Machine Learning Assignment Report

By 246743

Abstract

A classification problem can be a two-category problem (yes/no) or a multi-category problem. Its results are continuous values between (0,1). In this experiment, a series of models will be used on a two-category problem of whether a photograph is "memorable" and the results will be compared for different input features and classifiers. Here we show that the Linear Regression classifier is the most accurate, achieving an accuracy of around 72%-75%. The data from the test set has been predicted.

Introduction

Pre-Processing

- KNN Imputer

Instead of simply eliminating NaNs or converting them to averages, KNN (K-Nearest Neighbour) divides all data into clusters according to Euclidean distance and then fills in the missing values using nearest neighbours according to this distance. In addition, KNN Imputer itself does not need to be trained, so its final complexity is determined by the data (if the total amount of data is n, the final time complexity is n), which means it has a fast training process.

- Normalization

One type of data normalisation is the mapping of all data values proportionally to a smaller range of intervals. (For example, normalisation is the reduction of data to an interval of 0-1). However, another type of normalise is used in this experiment, which scales each sample to a certain number of unit bands (l1, l2, max). Normalization reduces the chance of overfitting the model and minimises structural risk.

- PCA (Principle Component Analysis)

PCA is essentially dimensionality reduction, i.e. mapping a high dimensional one to a low dimensional one by a certain ratio. It can retain dimensional features that contain most of the variance and ignore features that have almost zero variance. This concentrates the data and increases the speed of data processing.

Logistic Regression

Logistic regression can be understood as linear regression with the involvement of a non-linear function and is often used to solve classification problems. It uses the idea of linear regression to fit the approximation bound (sigmoid function, 0-1). Estimate the result by maximum likelihood estimation, with a result of 1 when the result is > 0.5, otherwise the result will be calculated as 0 [1].

SVC (Support Vector Classification) & Linear SVC

Both SVC and Linear SVC are linear models that use decision boundaries (or hyperplanes) to partition the two types of data. Both SVC and Linear SVC consider only the points on the classification plane, and adding or subtracting any points has no effect on the results [2].

Methods

In the Pre-processing phase, I split the training/validation set into four steps: split up the training/validation set, handling with missing values (NaN), normalisation and combine the data set. Firstly, I extracted 15% of the data from each of training1 and training2 as the validation set, in order to estimate the accuracy of the model by predicting the validation set at the end of the experiment.

Then I used scikit learn's KNN Imputer to achieve the missing values. Manual NaN patching has also been tried (inserting the mean of each feature), but the KNN Imputer was superior in terms of both running time and effects. The normalise function of scikit learn was used to re-scale the data. Finally, the training data from training1 and training2 were combined for subsequent training.

After that, I performed feature selection, and model selection. For feature selection, I extracted different subsets of features: CNN features only (4096), gist features only (512), and confidence = 1 only (about 900), and planned to train them separately before comparing the final results. For model selection, I tried scikit learn's Logistic Regression, SVC and Linear SVC, and analysed the results by means of cross-validation (sklearn.model_selection.cross_val_predict).

For the training and testing part of the model, I use PCA before training and afterwards use GridSearchCV to verify the performance of the pipeline through iterations. Finally training, and then estimating the accuracy of the model again by predicting the validation set. The results are plotted in the form of a graph.

Results & Discussion

As can be seen from the graph, after training with the full training set, logistic regression is the most accurate at 72%, while SVC and Linear SVC are only less than 50% accurate.

Regarding the different feature subsets (CNN only, gist only, confidence = 1 only), the accuracy of cross-validation was 74%, 67% and 78% respectively. The actual prediction accuracies on the validation set were 70%, 64%, and 71%. In addition Linear SVC's scored 69% at cross-validation, which is higher than the actual prediction accuracy.

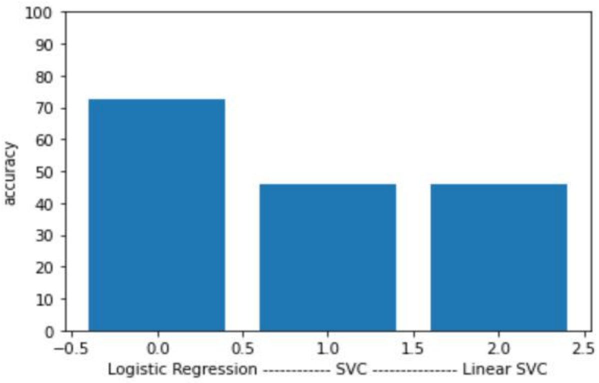

Fig 1. Comparison of different models

When choosing a norm for normalise, I chose L2 rather than L1, because L1 makes the optimal value of many parameters 0, leading to homogenisation. L2, on the other hand, makes the value smaller (near 0) but smoother (not equal to 0). [4]

For both CNN and gist features, the accuracy of the model after training CNN features is higher than that after training gist features in terms of results when cross-validating and predicting the validation set. It can be inferred from this that the CNN features relatively have a greater weighting. When compared to the model trained on the full training set (74% accuracy), it can be concluded that the CNN dominates the accuracy of the model due to its numerical superiority firstly, which is why the model is less accurate when only gist features are trained. But gist is also a low-dimensional signature vector for scenes, which allows for fast scene recognition and classification [5]. CNN features, on the other hand, lack an understanding of the overall space due to their displacement invariance [3]. Therefore, gist features play a greater role than CNN features in the "if the image is memorable" problem.

For the model that trained features with confidence=1, the overall performance was not as good as the model that trained the completed training data. Although confidence=1 obtained a high accuracy (78%) during cross-validation, it was unable to demonstrate good performance against real tests due to the limited validation set and the possible risk of overfitting.

Different SVCs (SVC and Linear SVC) were used to test the results after training. the difference between SVC and Linear SVC is mainly in the loss function, SVC uses hinge loss while Linear SVC uses squared hinge loss. But the two are extremely similar in the result of this problem. The reason for this is that the dimensionality of the dataset is not very high, and because SVC has its own internal penalised normalisation [2], which leads to repeated normalisation operations and over-flattening of the data, the results are not expected.

The present experiments on the hyper-parameters of the classifiers left much to be desired. The results would have been theoretically better if the preprocessing had been done specifically for the characteristics of the models of SVC.

References

1. Menard, S. W. (2010). *Logistic regression : from introductory to advanced concepts and applications. Thousand Oaks, Calif. ;: SAGE.* https://methods-sagepub-com.ezproxy.sussex.ac.uk/book/logistic-regression-from-introductory-to-advanced-concepts-and-applications
2. Gu, B., Wang, J.-D., Zheng, G.-S., & Yu, Y.-C. (2012). *Regularization Path for \nu -Support Vector Classification. IEEE Transaction on Neural Networks and Learning Systems*, 23(5), 800–811. https://doi.org/10.1109/TNNLS.2012.2183644
3. Roy, S. K., Krishna, G., Dubey, S. R., & Chaudhuri, B. B. (2020). *HybridSN: Exploring 3-D-2-D CNN Feature Hierarchy for Hyperspectral Image Classification.* IEEE Geoscience and Remote Sensing Letters, 17(2), 277–281. https://doi.org/10.1109/LGRS.2019.2918719
4. Wang, D. (2018). *Some further thoughts about spectral kurtosis, spectral L2/L1 norm, spectral smoothness index and spectral Gini index for characterizing repetitive transients*. Mechanical Systems and Signal Processing, 108, 360–368. https://doi.org/10.1016/j.ymssp.2018.02.034
5. Zhicheng Li, & Itti, L. (2011). *Saliency and Gist Features for Target Detection in Satellite Images. IEEE Transactions on Image Processing*, 20(7), 2017–2029. https://doi.org/10.1109/TIP.2010.2099128