16/01/2021

Monitorer les performances d'un modèle de machine learning en production

Etat de l'art

Introduction

Il n'existe rien de constant si ce n'est le changement. Le monde du machine learning n'échappe pas à cette règle. Le processus de création d'un modèle de Machine learning suit les étapes classiques d'apprentissage avec un jeu de données viable à l'instant t, d'évaluation du modèle avec un échantillon de test, puis une étape de mise en production lorsque les métriques d'évaluation du modèle sont satisfaisantes. Toutefois durant la phase de vie du modèle (post mise en production), les données entrantes peuvent être amenées à changer. Ces changements peuvent impacter la performance du modèle. Il est alors important de trouver un moyen de monitorer cette performance à travers le temps et de se créer des alertes en cas de dégradation (drift) de sa performance.

Nous allons donc chercher à comprendre quelles sont les meilleures pratiques sur ce sujet : comment évaluer le modèle de machine learning à travers le temps et s'assurer que celui-ci ne se dégrade pas ?

À l'heure actuelle les recherches sur les drifts se sont accentuées, celle-ci regroupent plusieurs champs de recherches et les terminologies ne sont pas encore unifiées au sein de la communauté scientifique.

Compte tenu de ces éléments et des contraintes du projet, nous concentrerons notre travail sur la présentation des différentes typologies de drifts, des différents outils pour le mesurer et des méthodes pour rendre les algorithmes adaptatifs. Nous ne rentrerons pas dans le détail statistique des différentes méthodes de calculs. De même, une fois la dégradation d'un modèle avérée, nous n'aborderons pas les options s'offrant au data scientist pour la mise à jour du modèle tant ceci peut être un sujet à part entière (ré apprentissage, ensemble learning, online learning...).

Présentation des sources :

Notre étude se fera en deux parties : nous commencerons par étudier le concept de drift en machine learning puis nous regarderons quelles sont à l'heure actuelle les meilleures pratiques pour s'adapter et palier à ces problématiques de drift. Durant notre recherche documentaire, nous avons souhaité mélanger deux types de sources :

1. Des sources de recherches académiques avec pour objectif de cadrer le problème et de poser des définitions claires sur les termes utilisés ;
2. Des sources provenant du "terrain" et de data scientist qui ont rencontré ces problématiques durant leur vie professionnelle avec pour objectif de voir les meilleures pratiques terrains mise en place ;

Les sources liées aux recherches académiques proviennent de plusieurs canaux :

L'association professionnelle nommée : Institute of Electrical and Electronics Engineers (IEEE). IEEE est la plus grande organisation professionnelle technique au monde dédié à l'avancement de la technologie au profit de l'humanité. Reconnue par la communauté scientifique, les articles publiés par cette organisation sont viables et suivent un processus de publication stricte : relecture par des confrères, vérification du plagiat etc...

Deux articles sélectionnés sont issus de ce canal :

- L'article "Multiscale Drift Detection Test to Enable Fast Learning in Nonstationary Environments" a été publié en 2021 par une équipe de l'université de Shangaï spécialiste dans le domaine des " computers sciences".
- L'article "*Learning under concept drift : a review"* a été publié en 2019 par plusieurs auteurs de la faculté of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia dont la directrice du pôle d'intelligence artificielle de la faculté pré cité.

Les autres canaux de recherches nous ont permis de sourcer plusieurs articles complémentaires :

- L'association for Computing Machinery (ACM) dont les articles publiés suivent un processus strict incluant une relecture par les pairs. Existant depuis plus de soixante ans, cette association est reconnue dans le monde de la recherche au même titre que IEEE. L'article sélectionné : " A survey on concept Drift adaption" date de 2013, cela rend son impact un peu moindre bien que les théories statistiques utilisées soient toujours d'actualité.
- Un article "A unifying view on dataset shift in classification" provient de la revue scientifique Elsevier, spécialiste dans la recherche et notamment la recherche dans le domaine de la santé. Ses auteurs sont là aussi tous spécialistes de sciences de l'informatique et de l'intelligence artificielle.

Pour finir, nous avons sélectionné des articles provenant d'un blog reconnu dans la data science : Towards Data Science Inc. est une société enregistrée au Canada qui se base sur la technologie Medium. Ils fournissent une plate-forme à des milliers de personnes pour échanger des idées et élargir la compréhension de la datascience. Les articles publiés sur cette plateforme doivent répondre à certaines règles et sont relus par une équipe spécialiste avant publication. Toutefois le contenu n'est pas de la responsabilité de l'entreprise mais des auteurs eux-mêmes. Cette source sera utilisée principalement pour sonder les pratiques de la communauté de développeurs IA/data scientist vis-à-vis des problématiques de drift des modèles.

Les tendances du domaine et l'impact sur la pratique professionnelle

Comme vu lors de l'introduction les données envoyées au cours du temps à un modèle en production peuvent évoluer. Lorsqu'un modèle fait référence à des informations obsolètes, la performance de celui-ci se dégrade.

Il existe plusieurs types de dégradation d'un modèle de Machine Learning et plusieurs terminologies pour les adresser :

Globalement on parle de dataset shift pour décrire l'évolution du jeu de données entre la phase d'apprentissage et la phase de test. Certains auteurs assimilent cette notion à celle de concept drift ou concept shift. Toutefois plus récemment, des auteurs [1] ont avancés que le dataset shift est la catégorie parents de plusieurs sous-ensembles :

- Covariate drift/shift ou data drift : modifications dans la distribution des variables explicatives du modèle. Exemple : un système de recommandation de film qui a appris sur les films vus par une population retraitée performera moins bien sur une population d'enfants.
- Prior probability drift/shift : l'inverse du covariate shift : modification dans la distribution de la variable cible. Exemple : jeu d'apprentissage 50% de variable cible spam et 50% de courriers sains alors que les données en réalité sont réparties : 10% de spam et 90% de sains
- Concept drift /shift: Les propriétés statistiques de la variable cible évoluent, l'algorithme mappant les variables explicatives à la cible n'est alors plus correct. On parle des relations entre variables explicatives et variable cible [2]. Exemple : la définition d'un spam change (la structure des spams est différente en 2021 vs. 2000).

Il existe plusieurs types de drift qui sont représentés dans l'annexe 1 : Un drift abrupt ne sera pas monitoré de la même manière qu'un drift progressif. Il est à noter que la terminologie de drift peut être remplacée par shift dans certains écris lorsque les évolutions ne sont pas abruptes mais progressives.

Pour finir, il faut distinguer deux classes de concept drift :

- Le virtual drift : changement dans la distribution des variables X et des frontières de décision sans pour autant affecter $p(y|X)$.
- Real drift : correspond aux changements de $p(y|X)$, ces changements peuvent être ou ne pas être dus à des modifications de $p(X)$. Cf. annexe 2

Il est souvent long voire impossible d'inspecter manuellement des modifications dans un environnement qui mèneraient à une modification des propriétés statistiques de la variable cible. Il est donc important d'automatiser leur détection.

Pour répondre à ces problématiques, les algorithmes doivent être capables de s'adapter à ces changements durant la phase de production (adaptive learning[4]). Les modèles de machine learning peuvent être soit "offline" soit "online". Les modèles "offline", une fois l'entrainement effectué, le modèle est mis en production. Pour ces modèles une fois l'alerte de drift déclenchée, un ré apprentissage global est appliqué.

Les modèles "online" vont traiter la donnée de manière séquentielle. Le modèle est mis en production sans voir toute la donnée d'apprentissage. Il apprend alors progressivement au fur et à mesure que la donnée arrive. Pour être moins restrictifs certains modèles apprennent de manière incrémentale par batch de données et non continuellement. Pour finir, les "streaming algorithms" sont de la catégorie des algorithmes "online" mais traitent des flux rapides en importants de données. Pour ces modèles, la détection d'un drift enclenche un réapprentissage partiel. Cf annexe 3.

L'annexe 4 montre que pour créer des alertes de drift, il y a plusieurs étapes : la récupération de données (historiques vs. Nouvelles données), la mise en forme des données, les tests de dissimilarité, les tests de significativités de cette dissimilarité. La problématique principale se situe dans les tests de dissimilarité. La sélection du test le plus pertinent reste une question ouverte dans la communauté scientifique.

Il y a trois catégories de tests qui peuvent être conduits :

1. Les tests basés sur les erreurs produites par le modèle : ces tests tracent l'évolution du taux d'erreur de l'algorithme à travers le temps ;
2. Les tests basés sur la distribution des données : Les algorithmes de cette catégorie utilisent une fonction/métrique de distance pour quantifier la dissemblance entre la distribution des données historiques et les nouvelles données ;
3. Les tests multiples : on applique une série de tests avant de déclencher une alerte. Ces tests peuvent être conduits en parallèle ou en série. Généralement sur la base de l'analyse en composantes principales pour minimiser les ressources mobilisées ;

La difficulté des modèles de la première catégorie est qu'il est nécessaire d'avoir des données labelisées pour évaluer le drift [3]. Les métriques utilisées dépendent du modèle, pour la classification principalement, les mesures d'accuracy, precision et recall peuvent être utilisées.

La deuxième méthode est plus consommatrice de ressources pour le calcul des distances.

Les deux premières méthodes vont comparer les indicateurs sur le concept de fenêtres (données historiques/ nouvelles données). Si la différence est significative alors drift il y a sinon les "nouveaux"

indicateurs sont basculés dans le dataset des indicateurs historiques qui serviront de base pour la prochaine évaluation. [3]

Il existe de nombreux algorithmes différents d'analyse du drift, ceux-ci apportent plus ou moins d'information sur le drift : quand (moment à partir duquel le drift devient significatif), où (quelle région des variables est impactée), comment (importance du drift). L'annexe 5 résume la pertinence de ces algorithmes selon les 3 axes précités.

Pour finir, nous avons observé ce qui se fait dans la communauté des data scientist à travers les blogs reconnus dans le domaine. Peu d'article sont parus visant à monitorer les modèles à travers le temps. Ce qui montre que l'approche de ces problématiques est encore au niveau de la recherche et ne sont pas encore stabilisées/ optimisées.

Toutefois certains articles publiés [5] montrent des approches à travers des outils "low code" tels que Microsoft Azure. Ces méthodologies sont actuellement en phase de test par la communauté d'utilisateurs. Elles sont basées sur les tests de distance et plus spécifiquement : "wassertein distance" et l'"euclidian distance".

L'article introduit aussi des bibliothèques telles que :

- "Evidently" qui permet d'évaluer et de monitorer de manière opensource les modèles en production
- FiddlerAI Monitoring qui aide à rendre les modèles d'intelligence artificiels explicable et monitorables
- Pour créer des méthodes de monitoring sur mesure : PageHinkley ou skmultiflow

Deux autres articles [6] et [7] parlent aussi de la bibliothèque Evidently. Les articles expliquent les tests utilisés par evidently : "Z Test", "chi-squared" ou "two-sample Kolmogorov-Smirnov" en fonction des variables à analyser. L'outil se base donc sur le monitoring de l'évolution de la distribution des données et permet de mettre en avant les évolutions significatives. L'article 6 met en avant que le monitoring des modèles ne doit pas se contenter d'évaluer si un drift a lieu mais doit aussi monitorer la qualité des données qui lui sont transmis pour évider de nombreux problèmes (perte d'accès aux données sources, requête SQL inopérante, changement dans l'infrastructure (changement des noms de colonnes...) etc... ces modifications peuvent alors dégrader la performance du modèle.

L'article [8] parle des méthodologies utilisées par FiddlerAI pour calculer le drift, notamment en précisant qu'ils utilisent : " La divergence de Jensen-Shannon (ou JS) est une méthode de mesure de la similarité entre deux distributions de probabilité. Il est basé sur la divergence KL, avec quelques

différences notables, notamment qu'il est symétrique et qu'il a toujours une valeur finie". Toutefois la solution FiddlerAI est une solution commerciale.

Un point à noter est qu'à l'heure actuelle, des solutions commerciales sont mises en place pour accompagner les entreprises à monitorer leurs modèles et peu de solution "opensource" sont à disposition.

Conclusion :

Pour conclure notre état de l'art, nous avons vu quels sont les différents types de drift. En se focalisant sur le concept drift nous avons vu quels sont les méthodes pour déterminer l'apparition d'un drift. Ces méthodes sont alors à coupler avec l'architecture de notre modèle, soit celui-ci est offline soit celui-ci est online. Pour finir, le champ de recherche sur le drift est un champ vaste qui recoupe plusieurs domaines de recherche. Nous pouvons notamment citer la problématique des biais. La grande majorité des travaux sur la détection de concept drift résumés dans cet état de l'art n'aborde pas le problème du biais de représentation qui est commun à la plupart des systèmes adaptatifs qui imposent ou suggèrent un type particulier de comportement.

De plus, nous ne sommes pas rentrés dans la problématique du choix d'architecture de son projet de ML : offline, online ou autre. Ni des méthodologies pratiques pour implémenter la détection du drift.

Il reste aussi à aborder quelle méthode utiliser en fonction de l'algorithme en production : classification, regression, clusterisation....

Pour finir, nous pouvons citer les bibliothèque Evidently IA et scikit-multiflow. Leurs modules de drift détection qui reprend les principaux algorithmes à disposition sur les méthodes basées sur les erreurs et les calculs de distance.

Sujet passionnant et complexe, le monitoring des projets de machine learning semble encore souvent abordé par des ré apprentissage réguliers sans mise en place de monitoring spécifiques au sein des entreprises. La mise en place de monitoring et d'architecture de modèle dédiée avec des alertes lorsque les modèles se dégradent sera un des prochains challenges à adresser par les data scientist. Pour accompagner cela, des entreprises commerciales commencent à proposer des solutions. Toutefois en vue des nombreuses méthodes abordées par la communauté des chercheurs et les solutions proposées par ces outils, il semblerait que le domaine devra encore maturer pour devenir performant.

Sources :

[1] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodŕıguez, N. V. Chawla, and F. Herrera, “A unifying view on dataset shift in classification,” Pattern Recognit., vol. 45, no. 1, pp. 521–530, 2013

[2] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama and G. Zhang, "Learning under Concept Drift: A Review," in IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 12, pp. 2346-2363, 2019

[3] X. Wang, Q. Kang, M. Zhou, L. Pan and A. Abusorrah, "Multiscale Drift Detection Test to Enable Fast Learning in Nonstationary Environments," in IEEE Transactions on Cybernetics, vol. 51, no. 7, pp. 3483-3495, July 2021

[4] J. Gama, I. ˇZliobaitˇe, A. Bifet, M. Pechenizkiy, and A. Bouchachia, “A Survey on Concept Drift Adaptation.”, ACM Comput. Surv. 1, 1, Article 1, 35 p, January 2013

[5] S. Machiraju, “why-data-drift-detection-is-important-and-how-do-you-automate-it-in-5-simple-steps" ,towardsdatascience, Nov 1, 2021

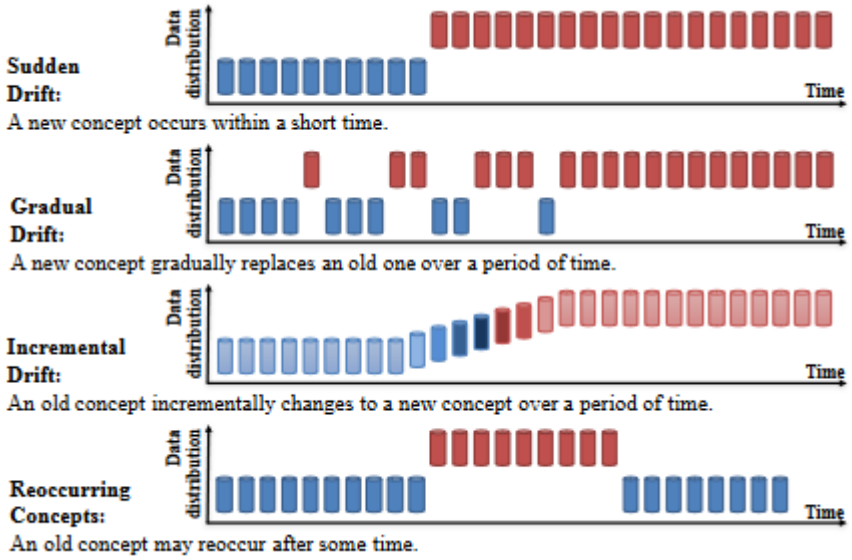
[6] E. Dral, “Monitoring Machine Learning Models in Production, towardsdatascience, Jan 2021

[7] D. Verma, “To Monitor or Not to Monitor a Model — Is there a question ? ”, towardsdatascience, Nov 1, 2021

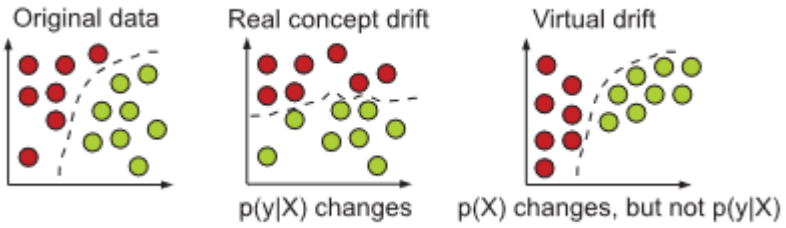
[8] Fiddler Labs Blog, “How-to-detect-model-drift-in-ml-monitoring", medium, sept 2020

Annexes :

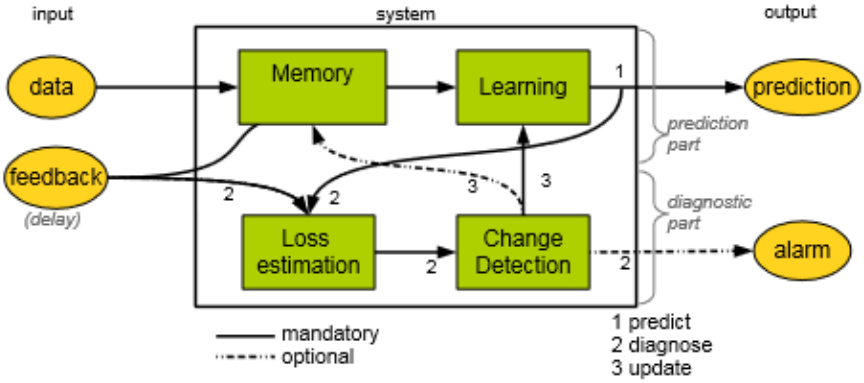
Annexe 1 : les différentes typologies de drift



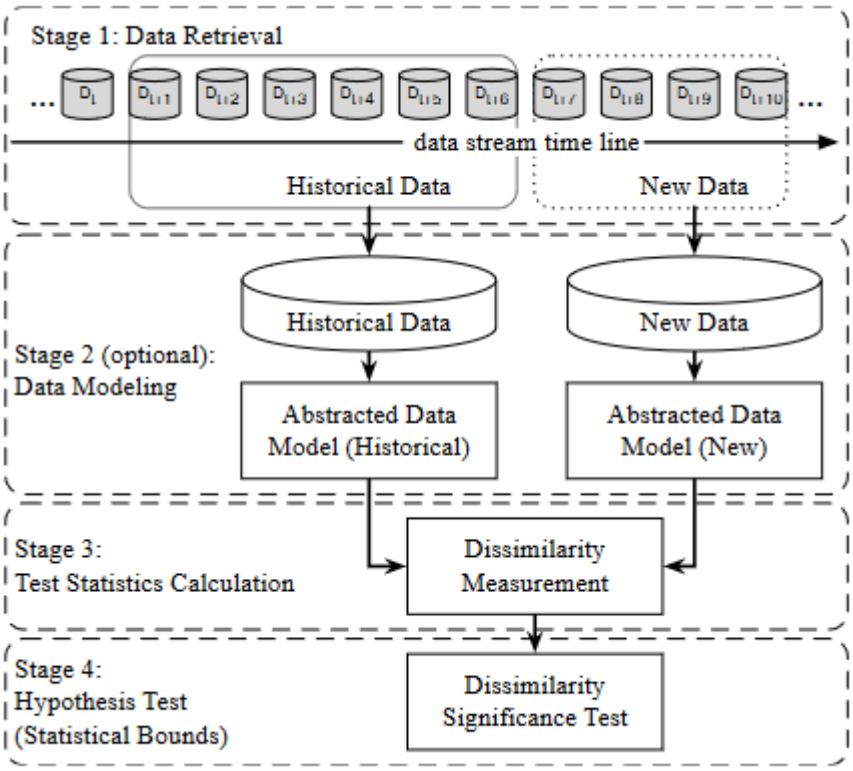
Annexe 2 : Les différents concept drift : virtual et real concept drift



Annexe 3: Schéma générique d'un algorithme online adaptatif [4]



Annexe 4 : Structure d'un projet de détection de drift [2]



Annexe 5 : Synthèse des algorithmes de drift analysis suivant les 3 axes d'analyse [2]:

Summary of drift understanding for drift detection algorithms

Category	Algorithms	When	How	Where
Error rate-based	DDM [20]	√		
	EDDM [26]	√		
	FW-DDM [5]	√		
	DEML [27]	√		
	STEPD [30]	√		
	ADWIN [31]	√		
	ECDD [29]	√		
	HDDM [23]	√		
	LLDD [25]	√		√
Data distribution-based	kdqTree [22]	√	√	√
	CM [2], [3]	√	√	√
	RD [37]	√	√	
	SCD [38]	√	√	
	EDE [40]	√		
	SyncStream [36]	√	√	
	PCA-CD [39]	√	√	
	LSDD-CDT [21]	√		
	LSDD-INC [41]	√		
	LDD-DSDA [4]	√	√	√
Multiple hypothesis tests	JIT [19]	√		
	LFR [46]	√		
	Three-layer drift detection [47]	√		
	e-Detector [48]	√		
	DDE [49]	√		
	EWMA [52]	√		
	HCDTs [50]	√		
	HLFR [51]	√		
	HHT-CU [53]	√		
	HHT-AG [53]	√		