Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

DECLARATION

I understand that this is an individual assessment and that collaboration is not permitted. I have not received any assistance with my work for this assessment. Where I have used the published work of others, I have indicated this with appropriate citation.

I have not and will not share any part of my work on this assessment, directly or indirectly, with any other student.

I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at http://www.tcd.ie/calendar.

I have also completed the Online Tutorial on avoiding plagiarism 'Ready Steady Write', located at http://tcd-ie.libguides.com/plagiarism/ready-steady-write."

I understand that by returning this declaration with my work, I am agreeing with the above statement. ☑

Name: Xiangyu Zheng

Date: 19/12/2021

1.

I chose No. 21 Leinster Street South Station and No. 5 Open Charlemont Street. Because No. 21 is near Trinity College, there is a lot of traffic, and the number of bicycles available must change very sharply, and No. 5 is far away from the city center. In places where people walk by the river, bicycles may change a lot at a certain time. Both stations are very meaningful for research.

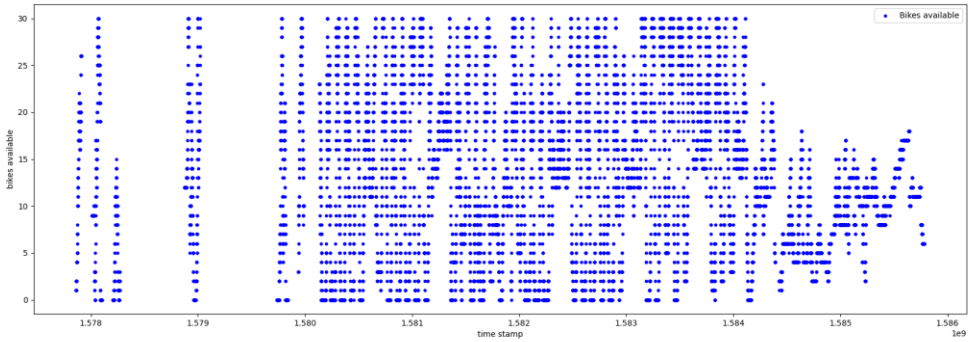

figure 1 station21 bikes available

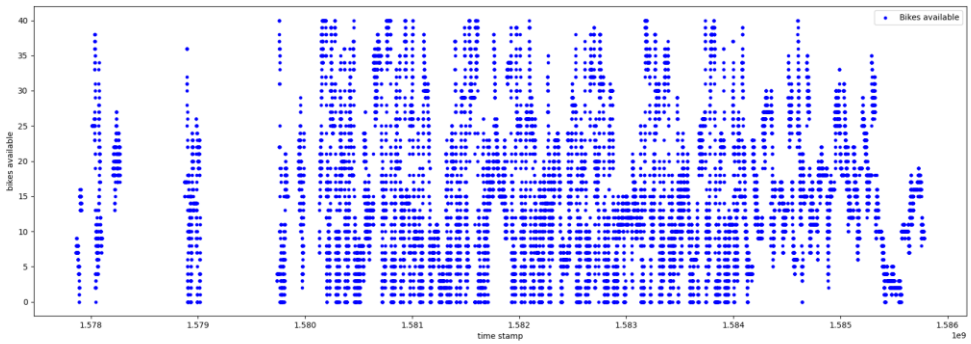

Figure 2 station5 bikes available

(i) feature engineering

First, I will remove the original features that are useless at first glance, such as the name of the station, ID, update time, etc., and then I plan to use the method of cross-validating the new features to improve the feature vector. I used the One-step ahead prediction method to select 3 preliminary features for predicting the availability of bicycles after 10 minutes, the number of bicycles available at the current time, 10 minutes ago and 20 minutes ago. The target is the data after 10 minutes.

Obviously, the three features are not enough. I analyzed the scatter plot of bicycle distribution and found that the data is missing in time. Then I found that the data is actually periodic on a daily basis, so I plan to abandon the insertion of the missing data and add it to the original data. To extract more features. Use lasso regression and 5-fold cross-validation to verify the feasibility of the new features. Because in the use of lasso regression, the model can calculate the coefficient of each feature, and some coefficients with low correlation can even reach 0, which can help me choose features.

Use the data of three features to make predictions, use 5-fold cross-validation to get the

best alpha value of 0.1, and get the feature coefficient:

20 minutes ago	10 minutes ago	current
-0.03933	-0.00000	1.02244

And mean absolute error: 0.7305

next, 5 features of the bicycle availability at the station are added 30 minutes ago, 40 minutes ago, 1 day ago, 3 days ago and 1 week ago.
Also use 5-fold cross-validation to obtain the best alpha of 0.18, and the characteristic coefficient table:ss

1week	3d ago	1d ago	40m ago	30m ago	20m ago	10m ago	current
0.0075	0.0041	0.0103	-0.0214	-0.0205	-0.00	-0.00	1.0097

And mean absolute error: 0.72

Then I thought, the data seems to have regularity every day, so I added a time feature, the value is Hours*60+minutes, which is the total number of minutes in the current time. Get the coefficient of the feature: 0.0003, and the average absolute error: 0.7363
Its low relevance also affects performance, so I removed it.

Next, I think that because the historical data has some relevance, I will continue to add the data 10 days ago and 2 weeks ago. Obtain an alpha of 0.18, the average absolute error: 0.714 and the coefficients 10 days ago and 2 weeks ago: 0 and 0.00347. I added the average number of bicycles available at the current time within 3 days and 7 days respectively, and the best absolute error was 0.839 and 0.841, so I also gave up these two features.

I used the same feature selection method. I selected the features that predicted the 30-minute lag and 60 minutes later. After the target value became 30 minutes and 60 minutes. the interval became 30 minutes and 60 minutes. The average error is increased to 1.59 and 2.55. So I thought, can I still use 10-minute forecasts to predict 30 minutes? That is, I want to predict the availability of bicycles after 30 minutes, I first predict the data after 10 minutes and 20 minutes as features, and then predict the 30 minutes.

I tried to use the predicted data as a feature to predict the availability of bicycles with a 30-minute lag, but the average absolute error is still 1.59. So I finally decided to use 2 weeks ago, 10 days ago, 1 week ago, 3 days ago, 1 day ago, 40/120/240 minutes ago, 30/90/180 minutes ago, 20/60/120 minutes ago, 10 /30/60 minutes ago, 10 features to predict the data after 10/30/60 minutes.

I also encountered a similar situation when selecting the eigenvalues of station5, so I used the same features as station21

(ii) machine learning methodology

1. Lasso Regression

Lasso regression uses L1 regularization as a penalty term on the basis of a linear model to make the model more distinctive. When the penalty is increased, the coefficient of the feature may become 0, which helps to reduce the dimensionality and our selection of features. Its cost function is:

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda\sum_{j=1}^{n}|\theta_j|$$

Equation 1 Lasso Regression cost function

2. Support vector regression

Taking into account the previous process of selecting features, since the previous data will be selected as features, some data may not be available, and blanks cannot be used as features, which will be meaningless. Therefore, as the number of features increases, the data set becomes less and less, so I choose SVR to explore its performance changes under the change of the data set.

The feature of support vector regression is that it has an interval band, as shown in Figure 3.

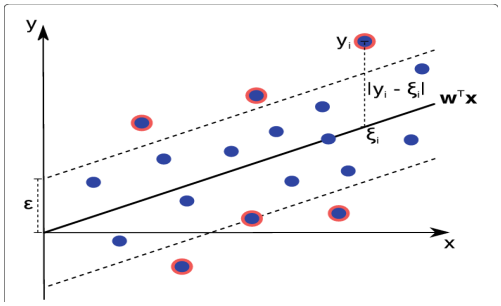

figure 3 support vector regression

The interval band is determined by some samples that are close to the dividing line. The data falling into the interval band is considered to be correct and the loss is not calculated. Therefore, a slack variable ξ is introduced to control the size of the interval band. The goal is[1]:

$$minimize \quad \frac{1}{2}||\varpi||^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*)$$

$$subject\ to \begin{cases} y_i - \langle \varpi, x_i \rangle - b \leq \varepsilon + \xi_i \\ \langle \varpi, x_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

Equation 2

The Gaussian kernel is used here to test this experiment, which maps features to high dimensions.

$$k(x_i, x_j) = exp\left(-\frac{||x_i - x_j||^2}{2a^2}\right)$$

Equation 3

(iii) Evaluation

First, compare 10 features, lasso regression and svr. I first used cross-validation to select the hyperparameters of the two models. Since it is to predict the number of available

bicycles, this is a regression problem, so the absolute average error is used as an indicator.

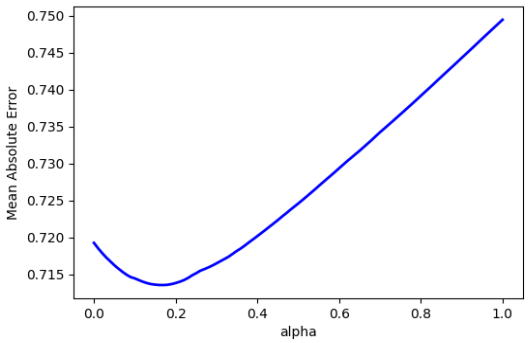

figure 4 Lasso regression MAE predicts 10 minutes later at station 21

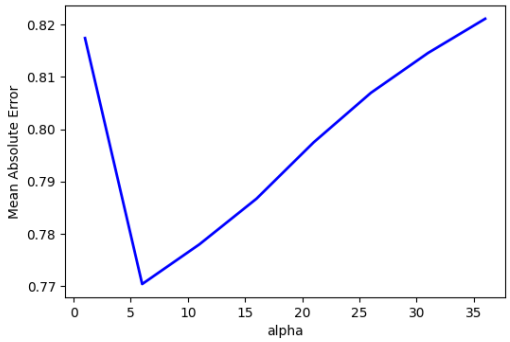

figure 5 SVR MAE predicts 10 minutes later at station 21

We can see from Figure 4 that under the 10-minute prediction, when lasso regression is alpha=0.18, the minimum MAE is 0.712. At this time, the model parameters are:

2 week ago	10 days ago	1 week ago	3 days ago	1 days ago	40 min ago	30 min ago	20 min ago	10 min ago	current
0.00308	0.0000	0.00246	0.00349	0.00798	-0.01817	-0.00364	-0.00000	-0.00000	0.99301

From picture 5, we can see that the minimum MAE of the SVR model is 0.77 when C=6 and has 7565 support vectors.

Table 1 shows the comparison between Lasso regression and SVR in six prediction situations at two stations. In each case, 10 features are used. I used a baseline model that randomly predicts available bicycles as a comparison

		Lasso		SVR		baseline
		MAE	alpha	MAE	C	MAE
Station21	Predict 10m	0.712	0.18	0.77	6	10.41
	Predict 30m	1.59	0.22	1.663	1	10.46
	Predict 60m	2.5479	0.49	2.5393	0.7	10.58
Station5	Predict 10m	1.296	0.05	1.4771	5.1	13.27
	Predict 30m	2.5084	0.09	2.738	0.9	13.15
	Predict 60m	3.7374	0.16	3.9236	0.5	13.20

Table 1 Lasso vs SVR

We can see through table1 that as the prediction time gets larger and larger, the performance of SVR and Lasso are getting closer and closer, even surpassing Lasso when station21 predicts 60 minutes. This may be because the Lasso regression value selects several important features. , Especially the characteristics of the current time, and the forecast time is relatively long, so it is becoming more and more inaccurate. SVR is always a dividing line constructed by several support vectors, so its performance is relatively stable.

			data	Lasso MAE	SVR MAE
Station21	Predict 10m	13features	212	0.7166	1.4354
	Predict 60m	13features	212	2.705	2.48

Table 2 Lasso vs SVR in 212 data

Finally, I tested the extreme case, the data volume was only 212. After 10 minutes of prediction, the best MAE of the Lasso available on the bicycle was 0.7166 and the SVR was 1.4354. At 60 minutes of prediction, SVR beat Lasso regression by 2.48. I think the reason is the same as that of table1. Maybe the data is still not enough and the features are not enough to show the characteristics of SVR.

2.

(i) What is a ROC curve. How can it be used to evaluate the performance of a classifier.

ROC is the abbreviation of receiver operating characteristic. It is used to evaluate the performance of the classifier. It introduces true positive (TP) (correctly predicted positive samples), true negative (TN), false positive (FP) and false negative (FN) (Wrongly predicted negative samples) Four main features, mainly through the calculation of true positive rate : $\mathrm{TPR} = \frac{TP}{TP+FN}$, false positive rate: $\mathrm{FPR} = \frac{FP}{FP+TN}$, accuracy: $\mathrm{ACC} = \frac{TP+TN}{TP+TN+FP+FN}$ and positive predictive value: $\mathrm{PPV} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}}$ etc. to evaluate performance, and then display them through graphs Come out, the x-axis is FPR, and the y-axis is TPR.

(ii) Give two examples of situations when a linear regression would give inaccurate predictions. Explain your reasoning.

1.As shown in Figure 1, unary linear regression may encounter inaccurate predictions when doing classification tasks. First, in order to minimize the mean square error, we fit a straight line (L1) through sample 1 and sample 2, and then set a threshold (P1) as the dividing line, one category is greater than 80kg, and the other category is less than 80kg. Then we introduce sample 3, fit the straight line L2, and then classify and find that the sub-sample S4 of sample 2 is above the horizontal dividing line, which is classified incorrectly. This leads to inaccurate classification, which indicates that the generalization ability of the data is poor when the unary linear regression is used for classification.

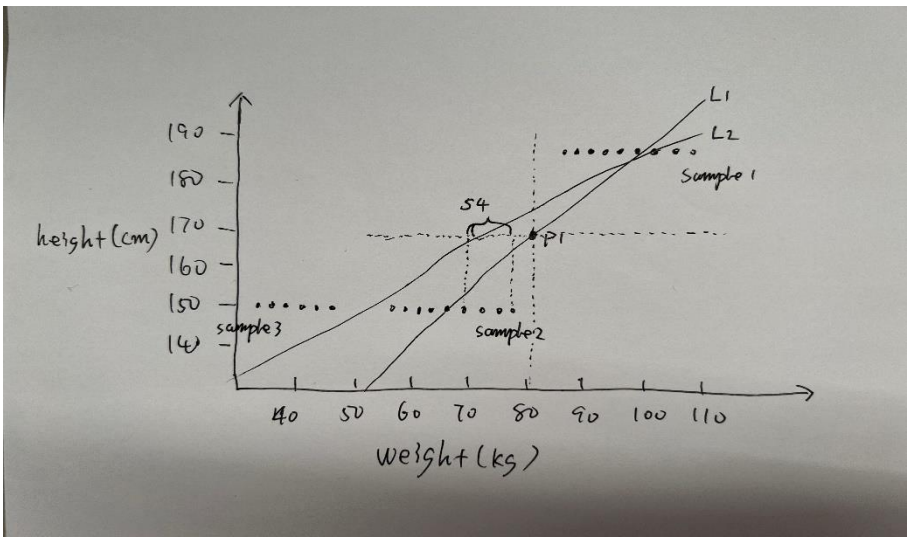

Figure 6

2. Overfitting

When we introduce too many useless features, if we do not take certain measures, the model will train weights for these features when training the data, resulting in these features are also calculated in the results when predicting, resulting in inaccurate predictions, such as We are calculating the height and weight to determine the obesity rate, adding the name to the feature.

(iii) Discuss three pros/cons of an SVM classifier vs a neural net classifier.

1. When the amount of training data is small, SVM will also perform better. This is because SVM maps features to high dimensions, and a small amount of support vectors affect the final result. We can grab key samples and eliminate redundant samples. The essence of the neural network is to fit a function by linear combination. When the amount of training data is small and the model is complex, it is difficult to fit a curve. For example, when the data is noisy, the model will cover the noise, which will cause over-fitting. combine.

2. In the problem of multi-classification, SVM seems to be relatively unsuitable. It is a two-class classifier. If you want to achieve multi-classification, you must modify the objective function or group multiple classifiers, or other methods. In any case, it will make the model extremely complex. The neural network is relatively simple, such as implementing a softmax classifier and adding multiple output layers to achieve multi-classification.

3. In terms of interpretability, neural networks are relatively poor. For example, after inputting some samples, it is difficult to understand or explain what happened to the neural unit in the middle to obtain the results. SVM has a lot of mathematical theoretical support, and Many kernel functions have relatively good interpretability.

(iv) Describe the operation of a convolutional layer in a convNet. Give a small example to illustrate.

In the convolution layer, use the kernel matrix to perform convolution calculations on the input matrix to obtain an output matrix.

$$\begin{bmatrix} 1 & 3 & 2 & 4 & 5 \\ 5 & 1 & 3 & 2 & 4 \\ 2 & 4 & 3 & 1 & 5 \\ 4 & 2 & 1 & 3 & 5 \\ 3 & 1 & 4 & 5 & 2 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 16 & & \\ & & \\ & & \end{bmatrix}$$

Step1

$$\begin{bmatrix} 1 & 3 & 2 & 4 & 5 \\ 5 & 1 & 3 & 2 & 4 \\ 2 & 4 & 3 & 1 & 5 \\ 4 & 2 & 1 & 3 & 5 \\ 3 & 1 & 4 & 5 & 2 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 16 & 15 & \\ & & \\ & & \end{bmatrix}$$

Step2

In addition, we can also use padding, step size, and multiple channels to operate the convolution process

(v) In k-fold cross-validation a dataset is resampled multiple times. What is the idea behind this resampling i.e. why does resampling allow us to evaluate the generalization performance of a machine learning model. Give a small example to illustrate.

Resampling is to solve the problem of data imbalance. For example, when we divide the training set/test set, a lot of noise happens to be in the training set. At this time, our model may overfit. In K-fold cross-validation, multiple re-sampling may extract noise into multiple training sets/test sets. We evaluate these training sets/test sets multiple times, which helps us evaluate the actual generalization ability of the model.

references

[1]. Smola, A.J., Schölkopf, B. A tutorial on support vector regression. *Statistics and Computing* **14,** 199–222 (2004). https://doi.org/10.1023/B:STCO.0000035301.49549.88

Appendix

Code: https://github.com/zhengtiantian/ML-final-exam.git