

Stay Alert! The Ford Challenge

—Kaggle 2011

Yuan, Jumao

03/31/2015

Contents

1	Abstract	2
2	Introduction	2
3	Data Description	2
4	Evaluation	3
5	Data Preprocessing	4
5.1	Missing values and typos	4
5.2	Remove redundant variables	4
5.3	Split into training and testing datasets	4
6	Build Moldes	5
6.1	GLM	5
6.2	Logistic Regression	5
6.3	3	5
6.4	4	5
6.5	5	5
6.6	6	5
7	Result	5
8	Discussion	5
9	References	5

1 Abstract

The “Stay Alert!” competition from Ford challenged competitors to predict whether a car driver was not alert based on various measured features.

The features were presented in 3 sets: physiological(P1 P8), environmental(E1 E11) and vehicular(V1 V11). Each feature was presented as a real number. For each measurement we were also told whether the driver was alert or not at that time(a boolean label called IsAlert).

We splitted data into 70% training data and 30% testing data. “Area under the curve”(AUC) was used as the accuracy assessment criteria.

We used “logistic regression(glmnet in R)”, “naive bayes”, “classification tree(tree in R)”, “random forest”and “SVM” models, and found “random forest” model works much better than others by comparison of AUC scores and ROC curves.

2 Introduction

Driving while distracted, fatigued or drowsy may lead to accidents. Activities that divert the driver’s attention from the road ahead, such as engaging in a conversation with other passengers in the car, making or receiving phone calls, sending or receiving text messages, eating while driving or events outside the car may cause driver distraction. Fatigue and drowsiness can result from driving long hours or from lack of sleep.

The objective of this challenge is to design a detector/classifier that will detect whether the driver is alert or not alert, employing any combination of vehicular, environmental and driver physiological data that are acquired while driving.

3 Data Description

We can download data from website <https://www.kaggle.com/c/stayalert/data>. The data for this challenge shows the results of a number of “trials”, each one representing about 2 minutes of sequential data that are recorded every 100 ms during a driving session on the road or in a driving simulator. The trials are samples from some 100 drivers of both genders, and of different ages and ethnic backgrounds. The files are structured as follows:

The first column is the Trial ID - each period of around 2 minutes of sequential data has a unique trial

ID. For instance, the first 1210 observations represent sequential observations every 100ms, and therefore all have the same trial ID

The second column is the observation number - this is a sequentially increasing number within one trial ID

The third column has a value X for each row where

- $X = 1$ if the driver is alert
- $X = 0$ if the driver is not alert

The next 8 columns with headers P1, P2 ,....., P8 represent physiological data;

The next 11 columns with headers E1, E2,....., E11 represent environmental data;

The next 11 columns with headers V1, V2,....., V11 represent vehicular data;

4 Evaluation

Entries will be evaluated using the area under the **receiver operator curve** (AUC). AUC was first used by the American army after the attack on Pearl Harbour, to detect Japanese aircraft from radar signals. Today, it is a commonly used evaluation method for binary choose problems, which involve classifying an instance as either positive or negative. Its main advantages over other evaluation methods, such as the simpler misclassification error, are:

1. It's insensitive to unbalanced datasets (datasets that have more installeds than not-installeds or vice versa).
2. For other evaluation methods, a user has to choose a cut-off point above which the target variable is part of the positive class (e.g. a logistic regression model returns any real number between 0 and 1 - the modeler might decide that predictions greater than 0.5 mean a positive class prediction while a prediction of less than 0.5 mean a negative class prediction). AUC evaluates entries at all cut-off points, giving better insight into how well the classifier is able to separate the two classes.

Understanding AUC

To understand the calculation of AUC, a few basic concepts must be introduced. For a binary choice prediction, there are four possible outcomes:

- true positive - a positive instance that is correctly classified as positive;
- false positive - a negative instance that is incorrectly classified as positive;
- true negative - a negative instance that is correctly classified as negative;
- false negative - a positive instance that is incorrectly classified as negative);

These possibilities can be neatly displayed in a confusion matrix:

	P	N
P	true positive	false positive
N	false positive	true positive

The true positive rate, or recall, is calculated as the number of true positives divided by the total number of positives. When identifying aircraft from radar signals, it is proportion that are correctly identified.

The false positive rate is calculated as the number of false positives divided by the total number of negatives. When identifying aircraft from radar signals, it is the rate of false alarms.

5 Data Preprocessing

5.1 Missing values and typos

Usually, we use data imputation to make up missing values; while, in this problem, not many missing values and typos exist, so we can just remove them.

5.2 Remove redundant variables

In the original dataset, we can see variables “P8”, “V7” and “V9” are redundant. Therefore, we can delete these 3 columns in the data preprocessing.

5.3 Split into training and testing datasets

We randomly split data into 70% test datasets and 30% training datasets and repeat all models with a few iterations.

6 Build Moldes

6.1 GLM

6.2 Logistic Regression

6.3 3

6.4 4

6.5 5

6.6 6

7 Result

8 Discussion

9 References

- [1] “An Introduction to Statistical Learning” by James, G., Witten, D., Hastie, T., Tibshirani, R.
- [2] <https://www.kaggle.com/c/stayalert>.
- [3] <https://www.kaggle.com/c/stayalert/data>.
- [4] “First place in the ‘Stay Alert!’ competiion”, inference, March *2011*.
- [5] “Stay Alert! The Ford Challenge”, L. Fourrier, F. Gaie, T. Rolf.
- [6] “Stay Alert! The Ford Challenge”, T. Tariq, A. Chen, Fall *2012*.
- [7] <http://stats.stackexchange.com/questions/76365/examples-for-one-class-svm-in-r>.
- [8] <https://www.kaggle.com/c/stayalert/forums>.
- [9] <http://en.wikibooks.org/wiki/LaTeX/Tables>.