

# Regression: a Probabilistic View

---

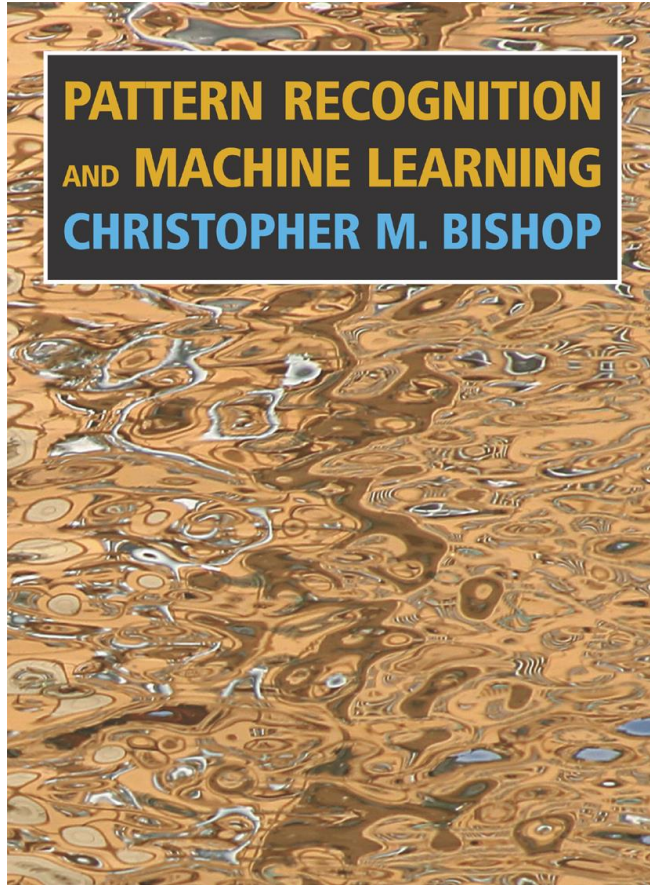
Song Liu ([song.liu@bristol.ac.uk](mailto:song.liu@bristol.ac.uk))

Office Hour: Thursday 3-4pm

Office: GA 18

# Reference

---



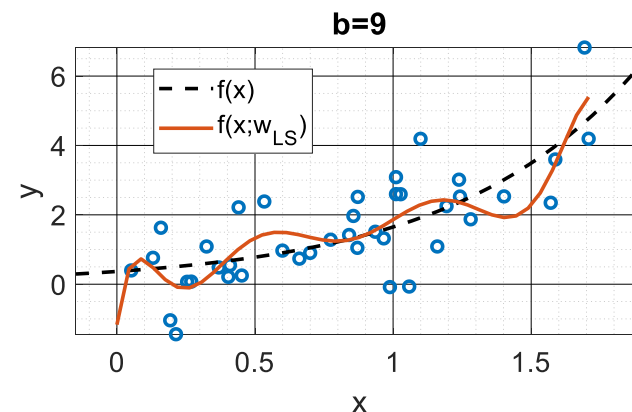
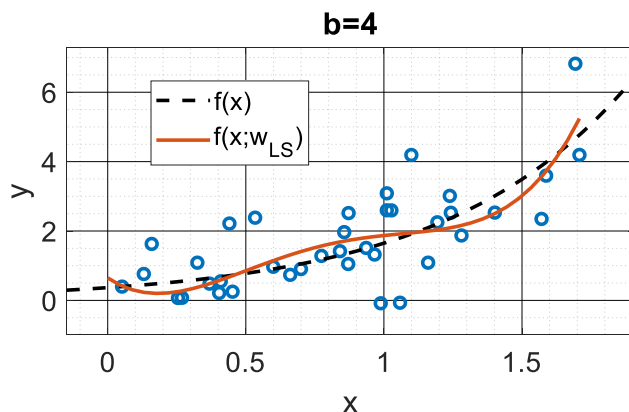
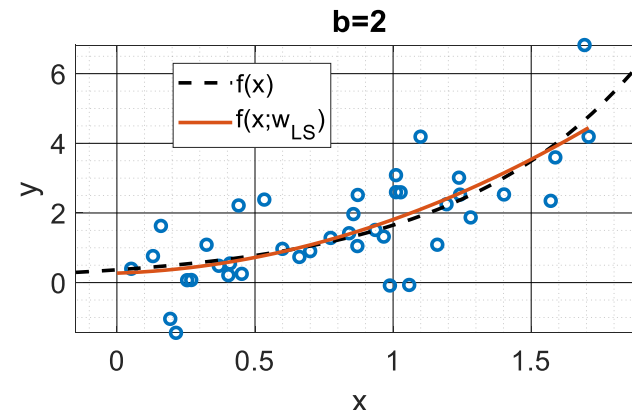
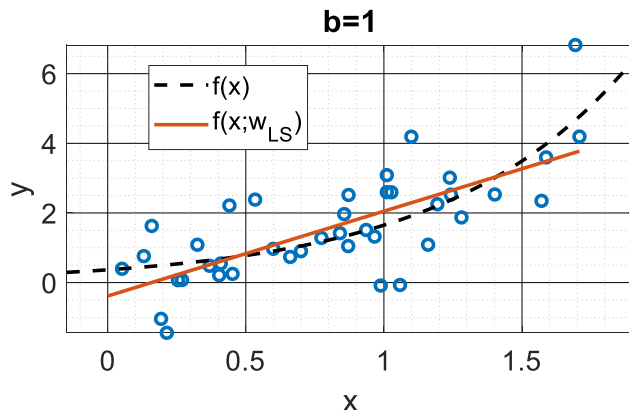
Today's class *roughly* follows Chapter 1.

Pattern Recognition and  
Machine Learning

Christopher Bishop, 2006

# The Overfitting Issue...

- Last class, we faced a dilemma:
  - By using poly. feature, we can increase the flexibility of  $f(x; w)$ .
  - The increased flexibility may also cause overfitting...



# Overfitting and Regularization

- Large  $b$  causes overfitting
  - Pick a smaller  $b$  to avoid overfitting (using CV).
- What if we want to use a larger  $b$ ?
  - We want the flexibility provided by high order polynomials.
- One trick we can do is called **regularization**.

$$\mathbf{w}_{\text{LS-R}} := \operatorname{argmin}_{\mathbf{w}} \sum_{i \in D} [y_i - f(\mathbf{x}_i; \mathbf{w})]^2 + \lambda \mathbf{w}^\top \mathbf{w}$$

regularization term

- By adding a **regularization term** to LS Error.
- Note:  $\lambda > 0$ .

# Overfitting and Regularization

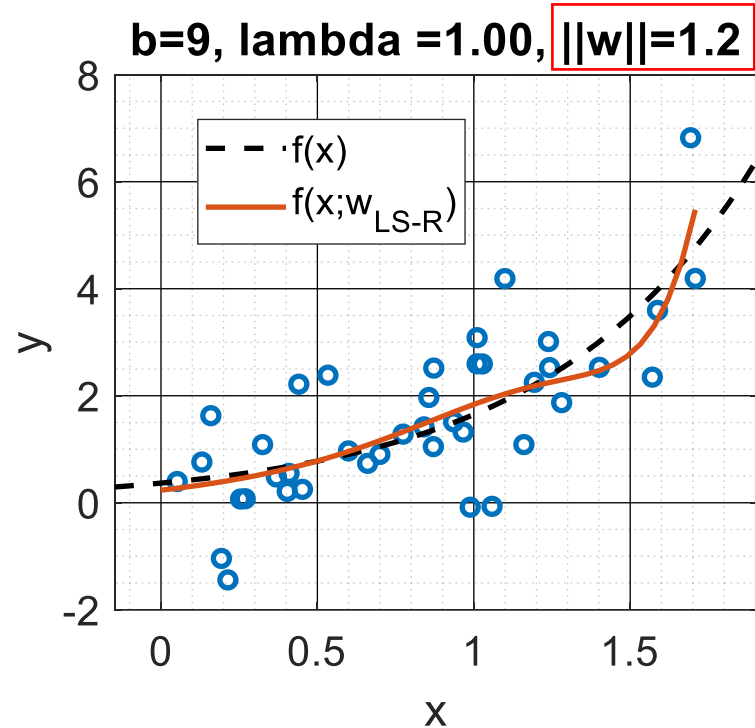
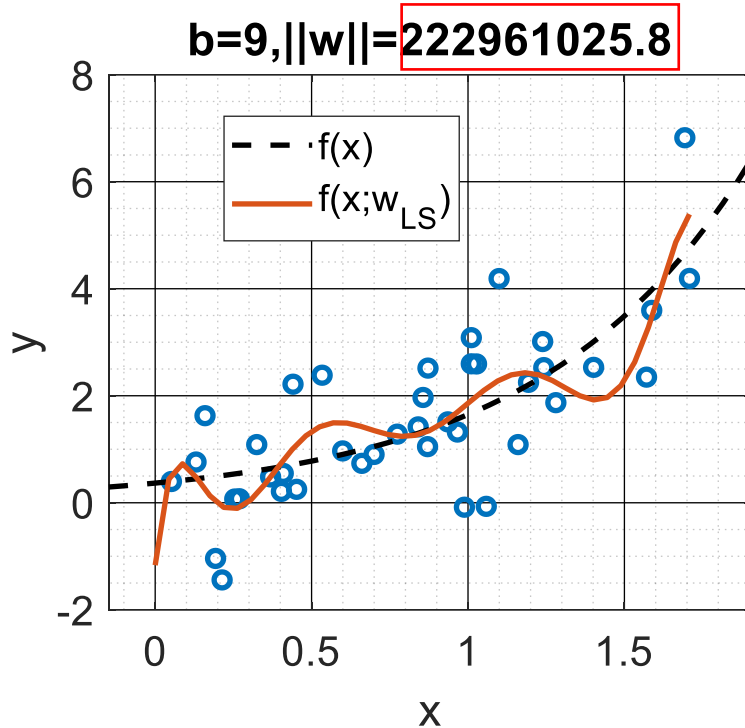
$$\mathbf{w}_{\text{LS-R}} := \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i \in D} [y_i - f(\mathbf{x}_i; \mathbf{w})]^2 + \lambda \mathbf{w}^\top \mathbf{w}$$

- $\mathbf{w}^\top \mathbf{w}$  is the magnitude of  $\mathbf{w}$
- Regularization term discourages  $\mathbf{w}$  taking large values.
- Why does the regularization help overcome overfitting?

# Overfitting and Regularization

- Prove that if regularization term is  $\lambda \mathbf{w}^\top \mathbf{w}$ ,
- $\mathbf{w}_{\text{LS-R}} := (\boldsymbol{\phi}(X)\boldsymbol{\phi}(X)^\top + \lambda \mathbf{I})^{-1} \boldsymbol{\phi}(X)\mathbf{y}^\top$ ,
  - $\mathbf{I} \in R^{b \times b}$  is identity matrix.
- $\lim_{\lambda \rightarrow \infty} \mathbf{w}_{\text{LS-R}} = \mathbf{0}$ .
  - $\lim_{\lambda \rightarrow \infty} f(\mathbf{x}; \mathbf{w}_{\text{LS-R}}) = 0$ .
  - As you enlarge  $\lambda$ , coefficients in  $\mathbf{w}_{\text{LS-R}}$  get smaller and smaller.
  - As you enlarge  $\lambda$ ,  $f(\mathbf{x}; \mathbf{w}_{\text{LS-R}})$  get flatter and flatter.
  - Which in turn reduces the complexity of  $f(\mathbf{x}; \mathbf{w}_{\text{LS-R}})$

# Example



- $f(x; w_{LS-R})$  is much less squiggly than  $f(x; w_{LS})$

See PRML, Table 1.2 for another example

# Overfitting and Regularization

$$\mathbf{w}_{\text{LS-R}} := \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i \in D} [y_i - f(\mathbf{x}_i; \mathbf{w})]^2 + \lambda \mathbf{w}^\top \mathbf{w}$$

- Regularization term does not have to be  $\mathbf{w}^\top \mathbf{w}$
- For example,  $\sum_i |w_i|$ , i. e.  $\|\mathbf{w}\|_1$  can be used too!
- $\|\mathbf{w}\|_1$  and  $\sqrt{\mathbf{w}^\top \mathbf{w}}$  are called “norms”.



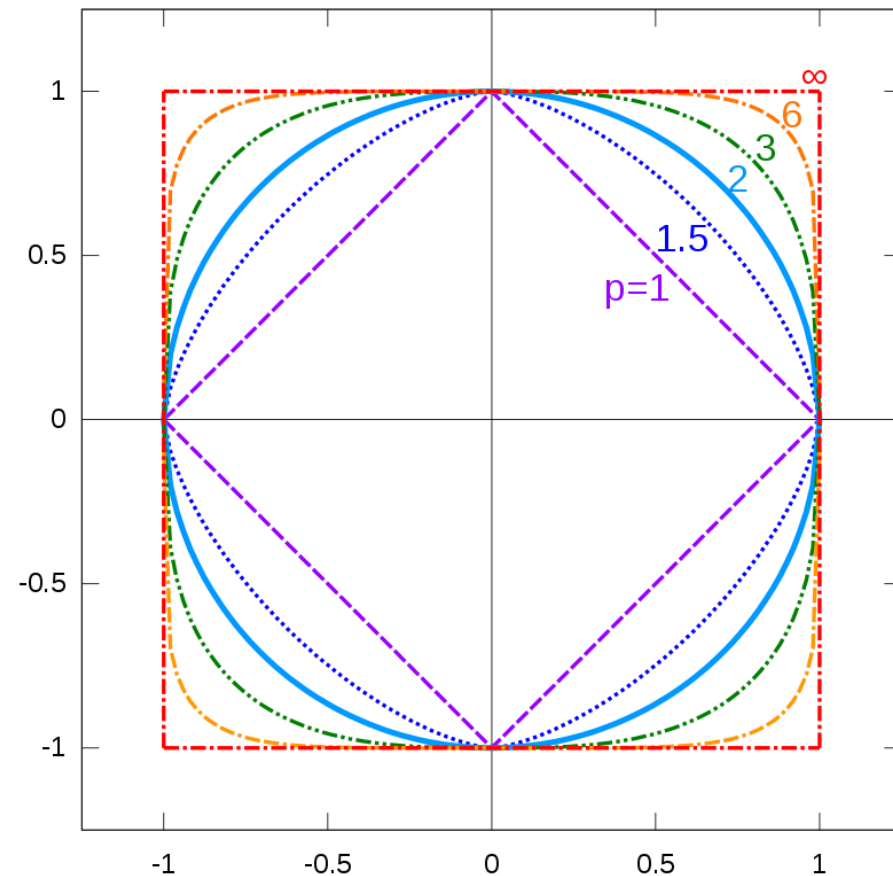
# Norms

---

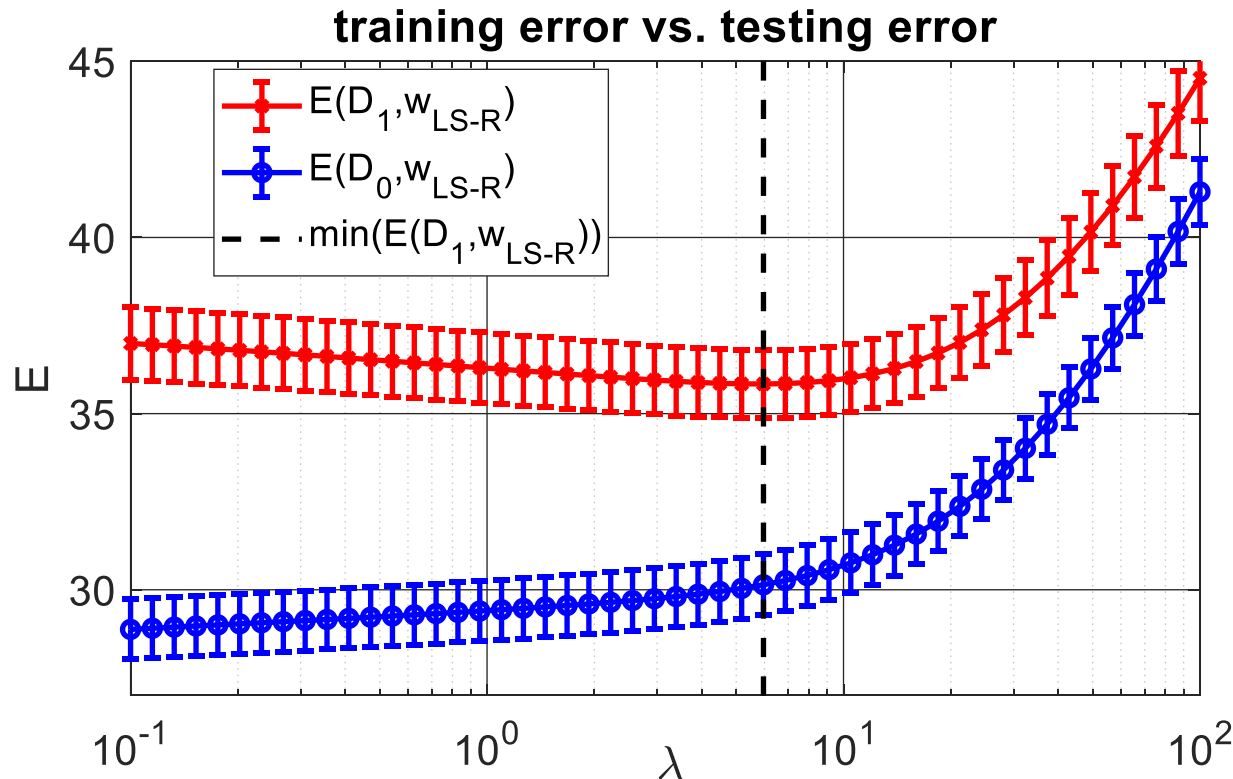
- Norms are widely used in machine learning.
- a generalization of the concept “length” in Euclidean spac.
  - $\sqrt{\mathbf{w}^\top \mathbf{w}}$  is the Euclidian distance from  $\mathbf{w}$  to the origin.
- To become a norm, a function  $t$  must satisfy
  - If  $t(\mathbf{x}) = 0$ , then  $\mathbf{x} = \mathbf{0}$
  - $t(\mathbf{x}) + t(\mathbf{y}) \geq t(\mathbf{x} + \mathbf{y})$ , Triangle Inequality
  - $t(a \cdot \mathbf{x}) = |a| \cdot t(\mathbf{x})$
- Matrix cookbook, page 60, 61, 62.

# $L^p$ norms

- An important class of norms is called  $L^p$  norm.
- $L^p$  norm for a real  $p \geq 1$ :
- $\|\mathbf{x}\|_p := (|x_1|^p + \cdots |x_d|^p)^{\frac{1}{p}}$
- Right: Unit “circle” defined by different  $L^p$  norms.



# $\lambda$ and Generalization



$D = D_0 \cup D_1$   
 $D_0$ : Training set  
 $D_1$ : Testing set

- Before the dash line, increasing  $\lambda$  reduces overfitting. After the dash line, increasing  $\lambda$  encourages underfitting.

See PRML, Figure 1.8

# Problem of Regularization

---

- How do you choose  $\lambda$ ?
- If we have plenty of i.i.d. data, we may choose a  $\lambda$  that minimizes the validation error using CV.
- However, what if we only have limited data.
- Frequentist approach does not offer a straightforward way for tuning  $\lambda$ . To choose  $\lambda$  we need to adopt a **probabilistic view of regression problem**.

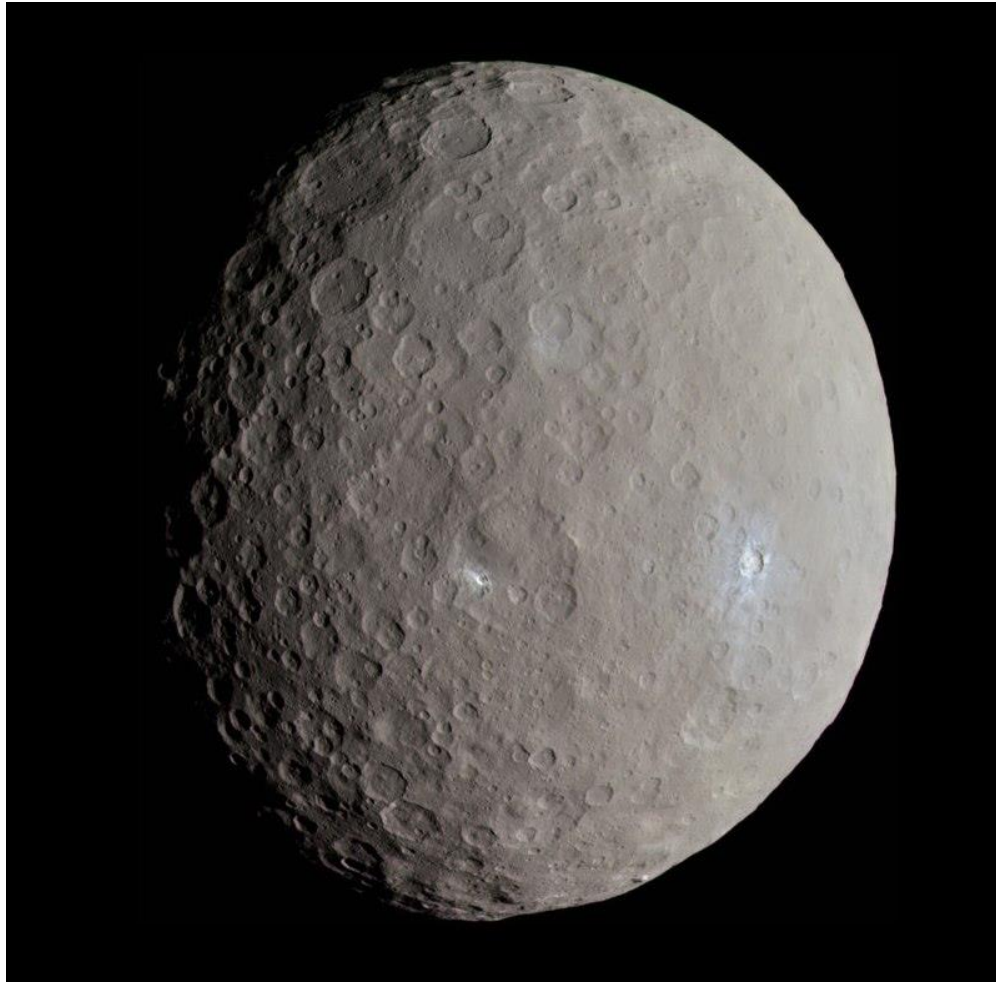
# Inverse Problems

---

- Many data science problems are **inverse problems**.
- We have a dataset of noisy observations  $D$
- We want to identify some **latent, unobserved** data generating mechanism.
- In regression, we observe  $y_i$  which is supposedly generated by
- $y_i = g(x_i) + \epsilon$ , where  $\epsilon$  is some noise.
- We are interested in finding the latent function  $g$ .

# The Prediction of Ceres

---



# Inverse Problems and Posterior

- The key of solving inverse problem is to infer posterior probability distribution  $p(g|D)$ .
  - The word “posterior” comes from the fact that  $p(g|D)$  is a probability obtained AFTER we observe  $D$ .
  - pp. 17, PRML
- The probability of a latent, data generating mechanism,  $g$ , given our dataset  $D$ .
- Problem: How do we obtain that posterior?

# Bayes' Rule (or Law, Theorem)

---

$$P(\textcolor{red}{A}|\textcolor{green}{B}) = \frac{P(\textcolor{green}{B}|\textcolor{red}{A})P(A)}{P(B)}$$

- You can calculate a conditional probability given its “inverse probability”.
- This theorem plays a key role in Bayesian statistics.
- Let us see how it helps us to obtain posterior  $p(g|D)$ .



# Inverting the Posterior by Bayes' Rule

- Using Bayes' Rule, we know

$$p(g|D) = \frac{p(D|g)p(g)}{p(D)}$$

Diagram illustrating Bayes' Rule with labels and arrows:

- $p(g|D)$  is labeled **Posterior** (blue arrow pointing up).
- $p(D|g)$  is labeled **Likelihood** (blue arrow pointing down).
- $p(g)$  is labeled **Prior** (blue arrow pointing down).
- $p(D)$  is labeled **Evidence** (blue arrow pointing up).

- $p(g)$  is called prior: the belief of our data generation mechanism  $g$  BEFORE the observation.
- $p(D|g)$  is called likelihood as it shows how likely we observe a specific dataset  $D$  given a data generator  $g$ .

# Regression using Bayes' Rule

- In regression, we want to infer  $p(g|D)$ , where  $g$  is the data generating function:
- $y_i = g(\mathbf{x}_i) + \epsilon$ .
- Suppose  $g$  admits a parametric form  $g(\mathbf{x}) = f(\mathbf{x}; \mathbf{w})$ , we only need to consider the parameter  $\mathbf{w}$ .
  - Once  $\mathbf{w}$  is determined,  $f$  is determined.
- **Task:** Infer  $p(\mathbf{w}|D)$
- **Bayes' Rule:**  $p(\mathbf{w}|D) = \frac{p(D|\mathbf{w})p(\mathbf{w})}{p(D)}$

# Regression using Bayes' Rule

- **Task:** Infer  $p(\mathbf{w}|D)$
- **Bayes' Rule:**  $p(\mathbf{w}|D) = \frac{p(D|\mathbf{w})p(\mathbf{w})}{p(D)}$
- If we assume  $\epsilon$  is drawn from a Normal dist and  $D$  is IID:
- $p(D|\mathbf{w}) = \prod_{i \in D} p(y_i|\mathbf{x}_i, \mathbf{w}, \sigma^2) = \prod_{i \in D} N_{y_i}(f(\mathbf{x}_i; \mathbf{w}), \sigma^2)$
- To compute the Bayes' rule, we also need a prior  $p(\mathbf{w})$ .
- For now, we just use a Normal dist.,  $p(\mathbf{w}) = N_{\mathbf{w}}(0, \sigma_{\mathbf{w}}^2 \mathbf{I})$ .

$$p(\mathbf{w}|D) = \frac{\prod_{i \in D} N_{y_i}(f(\mathbf{x}_i; \mathbf{w}), \sigma^2) \cdot N_{\mathbf{w}}(0, \sigma_{\mathbf{w}}^2 \mathbf{I})}{P(D)}$$

# Maximum A Posteriori (MAP)

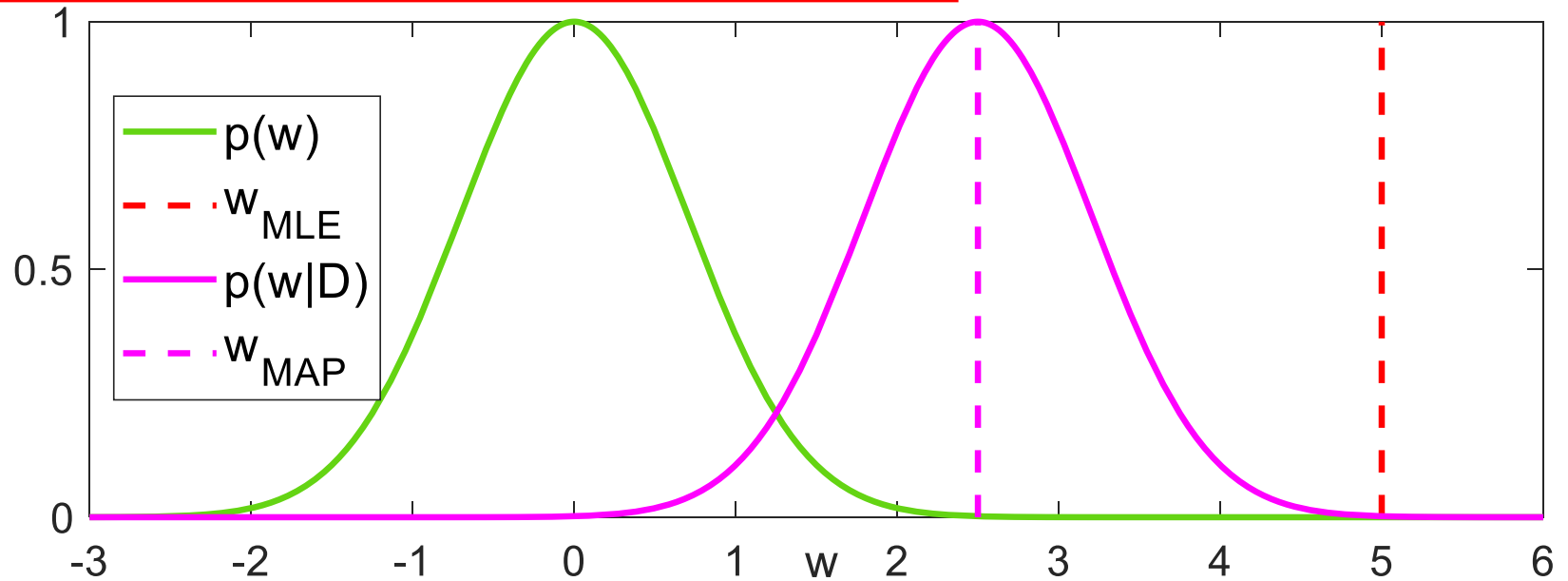
- $p(\mathbf{w}|D) = \frac{\prod_{i \in D} N_{y_i}(f(\mathbf{x}_i; \mathbf{w}), \sigma^2) \cdot N_{\mathbf{w}}(0, \sigma_{\mathbf{w}}^2 \mathbf{I})}{P(D)}$
- How to make a prediction?
  - Find a  $\mathbf{w}$  that is the most likely given our dataset  $D$ !
- To get a single  $\mathbf{w}$ , we can perform a maximization of  $p(\mathbf{w}|D)$  with respect to  $\mathbf{w}$ .
- This procedure is called Maximum A Posteriori (MAP)
- $\mathbf{w}_{\text{MAP}} := \underset{\mathbf{w}}{\operatorname{argmax}} p(\mathbf{w}|D)$   
$$= \underset{\mathbf{w}}{\operatorname{argmax}} \prod_{i \in D} N_{y_i}(f(\mathbf{x}_i; \mathbf{w}), \sigma^2) \cdot N_{\mathbf{w}}(0, \sigma_{\mathbf{w}}^2 \mathbf{I})$$

# Maximum A Posteriori (MAP)

- Prove,  $\mathbf{w}_{\text{MAP}} = \mathbf{w}_{\text{LS-R}}$  using  $\lambda = \frac{\sigma^2}{\sigma_{\mathbf{w}}^2}$ .
- After getting  $\mathbf{w}_{\text{MAP}}$ , we can plug it in  $f(\mathbf{x}; \mathbf{w}_{\text{MAP}})$  to make predictions.

# MAP vs. MLE

- $\mathbf{w}_{\text{MAP}} = \underset{\mathbf{w}}{\operatorname{argmax}} p(\mathbf{w}|D)$   
 $= \underset{\mathbf{w}}{\operatorname{argmax}} p(D|\mathbf{w})p(\mathbf{w})$



# A Full Probabilistic Approach

---

- However, why settle with a single  $\mathbf{w}$  when you already have access to  $p(\mathbf{w}|D)$ ?
- Using MAP to obtain a single  $\mathbf{w}$  for prediction **ignores the uncertainty information** represented in  $p(\mathbf{w}|D)$ .
- If not getting a single  $\mathbf{w}$ , how do we make prediction using a probability  $p(\mathbf{w}|D)$ ?

# A Full Probabilistic Approach

- Instead of making a single prediction  $\hat{y}$  given an  $\mathbf{x}$ .
  - We can calculate the predictive distribution  $p(\hat{y}|\mathbf{x}, D)$ ,
    - Probability of  $\hat{y}$  given our dataset and  $\mathbf{x}$ .
  - We know
  - $p(\hat{y}|\mathbf{x}, D) = \int p(\hat{y}|\mathbf{x}, \mathbf{w})p(\mathbf{w}|D)d\mathbf{w}$ , (why?)
  - Calculate  $p(\hat{y}|\mathbf{x}, D)$  as a marginalized probability.
- 
- How can we calculate the predictive distribution?
  - We can assume  $p(\hat{y}|\mathbf{x}, \mathbf{w}) = N_{\hat{y}}(f(\mathbf{x}, \mathbf{w}), \sigma^2)$
  - We can calculate  $p(\mathbf{w}|D)$  up to a constant  $p(D)$



# Calculating Predictive Distribution

likelihood

prior

- $p(\mathbf{w}|D) \propto \prod_{i \in D} N_{y_i}(f(\mathbf{x}_i; \mathbf{w}), \sigma^2) \cdot N_{\mathbf{w}}(0, \sigma_{\mathbf{w}}^2 \mathbf{I})$
- $p(\hat{y}|\mathbf{x}, \mathbf{w}) = N_{\hat{y}}(f(\mathbf{x}; \mathbf{w}), \sigma^2)$
- Suppose  $f(\mathbf{x}; \mathbf{w}) = \langle \mathbf{w}, \boldsymbol{\phi}(\mathbf{x}) \rangle$
- **Prove:**

- $\int p(\hat{y}|\mathbf{x}, \mathbf{w}) \cdot p(\mathbf{w}|D) d\mathbf{w} =$   
$$N_{\hat{y}} \left[ f(\mathbf{x}; \mathbf{w}_{\text{LS-R}}), \sigma^2 + \boldsymbol{\phi}^\top(\mathbf{x}) \sigma^2 \left( \boldsymbol{\phi} \boldsymbol{\phi}^\top + \frac{\sigma^2}{\sigma_{\mathbf{w}}^2} \mathbf{I} \right)^{-1} \boldsymbol{\phi}(\mathbf{x}) \right]$$
- Where  $\boldsymbol{\phi}$  is short for  $\boldsymbol{\phi}(\mathbf{X})$ , and  $\mathbf{w}_{\text{LS-R}}$  is the LS-R solution with  $\lambda = \frac{\sigma^2}{\sigma_{\mathbf{w}}^2}$ .

# The Predictive Distribution

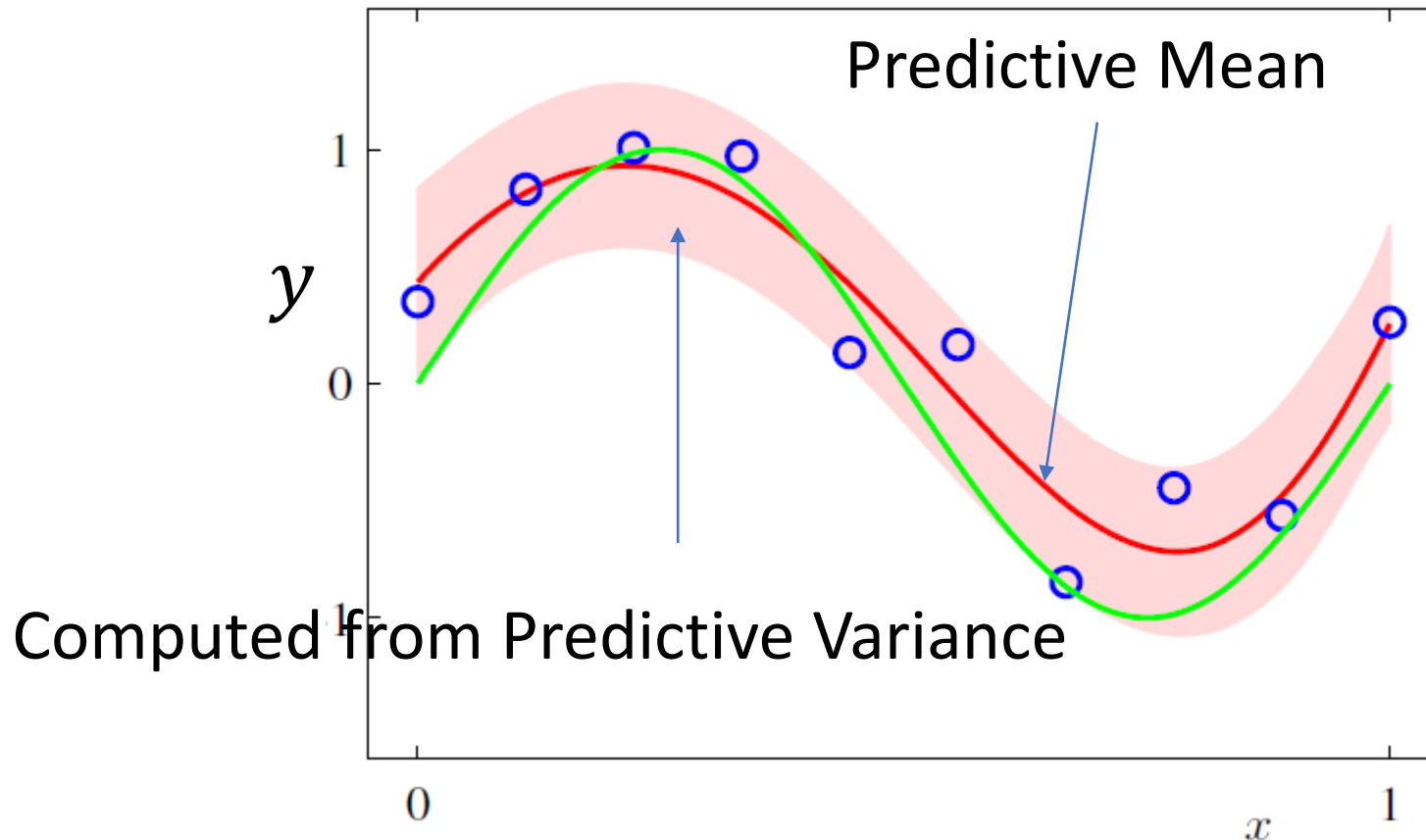
$$\bullet p(\hat{y}|\mathbf{x}, D) = \int p(\hat{y}|\mathbf{x}, \mathbf{w}) \cdot p(\mathbf{w}|D) d\mathbf{w} = N_{\hat{y}} \left[ f(\mathbf{x}; \mathbf{w}_{\text{LS-R}}), \sigma^2 + \boldsymbol{\phi}^\top(\mathbf{x}) \sigma^2 \left( \boldsymbol{\phi} \boldsymbol{\phi}^\top + \frac{\sigma^2}{\sigma_{\mathbf{w}}^2} \mathbf{I} \right)^{-1} \boldsymbol{\phi}(\mathbf{x}) \right]$$

- The mean of  $p(\hat{y}|\mathbf{x}, D)$  is the LS-R prediction!
- The idea of regularization naturally arises from both probabilistic modelling approaches.

# A Full Probabilistic Approach

- With the predictive distribution  $p(\hat{y}|\mathbf{x}, D)$ , we can compute:
- Prediction:  $\mathbb{E}_{p(\hat{y}|\mathbf{x}, D)}[\hat{y}|\mathbf{x}]$ ,
- Prediction uncertainty:  $\text{var}_{p(\hat{y}|\mathbf{x}, D)}[\hat{y}|\mathbf{x}]$ .
- We can also use the predictive distribution to calculate other interesting expected values, as we will see later.

# Example: $p(\hat{y}|x, D)$



- **PRML, Figure 1.17**

# Conclusion

---

- We looked at “Regularized LS” from three different perspectives:
  - Regularized LS (Frequentist)
  - MAP (Semi-Bayesian)
  - Probabilistic Approach (Full Bayesian)
- However, we still have not incorporated an important concept, risk function, in our decision making process.
  - Recall, making wrong decisions has different consequences.
- Next, we talk about statistical decision making.
  - We will finally wrap up Chapter 1, PRML.

# Homework

---

- Prove the statement on page 6
- Revisit: “The solution of  $\mathbf{w}_{LS}$  is useless if  $n < d$ . ”
  - Is this statement still true for  $\mathbf{w}_{LS-R}$ ?
- Prove the statement on page 21
- Prove the statement on page 25