Least squares in Julia

Reese Pathak    Stephen Boyd

EE103
Stanford University

November 15, 2016
Outline

Least squares

Multi-objective least squares

Linearly constrained least squares
Least squares approximate solution in Julia

the math:

- $\hat{x}$ minimizes $\|Ax - b\|^2$; $A$ has independent columns
- $\hat{x} = (A^T A)^{-1} A^T b = A^\dagger b = R^{-1} Q^T b$
  $(A = QR$ is $QR$-factorization of $A$)

in Julia:

- $\text{xhat} = \text{inv}(A'\ast A)\ast(A'\ast b)$
- $\text{xhat} = \text{pinv}(A)\ast b$
- $Q, R = \text{qr}(A); \text{xhat} = \text{inv}(R)\ast(Q'\ast b)$
- simplest method: $\text{xhat} = A\backslash b$
Example: Regression

- $N$ columns of $X$ are feature $n$-vectors
- $N$-vector $y$ gives associated outcomes
- regression model: find $n$-vector $\beta$, scalar $v$ that minimize

$$\|X^T \beta + v1 - y\|^2$$

- express objective as

$$\left\| \begin{bmatrix} 1 & X^T \end{bmatrix} \begin{bmatrix} v \\ \beta \end{bmatrix} - y \right\|^2$$

- in Julia:

```julia
beta_tilde = [ ones(N,1) X' ] \ y;
v = beta_tilde[1]; beta = beta_tilde[2:end];
```
The backslash operator

the backslash operator $x = A\backslash b$ is heavily overloaded

- if $A$ is square and invertible
  - $x = A^{-1}b$
  - the unique solution of square set of equations $Ax = b$
- if $A$ is tall with linearly independent columns
  - $x = (A^T A)^{-1} A^T b$
  - the unique least squares approximate solution of overdetermined equations $Ax = b$
- if $A$ is wide with linearly independent rows
  - $x = A^T (AA^T)^{-1} b$
  - the unique least norm solution of the underdetermined equations $Ax = b$
- in other cases, $A\backslash b$ will give an error message
Outline

Least squares

Multi-objective least squares

Linearly constrained least squares
Multi-objective least squares

the math (for two objectives):

- \( \hat{x} \) minimizes \( \lambda_1 \| A_1 x - b_1 \|^2 + \lambda_2 \| A_2 x - b_2 \|^2 \)
- \( \lambda_1, \lambda_2 > 0 \) are relative weights, trade off objectives
- solve by stacking:

\[
\hat{x} = \begin{bmatrix} \sqrt{\lambda_1} A_1 \\ \sqrt{\lambda_2} A_2 \end{bmatrix} \dagger \begin{bmatrix} \sqrt{\lambda_1} b_1 \\ \sqrt{\lambda_2} b_2 \end{bmatrix}
\]

- or \( \hat{x} = (\lambda_1 A_1^T A_1 + \lambda_2 A_2^T A_2)^{-1}(\lambda_1 A_1^T b_1 + \lambda_2 A_2^T b_2) \)

in Julia:

```julia
sl1=sqrt(lambda1); sl2=sqrt(lambda2);
x_hat = [sl1*A1; sl2*A2) \ [sl1*b1; sl2*b2 ]
```
Outline

Least squares

Multi-objective least squares

Linearly constrained least squares
Equality constrained least squares

the math:

- $\hat{x}$ minimizes $\|Ax - b\|^2$ subject to $Cx = d$
- $A$ is $m \times n$, $C$ is $p \times n$
- find $\hat{x}$ by solving $(n + p) \times (n + p)$ KKT system

$$
\begin{bmatrix}
2A^TA & C^T \\
C & 0 \\
\end{bmatrix}
\begin{bmatrix}
\hat{x} \\
z \\
\end{bmatrix}
=
\begin{bmatrix}
2A^Tb \\
d \\
\end{bmatrix}
$$

in Julia:

```julia
kkt_sol = [2*A'*A C'; C zeros(p,p)] \ [2*A'*b; d]
x_hat = kkt_sol[1:n]
```
Least norm problem

the math:

- $\hat{x}$ minimizes $\|x\|^2$ subject to $Cx = d$
- can solve by KKT system, or $\hat{x} = C^T(CCT)^{-1}d = C^\dagger d$

in Julia: $x_{\text{hat}} = C\backslash d$