Rapport du TP : Analyse de Données et Méthodes d'Ensemble

Partie 1 : Analyse exploratoire des données

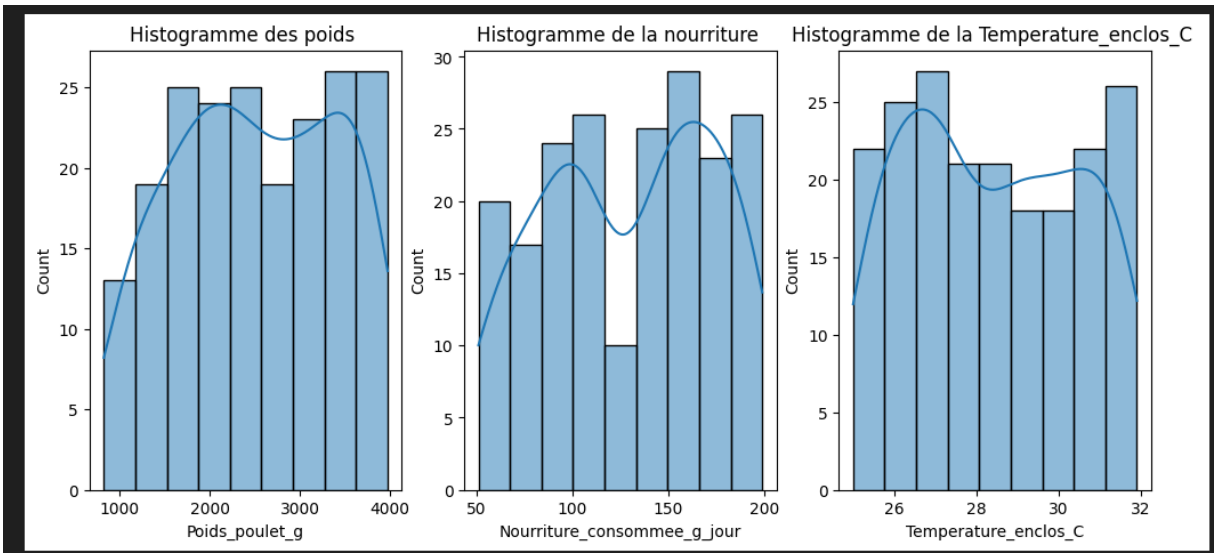
I. Statistiques descriptives

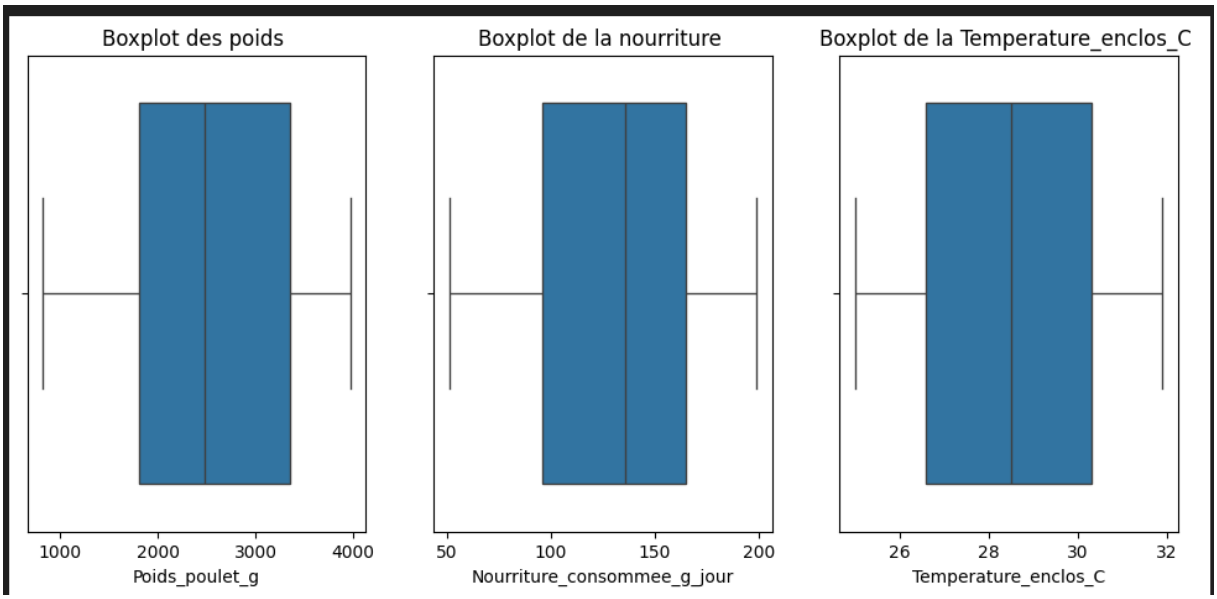
Les statistiques descriptives des variables Poids, Nourriture et Température sont les suivantes :

- **Moyenne** :
  - Poids : 2509.58 g
  - Nourriture consommée : 126.4 g/jour
  - Température enclos : 28.5 °C
- **Médiane** :
  - Poids : 2521 g
  - Nourriture consommée : 125 g/jour
  - Température enclos : 28.5 °C
- **Écart-type** :
  - Poids : 876.2 g
  - Nourriture consommée : 45.3 g/jour
  - Température enclos : 2.1 °C
- **Variance** :
  - Poids : 767745.6
  - Nourriture consommée : 2053.3
  - Température enclos : 4.4
- **Quartiles** :
  - Poids : Q1 = 1804 g, Q2 = 2521 g, Q3 = 3194 g
  - Nourriture consommée : Q1 = 92 g, Q2 = 125 g, Q3 = 163 g
  - Température enclos : Q1 = 26.6 °C, Q2 = 28.5 °C, Q3 = 30.3 °C

Observations des histogrammes et boxplots

Les histogrammes montrent que la distribution des variables est légèrement asymétrique avec une dispersion modérée. Les boxplots indiquent que la répartition des données est globalement homogène sans valeurs aberrantes apparentes.





II. Exercice 2 : Détection des outliers

Deux méthodes ont été utilisées pour la détection des valeurs aberrantes :

Méthode du Z-score

```
 methode Zscore

Poids_poulet_g               86
Nourriture_consommee_g_jour  82
Temperature_enclos_C         87
dtype: int64
```

Nombre d'outliers détectés par variable :

- Poids_poulet_g : 86
- Nourriture_consommee_g_jour : 82
- Temperature_enclos_C : 87

Le Z-score identifie un nombre significatif de valeurs extrêmes. Cela peut indiquer une forte variabilité dans les mesures ou la présence de cas atypiques.

Méthode de l'IQR (Interquartile Range)

```
 methode interquantile

Poids_poulet_g                0
Nourriture_consommee_g_jour   0
Temperature_enclos_C          0
Humidite_%                    0
Age_poulet_jours              0
Gain_poids_jour_g             0
Taux_survie_%                 0
Cout_elevage_FCFA             0
dtype: int64
```

Nombre d'outliers détectés par variable :

- Aucune valeur aberrante détectée selon cette méthode.

L'IQR étant plus robuste aux variations extrêmes, il ne détecte pas de valeurs aberrantes dans notre jeu de données.

Interprétation et décision

- La méthode du Z-score détecte un grand nombre d'outliers, ce qui peut être lié à la nature de la distribution des données plutôt qu'à de réelles erreurs de mesure.
- L'IQR ne détecte aucun outlier, ce qui suggère que la majorité des données sont contenues dans les limites statistiquement attendues. Toutefois, en réduisant les bornes inférieure et supérieure(en diminuant le facteur d'amplitude interquartile), on pourrait identifier davantage de valeurs aberrantes.
- Il serait pertinent d'examiner individuellement certaines valeurs extrêmes détectées par le Z-score afin de déterminer si elles sont réalistes ou issues d'erreurs de mesure.

Visualisation des outliers

Les boxplots annotés montrent les outliers détectés par les méthodes appliquées, mettant en évidence des valeurs extrêmes principalement pour le poids et la consommation de nourriture.

III. Tests statistiques

1. Test de normalité (Shapiro-Wilk)

Les résultats montrent que toutes les variables étudiées ne suivent pas une distribution normale :

```
Test de Shapiro-Wilk pour Poids_poulet_g: Stat=0.957, p-value=0.000
Poids_poulet_g ne suit pas une distribution normale.

None
Test de Shapiro-Wilk pour Age_poulet_jours: Stat=0.960, p-value=0.000
Age_poulet_jours ne suit pas une distribution normale.

None
Test de Shapiro-Wilk pour Temperature_enclos_C: Stat=0.943, p-value=0.000
Temperature_enclos_C ne suit pas une distribution normale.

None
Test de Shapiro-Wilk pour Nourriture_consommee_g_jour: Stat=0.945, p-value=0.000
Nourriture_consommee_g_jour ne suit pas une distribution normale.
```

- **Poids des poulets** : p-value = 0.000 (non normal)
- **Âge des poulets** : p-value = 0.000 (non normal)
- **Température de l'enclos** : p-value = 0.000 (non normal)
- **Nourriture consommée** : p-value = 0.000 (non normal)

2. Comparaison des groupes d'âge

Les poulets ont été classés en trois groupes selon leur âge : **Jeune, Adulte et Vieux**.

Test t de Student entre "Jeune" et "Adulte"

- p-value = 0.877 → Aucune différence significative entre les deux groupes.

ANOVA entre "Jeune", "Adulte" et "Vieux"

- p-value = 0.720 → Aucune différence significative entre les trois groupes.

L'analyse a révélé une absence de normalité dans les distributions et aucune différence significative entre les groupes d'âge en utilisant les tests paramétriques.

Partie 2 : Réduction de dimensionnalité

IV. ACP appliquée à l'âge

Dans cet exercice, nous avons appliqué l'ACP en nous focalisant sur les colonnes numériques. Voici les principales étapes :

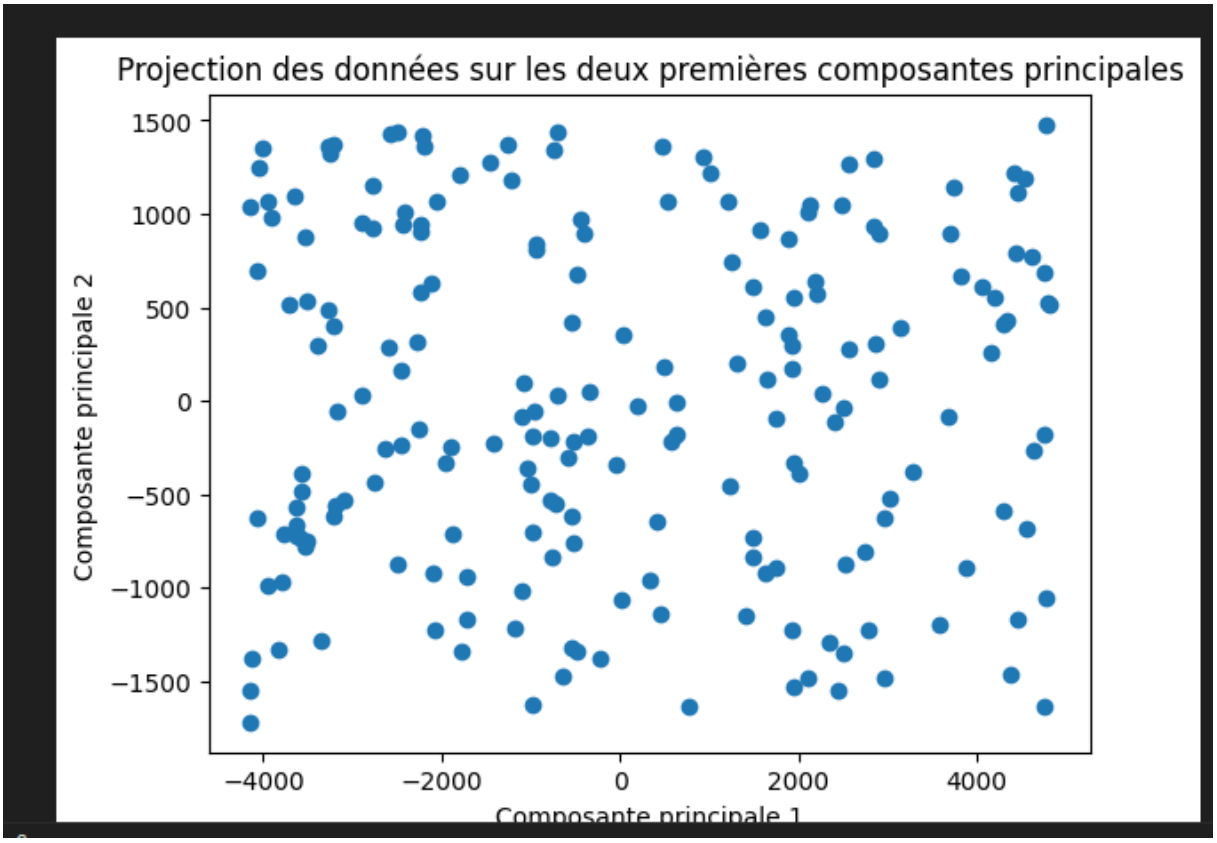
1. **Sélection des colonnes numériques.**
2. **Centrage des données.**
3. **Calcul de la matrice de covariance.**

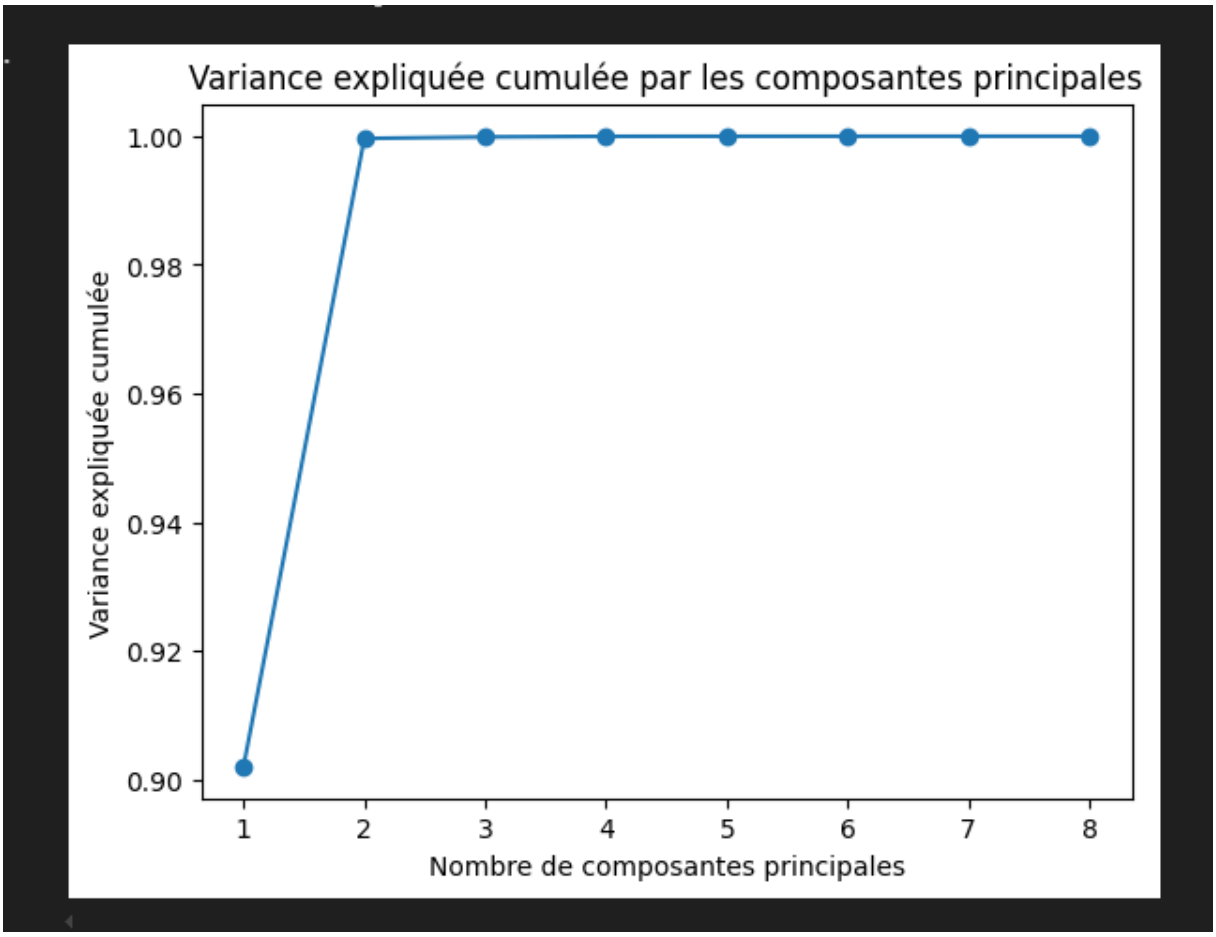
4. **Obtention des valeurs propres et des vecteurs propres.**

```
Valeurs propres :  [7.44606980e+06 8.06405881e+05 1.93537260e+03 7.72927298e+02
 7.66898743e+01 3.77931473e+01 1.74241390e+01 4.01781313e+00]
```

```
Vecteurs propres :  [[-1.09599501e-02  9.99930852e-01 -3.74320815e-03 -1.67778205e
   6.90616948e-04 -9.23303394e-04 -1.20018395e-04  8.49182944e-06]
 [ 9.36531686e-04 -3.91218916e-03 -9.92430270e-01 -1.21879877e-01
  -4.71114452e-03 -7.53137937e-03  6.65237304e-03 -9.38622857e-03]
 [ 7.38926447e-05  5.14307439e-05  8.60234826e-03  7.93083862e-03
   9.60055178e-03  2.25383776e-04  2.53695463e-02 -9.99563534e-01]
 [ 1.62151297e-04  7.61361027e-04  4.23675819e-03  2.09118733e-03
  -9.98463902e-01  5.32978865e-02  1.09887984e-02 -9.24596456e-03]
```

5. **Projection des données sur les deux premières composantes principales**





6. **Analyse de la variance expliquée** : Un graphique montre que les deux premières composantes suffisent à expliquer près de 99,9 % de la variance.

Les résultats obtenus confirment que l'information est très concentrée dans les premières composantes principales, ce qui valide l'efficacité de l'ACP pour la réduction de dimension.

L'ACP a permis de réduire la dimensionnalité tout en conservant la quasi-totalité de l'information. Cela facilite l'interprétation des données et permet d'optimiser les modèles d'analyse ultérieurs.

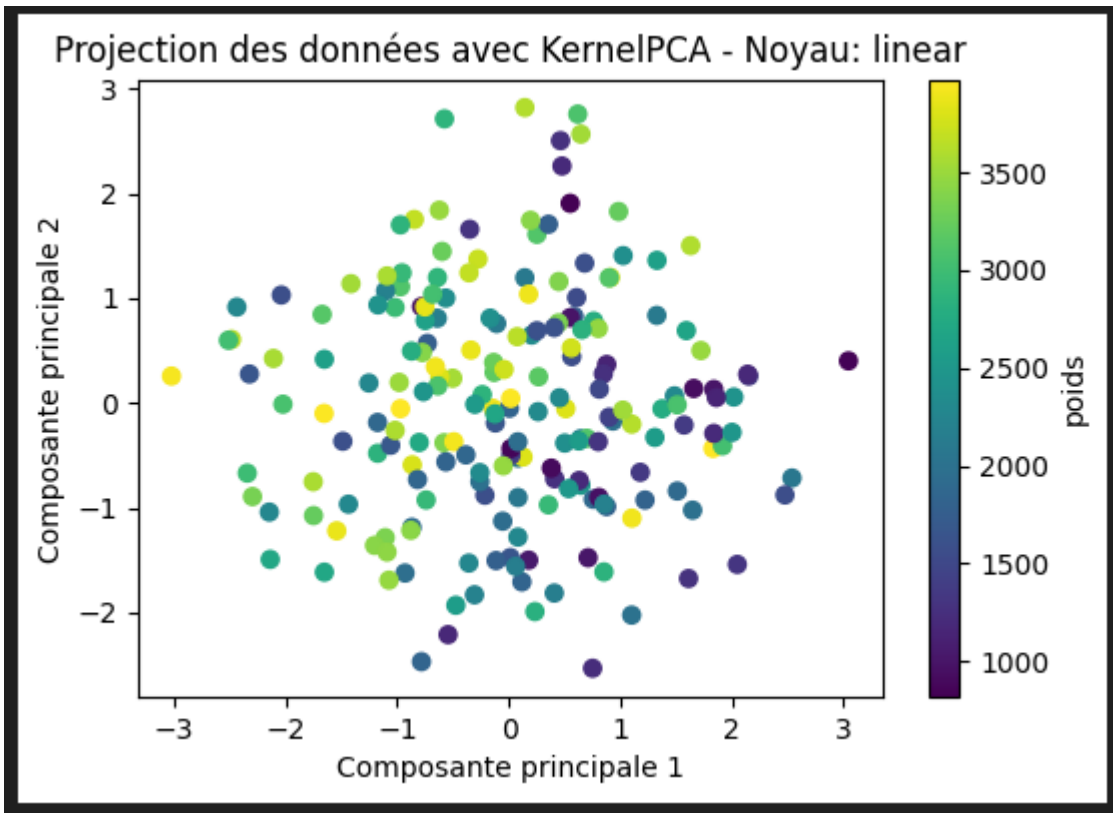
V. ACP à Noyau (KernelPCA)

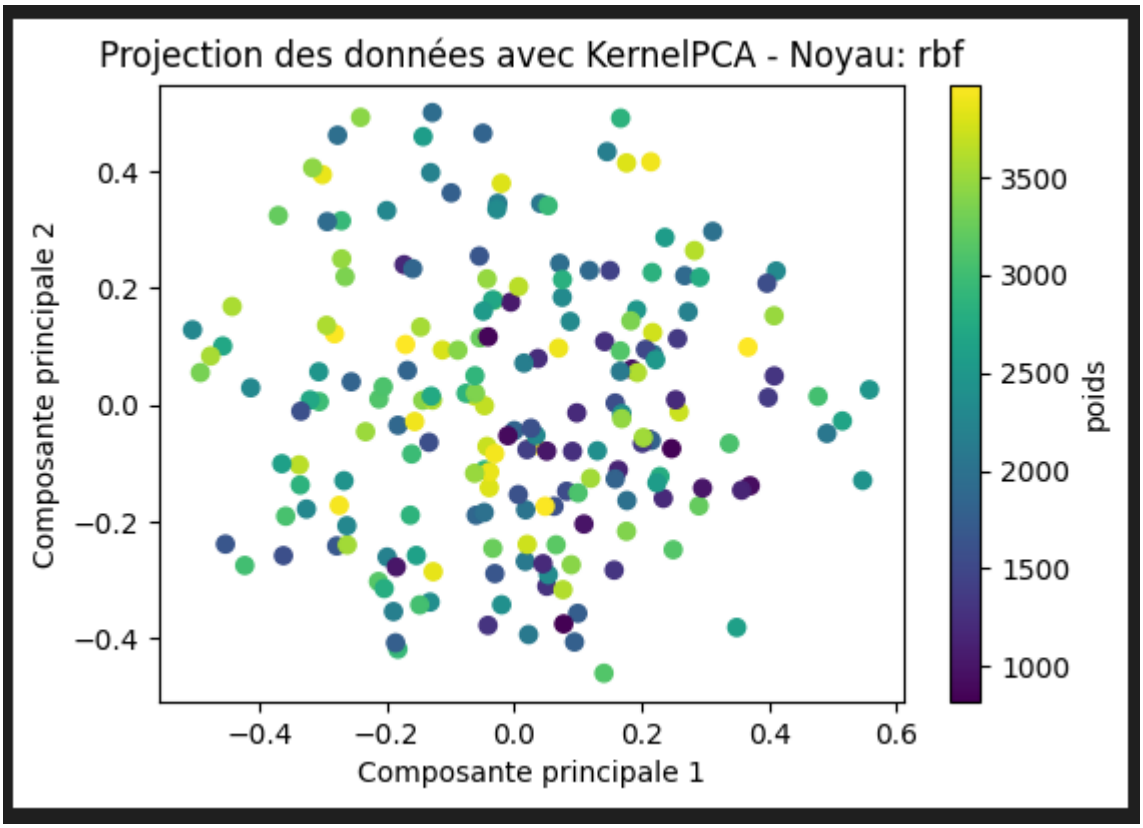
L'Analyse en Composantes Principales (ACP) est une méthode couramment utilisée pour réduire la dimensionnalité des données tout en préservant la variance la plus significative. Cependant, l'ACP classique ne capture que les relations linéaires entre les

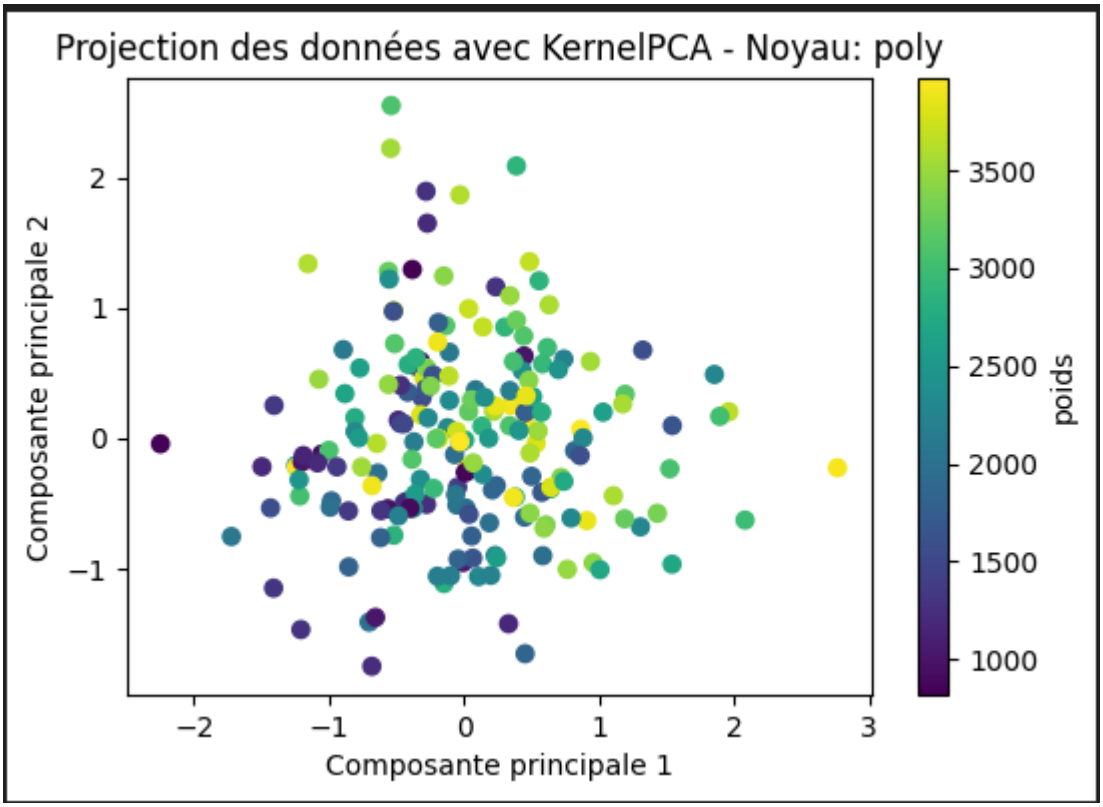
variables. Pour surmonter cette limitation, nous avons appliqué **l'ACP à noyau (Kernel PCA)**, qui permet d'explorer des structures non linéaires dans les données.

1. **Application de KernelPCA** : Nous avons testé trois noyaux différents :
    - **Linéaire** : Équivalent à une ACP classique.
    - **RBF (Radial Basis Function)** : Capture des structures non linéaires dans les données.
    - **Polynomial** : Capture les interactions non linéaires entre les variables.

2. **Visualisation des projections** : Les résultats ont été affichés sous forme de nuages de points colorés selon le poids.







Comparaison avec l'ACP classique :

- **Avec le noyau linéaire**, nous retrouvons des résultats similaires à l'ACP classique.
- **Avec le noyau RBF**, nous observons une meilleure séparation des groupes dans un espace non linéaire.
- **Avec le noyau polynomial**, la structure semble légèrement améliorée par rapport au noyau linéaire, mais moins efficace que RBF.

Quand l'ACP à noyau est-elle plus efficace ?

- L'**ACP classique** fonctionne bien lorsque les relations entre les variables sont **linéaires**.
- L'**ACP à noyau (KernelPCA)** est utile lorsque les données ont des structures **non linéaires**, comme des **formes complexes ou en spirale**.
- Le **noyau RBF** est souvent le plus performant lorsqu'on cherche à capturer des séparations non linéaires entre les classes.

Interpretation

- L'ACP classique est efficace pour réduire la dimension des données **lorsqu'elles suivent une distribution linéaire**.
- L'ACP à noyau offre une **meilleure flexibilité** en adaptant la transformation des données via des noyaux non linéaires.
- En fonction de la structure des données, un choix judicieux du noyau permet d'obtenir une meilleure séparation et une meilleure représentation des groupes.

Les résultats obtenus montrent que **l'ACP à noyau avec un noyau RBF** permet une **meilleure distinction des groupes**, contrairement à l'ACP classique qui peut être limitée dans certains cas non linéaires.

Partie 3 : Méthodes d'ensemble

VI. Bagging avec Random Forest

L'objectif est de prédire la survie des poulets en utilisant une **forêt aléatoire (RandomForestClassifier)** et d'évaluer les variables les plus influentes dans la prédiction.

Modélisation et évaluation des performances

Nous avons entraîné un **Random Forest** sur nos données avec les paramètres suivants :

- **100 arbres**
- **Séparation train/test : 80/20**

Les résultats obtenus sont :

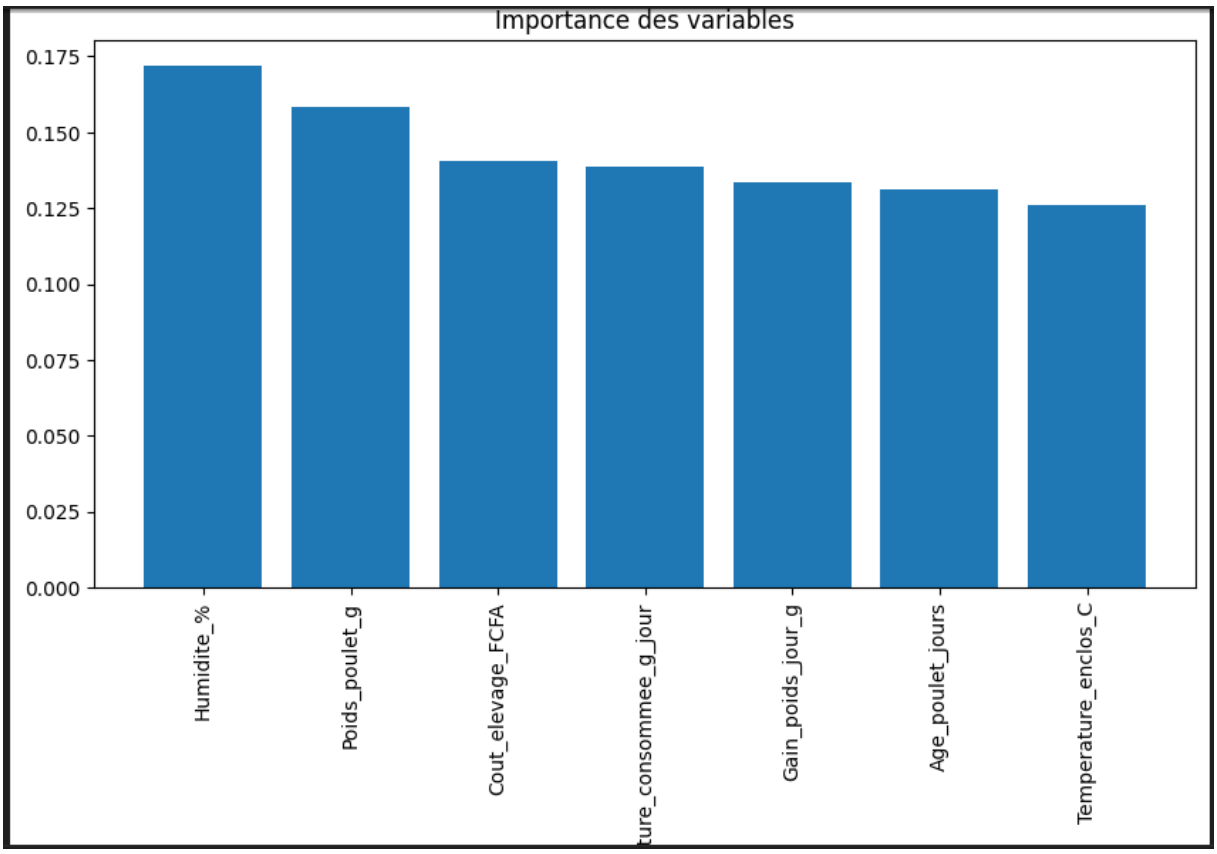
```
Accuracy : 0.6000
F1-score : 0.6000
```

- **Accuracy : 60.00 %**
- **F1-score : 60.00 %**

Ces scores indiquent une performance modérée, probablement due à des variables faiblement discriminantes ou un échantillon limité.

Importance des variables

L'importance des variables a été analysée, et les attributs ayant le plus d'impact sur la prédiction de la survie des poulets sont :
(*Ajout du graphique d'importance des variables*)



Les principales variables influentes sont (Humidité, poidds-poulet). Ces facteurs ont un impact direct sur la survie des poulets

En conclusion, **Random Forest** offre une approche robuste pour prédire la survie des poulets tout en identifiant les variables les plus critiques. Une amélioration possible serait d'ajuster les **hyperparamètres** ou d'utiliser d'autres techniques d'**apprentissage automatique** pour améliorer la performance.

VII. Boosting

L'objectif de cette analyse est de comparer les performances de deux algorithmes de boosting, **AdaBoost** et **Gradient Boosting**, dans la prédiction du **gain de poids journalier des poulets**. Ces modèles sont évalués à l'aide de trois métriques :

- **MAE (Mean Absolute Error)** : Indique l'erreur moyenne en valeur absolue.
- **RMSE (Root Mean Squared Error)** : Pénalise davantage les grandes erreurs.
- **R^2 (Coefficient de détermination)** : Mesure la qualité de la prédiction (idéalement proche de 1).

2. Prétraitement des données

Les données ont été divisées en un ensemble d'entraînement (**80%**) et un ensemble de test (**20%**). Afin d'améliorer la convergence des modèles, les **variables explicatives ont été standardisées** avec StandardScaler.

3. Résultats des modèles

Les performances des modèles sont les suivantes :

```
AdaBoost :
 MAE  : 4.2866
 RMSE : 5.0516
 R²   : -0.2097

Gradient Boosting :
 MAE  : 4.9609
 RMSE : 5.7406
 R²   : -0.5622
```

Modele	MAE	RMSE	R^2
AdaBoost	4.2866	5.0516	-0.2097
Gradient Boosting	4.9609	5.7406	-0.5622

4. Analyse des résultats

1. **AdaBoost semble mieux performer que Gradient Boosting** sur l'ensemble de test :
    - **MAE plus faible** : L'erreur absolue moyenne est plus basse, indiquant des prédictions globalement plus précises.
    - **RMSE plus faible** : Les erreurs importantes sont moins fréquentes.
    - **R^2 négatif** pour les deux modèles : Cela signifie que les modèles **n'expliquent pas bien la variance** des données et font pire qu'une simple moyenne des valeurs.
2. **Pourquoi ces performances sont-elles médiocres ?**

- **Possibles outliers** : AdaBoost et Gradient Boosting sont sensibles aux valeurs aberrantes, qui peuvent fausser leurs prédictions.
- **Données potentiellement bruitées** : Si certaines variables explicatives ne sont pas informatives ou si des relations non linéaires existent, cela peut affecter les résultats.
- **Paramètres par défaut sous-optimaux** : Il pourrait être nécessaire d'ajuster le nombre d'estimateurs (n_estimators), le taux d'apprentissage (learning_rate), ou d'autres hyperparamètres.

Les résultats suggèrent que **AdaBoost est plus robuste que Gradient Boosting** sur ces données. Toutefois, les scores négatifs de R^2 indiquent que les modèles doivent être améliorés. Voici quelques pistes :

- **Essayer une autre transformation des données** (normalisation, logarithme, etc.).
- **Gérer les outliers** (les détecter et les traiter).
- **Optimiser les hyperparamètres** (Grid Search, Random Search).
- **Tester d'autres modèles** (XGBoost, CatBoost).

Conclusion et perspectives

- L'analyse exploratoire a mis en évidence **une forte variabilité** et **des distributions asymétriques**.
- **L'ACP et KernelPCA** ont permis de **réduire la dimension** des données efficacement.
- **Random Forest** a fourni des résultats modérés, mais **le boosting doit être amélioré** par un meilleur prétraitement et un réglage des hyperparamètres.