NICE
ACTIMIZE

Online Machine Learning: An Introduction

Part 1

Insights Article

Danny Butvinik
Chief Data Scientist | NICE Actimize

With billions of financial transactions being processed every minute, financial services organizations (FSOs) face a high risk of fraud. To keep them a step ahead, it requires the high capabilities of machines to learn online effectively and fast.

As a mathematician and data scientist for many years, I've always considered Online Machine Learning as one of the most challenging and intriguing topics. While the theoretical framework of online learning has always been a key part of my diverse research, as a Chief Data Scientist at NICE Actimize, I now realize just how vital it is for theory to become practical.

Abstract

This article is meant to serve as an introduction to online machine learning, one of the most fascinating and challenging sub-domains in Computational Learning Theory. Recently, online learning and incremental learning gained attention, especially in the context of big data and learning from data streams, conflicting with the traditional assumption of complete data availability. Even though a variety of different methods are available, it often remains unclear which are suitable for specific tasks and how they perform in comparison to each other. Following are brief explanations of some of the basic concepts on online machine learning.

What is Online Machine Learning?

Online machine learning, a well-established learning paradigm that has both theoretical and practical applications, has been studied in many research fields, including online anomaly detection, incremental learning, game theory, and information theory. It also became of great interest to data science practitioners due to the recent emergence of large-scale applications such as financial fraud detection, social media, healthcare, and online web ranking.

Online machine learning is about making consequential predictions given a knowledge of the correct answer to previous predictions and potentially additional available information. An online learner needs to make predictions about a sequence of instances, one after the other and receives feedback after each prediction. The performance of the online learner is typically compared to the best predictor from a given class, often in terms of its excess loss (*the regret*) over the best predictor.

Example

For example, a learner might receive a pattern of financial transaction and the question is whether the transaction is fraudulent or not. To answer the question, the learner uses a prediction mechanism, termed a hypothesis, which is a mapping from the set of questions to the set of admissible answers. After predicting an answer, the correct answer is revealed, and the learner suffers a loss if there is a discrepancy between his answer and the correct one. The learner's goal is to minimize the *cumulative loss* suffered along its run. To achieve this goal, the learner may update the hypothesis after each round to be more accurate in later rounds.

What are Regret Bounds?

Regret bounds are the common thread on the analysis of online learning algorithms. A regret bound measures the performance of an online algorithm relative to the performance of a competing prediction mechanism called a competing hypothesis. The competing hypothesis can be chosen in hindsight from a class of hypotheses, after observing the entire sequence of question-answer pairs.

Regret bounds are universal in the sense that they hold for any possible fixed hypothesis in a given hypothesis class. Therefore, casting the universal bound as a lower bound for an optimization problem, in which we search for the optimal competing hypothesis. While the optimal competing hypothesis can only be found in hindsight, after observing the entire sequence of question-answer pairs, this viewpoint relates regret bounds to lower bounds of minimization problems.

Formally, we assess the performance of the learner using the notion of regret. Given any fixed hypothesis h∈ H, we define the regret of an online learning algorithm as the excess loss for not consistently predicting with the hypothesis *h,*

$$R(h,T) = \sum_{t=1}^{T} \ell(h_t,(\mathbf{x}_t,y_t)) - \sum_{t=1}^{T} \ell(h,(\mathbf{x}_t,y_t))$$

Similarly, given any fixed vector u ∈ S, we define the regret of an online convex programming procedure as the excess loss for not consistently choosing the vector u ∈ S,

$$R(\mathbf{u},T) = \sum_{t=1}^{T} g_t(\mathbf{w}_t) - \sum_{t=1}^{T} g_t(\mathbf{u})$$

What is Duality?

The notion of duality, commonly used on convex optimization theory, plays an important role in obtaining lower bounds for the minimal value of minimization problem. By generalizing the notion of *Fenchel duality,* we can derive a dual optimization problem, which can be optimized incrementally, as the online learning progresses. The main idea behind our derivation is the connection between regret bounds and Fenchel duality. This connection leads to a reduction from the process of online learning to the task of incrementally ascending the dual objective function.

To derive explicit quantitative regret bounds, we make immediate use of the *weak duality property*, which tells us that the dual objective lower bounds the primal objective. We, therefore, reduce the process of online learning to the task of incrementally increasing the dual objective function. The amount by which the dual increases serves as a new and natural notion of progress.

By doing so, we can associate the cumulative loss of the competing hypothesis and the cumulative loss of the online algorithm, using the increase in the dual.

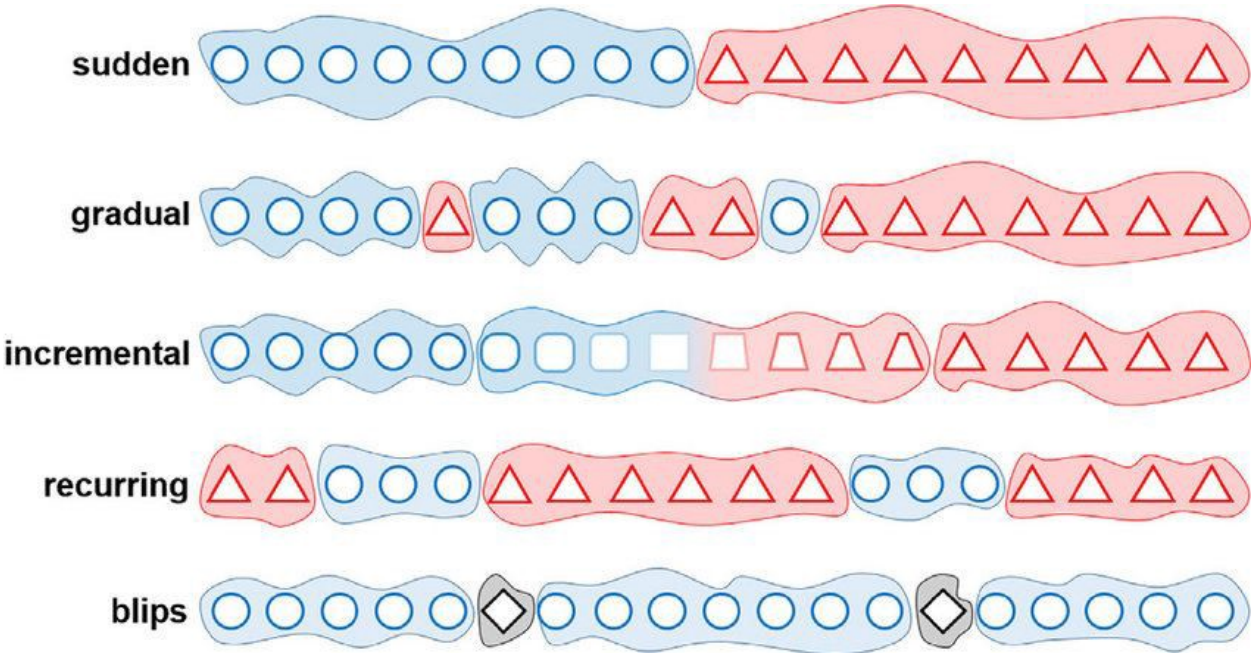
What is Incremental Learning?

Incremental learning refers to the situation of continuous model adaptation based on a constantly arriving data stream. Online learning becomes necessary in interactive scenarios where training examples are provided based on human feedback over time. Incremental learning refers to online learning strategies that work with limited memory resources.

Concept Drift

Incremental learning shares several challenges with online learning, with memory limitations creating even more challenges. One prominent problem is that when the temporal structure of the data samples is considered, one can observe changes in data statistics that occur over time. Changes in the data distribution over time are commonly referred to as *concept drift.* Different types of concept drift can be distinguished: changes in the input distribution only referred to as *virtual concept drift*, also referred to as *covariate shift*; or changes in the underlying functionality itself, referred to as *real concept drift*.

Further, *concept drift* can be *gradual* or *abrupt*. The *term local concept drift* characterizes changes of the data statistics only in a specific region of the data space. *Real concept drift* is problematic since it leads to conflicts in the classification – for example when a new but visually similar class appears in the data. This will, in any event, have an impact on classification performance until the model can be re-adapted accordingly.



Types of concept drifts

Stability-Plasticity Dilemma

For noisy environments or *concept drift*, a second challenge consists of the question of when and how to adapt the current model. A quick update enables a rapid adaptation according to new information, but old information is forgotten just as quickly. On the other hand, adaption can be performed slowly, in which old case information is retained longer but the reactivity of the system is decreased. The dilemma behind this trade-off is usually denoted as the *stability-plasticity dilemma*, which is a well-known constraint for artificial, as well as biological learning systems.

Incremental learning techniques, which adapt learned models to *concept drift* only in those regions of the data space where *concept drift* occurs, offer a partial remedy to this problem. When dealing with limited resources, many online learning methods are not able to solve this dilemma on their own because they exhibit a so-called *catastrophic forgetting* behavior even when the new data statistics do not invalidate the old ones.

One approach to deal with the *stability-plasticity dilemma* is to enhance the learning rules by explicit meta-strategies – that is, when and how to learn.

Model Benchmarking

There exist two fundamentally different possibilities to assess the performance of incremental learning algorithms:

Incremental vs non-incremental: In the absence of *concept drift*, the aim of learning consists of the inference of the stationary distribution *p(y|x)* for typical data characterized by *p(x).* For example, this setting occurs whenever incremental algorithms are used for big data sets, and they compete with parallelized batch algorithms. In such settings, the method of choice evaluates the classification accuracy of the final model Mt on a test set, or within cross-validation. While incremental learning should attain results in the same range as batch variants, one must consider that they deal with restricted knowledge due to their streaming data access. It has been shown, as an example, that incremental clustering algorithms cannot reach the same accuracy as batch versions if restricted in terms of their resources.

Incremental vs incremental: When facing concept drift, different cost functions can be of interest. Virtual concept drift aims for the inference of a stationary model *p(y|x)* with a drifting probability of *p(x)* the inputs. In such settings, the robustness of the model when evaluated on test data, following a possibly skewed distribution, is of interest. Such settings can easily be generated e.g. by enforcing imbalanced label distributions for test and training data. Whenever *real confidence drift* is present, the online behavior of the classification error, the next data point is usually the method of choice. Therefore, a simple average of these errors can be accompanied by a detailed inspection of the overall shape of the online error, since it provides insight into the rates of convergence, such as for *abrupt concept drift*.

$$\|M_t(\vec{x}_{t+1}) - y_{t+1}\|$$

Formal guarantees on the generalization behavior: Since many classical algorithms such as the simple perceptron or large margin methods have been proposed as online algorithms, there exists an extensive body of work investigating their learning behavior, convergence speed, and generalization ability, classically relying on the assumption of data being i.i.d. Some results weaken the i.i.d. assumption and require only interchangeability. Recently, popular settings, such as learning a generalized linear regression, could be accompanied by convergence guarantees for arbitrary distributions *p(x)* by taking a game-theoretic point of view. In such settings, classifier Mt and training example can be taken in an adversarial manner, still allowing for fast convergence rates in relevant situations.

x_{t+1}

x_{t+1}

While this article highlighted the basic concepts and several mathematical notions of Online Machine Learning, parts two and three elaborate more on each concept and also focus on Online Machine Learning Algorithms.

A note on undersampling: for unbalanced classification tasks, it is not clear how these two effects interact and when undersampling leads to better accuracy in the classification task.

ABOUT NICE ACTIMIZE

NICE Actimize is the largest and broadest provider of financial crime, risk and compliance solutions for regional and global financial institutions, as well as government regulators. Consistently ranked as number one in the space, NICE Actimize experts apply innovative technology to protect institutions and safeguard consumers and investors assets by identifying financial crime, preventing fraud and providing regulatory compliance. The company provides real-time, cross-channel fraud prevention, anti-money laundering detection, and trading surveillance solutions that address such concerns as payment fraud, cybercrime, sanctions monitoring, market abuse, customer due diligence and insider trading.

NICE · ACTIMIZE