BOUDOU Mathias, BOURRIER Maxime, DERAMBURE Gaëlle

TD - Atelier Composant Métier

Variables qualitatives	**Variables quantitatives**
APP_Libelle_etablissement Adresse_2_UA Libelle_commune Date_inspection APP_Libelle_activite_etablissement Synthèse_eval_sanit Agrément filtre ods_type_activite	SIRET Code_postal Numero_inspection geores

CONSIGNES

- Trouver une problématique
- Émettre au moins une hypothèse et y répondre
- Pas de limite de modèle
- A fournir : Github: code + fichier d'explication

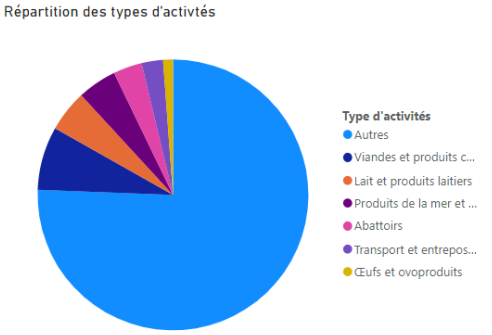
Problématiques :

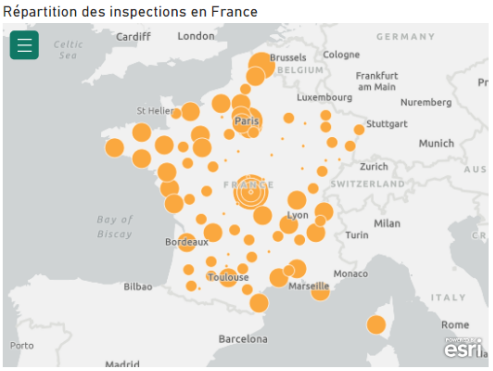
Contexte : La société 'SantéResto' exploite une chaîne de restaurants et souhaite maintenir des normes sanitaires élevées. Pour anticiper les inspections, ils développent un modèle de prédiction basé sur les données d'inspection passées, les pratiques d'hygiène, et d'autres variables. L'objectif est d'anticiper et d'améliorer les évaluations sanitaires de leurs restaurants pour garantir la sécurité des clients et le respect des réglementations.

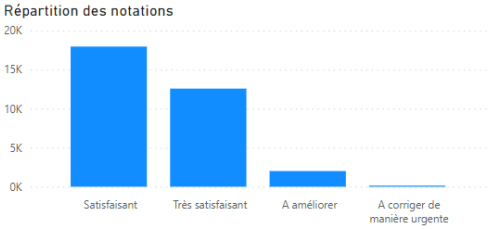
Comment améliorer la qualité des inspections sanitaires des établissements manipulant de la nourriture ?

Visualisation des données :

Voici quelques graphiques nous permettant de nous rendre compte des données disponibles afin de nous aider à mieux interpréter nos résultats.







32,72K
Taille de l'échantillon

Hypothèses :

- Souligner la corrélation entre la propreté (Synthese_eval_sanit) de l' établissement et son type (APP_Libelle_activite_etablissement).
    - On pourrait rajouter le type d'activité (ods_type_activite)
- D'après la géolocalisation (geores) ou le département de l'établissement (2 premiers caractères de Code_postal), prédire le niveau de propreté de l'établissement

Modèle :

Modèle 1 - D'après le libellé :

Nous avons choisi d'utiliser un modèle logistique multinomial car nous avons des variables catégorielles avec plus de deux catégories (Satisfait, Très satisfait…).
De plus, il était nécessaire d'encoder ces variables afin de les utiliser dans notre modèle.

Nous avons donc utilisé Labelencoder pour Synthese_eval_sanit pour encoder les étiquettes de classe en valeurs numériques.
Puis nous avons utilisé OneHotEncoder pour APP_Libelle_activite_etablissement pour transformer les catégories en variables binaires.

Les données ont été divisées en ensemble d'entraînement de test pour l'entraînement des données. Cela permet d'évaluer la performance du modèle. Ici nous avons un total de 6544 données ce qui représente 20% de l'ensemble des données du fichier csv.

Pour finir nous avons voulu faire des prédictions sur les évaluations sanitaires avec l'ensemble des tests. Les performances du modèle sont évaluées en utilisant l'accuracy et le rapport de classification(précision, rappel, F-score). Ici nous avons une précision de 0.61.

Nom du fichier sur Github : (modele_LibelleSanitaire_Regression_Logistic_Multinomial.py)

Résultat :

```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.01      0.01       425
           1       0.00      0.00      0.00        24
           2       0.59      0.96      0.73      3594
           3       0.77      0.23      0.35      2501

    accuracy                           0.61      6544
   macro avg       0.59      0.30      0.27      6544
weighted avg       0.68      0.61      0.54      6544
```

Modèle 2 - D'après la localisation :

1. D'après le code département

Modèle régression logistique

Pourquoi ?

Dans un premier temps, nous avons choisi d'utiliser une régression logistique. Nous avons choisi d'utiliser ce modèle car il est bon pour cibler une variable discrète. Ici, nous ciblons la note pour l'évaluation sanitaire, cette note pouvant avoir 4 valeurs différentes (À améliorer, A corriger de manière urgente, Satisfaisant, Très satisfaisant) donc elle est bien discrète.

Nous avons donc commencé par mapper les valeurs possibles des notes pour manipuler des entiers :

```
"A améliorer": 0,
"A corriger de manière urgente": 1,
"Satisfaisant": 2,
"Très satisfaisant": 3
```

Et nous avons récupéré les 2 premiers caractères de la colonne Code_postal pour avoir le numéro de département concerné.

Puis nous avons suivi le même principe qu'avant pour entraîner notre modèle

Résultat :

```
Précision du modèle : 54.58%
Prédiction pour le département  25  : 2
```

Nom du fichier sur Github : (modele_DepartementSanitaire_LogisticRegression.py)

Modèle Support Vector Regression

Pourquoi ?

Comme le modèle de régression logistique, le modèle SVR est utilisé pour régler des problèmes de classification (ici nos notes). Nous voulions donc tester deux modèles différents pour voir si l'un était plus performant que l'autre, toujours en utilisant le code de département comme paramètre pour prédire la note à l'évaluation sanitaire.

Nous avons donc suivi les mêmes étapes que juste avant, mais en utilisant ce nouveau modèle.

Résultat :

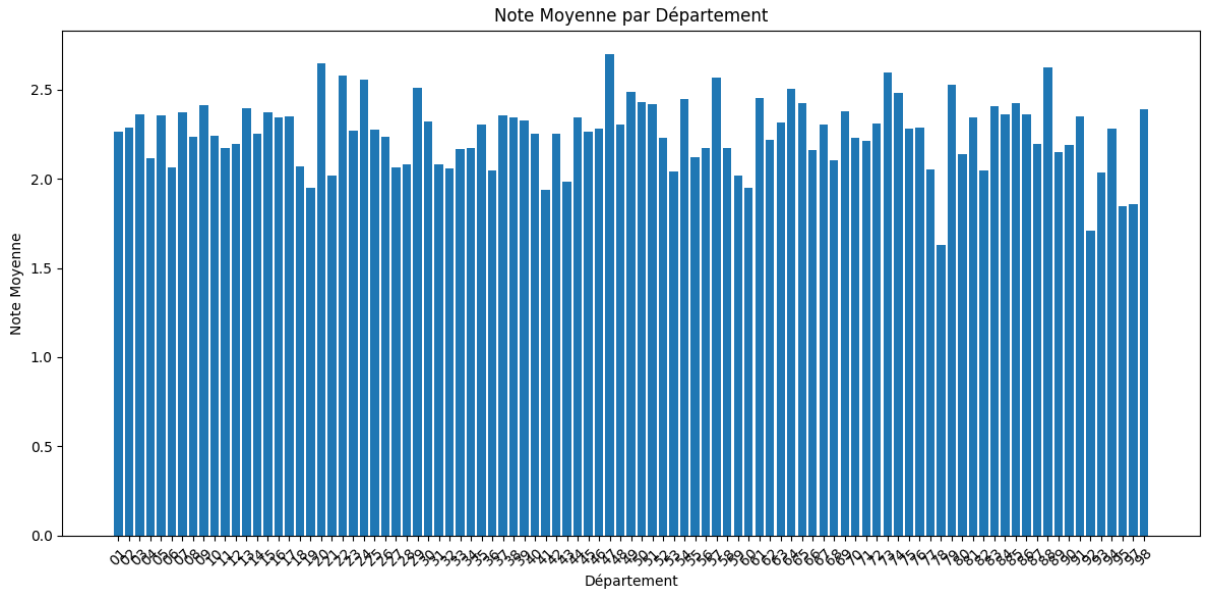
```
Précision du modèle : 54.56%
Prédiction pour le département  25  : 2
```

Les résultats et la précision sont très similaires, les deux modèles ont donc l'air aussi performants l'un que l'autre.

Nom du fichier sur Github : (modele_DepartementSanitaire_SVR.py)

Diagramme

Par ailleurs, nous avons créé un diagramme pour visualiser la répartition des notes moyennes par département :



Nom du fichier sur Github : (diagramme_LongitudeLatitude_Sanitaire.py)

2. D’après le code département, la Longitude et la Latitude

Modèle régression logistique

Nous avons voulu essayer d’utiliser le premier modèle que nous avions fait (avec comme paramètre département, en utilisant une régression logistique), mais cette fois en utilisant la colonne geores.

Nous avons donc séparé Longitude / Latitude qui sont présents dans cette colonne et nous les avons ajoutés au modèle précédent. Pour récupérer le code du département, nous avons procédé comme auparavant.

Résultat :

```
Précision du modèle : 53.68%
```

La précision n’est pas meilleure que le modèle précédent.

Nom du fichier sur Github : (modele_DepartementLongitudeLatitude_LogisticRegression.py)