Analyse_Modelisation

September 11, 2025

1 Phase 3 : Analyse Bivariée et Modélisation

Ce notebook est dédié à l'analyse des relations entre les variables (analyse bivariée) et à la construction du modèle de régression logistique pour répondre à la problématique du projet.

Nous utiliserons le jeu de données nettoyé et préparé lors de la phase précédente.

[58]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
from scipy.stats import chi2_contingency

# Configuration pour les graphiques
sns.set(style="whitegrid")
plt.rcParams['figure.figsize'] = (10, 6)
```

[63]:
```
# Chargement des données
try:
    df = pd.read_csv('df_analyse_borgou.csv')
    print("Le jeu de données 'df_analyse_borgou.csv' a été chargé avec succès.")
    print(f"Le DataFrame contient {df.shape[0]} lignes et {df.shape[1]}␣
 ↪colonnes.")
except FileNotFoundError:
    print("Erreur : Le fichier 'df_analyse_borgou.csv' est introuvable.␣
 ↪Assurez-vous qu'il se trouve dans le même répertoire que ce notebook.")

# Afficher les premières lignes pour vérifier
df.head()
```

```
Le jeu de données 'df_analyse_borgou.csv' a été chargé avec succès.
Le DataFrame contient 880 lignes et 36 colonnes.
```

[63]:

	region	usage_preservatif	connaissance_preservatif_vih	\
0	4.0	0.0	1.0	
1	4.0	0.0	1.0	
2	4.0	0.0	1.0	
3	4.0	NaN	1.0	

```
4     4.0                0.0                          1.0

   connaissance_transmission_sain  a_eu_ist_12mois  deja_teste_vih  \
0                             1.0              0.0             1.0
1                             1.0              0.0             0.0
2                             8.0              0.0             0.0
3                             0.0              0.0             0.0
4                             1.0              0.0             0.0

   niveau_instruction  annees_education  alphabetisation   age  …  \
0                 0.0               0.0              3.0  27.0  …
1                 0.0               0.0              0.0  20.0  …
2                 0.0               0.0              0.0  23.0  …
3                 0.0               0.0              0.0  15.0  …
4                 0.0               0.0              0.0  40.0  …

     age_cat  milieu_residence_cat  statut_marital_cat  indice_richesse_cat  \
0  25-34 ans                 Rural              Marié               Pauvre
1  15-24 ans                 Rural              Marié               Pauvre
2  15-24 ans                 Rural              Marié               Pauvre
3  15-24 ans                 Rural       Jamais marié          Très pauvre
4  35-44 ans                 Rural              Marié          Très pauvre

   travaille_actuellement_cat    frequence_radio_cat      frequence_tv_cat  \
0                         Oui            Pas du tout           Pas du tout
1                         Oui  Au moins une fois/sem  Moins d'une fois/sem
2                         Oui  Au moins une fois/sem           Pas du tout
3                         Oui  Au moins une fois/sem  Moins d'une fois/sem
4                         Oui            Pas du tout           Pas du tout

   utilise_internet_cat  age_premier_rapport_cat nb_partenaires_12mois_cat
0          Pas du tout               15-19 ans                         1
1          Pas du tout               15-19 ans                         1
2          Pas du tout               15-19 ans                        2+
3          Pas du tout          Moins de 15 ans                         0
4          Pas du tout               15-19 ans                         1

[5 rows x 36 columns]
```

[60]:
```python
# --- CELLULE DE CORRECTION ---
# Ce bloc recrée les variables catégorielles (_cat) et sauvegarde le DataFrame.
# Il peut être supprimé ou désactivé après sa première exécution.

print("Début de la recodification des variables...")

# Recharger les données brutes
df_corr = pd.read_csv('df_analyse_borgou.csv')
```

```
# Recodage des variables dépendantes
df_corr['usage_preservatif_cat'] = df_corr['usage_preservatif'].map({1.0:␣
 ↪'Oui', 0.0: 'Non'})
df_corr['connaissance_preservatif_vih_cat'] =␣
 ↪df_corr['connaissance_preservatif_vih'].map({1.0: 'Oui', 0.0: 'Non', 9.0:␣
 ↪'Ne sait pas'})
df_corr['connaissance_transmission_sain_cat'] =␣
 ↪df_corr['connaissance_transmission_sain'].map({1.0: 'Oui', 0.0: 'Non', 9.0:␣
 ↪'Ne sait pas'})
df_corr['a_eu_ist_12mois_cat'] = df_corr['a_eu_ist_12mois'].map({1.0: 'Oui', 0.
 ↪0: 'Non', 9.0: 'Ne sait pas'})
df_corr['deja_teste_vih_cat'] = df_corr['deja_teste_vih'].map({1.0: 'Oui', 0.0:␣
 ↪'Non', 9.0: 'Ne sait pas'})

# Recodage de la variable explicative principale
df_corr['niveau_instruction_cat'] = df_corr['niveau_instruction'].map({0:␣
 ↪'Aucun', 1: 'Primaire', 2: 'Secondaire', 3: 'Supérieur', 9: 'Ne sait pas'})

# Recodage des variables de contrôle
df_corr['alphabetisation_cat'] = df_corr['alphabetisation'].map({1: 'Capable de␣
 ↪lire', 2: 'Capable de lire avec difficulté', 3: 'Incapable de lire', 4:␣
 ↪'Analphabète (déclaré)'})
df_corr['age_cat'] = pd.cut(df_corr['age'], bins=[14, 24, 34, 44, 59],␣
 ↪labels=['15-24 ans', '25-34 ans', '35-44 ans', '45-59 ans'])
df_corr['milieu_residence_cat'] = df_corr['milieu_residence'].map({1: 'Urbain',␣
 ↪2: 'Rural'})
df_corr['statut_marital_cat'] = df_corr['statut_marital'].map({0: 'Jamais␣
 ↪marié', 1: 'Marié', 2: 'Vit avec partenaire', 3: 'Veuf', 4: 'Divorcé', 5:␣
 ↪'Séparé'})
df_corr['indice_richesse_cat'] = df_corr['indice_richesse'].map({1: 'Très␣
 ↪pauvre', 2: 'Pauvre', 3: 'Moyen', 4: 'Riche', 5: 'Très riche'})
df_corr['travaille_actuellement_cat'] = df_corr['travaille_actuellement'].
 ↪map({1.0: 'Oui', 0.0: 'Non'})
df_corr['frequence_radio_cat'] = df_corr['frequence_radio'].map({0.0: 'Pas du␣
 ↪tout', 1.0: 'Moins d\'une fois/sem', 2.0: 'Au moins une fois/sem', 3.0:␣
 ↪'Presque tous les jours'})
df_corr['frequence_tv_cat'] = df_corr['frequence_tv'].map({0.0: 'Pas du tout',␣
 ↪1.0: 'Moins d\'une fois/sem', 2.0: 'Au moins une fois/sem', 3.0: 'Presque␣
 ↪tous les jours'})
df_corr['utilise_internet_cat'] = df_corr['utilise_internet'].map({0.0: 'Pas du␣
 ↪tout', 1.0: 'Moins d\'une fois/sem', 2.0: 'Au moins une fois/sem', 3.0:␣
 ↪'Presque tous les jours'})
df_corr['age_premier_rapport_cat'] = pd.cut(df_corr['age_premier_rapport'],␣
 ↪bins=[0, 15, 20, 25, 59], labels=['Moins de 15 ans', '15-19 ans', '20-24␣
 ↪ans', '25 ans et plus'], right=False)
```

```
df_corr['nb_partenaires_12mois_cat'] = df_corr['nb_partenaires_12mois'].
 ↪apply(lambda x: '0' if x == 0 else ('1' if x == 1 else ('2+' if x >= 2 else␣
 ↪'Non spécifié')))

# Sauvegarde du fichier corrigé
df_corr.to_csv('df_analyse_borgou.csv', index=False)

print("Recodification terminée. Le fichier 'df_analyse_borgou.csv' a été mis à␣
 ↪jour avec les variables catégorielles.")
print(f"Le nouveau DataFrame contient {df_corr.shape[0]} lignes et {df_corr.
 ↪shape[1]} colonnes.")
df_corr.head()
```

```
Début de la recodification des variables…

Recodification terminée. Le fichier 'df_analyse_borgou.csv' a été mis à jour
avec les variables catégorielles.
Le nouveau DataFrame contient 880 lignes et 36 colonnes.
```

[60]:

	region	usage_preservatif	connaissance_preservatif_vih	\
0	4.0	0.0	1.0	
1	4.0	0.0	1.0	
2	4.0	0.0	1.0	
3	4.0	NaN	1.0	
4	4.0	0.0	1.0	

	connaissance_transmission_sain	a_eu_ist_12mois	deja_teste_vih	\
0	1.0	0.0	1.0	
1	1.0	0.0	0.0	
2	8.0	0.0	0.0	
3	0.0	0.0	0.0	
4	1.0	0.0	0.0	

	niveau_instruction	annees_education	alphabetisation	age	…	\
0	0.0	0.0	3.0	27.0	…	
1	0.0	0.0	0.0	20.0	…	
2	0.0	0.0	0.0	23.0	…	
3	0.0	0.0	0.0	15.0	…	
4	0.0	0.0	0.0	40.0	…	

	age_cat	milieu_residence_cat	statut_marital_cat	indice_richesse_cat	\
0	25-34 ans	Rural	Marié	Pauvre	
1	15-24 ans	Rural	Marié	Pauvre	
2	15-24 ans	Rural	Marié	Pauvre	
3	15-24 ans	Rural	Jamais marié	Très pauvre	
4	35-44 ans	Rural	Marié	Très pauvre	

	travaille_actuellement_cat	frequence_radio_cat	frequence_tv_cat	\

```
0                         Oui            Pas du tout           Pas du tout
1                         Oui  Au moins une fois/sem  Moins d'une fois/sem
2                         Oui  Au moins une fois/sem           Pas du tout
3                         Oui  Au moins une fois/sem  Moins d'une fois/sem
4                         Oui            Pas du tout           Pas du tout

   utilise_internet_cat  age_premier_rapport_cat nb_partenaires_12mois_cat
0          Pas du tout               15-19 ans                         1
1          Pas du tout               15-19 ans                         1
2          Pas du tout               15-19 ans                        2+
3          Pas du tout          Moins de 15 ans                         0
4          Pas du tout               15-19 ans                         1

[5 rows x 36 columns]
```

[61]:
```
# Vérification des colonnes disponibles
print("Colonnes disponibles dans le DataFrame chargé :")
print(list(df.columns))
```

```
Colonnes disponibles dans le DataFrame chargé :
['region', 'usage_preservatif', 'connaissance_preservatif_vih',
'connaissance_transmission_sain', 'a_eu_ist_12mois', 'deja_teste_vih',
'niveau_instruction', 'annees_education', 'alphabetisation', 'age',
'milieu_residence', 'statut_marital', 'indice_richesse',
'travaille_actuellement', 'frequence_radio', 'frequence_tv', 'utilise_internet',
'age_premier_rapport', 'nb_partenaires_12mois', 'usage_preservatif_cat',
'connaissance_preservatif_vih_cat', 'connaissance_transmission_sain_cat',
'a_eu_ist_12mois_cat', 'deja_teste_vih_cat', 'niveau_instruction_cat',
'alphabetisation_cat', 'age_cat', 'milieu_residence_cat', 'statut_marital_cat',
'indice_richesse_cat', 'travaille_actuellement_cat', 'frequence_radio_cat',
'frequence_tv_cat', 'utilise_internet_cat', 'age_premier_rapport_cat',
'nb_partenaires_12mois_cat']
```

1.1 1. Analyse Bivariée

Cette section explore les relations entre les paires de variables, en se concentrant sur l'influence de la variable explicative sur les variables dépendantes, tout en considérant les variables de contrôle.

1.1.1 1.1. Niveau d'instruction et Utilisation du préservatif

Objectif : Mesurer si le niveau d'instruction a un impact statistiquement significatif sur l'utilisation du préservatif.

Pour cela, nous allons utiliser le **test d'indépendance du Chi-carré (²)**.

Comprendre le Test du Chi-carré

- **Pour tous :** Imaginez que nous voulons savoir si les gens qui aiment le café (groupe A) sont plus susceptibles de préférer les matins (groupe B) que le reste de la population. Le test du Chi-carré nous aide à déterminer si ces deux préférences ("aime le café" et "préfère le matin") sont liées ou si c'est juste une coïncidence. Dans notre cas, nous voulons savoir si le "niveau d'instruction" et "l'utilisation du préservatif" sont liés.
- **Pourquoi ce test ? (L'intuition) :** Nous utilisons le test du Chi-carré car nous travaillons avec des **variables catégorielles** (des étiquettes comme "Primaire", "Secondaire", "Oui", "Non"). Ce test compare les données que nous avons *observées* (le nombre réel de personnes dans chaque croisement, par exemple "Secondaire" et "Oui") avec les données que nous *attendrions* si les deux variables étaient totalement indépendantes (c'est-à-dire si l'instruction n'avait absolument aucun effet sur l'utilisation du préservatif). Si l'écart entre ce que nous observons et ce que nous attendons est grand, alors nous suspectons que les variables ne sont pas indépendantes et qu'il y a une relation entre elles.
- **Pour les experts (Détails techniques) :**
  - **Hypothèse nulle (H) :** Le niveau d'instruction et l'utilisation du préservatif sont indépendants. La répartition de l'utilisation du préservatif est la même à travers tous les niveaux d'instruction.
  - **Hypothèse alternative (H) :** Il existe une dépendance entre le niveau d'instruction et l'utilisation du préservatif.
  - **Condition d'application :** Le test est fiable si les effectifs attendus dans chaque cellule du tableau de contingence sont majoritairement 5. Nous procédons en supposant cette condition remplie, ce qui est généralement le cas avec un échantillon de notre taille (N=880).
  - **Interprétation de la p-value :** La p-value représente la probabilité d'observer un écart au moins aussi grand que celui mesuré si l'hypothèse nulle (H) était vraie. Une p-value faible (typiquement < 0.05) nous conduit à rejeter H et à conclure à une association statistiquement significative.

[64]:
```
# Tableau croisé entre le niveau d'instruction et l'utilisation du préservatif
cross_tab = pd.crosstab(df['niveau_instruction_cat'],␣
 ↪df['usage_preservatif_cat'])

print("Tableau croisé : Niveau d'instruction vs Utilisation du préservatif")
print(cross_tab)

# Test du Chi-carré
chi2, p, dof, expected = chi2_contingency(cross_tab)
print(f"\nTest du Chi-carré :")
print(f"  - Statistique du Chi² = {chi2:.2f}")
print(f"  - p-value = {p:.4f}")

# Interprétation du test
alpha = 0.05
if p < alpha:
    print("\nInterprétation : La p-value est inférieure à 0.05. Nous rejetons␣
 ↪l'hypothèse nulle.")
```

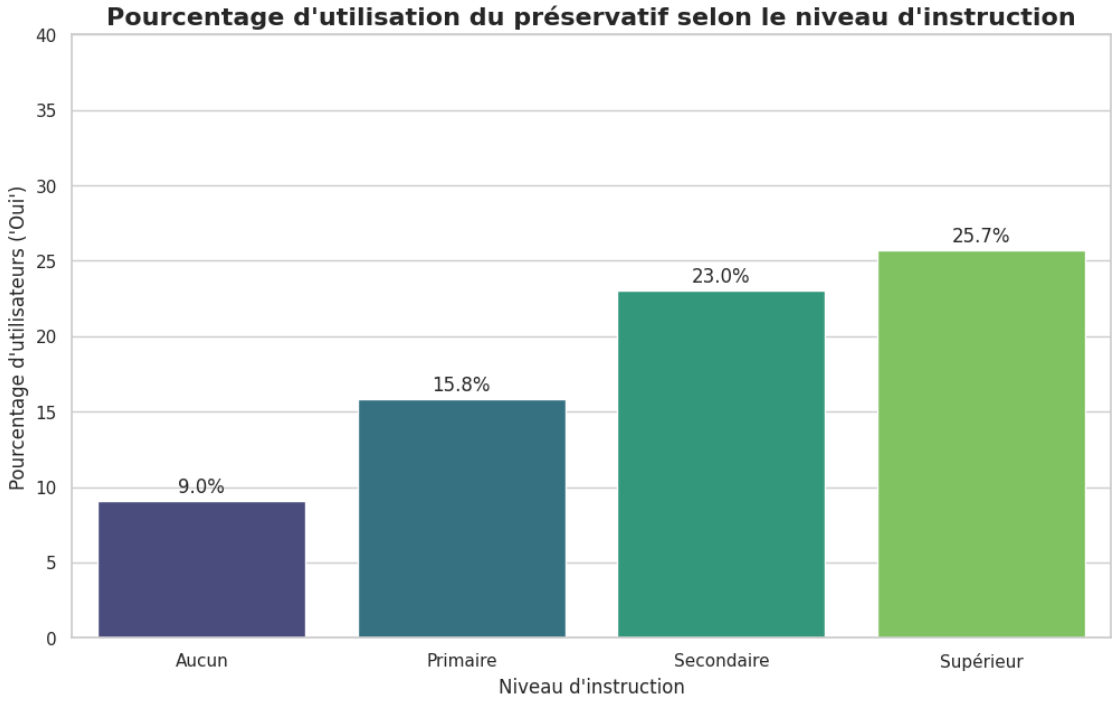
```
    print("Il existe une association statistiquement significative entre le␣
 ↪niveau d'instruction et l'utilisation du préservatif.")
else:
    print("\nInterprétation : La p-value est supérieure ou égale à 0.05. Nous␣
 ↪ne pouvons pas rejeter l'hypothèse nulle.")
    print("Il n'y a pas d'association statistiquement significative entre le␣
 ↪niveau d'instruction et l'utilisation du préservatif.")
```

```
Tableau croisé : Niveau d'instruction vs Utilisation du préservatif
usage_preservatif_cat   Non  Oui
niveau_instruction_cat
Aucun                   342   34
Primaire                 85   16
Secondaire               97   29
Supérieur                26    9

Test du Chi-carré :
  - Statistique du Chi² = 20.69
  - p-value = 0.0001

Interprétation : La p-value est inférieure à 0.05. Nous rejetons l'hypothèse
nulle.
Il existe une association statistiquement significative entre le niveau
d'instruction et l'utilisation du préservatif.
```

[65]:
```
# Visualisation de la relation

# Calculer les pourcentages pour la visualisation
cross_tab_prop = pd.crosstab(df['niveau_instruction_cat'],␣
 ↪df['usage_preservatif_cat'], normalize='index') * 100
cross_tab_prop = cross_tab_prop.reindex(['Aucun', 'Primaire', 'Secondaire',␣
 ↪'Supérieur'])

# Création du graphique
plt.figure(figsize=(12, 7))
ax = sns.barplot(x=cross_tab_prop.index, y=cross_tab_prop['Oui'],␣
 ↪palette='viridis')

# Ajout des pourcentages sur les barres
for p in ax.patches:
    ax.annotate(f'{p.get_height():.1f}%',
                (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center',
                xytext=(0, 9),
                textcoords='offset points',
                fontsize=12)
```

```
# Titres et labels
plt.title("Pourcentage d'utilisation du préservatif selon le niveau␣
 ↪d'instruction", fontsize=16, fontweight='bold')
plt.xlabel("Niveau d'instruction", fontsize=12)
plt.ylabel("Pourcentage d'utilisateurs ('Oui')", fontsize=12)
plt.ylim(0, 40) # Ajuster la limite de l'axe y pour une meilleure lisibilité

# Afficher le graphique
plt.show()
```

```
/tmp/ipykernel_53031/707288087.py:9: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  ax = sns.barplot(x=cross_tab_prop.index, y=cross_tab_prop['Oui'],
palette='viridis')
```



1.1.2 1.2. Niveau d'instruction et Connaissance de la protection du préservatif contre le VIH

Nous analysons maintenant la relation entre le niveau d'instruction et la connaissance que le préservatif est un moyen de prévention efficace contre le VIH.

Hypothèse : Un niveau d'instruction plus élevé est associé à une meilleure connaissance des méthodes de prévention du VIH.

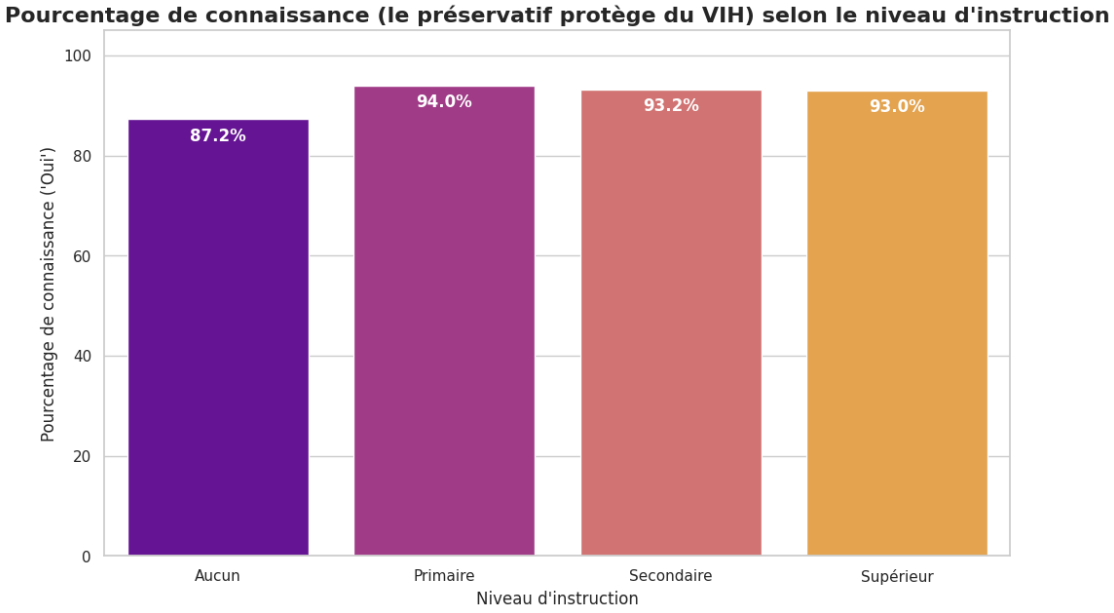
[66]:
```
# Tableau croisé
cross_tab_connaissance = pd.crosstab(df['niveau_instruction_cat'],␣
 ↪df['connaissance_preservatif_vih_cat'])

# Nous allons exclure la catégorie "Ne sait pas" du test statistique pour une␣
 ↪analyse plus claire
cross_tab_connaissance_test = cross_tab_connaissance[['Oui', 'Non']]

print("Tableau croisé : Niveau d'instruction vs Connaissance de la protection␣
 ↪VIH")
print(cross_tab_connaissance)

# Test du Chi-carré
chi2, p, dof, expected = chi2_contingency(cross_tab_connaissance_test)
print(f"\nTest du Chi-carré (sur 'Oui' et 'Non') :")
print(f"  - Statistique du Chi² = {chi2:.2f}")
print(f"  - p-value = {p:.4f}")

# Interprétation du test
alpha = 0.05
if p < alpha:
    print("\nInterprétation : La p-value est inférieure à 0.05. Nous rejetons␣
 ↪l'hypothèse nulle.")
    print("Il existe une association statistiquement significative entre le␣
 ↪niveau d'instruction et la connaissance sur le VIH.")
else:
    print("\nInterprétation : La p-value est supérieure ou égale à 0.05. Nous␣
 ↪ne pouvons pas rejeter l'hypothèse nulle.")
    print("Il n'y a pas d'association statistiquement significative entre le␣
 ↪niveau d'instruction et la connaissance sur le VIH.")
```

```
Tableau croisé : Niveau d'instruction vs Connaissance de la protection VIH
connaissance_preservatif_vih_cat  Non  Oui
niveau_instruction_cat
Aucun                              51  349
Primaire                            8  126
Secondaire                         14  191
Supérieur                           4   53

Test du Chi-carré (sur 'Oui' et 'Non') :
  - Statistique du Chi² = 8.79
  - p-value = 0.0322

Interprétation : La p-value est inférieure à 0.05. Nous rejetons l'hypothèse
nulle.
```

Il existe une association statistiquement significative entre le niveau d'instruction et la connaissance sur le VIH.

[67]:
```
# Visualisation de la relation

# Calculer les pourcentages
cross_tab_connaissance_prop = pd.crosstab(df['niveau_instruction_cat'],␣
 ↪df['connaissance_preservatif_vih_cat'], normalize='index') * 100
cross_tab_connaissance_prop = cross_tab_connaissance_prop.reindex(['Aucun',␣
 ↪'Primaire', 'Secondaire', 'Supérieur'])

# Création du graphique
plt.figure(figsize=(12, 7))
ax = sns.barplot(x=cross_tab_connaissance_prop.index,␣
 ↪y=cross_tab_connaissance_prop['Oui'], palette='plasma')

# Ajout des pourcentages sur les barres
for p in ax.patches:
    ax.annotate(f'{p.get_height():.1f}%',
                (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center',
                xytext=(0, -12), # Placer le texte à l'intérieur de la barre
                textcoords='offset points',
                fontsize=12, color='white', fontweight='bold')

# Titres et labels
plt.title("Pourcentage de connaissance (le préservatif protège du VIH) selon le␣
 ↪niveau d'instruction", fontsize=16, fontweight='bold')
plt.xlabel("Niveau d'instruction", fontsize=12)
plt.ylabel("Pourcentage de connaissance ('Oui')", fontsize=12)
plt.ylim(0, 105) # L'axe y va jusqu'à 100%

# Afficher le graphique
plt.show()
```

```
/tmp/ipykernel_53031/3449441891.py:9: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  ax = sns.barplot(x=cross_tab_connaissance_prop.index,
y=cross_tab_connaissance_prop['Oui'], palette='plasma')
```


Pourcentage de connaissance (le préservatif protège du VIH) selon le niveau d'instruction
Pourcentage de connaissance ('Oui')
100
80
60
40
20
0
87.2%
94.0%
93.2%
93.0%
Aucun
Primaire
Secondaire
Supérieur
Niveau d'instruction