Readings listed in class website

Gaussians
Linear Regression
Bias-Variance Tradeoff

Machine Learning – 10701/15781
Carlos Guestrin
Carnegie Mellon University

January 22nd, 2007

Maximum Likelihood Estimation

- **Data:** Observed set *D* of α_H Heads and α_T Tails
- **Hypothesis:** Binomial distribution
- Learning θ is an optimization problem
  - What's the objective function?
- MLE: Choose θ that maximizes the probability of observed data:

$$\hat{\theta} = \arg\max_{\theta} P(\mathcal{D} \mid \theta) = \frac{\alpha_H}{\alpha_H + \alpha_T}$$

$$= \arg\max_{\theta} \ln P(\mathcal{D} \mid \theta)$$

Bayesian Learning for Thumbtack

$$P(\theta \mid \mathcal{D}) \;\propto\; P(\mathcal{D} \mid \theta)P(\theta)$$

- Likelihood function is simply Binomial:

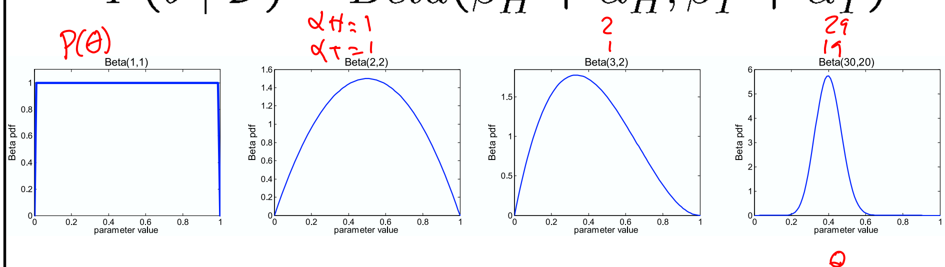
$$P(\mathcal{D} \mid \theta) = \theta^{\alpha_H}(1-\theta)^{\alpha_T}$$

- What about prior?
  - Represent expert knowledge
  - Simple posterior form
- Conjugate priors:
  - Closed-form representation of posterior
  - **For Binomial, conjugate prior is Beta distribution**

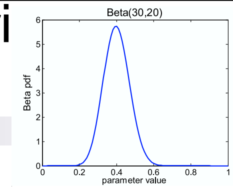
Posterior distribution

- Prior: $Beta(\beta_H, \beta_T)$
- Data: α_H heads and α_T tails
- Posterior distribution:

$$P(\theta \mid \mathcal{D}) \sim Beta(\beta_H + \alpha_H, \beta_T + \alpha_T)$$



MAP: Maximum a posteriori approximation



$$P(\theta \mid \mathcal{D}) \sim Beta(\beta_H + \alpha_H, \beta_T + \alpha_T)$$

$$E[f(\theta)] = \int_0^1 f(\theta) P(\theta \mid \mathcal{D}) d\theta$$

- As more data is observed, Beta is more certain

- MAP: use most likely parameter:

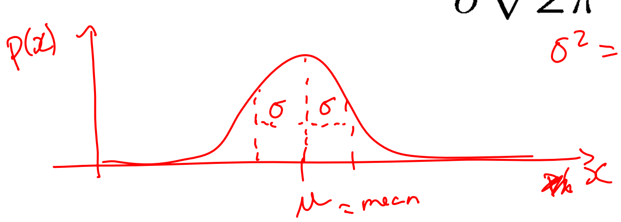
$$\underbrace{\widehat{\theta} = \arg\max_\theta P(\theta \mid \mathcal{D})}_{\text{most likeli parameter}} \qquad E[f(\theta)] \approx f(\widehat{\theta})$$

What about continuous variables?

- Billionaire says: If I am measuring a continuous variable, what can you do for me?
- **You say: Let me tell you about Gaussians…**

$$P(x \mid \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

σ^2 = Variance



Some properties of Gaussians

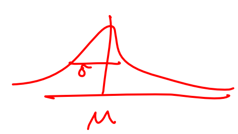
- affine transformation (multiplying by scalar and adding a constant)
  - X ~ $N(\mu,\sigma^2)$
  - Y = aX + b → Y ~ $N(\underbrace{a\mu+b}_{\text{mean}},\underbrace{a^2\sigma^2}_{\text{variance}})$
- Sum of Gaussians
  - X ~ $N(\mu_X,\sigma^2_X)$
  - Y ~ $N(\mu_Y,\sigma^2_Y)$
  - Z = X+Y → Z ~ $N(\underbrace{\mu_X+\mu_Y}_{\text{mean sum}}, \underbrace{\sigma^2_X+\sigma^2_Y}_{\text{var sum}})$

Learning a Gaussian

- Collect a bunch of data — N points
  - Hopefully, i.i.d. samples
  - e.g., exam scores

99
98
97
98
⋮
72
98
⋮

- Learn parameters
  - Mean $= \sum_i \frac{x_i}{N}$
  - Variance = ...



$$P(x \mid \mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

MLE for Gaussian

- Prob. of i.i.d. samples $D=\{x_1,\ldots,x_N\}$: $= \prod_i p(x_i|\mu,\sigma)$

$$P(\mathcal{D} \mid \mu,\sigma) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N \prod_{i=1}^{N} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}}$$

- Log-likelihood of data:

$$\begin{aligned} \ln P(\mathcal{D} \mid \mu,\sigma) &= \ln\left[\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N \prod_{i=1}^{N} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}}\right] \\ &= \underline{-N\ln\sigma\sqrt{2\pi}} - \underline{\sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2\sigma^2}} \end{aligned}$$

Your second learning algorithm: MLE for mean of a Gaussian

$\frac{d}{d\mu}[f+g] = \frac{d}{d\mu}f + \frac{d}{d\mu}g$

- What's MLE for mean?

$\frac{d}{d\mu}\frac{(x_i-\mu)^2}{2\sigma^2} = -\frac{2(x_i-\mu)}{2\sigma^2}$

$$\frac{d}{d\mu}\ln P(\mathcal{D} \mid \mu,\sigma) = \frac{d}{d\mu}\left[\underbrace{-N\ln\sigma\sqrt{2\pi}}_{\to 0} - \sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2\sigma^2}\right] = 0$$

$$= -\sum_{i=1}^{N} -\frac{2(x_i-\mu)}{2\sigma^2} = 0 \implies N\mu = \sum_{i=1}^{N} x_i$$

$$\implies \boxed{\mu = \frac{\sum_{i=1}^{N} x_i}{N}}$$

MLE for variance

$\frac{d}{d\sigma} - N \ln \sigma\sqrt{2\pi} = -N \log\sigma - N \log\sqrt{2\pi}$

$\frac{d}{d\sigma} \log\sigma = \frac{1}{\sigma}$

- Again, set derivative to zero:

$$
\begin{aligned}
\frac{d}{d\sigma} \ln P(\mathcal{D} \mid \mu, \sigma) &= \frac{d}{d\sigma}\left[-N \ln \sigma\sqrt{2\pi} - \sum_{i=1}^{N} \frac{(x_i-\mu)^2}{2\sigma^2}\right] \\
&= \frac{d}{d\sigma}\left[-N \ln \sigma\sqrt{2\pi}\right] - \sum_{i=1}^{N} \frac{d}{d\sigma}\left[\frac{(x_i-\mu)^2}{2\sigma^2}\right] = 0
\end{aligned}
$$

$$-\frac{N}{\sigma} \quad - \sum_{i=1}^{N} - \frac{(x_i-\mu)^2}{\sigma^3} \Rightarrow \sigma^2 = \frac{\sum_{i=1}^{N} (x_i-\mu)^2}{N}$$

Learning Gaussian parameters

- MLE:

$$\widehat{\mu}_{MLE} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\widehat{\sigma}^2_{MLE} = \frac{1}{N}\sum_{i=1}^{N} (x_i - \widehat{\mu})^2$$

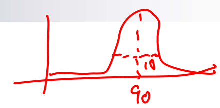
- BTW. MLE for the variance of a Gaussian is **biased**
  - Expected result of estimation is **not** true parameter!
  - Unbiased variance estimator:

$$\widehat{\sigma}^2_{unbiased} = \frac{1}{N-1}\sum_{i=1}^{N} (x_i - \widehat{\mu})^2$$

Bayesian learning of Gaussian parameters

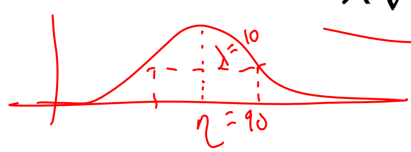
- Conjugate priors
  - Mean: Gaussian prior
  - Variance: Wishart Distribution

$P(\mu) =$



- Prior for mean:

$$P(\mu \mid \eta, \lambda) = \frac{1}{\lambda\sqrt{2\pi}} e^{\frac{-(\mu-\eta)^2}{2\lambda^2}}$$



MAP for mean of Gaussian

$P(\mu \mid D, \sigma) \propto P(\mu \mid \eta, \lambda) \cdot P(D \mid \mu, \sigma)$

prior

$$P(\mu \mid \eta, \lambda) = \frac{1}{\lambda\sqrt{2\pi}} e^{\frac{-(\mu-\eta)^2}{2\lambda^2}}$$

likelihood

$$P(\mathcal{D} \mid \mu, \sigma) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N \prod_{i=1}^{N} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}}$$

$$\boxed{\frac{d}{d\mu}\left[\ln P(\mathcal{D} \mid \mu) P(\mu)\right] = \frac{d}{d\mu}\left[\ln P(\mathcal{D} \mid \mu) + \ln P(\mu)\right]} = 0$$

compute MAP mode

$$-\frac{(\mu-\eta)}{\lambda^2} + \sum_{i=1}^{N} \frac{(x_i - \mu)}{\sigma^2} = 0$$

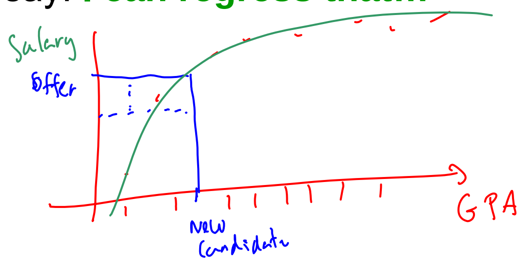
$$\Rightarrow \frac{N\mu}{\sigma^2} + \frac{\mu}{\lambda^2} = \left[\sum_{i=1}^{N} \frac{x_i}{\sigma^2}\right] + \frac{\eta}{\lambda^2}$$

$$\Rightarrow \quad \mu = \left[\left(\sum_{i=1}^{N} \frac{x_i}{\sigma^2}\right) + \frac{\eta}{\lambda^2}\right] \Big/ \left[\frac{N}{\sigma^2} + \frac{1}{\lambda^2}\right]$$

if I know nothing $\lambda^2 \to \infty$
$\Rightarrow$ estimate is same as MLE,
but $\lambda^2 < \infty$
then bring answer closer to η

Prediction of continuous variables

- Billionaire says: Wait, that’s not what I meant!
- You says: Chill out, dude.
- He says: I want to predict a continuous variable for continuous inputs: I want to predict salaries from GPA.
- You say: **I can regress that…**



The linear regression problem

- **Instances:** <$\mathbf{x}_j$, t_j> <(GPA, 10701 Grade, …, Salary)>
  x_j t_j
  < 200, 97, 150k>
- **Learn:** Mapping from x to t($\mathbf{x}$)
- **Hypothesis space:**
  - Given, basis functions $H = \{h_1, \ldots, h_K\}$
    $1, x, x^2, x^3 \ldots$ $-x_7$ e^{x_1}, $\sin x_2 x_1$
  - Find coeffs $\mathbf{w}=\{w_1,\ldots,w_k\}$ $\underbrace{t(\mathbf{x})}_{\text{data}} \approx \widehat{f}(\mathbf{x}) = \sum_i w_i h_i(\mathbf{x})$ $h_{81}(\mathbf{x})$
    coefficients
    linear combination
  - Why is this called linear regression?
    - model is linear in the parameters

    h_i not linear

    what does $\approx$ mean??

- Precisely, minimize the residual squared error:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum_{j=1}^{N} \Big(\overbrace{t(\mathbf{x}_j) - \sum_i w_i h_i(\mathbf{x}_j)}^{\text{residual}} \Big)^2 \quad \text{squared}$$

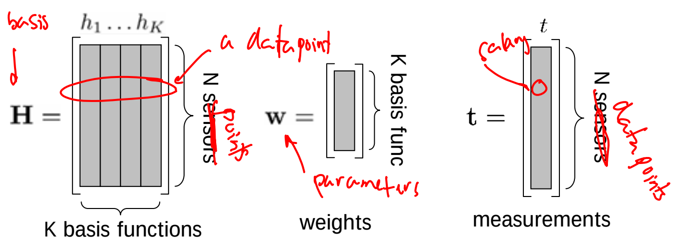
The regression problem in matrix notation

$(a-b)^2 = (b-a)^2$

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum_{j=1}^{N} \left(t(\mathbf{x}_j) - \sum_i w_i h_i(\mathbf{x}_j) \right)^2$$

transpose

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \underbrace{(\mathbf{H}\mathbf{w} - \mathbf{t})^T (\mathbf{H}\mathbf{w} - \mathbf{t})}_{\text{residual error}}$$



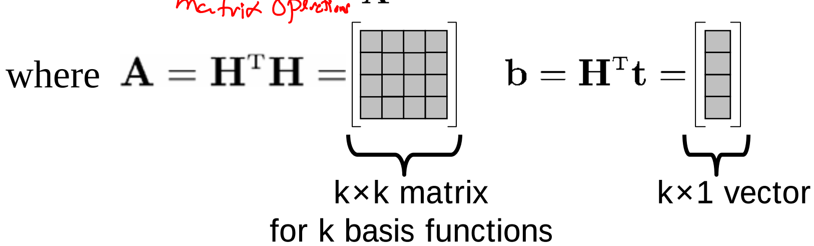
Regression solution = simple matrix operations

take derivative, set to zero

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \underbrace{(\mathbf{H}\mathbf{w} - \mathbf{t})^T (\mathbf{H}\mathbf{w} - \mathbf{t})}_{\text{residual error}}$$

$$\text{solution: } \mathbf{w}^* = \underbrace{(\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}}_{\mathbf{A}^{-1}} \underbrace{\mathbf{H}^{\mathrm{T}}\mathbf{t}}_{\mathbf{b}} = \mathbf{A}^{-1}\mathbf{b}$$

← simple matrix operations



But, why?

- Billionaire (again) says: Why sum squared error???
- You say: Gaussians, Dr. Gateson, Gaussians…
- Model: prediction is linear function plus Gaussian noise
  - $t = \sum_i w_i h_i(\mathbf{x}) + \varepsilon$

noise $\varepsilon \sim N(0, \sigma^2)$ mean variance

$t(x) \sim N(\sum_i w_i h_i(x), \sigma^2)$

- Learn **w** using MLE

$$P(t \mid \mathbf{x}, \mathbf{w}, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-[t - \sum_i w_i h_i(\mathbf{x})]^2}{2\sigma^2}}$$

Maximizing log-likelihood

$\arg\max_w \frac{f(w)}{a} = \arg\max_w f(w)$

Maximize:

$$\ln P(\mathcal{D} \mid \mathbf{w}, \sigma) = \ln \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N \prod_{j=1}^{N} e^{\frac{-[t_j - \sum_i w_i h_i(\mathbf{x}_j)]^2}{2\sigma^2}}$$

$$= \ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N + \ln \prod_{j=1}^{N} e^{-\frac{[t_j - \sum_i w_i h_i(x_j)]^2}{2\sigma^2}}$$

$$= \ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N - \underbrace{\sum_{j=1}^{N} \frac{[t_j - \sum_{i=1}^{k} w_i h_i(x_j)]^2}{2\sigma^2}}_{\text{maximize}_w}$$

residual error

constant no role in finding w

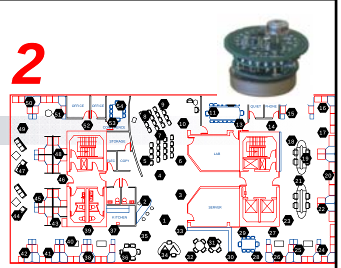
$\equiv \text{minimizing}_w \sum_{j=1}^{N} [t_j - \sum_{i=1}^{k} w_i h_i(x_j)]^2$

Least-squares Linear Regression is MLE for Gaussians!!!

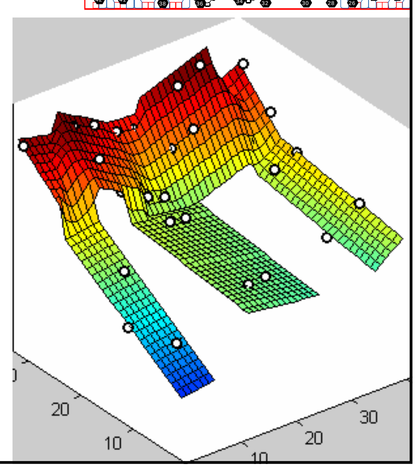
Applications Corner 1

- Predict stock value over time from
  - past values
  - other relevant vars
    - e.g., weather, demands, etc.

Applications Corner 2



- Measure temperatures at some locations
- Predict temperatures throughout the environment



[Guestrin et al. '04]

Applications Corner 3

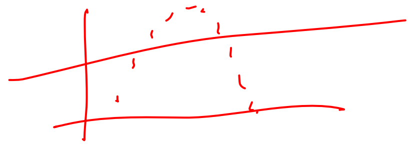
- Predict when a sensor will fail
  - based several variables
    - age, chemical exposure, number of hours used,…

Announcements

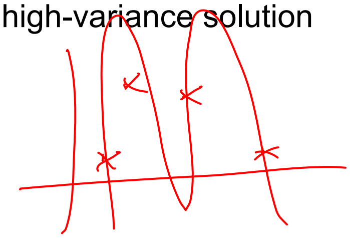
- Readings associated with each class
  - See course website for specific sections, extra links, and further details
  - Visit the website frequently
- Recitations
  - Thursdays, 5:30-6:50 in Wean Hall 5409
- Special recitation on Matlab
  - Jan. 24 Wed. 5:30-6:50pm NSH 1305

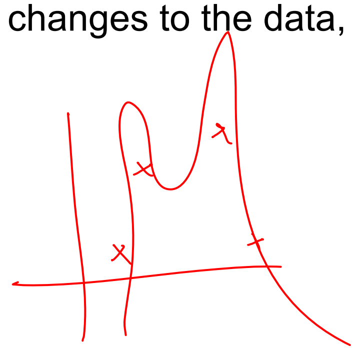
Bias-Variance tradeoff – Intuition

- Model too "simple" $\rightarrow$ does not fit the data well
  - A biased solution



- Model too complex $\rightarrow$ small changes to the data, solution changes a lot
  - A high-variance solution





(Squared) Bias of learner

- Given dataset *D* with *m* samples, learn function h(x)
- If you sample a different datasets, you will learn different h(x)
- **Expected hypothesis**: $E_D[h(x)]$ = average h over all possible D
- **Bias:** difference between what you expect to learn and truth
  - Measures how well you expect to represent true solution
  - Decreases with more complex model

expect to learn truth

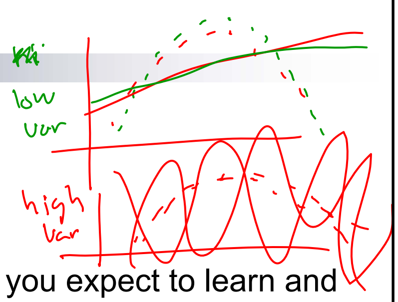
$$bias^2 = \int_x (E_D[h(x)] - t(x))^2 p(x) dx$$

(Squared) Bias of learner

- Given dataset *D* with *m* samples, learn function h(x)
- If you sample a different datasets, you will learn different h(x)
- **Expected hypothesis**: $E_D[h(x)]$
- **Bias:** difference between what you expect to learn and truth
  - Measures how well you expect to represent true solution
  - Decreases with more complex model

$$bias^2 = \int_x \{E_D[h(x)] - t(x)\}^2 p(x)dx$$

Variance of learner



- Given a dataset *D* with *m* samples, you learn function h(x)
- If you sample a different datasets, you will learn different h(x)
- **Variance:** difference between what you expect to learn and what you learn from a from a particular dataset
  - Measures how sensitive learner is to specific dataset
  - Decreases with simpler model

what you learn on average

$$\bar{h}(x) = E_D[h(x)]$$

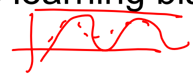
$$variance = \int E_D[(h(x) - \bar{h}(x))^2]p(x)dx$$

what you learn in this dataset

Bias-Variance Tradeoff

- Choice of hypothesis class introduces learning bias
  - More complex class → less bias
  - More complex class → more variance

poly degree 27



Bias–Variance decomposition of error

- Consider simple regression problem f:X→T

$t = f(x) = g(x) + \varepsilon$

truth

noise ~ N(0,σ)

deterministic

~ f(x)

Collect some data, and learn a function h(x)

What are sources of prediction error?

Sources of error 1 – noise

$f(x) = g(x) + \varepsilon$

- What if we have perfect learner, infinite data?
  - If our learning solution h(x) satisfies h(x)=g(x)
  - Still have remaining, *unavoidable error* of σ^2 due to noise ε

$$error(h) = \int_x \int_t (\underset{= g(x)}{h(x)} - \underset{f(x) = g(x) + \varepsilon}{t})^2 p(f(x) = t|x)p(x)dtdx$$

$\varepsilon \sim N(0, \sigma^2)$

$$\int_x \int_t (-\varepsilon)^2 \, p(\ldots) \, dx = \sigma^2$$

noise variance

Sources of error 2 – Finite data

- What if we have imperfect learner, or only m training examples?

  $\not\exists\, h$ that fit g exactly
- What is our expected squared error per example?
  - Expectation taken over random training sets *D* of size m, drawn from distribution P(X,T)

$$E_D \left[\int_x \int_t \underbrace{\{h(x) - t\}^2}_{\text{residual squared error}} \underbrace{p(f(x) = t|x)}_{\text{distribution salaries given GPA}} \underbrace{p(x)}_{\text{prob. obs. GPA}} dtdx \right]$$

Bias-Variance Decomposition of Error

Bishop Chapter 3

Assume target function: t = f(x) = g(x) + ε

Then expected sq error over fixed size training sets *D* drawn from P(X,T) can be expressed as sum of three components:

$$E_D\left[\int_x \int_t (h(x)-t)^2 p(t|x)p(x)dtdx\right]$$

$$= \underbrace{unavoidableError}_{\sigma^2} + bias^2 + variance$$

bias

variance

Where:

$$unavoidableError = \sigma^2$$

$$bias^2 = \int (E_D[h(x)] - g(x))^2 p(x)dx$$

$$\bar{h}(x) = E_D[h(x)]$$

$$variance = \int E_D[(h(x) - \bar{h}(x))^2]p(x)dx$$

What you need to know

- Gaussian estimation
  - MLE
  - Bayesian learning
  - MAP
- Regression
  - Basis function = features
  - Optimizing sum squared error
  - Relationship between regression and Gaussians
- Bias-Variance trade-off
- Play with Applet