

Machine Learning

Lec 5 and 6:

Regularization: LASSO and Ridge

Dr. Wazir Ali

Assistant Professor

Department of Artificial Intelligence and MMG

Aror Univeristy, Sukkur

Loss Functions..

1. Mean Absolute Error (MAE)

This is also known as the L1 loss

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|$$

This loss function is easy to compute and measures the absolute difference between the true and predicted value. It is not sensitive to outliers and it is also not differentiable at zero.

Loss Functions

Mean Squared Error (MSE)

This is also known as the L2 loss

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

This loss function handles outliers in an efficient manner as outliers are detected due to the quadratic loss. Convergence is also smooth as the gradient becomes smaller as the loss decreases.

Cross-validation

- primarily used in applied machine learning to estimate the skill of a machine learning model on unseen data.
- It is a popular method because it is simple to understand and because it generally results in a less biased or less optimistic estimate of the model skill than other methods, such as a simple train/test split.
- Note that k-fold cross-validation is to evaluate the model design, not a particular training. Because you re-trained the model of the same design with different training sets.

Procedure

1. Shuffle the dataset randomly.
2. Split the dataset into k groups
3. For each unique group:
 1. Take the group as a hold out or test data set
 2. Take the remaining groups as a training data set
 3. Fit a model on the training set and evaluate it on the test set
 4. Retain the evaluation score and discard the model
4. Summarize the skill of the model using the sample of model evaluation scores

Importantly, each observation in the data sample is assigned to an individual group and stays in that group for the duration of the procedure. This means that each sample is given the opportunity to be used in the hold out set 1 time and used to train the model $k-1$ times.

Reducing overfitting: Regularization

Regularization: An Overview

The idea of regularization revolves around modifying the loss function L ; in particular, we add a regularization term that penalizes some specified properties of the model parameters

$$L_{reg}(\beta) = L(\beta) + \lambda R(\beta),$$

where λ is a scalar that gives the weight (or importance) of the regularization term.

Fitting the model using the modified loss function L_{reg} would result in model parameters with desirable properties (specified by R).

LASSO Regression

Since we wish to discourage extreme values in model parameter, we need to choose a regularization term that penalizes parameter magnitudes. For loss function, we use MSE.

Together our regularized loss function is:

$$L_{LASSO}(\beta) = \frac{1}{n} \sum_{i=1}^n |y_i - \beta^\top \mathbf{x}_i|^2 + \lambda \sum_{j=1}^J |\beta_j|.$$

Note that $\sum_{j=1}^J |\beta_j|$ is the l_1 norm of the vector β

$$\sum_{j=1}^J |\beta_j| = \|\beta\|_1$$

Ridge Regression

Alternatively, we can choose a regularization term that penalizes the squares of the parameter magnitudes. Then, our regularized loss function is:

$$L_{Ridge}(\beta) = \frac{1}{n} \sum_{i=1}^n |y_i - \beta^\top \mathbf{x}_i|^2 + \lambda \sum_{j=1}^J \beta_j^2.$$

Note that $\sum_{j=1}^J |\beta_j|^2$ is the square of the l_2 norm of the vector β

$$\sum_{j=1}^J \beta_j^2 = \|\beta\|_2^2$$

Comparision

Ridge Regression: The penalty term adds a circular constraint around the origin in the coefficient space. This typically leads to coefficients being shrunk towards zero but not exactly to zero.

Lasso Regression: The penalty term adds a diamond-shaped constraint in the coefficient space. This can lead to some coefficients being exactly zero, effectively performing feature selection.

Choosing λ

In both ridge and LASSO regression, we see that the larger our choice of the **regularization parameter** λ , the more heavily we penalize large values in β ,

- If λ is close to zero, we recover the MSE, i.e. ridge and LASSO regression is just ordinary regression.
- If λ is sufficiently large, the MSE term in the regularized loss function will be insignificant and the regularization term will force β_{ridge} and β_{LASSO} to be close to zero.

To avoid ad-hoc choices, we should select λ using cross-validation.

The λ controls the amount of regularization applied to the model. A non-negative value represents a shrinkage parameter, which multiplies $P(\alpha, \beta)$ in the objective.

Ridge, LASSO - Computational complexity

Solution to ridge regression:

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

The solution to the LASSO regression:

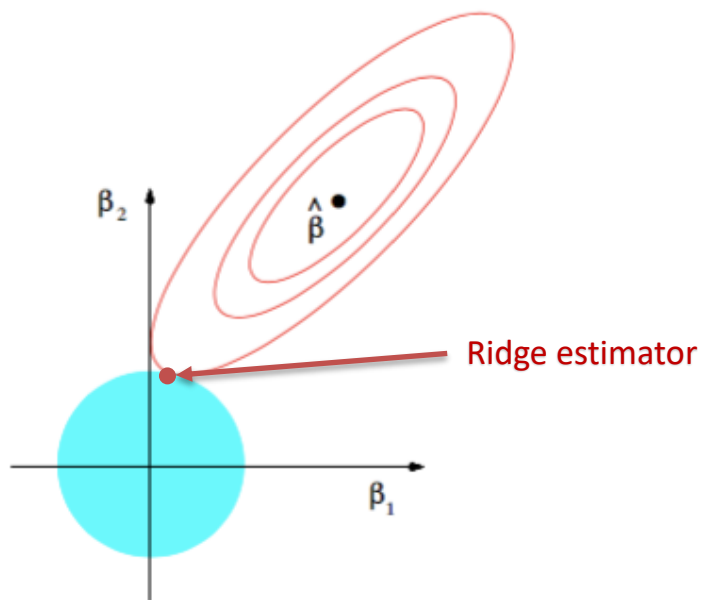
LASSO has no conventional analytical solution, as the L1 norm has no derivative at 0. We can, however, use the concept of **subdifferential** or **subgradient** to find a manageable expression.

Regularization Parameter with a Validation Set

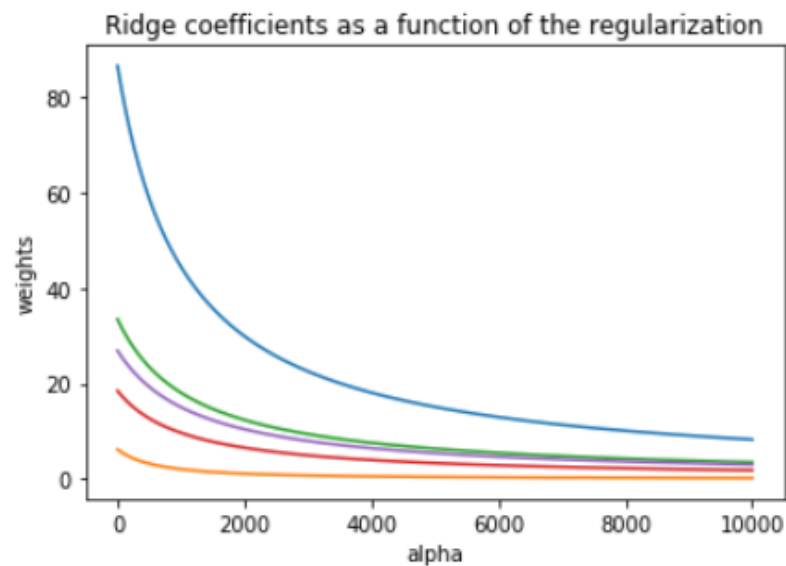
The solution of the Ridge/Lasso regression involves three steps:

- Select λ
- Find the minimum of the ridge/Lasso regression loss function (using the formula for ridge) and record the *MSE on the validation set*.
- Find the λ that gives the smallest *MSE*

Ridge visualized

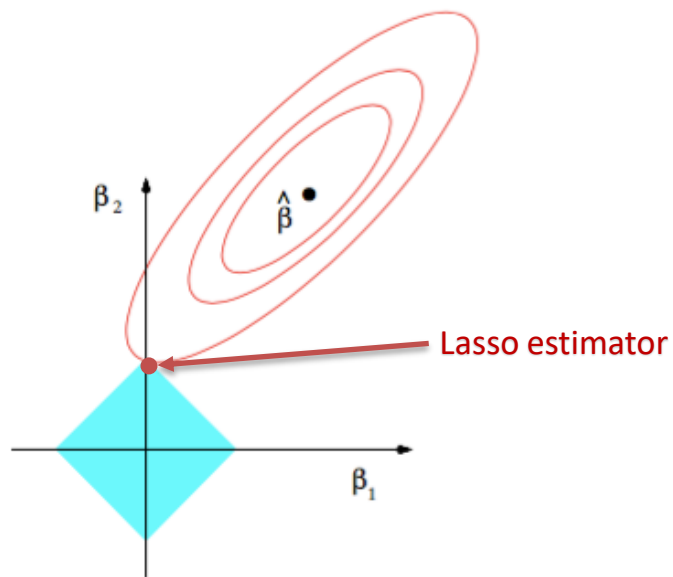


The ridge estimator is where the constraint and the loss intersect.

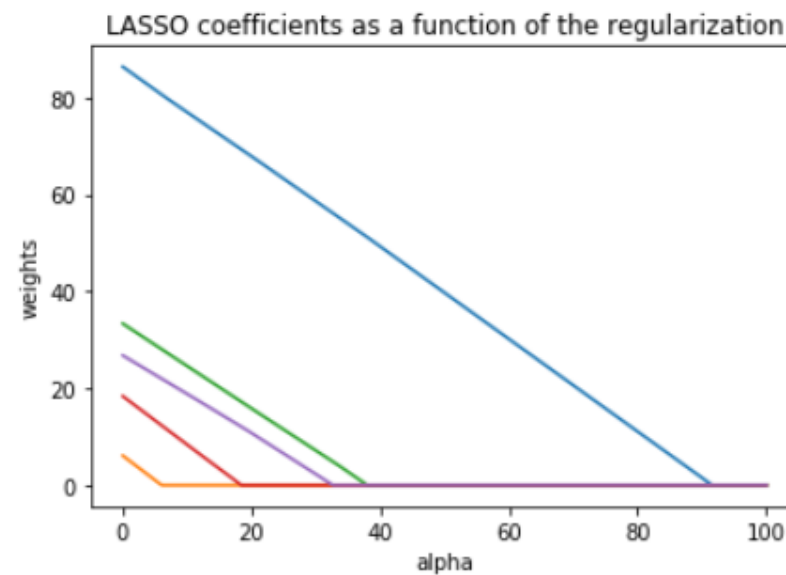


The values of the coefficients decrease as lambda increases, but they are not nullified.

LASSO visualized



The Lasso estimator tends to zero out parameters as the OLS loss can easily intersect with the constraint on one of the axes.

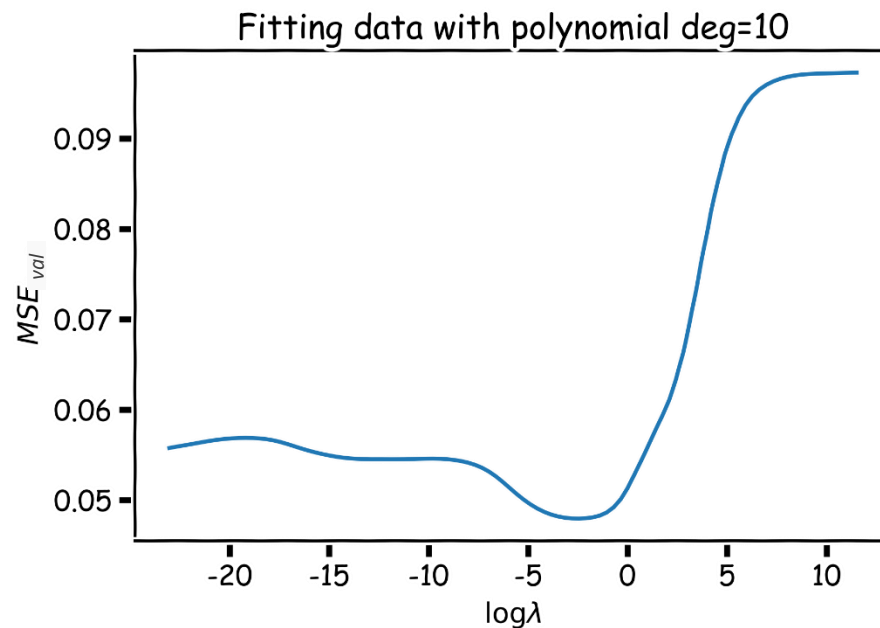


The values of the coefficients decrease as lambda increases, and are nullified fast.

Ridge regularization with only **validation** : step by step

1. split data into $\{\{X, Y\}_{train}, \{X, Y\}_{validation}, \{X, Y\}_{test}\}$
2. for λ in $\{\lambda_{min}, \dots, \lambda_{max}\}$:
 1. determine the β that minimizes the L_{ridge} , $\hat{\beta}_{Ridge}(\lambda) = (X^T X + \lambda I)^{-1} X^T Y$, using the train data.
 2. record $L_{MSE}(\lambda)$ using validation data.
3. select the λ that minimizes the loss on the validation data,
$$\lambda_{ridge} = \operatorname{argmin}_{\lambda} L_{MSE}(\lambda)$$
4. Refit the model using both train and validation data, $\{\{X, Y\}_{train}, \{X, Y\}_{validation}\}$, resulting to $\hat{\beta}_{ridge}(\lambda_{ridge})$
5. report MSE or R^2 on $\{X, Y\}_{test}$ given the $\hat{\beta}_{ridge}(\lambda_{ridge})$

Ridge regularization with **validation** only: step by step



Lasso regularization with validation only: step by step

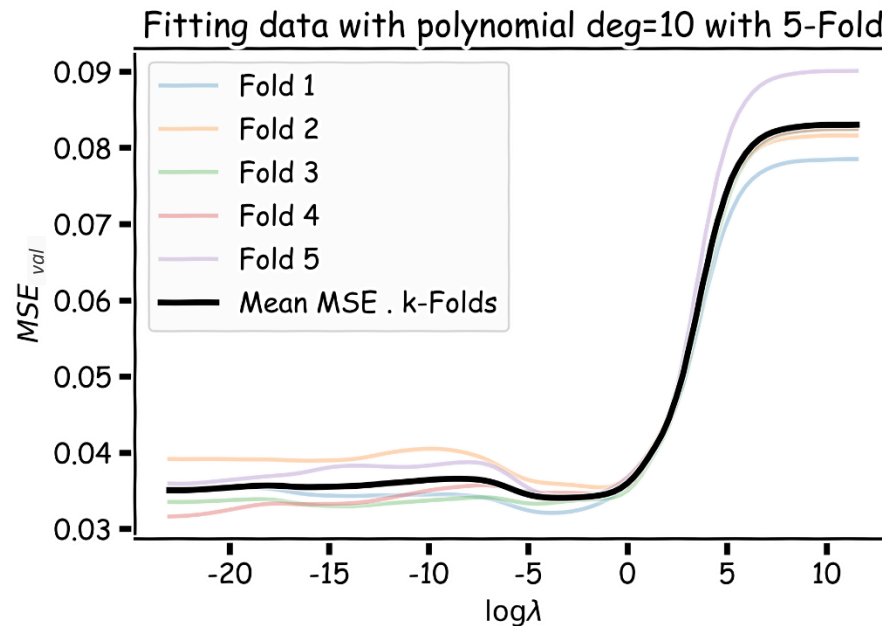
1. split data into $\{\{X, Y\}_{train}, \{X, Y\}_{validation}, \{X, Y\}_{test}\}$
2. for λ in $\{\lambda_{min}, \dots, \lambda_{max}\}$:
 - A. determine the β that minimizes the L_{lasso} , $\hat{\beta}_{lasso}(\lambda)$, using the train data. **This is done using a solver.**
 - B. record $L_{MSE}(\lambda)$ using validation data
3. select the λ that minimizes the loss on the validation data,
 $\lambda_{lasso} = \operatorname{argmin}_{\lambda} L_{MSE}(\lambda)$
4. Refit the model using both train and validation data,
 $\{\{X, Y\}_{train}, \{X, Y\}_{validation}\}$, resulting to $\hat{\beta}_{lasso}(\lambda_{lasso})$
5. report MSE or R^2 on $\{X, Y\}_{test}$ given the $\hat{\beta}_{lasso}(\lambda_{lasso})$

Ridge regularization with CV: step by step

1. remove $\{X, Y\}_{test}$ from data
2. split the rest of data into K folds, $\{\{X, Y\}_{train}^{-k}, \{X, Y\}_{val}^k\}$
3. for k in $\{1, \dots, K\}$
 1. for λ in $\{\lambda_0, \dots, \lambda_n\}$:
 - A. determine the β that minimizes the L_{ridge} , $\hat{\beta}_{ridge}(\lambda, k) = (X^T X + \lambda I)^{-1} X^T Y$, using the train data of the fold, $\{X, Y\}_{train}^{-k}$.
 - B. record $L_{MSE}(\lambda, k)$ using the validation data of the fold $\{X, Y\}_{val}^k$
At this point we have a 2-D matrix, rows are for different k , and columns are for different λ values.
4. Average the $L_{MSE}(\lambda, k)$ for each λ , $\bar{L}_{MSE}(\lambda)$.
5. Find the λ that minimizes the $\bar{L}_{MSE}(\lambda)$, resulting to λ_{ridge} .
6. Refit the model using the full training data, $\{\{X, Y\}_{train}, \{X, Y\}_{val}\}$, resulting to $\hat{\beta}_{ridge}(\lambda_{ridge})$
7. report MSE or R^2 on $\{X, Y\}_{test}$ given the $\hat{\beta}_{ridge}(\lambda_{ridge})$

	λ_1	λ_2	...	λ_n
k_1	L_{11}	L_{12}
k_2	L_{21}
...
k_n
E[]	\bar{L}_1	\bar{L}_2	...	\bar{L}_n

Ridge regularization with **validation** only: step by step



Variable Selection as Regularization

Since LASSO regression tend to produce zero estimates for a number of model parameters - we say that LASSO solutions are **sparse** - we consider LASSO to be a method for variable selection.

Many prefer using LASSO for variable selection (as well as for suppressing extreme parameter values) rather than stepwise selection, as LASSO avoids the statistic problems that arises in stepwise selection.

Question: What are the pros and cons of the two approaches?