Data Analysis Lab

Ridge Regression
Lasso Regression

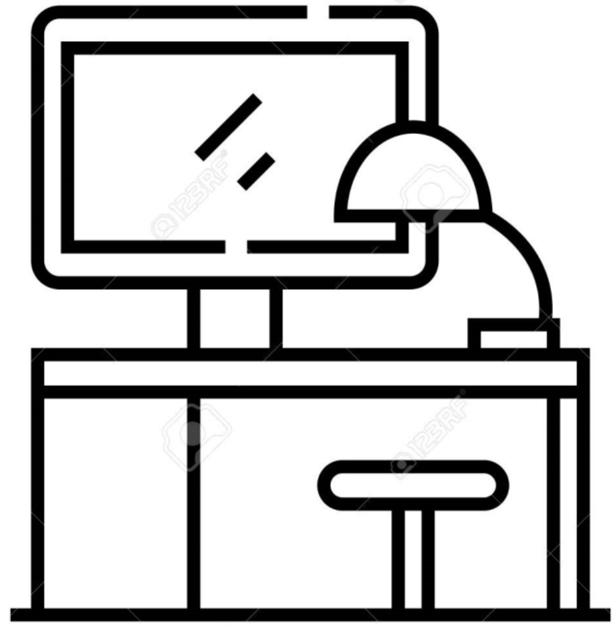
Fátima Leal
2022/2023

DCT DEPARTAMENTO CIÊNCIA E TECNOLOGIA

UPT UNIVERSIDADE PORTUCALENSE

Previous Lesson

- Machine Learning introduction
- Linear regression
- Logistic Regression



UPT DCT DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Outline

- Accuracy metrics for regressors
- Ridge Regression
- Practical Examples
- Lasso Regression
- Practical Examples

Accuracy metrics for regressors

- **R2** score to get the accuracy of your model on a percentage scale
- **Mean Absolute Error (MAE)** is simply defined as the sum of all the distances/residuals (the difference between the actual and predicted value) divided by the total number of points in the dataset. It is the absolute average distance of our model prediction.

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

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Accuracy metrics for regressors

- **Root Mean Squared Error (RMSE)** is the square root of the average squared distance (difference between actual and predicted value).

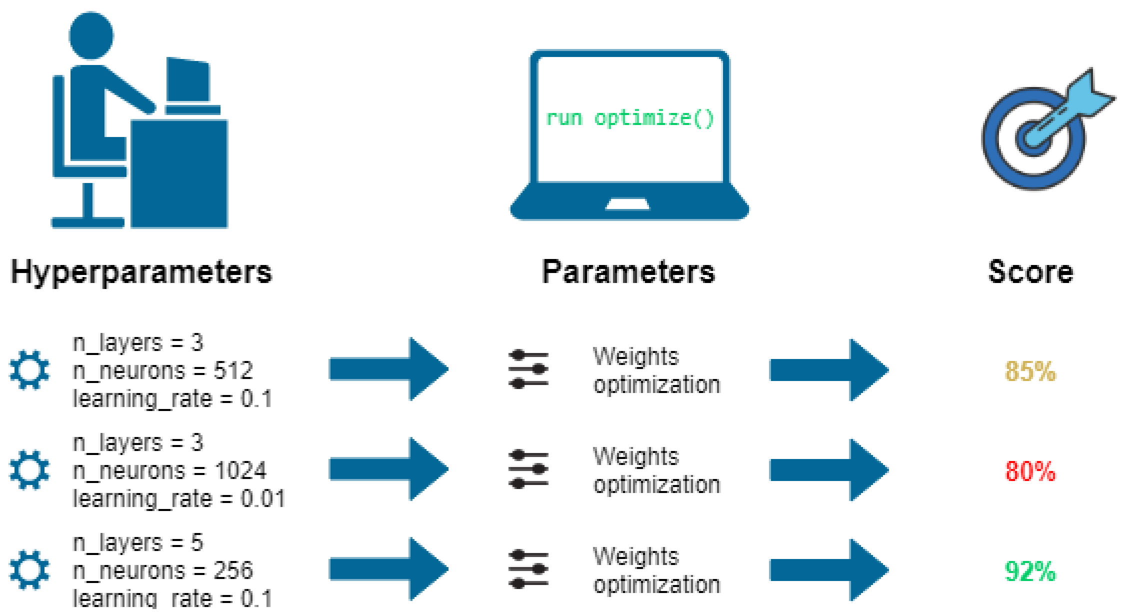
$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- RMSE is a popular evaluation metric for regression problems because it not only calculates how close the prediction is to the actual value on average, but it also indicates the effect of large errors. Large errors will have an impact on the RMSE result.

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

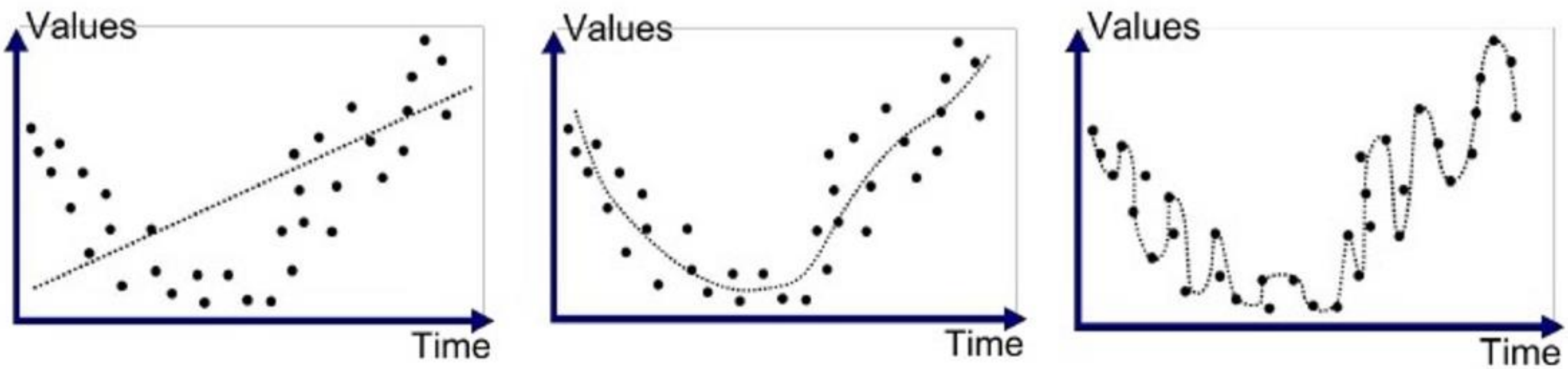
Hyperparameters

- Machine learning parameters whose value is chosen before a learning algorithm is trained



Regularization

– Good accuracy score isn't always good



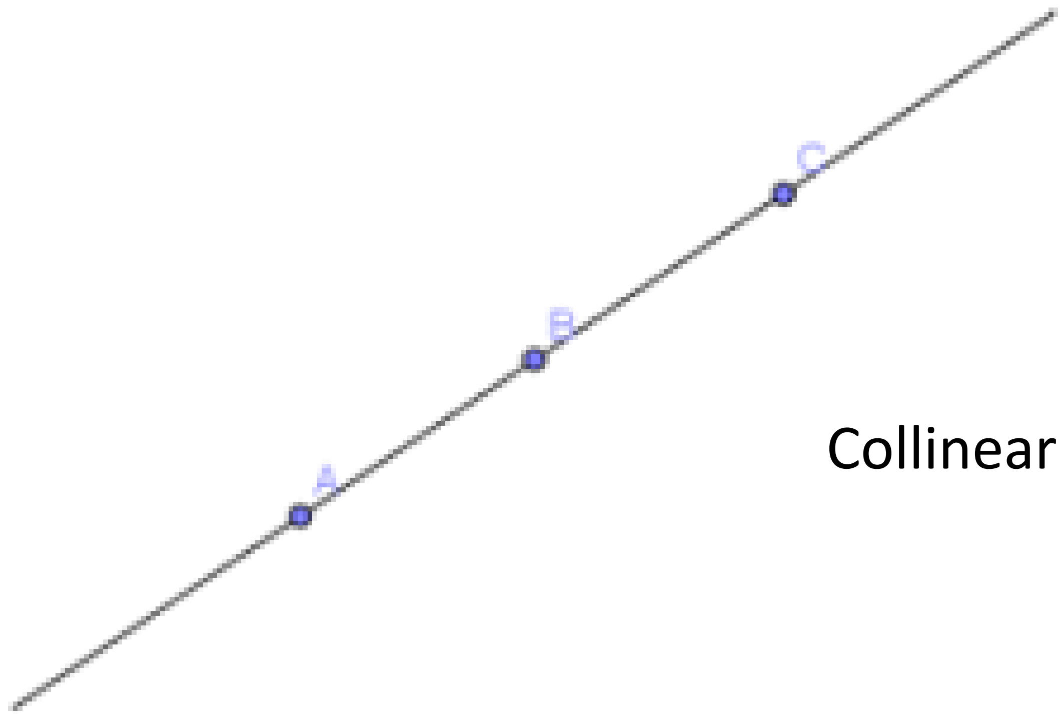
Underfitted | Good Fit/Robust | Overfitted

Regularization

- Good accuracy score isn't always good
- A model can predict the training set very well, but it can't perform new dataset.
- **Overfitting**
- **Regularization solves this problem**

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Ridge Regression



Collinearity

Ridge Regression

- One of the most common problems with linear regression is the ill-conditioning that causes instabilities in the solution. Ridge regression has been introduced to overcome this problem.
- The presence of multi-collinearities forces $\det(X^TX)$, and this implies that the inversion becomes extremely problematic.
- The presence of multi-collinearities is based on the computation of the condition number of X^TX defined as: $k(X^TX) = \frac{\sigma_{max}(X^TX)}{\sigma_{min}(X^TX)}$
- A small $k(X^TX)$ associated with a well-conditioned problem, hence the inversion is not problematic. On the contrary, when $k(X^TX) > 15$ the problem is ill-posed, and the result can change dramatically following small variations in X.

Ridge Regression

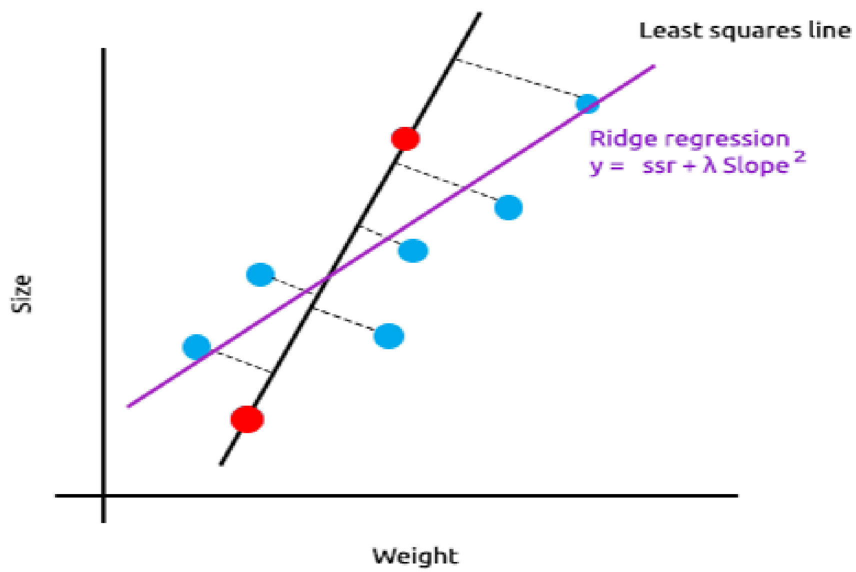
- A very simple and effective way to solve this problem is to employ a ridge (or Tikhonov) regularization, based on the L_2 norm of the parameter vector. Considering homoscedastic noise, the least-squares cost function becomes equal to: $L = (Y - X.\bar{\theta})^T(Y - X.\bar{\theta}) + \alpha\bar{\theta}^2\bar{\theta}$
- The parameter α determines the strength of the regularization and its role is immediately clear when considering the solution: $\bar{\theta} = (X^T X + \alpha I)^{-1} X^T Y$
- Ridge regularization plays a fundamental role in preventing overfitting, but, in the context of linear regression, its main effect is to bias the model in order to lower the variance.

Ridge Regression

- Ridge Regression **penalty = lambda1 x square of the magnitude of the coefficients**

- By shrinking the coefficients Ridge Regression reduces the model complexity and multi-collinearity. **But it keeps all the variables**.

- It's also called L2 regularization



DEPARTAMENTO **CIÊNCIA E TECNOLOGIA**

Ridge Regression example

Ridge Regression Documentation

```
from sklearn.datasets import load_diabetes
import numpy as np
from sklearn.linear_model import Ridge
from sklearn.metrics import r2_score,mean_absolute_error

data=load_diabetes()
X=data['data']
Y=data['target']

lrr=Ridge()
lrr.fit(X,Y)

print("R2={:.2f}".format(r2_score(Y,lrr.predict(X))))
print("MAE={:.2f}".format(mean_absolute_error(Y,lrr. predict(X))))
```


DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Optimization RidgeCV

```
from sklearn.linear_model import RidgeCV
from sklearn.datasets import load_diabetes
import numpy as np


data=load_diabetes(return_X_y=True)

rcv=RidgeCV(alphas=np.arange(0.1,1,0.01))
rcv.fit(X,Y)
print("Alpha:{:.2f}".format(rcv.alpha_))
print(rcv.coef_)
print(rcv.score(X,Y))
```


DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Exercise

- Using the Admission_Predict dataset implement the RidgeClassifier
- Using the RidgeClassifierCV to find the optimal hyperparameters

Lasso Regression – L1

- Ridge regression produces a global parameter shrinkage, but, the constraint surface is a hypersphere centered at the origin. Independent from the dimensionality, it is smooth, and this prevents the parameters from becoming null.
- On the other hand, an L_1 penalty has the advantage of performing an automatic feature selection, because the smallest weights are pushed toward an edge of the constraint hypercube.
- Lasso regression is formally equivalent to ridge, but it employs L_1 instead of L_2: $L = (Y - X.\bar{\theta})^T (Y - X.\bar{\theta}) + \alpha ||\bar{\theta}||_1$

Lasso Regression – L1

- The parameter α ontrols the strength of the regularization, which, in this case, corresponds to the percentage of parameters that are forced to become equal to zero.
- Lasso regression shares many of the properties of ridge, but its main application is feature selection. In particular, given that a linear model has a large number of parameters, we can consider the association: $\mathit{effect} \sim \mathit{cause\_1} + \mathit{cause\_2} + \ldots + \mathit{cause\_m}$

Lasso Regression example

```
from sklearn import linear_model

clf=linear_model.Lasso(alpha=1)
clf.fit(X,Y)

print("R2={:.2f}".format(r2_score(Y,clf.predict(X))))
print("MAE={:.2f}".format(mean_absolute_error(Y,clf.
predict(X))))
```

Lasso vs Ridge

- **Ridge Regression**
  - When all the features you have are important to your model
  - When you don't want to do feature selection as well as feature removing
- **Lasso Regression**
  - When you have too many features
  - And you know some of them don't have any significance to your model
  - When you want to remove the features with less importance

UPT DCT DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Exercises

- Use the datasets available on Moodle "Auto...Machine...Stock...USA_Housing"
- Do a exploratory/statistical analysis of the dataset
- Divide the dataset in train and test
- Apply different regression model
- Obtain the mean and std of the scores
- Obtain the R^2 and MAE
- Draw a line plot chart with the predictions of a Ridge and Linear Regression with also the ground truth

Exercises

- Continue now your project starting with this algorithms

UNIVERSIDADE
PORTUCALENSE

Do conhecimento à prática.