Machine Learning (ML) with Python

Regularization

Dr. Aeshah Alsughayyir

Collage of Computer Science and Engineering

Taibah University

2021-2022

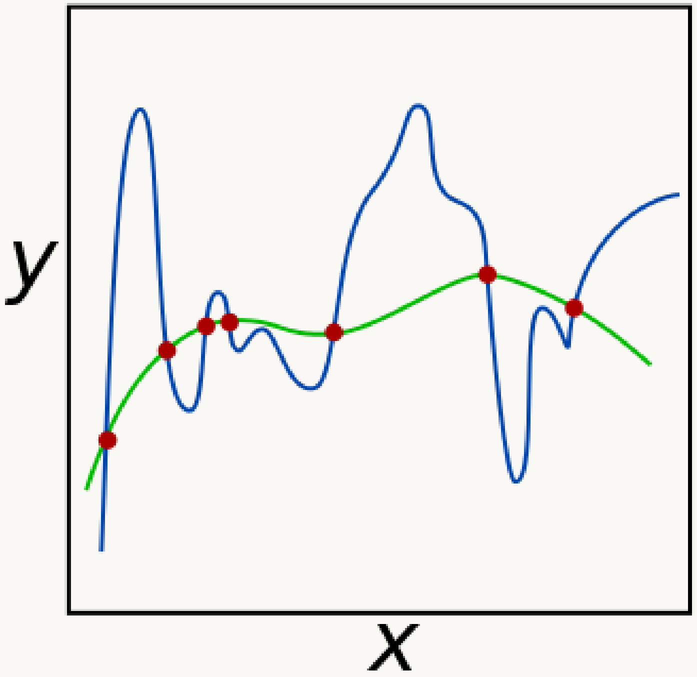
Outline:

- Regularization
    - Overfitting
    - How to Avoid Overfitting?
    - Understanding Regularization
    - Regularization Techniques of Linear Model
        - Ridge Regression
        - Lasso Regression
    - Impact of Regularization
    - Example (with Python)

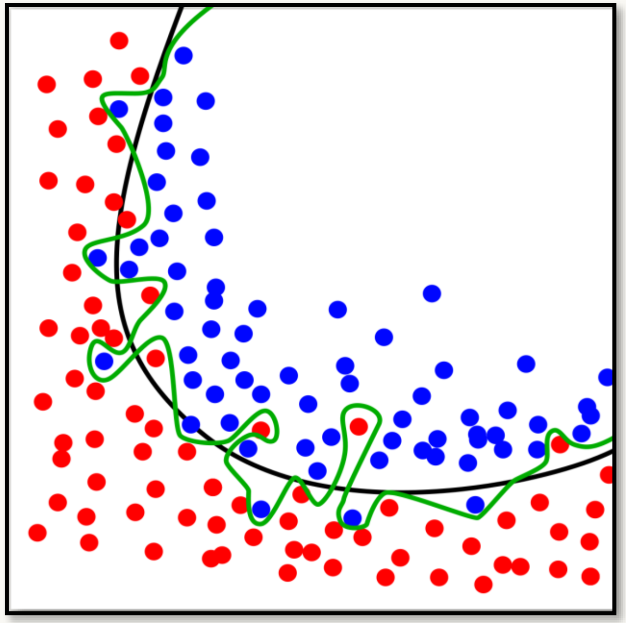
Overfitting

- Overfitting of the model occurs when the model learns just ‘too-well’ on the train data.
- This would sound like an advantage, but it is not.
- When a model is overtrained on training data, it performs worst on the test data, or any new data.
- Technically, the model learns the details as well as the noise of the train data.

Linear regression overfitting



Logistic regression overfitting



How to Avoid Overfitting?

- There are several ways of avoiding the overfitting of the model such as:
  - K-fold cross-validation
  - Resampling
  - *Reducing the number of features* (BUT this presents a disadvantage as removing features is sometimes equivalent to removing information)
  - *Applying Regularization to the model*

Regularization is a better technique than Reducing the number of features to overcome the overfitting problem as in Regularization we do not discard the features of the model.

Regularization works well when there are a lot of slightly useful features

Understanding Regularization

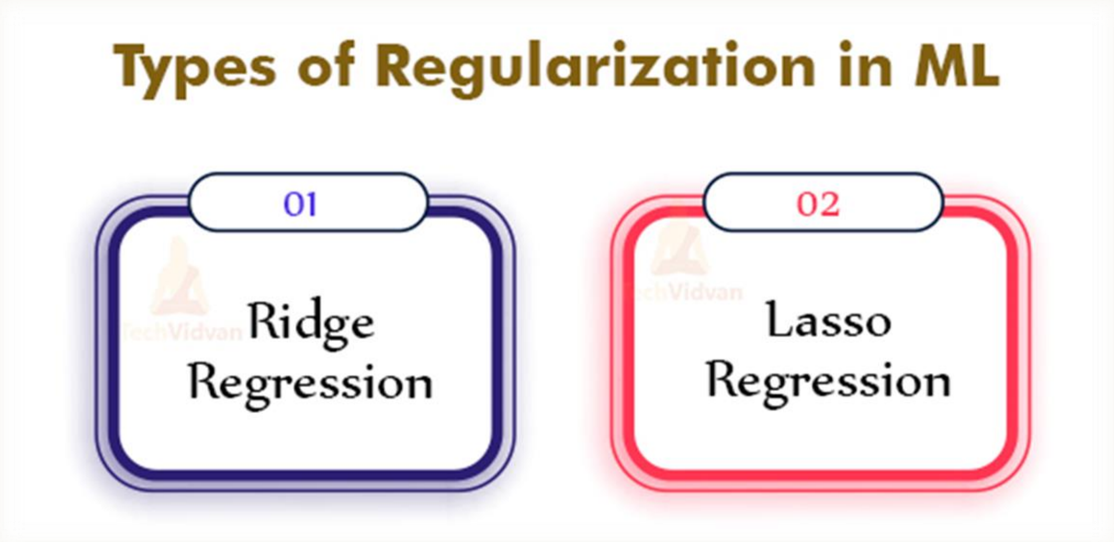
- Regularization is a technique that penalizes the coefficient.
- In an overfit model, the coefficients are generally inflated.
- Thus, Regularization adds penalties to the parameters and avoids them weigh heavily.
- The coefficients are added to the cost function of the linear equation.
- Thus, if the coefficient inflates, the cost function will increase. And Linear regression model will try to optimize the coefficient in order to minimize the cost function.

- Practically, you can check if the regression model is overfitting or not by RMSE (Root Mean Square Error).
- A good model has a similar RMSE for the train and test sets.
- If the difference is too large, we can say the model is overfitting to the training set.

- There are various techniques for adding penalties to the cost function

We will explore the most common Regularization Techniques of Linear Models

Regularization Techniques of Linear Model



Elastic-Net Regression Regularization (combines both L1 and L2)

Regularization Techniques (Cont.)

1. Ridge Regression (L2 Regularization):

- Here, we're going to minimize the sum of squared errors and sum of the squared coefficients (β).
- The coefficients (β) with a large magnitude will generate the graph peak and deep slope, to suppress this we're using the lambda (λ)
- lambda (λ) is called a Penalty Factor and help us to get a smooth surface instead of an irregular-graph.
- Ridge Regression is used to push the coefficients(β) value nearing **zero** in terms of magnitude.
- This is L2 regularization, since it's adding a penalty-equivalent to the **Square-of-the Magnitude** of coefficients.

Ridge Regression = Loss function + Regularized term

Transforming the Loss function into Ridge Regression

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 \Rightarrow \sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2$$

Loss function — *Loss function + Regularized term*

Designed by Author (Shanthababu)

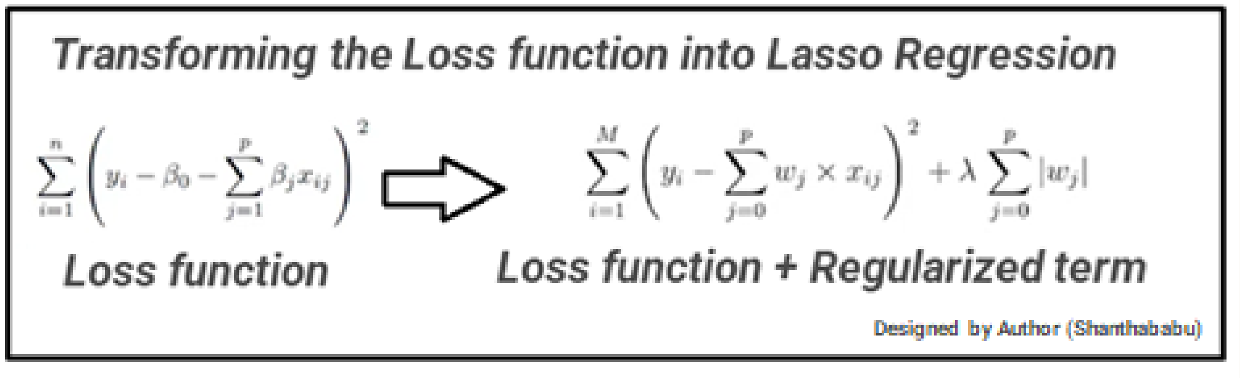
Note : $j = 1 \rightarrow n$, θ_0 is not regularized, it is the intercept

Regularization Techniques (Cont.)

2. Lasso Regression (L1 Regularization):

- LASSO stands for Least Absolute Shrinkage and Selection Operator.
- It is very similar to Ridge Regression, with little difference in Penalty Factor that coefficient is magnitude instead of squared.
- In Lasso there are possibilities of many coefficients becoming zero, so that corresponding attribute/features become zero and dropped from the list.
- This ultimately reduces the dimensions and supports for dimensionality reduction. So, it's deciding that those attributes (features) are not suitable for predicting target value.
- This is L1 regularization, because of adding the Absolute-Value as penalty-equivalent to the magnitude of coefficients.

Lasso Regression = Loss function + Regularized term

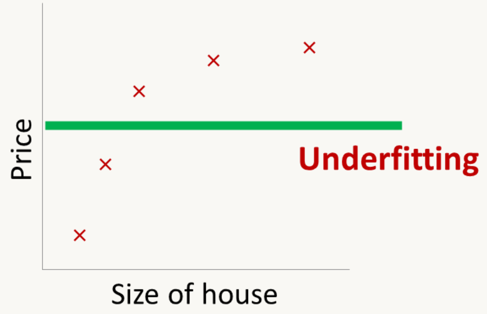


Regularization Techniques (Cont.)

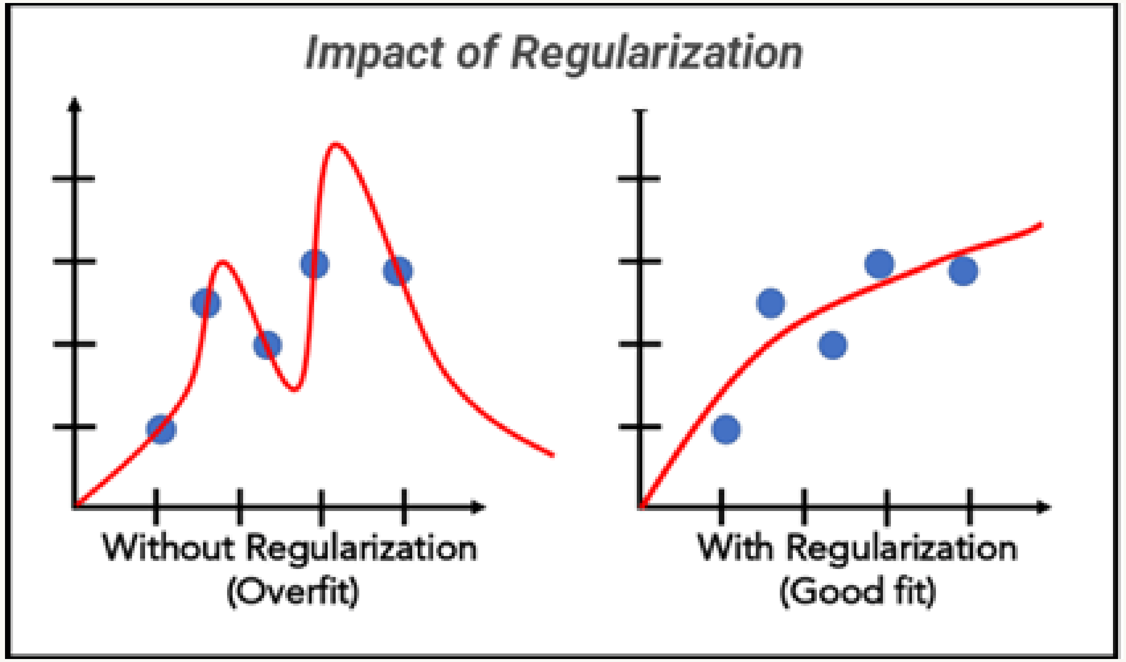
Characteristics of Lambda

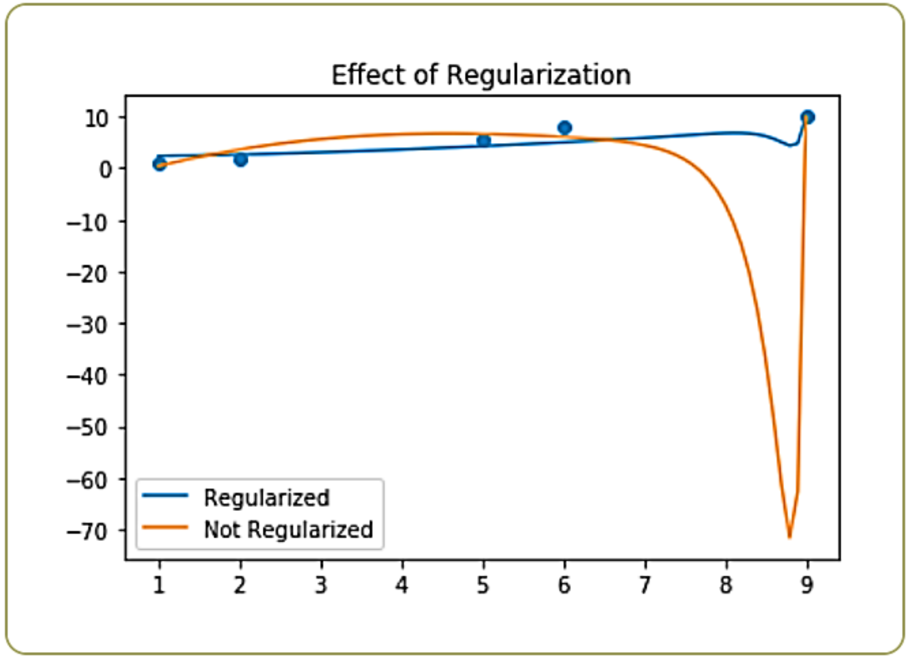
Remember one thing that the Ridge never make coefficients into zero, Lasso will do. So, you can use Lasso for feature selection.

	λ = 0	λ => Minimal	λ =>High
Lambda - Penalty Factor (λ)	• No impact on coefficients (β) and model gets Overfit • Not suitable for Production	• Generalized model. • Acceptable accuracy • Eligible for Test and Train • Fit for Production.	• Very high impact on coefficients (β). • Leading to underfit. • Ultimately not fit for Production.



Impact of Regularization





Regularizated Linear Regression

Example (with Python)

```
In [23]: from sklearn import datasets
         from sklearn.linear_model import Lasso
         from sklearn.model_selection import train_test_split
         #
         # Load the Boston Data Set
         #
         bh = datasets.load_boston()
         X = bh.data
         y = bh.target
         #
         # Create training and test split
         #
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
         #
         # Create an instance of Lasso Regression implementation
         #
         lasso = Lasso(alpha=1.0)
         #
         # Fit the Lasso model
         #
         lasso.fit(X_train, y_train)
         #
         # Create the model score
         #
         lasso.score(X_test, y_test), lasso.score(X_train, y_train)

Out[23]: (0.655906082915434, 0.6899591642958296)

In [13]: lasso.coef_

Out[13]: array([-0.        ,  0.        , -0.        ,  0.22497382, -0.        ,
                 2.73102016, -0.        , -0.        , -0.        , -0.        ,
                -1.24748188,  0.26711155, -3.75408325])
```

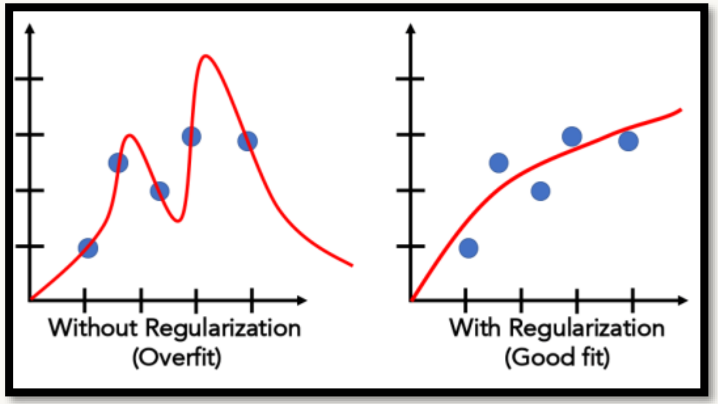
Summary

L1 Regularization (LASSO)

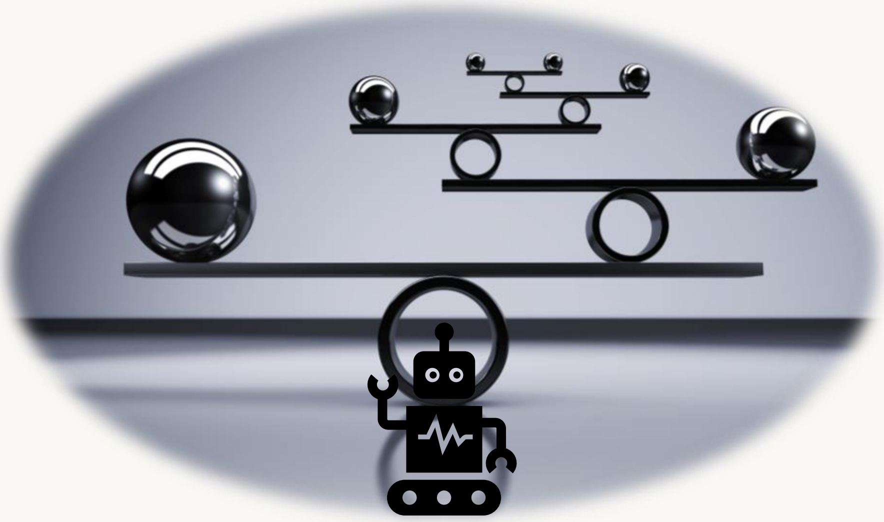
$$L(x,y) \equiv \sum_{i=1}^{n}(y_i - h_\theta(x_i))^2 + \boxed{\lambda \sum_{i=1}^{n}|\theta_i|}$$

L2 Regularization (RIDGE)

$$L(x,y) \equiv \sum_{i=1}^{n}(y_i - h_\theta(x_i))^2 + \boxed{\lambda \sum_{i=1}^{n}\theta_i^2}$$



The End..



Any Questions?