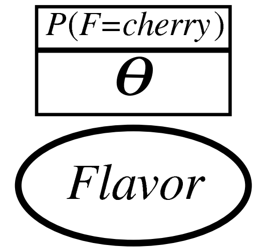
ML parameter learning in Bayes nets

Bag from a new manufacturer; fraction θ of cherry candies?
Any θ is possible: continuum of hypotheses h_θ
θ is a parameter for this simple (binomial) family of models



Suppose we unwrap N candies, c cherries and $\ell = N - c$ limes
These are i.i.d. (independent, identically distributed) observations, so

$$P(\mathbf{d}|h_\theta) = \prod_{j=1}^{N} P(d_j|h_\theta) = \theta^c \cdot (1-\theta)^\ell$$

Maximize this w.r.t. θ—which is easier for the log-likelihood:

$$L(\mathbf{d}|h_\theta) = \log P(\mathbf{d}|h_\theta) = \sum_{j=1}^{N} \log P(d_j|h_\theta) = c \log\theta + \ell \log(1-\theta)$$

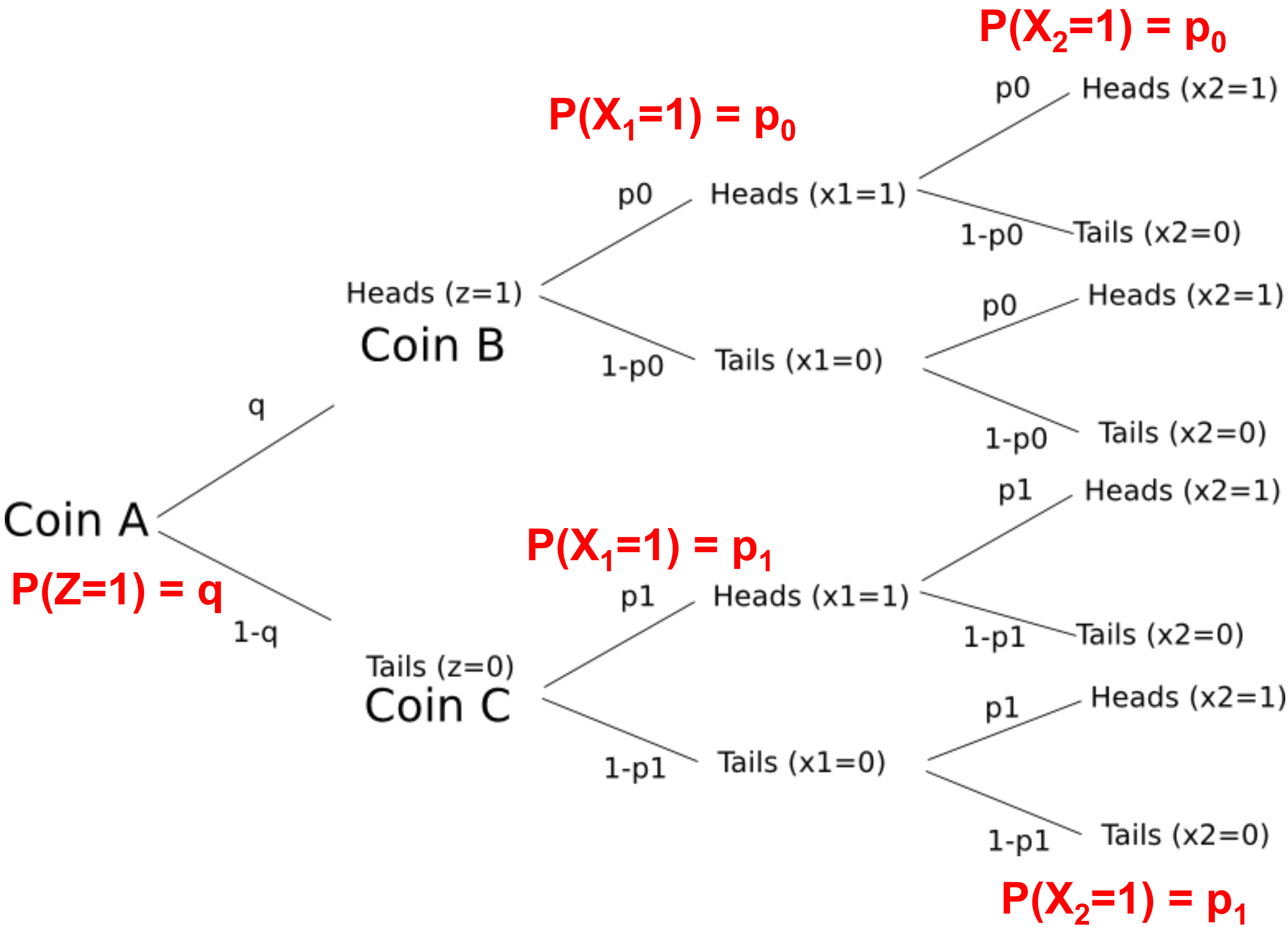
$$\frac{dL(\mathbf{d}|h_\theta)}{d\theta} = \frac{c}{\theta} - \frac{\ell}{1-\theta} = 0 \qquad \Rightarrow \qquad \theta = \frac{c}{c+\ell} = \frac{c}{N}$$

Seems sensible, but causes problems with 0 counts!

Coin-toss games

- Imagine we have three coins, A, B, C
- If toss of first coin:
  - A=heads, we then switch to coin B
  - A=tails, we then switch to coin C

- $q = P(A=\text{heads})$
- $p_0 = P(B=\text{heads})$
- $p_1 = P(C=\text{heads})$


P(X2=1) = p0
p0
Heads (x2=1)
P(X1=1) = p0
p0
Heads (x1=1)
1-p0
Tails (x2=0)
Heads (z=1)
Coin B
p0
Heads (x2=1)
1-p0
Tails (x1=0)
1-p0
Tails (x2=0)
q
p1
Heads (x2=1)
Coin A
P(X1=1) = p1
P(Z=1) = q
p1
Heads (x1=1)
1-q
1-p1
Tails (x2=0)
Tails (z=0)
Coin C
p1
Heads (x2=1)
1-p1
Tails (x1=0)
1-p1
Tails (x2=0)
P(X2=1) = p1

Flip the coins and observe…

z	x1	x2
0	1	0
1	0	0
0	1	1
0	0	1
1	1	1
0	0	0
1	1	0
0	0	1
0	1	0
0	0	0

Given:

$$q = 1/2$$

$$p_0 = 3/4$$

$$p_1 = 1/4$$

What if you're not given q?

$$q = \bar{z} = \frac{1}{n}\sum_{i=1}^{n} z_i$$

But what if we're not shown z?

z	x1	x2
?	1	0
?	0	0
?	1	1
?	0	1
?	1	1
?	0	0
?	1	0
?	0	1
?	1	0
?	0	0

Given only:

$$p_0 = 3/4$$

$$p_1 = 1/4$$

Use a first guess q = $q_{(0)}$

$$P(Z=z|X_1=x1,X_2=x_2)=\frac{P(X_1=x_1,X_2=x_2|Z=z)P(Z=z)}{\sum_{z_i=0}^{1}P(X_1=x_1,X_2=x_1|Z=z_i)P(Z=z_i)}$$

$$P(Z=1|X_1=1,X_2=1)=\frac{P(X_1=1,X_2=1|Z=1)P(Z=1)}{\sum_{z_i=0}^{1}P(X_1=1,X_2=1|Z=z_i)P(Z=z_i)}$$

$$=\frac{p_0^2 q_{(0)}}{p_0^2 q_{(0)}+p_1^2(1-q_{(0)})}$$

Let's guess $q_{(0)} = 0.1$

and use $p_0 = 3/4$

$p_1 = 1/4$

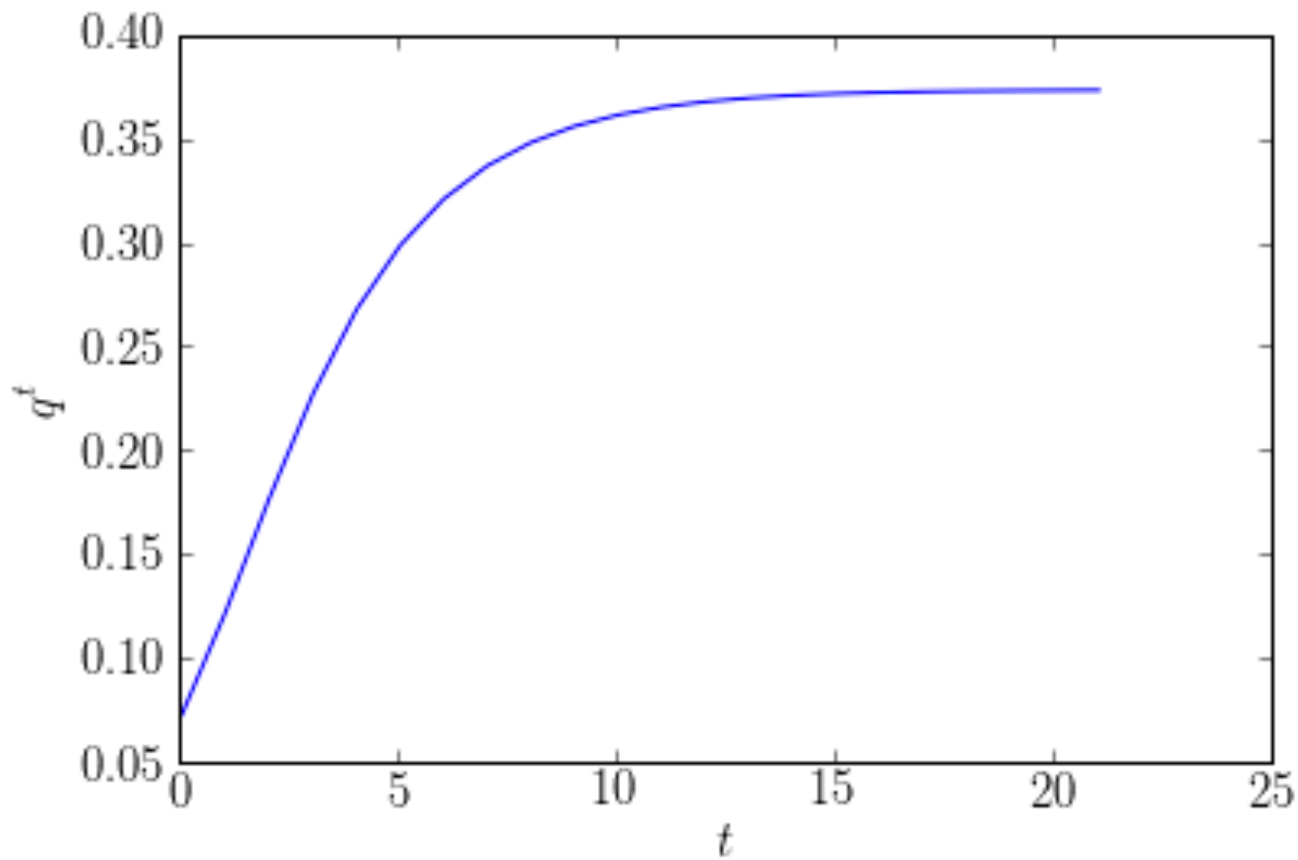
With a guess of $q_{(0)} = 0.1$, this gives

z	x1	x2
0.1	1	0
0.012195	0	0
0.5	1	1
0.1	0	1
0.5	1	1
0.012195	0	0
0.1	1	0
0.1	0	1
0.1	1	0
0.012195	0	0

So let's refine our guess:

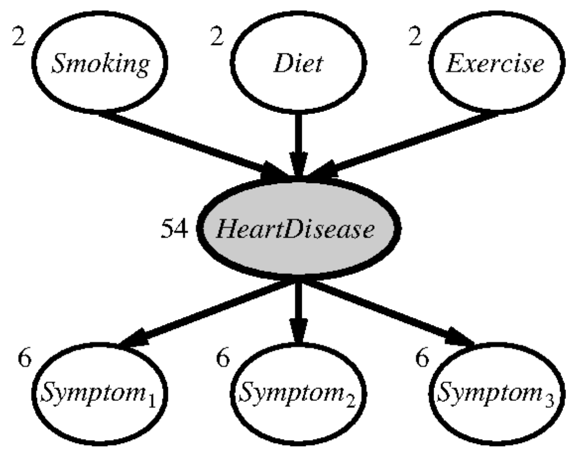
$$q_{(1)} = \bar{z} = \frac{1}{n}\sum_{i=1}^{n} z_i$$

What happens over time?



Expectation Maximization

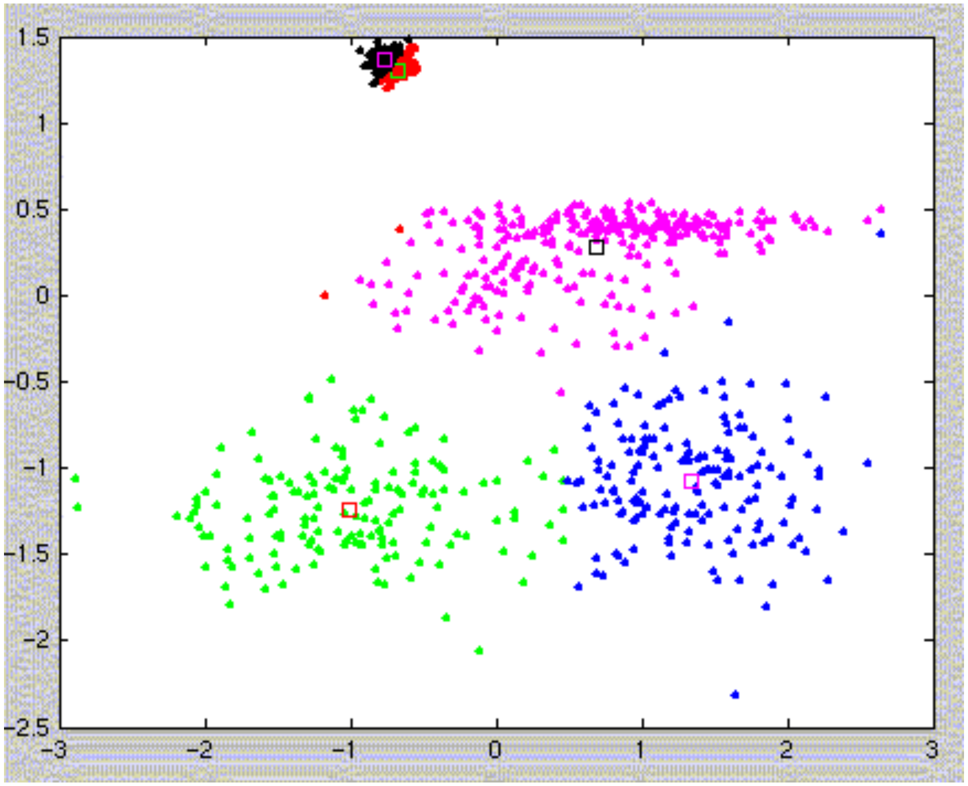
- MAP & ML learning deal with fully observable cases
- what if data is **incomplete** or has **missing** values?
- e.g., medical records contain:
  - health indicators
  - symptoms
  - **but not** the disease



- **goal** : assume the data comes from an underlying distribution; we need to guess the most likely (**maximum likelihood**) parameters of that model.

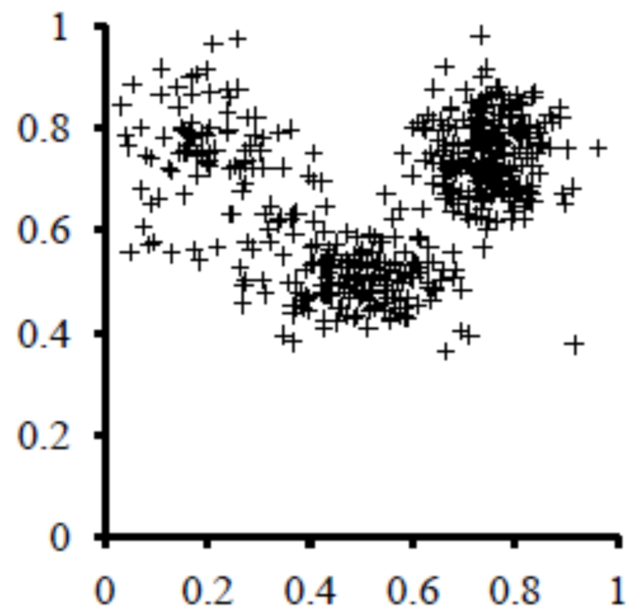
Example: Learning a Multivariate Gaussian Distribution

- suppose we have spectra of 100,000 stars
- how many stars of each type (white dwarf, red giant, etc.) are there?



Example: Learning a Multivariate Gaussian Distribution

- suppose we have spectra of 100,000 stars
- how many stars of each type (white dwarf, red giant, etc.) are there?

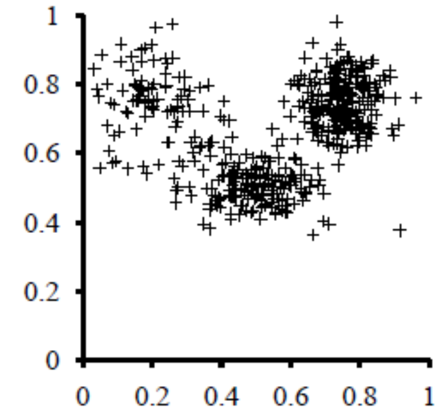


Learning a Multivariate Gaussian Distribution

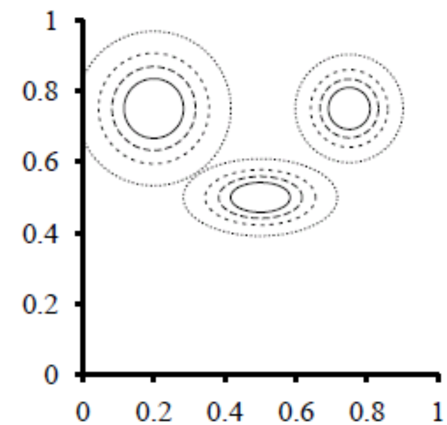
- given a set of *n* data points (e.g., stars)

$$(\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n) = \{\vec{X}\}$$

  whose attributes x_i represent spectral intensities at f_1 and f_2:
- assume underlying distribution is MoG with *k* components
- each component C_i has a:
  - $w_i = P(C = i)$ = weight or likelihood
  - μ_i = mean
  - Σ_i = co-variance
- **goal**: estimate parameters of each Gaussian distribution



500 points sampled from model



Gaussian mixture model

Learning Mixtures of Gaussians (MoG)

- mixture distribution given by: $P(\mathbf{x}) = \sum_{i=1}^{k} P(\mathbf{x} \mid C = i) P(C = i)$
- if we knew which component generated each data point…
  - we could solve for the Gaussian parameters directly
- or, if we knew parameters of each component…
  - we could assign each data point (probabilistically) to a component
- our problem:
  - we know neither!

Let's pretend we do! (Expectation)

- pretend we know the parameters of the model (weights, means, and co-variance of each Gaussian)
- compute probabilities that each data point belongs to component C_i

$$p_{ij} = P(C = i \mid \mathbf{x}_j)$$

$$= \alpha P(\mathbf{x}_j \mid C = i) P(C = i)$$

- for convenience, define: $p_i = \sum_j p_{ij}$
- equivalent to computing the expected values of a hidden "indicator" variable, Z_{ij}
  ($Z_{ij} = 1$ if data $\boldsymbol{x}_j$ was generated by component C_i)

Now find maximum likelihood of data given the expectation (Maximization)

- compute new model parameters based on the expectation

$$\mu_i \leftarrow \sum_j p_{ij} \mathbf{x}_j / p_i$$

$$\Sigma_i \leftarrow \sum_j p_{ij} (\mathbf{x}_j - \mu_i)(\mathbf{x}_j - \mu_i)^T$$

$$w_i \leftarrow p_i$$

- maximizes the log likelihood of the data, given the expected values of the hidden indicator variables

EM algorithm in a nutshell

- given a set of incomplete (observed) data
- assume observed data come from a specific model
- guess (or pretend we know) parameters for model
- repeat:
  - use this to guess the missing value/data:
    infer probability that each data point belongs to each component (**expectation step**)
  - refit the components to the data:
    from the missing data and observed data, find the most likely parameters (**maximization step**)
- until convergence

EM in general

- Let **x** be all the observed values
- Let **Z** denote all the hidden variables
- Let θ be all the parameters for the probability model

$$\theta^{(i+1)} = \arg\max_{\theta} \sum_{z} P(Z = z \mid x, \theta^{(i)}) L(x, Z = z \mid \theta)$$

- "expectation of the log likelihood of the completed data with respect to the distribution $P(Z = z \mid x, \theta^{(i)})$, which is the posterior over the hidden variables, given the data"
- "maximization of this expected log likelihood with respect to the parameters"