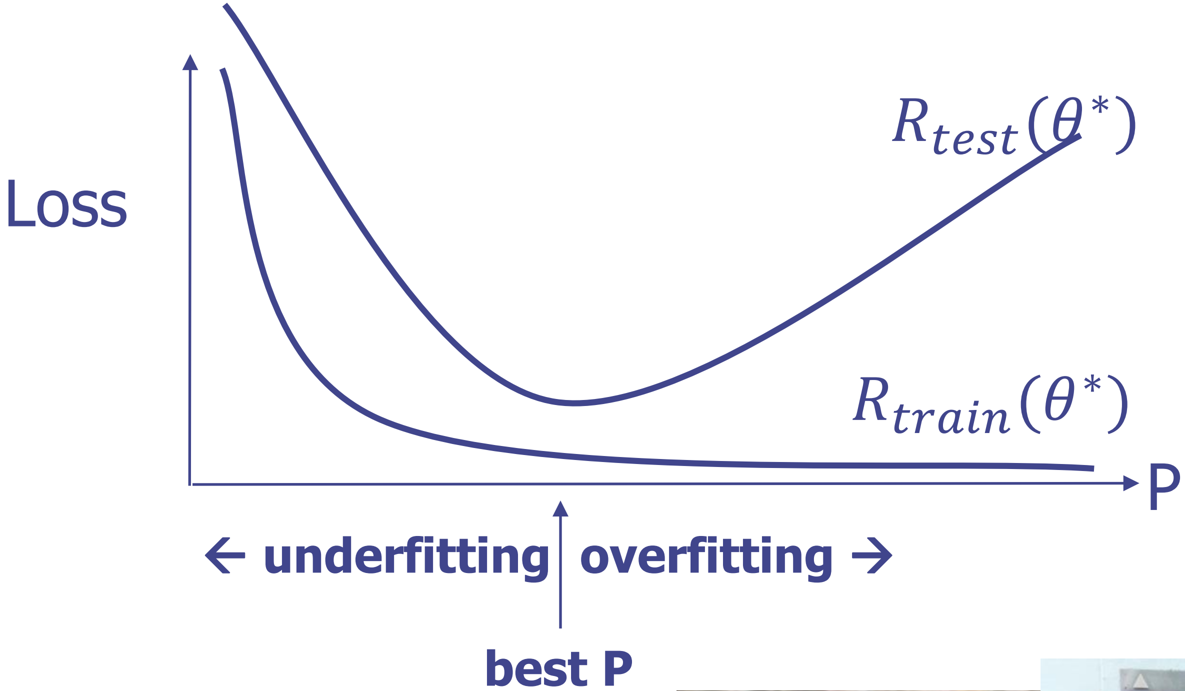
Machine Learning

4771

Instructor: Itsik Pe’er

Reminder: Cross Validation



General Additive Models

Class 5: How to stop Max Likelihood from Overfitting ?

- Estimating parameters of distributions
- Evidence vs. prior assumptions
- Regularizing regression

Example: Mean of Gaussian

- Can we recover most likely μ for height?

$x \sim Normal(\mu, \sigma^2)$

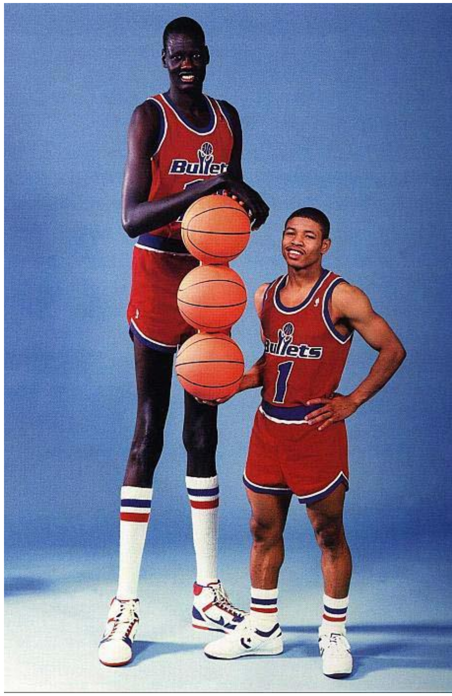
$$p(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

Example: Mean of Gaussian

- Can we recover most likely μ for height?

$$p(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

$$\log p(x_1,\dots,x_N|\mu,\sigma^2) =$$
$$= -\frac{N}{2}\log 2\pi\sigma^2 - \frac{\sum_{i=1}^{N}(x_i-\mu)^2}{2\sigma^2}$$



Example: Mean of Gaussian

- Can we recover most likely μ for height?

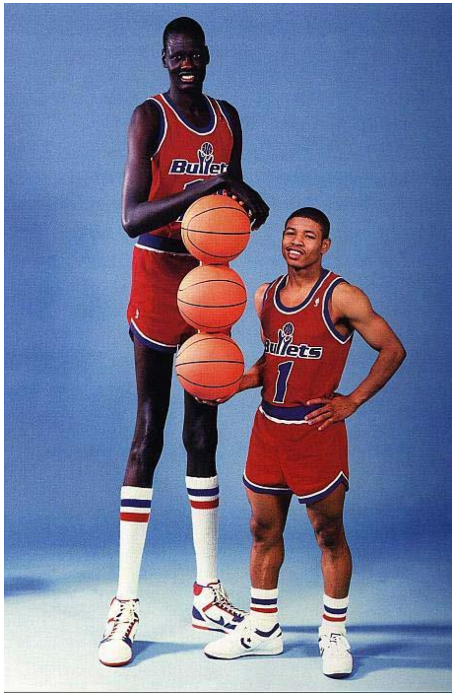
$$p(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

$$\log p(x_1,\ldots,x_N|\mu,\sigma^2) =$$

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

$$\frac{d}{d\mu}\log p(\boldsymbol{X}|\mu^*,\sigma^2) = \frac{\sum_{i=1}^{N}(x_i-\mu^*)}{\sigma^2} = 0$$

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



Example: Success rate

- Can we recover ML α for drawing a card?

$$x \sim Bernoulli(\alpha)$$

$$p(x|\alpha) = \alpha^x (1-\alpha)^{1-x}$$

Example: Success rate

- Can we recover ML α for drawing a card?

$$x \sim Bernoulli(\alpha)$$

$$p(x|\alpha) = \alpha^x (1-\alpha)^{1-x}$$

$$N_1 = \sum_i x_i$$

$$\log p(x_1, \ldots, x_N | \alpha) = N_1 \log \alpha - (N - N_1) \log(1 - \alpha)$$

Example: Success rate

- Can we recover ML α for drawing a card?

$$x \sim Bernoulli(\alpha)$$

$$p(x|\alpha) = \alpha^x (1-\alpha)^{1-x}$$

$$N_1 = \sum_i x_i$$

$$\log p(x_1, \ldots, x_N | \alpha) = N_1 \log \alpha - (N - N_1) \log(1-\alpha)$$

$$\frac{d}{d\alpha} \log p(\boldsymbol{X}|\alpha^*) = \frac{N_1}{\alpha^*} - \frac{N-N_1}{1-\alpha^*} = 0$$

$$\alpha^* = \frac{N_1}{N}$$

Best Guess

- Given evidence X, what's best guess α?

Best Guess

- Prior assumption about $\alpha : p(\alpha)$
- What's best guess α?
- Given evidence $\boldsymbol{X}$, what's best guess α?

Bayesian Inference

- Prior assumption about $\alpha : p(\alpha)$
  $$E[\alpha]$$
- Given evidence $\boldsymbol{X}$, what's best guess α?
- Bayesian answer: optimize $E[\alpha|\boldsymbol{X}]$ w.r.t. posterior $p(\alpha|\boldsymbol{X}) = \dfrac{p(\alpha)p(\boldsymbol{X}|\alpha)}{p(\boldsymbol{X})}$
- Optimal if we have true probability

Bayesian Inference

- Prior assumption about $\alpha : p(\alpha)$

$$E[\alpha]$$

- Given evidence $\boldsymbol{X}$, what's best guess α?

- Bayesian answer: optimize $E[\alpha|\boldsymbol{X}]$ w.r.t. posterior

$$p(\alpha|\boldsymbol{X}) = \frac{p(\alpha)p(\boldsymbol{X}|\alpha)}{p(\boldsymbol{X})}$$

prior

likelihood

Constant w.r.t. α

Bayesian Inference

- Prior assumption about $\alpha : p(\alpha)$
- Given evidence $\boldsymbol{X}$, what is the Expected A-Posteriori (EAP) $E_\alpha\left[\frac{p(\alpha)p(\boldsymbol{X}|\alpha)}{p(\boldsymbol{X})}\right]$

Bayesian Inference

- Prior assumption about $\alpha : p(\alpha)$
- Given evidence $\boldsymbol{X}$, what is the Expected A-Posteriori (EAP) $E_\alpha\left[\frac{p(\alpha)p(\boldsymbol{X}|\alpha)}{p(\boldsymbol{X})}\right]$
- Another approach:
  Maximum A-Posteriori (MAP)
  $argmax_\alpha[p(\alpha)p(\boldsymbol{X}|\alpha)] =$
  $= argmax_\alpha[\log p(\alpha) + \log p(\boldsymbol{X}|\alpha)]$

Bayesian Inference

- Prior assumption about $\alpha : p(\alpha)$
- Given evidence X, what is the Expected A-Posteriori (EAP) $E_\alpha\left[\frac{p(\alpha)p(\boldsymbol{X}|\alpha)}{p(\boldsymbol{X})}\right]$

Bayesian Inference

- Prior assumption about $\alpha : p(\alpha)$

  $$\alpha \sim Uniform(0,1) \;\; ; \;\; x \sim Bernoulli(\alpha\,)$$

- Given evidence $\boldsymbol{X}$, what is the Expected A-Posteriori (EAP) $E_\alpha\left[\frac{p(\alpha)p(\boldsymbol{X}|\alpha)}{p(\boldsymbol{X})}\right] =$

$$= \frac{1}{p(\boldsymbol{X})}\int_{\alpha=0}^{1} \alpha\, p(\alpha)p(\boldsymbol{X}|\alpha)d\alpha =$$

$$= \frac{\int_{\alpha=0}^{1} \alpha \cdot 1 \cdot \alpha^{N_1}(1-\alpha)^{N-N_1}d\alpha}{\int_{\alpha=0}^{1} \alpha^{N_1}(1-\alpha)^{N-N_1}d\alpha} = \frac{c(N_1+1, N-N_1)}{c(N_1, N-N_1)}$$

$$c(m,k) = \int_{\alpha=0}^{1} \alpha^m (1-\alpha)^k d\alpha$$

$$c(m,k) = \int_{\alpha=0}^{1} \alpha^m (1-\alpha)^k d\alpha$$

$$k = 0 : c(m,k) = \int_{\alpha=0}^{1} \alpha^m d\alpha = \frac{1}{m+1}$$

$k, m > 0 :$

$$0 = \alpha^m (1-\alpha)^k \Big|_0^1 = mc(m-1,k) - kc(m,k-1)$$

$$c(m,k) = \frac{k}{m+1} c(m+1,k-1) = \cdots =$$

$$= \frac{k!}{\frac{(m+k)!}{m!}} c(m+k,0) = \frac{m!\,k!}{(m+k)!} \int_{\alpha=0}^{1} \alpha^{m+k} d\alpha$$

$$= \frac{m!\,k!}{(m+k+1)!}$$

Bayesian Inference

- Prior assumption about $\alpha : p(\alpha)$
  $\alpha \sim Uniform(0,1)$; $x \sim Bernoulli(\alpha)$
- Given evidence $\boldsymbol{X}$, what is the Expected A-Posteriori (EAP) $E_{\alpha}\left[\frac{p(\alpha)p(\boldsymbol{X}|\alpha)}{p(\boldsymbol{X})}\right] =$

$$= \frac{c(N_1+1, N-N_1)}{c(N_1, N-N_1)}$$

Substitute $c(m,k) = \frac{m!k!}{(m+k+1)!}$

Bayesian Inference

- Prior assumption about $\alpha : p(\alpha)$

  $$\alpha \sim Uniform(0,1)\ ;\ x \sim Bernoulli(\alpha)$$

- Given evidence $\boldsymbol{X}$, what is the

  Expected A-Posteriori (EAP) $E_\alpha\left[\frac{p(\alpha)p(\boldsymbol{X}|\alpha)}{p(\boldsymbol{X})}\right] =$

  $$= \frac{c(N_1+1, N-N_1)}{c(N_1, N-N_1)} = \frac{\frac{(N_1+1)!(N-N_1)!}{(N+2)!}}{\frac{N_1!(N-N_1)!}{(N+1)!}} = \frac{N_1+1}{N+2}$$

- Additive smoothing, add-1 smoothing

  Chance for sunrise tomorrow[Laplace]

Bayesian approach to overfit prevention

- Prior assumption about $\alpha : p(\alpha)$
- Given evidence $\boldsymbol{X}$, what is the Maximum A-Posteriori (MAP) $argmax_{\alpha}[p(\alpha)p(\boldsymbol{X}|\alpha)] =$
  $= argmax_{\alpha}[\log p(\alpha) + \log p(\boldsymbol{X}|\alpha)]$

Regression: Assuming θ is small

- Prior: $\Pr(\theta) \propto e^{-\frac{\lambda}{2}\|\theta\|^2}$

Assuming θ is small

- Prior: $\Pr(\theta) \propto e^{-\frac{\lambda}{2}\|\theta\|^2}$
- $\Pr(Data) = \Pr(Data|\theta) \times \Pr(\theta)$

- Posterior = Likelihood × Prior

Assuming θ is small

- Prior: $\Pr(\theta) \propto e^{-\frac{\lambda}{2}\|\theta\|^2}$
- $\Pr(Data) = \Pr(Data|\theta) \times \Pr(\theta)$

  $\log \Pr(Data) = \mathrm{l}(\theta) + \log \Pr(\theta)$
- Posterior = Likelihood × Prior

$\theta^* =$Max-aposteriori$=\mathrm{argmax}[\mathrm{l}(\theta) + \log \Pr(\theta)]$

Regularized Risk Minimization

- Empirical Risk Minimization gave overfitting & underfitting
- We want to add a penalty for using too many theta values

Regularized Risk Minimization

- Empirical Risk Minimization gave overfitting & underfitting
- We want to add a penalty for using too many theta values
- This gives us the Regularized Risk

$$R_{regularized}(\theta) = R_{empirical}(\theta) + Penalty(\theta)$$
$$= \frac{1}{N}\sum_{i=1}^{N} Loss\big(y_i, f(x_i;\theta)\big) + \frac{\lambda}{2}\|\theta\|^2$$

- Solution for Regularized Risk with Least Squares Loss:

Regularized Risk Minimization

- Empirical Risk Minimization gave overfitting & underfitting
- We want to add a penalty for using too many theta values
- This gives us the Regularized Risk

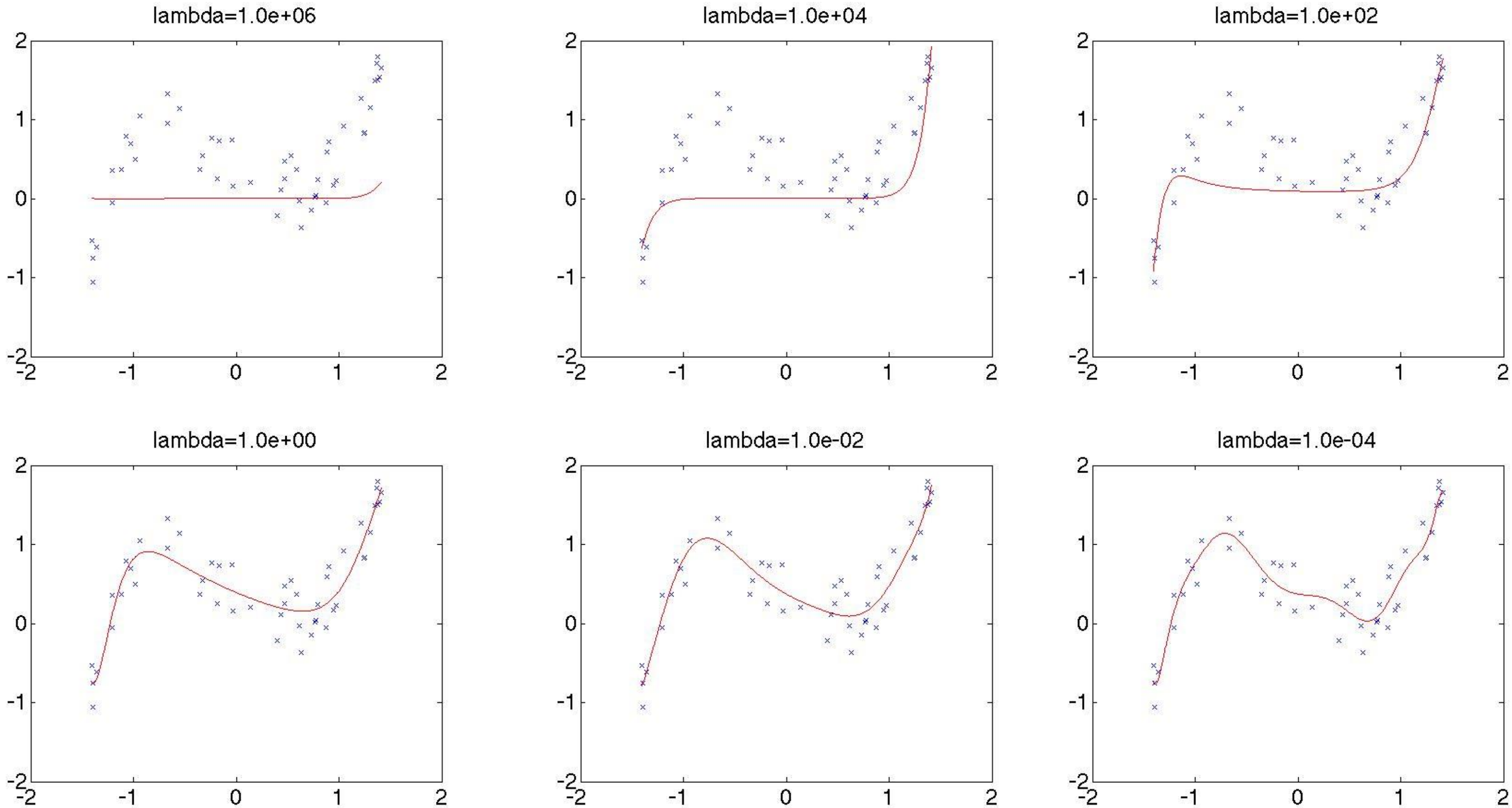
$$R_{regularized}(\theta) = R_{empirical}(\theta) + Penalty(\theta)$$
$$= \frac{1}{N}\sum_{i=1}^{N} Loss\big(y_i, f(x_i;\theta)\big) + \frac{\lambda}{2}\|\theta\|^2$$

- Solution for Regularized Risk with Least Squares Loss:

$$\nabla_\theta R_{regularized} = 0$$
$$\nabla_\theta\left(\frac{1}{2N}\|\boldsymbol{y} - \boldsymbol{X}\theta\|^2 + \frac{\lambda}{2}\|\theta\|^2\right) = 0$$
$$\frac{1}{2N}(-2\boldsymbol{X}^T\boldsymbol{y} + 2\boldsymbol{X}^T\boldsymbol{X}\theta) + \frac{\lambda}{2}(2\theta) = 0$$
$$\theta^* = (\boldsymbol{X}^T\boldsymbol{X} + \lambda N I)^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

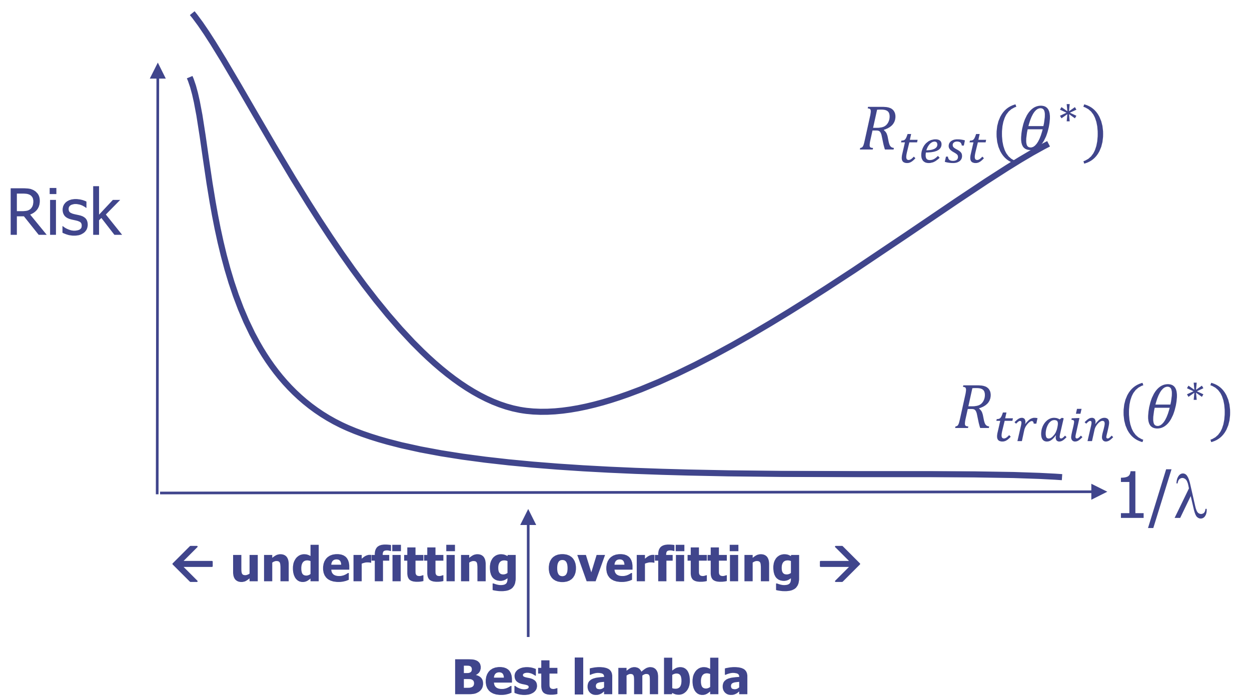
Regularized Risk Minimization

- Have D=16 features (or P=15 throughout)
- Try minimizing $R_{regularized}(\theta)$ to get θ* with different λ
- Note that λ=0 give back Empirical Risk Minimization



Crossvalidation

- Try fitting with different lambda regularization levels
- Select lambda which gives lowest $R_{test}(\theta^*)$



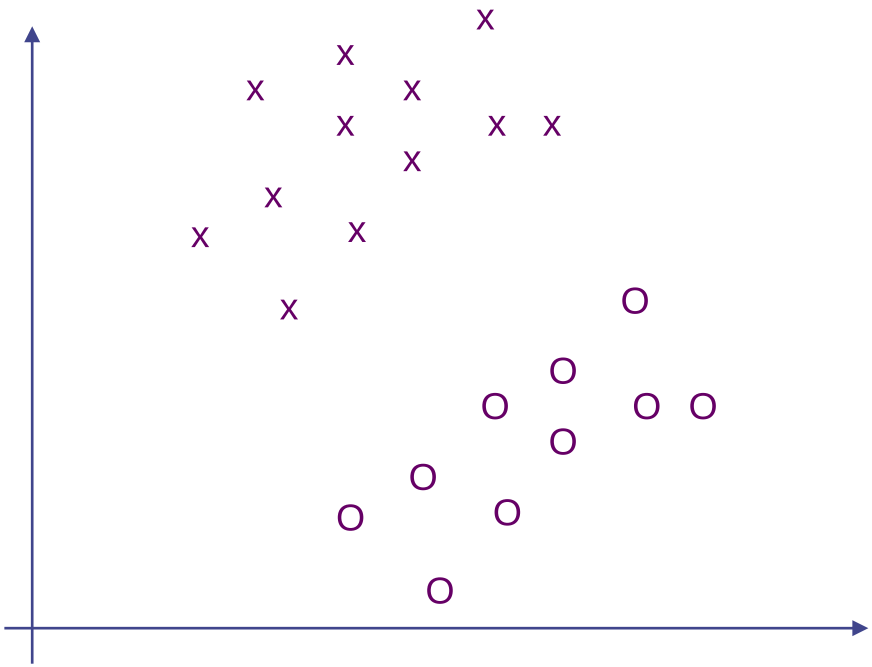
- Lambda measures simplicity of the model
- Models with low lambda are more flexible

Summary

- Inferring distribution parameters:
  - Max likelihood
  - Expected A-Posteriori
  - Maximum A-Posterior
- Regularization

Class 6

- **Classification**
- Logistic Regression
- Gradient Descent



Classification Problems

- Determine student admission to Columbia based on GPA, prev. school rank, tests

Classification Problems

- Determine student admission to Columbia based on GPA, prev. school rank, tests
- Decide malignant or benign tumors based on size, density, speed of growth

Kim et al. '07

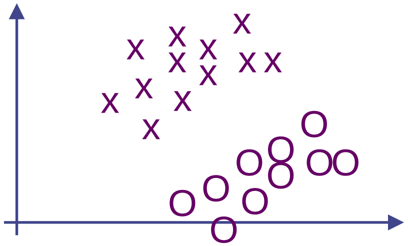
From Regression To Classification

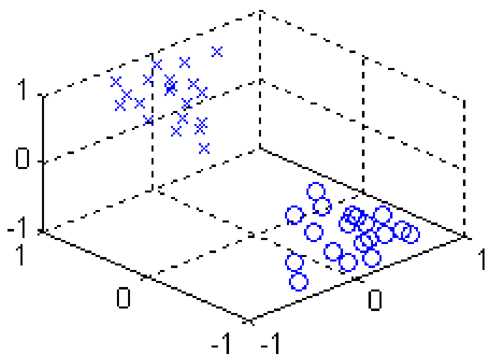
- Classification is another important learning problem

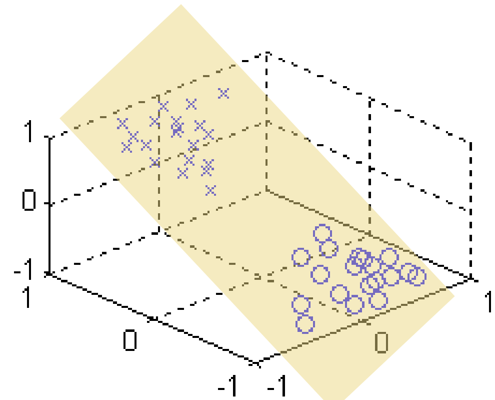
Classification: $X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \{0,1\}$

Regression: $X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \boldsymbol{R}$

- Should we solve this as a least squares regression problem?







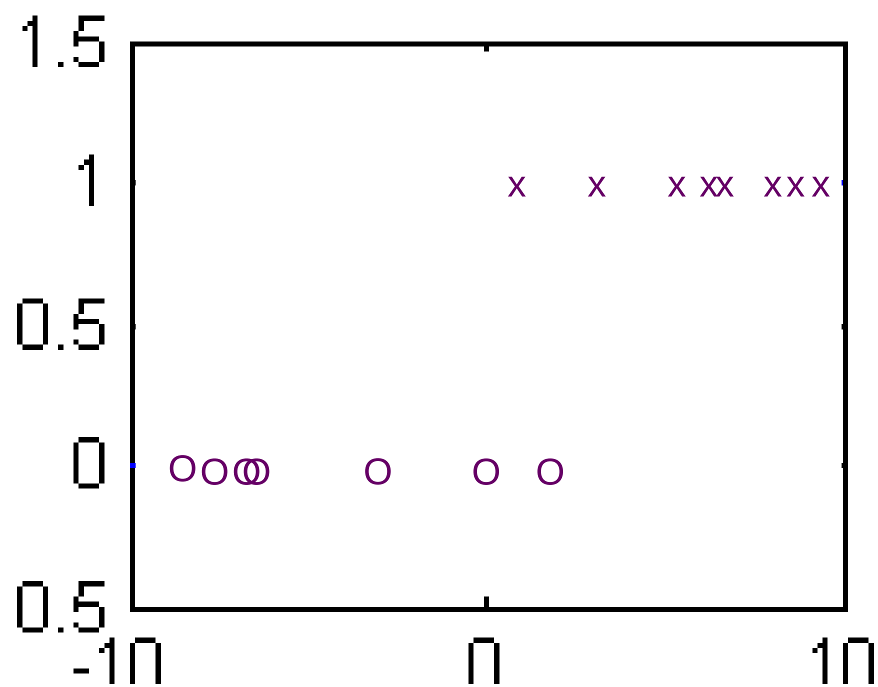
Logistic Regression

- Given a classification problem with binary outputs

$$X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \{0,1\}$$

- Use this function and output 1 if $f(\boldsymbol{x})>0.5$ and 0 otherwise

$$f(\boldsymbol{x};\theta) = \frac{1}{1 + exp(-\theta\boldsymbol{x})}$$



Short hand for Linear Functions

- What happened to adding the intercept?

$$f(\boldsymbol{x};\theta) = \theta^T \boldsymbol{x} + \theta_0$$

$$= \begin{bmatrix} \theta(1) \\ \theta(2) \\ \vdots \\ \theta(D) \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(1) \\ \boldsymbol{x}(2) \\ \vdots \\ \boldsymbol{x}(D) \end{bmatrix} + \theta_0 = \begin{bmatrix} \theta_0 \\ \theta(1) \\ \theta(2) \\ \vdots \\ \theta(D) \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{x}(1) \\ \boldsymbol{x}(2) \\ \vdots \\ \boldsymbol{x}(D) \end{bmatrix} = \vec{\theta}^T \vec{\boldsymbol{x}}$$

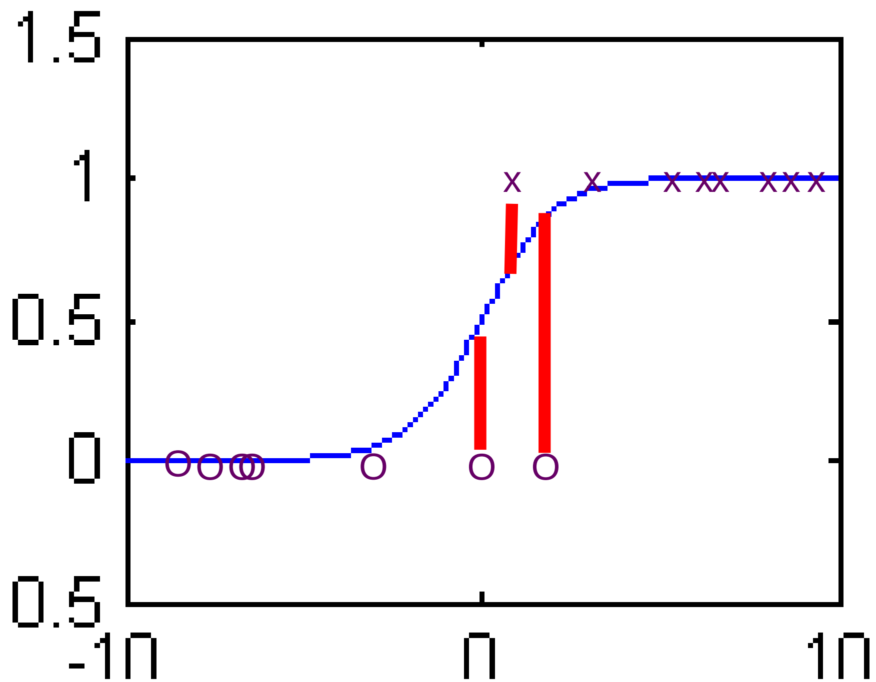
Logistic Regression

- Given a classification problem with binary outputs

$$X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \{0,1\}$$

- Use this function and output 1 if $f(\boldsymbol{x}) > 0.5$ and 0 otherwise

$$f(\boldsymbol{x}; \theta) = \frac{1}{1 + exp(-\theta \boldsymbol{x})}$$



Logistic Regression

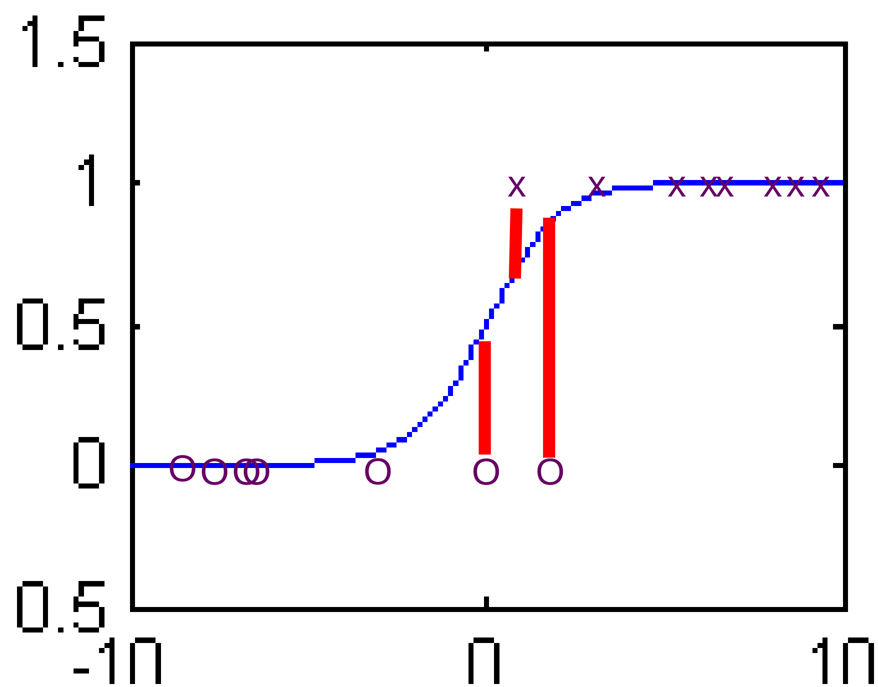
- Given a classification problem with binary outputs

$$X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \{0,1\}$$

- Use this function and output 1 if $f(\boldsymbol{x})>0.5$ and 0 otherwise

$$f(\boldsymbol{x}; \theta) = \frac{1}{1 + exp(-\theta \boldsymbol{x})}$$

- Assume $\Pr(y = 1) = f(\boldsymbol{x}; \theta)$



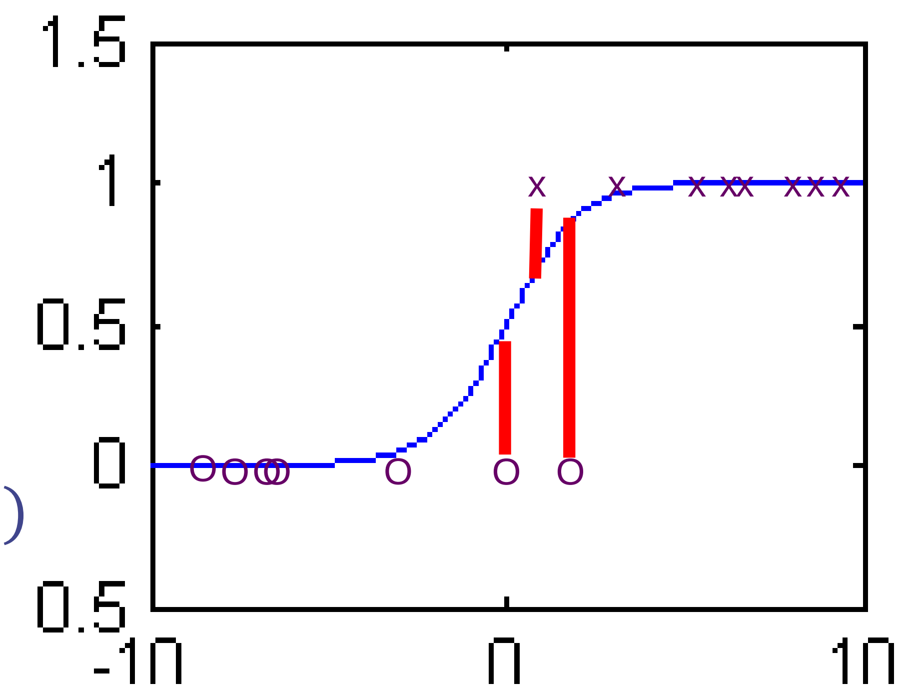
Logistic Regression

- Given a classification problem with binary outputs

$$X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \{0,1\}$$

- Use this function and output 1 if $f(\boldsymbol{x})>0.5$ and 0 otherwise

$$f(\boldsymbol{x};\theta) = \frac{1}{1+exp(-\theta\boldsymbol{x})}$$



- Assume $\Pr(y_i = 1) = f(\boldsymbol{x}_i;\theta)$

- $\Pr(y|f(\boldsymbol{x};\theta)) =$
$=\prod_{y_i=0}\left(1-f(\boldsymbol{x}_i;\theta)\right)\prod_{y_i=1} f(\boldsymbol{x}_i;\theta)$

Logistic Regression

- Given a classification problem with binary outputs

$$X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \{0,1\}$$

- Use this function and output 1 if $f(\boldsymbol{x})>0.5$ and 0 otherwise

$$f(\boldsymbol{x}; \theta) = \frac{1}{1 + exp(-\theta \boldsymbol{x})}$$

- Instead of squared loss, use Logistic Loss (i.e. negative binomial likelihood)

$$Loss_{log}\big(y, f(\boldsymbol{x}; \theta)\big) = (y - 1)\log\big(1 - f(\boldsymbol{x}; \theta)\big) - y\log(f(\boldsymbol{x}; \theta))$$

- The resulting method is called Logistic Regression.
- Empirical Risk:

Logistic Regression

- Given a classification problem with binary outputs

$$X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \{0,1\}$$

- Use this function and output 1 if $f(\boldsymbol{x})>0.5$ and 0 otherwise

$$f(\boldsymbol{x}; \theta) = \frac{1}{1 + exp(-\theta \boldsymbol{x})}$$

- Instead of squared loss, use Logistic Loss (i.e. negative binomial likelihood)

$$Loss_{log}\big(y, f(\boldsymbol{x}; \theta)\big) = (y-1)\log\big(1 - f(\boldsymbol{x}; \theta)\big) - y\log(f(\boldsymbol{x}; \theta))$$

- The resulting method is called Logistic Regression.
- Empirical Risk:

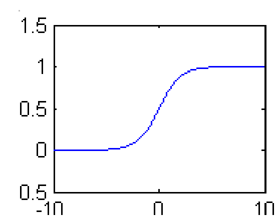
$$R_{emp}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - 1)\log\big(1 - f(\boldsymbol{x}_i; \theta)\big) - y_i\log(f(\boldsymbol{x}_i; \theta))$$

Logistic Regression

- With empirical logistic risk has no closed form solution:

$$R_{emp}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - 1)\log\big(1 - f(\boldsymbol{x_i};\theta)\big) - y_i\log(f(\boldsymbol{x_i};\theta))$$

$$f(\boldsymbol{x};\theta) = \frac{1}{1 + exp(-\theta\boldsymbol{x})}$$



Logistic Regression

- With empirical logistic risk has no closed form solution:

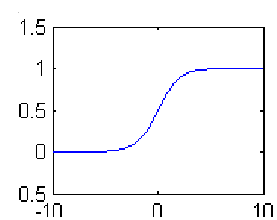
$$R_{emp}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - 1)\log\big(1 - f(\boldsymbol{x_i};\theta)\big) - y_i\log(f(\boldsymbol{x_i};\theta))$$

$$\nabla_\theta R = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1 - y_i}{1 - f(\boldsymbol{x_i};\theta)} - \frac{y_i}{f(\boldsymbol{x_i};\theta)}\right) f'(\boldsymbol{x_i};\theta) = 0 \quad ??????$$

Find best θ numerically!

where

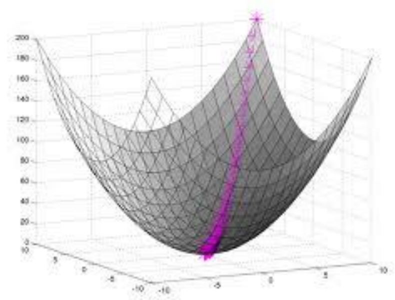
$$f(\boldsymbol{x};\theta) = \frac{1}{1 + exp(-\theta \boldsymbol{x})} = g(\theta^T \boldsymbol{x})$$



$$g(z) = \frac{1}{1 + exp(-z)} \qquad g'(z) = g(z)(1 - g(z))$$

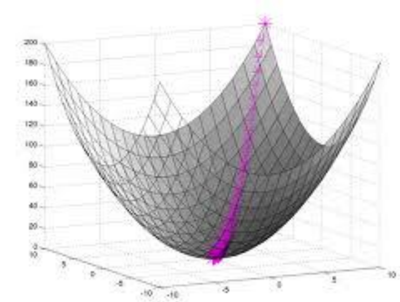
Gradient Descent

- Useful when we can't get minimum solution in closed form
- Gradient points in direction of fastest increase
- Take step in the opposite direction!



Gradient Descent

- Useful when we can't get minimum solution in closed form
- Gradient points in direction of fastest increase
- Take step in the opposite direction!



- Gradient Descent Algorithm

choose scalar step size η, *& tolerance* ε
initialize θ^0 *= small random vector*

$$\theta^1 \leftarrow \theta^0 - \eta \nabla_\theta R_{emp}|_{\theta^0} \;;t \leftarrow 1$$

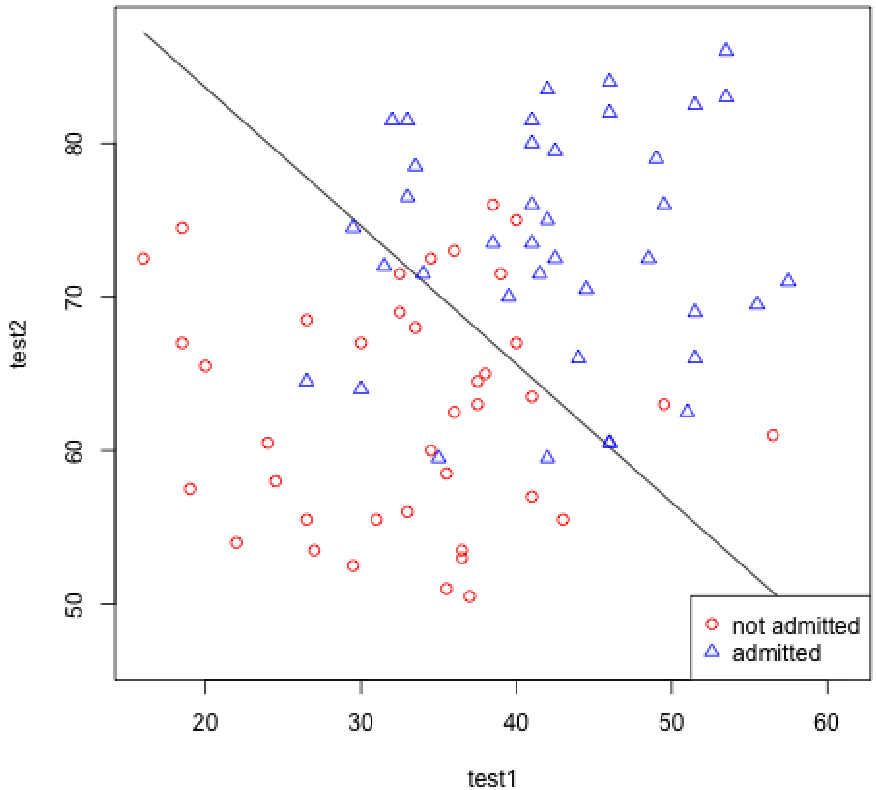
while $\|\theta^t - \theta^{t-1}\| \geq \epsilon$ *{*

$$\theta^{t+1} \leftarrow \theta^t - \eta \nabla_\theta R_{emp}|_{\theta^0} \;;t \leftarrow t+1 \quad \}$$

- For appropriate η, this will converge to local minimum

Logistic Regression

- Logistic regression gives better classification performance
- Its empirical risk is convex so gradient descent always converges to the same solution



Summary

- Additive models
- Classification
- Logistic Regression
- Gradient Descent