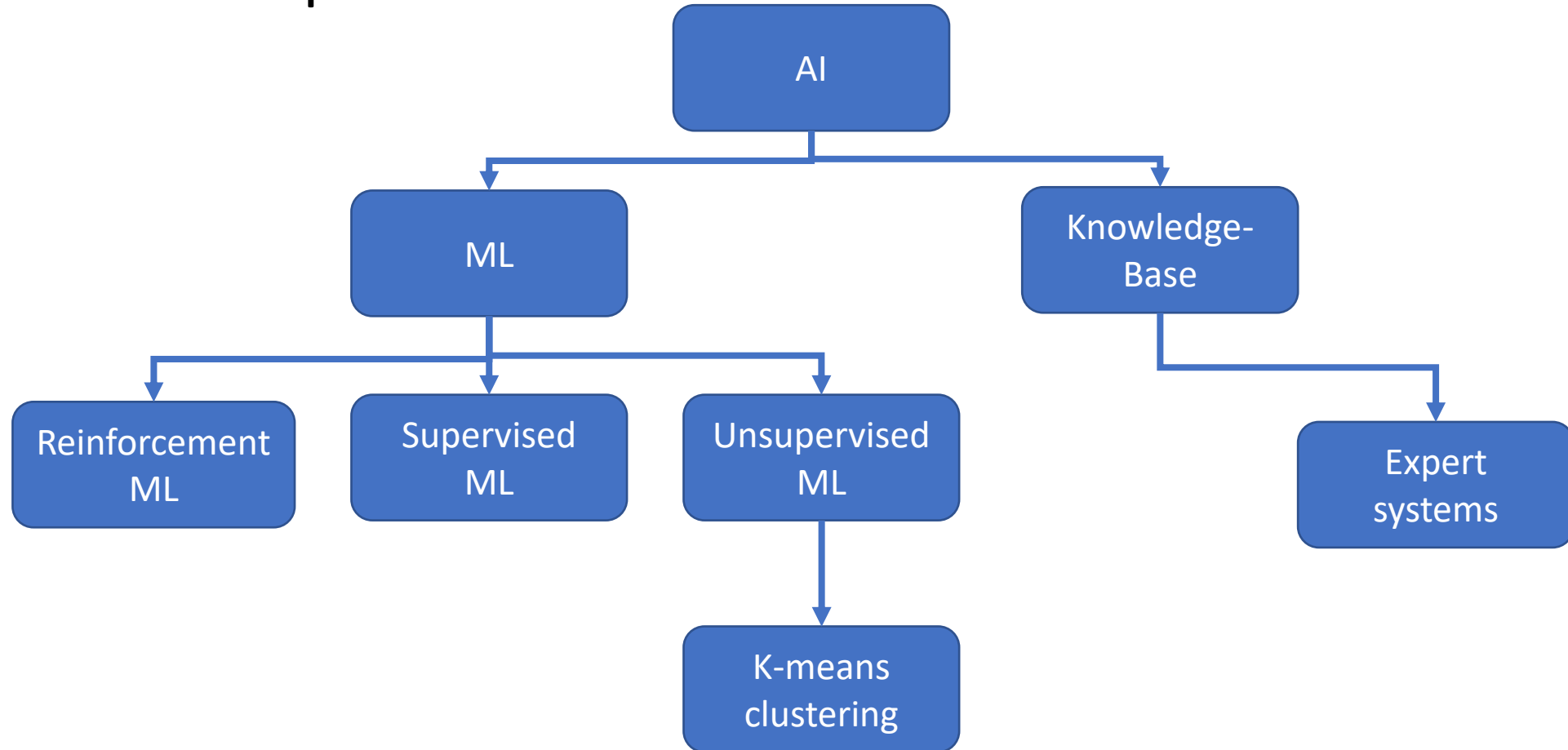
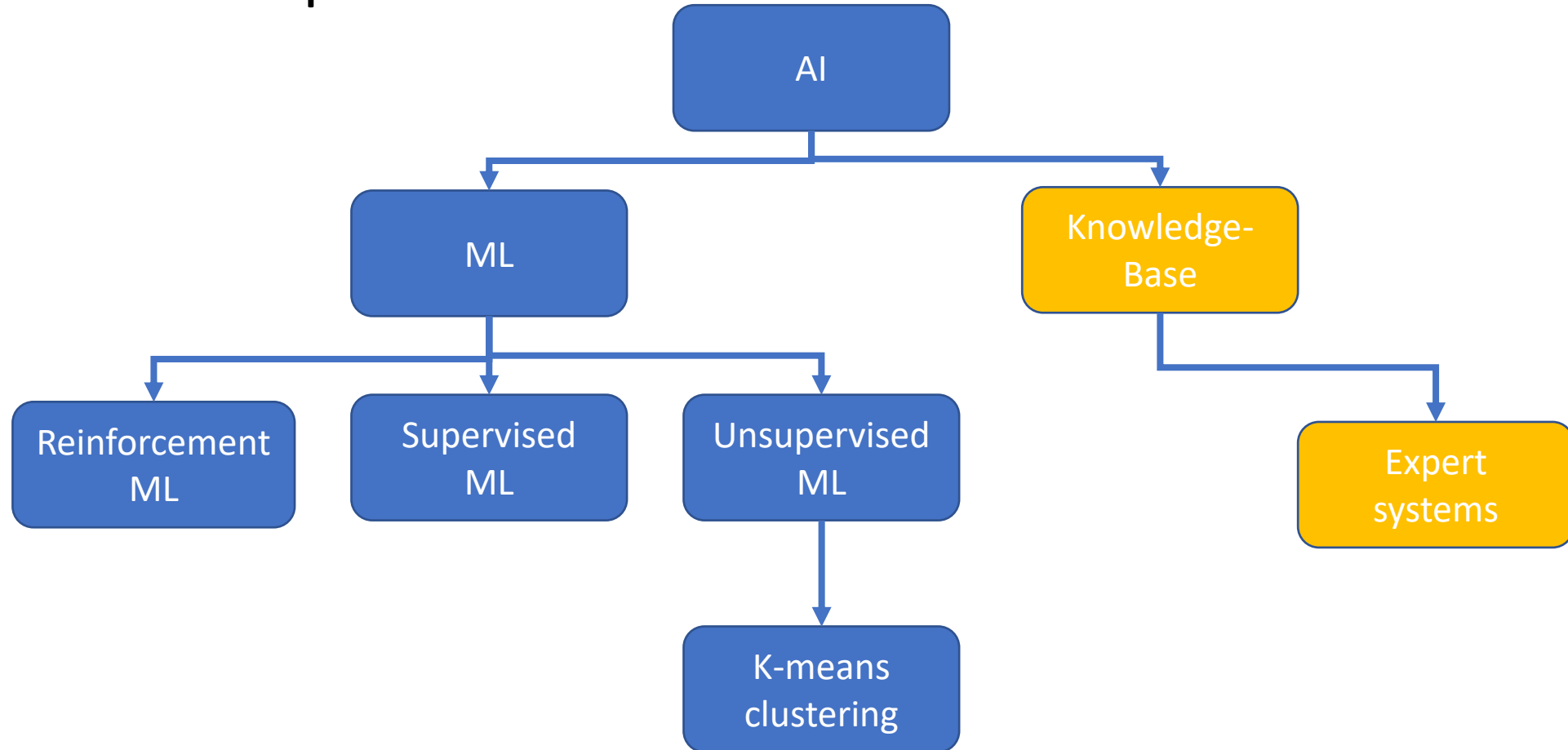


Supervised Machine learning

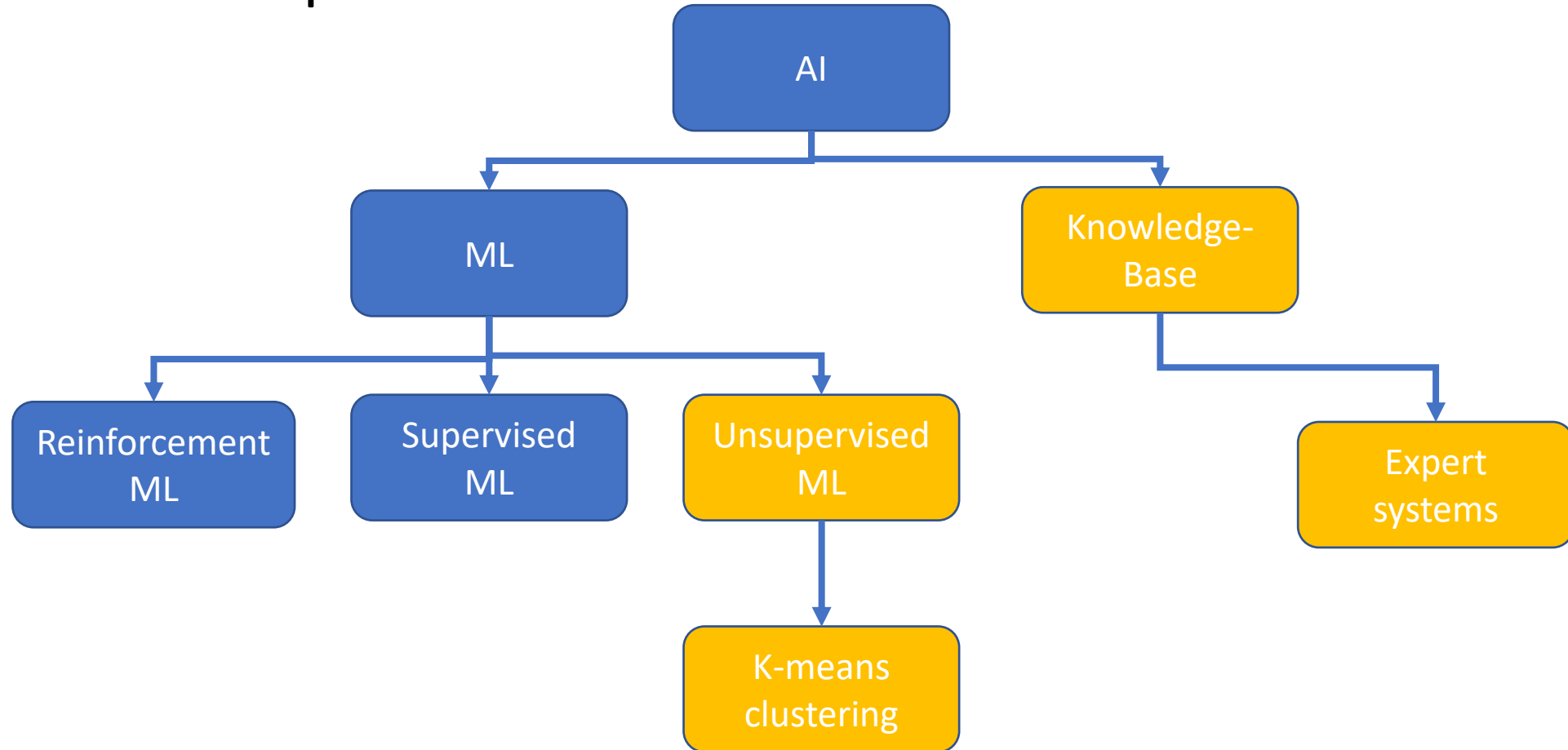
AI Techniques



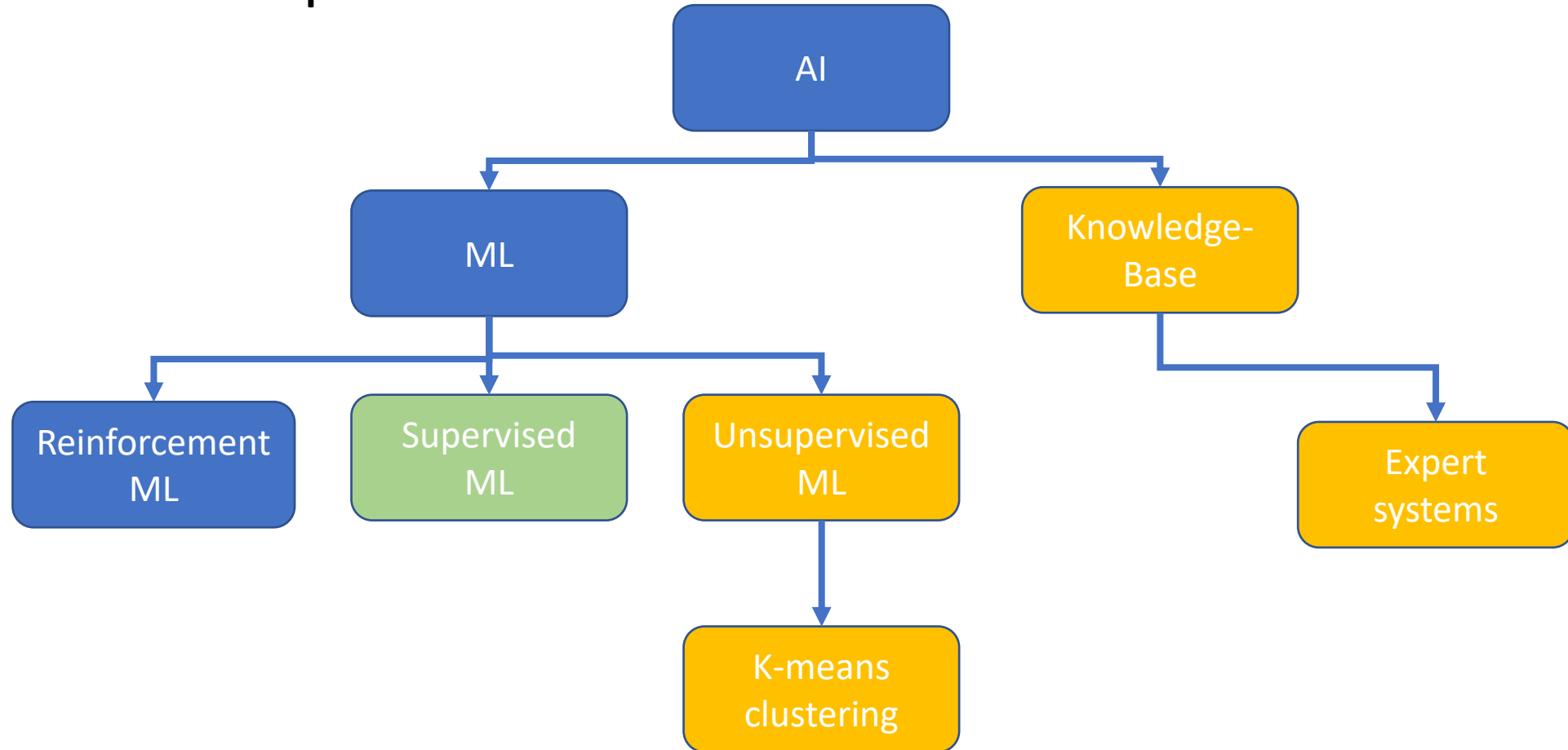
AI Techniques



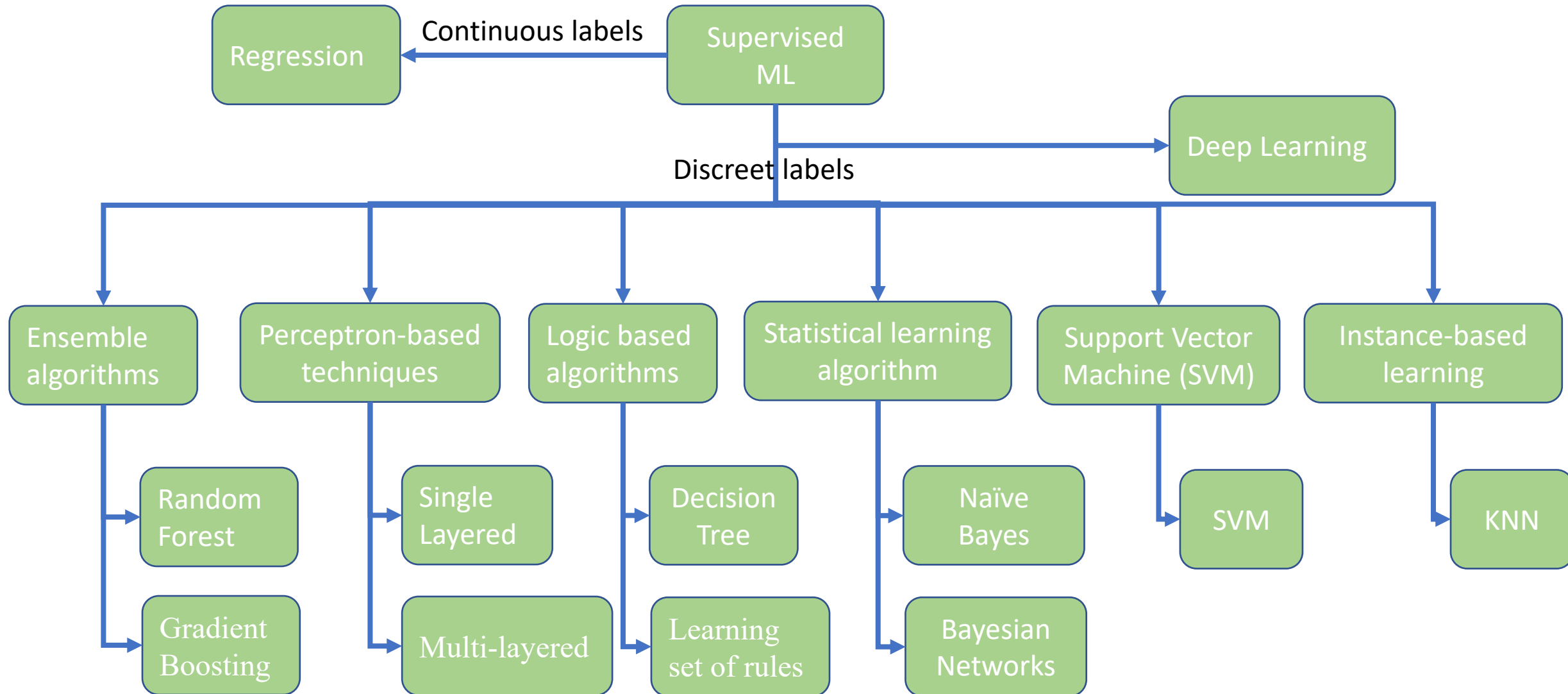
AI Techniques



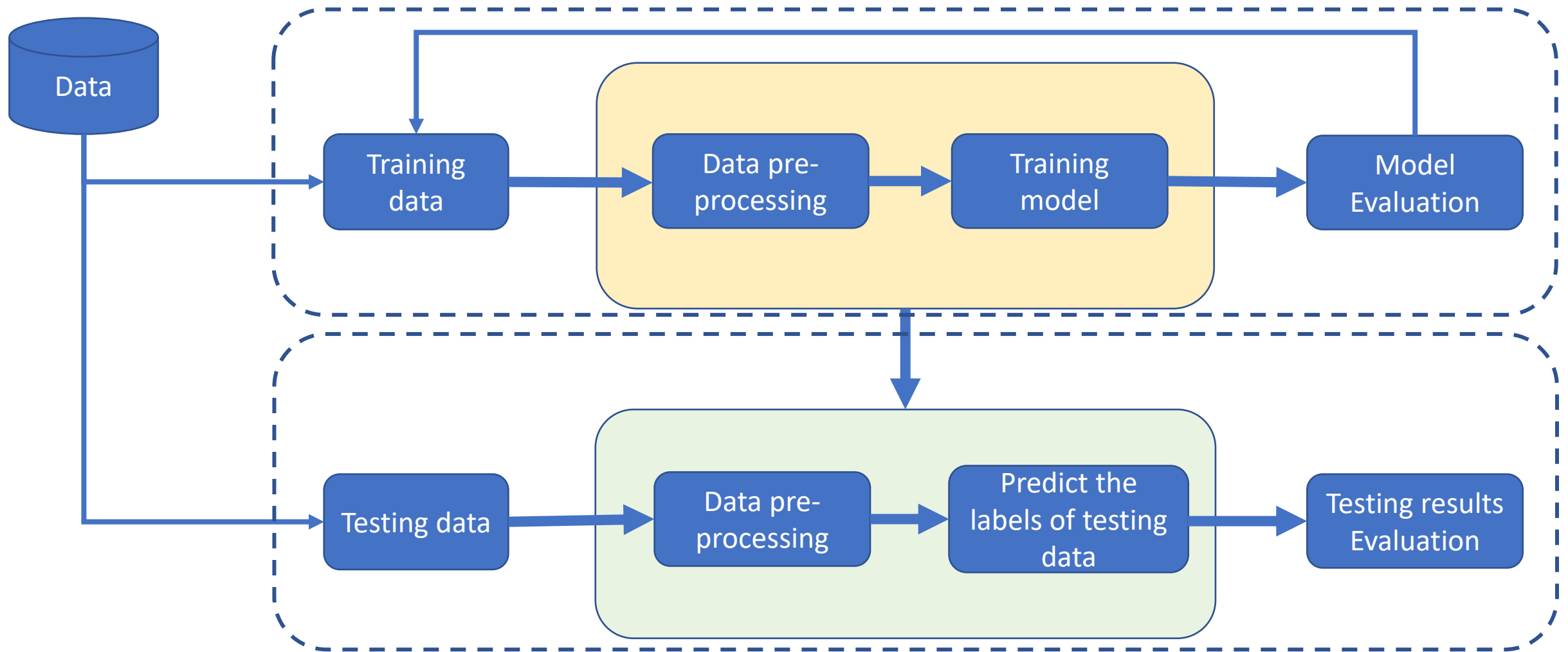
AI Techniques



Types of Supervised ML



Supervised ML pipeline



- Today (and for the next 5-6 weeks) we will focus on supervised ML.
- This means we will be given a dataset the we will divide into training and testing.
- The data consists of features (independent variable), and class label (target, ground truth, or output).

Task	Input	Labels
Object recognition	Image	Category
Image captioning	Image	Text
Machine translation	Text	Text
Speech recognition	Audio	Text
Sentiment analysis	Text	Category

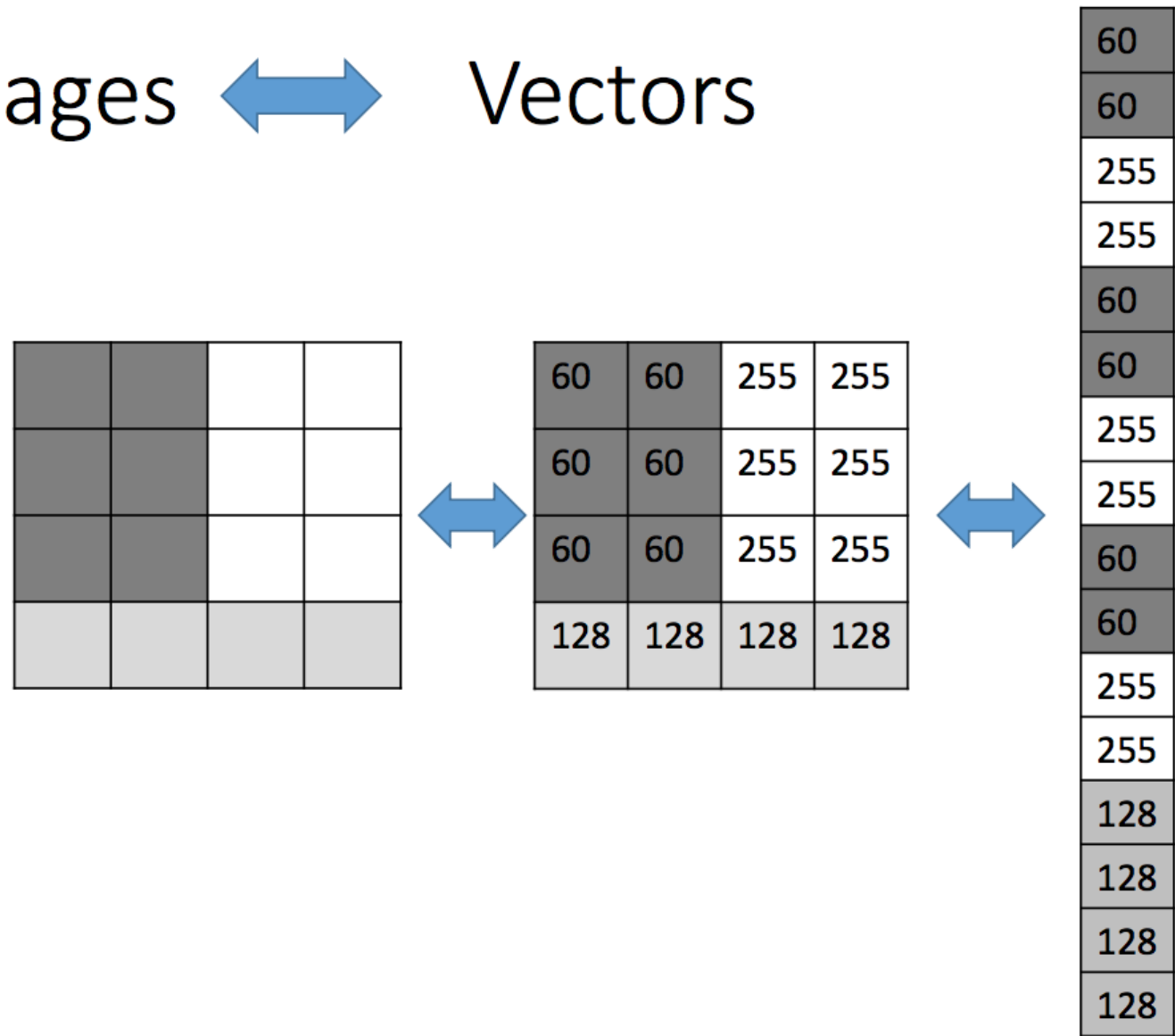
What an image looks like to the computer:



183	166	174	190	182	159	158	167	158	136	158	181	150	146	149	175	166	169	158	153	150	154	152	160	165	170	147	142	137	138	139
176	165	169	176	169	162	171	173	156	152	166	186	137	152	150	185	172	165	149	149	133	151	141	157	157	140	148	143	147	151	161
162	191	176	169	168	166	159	151	144	167	162	164	133	156	148	169	168	168	157	156	147	158	152	159	154	145	163	145	151	152	167
171	176	176	175	171	162	150	152	167	149	167	166	151	157	158	157	159	156	159	154	164	159	167	157	155	150	166	154	165	163	168
166	156	168	163	154	160	162	156	156	134	187	183	143	133	154	154	154	145	158	155	172	159	175	161	162	144	158	163	176	174	170
153	169	168	155	153	166	154	127	119	124	169	165	110	122	142	152	149	161	171	175	176	171	178	178	178	163	162	166	154	156	165
159	145	142	147	171	180	144	126	153	124	147	156	123	177	174	181	171	170	171	181	161	168	160	176	175	151	157	182	166	171	176
175	149	146	153	164	154	131	125	135	131	159	177	168	157	162	170	169	175	182	176	166	173	182	183	185	162	181	174	163	176	180
148	171	149	135	139	136	125	131	150	144	167	184	181	171	169	166	163	167	169	176	187	187	177	170	170	179	187	178	166	170	178
161	175	149	137	151	158	149	153	170	158	164	174	178	176	171	168	168	184	181	180	169	150	154	180	198	188	183	176	165	160	174
181	161	137	132	154	164	154	153	165	173	164	161	167	169	165	165	170	158	169	182	177	165	174	182	171	184	172	172	168	157	172
155	167	142	130	141	149	146	154	171	182	166	158	161	163	157	156	161	173	159	152	155	163	172	169	157	187	177	181	181	171	180
140	155	139	131	141	153	161	169	175	166	158	157	163	165	160	158	161	157	161	165	174	167	133	118	137	193	188	189	189	183	185
144	130	126	126	141	162	175	169	155	155	153	155	157	157	156	157	160	157	186	170	137	128	116	131	184	186	186	179	174	178	176
134	153	148	142	150	173	188	176	150	165	164	160	149	141	142	146	150	114	176	160	119	149	178	176	187	170	174	159	152	163	162
169	154	152	172	184	170	166	164	146	160	172	132	137	151	133	157	160	127	134	152	170	179	179	182	189	167	165	163	162	164	166
168	151	160	174	173	156	153	165	171	143	152	149	163	153	151	129	160	120	131	152	169	175	172	175	182	171	171	172	173	171	166
169	170	171	160	156	163	165	164	168	164	145	159	170	143	164	108	165	138	145	159	166	167	164	170	177	172	172	173	174	172	166
172	184	183	162	158	176	174	158	157	175	137	152	159	142	163	112	163	174	171	168	163	160	161	169	175	176	172	168	167	169	170
173	175	187	177	166	169	160	154	165	153	129	146	152	145	142	132	156	183	176	168	161	162	166	172	173	181	177	172	169	169	172
174	179	177	165	158	161	162	162	166	156	153	158	146	128	123	153	162	162	162	163	162	167	172	174	171	175	176	175	169	163	160
180	183	165	153	154	159	169	173	161	169	168	155	135	108	139	165	162	150	159	167	166	166	170	174	174	171	176	178	171	159	153
189	169	162	171	170	155	159	175	169	152	151	139	135	112	178	166	146	158	170	178	169	160	162	173	179	176	182	184	175	163	161
155	141	151	158	159	164	170	178	176	157	160	142	133	149	153	148	155	143	176	178	162	169	174	170	173	159	168	170	188	178	182
158	160	160	159	154	158	169	177	176	169	169	144	173	160	181	161	178	166	170	172	177	180	169	162	170	166	172	172	190	178	178
161	157	157	158	160	167	170	165	154	175	165	152	187	162	170	152	162	180	169	175	185	181	173	169	164	170	171	169	188	173	169
153	158	155	156	163	174	176	166	151	185	168	187	185	178	159	170	161	175	173	182	179	170	183	187	163	173	171	167	186	170	166
150	179	167	154	157	170	180	182	179	168	168	197	179	177	169	194	179	177	174	181	178	168	181	190	172	174	172	165	183	169	170
161	178	169	161	166	174	179	181	182	153	172	179	175	160	179	186	182	188	171	172	183	180	171	176	182	176	174	166	179	169	178
174	173	171	172	179	178	171	166	166	176	187	183	178	181	188	184	185	187	174	168	179	191	176	170	178	173	176	165	172	166	183
173	189	182	180	178	170	162	163	170	188	179	186	164	210	180	186	183	176	181	171	169	191	193	174	168	169	173	162	163	160	180

Input Vectors

Images $\longleftrightarrow$ Vectors



K-Nearest Neighbour (KNN)

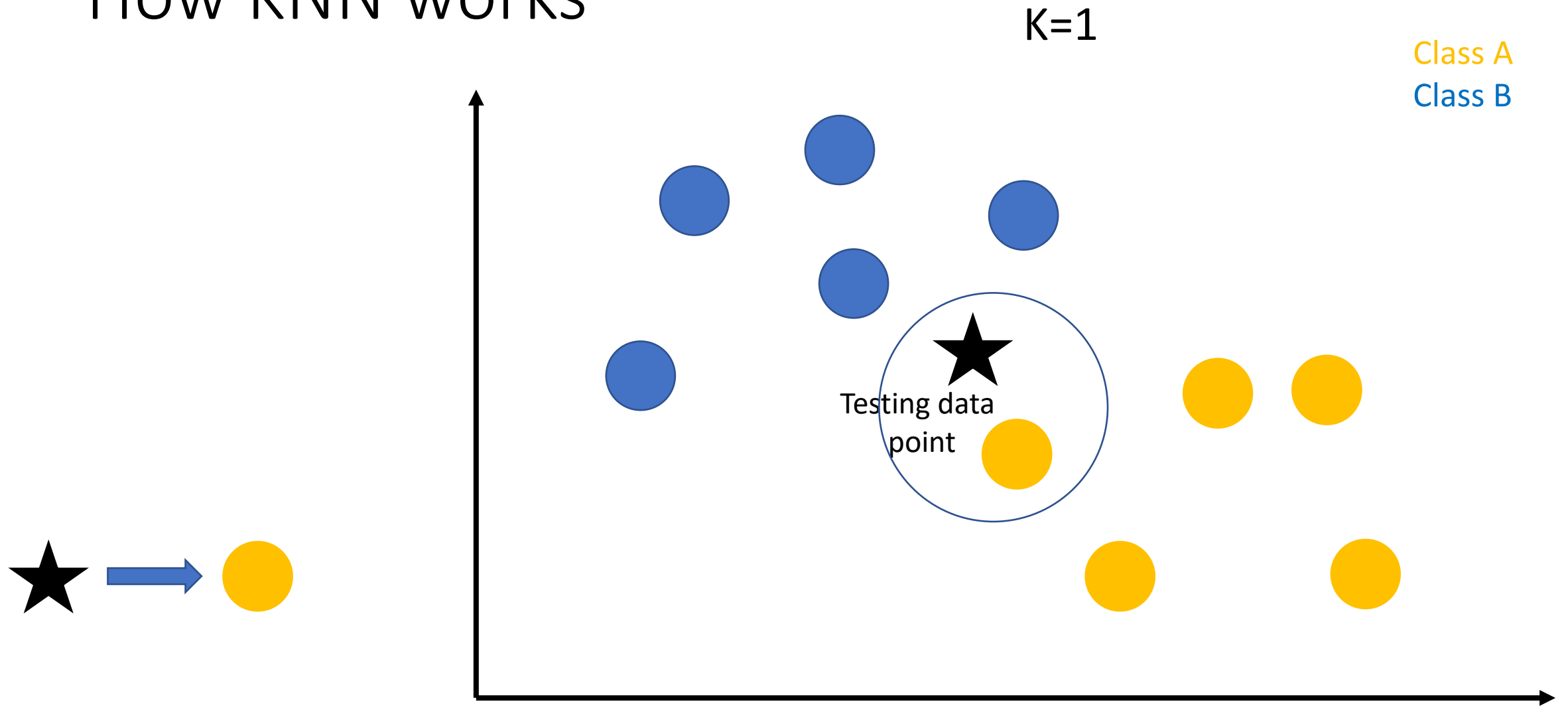
What is KNN

- KNN algorithm is one of the simplest classification algorithm
- non-parametric
 - it does not make any assumptions on the underlying data distribution
- lazy learning algorithm.
 - there is *no explicit training phase* or it is very minimal.
 - The training phase is fast .
 - KNN keeps all the training data.
- It uses a database in which the data points are separated into several classes to predict the classification of a new sample point.
- It classifies the new data point based on a similarity measure.

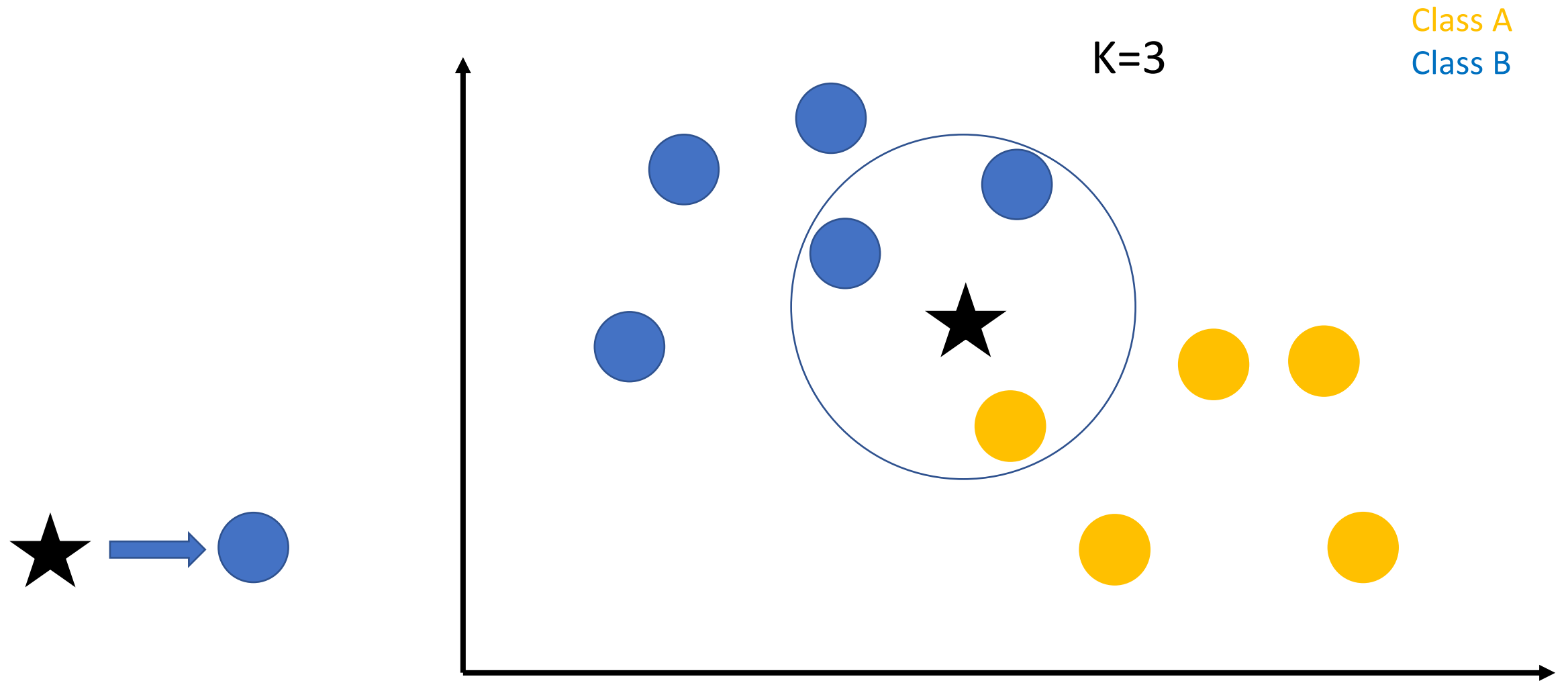
What is K in KNN ?

- K is the number of the nearest neighbour that the algorithm uses to find about the class of the new testing input data point.
- If $K=1$ then, the new testing data point is classified as the class of the nearest data point in the training data.
- If $K=3$, the label of the closest 3 datapoint in the training data is check and the majority class is assigned to the new testing data point.

How KNN works



How KNN works?



KNN Algorithm?

- Suppose we are given a new input vector (data point) x that we would like to classify.
- Measure the distance between the vector x and the vectors in the training data.
- Find the nearest input vectors to x in the training set. and copy its label.
- Using majority voting , assign the label that is appears the most in the K nearest.

Algorithm (kNN):

1. Find k examples $\{x^{(i)}, t^{(i)}\}$ closest to the test instance x
2. Classification output is majority class

$$y = \operatorname{argmax}_t \sum_{i=1}^k \mathbb{I}\{t = t^{(i)}\}$$

How do we choose the value of K?

- KNN is based on feature similarity.
- Choosing the right value of K is a process called parameter tuning.
- The value of K is important for getting the best performance.
- When the data is noisy, which means the data contains mislabelled data. The value of K should be big enough to avoid any bias of the noise.
- If the value of K is too big, the processing will take long time.

Trade-off in choosing k?

❑ Small k

- Good at capturing fine-grained patterns
- It can be sensitive to the noise in the training data

❑ Large k

- Makes stable predictions by using majority vote.
- May fail to capture important patterns.

❑ Balancing k:

- The optimal choice of k depends on the number of data points n.
- $k = n^{\frac{2}{2+d}}$, where d is the number of features.

When can KNN algorithm be used?

Data

Data is noise free

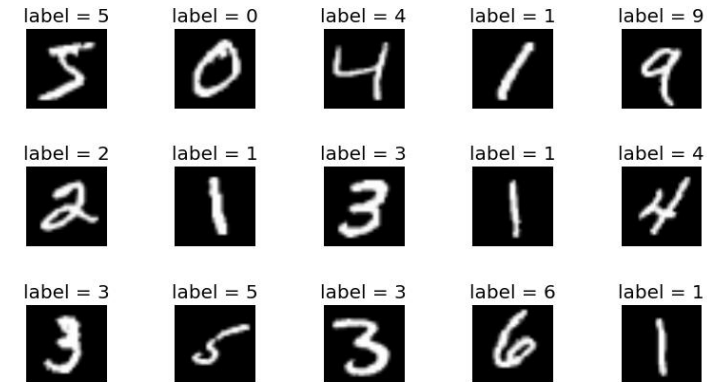
Weight (in Kgs)	Hight (in cms)	Jumper size
158	58	M
158	60	M
162	59	50
163	60	Hi
160	58	Fourty
165	66	L

Data is small



KNN is a lazy ML classifier i.e.
It doesn't learn a discriminative
function from the data

Data is labelled.



How KNN measure the closeness ?

- Euclidean Distance

The Euclidean distance, also known as L2 norm, between two data point x_i and x_j is

defined as: $D(X_i, X_j) = \sqrt{\sum_{l=1}^N (x_{il} - x_{jl})^2}$

- Where x_{il} and x_{jl} represent the l^{th} dimension of X_i and X_j respectively.
- If one of the input attributes has a relatively large range, then it can overcome the other attributes.

- Manhattan Distance

Manhattan distance also known as a city-block, rectilinear or L1 distance, is mathematically defined as:

$$D(X_i, X_j) = \sum_{l=1}^N |x_{il} - x_{jl}|$$

- The clusters formed using Manhattan distance tend to form rectangular shaped clusters.
- Its advantage over Euclidean distance is the reduction in computational time
- The disadvantage is that it depends upon the rotation of the coordinate system.

- Hamming Distance

- Calculates the distance between two vectors.
- It measures the minimum number of *substitutions* required to change one array into the other, or the minimum number of *errors* that could have transformed one string into the other.
- It works well with binary data.

- Cosine Distance (Similarity)

- It is a measure of similarity between two non-zero vectors.
- It works well with documents similarity.

$$D = \frac{\sum_{l=1}^N x_{jl} x_{il}}{\sqrt{\sum_{l=1}^N x_{il}^2} \sqrt{\sum_{l=1}^N x_{jl}^2}}$$

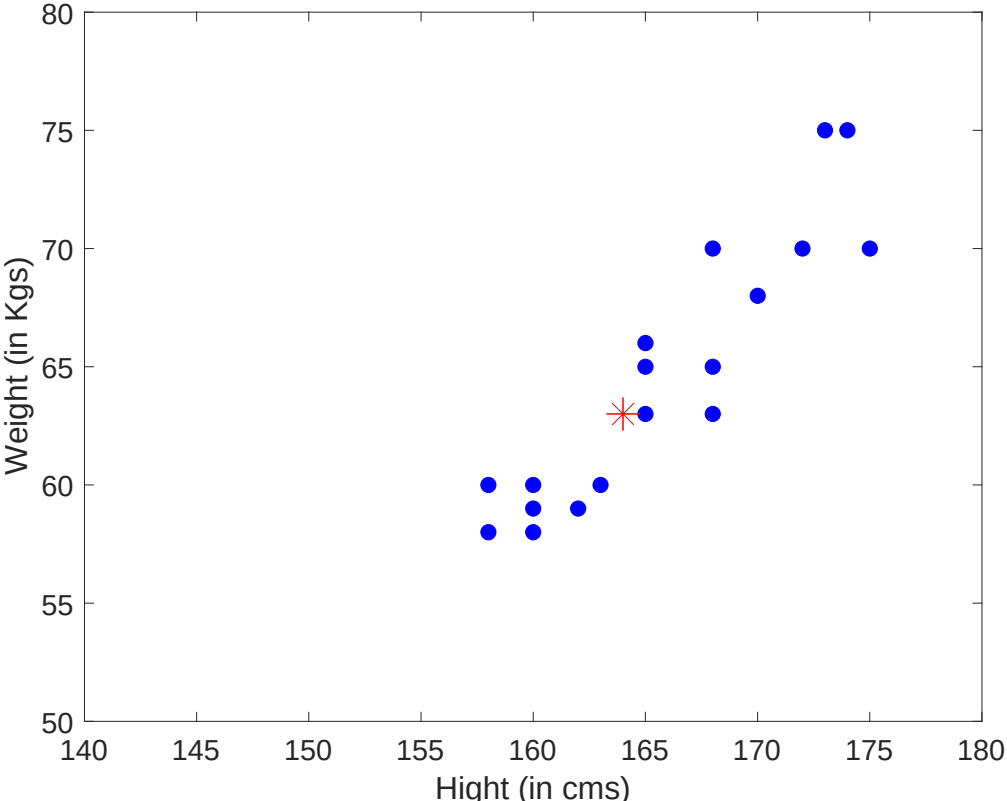
Example:

Suppose we have the dataset that is shown below. It contains the weight and hight of people as features and the jumper size is the class label. Every row vector is a data point that represent one person.

Hight (in cms)	Weight (in Kgs)	Jumper size
158	58	M
158	60	M
162	59	M
163	60	M
160	59	M
160	60	M
160	65	M
165	65	L
165	68	L
165	70	L
168	63	L
168	65	L
168	70	L
170	68	L
172	70	L
175	70	L
173	75	L
174	75	L

Suppose we have new query, we would like to know the size of this new person which we don't have in the data:

Hight	Weight	Size
164	63	?



K= 3

Hight (in cms)	Weight (in Kgs)	Jumper size	Euclidean distance
158	58	M	7.810
158	60	M	6.708
162	59	M	4.472
163	60	M	3.162
160	59	M	5.657
160	60	M	5.000
160	65	M	4.472
165	65	L	2.236
165	68	L	5.099
165	70	L	7.071
168	63	L	4.000
168	65	L	4.472
168	70	L	8.062
170	68	L	7.810
172	70	L	10.630
175	70	L	13.038
173	75	L	15.000
174	75	L	15.620

Matlab Session

Dataset for this session

- **Breast Cancer Wisconsin (Diagnostic) Data Set**
- **Attribute Information:**
- 1) ID number
2) Diagnosis (M = malignant, B = benign)
3-32)

Ten real-valued features are computed for each cell nucleus:

- a) radius (mean of distances from center to points on the perimeter)
- b) texture (standard deviation of gray-scale values)
- c) perimeter
- d) area
- e) smoothness (local variation in radius lengths)
- f) compactness ($\text{perimeter}^2 / \text{area} - 1.0$)
- g) concavity (severity of concave portions of the contour)
- h) concave points (number of concave portions of the contour)
- i) symmetry
- j) fractal dimension ("coastline approximation" - 1)