TD05 : La distribution des salaires

Décrire, mesurer les inégalités, comparer des groupes

Anderson VIL | L3 Économie | 90 minutes
Analyse des données | CY Cergy Paris Université
Année universitaire 2026–2027

Rappels : petits exemples numériques

TD05 : La distribution des salaires

Dix salaires fictifs, à vérifier à la main

Rang	1	2	3	4	5	6	7	8	9	10
Salaire (€)	1 000	1 200	1 300	1 500	1 600	1 800	2 000	2 400	3 000	4 200

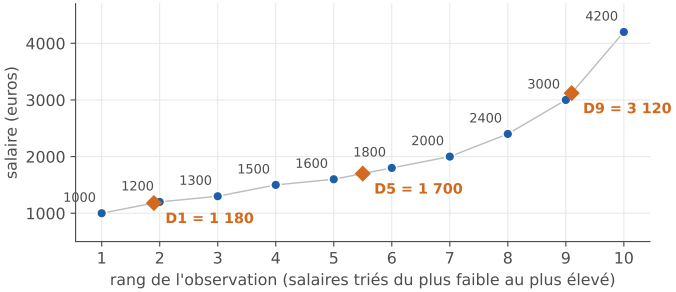
Masse salariale
somme des dix salaires
$1\,000 + \cdots + 4\,200$
$= 20\,000$ €

Moyenne
masse / effectif
$20\,000/10$
$= 2\,000$ €

Médiane
entre le 5^{e} et le 6^{e}
$(1\,600 + 1\,800)/2$
$= 1\,700$ €

La moyenne dépasse la médiane. Le salaire de 4 200 € tire la moyenne vers le haut. La médiane partage les observations en deux moitiés **d'effectifs** égaux.

Situer les déciles sur la série ordonnée




```
ex.quantile([.1, .5, .9])
```

SORTIE RÉELLE

```
0.1    1180.0
0.5    1700.0
0.9    3120.0
```

Probabilités **entre 0 et 1** : .1 pour 10 %. D_1 tombe **entre** deux observations.

Pourquoi $D_1 = 1\,180$: l'interpolation de `quantile`

Par défaut, pandas place le quantile p au rang $1 + (n-1)\,p$ des valeurs triées.
D_1 : rang $1 + 9 \times 0{,}1 = \mathbf{1}{,}\mathbf{9}$, soit 90 % du chemin entre le 1er et le 2e salaire.

$$D_1 = 1\,000 + 0{,}9 \times (1\,200 - 1\,000) = 1\,180 \qquad D_9 = 3\,000 + 0{,}1 \times (4\,200 - 3\,000) = 3\,120$$

D'autres logiciels ou d'autres options (`interpolation="lower"`...) donnent des valeurs voisines. Sur 6 812 salaires, l'écart entre deux observations voisines est minime.

Contrôle. La médiane de pandas est au rang $1 + 9 \times 0{,}5 = 5{,}5$: $(1\,600 + 1\,800)/2 = 1\,700$, comme à la main.

Un rapport de seuils n'est pas une part de masse

Rapport interdécile
$D_9/D_1 = 3\,120/1\,180 = \mathbf{2{,}64}$
Il compare **deux seuils** : la frontière du dixième le mieux payé et celle du dixième le moins payé.

Parts de la masse
Haut : $4\,200/20\,000 = \mathbf{21}$ %
Bas : $1\,000/20\,000 = \mathbf{5}$ %
Elles additionnent **tous les salaires** de chaque groupe.

Seuil et moyenne d'un groupe. Le dixième le mieux payé contient ici une seule observation : sa moyenne vaut 4 200 €, au-dessus du seuil $D_9 = 3\,120$ €. Un seuil ne dit rien de ce qui se passe au-delà.

Proportions conditionnelles et points de pourcentage

	Temps complet	Temps partiel	Total
Hommes	40	10	50
Femmes	30	20	50

Proportion conditionnelle
Parmi les hommes : $10/50 = 20$ %.
Parmi les femmes : $20/50 = 40$ %.
Le dénominateur est **le total de la ligne**.

Deux comparaisons
Différence : $40 - 20 =$ **20 points de pourcentage**.
Rapport : $40/20 =$ **2** : deux fois plus fréquent.
Écrire « 20 % de plus » serait ambigu.

Manquant, hors champ, plafonné

Code	Signification	Traitement dans le notebook
999 999	Non-réponse	Remplacé par `NaN` ? la valeur manquante
0	Non-salarié : hors champ	Ligne exclue : pas un salaire nul
10 000	Plafond de diffusion	la vraie valeur est **au moins** 10 000 €

`SALAIRE_NET`

Salaire mensuel net en euros, virgule décimale, espace pour les milliers.

Censure haute. Les salaires supérieurs à 10 000 euros sont ramenés à 10 000 : c'est le plafond de diffusion. Tout indicateur d'inégalité calculé sur cette colonne est une borne inférieure.

Champ. Salariés uniquement. La valeur est **0** pour les non-salariés : c'est du **sans objet**. Un non-salarié ne perçoit pas de salaire, il n'a pas un salaire nul. **Non-réponse.** codée **999 999**.

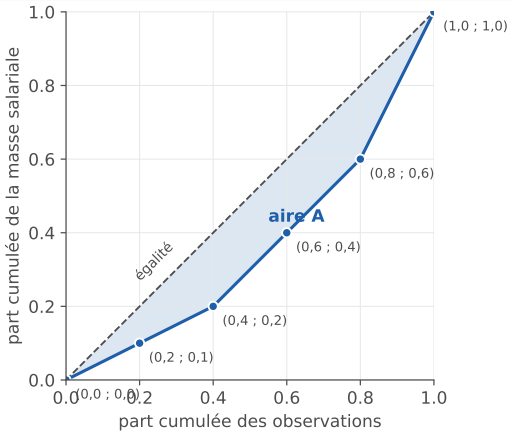
Cinq salaires, deux parts cumulées

Rang	Salaire	Masse cumulée	Part cumulée des observations	Part cumulée de la masse
0		0	0	0
1	1 000	1 000	0,2	0,1
2	1 000	2 000	0,4	0,2
3	2 000	4 000	0,6	0,4
4	2 000	6 000	0,8	0,6
5	4 000	10 000	1,0	1,0

On trie du plus faible au plus élevé. Les 40 % les moins payés reçoivent 20 % de la masse, les 20 % les mieux payés en reçoivent $1 - 0{,}6 = 40$ %.

Pause recherche : quelle serait la dernière colonne si les cinq salaires valaient 2 000 € ?

La courbe de Lorenz et la diagonale d'égalité



Abscisse : part cumulée des observations, classées du plus faible au plus élevé.
Ordonnée : part cumulée de la masse salariale.

Diagonale : situation où tous les salaires sont égaux. Chaque fraction des observations reçoit la même fraction de la masse.

Plus la courbe s'éloigne de la diagonale, plus la masse est concentrée en haut de la distribution.

Le Gini est un rapport d'aires

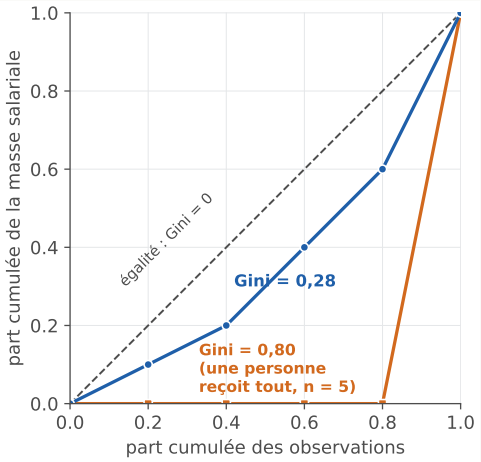
$$G = \frac{\text{aire A entre la diagonale et la courbe}}{\text{aire sous la diagonale}} = \frac{A}{1/2} = 2A$$

Aire sous la courbe, par trapèzes de largeur 0,2 : $0{,}2 \times \left(\frac{0+0{,}1}{2} + \frac{0{,}1+0{,}2}{2} + \frac{0{,}2+0{,}4}{2} + \frac{0{,}4+0{,}6}{2} + \frac{0{,}6+1}{2} \right) = 0{,}36$

$A = 0{,}5 - 0{,}36 = 0{,}14$
$G = 0{,}14/0{,}5 = \mathbf{0{,}28}$

0,28 **n'est pas** « 28 % des salariés » ni « 28 % des salaires ». C'est un indice sans unité, entre 0 et 1.

Égalité, situation inégalitaire, situation extrême



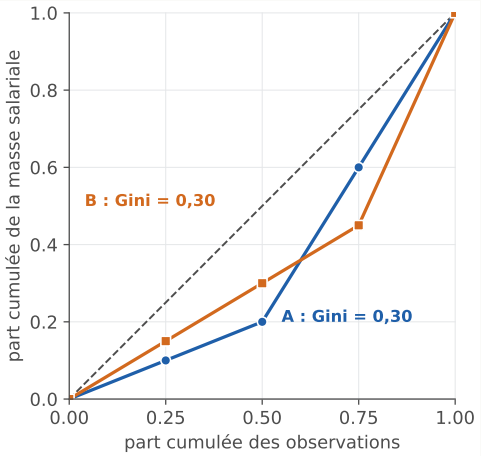
Égalité : 5 salaires de 2 000 € : $G = 0$.

Notre exemple : $G = 0{,}28$.

Une seule observation reçoit tout : $0, 0, 0, 0, 10\,000$. Avec $n = 5$, la formule du notebook donne $G = 0{,}80 = 1 - 1/5$, pas 1.

Approfondissement. Avec la formule empirique fournie, le maximum à effectif fini est $1 - 1/n$. Pour 6 812 salaires : 0,99985.

Deux distributions différentes, un même Gini



	Salaires (€)	Gini
A	1 000 ; 1 000 ; 4 000 ; 4 000	0,30
B	1 500 ; 1 500 ; 1 500 ; 5 500	0,30

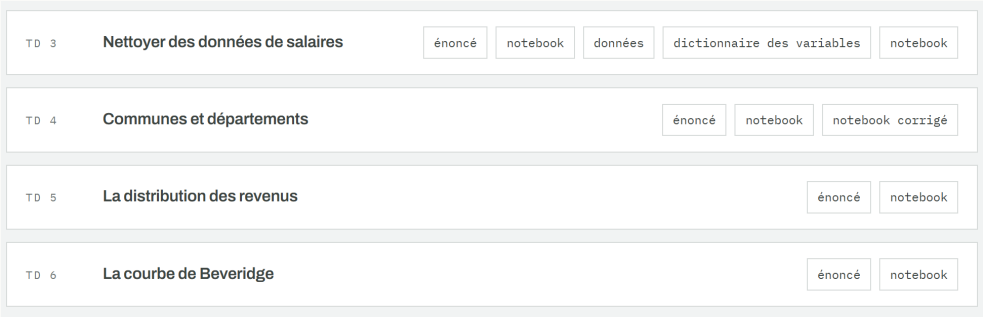
Les courbes se croisent. A défavorise davantage le **bas** : 20 % de la masse pour la moitié la moins payée, contre 30 % dans B. B concentre davantage le **haut** : 55 % pour le dernier quart, contre 40 %.

Le Gini résume en un nombre. On le présente **avec** les déciles et les parts de masse.

Ressources et prise en main

TD05 : La distribution des salaires

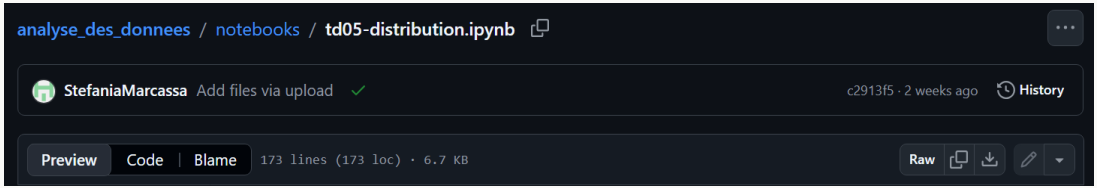
La page du cours : énoncé et notebook du TD05



Page du cours **Énoncé TD05** **Dictionnaire des variables**
La ligne TD 5 s'intitule en ligne « La distribution des revenus » ; l'énoncé et le notebook portent sur les **salaires**.

Source : capture de la page publique du cours, 14/10/2026, lignes TD 3 à TD 6.

GitHub : afficher ou télécharger le notebook

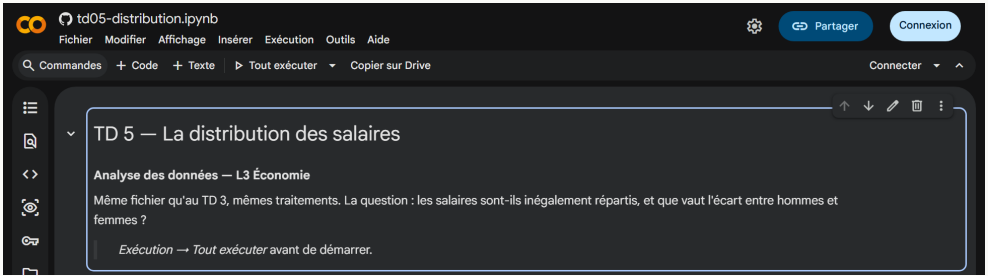


Notebook étudiant sur GitHub **Téléchargement direct du .ipynb**
Bouton **Raw** ou icône de téléchargement à droite.

Les blancs _____ sont à remplacer. La prévisualisation GitHub affiche les cellules sans les exécuter.

Source : capture GitHub, 14/10/2026.

Colab : ouvrir le notebook et garder une copie personnelle



Ouverture directe du TD05 dans Colab **Google Colab**
Se connecter, puis **Copier sur Drive** ou **Fichier / Enregistrer une copie dans Drive**.

Source : capture Colab sans connexion, 14/10/2026.

Importer, exécuter une cellule, compléter les blancs

Depuis le fichier téléchargé
1. Ouvrir **Colab** et se connecter.
2. **Fichier / Importer le notebook**.
3. Choisir `td05-distribution.ipynb`.
4. Enregistrer une copie personnelle.

Pendant le TD
1. Exécuter la cellule de la partie 0.
2. Lire la question, remplir les blancs.
3. Exécuter : triangle ou `Maj+Entrée`.
4. Lire la sortie sous la cellule.

La consigne « **Tout exécuter** avant de démarrer » s'arrête ici à la première cellule à trous : `NameError: name '_______' is not defined`. Compléter et tester les cellules une par une.

Contrôle. À la fin, redémarrer la session, puis **Exécution / Tout exécuter**. Aucune cellule ne doit afficher d'erreur.

Partie 0 — Comprendre le fichier nettoyé

TD05 : La distribution des salaires

1 Importer les données

```
import pandas as pd
import numpy as np
from scipy import stats
URL = ("https://raw.githubusercontent.com/StefaniaMarcassa/"
"analyse_des_donnees/main/data/cm03-salaires.csv")
def en_nombre(serie):
return pd.to_numeric(serie.astype(str).str.replace(" ", "", regex=False)
.str.replace(",", ".", regex=False), errors="coerce")
df = pd.read_csv(URL, sep=";", encoding="utf-8")
df["SALAIRE_NET"] = en_nombre(df["SALAIRE_NET"]).replace(999999, np.nan)
df = df[df["SALAIRE_NET"].isna() | (df["SALAIRE_NET"] > 0)].copy()
s  = df["SALAIRE_NET"].dropna()
print(len(df), "lignes |", len(s), "salaires observes")
```

SORTIE RÉELLE

```
7337 lignes | 6812 salaires observes
```

Partie 1 — Décrire la distribution : Q1–Q3

TD05 : La distribution des salaires

Q1–Q3. Les questions

1. Décrire la distribution

Question 1. Calculez les déciles du salaire. Que valent D_1, D_5 et D_9 ?

Question 2. Calculez le **rapport interdécile** D_9/D_1. Interprétez-le en une phrase.

Question 3. Quelle part de la masse salariale totale revient aux 10 % les mieux payés ? Aux 10 % les moins payés ?

Pause recherche : quelle probabilité faut-il donner à `quantile` pour D_1 ? Pour la médiane ?

Q1. Les neuf déciles en une ligne

```
s.quantile([.1, .2, .3, .4, .5,
            .6, .7, .8, .9]).round(0)
```

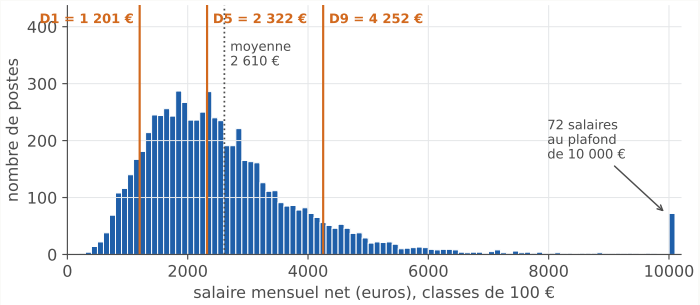
`s.quantile([...])` : un seuil par probabilité de la liste.
`.round(0)` : arrondi à l'euro, **pour l'affichage**.

SORTIE RÉELLE

```
0.1    1201.0
0.2    1523.0
0.3    1791.0
0.4    2040.0
0.5    2322.0
0.6    2596.0
0.7    2939.0
0.8    3419.0
0.9    4252.0
```

$D_1 = \mathbf{1\,201}$ €, $D_5 = \mathbf{2\,322}$ €, $D_9 = \mathbf{4\,252}$ €.

Q1. Interpréter D_1, D_5 et D_9



10 % des postes à salaire observé ont un salaire inférieur à 1 201 € ; la moitié, inférieur à 2 322 € ; 10 % sont au-dessus de 4 252 €.
Moyenne 2 610 € > médiane 2 322 € : distribution **étalée vers la droite**. Le pic à 10 000 € est le plafond.

Q1. Contrôle : valeurs non arrondies et interpolation

```
print(s.quantile([.1, .5, .9]))
```

SORTIE RÉELLE

```
0.1     1201.367
0.5     2322.045
0.9     4251.870
```

D_1 est au rang $1 + 6\,811 \times 0{,}1 = 682{,}1$: entre les 682^{e} et 683^{e} salaires triés, 1 201,36 € et 1 201,43 €.

Avec 6 812 observations, l'interpolation ne déplace les déciles que de quelques centimes. Elle comptait davantage dans l'exemple à dix salaires.

Q2. Le rapport interdécile

```
d1, d9 = s.quantile([.1, .9])
print(d1, d9, round(d9 / d1, 2))
```

SORTIE RÉELLE

```
1201.367 4251.870000000003 3.54
```

`d1, d9 = ...` : la sortie contient deux valeurs ; Python range la première dans `d1`, la seconde dans `d9`. Le rapport est calculé sur les valeurs **non arrondies**, puis arrondi.

Contrôle. $4\,251{,}87 / 1\,201{,}367 = 3{,}539$. Les huit chiffres après la virgule de `d9` sont un artefact de calcul en virgule flottante.

Q2. Interprétation

« Le seuil du neuvième décile représente environ **3,54 fois** celui du premier décile. Il faut au moins 4 252 € pour être parmi les 10 % de postes les mieux payés, contre au plus 1 201 € pour les 10 % les moins payés. »

Q3. Décomposer le calcul de la part du haut

```
print(round(100 * s[s >= d9].sum() / s.sum(), 1), "%")
print(round(100 * s[s <= d1].sum() / s.sum(), 1), "%")
```

Morceau	Rôle
`s >= d9`	Vrai ou Faux pour chaque salaire, est-il au moins égal à D_9 ?
`s[s >= d9]`	Garde seulement les salaires pour lesquels c'est Vrai
`.sum()`	Masse salariale du groupe sélectionné
`/ s.sum()`	Divisée par la masse salariale totale, le dénominateur
`100 *`, `round(..., 1)`	En pourcentage, arrondi à une décimale

Q3. Résultat et contrôle des effectifs sélectionnés

```
print((s >= d9).sum(), (s <= d1).sum(), len(s) / 10)   # effectifs selectionnes
print((s == d9).sum(), (s == d1).sum())                # ex aequo aux seuils
```

SORTIE RÉELLE parts de la masse (Q3)

```
22.4 %
3.6 %
```

SORTIE RÉELLE contrôle des effectifs

```
682  682  681.2
0  0
```

Les 10 % de postes les mieux payés reçoivent **22,4 %** de la masse salariale ; les 10 % les moins payés, **3,6 %**. Rapport des parts : environ 6,2.

Contrôle : 682 observations sélectionnées de chaque côté, pour $6\,812/10 = 681{,}2$. Aucun ex æquo au seuil ici. Avec des ex æquo, >= et <= pourraient sélectionner nettement plus de 10 %.

Partie 2 — L'indice de Gini : Q4–Q6

TD05 : La distribution des salaires

Q4–Q6. Les questions

2. Un seul chiffre : l'indice de Gini

L'indice de Gini résume la distribution : 0 si tout le monde gagne la même chose, 1 si une seule personne perçoit tout. La fonction vous est fournie dans le notebook.

Question 4. Que vaut l'indice de Gini de ce fichier ?

Question 5. Recalculez-le en écartant les salaires au plafond de 10 000 euros. L'indice *baisse*. Les inégalités ont-elles diminué ?

Question 6. Votre indice est-il une estimation juste, une borne haute ou une borne basse de l'inégalité réelle ? Justifiez en une phrase.

Pause recherche : avant d'exécuter, le Gini sans les salaires au plafond sera-t-il plus grand ou plus petit ?

Q4. La fonction fournie, étape par étape

```
def gini(x):
    """Indice de Gini d'une serie de valeurs positives."""
    x = np.sort(np.asarray(x))          # 1. tableau NumPy trie
    n = len(x)                          # 2. nombre d'observations
    return 2 * np.sum(np.arange(1, n + 1) * x) / (n * np.sum(x)) - (n + 1) / n
```

$$G = \frac{2\sum_{i=1}^{n} i\, x_{(i)}}{n\sum_{i=1}^{n} x_{(i)}} - \frac{n+1}{n} \qquad x_{(1)} \leq \cdots \leq x_{(n)}$$

`np.arange(1, n + 1)` donne les rangs $1, 2, \ldots, n$. Chaque salaire trié est multiplié par son rang. Les salaires élevés pèsent davantage.

Q4. Vérifier la formule sur l'exemple à la main

$x = (1\,000\,;\ 1\,000\,;\ 2\,000\,;\ 2\,000\,;\ 4\,000)$, $n = 5$, $\sum x = 10\,000$.
$\sum i\, x_{(i)} = 1 \times 1\,000 + 2 \times 1\,000 + 3 \times 2\,000 + 4 \times 2\,000 + 5 \times 4\,000 = 37\,000$

$$G = \frac{2 \times 37\,000}{5 \times 10\,000} - \frac{6}{5} = 1{,}48 - 1{,}20 = 0{,}28$$

Même résultat que l'aire : 0,28. La formule et le graphique décrivent le même indice.

Conditions d'application : valeurs observées, non négatives, $n > 0$ et somme positive. Avec un `NaN`, la fonction renvoie `nan`. D'où l'usage de `s`, sans manquants.

Q4. Le Gini du fichier

```
print(round(gini(s), 4))
```

SORTIE RÉELLE

```
0.2796
```

Indice de Gini des 6 812 salaires observés : **0,280**.

Q5. Le Gini sans les salaires au plafond

```
print(round(gini(s[s < 10000]), 4))
print((s == 10000).sum(), "salaires au plafond")
```

SORTIE RÉELLE

```
0.263
72 salaires au plafond
```

`s < 10000` retient 6 740 salaires ; `(s == 10000).sum()` compte les 72 salaires exactement égaux au plafond. Aucun salaire ne dépasse 10 000 €.

Pause recherche : l'indice passe de 0,280 à 0,263. Les inégalités ont-elles diminué ?

Q5. Le champ a changé, pas les inégalités

Non. On a retiré les 72 observations les plus élevées. Le second indice décrit un autre ensemble à savoir les 6 740 postes sous le plafond. Sa baisse est mécanique.

Q6. Estimation juste, borne haute ou borne basse ?

Le dictionnaire indique que les salaires supérieurs à 10 000 € sont ramenés à 10 000. Pour ces postes, le vrai salaire est **au moins** égal à 10 000 €.

Borne basse. Abaisser les plus hauts salaires, en conservant leur rang, réduit la part du haut et rapproche la courbe de Lorenz de la diagonale. Le Gini calculé, 0,280, **sous-estime** celui des salaires non plafonnés, toutes choses égales par ailleurs.

Nuance. Ce n'est pas une règle universelle. D_9/D_1 n'est pas modifié par ce plafond, car $D_9 = 4\,252$ €.

Partie 3 — Tableau croisé et khi-deux : Q7–Q8

TD05 : La distribution des salaires

Q7–Q8. Les questions

3. Croiser deux variables qualitatives

Question 7. Croisez `SEXE` et `TEMPS`, en proportions par ligne, après avoir écarté la non-réponse. Que lisez-vous ?

Question 8. Testez l'indépendance des deux variables avec un khi-deux. Que dit la p-value obtenue — et que ne dit-elle pas ?

Pause recherche : quel code de `TEMPS` représente la non-réponse ? Faut-il des effectifs ou des pourcentages pour le khi-deux ?

Q7. Lire les codes dans le dictionnaire

SEXE

Numérique. 1 = homme 2 = femme

Champ. Toutes les lignes. **Non-réponse.** Aucune.

TEMPS

Temps de travail déclaré. Numérique.

1 Temps complet
2 Temps partiel
9 Non-réponse

Champ. Toutes les lignes. **Non-réponse.** codée **9**.

`SEXE` : 1 = homme, 2 = femme, sans non-réponse.

`TEMPS` : 1 = temps complet, 2 = temps partiel, **9 = non-réponse**.

Le code 9 n'a pas le même sens partout. Pour `MOTIF_ABS`, 9 est une situation réelle (chômage partiel). On lit la ligne de la variable.

Source : dictionnaire `cm03-dictionnaire.pdf`, p. 2 et récapitulatif p. 4.

Q7. Vérifier les modalités avant de croiser

```
print(df["SEXE"].value_counts().
    sort_index())
print(df["TEMPS"].value_counts().
    sort_index())
```

`value_counts()` compte chaque modalité ; `sort_index()` les range par code.

SORTIE RÉELLE

```
SEXE
1     3598
2     3739
TEMPS
1     5303
2     1595
9      439
```

Aucun code imprévu. `TEMPS = 9` concerne 439 lignes. On les écarte. Ainsi, on obtient $7\,337 - 439 = 6\,898$.

Q7. Les effectifs, puis les proportions par ligne

```
c = df[df["TEMPS"] != 9]
pd.crosstab(c["SEXE"], c["TEMPS"],
            margins=True)
```

SORTIE RÉELLE effectifs

TEMPS	1	2	All
SEXE			
1	2906	478	3384
2	2397	1117	3514
All	5303	1595	6898

`df["TEMPS"] != 9` : Vrai pour les lignes à temps renseigné.
`pd.crosstab(lignes, colonnes)` compte les observations de chaque case.
`margins=True` ajoute les totaux.

Q7. `normalize="index"` : le dénominateur est la ligne

```
prop = (100 * pd.crosstab(c["SEXE"], c["TEMPS"], normalize="index")).round(1)
print(prop)
print(prop.sum(axis=1))
```

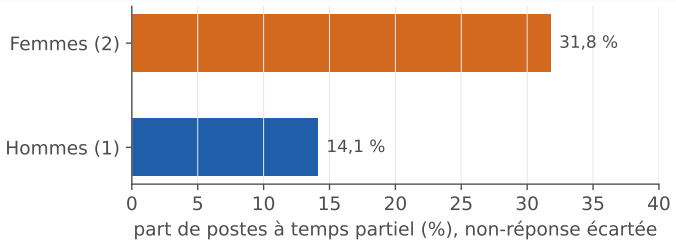
SORTIE RÉELLE

```
TEMPS      1      2
SEXE
1       85.9   14.1
2       68.2   31.8
SEXE
1     100.0
2     100.0
```

`normalize="index"` : chaque case divisée par le total de **sa ligne**.
`100 *` : en pourcentage. `.round(1)` : une décimale.
`.sum(axis=1)` : somme de chaque ligne.

Contrôle. Chaque ligne totalise 100 % : $478/3\,384 = 14{,}1$ % et $1\,117/3\,514 = 31{,}8$ %.

Q7. Que lisez-vous ?



31,8 % des postes occupés par des femmes sont à temps partiel, contre **14,1 %** pour les hommes.
Différence : **17,7 points de pourcentage**. Rapport : **2,25**, plus du double.

Champ : 6 898 lignes à temps renseigné. Sur les seuls salaires observés, les proportions sont presque identiques : 14,4 % et 31,8 %.

Q8. Le test sur les effectifs

```
tableau = pd.crosstab(c["SEXE"], c["TEMPS"])      # effectifs, sans normalize
k = stats.chi2_contingency(tableau)
print(round(k.statistic, 1), k.pvalue, k.dof)
print(k.expected_freq.round(1))
```

SORTIE RÉELLE

```
301.5 1.5361201212017604e-67 1
[[2601.5  782.5]
 [2701.5  812.5]]
```

`statistic` : χ^2 ; `pvalue` ; `dof` : degrés de liberté ; `expected_freq` : effectifs attendus.

Correction de Yates appliquée par défaut (2×2). Sans correction : 302,5. Les effectifs attendus dépassent tous 5.

Q8. Ce que dit la p-value, et ce qu'elle ne dit pas

Ce qu'elle dit
$p \approx 1{,}5 \times 10^{-67}$: si sexe et temps de travail étaient indépendants, un tel écart serait extrêmement improbable. Les données fournissent des éléments très nets **contre l'indépendance**.

Ce qu'elle ne dit pas
Ni la **force** de l'association, ni son **sens**, ni sa **cause**. Une p-value minuscule ne signifie pas « très fortement liées ».

Dans un rapport, on écrit d'abord l'ampleur : 31,8 % contre 14,1 %, soit 17,7 points. Le test vient en appui.

Partie 4 — Écarts de salaire : Q9–Q11

TD05 : La distribution des salaires

Q9–Q11. Les questions

4. Relier les deux

Question 9. Calculez le salaire moyen des hommes et celui des femmes. Quel est l'écart, en euros ?

Question 10. Refaites le calcul **parmi les seuls salariés à temps complet**. De combien l'écart change-t-il ?

Question 11. Quelle part de l'écart initial tient au temps de travail ? L'écart restant permet-il de conclure « à travail égal, salaire inégal » ? Qu'est-ce qui manque pour l'affirmer ?

Pause recherche : quelle colonne mettre dans `groupby` ? Quel code de `TEMPS` désigne le temps complet ?

Q9. Lire `groupby` en trois temps

```
df.groupby("SEXE")["SALAIRE_NET"].mean().round(0)
```

Morceau	Rôle
`df.groupby("SEXE")`	Sépare les lignes en deux groupes : `SEXE = 1` et `SEXE = 2`
`["SALAIRE_NET"]`	Choisit la colonne à résumer
`.mean()`	Moyenne dans chaque groupe ; les `NaN` sont **ignorés**
`.round(0)`	Arrondi à l'euro, pour l'affichage

La moyenne porte donc sur les seuls salaires observés. Il faut afficher ces effectifs : `.agg(["mean", "count"])`.

Q9. Moyennes, effectifs et écart calculé avant l'arrondi

```
m = df.groupby("SEXE")["SALAIRE_NET"].agg(["mean", "count"])
ecart = m.loc[1, "mean"] - m.loc[2, "mean"]        # hommes - femmes
```

SORTIE RÉELLE `print(m.round(1))`

SEXE	mean	count
1	2780.7	3345
2	2444.4	3467

Hommes : **2 781 €** (3 345 salaires) ; femmes : **2 444 €** (3 467).
Écart hommes − femmes : 336,34, soit **336 €** par mois, 12 % de la moyenne des hommes.

La différence des moyennes **arrondies** donne 2 781 − 2 444 = 337 €. On calcule l'écart sur les valeurs exactes, puis on arrondit : 336 €.

Q10. Le même calcul parmi les temps complets

```
plein = df[df["TEMPS"] == 1]                          # 1 = temps complet
mp = plein.groupby("SEXE")["SALAIRE_NET"].agg(["mean", "count"])
ecart_plein = mp.loc[1, "mean"] - mp.loc[2, "mean"]
```

SORTIE RÉELLE `print(mp.round(1))`

SEXE	mean	count
1	2974.4	2695
2	2828.5	2222

Parmi les temps complets : hommes **2 974 €**, femmes **2 829 €**.
Écart : 145,87, soit **146 €**. Moyennes arrondies : $2\,974 - 2\,829 = 145$.

L'écart passe de 336 € à 146 €, il diminue de **190 €**.

Q11. Une réduction descriptive de l'écart

$$100 \times \frac{\Delta_{\text{ensemble}} - \Delta_{\text{temps complet}}}{\Delta_{\text{ensemble}}} = 100 \times \frac{336{,}34 - 145{,}87}{336{,}34} = \mathbf{56{,}6}\ \%$$

Interprétation : « Parmi les seuls postes à temps complet, l'écart de salaire moyen entre hommes et femmes est inférieur de 57 % à l'écart observé sur l'ensemble des postes. »

Q11. « À travail égal, salaire inégal » ?

Il reste **146 €** parmi les temps complets. Cela ne prouve pas « à travail égal, salaire inégal ».

Ce qui peut encore différer	Pourquoi cela compte
Diplôme, âge, ancienneté	Ils sont liés au salaire et peuvent différer entre groupes
Métier, secteur, région	Les grilles salariales varient selon les emplois
Heures exactes	« Temps complet » regroupe des durées différentes

La **régression**, vue dans la suite du cours, compare des salaires à caractéristiques observées égales. Elle n'identifie pas automatiquement une discrimination, des différences non observées peuvent subsister.

Approfondissements facultatifs : Q12–Q14

TD05 : La distribution des salaires

Q12–Q14. Les questions

FACULTATIF

5. Pour ceux qui ont terminé

Question 12. [facultatif] Calculez l'indice de Gini par secteur, avec l'effectif de chaque secteur. Le classement est-il fiable partout ?

Question 13. [facultatif] Tracez la courbe de Lorenz : la part cumulée de la masse salariale en fonction de la part cumulée des salariés, classés du plus faible au plus élevé.

Question 14. [facultatif] Reprenez la question 10 en comparant aussi à diplôme identique. L'écart diminue-t-il encore ?

Pause recherche : dans quel secteur le Gini est-il le plus élevé ? Avec combien de salaires ?

Q12. Un Gini par secteur, avec son effectif

FACULTATIF

```
t = df.groupby("SECTEUR")["SALAIRE_NET"].agg(
        gini=lambda x: gini(x.dropna()), effectif="count").round(3)
t.sort_values("gini", ascending=False)
```

`lambda x: gini(x.dropna())` : petite fonction appliquée au salaire de chaque secteur, sans les manquants.
`effectif="count"` : nombre de salaires **valides**.
`sort_values` : classement décroissant.

SORTIE RÉELLE

SECTEUR	gini	effectif
HZ	0.299	317
OQ	0.288	1925
GZ	0.283	1307
BE	0.280	971
FZ	0.278	463
MN	0.272	861
JZ	0.267	343
RU	0.256	401
AZ	0.240	224

Q12. Le classement est-il fiable partout ?

FACULTATIF

AZ	Agriculture, sylviculture, pêche
BE	Industrie manufacturière, extractive
FZ	Construction
GZ	Commerce, réparation d'automobiles
HZ	Transports et entreposage
JZ	Information et communication
MN	Activités scientifiques, techniques, services administratifs
OQ	Administration publique, enseignement, santé, action sociale
RU	Autres activités de services

Champ. Toutes les lignes. **Non-réponse.** Aucune.

Entre secteurs voisins du classement, les écarts sont souvent de **0,002 à 0,006**. HZ, premier : 317 salaires ; AZ, dernier : 224.

Rééchantillonnage (2 000 tirages) : écart-type du Gini de 0,005 pour OQ ($n = 1\,925$), 0,014 pour HZ et AZ. Les rangs 2 à 7 changent souvent.

Classement fiable seulement aux extrêmes, et avec prudence. Les effectifs évitent de sur-interpréter des différences minimes.

Q13. Construire la courbe de Lorenz

FACULTATIF

```
x = np.sort(s.values)                                    # 1. trier
part_personnes = np.concatenate([[0], np.arange(1, len(x) + 1) / len(x)])
part_masse     = np.concatenate([[0], np.cumsum(x) / x.sum()])
print(part_personnes[0], part_masse[0], part_personnes[-1], part_masse[-1])
```

SORTIE RÉELLE

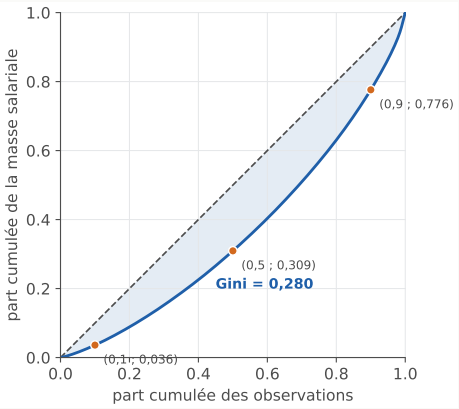
```
0.0 0.0 1.0 0.9999999999999996
```

`np.cumsum(x)` : sommes cumulées. `np.concatenate([[0], ...])` ajoute explicitement l'origine $(0\,;0)$, absente du code du corrigé.

Contrôle. La courbe part de $(0\,;0)$ et arrive en $(1\,;1)$: la dernière valeur vaut 1 aux erreurs d'arrondi près, ce que confirme `np.isclose(part_masse[-1], 1)`.

Q13. Tracer et lire la courbe

FACULTATIF




```
import matplotlib.pyplot as plt
plt.plot(part_personnes, part_masse)
plt.plot([0, 1], [0, 1], "--")
plt.xlabel("part cumulée des postes")
plt.ylabel("part cumulée de la masse")
plt.show()
```

Les 10 % de postes les moins payés reçoivent 3,6 % de la masse ; la moitié, 30,9 % ; les 90 % les moins payés, 77,6 %.

Aire entre diagonale et courbe rapportée à l'aire sous la diagonale : 0,280.

Q14. Temps complet et diplôme identique

FACULTATIF

```
plein2 = plein[plein["DIPLOME"] != 9]                    # 9 = non-reponse
q14 = (plein2.groupby(["DIPLOME", "SEXE"])["SALAIRE_NET"]
              .agg(["mean", "count"])
              .unstack("SEXE"))                            # une colonne par sexe
q14["ecart"] = q14[("mean", 1)] - q14[("mean", 2)]
q14.round(0)
```

`groupby(["DIPLOME", "SEXE"])` : un groupe par couple diplôme × sexe.
`unstack("SEXE")` : place hommes et femmes côte à côte.
L'écart est calculé **avant** l'arrondi.

DIPLOME

Diplôme le plus élevé obtenu. Numérique.

1	Aucun diplôme
2	BEPC, brevet des collèges
3	CAP, BEP
4	Baccalauréat
5	Bac + 2
6	Bac + 3 ou + 4
7	Bac + 5 ou plus

Champ. Toutes les lignes. **Non-réponse.** codée **9**.

Q14. Les écarts par diplôme

FACULTATIF

Diplôme (temps complet)	Moy. H	Moy. F	Écart	Eff. H	Eff. F
1 Aucun diplôme	2 100	2 028	71	340	267
2 BEPC, brevet	2 396	2 143	252	124	111
3 CAP, BEP	2 667	2 428	239	609	465
4 Baccalauréat	2 688	2 587	101	487	368
5 Bac + 2	3 345	3 024	321	344	323
6 Bac + 3 ou + 4	3 845	3 589	255	327	295
7 Bac + 5 ou plus	3 829	3 814	16	359	294

Moyenne des sept écarts pondérée par les effectifs : **175,8 €**, contre **148,9 €** sans tenir compte du diplôme sur le même champ (diplôme renseigné).

Q14. L'écart diminue-t-il encore ? Vérifions

FACULTATIF

Non, pas en moyenne. À diplôme identique, l'écart moyen est d'environ 176 €, au-dessus des 149 € observés parmi les temps complets à diplôme renseigné.

Pourquoi ? Parmi les temps complets, les femmes de ce fichier sont un peu plus souvent diplômées du supérieur (bac + 2 : 15,2 % contre 13,3 %). Comparer à diplôme égal retire cet avantage.

Limites. Écarts très variables, de 16 à 321 €, sur des groupes parfois petits (111 femmes en BEPC). 204 salaires perdus avec le diplôme non renseigné : 4 917 → 4 713.

Annexe. Ressources et liens vérifiés le 14/10/2026

Ressource	État
Page du cours	Accessible ; ligne TD 5 avec énoncé et notebook
Énoncé TD05	Accessible ; identique à la copie locale
Notebook sur GitHub	Accessible
Téléchargement direct	Accessible ; identique à la copie locale
Ouverture dans Colab	Notebook affiché sans connexion
Données	Accessible ; identique à la copie locale
Dictionnaire des variables	Accessible ; identique à la copie locale