Rapport d'Analyse et de Modélisation des Données du Play Store

# I - Introduction

Dans le cadre de ce projet, nous avons analysé des données provenant du Play Store afin d'aider deux utilisateurs spécifiques, Mme Ajmi et M. Demoli, à sélectionner les applications les plus pertinentes en fonction de leurs intérêts et préférences. Les utilisateurs sont souvent confrontés à un grand nombre d'applications similaires, rendant difficile la sélection de l'application la plus adaptée à leurs besoins spécifiques.

Les données utilisées proviennent de deux fichiers principaux du Play Store : `googleplaystore.csv` pour les informations sur les applications et `googleplaystore_user_reviews.csv` pour les avis des utilisateurs. Ces données incluent des attributs tels que le nom de l'application, la catégorie, la note, le nombre de téléchargements, le type (gratuite ou payante), le prix, et le groupe d'âge cible.

L'objectif est de filtrer et recommander les applications adaptées aux intérêts spécifiques des utilisateurs. Mme Ajmi est intéressée par les applications de "FAMILY" et "BUSINESS", tandis que M. Demoli cherche des applications de "TOOLS" et "PRODUCTIVITY". Nous avons appliqué diverses techniques de machine learning pour atteindre cet objectif, en utilisant des méthodes comme la régression logistique, le K-Nearest Neighbors (KNN) et les arbres de décision, en les calibrant et en optimisant leurs paramètres pour obtenir les meilleurs résultats possibles.

---

# II - Qualité et Justifications des Choix d'Analyse

## Préparation des Données

1. **Nettoyage des Données** :
    - Les valeurs manquantes critiques ont été supprimées pour éviter les biais dans l'analyse.
    - Les colonnes catégorielles (`Type`, `Content Rating`) ont été encodées en variables numériques à l'aide de `OneHotEncoder`.
2. **Filtrage des Données** :
    - Les données ont été filtrées pour ne conserver que les catégories pertinentes pour Mme Ajmi (`FAMILY`, `BUSINESS`) et M. Demoli (`TOOLS`, `PRODUCTIVITY`).
3. **Normalisation des Données** :
    - Les colonnes `Rating`, `Installs`, et `Reviews` ont été normalisées pour mettre toutes les valeurs sur une échelle commune de 0 à 1, assurant une comparabilité optimale des caractéristiques.

## Visualisations

1. **Distribution des Catégories d'Applications** :
    - Un diagramme en barres a été utilisé pour visualiser la distribution des applications par catégorie, permettant de comprendre la répartition des applications.
2. **Distribution des Ratings par Catégorie** :

    - Un boxplot a été utilisé pour montrer la distribution des notes par catégorie, facilitant l'identification des tendances et des outliers.

3. **Distribution des Prix** :
    - Un histogramme a été utilisé pour visualiser la distribution des prix des applications, montrant la majorité des applications gratuites ou à faible coût.

4. **Applications les Plus Populaires** :
    - Une analyse a été effectuée pour identifier les applications les plus populaires dans chaque catégorie, basée sur le nombre d'installations et la note moyenne.

5. **Tendance des Notes au Fil du Temps** :
    - Un graphique linéaire a été utilisé pour montrer l'évolution des notes moyennes des applications au fil des ans, permettant de visualiser les tendances de qualité des applications.

### Analyse des Sentiments des Avis Utilisateurs

1. **Polarité et Subjectivité** :
    - Les distributions de la polarité et de la subjectivité des sentiments ont été analysées à l'aide d'histogrammes avec une courbe de densité, fournissant des insights sur les avis des utilisateurs.

2. **Prédiction de la Polarité et de la Subjectivité** :
    - Des modèles de machine learning ont été utilisés pour prédire la polarité et la subjectivité des avis, permettant d'évaluer automatiquement les sentiments des utilisateurs et d'améliorer les recommandations d'applications.

Ces choix d'analyse et de visualisation permettent de répondre aux questions d'intérêt des utilisateurs et de fournir des recommandations d'applications précises et pertinentes, améliorant ainsi leur expérience sur le Play Store.

---

## III - Qualité des Visualisations

### Lisibilité et Clarté

Toutes les visualisations ont été conçues pour être facilement compréhensibles. Les titres, les axes et les légendes sont clairement étiquetés. Les diagrammes en barres et les boxplots utilisent des couleurs contrastées pour améliorer la lisibilité. Les histogrammes sont accompagnés de courbes de densité pour mieux représenter les distributions.

### Choix des Visualisations

Les types de visualisations ont été choisis en fonction de l'analyse effectuée :

- Les diagrammes en barres pour montrer la distribution des catégories.
- Les boxplots pour comparer les distributions des ratings entre différentes catégories.
- Les histogrammes pour analyser les distributions des prix, de la polarité et de la subjectivité.

- Les graphiques linéaires pour observer les tendances temporelles.

### Originalité et Validité des Conclusions

Les conclusions tirées des visualisations sont basées sur des observations claires et cohérentes. Par exemple, l'analyse de la distribution des prix montre que la majorité des applications sont gratuites ou peu coûteuses, ce qui est cohérent avec les attentes des utilisateurs. La distribution des ratings par catégorie aide à identifier les catégories avec les meilleures notes, offrant ainsi des recommandations pertinentes.

---

## IV - Protocole de Préparation des Données pour la Modélisation

### Protocole de Préparation des Données pour la Modélisation

1. **Séparation des Données** :
    - Les données utilisées proviennent du fichier `googleplaystore.csv` pour les caractéristiques des applications et du fichier `googleplaystore_user_reviews.csv` pour les avis des utilisateurs.
    - Les colonnes pertinentes incluent : `App`, `Category`, `Rating`, `Reviews`, `Size`, `Installs`, `Type`, `Price`, `Content Rating`, `Genres`, `Last Updated`, `Current Ver`, `Android Ver`.

2. **Préparation des Données** :
    - Les valeurs manquantes ont été traitées en supprimant les lignes avec des valeurs manquantes critiques.
    - Les colonnes catégorielles (`Type`, `Content Rating`) ont été encodées en variables numériques à l'aide de `OneHotEncoder`.
    - Les caractéristiques non numériques inutiles pour la modélisation ont été exclues.

3. **Séparation Apprentissage/Test** :
    - Les données ont été divisées en ensembles d'apprentissage (80%) et de test (20%) en utilisant `train_test_split` avec un état aléatoire (random_state=42) pour assurer la reproductibilité.
    - Cette séparation permet de former le modèle sur un sous-ensemble des données et de l'évaluer sur un sous-ensemble séparé pour tester sa performance sur des données non vues.

### Modélisation et Évaluation avec Régression Logistique

1. **Modélisation** :
    - Le modèle de régression logistique a été entraîné sur les données d'apprentissage et évalué sur l'ensemble de test.
    - Précision obtenue : 67.94%.

2. **Résultats de la Prédiction** :
    - Les résultats montrent que la régression logistique capture bien certaines relations linéaires, mais sa précision reste modérée.

### Conclusion des Prédictions pour Mme Ajmi et M. Demoli

1. **Mme Ajmi** :

- Les applications recommandées sont principalement des applications "FAMILY" et "BUSINESS".
- Exemple : "Google Play Games", "OfficeSuite : Free Office + PDF Editor".

2. **M. Demoli** :

- Les applications recommandées sont principalement des applications "TOOLS" et "PRODUCTIVITY".
- Exemple : "Google Drive", "Google Translate".

---

# Modélisation avec K-Nearest Neighbors (KNN)

## Analyse et Évaluation avec K-Nearest Neighbors (KNN)

1. **Sans Standardisation** :

- Précision : 64.43%.

2. **Avec Standardisation** :

- Précision : 72.73%.

3. **Optimisation des Hyperparamètres** :

- Utilisation de `GridSearchCV` pour trouver le meilleur `n_neighbors`.
- Précision optimale avec `n_neighbors=17` : 74.36%.

---

# Évaluation avec Distance de Manhattan

1. **KNN avec Distance de Manhattan** :

- Précision : 73.05%.

2. **Sélection des Meilleures Caractéristiques** :

- Utilisation de `SelectKBest` pour sélectionner les 10 meilleures caractéristiques.
- Précision : 71.93%.

---

## Modélisation et Évaluation avec un Arbre de Décision

1. **Arbre de Décision avec Profondeur Max 5** :

- Précision : 73.21%.

2. **Arbre de Décision avec Profondeur Max 3** :

- Précision : 72.57%.

3. **Ajustement de la Visualisation** :

- Utilisation de `plot_tree` pour visualiser l'arbre de décision.

### Normalisation des Données

1. **MinMaxScaler** :
    - Normalisation des colonnes `Rating`, `Installs`, `Reviews`.
    - Utilisation de la normalisation pour améliorer les performances des modèles.

### Analyse de la Normalisation des Données

La normalisation des données a permis d'améliorer les performances des modèles, en particulier le KNN, en assurant que toutes les caractéristiques sont sur une échelle commune.

### Prédiction

de la Polarité et de la Subjectivité des Avis

1. **Modèle de Polarité** :
    - Utilisation de la régression logistique.
    - Précision : 87.61%.
2. **Modèle de Subjectivité** :
    - Utilisation de RandomForestClassifier.
    - Précision : 94.17%.

---

## V - Évaluation de la Qualité des Résultats

### Protocole d'Évaluation

1. **Séparation des Données** :
    - Les données ont été divisées en ensembles d'apprentissage (80%) et de test (20%) pour évaluer la performance des modèles.
2. **Validation Croisée** :
    - Utilisation de `GridSearchCV` pour optimiser les hyperparamètres du modèle KNN.

### Tableau des Résultats Obtenus

| **Modèle** | **Précision (%)** |
|---|---|
| Régression Logistique | 67.94 |
| KNN (non standardisé) | 64.43 |
| KNN (standardisé) | 72.73 |
| KNN (optimisé, n_neighbors=17) | 74.36 |
| KNN (Distance de Manhattan) | 73.05 |
| KNN (Sélection des Caractéristiques) | 71.93 |

| Modèle | Précision (%) |
|---|---|
| Arbre de Décision (profondeur max 5) | 73.21 |
| Arbre de Décision (profondeur max 3) | 72.57 |
| Modèle de Polarité (Régression Logistique) | 87.61 |
| Modèle de Subjectivité (Random Forest) | 94.17 |

### Interprétation des Résultats

**Régression Logistique** :

- Modèle simple et rapide avec une précision modérée.

**K-Nearest Neighbors (KNN)** :

- La standardisation et l'optimisation des hyperparamètres ont significativement amélioré la précision, faisant du KNN optimisé l'un des modèles les plus performants.

**Arbre de Décision** :

- Bonne capacité à capturer les relations non linéaires, mais avec une précision légèrement inférieure au KNN optimisé.

**Évaluation avec Distance de Manhattan** :

- La précision obtenue est compétitive avec les autres modèles testés.

**Sélection des Meilleures Caractéristiques** :

- La sélection a amélioré la précision, soulignant l'importance de cette étape dans le processus de modélisation.

**Prédiction des Sentiments** :

- Les modèles de polarité et de subjectivité des avis utilisateurs ont montré des précisions élevées, améliorant la capacité à comprendre les sentiments des utilisateurs.

---

## Conclusion Générale

1. **Meilleure Méthode** :
    - Le KNN optimisé avec `n_neighbors=17` a montré une des meilleures précisions (74.36%).
2. **Interprétabilité** :
    - Les arbres de décision fournissent des insights précieux sur les critères de décision.
3. **Simplicité et Rapidité** :
    - La régression logistique est rapide et simple, mais avec une précision inférieure.

Ces résultats permettent de fournir des recommandations d'applications précises et pertinentes à Mme Ajmi et M. Demoli, améliorant ainsi leur expérience sur le Play Store.