National University of Singapore
School of Computing
CS3244: Machine Learning
Solution to Tutorial 06

Regression Metrics and Data Processing

1. **Regression Evaluation Metrics**

In this problem, we discuss the regression metric, Mean Absolute Percentage Error (MAPE). MAPE is a measure of prediction accuracy of a forecasting method. Mathematically, it is expressed by a ratio defined by:

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

where y_i and $\hat{y}_i$ denote the actual values and forecast/prediction values at data point i. Here, n denotes the number of data points in total.

(a) Suppose that $y = [1, 2, 3, 4, 5]$ and $\hat{y} = [1, 2.5, 3, 4.1, 4.9]$. Calculate the MAPE of this prediction.

n = 5
$MAPE$ = 5.90%

(b) Assume that you are a supply-chain manager. You are using MAPE to judge your regression forecasts about **product demands** for the next month. We list them here. Discuss whether the listed MAPE shortcomings below will or will not affect you.

i. Data with zeroes or close to zeroes.
ii. Heavier penalty when predictions are higher than actual data.
iii. Assumption that zero in the data's unit of measurement holds meaning.

i. Yes, it will affect. Product demand can reach values of zero. MAPE produces undefined or infinite values when the actual values(y_i) are zero or close to zero.

ii. Yes, it affects as product demand cannot be negative values. MAPE is asymmetric when forecasts are strictly non-negative. It places a heavier penalty when forecasts($\hat{y}_i$) are higher than actuals(y_i). Percentage error cannot exceed 100% for low forecasts, but there are no upper limits for high forecasts as our product demands cannot be a negative value.

iii. No, this is not a shortcoming for us. MAPE assumes that when the value is zero, it is meaningful. In our case, forecasts for product demand being zero would cause MAPE to return 100% if our actual is non-zero. For units of measurements that have arbitrary zero values, using MAPE no longer makes sense.

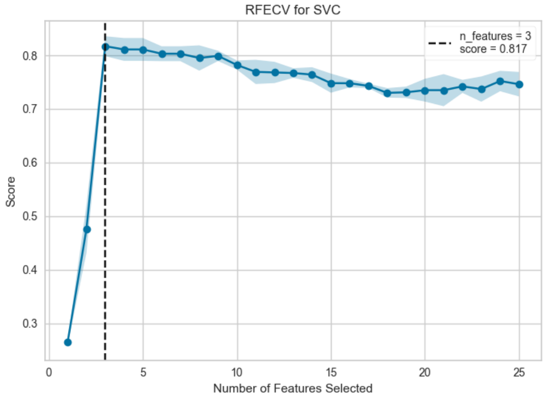

Figure 1: Recursive Feature Elimination - Image Credits

(c) Finally, we define an alternative to MAPE, which is SMAPE (Symmetric-MAPE). We define SMAPE as follows.

$$SMAPE = \frac{100\%}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{(|\hat{y}_i| + |y_i|)/2}$$

Analyze how SMAPE fixes MAPE's problems.

As its name suggests, SMAPE overcomes the asymmetry of negative and positive errors of MAPE. Its formulation results in a fixed lower bound of 0% and upper bound of 200%.
Can you think which shortcomings still persist from MAPE?
What other shortcomings does SMAPE introduce?

2. **Curse of Dimensionality**

(a) **Feature Selection (Wrapper):** Observe figure 1 regarding Recursive Feature Elimination (RFE) closely and answer the questions below.

i. Describe the general trend in the graph.
The graph peaked when only the top 3 features are considered. The score slowly decreases as additional features are added. This is due to the fact that the latter features do not provide any further information. It is possible that they are a source of noise, explaining the decrease in the score as more features are added.

ii. Is this graph theoretically possible if we implement the high-level RFE algorithm described in the lecture? Explain your answer.
The graph is theoretically possible. Even though RFE is to eliminate the features with most decrease, that does not mean that it has to be monotonically increasing. RFE retrains the model each time after a feature is removed. As such, the next feature to be removed (new highest decrease) can always have either a net positive or negative impact on performance after retraining.

(b) **Feature Selection (Filter):** Consider the following correlation matrix shown in figure 2 about cell nucleus data for breast cancer patients. There are six features and their correlations within each other.

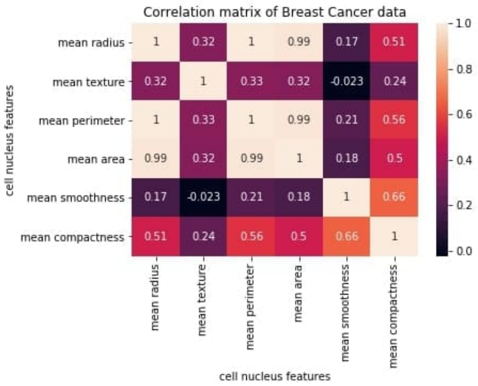

Figure 2: Correlation Matrix - Image Credits

i. Which feature(s) should we remove from the table to avoid redundant information?
We see that all three of mean radius, mean perimeter, and mean area are strongly correlated, with value very near to 1. Hence, we need to only keep one of them. The feature we end up keeping will be not very relevant, as shown below.

ii. There are some data with 1 correlation. Is this a coincide?
Note that there are a relation between radius r and area A of a circle, $A = \pi r^2$ and also a relation between radius r and perimeter P with $P = 2\pi r$. Hence, we should expect that the correlation is very high.

3. **Data Resampling Techniques**

You are deciding on which data resampling methods to use for each of the following *imbalanced* datasets. Which of the data resampling methods should be applied? Briefly explain when and how the method(s) can be used in tandem with train-test split.

(a) Dataset of 2 class labels. 80% majority class and 20% minority class.
Oversampling and SMOTE are plausible methods. Undersampling should not be used as we lose 75% of the majority class, which makes up 60% of the overall dataset. Data resampling methods are always done after the train-test split, and only on the training set.

(b) Dataset of 3 class labels and discrete and continuous features. 45% majority class and 55% from minority classes.
Here, the plausible methods depends on the composition of the 55% minority. If the minority classes are split 30%-25%, all three methods are applicable. However, if the minority classes are split 40%-15%, undersampling should be avoided for the same reason as in (a).

Extra care also has to be taken to determine if there are any logical errors when using SMOTE as we have discretized features. If the features are intended to be strictly categorical variables, SMOTE should be avoided due to its interpolation.

(c) Dataset with continuous output variable.
After train-test split, we assign the training instances to bins based on their output values and plot a simple histogram. With dataset domain knowledge, whether resampling is required or not is determined. If required, we choose the resampling method

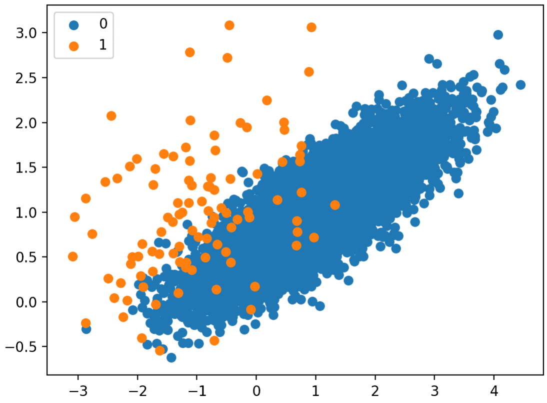

Figure 3: Scatter Plot of Imbalanced Classes - Image Credits

based on the overall distribution of the bins and the type of the dataset features. When is undersampling preferred over oversampling?

4. **Data Resampling: SMOTE**

Refer the general Steps of SMOTE given to you below.

1. From all the data points of your minority class, pick a random point.
2. Find the k nearest neighbours to that point.
3. Pick one of the neighbours randomly, now we have a pair from the minority class.
4. Draw an imaginary line between the pair and pick a random point along the line.
5. The new random point is added to the minority class.

Figure 3 is a scatter plot of an imbalanced dataset to be used in a binary classification problem. Roughly illustrate how the transformed dataset will look like after SMOTE.

Taking k to be the default value of 5, we can find out all combinations of the minority class pairs. Add points along the lines connecting each minority class pair and we would get a result similar to what we see in Figure 4 after SMOTE. (We will get varying results as k changes)

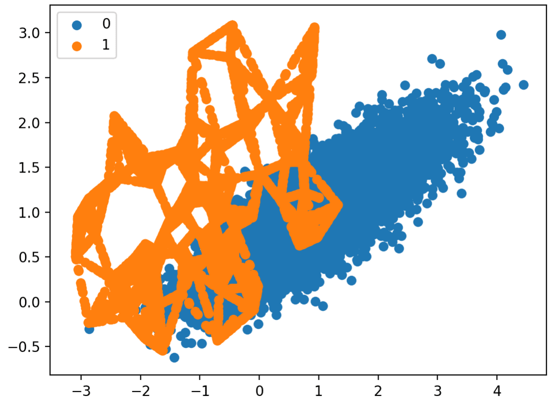

Figure 4: Scatter Plot after SMOTE transformation - Image Credits