

Introduction to Machine Learning

Extending Linear Regression

Varun Chandola

Computer Science & Engineering
State University of New York at Buffalo
Buffalo, NY, USA
chandola@buffalo.edu



University at Buffalo
Department of Computer Science
and Engineering
School of Engineering and Applied Sciences



Shortcomings of Linear Models

Handling Non-linear Relationships

- Handling Overfitting via Regularization

- Elastic Net Regularization

Handling Outliers in Regression

Issues with Linear Regression

1. Susceptible to outliers
2. *Too simplistic* - Underfitting
3. No way to control overfitting
4. Unstable in presence of correlated input attributes
5. Gets “confused” by unnecessary attributes

Biggest Issue with Linear Models

- ▶ They are linear!!
- ▶ Real-world is usually non-linear
- ▶ How do learn non-linear fits or non-linear decision boundaries?
 - ▶ Basis function expansion
 - ▶ Kernel methods (*will discuss this later*)

Handling Non-linear Relationships

- ▶ Replace \mathbf{x} with non-linear functions $\phi(\mathbf{x})$

$$y = \mathbf{w}^\top \phi(\mathbf{x})$$

- ▶ Model is still linear in \mathbf{w}
- ▶ Also known as **basis function expansion**

Example

$$\phi(x) = [1, x, x^2, \dots, x^p]$$

- ▶ Increasing p results in more complex fits

How to Control Overfitting?

- ▶ Use simpler models (linear instead of polynomial)
 - ▶ Might have poor results (underfitting)
- ▶ Use regularized complex models

$$\hat{\Theta} = \arg \min_{\Theta} J(\Theta) + \lambda R(\Theta)$$

- ▶ $R()$ corresponds to the penalty paid for complexity of the model

Ridge Regression

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} J(\mathbf{w}) + \lambda \|\mathbf{w}\|_2^2$$

- Helps in reducing impact of correlated inputs

Parameter Estimation for Ridge Regression

Exact Loss Function

$$\begin{aligned} J(\mathbf{w}) &= \frac{1}{2} \sum_{i=1}^N (y_i - \mathbf{w}^\top \mathbf{x}_i)^2 + \frac{1}{2} \lambda \|\mathbf{w}\|_2^2 \\ &= \frac{1}{2} (\mathbf{y} - \mathbf{X}\mathbf{w})^\top (\mathbf{y} - \mathbf{X}\mathbf{w}) + \frac{1}{2} \lambda \|\mathbf{w}\|_2^2 \end{aligned}$$

Ridge Estimate of \mathbf{w}

$$\hat{\mathbf{w}}_{\text{Ridge}} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}_D)^{-1} \mathbf{X}^\top \mathbf{y}$$

Using Gradient Descent with Ridge Regression

- ▶ Very similar to OLE
- ▶ Minimize the squared loss using *Gradient Descent*

$$J(\mathbf{w}) = \frac{1}{2}(\mathbf{y} - \mathbf{X}\mathbf{w})^\top(\mathbf{y} - \mathbf{X}\mathbf{w}) + \frac{1}{2}\lambda\|\mathbf{w}\|_2^2$$

$$\begin{aligned}\nabla J(\mathbf{w}) = \frac{d}{d\mathbf{w}}J(\mathbf{w}) &= \frac{1}{2}\frac{d}{d\mathbf{w}}(\mathbf{y} - \mathbf{X}\mathbf{w})^\top(\mathbf{y} - \mathbf{X}\mathbf{w}) + \frac{1}{2}\lambda\frac{d}{d\mathbf{w}}\|\mathbf{w}\|_2^2 \\ &= \mathbf{X}^\top\mathbf{X}\mathbf{w} - \mathbf{X}^\top\mathbf{y} + \lambda\mathbf{w}\end{aligned}$$

Using the above result, one can perform repeated updates of the weights:

$$\mathbf{w} := \mathbf{w} - \eta\nabla J(\mathbf{w})$$

Least Absolute Shrinkage and Selection Operator - LASSO

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} J(\mathbf{w}) + \lambda |\mathbf{w}|$$

- ▶ Helps in feature selection – favors sparse solutions
- ▶ Optimization is not as straightforward as in Ridge regression
 - ▶ Gradient not defined for $w_i = 0, \forall i$

LASSO vs. Ridge

- ▶ Both control overfitting
- ▶ Ridge helps reduce impact of correlated inputs, LASSO helps in feature selection
- ▶ Rule of thumb
 - ▶ If data has many features but only few are potentially useful, use LASSO
 - ▶ If data has potentially many correlated features, use Ridge

LASSO vs. Ridge

- ▶ Both control overfitting
- ▶ Ridge helps reduce impact of correlated inputs, LASSO helps in feature selection
- ▶ Rule of thumb
 - ▶ If data has many features but only few are potentially useful, use LASSO
 - ▶ If data has potentially many correlated features, use Ridge

Elastic Net Regularization

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} J(\mathbf{w}) + \lambda_1 |\mathbf{w}| + \lambda_2 \|\mathbf{w}\|_2^2$$

- ▶ The best of both worlds
- ▶ Again, optimizing for \mathbf{w} is not straightforward

Impact of outliers on regression

- ▶ Linear regression training gets impacted by the presence of outliers
- ▶ The square term in loss function is the culprit
- ▶ How to handle this (*Robust Regression*)?
 - ▶ *Least absolute deviations* instead of least squares

$$J(\mathbf{w}) = \sum_{i=1}^N |y_i - \mathbf{w}^T \mathbf{x}|$$

References