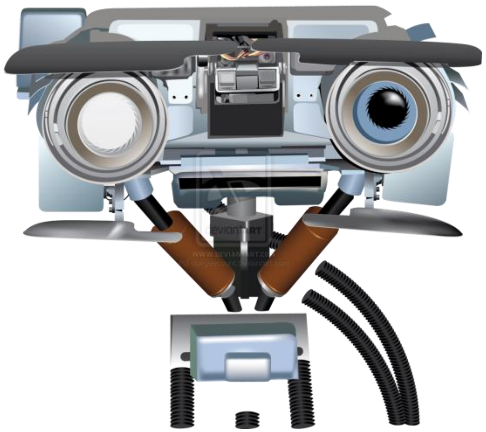
Today

- Discussing Pre-lecture Material

Reminders: PS2 is now posted, due Feb 24

Announcement: Pre-lecture Material for Feb 21



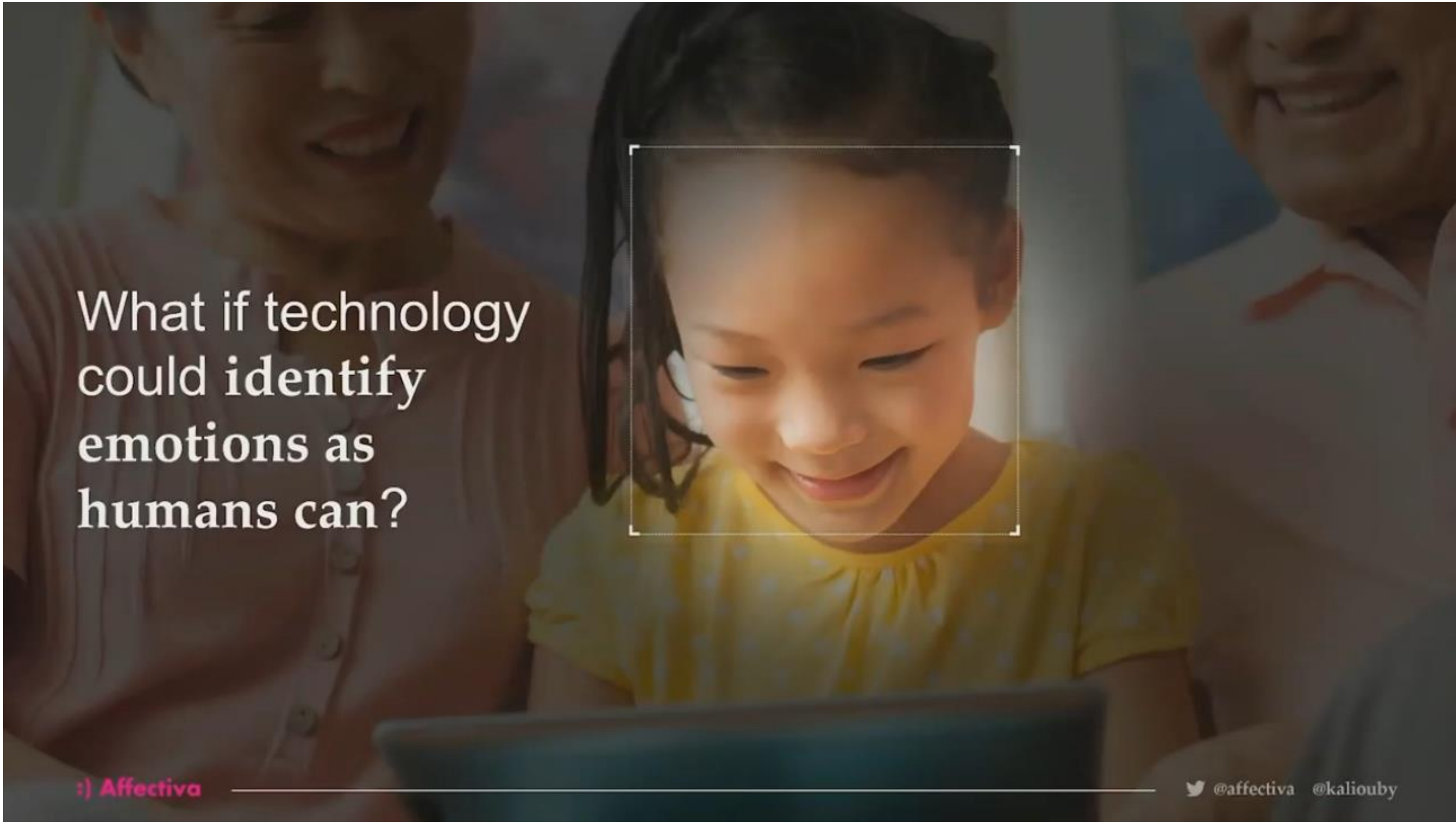
Pre-lecture Material

Humanizing Technology

:) Affectiva

Rana el Kaliouby

Co-founder and CEO


What if technology could identify emotions as humans can?
:) Affectiva
@affectiva @kaliouby

Emotional Intelligence

- What is the main problem definition?

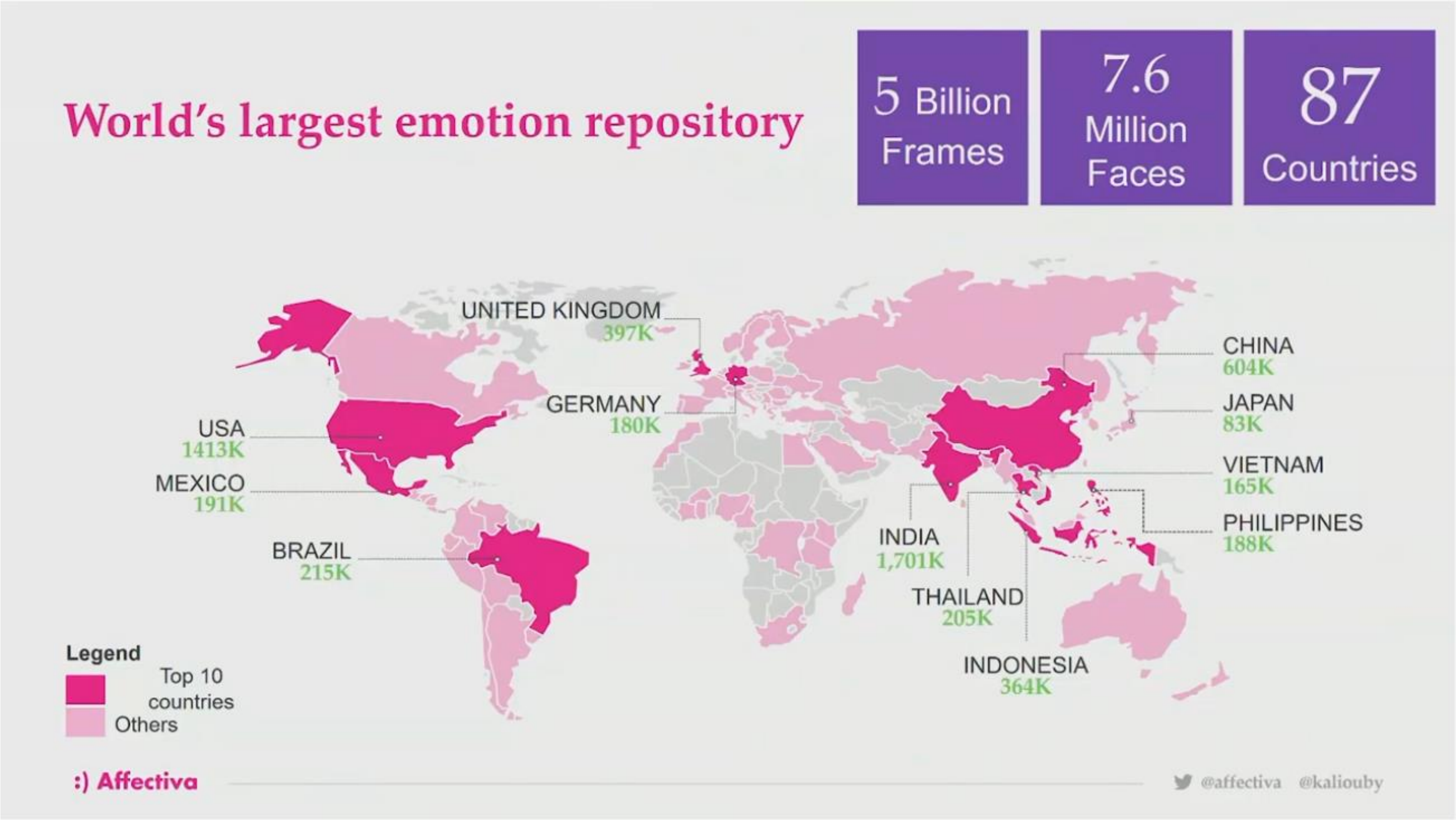
Emotional Intelligence

- What is the main problem definition?
  - Machine Learning Problem

Why?

Data Collection

- Consent, diversity, cultures



Emotional Intelligence

- What is the main problem definition?
  - Machine Learning Problem
  - Supervised vs. unsupervised

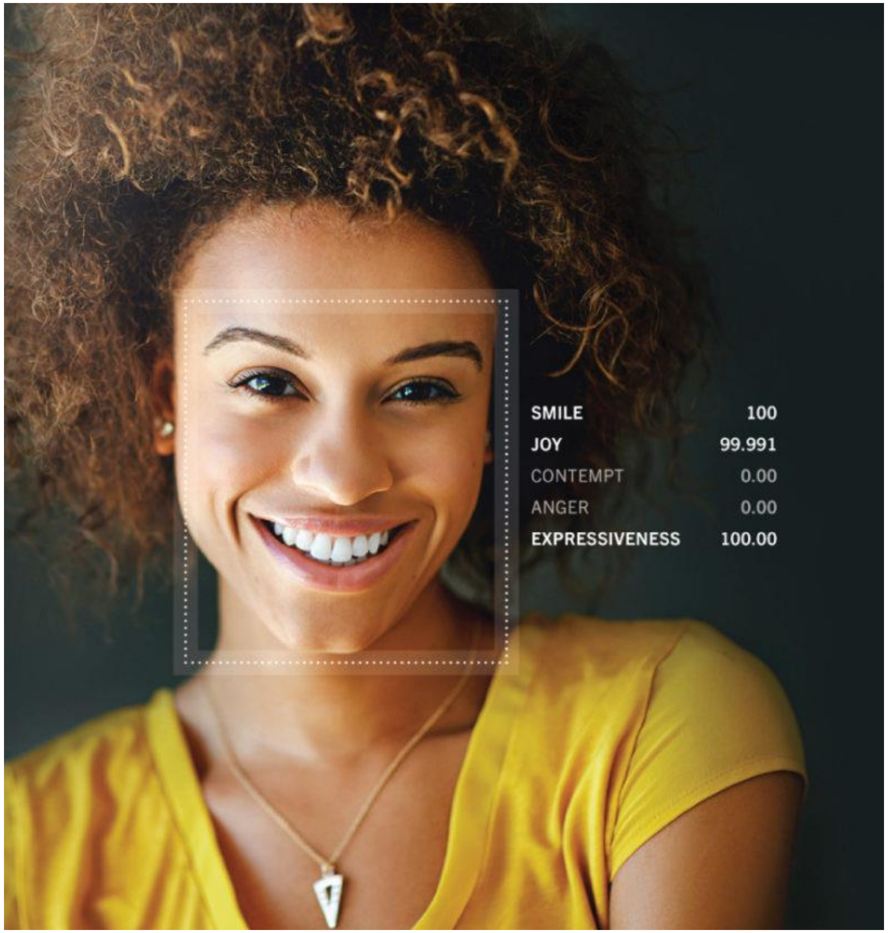
Labels *vs.* No labels

Emotional Intelligence

- What is the main problem definition?
  - Machine Learning Problem
  - Supervised vs. unsupervised
  - Classification: Happy, Sad, Angry, Surprised, Fear, …

Classification: Scale for Each Class

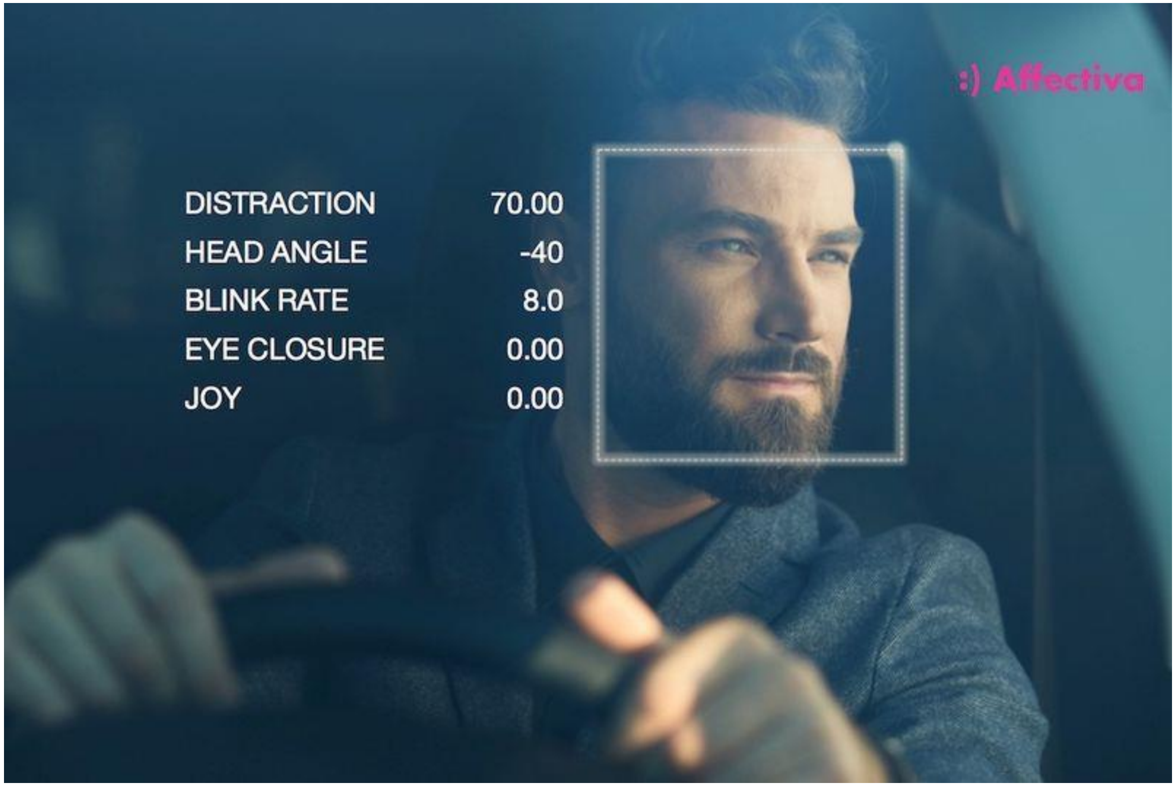
- Multi-class *vs.* Smile Classifier



:) Affectiva

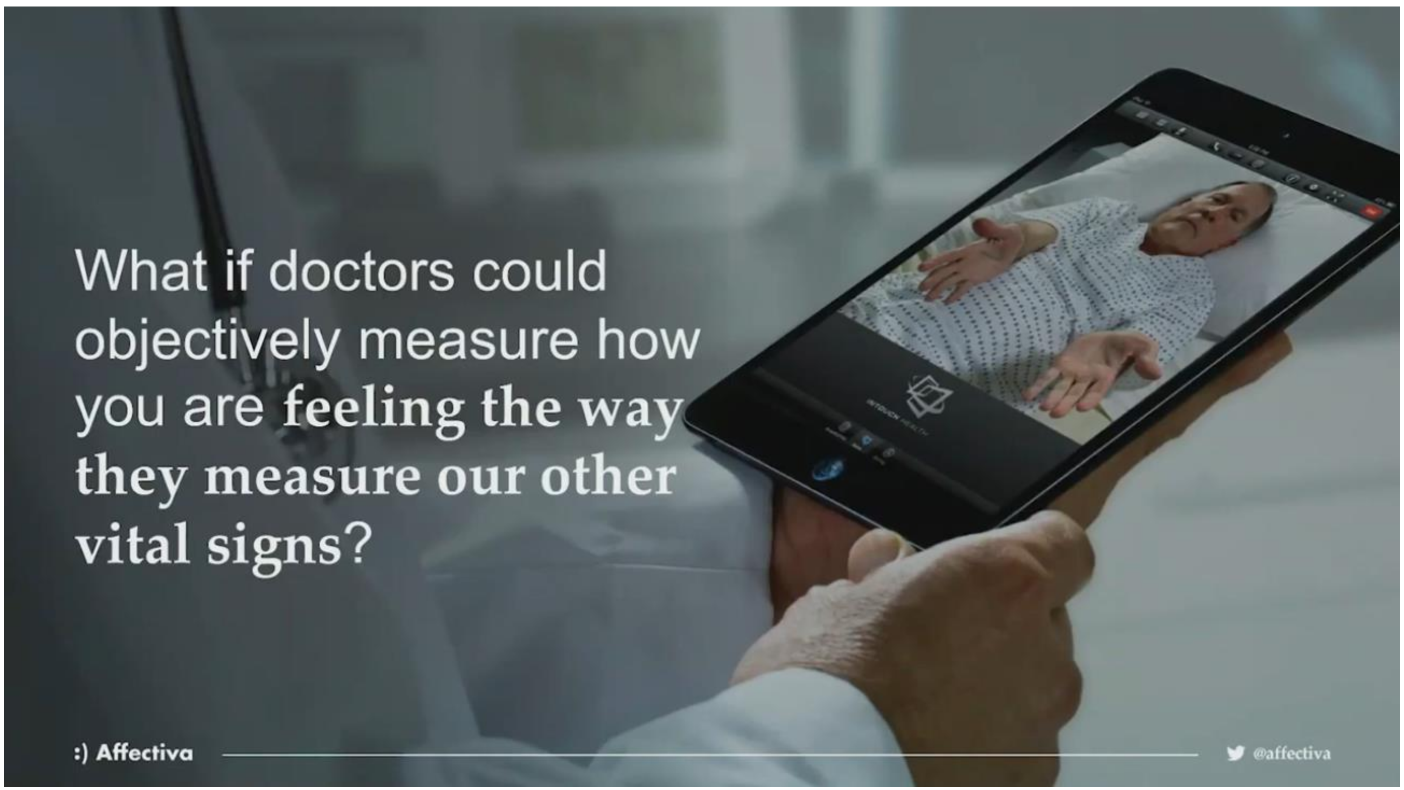
Applications

- Applications that benefit society:
  - Automotive Safety



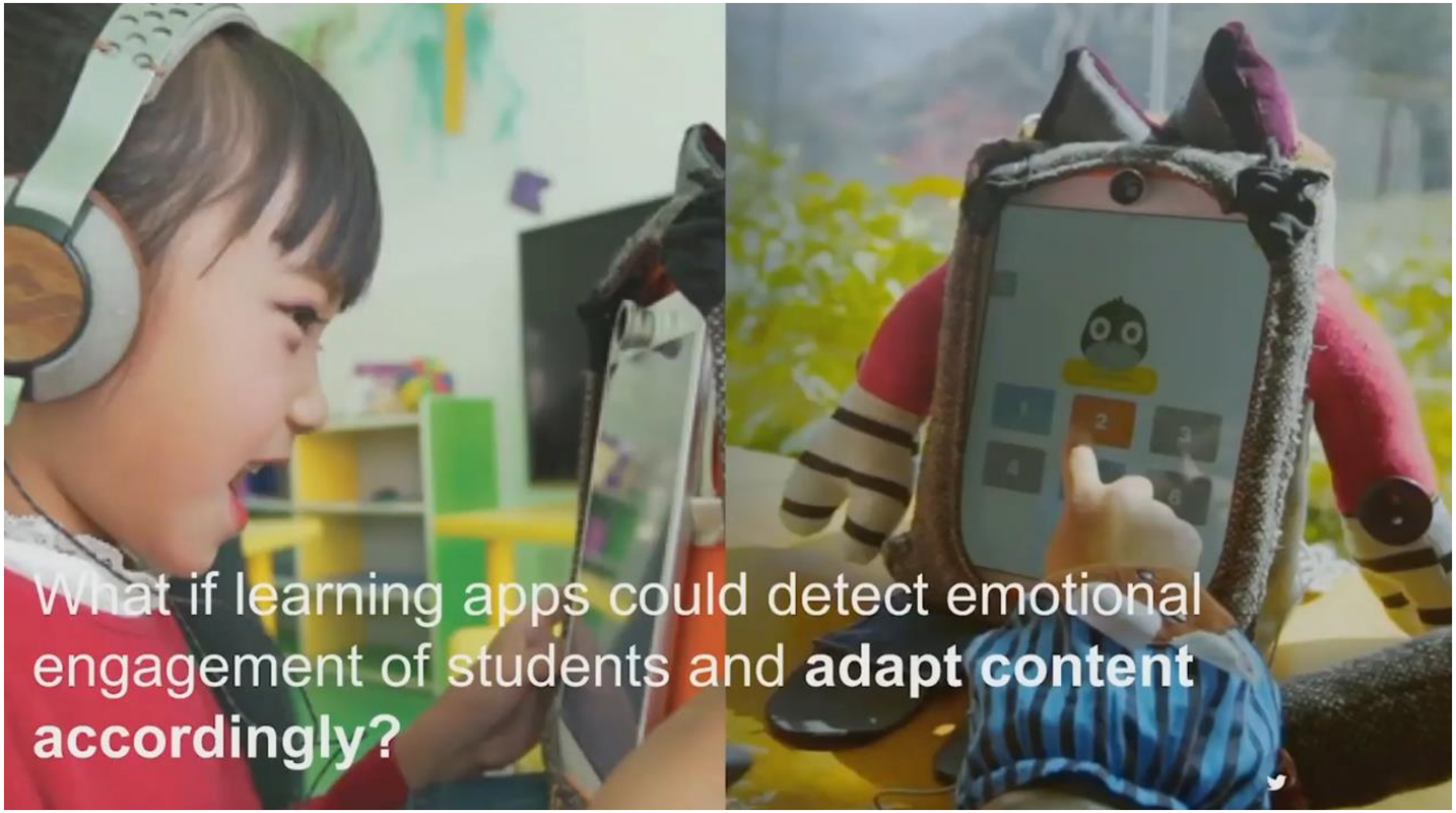
Applications

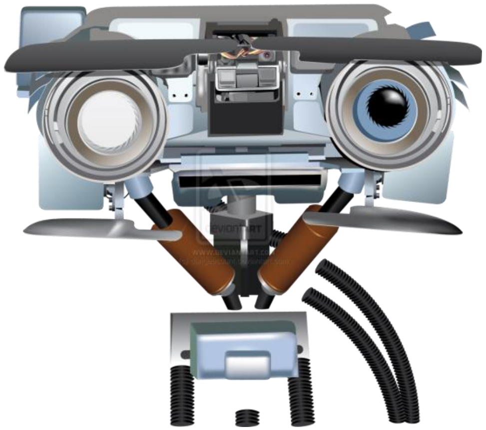
- Applications that benefit society:
  - Mental Health



Applications

- Applications that benefit society:
  – Education

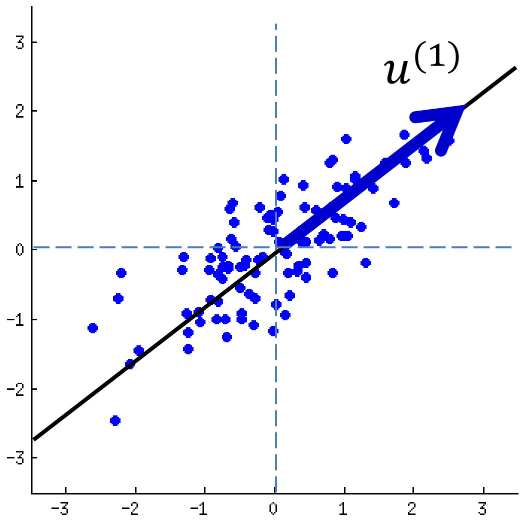


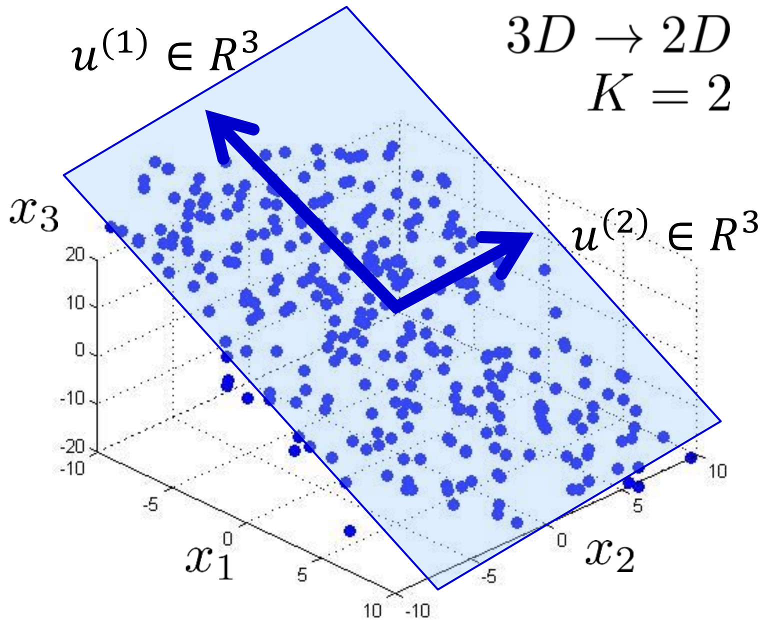


Unsupervised Learning II

Re-cap: Dimensionality Reduction

Choose subspace with minimal "information loss"





Reduce from 2-dimension to 1-dimension: Find a direction (a vector $u^{(1)}$) onto which to project the data, so as to minimize the projection error.

Reduce from n-dimension to K-dimension: Find K vectors $u^{(1)}, u^{(2)}, \ldots, u^{(K)}$ onto which to project the data so as to minimize the projection error.

PCA Algorithm

Normalize features (ensure every feature has zero mean) and optionally scale feature

Compute “covariance matrix” Σ:

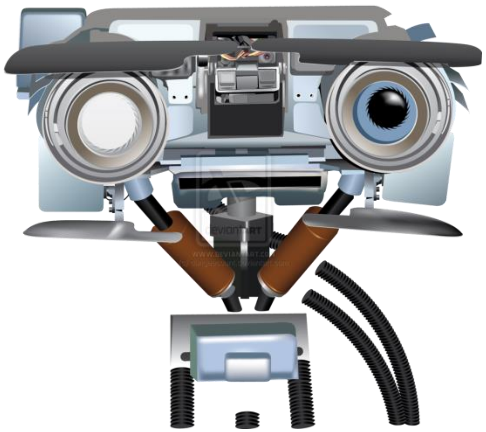
`Sigma =` $$\frac{1}{m}\sum_{i=1}^{m}(x^{(i)})(x^{(i)})^T$$

Compute its “eigenvectors”:

`[U,S,V] = svd(Sigma);` $$U = \begin{bmatrix} | & | & & | \\ u^{(1)} & u^{(2)} & \dots & u^{(n)} \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{n \times n}$$

Keep first K eigenvectors and project to get new features **`z`**

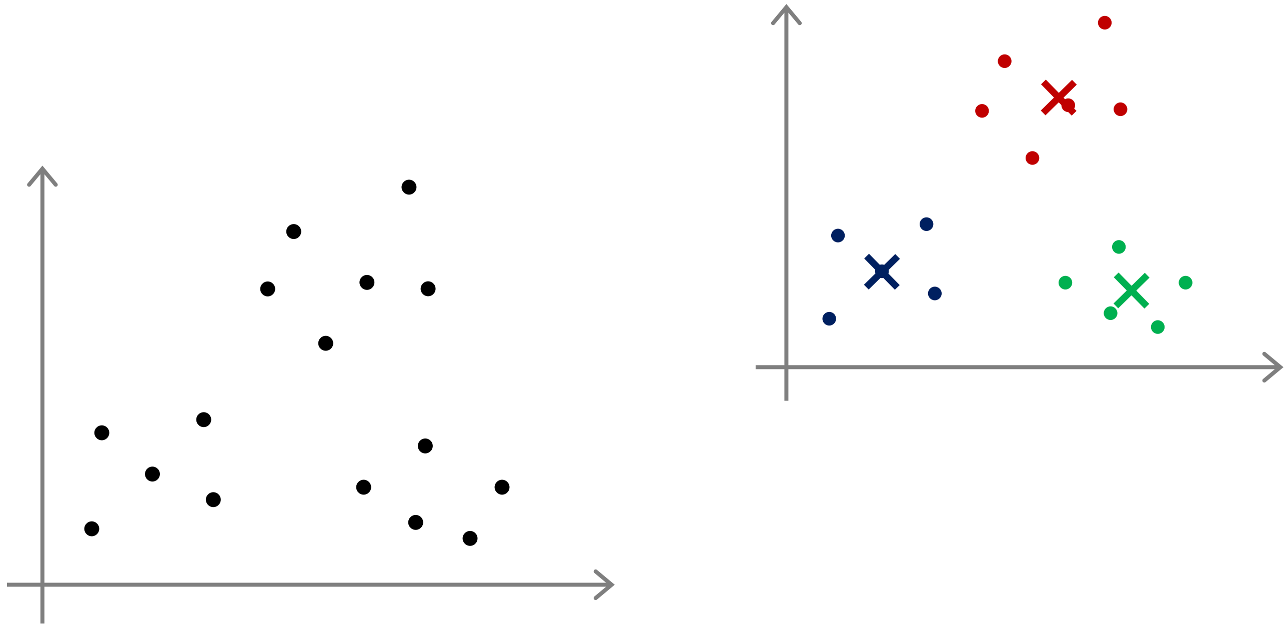
```
Ureduce = U(:,1:K);
z = Ureduce'*x;
```



Unsupervised Learning II

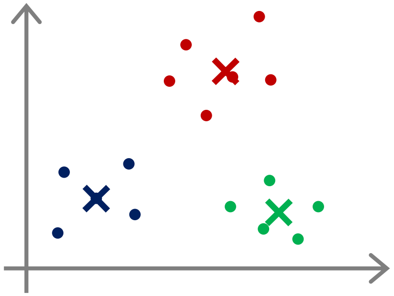
Re-cap: k-means Clustering

Goal: k-means Clustering



slide credit: Andrew Ng

K-means algorithm



Randomly initialize K cluster centroids $\mu_1, \mu_2, \ldots, \mu_K \in \mathbb{R}^n$

Repeat {

for i = 1 to m

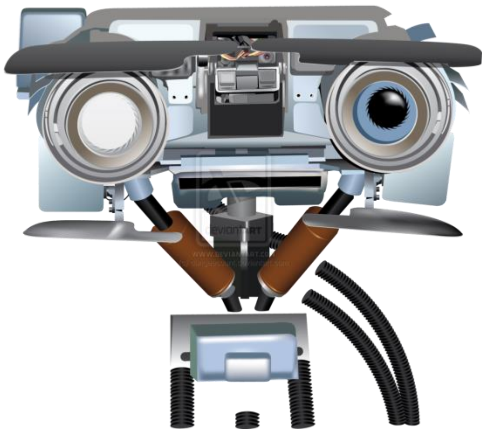
$c^{(i)}$:= index (from 1 to K) of cluster centroid closest to $x^{(i)}$

for k = 1 to K

μ_k := average (mean) of points assigned to cluster k

}

slide credit: Andrew Ng

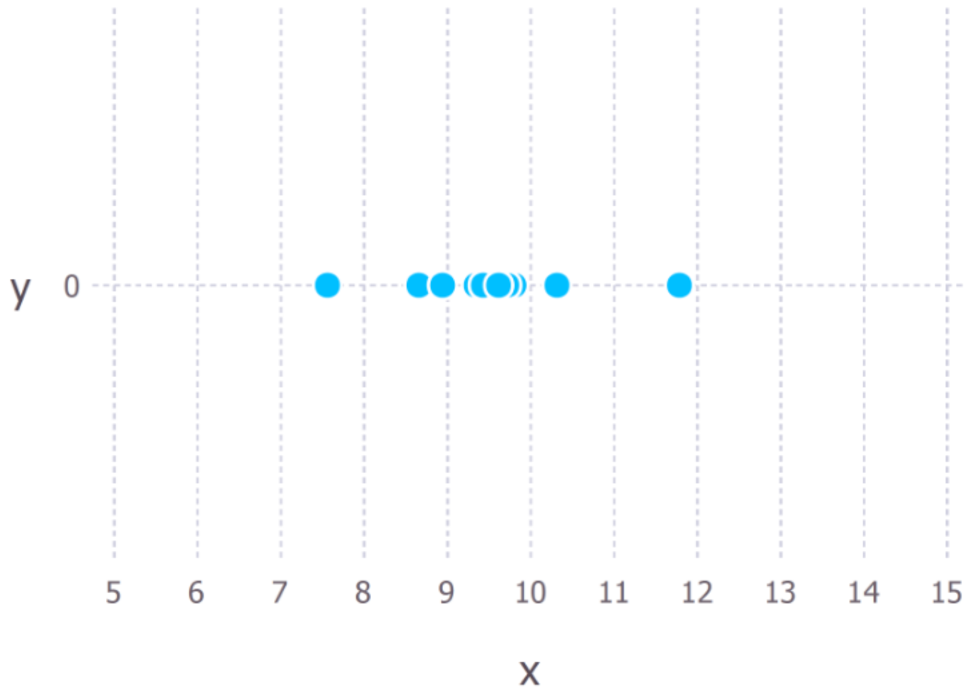


Unsupervised Learning II

Mixtures of Gaussians

Observed Data from a Single Gaussian

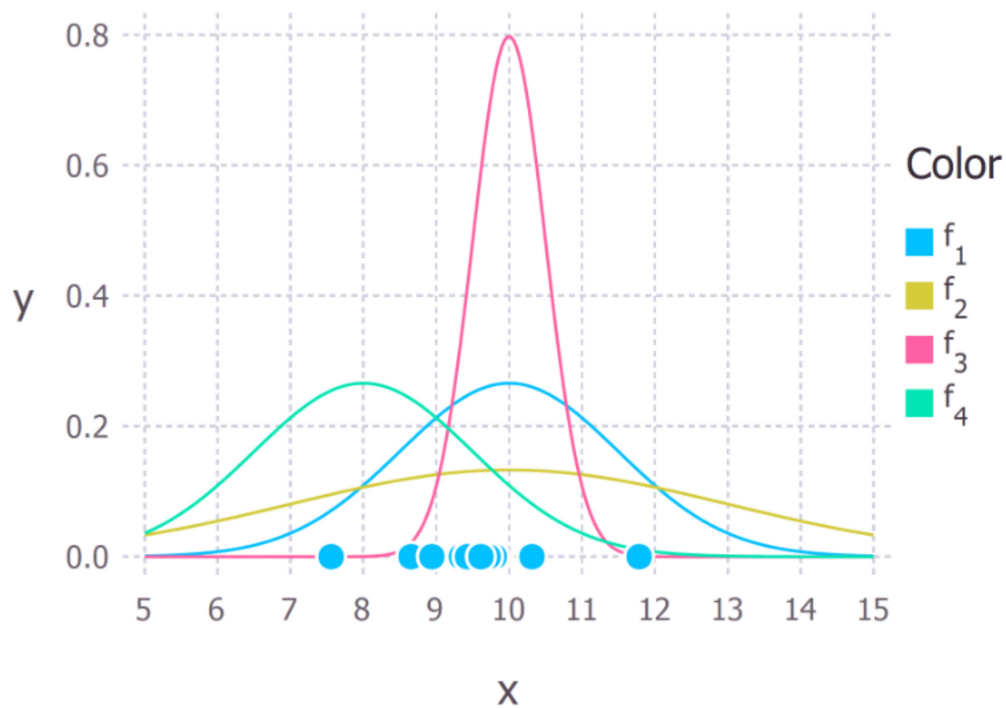
- Ten observed data points from some process



Jonny Brooks-Bartlett

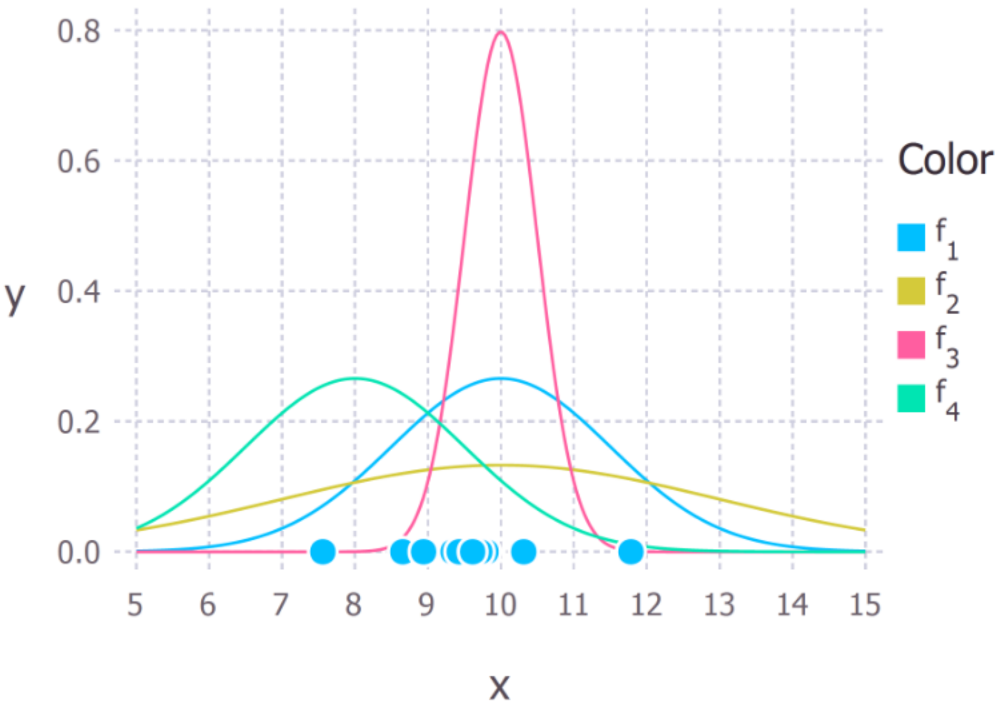
Learning the Model

- We want to know *which curve was most likely responsible for creating the data points that we observed?*



Maximum Likelihood

- Maximum likelihood estimation is a method that will find the values of μ and σ that result in the curve that best fits the data.



Jonny Brooks-Bartlett

Calculating Maximum Likelihood Estimates

- What we want to calculate is the total probability of observing all of the data, *i.e.* the joint probability distribution of all observed data points.

- To do this we would need to calculate some conditional probabilities, which can get very difficult.

- So it is here that we'll make our first assumption. *The assumption is that each data point is generated independently of the others*.

Calculating Maximum Likelihood Estimates

The probability density of observing a single data point x, that is generated from a Gaussian distribution is given by:

$$P(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

In our example the total (joint) probability density of observing the three data points is given by:

$$P(9,9.5,11;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(9-\mu)^2}{2\sigma^2}\right)\times\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(9.5-\mu)^2}{2\sigma^2}\right) \\ \times\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(11-\mu)^2}{2\sigma^2}\right)$$

Calculating Maximum Likelihood Estimates

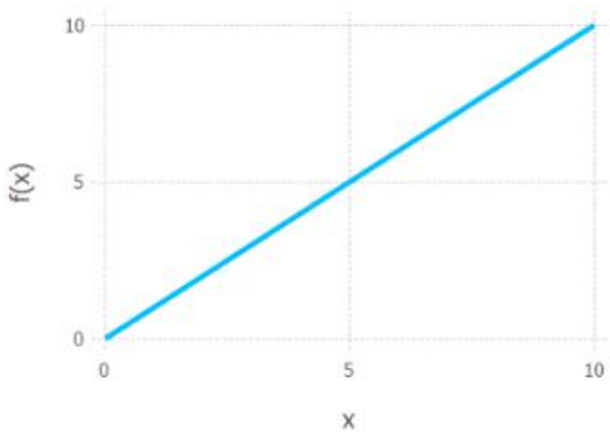
- We need to find the values of μ and σ that results in giving the maximum value of the above expression.

- The above expression for the total probability is difficult to differentiate.

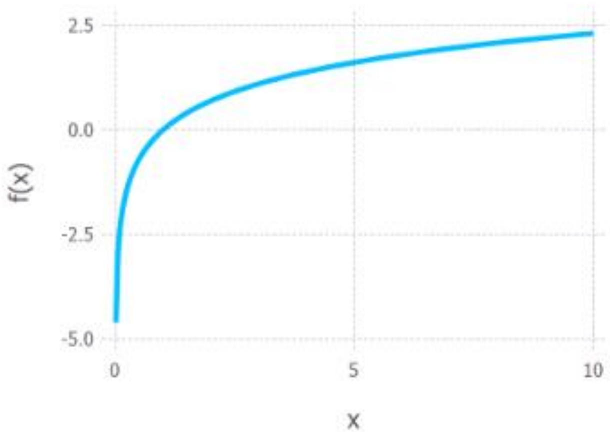
- It is almost always simplified by taking the natural logarithm of the expression.

Log Likelihood

- This is absolutely fine because the natural logarithm is a monotonically increasing function.



(a) $f(x) = x$



(b) $f(x) = \ln(x)$

Log Likelihood

Taking logs of the original expression gives us:

$$\ln(P(x;\mu,\sigma)) = \ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right) - \frac{(9-\mu)^2}{2\sigma^2} + \ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right) - \frac{(9.5-\mu)^2}{2\sigma^2} + \ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right) - \frac{(11-\mu)^2}{2\sigma^2}$$

This expression can be simplified again using the laws of logarithms to obtain:

$$\ln(P(x;\mu,\sigma)) = -3\ln(\sigma) - \frac{3}{2}\ln(2\pi) - \frac{1}{2\sigma^2}\left[(9-\mu)^2 + (9.5-\mu)^2 + (11-\mu)^2\right]$$

Computing μ_{ML}

This expression can be differentiated to find the maximum. In this example we'll find the MLE of the mean, μ. To do this we take the partial derivative of the function with respect to μ, giving

$$\frac{\partial \ln(P(x;\mu,\sigma))}{\partial \mu} = \frac{1}{\sigma^2}\left[9 + 9.5 + 11 - 3\mu\right].$$

Finally, setting the left hand side of the equation to zero and then rearranging for μ gives:

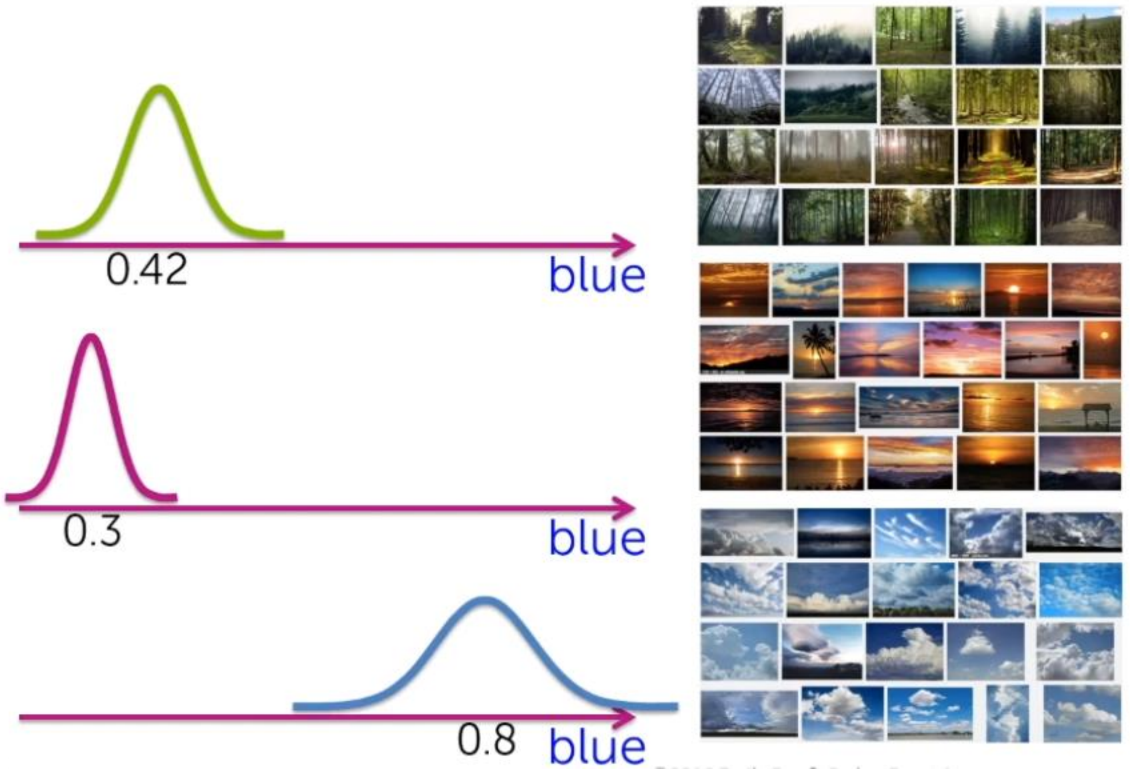
Do the same for σ

$$\mu = \frac{9 + 9.5 + 11}{3} = 9.833$$

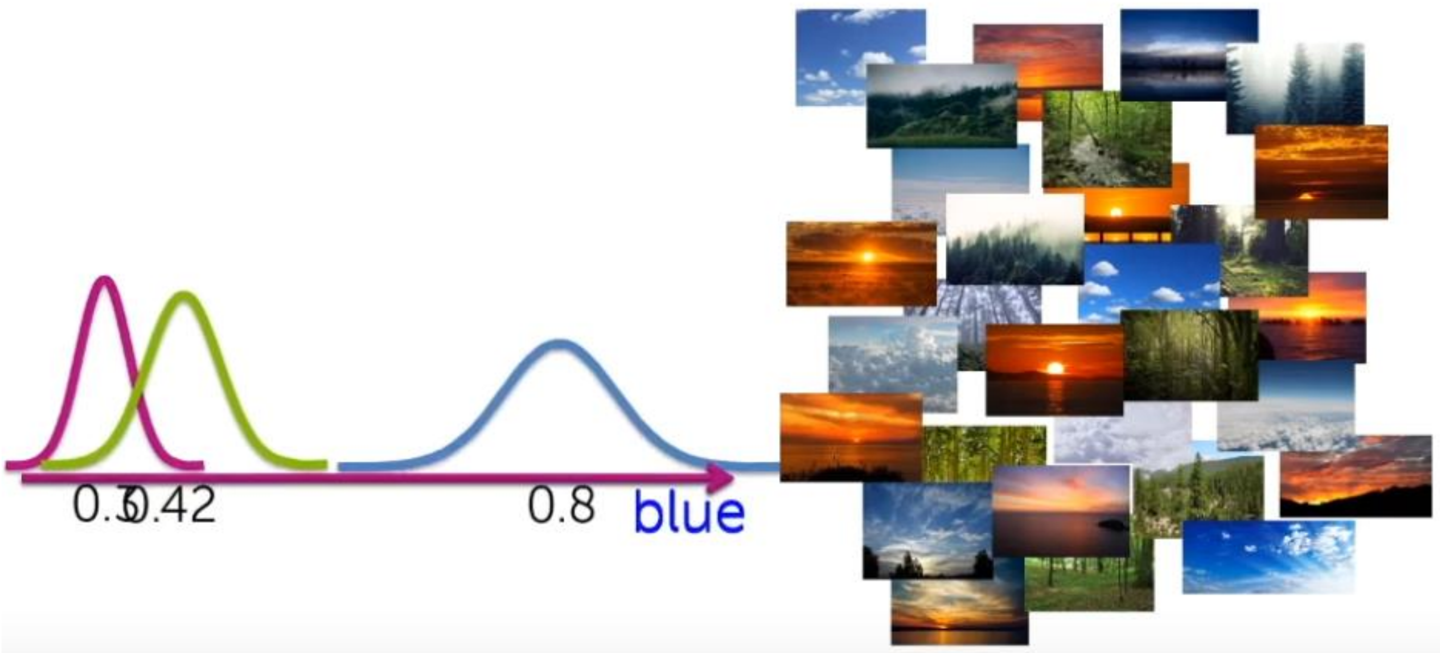
$\boldsymbol{\mu_{ML}}$

Jonny Brooks-Bartlett

Mixtures of Gaussians

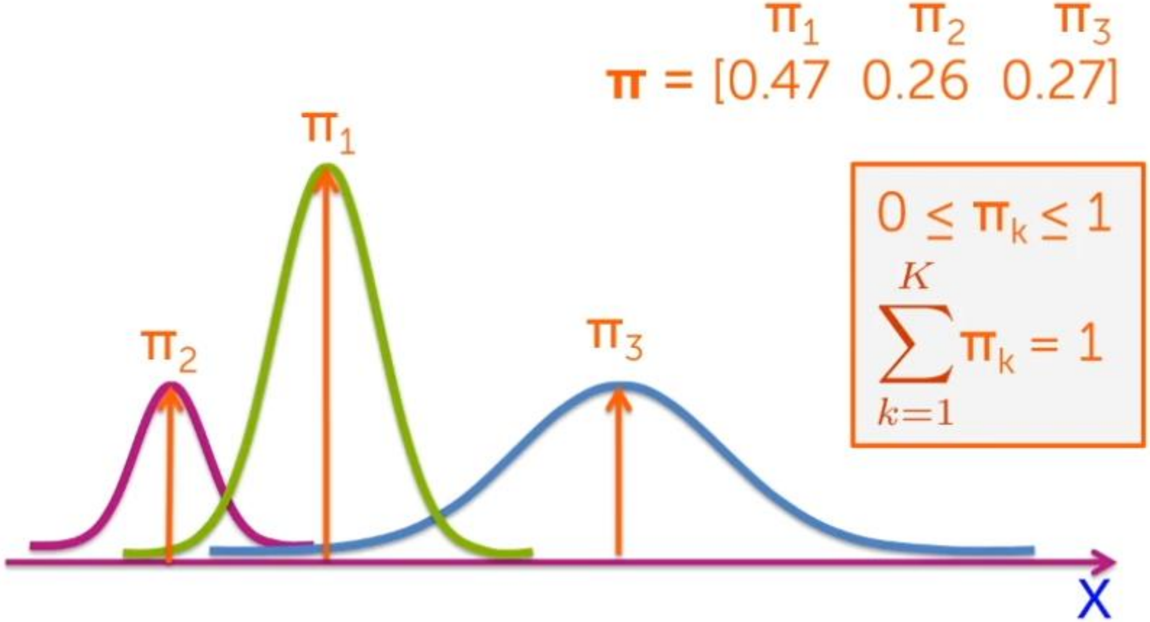


Mixtures of Gaussians



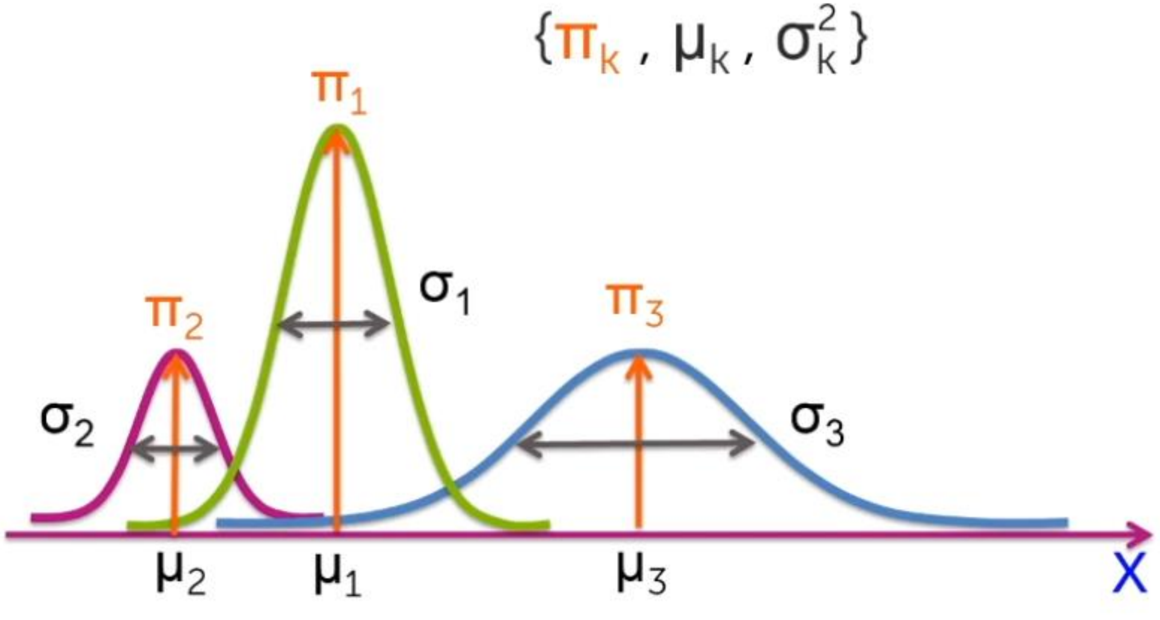
Mixtures of Gaussians

- Associate a weight π_k with each Gaussian Component: "The mixing coefficients"

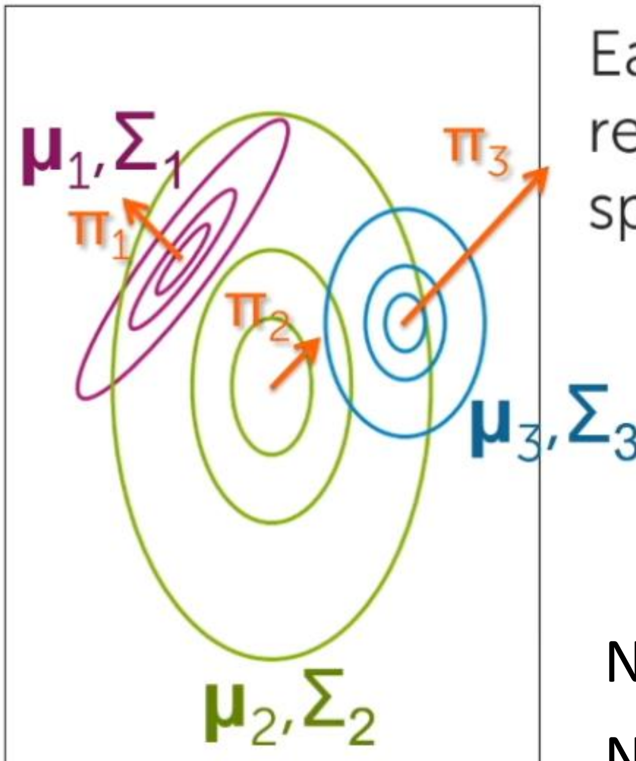


Mixtures of Gaussians

- Location and spread for the distributions comprising the Gaussians



Higher Dimensions



Each mixture component represents a unique cluster specified by:

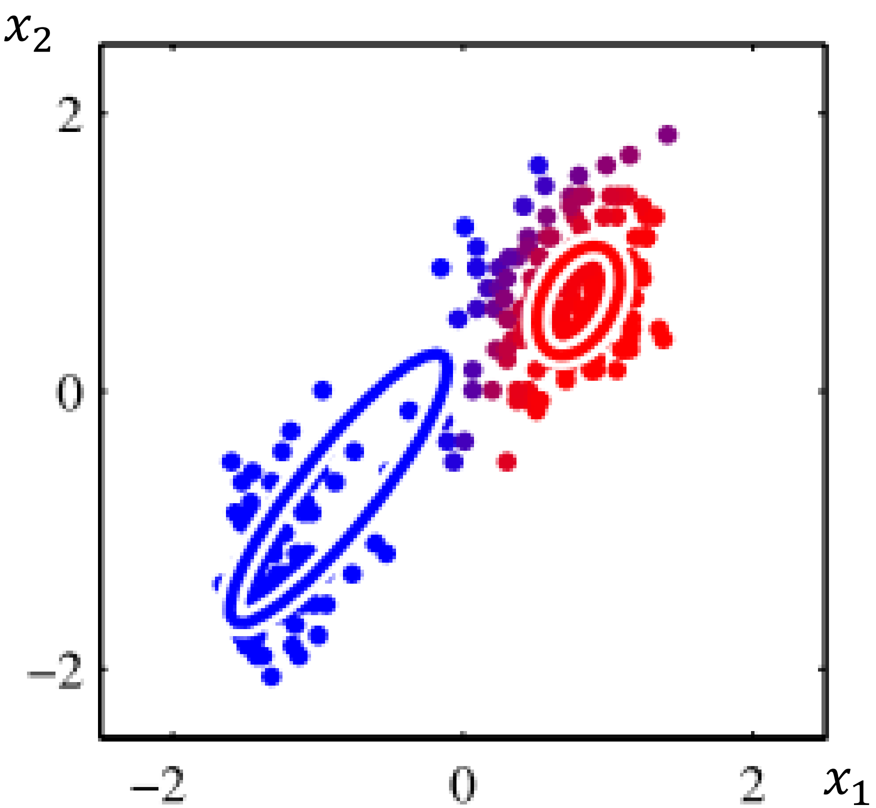
$$\{\pi_k , \mu_k , \Sigma_k\}$$

Naturally generated clusters!

Naturally a generative model!

vs. discriminative models

Mixtures of Gaussians: “Soft” cluster membership

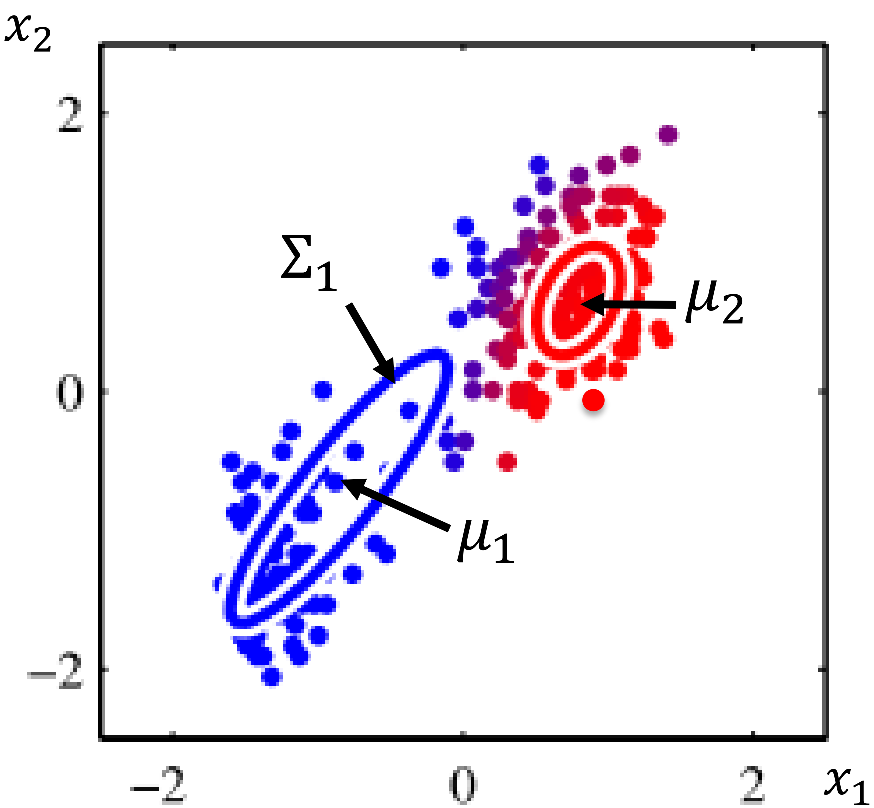


Define a distribution over x :

To generate each point x,

- Choose its cluster component z
- Sample x from the Gaussian distribution for that component

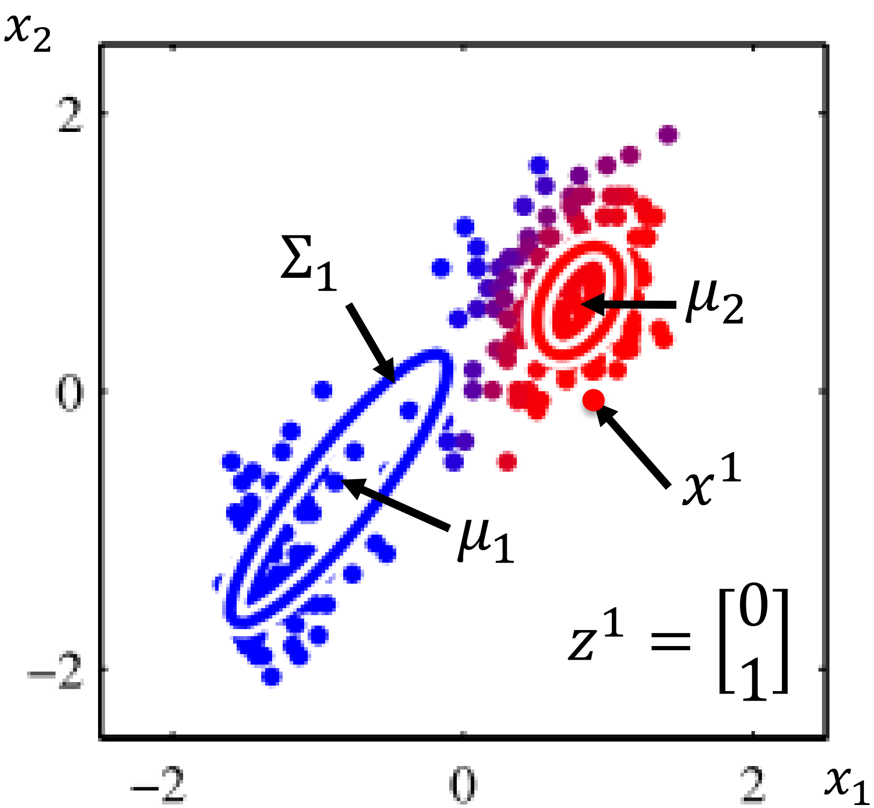
Mixtures of Gaussians:
component membership variable z



- Assume K components, k-th component is a Gaussian with parameters μ_k, Σ_k
- Introduce discrete r.v. $z \in R^K$ that denotes the component that generates the point
- one element of z is equal to 1 and others are 0, *i.e.* "one-hot":

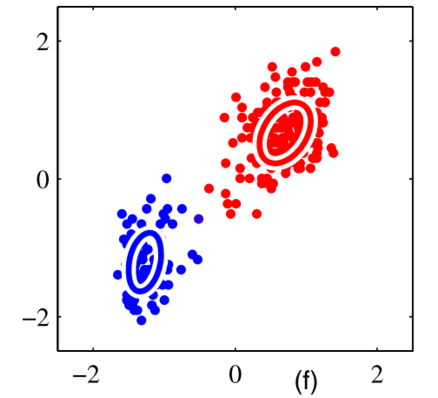
$$z_k \in \{0,1\} \text{ and } \sum_k z_k = 1$$

Mixtures of Gaussians:
Data generation example



- Suppose $K = 2$ components, k-th component is a Gaussian with parameters μ_k, Σ_k
- To sample i-th data point:
  - Pick component z^i with $\mathrm{p}(z_k = 1) = \pi_k$ (parameter)
  - for example, $\pi_1 = 0.5$, and we picked $z^1 = [0, 1]^T$
  - Pick data point x^i with probability $\mathrm{N}(\mathrm{x}; \mu_k, \Sigma_k)$

Mixtures of Gaussians



- $z_k \in \{0,1\}$ and $\sum_k z_k = 1$
- K components, k-th component is a Gaussian with parameters μ_k, Σ_k

- define the joint distribution $\mathrm{p}(\mathbf{x}, \boldsymbol{z})$ in terms of a marginal distribution $\mathrm{p}(\boldsymbol{z})$ and a conditional distribution $\mathrm{p}(\mathbf{x}|\boldsymbol{z})$

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{z})p(\mathbf{x}|\mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- where

$$p(z_k = 1) = \pi_k \qquad 0 \leqslant \pi_k \leqslant 1 \qquad \sum_{k=1}^{K} \pi_k = 1$$

$$p(\mathbf{x}|\mathbf{z}) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_k}$$

Substitute and simplify

Maximum Likelihood Solution for Mixture of Gaussians

- This distribution is known as a Mixture of Gaussians

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- We can estimate parameters using Maximum Likelihood, i.e. maximize

$$\ln p(\boldsymbol{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) =$$

$$\ln p(x^1, x^2, \dots, x^N|\pi_1, \dots, \pi_K, \mu_1, \dots, \mu_K, \Sigma_1, \dots, \Sigma_K)$$

- This algorithm is called Expectation Maximization (EM)

Expectation Maximization

- We can estimate parameters using Maximum Likelihood, i.e. minimize neg. log likelihood

$$-\ln p(\mathbf{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = -\sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

- Problem: don't know values of *"hidden"* (or "*latent*") variable z, we don't observe it

- Solution: treat z^i as parameters and use coordinate descent

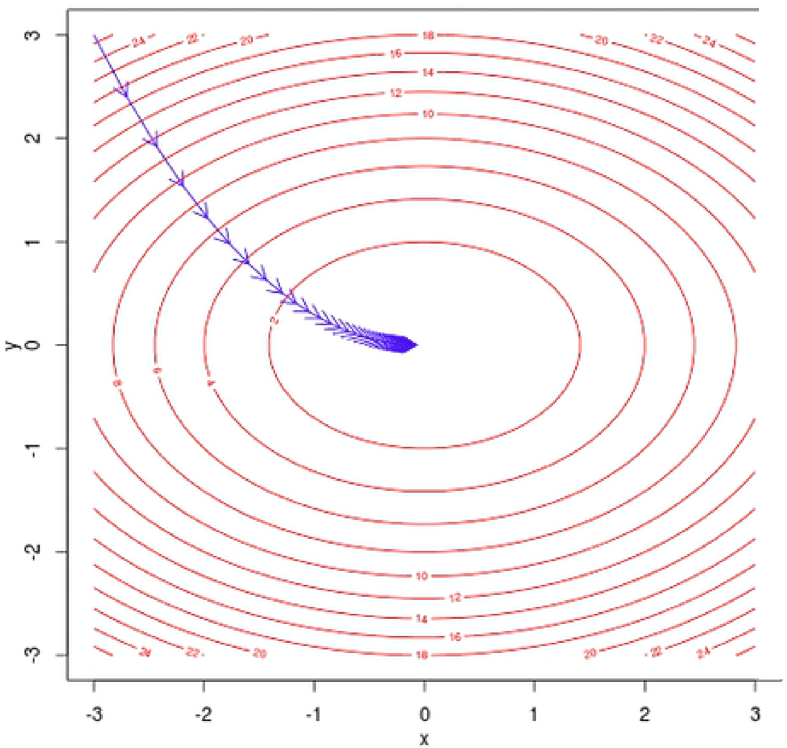
Coordinate Descent

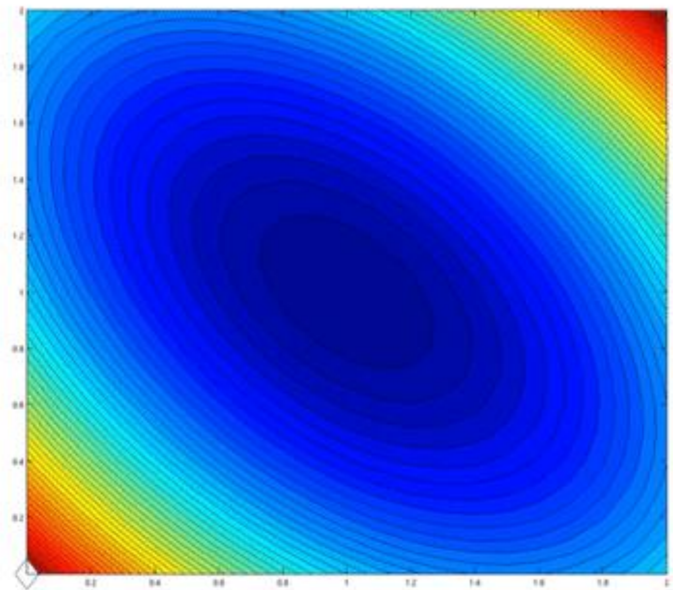
gradient descent:

- Minimize w.r.t all parameters at each step

coordinate descent:

- fix some coordinates, minimize w.r.t. the rest
- alternate

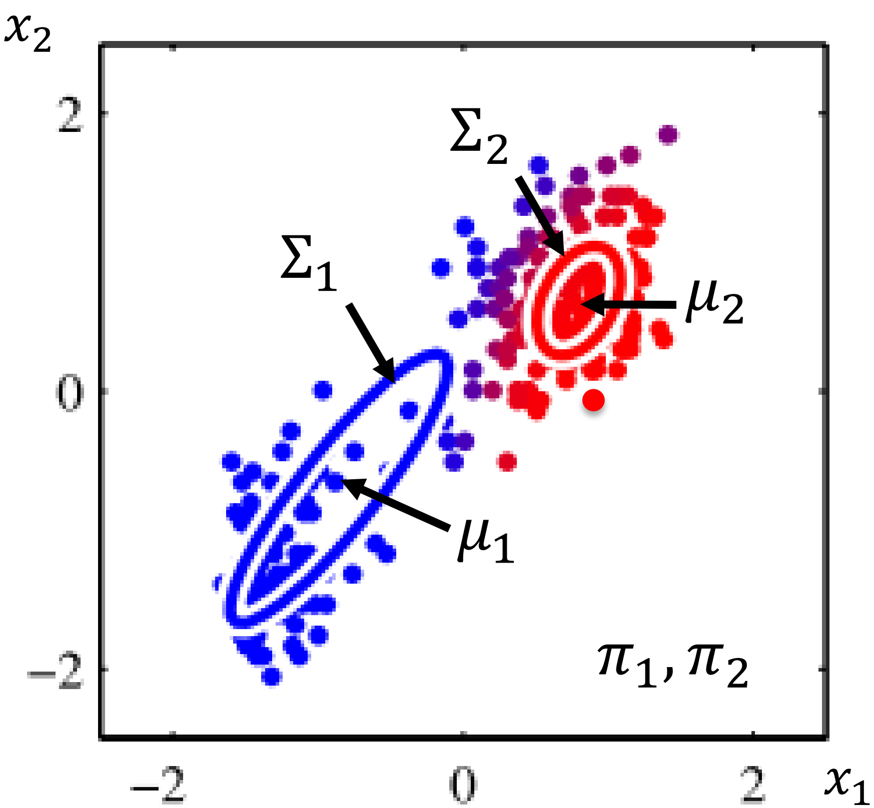




Credit: Martin Takac

Credit: http://vis.supstat.com/2013/03/gradient-descent-algorithm-with-r/

Expectation Maximization



Coordinate descent for Mixtures of Gaussians:

Alternate

- fix $\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$, update z^i
- fix z^i, update $\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$

Expectation Maximization Algorithm

- A general technique for finding maximum likelihood estimators in latent variable models
- Initialize and iterate until convergence:

  E-Step: estimate posterior probability of the latent variables $p(z_k|x)$, holding parameters fixed

  M-Step: maximize likelihood w.r.t parameters (here $\mu_k, \Sigma_\mathrm{k}, \pi_k$) using latent probabilities from E-step

EM for Gaussian Mixtures Example

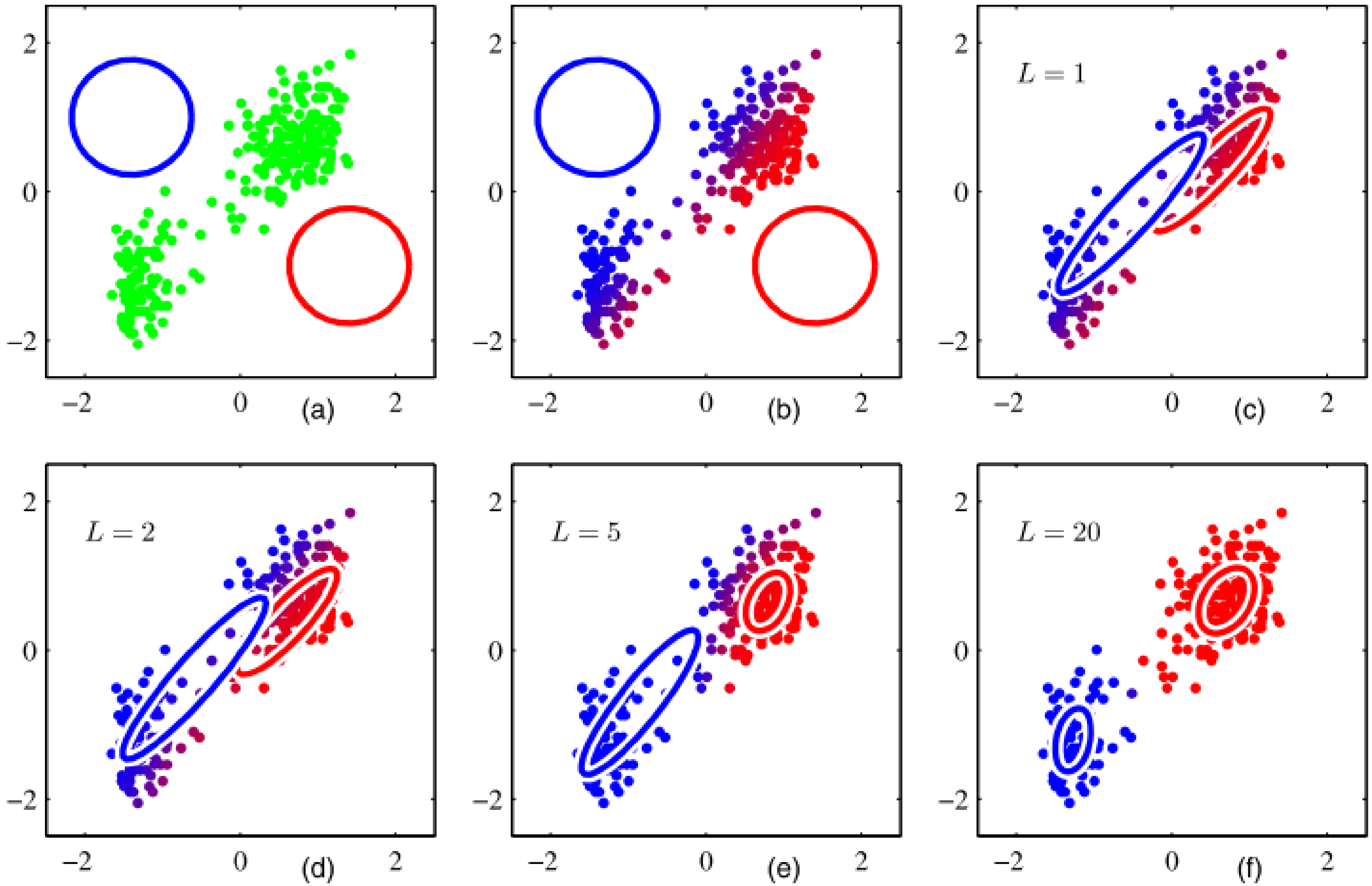


Figure 9.8 Illustration of the EM algorithm

EM for Gaussian Mixtures

1. Initialize the means $\boldsymbol{\mu}_k$, covariances $\boldsymbol{\Sigma}_k$ and mixing coefficients π_k, and evaluate the initial value of the log likelihood.
2. **E step**. Evaluate the responsibilities using the current parameter values

$$\gamma(z_k) \equiv p(z_k = 1|\mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}. \tag{9.23}$$

3. **M step**. Re-estimate the parameters using the current responsibilities

$$\boldsymbol{\mu}_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n \qquad N_k = \sum_{n=1}^{N} \gamma(z_{nk}) \tag{9.24}$$

$$\boldsymbol{\Sigma}_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \left(\mathbf{x}_n - \boldsymbol{\mu}_k^{\text{new}}\right) \left(\mathbf{x}_n - \boldsymbol{\mu}_k^{\text{new}}\right)^{\mathrm{T}} \tag{9.25}$$

$$\pi_k^{\text{new}} = \frac{N_k}{N} \tag{9.26}$$

see Bishop Ch. 9.2