Budget Participatif et Dette Propre : Deux Dynamiques Indépendantes

Présentation du Projet.

Cette étude se concentre sur la relation entre le capital initial des dettes propres et le montant du budget alloué aux projets du budget participatif. Nous utiliserons une analyse de régression linéaire pour examiner si un capital initial plus important est associé à un budget total plus bas, en contrôlant pour d'autres facteurs tels que la taille du projet et le Thématique du projet,..etc.

Data Source :

-**Budget Participatif - Les opérations des projets lauréats jusqu'à leurs réalisations(Citoyenneté):** https://opendata.paris.fr/explore/dataset/budget-participatif_operations-projets-gagnants-realisations/information/?disjunctive.thematique&disjunctive.type_financement_operation&disjunctive.arrondissement_operation&disjuncti

-**Dette de la Ville de Paris(administration et finance publique):** https://opendata.paris.fr/explore/dataset/dette-propre/information/?disjunctive.nature&disjunctive.organisme_preteur_ou_chef_de_file&disjunctive.type_de_taux_d_interet&disjunctiv

Nous formulons l'hypothèse suivante : **plus le montant des dettes propres est élevé, moins le budget total alloué à financer les projet sera important**. En d'autres termes, le recours aux dettes propres pourrait entraîner une diminution du financement global des projets et limiter leur portée.

- Quels sont les critères utilisés pour mesurer **l'avancement d'un projet** ? S'agit-il d'indicateurs quantitatifs (dépenses engagées) ou qualitatifs (Thématique, si l'operation est en quatier populaire ou pas, type de financement du projet.. etc)?
- Existe-t-il un niveau d'endettement à partir duquel l'avancement des projets est **significativement affecté** ?
- Comment comparons-nous l'avancement des projets dans les quartiers populaires par rapport aux autres quartiers ?
- Parmi les différentes **thématiques** étudiées, quelle est celle qui reçoit généralement le **budget global le plus élevé** pour ses projets lauréats ? Existe-t-il une différence significative entre les budgets alloués aux différentes thématiques ?
- Quelles sont les variables les plus pertinentes pour **prédire l'avancement du projet**, en plus de celles déjà mentionnées ?
- Comment **optimiser** la taille et la complexité de l'**arbre de décision** pour éviter le sur-apprentissage ?

Outils utilisés:

- Jupyter Notebook (Python)
- Power BI

Étude de la corrélation entre le niveau d'endettement et le budget alloué au financement de projets.

Récupération et Préparation des Données et de new DATAFRAME pour l'analyse

- **Récupération Par API:**

```
url1="https://opendata.paris.fr/api/explore/v2.1/catalog/datasets/dette-propre/records?limit=20"
r1=requests.get(url1)
print (r1.status_code)
if r1.status_code == 200:
    json_data1=r1.json()
    for key, value in json_data1.items():
        print(key+ ':',value)

else:
    print("Failed to fetch data:",r1.status_code)
```

```
import requests
import pandas as pd
url="https://opendata.paris.fr/api/explore/v2.1/catalog/datasets/budget-participatif_operations-projets-gagnants-realisations/records?limit=20&refine=t
r=requests.get(url)
print (r.status_code)
if r.status_code == 200:
    json_data=r.json()
    for key, value in json_data.items():
        print(key+ ':',value)

else:
    print("Failed to fetch data:",r.status_code)
```

- **Aprés la creation de new dataframe, j'ai calculé la correclation:**

```
   Year  capital_initial  budget_global_projet_gagnant
0  2015     1.150263e+08                       1500000
1  2017     5.000000e+07                         15000
2  2018     9.650000e+07                      14759000
```

```
#Correlation_analysis
correlation = new_df["capital_initial"].corr(new_df["budget_global_projet_gagnant"])
print(f"Correlation between City Debt Level and Budget for Project Funding: {correlation}")
```

```
Correlation between City Debt Level and Budget for Project Funding: 0.3288056782005793
```

▼ **Peut-on faire confiance à l'exactitude et à la fiabilité de ces résultats ?**

Face aux limitations de l'API qui renvoyait un nombre restreint de données, potentiellement introduisant un biais, j'ai d'abord opté pour une récupération par lots **(Batch Fetching)**. Cependant, cette méthode s'est avérée chronophage. Pour gagner en efficacité et éviter les biais, j'ai finalement choisi d'extraire les données directement d'un fichier Excel existant en utilisant Pandas, permettant une analyse plus rapide et fiable de la corrélation.

- **Récupération Par Download:**

```
#import data as xlsx
# Install openpyxl if not already installed
!pip install openpyxl

# Read the Excel file
Budget_Participatif = pd.read_excel("budget-participatif_operations-projets-gagnants-realisations (1).xlsx")

Budget_Participatif.head(100)
```

```
Dette_Propre = pd.read_excel("dette-propre.xlsx")

Dette_Propre.head(100)
```

```
Budget_Participatif.info()
```

```
Dette_Propre.info()
```

```
# Check for missing values
missingvalues = Budget_Participatif.isnull().sum()
print(missingvalues)
```

- **Refaire les memes etapes pour Creer new Dataframe pour afin d'observer la correlation:**

[26]:
```
# Extract year from relevant columns (assuming "Edition" and "Année de publication" are already in integer format)
Budget_Participatif["Year"] = Budget_Participatif["Edition"]
Dette_Propre["Year"] = Dette_Propre["Année de publication"]

# Filter data based on common years (if necessary)
min_year = min(Budget_Participatif["Year"].min(), Dette_Propre["Year"].min())
max_year = max(Budget_Participatif["Year"].max(), Dette_Propre["Year"].max())

Budget_Participatif = Budget_Participatif[(Budget_Participatif["Year"] >= min_year) & (Budget_Participatif["Year"] <= max_year)]
Dette_Propre = Dette_Propre[(Dette_Propre["Year"] >= min_year) & (Dette_Propre["Year"] <= max_year)]

# Calculate annual debt
annual_debt_df = calculate_annual_debt(Dette_Propre.copy(), "Year", "Capital initial")

# Calculate annual project budget
annual_budget_df = calculate_annual_project_budget(Budget_Participatif.copy(), "Year", "Budget global du projet lauréat")

# Combine data into new DataFrame
new_df = pd.merge(annual_debt_df, annual_budget_df, on="Year")

# Explore and analyze new_df
print(new_df.head())
```

```
   Year  Capital initial  Budget global du projet lauréat
0  2014     4.062139e+09                        753600000
1  2015     4.646021e+09                       1180105000
2  2016     5.193125e+09                       1874707407
3  2017     5.785625e+09                       1095851100
4  2018     5.901314e+09                        663421900
```

Comparons les résultats actuels à ceux obtenus précédemment via l'API :

	Year	capital_initial	budget_global_projet_gagnant
0	2015	1.150263e+08	1500000
1	2017	5.000000e+07	15000
2	2018	9.650000e+07	14759000

▼ **Corrélation:**

:
```
#Correlation_analysis
correlation = new_df["Capital initial"].corr(new_df["Budget global du projet lauréat"])
print(f"Correlation between City Debt Level and Budget for Project Funding: {correlation}")
```

```
Correlation between City Debt Level and Budget for Project Funding: 0.02791515735608592
```

La corrélation entre le niveau d'endettement des villes et le budget alloué aux projets est très faible (`0.02791515735608592`).

Cette faible corrélation indique qu'il n'y a pas de relation significative entre ces deux variables.

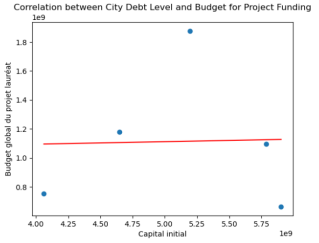
L'analyse initiale basée sur les données API s'est avérée erronée(`0.3288056782005793`).

Le niveau d'endettement d'une ville n'est probablement pas un facteur déterminant dans l'allocation des budgets pour les projets.

D'autres facteurs (taille de la ville, population, activité économique, politiques locales) pourraient avoir une influence plus importante sur le budget alloué aux projets.

- Nuage de points de **Capital initial** contre **Budget global du projet lauréat:**

```
import statsmodels.api as sm #for linear regression modeling



# Fit a linear regression model
X = sm.add_constant(new_df["Capital initial"])  # Add a constant term for the intercept
y = new_df["Budget global du projet lauréat"]
model = sm.OLS(y, X).fit()

# Create a scatter plot
plt.scatter(new_df["Capital initial"], new_df["Budget global du projet lauréat"])

# Plot the regression line
plt.plot(new_df["Capital initial"], model.fittedvalues, color='red')

plt.xlabel("Capital initial")
plt.ylabel("Budget global du projet lauréat")
plt.title("Correlation between City Debt Level and Budget for Project Funding")
plt.show()
```



▼ intérprétations:

- Les points sont assez **dispersés**, ce qui indique une **faible corrélation** entre les deux variables. Cela signifie que l'augmentation du capital initial n'entraîne pas nécessairement une augmentation proportionnelle du budget global du projet.
    - Il n'y a pas de tendance claire (ni ascendante ni descendante) visible parmi

les points, ce qui renforce l'idée d'une faible corrélation.

- La ligne rouge permet de comparer les différents points de données par rapport à une valeur de référence. Les points au-dessus de la ligne rouge ont un budget global supérieur à cette valeur de référence, tandis que ceux en dessous ont un budget inférieur.
- La ligne horizontale suggère que, indépendamment du capital initial, le budget global du projet reste relativement stable autour de cette valeur moyenne.

▼ Étude de la corrélation entre le niveau d'endettement et l'avancement du projet.

```
# Extract year from relevant columns (assuming "Edition" and "Année de publication" are already in integer format)
Budget_Participatif["Year"] = Budget_Participatif["Edition"]
Dette_Propre["Year"] = Dette_Propre["Année de publication"]

# Filter data based on common years
min_year = min(Budget_Participatif["Year"].min(), Dette_Propre["Year"].min())
max_year = max(Budget_Participatif["Year"].max(), Dette_Propre["Year"].max())

Budget_Participatif = Budget_Participatif[(Budget_Participatif["Year"] >= min_year) & (Budget_Participatif["Year"] <= max_year)]
Dette_Propre = Dette_Propre[(Dette_Propre["Year"] >= min_year) & (Dette_Propre["Year"] <= max_year)]

# Calculate annual debt
annual_debt_df = calculate_annual_debt(Dette_Propre.copy(), "Year", "Capital initial")

# Calculate annual project budget and advancement percentages
annual_project_df = Budget_Participatif.groupby("Year").agg(
    FIN=("Avancement du projet", lambda x: (x == "FIN").sum() / len(x) * 100),
    ETUDE=("Avancement du projet", lambda x: (x == "ETUDE").sum() / len(x) * 100),
    ABANDONNÉ=("Avancement du projet", lambda x: (x == "ABANDONNÉ").sum() / len(x) * 100),
    LIVRAISON=("Avancement du projet", lambda x: (x == "LIVRAISON").sum() / len(x) * 100),
    TRAVAUX=("Avancement du projet", lambda x: (x == "TRAVAUX").sum() / len(x) * 100),
    PROCEDURES=("Avancement du projet", lambda x: (x == "PROCEDURES").sum() / len(x) * 100),
    nondémarré=("Avancement du projet", lambda x: (x == "(non démarré)").sum() / len(x) * 100),
)

# Combine data into new DataFrame
new_df = pd.merge(annual_debt_df, annual_project_df, on="Year")

# Explore and analyze new_df
print(new_df.head())
```

```
   Year  Capital initial        FIN  ETUDE  ABANDONNÉ  LIVRAISON   TRAVAUX  \
0  2014     4.062130e+08  100.000000    0.0   0.000000   0.000000  0.000000
1  2015     4.644601e+08   90.295359    0.0   1.969058   7.172096  0.423941
2  2016     5.193125e+09   67.301174    0.0   2.054326  28.611370  0.652277
3  2017     5.785625e+09   56.910569    0.0   1.300813  36.910569  1.951220
4  2018     5.901314e+09   57.083333    0.0   0.416667  35.208333  1.458333

   PROCEDURES  nondémarré
0    0.000000    0.000000
1    0.140647    0.000000
2    0.372787    0.372787
3    1.138211    0.325203
4    2.500000    1.250000
```

- **Corrélation:**

```
# Correlation analysis
correlation_fin = new_df["Capital initial"].corr(new_df["FIN"])
correlation_etude = new_df["Capital initial"].corr(new_df["ETUDE"])
correlation_abandonné = new_df["Capital initial"].corr(new_df["ABANDONNÉ"])
correlation_livraison = new_df["Capital initial"].corr(new_df["LIVRAISON"])
correlation_travaux = new_df["Capital initial"].corr(new_df["TRAVAUX"])
correlation_procedures = new_df["Capital initial"].corr(new_df["PROCEDURES"])
correlation_nondémarré = new_df["Capital initial"].corr(new_df["nondémarré"])

print(f"Correlation between Capital initial and FIN: {correlation_fin}")
print(f"Correlation between Capital initial and ETUDE: {correlation_etude}")
print(f"Correlation between Capital initial and ABANDONNÉ: {correlation_abandonné}")
print(f"Correlation between Capital initial and LIVRAISON: {correlation_livraison}")
print(f"Correlation between Capital initial and TRAVAUX: {correlation_travaux}")
print(f"Correlation between Capital initial and PROCEDURES: {correlation_procedures}")
print(f"Correlation between Capital initial and nondémarré: {correlation_nondémarré}")
```

```
Correlation between Capital initial and FIN: -0.9840182920327816
Correlation between Capital initial and ETUDE: nan
Correlation between Capital initial and ABANDONNÉ: 0.1482710961219081
Correlation between Capital initial and LIVRAISON: 0.973479741141707
Correlation between Capital initial and TRAVAUX: 0.9339355833611364
Correlation between Capital initial and PROCEDURES: 0.8384009910452345
Correlation between Capital initial and nondémarré: 0.7718656908745235
```

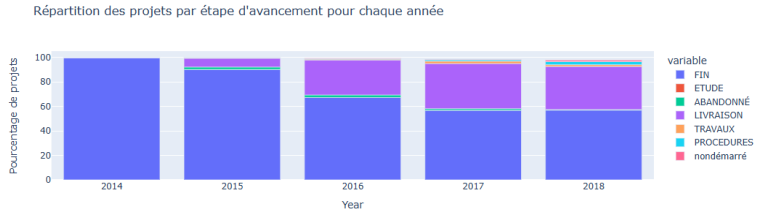
- **Corrélation entre Capital initial et FIN (-0.984)** : Une corrélation négative très forte. Cela signifie que plus le capital initial est élevé, moins le projet a de chances d'être terminé (FIN). Un endettement initial important pourrait donc freiner la finalisation des projets.

- **Corrélation entre Capital initial et ETUDE (nan)** : La valeur "nan" indique un manque de données pour calculer une corrélation entre ces variables. Il est possible qu'il y ait peu ou pas de projets dans la catégorie "ETUDE".
- **Corrélation entre Capital initial et ABANDONNÉ (0.148)** : Une corrélation faiblement positive. Cela suggère une légère tendance : les projets avec un capital initial plus élevé ont un peu plus de chances d'être abandonnés. Cette relation reste cependant très faible.
- **Corrélations entre Capital initial et LIVRAISON, TRAVAUX, PROCÉDURES, non démarré** : Des corrélations positives fortes à très fortes. Cela signifie que plus le capital initial est élevé, plus le projet a de chances d'atteindre ces étapes (LIVRAISON, TRAVAUX, PROCÉDURES) ou de ne pas démarrer du tout. Ces résultats semblent contradictoires avec la corrélation négative de "FIN". Cela pourrait indiquer que les projets avec un capital initial élevé sont plus susceptibles de démarrer et d'avancer dans les premières étapes, mais qu'ils rencontrent ensuite plus de difficultés à être finalisés.

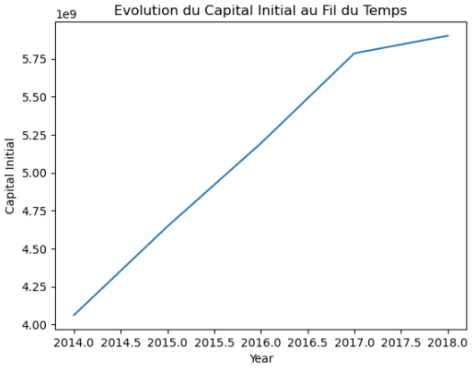
> → L'endettement initial semble avoir un impact complexe sur le cycle de vie des projets. Il favorise le démarrage et l'avancement initial des projets, mais il pourrait freiner leur finalisation. En résumé, ces résultats suggèrent que l'endettement initial joue un rôle ambivalent dans le cycle de vie des projets. Il peut être à la fois un moteur et un frein, selon l'étape du projet considérée.

- **visualisation:**

```python
import plotly.express as px #for interactive visualizations

# Supposons que votre DataFrame 'new_df' contient les colonnes 'Year', 'FIN', 'ETUDE', 'ABANDONNÉ', etc.

fig = px.bar(new_df, x='Year', y=['FIN', 'ETUDE', 'ABANDONNÉ', 'LIVRAISON', 'TRAVAUX', 'PROCEDURES', 'nondémarré'],
             title="Répartition des projets par étape d'avancement pour chaque année",  # Notez les guillemets doubles
             labels={'value':'Pourcentage de projets'})
fig.show()
```




```python
# Plot the line chart
plt.plot(new_df["Year"], new_df["Capital initial"])

# Add labels and title
plt.xlabel("Year")
plt.ylabel("Capital Initial")
plt.title("Evolution du Capital Initial au Fil du Temps")

# Show the plot
plt.show()
```



Corrélation entre l'endettement et l'avancement des projets daprés le visuel:

Année 2014: Le niveau d'endettement le plus bas a coïncidé avec un taux de finalisation de projets (FIN) à 100%. Cela suggère que, dans un contexte de faible endettement, les projets ont une meilleure probabilité

d'être achevés.

Années 2015 à 2017: L'augmentation rapide de l'endettement a été associée à une baisse du pourcentage de projets finalisés (FIN). Parallèlement, nous avons observé une hausse des projets en phase de "Livraison", ce qui pourrait indiquer que l'endettement a ralenti l'avancement des projets.

Année 2018: La stagnation de l'endettement a permis à un petit nombre de projets supplémentaires d'atteindre la phase "FIN", ce qui suggère que la réduction de l'endettement peut favoriser la finalisation des projets.

▼ **Étude de la corrélation entre Projet en Quartier populaire et l'avancement du projet(deux variables catégorielles).**

Pour étudier la corrélation entre "Projet en Quartier populaire" et "l'avancement du projet", nous utilisons une table de contingence et le coefficient V de Cramer. Voici les étapes de l'analyse :

1. Préparation des données

Nous filtrons le DataFrame pour ne conserver que les colonnes pertinentes :

```
quartier_projet_df = Budget_Participatif[["Projet en Quartier populaire", "Avancement du p
```

2. Création de la table de contingence

Une table de contingence est créée pour visualiser la distribution conjointe des deux variables :

```
contingency_table = pd.crosstab(quartier_projet_df["Projet en Quartier populaire"], quarti
```

3. Calcul du V de Cramer

Le V de Cramer est une mesure d'association entre deux variables catégorielles, basée sur le test du chi-carré :

```
chi2, p, dof, expected = chi2_contingency(contingency_table)
n = contingency_table.sum().sum()
phi = np.sqrt(chi2 / (n * (min(contingency_table.shape) - 1)))
cramer_v = np.min([np.sqrt(phi), 1])
```

Interprétation

Le coefficient V de Cramer varie de 0 (aucune association) à 1 (association parfaite). Il permet de quantifier la force de la relation entre les deux variables catégorielles, indépendamment de la taille de l'échantillon.

```
Cramer's V coefficient between 'Projet en Quartier populaire' and 'Avancement du projet': 0.3382203598507274
```

Il existe une relation notable, mais **pas forte**, entre la localisation d'un projet dans un quartier populaire et son état d'avancement.

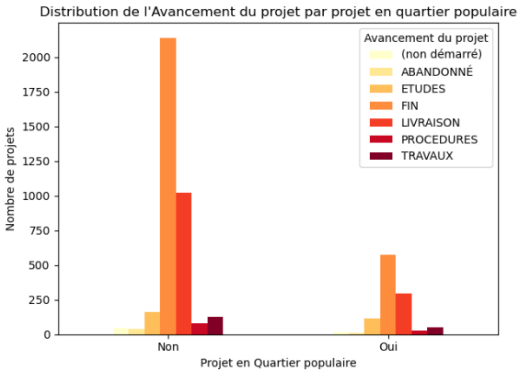
Les projets situés dans des quartiers populaires peuvent connaître des progressions légèrement différentes par rapport à ceux d'autres quartiers, mais cette différence n'est pas prononcée.

D'autres facteurs, tels que la nature du projet, le budget alloué, ou les caractéristiques spécifiques du quartier, pourraient avoir une influence plus significative sur l'avancement des projets.

Cette analyse suggère que bien que l'emplacement du projet dans un quartier populaire ait une certaine influence sur son avancement, **ce n'est probablement pas le facteur déterminant principal.**

- **Visualization:**

```
# Visualization using bar chart (consider using seaborn for more customization)
plt.figure(figsize=(8, 6))
contingency_table.plot(kind="bar", stacked=False, colormap=cm.YlOrRd)
plt.xlabel("Projet en Quartier populaire")
plt.ylabel("Nombre de projets")
plt.title("Distribution de l'Avancement du projet par projet en quartier populaire")
plt.xticks(rotation=0)  # Rotate x-axis labels for better readability
plt.tight_layout()
plt.show()  # Display the bar chart
```



Observations clés:

Projets Abandonnés et non démarrés: Les projets dans les quartiers non populaires un taux légèrement plus élevé d'abandon et de projets non démarrés.

Études et procédures: Les étapes d'études et de procédures ont un taux plus élevé dans les quartiers non populaires.

Réalisation: Les projets dans les quartiers populaires ont un taux inférieur de projets achevés (FIN) et en cours de livraison (LIVRAISON).

▼ Analyse de la Variance (ANOVA) : Impact de la Thématique sur le Budget Global des Projets Lauréats

```
import pandas as pd
# Read the Excel file
Budget_Participatifv = pd.read_excel("budget-participatif_operations-projets-gagnants-realisations (2).xlsx")

Budget_Participatifv.head(100)
```

→ Un nouveau fichier Excel a été utilisé pour l'ANOVA afin d'éliminer les espaces, supprimer les caractéres spéciaux, uniformiser les données et faciliter la manipulation, garantissant ainsi une analyse plus fiable et précise.

```
import statsmodels.api as sm
from statsmodels.formula.api import ols
# Rename columns to avoid spaces and special characters
Budget_Participatifv = Budget_Participatifv.rename(columns={
    "Budget global du projet lauréat": "Budget_global_du_projet_laureat",
    "Thématique": "Thematique"
})
#Create a formula string for the ANOVA model
formula = "Budget_global_du_projet_laureat ~ Thematique"

# Build the linear regression model with 'Thématique' as the categorical variable
model = ols(formula, data=Budget_Participatifv).fit()

# Perform ANOVA test using anova_lm from statsmodels
anova_table = sm.stats.anova_lm(model, typ=2)
# Print the ANOVA table results
print("ANOVA Results for Thématique and Budget_global_du_projet_laureat:")
print(anova_table)

# Interpretation:
# - Look at the p-value in the 'Pr(>F)' column.
#   - If p-value < significance level (e.g., 0.05), reject the null hypothesis
#     (no significant difference in budget allocation across themes).
#   - Otherwise, fail to reject the null hypothesis.
```

Output:

```
ANOVA Results for Thématique and Budget_global_du_projet_laureat:
                    sum_sq      df          F         PR(>F)
Thematique  1.903371e+15    10.0  88.771301  2.365959e-168
Residual    1.005596e+16  4690.0        NaN            NaN
```

Interprétation :

- La p-value est extrêmement petite ce qui est bien inférieur au seuil de signification commun de 0.05. Cela signifie qu'on va **rejeter l'hypothèse null**(il n'a pas de relation entre les thematiques et l'allocation du budget)e. Il y a une différence significative dans le "Budget global du projet lauréat" entre les différentes catégories de "Thématique"

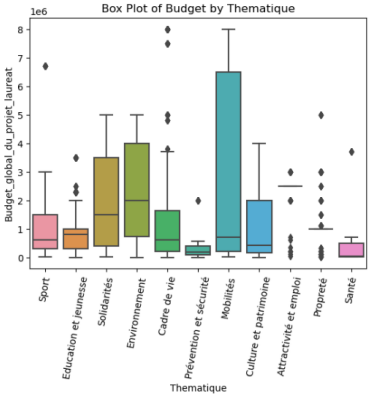
$$2.365959 \times 10^{-168}$$

- **La statistique F de 88.771301** est assez **élevée**, indiquant encore que la variation entre les moyennes des groupes est beaucoup plus grande que la variation au sein des groupes. En résu Les résultats de l'ANOVA montrent qu'il existe au moins une différence significative dans le budget global des projets lauréats entre les différentes catégories thématiques. Cela signifie qu'au moins une catégorie thématique reçoit un financement moyen différent des autres

> Récapitulatif des conclusions : Relation significative/ Hétérogénéité des budgets / Rejet de l'hypothèse nulle: **L'hypothèse selon laquelle les budgets sont identiques pour toutes les thématiques est rejetée. !**

Boîte à moustaches : Distribution du Budget Global des Projets Lauréats par Thématique

```python
import seaborn as sns
import matplotlib.pyplot as plt

sns.boxplot(x='Thematique', y='Budget_global_du_projet_laureat', data=Budget_Participatifv)
plt.title('Box Plot of Budget by Thematique')
plt.xticks(rotation=80)  # Rotate x labels if they are too long
plt.show()
```



- interprétations:

1. **Comparaison des Médianes** : La thématique "**Environnement**" a une médiane nettement plus élevée que les autres, ce qui suggère que cette catégorie reçoit généralement un budget plus important.
2. **Variabilité (IQR)** : La hauteur des boîtes montre où se situent la plupart des points de données. Une boîte plus haute indique **une plus grande variabilité dans le budget** pour cette thématique. Par exemple, "**Mobilités**" a une boîte plus haute, indiquant une **plus grande variabilité.**
3. **Valeurs Aberrantes** : Les points en dehors des moustaches sont des valeurs aberrantes(outliers), indiquant des budgets exceptionnellement élevés ou bas pour certaines thématiques.

Les résultats de l'ANOVA avec une valeur p extrêmement faible et un **F élevé ,**corroborant les observations visuelles des boxplots, suggèrent fortement qu'il existe **des disparités significatives dans les budgets alloués aux différentes thématiques**.

Pour aller plus loin, l'application d'un **test post-hoc comme Tukey's HSD** permettra de comparer toutes les paires de thématiques et de déterminer lesquelles présentent des différences statistiquement

significatives, tout en contrôlant le taux d'erreur de type I (faux positif, se produit lorsqu'on rejette par erreur une hypothèse nulle qui est pourtant vraie).

Test post-hoc : Tukey's HSD

```
from statsmodels.stats.multicomp import pairwise_tukeyhsd
# Replace infinity values with NaN before processing
Budget_Participatifv["Budget_global_du_projet_laureat"] = pd.to_numeric(Budget_Participatifv["Budget_global_du_projet_laureat"], errors='coerce')
# <Convert to numeric, replacing non-numerics with NaN

# Perform Tukey's HSD test
tukey_results = pairwise_tukeyhsd(Budget_Participatifv["Budget_global_du_projet_laureat"], Budget_Participatifv["Thematique"])

# Print the Tukey's HSD results
print(tukey_results)
```

Output:

```
Multiple Comparison of Means - Tukey HSD, FWER=0.05
=================================================================================================
        group1                group2            meandiff  p-adj      lower         upper     reject
-------------------------------------------------------------------------------------------------
Attractivité et emploi           Cadre de vie  -915768.5781    0.0 -1358918.8611  -472618.2952   True
Attractivité et emploi  Culture et patrimoine -1178283.7018    0.0 -1669163.9975   -687403.406   True
Attractivité et emploi  Education et jeunesse -1562157.7339    0.0 -1986697.0003 -1137618.4675   True
Attractivité et emploi          Environnement   -76200.2235    1.0  -509365.9065   356965.4594  False
Attractivité et emploi             Mobilités   564721.295 0.0639   -14806.7347  1144249.3247  False
Attractivité et emploi              Propreté -1092164.7462    0.0 -1581691.9407  -602637.5516   True
Attractivité et emploi Prévention et sécurité -2008819.6315    0.0 -2665613.9321 -1352025.3309   True
Attractivité et emploi                Santé -1600735.2941    0.0 -2565206.6578  -636263.9305   True
Attractivité et emploi           Solidarités  -425995.2185 0.0879   -879340.762    27350.325  False
Attractivité et emploi                Sport -1279628.7649    0.0 -1769423.1558  -789834.374   True
         Cadre de vie  Culture et patrimoine  -262515.1236 0.2788  -594726.6264   69696.3791  False
         Cadre de vie  Education et jeunesse  -646389.1557    0.0  -869177.4106  -423600.9009   True
         Cadre de vie          Environnement   839568.3546    0.0   600751.4233  1078385.2859   True
         Cadre de vie             Mobilités  1480489.8732    0.0  1027440.4774  1933539.2689   True
         Cadre de vie              Propreté  -176396.168 0.8254  -506605.0292   153812.6932  False
         Cadre de vie Prévention et sécurité -1093051.0533    0.0 -1641484.9792  -544617.1274   True
         Cadre de vie                Santé  -684966.716 0.3246 -1579173.6434   209240.2114  False
         Cadre de vie           Solidarités   489773.3596    0.0   216052.3345   763494.3848   True
         Cadre de vie                Sport  -363860.1868 0.0174  -694465.0311   -33255.3424   True
 Culture et patrimoine  Education et jeunesse  -383874.0321 0.0028  -690820.1153   -76927.9489   True
 Culture et patrimoine          Environnement  1102083.4782    0.0   783312.6858  1420854.2707   True
 Culture et patrimoine             Mobilités  1743004.9968    0.0  1243169.9775  2242840.0161   True
 Culture et patrimoine              Propreté    86118.9556 0.9998  -305823.8763   478061.7875  False
 Culture et patrimoine Prévention et sécurité  -830535.9297 0.0003 -1418210.2124  -242861.647   True
 Culture et patrimoine                Santé  -422451.5923 0.9264 -1341248.2622   496345.0775  False
 Culture et patrimoine           Solidarités   752288.4833    0.0   406594.2671  1097982.6995   True
 Culture et patrimoine                Sport  -101345.0631 0.9991  -493621.5659   290931.4396  False
 Education et jeunesse          Environnement  1485957.5103    0.0  1283757.2578  1688157.7628   True
 Education et jeunesse             Mobilités  2126879.0289    0.0  1692016.7858  2561741.272   True
 Education et jeunesse              Propreté   469992.9877    0.0   165215.5157   774770.4597   True
 Education et jeunesse Prévention et sécurité  -446661.8976 0.2818  -980170.2729    86846.4777  False
 Education et jeunesse                Santé   -38577.5602    1.0  -923708.8751   846553.7546  False
 Education et jeunesse           Solidarités  1136162.5154    0.0   893727.7588   1378597.28   True
 Education et jeunesse                Sport   282528.969 0.0995   -22677.4834   587735.4214  False
         Environnement             Mobilités   640921.5186 0.0002   197633.6994  1084209.3377   True
         Environnement              Propreté -1015964.5226    0.0 -1332647.6886  -699281.3567   True
         Environnement Prévention et sécurité -1932619.4079    0.0 -2473017.5081 -1392221.3077   True
         Environnement                Santé -1524535.0706    0.0 -2413836.1246  -635234.0166   True
         Environnement           Solidarités  -349794.995 0.0086  -607037.1673   -92552.8226   True
         Environnement                Sport -1203428.5414    0.0 -1520524.5816  -886332.5011   True
            Mobilités              Propreté -1656886.0412    0.0 -2155392.2658 -1158379.8166   True
            Mobilités Prévention et sécurité -2573540.9265    0.0 -3237054.5549 -1910027.2981   True
            Mobilités                Santé -2165456.5891    0.0 -3134516.2328 -1196396.9455   True
            Mobilités           Solidarités  -990716.5135    0.0 -1453743.2553  -527689.7718   True
            Mobilités                Sport -1844350.0599    0.0 -2343118.6707 -1345581.4492   True
             Propreté Prévention et sécurité  -916654.8853    0.0 -1503199.4037  -330110.3669   True
             Propreté                Santé  -508570.5479   0.79 -1426645.0166   409503.9207  False
             Propreté           Solidarités   666169.5277    0.0   322399.4001  1009939.6552   True
             Propreté                Sport  -187464.0187 0.9843  -578045.9744   203117.9369  False
Prévention et sécurité                Santé   408884.3373 0.9704  -609869.8344  1425238.5091  False
Prévention et sécurité           Solidarités  1582824.413    0.0  1026120.0213  2139528.8046   True
Prévention et sécurité                Sport   729190.8666 0.0031    142423.329  1315958.4042   True
               Santé           Solidarités  1174740.0756 0.0013   275436.9912  2074043.16   True
               Santé                Sport   321106.5292 0.9893  -597110.4392  1239323.4977  False
         Solidarités                Sport  -853633.5464    0.0 -1197784.0538  -509483.0391   True
-------------------------------------------------------------------------------------------------
```

> Le test de Tukey HSD nous a permis de répondre de manière précise à la question de savoir si les différents thèmes de projets reçoivent des budgets significativement différents. Les résultats montrent clairement qu'il existe de fortes disparités(true instances) entre les budgets alloués aux différents thèmes.

▼ **Interprétation des résultats spécifiques**

- **Hiérarchie des budgets:** Les résultats révèlent une hiérarchie claire dans l'allocation des budgets. Certains thèmes, comme "**Education et jeunesse**" et "**Prévention et sécurité**", reçoivent en moyenne des budgets nettement supérieurs à d'autres, tels que "**Environnement**" ou "**Mobilités**".

- **Groupes homogènes:** Le test identifie également des groupes de thèmes dont les budgets ne sont pas significativement différents. Par exemple, "**Attractivité et emploi**" et "**Environnement**" semblent appartenir à un même groupe en termes de niveau de financement.
- **Influences politiques et sociétales:** Ces disparités dans l'allocation des budgets reflètent probablement des choix politiques et des priorités sociétales. Par exemple, l'importance accordée à l'éducation et à la sécurité peut expliquer les budgets plus élevés alloués à ces thèmes.

▼ Construisez un classificateur d'arbre de décision pour prédire l'"Avancement du projet" en fonction des caractéristiques données.

```
#import the necessary libraries for decision tree classifier
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report, accuracy_score
```

```
df = Budget_Participatifv
# Encode categorical variables
df_encoded = pd.get_dummies(df, columns=['Thematique', "Type de financement de l'opération", 'Arrondissement du projet lauréat', "Quartier de l'opératior

# Define features and target variable
X = df_encoded[['Budget_global_du_projet_laureat'] + [col for col in df_encoded.columns if col.startswith(('Thématique_', "Type de financement de l'opéra
y = df['Avancement du projet']
```

```
#split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

```
clf = DecisionTreeClassifier(random_state=42)
clf.fit(X_train, y_train)
```

```
DecisionTreeClassifier
DecisionTreeClassifier(random_state=42)
```

```
#make predictions
y_pred = clf.predict(X_test)

# Evaluate the model
print("Accuracy:", accuracy_score(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))
```

```
Accuracy: 0.8490432317505315
Classification Report:
                precision    recall  f1-score   support

(non démarré)       0.38      0.46      0.42        24
    ABANDONNÉ       0.17      0.20      0.18        15
      ETUDES        0.65      0.58      0.61        83
         FIN        0.92      0.90      0.91       822
   LIVRAISON        0.85      0.90      0.87       383
  PROCEDURES        0.66      0.70      0.68        27
     TRAVAUX        0.63      0.65      0.64        57

    accuracy                           0.85      1411
   macro avg        0.61      0.63      0.62      1411
weighted avg        0.85      0.85      0.85      1411
```

Output et interpretations:

Accuracy: 0.8518781006378455

Cela signifie que le modèle a correctement prédit l'état d'avancement du projet dans environ 85% des cas. C'est une bonne précision globale.

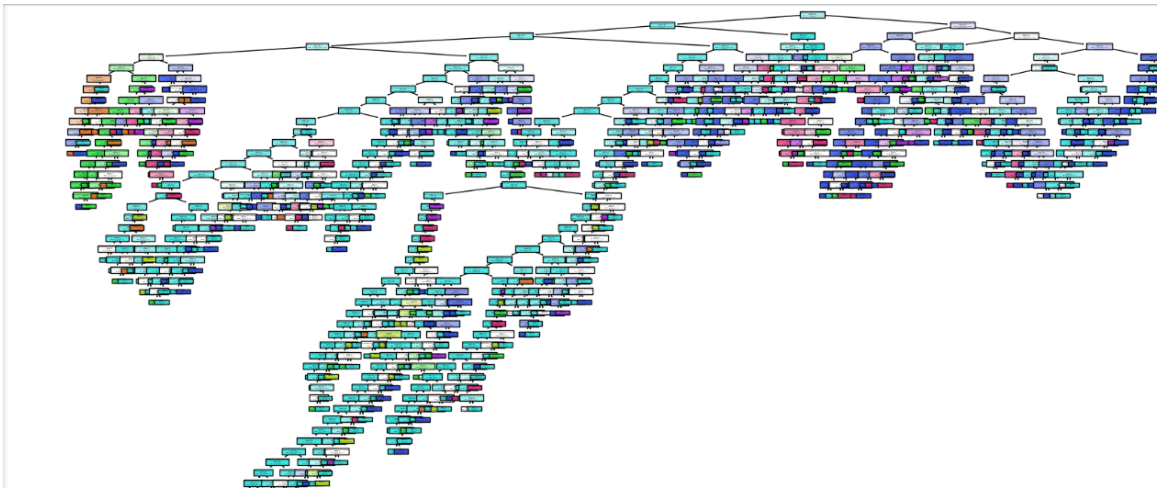
Détails par classe:

- **(non démarré)**: La précision (0.38) et le rappel (0.46) sont faibles, ce qui signifie que le modèle a des difficultés à prédire correctement cette classe.
- **ABANDONNÉ**: La précision (0.17) et le rappel (0.20) sont très faibles, indiquant que le modèle a du mal à identifier correctement les projets abandonnés.
- **ETUDES**: La précision (0.65) et le rappel (0.58) sont modérés, ce qui montre une performance moyenne pour cette classe.
- **FIN**: La précision (0.92) et le rappel (0.90) sont très élevés, ce qui signifie que le modèle est très bon pour prédire les projets terminés.
- **LIVRAISON**: La précision (0.85) et le rappel (0.90) sont également élevés, indiquant une bonne performance pour cette classe.
- **PROCEDURES**: La précision (0.66) et le rappel (0.70) sont modérés.
- **TRAVAUX**: La précision (0.63) et le rappel (0.65)

⇒ modèle de classification des arbres de décision fonctionne bien pour les classes majoritaires comme "FIN" et "LIVRAISON", mais a des difficultés avec les classes minoritaires comme "ABANDONNÉ" et "(non démarré)" sont modérés. .

Visualiser L'arbre de decision:

```python
#visualize decision tree
from sklearn.tree import plot_tree

plt.figure(figsize=(20,10))
plot_tree(clf, feature_names=X.columns, class_names=clf.classes_, filled=True)
plt.show()
```



> **la visualisation de cet arbre était difficile à interpréter car il possédait de nombreuses branches et feuilles. Cela rendait l'analyse des règles de décision complexes et peu claires.**

→ **Utilisation de l'importance des caractéristiques (Feature Importance):** Cette méthode permet d'identifier les caractéristiques les plus influentes dans la prédiction de la variable cible.

```python
# Get feature importances
feature_importances = clf.feature_importances_

# Print feature importances
for feature, importance in zip(X.columns, feature_importances):
    print(f"{feature}: {importance}")
```

```
Budget_global_du_projet_laureat: 0.4863268873467464
Type de financement de l'opération_179: 0.001910676289604481
Type de financement de l'opération_Régie: 0.04786572037816261
Type de financement de l'opération_Subvention: 0.021249269995485626
Arrondissement du projet lauréat_75001: 0.0
Arrondissement du projet lauréat_75002: 0.0
Arrondissement du projet lauréat_75003: 0.003518609793460691
Arrondissement du projet lauréat_75004: 0.04640047661382942
Arrondissement du projet lauréat_75005: 0.0013614417074597858
Arrondissement du projet lauréat_75006: 0.006773668084561228
Arrondissement du projet lauréat_75007: 0.0026159999357130087
Arrondissement du projet lauréat_75008: 0.005134356411596938
Arrondissement du projet lauréat_75009: 0.0017687649018014325
Arrondissement du projet lauréat_75010: 0.010778671250480898
Arrondissement du projet lauréat_75011: 0.015629507322448437
Arrondissement du projet lauréat_75012: 0.010766616950454666
Arrondissement du projet lauréat_75013: 0.018604576882849028
Arrondissement du projet lauréat_75014: 0.014223935536340259
```

OUTPUT

→ **Définition d'un seuil d'importance:** les caractéristiques avec une importance inférieure à 0.4 contribuent moins à la discrimination entre les classes et peuvent être supprimées sans perte d'information importante.

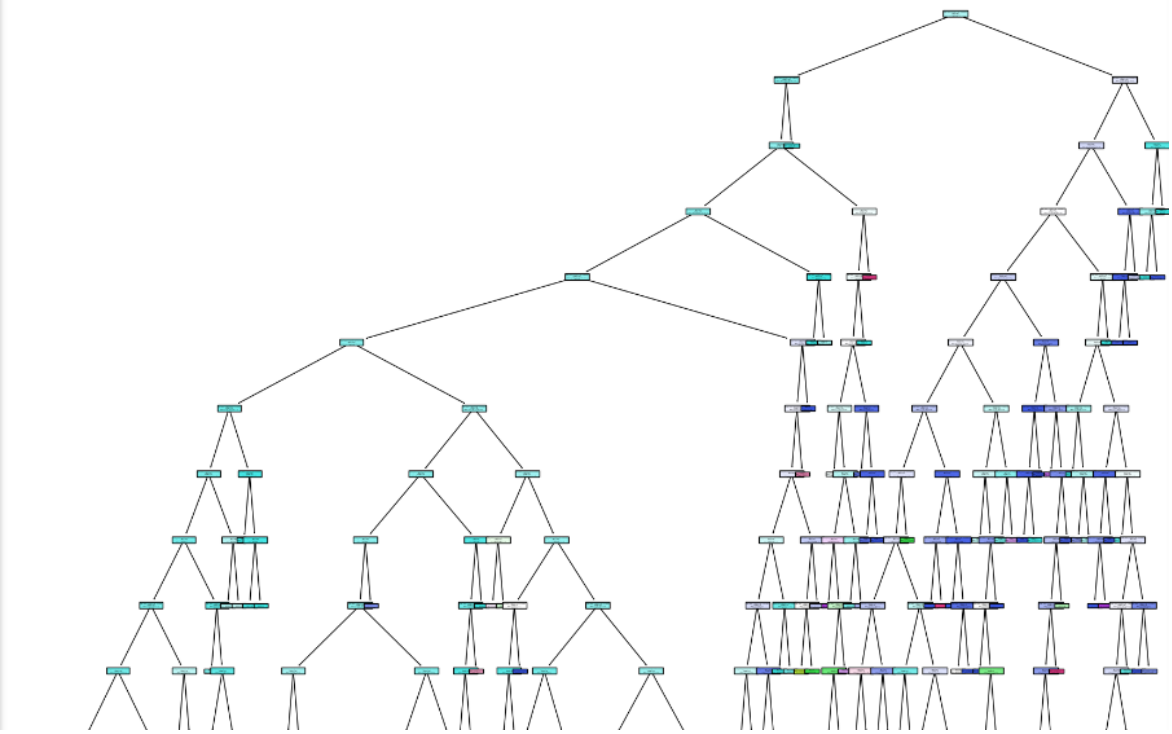
```python
# Set a threshold for feature importance
threshold = 0.4

# Remove features with importance below the threshold
selected_features = X.columns[feature_importances > threshold]

# Create a new DataFrame with only the selected features
X_selected = X[selected_features]

# Train a new decision tree model with the selected features
clf_pruned = DecisionTreeClassifier()
clf_pruned.fit(X_train[selected_features], y_train)
```

```
▾ DecisionTreeClassifier
DecisionTreeClassifier()
```

→ on a utilisé le seuil d'importance pour sélectionner uniquement les caractéristiques dont l'importance est supérieure à 0.4. Cette sélection permet d'obtenir un ensemble de caractéristiques plus pertinent pour la construction d'un nouvel arbre de décision plus simple et plus interprétable.

- **Construction d'un arbre de décision épuré:**

```
# Visualize the pruned tree
plt.figure(figsize=(20, 30))  # Adjust figure size as needed
plot_tree(clf_pruned, feature_names=selected_features, class_names=clf_pruned.classes_, filled=True)
plt.show()
```



Ce nouvel arbre est plus simple et plus facile à interpréter visuellement. La visualisation de cet arbre épuré permet de comprendre plus facilement les règles de décision utilisées pour la classification, mais on est encore en besoin d'ameliorer la lisibilité de cet arbre a fin d'observer d'autres données comme le gini index , nom du champ, etc..

- **Limites de l'élagage par importance des caractéristiques:**

Bien que l'élagage par importance des caractéristiques permette d'obtenir un arbre plus simple, il ne garantit pas nécessairement une amélioration des performances de prédiction. Il est possible que certaines caractéristiques moins importantes selon ce critère puissent être utiles pour la discrimination entre les classes.

Élagage par paramètres de l'arbre de décision:

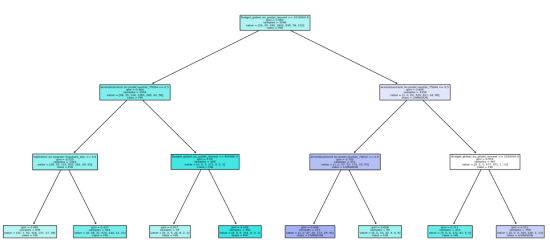
Une autre approche pour améliorer la lisibilité et les performances de l'arbre de décision consiste à utiliser l'élagage par paramètres. En ajustant ces paramètres, on peut contrôler la complexité de l'arbre et éviter le sur-apprentissage (overfitting).

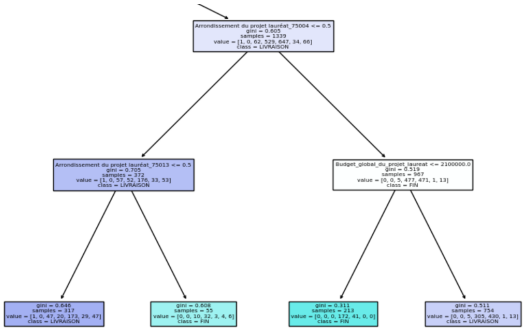
```
# Create a decision tree with pruning parameters
clf = DecisionTreeClassifier(max_depth=3, min_samples_split=20, min_samples_leaf=20, ccp_alpha=0.001)
clf.fit(X_train, y_train)
```

- **Construction d'un arbre de décision avec élagage par paramètres:**

```
# Visualize the pruned tree
plt.figure(figsize=(20, 10))
plot_tree(clf, feature_names=X.columns, class_names=clf.classes_, filled=True)
plt.show()
```

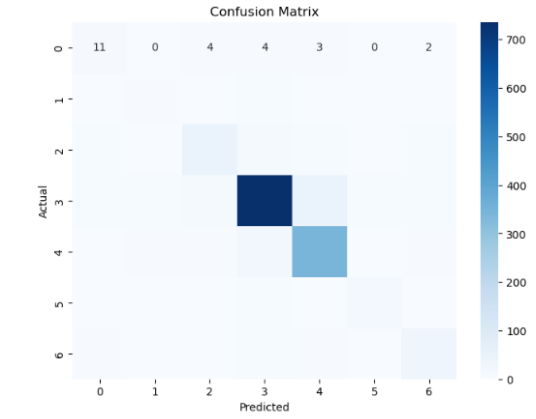
- **Visualisation de l'arbre de décision épuré:**

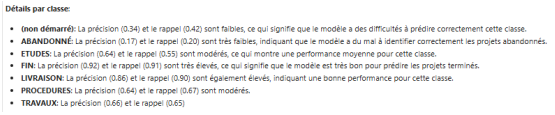




- **Génération de la matrice de confusion:**

```
from sklearn.metrics import confusion_matrix
confusion = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(8, 6))
sns.heatmap(confusion, annot=True, fmt="d", cmap="Blues")
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Confusion Matrix')
plt.show()
```





Détails par classe:

- **(non démarré):** La précision (0.34) et le rappel (0.42) sont faibles, ce qui signifie que le modèle a des difficultés à prédire correctement cette classe.
- **ABANDONNÉ:** La précision (0.17) et le rappel (0.20) sont très faibles, indiquant que le modèle a du mal à identifier correctement les projets abandonnés.
- **ETUDES:** La précision (0.64) et le rappel (0.55) sont modérés, ce qui montre une performance moyenne pour cette classe.
- **FIN:** La précision (0.92) et le rappel (0.91) sont très élevés, ce qui signifie que le modèle est très bon pour prédire les projets terminés.
- **LIVRAISON:** La précision (0.86) et le rappel (0.90) sont également élevés, indiquant une bonne performance pour cette classe.
- **PROCEDURES:** La précision (0.64) et le rappel (0.67) sont modérés.
- **TRAVAUX:** La précision (0.66) et le rappel (0.65)

Throwback: les resultats references qu'on a obtenus lorqu'on a evalué le model

D'après l'analyse de la matrice de confusion, nous pouvons tirer les conclusions suivantes concernant les performances de notre modèle de classification :

- **Dominance de la classe "FIN"(classée comme classe 3) :** Il semble que la majorité des instances correctement classées appartiennent à la classe "FIN". Cette observation est en parfaite cohérence avec les valeurs de précision et de rappel élevées que nous avons obtenues précédemment pour cette classe. Cela signifie que notre modèle est particulièrement doué pour identifier les projets qui sont effectivement terminés.
- **Bonne performance pour la classe "LIVRAISON"(classée comme classe4) :** Bien que dans une moindre mesure que pour la classe "FIN", la classe "LIVRAISON" présente également de bons résultats. La précision et le rappel élevés pour cette classe indiquent que le modèle est capable de distinguer correctement les projets qui sont en phase de livraison. La visualisation de la matrice de confusion confirme cette tendance : la couleur plus claire associée à la classe "LIVRAISON" suggère un nombre d'instances correctement classées significatif, mais inférieur à celui de la classe "FIN".

→ **notre modèle de classification semble particulièrement performant pour prédire les projets qui sont soit terminés ("FIN"), soit en phase de livraison ("LIVRAISON"). Ces deux classes bénéficient d'une précision et d'un rappel élevés, ce qui indique une bonne capacité du modèle à généraliser à de nouveaux exemples.**

- **Voir la structure d'arbre de decision:**

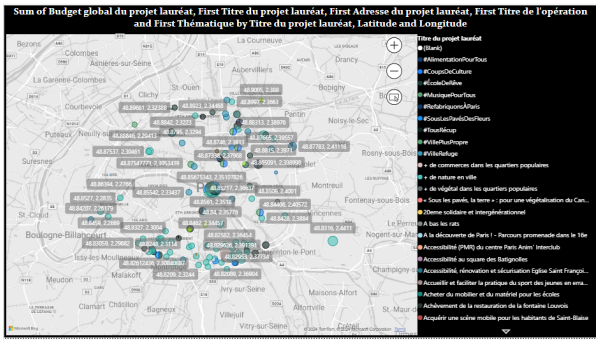
```python
from sklearn.tree import export_text
tree_structure = export_text(clf, feature_names=list(X.columns))
print("Decision Tree Structure:")
print(tree_structure)
```

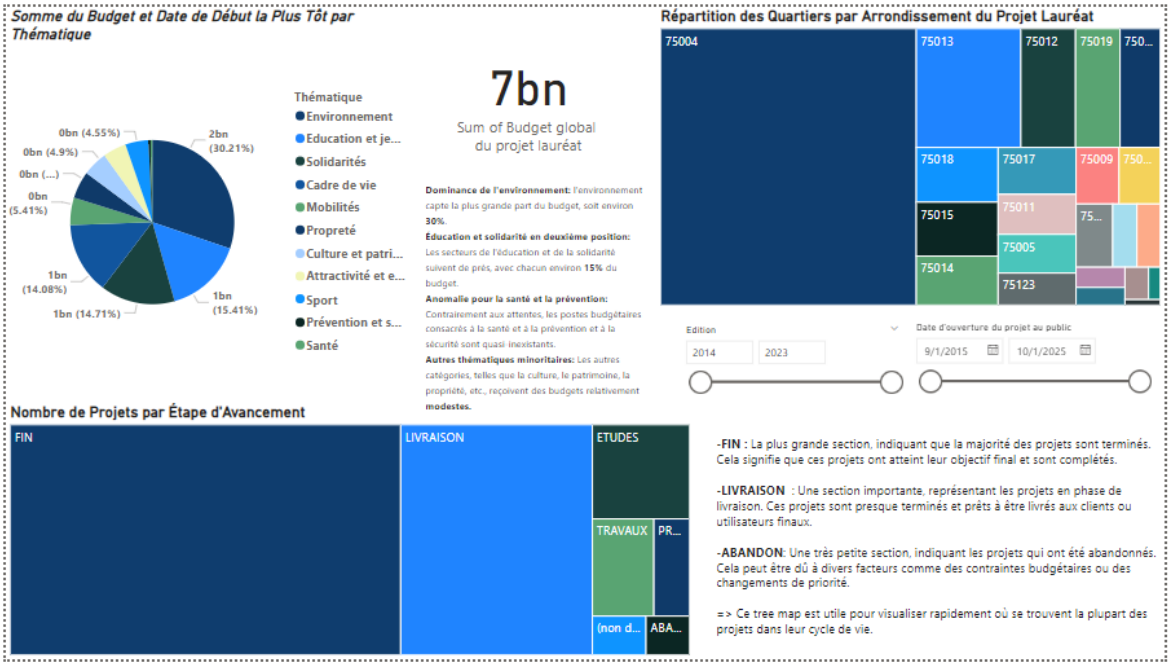
→Cette représentation textuelle de l'arbre de décision nous permet d'avoir une vue détaillée des règles de décision utilisées par notre modèle. En analysant ces règles, nous pouvons mieux comprendre comment le modèle arrive à ses prédictions et identifier les caractéristiques les plus importantes pour la classification.

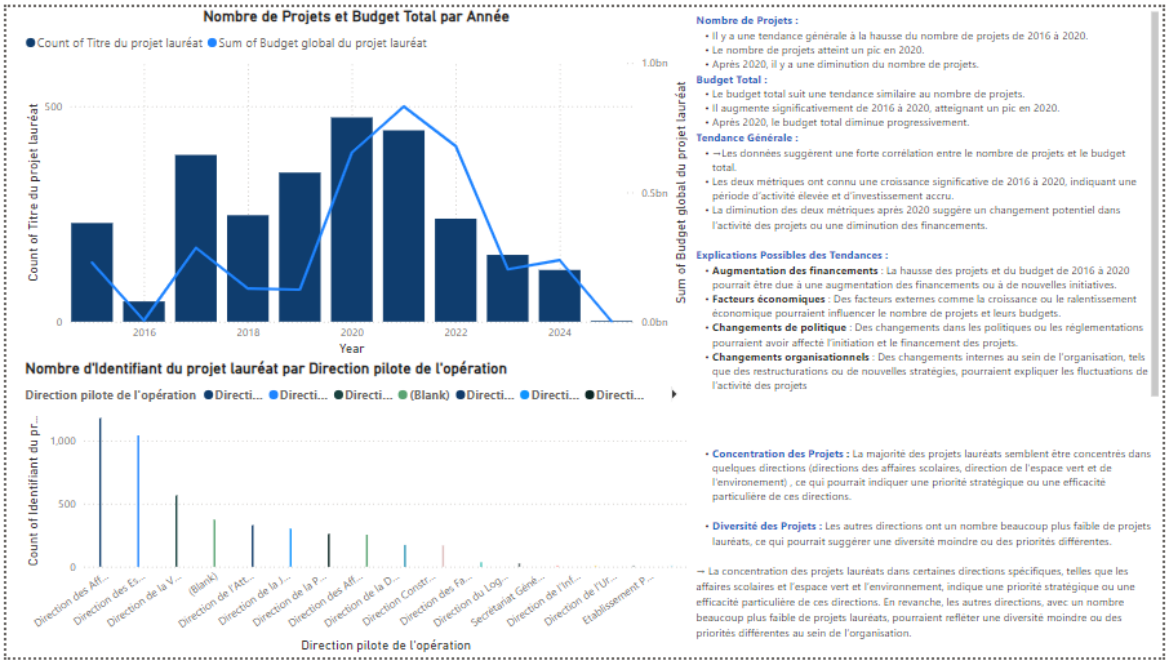
```
Decision Tree Structure:
|--- Budget_global_du_projet_laureat <= 1010000.00
|   |--- Arrondissement du projet lauréat_75004 <= 0.50
|   |   |--- Opération en Quartier Populaire_non <= 0.50
|   |   |   |--- class: FIN
|   |   |--- Opération en Quartier Populaire_non >  0.50
|   |   |   |--- class: FIN
|   |--- Arrondissement du projet lauréat_75004 >  0.50
|   |   |--- Budget_global_du_projet_laureat <= 800000.00
|   |   |   |--- class: FIN
|   |   |--- Budget_global_du_projet_laureat >  800000.00
|   |   |   |--- class: FIN
|--- Budget_global_du_projet_laureat >  1010000.00
|   |--- Arrondissement du projet lauréat_75004 <= 0.50
|   |   |--- Arrondissement du projet lauréat_75013 <= 0.50
|   |   |   |--- class: LIVRAISON
|   |   |--- Arrondissement du projet lauréat_75013 >  0.50
|   |   |   |--- class: FIN
|   |--- Arrondissement du projet lauréat_75004 >  0.50
|   |   |--- Budget_global_du_projet_laureat <= 2100000.00
|   |   |   |--- class: FIN
|   |   |--- Budget_global_du_projet_laureat >  2100000.00
|   |   |   |--- class: LIVRAISON
```

OUTPUT

▼ Power BI Dashboard interactive







Nombre de Projets :

- Il y a une tendance générale à la hausse du nombre de projets de 2016 à 2020.
- Le nombre de projets atteint un pic en 2020.
- Après 2020, il y a une diminution du nombre de projets.

Budget Total :

- Le budget total suit une tendance similaire au nombre de projets.
- Il augmente significativement de 2016 à 2020, atteignant un pic en 2020.
- Après 2020, le budget total diminue progressivement.

Tendance Générale :

- →Les données suggèrent une forte corrélation entre le nombre de projets et le budget total.
- Les deux métriques ont connu une croissance significative de 2016 à 2020, indiquant une période d'activité élevée et d'investissement accru.
- La diminution des deux métriques après 2020 suggère un changement potentiel dans l'activité des projets ou une diminution des financements.

Explications Possibles des Tendances :

- **Augmentation des financements** : La hausse des projets et du budget de 2016 à 2020 pourrait être due à une augmentation des financements ou à de nouvelles initiatives.
- **Facteurs économiques** : Des facteurs externes comme la croissance ou le ralentissement économique pourraient influencer le nombre de projets et leurs budgets.
- **Changements de politique** : Des changements dans les politiques ou les réglementations pourraient avoir affecté l'initiation et le financement des projets.
- **Changements organisationnels** : Des changements internes au sein de l'organisation, tels que des restructurations ou de nouvelles stratégies, pourraient expliquer les fluctuations de l'activité des projets

- **Concentration des Projets :** La majorité des projets lauréats semblent être concentrés dans quelques directions (directions des affaires scolaires, direction de l'espace vert et de l'environement) , ce qui pourrait indiquer une priorité stratégique ou une efficacité particulière de ces directions.

- **Diversité des Projets :** Les autres directions ont un nombre beaucoup plus faible de projets lauréats, ce qui pourrait suggérer une diversité moindre ou des priorités différentes.

→ La concentration des projets lauréats dans certaines directions spécifiques, telles que les affaires scolaires et l'espace vert et l'environnement, indique une priorité stratégique ou une efficacité particulière de ces directions. En revanche, les autres directions, avec un nombre beaucoup plus faible de projets lauréats, pourraient refléter une diversité moindre ou des priorités différentes au sein de l'organisation.