__B659 Project Report__

TIGER POPULATION GROWTH PREDICTION

-Tasneem Alowaisheq and Devendra Singh Dhami

1. Abstract

The purpose of this project is to predict the change in the population of the tigers in 29 reserves of India in the duration of 4 years (2006 – 2010). The first phase of the project involves the collection of data from the Project Tiger website of Government of India [1]. We collect the data for different reserves for the years 2006 and 2010. The next phase of the project is the task of Feature Extraction / Dimensionality Reduction. The final phase of the project is aimed at using Machine Learning algorithms for the task of predicting the change in the Tiger population whether positive or Negative. We present our findings in the form of accuracy, precision and recall of our methods and compare how the methods fare against each other in the final section of our report.

2. Introduction

The dwindling population of the tigers in the world has been a topic of great worries. Currently, India has the highest population of these creatures and its presence in the Tiger reserves of India is their brightest chance of survival.

The Indian government carries out a survey every four years that counts the total number of tigers in the various reserves and also evaluates all the reserves on various criteria's. This evaluation forms the dataset for the project. In 2006 and 2010 all the counting was done using the 'paws' method that was not completely reliable. Thus the only way to support the population figures for a reserve was how well they did in the various criteria. Since we still not have the 2014 data at our disposal, where the counting has been done using sensors and camera traps, we went ahead with the data from only those two years for our project.

Before using the classification algorithms we needed to reduce the dimensionality of our dataset which was of the order 29 * 35 i.e. having 29 examples and 35 features for each example. We used several methods for extracting the best features needed for prediction.

We used the supervised learning methods like Logistic Regression and Boosting in the Machine Learning literature to try and get a system to make predictions from the dataset.

We then measure the accuracy, precision and recall of our various methods to compare the effectiveness of all in the task of classification.

3. Project Goal

The project has the following goals:

- Reduce Dimensionality.
- Binary Classification Task (Increase/Decrease).
- Comparison of accuracy, precision and recall of various methods

Reduce Dimensionality:

Given the complex dimensionality of our dataset where the number of examples are less than the number of features it becomes necessary to reduce the dimensions in order to get good predictions. Thus this becomes the first goal of our project. We implement 4 methods to achiever this

1. k – Nearest Neighbors.
2. PCA
3. LDA with PCA as the predecessor.
4. Feature Boosting (embedded in the boosting algorithm)

For all the 4 methods above repeated random sub-sampling validation is used (10 runs)

Binary Classification Task:

For the classification of the examples into positive (increase in population) and negative (decrease in population) classes we take the supervised learning approach and use 2 algorithms to achieve this

1. Logistic Regression.
2. Boosting, using decision stumps as the weak classifiers.

Comparison of accuracy, precision and recall:

We present the following graphs in the result section.

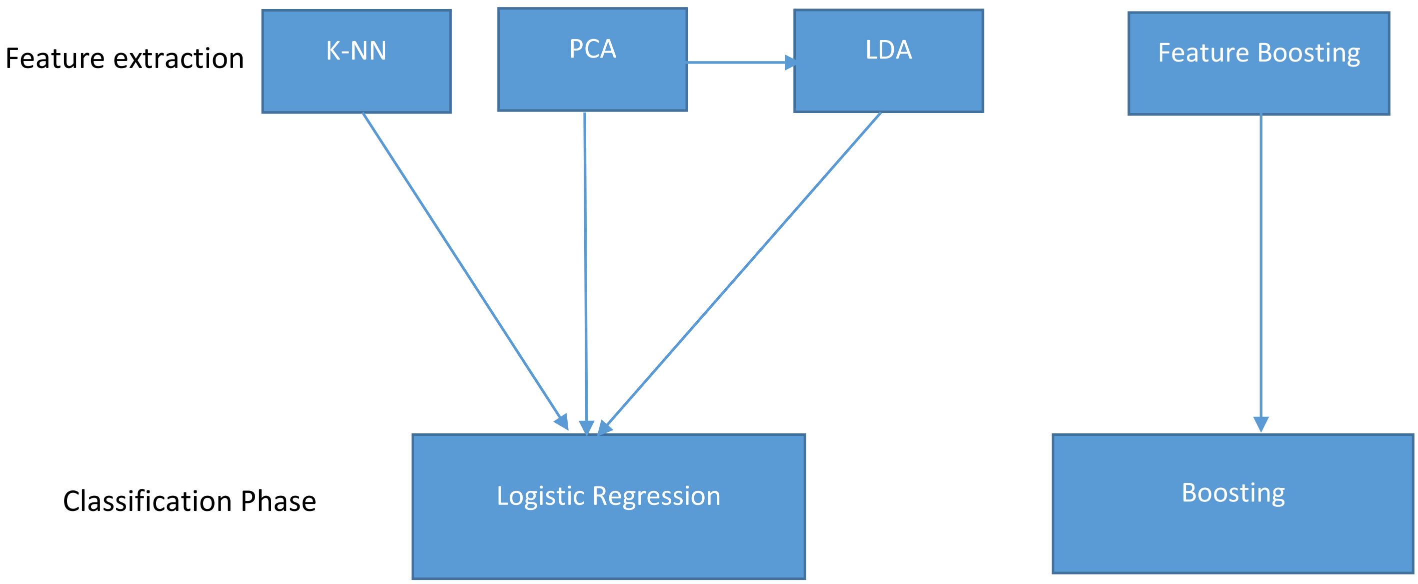
1. For K – NN and LR we compare the maximum and average accuracies, precision and recall for the most stable feature, the feature that decreases the most variance and the Logistic regression without feature extraction.

2. We compare the maximum and average accuracies, precision and recall for PCA as a standalone feature extractor and as a preprocessor for LDA.

3. We give the maximum and average accuracies, precision and recall for Boosting method

4. We plot the precision and recall for all the methods.

4. Methodology



K – NN as a feature extractor:

- Find the correlation between different features for 2 k values (3 and 5).
- Get the top 7 features that decrease the variance the most from k = 3 to k= 5.

PCA:

- PCA is used both as a stand-alone feature extractor and as a preprocessor to LDA.
- PCA as a preprocessor to LDA:
    - This is done to counter the Singular Matrix problem faced in LDA where the determinant of the matrix is zero and thus we cannot calculate the inverse of the ‘between class scatter matrix’.

5. Data Set Description

Here is a snapshot of our dataset.

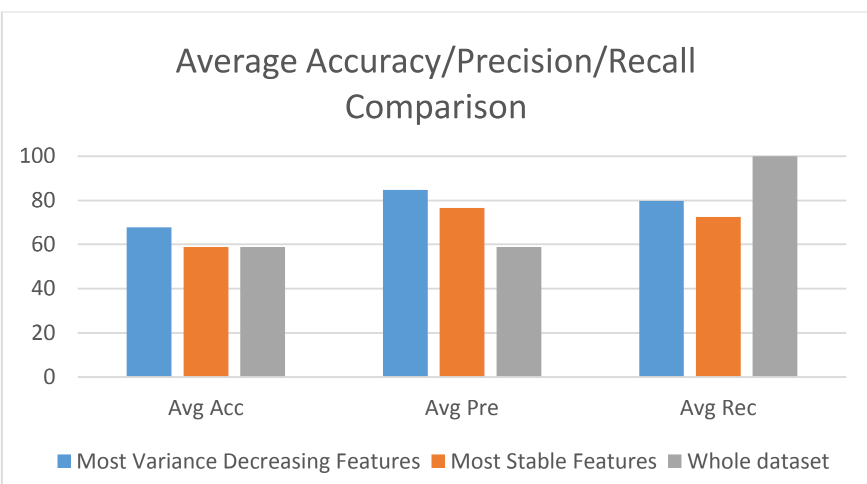
Ecodevelopment Activities	Villagers Intelliegence	Cattle Immunisation	Registration of Arms	Postmortem	Local People Trust	Control Over Use	Sustainability Efforts	Restorative Inputs	Increase
6	3	5	4	4	5	0	0	0	1
5	4	2	4	2	4	4	3	3	1
4	2	2	4	2	3	2	2	3	1
5	2	2	4	0	2	3	2	0	1
6	3	5	4	4	5	0	0	0	1
4.5	3	3.5	4	4	3.5	5	1.5	2.5	0
4	3	4	4	3	4	4	3	3	1
5	4	5	4	4	4	4	4	4	1
4	5	3	4	2	5	5	3	0	0
3	5	0	4	4	5	0	0	0	1
5	4	1	4	0	2	3	2	2	1
6	3	5	4	4	5	3	4	2	1
2	3	5	2	4	5	5	1	0	1
6	3	5	4	4	5	3	0	0	1
6	5	0	0	4	5	0	4	0	0
5	0	5	4	4	5	5	0	0	0
3	2	2	4	4	3	2	0	0	1

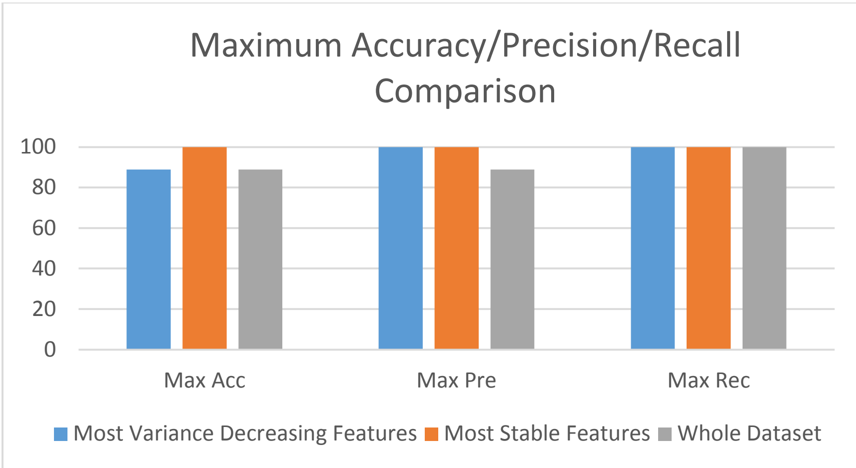
The characteristics of our datasets are:

Number of Examples	29
Number of Features	35
Number of Positive Examples	20
Number of Negative Examples	9
Number of Binary Features	4
Number of Non-Binary Features	31

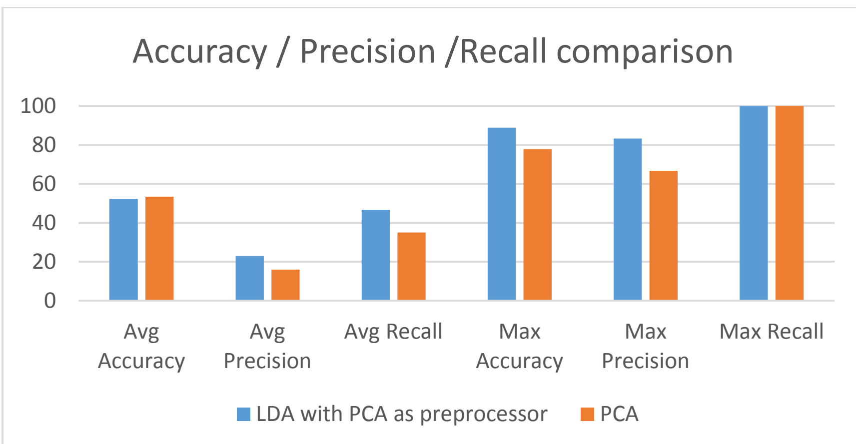
6. Results

<u>Logistic Regression with k-NN as feature Extractor:</u>

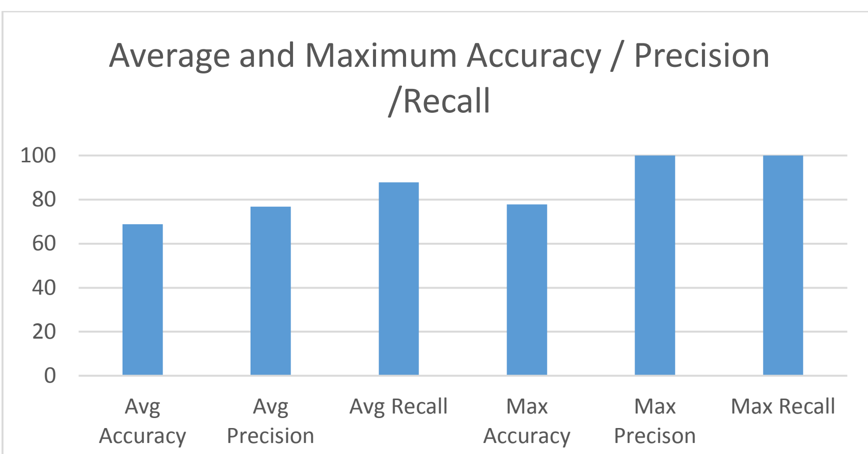




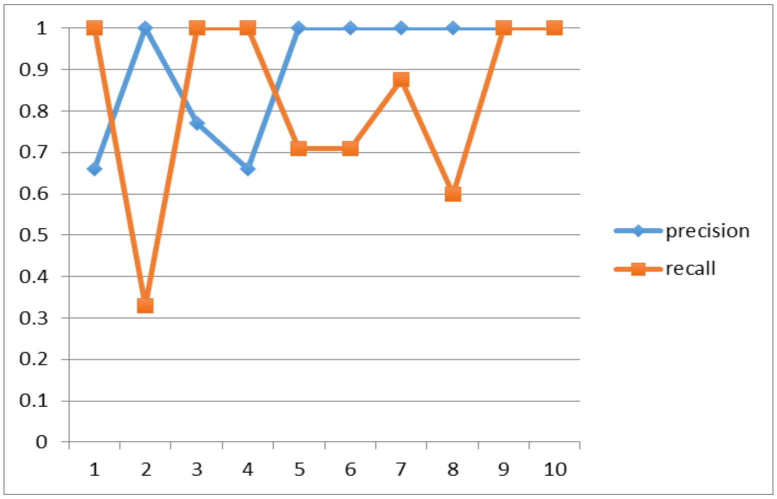
PCA (standalone) and LDA (with PCA):



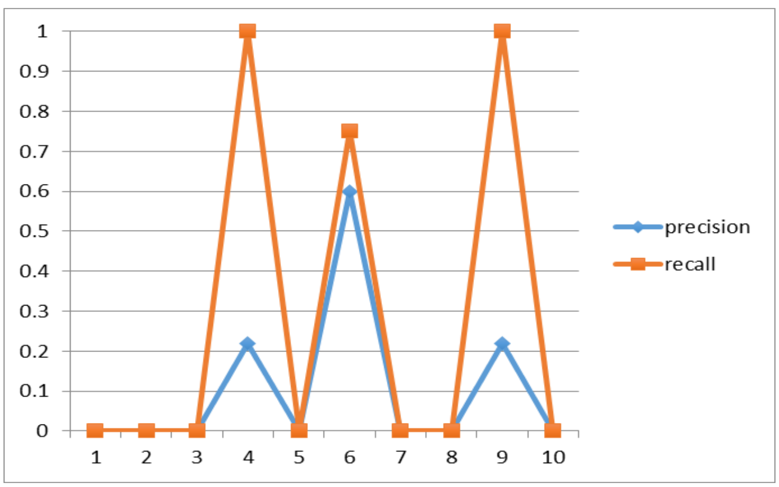
Boosting:



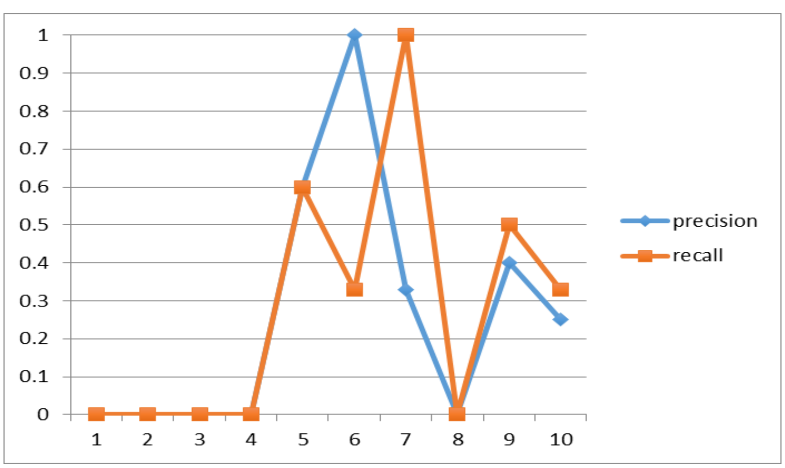
Precision / Recall graphs:



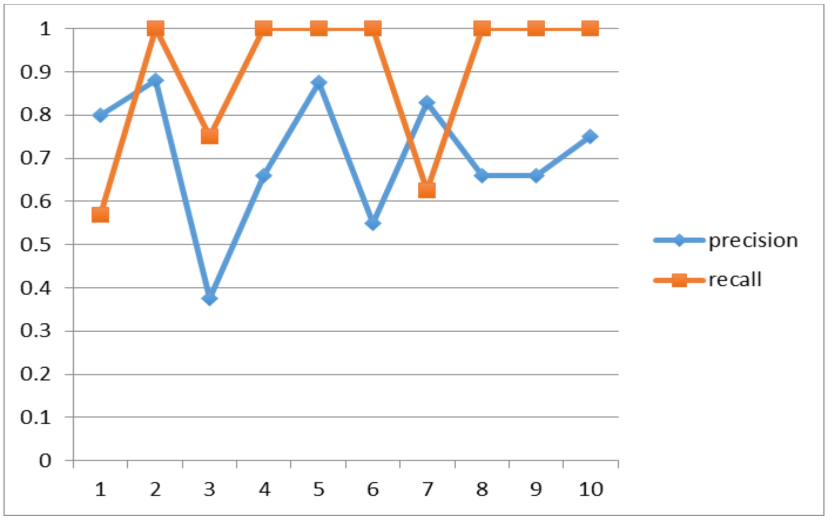
K - NN



LDA



PCA



Boosting

7. Analysis

We are using random sampling for all of our methods and running every algorithm 10 times.

For k – NN as a feature extractor with LR as the classifier we can see from the results that the 'Most Variance Decreasing' features give the best performance which is expected. When we increase the value of k from 3 to 5 we expect the variance to decrease and thus error from sensitivity to small fluctuations in the training set should decrease thereby giving us better features. This is what we can see from the results. When we used the most features we had kept the bias / variance tradeoff in mind as those features aren't having much effect on the variance and thus the bias of the dataset. But we record the accuracy and even precision and recall on the lower side as random sampling causes the training set to change for every iteration and thus increase the error. When we supply the whole dataset without feature extraction the difference in Accuracy and Precision can be seen clearly in the results as LR cannot decide on the relevant instances.

We believe that the reason why the average accuracy for the logistic regression classification using PCA is low is because of the following:

- The idea behind PCA is picking the dimensions with the highest variance, which does necessarily contain the most discriminant features.
- Our task is to find the most relevant features, not to reduce the dimensionality of the "already" relevant features (which PCA is typically used for).
- One of PCA data-preprocessing step is to exclude the class labels which will cause losing the discriminant properties of the data.

LDA (Linear Discriminant Analysis) has a similar concept to PCA, and since LDA includes the class labels within its calculations we thought that applying LDA on the data will perform better. However when we applied LDA on the raw data, we found that some of the 'between class scatter matrices' were singular matrices (where the determinant is zero). In order to overcome this problem, we used PCA as a preprocessing step before applying LDA as some papers suggested [2]. However classification using PCA already had low accuracy, so consequently classification LDA performed poorly too.

Boosting gives us good results as the weak classifier (decision stumps) serve as the feature extractor (Feature Boosting) which can be seen in high precision and recall values.

8. References

[1] http://projecttiger.nic.in/WriteReadData/userfiles/file/Report_EvaluationReportsofTRinIndia.pdf

[2] Nain, Neeta, et al. "Face recognition using pca and lda with singular value decomposition (svd) using 2dlda." Proceedings of the World Congress on Engineering. Vol. 1. 2008.