Linear Regression

Aarti Singh (Instructor), HMW-Alexander (Noter)

February 1, 2017

Back to Index

Contents

Resources

- Lecture

---

1 Discrete to Continuous Labels

From classification to regression

1.1 Task

Given $X \in \mathcal{X}$, predict $Y \in \mathcal{Y}$, Construct prediction rule $f : \mathcal{X} \to \mathcal{Y}$

1.2 Performance Measure

- Quantifies knowledge gained.
- Measure of closeness between true label Y and prediction f(X)
  - 0/1 lose:$loss(Y, f(X)) = 1_{f(X) \neq Y}$. Risk: probability of error
  - square loss: $loss(Y, f(X)) = (f(X) - Y)^2$. Risk: mean square error
- How well does the predictor perform on average?

$$Risk\ R(f) = \mathbb{E}[loss(Y, f(X))],\ (X, Y) \sim P_{XY}$$

1.3 Bayes Optimal Rule

- ideal goal: Construct prediction rule $f^* : \mathcal{X} \to \mathcal{Y}$

$$f^* = \arg\min_f E_{XY}[loss(Y, f(X))]$$

  (Bayes optimal rule)

- Best possible performance:

$$\forall f,\ R(f^*) \leq R(f)$$

  (Bayes Risk)

Problem: P_{XY} is unknown.
Solution: Training data provides a glimpse of P_{XY}

$$\text{(observed) } \{(X_i, Y_i)\} \sim_{i.i.d} P_{XY} \text{ unknown}$$

2 Macine Learning Algortihm

- Model based approach: use data to learn a model for P_{XY}
- Model-free approach: use data to learn mapping directly

2.1 Empirical Risk Minimization (model-free)

- Optimal predictor:

$$f^* = \arg\min_f \mathbb{E}[(f(X) - Y)^2]$$

- Empirical Minimizer:

$$\hat{f}_n = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (f(X) - Y)^2$$

$\mathcal{F}$ is the class of predictors:

- Linear
- Polynomial
- Nonlinear

3 Linear Regression

$$f(\vec{X}) = \sum_{i=0}^{p} \beta_0 X^i = \vec{X}^T \vec{\beta},\ where\ X^0 = 1,\ \vec{\beta} = [\beta_0, \ldots, \beta_p]^T$$

$$\hat{\vec{\beta}} = \arg\min_{\vec{\beta}} (A^T\vec{\beta} - \vec{Y})^T (A^T\vec{\beta} - \vec{Y}),\ where\ A = [\vec{X}_1, \ldots, \vec{X}_n]$$

$$J(\beta) = (A^T\vec{\beta} - \vec{Y})^T (A^T\vec{\beta} - \vec{Y})$$

$$\begin{aligned}
\frac{\partial J(\vec{\beta})}{\partial \vec{\beta}} &= \frac{\partial (A^T\vec{\beta}-\vec{Y})^T (A^T\vec{\beta}-\vec{Y})}{\partial \vec{\beta}} \\
&= \frac{\partial (\vec{\beta}^T A A^T \vec{\beta} - \vec{\beta}^T A \vec{Y} - \vec{Y}^T A^T \vec{\beta} + \vec{Y}^T \vec{Y})}{\vec{\beta}} \\
&= (AA^T + (AA^T)^T)\vec{\beta} - A\vec{Y} - A\vec{Y} \\
&= 2AA^T\vec{\beta} - 2A\vec{Y} = 0 \\
&\Rightarrow AA^T\vec{\beta} = A\vec{Y} \\
&\Rightarrow \hat{\vec{\beta}} = (AA^T)^{-1} A\vec{Y},\ \text{if } AA^T \text{ is invertible}
\end{aligned}$$

3.1 Gradient Descent

Even when AA^T is invertible, might be computationally expensive if A is huge; however, $J(\vec{\beta})$ is convex[1] in β.

Minimum of a convex function can be reached by gradient descent algorithm:

- Initialize: pick $\vec{w}$ at random
- Gradient:
$$\nabla_{\vec{w}} l(\vec{w}) = [\frac{\partial l(\vec{w})}{\partial w_0}, \ldots, \frac{\partial l(\vec{w})}{\partial w_d}]^T$$
- Update rule:
$$\Delta \vec{w} = \eta \nabla_{\vec{w}} l(\vec{w})$$
,
$$w_i^{t+1} \leftarrow w_i^t - \eta \frac{\partial l(\vec{w})}{\partial w_i}|_t$$
- Stop: when some criterion met $\frac{\partial l(\vec{w})}{\partial w_i}|_t < \epsilon$

3.2 If AA^T is not invertible

$Rank(AA^T)$ = number of non-zero eigenvalues of AA^T = number of non-zero singular values of A $\leq \min(n, p)$ since A is $n \times p$

$$A = U\Sigma V^T \Rightarrow AA^T = U\Sigma^2 U^T \Rightarrow AA^T U = U\Sigma^2$$

3.2.1 Regularized Leasts Squares

Ridge Regression (L2 penalty)

$$\begin{aligned}
\hat{\vec{\beta}}_{MAP} &= \arg\min_{\vec{\beta}} (A^T\vec{\beta} - \vec{Y})^T (A^T\vec{\beta} - \vec{Y}) + \lambda \vec{\beta}^T \vec{\beta} \quad (\lambda \geq 0) \\
&= (AA^T + \lambda I)^{-1} A\vec{Y}
\end{aligned} \tag{1}$$

$(AA^T + \lambda I)$ is invertible if $\lambda > 0$. Proof:

- the symmetric matrix AA^T is positive-semidefinite matrix, because a matrix is positive-semidefinite iff it arises as the Gram matrix of some set of vectors[2].

[1] A function is called convex if the line joining any two points on the function does not go below the function on the interval formed by these two points.

[2] In contrast to the positive-definite case, these vectors need not be linearly independent.

- $\therefore \forall\lambda > 0 \ and \ \vec{x} \neq \vec{0}$,

$$\vec{x}^T(AA^T)\vec{x} = (A^T\vec{x})^T(A^T\vec{x}) \geq 0$$
$$\vec{x}^T(AA^T + \lambda I)\vec{x} = \vec{x}^T(AA^T)\vec{x} + \lambda\vec{x}^T\vec{x} > 0$$

- $\therefore$ $(AA^T + \lambda I)$ is positive definite.
- $\therefore$ the eigenvalues of $B = (AA^T + \lambda I)$ are all positive.

$$B\vec{v} = \lambda\vec{v} \Rightarrow \vec{v}^T B\vec{v} = \lambda > 0$$

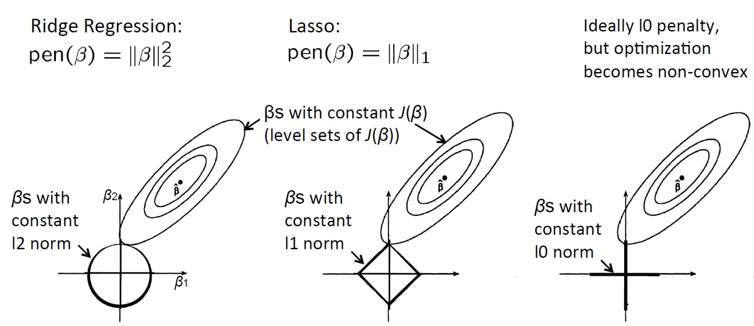
- $\therefore$ $(AA^T + \lambda I)$ is invertible if $\lambda > 0$

3.2.2 Understanding Regularized Least Squared

Why we need constraints: r equations, p unknowns - underdetermined system of linear equations.

$$\min_{\vec{\beta}} J(\beta) + \lambda pen(\vec{\lambda})$$

- Ridge Regression: $pen(\beta) = ||\beta||_2^2$
- Lasso Regression: $pen(\beta) = ||\beta||_1$. results in sparse solution - vector with more zero coordinates. Good for high-dimensional problems - don't have to store all coordinates, interpretable solution!



3.3 Regularized Least Squares - connection to MLE and MAP

- Least Squares and M(C)LE (maximum conditional LE)

$$Y = f^*(X) + \epsilon = X\beta^* + \epsilon$$
$$\epsilon \sim \mathcal{N}(0, \sigma^2 I) \ \ Y \sim \mathcal{N}(X\beta^*, \sigma^2 I)$$
$$\hat{\beta}_M LE = \arg\max$$

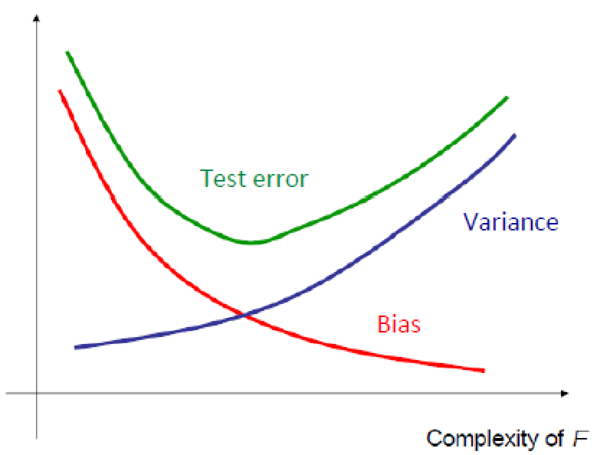
4 Polynomial Regression

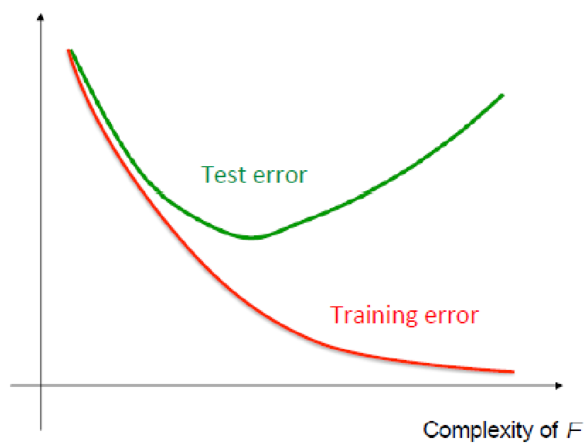
Univariate $f(X) = \sum \beta_i X^i$
Same with (Regular) Linear Regression:
$\hat{\beta} = (A^T A)^{-1} A^T Y$ or $(AA^T + \lambda I)^{-1} A^T Y$
Multivariate

4.1 Bias - Vairance Tradeoff

- Large bias, small variance: poor approximation but robust/stable
- Small bias, large variance: good approximation but unstable

Bias-Variance Decomposition: $E[(f(X))]$


Test error
Variance
Bias
Complexity of F


Test error
Training error
Complexity of F