Learning from Data
2. Probabilistic Framework

Spring 2017

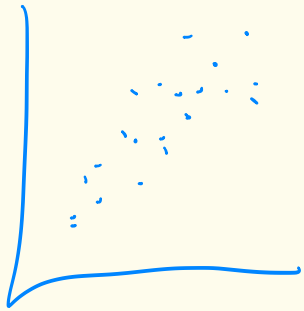
Need for Probability

- Inductive reasoning:
  - going from particular instances to broader generalizations, e.g., 1➔2, 2➔4, 3➔?
  - inherently uncertain: allows for the possibility of the conclusion to be false even if all premises are true
- Randomness (variability/complexity/uncertainty) inherent in most real-world data, e.g.,

- Examples, features, and labels ➔random variables

$D = ((x_1, y_1), \ldots, (x_n, y_n))$

$y = mx + b$



Want to know: $P(\theta \mid D)$

That is, what are most likely parameters given the data. This is the posterior.

Start with $P(D \mid \theta)$ distribution of data conditioned on parameters. This is called the likelihood. Also need $P(\theta)$ called the prior, which is initial belief about distribution of parameters (how likely we think parameters are before considering the data).

ML: choose θ that maximizes likelihood (mode of likelihood)
(What θ explains data the best [without knowing prior PDF])

MAP: choose θ that maximizes posterior (mode) ex , $\int p = 1$

If prior is uniform MAP reduces to ML

Probabilistic Model for Data

- Training data samples modeled as unordered collection of Independent and Identically Distributed (IID) samples drawn from some unknown underlying joint CDF/pmf/pdf:

$$(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n) \sim \text{IID } p(\mathbf{x}, y)$$

- Motivation/justification:
  - mathematical convenience
  - De Finetti's theorem: exchangeable (i.e., unordered) observations are conditionally independent given some underlying latent variable

- $(\mathbf{X}, Y)$ → random variables versus $(\mathbf{x}, y)$ → specific realization
- $p(\mathbf{x}, y)$:
  - joint pmf if both $\mathbf{x}, y$ are jointly discrete
  - joint pdf if they are jointly continuous
  - mixed probability distribution if one is discrete and the other continuous

Risk of a decision rule h

$E[\ell(h(x),y)] = \int \ell(h(x),y)\, p(x,y)\, dx\, dy$

$\ell^{ex.}\ (y-h(x))^2$

- Risk of h:
  - $R(h) = E[\ell(\mathbf{X},Y,h)]$ where $(\mathbf{X},Y) \sim p(\mathbf{x},y)$
  - Also called: Expected loss, Generalization error
  - This is the "true" or idealized risk
  - Need to know $p(\mathbf{x},y)$ to calculate it

- Empirical Risk of h: an estimate of the its true risk based on training samples:

$$R_{\text{emp}}(h) = \frac{1}{n}\sum_{j=1}^{n} \ell(\mathbf{X}_j, Y_j, h)$$

- Here, the empirical risk is a random variable being a function of the training data which are themselves IID random variables

$R_{emp}(h) = \frac{1}{n}\sum_{j=1}^{n} \ell(X_j, Y_j, h) \xrightarrow{n\to\infty} E[\ell(X,Y,h)] = R(h)$

Risk = expected loss

needs $p(x,y)$

Risk of a decision rule h

- Expected value of empirical risk = risk

$$E\left[R_{\text{emp}}(h)\right] = R(h)$$

- Thus, empirical risk provides an unbiased estimate of the risk

- Also, by the Law of Large Numbers,

$$R_{\text{emp}}(h) = \frac{1}{n}\sum_{j=1}^{n} \ell(\mathbf{X}_j, Y_j, h) \xrightarrow{n\to\infty} E[\ell(\mathbf{X}, Y, h)] = R(h)$$

Risk of a decision rule h

- Conditioned on a specific set of training samples:

$$\{(\mathbf{X}_1, Y_1) \ldots, (\mathbf{X}_n, Y_n)\} = \mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$$

the empirical risk is a deterministic number:

$$R_{\text{emp}}(h)|\mathcal{D} = \frac{1}{n}\sum_{j=1}^{n} \ell(\mathbf{x}_j, y_j, h) = \mathcal{L}_{\text{train}}(h, \alpha, \lambda = 0)$$

↑ Tuning Parameters

↑ complexity scaling param.

Ex. Polynomial degree in regression

- Useful: If we define $(\mathbf{X}_{\text{emp}}, Y_{\text{emp}}) \sim \text{Uniform}(\mathcal{D})$ then

$$R_{\text{emp}}(h)|\mathcal{D} = \frac{1}{n}\sum_{j=1}^{n} \ell(\mathbf{x}_j, y_j, h) = E[\ell(\mathbf{X}_{\text{emp}}, Y_{\text{emp}}, h)]$$

Statistical Decision Theory

- is the study of optimum decision rules when $p(\mathbf{x},y)$ is known

- Bayes Risk: minimum attainable risk over the family of ALL possible decision rules, with known $p(\mathbf{x},y)$:

$$R_{\text{Bayes}} = \inf_{h \text{ measurable}} R(h)$$

- Bayes hypothesis / decision rule: a rule h_{Bayes} which attains the Bayes risk:

$$R(h_{\text{Bayes}}) = R_{\text{Bayes}}$$

  - This is the best possible decision rule when $p(\mathbf{x},y)$ is known

Statistical Decision Theory

- Best-in-family hypothesis / decision rule: a rule $h_{\mathcal{H}}$ which attains the smallest risk among all h in $\mathcal{H}$:

$$R(h_{\mathcal{H}}) = \inf_{h \in \mathcal{H}} R(h)$$

risk: expected loss

- Risk decomposition:

risk min risk risk best risk best risk min risk

$$\underset{\text{known } p(x,y)}{R(h)} - R_{\text{Bayes}} = \underbrace{(R(h) - R(h_{\mathcal{H}}))}_{\text{estimation error}} + \underbrace{(R(h_{\mathcal{H}}) - R_{\text{Bayes}})}_{\text{approximation error}}$$

  - approximation error: measures how well Bayes risk can be approximated using $\mathcal{H}$; property of $\mathcal{H}$, its richness; inaccessible since joint distribution is unknown
  - estimation error: measures quality of h w.r.t. $h_{\mathcal{H}}$

Best-in-family decision rule

- Result: the best-in-family decision for a given $\mathbf{x}$ is the one that minimizes the in-family posterior risk for $\mathbf{x}$:

$$\begin{aligned} h_{\mathcal{H}} &= \arg\min_{h\in\mathcal{H}} E[\ell(\mathbf{X}, Y, h)] \\ \Leftrightarrow \forall \mathbf{x},\ h_{\mathcal{H}}(\mathbf{x}) &= \arg\min_{h\in\mathcal{H}} \underbrace{E[\ell(\mathbf{x}, Y, h)|\mathbf{X}=\mathbf{x}]}_{\text{posterior risk given } \mathbf{x}} \end{aligned}$$

- *Proof*: For all h in $\mathcal{H}$ and all $\mathbf{x}$, we have

$$E[\ell(\mathbf{x}, Y, h_{\mathcal{H}})|\mathbf{X}=\mathbf{x}] \leq E[\ell(\mathbf{x}, Y, h)|\mathbf{X}=\mathbf{x}]$$

equal when $h = h_{\mathcal{H}}$

The result follows by taking expectation on both sides with respect to $\mathbf{X}$

Optimum (Bayes) decision rule

- Note: the loss function is really a function of only y and $h(\mathbf{x})$: $\ell(\mathbf{x}, y, h) \equiv \ell(y, h(\mathbf{x}))$

- Result: For all $\mathbf{x}$ the Bayes decision rule is given by:

$R_{Bayes} = R(h_{Bayes})$

$$h_{\text{Bayes}}(\mathbf{x}) = \arg\min_{\hat{y} \in \mathcal{Y}} E[\ell(Y, \hat{y}) | \mathbf{X} = \mathbf{x}]$$

- *Proof*: For all h and all $\mathbf{x}$, we have

$$\min_{\hat{y} \in \mathcal{Y}} E[\ell(Y, \hat{y}) | \mathbf{X} = \mathbf{x}] \leq E[\ell(Y, h(\mathbf{x})) | \mathbf{X} = \mathbf{x}]$$

The result follows by taking expectation on both sides with respect to $\mathbf{X}$

Examples of Bayes decision rules

- Minimum Mean Square Error (MMSE) Estimate
- Here, $\mathcal{X} = \mathbb{R}^d, \mathcal{Y} = \mathbb{R}^m, \ell(\mathbf{x}, y, h) = \|y - h(\mathbf{x})\|^2$

- Result:
  $h_{\text{Bayes}}(\mathbf{x}) = h_{\text{MMSE}}(\mathbf{x})$ = posterior mean = $E[Y|\mathbf{X} = \mathbf{x}]$

- *Proof*: If $g(\mathbf{x}) = E[Y|\mathbf{X}=\mathbf{x}]$ then $E[(Y - g(\mathbf{x}))|\mathbf{X}=\mathbf{x}] = 0$.
  Thus, for any h,

$$E\left[\|Y - h(\mathbf{x})\|^2 \middle| \mathbf{X} = \mathbf{x}\right] = E\left[\|Y - g(\mathbf{x})\|^2 \middle| \mathbf{X} = \mathbf{x}\right] + E\left[\|g(\mathbf{x}) - h(\mathbf{x}))\|^2 \middle| \mathbf{X} = \mathbf{x}\right]$$

Examples of Bayes decision rules

- Minimum Mean Absolute Error (MMAE) Estimate
- Here, $\mathcal{X} = \mathcal{Y} = \mathbb{R}, \ell(\mathbf{x}, y, h) = |y - h(\mathbf{x})|$
- Result:

  $h_{\text{Bayes}}(\mathbf{x}) = h_{\text{MMAE}}(\mathbf{x})$ = posterior median, i.e.,

$$\int_{-\infty}^{h_{\text{MMAE}(\mathbf{x})}} p(y|\mathbf{x})dy = \int_{h_{\text{MMAE}(\mathbf{x})}}^{+\infty} p(y|\mathbf{x})dy$$

- *Proof*: Set $\frac{\partial}{\partial \hat{y}} \, E\left[|Y - \hat{y}| \,\middle|\, \mathbf{X} = \mathbf{x}\right]\Big|_{\hat{y}=h_{\text{Bayes}}(\mathbf{x})} = 0$ where

$$E\left[|Y - \hat{y}| \,\middle|\, \mathbf{X} = \mathbf{x}\right] = \int_{-\infty}^{\hat{y}} (\hat{y} - y)p(y|\mathbf{x})dy + \int_{\hat{y}}^{+\infty} (y - \hat{y})p(y|\mathbf{x})dy$$

Examples of Bayes decision rules

- Minimum Probability of Error (MPE) Estimate
- Here, $\mathcal{Y} = \{1, \ldots . m\}, \ell(\mathbf{x}, y, h) = 1\left(y \neq h(\mathbf{x})\right)$, i.e., unit cost for error
- $E[1(Y \neq h(\mathbf{X}))] = P(Y \neq h(\mathbf{X})) = P(\text{Error})$
- Result:

  $h_{\text{Bayes}}(\mathbf{x}) = h_{\text{MPE}}(\mathbf{x})$ = posterior mode = $\arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x})$

- *Proof*: Let $\hat{y} = h(\mathbf{x})$. Then for each $\mathbf{x}$

$$E[1(Y \neq \hat{y})|\mathbf{X} = \mathbf{x}] = 1 - E[1(Y = \hat{y})|\mathbf{X} = \mathbf{x}] = 1 - p(\hat{y}|\mathbf{x})$$

$$\Rightarrow \arg\min_{\hat{y}} E[1(Y \neq \hat{y})|\mathbf{X} = \mathbf{x}] = \arg\max_{\hat{y}} p(\hat{y}|\mathbf{x})$$

Examples of Bayes decision rules

Feature space $\mathcal{X}$	Label space $\mathcal{Y}$	Loss function	$\ell(\mathbf{x}, y, h)$	Bayes rule name	$h_{\text{Bayes}}(\mathbf{x})$	Posterior property:
$\mathbb{R}^d$	$\mathbb{R}^m$	Squared error	$\|y - h(\mathbf{x})\|^2$	Minimum Mean Squared Error (MMSE)	$E[Y\|\mathbf{X} = \mathbf{x}]$ $= \int_{\mathbb{R}^m} y\, p(y\|\mathbf{x}) dy$	Mean
$\mathbb{R}$	$\mathbb{R}$	Absolute error	$\|y - h(x)\|$	Minimum Mean Absolute Error (MMAE)	$\int_{-\infty}^{h_{\text{Bayes}}(\mathbf{x})} p(y\|\mathbf{x}) dy$ $= \int_{h_{\text{Bayes}}(\mathbf{x})}^{+\infty} p(y\|\mathbf{x}) dy$	Median
$\mathcal{X}$	$\{1, \ldots, m\}$	0-1 loss	$1(y \neq h(\mathbf{x}))$	Minimum Probability of Error (MPE)	$\arg\max_{y \in \mathcal{Y}} p(y\|\mathbf{x})$	Mode

Notation: max, argmax, sup, inf, etc.

- Explanation through example:
  - Say $f(z) = \sin(2\pi z)$

A	$\sup_{z \in A} f(z)$	$\max_{z \in A} f(z)$	$\arg\max_{z \in A} f(z)$	$\inf_{z \in A} f(z)$	$\min_{z \in A} f(z)$	$\arg\min_{z \in A} f(z)$
$[0,1]$	1	1	$\{1/4\}$	-1	-1	$\{3/4\}$
$[0,2]$	1	1	$\{1/4, 5/4\}$	-1	-1	$\{3/4, 7/4\}$
$[0,1/4)$	1	not reached	$\{\}$	0	0	$\{0\}$
$(0,1/4]$	1	1	$\{1/4\}$	0	not reached	$\{\}$
$(0,1/4)$	1	not reached	$\{\}$	0	not reached	$\{\}$

Maximum Aposteriori Probability (MAP) Decision Rule

- $h_{\text{MAP}}(\mathbf{x})$ = posterior mode = $\arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x})$

- Alternative forms:

$$
\begin{aligned}
h_{\text{MAP}}(\mathbf{x}) &= \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}) \\
&= \arg\max_{y \in \mathcal{Y}} p(\mathbf{x}, y) \\
&= \arg\max_{y \in \mathcal{Y}} p(\mathbf{x}|y)p(y) \\
&= \arg\min_{y \in \mathcal{Y}} -\ln p(\mathbf{x}|y) - \ln p(y)
\end{aligned}
$$

posterior

ranging through y, so x remains fixed

$p(x,y) = p(x)\,p(y|x)$

like scaling, doesn't change location of maximum

likelihood of x given y

+ to −
max to min

If true value was y, how likely is x

Maximum Likelihood (ML) Decision Rule

- $h_{\text{ML}}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(\mathbf{x}|y)$

- Result: If $Y \sim \text{Uniform}(\mathcal{Y})$ then the MAP rule reduces to the ML rule, i.e.,

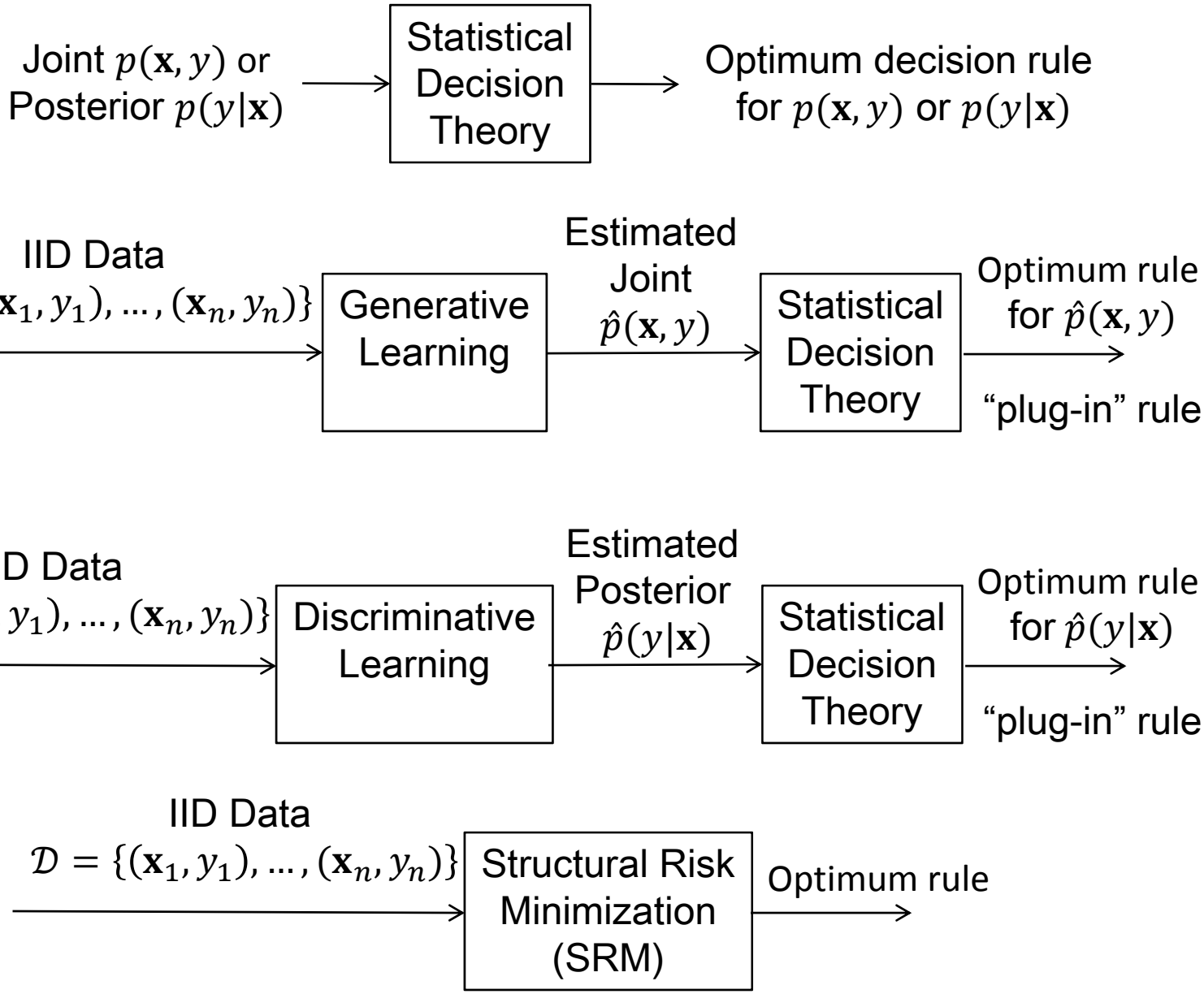
$$h_{\text{MAP}}(\mathbf{x}) = h_{\text{ML}}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(\mathbf{x}|y)$$

- *Proof*: For any $\mathbf{x}$, since Y is uniformly distributed,

$$p(y|\mathbf{x}) \propto p(\mathbf{x}|y) \cdot \underbrace{p(y)}_{\text{does not change with } y}$$

Proportional

Statistical Learning



Alternative forms:

$$h_{\text{MAP}}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(\mathbf{x}, y) = \arg\max_{y \in \mathcal{Y}} p(\mathbf{x}|y)p(y)$$

Statistical Learning

- Family of probability models $\{ p(\mathbf{x},y|\theta) \}$

- Parametric vs Non-parametric models
  - Parametric: fixed number of parameters, e.g., Gaussian with mean vector and covariance matrix
  - Non-parametric: number of parameters not fixed in advance, but grows with the amount of training data, e.g., series expansions of pdfs, e.g., Fourier/Wavelet, where the # terms depends on training data; Kernel Density Estimation (KDE), etc.

- Generative vs Discriminative Learning
  - Generative: training data $\mathcal{D}$ → $p(\mathbf{x},y|\hat{\theta})$ full model estimate. Can use this to generate samples of $(\mathbf{x},y)$
  - Discriminative: training data $\mathcal{D}$ → $p(y|\mathbf{x},\hat{\theta})$ posterior model estimate. Cannot generate $(\mathbf{x},y)$ since $p(\mathbf{x}|\hat{\theta})$ not learned. Recall that the posterior model is sufficient for optimum decision making.

Frequentist and Bayesian Generative Learning

- Frequentist: parameters θ viewed as deterministic unknown and estimated from training data via ML

$$\begin{aligned}
\widehat{\theta}_{\mathrm{ML}} &= \arg\max_{\theta} p(\mathcal{D}|\theta) \\
&= \arg\max_{\theta} \left[\prod_{j=1}^{n} p(\mathbf{x}_j, y_j|\theta) \right] \\
&= \arg\min_{\theta} \left[\frac{1}{n} \sum_{j=1}^{n} \underbrace{-\ln p(\mathbf{x}_j, y_j|\theta)}_{\text{like a loss function}} \right]
\end{aligned}$$

- analogous to ERM with a loss function defined by a generative probability model

Frequentist and Bayesian Generative Learning

- Bayesian: parameters θ viewed as random with given prior $\pi(\theta)$ and estimated from $\mathcal{D}$ via MAP

$$\begin{aligned}
\widehat{\theta}_{\mathrm{MAP}} &= \arg\max_{\theta} p(\theta|\mathcal{D}) \\
&= \arg\max_{\theta} \left[p(\mathcal{D}|\theta)\ \pi(\theta)\right] \\
&= \arg\min_{\theta} \left[\frac{1}{n}\sum_{j=1}^{n} \underbrace{-\ln p(\mathbf{x}_j, y_j|\theta)}_{\text{like a loss function}} - \frac{1}{n}\ln \pi(\theta)\right]
\end{aligned}$$

- analogous to SRM with a loss function defined a generative probability model and a structure penalty (complexity regularization) term determined by the prior

Bayesian Generative Learning as Structural Risk Minimization

- Bayesian Generative Learning:

$$\widehat{\theta}_{\mathrm{MAP}} = \arg\min_{\theta} \left[\frac{1}{n} \sum_{j=1}^{n} \underbrace{-\ln p(\mathbf{x}_j, y_j | \theta)}_{\text{like a loss function}} - \frac{1}{n} \ln \pi(\theta) \right]$$

- Structural Risk Minimization:

$$h^{\mathrm{SRM}}(\alpha, \lambda) = \arg\min_{h \in \mathcal{H}} \left[\underbrace{\frac{1}{n} \sum_{j=1}^{n} \ell(\mathbf{x}_j, y_j, h)}_{\substack{\text{Empirical Risk} \\ \text{(data term)}}} + \underbrace{\frac{\lambda}{n} \, \mathrm{complexity}(h)}_{\substack{\text{Structure Penalty} \\ \text{(prior beliefs)}}} \right]$$

Structural risk minimization

$h^{SRM}(\alpha, \lambda) = \arg\min$

Bayesian Generative Learning as Structural Risk Minimization

$$\widehat{\theta}_{\mathrm{MAP}} = \arg\min_{\theta} \left[\frac{1}{n} \sum_{j=1}^{n} \underbrace{-\ln p(\mathbf{x}_j, y_j|\theta)}_{\text{like a loss function}} - \frac{1}{n} \ln \pi(\theta) \right]$$

$$h^{\mathrm{SRM}}(\alpha, \lambda) = \arg\min_{h \in \mathcal{H}} \left[\underbrace{\frac{1}{n} \sum_{j=1}^{n} \ell(\mathbf{x}_j, y_j, h)}_{\substack{\text{Empirical Risk} \\ \text{(data term)}}} + \underbrace{\frac{\lambda}{n} \mathrm{complexity}(h)}_{\substack{\text{Structure Penalty} \\ \text{(prior beliefs)}}} \right]$$

- Formal connection:

Bayesian Generative Learning	SRM
model family: $\Omega = \{p(\mathbf{x}, y|\theta)\}$	hypothesis set: $\mathcal{H} = \{h_\theta$: Bayes rule for $\ell_B(\quad)$ for $p(\mathbf{x}, y|\theta\}$
loss function: $\ell_B()$	loss function: $\ell(\mathbf{x}, y, h_\theta) = -\ln p(\mathbf{x}, y|\theta)$
prior: $\pi(\theta)$	complexity(h_θ) = $-\frac{1}{\lambda}\ln \pi(\theta)$

Frequentist and Bayesian Discriminative Learning

- Frequentist: parameters θ viewed as deterministic unknown and estimated from training data via conditional ML

$$\widehat{\theta}_{\mathrm{ML}} = \arg\max_{\theta} \left[\prod_{j=1}^{n} p(y_j|\mathbf{x}_j, \theta) \right]$$

$$= \arg\min_{\theta} \left[\frac{1}{n} \sum_{j=1}^{n} \underbrace{-\ln p(y_j|\mathbf{x}_j, \theta)}_{\text{like a loss function}} \right]$$

- analogous to ERM with a loss function defined by a discriminative probability model

Frequentist and Bayesian Discriminative Learning

- Bayesian: parameters θ viewed as random with given prior $\pi(\theta)$ and estimated from $\mathcal{D}$ via conditional MAP

$$\widehat{\theta}_{\mathrm{MAP}} = \arg\max_{\theta} \left[\left(\prod_{j=1}^{n} p(y_j|\mathbf{x}_j, \theta) \right) \pi(\theta) \right]$$

$$= \arg\min_{\theta} \left[\frac{1}{n} \sum_{j=1}^{n} \underbrace{-\ln p(y_j|\mathbf{x}_j, \theta)}_{\text{like a loss function}} - \frac{1}{n} \ln \pi(\theta) \right]$$

- analogous to SRM with a loss function defined by a discriminative probability model and a structure penalty (complexity regularization) term determined by the prior

Bayesian Discriminative Learning as Structural Risk Minimization

- Bayesian Discriminative Learning:

$$\widehat{\theta}_{\mathrm{MAP}} = \arg\min_{\theta} \left[\frac{1}{n} \sum_{j=1}^{n} \underbrace{-\ln p(y_j|\mathbf{x}_j, \theta)}_{\text{like a loss function}} - \frac{1}{n} \ln \pi(\theta) \right]$$

- Structural Risk Minimization:

$$h^{\mathrm{SRM}}(\alpha, \lambda) = \arg\min_{h \in \mathcal{H}} \left[\underbrace{\frac{1}{n} \sum_{j=1}^{n} \ell(\mathbf{x}_j, y_j, h)}_{\substack{\text{Empirical Risk} \\ \text{(data term)}}} + \underbrace{\frac{\lambda}{n} \text{ complexity}(h)}_{\substack{\text{Structure Penalty} \\ \text{(prior beliefs)}}} \right]$$

Bayesian Discriminative Learning as Structural Risk Minimization

$$\widehat{\theta}_{\mathrm{MAP}} = \arg\min_{\theta} \left[\frac{1}{n} \sum_{j=1}^{n} \underbrace{-\ln p(y_j|\mathbf{x}_j,\theta)}_{\text{like a loss function}} - \frac{1}{n} \ln \pi(\theta) \right]$$

$$h^{\mathrm{SRM}}(\alpha,\lambda) = \arg\min_{h\in\mathcal{H}} \left[\underbrace{\frac{1}{n} \sum_{j=1}^{n} \ell(\mathbf{x}_j, y_j, h)}_{\text{Empirical Risk (data term)}} + \underbrace{\frac{\lambda}{n} \mathrm{complexity}(h)}_{\text{Structure Penalty (prior beliefs)}} \right]$$

- Formal connection:

Bayesian Discriminative Learning	SRM
model family: $\Omega = \{p(y\|\mathbf{x},\theta)\}$	hypothesis set: $\mathcal{H} = \{h_\theta$: Bayes rule for $\ell_B(\quad)$ for $p(y\|\mathbf{x},\theta\}$
loss function: $\ell_B()$	loss function: $\ell(\mathbf{x},y,h_\theta) = -\ln p(y\|\mathbf{x},\theta)$
prior: $\pi(\theta)$	complexity(h_θ) = $-\frac{1}{\lambda}\ln \pi(\theta)$

More on the connection between SRM and Bayesian Learning

- To every pair (model, loss function), we can associate a decision rule, namely the Bayes decision rule for that model and loss.

- Question: Is the reverse statement true? Meaning, is every decision rule $h(\mathbf{x})$ the Bayes decision rule for some model and loss?

- Answer: Yes: If $\mathcal{Y}$ is $\mathbb{R}^m$, then $h(\mathbf{x})$ is the MMSE rule for any model in which $p(y|\mathbf{x})$ has mean $h(\mathbf{x})$, e.g., $p(y|\mathbf{x}) = \mathcal{N}(h(\mathbf{x}), I_m)(y)$. If $\mathcal{Y}$ is discrete, then $h(\mathbf{x})$ is the MPE rule for any model in which $p(y|\mathbf{x})$ is maximum when $\mathrm{y} = h(\mathbf{x})$, e.g., $p(y|\mathbf{x}) = 1(y = h(\mathbf{x}))$.

More on the connection between SRM and Bayesian Learning

- We saw that Bayesian learning can be viewed as SRM:

Bayesian (e.g., Discriminative) Learning	SRM
model family: $\Omega = \{p(y\|\mathbf{x},\theta)\}$	hypothesis set: $\mathcal{H} = \{h_\theta\text{: Bayes rule for}\ell_B(\quad)\text{for } p(y\|\mathbf{x},\theta\}$
loss function: $\ell_B()$	loss function: $\ell(\mathbf{x},y,h_\theta) = -\ln p(y\|\mathbf{x},\theta)$
prior: $\pi(\theta)$	complexity(h_θ) = $-\frac{1}{\lambda}\ln \pi(\theta)$

- Question: Is the reverse true? i.e., given $\mathcal{H}$, ℓ, and complexity(h_θ) in SRM, can we define $\theta_h, p(y|\mathbf{x},\theta_h), \pi(\theta_h)$, and $l_B()$ so that the cost function in Bayesian learning matches that in SRM and h is the Bayes rule for $p(y|\mathbf{x},\theta_h)$ and $l_B()$?
- Answer: Not without suitable conditions

More on the connection between SRM and Bayesian Learning

- Conditions:
  - $\theta_h := h$
  - Normalization condition for model:

$$p(y|\mathbf{x},\theta_h) \propto e^{-\ell(\mathbf{x},y,h)} \Rightarrow \int_{\mathcal{Y}} e^{-\ell(\mathbf{x},y,h)}\, dy < \infty$$

  - Normalization condition for prior:

$$\pi(\theta_h) \propto e^{-\lambda \text{complexity}(h)} \Rightarrow \int_{\mathcal{H}} e^{-\lambda \text{complexity}(h)}\, dh < \infty$$

    and the integral has to be well defined
  - Conditions for the existence of a loss function $l_B()$ such that for each h, the Bayes rule for $p(y|\mathbf{x},\theta_h)$ and $l_B()$ is h. Example, for each for each $\mathbf{x}, h$, if $l(\mathbf{x},y,h)$ is a "smooth function" of y which has a unique minimum at $h(\mathbf{x})$, then we can define $l_B(\mathbf{x},y,h) = 1 - \delta(y - h(\mathbf{x}))$, where $\delta()$ is the unit impulse (Dirac delta) function.
- Conclusion: SRM is more general than Bayesian learning

Learning guarantees

- For any algorithm which learns a decision rule from data, e.g., $h^{\mathrm{SRM}}(\alpha,\lambda)$:
  - Asymptotic consistency: as $n \to \infty$ will the learned rule converge to the best-in-class decision rule maybe even the Bayes decision rule? Does the risk of the rule converge to the best-in-class risk or even the Bayes risk?
  - Sample complexity: how many training samples n does one need in order to guarantee that the estimated risk with the learned rule will be within some ϵ neighborhood of the best-in-class risk with confidence at least $1-\delta$ →Probably Approximately Correct (PAC)
  - Computational complexity: training time, testing time, memory requirements for training and testing
  - Complexity (capacity) of hypothesis set: VC, doubling, Natarajan … dimensions, Rademacher complexity,…

Universal Asymptotic Consistency Theorem

- For any joint distribution $p(\mathbf{x},y)$ on features $\mathbf{x}$ and labels y, there exists a decision rule whose risk converges to the Bayes risk as the training set size n goes to infinity.

Rate of convergence can be arbitrarily slow

- Here's one version of this result [Luc Devroye'82]:

  – For any binary classification rule $h(\mathbf{x})$ and any training set size n, there exists a joint distribution $p(\mathbf{x},y)$ on features $\mathbf{x}$ and labels y such that

$$P(h_{\text{Bayes}}(\mathbf{X}) \neq Y) = 0$$

  and yet

$$P(h(\mathbf{X}) \neq Y) > 1/2 - \epsilon$$

  i.e., only slightly better than random guessing, for any $\epsilon > 0$.

Rate of convergence can be arbitrarily slow

- More generally [Luc Devroye'82]:

  – For any sequence of decision rules $h_n(\mathbf{x})$ and any sequence of positive numbers a_n (not more than $1/16$) which converge to zero, there exists a joint distribution $p(\mathbf{x},y)$ on features $\mathbf{x}$ and labels y such that

$$P(h_{\text{Bayes}}(\mathbf{X}) \neq Y) = 0$$

and

$$P(h_n(\mathbf{X}) \neq Y) \geq a_n$$

No Free Lunch Theorem [Wolpert-Macready]

- All models are wrong, but some are useful – George Box
- There is no universally best model
- Assumptions that work well in one domain may work poorly in another
- ➔ need to develop many different types of models to cover wide variety of real-world data
- For each model there may be many different algorithms to train the model, each with its own speed-accuracy-complexity tradeoffs
- combination of data, models, algorithms➔ rest of this course