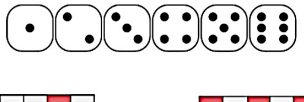
Think Bayesian

There are few rules to think Bayesian:

1- Use prior knowledge
2- Choose answer that explain the observation the most
3- Avoid making extra assumptions, this principle is known as outcome racer.

Review of Probability

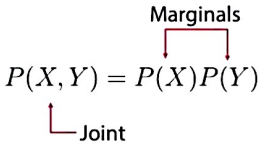
When tossing, probability of toss showing a specific number is 1/6 or the probability of toss showing odd number is ½.



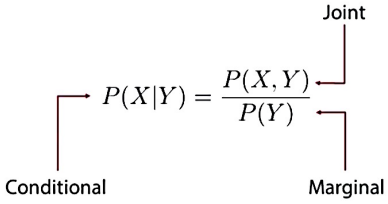
$P(threw\ 5) = \frac{1}{6}$ $P(threw\ odd) = \frac{1}{2}$

We have two variables: discrete or continuous. The discrete distribution is defined by probability mass function (PMS). It maps a number to specific point, refers as probability. The continuous distribution is defined by probability density function(PDF). It assigns non-negative value to a range of numbers.

The tow variable is defined as independent if their joint probability is equal to the product of their marginal.

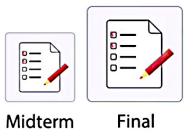


Conditional probability is the probability of x when y happens is calculated as below. It is defined by probability of joint over probability of marginal.



For example, imagine we have the probability of student pass the midterm is 0.4 and probability of student pass both midterm and final is 0.25 and we like to know the probability of student passing final while they already have passed the midterm. On the right, it shows how to calculate the probability of it.

$P(M) = 0.4$
$P(M\ \&\ F) = 0.25$



$$P(F|M) = \frac{P(M\ \&\ F)}{P(M)} = \frac{0.25}{0.4} = 0.625$$

Chain rule: the joint probability of x and y is calculated by product of x given by and probability of y. It can be generalized to joint probability of 3 variables as well.

$$P(X,Y) = P(X|Y)P(Y)$$
$$P(X,Y,Z) = P(X|Y,Z)P(Y|Z)P(Z)$$
$$P(X_1,\ldots,X_N) = \prod_{i=1}^{N} P(X_i|X_1,\ldots,X_{i-1})$$

Sum rule: It is used when we want to find out the marginal distribution p(x) and we only know the probability of p (x, y).

Marginalization

$$p(X) = \int_{-\infty}^{\infty} p(X,Y)dY$$

Bayes Theorem: we would like to get the probability of x given theta. X is observation and theta is the parameter. For example, neural network is like parameter and images are observations. It is calculated as shown on the right. In this formula we have prior, meaning what's our prior knowledge we know about these parameters. Likelihood shows how well the parameters explains the data. The posterior is the probability of the parameters after we observe the data.

θ — parameters
X — observations

Posterior Likelihood Prior

$$P(\theta|X) = \frac{P(X,\theta)}{P(X)} = \frac{P(X|\theta)P(\theta)}{P(X)}$$

Evidence

Naïve Bayes Approach vs Frequentists

There are two main approaches in statistics: normal approach and Bayesian. The main different is that frequencies treat the objective and Bayesian treat it as subjective. Let's explain it more deeply: we have a coin and we toss it. The frequentist says it's half-half probability for each side, random outcome. On the other hand, the Bayesian says that if we know the velocity of coin and how it's tossed, we can predict the outcome, therefore the outcome is not random.

Another difference between the Bayesian and frequencies are how they treat the parameters of data. Frequencies say that parameters are fixed and data is random and they look for optimal point. On the other hand, the Bayesian says parameters are random and the data is fixed. It makes sense, because when we are training the data, we know the data and it's not random anymore.

Bayesian works for arbitrary number of points, while the frequencies works only when the number of data points is much greater than number of parameters. It's not a big problem for big data. However, it's problematic in neural network that there are millions of parameters, while number of data points are just thousands.

Another difference between the Bayesian and frequentists is how they train the model. The frequentists train model using the maximum likehood principle, meaning the try to find the

parameter theta that maximizes the likehood. On the other hand, the Bayesian try to compute the posteriors, the probability of the parameters given the data, using Bayes formula. For example, is classification problem, they train and predict as follow:

Training:

$$P(\theta|X_{\text{tr}}, y_{\text{tr}}) = \frac{P(y_{\text{tr}}|X_{\text{tr}}, \theta)P(\theta)}{P(y_{\text{tr}}|X_{\text{tr}})}$$

Prediction:

$$P(y_{\text{ts}}|X_{\text{ts}}, X_{\text{tr}}, y_{\text{tr}}) = \int P(y_{\text{ts}}|X_{\text{ts}}, \theta)P(\theta|X_{\text{tr}}, y_{\text{tr}})d\theta$$

The Bayesian also can be used for regularization as well. We can treat the prior as regularizer.

Regularizer

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)}$$

Bayesian is great for online learning too. We update our parameters and then use the posterior as a prior to the next set.

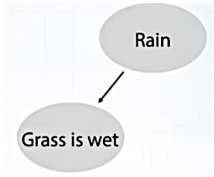
New prior — Likelihood — Prior

$$P_k(\theta) = P(\theta|x_k) = \frac{P(x_k|\theta)P_{k-1}(\theta)}{P(x_k)}$$

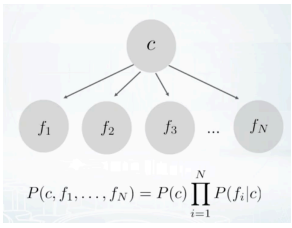
Posterior

Define a Probabilistic Model

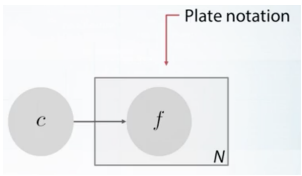
Let's do it for Bayesian network. It is a graph that nodes are variables and edges are the direct impact. For example, in image below we have two invariables, rain and grass is wet. The edge is showing if it is raining, then the grass gets wet. The network is better explained at Wikipedia.



The Naïve Bayes classifier can be a little complex, the joint probability is defined as the probability of class times the product over the features, shown in the image on the right-hand side.

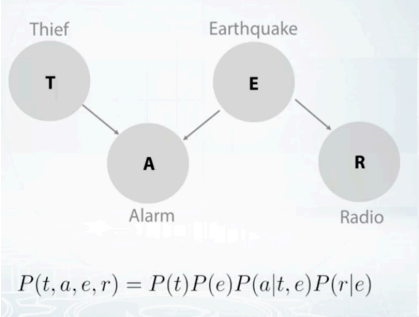


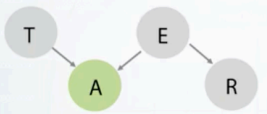
The better way of showing the graph is called plate notation, since some of subgraph might be identical.



Thief and Alarm Example

Sometimes, alarm house alarms might give false alarm for example in case of earthquake. Also, if the earthquake is strong enough, we get a notification from radio. Here is the Bayesian graph for it. We would like to know the general probability of four variables. The table below shows the prior knowledge and the probability of alarm goes off in different cases.



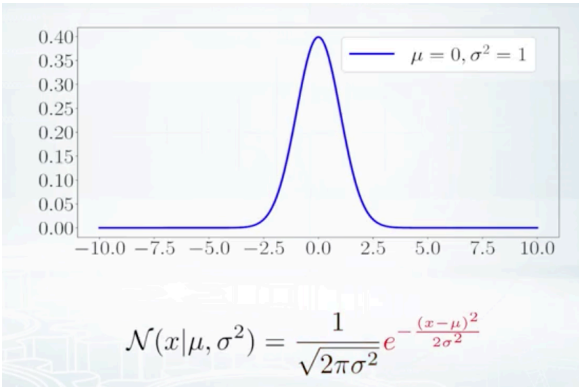


Priors	
$P(T = 1)$	10^{-3}
$P(E = 1)$	10^{-2}

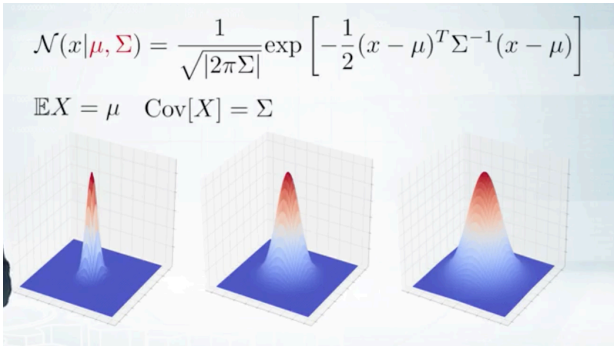
$P(A = 1 \mid T, E)$	E = 0	E = 1
T = 0	0	1/10
T = 1	1	1

Linear Regression (Naïve Bayes)

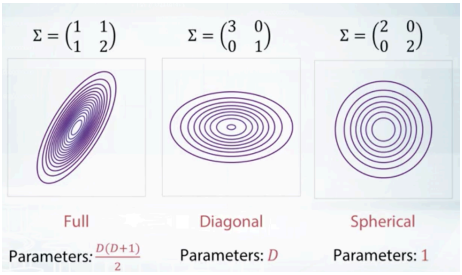
Before starting, let's get remind of univariate and multivariate. The univariate normal distribution is a Gaussian density distribution, when mean of random variables of 0 and variance of 1. By changing the mean and variance, we get different density function.



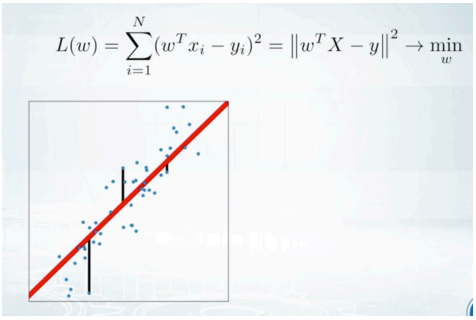
In multivariate, we have similar density function to univariate. The mean is a vector and variance is a covariance. This assures that the area under density function is equal to 1. The maximum value occurs in mu.



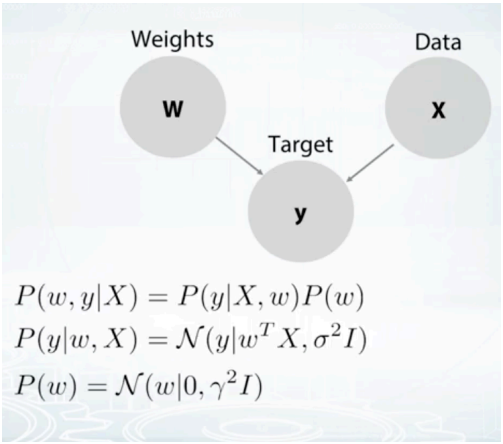
Sigma is a symmetry variable and has a bunch of parameters in the matrix. In image below shows how the parameters are approximate.



In linear regression, we like to fit a line into our parameters shown in red line. In linear regression, the least square value between the line and point should be minimal.



Linear regression using Bayesian model will be as follow. Given a data and weights, we find a probability of target. By using this model, we end up with L2 regularization in linear regression.



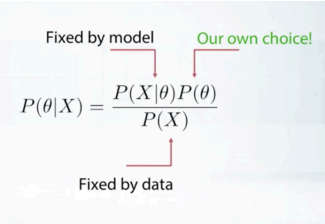
Analytical Inference

The Bayesian formula is the likelihood times prior divided by evidence. What's exactly the evidence? Imagine we are training a neural network to play games. X is the images of game scream and theta is a network parameter. By knowing the p(x), the evidence we can generate a new game like frames.

Naïve Bayes is easy to implement; however, it has some cons such as:

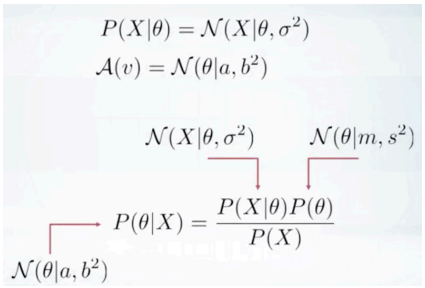
- Can't use it as prior
- It finds untypical points
- Can't compute credible intervals
- Not variant to reparametrization

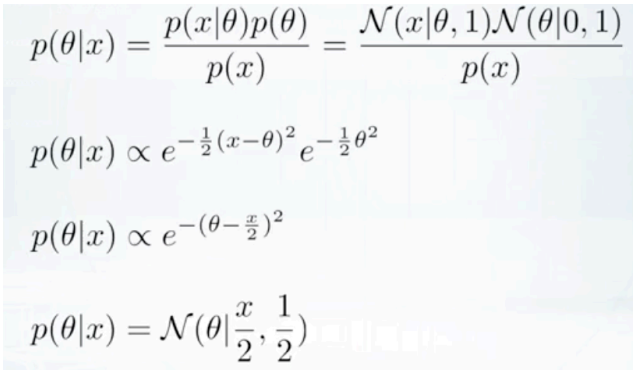
Conjugate Distributions



In Bayes formula, the evidence is fixed by data, prior is our choice and the likelihood is fixed by our model.

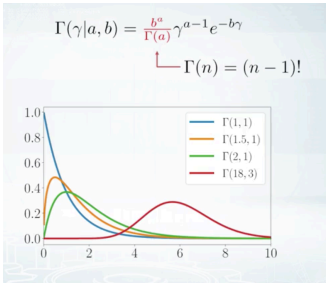
The prior is the conjugate of posterior if they are from same distribution, meaning having same mean and variance. By knowing the variance and mean of the prior and likelihood, we can get the value of mean and variance for the posterior. More information about conjugate distribution can be found at Wiki.

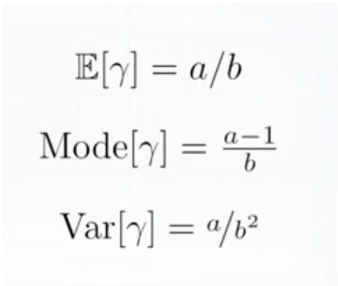




Gamma Distribution

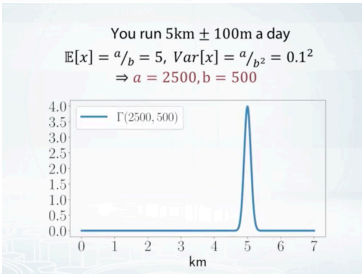
It is defined by two positive parameters of a and b. Gamma is a positive value as well.





Here is an example of runner, by knowing the var and mean, we can calculate the gamma function.

The gamma function is conjugated to be normal with respect to the precision. Precision is inverse of variance. Low precision means high variance. High precision means low variance. We can avoid computing the evidence simply by choosing the conjugate prior.

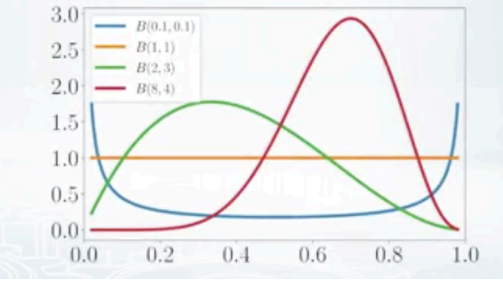


Beta Distribution - Bernoulli

It has some normalization constant and some parameters with power a-1 and b-1.

$$B(x|a,b) = \frac{1}{B(a,b)} x^{a-1}(1-x)^{b-1}$$

$$x \in [0,1], \quad a,b > 0$$



$$\mathbb{E}x = \frac{a}{a+b}$$

$$\text{Mode}[x] = \frac{a-1}{a+b-2}$$

$$\text{Var}[x] = \frac{ab}{(a+b)^2(a+b-1)}$$

Beta distribution is conjugated with Bernoulli.

Latent Variable Models & Expectation Maximization

Latent (hidden) variable is a variable that we never observe. height and weight can be measured directly but some other variables such as incidences.

Imagine the company is hiring new employees, we have data from each employee. Also, we have some data about their previous work experience and we would like to predict which of the candidates do better on onsite interview and bring them for interview to eliminates the cost. Here is a simple regression problem. However, it's not great to use standard regression methods such as linear regression or neural network. ***Because we may want to quantify uncertainty in our predictions***.

	High school grade	University grade	IQ score	Phone Interview	Onsite interview
John	4.0	4.0	120	3/4	?
Helen	3.7	3.6	N/A	4/4	?
Jack	3.2	N/A	112	2/4	?
Emma	2.9	3.2	N/A	3/4	?
	High school grade	University grade	IQ score	Phone Interview	Onsite interview
Sophia	3.5	3.6	N/A	4/4	85/100
...					

The missing values in data and wanting to quantify the uncertainty bring us the need to probabilistic modelling of the data. In this case, we need to find the relation between random variables. In our case, it appears everything is connected to everything. This model failed to capture the structure of our probabilistic model. Basically, this model is very flexible with little structure that we possibly can have.

Next step is to assign probability to each possible combination of our feature.. We can come up with a method to do this. For example, using equation below:

$$p(x_1, x_2, x_3, x_4, x_5) = \frac{exp(-w^\intercal x)}{Z}$$

the normalization constant above is sum of all possible configurations (gigantic sum, billions of sum). This means that training will be so impractical. The solution is to introduce a new variable that we don't have in our model which is called *intelligence*. So, we can assume that each employee has an internal and hidden property of him. The connection between the intelligence and variables are non-deterministic, but direct causation. Using intelligence, the complexity of model has been reduced substantially. We may now write the probabilistic model using the rule of sum of probabilities. It is the sum of all possible configuration given the intelligence times the prior probabilities. These are conditional probabilities. Here we reduce the complexity without changing the flexibility.

$$p(x_1, x_2, x_3, x_4, x_5) = \sum_{I=1}^{100} p(x_1, x_2, x_3, x_4, x_5 \mid I)\, p(I)$$

In summary, latent variables:

Pros:

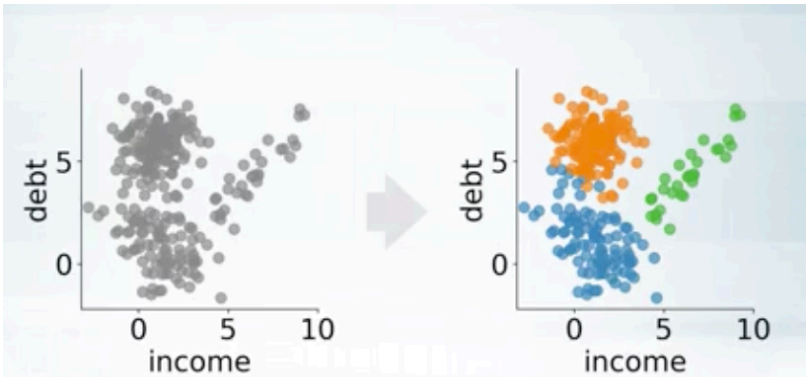
- Simpler model (less edges)
- Fewer parameters
- Sometimes latent variables are meaningful

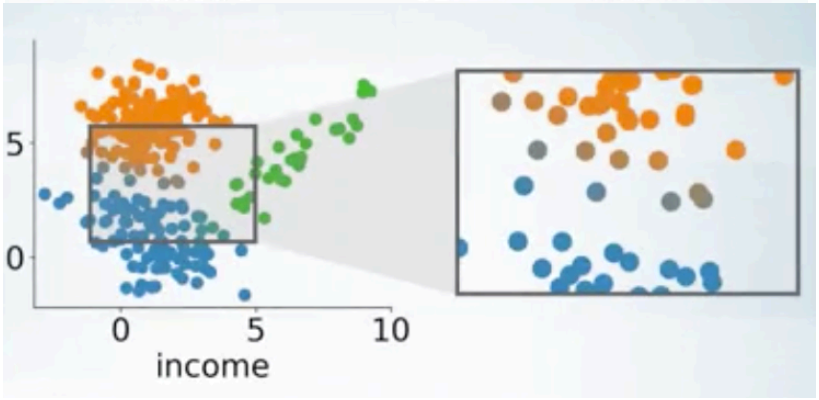
Cons

- Harder to work with (rely on so much math)

Probabilistic Clustering

Clustering is when we have unlabeled data. Usually clustering is done in a hard way (hard clustering) as shown in top picture below.

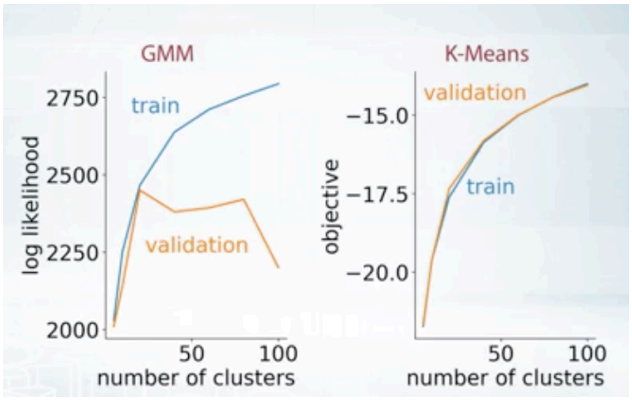




In soft clustering (shown in bottom picture above) is when we assign each data point from probability distribution over cluster. The blue and orange line are 100% probabilities, but in middle point is 60% belongs to the blue data points for example or 40% orange data points.

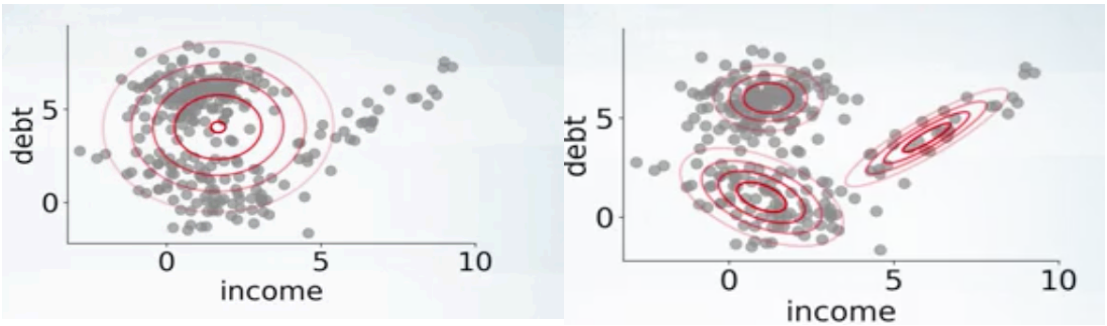
There are several reasons to use soft clustering such as: handling the missing data naturally, to tune the hyper parameters, building a generative model of data.

Charts below shows the performance of soft vs hard clustering technique. The higher value of log likehood the better it is. The hard clustering doesn't show what is the best number of clusters to choose. The soft clustering makes it easy to choose the number of clustering.



Gaussian Mixture Model

GMM is a way to model our data probabilistically. Gaussian model has to model all data points as one big circle or maybe ellipse. In the figure below, the center of Gaussian is falling into the between clusters. The center of Gaussian has to assign the hyper growth in it. The solution is to use GMM, using three Gaussians for our model. Each of them explains one clusters of data points. In this case more density is located at the center of each Gaussian distribution. The density of each data point is equal to the weighted sum of three Gaussian densities.



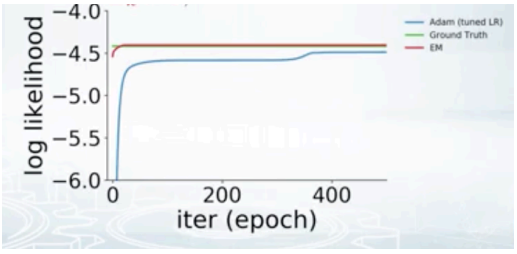
$$p(x \mid \theta) = \pi_1 \mathcal{N}(x|\mu_1, \Sigma_1) + \pi_2 \mathcal{N}(x|\mu_2, \Sigma_2) + \pi_3 \mathcal{N}(x|\mu_3, \Sigma_3)$$

$$\theta = \{\pi_1, \pi_2, \pi_3, \mu_1, \mu_2, \mu_3, \Sigma_1, \Sigma_2, \Sigma_3\}$$

GMM is more flexible than Gaussian. The downside is number of parameter we deal with in GMM. We find the parameters of GMM by mean and std, using maximum likelihood estimation. In ML, we assume that data has n independent data points. Therefor the likelihood is equal to:

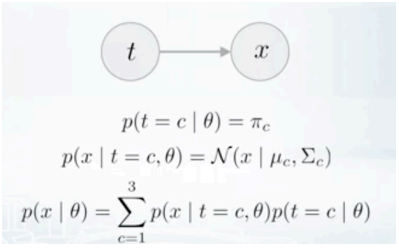
$$\max_{\theta} \prod_{i=1}^{N} p(x_i \mid \theta) = \prod_{i=1}^{N} (\pi_1 \mathcal{N}(x_i \mid \mu_1, \Sigma_1) + \ldots)$$
$$\text{subject to} \quad \pi_1 + \pi_2 + \pi_3 = 1; \; \pi_k \geq 0; \; k = 1, 2, 3.$$

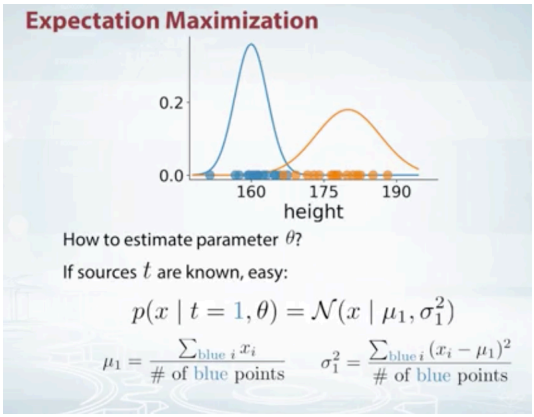
Now we may use our model (NN or any other thing) to optimize the parameters. In Gaussian distribution, we need to calculate the inverse of covariance. So, the covariance matrix (sigma) can't be arbitrary. The set of valid covariance matrices is called positive, which is a very hard constraint. Another way of constraining the covariance matrix is to use a diagonal matrix which corresponds to the elliptical Gaussian that can't be rotated. This is easier to constraint. In image below you can see the log likelihood (we try to maximize it) vs iteration. Adam optimization is doing a great job in compare to the Ground Truth. The expectation model works better than both.



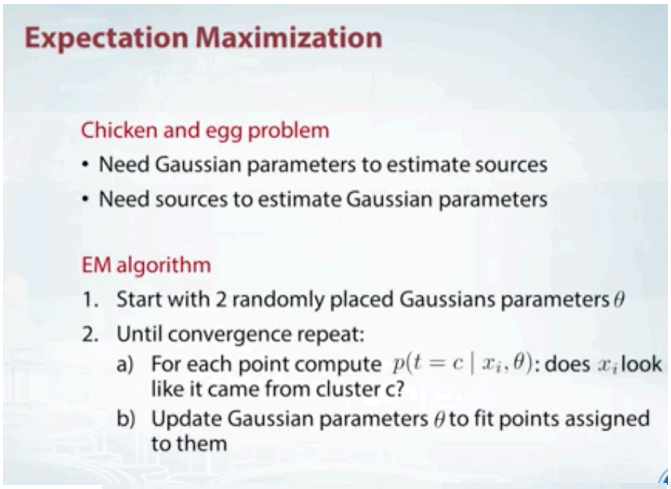
Training GMM & Expectation algorithm

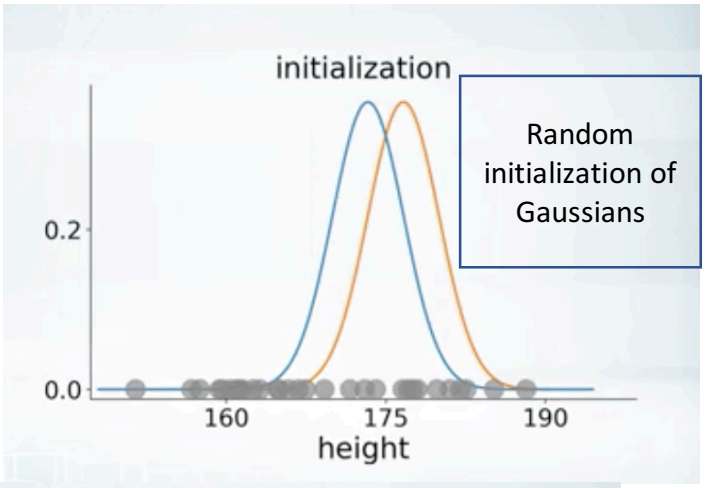
Let's introduce a latent variable to make the reasoning about our model easier. We can say each data point is using some information from latent variable T and it causes x to happen. Variable T gets 3 radiuses, one two or three which that data point came from. For each data points, we don't know to which Gaussian they belong to, so it's latent variable. We don't observe it either in testing nor in testing. Maybe later we find the distribution on the latent variable even the data. T has exact weight of our Gaussians, and T equals to some cluster number. Using latent variable will give same result as using original model.

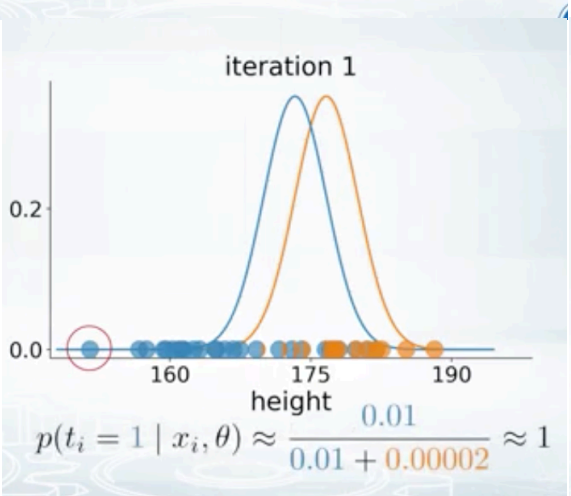


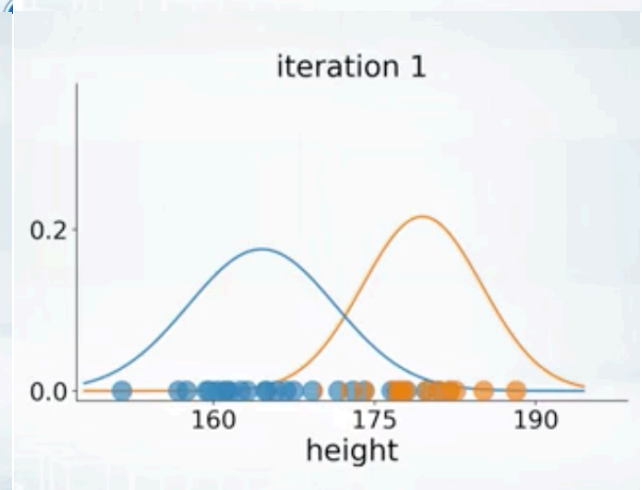


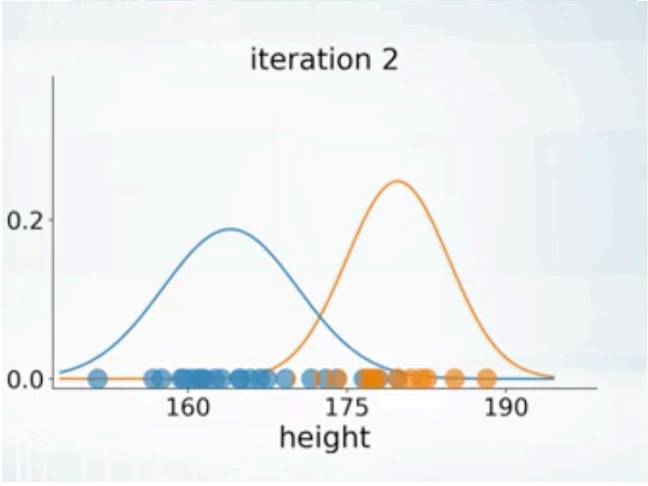
Example of GMM training

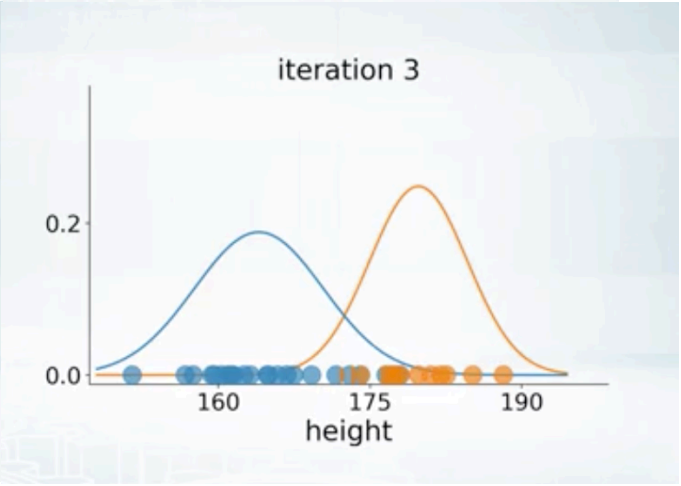












GMM may stuck in local maxima which is not optimal. In other word, the GMM suffers from local maximums. For best result, its best to start initialization several times and choose the

best at the end by tracking the log likelihood performance. Its sometimes faster that stochastic Gradient descent. It also handles complicated constraints.