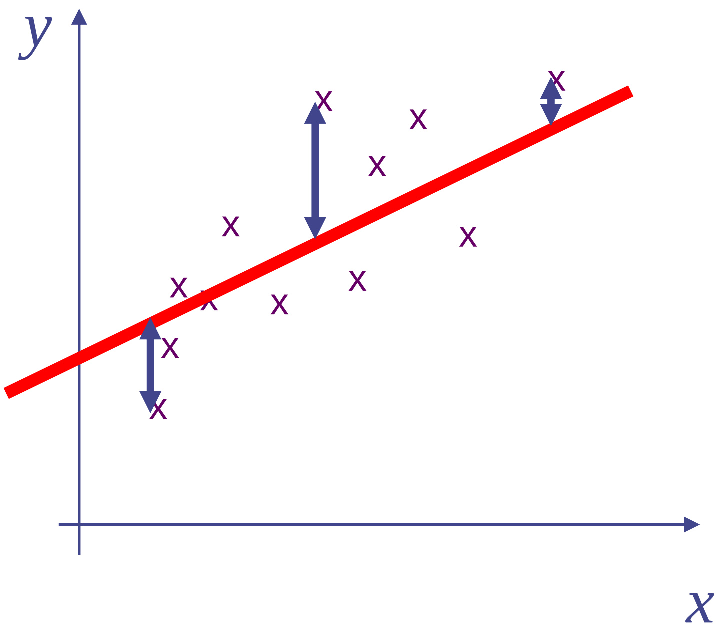
Machine Learning

4771

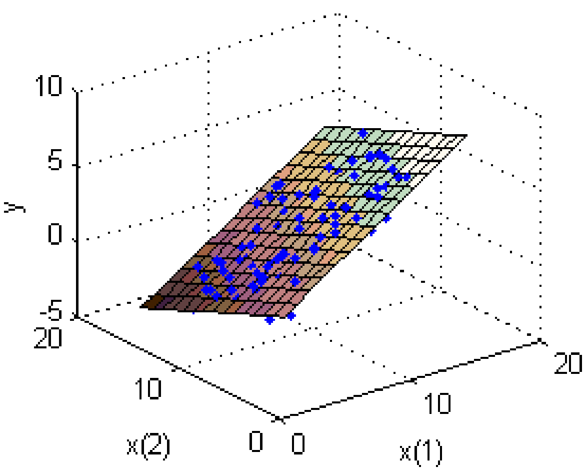
Instructor: Itsik Pe’er

Reminder: Regression

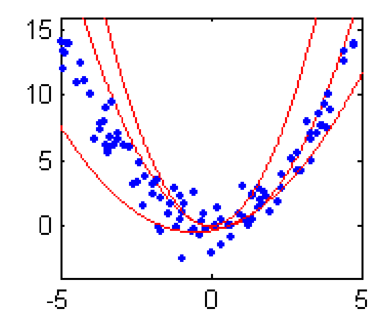


- 1D Linear:
  - Loss
  - Empirical risk
  - Least-squares

- Multi-D Linear: matrix form

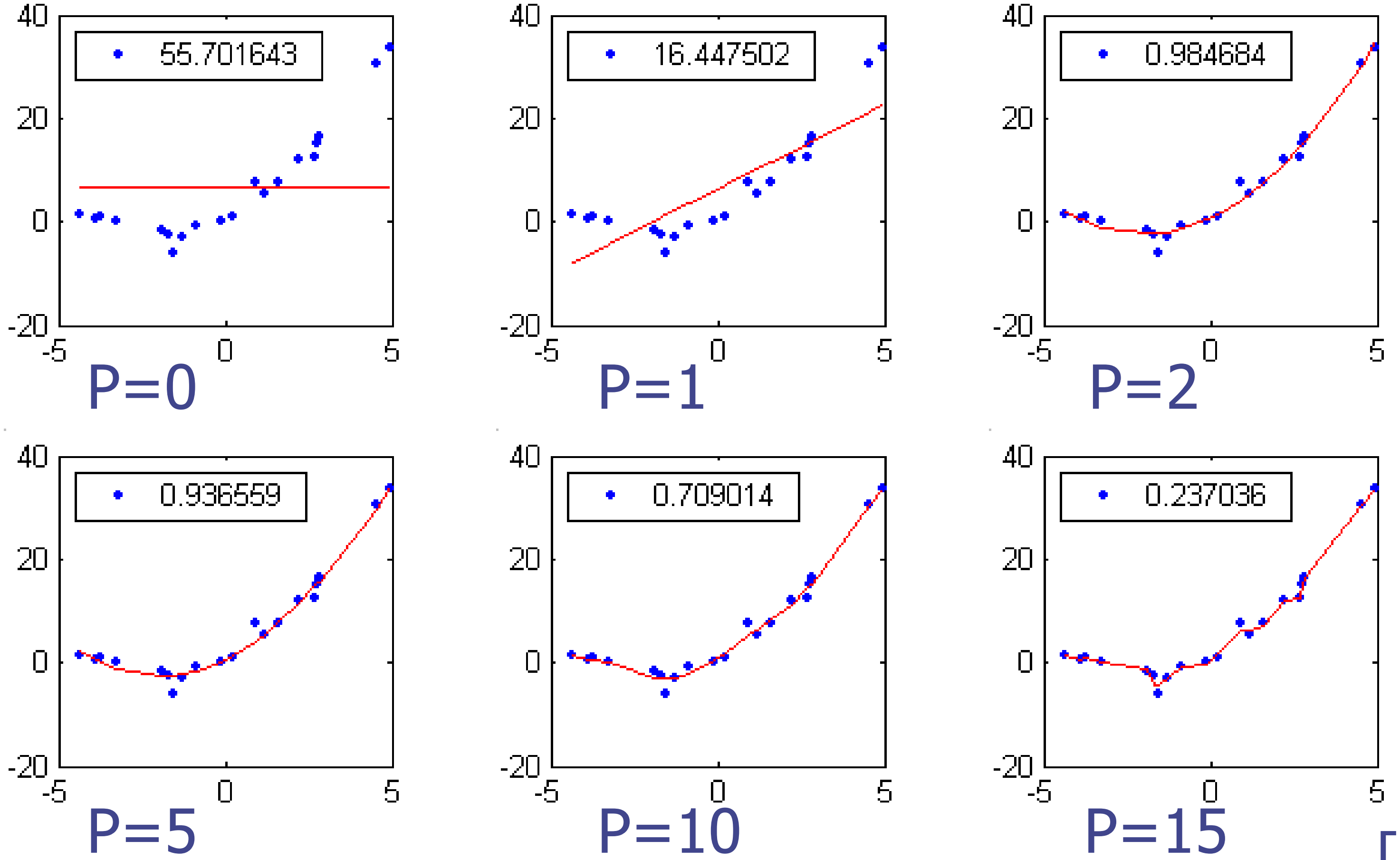


- Polynomial



Underfitting/Overfitting

- Try varying P. Higher P fits a more complex function class
- Observe $R(\theta^*)$ drops with bigger P



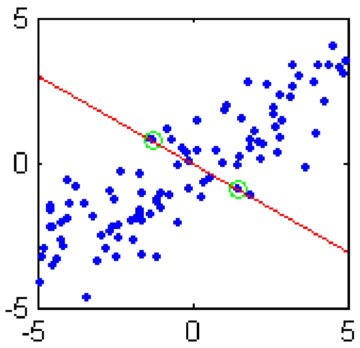
[Bishop]

Class 4

- Overfitting
- Additive models: Fourier
- Radial basis functions

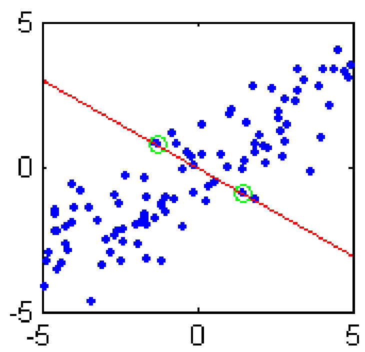
Evaluating The Regression

- Unfair to use empirical to find best order P
- High P (vs. N) can overfit, even linear case!
- min $R(\theta^*)$ not on training but on future data
- Want model to *Generalize* to future data



Evaluating The Regression

- Unfair to use empirical to find best order P
- High P (vs. N) can overfit, even linear case!
- min $R(\theta^*)$ not on training but on future data
- Want model to *Generalize* to future data



True loss: $R_{true}(\theta) = \int p(x,y)\frac{1}{2}(y-\theta^T x)^2 dx\, dy$

- One approach: split data into training / testing portion

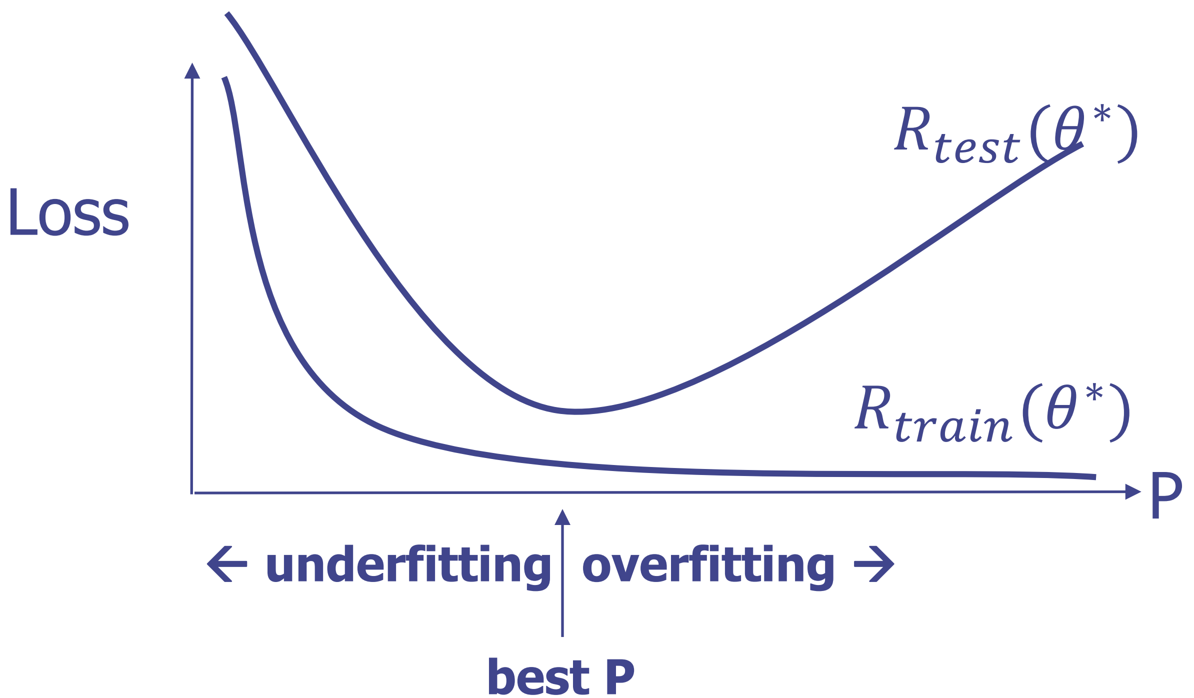
$\{(x_1,y_1),\ldots,(x_N,y_N)\}$ $\{(x_{N+1},y_{N+1}),\ldots,(x_{N+M},y_{N+M})\}$

- Estimate θ^* with training loss: $R_{train}(\theta) = \frac{1}{2N}\sum_{i=1}^{N}(y_i-\theta^T x_i)^2$
- Evaluate P with testing loss:

$$R_{test}(\theta^*) = \frac{1}{2M}\sum_{i=N+1}^{N+M}(y_i-\theta^{*T} x_i)^2$$

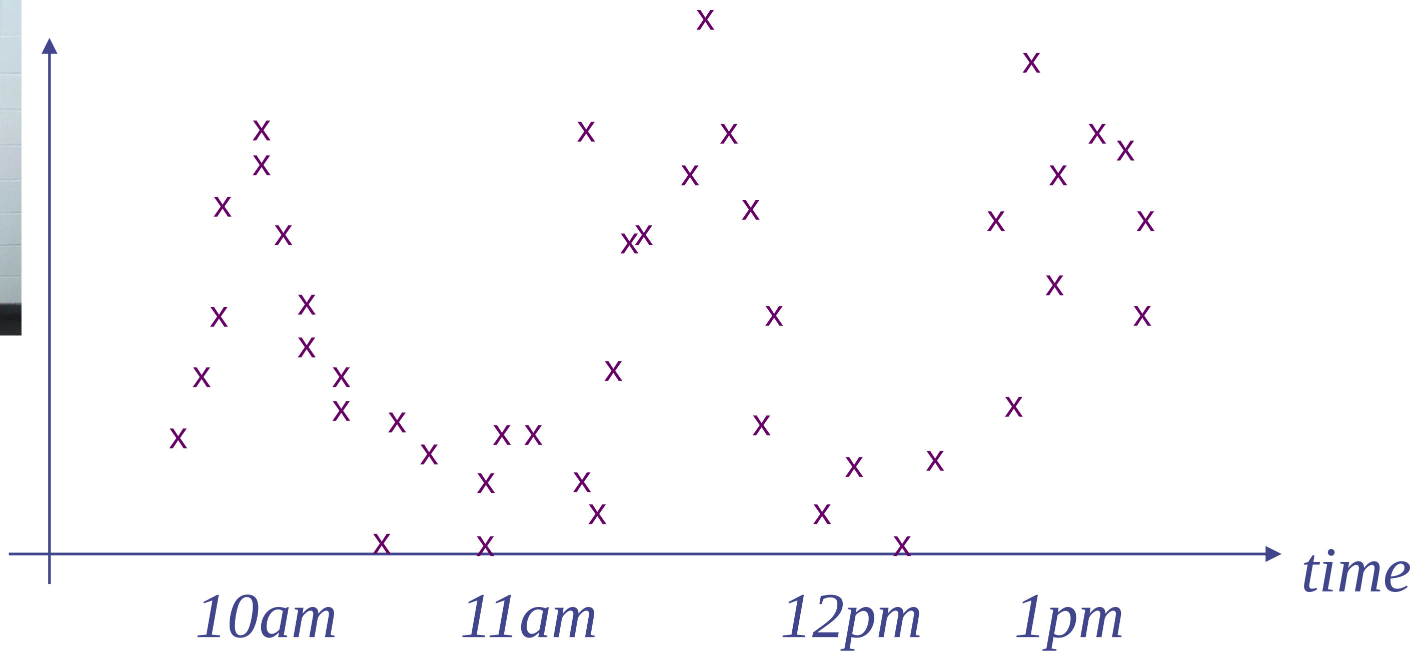
Crossvalidation

- Try fitting with different polynomial order P
- Select P which gives lowest $R_{test}(\theta^*)$

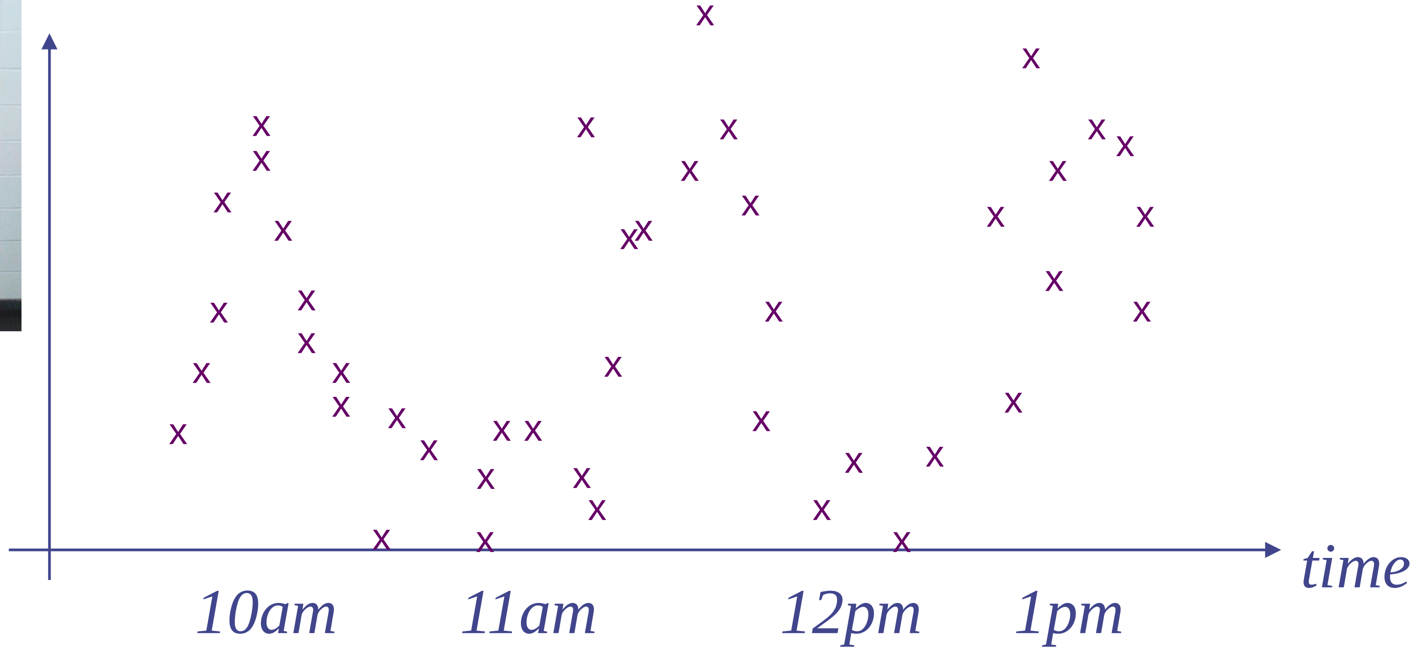


- Think of P as a measure of the complexity of the model
- Higher order polynomials are more flexible and complex

Example: Temporal data

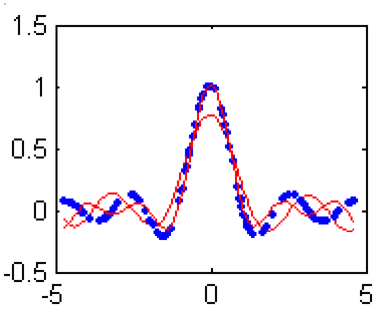


Example: Temporal data



- Need to fit periodic behavior
- Cycle: 90min, daily, weekly, annual

Sinusoidal Basis Functions

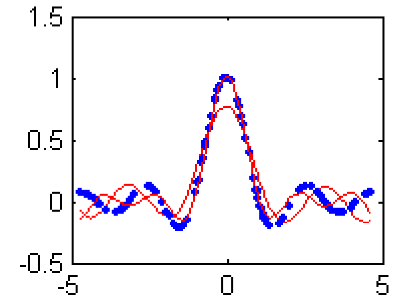


- General functions, not just polynomials:

$$f(x;\theta) = \sum_{p=1}^{P} \theta_p \phi_p(x) + \theta_0$$

- These are generally called Additive Models
- Regression adds linear combinations of the basis fn's

Sinusoidal Basis Functions



- General functions, not just polynomials:

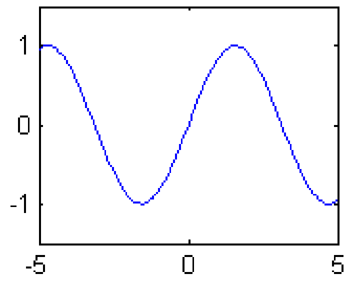
$$f(x;\theta) = \sum_{p=1}^{P} \theta_p \phi_p(x) + \theta_0$$

- These are generally called Additive Models
- Regression adds linear combinations of the basis fn's
- For example: Fourier (sinusoidal) basis

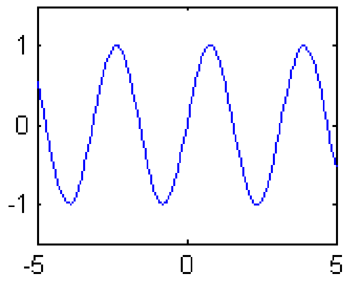
$$\phi_{2k}(x_i) = \sin(kx_i) \qquad \phi_{2k+1}(x_i) = \cos(kx_i)$$

- Note, don't have to be a basis per se, usually subset

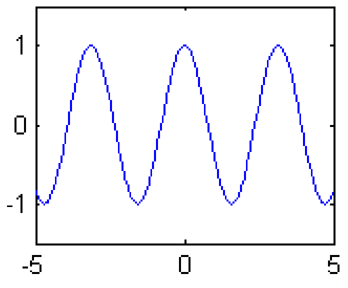
$\theta_1 \times$



$+\theta_2 \times$



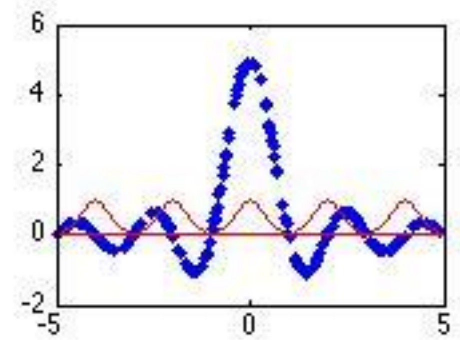
$+\theta_3 \times$



Patterson
Gimlin

Example: Bigfoot Sightings

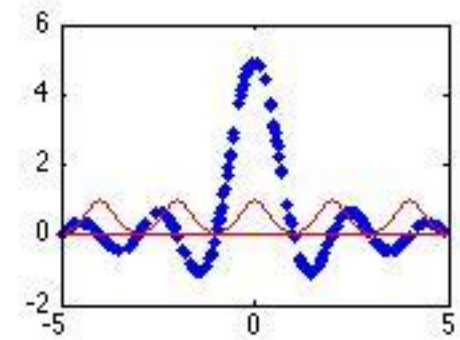
Radial Basis Functions (RBF)



- Can act as prototypes of the data itself

$$f(\boldsymbol{x};\theta) = \sum_{k=1}^{N} \theta_k \exp\left(-\frac{1}{\sigma^2}\|\boldsymbol{x} - \boldsymbol{x}_k\|^2\right)$$

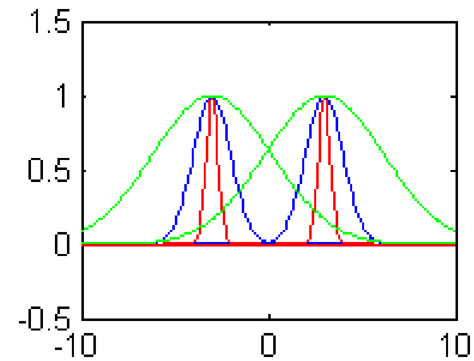
Radial Basis Functions (RBF)



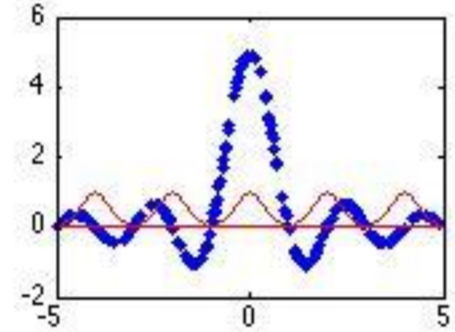
- Can act as prototypes of the data itself

$$f(\boldsymbol{x};\theta)=\sum_{k=1}^{N}\theta_k \exp\left(-\frac{1}{\sigma^2}\|\boldsymbol{x}-\boldsymbol{x}_k\|^2\right)$$

- Parameter σ = standard deviation controls how wide bumps are

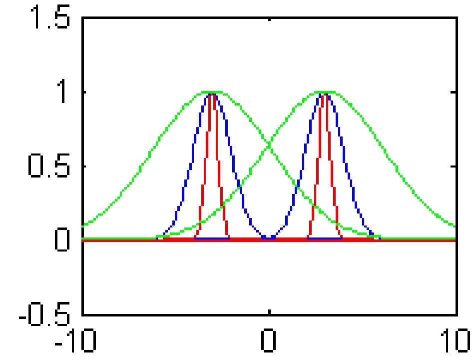


Radial Basis Functions (RBF)



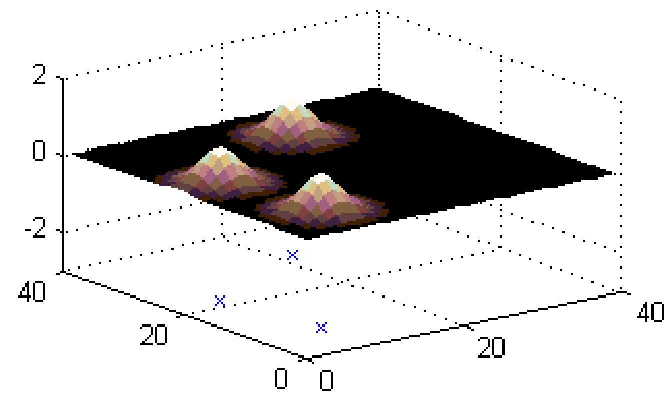
- Can act as prototypes of the data itself

$$f(\boldsymbol{x};\theta) = \sum_{k=1}^{N} \theta_k \exp\left(-\frac{1}{\sigma^2}\|\boldsymbol{x}-\boldsymbol{x}_k\|^2\right)$$



- Parameter σ = standard deviation controls how wide bumps are

- Also works in multi-dimensions



Radial Basis Functions

- Training point $\rightarrow$ bump function

$$f(\boldsymbol{x};\theta) = \sum_{k=1}^{N} \theta_k \underbrace{\exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}-\boldsymbol{x}_k\|^2)}_{\phi_k(x)}$$

- Reuse solution from linear regression: $\theta^* = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T y$
- Can view the data instead as $\boldsymbol{X}$, a big matrix of size N x N

Radial Basis Functions

- Training point $\rightarrow$ bump function

$$f(\boldsymbol{x};\theta) = \sum_{k=1}^{N} \theta_k \overbrace{\exp\left(-\frac{1}{\sigma^2}\|\boldsymbol{x}-\boldsymbol{x}_k\|^2\right)}^{\phi_k(x)}$$

- Reuse solution from linear regression: $\theta^* = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T y$
- Can view the data instead as $\boldsymbol{X}$, a big matrix of size N x N

$$\boldsymbol{X} = \begin{bmatrix} \phi_1(x_1) & \phi_2(x_1) & \cdots & \phi_k(x_1) \\ \phi_1(x_2) & \phi_2(x_2) & \cdots & \phi_k(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x_k) & \phi_2(x_k) & \cdots & \phi_k(x_k) \end{bmatrix}$$

Radial Basis Functions

- Training point $\rightarrow$ bump function

$$f(\boldsymbol{x};\theta)=\sum_{k=1}^{N}\theta_k \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}-\boldsymbol{x}_k\|^2)$$

- Reuse solution from linear regression: $\theta^*=(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T y$
- Can view the data instead as $\boldsymbol{X}$, a big matrix of size N x N

$$\boldsymbol{X}=\begin{bmatrix}\exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_1-\boldsymbol{x}_1\|^2) & \cdots & \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_1-\boldsymbol{x}_k\|^2)\\ \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_2-\boldsymbol{x}_1\|^2) & \cdots & \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_2-\boldsymbol{x}_k\|^2)\\ \vdots & \ddots & \vdots\\ \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_k-\boldsymbol{x}_1\|^2) & \cdots & \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_k-\boldsymbol{x}_k\|^2)\end{bmatrix}$$

Radial Basis Functions

- training point → bump function

$$f(\boldsymbol{x};\theta) = \sum_{k=1}^{N} \theta_k \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x} - \boldsymbol{x}_k\|^2)$$

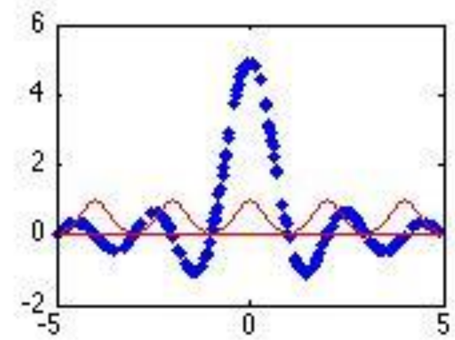
- Reuse solution from linear regression: $\theta^* = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T y$
- Can view the data instead as $\boldsymbol{X}$, a big matrix of size N x N

$$\boldsymbol{X} = \begin{bmatrix} \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_1 - \boldsymbol{x}_1\|^2) & \cdots & \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_1 - \boldsymbol{x}_k\|^2) \\ \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_2 - \boldsymbol{x}_1\|^2) & \cdots & \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_2 - \boldsymbol{x}_k\|^2) \\ \vdots & \ddots & \vdots \\ \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_k - \boldsymbol{x}_1\|^2) & \cdots & \exp(-\frac{1}{\sigma^2}\|\boldsymbol{x}_k - \boldsymbol{x}_k\|^2) \end{bmatrix}$$

- For RBFs, X is square and symmetric, so solution is just

$$\theta^* = \boldsymbol{X}^{-1} y$$

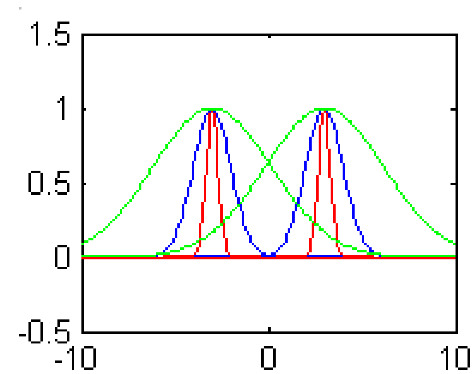
Bump Width for RBF



- Can act as prototypes of the data itself

$$f(\boldsymbol{x};\theta) = \sum_{k=1}^{N} \theta_k \exp(-\frac{1}{\sigma^2} \|\boldsymbol{x} - \boldsymbol{x}_k\|^2)$$

- Parameter σ = standard deviation controls how wide bumps are



What happens if too big/small?

How would we know that?

Evaluating Our Learned Function

- We minimized empirical risk to get θ^*
- How well does $f(x;\theta^*)$ perform on future data?
- It should *Generalize* and have low True Risk:

$$R_{true}(\theta) = \int P(x,y)\frac{1}{2}(y-\theta^T x)^2 dx\, dy$$

- Can't compute true risk, instead use Testing Empirical Risk
- We randomly split data into training and testing portions

$$\{(x_1,y_1),\dots,(x_N,y_N)\} \qquad \{(x_{N+1},y_{N+1}),\dots,(x_{N+M},y_{N+M})\}$$

- Find θ^* with training data:

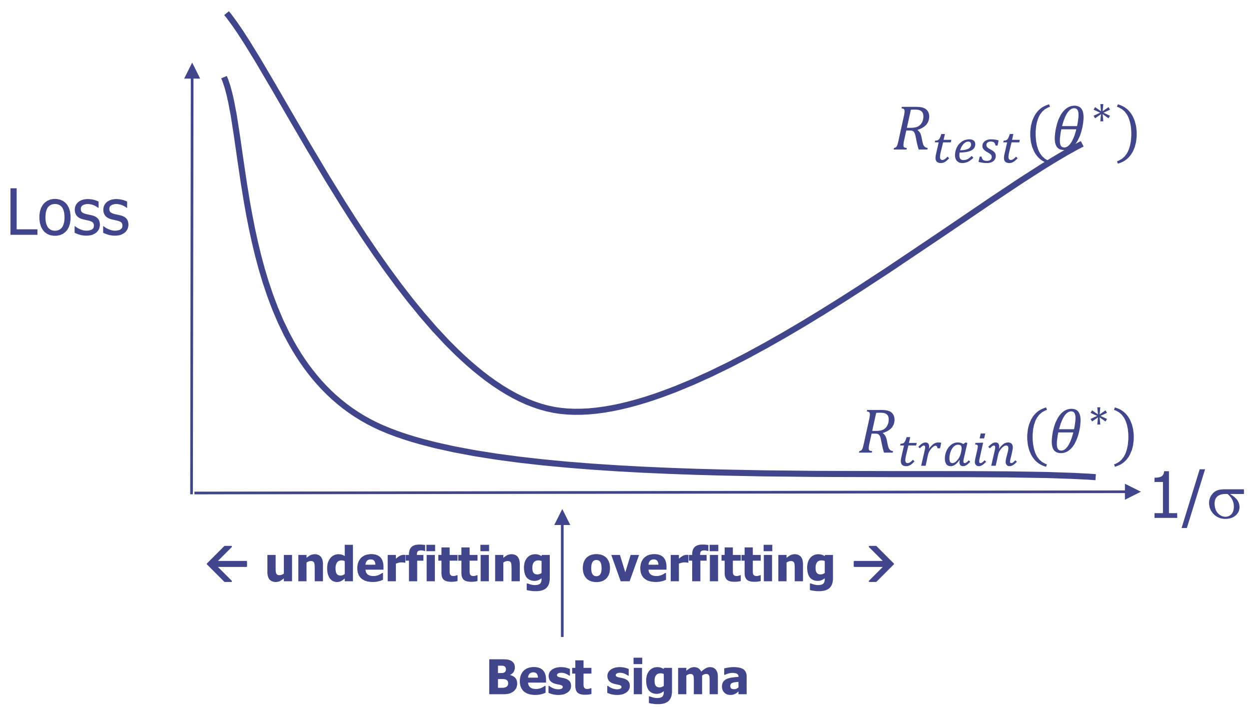
$$R_{train}(\theta) = \frac{1}{2N}\sum_{i=1}^{N}(y_i-\theta^T x_i)^2$$

- Evaluate it with testing data:

$$R_{train}(\theta) = \frac{1}{2M}\sum_{i=N+1}^{N+M}(y_i-\theta^T x_i)^2$$

Crossvalidation

- Try fitting with different sigma radial basis function widths
- Select sigma which gives lowest $R_{test}(\theta^*)$



- Think of sigma as a measure of the simplicity of the model
- Thinner RBFs are more flexible and complex

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

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

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

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

- 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

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

- Solution for Regularized Risk with Least Squares Loss:

$$\nabla_\theta R_{regularized} = 0$$

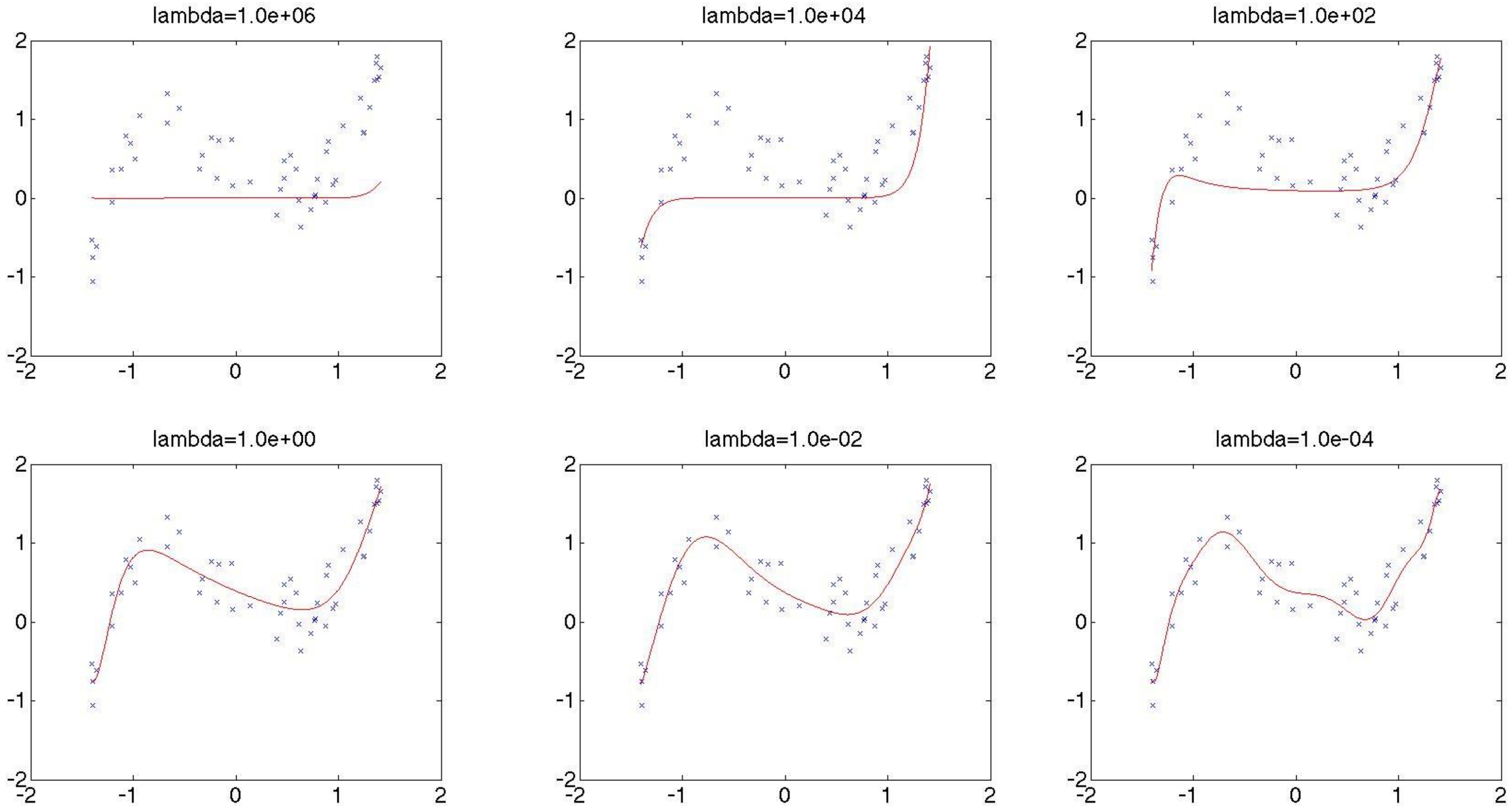
$$\nabla_\theta \left(\frac{1}{2N}\|\boldsymbol{y} - \boldsymbol{X}\theta\| + \frac{\lambda}{2}\|\theta\|^2\right) = 0$$

$$\frac{1}{2N}(-2\boldsymbol{X}^T\boldsymbol{y} + 2\boldsymbol{X}^T\boldsymbol{X}\theta) + \frac{\lambda}{2}(2\theta) = 0$$

$$\theta^* = (\boldsymbol{X}^T\boldsymbol{X} + \lambda N I)^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

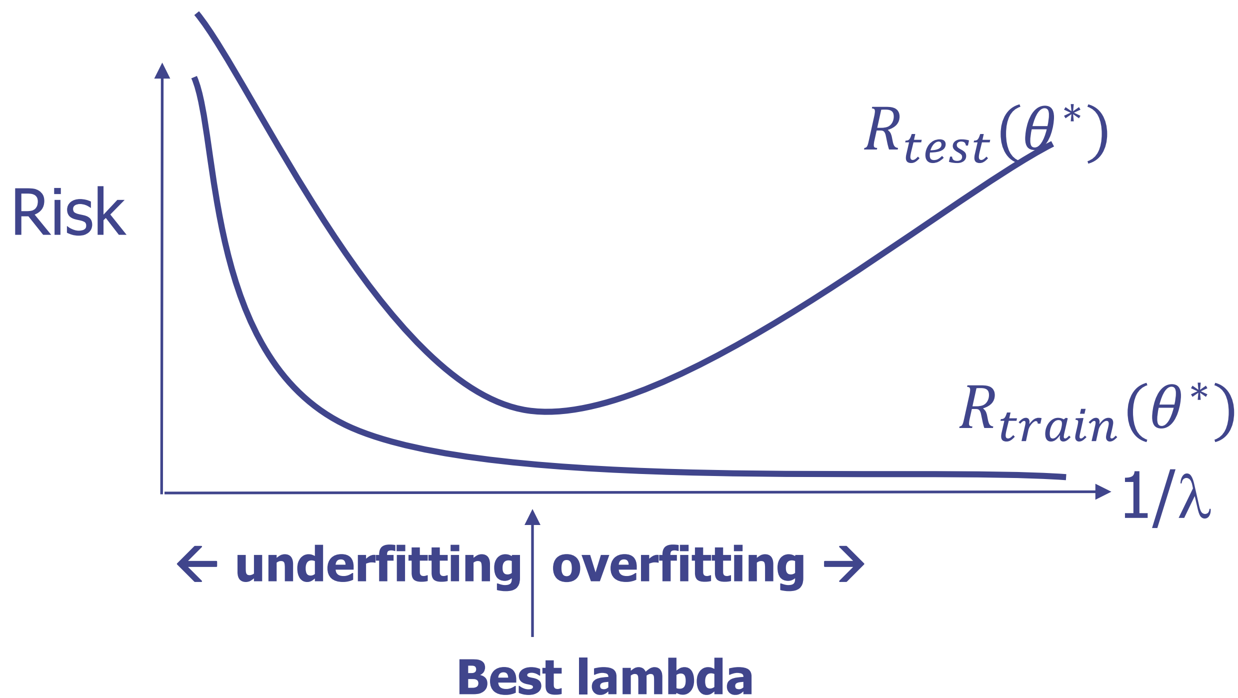
Regularized Risk Minimization

- 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

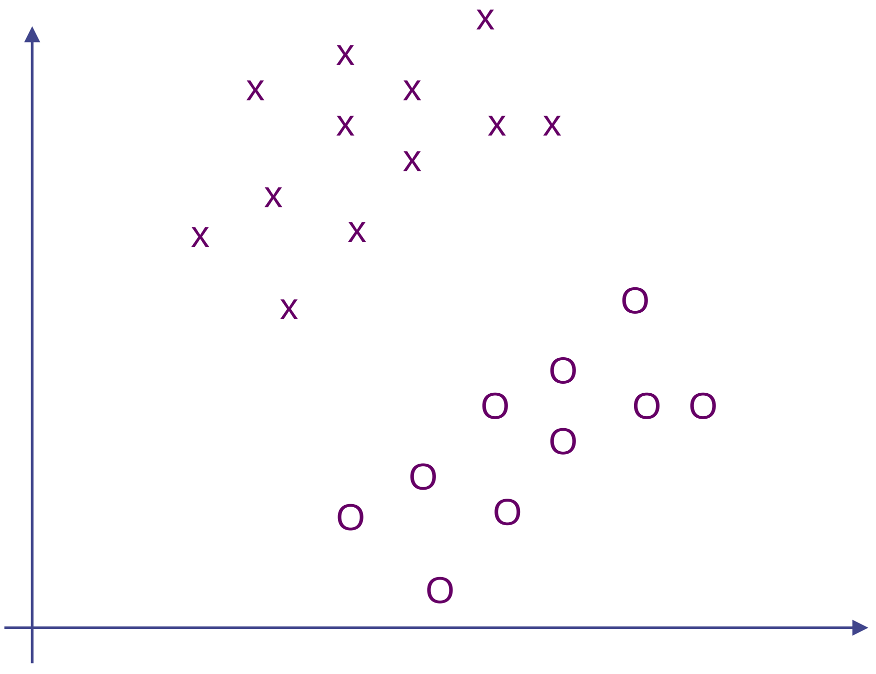
- Try fitting with different lambda regularization levels
- Select lambda which gives lowest $R_{test}(\theta^*)$



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

Class 5

- **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

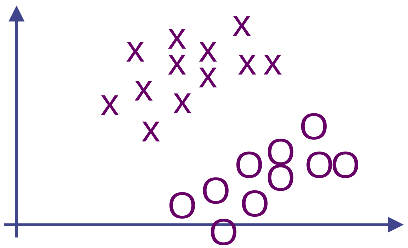
From Regression To Classification

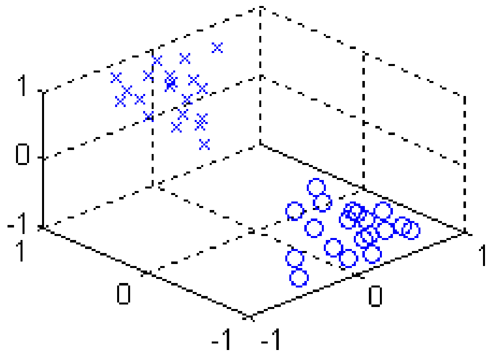
- Classification is another important learning problem

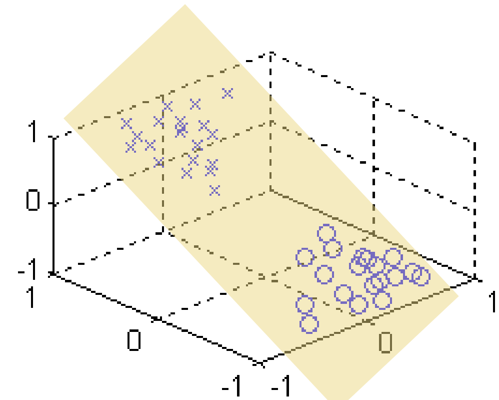
Classification: $X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_N, y_N)\}, \boldsymbol{x} \in \boldsymbol{R}^D, y \in \{0,1\}$

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?







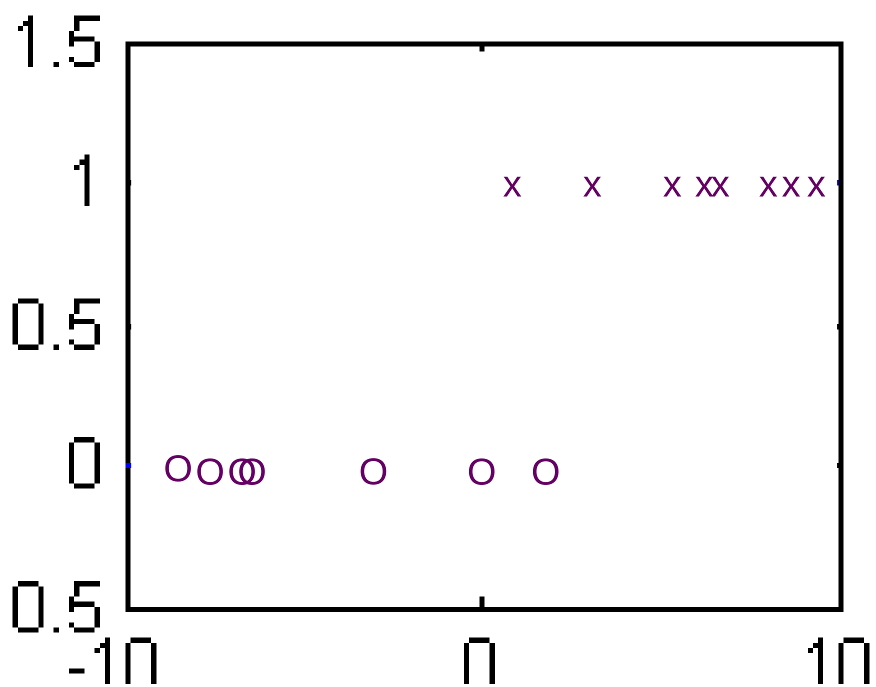
Logistic Regression

- 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?

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

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

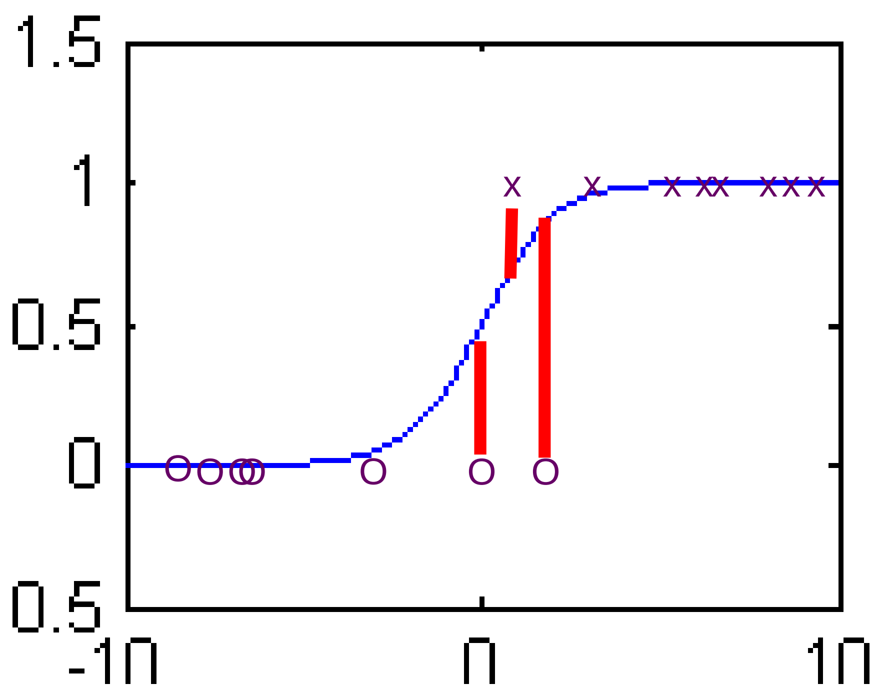
Logistic Regression

- Given a classification problem with binary outputs

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

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

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



Logistic Regression

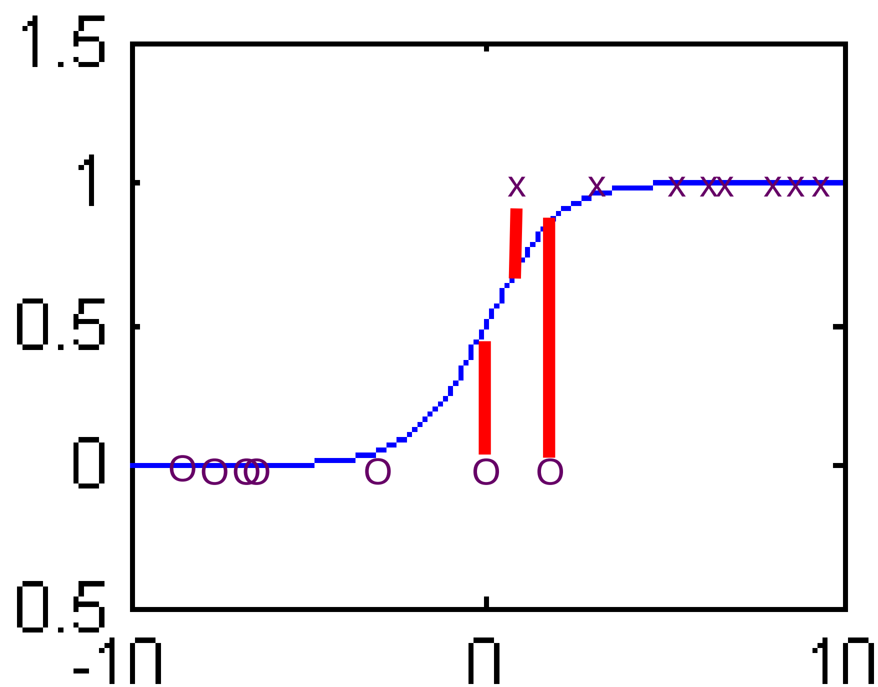
- Given a classification problem with binary outputs

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

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



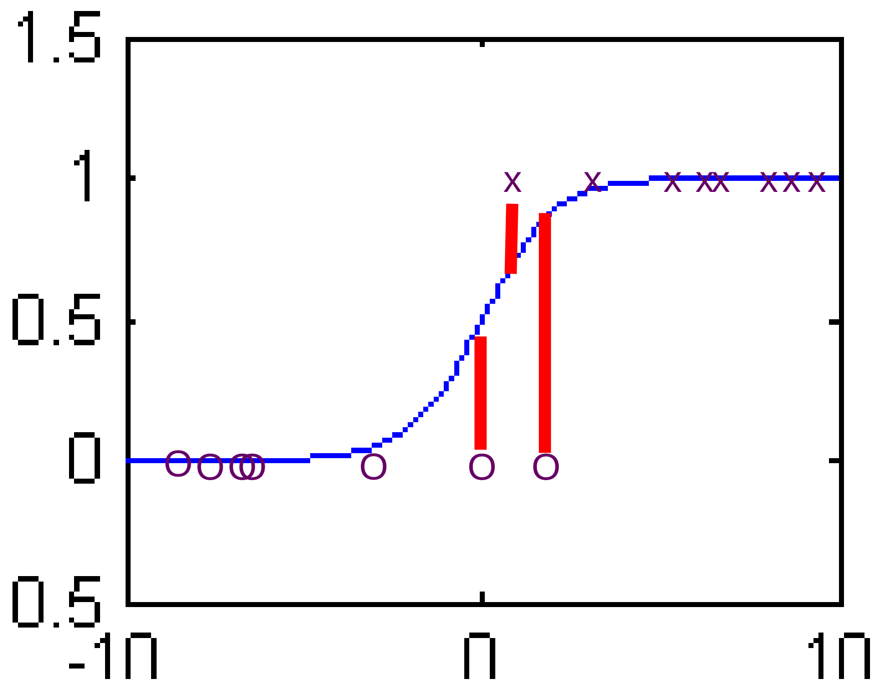
Logistic Regression

- Given a classification problem with binary outputs

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

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

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

$$= \prod_{y_i=0} (1 - f(\boldsymbol{x}_i; \theta)) \prod_{y_i=1} f(\boldsymbol{x}_i; \theta)$$

Logistic Regression

- Given a classification problem with binary outputs

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

- 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:

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:

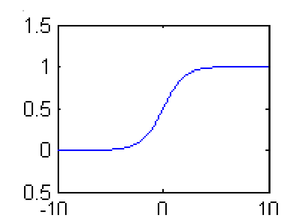
$$R_{emp}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - 1)\log\big(1 - f(\boldsymbol{x}_i; \theta)\big) - y_i\log(f(\boldsymbol{x}_i; \theta))$$

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:

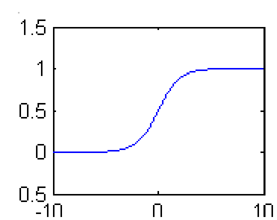
$$R_{emp}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - 1)\log\big(1 - f(\boldsymbol{x}_i;\theta)\big) - y_i\log(f(\boldsymbol{x}_i;\theta))$$

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

Find best $\boldsymbol{\theta}$ numerically!

where

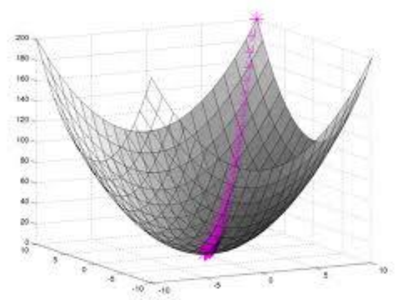
$$f(\boldsymbol{x};\theta) = \frac{1}{1 + exp(-\theta\boldsymbol{x})} = g(\theta^T\boldsymbol{x})$$



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

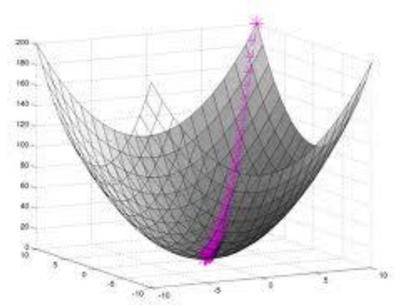
Gradient Descent

- 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*

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

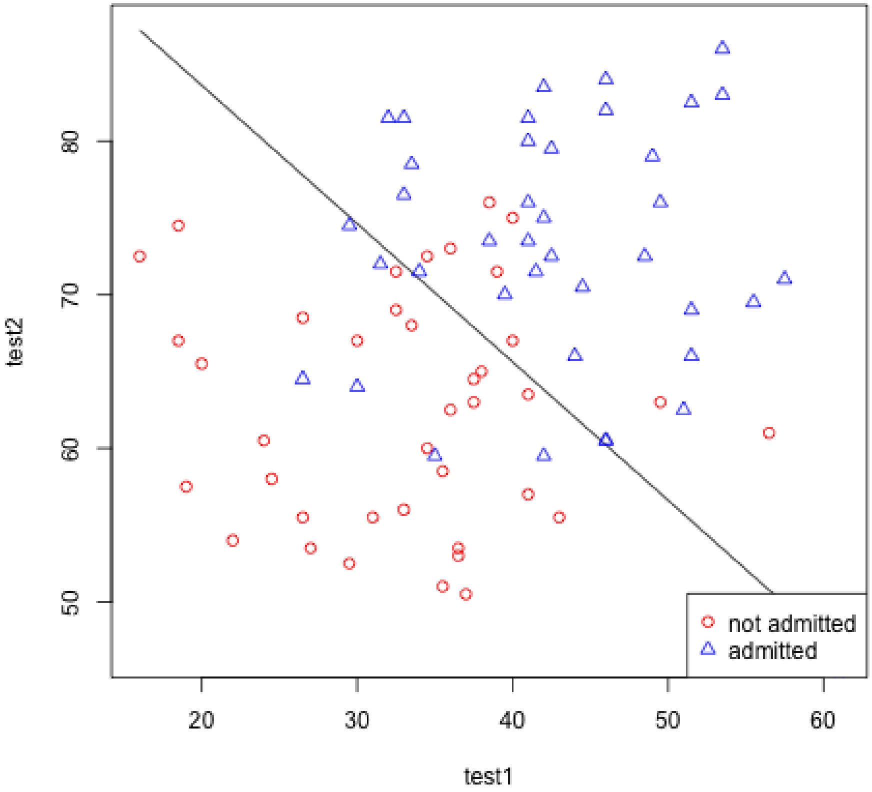
while $\|\theta^t - \theta^{t-1}\| \geq \epsilon$ *{*

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

- 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