Machine Learning and Photonics

Week 5

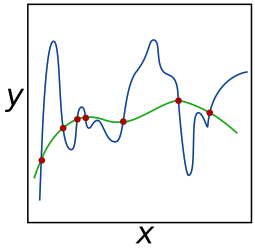
Ergun Simsek

University of Maryland Baltimore County
simsek@umbc.edu

Regularization

February 28, 2023

1. Brief Review of Last Week
2. Regression with Regularization
   - What's Regularization
   - Ridge Regularization
   - Lasso Regularization
   - Elastic Net Regularization



Announcements

TA: Arushi Agarwal
Time: Thursday 2 pm to 4 pm
Webex Room: https://umbc.webex.com/meet/arushia1
E-mail: arushia1@umbc.edu

An advanced reference about regularization methods:
https://arxiv.org/pdf/1509.09169

Remember the Linear Regression from Week#4?

Assume a dataset in which the features (d of them) are denoted by $\vec{x} \in \mathbb{R}^d$, and y denotes the target variable.
The dataset consists of n observations, and both $\vec{x}$ and y are continuous variables.

Observation	$\vec{x} \in \mathbb{R}^d$	y
1	$\vec{x}^{(1)}$	y_1
2	$\vec{x}^{(2)}$	y_2
3	$\vec{x}^{(3)}$	y_3
$\vdots$	$\vdots$	$\vdots$
n	$\vec{x}^{(n)}$	y_n

Remember the Linear Regression from Week#4?

$$\mathbb{X} = (\mathbb{1}, x_1, x_2, \cdots, x_d) = \begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_d^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \cdots & x_d^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_1^{(n)} & x_2^{(n)} & \cdots & x_d^{(n)} \end{pmatrix} . \tag{1}$$

$$\hat{\mathbb{Y}} = \mathbb{X} \cdot \omega^{\mathsf{T}} , \tag{2}$$

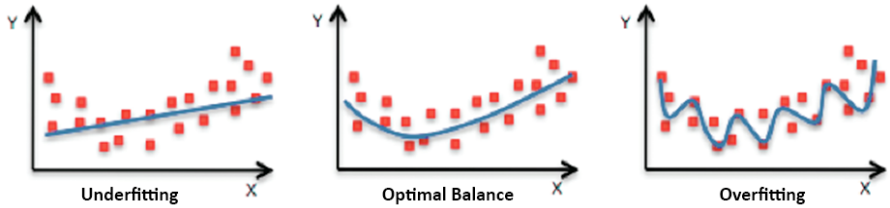
$$J(\omega) = \frac{1}{2}(\mathbb{Y} - \hat{\mathbb{Y}})^{\mathsf{T}} \cdot (\mathbb{Y} - \hat{\mathbb{Y}}) , \tag{3}$$

$$\omega^{\mathsf{T}} = \left(\mathbb{X}^{\mathsf{T}}\mathbb{X}\right)^{-1} \mathbb{X}^{\mathsf{T}}\mathbb{Y} . \tag{4}$$

Assessing the Accuracy of the Model

- Residual Standard Error (RSE)
- R^2
- F-statistics
- Pearson Correlation
- Spearman Rank Correlation Coefficient
- Kendall correlation

Motivation

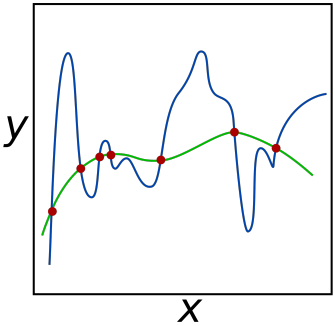


1. Errors that are due to *high bias* often lead to an *underfit* of the training data.
2. Errors caused by *high variance* often lead to an *overfit* of the training data.
3. An irreducible error is caused by the noise in the data itself. The only way to reduce this type of error is to clean up the data.

Today, we focus on errors due to high variance.

What's Regularization?

- Regularization is a method to reduce degrees of freedom of a model to avoid overfitting the data.
- We will introduce regularization in the context of regression analysis. However, regularization methods work for classification algorithms as well.



Why do we need regularization?

- Our linear model had $d + 1$ **degrees of freedom**.
- If one uses polynomial regression, then one would have even more degrees of freedom.
  - If one uses quadratic regression, one would find $\frac{d(d+1)}{2}$ more degrees of freedom (in addition to $d + 1$ linear degrees of freedom).
- *The greater degrees of freedom a model possesses, the easier it will overfit the data*.
- The fewer degrees of freedom a model has, the harder it will be for it to overfit the data.
- To avoid overfitting, one has to reduce the degrees of freedom of the model.
- **Regularization is a method to reduce degrees of freedom of a model to avoid overfitting the data**.

Important Note

- We *do not eliminate any of the features by hand*.
- We rather *constrain the model*.
- The model will then learn through the process of training to identify irrelevant (or less relevant) features.

Regularization Method

- Here, we consider three common types of regularizations, namely the
  1. *Ridge*,
  2. *Lasso*, and
  3. *Elastic Net* regressions.
- In these regularization methods, a constraint on the weights ω (*i.e.* parameters of the model) is imposed.
- The cost function is then minimized in the presence of the introduced constraint.

Parameters and *Hyperparameters* of ML Models

- **Parameter:** A factor that is introduced by the model, and its optimal value is determined through the training process once the training is complete.
  - Learned by through the training process.
  - e.g. Coefficients of linear regression ω are parameters of the linear model, and their optimal values are determined by minimizing the cost function through the training process.
- **Hyperparameter:** Set by a user before the training process starts.
  - Not learned by the model, and their values are are fixed before training starts.
  - Without specifying fixed values for hyperparameters, the training process cannot start.
  - e.g., *train-test split ratio*.

Contraints

- The constraint is **independent of the training data**.
- The constraint involves a **positive definite term** that only depends on the magnitude of the weights ω.
- The constraint puts an **upper bound on the magnitude of the parameters** of the model (in this case the coefficients of linear model ω).
- **Regularization factor** or **penalty factor**: λ is the positive semi-definite hyperparameter that we use to incorporate and control the positive definite constraint in the minimization of the cost function
  - The case $\lambda = 0$ corresponds to the unconstrained linear model we studied last time.
  - The greater λ is, the stronger the penalty term will be.
  - A stronger penalty term forces the coefficients ω to be smaller in magnitude.

Ridge Regularization (a.k.a. Tikhonov Regularization)

- Based on the L_2-norm.
- The weight vector: $\tilde{\omega} = (\omega_1, \cdots, \omega_d)$
  - Note that ω_0 is not included!

$$\begin{aligned} &\underset{\omega}{\text{argmin}} \Big\{ J(\omega) = \frac{1}{2} (\mathbb{Y} - \hat{\mathbb{Y}})^{\mathsf{T}} \cdot (\mathbb{Y} - \hat{\mathbb{Y}}) \Big\} \\ &\text{subject to: } (\|\tilde{\omega}\|_2)^2 = \sum_{i=1}^{d} \omega_i^2 \leq s^2 \, . \end{aligned} \tag{5}$$

Ridge Regularization (a.k.a. Tikhonov Regularization)

- Based on the L_2-norm.
- The weight vector: $\tilde{\omega} = (\omega_1, \cdots, \omega_d)$
  - Note that ω_0 is not included!

$$\begin{aligned} &\underset{\omega}{\text{argmin}} \Big\{ J(\omega) = \frac{1}{2} (\mathbb{Y} - \hat{\mathbb{Y}})^{\mathsf{T}} \cdot (\mathbb{Y} - \hat{\mathbb{Y}}) \Big\} \\ &\text{subject to: } (\|\tilde{\omega}\|_2)^2 = \sum_{i=1}^{d} \omega_i^2 \leq s^2 \, . \end{aligned} \tag{5}$$

Question: How can we implement this?

Ridge Regularization (a.k.a. Tikhonov Regularization)

Trick: Lagrange Multiplier Method

Instead of minimizing the cost function $J(\omega)$ in (5),
we minimize $\tilde{J}(\omega)$ in the following equation

$$\tilde{J}(\omega) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2 + \frac{\lambda}{2}\sum_{i=1}^{d}\omega_i^2 = \frac{1}{2}(\mathbb{Y} - \hat{\mathbb{Y}})^{\mathsf{T}} \cdot (\mathbb{Y} - \hat{\mathbb{Y}}) + \frac{\lambda}{2}\,\tilde{\omega}\,\tilde{\omega}^{\mathsf{T}}\,. \tag{6}$$

No more constraints! Just minimize the cost function!

Ridge Regularization (a.k.a. Tikhonov Regularization)

Trick: Lagrange Multiplier Method

Instead of minimizing the cost function $J(\omega)$ in (5),
we minimize $\tilde{J}(\omega)$ in the following equation

$$\tilde{J}(\omega) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2 + \frac{\lambda}{2}\sum_{i=1}^{d}\omega_i^2 = \frac{1}{2}(\mathbb{Y} - \hat{\mathbb{Y}})^{\mathsf{T}} \cdot (\mathbb{Y} - \hat{\mathbb{Y}}) + \frac{\lambda}{2}\,\tilde{\omega}\,\tilde{\omega}^{\mathsf{T}}\,. \tag{6}$$

No more constraints! Just minimize the cost function!

- If we set $\lambda = 0$, $\tilde{J}(\omega)$ reduces to the cost function $J(\omega)$, and we have the ordinary regression
- The greater the penalty factor λ is, the greater the effect of regularization will be.

Ridge Regression: Exact Solution

Question: How does regularization treat weakly correlated features?

Ridge Regression: Exact Solution

Question: How does regularization treat weakly correlated features?
Side Question: We saw that it was possible to solve the regression problem exactly. Is it still possible to obtain an exact solution for the Ridge regression when a penalty term is present?

Ridge Regression: Exact Solution

Question: How does regularization treat weakly correlated features?
Side Question: We saw that it was possible to solve the regression problem exactly. Is it still possible to obtain an exact solution for the Ridge regression when a penalty term is present?

$$\tilde{\mathbb{X}} = (x_1 - \bar{x}_1, \cdots, x_d - \bar{x}_d) = \begin{pmatrix} x_1^{(1)} - \bar{x}_1 & x_2^{(1)} - \bar{x}_2 & \cdots & x_d^{(1)} - \bar{x}_d \\ x_1^{(2)} - \bar{x}_1 & x_2^{(2)} - \bar{x}_2 & \cdots & x_d^{(2)} - \bar{x}_d \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{(n)} - \bar{x}_1 & x_2^{(n)} - \bar{x}_2 & \cdots & x_d^{(n)} - \bar{x}_d \end{pmatrix} . \tag{7}$$

$$\boxed{\omega_0 = \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y^{(i)} , \quad \tilde{\omega}^{\mathsf{T}} = \left(\tilde{\mathbb{X}}^{\mathsf{T}}\tilde{\mathbb{X}} + \lambda \mathbb{1}_{d\times d}\right)^{-1} \tilde{\mathbb{X}}^{\mathsf{T}}\mathbb{Y} ,} \tag{8}$$

Ridge Regression: Exact Solution

$$\omega_0 = \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y^{(i)}\,, \quad \tilde{\omega}^{\mathsf{T}} = \left(\tilde{\mathbb{X}}^{\mathsf{T}}\tilde{\mathbb{X}} + \lambda \mathbb{1}_{d\times d}\right)^{-1}\tilde{\mathbb{X}}^{\mathsf{T}}\mathbb{Y}\,, \tag{8}$$

- The bias term ω_0 is nothing but the *mean of the target variable* y.
- The greater λ is, the smaller $\tilde{\omega}$ will be.
- IOW, $\tilde{\omega} \to \vec{0}$ as $\lambda \to \infty$.

Ridge Regression: Exact Solution

$$\omega_0 = \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y^{(i)}\,, \quad \tilde{\omega}^{\mathsf{T}} = \left(\tilde{\mathbb{X}}^{\mathsf{T}}\tilde{\mathbb{X}} + \lambda \mathbb{1}_{d\times d}\right)^{-1}\tilde{\mathbb{X}}^{\mathsf{T}}\mathbb{Y}\,, \tag{8}$$

- The bias term ω_0 is nothing but the *mean of the target variable* y.
- The greater λ is, the smaller $\tilde{\omega}$ will be.
- IOW, $\tilde{\omega} \to \vec{0}$ as $\lambda \to \infty$.

Question: Does scikit-learn use the exact formula (8) to calculate the weights of the Ridge regression?

Lasso Regularization

LASSO: Least Absolute Shrinkage and Selection Operator

$$\begin{aligned} &\underset{\omega}{\operatorname{argmin}} \left\{ J(\omega) = \frac{1}{2} (\mathbb{Y} - \hat{\mathbb{Y}})^{\mathsf{T}} \cdot (\mathbb{Y} - \hat{\mathbb{Y}}) \right\} \\ &\text{subject to: } \|\tilde{\omega}\|_1 = \sum_{i=1}^{d} |\omega_i| \leq s \,. \end{aligned} \tag{9}$$

Lasso Regularization

LASSO: Least Absolute Shrinkage and Selection Operator

$$\begin{aligned} &\underset{\omega}{\text{argmin}}\Big\{J(\omega) = \frac{1}{2}(\mathbb{Y} - \hat{\mathbb{Y}})^{\mathsf{T}} \cdot (\mathbb{Y} - \hat{\mathbb{Y}})\Big\} \\ &\text{subject to: } \|\tilde{\omega}\|_1 = \sum_{i=1}^{d} |\omega_i| \leq s \,. \end{aligned} \tag{9}$$

- Based on the L_1-norm.
- Deals with the absolute value of the weights
- Again, the bias term ω_0 is never regularized.
- Again, by employing the Lagrange multiplier method, we can obtain the cost function

$$\tilde{J}(\omega) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2 + \frac{\lambda}{2}\sum_{i=1}^{d}|\omega_i| \,. \tag{10}$$

Ridge vs. Lasso Regression

- Ridge regression shrinks all the weights toward smaller values as λ increases
  - But, it will not set any of the weights ω_i exactly to zero (unless you set $\lambda \to \infty$).
  - This can pose a challenge in interpreting the model in which the number of features is quite large.
- Lasso regularization has the effect of forcing some of the weights ω_i to be exactly zero when the penalty factor λ is sufficiently large.
- *Lasso regularization can be used as a feature selection criterion*, and hence, it offers a simpler model to interpret!

Ridge vs. Lasso Regression

How does the Lasso regularization set some of the weights *exactly* to zero?

Suppose we have a model that possesses only two features x_1 and x_2, and a target variable y.

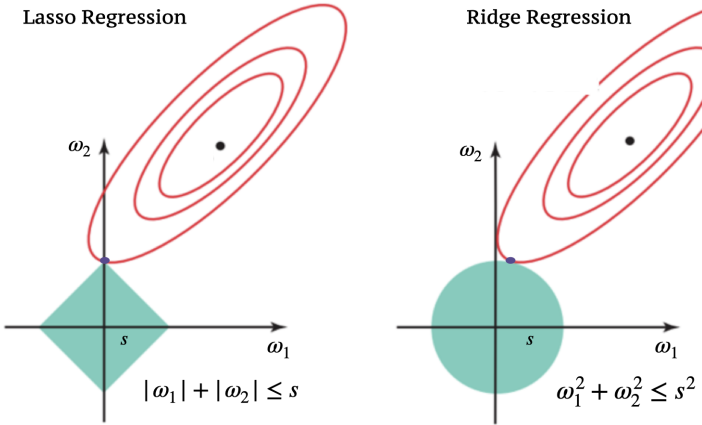
In this case, $\tilde{\omega} = (\omega_1, \omega_2)$, and the cost function $J(\omega)$ is given by

$$\begin{aligned} J(\omega_0, \omega_1, \omega_2) &= \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2 = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \omega_1 x_1^{(i)} - \omega_2 x_2^{(i)} - \omega_0)^2 \\ &= A\,\omega_1^2 + B\,\omega_2^2 + C\,\omega_1\omega_2 + D\,\omega_1 + E\,\omega_2 + F\,, \end{aligned} \tag{13}$$

Note that this is an elliptical equation!

Ridge vs. Lasso Regression

Minimizing the cost function = Finding the intersection of the contour plots (*i.e.* the red ellipses) with the regions corresponding to the constraints.



Elastic Net Regularization

$$\underset{\omega}{\text{argmin}}\Big\{J(\omega) = \frac{1}{2}(\mathbb{Y} - \hat{\mathbb{Y}})^{\mathsf{T}} \cdot (\mathbb{Y} - \hat{\mathbb{Y}})\Big\}$$

$$\text{subject to: } \begin{cases} \|\tilde{\omega}\|_1 = \sum_{i=1}^{d} |\omega_i| \leq s_1 \\ (\|\tilde{\omega}\|_2)^2 = \sum_{i=1}^{d} \omega_i^2 \leq s_2 \end{cases} \tag{11}$$

By employing the same Lagrange multiplier technique, the cost function

$$\tilde{J}(\omega) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2 + \frac{\lambda_1}{2}\sum_{i=1}^{d}\omega_i^2 + \frac{\lambda_2}{2}\sum_{i=1}^{d}|\omega_i| \; . \tag{12}$$

Elastic Net Regularization

$$\tilde{J}(\omega) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2 + \frac{\lambda_1}{2}\sum_{i=1}^{d}\omega_i^2 + \frac{\lambda_2}{2}\sum_{i=1}^{d}|\omega_i| \,. \tag{12}$$

- Here, we have two penalty terms!
- Setting $\lambda_1 = 0$ reduces the Elastic Net regression to the Lasso regression
- Setting $\lambda_2 = 0$ reduces the Elastic Net regression to the Ridge regression

Final Notes

- Regularization techniques can be applied to classification problems as well.
- If your dataset is large and if you have very few features, you don't need regularization.
- If your dataset is small and if you have lots of features, then the regularization methods are your best friends.