Linear Regression

Rob Weston

Applied Artificial Intelligence Lab

March 23, 2021

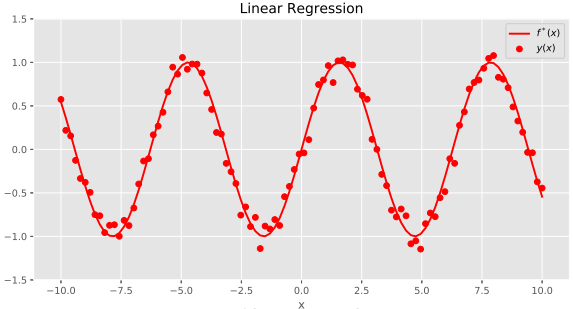
Overview

Linear Regression

Estimate a function $f^* : \mathbb{R}^D \to \mathbb{R}$ with a linear model $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{\phi}(x)$ from observations $y(\boldsymbol{x}) = f^*(\boldsymbol{x}) + \epsilon$ corrupted by noise $\epsilon \sim \text{Norm}(0, \alpha^{-1})$ where...

- $\boldsymbol{\phi} = \mathbb{R}^D \to \mathbb{R}^M$ is a *known* function mapping $\boldsymbol{x} \in \mathbb{R}^D$ to a feature space $\boldsymbol{\phi}(\boldsymbol{x}) \in \mathbb{R}^M$.
- $\boldsymbol{w} \in \mathbb{R}^M$ is used to weight each feature in $\boldsymbol{\phi}$
- α is the *observation precision* and is assumed a hyper-parameter

The *bias* is assumed to be included in $\boldsymbol{w}$ corresponding to a fixed feature $\phi(\boldsymbol{x}) = 1$. Code for the example below can be found *here*.



Overview

Linear Regression - A frequentist approach

- Find the weights that maximise the likelihood $\boldsymbol{w}^* = \arg\max_{\boldsymbol{w}} \log p(\boldsymbol{y}|\boldsymbol{w}, \boldsymbol{X}, \alpha)$ given N observations $\boldsymbol{y} = [y_1, .., y_N] \in \mathbb{R}^N$ where

$$\log p(\boldsymbol{y}|\boldsymbol{w}, \boldsymbol{X}, \alpha) = \log \prod_{n=1}^{N} \text{Nor}(y_n|\boldsymbol{w}^\top \boldsymbol{\phi}_n, \alpha^{-1}) = \frac{N}{2}\log\alpha - \frac{N}{2}\log 2\pi - \sum_{n=1}^{N}(y_n - \boldsymbol{w}^\top \boldsymbol{\phi}_n)^2 = -E(\boldsymbol{w})$$

- Differentiating with respect to $\boldsymbol{w}$ and setting to 0 gives $\boldsymbol{w}^* = (\boldsymbol{\Phi}^\top \boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^\top \boldsymbol{y}$ is equal to the *Moore-Penrose Inverse* where and $\boldsymbol{\Phi} = [\boldsymbol{\phi}(\boldsymbol{x}_1), ..., \boldsymbol{\phi}(\boldsymbol{x}_M)]^\top \in \mathbb{R}^{N\times M}$ is the *design matrix*.
- Similarly for α we have $(\alpha^*)^{-1} = \frac{1}{N}\sum_{n=1}^{N}(y_n - \boldsymbol{w}^{*\top}\boldsymbol{\phi}(\boldsymbol{x}_n))^2$
- Assuming we have inputs $\boldsymbol{x} \sim p(\boldsymbol{x})$ and letting $N \to \infty$, optimising over $\boldsymbol{w}$ is equivalent to[1]

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \mathbb{E}_{p(\boldsymbol{x},\boldsymbol{y})}[(\boldsymbol{y}(\boldsymbol{x}) - \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}))^2] = \arg\min_{\boldsymbol{w}} \mathbb{E}_{p(\boldsymbol{x},\boldsymbol{y})}[\ell(\boldsymbol{y}(\boldsymbol{x}), f(\boldsymbol{x}; \boldsymbol{w}))]$$

[1] This comes from the fact that $\boldsymbol{w}$ only depends on the term $\sum_{n=1}^{N}(y_n - \boldsymbol{w}^\top \boldsymbol{\phi}_n)^2$ in $E(\boldsymbol{w})$

Linear Regression as Optimal Function Estimation

Instead of viewing the optimisation over parameters $\boldsymbol{w}$ we can view the optimisation over functions f

- We aim to estimate a function f^* from observations $y(\boldsymbol{x}) = f^*(\boldsymbol{x}) + \epsilon$ corrupted by noise ϵ. We have

$$\hat{f} = \arg\min_f \left\{\mathbb{E}_{p(\boldsymbol{x},y)}[\ell\left(y(\boldsymbol{x}), f(\boldsymbol{x})\right)]\right\} = \arg\min_f \left\{L[f]\right\}$$

  where $\ell\left(y(\boldsymbol{x}), f(\boldsymbol{x})\right)$ is a loss function between points $y(\boldsymbol{x})$ and $f(\boldsymbol{x})$.

- This can be viewed as a *calculus of variations* problem $\implies$ "determine the *function* $f(\boldsymbol{x})$ which minimises the *functional* $L[f] = \mathbb{E}_{p(\boldsymbol{x},y)}[\ell\left(y(\boldsymbol{x}), f(\boldsymbol{x})\right)]$".
- The optimum estimator $\hat{f}$ is determined by our *choice* of loss function ℓ.
- For $\ell(y(\boldsymbol{x}), f(\boldsymbol{x})) = (y(\boldsymbol{x}) - f(\boldsymbol{x}))^2$ it can be shown that $\hat{f}(\boldsymbol{x}) = \mathbb{E}_{p(y|\boldsymbol{x})}[y(\boldsymbol{x})]$ and so $\hat{f}(\boldsymbol{x}) = f^*(\boldsymbol{x})$ (as the mean of $p(y|\boldsymbol{x}) = \mathcal{N}(y(\boldsymbol{x})|f^*(\boldsymbol{x}), \sigma^2)$ is $f^*(\boldsymbol{x})$).
- In this case $L[f]$ becomes the *mean squared error* $\text{Mse}[f]$

Learning from Noisy Observations

Maximum Likelihood with Noisy Observations

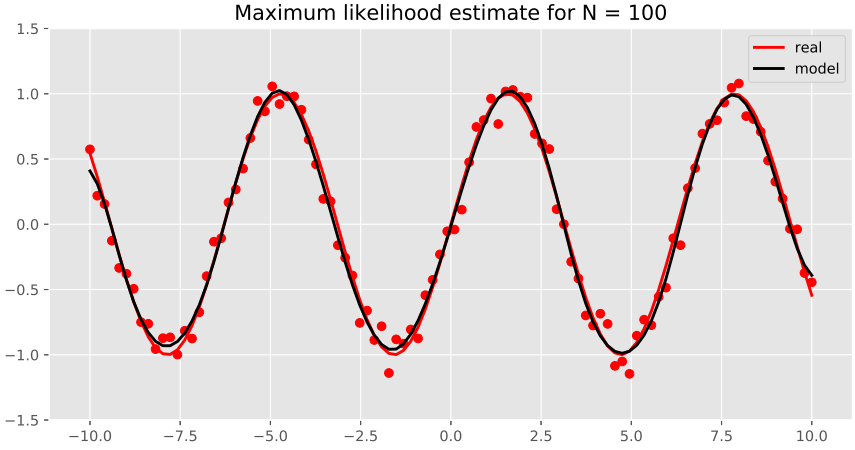
When we estimate $f(\boldsymbol{x})$ from noisy observations $y(\boldsymbol{x}) = f^*(\boldsymbol{x}) + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ using maximum likelihood we are guaranteed to converge to the true solution $f(\boldsymbol{x}) = f^*(\boldsymbol{x})$ corresponding to an optimal loss $\text{Mse}[f] = \mathbb{E}_{p(\boldsymbol{x},y)}[(y(\boldsymbol{x}) - f(\boldsymbol{x}))^2] = \sigma^2$

Proof Writing $y = y(\boldsymbol{x})$, $f = f(\boldsymbol{x})$ and $f^* = f^*(\boldsymbol{x})$

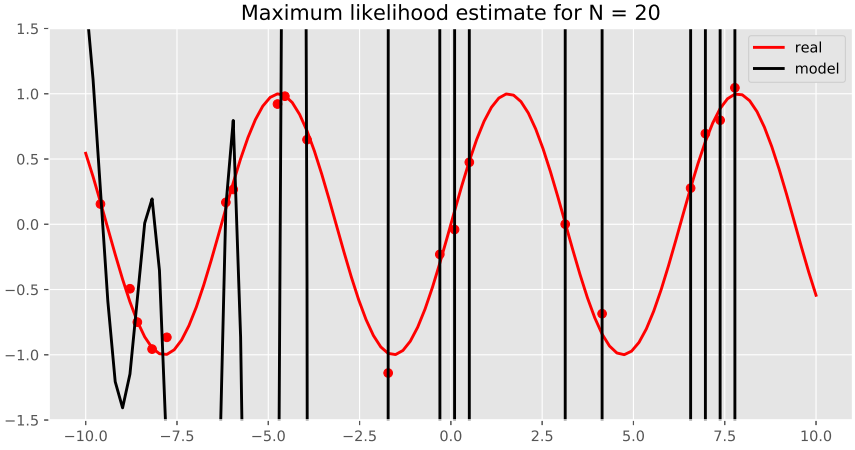
$$\begin{aligned}
\text{Mse}[f] &= \mathbb{E}_{p(\boldsymbol{x},y)}[(y - f)^2] && (1)\\
&= \mathbb{E}_{p(\boldsymbol{x},y)}[(y - f^* + f^* - f)^2] && (2)\\
&= \mathbb{E}_{p(\boldsymbol{x},y)}[(y - f^*)^2] + \mathbb{E}_{p(\boldsymbol{x},y)}[(f^* - f)^2)] + 2\mathbb{E}_{p(\boldsymbol{x},y)}[(f^* - f)(y - f^*)] && (3)\\
&= \mathbb{E}_{p(\boldsymbol{x},y)}[(y - f^*)^2] + \mathbb{E}_{p(\boldsymbol{x},y)}[(f^* - f)^2)] \quad \text{assuming } (f^* - f) \text{ and } (y - f) \text{ are uncorrelated} && (4)
\end{aligned}$$

If we have $\mathbb{E}[\epsilon] = 0$ then $\mathbb{E}_{p(y|\boldsymbol{x})}[y] = \mathbb{E}_{p(y|\boldsymbol{x})}[f^*(\boldsymbol{x}) + \epsilon] = f^*(\boldsymbol{x})$. The expected loss in this case is given as $\text{Mse}[f] = \mathbb{E}_{p(\boldsymbol{x},y)}[(y(\boldsymbol{x}) - f^*(\boldsymbol{x}))^2] = \mathbb{E}_{p(\boldsymbol{x},y)}[\epsilon^2] = \sigma^2$.

Maximum Likelihood is good when N is big...



... but bad when N is small!



Why?

Working with Finite datasets

When we work with a *finite* dataset $\mathcal{D}$ we are no longer estimating the *ideal* maximum likelihood estimate $\hat{f}(\boldsymbol{x})$ but instead approximating it with $\hat{f}(\boldsymbol{x}) \approx \tilde{f}(\boldsymbol{x}; \mathcal{D})$. Different datasets $\mathcal{D} \sim p(\mathcal{D})$ will result in different estimates $\tilde{f}(\boldsymbol{x}; \mathcal{D})$. In this case our estimate $\tilde{f}$ is no longer ideal and the expected mean squared error becomes

$$\text{Mse}[\tilde{f}] = \sigma^2 + \text{Bias}^2[\tilde{f}] + \text{Var}[\tilde{f}]$$

where σ^2 is the *observation* variance, $\text{Bias}[\tilde{f}] = \mathbb{E}_{p(x)}\{(f^* - \mathbb{E}_{p(\mathcal{D})}[\tilde{f}])^2\}$ is the bias in the estimator $\tilde{f}$ and $\text{Var}[\tilde{f}] = \mathbb{E}_{p(x)}\{(\mathbb{E}_{p(\mathcal{D})}[\tilde{f}] - \tilde{f})^2\}$ is the variance in $\tilde{f}$.

Proof As before we have $L[\tilde{f}] = \mathbb{E}_{p(x,y)}[(y - f^*)^2] + \mathbb{E}_{p(x)}[(f^* - \tilde{f})^2)]$. Expanding the second term gives,

$$\begin{aligned}
\mathbb{E}_{p(x)}\{(f^* - \tilde{f})^2\} &= \mathbb{E}_{p(x)}\{(f^* - \mathbb{E}_{p(\mathcal{D})}[\tilde{f}] + \mathbb{E}_{p(\mathcal{D})}[\tilde{f}] - \tilde{f})^2\} && (5)\\
&= \mathbb{E}_{p(x)}\{(f^* - \mathbb{E}_{p(\mathcal{D})}[\tilde{f}])^2 + (\mathbb{E}_{p(\mathcal{D})}[\tilde{f}] - \tilde{f})^2 + 2(f^* - \mathbb{E}_{p(\mathcal{D})}[\tilde{f}])(\mathbb{E}_{p(\mathcal{D})}[\tilde{f}] - \tilde{f})\} && (6)\\
&= \mathbb{E}_{p(x)}\{(f^* - \mathbb{E}_{p(\mathcal{D})}[\tilde{f}])^2\} + \mathbb{E}_{p(x)}\{(\mathbb{E}_{p(\mathcal{D})}[\tilde{f}] - \tilde{f})^2\} && (7)\\
&= \text{Bias}^2(\tilde{f}) + \text{Var}(\tilde{f}) && (8)
\end{aligned}$$

Noting that $\mathbb{E}_{p(x,y)}[(y - f^*)^2] = \sigma^2$ gives $L[\tilde{f}] = \sigma^2 + \text{Bias}^2(\tilde{f}) + \text{Var}(\tilde{f})$.

Why?

Working with Finite datasets

When we work with a *finite* dataset $\mathcal{D}$ we are no longer estimating the *ideal* maximum likelihood estimate $\hat{f}(\boldsymbol{x})$ but instead approximating it with $\hat{f}(\boldsymbol{x}) \approx \tilde{f}(\boldsymbol{x}; \mathcal{D})$. Different datasets $\mathcal{D} \sim p(\mathcal{D})$ will result in different estimates $\tilde{f}(\boldsymbol{x}; \mathcal{D})$. In this case our estimate $\tilde{f}$ is no longer ideal and the expected mean squared error becomes

$$\text{Mse}[\tilde{f}] = \sigma^2 + \text{Bias}^2[\tilde{f}] + \text{Var}[\tilde{f}]$$

where σ^2 is the *observation* variance, $\text{Bias}[\tilde{f}] = \mathbb{E}_{p(x)}\{(f^* - \mathbb{E}_{p(\mathcal{D})}[\tilde{f}])^2\}$ is the bias in the estimator $\tilde{f}$ and $\text{Var}[\tilde{f}] = \mathbb{E}_{p(x)}\{(\mathbb{E}_{p(\mathcal{D})}[\tilde{f}] - \tilde{f})^2\}$ is the variance in $\tilde{f}$.

Note:

- In the limit as $N \to \infty$ we have $p(\mathcal{D}) = \delta(\mathcal{D} - \mathcal{D}^*)$ and $\tilde{f} \to \hat{f} = f^*(\boldsymbol{x})$: this is why maximum likelihood is good for large N!

Regularisation

From the perspective of Occams Razor

- When N is small the maximum likelihood solution *overfits* to the data (good at predicting training data but bad at generalising)
- Solution = *Occam's Razor* $\implies$ "Prefer the simpler model"
- Could make the model simpler by reducing the dimensionality of the feature space M and so limiting the number of parameters $\boldsymbol{w} \in \mathbb{R}^M$ in our model. But difficult to optimise over...
- Instead penalise the norm of the weights $\|\boldsymbol{w}\|$ and consider $\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w})$ with

$$\mathcal{L}(\boldsymbol{w}) = E(\boldsymbol{w}) + \lambda \|\boldsymbol{w}\|$$

  where $E(\boldsymbol{w})$ is as defined before.
- For $\|\boldsymbol{w}\| = \|\boldsymbol{w}\|_2^2$ we can arrive at $\mathcal{L}(\boldsymbol{w})$ considering a MAP estimate for $\boldsymbol{w}$ with a prior $p(\boldsymbol{w}) = \text{Norm}(\boldsymbol{w}|0, \beta^{-1}\boldsymbol{I})$ and in this case $\boldsymbol{w}^* = (\lambda \boldsymbol{I} + \boldsymbol{\Phi}^\top \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^\top \boldsymbol{y}$.
- Could also consider $\|\boldsymbol{w}\| = \|\boldsymbol{w}\|_1$. In this case the weights in $\boldsymbol{w}$ are encouraged to be sparse (eg. identical to 0).

Regularisation

From the perspective of Bias and Variance

- The *expected mean squared error* for a non-ideal estimator is given as

$$\text{Mse}[\tilde{f}] = \sigma^2 + \text{Bias}^2[\tilde{f}] + \text{Var}[\tilde{f}]$$

  where σ^2 is the *observation* variance, $\text{Bias}[\tilde{f}] = \mathbb{E}_{p(x)}\{(f^* - \mathbb{E}_{p(\mathcal{D})}[\tilde{f}])^2\}$ is the bias in the estimator $\tilde{f}$ and $\text{Var}[\tilde{f}] = \mathbb{E}_{p(x)}\{(\mathbb{E}_{p(\mathcal{D})}[\tilde{f}] - \tilde{f})^2\}$ is the variance in $\tilde{f}$.

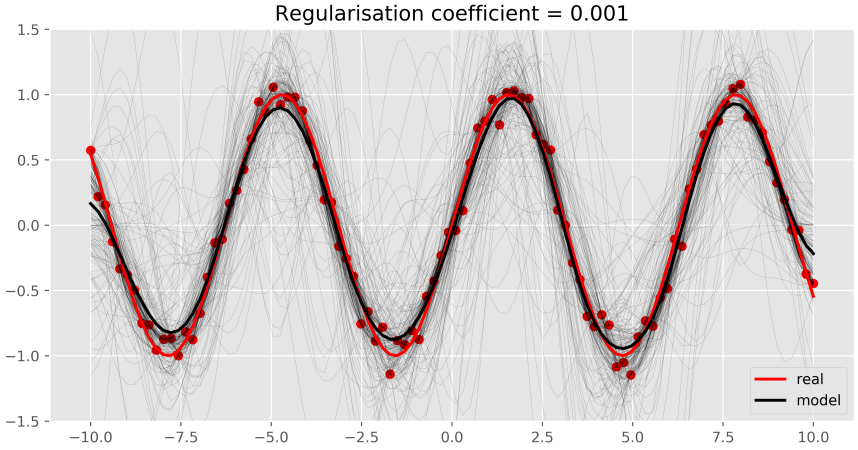
- *Observation*: The maximum-likelihood estimate is just one of a host of estimates we could use for $\tilde{f}$ and the above decomposition remains true independent of the loss function we actually optimise to find $\tilde{f}$
- *Idea*: "design a new loss function $\mathcal{L}$ which results in smaller mean squared error $L[\tilde{f}]$ when N is small"
- By introducing regularisation

$$\mathcal{L}(\boldsymbol{w}) = E(\boldsymbol{w}) + \lambda\|\boldsymbol{w}\|$$

  we increase the $\text{Bias}[\tilde{f}]$ but *reduce* $\text{Var}[\tilde{f}]$ for small N. For a correct weighting λ this *can* result in smaller expected mean sqaured error $\text{Mse}[\tilde{f}] = \mathbb{E}_{p(\boldsymbol{x})}[(\tilde{f}(\boldsymbol{x}) - f^*(\boldsymbol{x}))^2]$

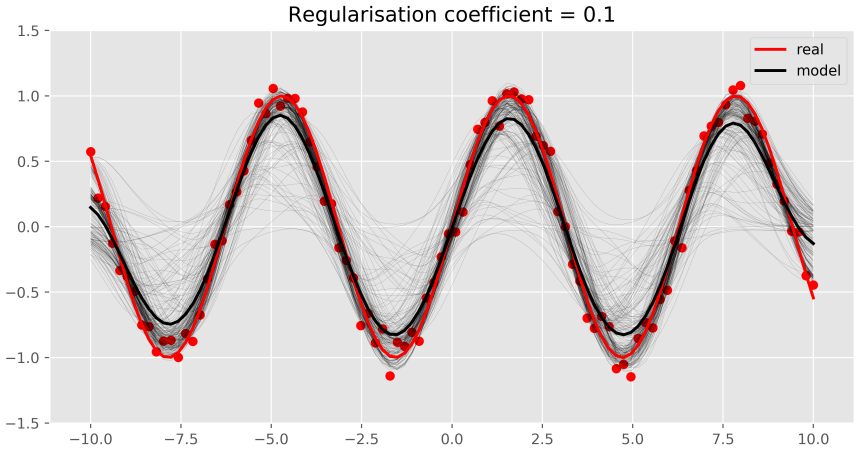
Regularisation

High Variance - Low Bias...



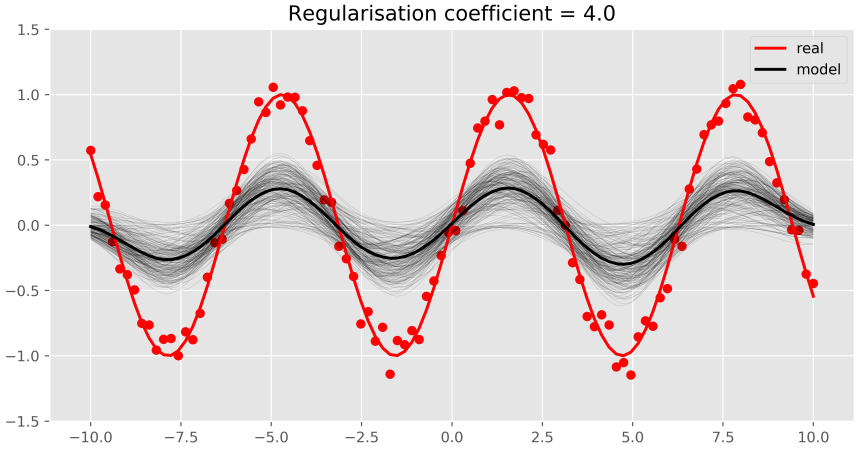
Regularised Linear Regression

Medium Variance - Medium Bias...

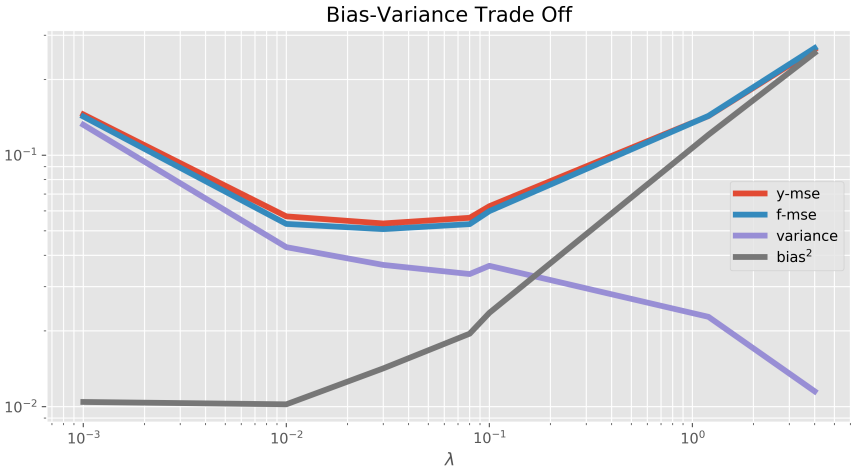


Regularised Linear Regression

Low Variance - High Bias...



Regularised Linear Regression



Choosing λ...

- In reality we do not have access to the true function f^* and so evaluating Bias$[\tilde{f}]$ is impossible
- We *can* estimate the expected mean squared error over a test set held out from training and use this to tune λ (cross validation)
- But relies on splitting our dataset reducing the size of N and increasing our chances of overfitting
- Is there a better way?

Overview

Bayesian Linear Regression

Bayesian Linear regression offers a valuable alternative...

1. Assume that the weights are no longer fixed but instead a *random variable* $\boldsymbol{w} \sim p(\boldsymbol{w}|\beta)$ with *prior* $p(\boldsymbol{w}|\beta) = \text{Norm}(\boldsymbol{w}|0, \beta^{-1}\boldsymbol{w})$ (shape of prior governed by the prior *precision* β).
2. As before assume the *likelihood* of our observations is given as $p(\boldsymbol{y}|\boldsymbol{w}, \alpha) = \prod_{n=1}^{N} \text{Norm}(y|\boldsymbol{w}^\top \boldsymbol{x}, \alpha^{-1})$
3. Calculate the posterior distribution $p(\boldsymbol{w}|\boldsymbol{y}, \alpha, \beta)$. In this case we have

$$p(\boldsymbol{w}|\boldsymbol{y}, \alpha, \beta) = \text{Norm}(\boldsymbol{w}|\boldsymbol{m}_N, \boldsymbol{S}_N) \quad \text{with} \quad \boldsymbol{m}_N = \alpha \boldsymbol{S}_N \boldsymbol{\Phi}^\top \boldsymbol{y} \quad \text{and} \quad \boldsymbol{S}_N^{-1} = \beta \boldsymbol{I} + \alpha \boldsymbol{\Phi}^\top \boldsymbol{\Phi}$$

4. Use the *posterior-predictive* distribution

$$p(y|\boldsymbol{y}, \alpha, \beta) = \int p(y|\boldsymbol{w}) p(\boldsymbol{w}|\boldsymbol{y}, \alpha, \beta) d\boldsymbol{w} = \text{Norm}(y|\boldsymbol{m}_N^\top \boldsymbol{\phi}, \alpha^{-1} + \boldsymbol{\phi}^\top \boldsymbol{S}_N \boldsymbol{\phi})$$

to predict *new* y averaging across *all* possible values of weights $\boldsymbol{w}$.

Bayesian Linear Regression

What about the hyper-parameters α and β?

- Could assume conjugate priors and add them to the latents adopting a full Bayesian approach BUT results in intractable posterior distribution for $\boldsymbol{w}$
- Instead consider maximum likelihood estimators using $\hat{\alpha}, \hat{\beta} = \arg\max_{\alpha,\beta} p(\boldsymbol{y}|\alpha,\beta)$ maximising the *marginal likelihood* $p(\boldsymbol{y}|\alpha,\beta) = \int p(\boldsymbol{y}|\boldsymbol{w},\alpha)p(\boldsymbol{w}|\beta)d\boldsymbol{w} \rightarrow$ Known as *Empirical Bayes* or *Type II Maximum Likelihood*
- Marginal likelihood is in this case given as

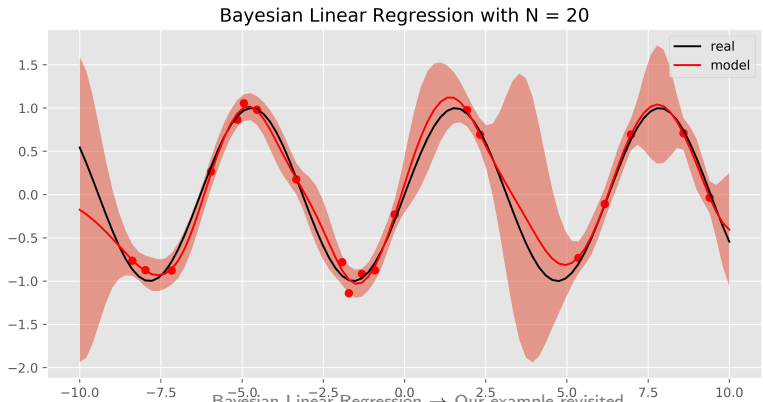
$$\log p(\boldsymbol{y}|\alpha,\beta) = \frac{M}{2}\log\beta + \frac{N}{2}\log\alpha - \frac{\alpha}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{m}_N\|^2 - \frac{\beta}{2}\boldsymbol{m}_N^\top \boldsymbol{m}_N - \frac{1}{2}|\boldsymbol{S}_N^{-1}| - \frac{N}{2}\log 2\pi$$

- Differentiating and setting to 0 an iterative scheme emerges for estimating $\hat{\alpha}$ and $\hat{\beta}$...
  1. Choose starting values for α and β
  2. Calculate $\boldsymbol{\lambda} = \text{eig}(\boldsymbol{\Phi}^\top\boldsymbol{\Phi})$
  3. Calculate $\boldsymbol{S}_N = (\beta\boldsymbol{I} + \alpha\boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}$
  4. Calculate $\boldsymbol{m}_N = \alpha\boldsymbol{S}_N\boldsymbol{\Phi}^\top\boldsymbol{y}$
  5. Calculate $\gamma = \sum_i \frac{\lambda_i}{\lambda_i+\beta}$
  6. Update $\alpha = \frac{(N-\gamma)}{\sum_n (y_n - \boldsymbol{m}_N^\top\boldsymbol{\phi}(\boldsymbol{x}_n))^2}$
  7. Update $\beta = \frac{\gamma}{\boldsymbol{m}_N^\top\boldsymbol{m}_N}$
  8. Repeat steps 3-7 until convergence

Bayesian Linear Regression

Our earlier example revisited...

- Bayesian linear regression *automatically* results in a regularised solution as a result of *marginalising* out the uncertainty in $\boldsymbol{w}$
- Hyper-parameters α and β are estimated from the *training* set - all available data used for training



What is γ?

- Noting that $\gamma = \sum_i \frac{\lambda_i}{\lambda_i+\beta}$ it is equal to the sum of terms $\frac{\lambda_i}{\lambda_i+\beta}$.
- As $\mathbf{\Phi}^\top \mathbf{\Phi}$ is positive definite matrix we have $\lambda_i > 0$ and so $0 \leq \frac{\lambda_i}{\lambda_i+\beta} \leq M$
- Two cases:
  1. $\lambda_i >> \beta$: the estimate for w_i will be close to its ML estimate and $\frac{\lambda_i}{\lambda_i+\beta} \approx 1$
  2. $\lambda_i << \beta$: The estimate for $w_i \approx 0$ (the prior value) and $\frac{\lambda_i}{\lambda_i+\beta} \approx 0$
- The quantity γ has an intuitive interpretation as the **effective number of parameters** in the model.
- γ parameters are set by ML and $M - \gamma$ are set from the prior.
- This can be seen in the estimate for the likelihood variance $\alpha^{-1} = \frac{1}{N-\gamma} \sum_n (y_n - \boldsymbol{m}_n^\top \boldsymbol{\phi}(\boldsymbol{x}_m))^2$