CS412 Machine Learning - Homework 4 Linear Regression and Evaluation Metrics

Engincan Varan 25050

Deadline: 30 April 2020, 23:55
Late submission: till 2 May 2020, 23:55
(-10pts penalty for **each** late submission day)

Submission
For your notebook results, make sure to run all of the cells and the output results are there.
Please submit your homework as follows:

- Download the .ipynb and the .py file and upload both of them to SuCourse.
- Also submit a single pdf document by solving questions on the sheet.
- Link to your Colab notebook (obtained via the share link in Colab) in the sheet:

Objective
The topic of this homework assignment is supervised learning. The first half is concerned with linear regression, and the second half, performance measure on classification tasks.

Startup Code
https://colab.research.google.com/drive/1W80EpGJYudkQ7Sz2pbAHffvt9bo_ITHH
To start working for your homework, take a copy of this folder to your own google drive.

Software: You may find the necessary function references here:

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.KFold.html

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.RidgeCV.html

Question 1: 75 pts - Predict the price of houses.

Dataset Description

https://raw.githubusercontent.com/OpenClassrooms-Student-Center/Evaluate-Improve-Models/master/house_prices.csv

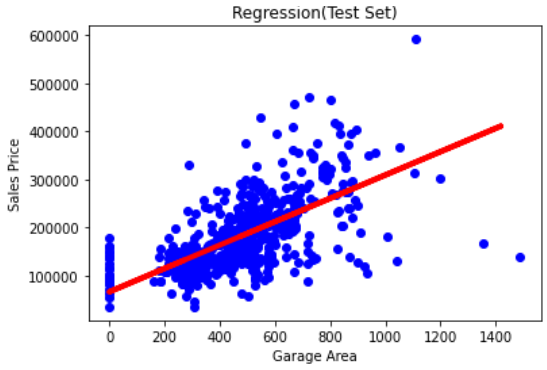
In this dataset, there are 2930 observations with 305 explanatory variables describing (almost) every aspect of residential homes.

a) **Find the correlation between garage area and sale price by applying linear regression. Print the bias and slope. Print the train and test R2. Plot the test set with a scatter plot and add the linear regression model line.**

   From the linear regression we can see that as the garage area gets bigger, sale prices increase with it.

```
Slope → Regressor coefficient or slope: 243.15503241812982
Bias  → Interception point with axis: 65960.99071000557

Train R2 → 0.42146105689364743
Test R2 → 0.35547211068663587
```



b) **Apply multiple linear regression by taking all input features. Print the train and test R2.**

```
Train R2 → 0.9347186459931469
Test → -3.0013667439899635e+17
```

c) Comment on part a and b results. Why R2 is low in part a? Why test R2 is low although train R2 is quite high in part b?

In part A, we used only 1 input feature “garage area” so our R2 score is low. Since, garage area is not the only input that is effective on price of the house, at some values our predictions are not correct. As garage area increases the house price also increases. However, there may be cases that has enormous garage area, but the house is very bad therefore the price is low or vice versa.

In part B, we used every possible input features we have to decide on the price of the house. In the end we had a very good train R2 very close to 1. However, by using every input feature we overfit our model and we failed hard in the test R2, so it becomes very low compared to the train R2.

d) Apply ridge regression with cross-validation by taking all input features. Print optimal alpha. Also print the train and test R2.

```
Train R2 → 0.9083847482786301
Test R2 → 0.9004485869152923

Optimal Alpha → 5.0
```

e) Discuss on regularization. What is ridge regression? When do we use it? And what is the effect on features?

Ridge regression is used to regularize the weights of the input features. By using ridge regression, we regulate the input weight and optimize them, so that one weight will not dominate the other weights. In our case for example, using multivariable linear regression caused our line to be overfit so that our test results were very low. We were not certain about the weights of the features. Some of them may be very dominant over the others, yet very unrelated to our label. Therefore, using ridge regression prevent our data to overfit and overcomes this problem.

1) Ridge regression regulates the input features and optimize them.
2) We use it to prevent our data to overfit.
3) It affects the weight on the features (coefficients)

f) Print regression coefficients for multiple linear regression and ridge regression. Comment on the change of feature weights. What is the effect of ridge regression on feature weights?

Since there are lots of them, to see more please check Colab.

Some coefficients for multiple linear regression (first part):

```
[-1.06455223e+04  1.66987443e+04  1.19278927e+05  5.85444713e+04
  4.41405396e+04  3.86666317e+04  5.42380923e+03  4.04458484e+04
 -2.20712468e+15 -5.96752705e+14 -9.13508728e+14  2.38935716e+15
 -1.99613356e+15 -8.65787819e+14 -4.46100842e+14  2.22547300e+15
  6.32176747e+03 -3.60854301e+02  1.59167186e+04  3.69108761e+03
 -2.61469007e+04 -3.94701083e+04  1.20117063e+04  2.96057125e+04
  4.53358664e+03  1.44617269e+04  2.73779602e+04  1.22814560e+04
 -4.57801402e+03  1.55042520e+04 -7.25023167e+03  2.35798569e+04 …]
```

Some coefficients for ridge regression (first part):

```
[-8.55688261e+03 -2.24947003e+03  2.51985809e+04  6.11176862e+04
  3.05688283e+04  1.54705088e+04  6.42802448e+03  3.43299527e+04
  3.11874100e+04  1.08561215e+04  4.41285001e+02  3.16888749e+04
  5.88293295e+04  4.46935789e+04  3.22251349e+03  7.08001945e+04
  1.84925753e+04 -4.22889476e+03  3.46719489e+04  1.03066466e+04
  5.21474078e+03 -1.06430026e+04  2.76039916e+04  2.62569504e+04
 -2.13425715e+01  3.07977027e+04  2.28947048e+04  1.45086270e+04
 -5.71374758e+02  9.26247548e+03  9.93966295e+02  2.43749213e+04 …]
```

The effects are clearly visible for each feature weights. Some of them increased in weight whereas some of them decreased. It normalizes the weights and put the values between in some certain floor attic.

In the end, ridge regression manipulates the weights of the input features to prevent overfitting problem and creates a more correct test results.

Question 2: 25 pts - Evaluation metrics.

a) 15 pts - Provide the Confusion Matrix, Accuracy, Error, Precision, Recall, and F1-Score for the fruit classification problem. The output of test data classification results is given in the following table.
Use both macro and micro averaging methods.

mass	width	height	color_score	class	prediction	T/F
154	7.1	7.5	0.78	orange	lemon	F
180	7.6	8.2	0.79	orange	lemon	F
154	7.2	7.2	0.82	orange	apple	F
160	7.4	8.1	0.80	orange	orange	T
164	7.5	8.1	0.81	orange	apple	F
152	6.5	8.5	0.72	lemon	lemon	T
118	6.1	8.1	0.70	lemon	apple	F
166	6.9	7.3	0.93	apple	apple	T
172	7.1	7.6	0.92	apple	apple	T

General Accuracy → **4/9**
General Error (Misprediction Rate) → **5/9**

		Gold Output					
		Orange	Lemon	Apple			
System Output	Orange	1	0	0	precision_o	1/1+0+0	1
	Lemon	2	1	0	precision_l	2/2+1+0	2/3
	Apple	2	1	2	precision_a	2/2+1+2	2/5
		recall_o	recall_l	recall_a			
		1/1+2+2	1/0+1+1	2/0+0+2			
		1/5	1/2	1			

f1_score (orange) = $\frac{2*1*\left(\frac{1}{15}\right)}{1+\left(\frac{1}{15}\right)}$ = **0.125**

f1_score (lemon) = $\frac{2*\left(\frac{2}{3}\right)*\left(\frac{1}{2}\right)}{\left(\frac{2}{3}\right)+\left(\frac{1}{2}\right)}$ = **0.57**

f1_score (apple) = $\frac{2*\left(\frac{2}{5}\right)*1}{\left(\frac{2}{5}\right)+1}$ = **0.57**

f1_score (average) = $\frac{(0.125+0.57+0.57)}{3}$ = **0.42**

MicroAvg Accuracy → **17 / 27** || MacroAvg Accuracy → **17 / 27**

MicroAvg Error → **10 /27** || MacroAvg Error → **10 / 27**

MicroAvg Precision → **0.44** || MacroAvg Precision → **0.57**

MicroAvg Recall → **0.44** || MacroAvg Recall → **0.56**

MicroAvg f1 Score = $\frac{2*\left(\frac{4}{9}\right)*\left(\frac{4}{9}\right)}{\left(\frac{4}{9}\right)+\left(\frac{4}{9}\right)}$ = **0.45** || MacroAvg f1 Score = $\frac{2*0.57*0.56}{0.57+0.56}$ = **0.56**

MACROAVERAGE

	Class 1: Orange	
	True Orange	True Not
System Orange	1	0
System Not	4	4
Precision	1/1	1
Recall	1/1+4	1/5

	Class 2: Lemon	
	True Lemon	True Not
System Lemon	1	2
System Not	1	5
Precision	1/1+2	1/3
Recall	1/1+1	1/2

	Class 3: Apple	
	True Apple	True Not
System Apple	2	3
System Not	0	4
Precision	2/2+3	2/5
Recall	2/2+0	1

MacroAvg Precision	(1+1/3+2/5) / 3	0.577778
MacroAvg Recall	(1/5+1/2+1) / 3	0.566667

MICROAVERAGE

	POOL	
	True Yes	True No
System Yes	4	5
System No	5	13
MicroAvg Precision	4/4+5	0.44
MicroAvg Recall	4/4+5	0.44

b) 10 pts - The table shows 18 data and the score assigned to each by a classifier. It is a binary classification problem. The active/decoy column shows the ground truth labels. Plot the corresponding ROC curve.

id	score	active/decoy	id	score	active/decoy
O	0.03	a	L	0.48	a
J	0.08	a	K	0.56	d
D	0.10	d	P	0.65	d
A	0.11	a	Q	0.71	d
I	0.22	d	C	0.72	d
G	0.32	a	N	0.73	a
B	0.35	a	H	0.80	d
M	0.42	d	R	0.82	d
F	0.44	d	E	0.99	d

Assume that above the threshold, we predict it as decoy. The graph would look like this, and the ordering is like:

AADADAADDADDDDADDD

0 1