



Save the date !

Mardi 20 Septembre 2022

Je suis conférencier au Salon de la Data à Nantes

MLOoops...

Ou comment faire échouer la mise en production du ML

Rendez-vous à 10 h à la Cité des Congrès
Salle Nantes Métropole et sur la plateforme Imagina


Événement gratuit pour les visiteurs, réservez
votre billet !



Paul PETON

Microsoft MVP





MLOops : comment faire échouer la mise en production du ML



factory going to crash



Speaker : qui suis-je ?

- Paul PETON
- Tech Lead Intelligence Artificielle
- Data Platform & AI MVP since 2018

Présentation illustrée par DALL-E mini

<https://huggingface.co/spaces/dalle-mini/dalle-mini>

MLOps : qu'est-ce que c'est ?

- Pourquoi industrialiser ?
 - Pour **diminuer les coûts** (d'opérations manuelles, d'erreurs...)
 - Pour **augmenter la qualité** (par des opérations testées et automatisées)
- MLOps = ML + DevOps, mais pas que
 - **Framework complet** : design + model development + operations
 - Des niveaux de **maturité**
 - Une **stack** construite à partir d'un **tooling**
 - Sites de référence :
 - <https://ml-ops.org/>
 - <https://mlops-guide.github.io/>
- Une **complexité forte** lié à la nature de l'apprentissage automatique
 - Notion de **reproductibilité**
 - Notions de **dérive** (*drift*)





Sans création de valeur, inutile d'aller plus loin

- Réussir un **PoV** plutôt qu'un PoC
 - Prouver la **valeur** plutôt que le concept
 - = prouver qu'il existe des **facteurs explicatifs de la cible**
- Respecter **l'équation coût – valeur** :
coût de run < création de valeur
 - Difficile sans **mutualisation** des projets
 - ... qui ne seront peut-être pas tous **similaires**

Agenda

- Ne pas **réusiner le code** des notebooks
- Ne pas bien séparer les **environnements**
- Ne pas assurer la **reproductibilité** des modèles
- Ne pas s'adapter aux scénarios de ***serving***
- Ne pas contrôler la **dérive** du modèle ou des données

BONUS :

- Faire confiance (aveuglement) à l'Automated ML
 - Mais qui sont donc les Citizen Data Scientists ?
- Se rassurer (illusoirement) avec le Continuous Training

Ne pas réusiner le code des notebooks

- Chacun développe ses propres **fonctions**
- Aucune stratégie de **tests** (unitaires, fonctionnels, etc.) n'est mise en place
- Les résultats de sortie des cellules sont *pushés* dans **Git**




Ne pas bien séparer les environnements

- Le modèle a été entraîné sur la dev **sans la dernière version des données de production.**
- Des **utilisateurs (DS)** accèdent à la **production** pour entraîner manuellement les modèles.
 - Et faire n'importe quoi.
- Le modèle de production a été entraîné sur la dev.
 - Comment le **livrer en production** ?
- Les modèles sont enregistrés dans des registres propres à chaque environnement.
 - Comment **déployer en production** ?



separate development from production



Ne pas assurer la reproductibilité

- Les **données qui ont servi à l'entraînement** ne sont plus disponibles
 - *elles ont évolué depuis !*
- Le **code qui a servi à l'entraînement** n'a pas été versionné
 - Et les librairies n'ont pas été fixées==1.2.3
- L'**artefact de modèle** est venu écraser la version préalablement entraînée.
 - *Mince, c'est celle qui est déployée en prod...*

Ne pas s'adapter aux scénarios de *serving*

Serving : mise à disposition (« au service ») à l'utilisateur final sous forme de rapports, inférence, etc.

- Ne pas **s'adapter aux scénarios de batch**
 - n'offrir qu'un *endpoint* https, peu adapté à de nombreuses interrogations
 - stocker de nombreuses images Docker
- Déplacer les données plutôt que venir **faire la prévision au plus proche des données**



choosing between batch or real time

Ne pas surveiller la dérive du modèle

- La **distribution des données** (*data*) en entrée a changé
- Le lien (*concept*) entre la cible et les données a changé
- La **préparation des données** a changé

Et rien de cela n'a été détecté...



detect drift on data or on concept

Faire (aveuglément) confiance à l'automated ML

Automated ML : « force brute » de l'apprentissage réalisant un grand nombre d'entraînements à partir de différents algorithmes pour ne conserver que le meilleur (sur la base d'une métrique d'évaluation)

- Ne pas **préparer** les données en amont
- Ne pas (savoir) **interpréter** les métriques d'évaluation
- Abuser des modèles **Ensemble**

blindly trusting automation

Qui sont les Citizen Data Scientists ?

Des personnes qui

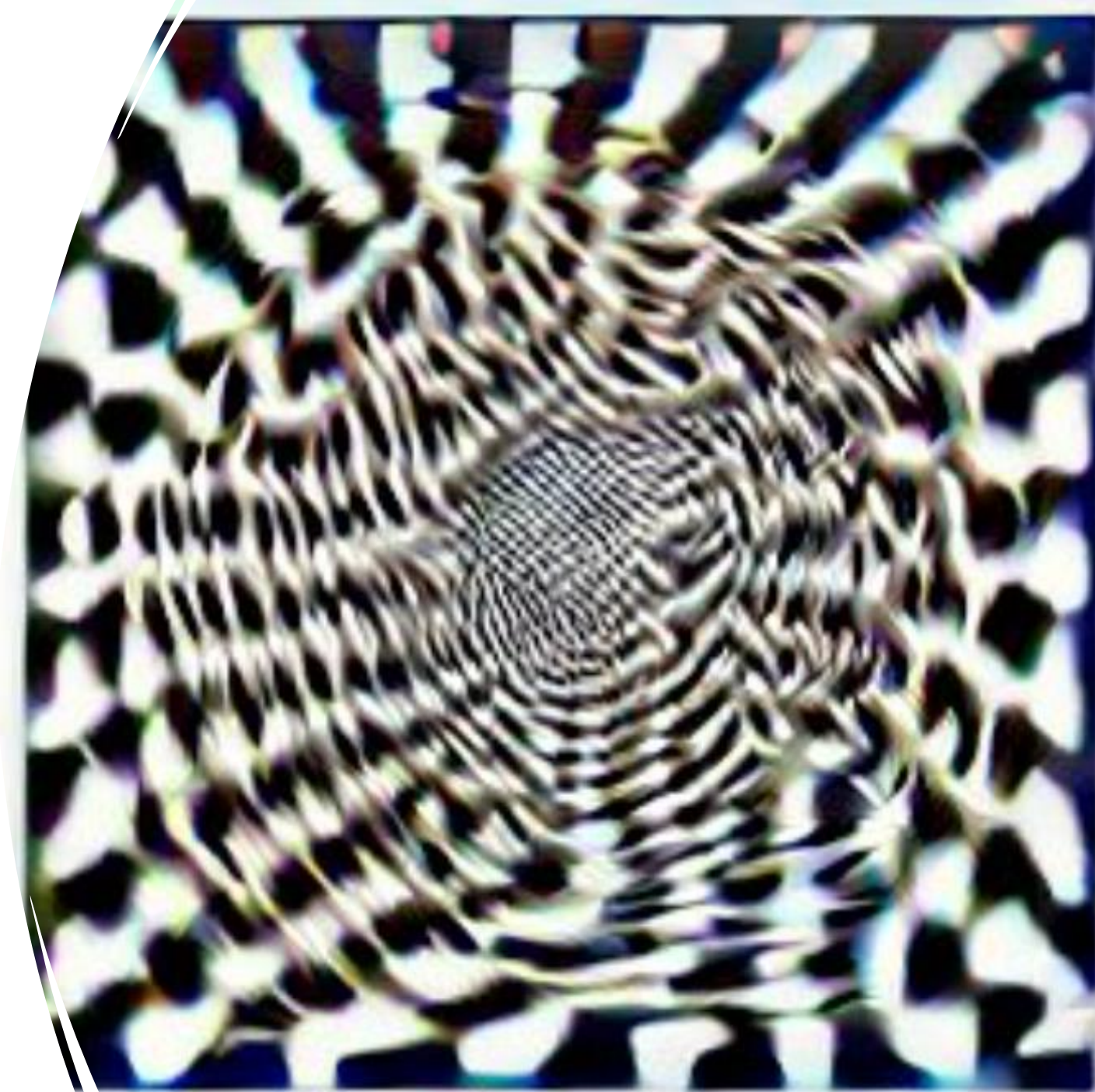
- **ne font pas de code**
 - utilisent des plateformes de ML
- ont des **bases théoriques de Data Science**
 - Spécificités des algorithmes
 - Stratégies d'imputation
 - Métriques d'évaluation
 - Etc.
- explorent des sujets **avant industrialisation**



who are the citizen data scientists

Se rassurer (illusoirement) avec le Continuous Training

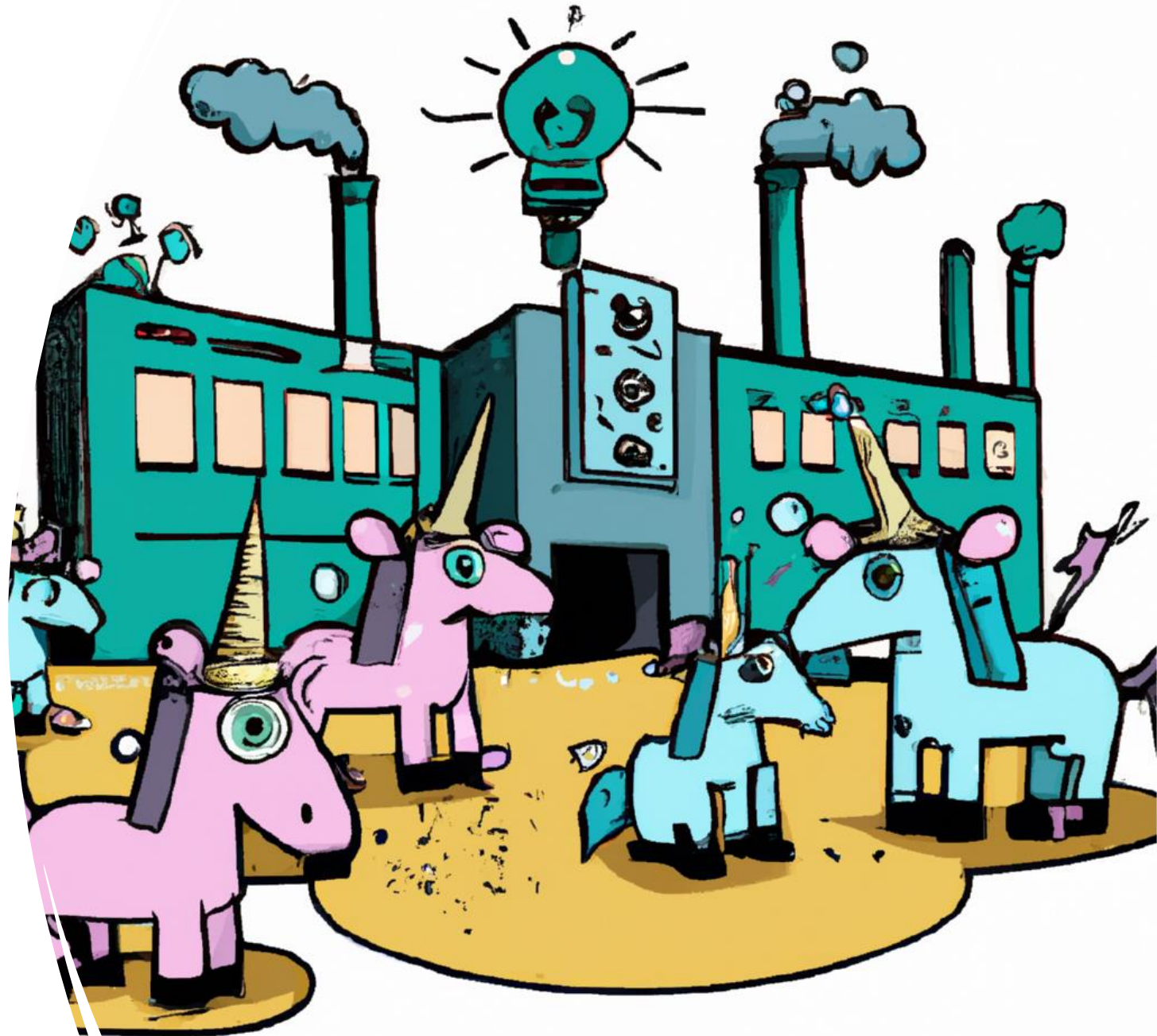
- « on ne peut que progresser »
 - *Sur une métrique, oui*
 - *Mais sur le temps de traitement ?*
 - *Sur la complexité ?*
- « l'interprétabilité est secondaire »
 - *Posez donc la question au métier !*
- « cela dispense de monitorer »
 - *Bien au contraire !*
 - *La drift detection est indispensable*
 - *... et peut éviter le CT*



reassuring illusion of continuity

Résumé des bonnes pratiques

- Seul un **code** robuste, testé et versionné peut aller en production.
- La gestion (environnement, accès, versioning) des **données d'entraînement** est fondamentale.
- Le triptyque **{code, data, model}** permet d'assurer la reproduction d'une prévision.
- Le *serving* se fait de plusieurs manières : **batch** (fichiers ou tables) ou **real time**.
- La **détection des dérives** est un élément incontournable d'une stack MLOps.



a data factory surrounded by five-legged unicorns, pixar style

Meetup: Dernières tendances sur Azure

Le 12 Octobre en présentiel à 18h45

Amis Data engineers et MLOps, rejoignez-nous :

- Migration et insertion massives avec Cassandra sur Azure
- Déployer une stack MLOps sous Azure : gagnez en maturité
- Les dernières innovations Data sur Microsoft Azure



Vincent Devilliers
Head of Delivery



Thomas Dunglas
Data Engineer



Paul Peton
AI Manager - Azure MVP



Eric Charbonnier
Azure CTO Data & AI



cassandra