# Lecture 17. Bayesian regression

COMP90051 Statistical Machine Learning

Semester 2, 2020
Lecturer: Ben Rubinstein

THE UNIVERSITY OF MELBOURNE

# This lecture

- Uncertainty not captured by point estimates
- Bayesian approach preserves uncertainty
- Sequential Bayesian updating
- Conjugate prior (Normal-Normal)
- Using posterior for Bayesian predictions on test

# Training == optimisation (?)

Stages of learning & inference:

- Formulate model

**Regression**

$$p(y|\mathbf{x}) = \text{sigmoid}(\mathbf{x}'\mathbf{w})$$

$$p(y|\mathbf{x}) = \text{Normal}(\mathbf{x}'\mathbf{w}; \sigma^2)$$

- Fit parameters to data

Like a penalty term

$$\hat{\mathbf{w}} = \text{argmax}_{\mathbf{w}}\, p(\mathbf{y}|\mathbf{X}, \mathbf{w})p(\mathbf{w})$$

*ditto*

- Make prediction

$$p(y_*|\mathbf{x}_*) = \text{sigmoid}(\mathbf{x}_*'\hat{\mathbf{w}})$$

$$E[y_*] = \mathbf{x}_*'\hat{\mathbf{w}}$$

$\hat{\boldsymbol{w}}$ referred to as a '*point estimate*'

# Bayesian Alternative

Nothing special about $\widehat{\boldsymbol{w}}$... use more than one value?

- Formulate model

**Regression**

$$p(y|\mathbf{x}) = \text{sigmoid}(\mathbf{x}'\mathbf{w}) \qquad p(y|\mathbf{x}) = \text{Normal}(\mathbf{x}'\mathbf{w}; \sigma^2)$$

- Consider the space of likely parameters – those that fit the training data well

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y})$$

We don't want to pick a single point estimation in Baysian stats

- Make 'expected' prediction

$$p(y_*|\mathbf{x}_*) = E_{p(\mathbf{w}|\mathbf{X}\ ,\mathbf{y})}\left[\text{sigmoid}(\mathbf{x}_*'\mathbf{w})\right]$$

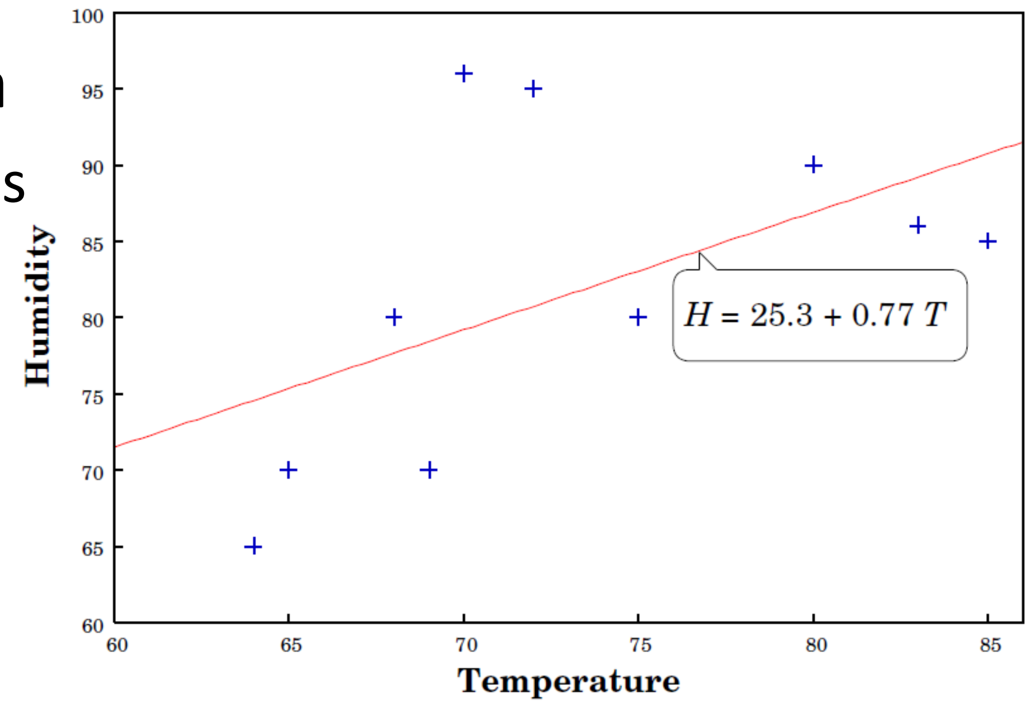
$$p(y_*|\mathbf{x}_*) = E_{p(\mathbf{w}|\mathbf{X}\ ,\mathbf{y})}\left[\text{Normal}(\mathbf{x}_*'\mathbf{w}, \sigma^2)\right]$$

# Uncertainty

*From small training sets, we rarely have complete confidence in any models learned. Can we quantify the uncertainty, and use it in making predictions?*

# Regression Revisited

- Learn model from data
  - minimise error residuals by choosing weights $\hat{\mathbf{w}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$
- But… how confident are we
  - in $\hat{\mathbf{w}}$?
  - in the predictions?




Linear regression: $y = w_0 + w_1 x$

(here $y$ = humidity, $x$ = temperature)

# Do we trust point estimate $\widehat{\mathbf{w}}$ ?

- How *stable* is learning?
  - $\widehat{\mathbf{w}}$ highly sensitive to noise
  - how much uncertainty in parameter estimate?
  - more *informative* if neg log likelihood objective highly peaked

- Formalised as *Fisher Information matrix*
  - E[2nd deriv of NLL]

$$\mathcal{I} = \frac{1}{\sigma^2}\mathbf{X}'\mathbf{X}$$

  - measures *curvature of objective* about $\widehat{\mathbf{w}}$

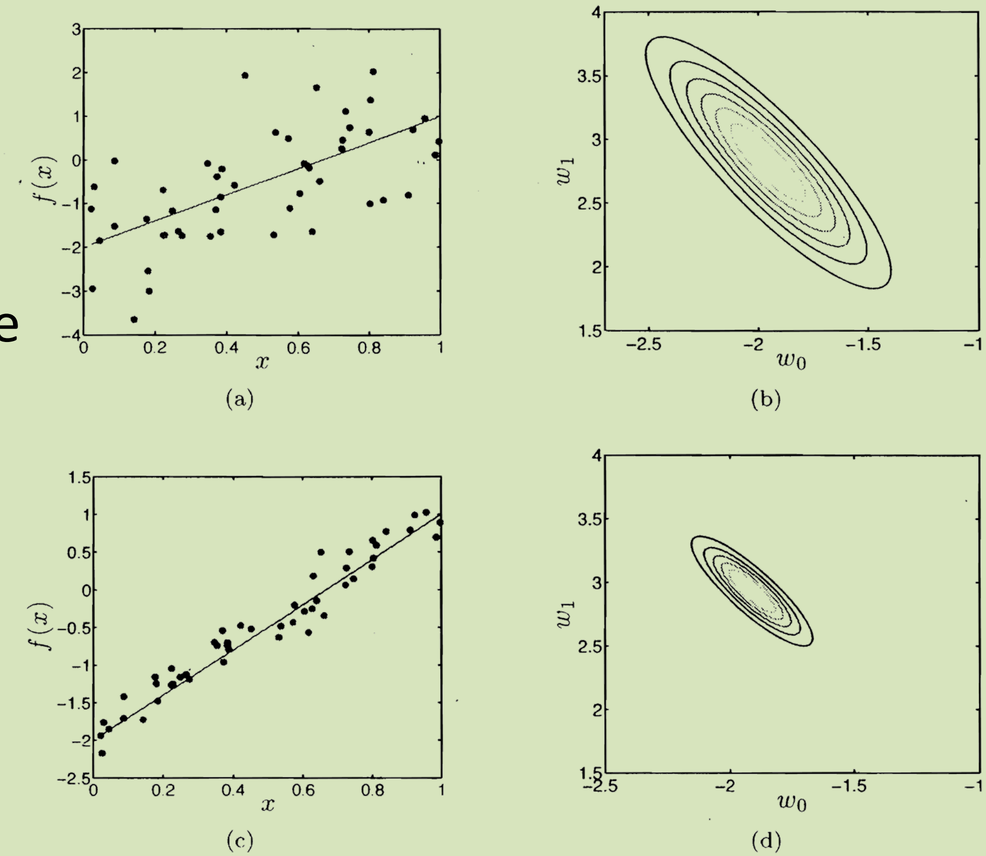



Figure: Rogers and Girolami p81

# Mini Summary

- Uncertainty not captured by point estimates (MLE, MAP)
- Uncertainty might capture range of plausible parameters
- (Frequentist) idea of Fisher information as likelihood sensitivity at point estimates

Next time: The Bayesian view (reminder)

# The Bayesian View

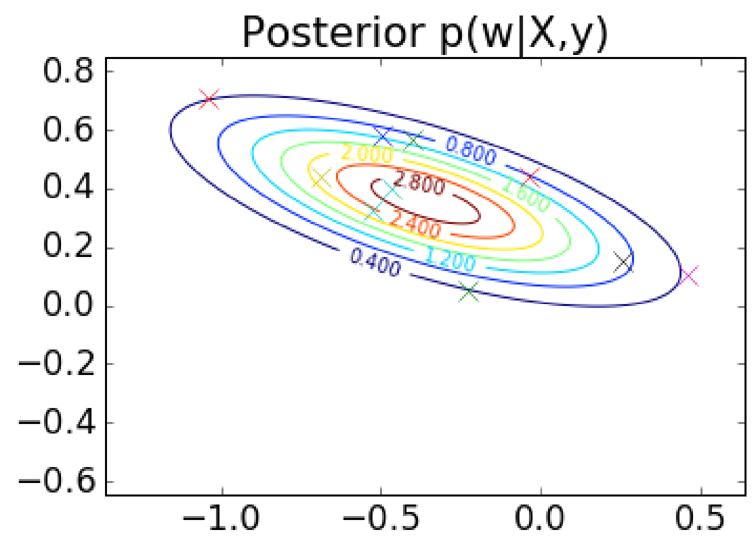
*Retain and model all unknowns (e.g., uncertainty over parameters) and use this information when making inferences.*

# A Bayesian View

- Could we reason over *all* parameters that are consistent with the data?
  - weights with a better fit to the training data should be more probable than others
  - make predictions with all these weights, *scaled by their probability*
- This is the idea underlying **Bayesian** inference

# Uncertainty over parameters

- Many reasonable solutions to objective
  - why select just one?
- Reason under ***all*** possible parameter values
  - weighted by their *posterior probability*
- More robust predictions
  - less sensitive to overfitting, particularly with small training sets
  - can give rise to more expressive model class (Bayesian logistic regression becomes non-linear!)



# Frequentist vs Bayesian “divide”

- **Frequentist**: learning using *point estimates*, regularisation, *p*-values …
  - backed by sophisticated theory on simplifying assumptions
  - mostly simpler algorithms, characterises much practical machine learning research
- **Bayesian**: maintain *uncertainty*, marginalise (sum) out unknowns during inference
  - some theory
  - often more complex algorithms, but not always
  - often (not always) more computationally expensive

# Mini Summary

- Frequentist's central preference of point estimates don't capture uncertainty
- Bayesian view is to quantify belief in prior, update it to posterior using observations

Next time: Bayesian approach to linear regression

# Bayesian Regression

*Application of Bayesian inference to linear regression, using Normal prior over* $\mathbf{w}$

# Revisiting Linear Regression

Identity matrix since we normally have D number of w in out model

$I_D$ = *D* x *D* identity matrix

- Recall probabilistic formulation of linear regression

Likelihood of our data

$$y \sim \text{Normal}(\mathbf{x}'\mathbf{w}, \sigma^2)$$

More specificly

$$y_i|\mathbf{x}_i \overset{\text{ind}}{\sim} \text{Normal}(\mathbf{x}_i^\top \mathbf{w}, \sigma^2), \qquad i \in \{1, \dots, n\}$$

$$\mathbf{w} \sim \text{Normal}(\mathbf{0}, \gamma^2 \mathbf{I}_D)$$

- Bayes rule:

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

$$\max_{\mathbf{w}} p(\mathbf{w}|\mathbf{X}, \mathbf{y}) = \max_{\mathbf{w}} p(\mathbf{y}|\mathbf{X}, \mathbf{w})p(\mathbf{w})$$

- Gives rise to penalised objective (ridge regression)

point estimate taken here, avoids computing marginal likelihood term

# Bayesian Linear Regression

- Rewind one step, consider full posterior

We are not only considering the point estimation but considering the w in a distribution context

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}, \sigma^2) = \frac{p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \sigma^2)p(\mathbf{w})}{p(\mathbf{y}|\mathbf{X}, \sigma^2)}$$

Here we assume noise var. known

$$= \frac{p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \sigma^2)p(\mathbf{w})}{\int \underbrace{p(\mathbf{y}, |\mathbf{X}, \mathbf{w}, \sigma^2)p(\mathbf{w})}_{p(y, w | X)} d\mathbf{w}}$$

- Can we compute the denominator (marginal likelihood or evidence)?
  * if so, we can use the full posterior, not just its mode

# Bayesian Linear Regression (cont)

- We have two Normal distributions
  - normal likelihood x normal prior

$$\frac{p(\mathbf{y}|\mathbf{X},\mathbf{w},\sigma^2)p(\mathbf{w})}{p(\mathbf{y}|\mathbf{X},\sigma^2)}$$

- Their product is also a Normal distribution
  - ***conjugate prior:*** *when product of likelihood x prior results in the same distribution as the prior*
  - *evidence* can be computed easily using the normalising constant of the Normal distribution

$$\begin{aligned} p(\mathbf{w}|\mathbf{X},\mathbf{y},\sigma^2) &\propto \mathrm{Normal}(\mathbf{w}|\mathbf{0},\gamma^2\mathbf{I}_D)\mathrm{Normal}(\mathbf{y}|\mathbf{X}\mathbf{w},\sigma^2\mathbf{I}_N) \\ &\propto \mathrm{Normal}(\mathbf{w}|\mathbf{w}_N,\mathbf{V}_N) \end{aligned}$$

closed form solution for posterior!

# Bayesian Linear Regression (cont)

$$p(\mathbf{w}|\mathbf{X},\mathbf{y},\sigma^2) \propto \mathrm{Normal}(\mathbf{w}|\mathbf{0},\gamma^2\mathbf{I}_D)\mathrm{Normal}(\mathbf{y}|\mathbf{X}\mathbf{w},\sigma^2\mathbf{I}_N)$$
$$\propto \mathrm{Normal}(\mathbf{w}|\mathbf{w}_N,\mathbf{V}_N)$$

where $$\mathbf{w}_N = \frac{1}{\sigma^2}\mathbf{V}_N\mathbf{X}'\mathbf{y}$$

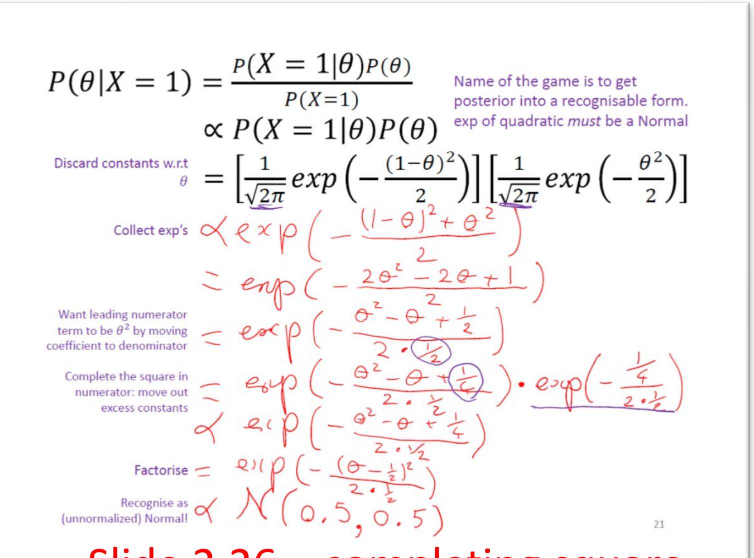
This represent multiple mean in a array where each one represent the mean of the distribution for that weight

$$\mathbf{V}_N = \sigma^2(\mathbf{X}'\mathbf{X} + \frac{\sigma^2}{\gamma^2}\mathbf{I}_D)^{-1}$$

This is a multi-dimensional covariance that represent the corresponding variance of each weights

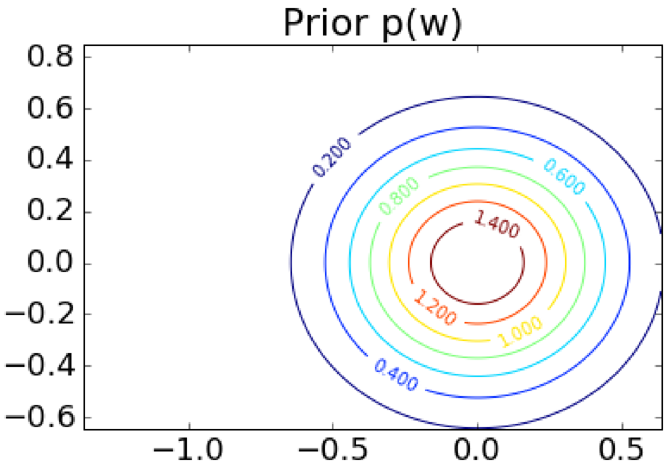
When new data come, we are updating the distribution on each dimension simultaneously

**Advanced:** verify by expressing product of two Normals, gathering exponents together and 'completing the square' to express as squared exponential (i.e., Normal distribution).

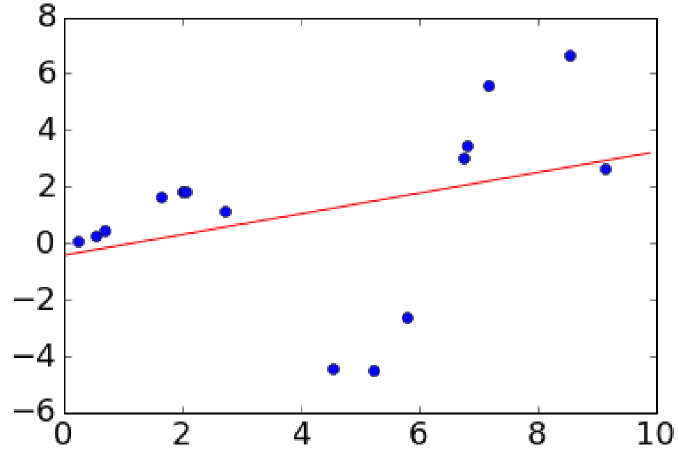


Slide 2.26 – completing square

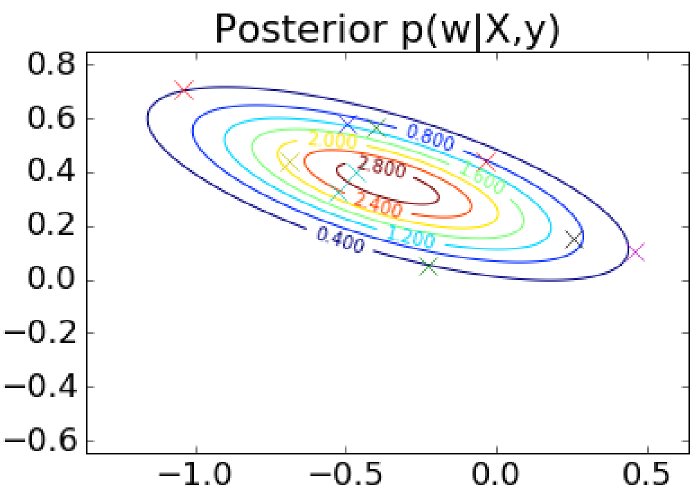
# Bayesian Linear Regression example



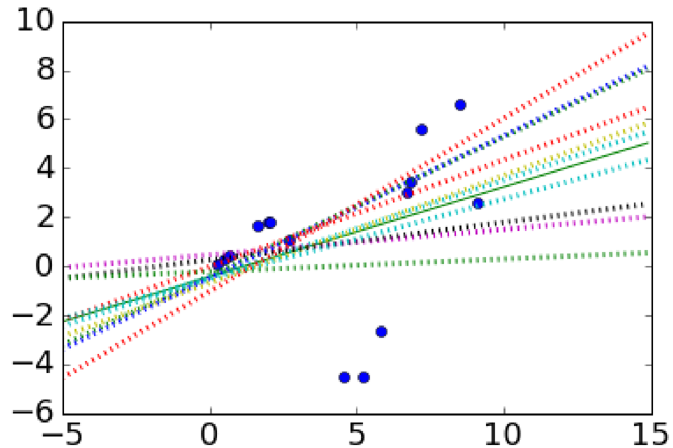

Step 1: select prior, here spherical about **0**




Step 2: observe training data




Step 3: formulate posterior, from prior & likelihood

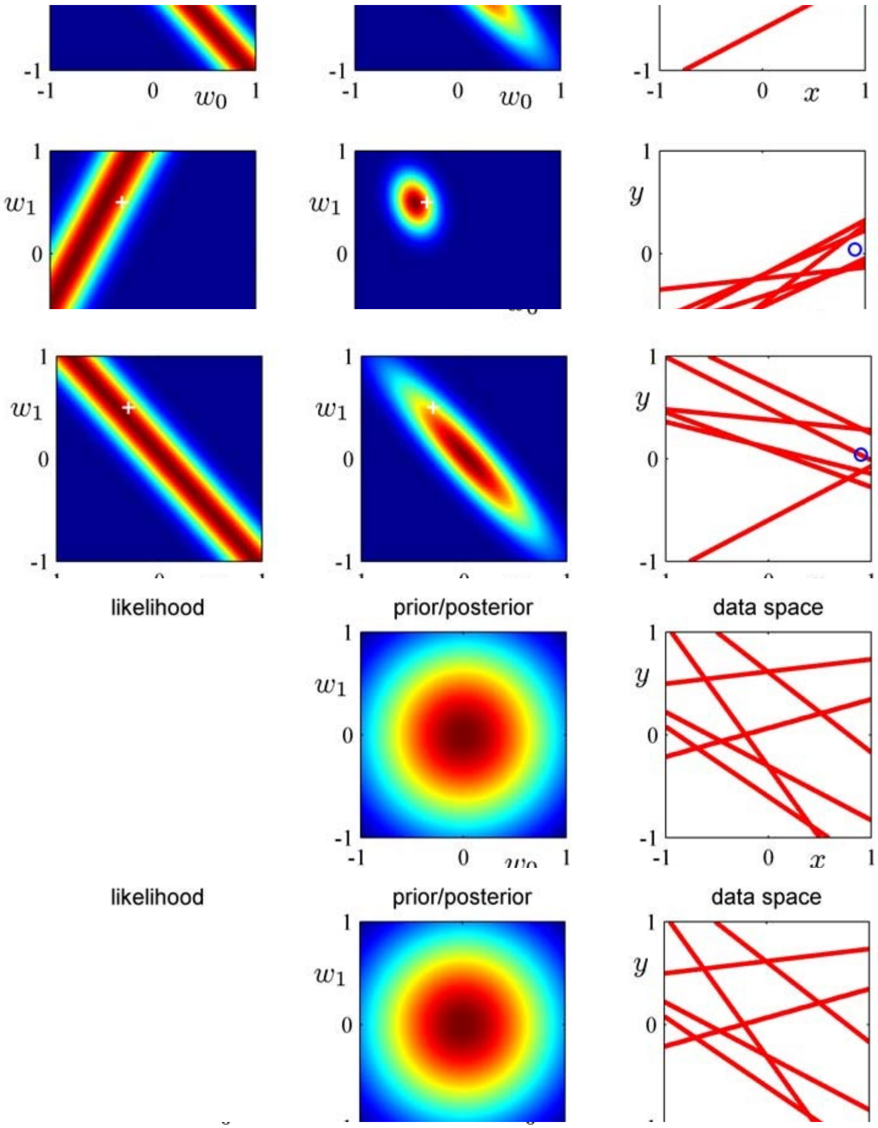



Samples from posterior

# Sequential Bayesian Updating

- Can formulate $p(\mathbf{w}|\mathbf{X}, \mathbf{y}, \sigma^2)$ for given dataset
- What happens as we see more and more data?
  1. Start from prior $p(\mathbf{w})$
  2. See new labelled datapoint
  3. Compute posterior $p(\mathbf{w}|\mathbf{X}, \mathbf{y}, \sigma^2)$
  4. The ***posterior now takes role of prior***
     & repeat from step 2

# Sequential Bayesian Updating




- Initially know little, many regression lines licensed
- Likelihood constrains possible weights such that regression is close to point
- Posterior becomes more refined/peaked as more data introduced
- Approaches a point mass

*Bishop Fig 3.7, p155*

# Stages of Training

1. Decide on model formulation & prior
2. Compute *posterior* over parameters, $p(\boldsymbol{w}|\boldsymbol{X},\boldsymbol{y})$

**MAP**

3. Find *mode* for ***w***
4. Use to make prediction on test

**approx. Bayes**

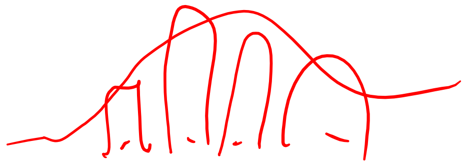
3. Sample many ***w***
4. Use to make *ensemble* average prediction on test

**exact Bayes**

3. Use *all* ***w*** to make *expected* prediction on test

# Prediction with uncertain *w*

- Could predict using sampled regression curves
  - sample $S$ parameters, $\boldsymbol{w}^{(s)}, s \in \{1, \dots, S\}$
  - for each sample compute prediction $y_*^{(s)}$ at test point $\boldsymbol{x}_*$
  - compute the mean (and var.) over these predictions
  - this process is known as **Monte Carlo integration**
- For Bayesian regression there's a simpler solution
  - integration can be done analytically, for



$$p(\hat{y}_* \,|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}_*, \sigma^2) = \int \underbrace{p(\boldsymbol{w}\,|\boldsymbol{X}, \boldsymbol{y}, \sigma^2)}_{\text{posterior}} \underbrace{p(y_* |\, \boldsymbol{x}_*, \boldsymbol{w}, \sigma^2)}_{\text{likelihood}}\, d\boldsymbol{w}$$

# Prediction (cont.)

- Pleasant properties of Gaussian distribution means integration is tractable

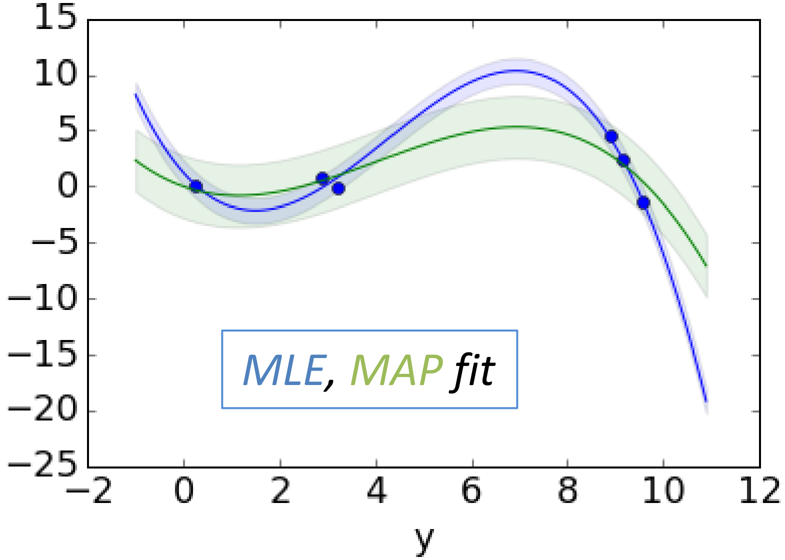
$$
\begin{aligned}
p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}, \sigma^2) &= \int p(\mathbf{w}|\mathbf{X}, \mathbf{y}, \sigma^2) p(y_*|\mathbf{x}_*, \mathbf{w}, \sigma^2) d\mathbf{w} \\
&= \int \mathrm{Normal}(\mathbf{w}|\mathbf{w}_N, \mathbf{V}_N) \mathrm{Normal}(y_*|\mathbf{x}_*'\mathbf{w}, \sigma^2) d\mathbf{w} \\
&= \mathrm{Normal}(y_*|\mathbf{x}_*'\mathbf{w}_N, \sigma_N^2(\mathbf{x}_*)) \\
\sigma_N^2(\mathbf{x}_*) &= \sigma^2 + \mathbf{x}_*'\mathbf{V}_N\mathbf{x}_*
\end{aligned}
$$

  - additive variance based on $x_*$ match to training data
  - *cf.* MLE/MAP estimate, where variance is a fixed constant

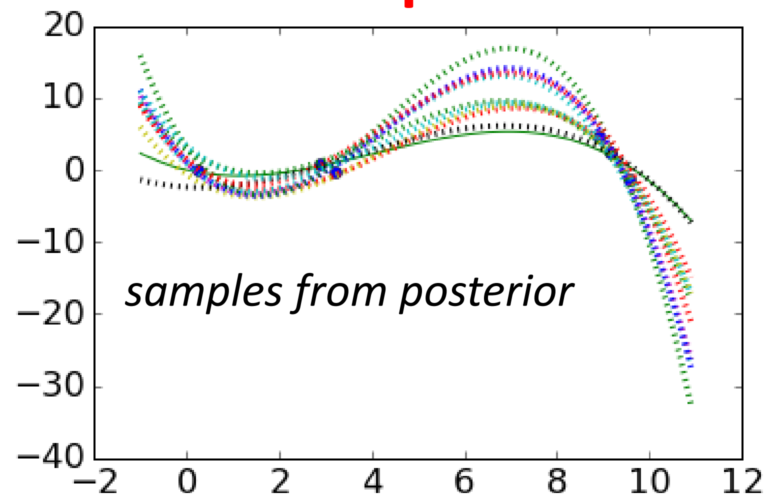
($\boldsymbol{w}_N$ and $\boldsymbol{V}_N$ defined in posterior when fitting Bayesian linear regression)

# Bayesian Prediction example

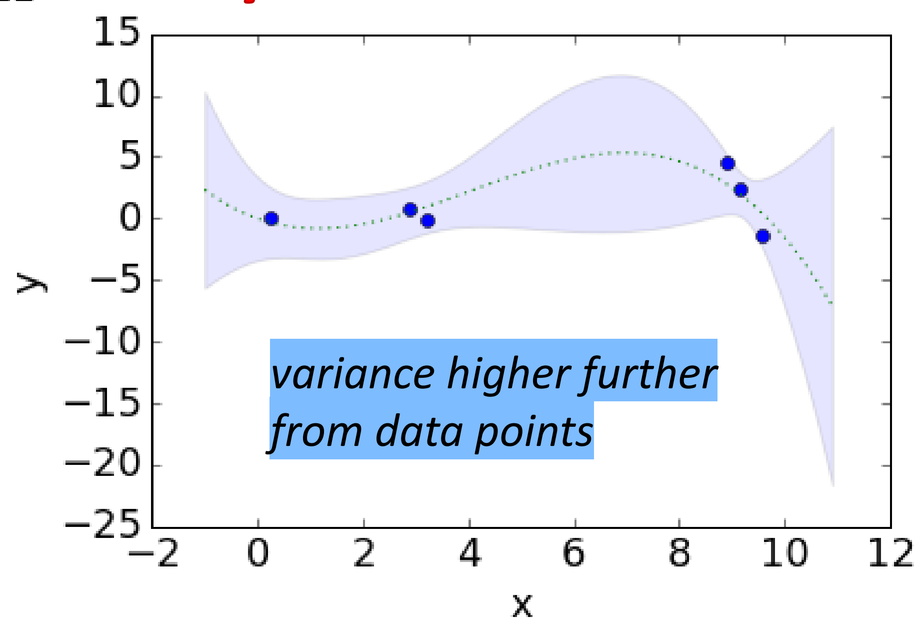
## Point estimate




MLE (blue) and MAP (green) point estimates, *with fixed variance*




## Bayesian inference




*Data: y = x sin(x); Model = cubic*

# Caveats

- Assumptions
  - known data noise parameter, $\sigma^2$
  - data was drawn from the model distribution
- In real settings, $\sigma^2$ is unknown
  - has its own conjugate prior
    *Normal* likelihood × *InverseGamma* prior
    results in *InverseGamma* posterior
  - closed form predictive distribution, with student-T likelihood
    *(see Murphy, 7.6.3)*

# Mini Summary

- Uncertainty not captured by point estimates (MLE, MAP)
- Bayesian approach preserves uncertainty
  - care about predictions NOT parameters
  - choose prior over parameters, then model posterior
- New concepts:
  - sequential Bayesian updating
  - conjugate prior (Normal-Normal)
- Using posterior for Bayesian predictions on test

Next time: Bayesian classification, then PGMs