

WEBINAIRE

# REPRODUCTIBILITÉ EN APPRENTISSAGE AUTOMATIQUE

30 OCTOBRE 2020

## OBJECTIFS DE LA PRÉSENTATION

- Inciter l'intégration des solutions permettant une meilleure reproductibilité dans vos solutions d'affaires et académiques.
- Améliorer la reproductibilité de vos projets.
- Améliorer votre productivité.

## VOTRE CONFÉRENCIER



- Introduit à la recherche reproductible en 2016 (R Markdown et **git**)
- Participation à REPROLANG de la conférence LREC [Garneau et al., 2020]
- Membre actif dans le développement d'une librairie facilitant la reproductibilité (**Poutyne**)

**DAVID BEAUCHEMIN**

Candidat au doctorat

Département d'informatic  
tique et de génie logiciel

## AU MENU



Gestion version



Productivité



Présenter



Réutiliser

# **Introduction**

## C'EST QUOI LA REPRODUCTIBILITÉ?

La reproductibilité est le principe qui dit qu'on ne peut tirer de conclusions que d'un événement bien décrit, qui est apparu plusieurs fois, provoqué par des **personnes différentes**.

Par contre, en apprentissage automatique, la reproductibilité correspond (surtout) soit à être capable de reproduire des résultats, soit d'obtenir des résultats similaires en réexécutant un code source[Pineau et al., 2020].

## POURQUOI S'Y INTÉRESSER ?

70 %<sup>1</sup>

---

1. [Baker, 2016]

## POURQUOI S'Y INTÉRESSER ?

50 %<sup>1</sup>

---

1. [Baker, 2016]

## POURQUOI S'Y INTÉRESSER ?

40 %<sup>2</sup>

---

2. [Raff, 2019]

# MOTIVATION



Réutilisation

## MOTIVATION



Réutilisation



Productivité

# MOTIVATION



Réutilisation



Productivité



Transfert

# MOTIVATION



Réutilisation



Productivité

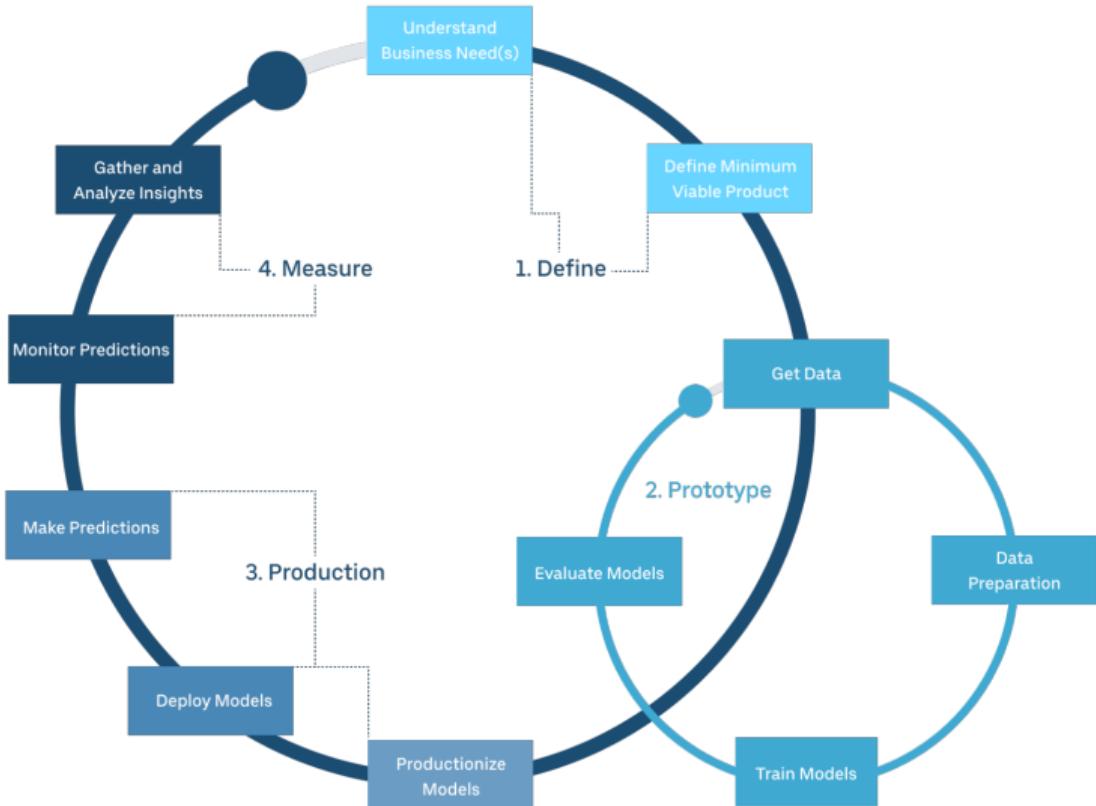


Transfert

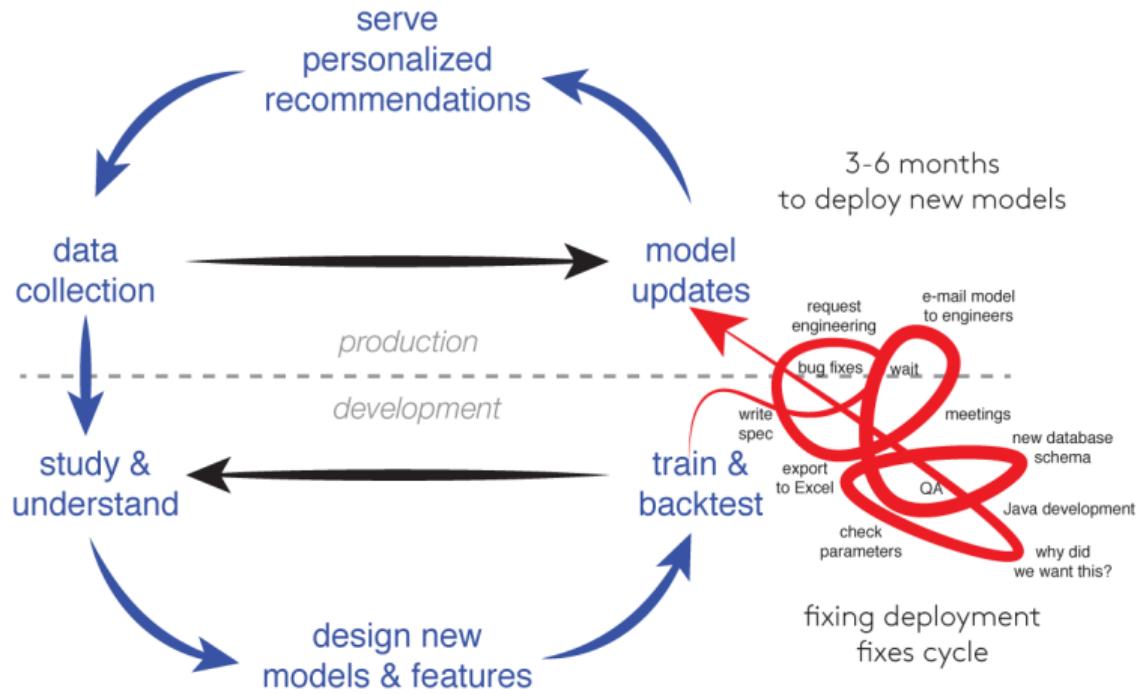


Se faire connaître

# **Les barrières à la reproductibilité**



**Figure 1 – From Uber Engineering**



**Figure 2 – The need for Agile machine learning**

**OK, mais comment?**

## AU MENU



Gestion version



Productivité



Présenter



Réutiliser

## VERSION DES DONNÉES & ÉTAPES DE PRÉTRAITEMENT



Version

## VERSION DES DONNÉES & ÉTAPES DE PRÉTRAITEMENT



Version



Gestion des versions

## VERSION DES DONNÉES & ÉTAPES DE PRÉTRAITEMENT



Version



Gestion des versions



Étapes prétraitement

## VERSION DES DONNÉES & ÉTAPES DE PRÉTRAITEMENT



Data Version Control



Dask

# CODE



Version

# CODE



Version



Différence

# CODE



Version



Différence



Divergences

# CODE

git

Git



GitHub



GitLab



Bitbucket

## AU MENU



Gestion version



Productivité



Présenter



Réutiliser

## DÉVELOPPEMENT DES MODÈLES



Réinventer

## DÉVELOPPEMENT DES MODÈLES



Réinventer



Simplification

## DÉVELOPPEMENT DES MODÈLES



Réinventer



Simplification



Facilite

# DÉVELOPPEMENT DES MODÈLES



Poutyne



PyTorch  
Lightning



Scikit-learn



Gensim



Allen NLP

## ENTRAÎNEMENT, CONFIGURATION ET RÉSULTATS



Version de  
l'entraînement

# ENTRAÎNEMENT, CONFIGURATION ET RÉSULTATS



Version de  
l'entraînement



Résultats

# ENTRAÎNEMENT, CONFIGURATION ET RÉSULTATS



Version de  
l'entraînement



Résultats



Visualisation

## ENTRAÎNEMENT, CONFIGURATION ET RÉSULTATS



Version de  
l'entraînement



Résultats



Visualisation



Erreurs  
d'entraînement

# ENTRAÎNEMENT, CONFIGURATION ET RÉSULTATS



MLflow



Hydra



Sacred



Notif

## AU MENU



Gestion version



Productivité



Présenter



Réutiliser

# RAPPORT ET ANALYSE DES RÉSULTATS



Tableau des résultats

## RAPPORT ET ANALYSE DES RÉSULTATS



Tableau des résultats



Mise à jour

## RAPPORT ET ANALYSE DES RÉSULTATS



Tableau des résultats



Mise à jour



Visualisation configuration

# RAPPORT ET ANALYSE DES RÉSULTATS



1

Python2LaTeX



TensorBoard



2

Jupyter  
notebook



Markdown



3

Dash

- 
1. Ou en HTML avec Pandas
  2. *I don't like notebooks* - Joel Grus
  3. *New York Oil and Gas*

## AU MENU



Gestion version



Productivité



Présenter



Réutiliser

# ENVIRONNEMENT



Différents environnements

# ENVIRONNEMENT



Différents environnements



Réutilisation

# ENVIRONNEMENT



Docker



kubernetes

Kubernetes

**La suite**



Itérations d'expérimentations

## POUR ALLER PLUS LOIN (EN ORDRE)

- Clean code
- Continuous Machine Learning
- Reproducibility in ML : Why it Matters and How to Achieve it
- Faire des tests !
- Writing Code for NLP Research [Gardner et al., 2018]
- *Improving Reproducibility in Machine Learning Research (A Report from the NeurIPS 2019 Reproducibility Program* [Pineau et al., 2020]
- Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning
- SOLID

## PÉRIODE DE QUESTIONS



# WEBINAIRE

# MERCI DE VOTRE ÉCOUTE !

## REFERENCES i

-  Baker, M. (2016).  
**1,500 Scientists Lift the Lid on Reproducibility.**  
*Nature News*, 533(7604) :452.
-  Gardner, M., Neumann, M., Grus, J., and Lourie, N. (2018).  
**Writing Code for NLP Research.**  
In *Conference on Empirical Methods in Natural Language Processing : Tutorial Abstracts*.
-  Garneau, N., Godbout, M., Beauchemin, D., Durand, A., and Lamontagne, L. (2020).  
**A Robust Self-Learning Method for Fully Unsupervised Cross-Lingual Mappings of Word Embeddings : Making the Method Robustly Reproducible as Well.**

## REFERENCES ii

-  Pineau, J., Vincent-Lamarre, P., Sinha, K., Larivière, V., Beygelzimer, A., d'Alché Buc, F., Fox, E., and Larochelle, H. (2020).  
**Improving Reproducibility in Machine Learning Research (A Report from the NeurIPS 2019 Reproducibility Program).**
-  Raff, E. (2019).  
**A Step Toward Quantifying Independently Reproducible Machine Learning Research.**