Chapter 3: Classification

Warren Alphonso

March 23, 2019

Contents

1 Training a Binary Classifier

We begin by creating a classifier that can only identify one digit. A binary classifier distinguishes between just two classes, 5 and not 5.

The following creates target vectors for this task:

```
y_train_5 = (y_train == 5)
y_test_5 = (y_test == 5)
```

Now, we pick Stochastic Gradient Descent as the classifier and train it. SGD can handle very large datasets efficiently.

2 Performance Measures

One measure is cross-validation. We can use `cross_val_score()` function evaluates SGDClassifier model using K-fold cross-validation, with 3 folds. K-fold cross-validation means splitting training set into K-folds, then making predictions and evaluating them on each fold using a model trained on the remaining folds.

We get a 95% accuracy! This isn't that great actually because only 10% of images are 5's so always guessing an image is *not* a 5 gets us 90% accuracy. Accuracy is generally not the preferred performance measure for classifiers.

3 Confusion Matrix

The general idea is to count the number of times instances of class A are classified as class B. Each row in a confusion matrix represents an *actual* class, while each column represents a *predicted* class.

If you prefer a more concise metric, we can look at the precision of the classifier, which is defined as:

$$precision = \frac{TP}{TP + FP}$$

where TP is number of true positives and FP is number of false positives.

However, we can say we have perfect precision by only making a single positive prediction so typically we also use recall, aka sensitivity or true positive rate, which is defined as:

$$recall = \frac{TP}{TP + FN}$$

where FN is the number of false negatives.

4 Precision and Recall

We usually combine precision and recall into a single metric called F_1. The F_1 score is a *harmonic mean* of precision and recall so it gives much more weight to low values - the classifier will only get a high F_1 if *both* recall and precision are high. Logically, increasing precision reduces recall , and vice versa.

5 The ROC Curve

The *receiver operating characteristic* curve is common with binary classifiers. It plots the true positive rate (aka recall) against the false positive rate. The FPR is the ratio of negative instances that are incorrectly classified as positive, which is equal to 1 - true negative rate. TNR is also known as specificity. The ROC plots sensitivity (recall) versus 1 - specificity.

One way to compare classifiers is to measure the area under the curve. A perfect classifier will have a ROC AUC equal to 1, while a random classifier will have a ROC AUC equal to 0.5.

6 Error Analysis

Most misclassified images seem like obvious erros to us. The reason is that we used a simple `SGDClassifier` which is a linear model. It jsut assigns weight per class to each pixel and sums those weighted pixels to get a score. Since 3s and 5s differ only by a few pixels, the model easily confuses them.

Since the main difference between 3s and 5s is the line that joins the top line to bottom arc, one way to reduce this confusion error woudl be to preprocess images to center them and ensure they are not rotated.