Lecture 4: The Multivariate Gaussian & Bayesian Regression - 03/02/20

Revision: Bayes Rule

Bayes Rule allows us to reason about random variables:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \propto p(y|x)p(x)$$

Conjugacy means that prior and posterior have the same form, e.g.

$$Beta(\mu|a+m, b+l) \propto \mu^m(1-\mu)^l \times Beta(\mu|a,b)$$

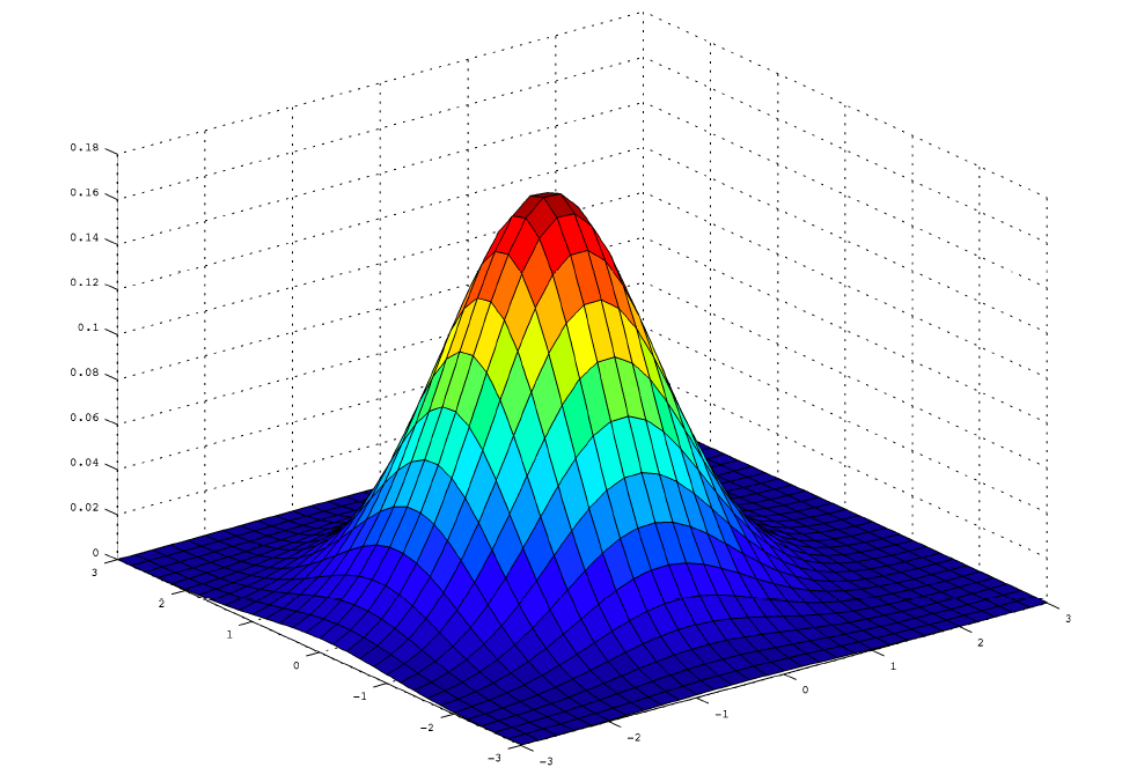
Bayesian Inference on Regression Weights

By combining the ***linear model regression*** and ***Bayes rule***, we can reason about ***regression weights***, $\mathbf{w}$

$$p(\mathbf{w}|D,\theta) = \frac{p(D|\mathbf{w})p(\mathbf{w}|\theta)}{p(D)} \propto p(D|\mathbf{w})p(\mathbf{w}|\theta)$$

with θ as the initial parameters

The Multivariate Gaussian Distribution



- a multi-dimensional ***generalisation*** of the Gaussian
- data-points are $D > 1$ dimensional vectors, e.g. $\mathbf{x}_n \in \mathbb{R}^D$

The probability density function (p.d.f) is:

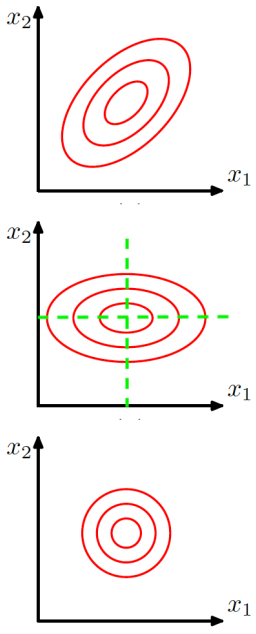
$$N(\mathbf{x}|\mu,\mathbf{\Sigma}) = \frac{1}{(2\pi)^{\frac{D}{2}}}\frac{1}{|\mathbf{\Sigma}|^{\frac{1}{2}}}\exp[-\frac{1}{2}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)]$$

with

- $\mu = E[\mathbf{x}]$ is the ***vector mean***
- $\mathbf{\Sigma}$ is the ***covariance matrix***
- $|\mathbf{\Sigma}|$ is its ***scalar determinant***
- $\mathbf{\Sigma}^{-1}$ is its ***inverse matrix***

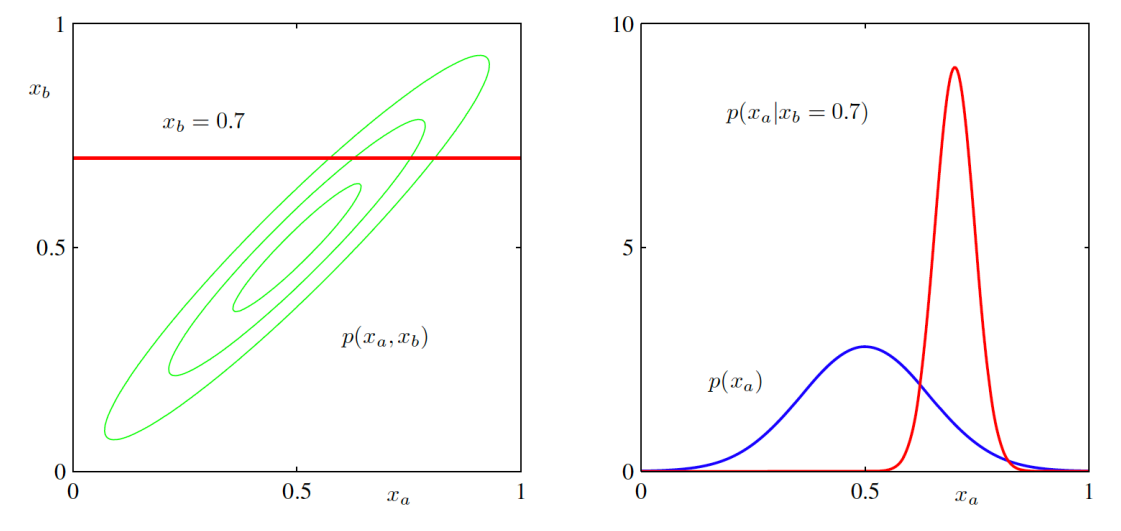
> The ***Covariance Matrix*** $\mathbf{\Sigma}$ is the multi-dimensional analog of the variance. It is ***symmetric***, and influences the ***shape*** and ***dispersion*** of the distribution.
>
> $$[\mathbf{\Sigma}]_{i,j} = \text{cov}[x_i, x_j]$$

> The Mahalanobis Distance Δ is the analog of $\frac{x-\mu}{\sigma}$ in unidimensional Gaussian Distribution under the multivariate case. It is the distance of a data point from the mean value generalised by the co-variance.
>
> $$\Delta^2 = (\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)$$

> ***Note that***: two data points that have the same Mahalanobis Distance from the data mean have the same probability density. All of these points form the ***Equiprobability Surface***
>
> - **Equiprobability surfaces** are those with constant Δ^2 (contour lines in 2d)
> - General $\mathbf{\Sigma}$ has ellipsoid equiprobability surfaces
> - If $\mathbf{\Sigma}$ is diagonal then equiprobability surfaces are axis aligned ellipsoids
> - If $\mathbf{\Sigma} = \boldsymbol{I}$ (identity mtx) then Δ^2 is the Euclidean distance
> - Equiprobability surfaces are spherical
>
> 
>

H3 Conditionals and Marginals

Conveniently for Multivariate Gaussian, many conditionals and marginals are also ***Gaussian***



Block Matrix Notation

Block matrix notation expresses larger arrays in terms of smaller ones

For vectors:

$$\mathbf{x} = (x_1, x_2, \ldots, x_k)^T = \begin{pmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{pmatrix} = \left(\mathbf{x}_a^T \mathbf{x}_b^T \right)^T$$

Where $\mathbf{x}_a = (x_1, x_2, \ldots, x_j)^T$ and $\mathbf{x}_b = (x_{j+1}, x_{j+2}, \ldots, x_k)^T$

For matrices:

$$\mathbf{\Sigma} = \begin{pmatrix} \mathbf{\Sigma}_{aa} & \mathbf{\Sigma}_{ab} \\ \mathbf{\Sigma}_{ba} & \mathbf{\Sigma}_{bb} \end{pmatrix}$$

Where $\mathbf{\Sigma}_{aa}, \mathbf{\Sigma}_{ab}, \mathbf{\Sigma}_{ba}, \mathbf{\Sigma}_{bb}$ are the submatrices. $\mathbf{\Sigma}_{aa}$ and $\mathbf{\Sigma}_{bb}$ are ***square*** matrices.

Just like the unidimensional β, ***Precision Matrix***:

$$\mathbf{\Lambda}^{-1} \equiv \mathbf{\Sigma}$$

> ***Note that***: $\mathbf{\Lambda}_{aa}^{-1} \neq \mathbf{\Sigma}_{aa}$

Partitioned Gaussians

For any Gaussian, $N(\mathbf{x}|\mu, \mathbf{\Lambda}^{-1})$ with $\mathbf{\Lambda^{-1}} = \mathbf{\Sigma}$, $\mathbf{x} = \begin{pmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{pmatrix}$, $\mu = \begin{pmatrix} \mu_a \\ \mu_b \end{pmatrix}$, $\mathbf{\Lambda} = \begin{pmatrix} \mathbf{\Lambda_{aa}} & \mathbf{\Lambda_{ab}} \\ \mathbf{\Lambda_{ba}} & \mathbf{\Lambda_{bb}} \end{pmatrix}$

> The ***conditional distribution*** of $\mathbf{x}_a$ given $\mathbf{x}_b$ is ***Gaussian***
>
> $$p(\mathbf{x}_a|\mathbf{x}_b) = N(\mathbf{x}|\mu_{a|b}, \mathbf{\Lambda}_{a|b}^{-1})$$
>
> Where $\mu_{a|b} = \mu_a - \mathbf{\Lambda}_{aa}^{-1}\mathbf{\Lambda}_{ab}(\mathbf{x}_b - \mu_b)$ and $\mathbf{\Lambda}_{a|b} = \mathbf{\Lambda}_{aa}$

> The ***marginal distribution*** of $\mathbf{x}_a$ is also ***Gaussian***
>
> $$p(\mathbf{x}_a) = N(\mathbf{x}|\mu_a, \mathbf{\Lambda}_a^{-1})$$
>
> where $\mathbf{\Lambda}_a = \mathbf{\Lambda}_{aa}$

Dependent Gaussian

Similarly nice properties hold when we combine Gaussians. For instance, assume we have marginal and conditional Gaussian distributions:

$$p(\mathbf{x}) = N(\mathbf{x}|\mu, \mathbf{\Lambda}^{-1})$$
$$p(\mathbf{y}|\mathbf{x}) = N(\mathbf{y}|\mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{L}^{-1})$$

In which the ***mean*** for $p(\mathbf{y}|\mathbf{x})$ is some matrix linear operations $\mathbf{A}\mathbf{x} + \mathbf{b}$ involving $\mathbf{x}$ and the some ***precision matrix*** $\mathbf{L}$.

> The ***marginal distribution*** of $\mathbf{y}$ is Gaussian
>
> $$p(\mathbf{y}) = N(\mathbf{y}|\mu_y, \mathbf{\Lambda}_y^{-1}) = N(\mathbf{y}|\mathbf{A}\mu + \mathbf{b}, \mathbf{L}^{-1} + \mathbf{A}\mathbf{\Lambda}^{-1}\mathbf{A}^{\mathbf{T}})$$

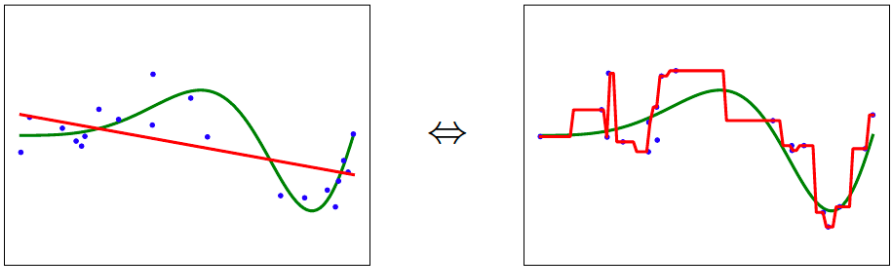
> The ***conditional distribution*** of $\mathbf{x}$ given $\mathbf{y}$ is Gaussian
>
> $$p(\mathbf{x}|\mathbf{y}) = N(\mathbf{x}|\mu_{x|y}, \mathbf{\Lambda}_{x|y}^{-1}) = N(\mathbf{x}|\mathbf{S}[\mathbf{A}^T\mathbf{L}(\mathbf{y} - \mathbf{b}) + \mathbf{\Lambda}\mu], \mathbf{S})$$
>
> where $\mathbf{S} = (\mathbf{\Lambda} + \mathbf{A}^T\mathbf{L}\mathbf{A})^{-1}$

Generalisation

- The ***real aim*** of supervised learning is to perform well on ***test data***
- Choosing the values for the parameters $\mathbf{w}$ that minimise the loss function on the training data is not always the best policy as it may lead to ***over-fitting***
- Want to model true regularities in data, and ignore noise. . . but the learning machine doesn't know ***which regularities are real, and which are quirks of the current training examples*** .

> How can we ensure the machine generalises correctly to new data, ignoring the training data and fitting to the trend?

Goodness of Fit vs. Model Complexity



- Intuitively, we only say a model generalises well, if it explains data surprisingly well given its complexity.
- If the model has as many degrees of freedom as the data, it can fit the data perfectly but so what?
- Lots of theory about how to measure model complexity and control it to optimise generalisation.
  (We are not going to cover this.)

But, the Bayesian framework allows for a natural solution. . .

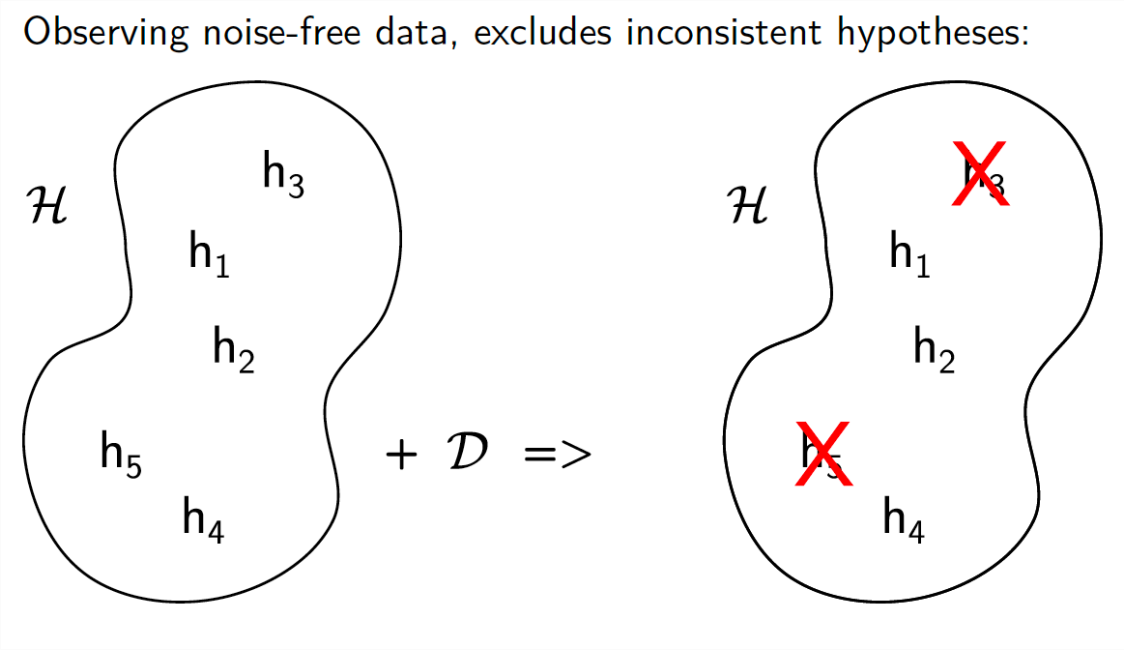
H2 Hypothesis Space

One way to think about a supervised learning machine is as a device that explores a ***hypothesis space***:

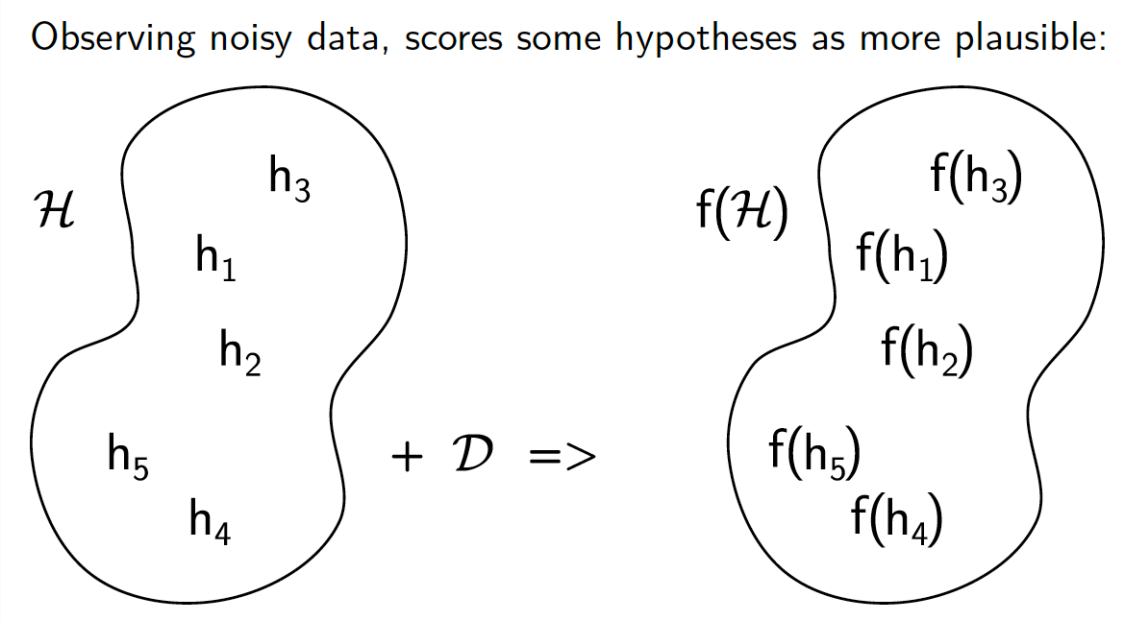
- Each ***setting of the parameters*** is a ***different hypothesis*** about the function that maps inputs to outputs.
- If data is ***noise-free*** , each training example rules out a region of hypothesis space.
- If data is ***noisy*** , each training example is scored according to how plausible it is given the (validation?) data.

> The ***art*** of supervised machine learning then becomes:
>
> - Decide how to ***represent inputs and outputs***
> - Select a ***hypothesis space*** – ***powerful*** enough to represent input → output relationship, ***simple*** enough to be searched

H3 Hypothesis Space: Noise-Free Data



H3 Hypothesis Space: Noisy Data



> The likelihood functions are one way of measuring the plausibility

H2 Optimising Plausibility

Having formulated a ***plausibility function***, one can adjust the parameters to maximise the plausibility.

- This allows optimisation to be separated from the function that is being optimised
- maximum-likelihood/least-squares does this
- regularised least-squares also does this

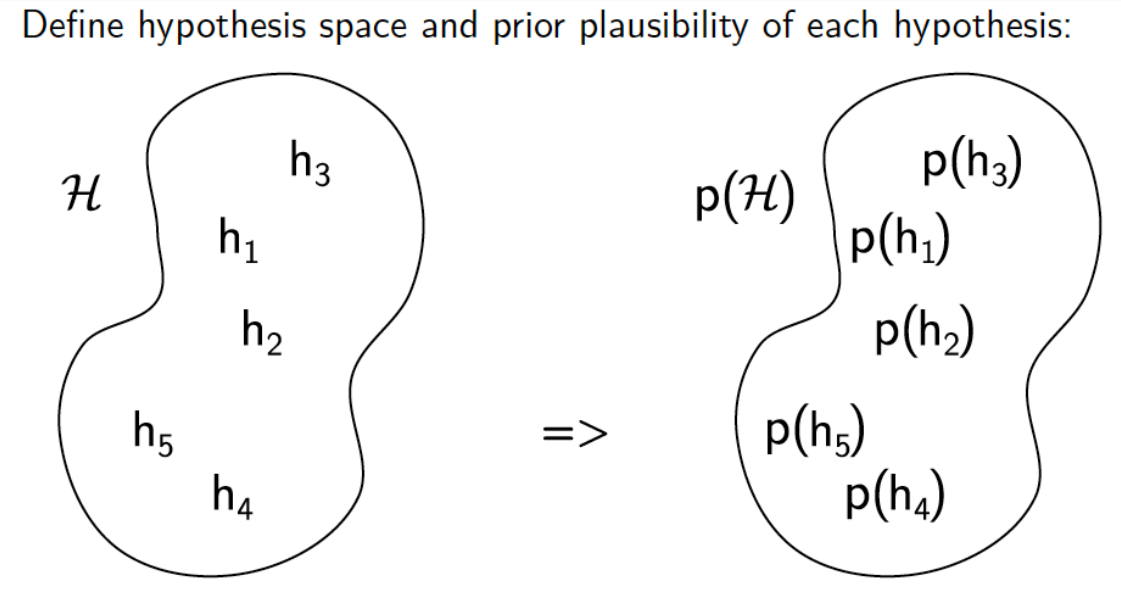
Bayesians do not ***search for a single set of parameter values*** that are most plausible. Instead they:

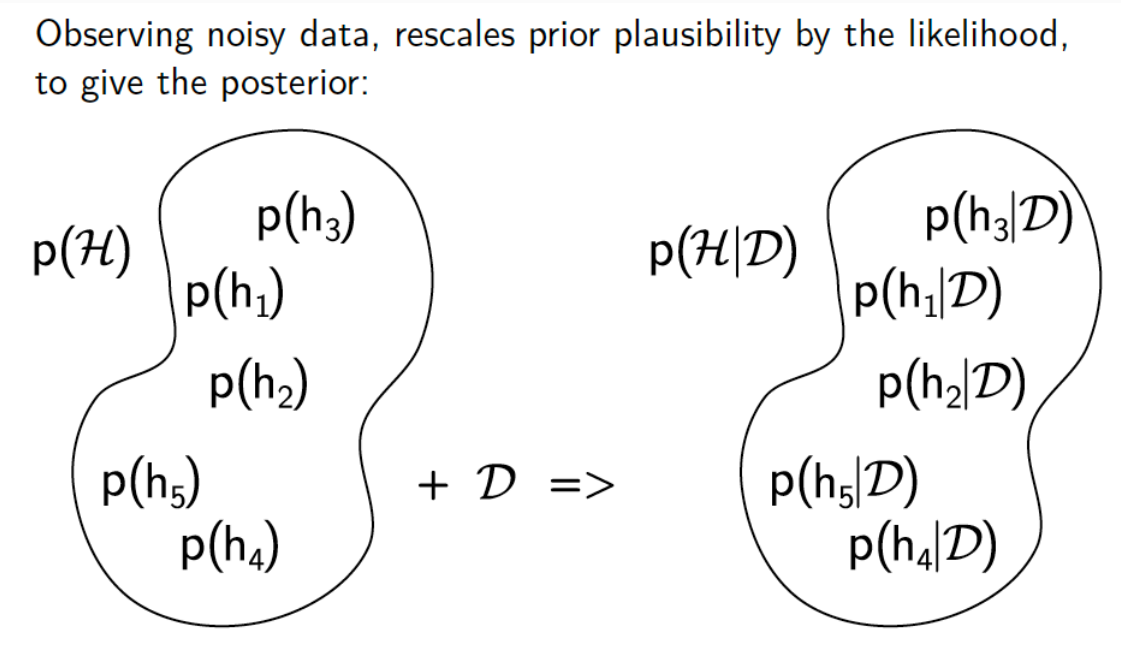
- define a probabilistic model
- choose a prior distribution over parameter values
- the prior represents initial belief

- combine prior with training data to compute a posterior distribution over the whole hypothesis space
- the posterior captures updated belief

> Ideally, predictions should be made using this posterior

H3 Hypothesis Space: Bayesian

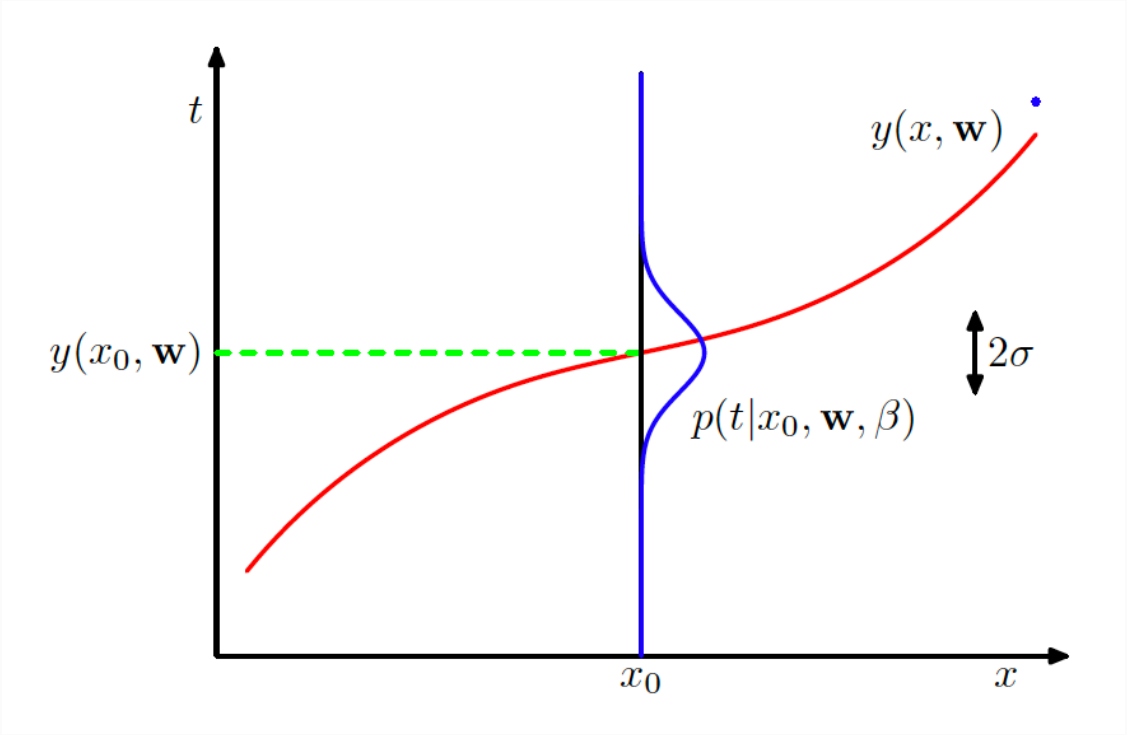




> weights prior plausibility by the likelihood to give the posterior (when normalised)

H2 Bayesian Inference for Linear Models

Consider the following data, fitting a linear model to the data



- N Inputs $\mathbf{x}_n$
- N targets t_n
- $M-1$ basis functions ϕ_j
- Want to fit prediction function: $y(\mathbf{x}; \mathbf{w}) = \phi(\mathbf{x})^T \mathbf{w}$
- Likelihood of single target: $p(t|\mathbf{x}, \mathbf{w}, \beta) = N(t|\phi(\mathbf{x})^T \mathbf{w}, \beta^{-1})$
- β is the known noise precision

For Bayesian inference, we will also need a ***prior*** on $\mathbf{w}$. We can make this Multivariate Gaussian

$$p(\mathbf{w}) = N(\mathbf{w}|\mathbf{m}_0, \mathbf{S}_0)$$

with mean $\mathbf{m}_0$ and covariance matrix $\mathbf{S}_0$.

The ***likelihood*** can also be viewed as a ***Multivariate Gaussian***:

$$\begin{aligned} p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) &= \prod_{n=1}^{N} N(t_n|\mathbf{w}^T \phi(\mathbf{x}_n), \beta^{-1}) \\ &= N(\mathbf{t}|\mathbf{\Phi}\mathbf{w}, \beta^{-1}\mathbf{I}) \end{aligned}$$

where $\mathbf{w}^T \phi(\mathbf{x}_n)$ is the prediction of the prediction of the function of the n-th data point, with precision β. $\mathbf{I}$ is the identity matrix, thus the ***covariance matrix*** is $\beta^{-1}\mathbf{I}$.

> ***Note that***: The covariance matrix is diagnol, which suggest that for all ***equiprobability spaces*** are ***spherical***, or "***isotropic***". This also suggests that *there's no covariance between different dimensions, so they are independently sampled or they are independent variables.*

$\prod_{n=1}^{N}$ can be interpreted as an isotropic multivariate Gaussian with N dimensions, so one dimension for every data point. Thus, the taget vector $\mathbf{t}$ is a single random variable with N elements in it.

Using the rule of the conditional distribution of dependent Gaussian $p(\mathbf{x}|\mathbf{y}) = N(\mathbf{x}|\mu_{x|y}, \mathbf{\Lambda}_{x|y}^{-1})$, the posterior over $\mathbf{w}$ given $\mathbf{t}$ will also be Multivariate Gaussian

$$p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \beta) = N(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$$

> ***Conclusion***
>
> For a linear regression model with prior on weights:
>
> $$p(\mathbf{w}) = N(\mathbf{w}|\mathbf{m}_0, \mathbf{S}_0)$$
>
> Assuming Gaussian noise with known precision β, e.g.
>
> $$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = N(\mathbf{t}|\mathbf{\Phi}\mathbf{w}, \beta^{-1}\mathbf{I})$$
>
> Then the posterior distribution over weights is:
>
> $$p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \beta) = N(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$$
>
> where $\mathbf{m}_N = \mathbf{S}_N(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta\mathbf{\Phi}^T\mathbf{t})$ and $\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \beta\mathbf{\Phi}^T\mathbf{\Phi}$
>
> > ***Notes on*** $\mathbf{m}_N$: clearly, the posterior mean will be a trade-off between the prior mean and the maximum-likelihood solution. The more data we have, the more certain we are going to be about the optimal parameters.
> >
> > ***Notes on*** $\mathbf{S}_N^{-1}$: clearly, the magnitude of the inverse of covariance matrix or the precison matrix gets greater as the more data been used in training, which suggests that the covariance matrix gets smaller. So the spread or the uncertainty is going to fall as we add more data.
> >
> > *These characteristics are identical as that in the univariate Gaussian.*

H2 Example: Bayesian Univariate Linear Regression

Generate data from function:

$$f(x, \mathbf{a}) = a_0 + a_1 x + \epsilon$$

Where $\epsilon \sim N(\epsilon|0, \beta^{-1})$ and $\beta = (\frac{1}{0.2})^2 = 25$

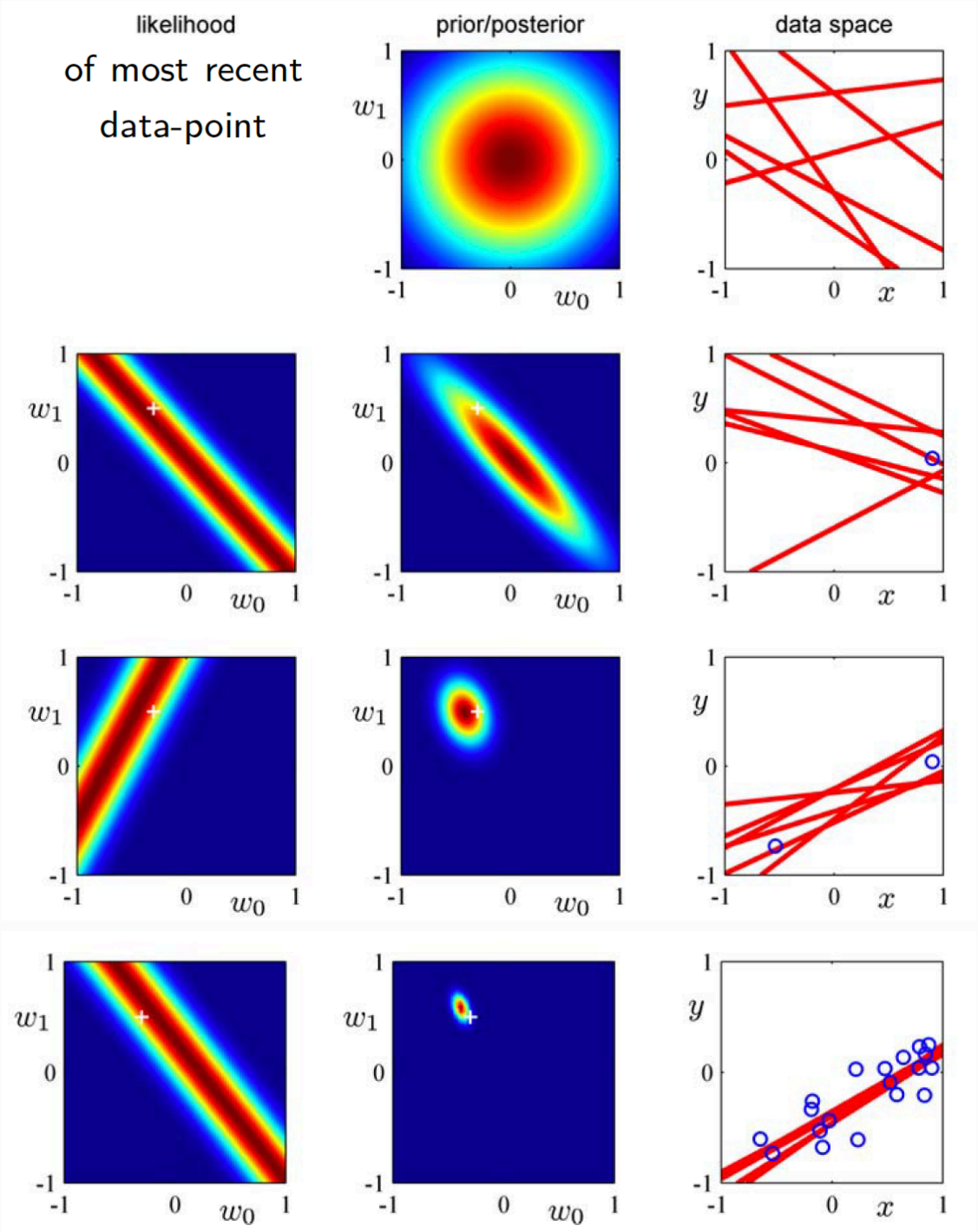
We are trying to fit this data with regression function:

$$y(x, \mathbf{w}) = w_0 + w_1 x$$

Assume, we know β and have a prior

$$p(\mathbf{w}) = N(\mathbf{w}|0, \alpha\mathbf{I})$$

With $\alpha = 2$. So covariane matrix is $\mathbf{S}_0 = \alpha\mathbf{I} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$



> ***Note that***: the white cross is the ***unknown optumal parameter setting***, which is the answer.

H2 Maximum a-Posteriori (MAP) Regression

Consider an isotropic prior centred at the origin:

$$p(\mathbf{w}) = N(\mathbf{w}|0, \alpha^{-1}\mathbf{I})$$

Then use $p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \beta) = N(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$ where $\mathbf{m}_N = \mathbf{S}_N(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta\mathbf{\Phi}^T\mathbf{t})$ and $\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \beta\mathbf{\Phi}^T\mathbf{\Phi}$, with $\mathbf{m}_0 = 0$ and $\mathbf{S}_0 = \alpha^{-1}\mathbf{I}$

$$\mathbf{m}_N = \mathbf{S}_N\beta\mathbf{\Phi}^T\mathbf{t}$$
$$\mathbf{S}_N^{-1} = \alpha\mathbf{I} + \beta\mathbf{\Phi}^T\mathbf{\Phi}$$

the posterior is:

$$p(\mathbf{w}|\mathbf{t}, \Phi, \mathbf{m}_0, \mathbf{S}_0) = N(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$$

Using the p.d.f of multivariate Gaussian $N(\mathbf{x}|\mu, \mathbf{\Sigma}) = \frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{|\mathbf{\Sigma}|^{\frac{1}{2}}} \exp[-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)]$:

$$\begin{aligned} N(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N) &= \frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{|\mathbf{S}_N|^{\frac{1}{2}}} \exp[-\frac{1}{2}(\mathbf{w}-\mathbf{m}_N)^T \mathbf{S}_N^{-1}(\mathbf{w}-\mathbf{m}_N)] \\ &= \frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{|\mathbf{S}_N|^{\frac{1}{2}}} \exp[-\frac{1}{2}(\mathbf{w}-\mathbf{S}_N\beta\mathbf{\Phi}^T\mathbf{t})^T (\alpha\mathbf{I} + \beta\mathbf{\Phi}^T\mathbf{\Phi})(\mathbf{w}-\mathbf{S}_N\beta\mathbf{\Phi}^T\mathbf{t})] \end{aligned}$$

> ***to be implemented***

and the log of this posterior is

$$\ln p(\mathbf{w}|\mathbf{t}) = -\frac{\beta}{2}\sum_{n=1}^{N}(t_n - \mathbf{w}^T\Phi(\mathbf{x}_n))^2 - \frac{\alpha}{2}\mathbf{w}^T\mathbf{w} + \text{const.}$$

Finding the maximum of this:

$$\nabla_{\mathbf{w}} \ln p(\mathbf{w}|\mathbf{t}) = 0$$

> ***Note that***: MAP regression with an isotropic prior is equivalent to ridge regression with $\lambda = \frac{\alpha}{\beta}$. So maximising the posterior is equivalent to minimising the squared ridge regression loss.
>
> ***Previously, for ridge regression, a lot of experiments is required to find what the best λ was, even then it was not entirely sure. Where as in MAP regression, if we knew β and we choose α a priori, then we don't have to do model selection on λ.***

H2 Predictive Distribution: Bayesian

We are less interested in $\mathbf{w}$ than in the ***posterior predictions***. We can integrate out $\mathbf{w}$ (marginalise) to get the predictive distribution for the target t for some new input $\mathbf{x}$:

$$p(t|\mathbf{x}, D, \alpha, \beta) = \int p(t|\mathbf{x}, \mathbf{w}, \beta)p(\mathbf{w}, D, \alpha, \beta)\, d\mathbf{w}$$

Where D is our training data.

The likelihood is defined in our model:

$$p(t|\mathbf{x}, \mathbf{w}, \beta) = N(t|\phi(\mathbf{x})^T\mathbf{w}, \beta^{-1})$$

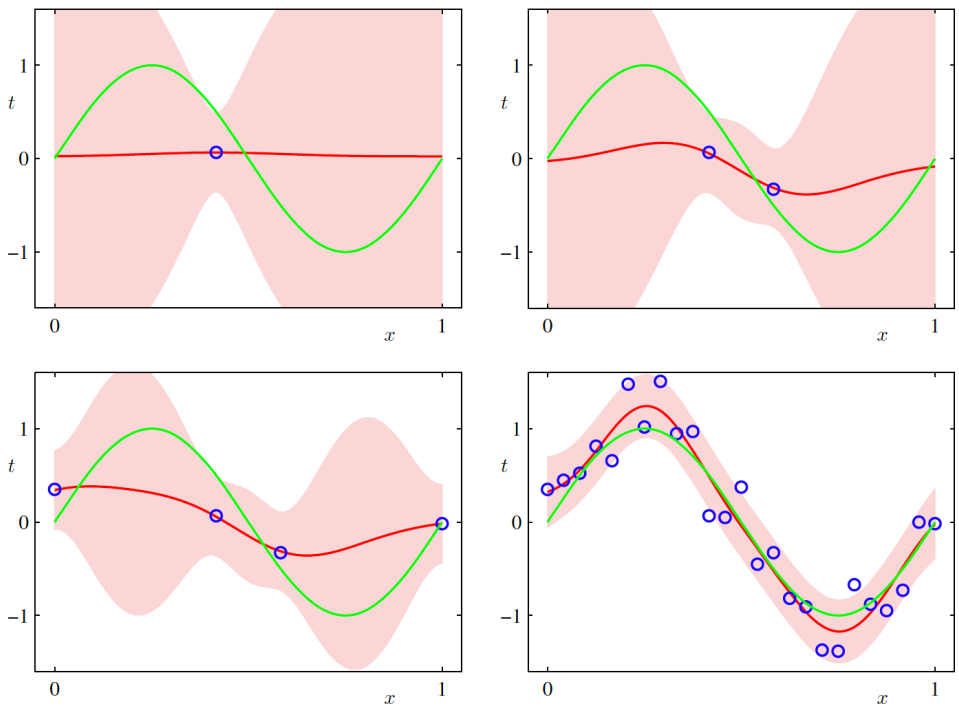
The posterior over weights:

$$p(\mathbf{w}|D, \alpha, \beta) = N(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$$

> Using the property of combined Gaussian $p(\mathbf{y}) = N(\mathbf{y}|\mu_y, \mathbf{\Lambda}_y^{-1})$ gives the ***predictive distribution***:
>
> $$p(t|\mathbf{x}, D, \alpha, \beta) = N(t|\phi(\mathbf{x})^T\mathbf{m}_N, \sigma_N^2(\mathbf{x}))$$

where the mean prediction is the feature vector of our unseen data point multipied by the mean of the posterior, the variance $\sigma_N^2(\mathbf{x}) = \frac{1}{\beta} + \phi(\mathbf{x})^T \mathbf{S}_N \phi(\mathbf{x})$.

> ***Note that***:
>
> The uncertainty in t depends on two terms:
>
> - first term: the noise in the data
> - second term: from uncertainty about $\mathbf{w}$, which is input dependent



Predictive distribution for model with 9 Gaussian basis functions fitting synthetic data ($t = \sin(x) + \varepsilon$).