

Applications of Big Data - Part. 2

Vincent DOMINGUES

vincent.domingues@intervenants.efrei.net

<https://www.linkedin.com/in/vincentdmg/>



Objectifs

Déployer, à travers d'une API, un modèle entraîné de prédiction en utilisant la philosophie DevOps sur un fournisseur de services cloud.



Notation du module

Par groupe de 6 personnes, vous devrez présenter votre projet pendant une soutenance de 20 minutes et fournir tous les livrables demandés par e-mail.

Déroulement de la soutenance par groupe

15 minutes de présentation

5 minutes de questions

Livrables

- URL du référentiel Github avec tous les éléments (code, Dockerfile, configuration du workflow Github)
- Nom de l'image Docker sur Azure Container Registry (ACR)
- Endpoint API Azure Container App
- Rapport présentant, étape par étape, les choix techniques, les commandes utilisées, la stratégie de test de charge accompagnée de vos observations et les difficultés rencontrées si vous n'avez pas pu terminer.

Template

Dataset: <https://archive.ics.uci.edu/ml/datasets/iris>

Algorithme: scikit-learn KNN (K-nearest neighbor)

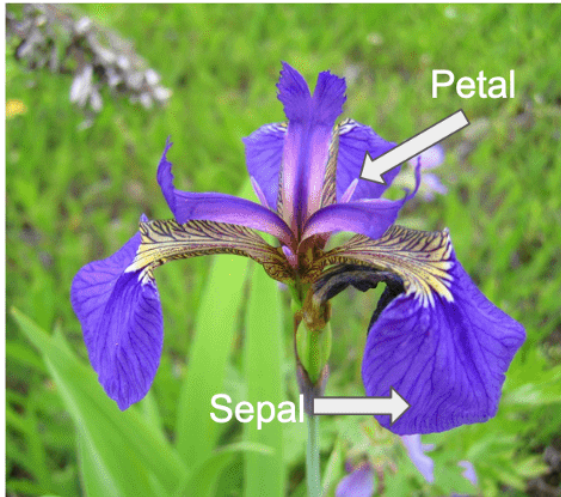
Langage de programmation: Python

Fournisseur de services cloud: Azure (Groupe de ressources : **ADDE92-CTP**)

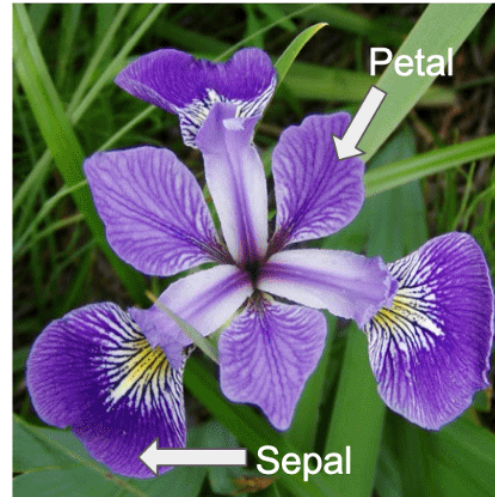
Organisation Github: EFREI-2024-ADDE92

Azure Container Registry: efreibigdata.azurecr.io

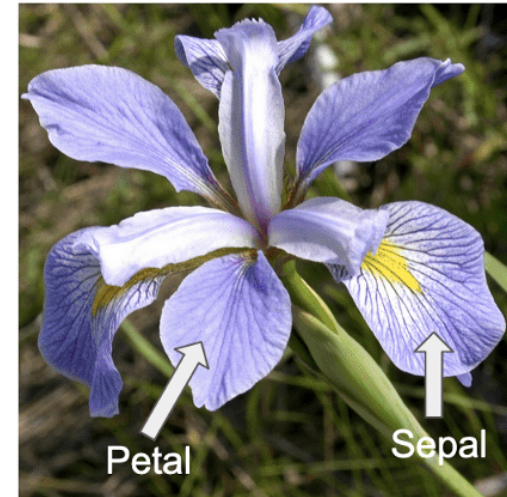
Iris setosa



Iris versicolor



Iris virginica



Étapes

- Entraînement du modèle (Iris avec scikit-learn KNN) et exportation (joblib.dump)
- Intégration du modèle entraîné dans une API
- Packaging de l'API dans une image Docker en local
- Publication du code sur Github
- Configuration de la pipeline de déploiement avec les éléments suivants
 - Construction de l'image Docker
 - Publication de l'image Docker sur Azure Container Registry (ACR)
 - Déploiement sur Azure Container App
 - Configuration de l'autoscaling en utilisant comme paramètre le nombre de requêtes en simultané
 - Réalisation d'un test de charge avec l'outil de votre choix

Dépôt des livrables

À envoyer par e-mail au plus tard 24 heures après la fin de la soutenance à

vincent.domingues@intervenants.efrei.net avec comme objet :

[PROJET-BIGDATA-GROUP_NUMBER] NOM1 NOM2 NOM3 NOM4 NOM5 NOM6

Pensez à bien renseigner le nom de chaque membre du groupe.

Bonus

- Utilisation d'un linter pour Dockerfile dans la pipeline de déploiement pour s'assurer de sa cohérence
- Ajout d'un endpoint `/metrics` en utilisant la bibliothèque `prometheus-client`, puis mise à disposition d'une ou plusieurs métriques d'utilisation de l'API (exemple : counter avec le nombre d'appels)

Informations de connexion disponibles sur l'organisation Github

AZURE_CREDENTIALS : compte de service pour s'authentifier avec l'API Azure

REGISTRY_LOGIN_SERVER : lien du serveur de la registry (efreibigdata.azurecr.io)

REGISTRY_USERNAME : nom d'utilisateur de la registry

REGISTRY_PASSWORD : mot de passe pour la registry

RESOURCE_GROUP : groupe de ressources Azure (ADDE92-CTP)

Les projets doivent être en mode `public` pour pouvoir accéder aux secrets de l'organisation

Soutenance

- Déploiement sur Azure Container Apps
- Revue de code (API, méthode d'entraînement, Dockerfile)
- Test local
- Test de l'endpoint Azure Container Apps
- Bonus
- Problèmes rencontrés