# Machine Learning:
# A Probabilistic Perspective

CS 480
Intro to AI

# Why probabilities?

Gives us a formal way to talk about **noise** (Frequentist)

Gives us a formal way to talk about **belief** (Bayesian)

Useful probability facts/definitions:

Notation

$$p(X) = \text{Probability of } X$$

$$0 \leq p(X) \leq 1$$

$$\int p(x)dx = 1$$

Independence

$$p(X,Y) = p(X) \cdot p(Y) \Leftrightarrow X \text{ and } Y \text{ are independent}$$

Conditional

$$p(X \mid Y) = \frac{p(X,Y)}{p(Y)}, \text{ if } p(Y) > 0$$

Bayes Rule

$$p(X \mid Y) = \frac{p(Y \mid X)p(X)}{p(Y)}$$

# Expected Value

Useful facts about how **expectation** works

Definition

$$\mathbb{E}[X] = \int x \cdot p(x) dx$$

Linearity

$$\mathbb{E}[X + Y] = \mathbb{E}[X] + \mathbb{E}[Y]$$
$$\mathbb{E}[aX + b] = a\mathbb{E}[X] + b$$

Conditional

$$\mathbb{E}[X \mid Y] = \int x \cdot p(x \mid y) dx$$

Total Expectation

$$\mathbb{E}[X] = \mathbb{E}_{\mathcal{Y}}[\mathbb{E}_{\mathcal{X}}[X \mid Y]]$$

Expectation is a statistical measure of the **central tendency** of a random variable, and tells us where the "middle" of the distribution of a random variable is

# Variance

Useful facts about variance

Definition
$$\text{Var}[X] = \mathbb{E}\left[(X - \mathbb{E}[X])^2\right] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$$

Constants
$$\text{Var}[X + a] = \text{Var}[X]$$
$$\text{Var}[aX] = a^2\text{Var}[X]$$
$$\text{Var}[a] = 0$$

The variance is a statistical measure of **deviation from the mean** and gives a number for how "noisy" a random variable is.

# The Bias-Variance tradeoff Proof (1)

Start with

Model, trained on S

$$\mathbb{E}_S\left[(y - h_\theta(\mathbf{x}))^2\right]$$

End with

Arbitrary output

Arbitrary input

$$(y - \mathbb{E}[h_\theta(\mathbf{x})])^2 + \mathrm{Var}[h_\theta(\mathbf{x})]$$

Where **y** is an arbitrary output, and **x** is an arbitrary input, and the expectation is taken with respect to the **distribution** of the **training data**

# The Bias-Variance tradeoff Proof (2)

$$\mathbb{E}_S[(y - h_\theta(\mathbf{x}))^2] = \mathbb{E}_S[(y - \mathbb{E}_S[h_\theta(\mathbf{x})] + \mathbb{E}_S[h_\theta(\mathbf{x})] - h_\theta(\mathbf{x}))^2]$$

$$= \mathbb{E}_S[(y - \mathbb{E}_S[h_\theta(\mathbf{x})])^2] +$$

$$\mathbb{E}_S[(\mathbb{E}_S[h_\theta(\mathbf{x})] - h_\theta(\mathbf{x}))^2] +$$

$$\mathbb{E}_S[2(y - \mathbb{E}_S[h_\theta(\mathbf{x})])(\mathbb{E}_s[h_\theta(\mathbf{x})] - h_\theta(\mathbf{x}))]$$

We **add** and **subtract** E[h(x)], then (partially) expand out the square

# The Bias-Variance tradeoff Proof (3)

Let's take a closer look at the last term

$$\mathbb{E}_S[2(y - \mathbb{E}_S[h_\theta(\mathbf{x})])(\mathbb{E}_s[h_\theta(\mathbf{x})] - h_\theta(\mathbf{x}))] = 2(y - \mathbb{E}_S[h_\theta(\mathbf{x})])\mathbb{E}_S[(\mathbb{E}_S[h_\theta(\mathbf{x})] - h_\theta(\mathbf{x}))]$$
$$= 2(y - \mathbb{E}_S[h_\theta(\mathbf{x})])(\mathbb{E}_S[h_\theta(\mathbf{x})] - \mathbb{E}_S[h_\theta(\mathbf{x})])$$
$$= 2(y - \mathbb{E}_S[h_\theta(\mathbf{x})])(0)$$
$$= 0$$

Since **y** and **E[h(x)]** are **constants**, we can push the expectation inside, and the cross term vanishes!

# The Bias-Variance tradeoff Proof (4)

$$\mathbb{E}_S[(y - h_\theta(\mathbf{x}))^2] = \mathbb{E}_S[(y - \mathbb{E}_S[h_\theta(\mathbf{x})])^2] + \mathbb{E}_S[(\mathbb{E}_S[h_\theta(\mathbf{x})] - h_\theta(\mathbf{x}))^2]$$

$$= \mathbb{E}_S[(y - \mathbb{E}_S[h_\theta(\mathbf{x})])^2] + \mathrm{Var}[h_\theta(\mathbf{x})]$$

$$= (y - \mathbb{E}_S[h_\theta(\mathbf{x})])^2 + \mathrm{Var}[h_\theta(\mathbf{x})]$$

$$= \mathrm{Bias}(h_\theta(\mathbf{x}))^2 + \mathrm{Var}[h_\theta(\mathbf{x})]$$

So the expected loss of any hypothesis is a combination of its **bias** and its **variance**.
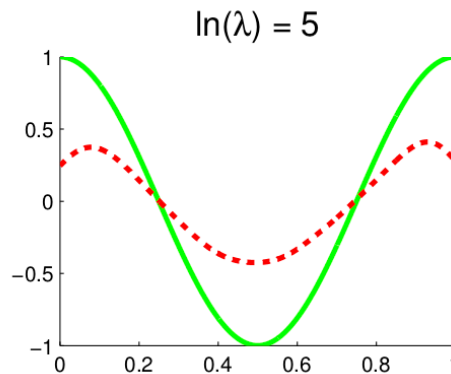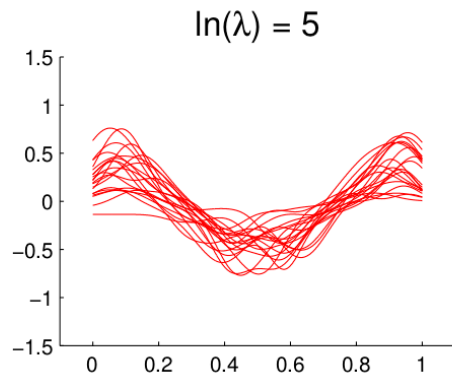
**Bias** is reduced by **increasing** complexity
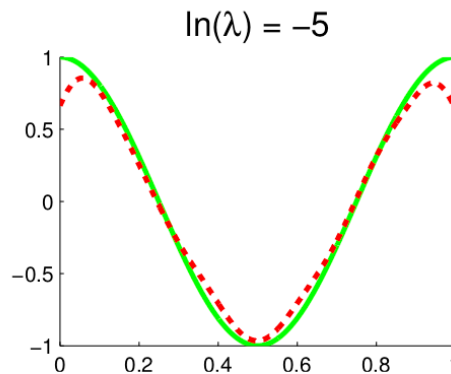**Variance** can be reduced by **decreasing** complexity

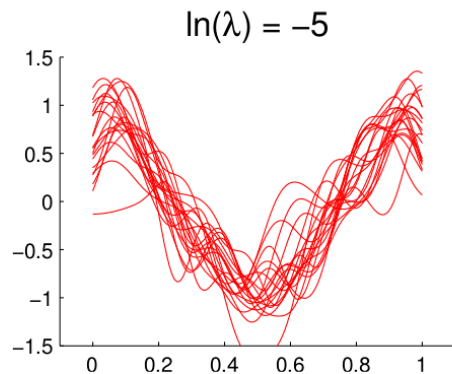# The Bias-Variance tradeoff visually

$\lambda$ = regularization

# Minimizing the Expected Loss (1)

Let's revisit how we choose the "best" hypothesis. To start, what's the expected loss for an arbitrary hypothesis at a **given** datapoint?

$$\mathbb{E}[(h(\mathbf{x}) - y)^2 \mid \mathbf{x}] = \mathbb{E}\left[(h(\mathbf{x}) - \mathbb{E}[y \mid \mathbf{x}] + \mathbb{E}[y \mid \mathbf{x}] - y)^2\right]$$

$$= \mathbb{E}\left[(h(\mathbf{x}) - \mathbb{E}[y \mid \mathbf{x}])^2 \mid \mathbf{x}\right] +$$

$$\mathbb{E}\left[(\mathbb{E}[y \mid \mathbf{x}] - y)^2 \mid \mathbf{x}\right] +$$

$$2\mathbb{E}\left[(h(\mathbf{x}) - \mathbb{E}[y \mid \mathbf{x}])(\mathbb{E}[y \mid \mathbf{x}] - y) \mid \mathbf{x}\right]$$

$$= \mathbb{E}\left[(h(\mathbf{x}) - \mathbb{E}[y \mid \mathbf{x}])^2 \mid \mathbf{x}\right] + \mathbb{E}\left[(\mathbb{E}[y \mid \mathbf{x}] - y)^2 \mid \mathbf{x}\right]$$

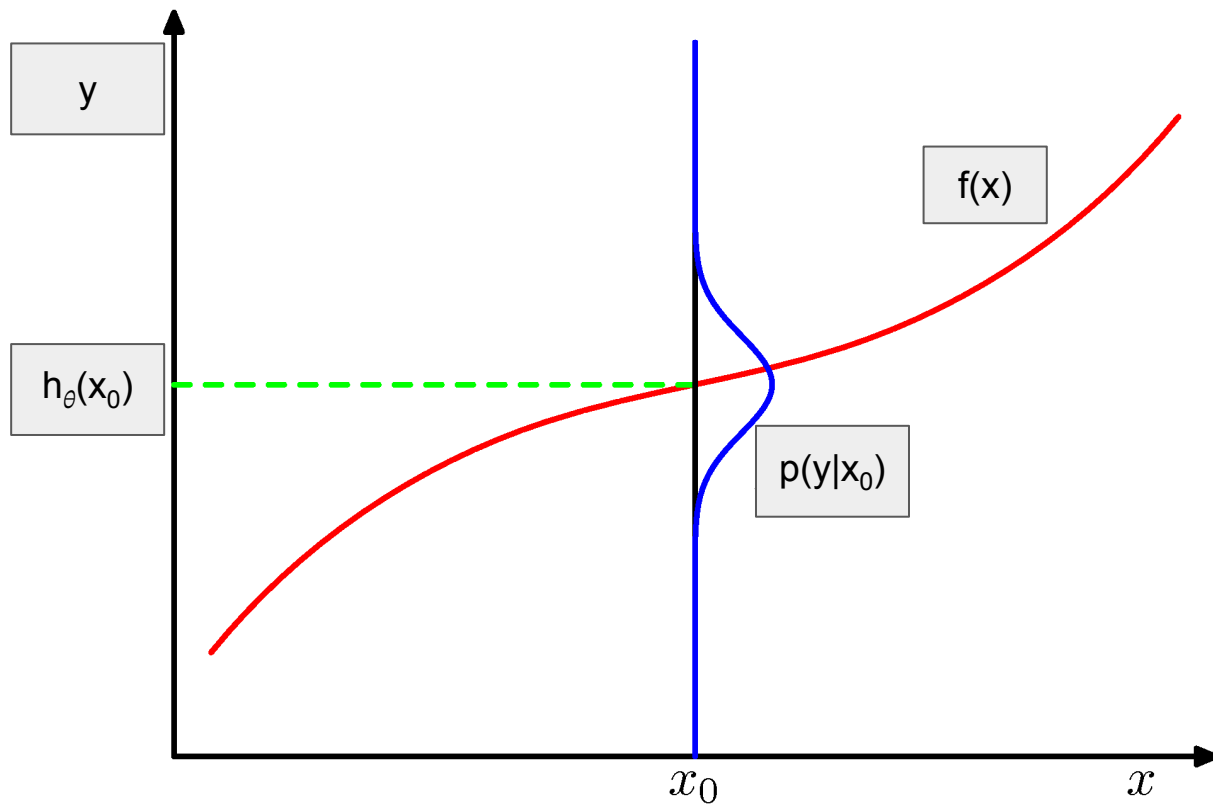$$\geq \mathbb{E}\left[(\mathbb{E}[y \mid \mathbf{x}] - y)^2 \mid \mathbf{x}\right]$$

# Minimizing the Expected Loss (2)

$$\mathbb{E}[(h(\mathbf{x}) - y)^2 \mid \mathbf{x}] \geq \mathbb{E}[(\mathbb{E}[y \mid \mathbf{x}] - y)^2 \mid \mathbf{x}]$$

$$\mathbb{E}[\mathbb{E}[(h(\mathbf{x}) - y)^2 \mid \mathbf{x}]] \geq \mathbb{E}[\mathbb{E}[(\mathbb{E}[y \mid \mathbf{x}] - y)^2 \mid \mathbf{x}]]$$

$$\mathbb{E}[(h(\mathbf{x}) - y)^2] \geq \mathbb{E}[(\mathbb{E}[y \mid \mathbf{x}] - y)^2]$$

$$\mathbb{E}[\mathcal{J}_S(h)] \geq \mathbb{E}[\mathcal{J}_S(\mathbb{E}[y \mid \mathbf{x}])]$$

No hypothesis can do better than predicting the expected value of y given x!

This makes sense if we think of the **noise** in the training data as being a small additive error

# Conditional Expectation - Graphical View

# Modeling Noise Probabilistically

Let's assume there is a "ground truth" deterministic function which generates our data, and that the samples in our dataset **S** have some small noise.

$$y^{(i)} = f(\mathbf{x}^{(i)}) + \epsilon^{(i)}$$

$$\epsilon^{(i)} \sim \mathcal{N}(0, \sigma^2)$$

$$p\left((\mathbf{x}^{(i)}, y^{(i)}) \mid f\right) = \mathcal{N}(f(\mathbf{x}^{(i)}), \sigma^2)$$

For a model parameterized by $\theta$, we can talk about the **likelihood** that a fixed set of data was generated by that model.

$$\mathcal{L}(h_\theta; S) = p(S \mid h_\theta)$$

$$= p((\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \ldots, (\mathbf{x}^{(N)}, y^{(N)}) \mid h_\theta)$$

# Maximum Likelihood Estimation (1)

If we assume the training data is drawn I.I.D (**independent** and **identically distributed**), we can factor the likelihood

$$\mathcal{L}(h_\theta; S) = \prod_{i=1}^{N} p((\mathbf{x}^{(i)}, y^{(i)}) \mid h_\theta)$$

Which h maximizes the likelihood?

$$\arg\max_{h \in \mathcal{H}} \mathcal{L}(h_\theta; S) = \arg\max_{h \in \mathcal{H}} \log \mathcal{L}(h_\theta; S)$$

$$= \arg\min_{h \in \mathcal{H}} (-\log \mathcal{L}(h_\theta; S))$$

Maximizing the likelihood is the same thing as **minimizing** the **negative log-likelihood** (NLL)

# Maximum Likelihood Estimation (2)

Putting it together in the case of Gaussian noise

$$\arg\max_{h\in\mathcal{H}}\mathcal{L}(h_\theta; S) = \arg\min_{h\in\mathcal{H}} -\log\prod_{i=1}^{N} p((\mathbf{x}^{(i)}, y^{(i)}) \mid h_\theta)$$

$$= \arg\min_{h\in\mathcal{H}} -\log\prod_{i=1}^{N} \exp\left\{\frac{-(y^{(i)} - h_\theta(\mathbf{x}^{(i)}))^2}{2\sigma^2}\right\}$$

$$= \arg\min_{h\in\mathcal{H}} -\sum_{i=1}^{N} \frac{-(y^{(i)} - h_\theta(\mathbf{x}^{(i)}))^2}{2\sigma^2}$$

$$= \arg\min_{h\in\mathcal{H}} \sum_{i=1}^{N} (y^{(i)} - h_\theta(\mathbf{x}^{(i)}))^2$$

# Maximum Likelihood Estimation

The MLE estimate **also** minimizes the sum of squared errors!

$$\arg \max_{h \in \mathcal{H}} \mathcal{L}(h_\theta; S) = \arg \min_{h \in \mathcal{H}} \sum_{i=1}^{N} (y^{(i)} - h_\theta(\mathbf{x}^{(i)}))^2$$

Notes:

- We made no assumption about the hypothesis class, just the distribution of errors (zero mean normal)
- Minimizing the sum of squared errors is **equivalent** to assuming that the data has Gaussian distributed noise

# Summary and preview

Wrapping up

- Probabilities let us formalize our assumptions about noise and loss functions
- The Bias-Variance tradeoff shows us how complexity, bias, and variance are related
- Regression can be thought of as estimating the conditional expectation
- Maximum Likelihood Estimation under the assumption of Gaussian noise and IID data is equivalent to minimizing the sum of squared errors

Next time

- Moving from Regression to Classification