CS540 Machine learning
Lecture 4

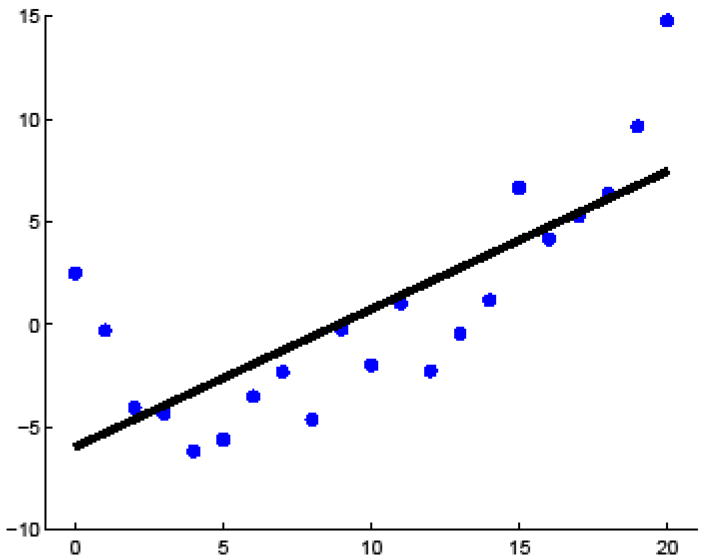
Last time

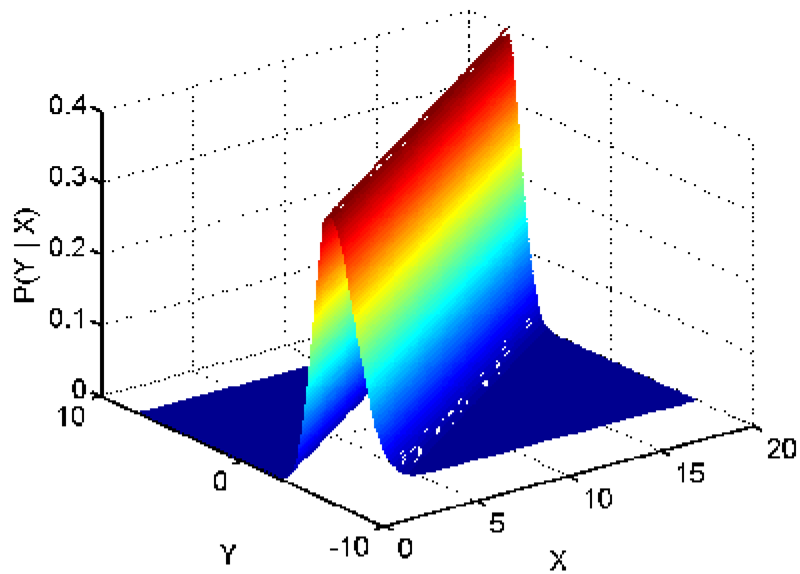
- Basic concepts
  - Loss functions
  - Estimation vs inference
  - Decision boundaries
  - Overfitting
  - Regularization
  - Model selection
  - Structural error vs approximation error

This time

- Basis functions
- Normal equations
- QR
- SVD

Linear regression



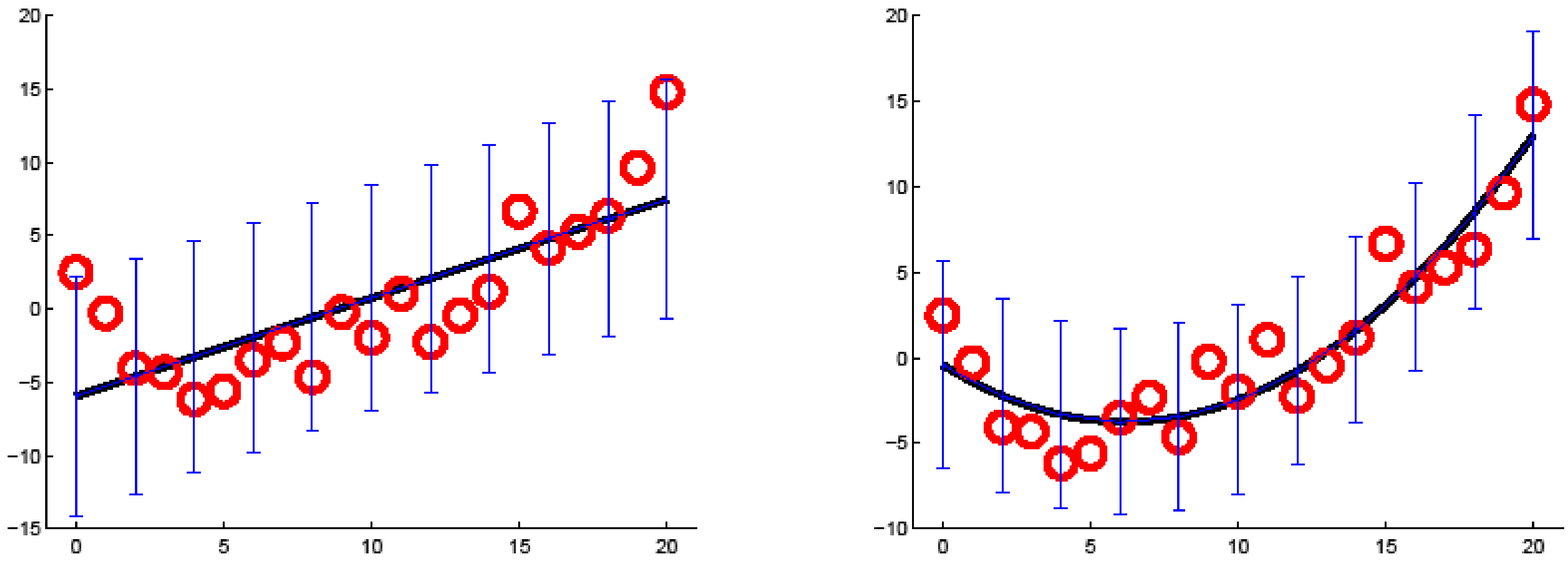


$$p(y|\mathbf{x}, \boldsymbol{\theta}) \quad = \quad \mathcal{N}(y|\mathbf{w}^T\mathbf{x}, \sigma^2)$$

Polynomial Regression

$$f(x) = w_0 + w_1 x + w_2 x^2 + \cdots + w_d x^D$$

$$f(x) = \mathbf{w}^T \boldsymbol{\phi}(x) = \sum_{j=1}^{d} w_k \phi_j(\mathbf{x})$$

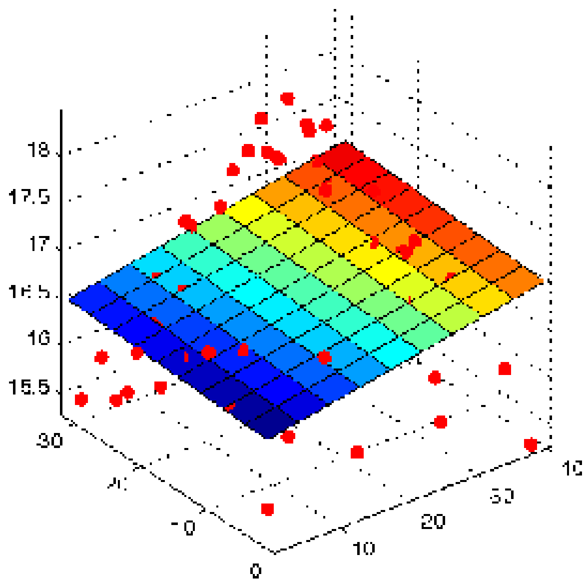


Line denotes posterior mode arg $\max_y$ p(y|x)

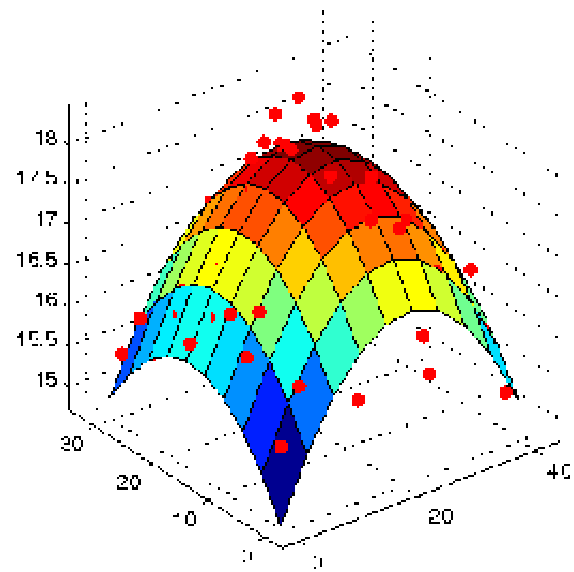
Error bars denote 95% credible interval $p(y \in I|\mathbf{x}) = 0.95$

Polynomial Regression

$$f(x) = \mathbf{w}^T \boldsymbol{\phi}(x) = \sum_{j=1}^{d} w_k \phi_j(\mathbf{x})$$



$$f(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2$$



$$f(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2$$

$$f(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 + w_5 x_1 x_2$$

Interaction term

Polynomial basis

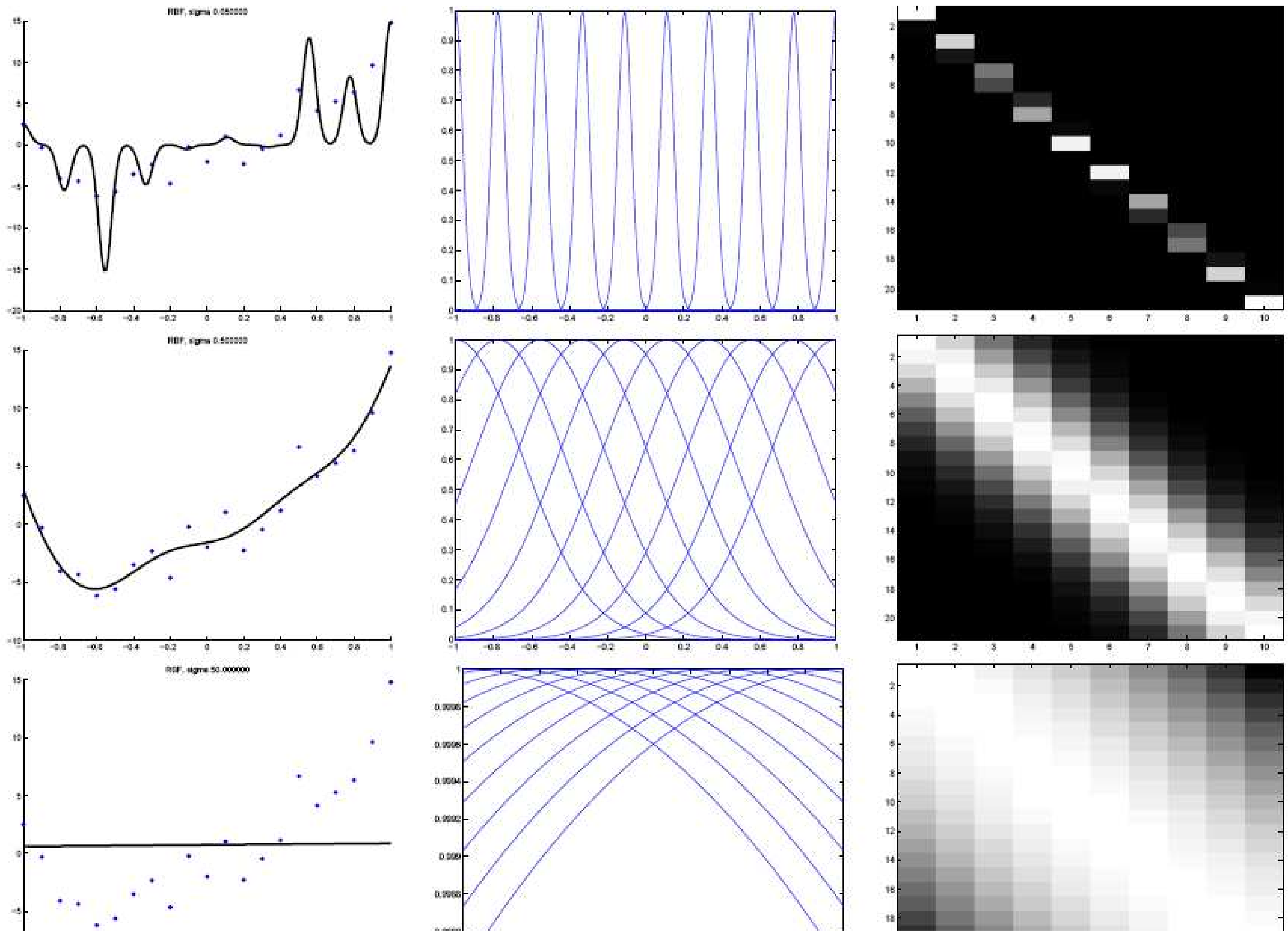
- Linear regression can fit nonlinear functions, provided the nonlinearity is fixed

$$\mathbf{\Phi} = \begin{pmatrix} 1 & x_1 & x_1^2 & x_1^3 \\ & \vdots & & \\ 1 & x_n & x_n^2 & x_n^3 \end{pmatrix}$$

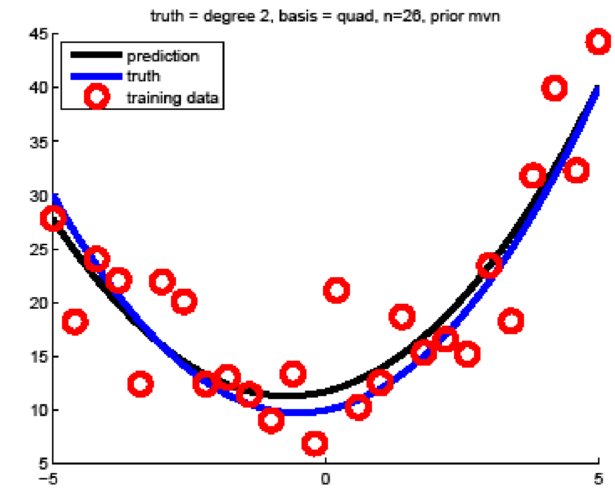
Radial basis functions (RBF)

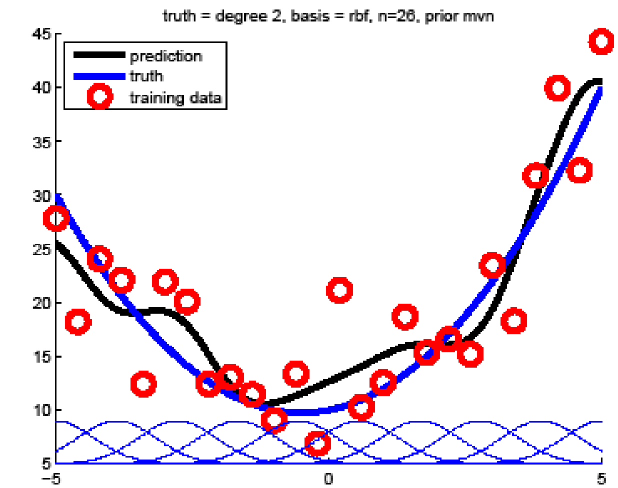
- Measure distance to examplars

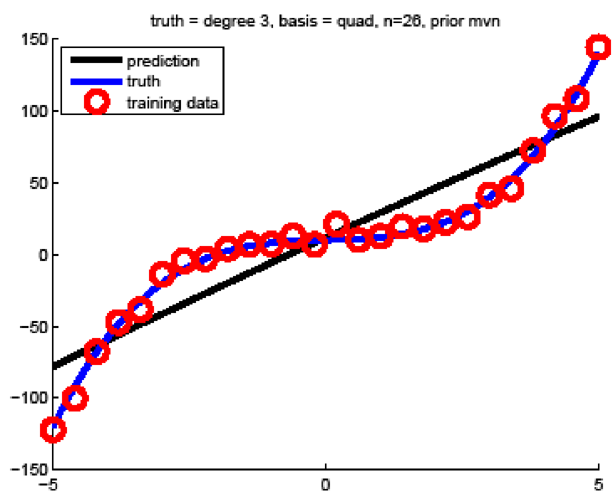
$$\phi(\mathbf{x}) = [K(\mathbf{x}, \boldsymbol{\mu}_1), \ldots, K(\mathbf{x}, \boldsymbol{\mu}_d)], \quad K(\mathbf{x}, \boldsymbol{\mu}) = \exp\left(-\frac{||\mathbf{x} - \boldsymbol{\mu}||^2}{2\sigma^2}\right)$$

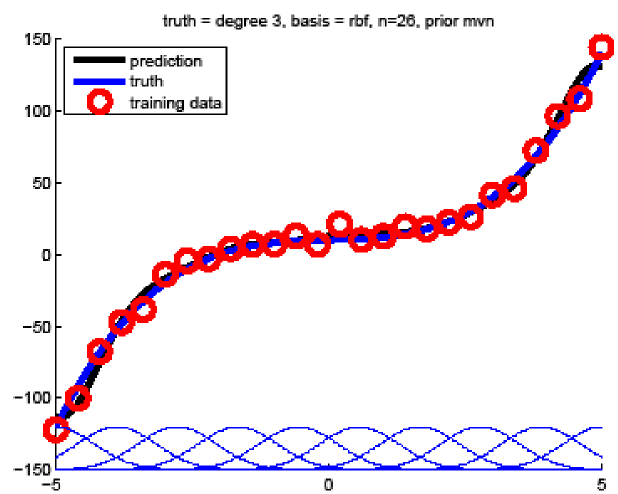


RBF vs polynomials









Categorical features

- Not meaningfully ordered, so use 1-of-K encoding to embed into a vector space

$$\begin{aligned}
\boldsymbol{\phi}(x) &= [I(x=r), I(x=g), I(x=b)] \\
\boldsymbol{\phi}(x) &= [1, I(x=r), I(x=g)] \\
p(y|x,\boldsymbol{\theta}) &= \mathcal{N}(y|w_0 + w_1 I(x=r) + w_2 I(x=g), \sigma^2) \\
E(y|x=r,\boldsymbol{\theta}) &= w_0 + w_1, \;\; E(y|x=g,\boldsymbol{\theta}) = w_0 + w_2, \;\; E(y|x=b,\boldsymbol{\theta}) = w_0
\end{aligned}$$

Standardization

- Often need to ensure features are on same scale (numerics, ridge)

$$z_{ij} = \frac{x_{ij} - \overline{x}_j}{\sigma_j}$$

BLT

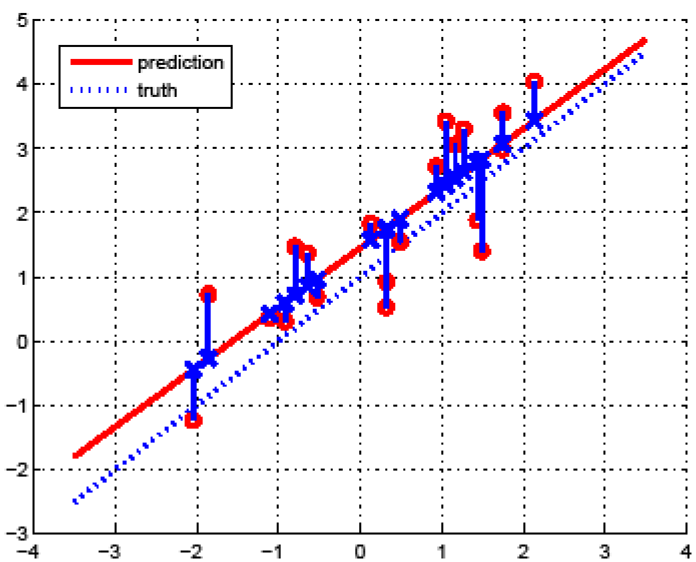
Listing 1: :

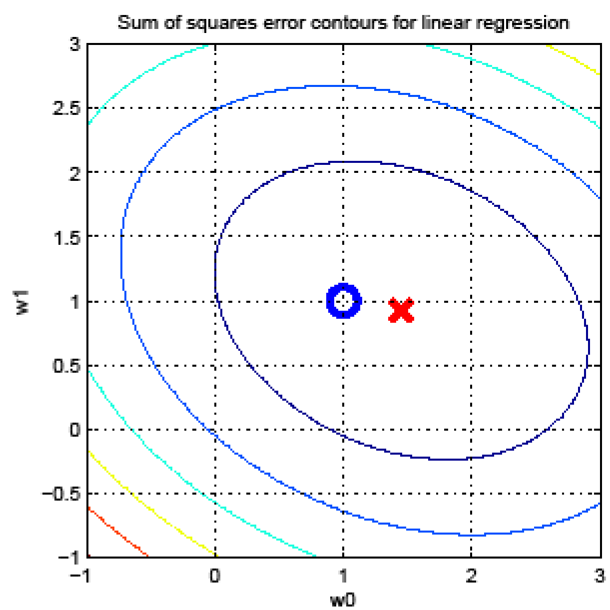
```
%Part of \codename{linregDist.demoPolyfitDegree}}
m = linregDist;
m.transformer =  chainTransformer({rescaleTransformer, polyBasisTransformer(deg)})
m = fit(m, 'X', xtrain, 'y', ytrain);
ypredTest = predict(m, xtest);
testMse = mean((ypredTest - ytest).^2);
```

MLE for linear regression (least squares)

$$
\begin{aligned}
p(\mathcal{D}|\mathbf{w},\sigma^2) &= \prod_{i=1}^{n} \mathcal{N}(y_i|\mathbf{w}^T\mathbf{x}_i,\sigma^2) \\
&= (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{y}-\mathbf{X}\mathbf{w})^T(\mathbf{y}-\mathbf{X}\mathbf{w})\right) \\
J(\mathbf{w},\sigma^2) &= -\log p(\mathbf{y}|X,\mathbf{w},\sigma^2) \\
&= \frac{n}{2}\log(\sigma^2) + \frac{1}{2\sigma^2} RSS(\mathbf{w}) \\
RSS(\mathbf{w}) &= ||\mathbf{X}\mathbf{w}-\mathbf{y}||_2^2 = (\mathbf{y}-\mathbf{X}\mathbf{w})^T(\mathbf{y}-\mathbf{X}\mathbf{w})
\end{aligned}
$$

Negative log likelihood





Normal equations

$$
\begin{aligned}
\nabla_{\mathbf{w}} RSS(\mathbf{w}) &= \mathbf{0} \\
\hat{\mathbf{w}} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = (\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^T)^{-1}(\sum_{i=1}^{n} y_i\mathbf{x}_i)
\end{aligned}
$$

See book for derivation

MLE = OLS estimate

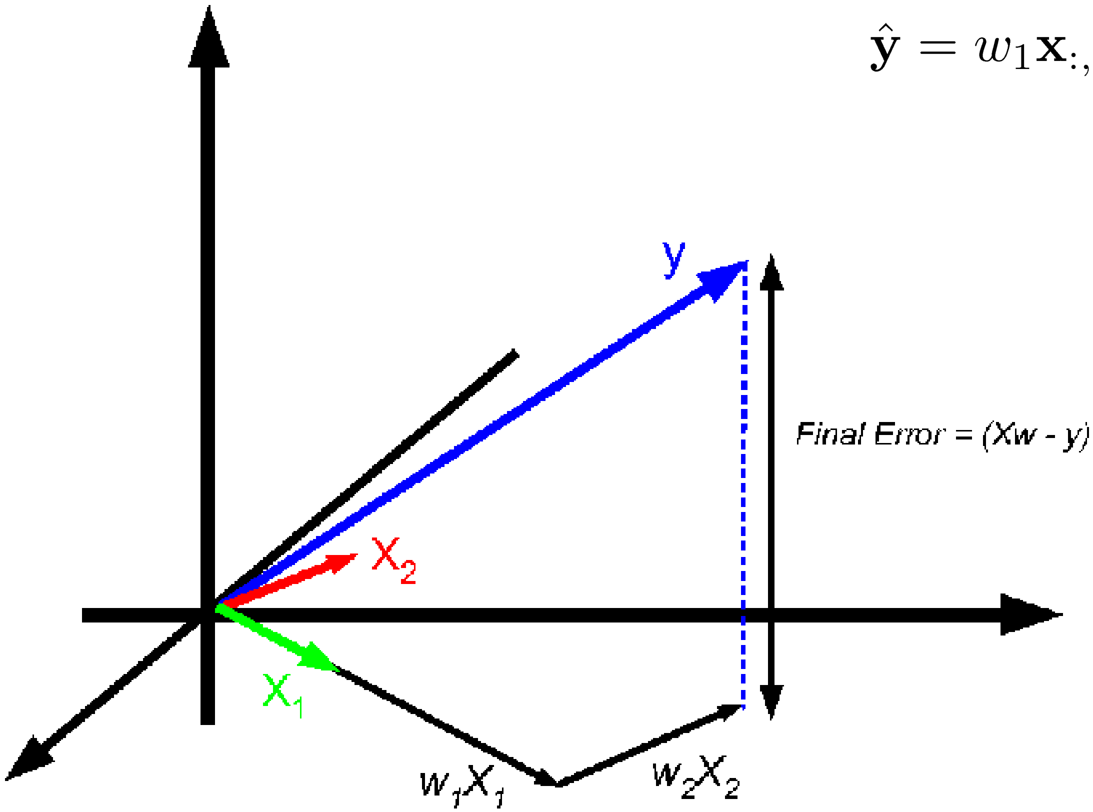
Uncertainty in estimate – see later

Geometry of least squares

$$\mathbf{X} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} 3 \\ 2 \\ 3 \end{pmatrix}$$

Minimize RSS by orthogonal projection of y into column space of X

$$\hat{\mathbf{y}} = w_1 \mathbf{x}_{:,1} + \cdots + w_d \mathbf{x}_{:,d}$$



Orthogonal projection

- Projection of y onto X

$$\mathrm{Proj}(\mathbf{y};\mathbf{X}) = \mathrm{argmin}_{\hat{\mathbf{y}}\in\mathrm{span}(\{\mathbf{x}_1,\ldots,\mathbf{x}_n\})} \|\mathbf{y}-\hat{\mathbf{y}}\|_2.$$

- Let r = y - \hat{y}. Residual must be orthogonal to X. Hence

$$\mathbf{x}_j^T(\mathbf{y}-\hat{\mathbf{y}}) = 0 \Rightarrow \mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}) = \mathbf{0} \Rightarrow \mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

- Prediction on training set

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\mathbf{w}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \stackrel{\mathrm{def}}{=} \mathbf{H}\mathbf{y}$$

Hat matrix

- Residual is orthogonal

$$\mathbf{X}^T(\mathbf{y}-\mathbf{H}\mathbf{y}) = \mathbf{X}^T(\mathbf{y}-\mathbf{X}\hat{\mathbf{w}}) = \mathbf{X}^T\mathbf{y} - \mathbf{X}^T\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{0}$$

Solving for offset

- Let us separate w_0 from the other weights

$$J(\mathbf{w}, \hat{w}_0) \quad = \quad \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \mathbf{w} - w_0)^2$$

- One can show (homework) that

$$\hat{w}_0 \quad = \quad \frac{1}{n} \sum_i y_i - \frac{1}{n} \sum_i \mathbf{x}_i^T \mathbf{w} = \overline{y} - \overline{\mathbf{x}}^T \mathbf{w}$$

- And

$$\hat{\mathbf{w}} = (\mathbf{X}_c^T \mathbf{X}_c)^{-1} \mathbf{X}_c^T \mathbf{y}_c = [\sum_{i=1}^{n} (\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T]^{-1} [\sum_{i=1}^{n} (y_i - \overline{y})(\mathbf{x}_i - \overline{\mathbf{x}})]$$

- For 1d data:

$$\begin{aligned} w_1 &= \frac{\sum_i (x_i - \overline{x})(y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2} = \frac{\sum_i x_i y_i - n\overline{xy}}{\sum_i x_i^2 - n\overline{x}^2} \\ w_0 &= \bar{y} - w_1 \bar{x} \end{aligned}$$

Solving for σ^2

- One can show

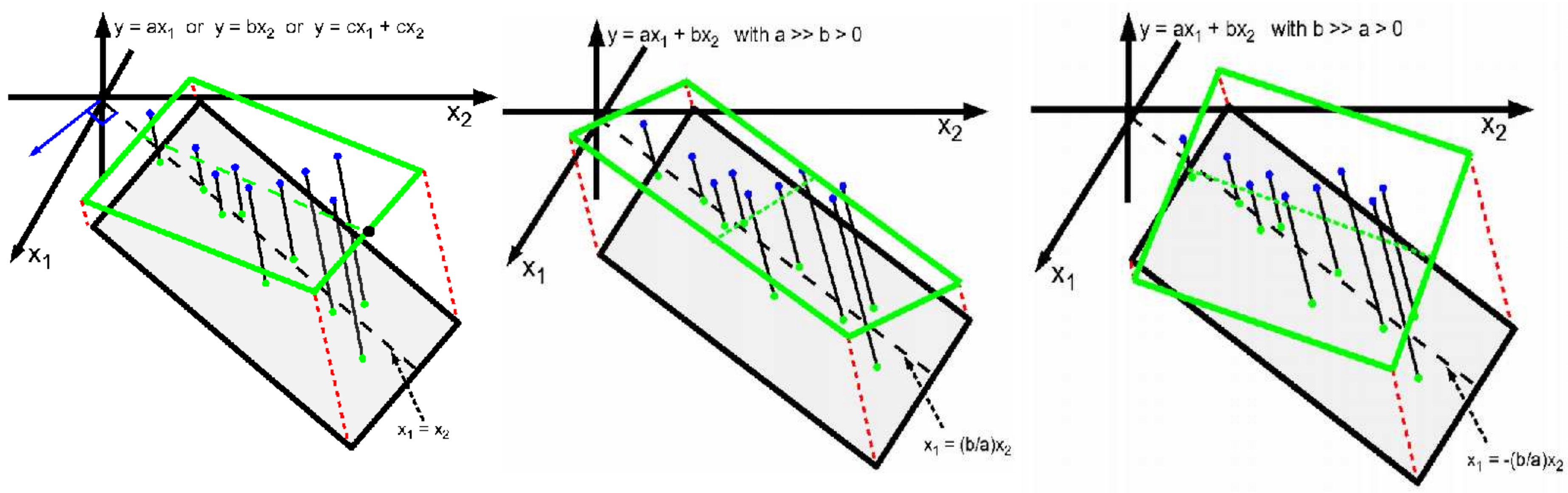
$$\hat{\sigma}^2 = \frac{1}{n}(\mathbf{y} - X\hat{\mathbf{w}})^T(\mathbf{y} - X\hat{\mathbf{w}}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{x}_i^T\hat{\mathbf{w}})^2$$

Colinearity

- Consider if x1=x2

$$w_1\mathbf{x}_1 + w_2\mathbf{x}_2 = (w_1 + w_2)\mathbf{x}_1 = (w_1 + w_2)\mathbf{x}_2$$

What solution should we return?



Null space

- Consider rank 2 matrix (2nd = avg of 1 + 3)

$$\mathbf{X} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \\ 13 & 14 & 15 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} 16 \\ 17 \\ 18 \\ 19 \\ 20 \end{pmatrix}$$

- Let z be in the null space of X, ie Xz = 0. Then

$$\mathbf{X}\mathbf{w} = \mathbf{y} \Rightarrow \mathbf{X}(\mathbf{w} + c\mathbf{z}) = \mathbf{y}$$

```
X = reshape(1:15, [3 5])'; y = (16:20)';
w = X\y; z =[1;-2;1];
c = rand; assert(approxeq(norm(X*(w+c*z) - y),0))
```

- What solution should we return?

Condition number

- Suppose X is full rank so solution is theoretically unique. May be hard to find numerically.

$$\mathbf{X} = \begin{pmatrix} 1 & 1 \\ \delta & 0 \\ 0 & \delta \end{pmatrix}, \quad \mathbf{X}^T\mathbf{X} = \begin{pmatrix} 1+\delta^2 & 1 \\ 1 & 1+\delta^2 \end{pmatrix}, \quad \kappa(\mathbf{X}^T\mathbf{X}) = \kappa(\mathbf{X})^2$$

- We see methods for finding the MLE that do not invert $X^T X$
- Each method will resolve the ambiguity issue in a different way

QR decomposition

- We find a set of orthonormal vectors q_j that span successive columns of X (using Gram-Schmidt orthogonalization)

$$\begin{aligned} \mathbf{x}_1 &= r_{11}\mathbf{q}_1 \\ \mathbf{x}_2 &= r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2 \\ &\vdots \\ \mathbf{x}_n &= r_{1n}\mathbf{q}_1 + \cdots + r_{nn}\mathbf{q}_n \end{aligned}$$

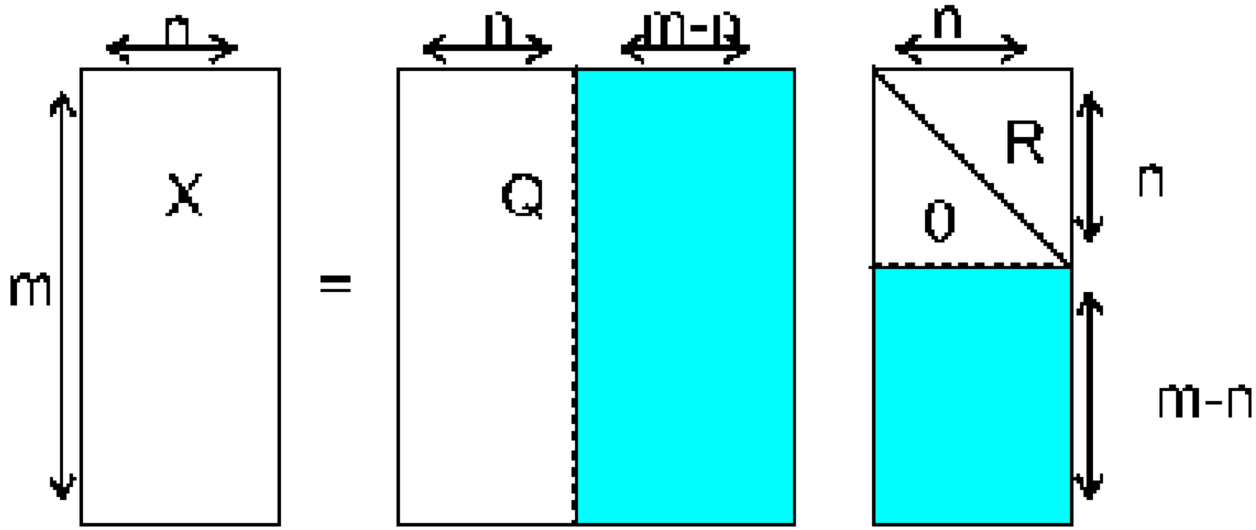
$$\mathbf{q}_i^T\mathbf{q}_j = \delta_{ij}$$

$$\begin{pmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_n \\ | & | & & | \end{pmatrix} = \begin{pmatrix} | & | & & | \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_n \\ | & | & & | \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & & \cdots \\ & & \ddots & \\ & & & r_{nn} \end{pmatrix}$$

QR decomposition

- Can make Q and R be square m x m matrices so we can write $\mathbf{Q}^T\mathbf{Q} = \mathbf{Q}\mathbf{Q}^T = \mathbf{I}$

$$\begin{pmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_n \\ | & | & & | \end{pmatrix} = \begin{pmatrix} | & | & & | \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_n \\ | & | & & | \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & & \cdots \\ & & \ddots & \\ & & & r_{nn} \end{pmatrix}$$



Least squares with QR

- We have

$$
\begin{aligned}
\hat{\mathbf{w}} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \\
&= (\mathbf{R}^T\mathbf{Q}^T\mathbf{Q}\mathbf{R})^{-1}\mathbf{R}^T\mathbf{Q}^T\mathbf{y} \\
&= (\mathbf{R}^T\mathbf{R})^{-1}\mathbf{R}^T\mathbf{Q}^T\mathbf{y} \\
&= \mathbf{R}^{-1}\mathbf{R}^{-T}\mathbf{R}^T\mathbf{Q}^T\mathbf{y} \\
&= \mathbf{R}^{-1}\mathbf{Q}^T\mathbf{y}
\end{aligned}
$$

- Let $z = Q^T y$. Solve $w=R^{-1} z$ by back substitution, w = R \ z.

```
[Q,R] = qr(X,0);
w = R\(Q'*y);
```

Shorthand `w=X\y;`

Basic solution

- Let r = rank(X). Basic solution has r non-zeros.
- w=X\y returns one of many possible basic solutions.

$$\mathbf{X} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \\ 13 & 14 & 15 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} 16 \\ 17 \\ 18 \\ 19 \\ 20 \end{pmatrix}$$

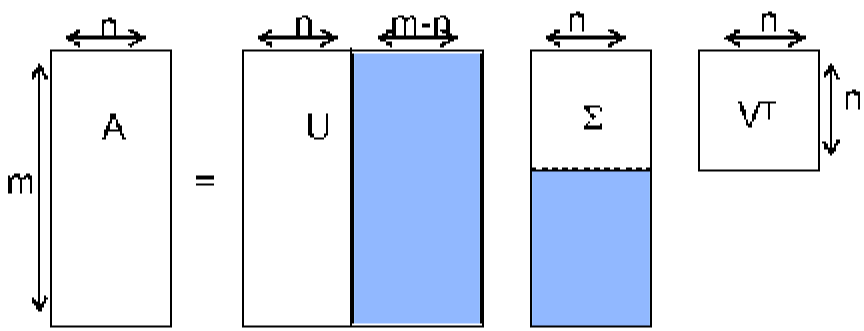
```
X = reshape(1:15, [3 5])'; y = (16:20)';
w = X\y % [-7.5, 0. 7.83]
norm(X*w - y) % 0.00

w = [0,-15,15.3333]';
norm(X*w - y) % 0.00
```

SVD

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sigma_1 \begin{pmatrix} | \\ \mathbf{u}_1 \\ | \end{pmatrix} \begin{pmatrix} - & \mathbf{v}_1^T & - \end{pmatrix} + \cdots + \sigma_r \begin{pmatrix} | \\ \mathbf{u}_r \\ | \end{pmatrix} \begin{pmatrix} - & \mathbf{v}_r^T & - \end{pmatrix}$$

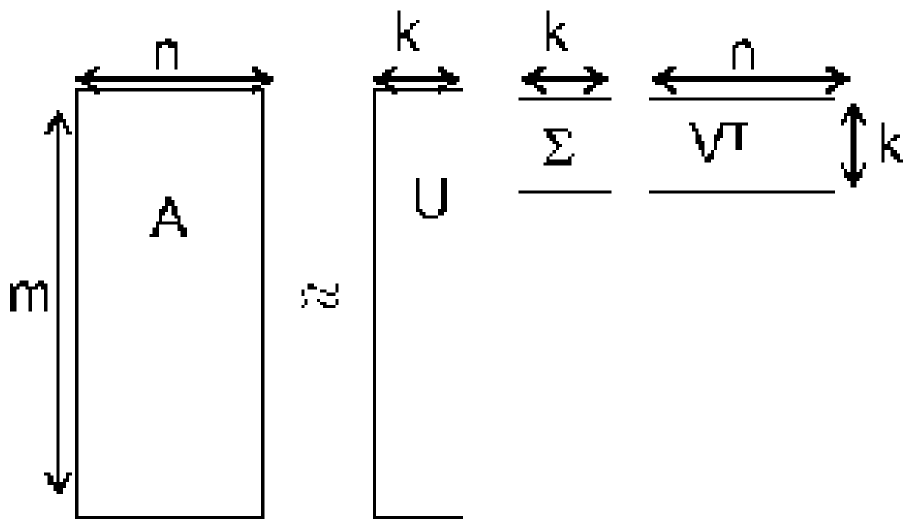
$$\begin{aligned} \mathbf{U}^T\mathbf{U} &= \mathbf{I} \\ \mathbf{V}^T\mathbf{V} &= \mathbf{V}\mathbf{V}^T = \mathbf{I} \end{aligned}$$



Truncated SVD

- Rank k approximation to a matrix

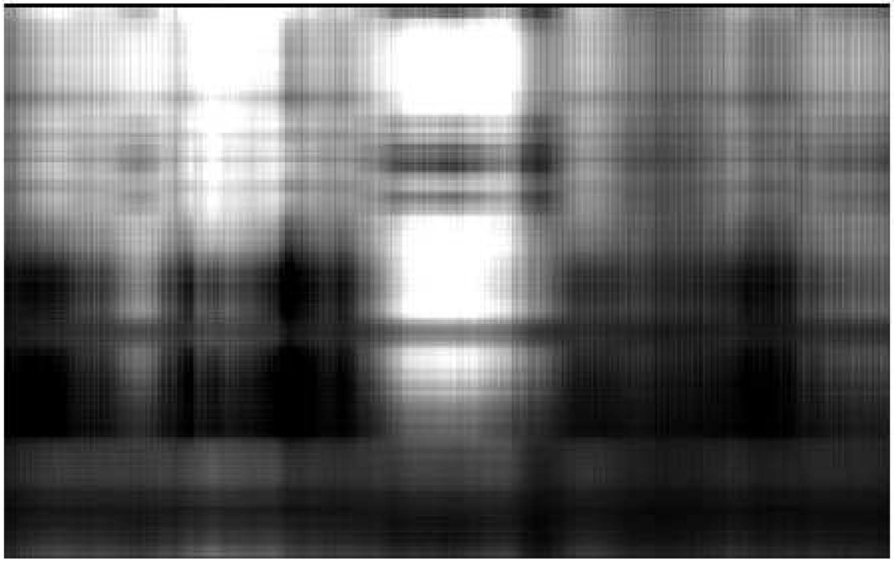
$$\mathbf{A}_k = \sum_{j=1}^{k} \sigma_j \mathbf{u}_j \mathbf{v}_k^T = \mathbf{U}_{:,1:k} \ \ \mathbf{\Sigma}_{1:k,1:k} \ \ \mathbf{V}_{:,1:k}^T$$



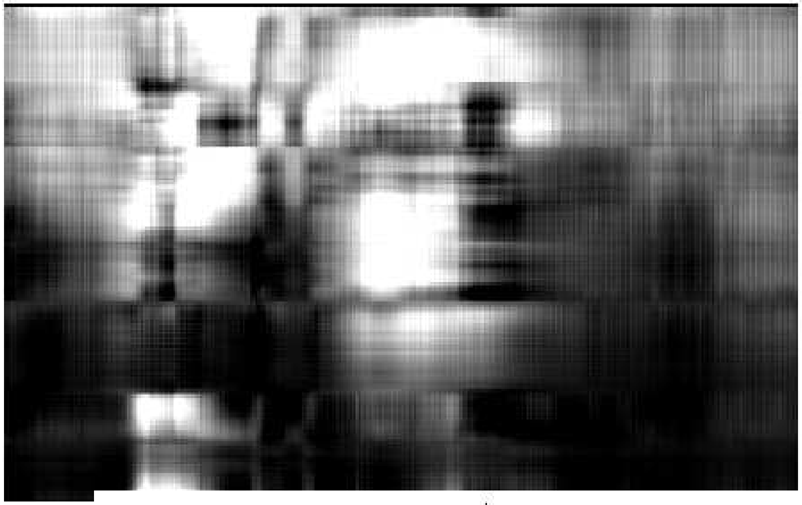
Equivalent to PCA

Truncated SVD

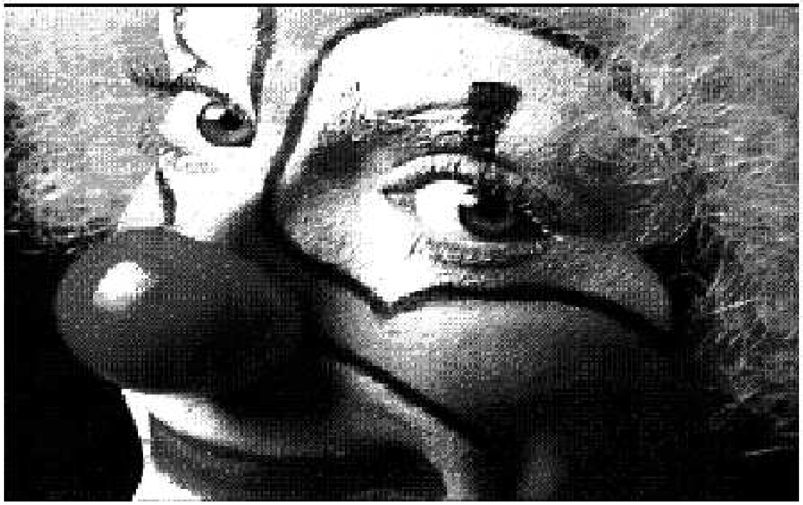
rank 2



rank 5



rank 200



```
load clown; % built-in image
[U,S,V] = svd(X,0);
k = 20;
Xhat = (U(:,1:k)*S(1:k,1:k)*V(:,1:k)');
image(Xhat);
```

SVD for least squares

- We have

$$
\begin{aligned}
\hat{\mathbf{w}} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \\
\mathbf{X}^T\mathbf{X}\mathbf{w} &= \mathbf{X}^T\mathbf{y} \text{ (premultiply by } \mathbf{X}^T\mathbf{X}) \\
\mathbf{V}\mathbf{D}\mathbf{U}^T\mathbf{U}\mathbf{D}\mathbf{V}^T\mathbf{w} &= \mathbf{V}\mathbf{D}\mathbf{U}^T\mathbf{y} \text{ (SVD expansion)} \\
\mathbf{V}\mathbf{D}^2\mathbf{V}^T\mathbf{w} &= \mathbf{V}\mathbf{D}\mathbf{U}^T\mathbf{y} \text{ (since } \mathbf{U}^T\mathbf{U} = \mathbf{I} \text{ and } \mathbf{D}\mathbf{D} = \mathbf{D}^2) \\
\mathbf{D}^2\mathbf{V}^T\mathbf{w} &= \mathbf{D}\mathbf{U}^T\mathbf{y} \text{ (premultiply by } \mathbf{V}^T) \\
\mathbf{V}^T\mathbf{w} &= \mathbf{D}^{-1}\mathbf{U}^T\mathbf{y} \text{ (premultiply by } \mathbf{D}^{-2}) \\
\mathbf{w} &= \mathbf{V}\mathbf{D}^{-1}\mathbf{U}^T\mathbf{y} \text{ (premultiply by } \mathbf{V})
\end{aligned}
$$

```
[U,D,V]=svd(X,0);
Dinv = diag(1./(diag(D)));
w = V*Dinv*U'*y;
```

What if $D_j = 0$ (so rank of X is less than d)?

Pseudo inverse

- If D_j=0, use

$$\mathbf{w} = \mathbf{V}\mathbf{D}^{\dagger}\mathbf{U}^T\mathbf{y} \stackrel{\text{def}}{=} \mathbf{X}^{\dagger}\mathbf{y}, \quad \mathbf{D}^{\dagger} = \text{diag}(\sigma_1^{-1}, \ldots, \sigma_r^{-1}, 0, \ldots, 0)$$

```
function B = pinv(A)
[U,S,V] = svd(A,0);
s = diag(S);
r = sum(s > tol); % rank
w = diag(ones(r,1) ./ s(1:r));
B = V(:,1:r) * w * U(:,1:r)';
```

- Of all solutions w that minimize ||Xw – y||, the pinv solution also minimizes ||w||

```
w = X\y;
w2 = pinv(X)*y;
[norm(w) norm(w2)]
>> 10.8449   10.8440
```