}^{(0)}$ and iterate

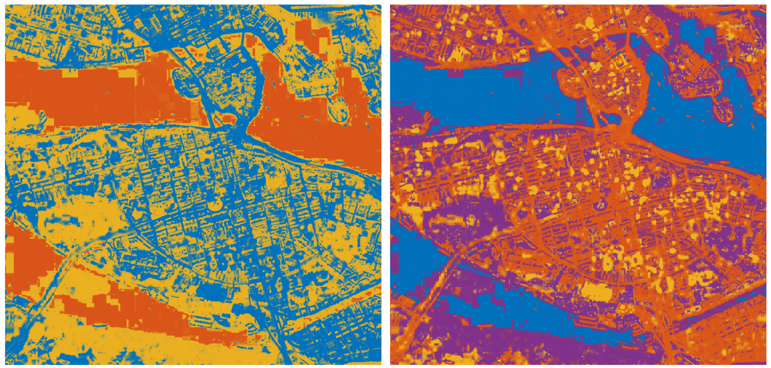
$$\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)} + \gamma \nabla \log \pi(\mathbf{Y}; \boldsymbol{\theta}^{(i)})$$

  where γ determines the step length.
- EM-algorithm: Set $\boldsymbol{\theta}^{(i+1)}$ such that

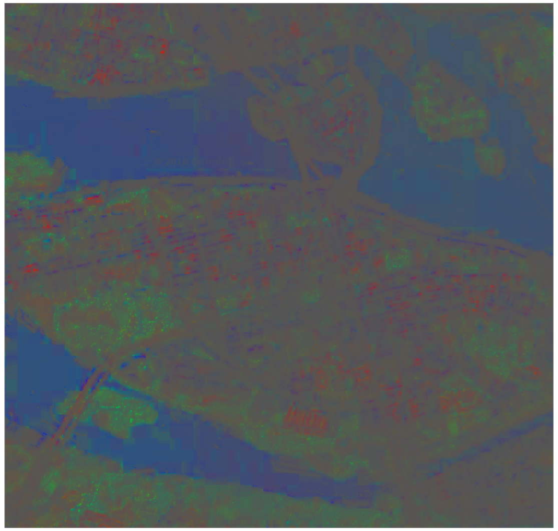
$$\sum_{i=1}^N \sum_{k=1}^K \mathsf{P}(z_i = k | \mathbf{Y}_i, \boldsymbol{\theta}^{(i)})(\nabla \log \pi_G(\mathbf{Y}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \nabla \log \pi_k) = 0$$

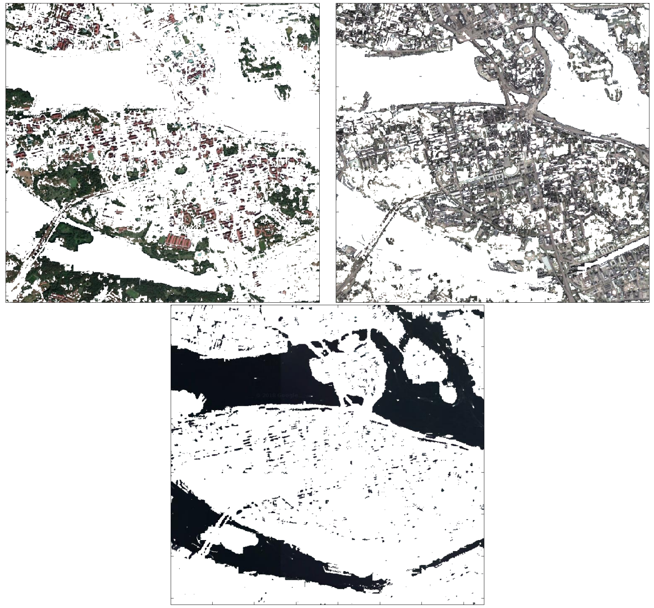
RGB classification

$K = 3$ $\quad$ $K = 4$

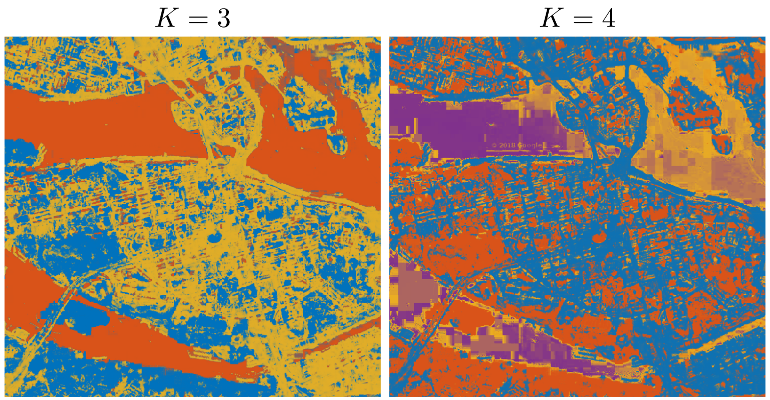


Relative colors





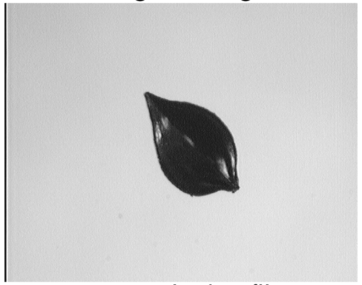
Classification using relative amount of green and blue



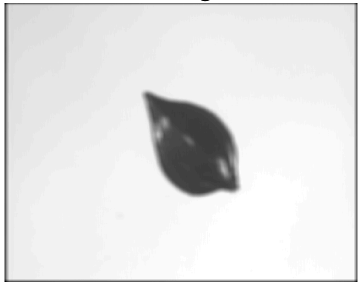
Including additional features

- The GMM is does not take spatial dependencies into account.
- The classes may have additional features except for raw pixel values which we may want to use.
- Today we will introduce some common image features that are useful both for segmentation and classification.
- On Wednesday, we will extend the mixture model so that the dependencies are modeled directly.

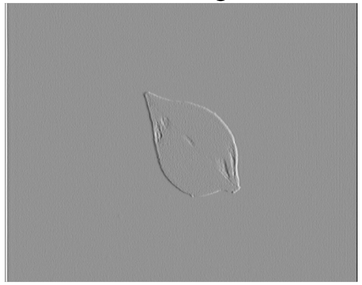
Original image



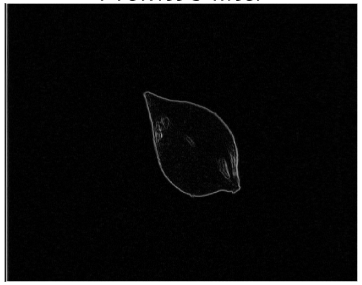
Smoothing filter



Horizontal edge filter



Prewitt's filter



Binary image

Image erotion

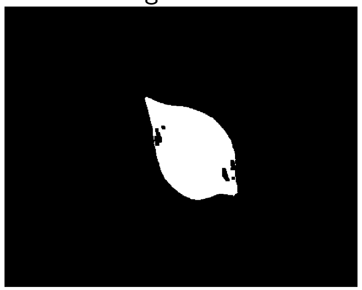


Image dilation

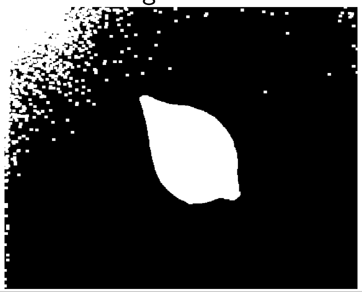


Image opening