KDnuggets

Subscribe to KDnuggets News | | Contact

search KDnuggets Search

- SOFTWARE
- NEWS
- Top stories
- Opinions
- Tutorials
- JOBS
- Companies
- Courses
- Datasets
- EDUCATION
- Certificates
- Meetings
- Webinars

KDnuggets Home » News » 2017 » Jun » Tutorials, Overviews » 7 Techniques to Handle Imbalanced Data (17:n22)

7 Techniques to Handle Imbalanced Data

Previous post
Next post

Share 215 G+1 7 Share 187

Tags: Balancing Classes, Data Preparation, Data Science, Unbalanced

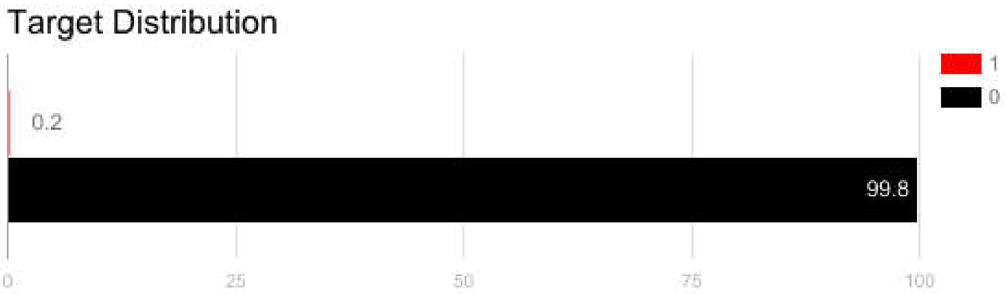
This blog post introduces seven techniques that are commonly applied in domains like intrusion detection or real-time bidding, because the datasets are often extremely imbalanced.

By Ye Wu & Rick Radewagen, IE Business School.

Introduction

What have datasets in domains like, fraud detection in banking, real-time bidding in marketing or intrusion detection in networks, in common?

Data used in these areas often have less than 1% of rare, but "interesting" events (e.g. fraudsters using credit cards, user clicking advertisement or corrupted server scanning its network). However, most machine learning algorithms do not work very well with imbalanced datasets. The following seven techniques can help you, to train a classifier to detect the abnormal class.



1. Use the right evaluation metrics

Applying inappropriate evaluation metrics for model generated using imbalanced data can be dangerous. Imagine our training data is the one illustrated in graph above. If accuracy is used to measure the goodness of a model, a model which classifies all testing samples into "0" will have an excellent accuracy (99.8%), but obviously, this model won' t provide any valuable information for us.

In this case, other alternative evaluation metrics can be applied such as:

- Precision/Specificity: how many selected instances are relevant.
- Recall/Sensitivity: how many relevant instances are selected.
- F1 score: harmonic mean of precision and recall.
- MCC: correlation coefficient between the observed and predicted binary classifications.
- AUC: relation between true-positive rate and false positive rate.

2. Resample the training set

Apart from using different evaluation criteria, one can also work on getting different dataset. Two approaches to make a balanced dataset out of an imbalanced one are under-sampling and over-sampling.

2.1. Under-sampling

Under-sampling balances the dataset by reducing the size of the abundant class. This method is used when quantity of data is sufficient. By keeping all samples in the rare class and randomly selecting an equal number of samples in the abundant class, a balanced new dataset can be retrieved for further modelling.

2.2. Over-sampling

On the contrary, oversampling is used when the quantity of data is insufficient. It tries to balance dataset by increasing the size of rare samples. Rather than getting rid of abundant samples, new rare samples are generated by using e.g. repetition, bootstrapping or SMOTE (Synthetic Minority Over-Sampling Technique) [1].

Note that there is no absolute advantage of one resampling method over another. Application of these two methods depends on the use case it applies to and the dataset itself. A combination of over- and under-sampling is often successful as well.

3. Use K-fold Cross-Validation in the right way

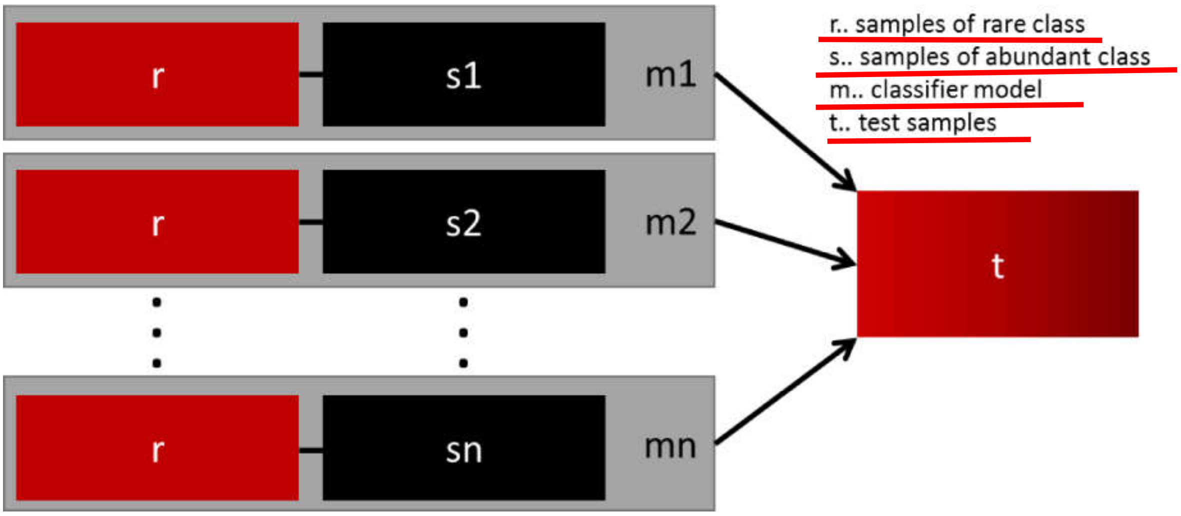
It is noteworthy that cross-validation should be applied properly while using over-sampling method to address imbalance problems.

Keep in mind that over-sampling takes observed rare samples and applies bootstrapping to generate new random data based on a distribution function. If cross-validation is applied after over-sampling, basically what we are doing is overfitting our model to a specific artificial bootstrapping result. That is why cross-validation should always be done before over-sampling the data, just as how feature selection should be implemented. Only by resampling the data repeatedly, randomness can be introduced into the dataset to make sure that there won’ t be an overfitting problem.

4. Ensemble different resampled datasets

The easiest way to successfully generalize a model is by using more data. The problem is that out-of-the-box classifiers like logistic regression or random forest tend to generalize by discarding the rare class. One easy best practice is building n models that use all the samples of the rare class and n-differing samples of the abundant class. Given that you want to ensemble 10 models, you would keep e.g. the 1.000 cases of the rare class and randomly sample 10.000 cases of the abundant class. Then you just split the 10.000 cases in 10 chunks and train 10 different models.

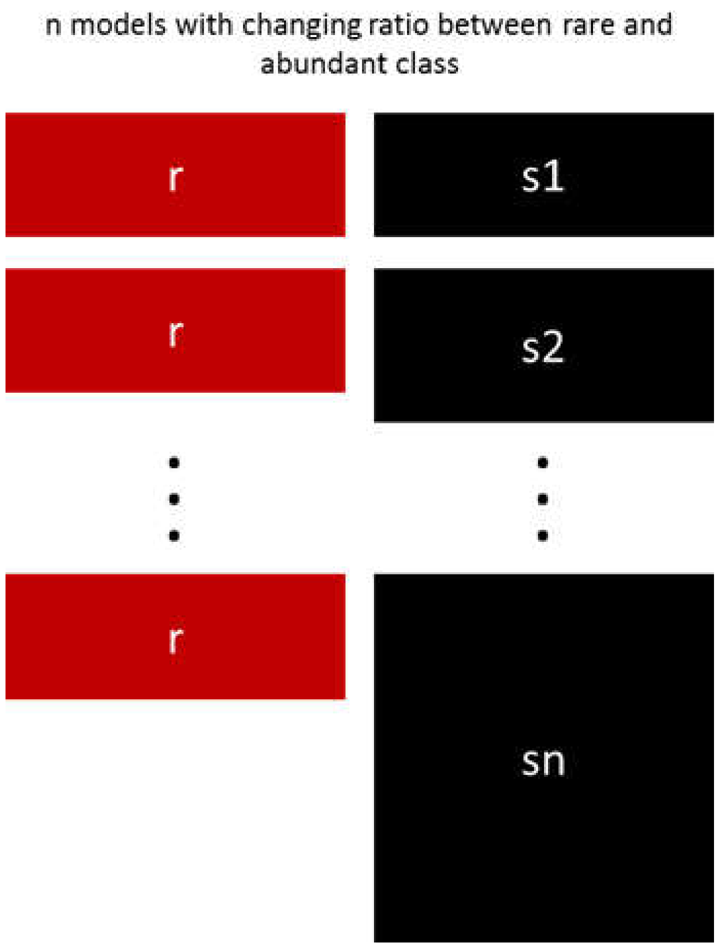
n models with changing data samples for the abundant class



This approach is simple and perfectly horizontally scalable if you have a lot of data, since you can just train and run your models on different cluster nodes. Ensemble models also tend to generalize better, which makes this approach easy to handle.

5. Resample with different ratios

The previous approach can be fine-tuned by playing with the ratio between the rare and the abundant class. The best ratio heavily depends on the data and the models that are used. But instead of training all models with the same ratio in the ensemble, it is worth trying to ensemble different ratios. So if 10 models are trained, it might make sense to have a model that has a ratio of 1:1 (rare:abundant) and another one with 1:3, or even 2:1. Depending on the model used this can influence the weight that one class gets.


n models with changing ratio between rare and abundant class
r
s1
r
s2
r
sn

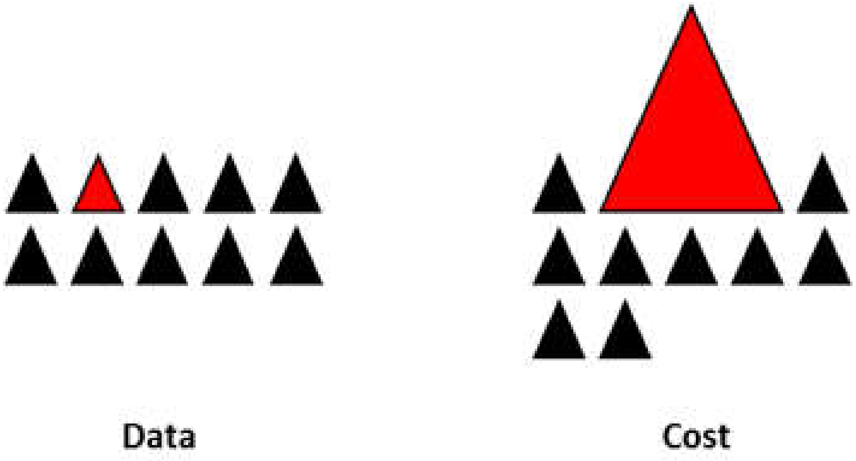
6. Cluster the abundant class

An elegant approach was proposed by Sergey on Quora [2]. Instead of relying on random samples to cover the variety of the training samples, he suggests clustering the abundant class in r groups, with r being the number of cases in r. For each group, only the medoid (centre of cluster) is kept. The model is then trained with the rare class and the medoids only.

7. Design your own models

All the previous methods focus on the data and keep the models as a fixed component. But in fact, there is no need to resample the data if the model is suited for imbalanced data. The famous XGBoost is already a good starting point if the classes are not skewed too much, because it internally takes care that the bags it trains on are not imbalanced. But then again, the data is resampled, it is just happening secretly.

By designing a cost function that is penalizing wrong classification of the rare class more than wrong classifications of the abundant class, it is possible to design many models that naturally generalize in favour of the rare class. For example, tweaking an SVM to penalize wrong classifications of the rare class by the same ratio that this class is underrepresented.



Final Remarks

This is not an exclusive list of techniques, but rather a starting point to handle imbalanced data. There is no best approach or model suited for all problems and it is strongly recommended to try different techniques and models to evaluate what works best. Try to be creative and combine different approaches. It is also important, to be aware that in many domains (e.g. fraud detection, real-time-bidding), where imbalanced classes occur, the "market-rules" are constantly changing. So, check if past data might have become obsolete.

[1] arxiv.org/pdf/1106.1813.pdf

[2] www.quora.com/In-classification-how-do-you-handle-an-unbalanced-training-set/answers/1144228?srid=h3G6o

Ye Wu is pursuing the Master in Business Analytics & Big Data at the IE Business School. She has a background in Accounting and hands-on experience in Marketing and Sales Forecasting.

Rick Radewagen is an aspiring Data Scientist with a background in Computer Science. He is also pursuing the Master in Business Analytics & Big Data at the IE Business School.

Related:

- Learning from Imbalanced Classes
- Dealing with Unbalanced Classes, SVMs, Random Forests, and Decision Trees in Python
- Tidying Data in Python

◀ **Previous post**
Next post ▶

Top Stories Past 30 Days

Most Popular

1. **Top 15 Python Libraries for Data Science in 2017**
2. **The 10 Algorithms Machine Learning Engineers Need to Know**
3. **6 Interesting Things You Can Do with Python on Facebook Data**
4. **10 Free Must-Read Books for Machine Learning and Data Science**
5. **7 Steps to Mastering Data Preparation with Python**
6. **Is Regression Analysis Really Machine Learning?**
7. **Machine Learning Workflows in Python from Scratch Part 1: Data Preparation**

Most Shared

1. **Top 15 Python Libraries for Data Science in 2017**
2. **6 Interesting Things You Can Do with Python on Facebook Data**
3. **Is Regression Analysis Really Machine Learning?**
4. **Deep Learning Papers Reading Roadmap**
5. **A Practical Guide to Machine Learning: Understand, Differentiate, and Apply**
6. **Applying Deep Learning to Real-world Problems**
7. **Emerging Ecosystem: Data Science and Machine Learning Software, Analyzed**

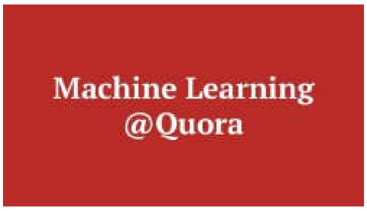
Latest News

- New Poll: Will society become better from inc...
- Upcoming Meetings in Analytics, Big Data, Dat...
- Analytically Speaking Featuring Pedro Saraiva...
- Fidelity Investments: Vice President (AI Lead...
- Improving Zillow Zestimate with 36 Lines of Code
- Exploratory Data Analysis in Python

Top Stories Last Week

Most Popular

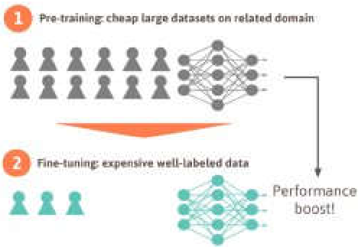
1. NEW **Top 10 Quora Machine Learning Writers and Their Best Advice, Updated**



2. NEW **Emerging Ecosystem: Data Science and Machine Learning Software, Analyzed**
3. NEW **Applying Deep Learning to Real-world Problems**
4. NEW **Top 15 Python Libraries for Data Science in 2017**
5. NEW **Using the TensorFlow API: An Introductory Tutorial Series**
6. NEW **Text Clustering: Get quick insights from Unstructured Data**
7. ▼ **The 10 Algorithms Machine Learning Engineers Need to Know**

Most Shared

1. **Applying Deep Learning to Real-world Problems**


1 Pre-training: cheap large datasets on related domain
2 Fine-tuning: expensive well-labeled data
Performance boost!

2. **Text Clustering: Get quick insights from Unstructured Data**
3. **Using the TensorFlow API: An Introductory Tutorial Series**
4. **Top 10 Quora Machine Learning Writers and Their Best Advice, Updated**
5. **Why Artificial Intelligence and Machine Learning?**
6. **For data scientists, now is the time to act; Forrester has insights to help you get started**
7. **Web Scraping with R: Online Food Blogs Example**

KDnuggets Home » News » 2017 » Jun » Tutorials, Overviews » 7 Techniques to Handle Imbalanced Data (17:n22)

Subscribe to KDnuggets News

X