

Data Analysis & Visualization (ADDA71)



Partie II

Méthodes d'Analyse Factorielle



01

Introduction aux méthodes factorielles



Introduction aux méthodes factorielles - Objectifs

- ✓ Comprendre le rôle des méthodes factorielles en analyse de données.
- ✓ Différencier réduction de dimension (ACP) et analyse exploratoire (AFC, ACM).
- ✓ Poser les bases statistiques nécessaires.

Plan de la séance

1. Introduction

1. Pourquoi réduire la dimension ? (curse of dimensionality, simplification, interprétation)
2. Exploratoire vs confirmatoire
3. Panorama des méthodes factorielles : ACP, AFC, ACM.

2. Rappels mathématiques/statistiques

1. Variance, covariance, corrélation
2. Notion de matrice de variance-covariance et de matrice de corrélation
3. Valeurs propres et vecteurs propres (intuition + exemple numérique simple)

3. Mise en contexte pratique

4. TP/exercice

1. Chargement d'un jeu de données en Python (pandas, seaborn).
2. Calcul et visualisation d'une matrice de corrélation.
3. Petit exercice manuel : calcul d'une ACP en 2D sur une matrice (2x2 ou 3x3).

1. Qu'est-ce qu'un jeu de données ?

Définition

Un *jeu de données* (ou *tableau de données*) est généralement un tableau :

- **Lignes** = individus / observations (ex. un client, un pays, un élève, une image...).
- **Colonnes** = variables / caractéristiques (ex. âge, revenu, taille, poids, note, catégorie...).

Exemple. 100 étudiants (100 lignes) \times 5 variables (taille, poids, âge, spécialité, réussite).

Types de variables.

- **Quantitatives** : nombres mesurables (taille, revenu).
- **Qualitatives** (catégorielles) : étiquettes/valeurs discrètes (sexe, spécialité, pays).

Vocabulaire minimal.

- **Dimension p** : nombre de variables utilisées en même temps. Si on travaille avec 7 variables \rightarrow dimension = 7.
- **Matrice de données** : notée X, de taille $n \times p$ (n lignes, p colonnes).
- **Centre d'une variable** : sa moyenne
- **écart-type d'une variable** : sa dispersion.

2. Visualiser et comprendre : 2D, 3D... et au-delà

On comprend bien un nuage de points en **2D** (plan) et parfois en **3D** (espace).

Mais quand $p > 3$ (ex. 20 variables), on ne peut plus « voir » directement.

Idée clé : Trouver des **axes synthétiques** (combinaisons des variables) pour projeter le nuage en 2D/3D **sans perdre trop d'information**. C'est le rôle des **méthodes d'analyse factorielle**.

Imaginez une **statue 3D** éclairée par une lampe : son **ombre 2D** garde une partie de l'information. On veut une ombre **la plus informative possible**.

3. Pourquoi réduire la dimension ?

Données **haute dimension** → complexité, bruit, coût de calcul.

- 1. Simplifier** l'analyse : passer de 50 variables à 2–3 axes lisibles.
- 2. Visualiser** : cartes, biplots, cercles des corrélations.
- 3. Denoising** : diminuer le bruit et les redondances (variables corrélées).
- 4. Calcul** : certains algorithmes deviennent plus rapides/stables.
- 5. Communication** : raconter l'histoire des données à un public non technique.

Attention. Réduire la dimension ≠ prédire une cible. Les méthodes factorielles sont d'abord **exploratoires**.

4. Trois méthodes sœurs, trois types de données

Situation	Type de données	Méthode	Sorties usuelles
Variables quantitatives seulement	Mesures numériques	ACP (PCA)	Variance expliquée, axes/composantes , biplot, cercle des corrélations
Tableau croisé (2 qualitatives)	Lignes et colonnes = modalités	AFC (CA)	Carte lignes/colonnes, inertie par axe
Plusieurs qualitatives (≥ 3)	Profil d'individus par catégories	ACM (MCA)	Carte individus + modalités, inertie par axe

Règle pratique.

- Si *chaque colonne est un nombre* → **ACP**.
- Si on a une **table de contingence** (comptages) → **AFC**.
- Si on a **plusieurs colonnes catégorielles** → **ACM**.

5. Rappels indispensables

5.1. Variance, covariance, corrélation

- **Variance** : mesure la **dispersion** d'une variable autour de sa moyenne. Plus la variance est grande, plus les valeurs sont éparpillées.
- **Covariance** $\text{Cov}_{(X,Y)}$: indique si deux variables **varient ensemble**. Positive (elles montent ensemble), négative (l'une monte quand l'autre baisse), proche de 0 (pas de lien linéaire fort).
- **Corrélation** $\rho_{(X,Y)}$: covariance **normalisée** entre -1 et $+1 \rightarrow$ comparables entre variables d'unités différentes.

5. Rappels indispensables

5.2. Standardisation (centrer-réduire)

Pour comparer des variables **d'unités différentes** (euros, cm, kg...), on **centre** (enlève la moyenne) et on **réduit** (divise par l'écart-type).

- Après standardisation, chaque variable a moyenne 0 et écart-type 1.
- **En ACP**, on travaille souvent sur les **données standardisées** (ou sur la **matrice de corrélations**).

5. Rappels indispensables

Exemple : Partons d'un tableau **X** avec n lignes (individus) et p colonnes (variables).

Pour **chaque variable** (chaque **colonne j**), on fait la même recette :

1.Centrer = *enlever la moyenne de la colonne.*

Pour chaque valeur x_{ij} (ligne i , colonne j), on calcule :

$$x_{ij}^{\text{centré}} = x_{ij} - \bar{x}_j$$

où \bar{x}_j = moyenne de la **colonne j** calculée sur **toutes** les lignes.

2.Réduire = *diviser par l'écart-type de la colonne.*

$$z_{ij} = \frac{x_{ij} - \bar{x}_j}{S_j}$$

où S_j = écart-type de la **colonne j** ;

Au final, **chaque colonne j standardisée a moyenne 0 et écart-type 1.**

On appelle z_{ij} un **z-score**.

5. Rappels indispensables

Pourquoi on fait ça en ACP ?

Parce que sinon, les variables **à grande échelle/variance** (ex. revenus en €) **écrasent** les petites (ex. taille en cm).

- **ACP sur covariances** (sans standardiser) : utile si toutes les variables sont sur **la même unité et comparables**.
- **ACP sur corrélations** (après standardisation) : met **chaque variable sur le même pied** (variance = 1). C'est ce qu'on fait **le plus souvent** quand les unités diffèrent.

Dire "on travaille sur la **matrice de corrélations**" → c'est **exactement** faire une ACP sur les **données standardisées** (ou, équivallement, diagonaliser la matrice des corrélations).

5. Rappels indispensables

5.3. Matrices utiles

- **Matrice de covariance** $\Sigma_{(p \times p)}$: toutes les covariances entre variables.
- **Matrice de corrélation** $R_{(p \times p)}$: covariances normalisées (entre -1 et +1).

5. Rappels indispensables

Mini-exemple (avec chiffres propres)

Étudiant	Taille (cm)	Revenu (€)
A	160	1000
B	170	2000
C	180	3000

Moyennes par colonne : $\bar{x}_{\text{taille}} = 170$, $\bar{x}_{\text{revenu}} = 2000$

Écarts-types (échantillon, $n - 1$) : $s_{\text{taille}} = 10$, $s_{\text{revenu}} = 1000$

Centrage (on enlève la moyenne de la colonne)

Étudiant	Taille centrée	Revenu centré
A	$160 - 170 = -10$	$1000 - 2000 = -1000$
B	$170 - 170 = 0$	$2000 - 2000 = 0$
C	$180 - 170 = +10$	$3000 - 2000 = +1000$

5. Rappels indispensables

Réduction (on divise par l'écart-type de la colonne)

Étudiant	Taille z-score	Revenu z-score
A	$-10 / 10 = -1$	$-1000 / 1000 = -1$
B	$0 / 10 = 0$	$0 / 1000 = 0$
C	$+10 / 10 = +1$	$+1000 / 1000 = +1$

Après standardisation : moyenne = 0, écart-type = 1 pour chaque colonne.

Du point de vue de l'ACP, taille et revenu pèsent autant au départ.

Règle pratique :

- Variables sur des unités différentes → standardiser et faire ACP (corrélation).
- Variables comparables (même unité, mêmes ordres de grandeur) → possible de faire ACP (covariance) sans réduire (on centre tout de même).
- Variable quasi constante ($\text{écart-type} \approx 0$) → la retirer ou la regrouper : division par 0 impossible et elle n'apporte rien.

6. Intuition de l'ACP (Analyse en Composantes Principales)

Si je dois tracer les données sur **un seul axe**, **lequel** garde le plus d'information ?

- Chercher la **direction** (unité) qui **maximise la variance projetée** des points.
- Cet axe s'appelle la **1ère composante**.
- Puis on cherche une **2e composante orthogonale** à la première, etc.

Lien mathématique (idée simple). Les axes obtenus sont les **vecteurs propres** de la matrice de covariance (ou corrélation) ; la **variance captée par un axe = sa valeur propre**.

- **Grande valeur propre** → axe très informatif.
- **Somme des valeurs propres** = variance totale (information totale) du jeu de données (après centrage/réduction).

6. Intuition de l'ACP (Analyse en Composantes Principales)

C'est quoi une valeur propre ?

Pour une matrice carrée A (par ex. une **matrice de covariance ou de corrélation**), une **valeur propre** λ est un nombre pour lequel il existe un **vecteur non nul** u tel que

$$A u = \lambda u.$$

- u = **vecteur propre** (la **direction** de l'axe).
- λ = **valeur propre** (la **quantité d'information/variance** que cet axe capture).

Dans l'ACP, A est symétrique (covariance/correlation) \Rightarrow les λ sont **réels et ≥ 0** .

La **somme** des λ = **variance totale** (la trace de A).

Sur une **matrice de corrélations** avec p variables standardisées, $\sum \lambda = p$.

6. Intuition de l'ACP (Analyse en Composantes Principales)

À partir de la matrice de corrélation/covariance

```
import numpy as np

# A = matrice de corrélation ou covariance (p×p)
eigvals, eigvecs = np.linalg.eigh(A)          # A symétrique → eigh
idx = eigvals.argsort()[-1:]                  # tri décroissant
eigvals = eigvals[idx]
eigvecs = eigvecs[:, idx]                      # colonnes = vecteurs propres
explained_ratio = eigvals / eigvals.sum()      # % de variance par axe
```

Directement avec **scikit-learn** (sur données standardisées Z)

```
from sklearn.decomposition import PCA
pca = PCA().fit(Z)
eigvals = pca.explained_variance_              # ≈ valeurs propres
explained_ratio = pca.explained_variance_ratio_
cum_ratio = explained_ratio.cumsum()
```

6. Intuition de l'ACP (Analyse en Composantes Principales)

Choisir le nombre d'axes. On regarde :

- la courbe des éboulis (*scree plot*) : on garde jusqu'au coude ;
- la variance expliquée cumulée (ex. viser 70–90% selon le contexte) ;
- parfois le critère de Kaiser (garder $\lambda > 1$ quand on travaille sur corrélations).

Sorties visuelles.

- Biplot : individus projetés + vecteurs des variables.
- Cercle des corrélations : montre comment chaque variable « charge » sur les axes.

À quoi ça sert concrètement ?

- Repérer des groupes d'individus, des tendances et des oppositions.
- Comprendre quelles variables définissent chaque axe (ex. « axe 1 = taille/poids », « axe 2 = age/activité »).

6. Intuition de l'ACP (Analyse en Composantes Principales)

Préparer les données

- Garde uniquement les colonnes numériques.
- Standardise (toujours, sauf si toutes les colonnes ont la même unité/échelle).
- Gère les NA (supprimer/imbriquer un imputer).

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA

# EXEMPLE : prends ton DataFrame df (numérique). Sinon, dataset de démo :
# from sklearn.datasets import load_wine
# X = pd.DataFrame(load_wine().data, columns=load_wine().feature_names)

X = df.select_dtypes(include=[np.number]).dropna() # <- remplace df par ton tableau
scaler = StandardScaler()
Z = scaler.fit_transform(X)                      # données centrées-réduites
feature_names = X.columns.tolist()
```

6. Intuition de l'ACP (Analyse en Composantes Principales)

1) Lancer l'ACP

```
pca = PCA()                      # toutes les composantes
scores = pca.fit_transform(Z)      # projections des individus ( $n \times p$ )
eigvals = pca.explained_variance_  # ~ "valeurs propres"
ratio = pca.explained_variance_ratio_
cum    = ratio.cumsum()           # variance expliquée cumulée
```

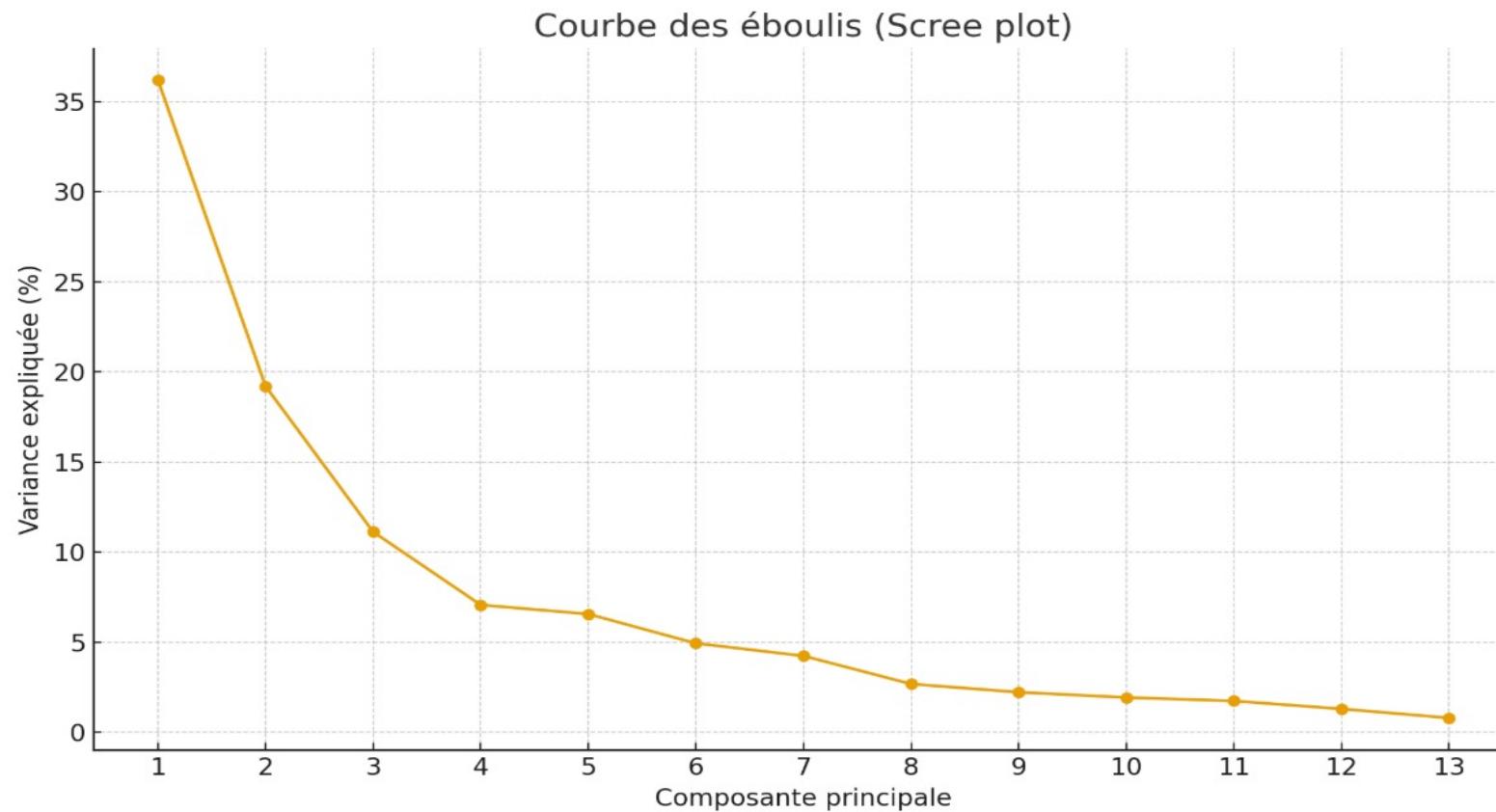
2) Courbe des éboulis (Scree plot) → combien d'axes garder ?

```
plt.figure()
plt.plot(range(1, len(ratio)+1), ratio*100, marker='o')
plt.xlabel("Composante principale"); plt.ylabel("Variance expliquée (%)")
plt.title("Courbe des éboulis"); plt.xticks(range(1, len(ratio)+1))
plt.show()
```

Comment lire (simple) :

- Cherche le **coude** : après ce point, les barres chutent doucement ⇒ peu d'info nouvelle.
- Si tu vois 2–3 très hautes barres puis un plateau → garde 2–3 axes.

6. Intuition de l'ACP (Analyse en Composantes Principales)



Comment lire (simple) :

- Cherche le **coude** : après ce point, les barres chutent doucement \Rightarrow peu d'info nouvelle.
- Si tu vois 2-3 très hautes barres puis un plateau \rightarrow garde 2-3 axes.

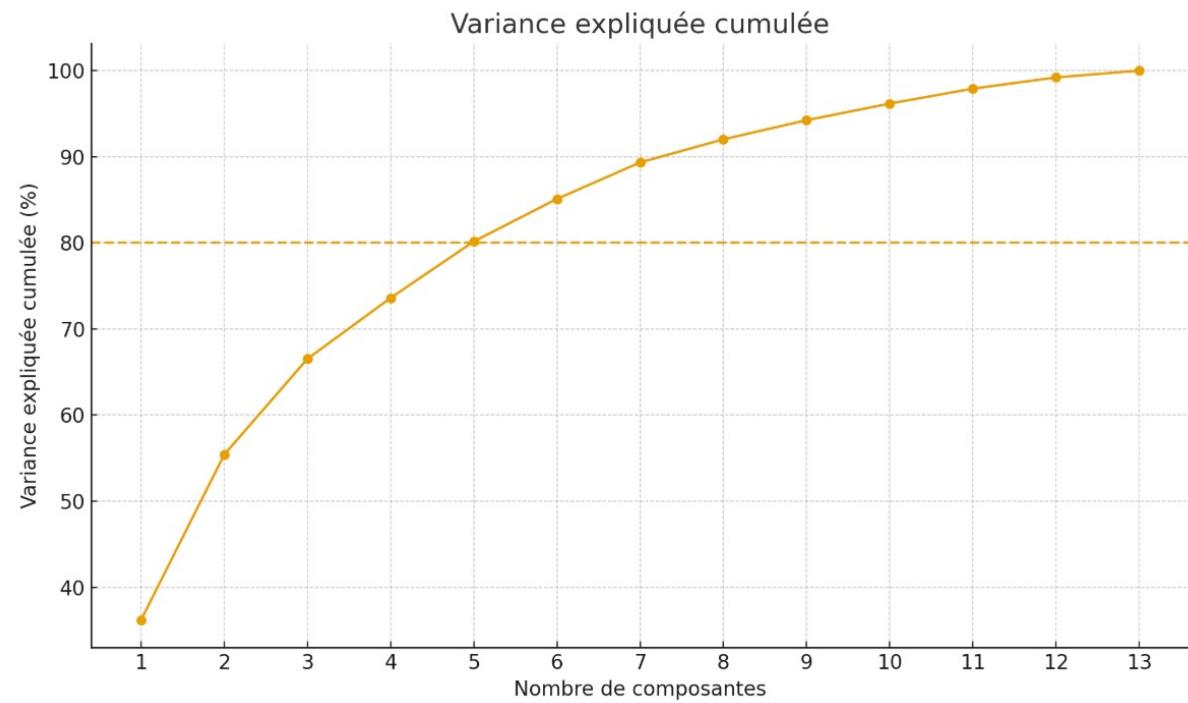
6. Intuition de l'ACP (Analyse en Composantes Principales)

3) Variance expliquée cumulée → niveau d'info global

```
plt.figure()
plt.plot(range(1, len(cum)+1), cum*100, marker='o')
plt.axhline(80, linestyle='--') # exemple de seuil 80%
plt.xlabel("Nombre de composantes"); plt.ylabel("Variance expliquée cumulée (%)")
plt.title("Variance expliquée cumulée"); plt.xticks(range(1, len(cum)+1))
plt.show()
```

Comment lire :

- Choisis le **plus petit K** tel que le **cumul \geq 70–90%** (selon ton besoin).
- Ex. si K=4 atteint 80%, **4 axes** suffisent.



6. Intuition de l'ACP (Analyse en Composantes Principales)

4) Kaiser (si ACP sur corrélations \Rightarrow après standardisation)

python

```
kaiser_keep = np.sum(eigvals > 1.0)
print("Nb d'axes (Kaiser,  $\lambda>1$ ) :", kaiser_keep)
```

Comment lire :

- **Garde** les axes avec $\lambda > 1$ (ils “valent” plus qu’une variable standardisée).
- Compare avec le coude + cumul pour décider.

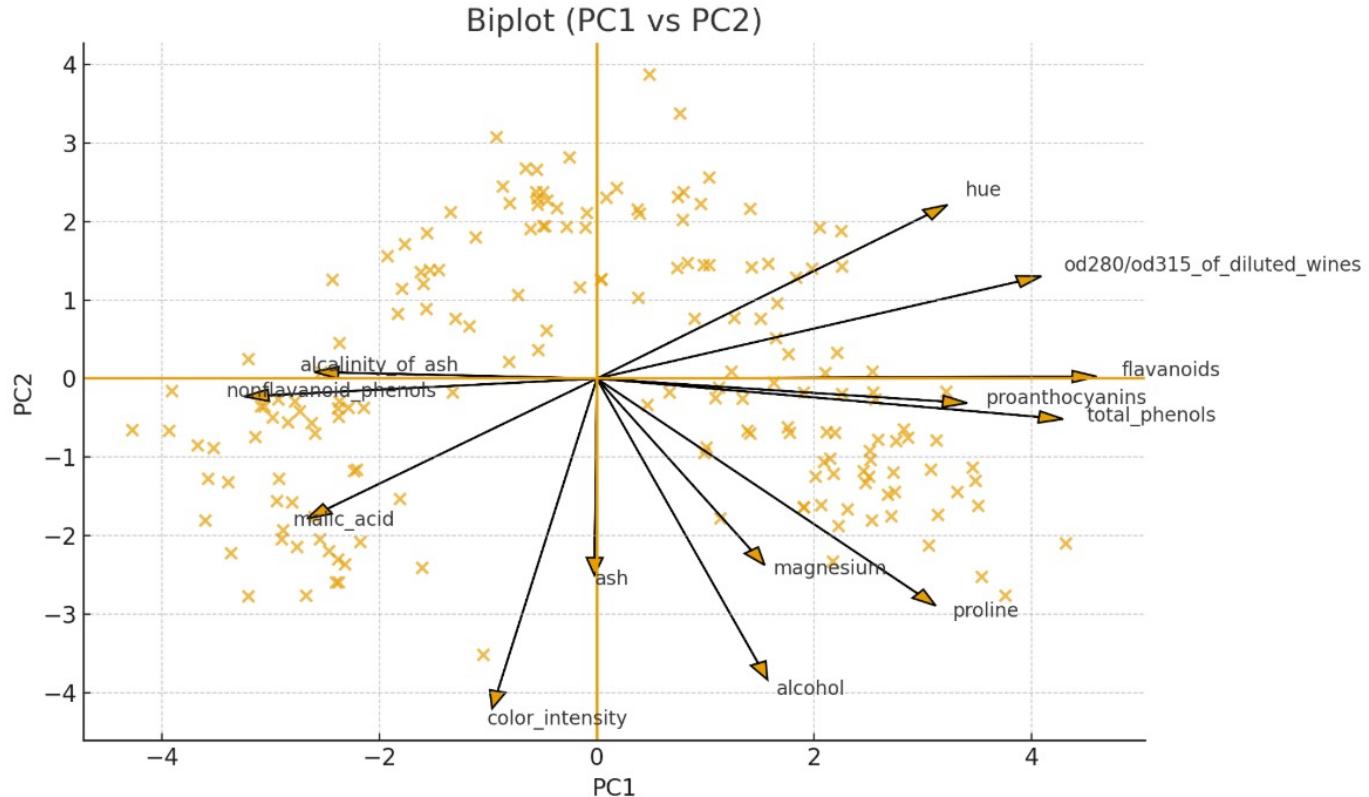
5) Biplot (PC1 vs PC2) → individus + variables

```
# "Loadings" = corrélation variable ↔ composante (avec données standardisées)
loadings = pca.components_.T * np.sqrt(pca.explained_variance_) # (p, p)

def biplot(scores2d, loads2d, names):
    plt.figure()
    plt.scatter(scores2d[:,0], scores2d[:,1], alpha=0.7)
    scale = 5.0 # agrandit les flèches pour les voir
    for i, name in enumerate(names):
        x, y = loads2d[i,0]*scale, loads2d[i,1]*scale
        plt.arrow(0,0,x,y, head_width=0.15, length_includes_head=True)
        plt.text(x*1.05, y*1.05, name)
    plt.axhline(0); plt.axvline(0)
    plt.xlabel("PC1"); plt.ylabel("PC2"); plt.title("Biplot (PC1 vs PC2)")
    plt.show()

biplot(scores[:, :2], loadings[:, :2], feature_names)
```

5) Biplot (PC1 vs PC2) → individus + variables



Comment lire :

- Points = individus ; flèches = variables.
- Flèche longue ⇒ variable **importante** dans le plan PC1–PC2 (bien expliquée).
- Flèches proches ⇒ variables **corrélées** ; opposées ⇒ **négativement corrélées** ; à 90° ⇒ peu corrélées.
- Un individu projeté dans le sens d'une flèche ⇒ valeur élevée pour cette variable.

6. Intuition de l'ACP (Analyse en Composantes Principales)

6) Cercle des corrélations → qualité de représentation des variables

```
def correlation_circle.loads2d, names):
    plt.figure()
    ax = plt.gca()
    circle = plt.Circle((0,0), 1, fill=False)
    ax.add_artist(circle)
    for i, name in enumerate(names):
        x, y = loads2d[i,0], loads2d[i,1]
        plt.scatter(x, y); plt.arrow(0,0,x,y, head_width=0.03, length_includes_head=True)
        plt.text(x*1.08, y*1.08, name)
    plt.axhline(0); plt.axvline(0)
    plt.xlim(-1.1,1.1); plt.ylim(-1.1,1.1); plt.gca().set_aspect("equal", "box")
    plt.xlabel("PC1 (corr)"); plt.ylabel("PC2 (corr)"); plt.title("Cercle des corrélations")
    plt.show()

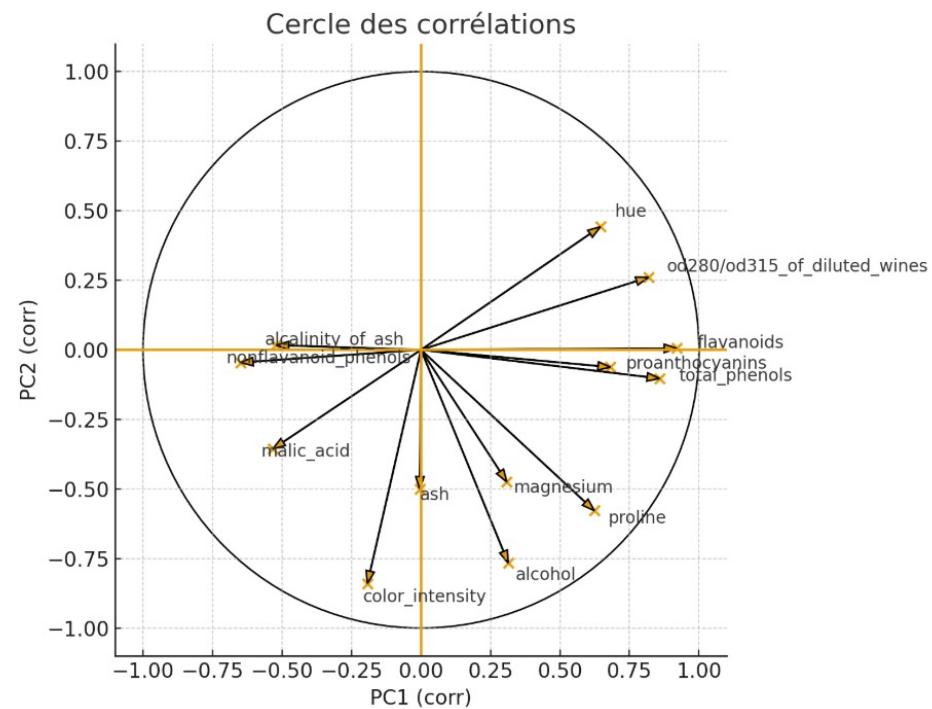
correlation_circle(loadings[:, :2], feature_names)
```

Comment lire :

- Chaque flèche = **corrélation** de la variable avec **PC1** (x) et **PC2** (y).
- **Près du bord** du cercle ⇒ très bien expliquée par PC1–PC2.
- **Près de l'origine** ⇒ peu expliquée par ces 2 axes (regarder PC3/PC4...).
- **Angle** entre flèches = **corrélation** entre variables (mêmes règles que biplot).

6. Intuition de l'ACP (Analyse en Composantes Principales)

6) Cercle des corrélations → qualité de représentation des variables



Comment lire :

- Chaque flèche = **corrélation** de la variable avec **PC1** (x) et **PC2** (y).
- **Près du bord** du cercle ⇒ **très bien expliquée** par PC1–PC2.
- **Près de l'origine** ⇒ **peu** expliquée par ces 2 axes (regarder PC3/PC4...).
- **Angle** entre flèches = **corrélation** entre variables (mêmes règles que biplot).

7. Intuition de l'AFC (Analyse Factorielle des Correspondances)

Quand ? Quand on étudie un **tableau croisé** (contingence) : lignes = **modalités d'une variable** (ex. Pays), colonnes = **modalités d'une autre** (ex. *Plat préféré*), cellules = **effectifs** (comptages).

Idée. L'AFC cherche une **projection** qui met en évidence les **proximités** entre **modalités** (lignes et colonnes) en utilisant une **distance du khi-deux** (qui tient compte des tailles marginales).

Mots clés.

- **Inertie** : analogue de la variance en ACP, mesure la dispersion autour d'un modèle d'indépendance.
- **Axes factoriels** : directions qui expliquent le plus d'inertie.
- **Cartes** : positions des modalités lignes/colonnes ; on lit les **rapprochements**.

Applications. Marketing (profils produits × clients), sociologie (pratiques × catégories sociales), textes (mots × documents, après mise en tableau).

8. Intuition de l'ACM (Analyse des Correspondances Multiples)

Quand ? Quand on a **plusieurs variables qualitatives** pour les mêmes individus (ex. sexe, *niveau d'étude, profession, ville...*).

Idée. Transformer les variables en un **grand tableau indicateur** (0/1) des **modalités** et appliquer une logique proche de l'AFC.

Lecture. La **carte** montre quelles **modalités** et quels **profils d'individus** se rapprochent (vont souvent ensemble).

Usages. Typologies, segmentation de clientèle, enquêtes de satisfaction.

9. Pas-à-pas : de la table brute à la première projection

1. **Nettoyer** : valeurs manquantes, aberrantes, types corrects (nombre vs catégorie).
2. **Choisir la méthode** selon les variables (ACP / AFC / ACM).
3. **Standardiser** (si ACP avec échelles différentes).
4. **Calculer** l'ACP/AFC/ACM (logiciel ou Python).
5. **Lire** les graphiques : variance/inertie par axe, biplot/cercle/cartes.
6. **Interpréter** : nommer les axes, raconter les proximités/contrastes.
7. **Communiquer** : 2–3 graphiques + messages clés (qui ? quoi ? pourquoi ?).

Bonnes pratiques & pièges

- Toujours **centrer** (et souvent **réduire**).
- Vérifier **valeurs extrêmes** / outliers.
- Interprétation : un **axe** = un **compromis** de variables.
- Corrélation \neq causalité ; ACP \neq clustering.