Regression: a Probabilistic View

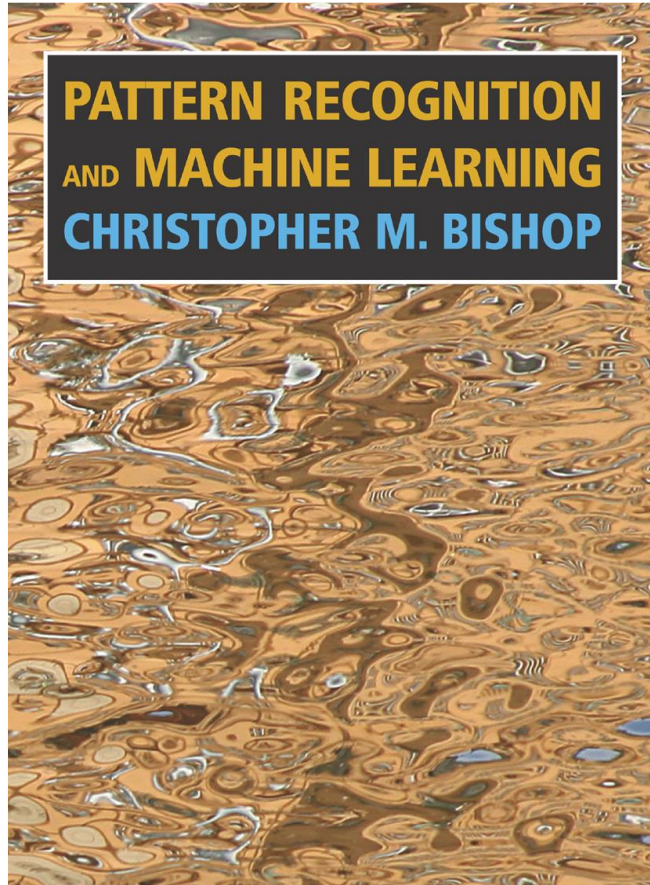Song Liu (song.liu@bristol.ac.uk)

Office Hour: Tuesday 2-3pm
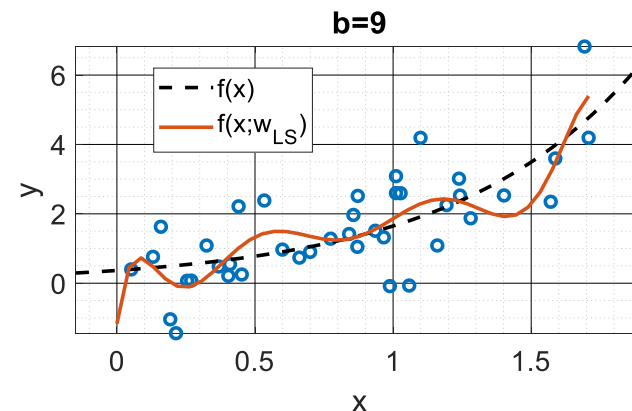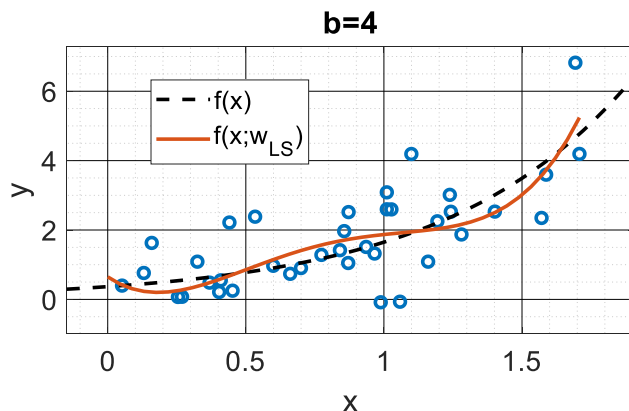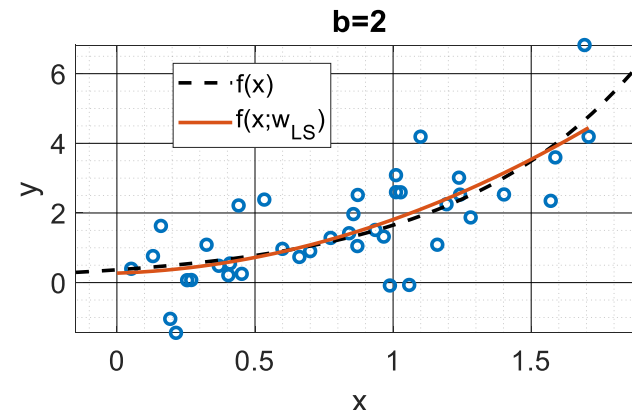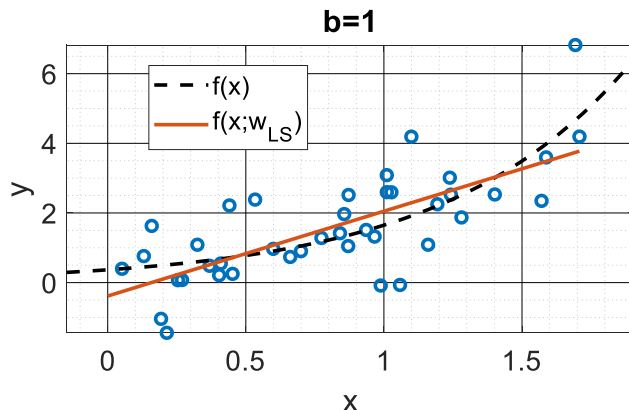
Office: GA 18

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 $f(\boldsymbol{x}; \boldsymbol{w})$.
  - 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**.

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

regularization term

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

Overfitting and Regularization

$$w_{\mathrm{LS-R}} := \underset{\boldsymbol{w}}{\mathrm{argmin}} \sum_{i \in D} [y_i - f(\boldsymbol{x}_i; \boldsymbol{w})]^2 + \boxed{\lambda \boldsymbol{w}^\top \boldsymbol{w}}$$

- $\boldsymbol{w}^\top \boldsymbol{w}$ is the magnitude of $\boldsymbol{w}$

- Regularization term <mark>discourages</mark> $\boldsymbol{w}$ taking large values.

- Why does the regularization help overcome overfitting?

Overfitting and Regularization

- Prove that if regularization term is $\lambda \boldsymbol{w}^\top \boldsymbol{w}$,

- $\boldsymbol{w}_{\mathrm{LS-R}} \coloneqq (\boldsymbol{\phi}(X)\boldsymbol{\phi}(X)^\top + \lambda \boldsymbol{I})^{-1}\boldsymbol{\phi}(X)\boldsymbol{y}^\top$,
  - $\boldsymbol{I} \in R^{b \times b}$ is identity matrix.

- $\mathrm{limit}_{\lambda \to \infty} ||\boldsymbol{w}_{\mathrm{LS-R}}|| = 0$.
  - $\mathrm{limit}_{\lambda \to \infty} f(\boldsymbol{x}; \boldsymbol{w}_{\mathrm{LS-R}}) = 0$.
  - As you enlarge λ, coefficients' magnitude in $\boldsymbol{w}_{\mathrm{LS-R}}$ get <mark>smaller and smaller.</mark>
  - As you enlarge λ, $f(\boldsymbol{x}; \boldsymbol{w}_{\mathrm{LS-R}})$ get <mark>flatter and flatter.</mark>
  - Which in turn reduces the complexity of $f(\boldsymbol{x}; \boldsymbol{w}_{\mathrm{LS-R}})$

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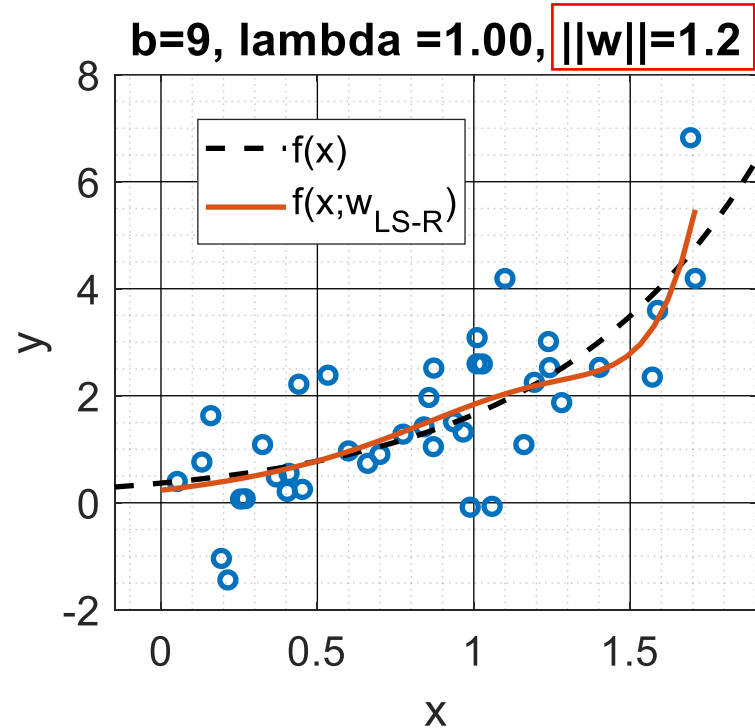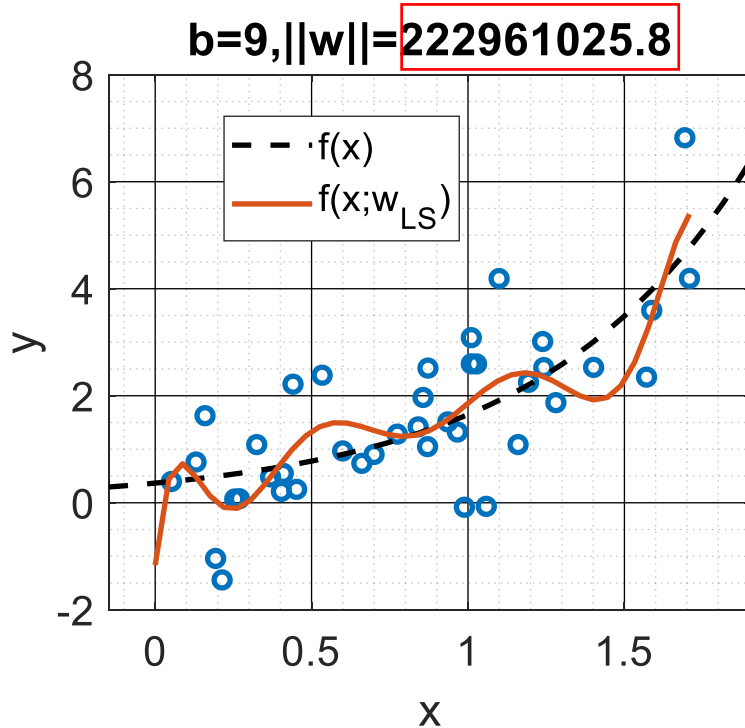
- $f(\boldsymbol{x}; w_{\mathrm{LS-R}})$ is much less squiggly than $f(\boldsymbol{x}; w_{\mathrm{LS}})$

  See PRML, Table 1.2 for another example

Overfitting and Regularization

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

- Regularization term does not have to be $w^\top w$

- For example, $\sum_i |w_i|$, i.e. $||w||_1$ can be used too!

- $||w||_1$ and $\sqrt{w^\top w}$ are called "norms".

Norms

- Norms are widely used in machine learning.
- a generalization of the concept "length" in Euclidean spac.
  - $\sqrt{w^\top w}$ is the Euclidian distance from w to the origin.

- To become a norm, a positive function t must satisfy
  - If $t(x) = 0$, then $x = 0$
  - $t(x) + t(y) \geq t(x + y)$, Triangle Inequality
  - $t(a \cdot x) = |a| \cdot t(x)$

- 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 $p \geq 1$:

- $||\boldsymbol{x}||_p := (|x_1|^p + \cdots |x_d|^p)^{\frac{1}{p}}$

- Right: Unit "circle" defined by different L^p norms.

λ and Generalization

training error vs. testing error

Legend:
- $E(D_1, w_{LS\text{-}R})$
- $E(D_0, w_{LS\text{-}R})$
- $\min(E(D_1, w_{LS\text{-}R}))$
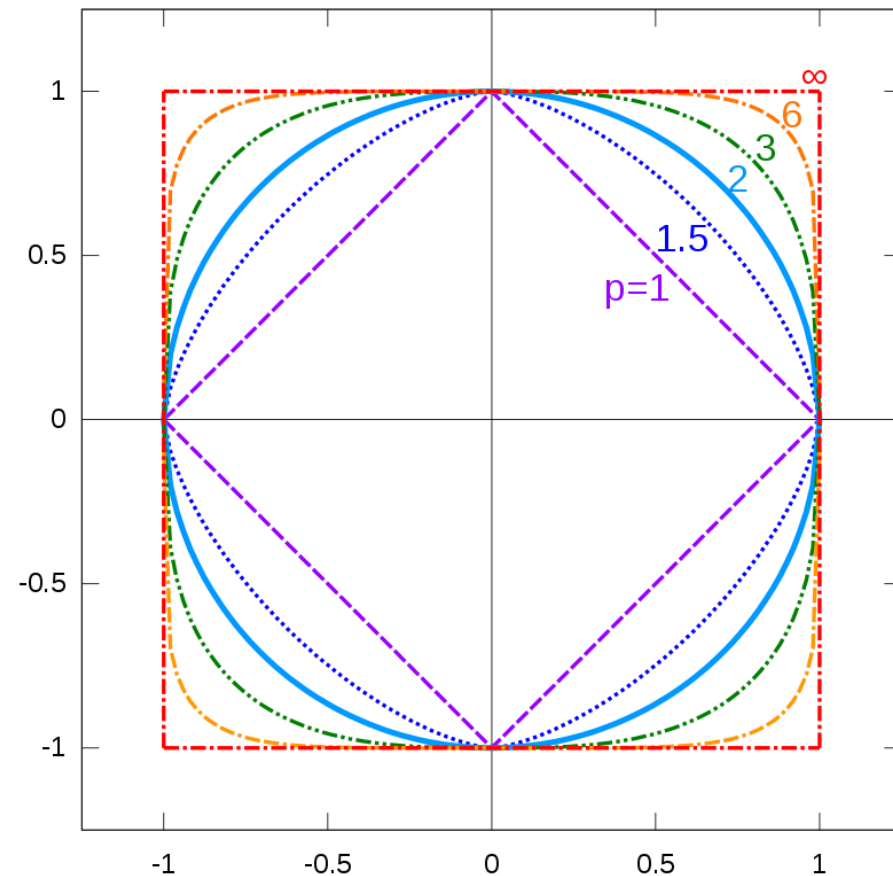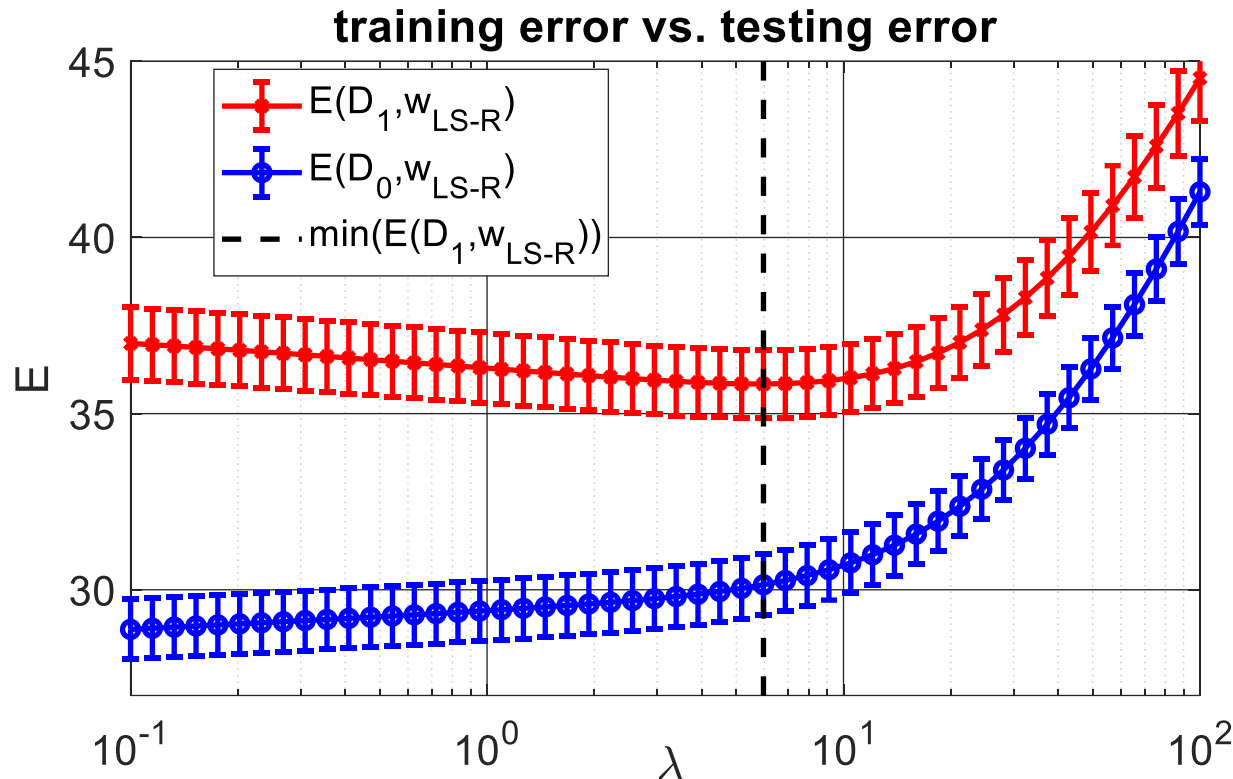
$D = D_0 \cup D_1$
D_0: Training set
D_1: Testing set

- Before the dash line, increasing λ reduces overfitting. After the dash line, increasing λ encourages underfitting.

  See PRML, Figure 1.8

Problem of Regularization

- How do you choose λ?

- If we have plenty of i.i.d. data, we may choose a λ 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 λ. To choose λ we need to adopt **a probabilistic view of regression problem.**

Inverse Problems

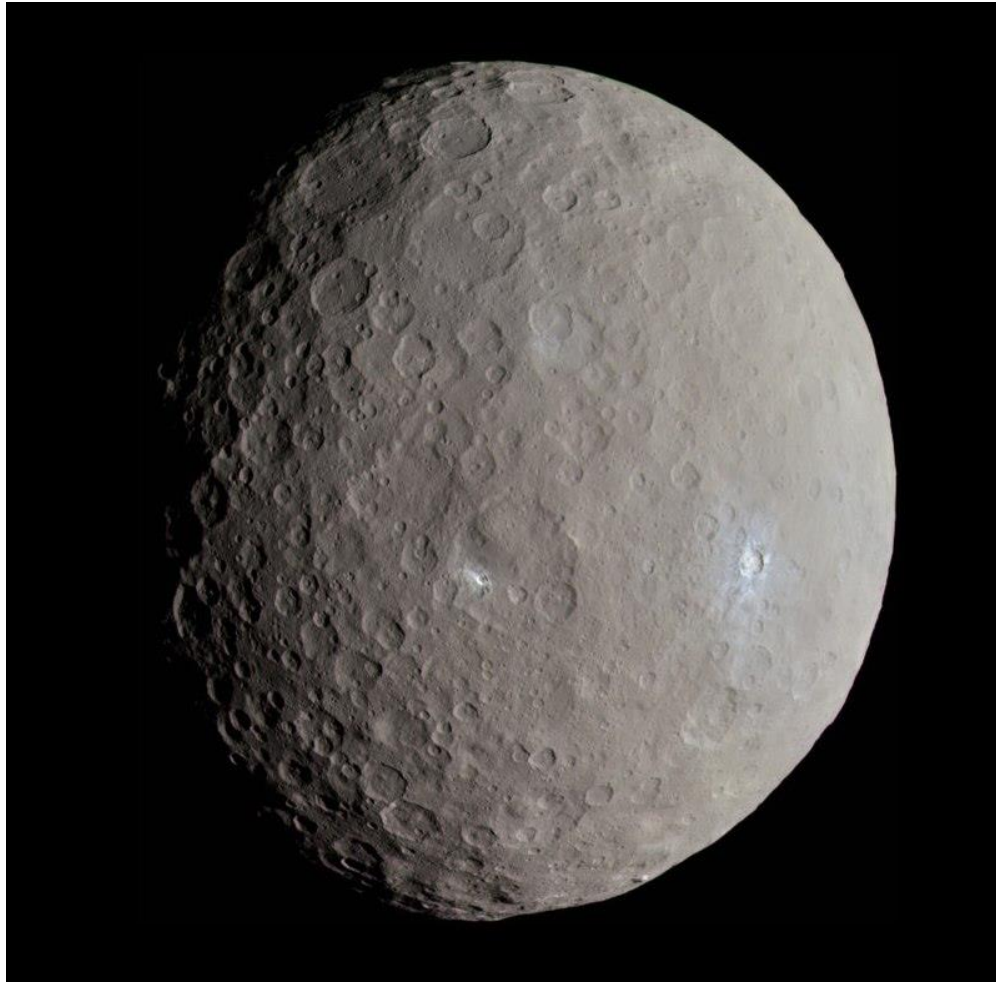- Many data science problems are <mark>inverse problems</mark>.

- 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 ϵ 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 <mark>infer posterior probability distributrion $p(g|D)$.</mark>
  - 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

Likelihood   Prior

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

Posterior

Evidence

- $p(g)$ 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:

- $y_i = g(\boldsymbol{x}_i) + \epsilon$.

- Suppose g admits a parametric form $g(\boldsymbol{x}) = f(\boldsymbol{x}; \boldsymbol{w})$, we only need to consider the parameter $\boldsymbol{w}$.
  - Once $\boldsymbol{w}$ is determined, f is determined.

- **Task:** Infer $p(\boldsymbol{w}|D)$

- **Bayes' Rule:** $p(\boldsymbol{w}|D) = \dfrac{p(D|\boldsymbol{w})p(\boldsymbol{w})}{p(D)}$

Regression using Bayes' Rule

- **Task:** Infer $p(\boldsymbol{w}|D)$

- **Bayes' Rule:** $p(\boldsymbol{w}|D) = \dfrac{p(D|\boldsymbol{w})p(\boldsymbol{w})}{p(D)}$

- If we assume ϵ is drawn from a Normal dist and D is IID:

- $p(D|\boldsymbol{w}) = \prod_{i \in D} p\left(y_i \middle| \boldsymbol{x}_i, \boldsymbol{w}, \sigma^2\right) = \prod_{i \in D} N_{y_i}\left(f(\boldsymbol{x}_i; \boldsymbol{w}), \sigma^2\right)$

- To compute the Bayes' rule, we also need a prior $p(\boldsymbol{w})$.

- For now, we just use a Normal dist., $p(\boldsymbol{w}) = N_{\boldsymbol{w}}\left(0, \sigma_{\boldsymbol{w}}^2 I\right)$.

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

Maximum A Posteriori (MAP)

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

- How to make a prediction?
  - Find a $\boldsymbol{w}$ that is the most likely given our dataset D!

- To get a single $\boldsymbol{w}$, we can perform a maximization of $p(\boldsymbol{w}|D)$ with respect to $\boldsymbol{w}$.

- This procedure is called Maximum A Posteriori (MAP)

- $\boldsymbol{w}_{\text{MAP}} := \underset{\boldsymbol{w}}{\text{argmax}}\, p(\boldsymbol{w}|D)$

$$= \text{argmax}_{\boldsymbol{w}} \prod_{i \in D} N_{y_i}(f(\boldsymbol{x}_i; \boldsymbol{w}), \sigma^2) \cdot N_{\boldsymbol{w}}(0, \sigma_{\boldsymbol{w}}^2 \mathbf{I})$$

Maximum A Posteriori (MAP)

- Prove, $\mathbf{w}_{\mathrm{MAP}} = \boldsymbol{w}_{\mathrm{LS-R}}$ using $\lambda = \dfrac{\sigma^2}{\sigma_{\mathbf{w}}^2}$.

- After getting $\boldsymbol{w}_{\mathrm{MAP}}$, we can plug it in $f(\boldsymbol{x}; \boldsymbol{w}_{\mathrm{MAP}})$ to make predictions.

MAP vs. MLE

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

$$= \underset{\boldsymbol{w}}{\text{argmax}}\, p(D|\boldsymbol{w})p(\boldsymbol{w})$$
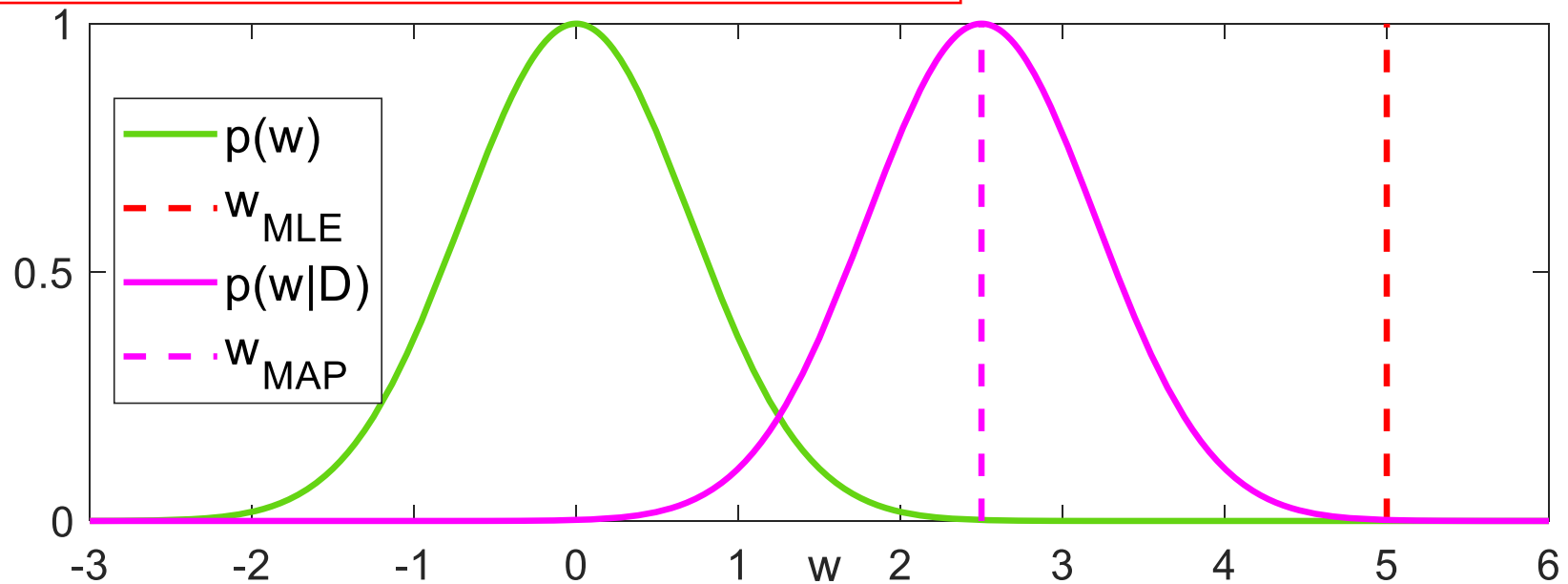
A Full Probabilistic Approach

- However, why settle with a single $\boldsymbol{w}$ when you already have access to $p(\boldsymbol{w}|D)$?

- Using MAP to obtain a single $\boldsymbol{w}$ for prediction **ignores the uncertainty information** represented in $p(\boldsymbol{w}|D)$.

- If not getting a single $\boldsymbol{w}$, how do we make prediction using a probability $p(\boldsymbol{w}|D)$?

A Full Probabilistic Approach

- Instead of making a single prediction $\hat{y}$ given an $\boldsymbol{x}$.
- We can calculate the predictive distribution $p(\hat{y}|\boldsymbol{x}, D)$,
  - Probability of $\hat{y}$ given our dataset and $\boldsymbol{x}$.
- We know
- $p(\hat{y}|\boldsymbol{x}, D) = \int p(\hat{y}|\boldsymbol{x}, \boldsymbol{w}) p(\boldsymbol{w}|D) d\boldsymbol{w}$, (why?)
- Calculate $p(\hat{y}|\boldsymbol{x}, D)$ as a marginalized probability.

- How can we calculate the predictive distribution?
- We can assume $p(\hat{y}|\boldsymbol{x}, \boldsymbol{w}) = N_{\hat{y}}(f(\boldsymbol{x}, \boldsymbol{w}), \sigma^2)$
- We can calculate $p(\boldsymbol{w}|D)$ up to a constant $p(D)$

Calculating Predictive Distribution

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

- $p(\hat{y}|\boldsymbol{x}, \boldsymbol{w}) = N_{\hat{y}}\big(f(\boldsymbol{x}; \boldsymbol{w}), \sigma^2\big)$

- Suppose $f(\boldsymbol{x}; \boldsymbol{w}) = \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}) \rangle$

- **Prove:**

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

- Where $\boldsymbol{\phi}$ is short for $\boldsymbol{\phi}(\boldsymbol{X})$, and $\boldsymbol{w}_{\mathrm{LS-R}}$ is the LS-R solution with $\lambda = \frac{\sigma^2}{\sigma_{\boldsymbol{w}}^2}$.

The Predictive Distribution

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

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

A Full Probabilistic Approach

- With the predictive distribution $p(\hat{y}|\boldsymbol{x}, D)$, we can compute:

- Prediction: $\mathbb{E}_{p(\hat{y}|\boldsymbol{x}, D)}[\hat{y}|\boldsymbol{x}]$,

- Prediction uncertainty: $\text{var}_{p(\hat{y}|\boldsymbol{x}, D)}[\hat{y}|\boldsymbol{x}]$.

- We can also use the predictive distribution to ==calculate other interesting expected values==, as we will see later.

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

Predictive Mean

Computed from Predictive Variance
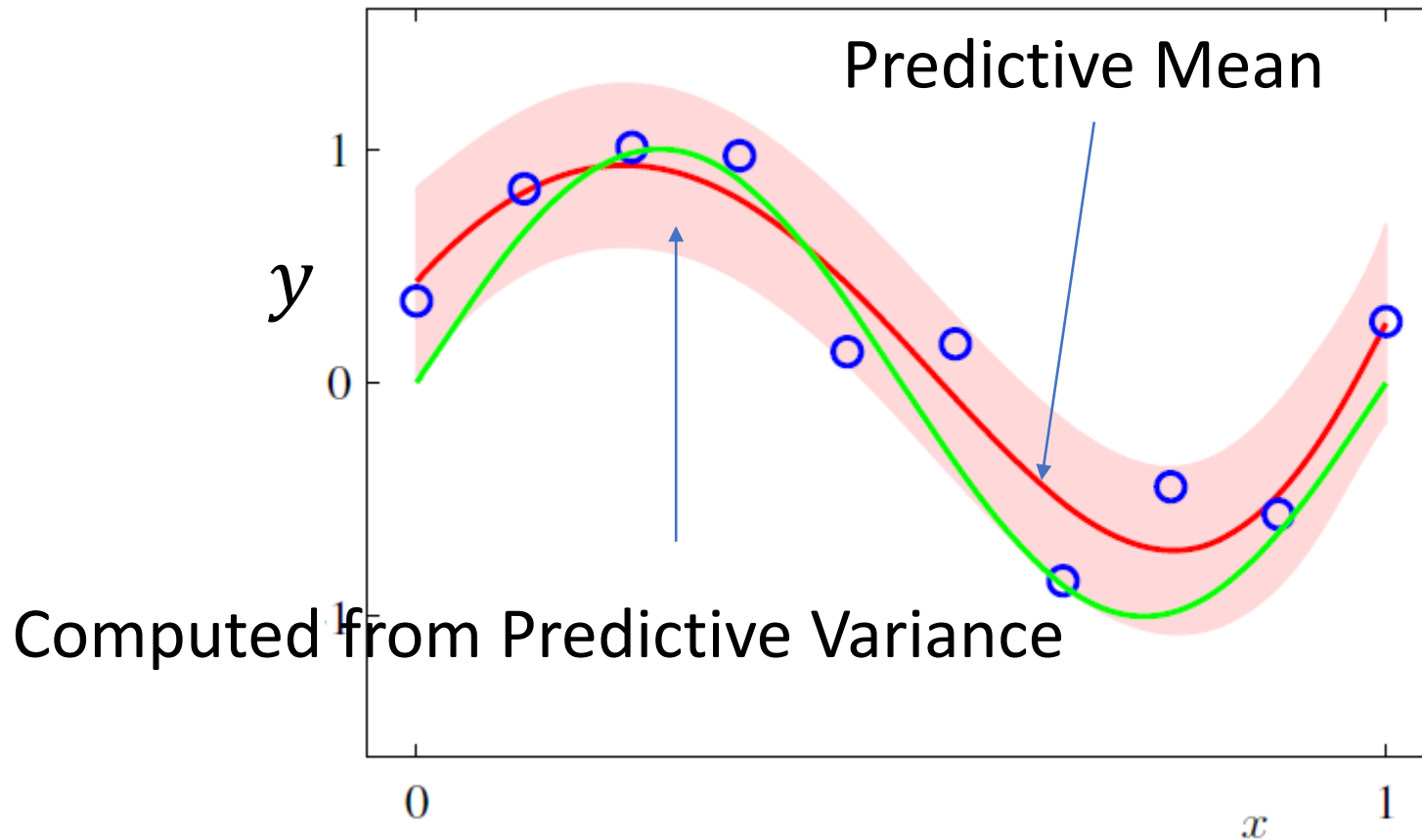
- **PRML, Figure 1.17**

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 $\boldsymbol{w}_{\mathrm{LS}}$ is useless if $n < d$."
  - Is this statement still true for $\boldsymbol{w}_{\mathrm{LS-R}}$?

- Prove the statement on page 21
- Prove the statement on page 25