<table>
<thead>
<tr>
<th></th>
<th>First principles</th>
<th></th>
<th>Data driven</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Small data</td>
<td></td>
<td>Big data to train</td>
</tr>
<tr>
<td>Domain expertise</td>
<td>High reliance on domain expertise</td>
<td></td>
<td>Results with little domain knowledge</td>
</tr>
<tr>
<td>Fidelity/Robustness</td>
<td>Universal link can handle non-linear complex relations</td>
<td></td>
<td>Limited by the range of values spanned by training data</td>
</tr>
<tr>
<td>Adaptability</td>
<td>Complex and time consuming derivation to use new relations</td>
<td></td>
<td>Rapidly adapt to new problems</td>
</tr>
<tr>
<td>Interpretability</td>
<td>Parameters are physical!</td>
<td></td>
<td>Physically agnostic, limited by the rigidity of the functional form</td>
</tr>
</tbody>
</table>

Perceived Importance. SIO       SP       Peter       Google
Machine learning versus knowledge based
Supervised learning

Supervised $y = w^T x$

Training set $\{(x^1, y^1), (x^2, y^2), (x^3, y^3)\}$

We are given the two classes.
Unsupervised learning

Supervised $y = wx$

Training set $\{(x_1^1, x_2^1), (x_1^2, x_2^2), (x_1^3, x_2^3)\}$
Unsupervised learning

Unsupervised machine learning is inferring a function to describe hidden structure from "unlabeled" data (a classification or categorization is not included in the observations). Since the examples given to the learner are unlabeled, there is no evaluation of the accuracy of the structure that is output by the relevant algorithm—which is one way of distinguishing unsupervised learning from supervised learning.

We are not interested in prediction

Supervised learning: all classification and regression.

\[ Y = W^T X \]

Prediction is important.
Unsupervised learning

- Unsupervised learning is more subjective than supervised learning, as there is no simple goal for the analysis, such as prediction of a response.
- But techniques for unsupervised learning are of growing importance in several fields:
  - subgroups of breast cancer patients grouped by their gene expression measurements,
  - groups of shoppers characterized by their browsing and purchase histories,
  - movies grouped by the ratings assigned by movie viewers.
- It is often easier to obtain unlabeled data — from a lab instrument or a computer — than labeled data, which can require human intervention.
  - For example it is difficult to automatically assess the overall sentiment of a movie review: is it favorable or not?
Kmeans

- **Input**: Points $x_1, \ldots, x_N \in \mathbb{R}^p$; integer $K$
- **Output**: “Centers”, or representatives, $\mu_1, \ldots, \mu_K \in \mathbb{R}^p$
- Output also $z_1, \ldots, z_N \in \mathbb{R}^K$

**Goal**: Minimize average squared distance between points and their nearest representatives:

$$ cost(\mu_1, \ldots, \mu_K) = \sum_{n=1}^{N} \min_j \|x_n - \mu_j\| $$

The centers carve $\mathbb{R}^p$ up into $k$ convex regions: $\mu_j$’s region consists of points for which it is the closest center.
K-means

\[ J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \| x_n - \mu_k \|^2 \]  \hspace{1cm} (9.1)

Solving for \( r_{nk} \)

\[ r_{nk} = \begin{cases} 
1 & \text{if } k = \arg \min_j \| x_n - \mu_j \|^2 \\
0 & \text{otherwise.} 
\end{cases} \]  \hspace{1cm} (9.2)

Differentiating for \( \mu_k \)

\[ \frac{\partial}{\partial \mu_k} \left( 2 \sum_{n=1}^{N} r_{nk} (x_n - \mu_k) \right) = 0 \]  \hspace{1cm} (9.3)

which we can easily solve for \( \mu_k \) to give

\[ \mu_k = \frac{\sum_{n} r_{nk} x_n}{\sum_{n} r_{nk}} \]  \hspace{1cm} (9.4)
Section 9.4

Exercise 9.1

Consider first the determination of the $\mu_i$-means algorithm. We can do this through an iterative procedure in which each iteration minimizes the objective function $J$ with respect to the $\mu_i$ and updating $r_{nk}$ fixed. This two-stage procedure is known as the $K$-means algorithm. For this reason, the procedure is repeated until convergence. We shall see that these two stages correspond respectively to the E (expectation) and M (maximization) steps of the EM algorithm, and to emphasize this we shall use the notation $E$ step and $M$ step in the context of the algorithm. The convergence of the $K$-means algorithm were studied by MacQueen (1967).

Now consider the optimization of the $J$ function which we can easily solve for $\mu_i$. In other words, we simply assign the points to the closest cluster centre. More formally, this can be expressed as

$$
J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \| x_n - \mu_k \|^2
$$

where $r_{nk} = \begin{cases} 1 & \text{if } k = \arg \min_j \| x_n - \mu_j \|^2 \\ 0 & \text{otherwise.} \end{cases}$

Then in the first $E$ step, we assign each point $x_n$ to the closest cluster centre for whichever value of $k$, this optimization can be performed easily to give a closed form solution. For this example, we have made a linear re-scaling of the variables outside of standardizing with respect to the unit standard deviation. For this example, we have chosen $K = 2$, and so this result has a simple interpretation, namely set $\mu_1$ and $\mu_2$ to be the values of the sample means $\bar{x}_1$ and $\bar{x}_2$. In other words, we simply assign the points to be

$$
r_{nk} = \begin{cases} 1 & \text{if } k = \arg \min_j \| x_n - \mu_j \|^2 \\ 0 & \text{otherwise.} \end{cases}
$$

Finally, we have the $M$ steps. First we choose some initial values for the $\mu_i$. We can do this through an iterative procedure in which each iteration minimizes the objective function $J$ with respect to the $\mu_i$ and updating $r_{nk}$ fixed. This two-stage procedure is known as the $K$-means algorithm. For this reason, the procedure is repeated until convergence. We shall see that these two stages correspond respectively to the E (expectation) and M (maximization) steps of the EM algorithm, and to emphasize this we shall use the notation $E$ step and $M$ step in the context of the algorithm. The convergence of the $K$-means algorithm were studied by MacQueen (1967).
Old Faithful, Kmeans from Murphy
The progress of the K-means algorithm with $K=3$.

- **Top left:** The observations are shown.
- **Top center:** In Step 1 of the algorithm, each observation is randomly assigned to a cluster.
- **Top right:** In Step 2(a), the cluster centroids are computed. These are shown as large colored disks. Initially the centroids are almost completely overlapping because the initial cluster assignments were chosen at random.
- **Bottom left:** In Step 2(b), each observation is assigned to the nearest centroid.
- **Bottom center:** Step 2(a) is once again performed, leading to new cluster centroids.
- **Bottom right:** The results obtained after 10 iterations.
Different starting values

*K*-means clustering performed six times on the data from previous figure with $K = 3$, each time with a different random assignment of the observations in Step 1 of the *K*-means algorithm.

Above each plot is the value of the objective (4). Three different local optima were obtained, one of which resulted in a smaller value of the objective and provides better separation between the clusters.

Those labeled in red all achieved the same best solution, with an objective value of 235.8
Each pixel $x_i$ is represented by codebook of $K$ entries $\mu_k$.

$$\text{Encode}(x_i) = \text{argmin}_k ||x_i - \mu_k||$$

Consider $N=64k$ observations, of $D=1$ (b/w) dimension, $C=8$ bit.

$NC=513k$

$N\log_2 K + KC$ bits is needed. $K=4$ gives $128k$ a factor 4.
Mixtures of Gaussians (1)

Old Faithful geyser:
The time between eruptions has a bimodal distribution, with the mean interval being either 65 or 91 minutes, and is dependent on the length of the prior eruption. Within a margin of error of ±10 minutes, Old Faithful will erupt either 65 minutes after an eruption lasting less than $2 \frac{1}{2}$ minutes, or 91 minutes after an eruption lasting more than $2 \frac{1}{2}$ minutes.
Mixtures of Gaussians (2)

Combine simple models into a complex model:

\[
p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k)
\]

Component Mixing coefficient

\[
\forall k : \pi_k \geq 0 \quad \sum_{k=1}^{K} \pi_k = 1
\]

\[
\prod_{k=1}^{K} \pi_k \leq 1
\]
Mixtures of Gaussians (3)
• Gaussian mixture
  \[ p(x) = \sum_k^K \pi_k N(x; \mu_k, \Sigma_k) \]

• Latent variable:
  - Un-observed
  - Often hidden

\[ Z = \prod_k \pi_k \quad z_n \in \{0, 1\} \]

• Here \( p(z_k) = \pi_k \)

\[ p(z)p(x|z) \quad \text{N iid } \{x_n\} \text{ with latent } \{z_n\} \]
\[ p(x|z_k = 1) = N(x; \mu_k, \Sigma_k) \]
\[ p(x|z) = \prod N(x; \mu_n, \Sigma_n)^{z_{nk}} \]
\[ p(x, z) = p(x|z) p(z) \]
\[ p(x) = \sum_{z} p(x,z) = \sum_{z} p(x|z) p(z) = \sum_{z} \prod_{k} N(x; \mu_{nk}, \Sigma_{nk}) \]

Responsibilities

\[ \gamma(z_k) = p(z_k = 1|x) = \frac{p(x|z_k = 1) p(z_k = 1)}{\sum_{j} p(x|z_j = 1) p(z_j = 1)} = \frac{\prod_{k} N(x; \mu_{nk}, \Sigma_{nk})}{\sum_{j} \prod_{k} N(x; \mu_{nk}, \Sigma_{nk})} \]
Mixture of Gaussians

• Mixtures of Gaussians

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k). \]

• Expressed with latent variable \( z \)

\[ p(x) = \sum_{z} p(z)p(x|z) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \]

• Posterior probability: responsibility

\[
\gamma(z_k) \equiv p(z_k = 1|x) = \frac{p(z_k = 1)p(x|z_k = 1)}{K \sum_{j=1}^{K} p(z_j = 1)p(x|z_j = 1)} = \frac{\pi_k \mathcal{N}(x|\mu_k, \Sigma_k)}{K \sum_{j=1}^{K} \pi_j \mathcal{N}(x|\mu_j, \Sigma_j)}. \]

\( p(z)p(x|z) \) \n iid \( \{x_n\} \) with latent \( \{z_n\} \)
Max Likelihood

- \( p(x) = \sum_k^K \pi_k N(x; \mu_k, \Sigma_k) \)
- \( N \) observations \( X \)
- \( \ln[p(X|\pi, \mu, \Sigma)] = \prod_N \ln[\sum_k^K \pi_k N(x_n; \mu_k, \Sigma_k)] \)

\[
\frac{\delta L}{\delta \mu_k} = 0 = \sum \frac{N(x_n; \mu_k, \Sigma_k)}{\Sigma_k} + \lambda
\]

\[
\frac{\delta L}{\delta \Sigma_k} = 0 = \sum \frac{N(x_n; \mu_k, \Sigma_k)}{\Sigma_k} - 1
\]

N iid \( \{x_n\} \) with latent \( \{z_n\} \)
1. Initialize the means $\mu_k$, covariances $\Sigma_k$ and mixing coefficients $\pi_k$, and evaluate the initial value of the log likelihood.

2. **E step.** Evaluate the responsibilities using the current parameter values

$$
\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n | \mu_j, \Sigma_j)}.
$$

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

$$
\mu_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_n
$$

$$
\Sigma_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) (x_n - \mu_k^{\text{new}})(x_n - \mu_k^{\text{new}})^T
$$

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

where

$$
N_k = \sum_{n=1}^{N} \gamma(z_{nk}).
$$

4. Evaluate the log likelihood

$$
\ln p(X | \mu, \Sigma, \pi) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right\}
$$

and check for convergence of either the parameters or the log likelihood. If the convergence criterion is not satisfied return to step 2.
Given a joint distribution \( p(X, Z|\theta) \) over observed variables \( X \) and latent variables \( Z \), governed by parameters \( \theta \), the goal is to maximize the likelihood function \( p(X|\theta) \) with respect to \( \theta \).

1. Choose an initial setting for the parameters \( \theta^{\text{old}} \).

2. **E step** Evaluate \( p(Z|X, \theta^{\text{old}}) \).

3. **M step** Evaluate \( \theta^{\text{new}} \) given by

\[
\theta^{\text{new}} = \arg \max_{\theta} Q(\theta, \theta^{\text{old}}) \tag{9.32}
\]

where

\[
Q(\theta, \theta^{\text{old}}) = \sum_{Z} p(Z|X, \theta^{\text{old}}) \ln p(X, Z|\theta). \tag{9.33}
\]

4. Check for convergence of either the log likelihood or the parameter values. If the convergence criterion is not satisfied, then let

\[
\theta^{\text{old}} \leftarrow \theta^{\text{new}} \tag{9.34}
\]

and return to step 2.
EM in general

\[ p(X|\theta) = \sum_{Z} p(X, Z|\theta). \]  

\[ \ln p(X|\theta) = \mathcal{L}(q, \theta) + \text{KL}(q\|p) \]  

where we have defined

\[ \mathcal{L}(q, \theta) = \sum_{Z} q(Z) \ln \left\{ \frac{p(X, Z|\theta)}{q(Z)} \right\} \]  

\[ \text{KL}(q\|p) = -\sum_{Z} q(Z) \ln \left\{ \frac{p(Z|X, \theta)}{q(Z)} \right\}. \]

\[ \ln p(X, Z|\theta) = \ln p(Z|X, \theta) + \ln p(X|\theta) \]

\[ \mathcal{L}(q, \theta) = \sum_{Z} p(Z|X, \theta^{\text{old}}) \ln p(X, Z|\theta) - \sum_{Z} p(Z|X, \theta^{\text{old}}) \ln p(Z|X, \theta^{\text{old}}) \]

\[ = Q(\theta, \theta^{\text{old}}) + \text{const} \]
Gaussian Mixtures

(a) $L = 1$

(b) $L = 2$

(c) $L = 5$

(d) $L = 20$

(e) $L = 1$

(f) $L = 5$
Kmeans and EM (9.3.2)

\[ \Sigma_k = \epsilon I \]

\[ p(x|\mu_k, \Sigma_k) = \frac{1}{(2\pi \epsilon)^{1/2}} \exp \left\{ -\frac{1}{2\epsilon} \|x - \mu_k\|^2 \right\}. \quad (9.41) \]

Whereby the responsibilities

\[ \gamma(z_{nk}) = \frac{\pi_k \exp \left\{ -\|x_n - \mu_k\|^2 / 2\epsilon \right\}}{\sum_j \pi_j \exp \left\{ -\|x_n - \mu_j\|^2 / 2\epsilon \right\}}. \quad (9.42) \]

Becomes delta functions. \( \mu_k = x_n \)

And the EM means approach the Kmeans

\[ \mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_n \quad (9.17) \]
Hierarchical Clustering

- \( K \)-means clustering requires us to pre-specify the number of clusters \( K \). This can be a disadvantage (later we discuss strategies for choosing \( K \))
- *Hierarchical clustering* is an alternative approach which does not require that we commit to a particular choice of \( K \).
- In this section, we describe *bottom-up* or *agglomerative* clustering. This is the most common type of hierarchical clustering, and refers to the fact that a dendrogram is built starting from the leaves and combining clusters up to the trunk.
Hierarchical Clustering Algorithm

The approach in words:
- Start with each point in its own cluster.
- Identify the closest two clusters and merge them.
- Repeat.
- Ends when all points are in a single cluster.
45 observations generated in 2-dimensional space. In reality there are three distinct classes, shown in separate colors. However, we will treat these class labels as unknown and will seek to cluster the observations in order to discover the classes from the data.
45 observations generated in 2-dimensional space. In reality there are three distinct classes, shown in separate colors. However, we will treat these class labels as unknown and will seek to cluster the observations in order to discover the classes from the data.

• **Left:** Dendrogram obtained from hierarchically clustering the data from previous slide, with complete linkage and Euclidean distance.

• **Center:** The dendrogram from the left-hand panel, cut at a height of 9 (indicated by the dashed line). This cut results in two distinct clusters, shown in different colors.

• **Right:** The dendrogram from the left-hand panel, now cut at a height of 5. This cut results in three distinct clusters, shown in different colors. Note that the colors were not used in clustering, but are simply used for display purposes in this figure.