

# Title

xxx<sup>1</sup>, yy<sup>1</sup>, and zzz<sup>1</sup>

<sup>1</sup>Faculty of Mathematics and Information Science, Warsaw University of Technology  
ul. Koszykowa 75, 00-662 Warsaw, Poland  
{xxx,yy,zzz}@mini.pw.edu.pl

**Abstract.** In the article we present

## 1 Introduction

AJ

## 2 Literature Review

AJ

## 3 Preliminaries

### 3.1 The Task of Classification with Rejection

AJ

### 3.2 Regression

AZ, PW Regression analysis is a statistical process for estimating the relationships between variables. It includes many techniques for modeling and analyzing several variables, when the focus is on the relationship between a dependent variable and one or more independent variables. The estimation target is a function of the independent variables called the regression function. There are many different regression models: linear regression, polynomial regression, logistic regression, Bayesian Ridge regression, etc. In this paper we will focus mainly on two regression variants: logistic regression and polynomial one (please note that linear regression is in fact polynomial regression using polynomial of degree 2).

Logistic regression relies on the assumption that the independent variables are dichotomous (it means that their values are either 0 or 1), in other words they describe presence or absence of certain facts. In this model, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function. The received answer is evaluated using maximum likelihood method. Predicted values are probabilities and are therefore restricted to (0,1) through

the logistic distribution function.

Polynomial regression can be viewed as a model that uses linear regression extended by constructing polynomial features from the coefficients. This approach maintains the generally fast performance of linear methods, while allowing them to fit a much wider range of data. The predictors resulting from the polynomial expansion of the "baseline" predictors are known as interaction features, and can be used in classification problem. Just like linear regression, polynomial regression models are usually fit using the method of least squares.

### 3.3 Rejection Mechanism Based on Regression

**AZ, PW** Calculations presented in this paper were conducted on a mixed set consisting of letters and digits. Because the aim of our work was to provide classifier-based solution for classifying digits and rejecting other symbols, we decided to compare three different approaches towards this problem. All of them based on the assumption that there is no information regarding outliers during classifier training.

As the first way to deal with presented issue "one-versus-all" method was prepared. This approach requires creating a vector of classifiers constructed in a specific way. Each classifier has to be trained on a specially prepared training data, consisting of two sets: the first one (denoted as class\_1) holding all training data entries for certain native class, and the second one (denoted as class\_2) being the result of a subset sum operation performed on the rest of the classes except for the class used in class\_1 set. Please note that both class\_1 and class\_2 sets should have the same size, so it is advisable to randomly choose elements when creating class\_2. One classifier for each native class has to be present in a final vector. The actual classification with rejection is performed by presenting the unknown pattern to each of the classifiers from the vector. When the classifier recognizes element as a native one (belonging to class\_1) then the pattern is treated as a recognized, and classified native element. In case of all classifiers rejecting such pattern (classifying it as element from class\_2), it is treated as outlier and rejected. It is also worth noting that there is a possibility of more than one classifier recognizing the pattern as a native element. In such case randomly chosen class is assigned to this pattern.

The second approach uses "one-versus-one" method. Similarly to the previous one it requires preparing vector of classifiers, but this time it consists of  $\binom{n}{2}$  classifiers, where  $n$  is the number of unique native classes. Each classifier is trained on data consisting of two sets: the first one (denoted as class\_1) holding all training data entries for certain native class, and the second one (denoted as class\_2) holding all training data entries for some other class (not the same as class\_1). In the end there is one classifier for each pair of classes: 1 vs. 2, 1 vs. 3, ..., 1 vs.  $n$ , ...,  $(n-1)$  vs.  $n$ . Classification with rejection mechanism is based on

presenting unknown pattern to every classifier in the vector and remembering their answers (e.g. classifier constructed for 1 vs. n classes can classify pattern as belonging to class 1 or class n). In the end those answers can be presented as a n-wide array with each element being the number of times pattern was classified as belonging to certain class. The pattern is rejected when difference between two biggest values in the result array is smaller than two. In such case it is assumed that classifiers were uncertain as to which class should this unknown element belong to. Otherwise the pattern is classified as an element from the class which had the biggest value in the result array. [tutaj mona doda schemat prezentujcy opisan koncepcj](#)

The last prepared and examined method, presented in this work, bases on the way of constructing classifiers vector used in the second approach ("one-versus-one"). The difference between those two methods lies in a rejection mechanism. In this method an unknown pattern is treated as a foreign element if its biggest value in the result array is lesser than (n-1). What it actually means is that there must be a certain class that has always been chosen by a classifier from the vector. [tutaj mona doda schemat prezentujcy opisan koncepcj](#)

### 3.4 Quality Evaluation

[A.J](#)

- CC (Correctly Classified) - the number of correctly classified patterns, i.e. native patterns classified as native ones with the correct class,
- TP (True Positives) - the number of native patterns classified as native (no matter, into which native class),
- FN (False Negatives) - the number of native patterns incorrectly classified as foreign,
- FP (False Positives) - the number of foreign patterns incorrectly classified as native,
- TN (True Negatives) - the number of foreign patterns correctly classified as foreign.

## 4 Experiments

### 4.1 Presentation of Datasets

[A.J](#)

We present a study on handwritten digits recognition and handwritten letters (from the Latin alphabet) rejection. In other words, native patterns set is made of digits (it is a ten class problem), while foreign patterns are 26 different handwritten letters. The justification to assume such foreign dataset for testing

00112233445566778899  
00112233445566778899

Fig. 1. ...

**Table 1.** Results for classification without rejection on train and test sets of native patterns in comparison with classification results with rejection mechanism.

|  |              |                |
|--|--------------|----------------|
|  | no rejection | with rejection |
|  |              |                |

purposes is that appearance of other real-world symbols, but not belonging to any proper class, is a common issue in a character recognition problem.

We would like to stress again, that foreign patterns do not participate in the model building phase. The entire scheme is based on native patterns only. Handwritten letters are used only for rejection mechanisms quality evaluation. Samples of processed patterns are displayed in Figure 1.

The native training dataset consisted of 10,000 handwritten digits with approximately 1,000 observations in each class taken from publicly available MNIST database, [2]. We split each class in proportion ca. 7:3 and as a result we got two sets. The first one includes 6,996 patterns and is used for training. The second set, the test set, contains 3,004 patterns. The dataset of foreign patterns contains 26,383 handwritten Latin letters, ca. 1,000 letters in each class. This dataset was created by 16 students, writing about 70 copies of each letter.

All patterns were normalized and feature vectors comprising of 106 numerical features were created. Examples of features are: maximum/position of maximum values of projections, histograms of projections, transitions, offsets; raw moments, central moments, Euler numbers etc. The best first search for the optimal feature subset has been performed using FSelector R package, [3] and then analysis of variance was employed to select good features. The final feature vector contained 24 elements. We considered features standardization but the training data is sufficiently consistent (there are no outliers), so we normalized those features to bring linearly all values into the range [0,1].

4.2 Impact on Classification

AZ, PW

4.3 Rejection Quality

AZ, PW

The results of our study is presented in Table 2. As it can be seen Polynomial Regression model achieved better accuracy in classifying native patterns

and was better at rejecting the foreign ones. It turned out that the third approach (Method 3) yielded the best results whereas the second one performed the worst. Method 2 was the one that "destroyed" classification capabilities of classifiers and had tendency to reject native patterns. On the other hand Method 1 accepted most of the foreign elements. Those results become more obvious if one look at them from the methods' construction point of view.

The first approach rejected input only after  $n$  failed tests which makes rejection less probable the more tests are performed (more native classes means more tests). Provided pattern was treated as native and labelled as an element from the first class that had its test passed. Because tests were done in order (from class 1 to class  $n$ ) we observed increased number of classified elements in the first few classes.

The second method required certain class to be obviously "stronger" than the others in terms of classification. All patterns that had similarities between two or more different classes were rejected which resulted in poor classification capabilities.

The third way provided some sort of equilibrium between classification and rejection by modifying method 2. Elements were rejected only when there was no strong evidence that they belong to certain class. This evidence was in fact the number of times pattern was assigned certain class' label. If there was no class with required number of labels the element was identified as foreign one.

#### 4.4 Experimental Settings

Solutions presented in this paper have been implemented in Python programming language, using scientific libraries [4, 5] and their implementations for logistic and polynomial regression. Several tests have been performed in order to find best suited method parameters for both classifiers. For finding those values the Grid Search [5] has been used.

**Table 2.** Results for classification with rejection when using only classifiers with three different approaches.

|                     | Method 1                   |          | Method 2   |          | Method 3   |          |
|---------------------|----------------------------|----------|------------|----------|------------|----------|
| Regression type     | Polynomial                 | Logistic | Polynomial | Logistic | Polynomial | Logistic |
| Data Set            | Native Patterns, Train Set |          |            |          |            |          |
| Strict Accuracy     | 0.137                      | 0.186    | 0.716      | 0.726    | 0.888      | 0.236    |
| Fine Accuracy       | 0.645                      | 0.731    | 1          | 0.897    | 1          | 0.926    |
| Strict Native Sens. | 0.645                      | 0.720    | 0.610      | 0.030    | 1          | 0.921    |
| Data Set            | Native Patterns, Test Set  |          |            |          |            |          |
| Strict Accuracy     | 0.064                      | 0.114    | 0.719      | 0.820    | 0.853      | 0.143    |
| Fine Accuracy       | 0.602                      | 0.738    | 0.969      | 0.882    | 0.959      | 0.925    |
| Strict Native Sens. | 0.601                      | 0.727    | 0.494      | 0.030    | 0.811      | 0.920    |

## 5 Conclusion

AJ Proposed ...  
In future ...

## Acknowledgment

The research is partially supported by the .....

## References

1. Hempstalk, K., Frank, E., Witten, I., *One-class classification by combining density and class probability estimation*, Machine Learning and Knowl. Disc. in Databases, pp. 505-519, 2008.
2. LeCun, Y., Cortes, C., and Burges, C., *The MNIST database of handwritten digits*, in: <http://yann.lecun.com/exdb/mnist>.
3. Romanski, P., Kotthoff, L., *Package FSelector*, <http://cran.r-project.org/web/packages/FSelector/FSelector.pdf>
4. <http://www.numpy.org/>
5. <http://scikit-learn.org/stable/>