Introduction au Machine Learning

Les arbres rencontrent les forêts

Trees work together in forests...
Source : Interview with Peter Wohlleben

Objectifs du module

Complete an end-to-end machine learning project using a real data set

Pour vous donner la confiance (et un cadre général) pour résoudre les problèmes d'apprentissage automatique dans ce module, vous effectuerez une tâche d'apprentissage automatique complète sur un ensemble de données réel. Nous utiliserons un ensemble de données que vous connaissez déjà, les arbres de données ouvertes de Grenoble. Vous apprendrez à créer des pipelines pour traiter plus facilement les données des algorithmes d'apprentissage automatique. Vous utiliserez de nouveaux algorithmes tels que des arbres de décision, des forêts aléatoires et des machines vectorielles de support. Vous les utiliserez pour construire le meilleur modèle possible qui prédit «l'année de plantation» des arbres.

Démarche pédagogique

- Durée du projet : 3 jours
- Travailler en équipes de quatre
- Produire vos propres scripts individuels pour terminer le projet

Compétences développées

- Développer un projet de Machine Learning dans son intégralité

- Utiliser des pipelines pour prétraiter les données avant l'ajustement du modèle
- Faire des prédictions en utilisant différents types d'algorithmes d'apprentissage automatique
- Utiliser le réglage des hyperparamètres d'un modèle et évaluer les performances du modèle choisi
- Créer de nouvelles fonctionnalités (features) et évaluer les performances du modèle avec celles-ci
- Expliquer en termes simples comment fonctionne un algorithme ML

Contexte

La ville de Grenoble prend très au sérieux sa gestion des espaces verts. Elle prévoit de planter plus de 1 000 arbres au cours des prochaines décennies afin d'aider les citoyens à faire face au nombre croissant de jours de canicule qui sont prévus. La métropole a publié un ensemble de données ouvertes qui répertorie tous les arbres à Grenoble dans l'espoir qu'il pourrait être utilisé par les citoyens de manière innovante.

Grenoble trees need YOU!

Source : motleynews.net

Certaines données manquent ! Par exemple, certains arbres n'ont pas d'année de plantation. Il est important que les arboriculteurs sachent quand les arbres ont été plantés afin d'en prendre soin et de planifier de nouveaux plans de plantation réussis. Votre mission est d'utiliser des techniques d'apprentissage automatique pour combler ces valeurs manquantes afin d'aider la ville de Grenoble à prendre soin de ses arbres.

Etape 1

Importez, nettoyez

Objectifs de l'activité

- Téléchargez, importez et nettoyez l'ensemble de données d'arbres ouverts
- Enquêter sur la structure des données et visualiser les données pour mieux les comprendre
- Étudier un exemple de projet d'apprentissage automatique de bout en bout

Compétences

- Terminer un projet ML du début à la fin (pour un problème de régression)

Consignes

- Regardez les exemples de projets d'apprentissage automatique de bout en bout suggérés ci-dessous, exécutez leur exemple de code, **lisez les explications dans les cahiers jupyter** et dans le livre
- Importez les données
- Supprimer les lignes avec des valeurs manquantes «année de plantation»
- Examiner la structure des données
- Pensez à **enregistrer vos données semi-préparées sous forme de fichier pickle** pour les utiliser à l'étape suivante. Cela vous permettra d'utiliser des cahiers plus petits séparés pour chaque étape!

Ressources

- Données d'arbres ouverts : https://data.metropolegrenoble.fr/visualisation/information/?id=arbres-grenoble&disjunctive.sous_categorie_desc&disjunctive.espece&location=12,45.18821,5.74699
- Exemple de "End-to-end machine learning project" dans scikit-learn (voir chapitre 2) : https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/
- Référentiels avec le code associé à l'exemple ci-dessus : https://github.com/ageron/handson-ml2
- Blog du visualisation de données manquantes : https://dev.to/tomoyukiaota/visualizing-the-patterns-of-missing-value-occurrence-with-python-46dj

Livrables

- Script / cahier python contenant :
    - Importation de données
    - Sauvegarde des fichiers de données sous forme de 'pickle'

Etape 2

Préparer les données, former et évaluer les modèles ML

Objectifs de l'activité

- Utiliser des pipelines Scikit-Learn pour préparer les données pour les modèles d'apprentissage automatique
- Appliquer plusieurs modèles d'apprentissage automatique non accordés pour prédire « l'année de plantation »
- Réglez les hyperparamètres d'un modèle

Compétences

- Terminer un projet ML du début à la fin (pour un problème de régression)
- Utiliser des pipelines pour prétraiter les données avant l'ajustement du modèle
- Faire des prédictions en utilisant différents types d'algorithmes d'apprentissage automatique
- Utiliser le réglage hyperparamétrique et évaluer les performances d'un modèle choisi

Consignes

- Suivez le processus de la ressource "end-to-end machine learning project" pour prédire «l'année de plantation» (n'ajoutez pas de variables supplémentaires à cette étape)
- Visualisez l'ensemble de données d'arbres pour comprendre ce qu'il contient
- Envisagez de créer un modèle naïf à utiliser comme référence pour les performances. Évaluer ses performances sur l'ensemble d'entraînement.
- Former plusieurs types différents de modèles d'apprentissage automatique (sans réglage). [Si vous utilisez des forêts aléatoires, commencez par un modèle utilisant 'n_estimators = 10']
- Choisissez l'un des modèles les plus prometteurs et réglez ses hyperparamètres
- Évaluer les performances du modèle réglé sur l'ensemble de test

Livrables

- Script / cahier python contenant :
    - Au moins une visualisation
    - Un pipeline scikit-learn pour le prétraitement des données
    - Évaluation des performances d'au moins trois performances de modèles non réglés
    - Évaluation des performances d'un modèle final réglé

Etape 3

Ajoutez de nouvelles variables à votre modèle pour améliorer les performances

Objectifs de l'activité

- Tenter d'ajouter des variables pour améliorer les performances du modèle
- Prenez un peu de temps pour comprendre comment fonctionnent les nouveaux algorithmes ML que vous utilisez

Compétences

- Terminer un projet ML du début à la fin (pour un problème de régression)
- Créez de nouvelles variables (features) et évaluez les performances du modèle avec celles-ci
- Expliquer en termes simples comment fonctionne un algorithme ML

Consignes

- La ressource de « end-to-end machine learning project » décrit comment créer de nouvelles variables susceptibles d'améliorer les performances du modèle. Essayez de créer au moins une nouvelle variable en utilisant les données que vous possédez déjà (si vous le pouvez, essayez de les intégrer dans votre pipeline scikit-learn).
- Évaluez les performances de votre nouveau modèle sur l'ensemble d'entraînement pour voir si ces nouvelles variables améliorent les performances.
- Si vous trouvez de nouvelles variables qui, selon vous, peuvent améliorer les performances du modèle. Évaluez votre nouveau modèle sur votre kit de test.
- Produisez un mémo qui décrit dans vos propres mots les principes / processus de base de deux éléments suivants : arbre de décision, forêt aléatoire, machine à vecteurs de support.

Livrables

- Script / cahier python contenant : :
    - Création d'une nouvelle fonctionnalité (de préférence dans un pipeline scikit-learn)
    - La formation et l'évaluation d'un modèle ML qui comprend les nouvelles fonctionnalités
    - Une évaluation des performances de ce nouveau modèle
- Un mémo qui décrit dans vos propres mots les principes / processus de base de deux éléments suivants: arbre de décision, forêt aléatoire, machine à vecteurs de support.

Pour aller plus loin

- Découvrez quelles variables sont les plus importantes dans votre modèle.
- Comparez les meilleures performances de votre modèle avec celles de la classe. Quels modèles fonctionnent mieux? Selon vous, pourquoi? Y a-t-il quelque chose que vous pouvez utiliser à partir de leur modélisation pour améliorer votre modèle?
- Y a-t-il d'autres données ouvertes que vous pourriez ajouter pour améliorer les performances de votre modèle? Produisez quelques recommandations pour d'autres fonctionnalités qui pourraient être ajoutées.
- Il manque dans les lignes des données manquantes «annee de plantation» dans les données des arbres de Grenoble certaines colonnes que vous avez pu utiliser dans le modèle.

Essayez de créer un nouveau modèle en utilisant uniquement les données d'entité disponibles. Produisez une évaluation des performances que vous attendez de votre nouveau modèle à fonctionnalités réduites. Faites un csv contenant les prédictions des données manquantes à l'aide de votre modèle.