MODEL FITTING I

Prof. Alexander Huth
2.18.2020

RECAP

$$Y = f(X)$$

* System identification
  * Linear
  * Linearized
  * Nonlinear

RECAP

$$Y = f(X)$$

* System identification
  * Linear $$Y = X\beta$$
  * Linearized $$Y = \mathbb{L}(X)\beta$$
  * Nonlinear $$Y = \Theta(X)$$

LINEAR REGRESSION

RESPONSE VARIABLES WEIGHTS NOISE

$$\underbrace{Y}_{\text{RESPONSE}} = \underbrace{X}_{\text{VARIABLES}}\underbrace{\beta}_{\text{WEIGHTS}} + \underbrace{\epsilon}_{\text{NOISE}}$$

LINEAR REGRESSION

$$\overset{\text{RESPONSE}}{Y} = \overset{\text{VARIABLES}}{X}\overset{\text{WEIGHTS}}{\beta} + \overset{\text{NOISE}}{\epsilon}$$

Loss function:

$$||Y - X\beta||_2$$

LINEAR REGRESSION

$$\overset{\text{RESPONSE}}{Y} = \overset{\text{VARIABLES}}{X}\,\overset{\text{WEIGHTS}}{\beta} + \overset{\text{NOISE}}{\epsilon}$$

* How do we solve for beta?
    * Analytically
    * Iteratively

ANALYTIC SOLUTION TO LINEAR REGRESSION

$$\underbrace{Y}_{\text{RESPONSE}} = \underbrace{X}_{\text{VARIABLES}}\underbrace{\beta}_{\text{WEIGHTS}} + \underbrace{\epsilon}_{\text{NOISE}}$$

$$\beta = f(X, Y)$$

ITERATIVE SOLUTION TO LINEAR REGRESSION

$$\overset{\text{RESPONSE}}{Y} = \overset{\text{VARIABLES}}{X}\,\overset{\text{WEIGHTS}}{\beta} + \overset{\text{NOISE}}{\epsilon}$$

$$\Delta\beta \propto -\frac{\partial(||Y - X\beta||_2)}{\partial\beta}$$

LINEAR REGRESSION

RESPONSE VARIABLES WEIGHTS NOISE

$$Y = X\beta + \epsilon$$

Constraining the values **beta** can take improves model performance:

REGULARIZATION

REGULARIZATION

* Regularization can be thought of in three ways:

  * **Prior**

  * **Penalty**

  * **Geometry**

REGULARIZATION AS PRIOR

RESPONSE VARIABLES WEIGHTS NOISE

$$Y = X\beta + \epsilon$$

$$Y_{t,j} \sim \mathcal{N}(X\beta, \sigma^2)$$

$$\hat\beta = \underset{\beta}{\mbox{argmax}}\ P(Y|X,\beta)$$

Y_{t,j} \sim \mathcal{N}(X\beta, \sigma^2)

\hat\beta = \underset{\beta}{\mbox{argmax }} P(Y|X,\beta)

REGULARIZATION AS PRIOR

RESPONSE VARIABLES WEIGHTS NOISE

$$Y = X\beta + \epsilon$$

$$Y_{t,j} \sim \mathcal{N}(X\beta, \sigma^2)$$

$$\hat\beta = \underset{\beta}{\mbox{argmax }} P(Y|X,\beta)P(\beta)$$

\hat\beta = \underset{\beta}
{\mbox{argmax }} P(Y|X,\bet

REGULARIZATION AS PENALTY

RESPONSE VARIABLES WEIGHTS NOISE

$$Y = X\beta + \epsilon$$

$$E(\beta) = ||Y - X\beta||_2^2$$

$$\hat{\beta} = \underset{\beta}{\mbox{argmin }} E(\beta)$$

E(\beta) = ||Y - X\beta||_2^2

\hat{\beta} = \underset{\beta}{\mbox{argmin }} E(\beta)

REGULARIZATION AS PENALTY

E_{pen}(\beta) = ||Y-X\beta||_2^2 + \lambda ||\beta||_2^2

\hat{\beta} = \underset{\beta}{\mbox{argmin }} E_{pen}(\beta)

$$\underbrace{Y}_{\text{RESPONSE}} = \underbrace{X}_{\text{VARIABLES}}\underbrace{\beta}_{\text{WEIGHTS}} + \underbrace{\epsilon}_{\text{NOISE}}$$

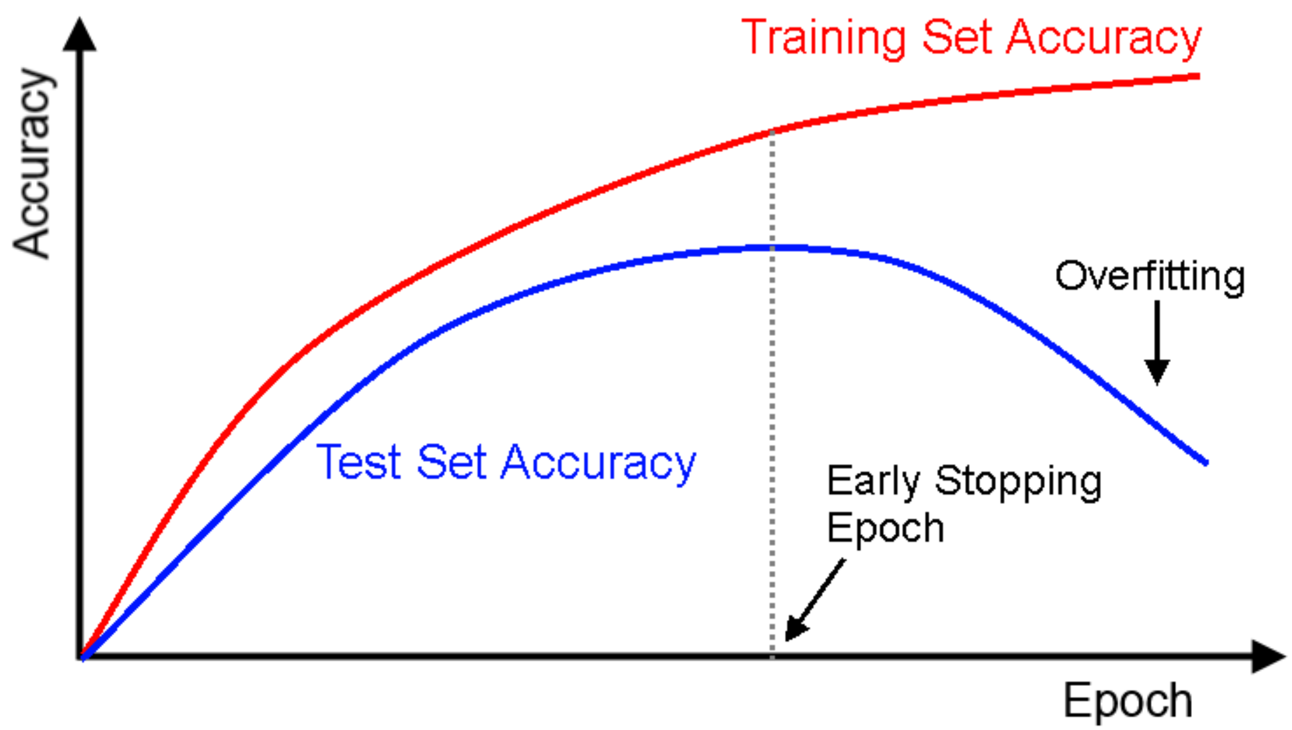
$$E_{pen}(\beta) = ||Y - X\beta||_2^2 + \lambda||\beta||_2^2$$

$$\hat{\beta} = \underset{\beta}{\mbox{argmin }} E_{pen}(\beta)$$

REGULARIZATION AS GEOMETRY - EARLY STOPPING

RESPONSE VARIABLES WEIGHTS NOISE

$$Y = X\beta + \epsilon$$



https://deeplearning4j.org/earlystopping

COMMON TYPES OF REGULARIZATION

* **Beta is small (L2-sense)** = ridge = gradient descent w/ early stopping

* **Beta is small, sparse (L1-sense)** = LASSO = coord. descent w/ early stopping

  * Beta is small & sparse (L1+L2 sense) = elastic net

* **Beta is sparse (L0-sense)** = variable selection

RIDGE REGRESSION

* Multivariate normal (MVN) prior on beta

* L2 penalty on beta

* Gradient descent w/ early stopping

RIDGE REGRESSI

\hat\beta = \underset{\beta}{\mbox{argmin}} \left[||Y-X\beta||_2^2 + \lambda ||\beta||_2^2 \right]

\hat\beta = \underset{\beta}

$$Y = X\beta + \epsilon$$

$$\hat{\beta} = \underset{\beta}{\mathrm{argmin}} \left[||Y - X\beta||_2^2 + \lambda ||\beta||_2^2 \right]$$

ERROR or LOSS **PENALTY**

RIDGE REGRESSION

$$\hat{\beta} = (X^\top X + \lambda I)^{-1} X^\top Y$$

$$\hat{\beta} = X^+_{ridge} Y$$

\hat\beta = (X^\top X + \lambda I)^{-1} X^\top Y

RIDGE REGRESSION

* Efficient solution with SVD

$$\hat{\beta} = (X^\top X + \lambda I)^{-1} X^\top Y$$

(SVD) $X = USV^\top$ $$D = \frac{S}{S^2 + \lambda^2}$$

$$\hat{\beta} = VDU^\top Y$$

\hat\beta = (X^\top X + \lambda I)^{-1} X^\top Y

D = \frac{S}{S^2 + \lambda^2}
\hat\beta = V D U^\top Y

RIDGE REGRESSION

* How to choose lambda?

* GCV - Generalized Cross Validation 👎

* Block-wise cross-validation 👍

RIDGE REGRESSION

* Good implementation: scikit-learn

* Awesome implementation:
  http://github.com/alexhuth/ridge

LASSO

* Laplacian prior on beta_i

* L1 penalty on beta

* Coordinate descent w/ early stopping

LASSO

\hat\beta = \underset{\beta}{\mbox{argmin}} \left[||Y-X\beta||_2^2 + \lambda ||\beta||_1 \right]

\hat\beta = \underset{\beta}

$$Y = X\beta + \epsilon$$

$$\hat\beta = \underset{\beta}{\mbox{argmin}} \left[||Y-X\beta||_2^2 + \lambda ||\beta||_1 \right]$$

ERROR or LOSS

PENALTY

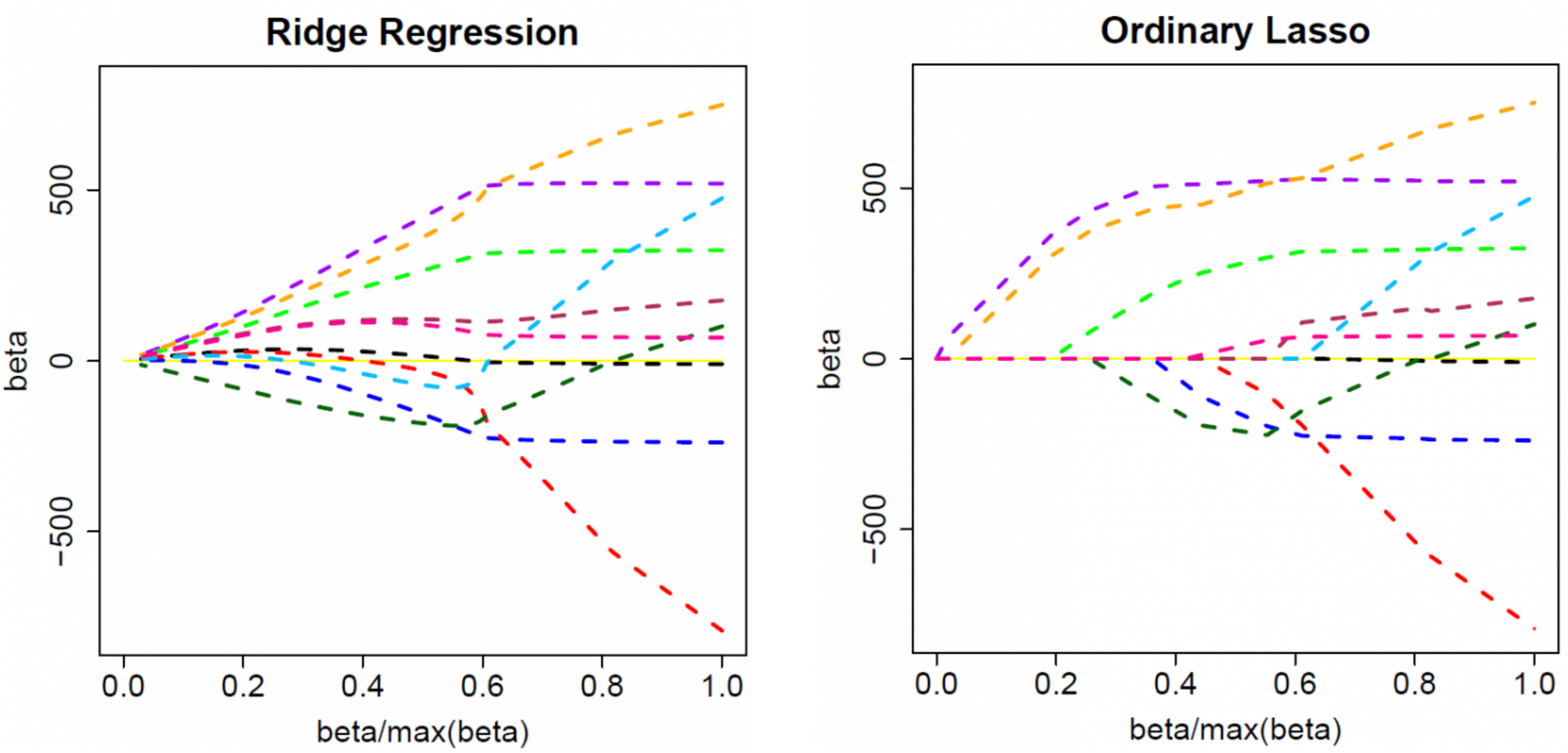
LASSO

* No closed form solution

* Solved via coordinate descent, LARS (least-angle regression) or other methods

* *Ssssslllllooooowwwww……………*

LASSO



h/t https://onlinecourses.science.psu.edu/stat857/node/158

OTHER METHODS

* Neural networks (linear or nonlinear)

* Random forests (nonlinear)

* Feature selection (~L0-norm)

NEXT TIME

* Tikhonov regression