**Introduction**

Caltech Pedestrian dataset consists of 11 subsets of videos, the first 6 for training and the last 5 for testing. The videos are taken from a vehicle driving in urban areas, and very 30 th frame is used. The data of the project is obtained from the Caltech pedestrian dataset including a training set (3605 positive samples and 10055 negative samples) and a test set (2043 positive samples and 4832 negative samples). A 2330-dimensional Haar-like feature was extracted from each image patch. The purpose of the project is to to develop classifiers, which take input features and predict the labels, compute the precision and recall values of the classifier and find out the best result. In this project, I have carried out both direct experiments and cross-validation experiments to find out best result. Besides, I have experimented K-NN, Decision Trees, Random Forest and SVM to find out the best algorithm with best parameters.

**Classification Models**

**Support vector machine**

In my opinion, SVM is the most suitable solution for this issue, because it is seen by many as the most successful current classification method. It is good generalization in theory and always work well with few training instance and high dimension datasets which as required by this project. In this project, I have verified that SVM can get the best results and choose it as the classification models. Support vector machines (SVM) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. To verify SVM, I have performed several sets of comparative experiments between SVM, KNN, Decision Tree and Random Forest and the results have shown that SVM has had better performance than other algorithms.

**Contrast experimental algorithm**

**1. k-nearest neighbors algorithm (k-NN)**

The k-nearest neighbors algorithm (k-NN) is a non-parametric method used for classification. The input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification.

**2. Decision Trees**

A decision tree is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements.A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). The paths from root to leaf represent classification rules.

**3. Random Forests**

Random forests is an ensemble learning method for classification, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes of the individual trees. Random forest is like bootstrapping algorithm with Decision tree (CART) model. It tries to build multiple CART model with different sample and different initial variables. Random forests correct for decision trees' habit of over-fitting to their training set.

**Experiments**

The experiments consist of 3 parts. Part 1 is direct experiments to verify the performances of different algorithms in the normalized data and part 2 is the direct experiments in the sampling training datasets. Part 3 is cross-validation experiments in Equal Proportions Datasets.

**1. Direct Experiments on Normalized Dataset**

In this section, I have carried out the experiments of SVM, Random Forest, Decision Tree and KNN. The result and the analysis can be seen as follows. Firstly, I have experimented to find out the best parameters of Random Forest.

Fig. 1. Number of trees.

As the Fig. 1 shows, the parameter of number of trees take effect in precision, recall and f1 score. When the value of the number of trees is 60, Random Forest take the best result. Thus, I have set the number of trees to 60 in order to find out the best value of the minimum number of samples required to split an internal node and the best value of the minimum number of samples required to be at a leaf node.

Fig. 2. The minimum number of samples required to split an internal node.

Fig. 3. The minimum number of samples required to be at a leaf node.

As we can see in Fig. 2 and Fig. 3, both parameters have little effect on the results. Therefore, I get the best parameters of Random Forest, where the best value of the number of trees is 60, the best value of the minimum number of samples required to split an internal node is 2 and the best value of the minimum number of samples required to be at a leaf node is 10.Then, I have experimented the KNN to obtain the best neighbor number.

Fig. 4. Number of neighbor

As shown in Fig. 4, the results of KNN have little effect in precision. However, when the neighbor number is 5, I get best result of recall and f1 score, even though they are not well. Besides, I have carry out the experiments of Decision Tree which have took the result as the value of precision is 68.83%, the value of recall is 52.86% which is the only evaluation criteria that better than Random Forest, and the value of f1 score is 59.80%. Finally, I have carried out the experiments of SVM, it get the results that precision is 94.76%, recall is 69.02% and f1 score is 79.86%. Thus, the best result of SVM, Random Forest, Decision Tree and KNN is shown in Fig. 5.

Fig. 5. Best result of different algorithms

In this project, Fig. 5 shows that SVM can get better result in all evaluation indicators than other three algorithms. Therefore, the results prove that SVM is a great method to address this project.

**2. Direct Experiments on Sampling Training Datasets**

In this part, I have sampled several groups of dataset which are sampling by random generator. The first group of datasets are the original dataset with normalized processing called Initial Datasets. The second group of datasets are about 3605 positive samples and 7210 negative samples which is the close proportion with the test dataset called Equal Proportions Datasets. The last group of datasets are about 2043 positive samples and 4832 negative samples which are the same as the test dataset. Both groups of datasets contain several types of dimension number called Equal Amount Datasets, including full dimension (2330), 2000, 1500, 1000, where the last three types are obtained by using PCA to reduce the dimension.

a. Experiments on Initial Datasets.

TableⅠ Contrast Results in Initial Datasets

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | Evaluation Indicators | 2330 | 2000 | 1500 | 1000 |
| SVM | Precision | 94.76% | 95.42% | 95.90% | 96.47% |
| Recall | 69.02% | 67.30% | 63.04% | 53.55% |
| F1 score | 79.86% | 78.93% | 76.08% | 68.88% |
| Random Forest | Precision | 93.01% | 44.11% | 49.90% | 54.19% |
| Recall | 46.26% | 22.91% | 24.42% | 24.38% |
| F1 score | 61.78% | 30.15% | 32.80% | 33.63% |
| Decision Tree | Precision | 68.83% | 52.40% | 52.90% | 52.38% |
| Recall | 52.86% | 35.34% | 36.22% | 35.00% |
| F1 score | 59.80% | 42.21% | 43.00% | 41.96% |
| KNN | Precision | 76.13% | 76.13% | 76.13% | 76.12% |
| Recall | 33.09% | 30.09% | 33.09% | 33.24% |
| F1 score | 46.13% | 46.13% | 46.13% | 46.27% |

The results of these experiments is shown in table Ⅰ. From the results, we can obtain that SVM have better performances than other algorithms in different types of dimension number while the result is stable. Besides, the dimensionality reductions of data have a great influence on SVM, Random Forest and Decision Tree which make the result of the experiment worse. However, KNN has a stable consequence in all these experiments.

b. Experiments on Equal Proportions Datasets

TableⅡ Contrast Results in Equal Proportions Datasets

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | Evaluation Indicators | 2330 | 2000 | 1500 | 1000 |
| SVM | Precision | 89.80% | 92.24% | 92.63% | 93.58% |
| Recall | 80.61% | 76.75% | 73.23% | 64.95% |
| F1 score | 84.96% | 83.78% | 81.79% | 76.68% |
| Random Forest | Precision | 86.83% | 49.74% | 51.78% | 65.61% |
| Recall | 69.40% | 47.72% | 42.68% | 41.56% |
| F1 score | 77.15% | 48.71% | 46.79% | 50.88% |
| Decision Tree | Precision | 57.03% | 48.41% | 45.96% | 50.58% |
| Recall | 60.00% | 46.89% | 43.76% | 47.14% |
| F1 score | 58.46% | 47.64% | 44.83% | 48.80% |
| KNN | Precision | 73.95% | 73.19% | 73.23% | 72.06% |
| Recall | 48.07% | 45.42% | 44.73% | 43.56% |
| F1 score | 58.265 | 56.06% | 55.55% | 54.30% |

Table Ⅱ shows that reasonable sampling is beneficial to improving the performances of the algorithms. As we can see, the samples are good for SVM in improving the recall and F1 score but bad for precision. But in general, the improvement of performance is greater than the reducing part. This result also appears in Random Forest and KNN except Decision Tree. In addition, the dimensionality reductions of data also have poor performance on SVM, Random Forest and Decision Tree.

c. Experiments on Equal Amount Datasets

Table Ⅲ Contrast Results in Equal Amount Datasets

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | Evaluation Indicators | 2330 | 2000 | 1500 | 1000 |
| SVM | Precision | 93.04% | 93.38% | 92.69% | 94.22% |
| Recall | 69.36% | 69.11% | 65.83% | 55.02% |
| F1 score | 79.47% | 79.44% | 76.99% | 69.47% |
| Random Forest | Precision | 91.18% | 39.56% | 46.04% | 44.56% |
| Recall | 50.61% | 28.39% | 33.92% | 28.24% |
| F1 score | 65.09% | 33.06% | 39.26% | 34.57% |
| Decision Tree | Precision | 61.72% | 42.58% | 44.60% | 45.83% |
| Recall | 54.77% | 38.33% | 41.80% | 37.10% |
| F1 score | 58.04% | 40.34% | 43.15% | 41.01% |
| KNN | Precision | 74.34% | 74.59% | 71.93% | 69.16% |
| Recall | 40.43% | 38.08% | 42.78% | 37.98% |
| F1 score | 52.38% | 50.42% | 53.65% | 49.04% |

As shown in Table Ⅲ, sampling with a small amount volume may reduce the performance of the algorithms. Though in original dimensions, all algorithms can get better performances in precision, but they all have worse effects in other evaluation indicators and other dataset of dimensionality reductions than the second group of experiments.

In summary, from above three groups of experiments, reasonable sampling can improve the performance while too little sampling may be counterproductive. Therefore, I select the second group of datasets to carry out the cross-validation experiments.

**3. Cross-validation Experiments on Normalized Sampling Datasets**

In this part, I have carried out two types of cross-validation. The first one partitions data into 5 parts and the other partitions data into 2 parts to do cross-validation experiment, select the best classifier and verify the test dataset.

a. Equal Proportions Dataset with 2330 Dimensions

Table Ⅳ Contrast Results in 2330 Dimensions of cross-validation

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Evaluation Indicators | Best in 2 classifier | Best in 5 classifier |
| SVM | Precision | 95.62% | 97.00% |
| Recall | 91.35% | 95.56% |
| F1 score | 93.43% | 96.27% |
| Random Forest | Precision | 96.86% | 97.57% |
| Recall | 86.78% | 93.45% |
| F1 score | 91.54% | 95.46% |
| Decision Tree | Precision | 77.94% | 85.75% |
| Recall | 76.31% | 82.66% |
| F1 score | 77.12% | 84.18% |
| KNN | Precision | 96.01% | 97.67% |
| Recall | 88.87% | 97.67% |
| F1 score | 92.30% | 97.65% |

b.Equal Proportions Dataset with 2000 Dimensions

Table Ⅴ Contrast Results in 2000 Dimensions of cross-validation

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Evaluation Indicators | Best in 2 classifier | Best in 5 classifier |
| SVM | Precision | 96.02% | 98.55% |
| Recall | 91.73% | 95.35% |
| F1 score | 93.83% | 96.93% |
| Random Forest | Precision | 79.82% | 87.34% |
| Recall | 49.27% | 64.19% |
| F1 score | 60.93% | 73.99% |
| Decision Tree | Precision | 71.02% | 81.19% |
| Recall | 69.53% | 76.28% |
| F1 score | 70.27% | 78.66% |
| KNN | Precision | 94.73% | 96.36% |
| Recall | 89.09% | 98.60% |
| F1 score | 91.82% | 97.47% |

c. Equal Proportions Dataset with 1500 Dimensions

Table Ⅵ Contrast Results in 1500 Dimensions of cross-validation

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Evaluation Indicators | Best in 2 classifier | Best in 5 classifier |
| SVM | Precision | 97.12% | 99.52% |
| Recall | 92.78% | 96.74% |
| F1 score | 94.90% | 98.11% |
| Random Forest | Precision | 83.01% | 82.84% |
| Recall | 50.78% | 65.12% |
| F1 score | 63.02% | 72.92% |
| Decision Tree | Precision | 71.35% | 86.70% |
| Recall | 70.11% | 78.75% |
| F1 score | 70.72% | 82.53% |
| KNN | Precision | 94.81% | 98.15% |
| Recall | 90.84% | 98.60% |
| F1 score | 92.78% | 98.38% |

d. Equal Proportions Dataset with 1000 Dimensions

Table Ⅶ Contrast Results in 1000 Dimensions of cross-validation

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Evaluation Indicators | Best in 2 classifier | Best in 5 classifier |
| SVM | Precision | 97.18% | 98.56% |
| Recall | 89.31% | 95.81% |
| F1 score | 93.08% | 97.17% |
| Random Forest | Precision | 89.95% | 92.64% |
| Recall | 54.97% | 70.23% |
| F1 score | 68.24% | 79.89% |
| Decision Tree | Precision | 73.08% | 80.36% |
| Recall | 69.42% | 76.11% |
| F1 score | 71.21% | 78.18% |
| KNN | Precision | 94.88% | 98.32% |
| Recall | 90.00% | 97.50% |
| F1 score | 92.37% | 97.91% |

From the above classifiers, we can know that every algorithm can get great classifiers in cross-validation. Thus, I have selected the best classifier of different algorithms from the above results to verity its performance in the test dataset with same number of dimension it had.

f. Verification of classifiers from cross-validation

Fig. 6. Best result of different algorithms

The results in Fig. 6 show that SVM can get the best performance in this project utilizing the best classifiers from the cross-validation with five classifiers that gets 90.11% in precision, 80.71% in recall and 85.15% in f1 score which is better than other algorithms and the results in table Ⅱ. In addition, all the results are obtained from Equal Proportions Datasets with 2330 dimensions. It means that dimensionality reduction may not be a great solution in the project.

**Conclusion**

In the project, I have verity that SVM has the best performance to address the issue where Random Forest, Decision Tree and KNN are utilized for contrast experimental study. Eventually, I have got the performances of SVM with 90.11% in precision, 80.71% in recall and 85.15% in f1 score. In order to get the consequence, I have sampled 3605 positive samples and 7210 negative samples from the train datasets and normalized them. Then, I have used the cross-validation with five classifiers to obtain the best classifier and have utilized it to get the best performance of this project. The parameters of the final result is SVC(C=1.0, kernel='rbf', degree=3, gamma='auto', coef0=0.0,shrinking=True, probability=False, tol=0.001, cache\_size=200, class\_weight=None, verbose=False, max\_iter=-1, decision\_function\_shape='ovr', random\_state=None). For future works, the results I have obtained may not be the best for the project and I will continue to experiment to find the best results.