We’ve covered both parametric and nonparametric models for regression and classification.

- Parametric models summarize the data into a model with a finite number of parameters. E.g., linear regression, logistic regression, neural nets, (linear) SVM, Naïve Bayes, GDA
- Nonparametric models refer back to the data to make predictions. E.g., KNN

The next two lectures are about Bayesian approaches to regression.

- This lecture: Bayesian linear regression, a parametric model
- Next lecture: Gaussian processes, a nonparametric model
Overview

- We’re going to be Bayesian about the parameters of the model, i.e. model them as random variables
  - Do not confuse a Bayesian approach with using Bayes rule! i.e. naïve Bayes and GDA used Bayes’ rule to infer the class, but used point estimates of the parameters.
  - By inferring a posterior distribution over the parameters, the model can know what it doesn’t know.
- How can uncertainty in the predictions help us?
  - Smooth out the predictions by averaging over lots of plausible explanations (just like ensembles!)
  - Assign confidences to predictions
  - Make more robust decisions
  - Guide exploration (focus on areas you’re uncertain about)
    - E.g., Bayesian optimization (see next tutorial)
Recap: Linear Regression

- Given a training set $\mathcal{D}$ of inputs and targets $\{(x^{(n)}, t^{(n)})\}^N_{n=1}$, use linear model with fixed feature mapping $\phi$
  
  $$y = w^\top \phi(x)$$

- Squared error cost (aka least-squares objective):
  
  $$\frac{1}{2} \left[ \sum_{n=1}^{N} (w^\top \phi(x^{(n)}) - t^{(n)})^2 \right]$$

- $L_2$ regularization:
  
  $$\frac{\lambda}{2} \|w\|^2$$

- Let $\Phi$ be data matrix:
  
  $$\Phi = \begin{bmatrix} \phi(x^{(1)})^\top & \vdots & \phi(x^{(N)})^\top \end{bmatrix}$$
Recap: Linear Regression

- Solution 1: solve analytically by setting the gradient to 0
  \[ w = (\Phi^\top \Phi + \lambda I)^{-1} \Phi^\top t \]

- Solution 2: solve approximately using gradient descent
  \[ w \leftarrow (1 - \alpha \lambda)w - \alpha \Phi^\top (y - t) \]
Recap: Linear Regression

- We can give linear regression a probabilistic interpretation by assuming a Gaussian noise model:

\[ t \mid x \sim \mathcal{N}(w^\top \phi(x), \sigma^2) \]

- Minimizing least squares objective is equivalent to maximizing likelihood under this model:

\[
\frac{1}{N} \sum_{n=1}^{N} \log p(t^{(n)} \mid x^{(n)}; w) = \frac{1}{N} \sum_{n=1}^{N} \log \mathcal{N}(t^{(n)}; w^\top \phi(x^{(n)}), \sigma^2) \\
= \frac{1}{N} \sum_{n=1}^{N} \log \left[ \frac{1}{\sqrt{2\pi\sigma}} \exp \left( -\frac{(t^{(n)} - w^\top \phi(x^{(n)}))^2}{2\sigma^2} \right) \right] \\
= \text{const} - \frac{1}{2N\sigma^2} \sum_{n=1}^{N} (t^{(n)} - w^\top \phi(x^{(n)}))^2
\]
Recap: Linear Regression

We can view an $L_2$ regularizer as MAP inference with a Gaussian prior.

Recall MAP inference:

$$\text{arg max}_w \log p(w \mid D) = \text{arg max}_w \left[ \log p(w) + \log p(D \mid w) - \underbrace{\log p(D)}_{\text{constant w.r.t. } w} \right]$$

We just derived the likelihood term $\log p(D \mid w)$:

$$\log p(D \mid w) = -\frac{1}{2N\sigma^2} \sum_{n=1}^{N} (t^{(n)} - w^\top \phi(x^{(n)}))^2 + \text{const}$$

Assume a Gaussian prior, $w \sim \mathcal{N}(\mathbf{m}, \mathbf{S})$. Commonly, $\mathbf{m} = 0$ and $\mathbf{S} = \eta \mathbf{I}$, so:

$$\log p(w) = \log \mathcal{N}(w; \mathbf{0}, \eta \mathbf{I})$$

$$= \log \left( \frac{1}{(2\pi)^{D/2} \eta^{D/2}} \exp \left( -\frac{1}{2\eta} w^\top w \right) \right)$$

$$= -\frac{1}{2\eta} \|w\|^2 + \text{const.}$$

This is just $L_2$ regularization!
Recap: Full Bayesian Inference

Recall: full Bayesian inference makes predictions by averaging over all likely explanations under the posterior distribution.

- Compute posterior using Bayes’ Rule:
  \[ p(w \mid D) \propto p(w)p(D \mid w) \]

- Make predictions using the posterior predictive distribution:
  \[ p(t \mid x, D) = \int p(w \mid D) p(t \mid x, w) \, dw \]

- Doing this lets us quantify our uncertainty.
Bayesian Linear Regression

- **Bayesian linear regression** considers various plausible explanations for how the data were generated.
- It makes predictions using all possible regression weights, weighted by their posterior probability.
  - We can visualize how $p(w|D)$ changes with more data by sampling $w \sim p(w|D)$ and plotting $y = w^T x$:

  ![Graphs showing changes in $p(w|D)$ with more data](image)

  - no observations
  - one observation
  - two observations

- **Prior distribution:** $w \sim \mathcal{N}(0, \eta I)$
- **Likelihood:** $t|x, w \sim \mathcal{N}(w^T \phi(x), \sigma^2)$
- $\eta$ and $\sigma^2$ are hyperparameters
Bayesian Linear Regression: Posterior

Deriving the posterior distribution:

\[
\log p(w | D) = \log p(w) + \log p(D | w) + \text{const}
\]

\[
= -\frac{1}{2\eta} w^T w - \frac{1}{2\sigma^2} \sum_{n=1}^{N} (w^T \phi(x^{(n)}) - t^{(n)})^2 + \text{const}
\]

\[
= -\frac{1}{2\eta} w^T w - \frac{1}{2\sigma^2} \|\Phi w - t\|^2 + \text{const}
\]

\[
= -\frac{1}{2\eta} w^T w - \frac{1}{2\sigma^2} (\Phi w - t)^T (\Phi w - t) + \text{const}
\]

\[
= -\frac{1}{2\eta} w^T w - \frac{1}{2\sigma^2} \left( w^T \Phi^T \Phi w - 2t^T \Phi w + t^T t \right) + \text{const}
\]

\[
= -\frac{1}{2} (w - \mu)^T \Sigma^{-1} (w - \mu) + \text{const} \quad \text{(complete the square!)}
\]

where

\[
\mu = \sigma^{-2} \Sigma \Phi^T t
\]

\[
\Sigma^{-1} = \sigma^{-2} \Phi^T \Phi + \eta^{-1} I
\]
Bayesian Linear Regression: Posterior

\[
\log p(w | D) = -\frac{1}{2}(w - \mu)\top \Sigma^{-1}(w - \mu) + \text{const}
\]

where

\[
\mu = \sigma^{-2} \Sigma \Phi \top t, \quad \Sigma^{-1} = \sigma^{-2} \Phi \top \Phi + \eta^{-1} I
\]

• Hence:

\[
p(w | D) = \exp(-\frac{1}{2}(w - \mu)\top \Sigma^{-1}(w - \mu)) \exp(\text{const})
\]

\[
\propto \exp(-\frac{1}{2}(w - \mu)\top \Sigma^{-1}(w - \mu))
\]

• This is a multivariate Gaussian distribution, i.e.

\[
w | D \sim \mathcal{N}(\mu, \Sigma)
\]
Bayesian Linear Regression: Posterior

- Just showed:

\[ w \mid D \sim \mathcal{N}(\mu, \Sigma) \]
\[ \mu = \sigma^{-2} \Sigma \Phi^\top t \]
\[ \Sigma^{-1} = \sigma^{-2} \Phi^\top \Phi + \eta^{-1} I \]

- Since a Gaussian prior leads to a Gaussian posterior, this means the Gaussian distribution is the conjugate prior for linear regression!

- Compare \( \mu \) with the closed-form solution for linear regression:

\[ \mu = \sigma^{-2} (\sigma^{-2} \Phi^\top \Phi + \eta^{-1} I)^{-1} \Phi^\top t \]
\[ w = (\Phi^\top \Phi + \lambda I)^{-1} \Phi^\top t \]
Bayesian Linear Regression

— Bishop, Pattern Recognition and Machine Learning
Aside: why are likelihood contours lines?

\[ c = \mathcal{N}(t | w_0 x_0 + w_1 x_1, \sigma^2) \]

\[ \implies c = d \exp \left( -\frac{1}{2\sigma^2} (t - w_0 x_0 - w_1 x_1)^2 \right) \]

\[ \implies \sqrt{-2\sigma^2 \log(c/d)} = t - w_0 x_0 - w_1 x_1 \]

Set \( e = \sqrt{-2\sigma^2 \log(c/d)} \):

\[ e = t - w_0 x_0 - w_1 x_1 \]

\[ \implies w_1 = \frac{1}{x_1} (t - w_0 x_0 - e) \]

We find \( w_1 \) is a linear function of \( w_0 \)
Bayesian Linear Regression

- Example with radial basis function (RBF) features

\[ \phi_j(x) = \exp \left( -\frac{(x - \mu_j)^2}{2s^2} \right) \]

- Bishop, Pattern Recognition and Machine Learning

\[ \phi(x) = \begin{pmatrix} \phi_1(x) \\ \vdots \\ \phi_J(x) \end{pmatrix} \]
Bayesian Linear Regression

Functions sampled from the posterior:

— Bishop, Pattern Recognition and Machine Learning
Bayesian Linear Regression

- Posterior predictive distribution:

\[ p(t \mid x, D) = \int \frac{p(t \mid x, w)}{\mathcal{N}(w \mid \mu, \Sigma)} p(w \mid D) \, dw \]

\[ \mathcal{N}(t; w^\top \phi(x), \sigma^2) \mathcal{N}(w; \mu, \Sigma) \]

- Another interpretation: \( t = w^\top \phi(x) + \varepsilon \), where \( \varepsilon \sim \mathcal{N}(0, \sigma^2) \) is independent of \( w \).

- By the linear combination rules for Gaussian random variables, \( t \) is a Gaussian distribution with parameters

\[
\mu_{\text{pred}} = \mu^\top \phi(x) \\
\sigma^2_{\text{pred}} = \phi(x)^\top \Sigma \phi(x) + \sigma^2
\]

- Hence, the posterior predictive distribution is \( \mathcal{N}(t \mid \mu_{\text{pred}}, \sigma^2_{\text{pred}}) \).
Bayesian Linear Regression

Here we visualize confidence intervals based on the posterior predictive mean and variance at each point:

— Bishop, Pattern Recognition and Machine Learning
What do we actually do with the posterior predictive distribution $p(t \mid x, D)$?

Often, we want to make a decision. We can formulate this as minimizing an expected loss under the posterior predictive distribution. This is known as **decision theory**.

Simple example: we have an entire distribution over targets $p(t \mid x, D)$. How should we choose a single prediction to make?

One criterion: choose single prediction $y$ to minimize the expected squared error loss.

$$\arg \min_y \mathbb{E}_{p(t \mid x, D)}[(y - t)^2] = \mathbb{E}_{p(t \mid x, D)}[t] = \mu_{\text{pred}}$$

- Same derivation as bias/variance from Lecture 4

Another criterion: minimize the expected absolute value loss. You can show that you should pick the median of $p(t \mid x, D)$.
Optional material
Occam’s Razor: “Entities should not be multiplied beyond necessity.”
- Named after the 14th century British theologian William of Occam
- Huge number of attempts to formalize mathematically
  - See Domingos, 1999, “The role of Occam’s Razor in knowledge discovery” for a skeptical overview.
- Common misinterpretation: your prior should favor simple explanations
Occam’s Razor (optional)

- Suppose you have a finite set of models, or hypotheses \( \{ \mathcal{H}_i \}_{i=1}^{M} \) (e.g. polynomials of different degrees)
- Posterior inference over models (Bayes’ Rule):

\[
p(\mathcal{H}_i | D) \propto p(\mathcal{H}_i) p(D | \mathcal{H}_i)
\]

- Which of these terms do you think is more important?
- The evidence is also called \textbf{marginal likelihood} since it requires marginalizing out the parameters:

\[
p(D | \mathcal{H}_i) = \int p(w | \mathcal{H}_i) p(D | w, \mathcal{H}_i) \, dw
\]

- If we’re comparing a handful of hypotheses, \( p(\mathcal{H}_i) \) isn’t very important, so we can compare them based on marginal likelihood.
Suppose $M_1$, $M_2$, and $M_3$ denote a linear, quadratic, and cubic model. $M_3$ is capable of explaining more datasets than $M_1$. But its distribution over $\mathcal{D}$ must integrate to 1, so it must assign lower probability to ones it can explain.
Occam’s Razor (optional)

- How does the evidence (or marginal likelihood) penalize complex models?

\[
p(D | H_i) = \int p(D | w, H_i) p(w | H_i)
\]

\[
\simeq p(D | w_{MAP}, H_i) \cdot p(w_{MAP} | H_i) \cdot \Delta w
\]

- Approximating the integral:

\[
p(D | H_i) = \int p(D | w, H_i) p(w | H_i)
\]

\[
\simeq p(D | w_{MAP}, H_i) \cdot p(w_{MAP} | H_i) \cdot \Delta w
\]
Occam’s Razor (optional)

- Multivariate case:
  \[ p(D | H_i) \sim p(D | w_{MAP}, H_i) \cdot p(w_{MAP} | H_i) |A|^{-1/2} \]
  where \( A = \nabla^2_w \log p(D | w, H_i) \)
  - The determinant appears because we’re taking the volume.
  - The more parameters in the model, the higher dimensional the parameter space, and the faster the volume decays.

— Bishop, Pattern Recognition and Machine Learning
Occam’s Razor (optional)

Analyzing the asymptotic behavior:

\[ A = \nabla_w^2 \log p(D \mid w, \mathcal{H}_i) \]

\[ = \sum_{j=1}^{N} \nabla_w^2 \log p(y_i \mid x_i, w, \mathcal{H}_i) \]

\[ \triangleq A_i \]

\[ \approx N \mathbb{E}[A_i] \]

log Occam factor = \( \log p(w_{\text{MAP}} \mid \mathcal{H}_i) + \log |A|^{-1/2} \)

\[ \approx \log p(w_{\text{MAP}} \mid \mathcal{H}_i) + \log |N \mathbb{E}[A_i]|^{-1/2} \]

\[ = \log p(w_{\text{MAP}} \mid \mathcal{H}_i) - \frac{1}{2} \log |\mathbb{E}[A_i]| - \frac{D \log N}{2} \]

\[ = \text{const} - \frac{D \log N}{2} \]

Bayesian Information Criterion (BIC): penalize the complexity of your model by \( \frac{1}{2} D \log N \).
Occam’s Razor (optional)

Summary

\[ p(\mathcal{H}_i \mid \mathcal{D}) \propto p(\mathcal{H}_i) p(\mathcal{D} \mid \mathcal{H}_i) \]
\[ p(\mathcal{D} \mid \mathcal{H}_i) \approx p(\mathcal{D} \mid \mathbf{w}_{\text{MAP}}, \mathcal{H}_i) p(\mathbf{w}_{\text{MAP}} \mid \mathcal{H}_i) |\mathbf{A}|^{-1/2} \]

Asymptotically, with lots of data, this behaves like

\[ \log p(\mathcal{D} \mid \mathcal{H}_i) = \log p(\mathcal{D} \mid \mathbf{w}_{\text{MAP}}, \mathcal{H}_i) - \frac{1}{2} D \log N. \]

Occam’s Razor is about integration, not priors (over hypotheses).