Outline

Nonlinear in the Inputs
  Nonlinear Functions of the Inputs
  Example with Polynomials
  Example with Radial Basis Functions

Probabilistic Framework
  Linear Model as Probabilistic Model
  Fitting the Linear, Probabilistic Model
  ML versus MAP
Linear in Parameters, Nonlinear in Inputs

- Linear dependence on parameters can be maintained if we substitute for the $x_i$ some nonlinear functions of $x$.

\[ y(x, w) = w_0 + w_1 \phi_1(x) + w_2 \phi_2(x) + \cdots + w_{M-1} \phi_{M-1}(x) \]

\[ = w^T \phi(x), \text{ where } \phi_0(x) = 1 \]
Linear in Parameters, Nonlinear in Inputs

- Linear dependence on parameters can be maintained if we substitute for the $x_i$ some nonlinear functions of $x$.

$$y(x, w) = w_0 + w_1\phi_1(x) + w_2\phi_2(x) + \cdots + w_{M-1}\phi_{M-1}(x)$$

$$= w^T\phi(x), \text{ where } \phi_0(x) = 1$$

- The functions $\phi_i$ could be any functions, but typical examples are

$$\phi_i(x) = x_i \text{ makes } y(x, w) \text{ linear in } x$$

$$= x_j^k \text{ or } x_j x_k$$

$$= e^{-\frac{1}{2}(x - \mu)^T\Sigma^{-1}(x - \mu)}$$

$$= \frac{1}{1 + e^{-a^T x}}$$

$$= \tanh(a^T x)$$
Example: Noisy Sine

\[ f(x) = \sin(3x) + \mathcal{N}(0, 0.2) \text{ where } x = [0, 1] \]
Example: Noisy Sine

- \( f(x) = \sin(3x) + \mathcal{N}(0, 0.2) \) where \( x = [0, 1] \)
Example: Noisy Sine

\[ f(x) = \sin(3x) + \mathcal{N}(0, 0.2) \text{ where } x = [0, 1] \]

```r
X <- matrix(seq(0,1,len=10))
f <- function(X) matrix(sin(3*X) + rnorm(length(X),0,0.2))
T <- f(X)
Xtest <- X + 0.05
Ttest <- f(Xtest)
```

![Data](chart.png)
Polynomial Functions

- Linear model won’t work well. Need to use some nonlinear functions, $\phi(x)$ of the input. Which ones?

Polynomial Functions

- Linear model won’t work well. Need to use some nonlinear functions, $\phi(x)$ of the input. Which ones?
Polynomial Functions

- Linear model won’t work well. Need to use some nonlinear functions, $\phi(x)$ of the input. Which ones?
- Let’s try polynomials.
Polynomial Functions

- Linear model won’t work well. Need to use some nonlinear functions, \( \phi(x) \) of the input. Which ones?
- Let’s try polynomials.
Polynomial Functions

- Linear model won’t work well. Need to use some nonlinear functions, $\phi(x)$ of the input. Which ones?
- Let’s try polynomials.

$$\phi_1(x) = x$$
$$\phi_2(x) = x^2$$
$$\phi_3(x) = x^3$$
$$\vdots$$
$$\phi_d(x) = x^d$$
Polynomial Functions

- Linear model won’t work well. Need to use some nonlinear functions, $\phi(x)$ of the input. Which ones?
- Let’s try polynomials.

\[
\begin{align*}
\phi_1(x) &= x \\
\phi_2(x) &= x^2 \\
\phi_3(x) &= x^3 \\
& \vdots \\
\phi_d(x) &= x^d
\end{align*}
\]
Results for Different Maximum Degrees $d$

black - training data red - testing data blue - model on testing data

Nonlinear in the Inputs
Nonlinear Functions of the Inputs
Example with Polynomials
Example with Radial Basis Functions

Probabilistic Framework
Linear Model as Probabilistic Model
Fitting the Linear, Probabilistic Model
ML versus MAP
Results for Different Maximum Degrees $d$

![Graph showing test RMSE values for different polynomial degrees](image-url)
Another popular choice is radial basis functions, or RBFs.
Radial Basis Functions

- Another popular choice is radial basis functions, or RBFs.
Radial Basis Functions

- Another popular choice is radial basis functions, or RBFs.

\[ \phi_1(x) = e^{-\frac{(x-0)^2}{0.1}} \]
\[ \phi_2(x) = e^{-\frac{(x-0.2)^2}{0.1}} \]
\[ \phi_3(x) = e^{-\frac{(x-0.4)^2}{0.1}} \]
\[ \vdots \]
Another popular choice is radial basis functions, or RBFs.

\[
\begin{align*}
\phi_1(x) &= e^{-\frac{(x-0)^2}{0.1}} \\
\phi_2(x) &= e^{-\frac{(x-0.2)^2}{0.1}} \\
\phi_3(x) &= e^{-\frac{(x-0.4)^2}{0.1}} \\
&\vdots
\end{align*}
\]
Results for Different Numbers of RBFs

black - training data red - testing data blue - model on testing data

Nonlinear in the Inputs

Nonlinear Functions of the Inputs
Example with Polynomials
Example with Radial Basis Functions

Probabilistic Framework
Linear Model as Probabilistic Model
Fitting the Linear, Probabilistic Model
ML versus MAP
Results for Different Numbers of RBFs
Outline

Nonlinear in the Inputs
Nonlinear Functions of the Inputs
Example with Polynomials
Example with Radial Basis Functions

Probabilistic Framework
Linear Model as Probabilistic Model
Fitting the Linear, Probabilistic Model
ML versus MAP
To enter the probabilistic world, let’s say our model $y(x_n, w)$ predicts $t_n$ with an error that is modeled as a Gaussian random variable with precision $\beta$.

$$t_n = y(x_n, w) + \epsilon$$

$$p(t_n|x_n, w, \beta) = \mathcal{N}(t_n|y(x_n, w), \beta^{-1})$$
To enter the probabilistic world, let’s say our model $y(x_n, w)$ predicts $t_n$ with an error that is modeled as a Gaussian random variable with precision $\beta$.

$$t_n = y(x_n, w) + \epsilon$$

$$p(t_n| x_n, w, \beta) = \mathcal{N}(t_n | y(x_n, w), \beta^{-1})$$

Our prediction of $t_n$ for a given sample $x_n$ is now not a single value, but a distribution over possible values—its expected value conditioned on $x_n$.

$$E[t_n| x_n] = \int t \ p(t_n| x_n) dt = y(x_n, w)$$
The likelihood for all data samples is

\[
p(T|X, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|y(x_n, w), \beta^{-1})
\]

\[
= \prod_{n=1}^{N} \mathcal{N}(t_n|w^T \phi(x_n), \beta^{-1})
\]

where \( \phi(x_n) \) is a vector of the values of all of the nonlinear functions (sometimes called basis functions) applied to the sample \( x_n \).
Fitting Model by Maximum Likelihood

- Taking the logarithm of the likelihood we get

\[
\ln p(T|X, w, \beta) = -\frac{1}{2} \sum_{n=1}^{N} (t_n - w^T \phi(x_n))^2 - \frac{N}{2} \ln \beta
\]

\[-\frac{N}{2} \ln(2\pi)\]

- Then take derivative, actually gradient, with respect to \(w\). (Just like minimizing squared error before we entered the probabilistic realm!)
Fitting Model by Maximum Likelihood

- Taking the logarithm of the likelihood we get

\[
\ln p(T|X, w, \beta) = -\frac{1}{2} \sum_{n=1}^{N} (t_n - w^T \phi(x_n))^2 - \frac{N}{2} \ln \beta
\] - \frac{N}{2} \ln(2\pi)

- Then take derivative, actually gradient, with respect to \(w\). (Just like minimizing squared error before we entered the probabilistic realm!)

\[
\nabla_w \ln p(T|X, w, \beta) = \sum_{n=1}^{N} (t_n - w^T \phi(x_n)) \phi(x_n)^T
\]

- Setting this equal to zero we can solve for \(w\).

\[
0 = \sum_{n=1}^{N} t_n \phi(x_n)^T - w^T \left( \sum_{n=1}^{N} \phi(x_n) \phi(x_n)^T \right)
\]
Fitting Model by Maximum Likelihood

- These sums can be expressed as matrix operations if we define

\[ \Phi = \begin{pmatrix}
\phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_{D-1}(x_1) \\
\phi_0(x_2) & \phi_1(x_2) & \cdots & \phi_{D-1}(x_2) \\
\vdots & \vdots & \ddots & \vdots \\
\phi_0(x_N) & \phi_1(x_N) & \cdots & \phi_{D-1}(x_N)
\end{pmatrix} \]

Now the above equation becomes the following one and the solution for \( w \) continues.

\[ 0 = \Phi^T T - \Phi^T \Phi w \]
\[ \Phi^T T = \Phi^T \Phi w \]
\[ w = (\Phi^T \Phi)^{-1} \Phi^T T \]

- So, fitting a probabilistic model defined for Gaussian distribution of fixed precision and mean as a linear function of inputs, gives same solution as the non-probabilistic least squares solution.
Maximum a Posterior

- Alternative to maximizing the likelihood of the data is to maximize the posterior distribution of $w = p(w|X, T, \text{parameters})$, where parameters represent basis functions and distribution parameters our model will be based on.

Alternative to maximizing the likelihood of the data is to maximize the posterior distribution of $w = p(w|X, T, \text{parameters})$, where parameters represent basis functions and distribution parameters our model will be based on.
Maximum a Posterior

- Alternative to maximizing the likelihood of the data is to maximize the posterior distribution of \( w = p(w | X, T, \text{parameters}) \), where parameters represents basis functions and distribution parameters our model will be based on.
- Called the MAP solution (versus the ML, or maximum likelihood solution).
Maximum a Posterior

- Alternative to maximizing the likelihood of the data is to maximize the posterior distribution of \( w = p(w|\mathbf{X}, \mathbf{T}, \text{parameters}) \), where parameters represents basis functions and distribution parameters our model will be based on.

- Called the MAP solution (versus the ML, or maximum likelihood solution).

- Using Bayes Theorem, we know (leaving out the parameters)

\[
p(w|\mathbf{X}, \mathbf{T}) = \frac{p(\mathbf{T}|\mathbf{X}, w)p(w)}{p(\mathbf{T})}
\]

We will be maximizing this by finding best \( w \) which does not affect the denominator, so can work just with the numerator.
Maximum a Posterior

- Alternative to maximizing the likelihood of the data is to maximize the posterior distribution of \( w = p(w|X, T, \text{parameters}) \), where parameters represents basis functions and distribution parameters our model will be based on.

- Called the MAP solution (versus the ML, or maximum likelihood solution).

- Using Bayes Theorem, we know (leaving out the parameters)

\[
p(w|X, T) = \frac{p(T|X, w)p(w)}{p(T)}
\]

- We will be maximizing this by finding best \( w \) which does not affect the denominator, so can work just with the numerator.
Choosing $p(w)$

- Must choose a prior distribution for $w$. 

$\text{Choosing } p(w)$ 

$\text{Must choose a prior distribution for } w.$
Choosing $p(w)$

- Must choose a prior distribution for $w$.
- What value of $w$ would we prefer if no training data exists?

\[ p(w | \alpha) = N(w | \alpha^{-1} I) \]
Choosing $p(w)$

- Must choose a prior distribution for $w$.
- What value of $w$ would we prefer if no training data exists?
- $w = 0$
Choosing \( p(w) \)

- Must choose a prior distribution for \( w \).
- What value of \( w \) would we prefer if no training data exists?
- \( w = 0 \)
- With small amount of data, let’s still strongly bias it towards something close to zero.

Must choose a prior distribution for \( w \).

What value of \( w \) would we prefer if no training data exists?

\( w = 0 \)

With small amount of data, let’s still strongly bias it towards something close to zero.

Must choose a prior distribution for \( w \).

What value of \( w \) would we prefer if no training data exists?

\( w = 0 \)

With small amount of data, let’s still strongly bias it towards something close to zero.
Choosing $p(w)$

- Must choose a prior distribution for $w$.
- What value of $w$ would we prefer if no training data exists?
- $w = 0$
- With small amount of data, let’s still strongly bias it towards something close to zero.
- Gaussian distribution will do this, and it is mathematically convenient. Use mean of zero and precision that we choose empirically to control how strongly we want to force $w$ to stay close to zero.

$$p(w|\alpha) = \mathcal{N}(w|\alpha^{-1}I)$$
Choosing $p(w)$

- Now, using our previous probabilistic model for $p(T|X, w, \beta)$, we have

$$p(w|X, T, \alpha, \beta) \propto p(T|X, w, \beta)p(w|\alpha)$$

$$p(T|X, w, \beta)p(w|\alpha) = \prod_{n=1}^{N} \mathcal{N}(t_n|\phi(x_n)w, \beta^{-1})\mathcal{N}(w|\alpha^{-1}I)$$
Choosing $p(w)$

- Now, using our previous probabilistic model for $p(T|X, w, \beta)$, we have

\[
p(w|X, T, \alpha, \beta) \propto p(T|X, w, \beta)p(w|\alpha)
\]

\[
p(T|X, w, \beta)p(w|\alpha) = \prod_{n=1}^{N} \mathcal{N}(t_n|\phi(x_n)w, \beta^{-1})\mathcal{N}(w|\alpha^{-1}I)
\]

- To maximize this, take the logarithm.

\[
\sum_{n=1}^{N} \ln \mathcal{N}(t_n|\phi(x_n)w, \beta^{-1}) + \ln \mathcal{N}(w|\alpha^{-1}I)
\]

\[
= -\frac{1}{2} \beta \sum_{n=1}^{N} (t_n - \phi(x_n)w)^2 + \frac{1}{2} N \ln \beta - \frac{1}{2} N \ln(2\pi)
\]

\[
- \frac{1}{2} w^T \alpha/w + \frac{1}{2} \alpha/l - \frac{1}{2} \ln(2\pi)
\]
Choosing $p(w)$

- Now, using our previous probabilistic model for $p(T|X, w, \beta)$, we have

$$p(w|X, T, \alpha, \beta) \propto p(T|X, w, \beta)p(w|\alpha)$$

$$p(T|X, w, \beta)p(w|\alpha) = \prod_{n=1}^{N} \mathcal{N}(t_n|\phi(x_n)w, \beta^{-1})\mathcal{N}(w|\alpha^{-1}I)$$

- To maximize this, take the logarithm.

$$\sum_{n=1}^{N} \ln \mathcal{N}(t_n|\phi(x_n)w, \beta^{-1}) + \ln \mathcal{N}(w|\alpha^{-1}I)$$

$$= -\frac{1}{2} \beta \sum_{n=1}^{N}(t_n - \phi(x_n)w)^2 + \frac{1}{2} N \ln \beta - \frac{1}{2} N \ln(2\pi)$$

$$- \frac{1}{2} w^T \alpha / w + \frac{1}{2} \alpha I - \frac{1}{2} \ln(2\pi)$$

- Now take the gradient of this with respect to $w$. 
The gradient of this with respect to $\mathbf{w}$ is

$$
\beta \sum_{n=1}^{N} (t_n - \phi(x_n)\mathbf{w})\phi(x_n)^T - \alpha I \mathbf{w}
$$
• The gradient of this with respect to $\mathbf{w}$ is

$$
\beta \sum_{n=1}^{N} (t_n - \phi(x_n)\mathbf{w})\phi(x_n)^T - \alpha / \mathbf{w}
$$

• Setting this equal to zero and rearranging a bit we get

$$
0 = \beta \left( \sum_{n=1}^{N} t_n \phi(x_n)^T - \mathbf{w}^T \sum_{n=1}^{N} \phi(x_n)\phi(x_n)^T \right) - \alpha / \mathbf{w}
$$

$$
= \beta (\Phi^T \mathbf{T} - \Phi^T \Phi \mathbf{w}) - \alpha / \mathbf{w}
$$
The gradient of this with respect to $\mathbf{w}$ is

$$\beta \sum_{n=1}^{N} (t_n - \phi(x_n)^T \mathbf{w}) \phi(x_n)^T - \alpha / \mathbf{w}$$

Setting this equal to zero and rearranging a bit we get

$$0 = \beta \left( \sum_{n=1}^{N} t_n \phi(x_n)^T - \mathbf{w}^T \sum_{n=1}^{N} \phi(x_n) \phi(x_n)^T \right) - \alpha / \mathbf{w}$$

$$= \beta (\Phi^T \mathbf{T} - \Phi^T \Phi \mathbf{w}) - \alpha / \mathbf{w}$$

Solving for $\mathbf{w}$ results in

$$0 = \beta (\Phi^T \mathbf{T} - \Phi^T \Phi \mathbf{w}) - \alpha / \mathbf{w}$$

$$0 = \Phi^T \mathbf{T} - \Phi^T \Phi \mathbf{w} - \frac{\alpha}{\beta} / \mathbf{w}$$

$$\Phi^T \Phi \mathbf{w} + \frac{\alpha}{\beta} / \mathbf{w} = \Phi^T \mathbf{T}$$

$$(\Phi^T \Phi + \frac{\alpha}{\beta} I) \mathbf{w} = \Phi^T \mathbf{T}$$

$$\mathbf{w} = (\Phi^T \Phi + \frac{\alpha}{\beta} I)^{-1} \Phi^T \mathbf{T}$$
Does this look familiar? Let’s call the solution $w_{MAP}$.

$$w_{MAP} = \left( \Phi^T \Phi + \frac{\alpha}{\beta} I \right)^{-1} \Phi^T \mathbf{T}$$
Does this look familiar? Let’s call the solution $w_{\text{MAP}}$.

$$w_{\text{MAP}} = \left( \Phi^T\Phi + \frac{\alpha}{\beta} I \right)^{-1} \Phi^T \mathbf{t}$$

Should remind you of our least squares solution with a weight penalty.

$$w_{\text{LS}} = \left( \Phi^T\Phi + \lambda I \right)^{-1} \Phi^T \mathbf{t}$$