Test Technique Quantmetry

Par : Idris Dada

1. <u>Statistiques descriptives :</u>

1. Décrivez le jeu de données. Présentez seulement les analyses et éventuels retraitements qui vous paraissent les plus pertinents et faites une première conclusion sur les variables à sélectionner en vue de la prédiction du succès ou de l'échec d'une candidature.

- On remarque que la dataset est déséquilibré et qu'elle contient majoritairement des qui variable catégorielles de ce fait un Processing, pour encoder ces catégories en valeurs numérique.
- J'ai choisi de supprimer les valeurs Null qui représentait 2% de la dataset
- j'ai corrigé quelque Outliers sur l'Age et l'expérience
- on remarque que les variables numérique suivent une loi de distribution normale ce qui favorise un modèle linéaire
- 'J'ai représenté la série temporelle d'embauche, j'ai remarqué une tendance et une saisonnalité sur le rythme d'embauche, c'est pour cela que j'ai préféré prendre en considération que le mois d'embauche comme indicateur significatif.
- Des intervalles de valeur on était défini pour réduire la variance de ces données :

age : 18, 30, 45, 65, 75
expérience : 0, 5, 10, 25
salaire : 14k - 25k - 35k - 45k - 55k

- Une matrice de corrélation a était établie afin de révéler de potentielle similitude de variable, mais sans succès

- Voici un tri des variables par dépendance et d'importance vis-à-vis de la décision d'embauche :

['note' 'cheveux' 'exp' 'dispo' 'age' 'specialite' 'sexe' 'diplome' 'salaire']

2. Y a-t-il une dépendance statistiquement significative entre :

(a) La spécialité et le sexe ?

Non, mais on peut dire que y a une légère dépendance négligeable.

	specialite	sexe
specialite	1.00	0.37
sexe	0.37	1.00

(b) La couleur de cheveux et le salaire demandé ?

Non

	cheveux	salaire
cheveux	1.00	0.02
salaire	0.02	1.00

(c) Le nombre d'années d'expérience et la note à l'exercice ?

Non

	exp	note
exp	1.00	-0.01
note	-0.01	1.00

2. Machine Learning

1. Pour cette problématique de classification binaire, j'ai opté pour un modèle à base d'arbre de décision, un **RandomForest** basé sur un algorithme de bagging qui permettra de créer plusieurs instances d'arbre et d'agréger les prédictions de chaque arbre afin de déduire le résultat final, mon choix, c'est porté sur cet algo, car la dataset comprends plusieurs valeurs catégorielles, pour ce fait les arbres de décision sont le meilleur choix.

    a. Hyperparametres:
        - n_estimators : 200

        qui représente le nombre d'arbres de décision à créer.
        - max_depth = 20

        représente la profondeur de l'arbre plus l'arbre est grand plus il a tendance à overfiter
        - class_weight='balanced'

        j'ai appliqué cette métrique pour pondérer l'impact de chaque ligne par rapport à sa classe de prédiction
        - min_samples_split = 5

        qui représente le nombre d'exemples à prendre pour calculer le criteron (Ginni) afin de diviser la décision

    b. On a eu un modéle avec :
        F1 score : 74
        Par contre, vu la dataset unbalancé on doit prendre en compte le F1-score du Label 1 qui représente 15% de la dataset.
        Ce dernier atteint : 55 sur test

2. les variables les plus importantes du modèle sont les suivantes :
    ['date', 'cheveux', 'note']

3. Pour évaluer notre modèle, j’ai choisi le F1 score un compromis idéal entre Précision et Recall un critère important quand on a affaire à une dataset déséquilibré.

4. Afin d’améliorer mon modèle, je propose d’appliquer une GridSearch afin d’optimiser les Hyperparametre du modèle, est un stacking de modèle afin de créer un modèle hybrid ensembliste afin de tirer profit des différentes familles d’algorithmes.
   Enfin, une imputation des lignes de données contenant des nans pourrais être récupéré et imputer en utilisant un KNN afin de remplir les vides trouvés, en calculant les similitudes avec d’autre point de la dataset.