Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

MACHINE LEARNING

XGBoost for Classification[Ca

By Sudhanshu Kumar on September 16, 2018

Boost Your ML skills with XGBoost

Introduction :

In this blog we will discuss one of the Popular Boo

XGBoost is the most popular machine learning alg
type (regression or classification), it is well known
algorithms.

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

Extreme Gradient Boosting (xgboost) is similar to
efficient. It has both linear model solver and tree l
is its capacity to do parallel computation on a sing

This makes xgboost at least 10 times faster than ex
It supports various objective functions, including r

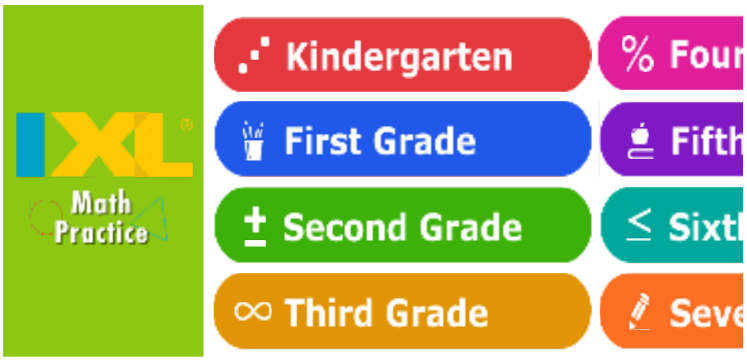
Since it is very high in predictive power but relativ
becomes an ideal fit for many competitions. It alsc
validation and finding important variables.

Idea of boosting

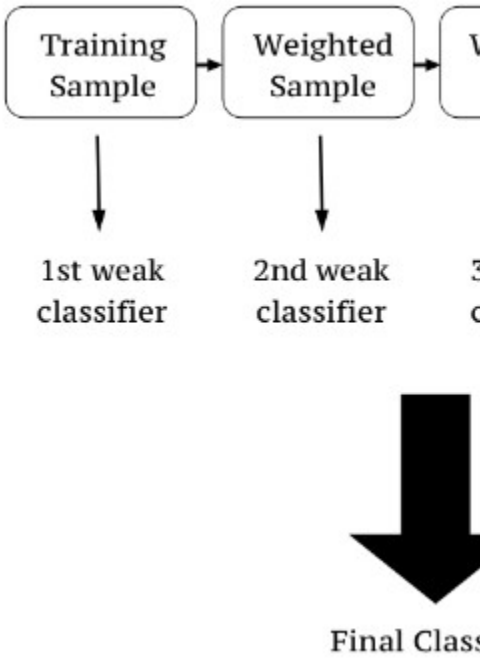
Let's start with intuitive definition of the concept:

Boosting (*Freud and Shapire, 1996*) – algorithm
reweighted versions of the training data. Classify

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]



When using boosting techinque all instance in dat
difficult to classify they are. In each following itera
(assign bigger weights) to instances that were wro



Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

In the first iteration all instance weights are equal.

Ensemble parameters are optimized in **stagewise** optimal parameters for the next classifier holding might sound like a limitation but turns out it's a ve model.

Pro's

- computational scalability,
- handling missing values,
- robust to outliers,
- does not require feature scalling,
- can deal with irrelevant inputs,
- interpretable (if small),
- can handle mixed predictors (quantitive and qu

Con's

- can't extract linear combination of features
- small predictive power (high variance)

Boosting techinque can try to reduce the variance (where each one is solving the same problem)

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

How XGBoost helps

The problem with most tree packages is that they
seriously – they allow to grow many very similar t
bushy.

GBT tries to approach this problem by adding som

- control tree structure (maximum depth, minim
- control learning rate (shrinkage),
- reduce variance by introducing randomness (st
  random subsamples of instances and features)

But it could be improved even further. Enter XGB

XGBoost (*extreme gradient boosting*) is a **more**
Boosted Trees.

It was develop by Tianqi Chen in C++ but also ena

The main advantages:

- good bias-variance (simple-predictive) trade-of
- great computation speed,
- package is evolving (author is willing to accept

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

XGBoost's objective function is a sum of a specific
predictions and a sum of regularization term for a

Mathematically, it can be represented as :

$$obj(\theta) = \sum_{i}^{n} l(y_i - \hat{y}$$

XGBoost handles only numeric variables.

Problem Statement :

To build a simple boosting classification model cal
the car given few of other car attributes.

Data details

==========

1. Title: Car Evaluation Database

==========

The dataset is available at "http://archive.ics.uci.e
2. Sources:
(a) Creator: Marko Bohanec
(b) Donors: Marko Bohanec (marko.bohanec@ijs
Blaz Zupan (blaz.zupan@ijs.si)
(c) Date: June, 1997

3. Past Usage:

The hierarchical decision model, from which this

M. Bohanec and V. Rajkovic: Knowledge acquisitio
decision making. In 8th Intl Workshop on Expert
Systems and their Applications, Avignon, France.

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

Within machine-learning, this dataset was used fo
Induction Tool), which was proved to be able to cc
hierarchical model. This, together with a comparis

B. Zupan, M. Bohanec, I. Bratko, J. Demsar: Mach
ICML-97, Nashville, TN. 1997 (to appear)

4. Relevant Information Paragraph:

Car Evaluation Database was derived from a simpl
developed for the demonstration of DEX
(M. Bohanec, V. Rajkovic: Expert system for decisi
1990.). The model evaluates
cars according to the following concept structure:

CAR	car acceptability
. PRICE	overall price
. . buying	buying price
. . maint	price of the maintenance
. TECH	technical characteristics
. . COMFORT	comfort
. . . doors	number of doors
. . . persons	capacity in terms of persons to c
. . . lug_boot	the size of luggage boot
. . safety	estimated safety of the car

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

Input attributes are printed in lowercase. Besides
includes three intermediate concepts:
PRICE, TECH, COMFORT. Every concept is in the
descendants by a set of examples.

The Car Evaluation Database contains examples w
i.e., directly relates CAR to the six input
attributes: buying, maint, doors, persons, lug_boo

Because of known underlying concept structure, th
testing constructive induction and
structure discovery methods.

5. Number of Instances: 1728
(instances completely cover the attribute space)

6. Number of Attributes: 6

7. Attribute Values:

buying v-high, high, med, low
maint v-high, high, med, low
doors 2, 3, 4, 5-more
persons 2, 4, more

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

lug_boot small, med, big
safety low, med, high

8. Missing Attribute Values: none

9. Class Distribution (number of instances per clas

class N N[%]
——————————
unacc 1210 (70.023 %)
acc 384 (22.222 %)
good 69 (3.993 %)
v-good 65 (3.762 %)

Tools to be used :
Numpy,pandas,scikit-learn

Python Implementation with code :

Import necessary libraries

Import the necessary modules from specific librar

```
import os
import numpy as np, pandas as pd
import matplotlib.pyplot as plt
from sklearn import  metrics, model_sel
from xgboost.sklearn import XGBClassifi
```

Load the data set

Use pandas module to read the bike data from the dataset.

```
data =
pd.read_csv('data/car_quality/car.data'
,'persons','lug_boot','safety','class']
```

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

```
data.head()
```

	buying	maint	doors	persons	lug_boo
0	vhigh	vhigh	2	low	unacc
1	vhigh	vhigh	2	unacc	
2	vhigh	vhigh	2	unacc	
3	vhigh	vhigh	2	low	unacc
4	vhigh	vhigh	2	unacc	

Check few information about the data set

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1728 entries, 0 to 1727
```

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

```
Data columns (total 7 columns):
buying      1728 non-null object
maint       1728 non-null object
doors       1728 non-null object
persons     1728 non-null object
lug_boot    1728 non-null object
safety      1728 non-null object
class       1728 non-null object
dtypes: object(7)
memory usage: 94.6+ KB
```

The train data set has 1728 rows and 7 columns.

There are no missing values in the dataset

Identify the target variable

```
data['class'],class_names = pd.factoriz
```

The target variable is marked as class in the datafr
format. However the algorithm requires the variab
codes. We can convert the string categorical value
method of the pandas library.

Let's check the encoded values now.

```
print(class_names)

print(data['class'].unique())
```

```
Index([u'unacc', u'acc', u'vgood', u'good'], dtype
[0 1 2 3]
```

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

As we can see the values has been encoded into 4 c

Identify the predictor variables and encode integer codes

```
data['buying'],_ = pd.factorize(data['b

data['maint'],_ = pd.factorize(data['ma

data['doors'],_ = pd.factorize(data['do

data['persons'],_ = pd.factorize(data['

data['lug_boot'],_ = pd.factorize(data[

data['safety'],_ = pd.factorize(data['s

data.head()
```

buying	maint	doors	persons
0	0	0	0
0	0		
1	0	0	0
1	0		
2	0	0	0
2	0		
3	0	0	0
0	0		
4	0	0	0
1	0		

Check the data types now :

```
data.info()
```

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1728 entries, 0 to 1727
Data columns (total 7 columns):
buying      1728 non-null int64
maint       1728 non-null int64
doors       1728 non-null int64
persons     1728 non-null int64
lug_boot    1728 non-null int64
safety      1728 non-null int64
class       1728 non-null int64
dtypes: int64(7)
memory usage: 94.6 KB
```

Everything is now converted in integer form.

Select the predictor feature and select the t

```
X = data.iloc[:,:-1]

y = data.iloc[:,-1]
```

Train test split :

```
# split data randomly into 70% training

X_train, X_test, y_train, y_test = mode
y, test_size=0.3, random_state=123)
```

Training / model fitting

```
params = {

'objective': 'binary:logistic',

'max_depth': 2,

'learning_rate': 1.0,
```

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

```
'silent': 1.0,

'n_estimators': 5

}

model = XGBClassifier(**params).fit(X_t
```

Model parameters study :

```
# use the model to make predictions wit

y_pred = model.predict(X_test)

# how did our model perform?

count_misclassified = (y_test != y_prec

print('Misclassified samples: {}'.forma

accuracy = metrics.accuracy_score(y_tes

print('Accuracy: {:.2f}'.format(accurac
```

```
Misclassified samples: 58
Accuracy: 0.89
```

The model actually has a 89% accuracy score,Not l
implement your first xgboost model with scikit-lea
a try!

Algo Advantages :

Parallel Computing: It is enabled with parallel
you run xgboost, by default, it would use all the co

Multivariate MultiLabel Classification with Logistic Regression[Case Study]
Naive Bayes Algorithm [Case Study]
Understanding Principal Component Analysis(PCA)
PCA for Fast ML
Polynomial Logistic Regression[Case Study]
Random Forest for Car Quality[Case Study]
Random Forest for Regression[Case Study]
Simple Linear Regression[Case Study]
Simple Logistic Regression[Case Study]
XGBoost for Classification[Case Study]
XGBoost for Regression[Case Study]

Regularization: I believe this is the biggest adva
for regularization. Regularization is a technique us
based models.

Enabled Cross Validation: In R, we usually use
to obtain CV results. But, xgboost is enabled with i

Missing Values: XGBoost is designed to handle
values are treated in such a manner that if there ex
captured by the model.

Flexibility: In addition to regression, classificatio
user-defined objective functions also. An objective
performance of the model given a certain set of pa
defined evaluation metrics as well.

Availability: Currently, it is available for progran
Julia, and Scala.

Save and Reload: XGBoost gives us a feature to
reload it later. Suppose, we have a large data set, w
future instead of wasting time redoing the comput

Tree Pruning: Unlike GBM, where tree pruning
XGBoost grows the tree upto max_depth and then
in loss function is below a threshold.

case study Machine Learning xgboost

SHARE THIS POST



Sudhanshu Kumar

Multivariate MultiLabel Classification with Logistic Regression[Case Study]

Naive Bayes Algorithm [Case Study]

Understanding Principal Component Analysis(PCA)

PCA for Fast ML

Polynomial Logistic Regression[Case Study]

Random Forest for Car Quality[Case Study]

Random Forest for Regression[Case Study]

Simple Linear Regression[Case Study]

Simple Logistic Regression[Case Study]

XGBoost for Classification[Case Study]

XGBoost for Regression[Case Study]

Data Scientist at Verizon Labs

Related Articles