Mixture Density Networks

Sargur Srihari

Mixture Density Networks

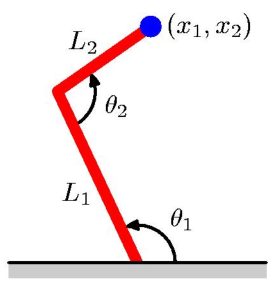
- Goal of supervised learning is to model the conditional distribution $p(t|x)$
- In some problems distribution can be multimodal
  - Particularly in inverse problems
- Gaussian assumption can lead to poor results
  - In regression $p(t|x)$ is typically assumed to be Gaussian
  - i.e., $p(t|x)=N(t|y(x,w),b^{-1})$

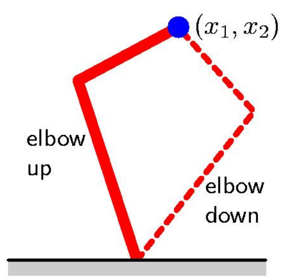
Kinematics of a robot arm

- Robot arm with two links

Forward problem:
Find end effector position given joint angles
Has a unique solution

Inverse kinematics has two solutions:
Elbow-up and elbow-down





- Inverse problem is a regression problem with
  - two inputs:
    - desired location of arm (x_1,x_2)
  - two outputs:
    - angles for links (θ_1,θ_2)
  - Has two solutions (elbow up, elbow down)

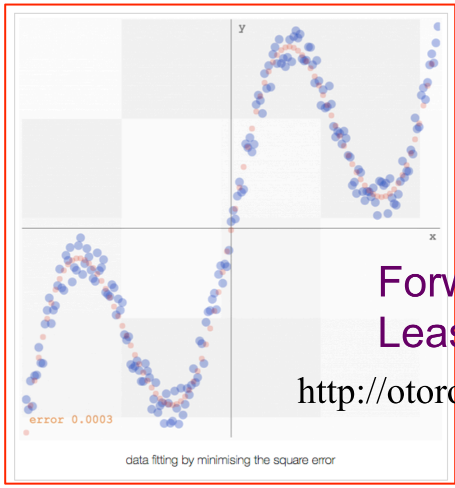
Forward and Inverse Problems

- Forward problems correspond to causality in a physical system
  - Have a unique solution
  - A disease causes a set of symptoms
- If forward problem is a many-to-one mapping,
  - Several diseases have the same symptoms
  - Inverse has multiple solutions
    - Same symptoms caused by several diseases

An Example:

x is disease, y is symptom;
x is elbow position, y is effector position

$$\boxed{y = x + 0.3sin(4\pi x) + \varepsilon}$$



data fitting by minimising the square error

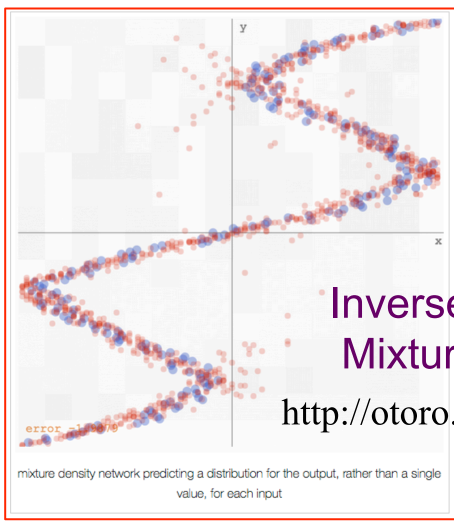
Forward Data and
Least squares regression

http://otoro.net/ml/mixture/index.html

minimum square error data fitting of inverse data

Inverse Data and
Least squares
regression

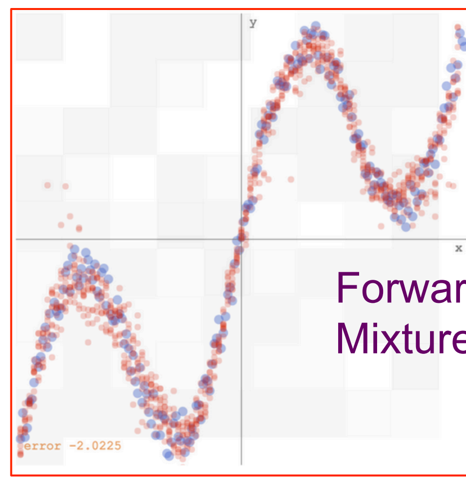
http://otoro.net/ml/mixture/inverse.html



mixture density network predicting a distribution for the output, rather than a single value, for each input

Inverse Data and
Mixture model

http://otoro.net/ml/mixture/mixture.html



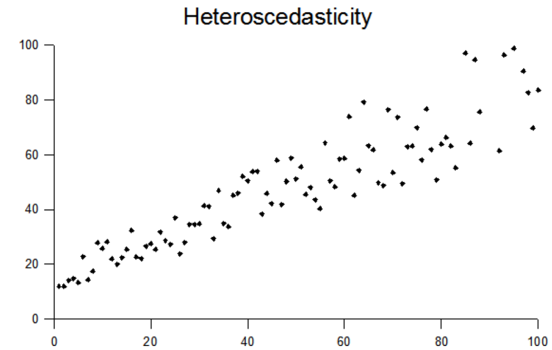
Forward Data and
Mixture model

A Mixture Density

- Generative model with K components

$$p(\mathrm{t} \mid \mathrm{x}) = \sum_{k=1}^{K} \pi_k(\mathrm{x}) \mathrm{N}(\mathrm{t} \mid \mu_k(\mathrm{x}), \sigma_k^2(\mathrm{x}))$$

  - Components can be Gaussian for continuous variables, Bernoulli for binary target variables, etc
- Note that mixing coefficient π is dependent on x and sums to 1 for each x
- So also mean and variance
  - An example of a hetero-scedastic model since variance is a function of the input vector x

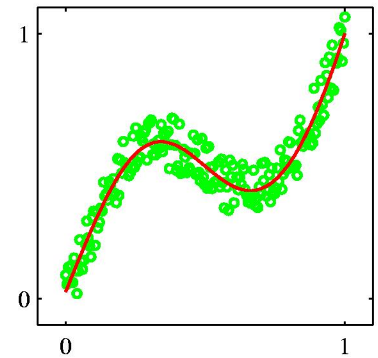


Variance does not remain constant

Data Set for Forward and Inverse Problems

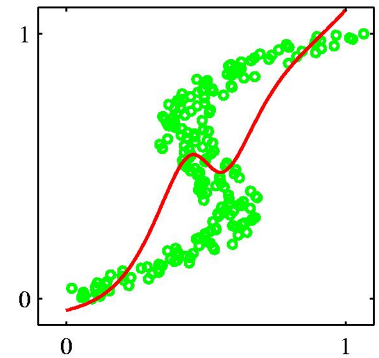
- Least squares corresponds to
  - Maximum likelihood under a Gaussian assumption
  - Leads to a poor result for highly non-Gaussian inverse problem
- Seek a general framework for modeling conditional probability distributions
- Achieved by using a mixture model $p(\mathrm{t}|\mathrm{x})$

Forward problem data set:
x is sampled uniformly over $(0,1)$ to give values $\{x_n\}$
Target t_n obtained by function
$x_n+0.3\sin(2\pi x_n)$
Then add noise over $(-0.1,0.1)$



Red curve is result of fitting a two-layer neural network by minimizing sum-of-squared error

Corresponding inverse problem by reversing x and t



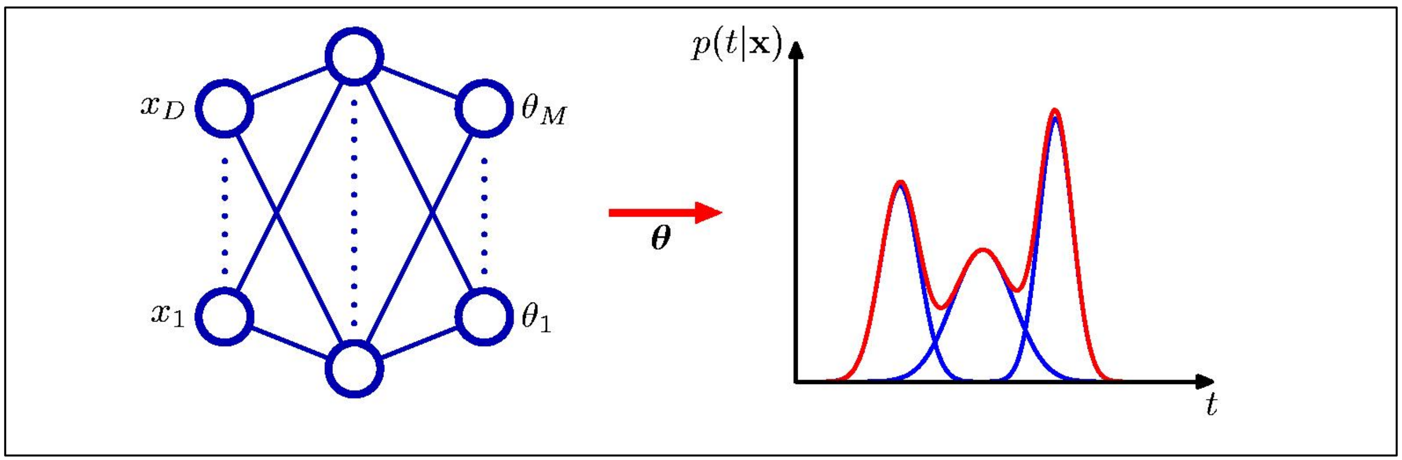
Very poor fit to data

Parameters of Mixture Model

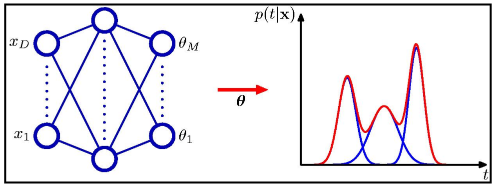
- Parameters of the mixture density:
  1. Mixing coefficients $\pi_k(\mathrm{x})$
  2. Means $\mu_k(\mathrm{x})$
  3. Variances $\sigma_k^2(\mathrm{x})$
- Governed by the outputs of a neural network
  - With x as input
- A single network predicts the parameters of all the component densities

Mixture density network

- Network represents general conditional probability densities $p(\mathrm{t}|\mathrm{x})$ by considering a parametric mixture model
- It takes x as input and provides the parameters of the distribution as output
  - In effect, it specifies the distribution of $p(\mathrm{t}|\mathrm{x})$
  - Takes x as input vector



No. of Outputs of Neural Network



- Two-layer network with sigmoidal *(tanh)* hidden units
- No. of output units calculated as follows:
  - If K components in mixture model then there are K mixing coefficients $\pi_k(\mathrm{x})$ determined by activations a_k^{π}
  - K outputs a_k^{σ} that determine kernel widths $\sigma_k(\mathrm{x})$
  - If Target t has L components then there are K x L outputs denoted a_{kj}^{μ} that determine components $\mu_{kj}(\mathrm{x})$ of kernel centres $\mu_k(\mathrm{x})$
- Then network will have $(L+2)K$ outputs
  - Instead of usual L outputs of a network, which simply predict the conditional means of target variables

Outputs of Mixture Density Network

1. Mixing coefficients
   - must satisfy $$\sum_{k=1}^{K} \pi_k(\mathrm{x}) = 1, \qquad 0 \le \pi_k(\mathrm{x}) \le 1$$
   - Achieved using softmax outputs

$$\pi_k(\mathrm{x}) = \frac{\exp(a_k^{\pi})}{\sum_{l=1}^{K} \exp(a_l^{\pi})}$$

2. Variances
   - Must satisfy $\sigma_k^2(\mathrm{x}) \ge 0$
   - Represented as exponentials of activations $\sigma_k(\mathrm{x}) = \exp(a_k^{\sigma})$
3. Means
   - Real components represented directly by output activations

$$\mu_{kj}(\mathrm{x}) = a_{kj}^{\mu}$$

Error Function for Mixture Density Network

- Can be set by maximum likelihood
- From distribution

$$p(\mathrm{t} \mid \mathrm{x}) = \sum_{k=1}^{K} \pi_k(\mathrm{x}) \mathrm{N}(\mathrm{t} \mid \mu_k(\mathrm{x}), \sigma_k^2(\mathrm{x}))$$

This is what will be used to predict the output for a given input

- Negative logarithm of Likelihood function is

$$E(\mathrm{w}) = -\sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k(\mathrm{x}_\mathrm{n}, \mathrm{w}) \mathrm{N}(\mathrm{t} \mid \mu_k(\mathrm{x}_\mathrm{n}, \mathrm{w}), \sigma_k^2(\mathrm{x}_\mathrm{n}, \mathrm{w})) \right\}$$

Minimization of Error Function

- Need to calculate derivatives of error $E(\mathrm{w})$ wrt component
- Can be evaluated provided we find suitable expressions for for derivatives of error wrt output unit activations
  - They represent error signals for each pattern and output unit
- Derivative terms are easily obtained due to summation of terms one for each data point

View of mixing coefficients

- Convenient to view mixing coefficients $\pi_k(\mathrm{x})$ as x-dependent prior probabilities
- Corresponding posterior probabilities are

$$\gamma_k(\mathrm{t} \mid \mathrm{x}) = \frac{\pi_k N_{nk}}{\sum_{l=1}^{K} \pi_l N_{nl}}$$

- where N_{nk} denotes $N(\mathrm{t}_n|\mu_k(\mathrm{x}_n),\sigma_k^2(\mathrm{x}_n))$

Derivatives with respect to Network Output Activations

1. Mixing coefficients

$$\frac{\partial E_n}{\partial a_k^{\pi}} = \pi_k - \gamma_k$$

2. Component means

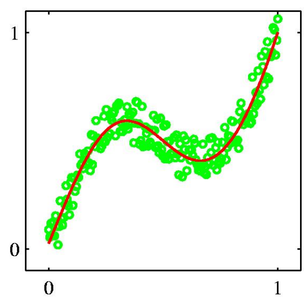
$$\frac{\partial E_n}{\partial a_{kl}^{\mu}} = \gamma_k \left\{ \frac{\mu_{kl} - t_l}{\sigma_k^2} \right\}$$

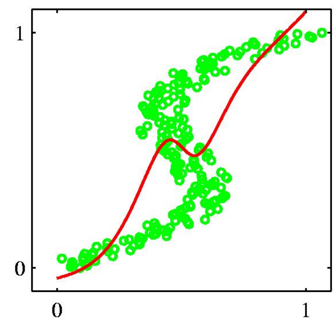
3. Component variances

$$\frac{\partial E_n}{\partial a_{kl}^{\sigma}} = -\gamma_k \left\{ \frac{\| t - \mu_k \|^2}{\sigma_k^2} - \frac{1}{\sigma_k} \right\}$$

Training Data for Mixture Density Network

- Obtained easily from forward data by exchanging roles of x and t
  - For different joint angles, the position of end effector

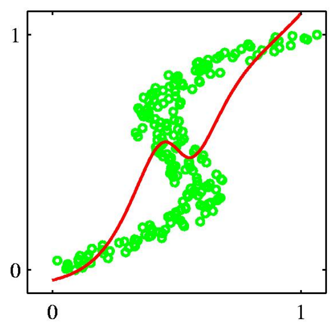




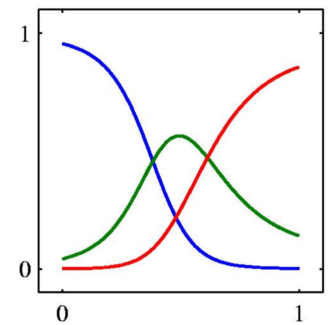
  - Note that we are using x as input and t as output <u>after</u> data exchange

Output of Mixture Density Network

Data Set



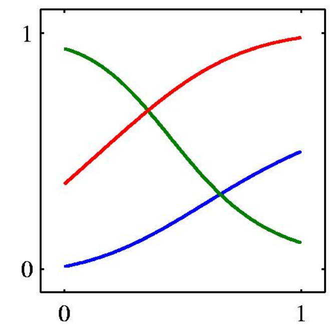
Mixing Coefficients
$\pi_k(x)$ versus x



(a)

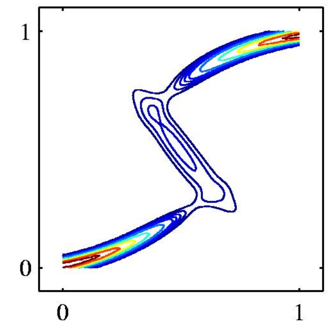
Three components have to sum to Unity

Means $\mu_k(x)$



(b)

Contours of Conditional probability density of target data



(c)

While the outputs of the neural network (and hence the parameters) are necessarily single valued, The model is able to produce a conditional density that is unimodal for some values of x and trimodal for other values

Use of Mixture Density Network

- Once mixture density network has been trained
    - can predict conditional density function of the target data for given value of input vector

- From this density can calculate more specific quantities of interest in applications

    e.g., mean of the target data

Predicting value of output vector

- Conditional distribution represents complete description of generator of data
- From this density we can calculate the mean
  - Which is the conditional average of target data

$$E[\mathrm{t} \mid \mathrm{x}] = \int \mathrm{t} p(\mathrm{t} \mid \mathrm{x}) d\mathrm{t} = \sum_{k=1}^{K} \pi_k(\mathrm{x}) \mu_k(\mathrm{x})$$

This is expected value, Not Error!

  - This is same as least squared solution and is limited value
    - Average of two solutions is not a solution
  - Variance of density function about the conditional average

$$s^2(\mathrm{x}) = E\left[\| t - E\left[t \mid \mathrm{x}\right] \|^2 \mid \mathrm{x}\right]$$

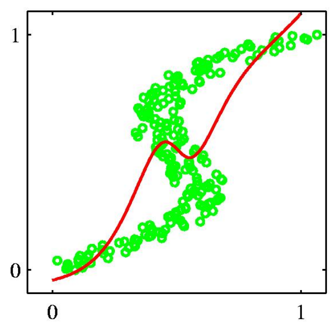
$$= \sum_{\mathrm{k}=1}^{\mathrm{K}} \pi_{\mathrm{k}}(\mathrm{x}) \left\{ \sigma_k^2(\mathrm{x}) + \left\| \mu_k(\mathrm{x}) - \sum_{l=1}^{K} \pi_l(\mathrm{x}) \mu_l(\mathrm{x}) \right\|^2 \right\}$$

Mode as the solution

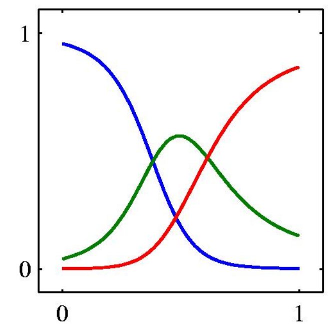
- Each of the modes of mixture density is a better solution than the single mean
- Does not have a simple analytical solution
  - Need numerical iteration
- Simple alternative:
  - Take mean of the most probable component
    - One with largest mixing coefficient for each value of x

Example of Mixture Density Network

Data Set

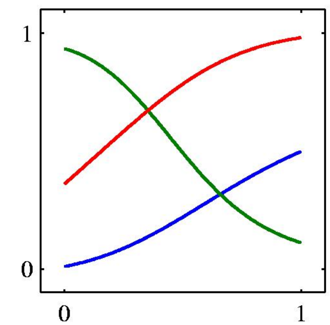


Mixing Coefficients
$\pi_k(x)$ versus x



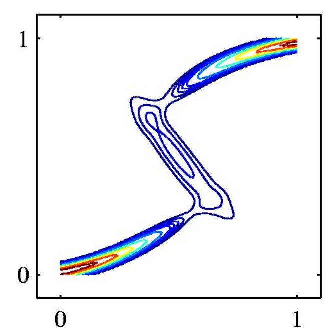
(a)

Means $\mu_k(x)$



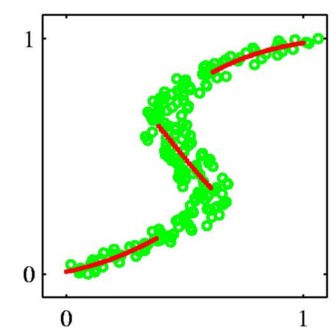
(b)

Contours of
Conditional
Probability density



(c)

Approximate
Conditional mode
(red points of
Conditional density)



(d)