More Bayes

CS4061 / CS5014 Machine Learning

Paul Henderson
University of Glasgow

paul.henderson@glasgow.ac.uk

Based on previous material by
Simon Rogers & Ke Yuan

Bayesian Probability

- what does $p(\text{head}) = 0.5$ **mean**?

Bayesian Probability

- what does $p(\text{head}) = 0.5$ **mean**?
- **frequentist** view: if I flip the coin infinitely many times, it shows heads 50% of the time

Bayesian Probability

- what does $p(\text{head}) = 0.5$ **mean**?
- **frequentist** view: if I flip the coin infinitely many times, it shows heads 50% of the time
- **Bayesian** view: I have 50% confidence that if I flip the coin **once**, it'll show heads
- probability = **degree of belief**

Bayesian Linear Regression

- model parameters $\mathbf{w}$ are a random variable
  - we're uncertain what values they take
  - **prior** $p(\mathbf{w})$ represents our assumptions about 'sensible' models

Bayesian Linear Regression

- model parameters $\mathbf{w}$ are a random variable
  - we're uncertain what values they take
  - **prior** $p(\mathbf{w})$ represents our assumptions about 'sensible' models

- we have training data $(\mathbf{X}, \mathbf{t})$
  - **likelihood** $p(\mathbf{t} \mid \mathbf{X}, \mathbf{w})$ says how probable it is for given model $\mathbf{w}$

Bayesian Linear Regression

- model parameters $\mathbf{w}$ are a random variable
  - we're uncertain what values they take
  - **prior** $p(\mathbf{w})$ represents our assumptions about 'sensible' models

- we have training data $(\mathbf{X}, \mathbf{t})$
  - **likelihood** $p(\mathbf{t} \mid \mathbf{X}, \mathbf{w})$ says how probable it is for given model $\mathbf{w}$

- want a density $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})$ that combines these
- it should give high probability to outcomes consistent with **both** data **and** prior

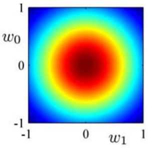
Bayesian Linear Regression

- model parameters $\mathbf{w}$ are a random variable
  - we're uncertain what values they take
  - **prior** $p(\mathbf{w})$ represents our assumptions about 'sensible' models
- we have training data $(\mathbf{X}, \mathbf{t})$
  - **likelihood** $p(\mathbf{t} \,|\, \mathbf{X}, \mathbf{w})$ says how probable it is for given model $\mathbf{w}$
- want a density $p(\mathbf{w} \,|\, \mathbf{X}, \mathbf{t})$ that combines these
- it should give high probability to outcomes consistent with **both** data **and** prior
- called the **posterior**; given by Bayes' rule:

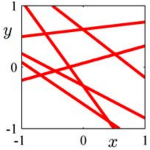
$$p(\mathbf{w} \,|\, \mathbf{X}, \mathbf{t}) = \frac{p(\mathbf{t} \,|\, \mathbf{X}, \mathbf{w})\, p(\mathbf{w})}{p(\mathbf{t} \,|\, \mathbf{X})}$$

Bayesian Linear Regression

prior

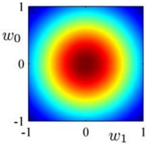


data & model

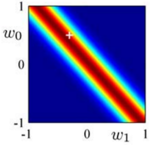


Bayesian Linear Regression

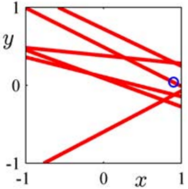
prior



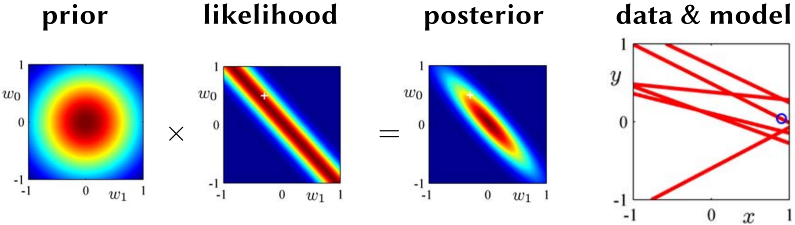
likelihood



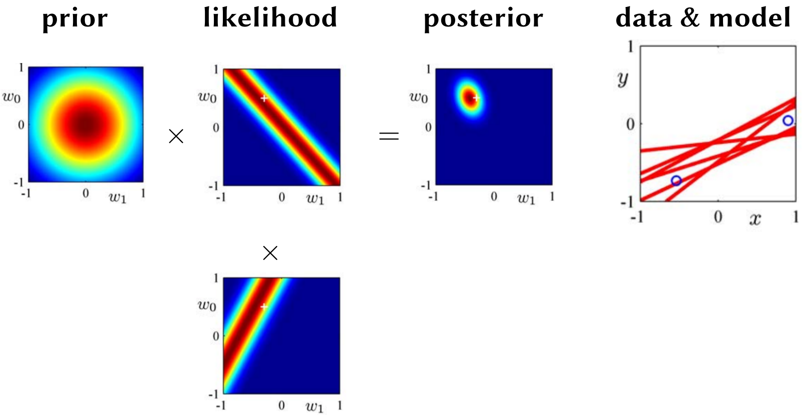
data & model



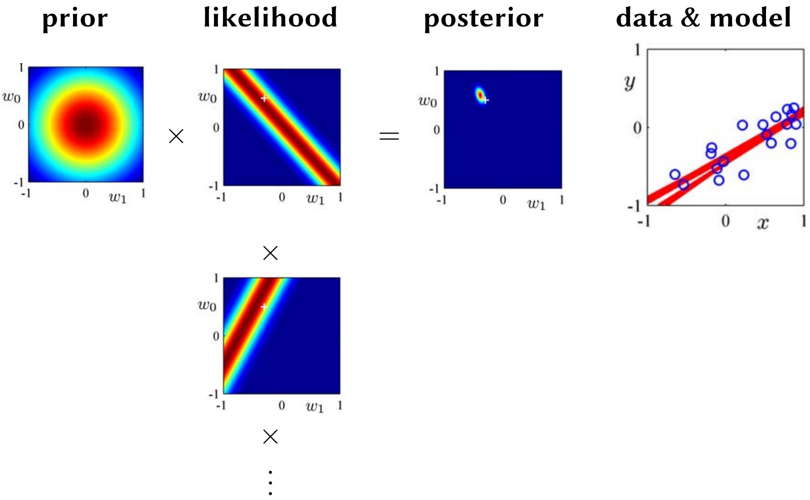
Bayesian Linear Regression



Bayesian Linear Regression



Bayesian Linear Regression



Computing the posterior

- Bayes rule:

$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}) = \frac{p(\mathbf{t}|\mathbf{X}, \mathbf{w})p(\mathbf{w})}{\color{red}{p(\mathbf{t}|\mathbf{X})}}$$

- Unfortunately, computing the posterior is hard...
- ...because marginal likelihood $p(\mathbf{t}|\mathbf{X})$ is hard to compute:

$$p(\mathbf{t}|\mathbf{X}) = \int p(\mathbf{t}|\mathbf{w}, \mathbf{X})p(\mathbf{w})\ d\mathbf{w}$$

- Sometimes we can do it (e.g. everything Gaussian)
- Usually we can't and need some trick/alternative

Example – Olympic data

- Model predictions are Gaussian

$$\mathbf{t} \sim \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$$

- $\mathbf{t}$ is a vector containing all the t_n
- $\mathbf{X}$ is a matrix containing all the $\mathbf{x}_n$

Example – Olympic data

- Model predictions are Gaussian

$$\mathbf{t} \sim \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$$

- $\mathbf{t}$ is a vector containing all the t_n
- $\mathbf{X}$ is a matrix containing all the $\mathbf{x}_n$

- Joint likelihood is given by

$$p(\mathbf{t}|\mathbf{w}, \mathbf{X}, \sigma^2) = \prod_{n=1}^{N} p(t_n|\mathbf{w}, \mathbf{x}_n, \sigma^2)$$

Example – Olympic data

- Model predictions are Gaussian

$$\mathbf{t} \sim \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$$

- $\mathbf{t}$ is a vector containing all the t_n
- $\mathbf{X}$ is a matrix containing all the $\mathbf{x}_n$

- Joint likelihood is given by

$$p(\mathbf{t}|\mathbf{w}, \mathbf{X}, \sigma^2) = \prod_{n=1}^{N} p(t_n|\mathbf{w}, \mathbf{x}_n, \sigma^2)$$

- Ignoring a constant, this is

$$p(\mathbf{t}|\mathbf{w}, \mathbf{X}, \sigma^2) \propto \exp\left\{-\frac{1}{2\sigma^2}(\mathbf{t} - \mathbf{X}\mathbf{w})^{\mathsf{T}}(\mathbf{t} - \mathbf{X}\mathbf{w})\right\}$$

Example – Olympic data

- Choose a Gaussian prior:

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \mathbf{S}), \ \mathbf{S} = \begin{bmatrix} 100 & 0 \\ 0 & 5 \end{bmatrix}$$

- Mean ($\mathbf{0}$) and covariance ($\mathbf{S}$) are design choices
- Lets us inject our own knowledge about what $\mathbf{w}$ are likely

Example – Olympic data

- Choose a Gaussian prior:

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \mathbf{S}), \ \mathbf{S} = \begin{bmatrix} 100 & 0 \\ 0 & 5 \end{bmatrix}$$

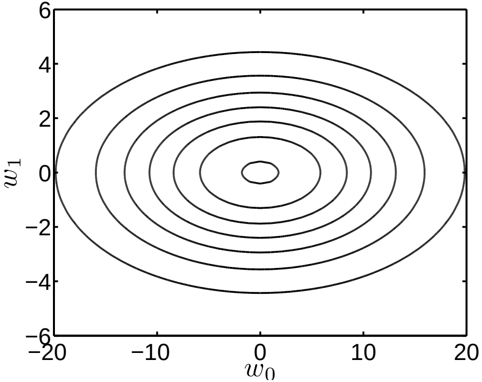
- Mean ($\mathbf{0}$) and covariance ($\mathbf{S}$) are design choices
- Lets us inject our own knowledge about what $\mathbf{w}$ are likely

- Density is (ignoring a constant)

$$p(\mathbf{w}) \propto \exp\left\{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{S}^{-1}\mathbf{w}\right\}$$

Example – Olympic data

- Rescale Olympic year:
  - $1896 = 1, 1900 = 2, \ldots, 2008 = 27, 2012 = 28$
- Prior density:



- Mean ($\mathbf{0}$) and covariance ($\mathbf{S}$)

Example – Olympic data

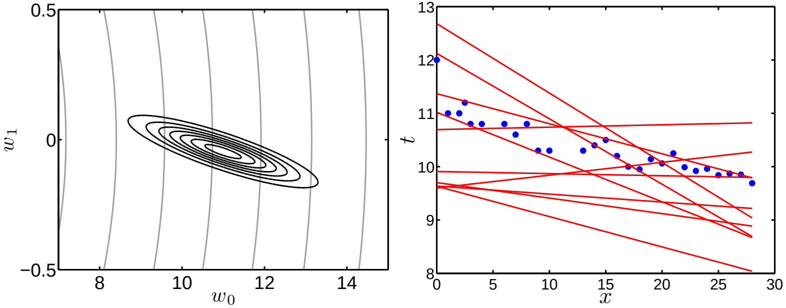
- Ignoring non $\mathbf{w}$ terms, the prior multiplied by the likelihood is:

$$
\begin{aligned}
& p(\mathbf{t} \mid \mathbf{w}, \mathbf{X}, \sigma^2)\, p(\mathbf{w}) \\
\propto \quad & \exp\left\{-\frac{1}{2\sigma^2}(\mathbf{t}-\mathbf{X}\mathbf{w})^{\mathsf{T}}(\mathbf{t}-\mathbf{X}\mathbf{w})\right\} \exp\left\{-\frac{1}{2}\mathbf{w}^{\mathsf{T}}\mathbf{S}^{-1}\mathbf{w}\right\} \\
\propto \quad & \exp\left\{-\frac{1}{2}\left(\mathbf{w}^{\mathsf{T}}\left[\frac{1}{\sigma^2}\mathbf{X}^{\mathsf{T}}\mathbf{X}+\mathbf{S}^{-1}\right]\mathbf{w}-\frac{2}{\sigma^2}\mathbf{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\mathbf{t}\right)\right\}
\end{aligned}
$$

- Can be rearranged to (yet another) Gaussian
- It has parameters:

$$
\boldsymbol{\mu} = \frac{1}{\sigma^2}\boldsymbol{\Sigma}\mathbf{X}^{\mathsf{T}}\mathbf{t} \qquad\qquad \boldsymbol{\Sigma} = \left(\frac{1}{\sigma^2}\mathbf{X}^{\mathsf{T}}\mathbf{X}+\mathbf{S}^{-1}\right)^{-1}
$$

Olympic data – Posterior



- *Left*: posterior (black) and prior (grey), zoomed in
- *Right*: functions corresponding to some $\mathbf{w}$ sampled from posterior

Olympic data – Predictions

- Our motivation for being Bayesian was to be able to average predictions (at $\mathbf{x}_{\text{new}}$) over all $\mathbf{w}$:

$$\mathbf{E}_{p(\mathbf{w}|\mathbf{X},\mathbf{t},\sigma^2)}\left\{f(\mathbf{w})\right\} = \int f(\mathbf{w})p(\mathbf{w}|\mathbf{t},\mathbf{X},\sigma^2)\,d\mathbf{w}$$

Olympic data – Predictions

- Our motivation for being Bayesian was to be able to average predictions (at $\mathbf{x}_{\text{new}}$) over all $\mathbf{w}$:

$$\mathbf{E}_{p(\mathbf{w}|\mathbf{X},\mathbf{t},\sigma^2)}\left\{f(\mathbf{w})\right\} = \int f(\mathbf{w})p(\mathbf{w}|\mathbf{t},\mathbf{X},\sigma^2)\ d\mathbf{w}$$

- For our model, $f(\mathbf{w})$ is a Gaussian density:

$$f(\mathbf{w}) = f_{\mathcal{N}}(t_{\text{new}}; \mathbf{w}^{\mathsf{T}}\mathbf{x}_{\text{new}}, \sigma^2)$$

Olympic data – Predictions

- Our motivation for being Bayesian was to be able to average predictions (at $\mathbf{x}_{\text{new}}$) over all $\mathbf{w}$:

$$\mathbf{E}_{p(\mathbf{w}|\mathbf{X},\mathbf{t},\sigma^2)}\left\{f(\mathbf{w})\right\} = \int f(\mathbf{w}) p(\mathbf{w}|\mathbf{t},\mathbf{X},\sigma^2)\, d\mathbf{w}$$

- For our model, $f(\mathbf{w})$ is a Gaussian density:

$$f(\mathbf{w}) = f_{\mathcal{N}}(t_{\text{new}}; \mathbf{w}^{\mathsf{T}}\mathbf{x}_{\text{new}}, \sigma^2)$$

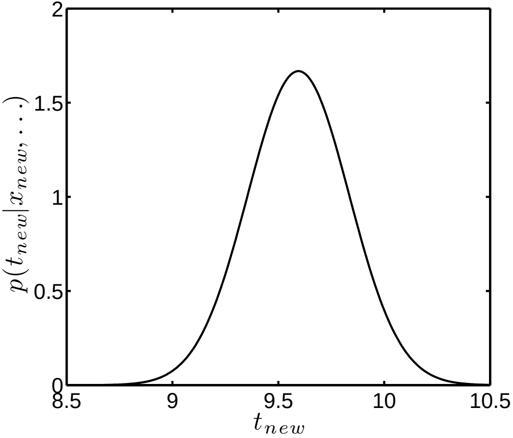
- We can compute this expectation exactly, to give predictive **density**:

$$p(t_{\text{new}}|\mathbf{X},\mathbf{t},\mathbf{x}_{\text{new}},\sigma^2) = \mathcal{N}(\mathbf{x}_{\text{new}}^{\mathsf{T}}\boldsymbol{\mu}, \sigma^2 + \mathbf{x}_{\text{new}}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{x}_{\text{new}})$$

  ...where posterior parameters were:

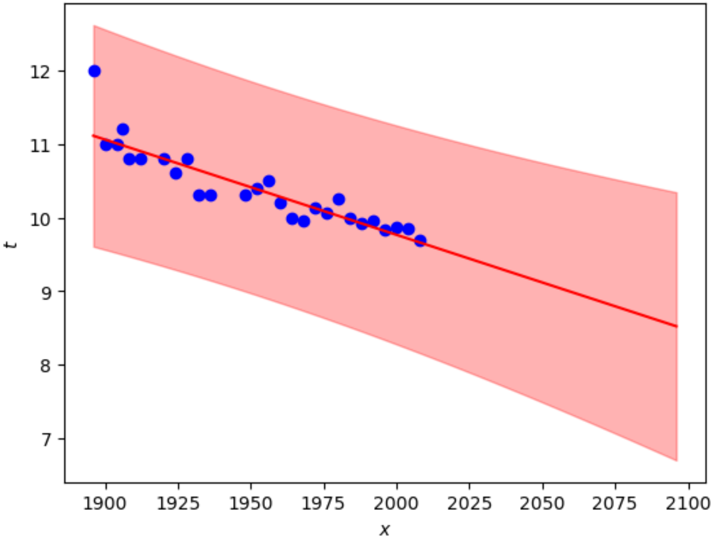
$$\boldsymbol{\mu} = \frac{1}{\sigma^2}\boldsymbol{\Sigma}\mathbf{X}^{\mathsf{T}}\mathbf{t} \qquad \boldsymbol{\Sigma} = \left(\frac{1}{\sigma^2}\mathbf{X}^{\mathsf{T}}\mathbf{X} + \mathbf{S}^{-1}\right)^{-1}$$

Olympic data – Predictions



Predictive density for 2012 Olympics

Olympic data – Predictions



Maximum *a posteriori*

- posterior $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})$ incorporates observations and prior knowledge
- posterior $\propto$ likelihood $\times$ prior

Maximum *a posteriori*

- posterior $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})$ incorporates observations and prior knowledge
- posterior $\propto$ likelihood $\times$ prior

- so far: made predictions by considering entire posterior, i.e.

$$\mathbb{E}_{p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})} \{ \dots \}$$

...called **fully Bayesian** approach

Maximum a posteriori

- posterior $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})$ incorporates observations and prior knowledge
- posterior $\propto$ likelihood $\times$ prior

- so far: made predictions by considering entire posterior, i.e.

$$\mathbb{E}_{p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})} \{ \dots \}$$

  ...called **fully Bayesian** approach

- alternative: just consider **mode** of posterior

  ...called ***maximum a posteriori*** approach

Maximum *a posteriori*

- posterior $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})$ incorporates observations and prior knowledge
- posterior $\propto$ likelihood $\times$ prior

- for Olympics, posterior was Gaussian with parameters

$$\boldsymbol{\mu} = \frac{1}{\sigma^2}\boldsymbol{\Sigma}\mathbf{X}^\mathsf{T}\mathbf{t} \qquad\qquad \boldsymbol{\Sigma} = \left(\frac{1}{\sigma^2}\mathbf{X}^\mathsf{T}\mathbf{X} + \mathbf{S}^{-1}\right)^{-1}$$

Maximum *a posteriori*

- posterior $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})$ incorporates observations and prior knowledge
- posterior $\propto$ likelihood $\times$ prior

- for Olympics, posterior was Gaussian with parameters

$$\boldsymbol{\mu} = \frac{1}{\sigma^2}\boldsymbol{\Sigma}\mathbf{X}^\mathsf{T}\mathbf{t} \qquad \boldsymbol{\Sigma} = \left(\frac{1}{\sigma^2}\mathbf{X}^\mathsf{T}\mathbf{X} + \mathbf{S}^{-1}\right)^{-1}$$

- mode = mean for Gaussian, hence MAP estimate of $\mathbf{w}$ is

$$\mathbf{w}^* = \frac{1}{\sigma^2}\boldsymbol{\Sigma}\mathbf{X}^\mathsf{T}\mathbf{t}$$

Maximum *a posteriori*

$$
\begin{aligned}
\mathbf{w}^* &= \operatorname*{argmax}_{\mathbf{w}} p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}) \\
&= \operatorname*{argmax}_{\mathbf{w}} \, \log p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})
\end{aligned}
$$

Maximum *a posteriori*

$$\begin{aligned}\mathbf{w}^* &= \operatorname*{argmax}_{\mathbf{w}} p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}) \\ &= \operatorname*{argmax}_{\mathbf{w}} \log p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})\end{aligned}$$

- posterior $\propto$ likelihood $\times$ prior
- for Olympics, chose Gaussian prior and Gaussian likelihood

$$p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}) \propto \exp\left\{-\frac{1}{2\sigma^2}(\mathbf{t} - \mathbf{X}\mathbf{w})^\mathsf{T}(\mathbf{t} - \mathbf{X}\mathbf{w})\right\} \exp\left\{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{S}^{-1}\mathbf{w}\right\}$$

Maximum a posteriori

$$\begin{aligned}\mathbf{w}^* &= \operatorname*{argmax}_{\mathbf{w}} p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}) \\ &= \operatorname*{argmax}_{\mathbf{w}} \log p(\mathbf{w} \mid \mathbf{X}, \mathbf{t})\end{aligned}$$

- posterior $\propto$ likelihood $\times$ prior
- for Olympics, chose Gaussian prior and Gaussian likelihood

$$p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}) \propto \exp\left\{-\frac{1}{2\sigma^2}(\mathbf{t} - \mathbf{X}\mathbf{w})^\mathsf{T}(\mathbf{t} - \mathbf{X}\mathbf{w})\right\} \exp\left\{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{S}^{-1}\mathbf{w}\right\}$$

$$\log p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}) = -\frac{1}{2\sigma^2}\sum_{n=1}^{N}(t_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n)^2 - \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{S}^{-1}\mathbf{w} + \text{const.}$$

Maximum a posteriori

$$\begin{aligned}\mathbf{w}^* &= \operatorname*{argmax}_{\mathbf{w}} p(\mathbf{w} \,|\, \mathbf{X},\, \mathbf{t}) \\ &= \operatorname*{argmax}_{\mathbf{w}} \; \log p(\mathbf{w} \,|\, \mathbf{X},\, \mathbf{t})\end{aligned}$$

- posterior $\propto$ likelihood $\times$ prior
- for Olympics, chose Gaussian prior and Gaussian likelihood

$$p(\mathbf{w} \,|\, \mathbf{X},\, \mathbf{t}) \propto \exp\left\{-\frac{1}{2\sigma^2}(\mathbf{t} - \mathbf{X}\mathbf{w})^\mathsf{T}(\mathbf{t} - \mathbf{X}\mathbf{w})\right\} \exp\left\{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{S}^{-1}\mathbf{w}\right\}$$

$$\log p(\mathbf{w} \,|\, \mathbf{X},\, \mathbf{t}) = -\frac{1}{2\sigma^2}\sum_{n=1}^{N}(t_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n)^2 - \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{S}^{-1}\mathbf{w} + \text{const.}$$

$$\therefore \quad \mathbf{w}^* = \operatorname*{argmax}_{\mathbf{w}} \left\{ -\sum_{n=1}^{N}(t_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n)^2 - \frac{\lambda}{2}\sum_{k=1}^{K} w_k^2 \right\}$$

Maximum a posteriori

$$
\begin{aligned}
\mathbf{w}^* &= \underset{\mathbf{w}}{\operatorname{argmax}}\, p(\mathbf{w} \,|\, \mathbf{X},\, \mathbf{t}) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}}\; \log p(\mathbf{w} \,|\, \mathbf{X},\, \mathbf{t})
\end{aligned}
$$

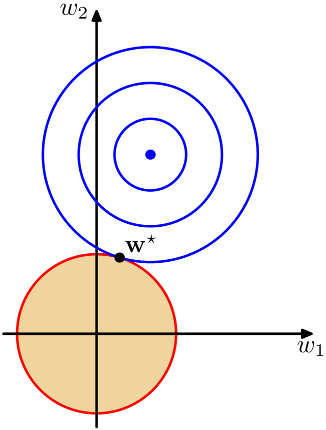
- posterior $\propto$ likelihood $\times$ prior
- for Olympics, chose Gaussian prior and Gaussian likelihood

$$
p(\mathbf{w} \,|\, \mathbf{X},\, \mathbf{t}) \propto \exp\left\{ -\frac{1}{2\sigma^2}(\mathbf{t} - \mathbf{X}\mathbf{w})^\mathsf{T}(\mathbf{t} - \mathbf{X}\mathbf{w}) \right\} \exp\left\{ -\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{S}^{-1}\mathbf{w} \right\}
$$

$$
\log p(\mathbf{w} \,|\, \mathbf{X},\, \mathbf{t}) = -\frac{1}{2\sigma^2}\sum_{n=1}^{N}(t_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n)^2 - \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{S}^{-1}\mathbf{w} + \text{const.}
$$

$$
\therefore \quad \mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \left\{ \sum_{n=1}^{N}(t_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n)^2 + \frac{\lambda}{2}\sum_{k=1}^{K} w_k^2 \right\}
$$

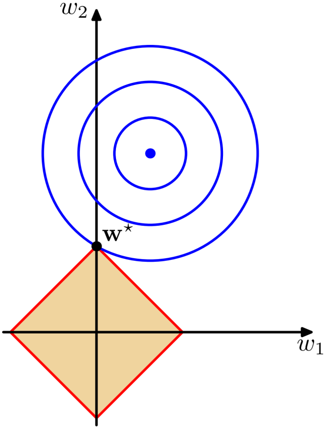
Regularised Least Squares



$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{n=1}^{N} (t_n - \mathbf{w}^\mathsf{T} \mathbf{x}_n)^2 + \frac{\lambda}{2} \sum_{k=1}^{K} w_k^2$$

- 2nd term = **L2 regulariser**

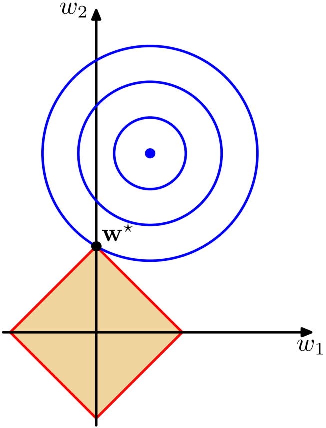
Regularised Least Squares



$$\mathbf{w}^* = \operatorname*{argmin}_{\mathbf{w}} \sum_{n=1}^{N} (t_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n)^2 + \frac{\lambda}{2} \sum_{k=1}^{K} |w_k|$$

- 2nd term = **L1 regulariser**

Regularised Least Squares



$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{n=1}^{N} (t_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n)^2 + \frac{\lambda}{2} \sum_{k=1}^{K} |w_k|$$

- 2^{nd} term = **L1 regulariser**

- encourages **sparsity**
- many elements of **w** will be zero

Summary

- Saw how predictions could be made by averaging over all possible parameter values $\mathbf{w}$, conditioned on training data – Bayesian
- Handles uncertainty in measurements *and* in $\mathbf{w}$ itself
- Can use marginal likelihood to compare models
- Maximum *a posteriori* is a simpler alternative to full Bayesian approach

Deep learning for regression

- linear model: $t = \mathbf{w}^\mathsf{T}\mathbf{x}$
  - $\mathbf{w}$ is a vector

Deep learning for regression

- linear model: $t = \mathbf{w}^\mathsf{T}\mathbf{x}$
  - $\mathbf{w}$ is a vector

- deep linear model: $t = \mathbf{W}_1\mathbf{x}'$ with $\mathbf{x}' = \mathbf{W}_0\mathbf{x}$
  - $\mathbf{W}_i$ are matrices
  - $\mathbf{x}'$ is a vector

Deep learning for regression

- linear model: $t = \mathbf{w}^\mathsf{T}\mathbf{x}$
  - $\mathbf{w}$ is a vector

- deep linear model: $t = \mathbf{W}_1\mathbf{x}'$ with $\mathbf{x}' = \mathbf{W}_0\mathbf{x}$
  - $\mathbf{W}_i$ are matrices
  - $\mathbf{x}'$ is a vector

- even deeper linear model:

$$\begin{aligned} \mathbf{x}' &= \mathbf{W}_0\mathbf{x} \\ \mathbf{x}'' &= \mathbf{W}_1\mathbf{x}' \\ t &= \mathbf{W}_2\mathbf{x}'' \end{aligned}$$

- ...etc.

Deep learning for regression

- deep linear model:

$$
\begin{aligned}
\mathbf{x}' &= \mathbf{W}_0\mathbf{x} \\
\mathbf{x}'' &= \mathbf{W}_1\mathbf{x}' \\
t &= \mathbf{W}_2\mathbf{x}''
\end{aligned}
$$

Deep learning for regression

- deep ~~linear~~ model:

$$
\begin{aligned}
\mathbf{x}' &= g(\mathbf{W}_0 \mathbf{x}) \\
\mathbf{x}'' &= g(\mathbf{W}_1 \mathbf{x}') \\
t &= \quad \mathbf{W}_2 \mathbf{x}''
\end{aligned}
$$

Deep learning for regression

- deep ~~linear~~ model:

$$\begin{aligned} \mathbf{x}' &= g(\mathbf{W}_0\mathbf{x}) \\ \mathbf{x}'' &= g(\mathbf{W}_1\mathbf{x}') \\ t &= \quad \mathbf{W}_2\mathbf{x}'' \end{aligned}$$

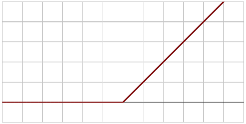
- g is...
  - nonlinear
  - elementwise, i.e. if $\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \end{pmatrix}$ then $g(\mathbf{y}) = \begin{pmatrix} g(y_1) \\ g(y_2) \\ \vdots \end{pmatrix}$

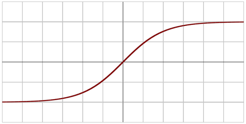
Deep learning for regression

- deep ~~linear~~ model:

$$\begin{aligned} \mathbf{x}' &= g(\mathbf{W}_0\mathbf{x}) \\ \mathbf{x}'' &= g(\mathbf{W}_1\mathbf{x}') \\ t &= \quad \mathbf{W}_2\mathbf{x}'' \end{aligned}$$

- g is...
  - nonlinear
  - elementwise, i.e. if $\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \end{pmatrix}$ then $g(\mathbf{y}) = \begin{pmatrix} g(y_1) \\ g(y_2) \\ \vdots \end{pmatrix}$
- common choices for g:





Deep learning for regression

- deep nonlinear model:

$$\begin{aligned} \mathbf{x}' &= g(\mathbf{W}_0\mathbf{x}) \\ \mathbf{x}'' &= g(\mathbf{W}_1\mathbf{x}') \\ t &= \quad \mathbf{W}_2\mathbf{x}'' \end{aligned}$$

- this can closely approximate any function $t^*(\mathbf{x})$ if either
  - two layers, and $\mathbf{W}_0$ has 'enough' rows/columns
  - fixed size $\mathbf{W}_i$ (as large as $\mathbf{x}$), and 'enough' layers
- can approximate an arbitrarily wiggly line!

Deep learning for regression

- deep nonlinear model:

$$\begin{aligned}\mathbf{x}' &= g(\mathbf{W}_0\mathbf{x})\\ \mathbf{x}'' &= g(\mathbf{W}_1\mathbf{x}')\\ t &= \quad \mathbf{W}_2\mathbf{x}''\end{aligned}$$

- this can closely approximate any function $t^*(\mathbf{x})$ if either
  - two layers, and $\mathbf{W}_0$ has 'enough' rows/columns
  - fixed size $\mathbf{W}_i$ (as large as $\mathbf{x}$), and 'enough' layers
- can approximate an arbitrarily wiggly line!

- still fit by minimising mean square error (or maximising likelihood):

$$\mathbf{W}_0^*,\ \mathbf{W}_1^*,\ \mathbf{W}_2^* = \underset{\mathbf{W}_0,\mathbf{W}_1,\mathbf{W}_2}{\operatorname{argmin}} \frac{1}{N}\sum_{n=1}^{N}\big(t_n - \mathbf{W}_2\, g(\mathbf{W}_1\, g(\mathbf{W}_0\mathbf{x}_n))\big)^2$$

Deep learning for regression

- deep nonlinear model:

$$\begin{aligned} \mathbf{x}' &= g(\mathbf{W}_0\mathbf{x}) \\ \mathbf{x}'' &= g(\mathbf{W}_1\mathbf{x}') \\ t &= \quad \mathbf{W}_2\mathbf{x}'' \end{aligned}$$

- this can closely approximate any function $t^*(\mathbf{x})$ if either
  - two layers, and $\mathbf{W}_0$ has ‘enough’ rows/columns
  - fixed size $\mathbf{W}_i$ (as large as $\mathbf{x}$), and ‘enough’ layers
- can approximate an arbitrarily wiggly line!

- still fit by minimising mean square error (or maximising likelihood):

$$\mathbf{W}_0^*,\ \mathbf{W}_1^*,\ \mathbf{W}_2^* = \underset{\mathbf{W}_0,\mathbf{W}_1,\mathbf{W}_2}{\operatorname{argmin}} \frac{1}{N}\sum_{n=1}^{N}\big(t_n - \mathbf{W}_2\, g(\mathbf{W}_1\, g(\mathbf{W}_0\mathbf{x}_n))\big)^2$$

- use gradient descent!

Deep learning for regression

- a big enough nonlinear regression model can predict lots of interesting things!

Deep learning for regression

- a big enough nonlinear regression model can predict lots of interesting things!

- e.g. image $\rightarrow$ 3D object locations

Deep learning for regression

- a big enough nonlinear regression model can predict lots of interesting things!

- e.g. image $\rightarrow$ other viewpoints

Deep learning for regression

- a big enough nonlinear regression model can predict lots of interesting things!

- e.g. random noise $\rightarrow$ cats