



Rapport technique d'évaluation French Industry

Promotion: DPM_avril 23_continu

Auteur: Lucie Duhin

Contexte

L'INSEE est l'institut officiel français qui collecte des données de tous types sur le territoire français. Elles peuvent être démographiques (Naissances, Décès, Densité de la population...), économiques (Salaires, Entreprises par activité / taille...) et plus encore. Ces données peuvent être d'une grande aide pour observer et mesurer les inégalités au sein de la population française.

C'est dans ce contexte et avec ces données concernant la population, les entreprises et les salaires horaires net que nous allons étudier la répartition des entreprises et la comparaison des salaires sur le territoire Français dans le projet technique French Industry.

Objectifs

Dans une première étape, l'objectif est de comparer les données à l'échelle nationale :

- Entreprises en fonction de leur localisation, de leur taille.
- Population en fonction du salaire et de la localisation.

Puis, la deuxième étape du projet est de faire une analyse sur une ville au choix

Audit des données

Pour cet exercice, nous avons à disposition un dataset visible et téléchargeable via le lien suivant:

- https://assets-datascientest.s3.eu-west-1.amazonaws.com/notebooks/power_bi/power_bi_datasets_projet/french_industry.zip.

Les données disponibles dans le dossier zip sont séparées en 4 fichiers csv qui ont été étudiés séparément dans une première phase d'exploration

Le tableau suivant donne un aperçu du nombre de lignes, des variables pour chaque fichier. l'intégralité du rapport d'exploration des données étant accessible via ce lien:


 [DPM_avr23_continu_French_Industry_Rapport exploration des données](#)

Tableau des sources de données

nom du fichier	nom du dataframe dans le notebook	nombre d'attributs (= nombre de colonnes)	nombre d'enregistrement (nombre de lignes)
base_etablissement_par_tranche_effectif.csv	df_entreprise	14	36681
name_geographic_information.csv	df_ville	14	36840

population.csv	df_population	7	8 536 584
net_salary_per_town_categories.csv	df_salaire	26	5136

L'audit des données a été réalisé dans un jupyter notebook partagé avec ce rapport.

1) Présentation de df_entreprise

Dans un premier temps, il s'agit de lire le csv puis d'afficher les informations relatives au df_entreprise.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 36681 entries, 0 to 36680
Data columns (total 14 columns):
#   Column      Non-Null Count  Dtype
---  -
0   CODGEO      36681 non-null  object
1   LIBGEO      36681 non-null  object
2   REG         36681 non-null  int64
3   DEP         36681 non-null  object
4   E14TST      36681 non-null  int64
5   E14TS0ND    36681 non-null  int64
6   E14TS1      36681 non-null  int64
7   E14TS6      36681 non-null  int64
8   E14TS10     36681 non-null  int64
9   E14TS20     36681 non-null  int64
10  E14TS50     36681 non-null  int64
11  E14TS100    36681 non-null  int64
12  E14TS200    36681 non-null  int64
13  E14TS500    36681 non-null  int64
dtypes: int64(11), object(3)
memory usage: 3.9+ MB
```

Nous constatons une base de données contenant 14 colonnes totalement remplies (sans valeur nulle) et avec des types de variables différents (object et int);

Les 4 premières colonnes de ce df_entreprise représentent des codes de référence (pour CODGEO, REG) ou des intitulés de commune (LIBGEO) ou numéro de département (DEP). La variable REG représentant un code région a été transformée en variable de type object. De plus, une vérification a été faite sur l'éventuelle présence de doublon.

Les colonnes 4 à 13 sont de type int et représentent le nombre d'entreprises sur la commune en fonction du nombre d'employés (0,1,6,10,20,50,100,200,500). La colonne 4

'E14TST' représente le total de la ligne (c'est-à-dire le nombre total d'entreprise sur la commune toute catégorie du nombre d'employés confondues).

2) Présentation de df_ville

Tout comme le précédent dataframe, une première lecture et un affichage des informations est nécessaires pour comprendre les données de df_ville.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 36840 entries, 0 to 36839
Data columns (total 14 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   EU_circo                             36840 non-null  object
1   code_région                          36840 non-null  int64
2   nom_région                           36840 non-null  object
3   chef.lieu_région                     36840 non-null  object
4   numéro_département                   36840 non-null  object
5   nom_département                       36840 non-null  object
6   préfecture                           36840 non-null  object
7   numéro_circonscription               36840 non-null  int64
8   nom_commune                          36840 non-null  object
9   codes_postaux                        36840 non-null  object
10  code_insee                           36840 non-null  int64
11  latitude                             33911 non-null  float64
12  longitude                             33999 non-null  object
13  éloignement                           33878 non-null  float64
dtypes: float64(2), int64(3), object(9)
memory usage: 3.9+ MB
```

Ce dataframe est constitué de 14 colonnes de 36840 lignes; nous constatons que les variables latitude, longitude et éloignement contiennent des valeurs nulles que nous remplaçons par -100 (valeur arbitraire pour différencier des chiffres du tableau).

La majeure partie des variables sont de type object telle que le code_region, les intitulés de région(nom_region), de chef lieu, département (nom_département), commune, code_insee...

Code_region, numéro_circonscription et code_insee ont été changé en variable de type object.

Les seules variables autre que object, de type int ou float sont les coordonnées de latitude et longitude (a été changée en float) ainsi que la colonne éloignement.

Cette database a pour but de localiser les différentes communes présentes sur le territoire français sur différentes échelles:

- EU circo: représente la zone géographique (Est, Nord-ouest, Ouest, Sud Est, Centre, Ile-de France, Outre-Mer),
- nom_région: représente la région dans laquelle se situe la commune (28 au total)
- nom_département: représente le département de la commune
- numéro_circonscription: fait un zoning des communes dans le département
- nom_commune: plus petite échelle du df_ville

Cette base de données pourra éventuellement permettre de faire le lien entre les différentes databases de l'exercice.

3) Présentation de df_population

Ce dataframe contient très peu de variables (7 colonnes au total) mais un nombre de lignes gigantesques (plus de 8 millions). Il s'agit des données démographiques du territoire Français, c'est-à-dire la répartition de la population à travers une certaine classe d'âge (AGEQ80_17), le genre ('SEXE'), et le mode de cohabitation dans le foyer ('MOCO'). Pour chacune de ces variables un nombre de personnes est indiqué dans la colonne NB.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8536584 entries, 0 to 8536583
Data columns (total 7 columns):
#   Column      Dtype
---  -
0   NIVGEO      object
1   CODGEO      object
2   LIBGEO      object
3   MOCO        int64
4   AGEQ80_17   int64
5   SEXE        int64
6   NB          int64
dtypes: int64(4), object(3)
memory usage: 455.9+ MB
```

Cette database ne contient pas de valeur nulle, elle est constituée des colonnes 0 à 5 de type catégorielle (donc à transformer en object) et de la dernière colonne (NB) de type int qui permet le calcul dans les différentes classes.

La colonne CODGEO n'est constituée que d'une valeur 'COM', cette colonne pourrait être supprimée si besoin. Les colonnes CODGEO et LIBGEO sont reprises dans le df_entreprise et pourront permettre de faire le lien entre la population et les entreprises d'une même zone géographique.

4) Présentation de df_salaire

Df_salaire est une base de données constituée de variables de type float en majeure partie car représente le salaire horaire net moyen pour plusieurs catégorie de personnes (soit par âge, par genre ou par type d'emploi).

```

<class 'pandas.core.frame.DataFrame'>
Int64Index: 6044 entries, 0 to 6043
Data columns (total 27 columns):
#   Column              Non-Null Count  Dtype
---  -
0   CODGEO              6044 non-null   object
1   LIBGEO              6044 non-null   object
2   SNHM14              6044 non-null   float64
3   SNHMC14             6044 non-null   float64
4   SNHMP14             6044 non-null   float64
5   SNHME14             6044 non-null   float64
6   SNHMO14             6044 non-null   float64
7   SNHMF14             6044 non-null   float64
8   SNHMF14             6044 non-null   float64
9   SNHMF14             6044 non-null   float64
10  SNHMF14             6044 non-null   float64
11  SNHMF14             6044 non-null   float64
12  SNHMF14             6044 non-null   float64
13  SNHMF14             6044 non-null   float64
14  SNHMF14             6044 non-null   float64
15  SNHMF14             6044 non-null   float64
16  SNHMF14             6044 non-null   float64
17  SNHMF14             6044 non-null   float64
18  SNHMF14             6044 non-null   float64
19  SNHMF14             6044 non-null   float64
20  SNHMF14             6044 non-null   float64
21  SNHMF14             6044 non-null   float64
22  SNHMF14             6044 non-null   float64
23  SNHMF14             6044 non-null   float64
24  SNHMF14             6044 non-null   float64
25  SNHMF14             6044 non-null   float64
26  DEP                 6044 non-null   object
dtypes: float64(24), object(3)
memory usage: 1.3+ MB

```

Dans cette base de données, nous avons également les colonnes CODGEO et LIBGEO qui permettent de faire le lien avec les autres df, nous avons choisi d'ajouter la colonne DEP afin de faire une analyse par département. Ce dataframe ne contient pas de valeurs nulle ni de doublon.

Visualisations et Statistiques

L'approche suivie dans ce projet, a été de réaliser des visualisations pour chaque dataframe (excepté le df_ville qui est une base de lien) puis de tenter de faire des croisements de données entre chaque database.

1) Visualisations sur df_entreprise

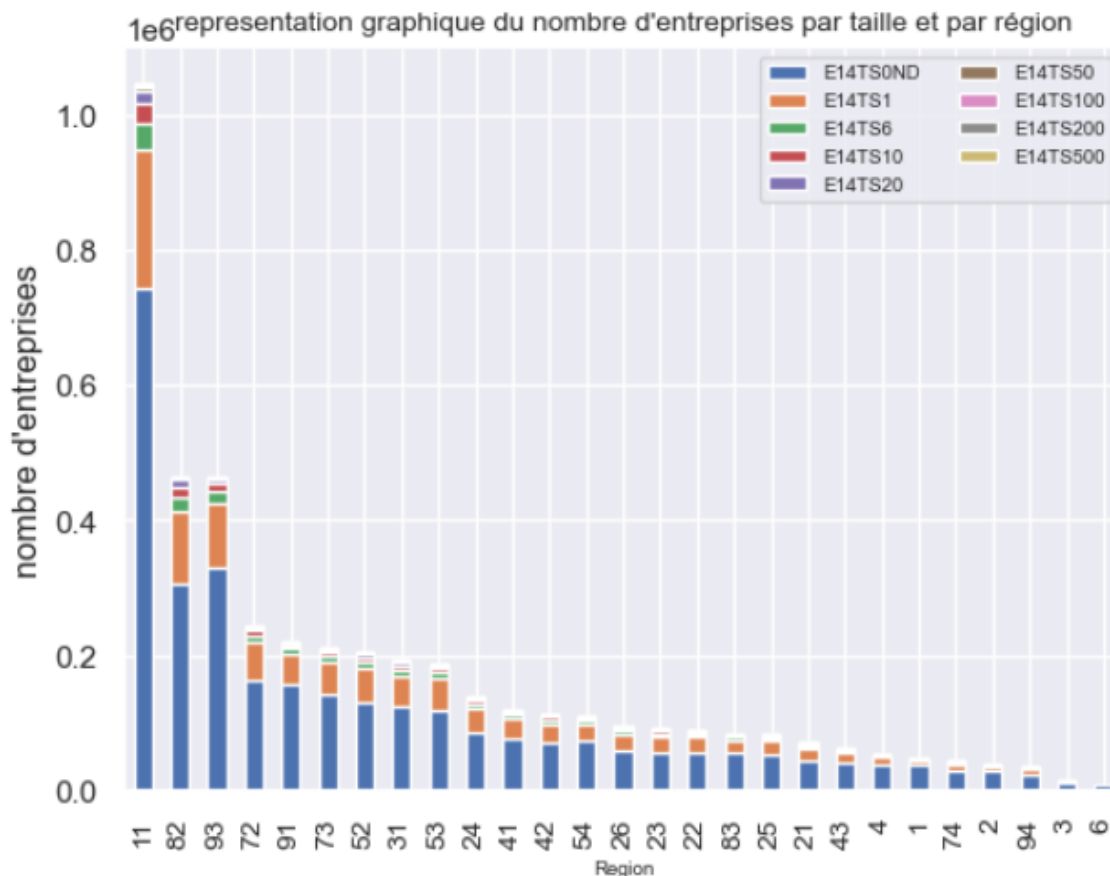
Une première analyse statistique permet de visualiser très rapidement les disparités dans la localisation des entreprises:

	E14TST	E14TS0ND	E14TS1	E14TS6	E14TS10	E14TS20	E14TS50	E14TS100	E14TS200	E14TS500
count	36681.000000	36681.000000	36681.000000	36681.000000	36681.000000	36681.000000	36681.000000	36681.000000	36681.000000	36681.000000
mean	123.456067	83.555301	27.291486	5.220550	3.800333	2.296448	0.738339	0.332434	0.172760	0.048417
std	2353.384846	1729.874812	432.062116	83.685519	60.961216	32.597382	9.882131	4.850211	2.783668	1.091031
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	8.000000	6.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	19.000000	14.000000	3.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
75%	54.000000	39.000000	11.000000	2.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000
max	427385.000000	316603.000000	76368.000000	14836.000000	10829.000000	5643.000000	1658.000000	812.000000	456.000000	180.000000

Nous constatons que les moyennes sont très largement éloignées des écart_type pour l'ensemble des catégories excepté les grandes entreprises.

le 3ème quartile est minime comparé aux données max ce qui montre des villes très fortement “industrialisées” et d'autres au contraire sont désertiques.

La première visualisation consiste à faire le graphique du nombre d'entreprise par catégorie de taille et par région. Pour cela, nous avons utilisé la fonction df.plot en rangeant au préalable les données de la colonne E14TST (total entreprise) par ordre décroissant.

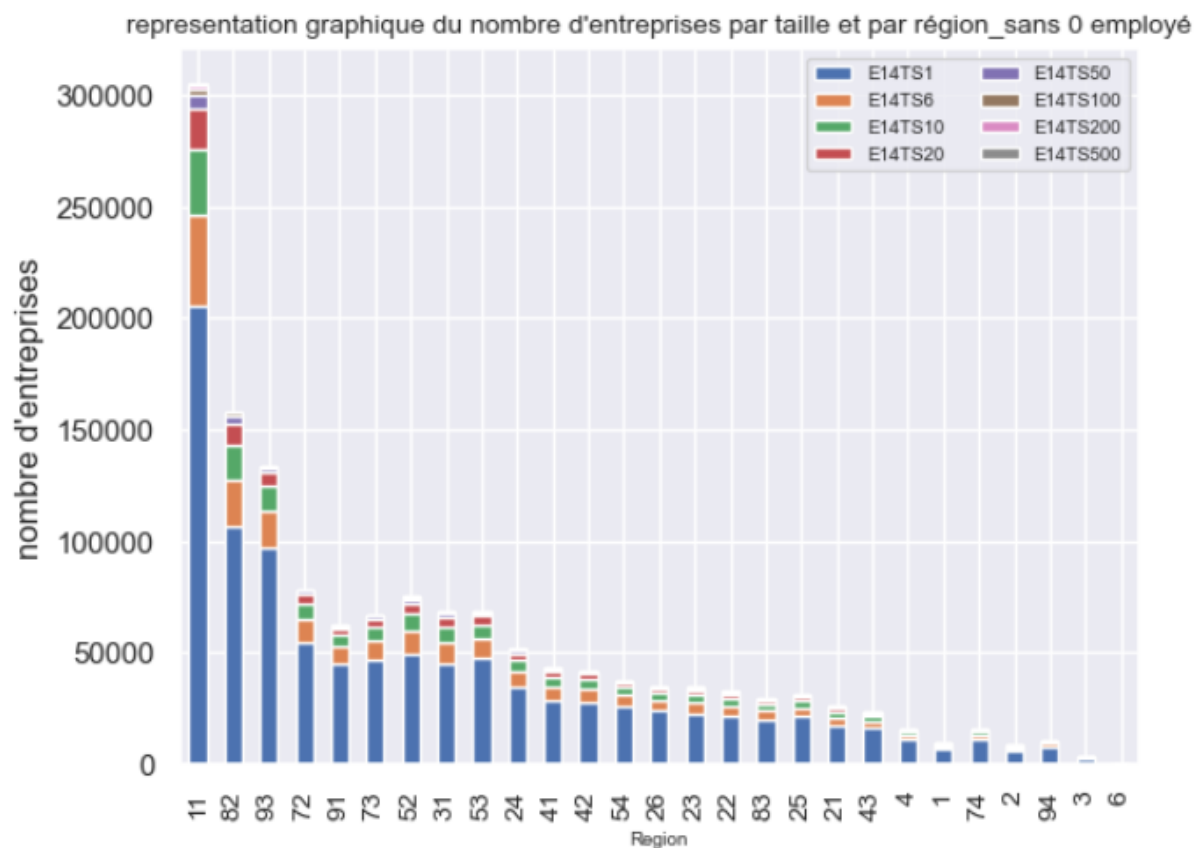


Nous constatons 2 points importants:

Le premier est que nous observons une catégorie d'entreprise qui est sur-représentée sur l'ensemble des catégories, il s'agit des entreprises à 0 employé ou nombre indéterminé.

Le deuxième point est qu'une région se détache grandement des autres, il s'agit de la région '11', nous tenterons d'en savoir plus sur cette région.

Nous avons retiré la catégorie 0/ND employé pour voir si cela avait une influence sur l'ordre des régions:



Pas de changement, les régions à forte présence d'entreprise sont toujours les mêmes (11,82,93,72,91), les régions pauvres en entreprise restent les 6 et 3.

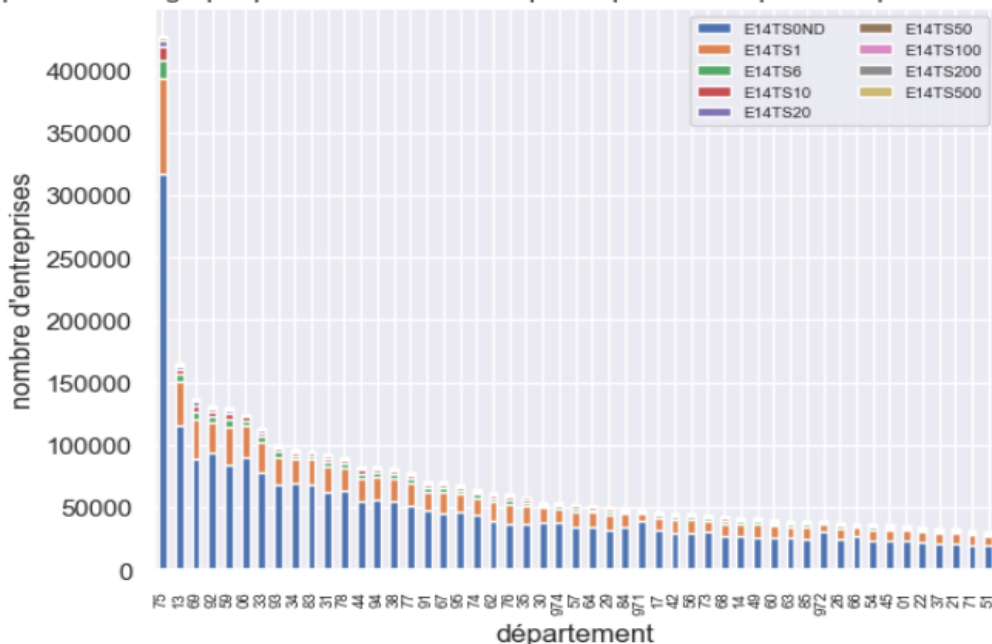
Nous constatons également qu'il existe très peu d'entreprises au-delà de 50 employés sur le territoire.

Au vu de ses graphiques, nous nous intéressons à ses régions extrêmes et tentons de comprendre quelles sont les zones géographiques du territoire concerné.

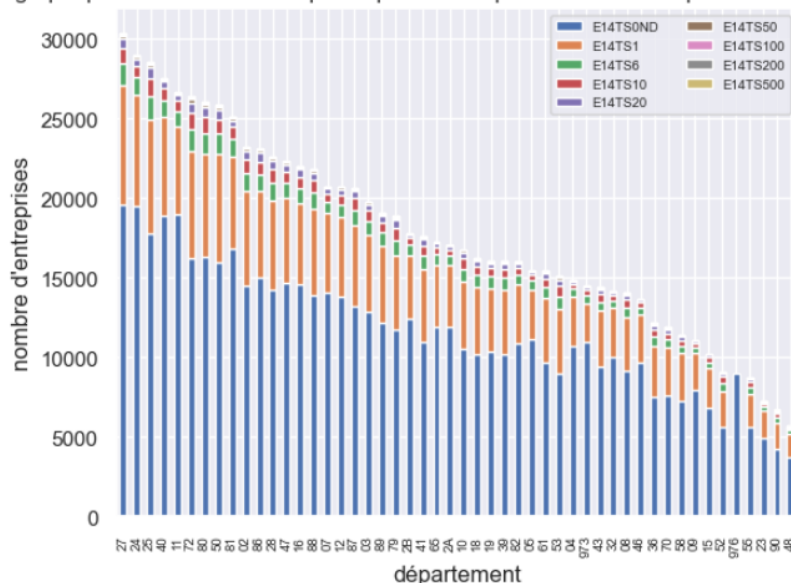
En filtrant le df_entreprise sur le top 5 des régions, nous allons faire apparaître les départements de cette région ainsi que le nombre total d'entreprises par département:

Région 11	Région 82	Région 93	Région 72	Région 91
E14TST	E14TST	E14TST	E14TST	E14TST
DEP	DEP	DEP	DEP	DEP
75 427385	01 35464	04 14806	24 28946	11 26636
77 77591	07 20741	05 15444	33 112887	30 53583
78 90302	26 37237	06 124198	40 27552	34 95799
91 70468	38 81610	13 164883	47 22298	48 5681
92 131528	42 45573	83 95074	64 50907	66 37019
93 100131	69 136867	84 48567		
94 82894	73 43311			
95 67953	74 64034			

En réalisant les graphiques à l'échelle départemental voici ce que nous obtenons:
 représentation graphique du nombre d'entreprises par taille et pour le top 51 des départements



représentation graphique du nombre d'entreprises par taille et pour le 51ème département au classement et au-delà



Nous observons un pic du nombre d'entreprises dans le département 75 avec plus de 400 000 entreprises réparties disproportionnellement (surreprésentation des entreprises à 0 ou ND).

Le top 5 des départements riches en entreprise toutes catégories de nombre d'employés confondus est 75,13,69,92,59.

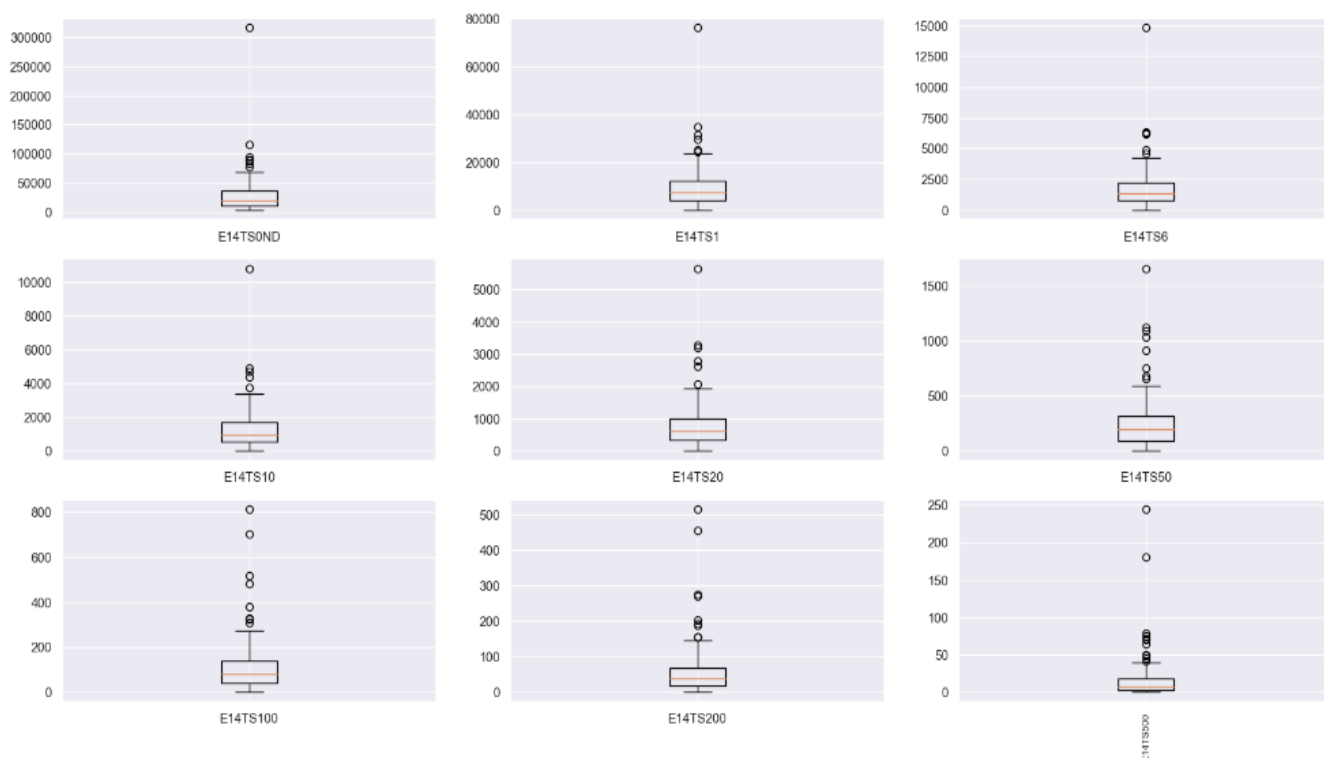
A l'exception du département 59, ces 4 autres départements listés font partie du top 5 région.

L'absence du département 59 dans le top 5 région montre une inégalité de la répartition des entreprises dans la région. Nous constatons cet effet aussi malgré tout sur la région 82 (avec un département très fort, 69).

Quant aux départements 976, 56,23,90 et 48, ceux-ci constituent la queue de peloton du nombre d'entreprises dans leur département. Au même titre que les top régions, les flop régions sont aussi dépendantes de la répartition des entreprises dans leur département.

Un second type de représentation permet de faire voir la répartition statistique du nombre d'entreprise par taille d'employés toutes régions confondues.

Il s'agit des diagrammes de type boîte à moustache:



Nous pouvons observer d'une part que les échelles sont différentes entre les boîtes à moustaches, que pour chaque classe du nombre d'employés il existe des valeurs en dehors des quantiles et donc des départements extrêmement riches en nombre d'entreprise comparé à leur classe de nombre d'employés. Nous observons logiquement grâce à l'échelle que plus le nombre d'employés augmente plus le nombre d'entreprise dans cette classe diminue avec un extrême situé à 244 entreprises de 500+ employés dans le 92 (cf dataframe ci-dessous).

	DEP	E14TST	E14TS0ND	E14TS1	E14TS6	E14TS10	E14TS20	E14TS50	E14TS100	E14TS200	E14TS500
75	75	427385	316603	76368	14836	10829	5643	1658	812	456	180
12	13	164883	115300	34729	6322	4388	2617	915	379	188	45
69	69	136867	89068	31460	6252	4895	3279	1092	482	275	64
92	92	131528	93845	23646	4884	3784	2785	1121	704	515	244
59	59	129819	84394	29479	6170	4683	3197	1033	516	272	75
5	06	124198	90148	25224	3967	2772	1423	397	172	71	24
33	33	112887	77462	24314	4561	3392	2060	680	260	119	39
93	93	100131	67911	22200	4265	2918	1667	559	329	204	78
34	34	95799	69339	19049	3251	2255	1298	379	142	69	17
83	83	95074	68405	20263	3010	1968	968	315	93	45	7

Conclusion intermédiaire sur df_entreprise:

ce dataframe nous montre une très forte disparité quant à la situation géographique des entreprises. En effet, le département 75 est fortement représenté avec ses 400 000 entreprises. l'écart entre le département 75 et les autres est très significatif et il serait intéressant de faire une analyse sur la population de ce département.

Nous avons constaté au travers des graphiques et des filtrations sur le dataframe que les entreprises ne sont pas réparties de manière équilibrée à l'intérieur des régions mais qu'il peut très bien y avoir un département très fort et d'autres moins représentés pour une même zone géographique.

2) Visualisations sur df_population

Ce dataframe est une base regroupant l'ensemble des données de la population sur le territoire, en voici un aperçu

	NIVGEO	CODGEO	LIBGEO	MOCO	AGEQ80_17	SEXE	NB
0	COM	1001	L'Abergement-Clémenciat	11	0	1	15
1	COM	1001	L'Abergement-Clémenciat	11	0	2	15
2	COM	1001	L'Abergement-Clémenciat	11	5	1	20
3	COM	1001	L'Abergement-Clémenciat	11	5	2	20
4	COM	1001	L'Abergement-Clémenciat	11	10	1	20

NB représente le nombre de personnes de chaque catégorie (MOCO correspond au mode de cohabitation, en couple avec/sans enfant; enfant avec 1 ou 2 parents, personnes seules...; SEXE 1 pour homme, 2 pour femme et AGEQ80-17 correspond aux tranches d'âge de 5 ans).

Sur cette base de données, nous avons filtré les valeurs de NB et affiché d'une part le tableau par ordre décroissant (qui montrera le top NB et ses informations), puis par ordre croissant qui affichera les zones géographiques (communes) avec des catégories sous ou non représentées

	NIVGEO	CODGEO	LIBGEO	MOCO	AGEQ80_17	SEXE	NB	DEP
9319842	COM	75056	Paris	11	0	1	48873	75
9319843	COM	75056	Paris	11	0	2	46883	75
9320079	COM	75056	Paris	32	80	2	46700	75
9319844	COM	75056	Paris	11	5	1	40223	75
9320057	COM	75056	Paris	32	25	2	40147	75
9319845	COM	75056	Paris	11	5	2	38818	75
9319959	COM	75056	Paris	22	35	2	37909	75
9319921	COM	75056	Paris	21	25	2	37383	75
9319961	COM	75056	Paris	22	40	2	36125	75
9319846	COM	75056	Paris	11	10	1	35769	75

Cette représentation du top 10 du nombre de population permet de montrer que la population de Paris est la plus dense du territoire; la catégorie enfant(sans distinction de genre) vivant avec 2 parents représente le maximum de personnes ce qui peut également faire le lien avec le poids du nombre d'entreprise dans ce département. Puis vient la catégorie femmes de 80+ ans vivant seule.

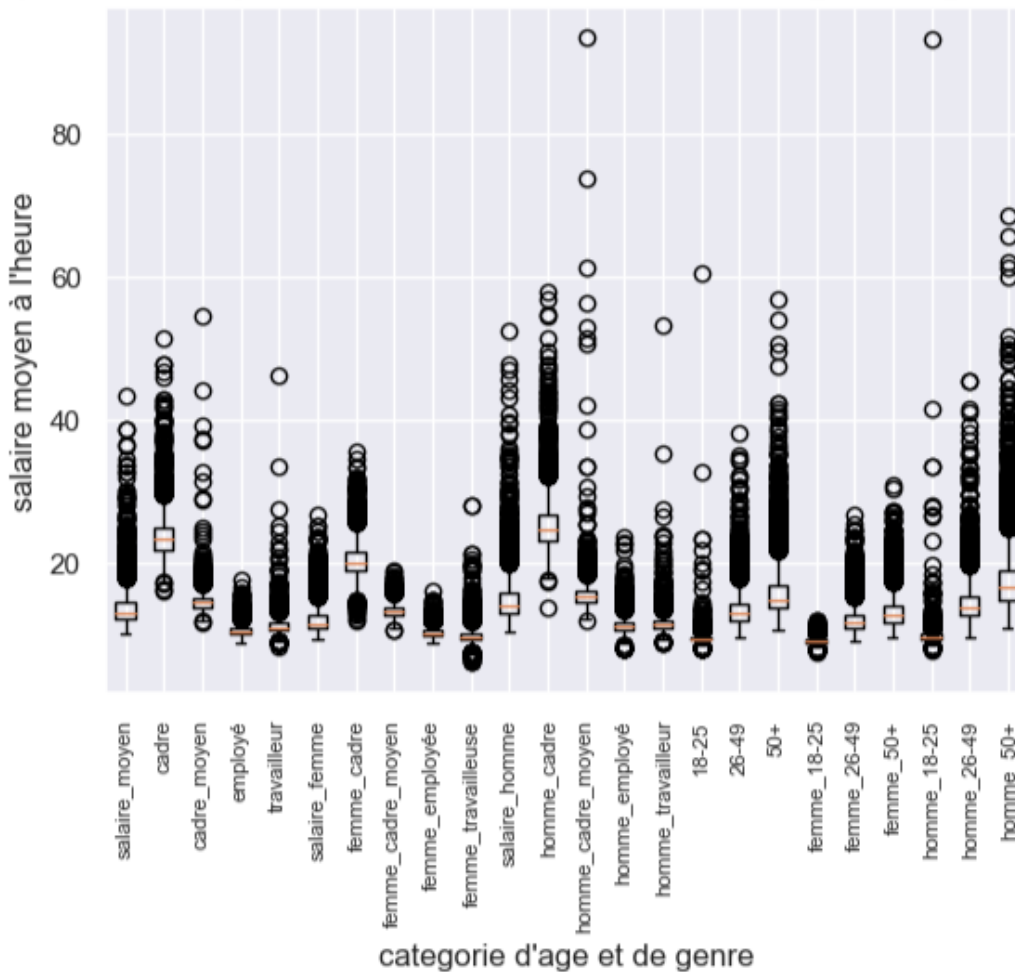
	NIVGEO	CODGEO	LIBGEO	MOCO	AGEQ80_17	SEXE	NB	DEP
4541991	COM	2B194	Ortale	32	80	2	1	2B
4559384	COM	2B302	San-Giovanni-di-Moriani	11	45	1	1	2B
4505710	COM	2A271	Sarrola-Carcopino	22	75	1	1	2A
4490542	COM	2A070	Casaglione	31	65	1	1	2A
4553452	COM	2B264	Rusio	12	5	1	1	2B
4490530	COM	2A070	Casaglione	31	35	1	1	2A
5819872	COM	40046	Biscarrosse	12	60	1	1	40
4548644	COM	2B234	Piobetta	32	55	1	1	2B
10503252	COM	95176	Cormeilles-en-Parisis	21	15	1	1	95
1295820	COM	7019	Aubenas	23	30	1	1	07
4535996	COM	2B164	Monacia-d'Orezza	31	55	1	1	2B
4513435	COM	2B005	Alando	11	5	2	1	2B
4513434	COM	2B005	Alando	11	5	1	1	2B
4505383	COM	2A270	Sari-d'Orcino	12	20	2	1	2A
10476344	COM	93027	La Courneuve	12	65	1	1	93
4513433	COM	2B005	Alando	11	0	2	1	2B
4513585	COM	2B005	Alando	23	40	2	1	2B
4559371	COM	2B302	San-Giovanni-di-Moriani	11	10	2	1	2B
4563163	COM	2B320	Tallone	32	55	2	1	2B
4527054	COM	2B102	Crocicchia	12	55	1	1	2B

Cette représentation du bottom de population nous montre qu'il y a un gap dans les départements 2A et 2B sur une grande partie des classes d'âge.

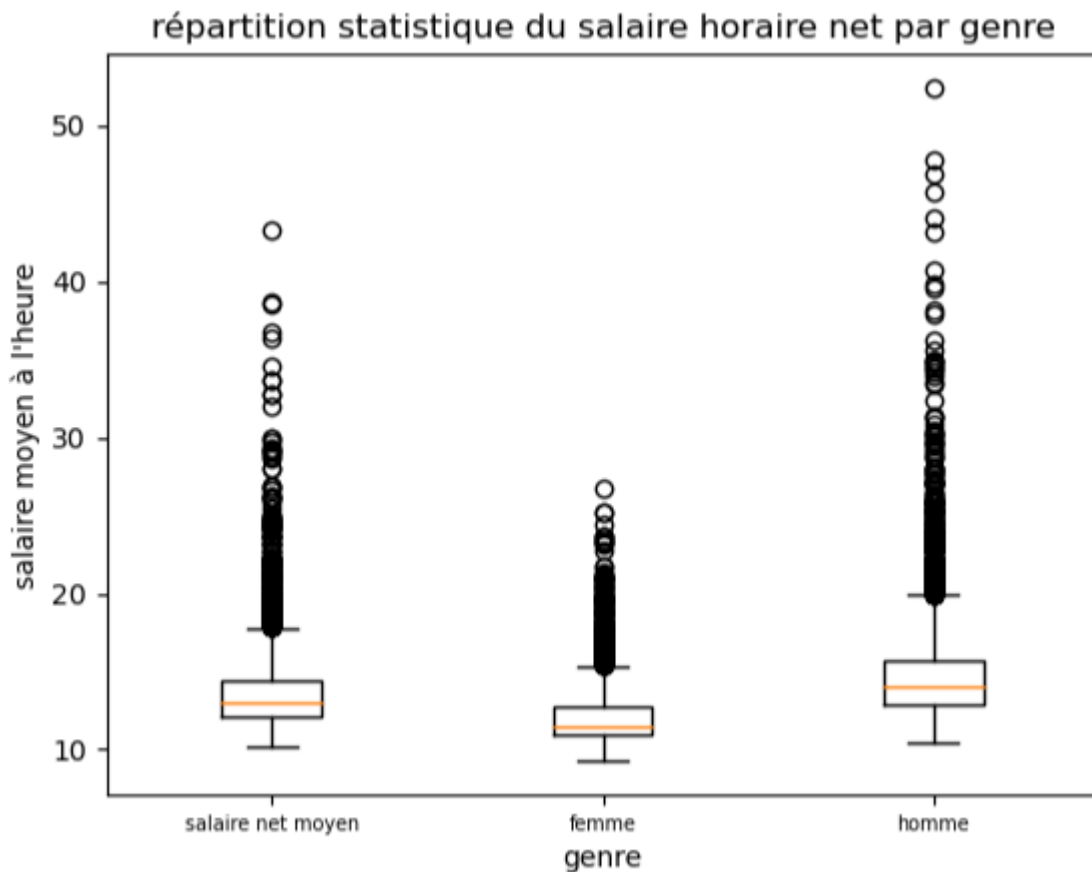
3) Visualisations sur df_salaire

Dans cette table est représenté le salaire moyen net horaire pour certaine classe (d'âge, de genre ou de type d'emploi). En voici une répartition sous forme de boîte à moustache pour l'ensemble des catégories

répartition statistique du salaire horaire en fonction des categories d'age et de genre



Nous constatons sans surprise que le salaire horaire net moyen est plus élevé pour les cadres. Nous constatons également que les valeurs en dehors des quartiles sont moins nombreuses pour les employés, alors que l'on peut constater de fortes inégalités dans les reste des catégories.



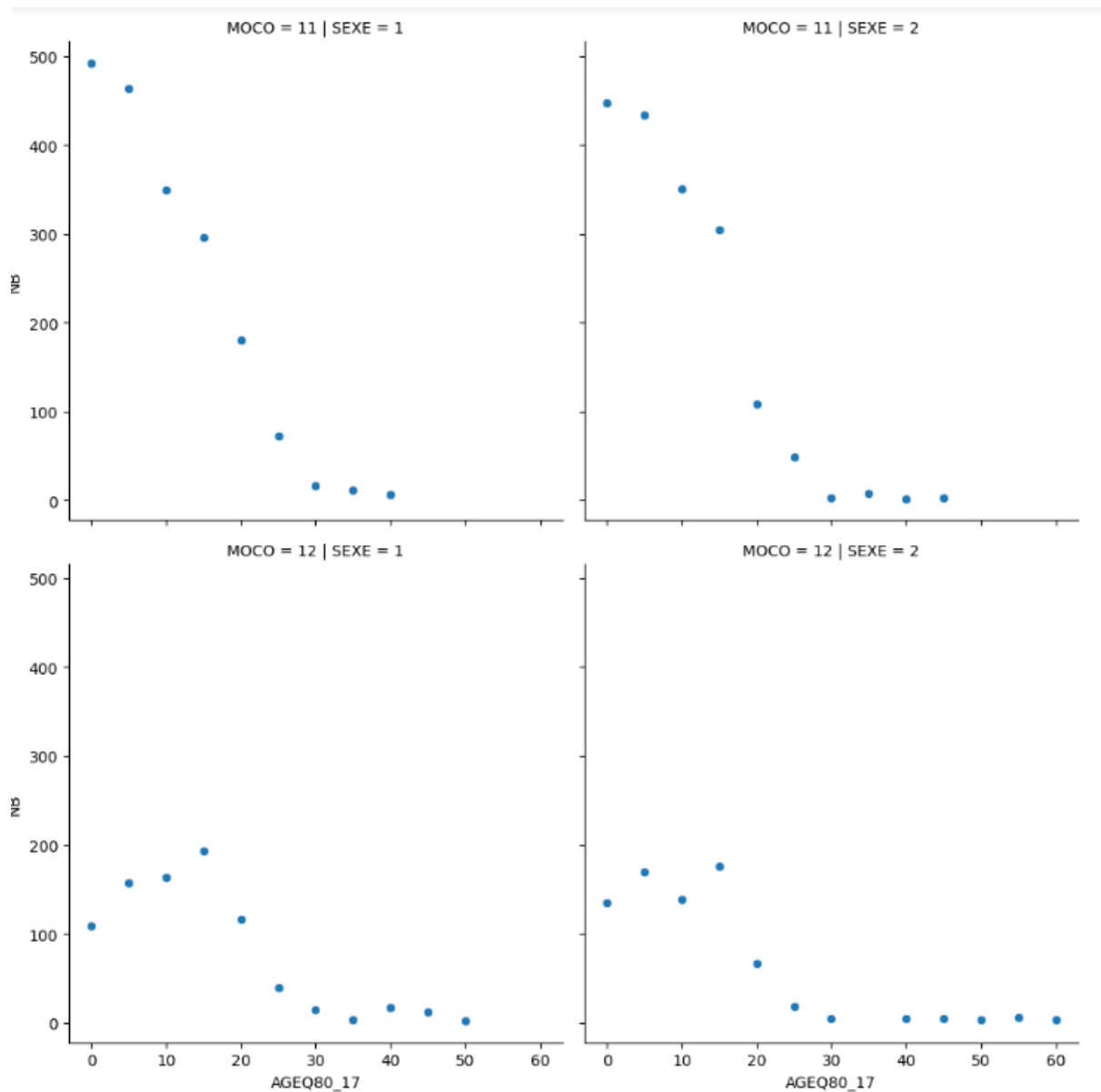
Nous constatons un salaire net moyen horaire légèrement supérieur chez les hommes, un salaire moyen horaire net plus bas que le salaire moyen horaire net. Les inégalités sont plus nombreuses et les extrêmes plus élevés pour les hommes que pour les femmes. D'autres visualisations par catégories d'âge, de genre ou d'emploi sont disponibles dans le notebook.

Voyons si toutes les observations faites à travers l'ensemble des tables se vérifient également sur une ville choisie parmi la France métropolitaine.

Focus sur une ville

D'après les informations récupérées dans les différentes tables, la ville choisie pour ce focus se situe dans la zone Nord-Ouest du territoire, en Haute-Normandie dans le département 27. Cette ville est rattachée à la préfecture d'Evreux et se nomme Val-de-Reuil. Il s'agit d'une commune de 12135 habitants, répartis de la façon suivante:

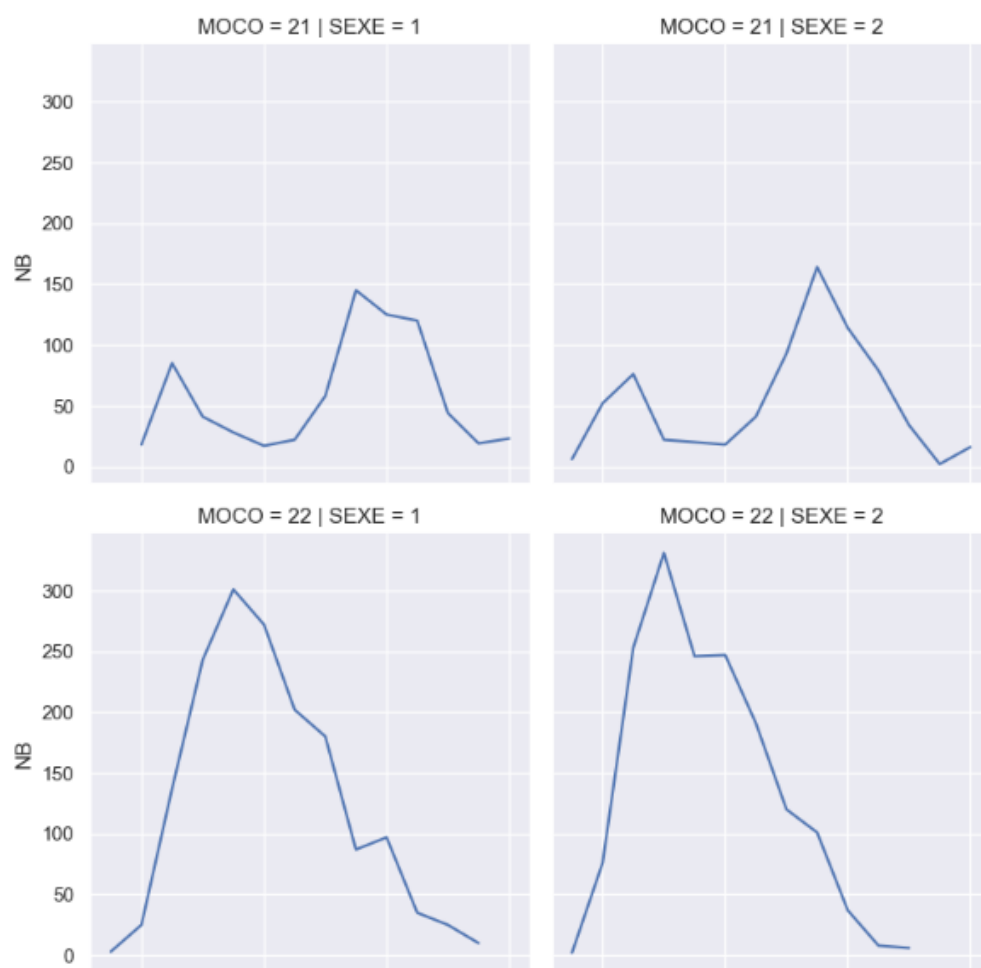
Concernant les enfants (MOCO 11 et 12), d'après la visualisation ci-dessous, il n'y a pas d'influence du genre. Les enfants vivants avec 2 parents sont 1.5 à 3 fois plus nombreux jusqu'à l'âge de 15 ans que les familles d'enfants vivant avec 1 parent. Nous pouvons observer que les femmes restent plus longtemps avec leurs parents (catégorie d'âge supérieur à 50 ans)

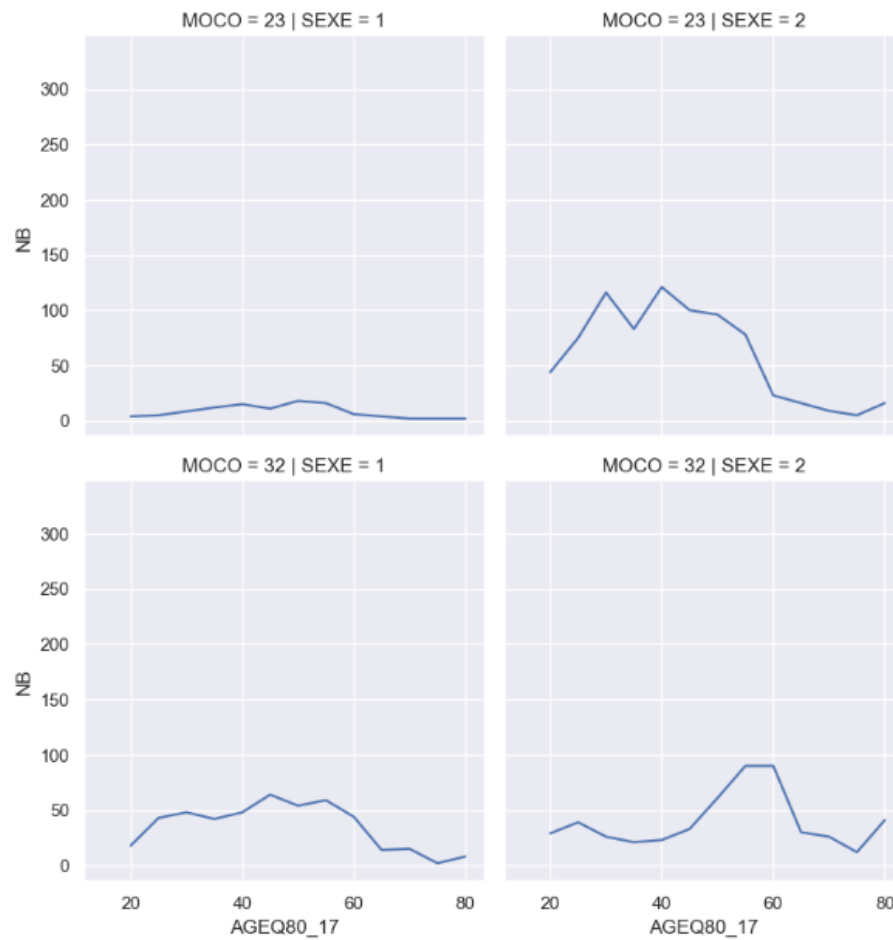


Concernant la population adulte (MOCO 21, 22, 23, 32), Nous ne constatons pas de grande différence entre les genres excepté pour la classe MOCO 23 (adult vivant seul avec un enfant) où la classe féminine est davantage présente en nombre tout au long des classes d'âge de 20 à 60 ans.

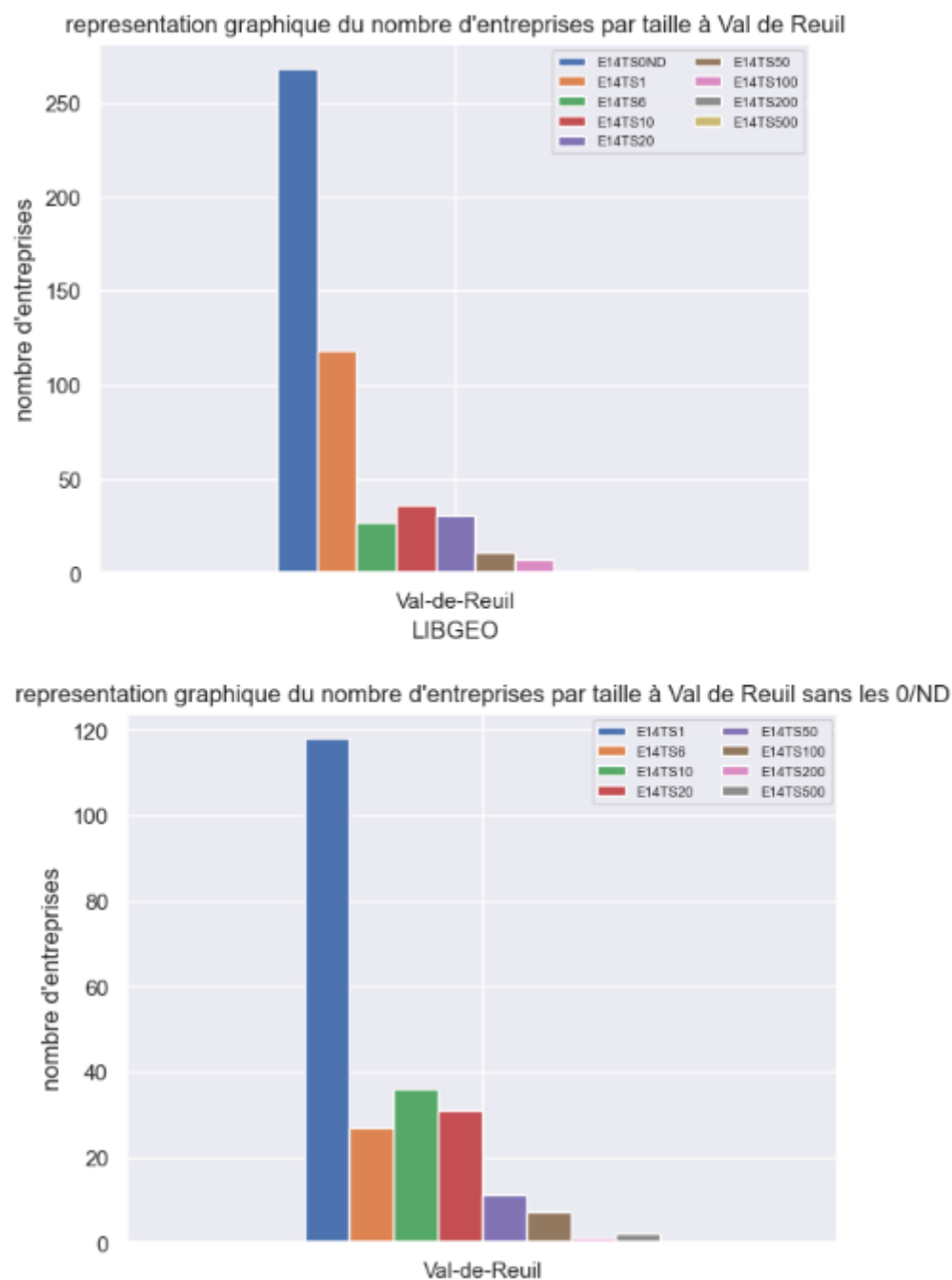
Nous observons que la classe 22 (couple avec enfant) équivaut au double de la classe 21 (couple sans enfant) sans distinction de genre.

Concernant la classe 32 (personnes vivant seules), nous observons qu'il y a légèrement plus de femmes de +de 50 ans vivant seules que d'hommes.

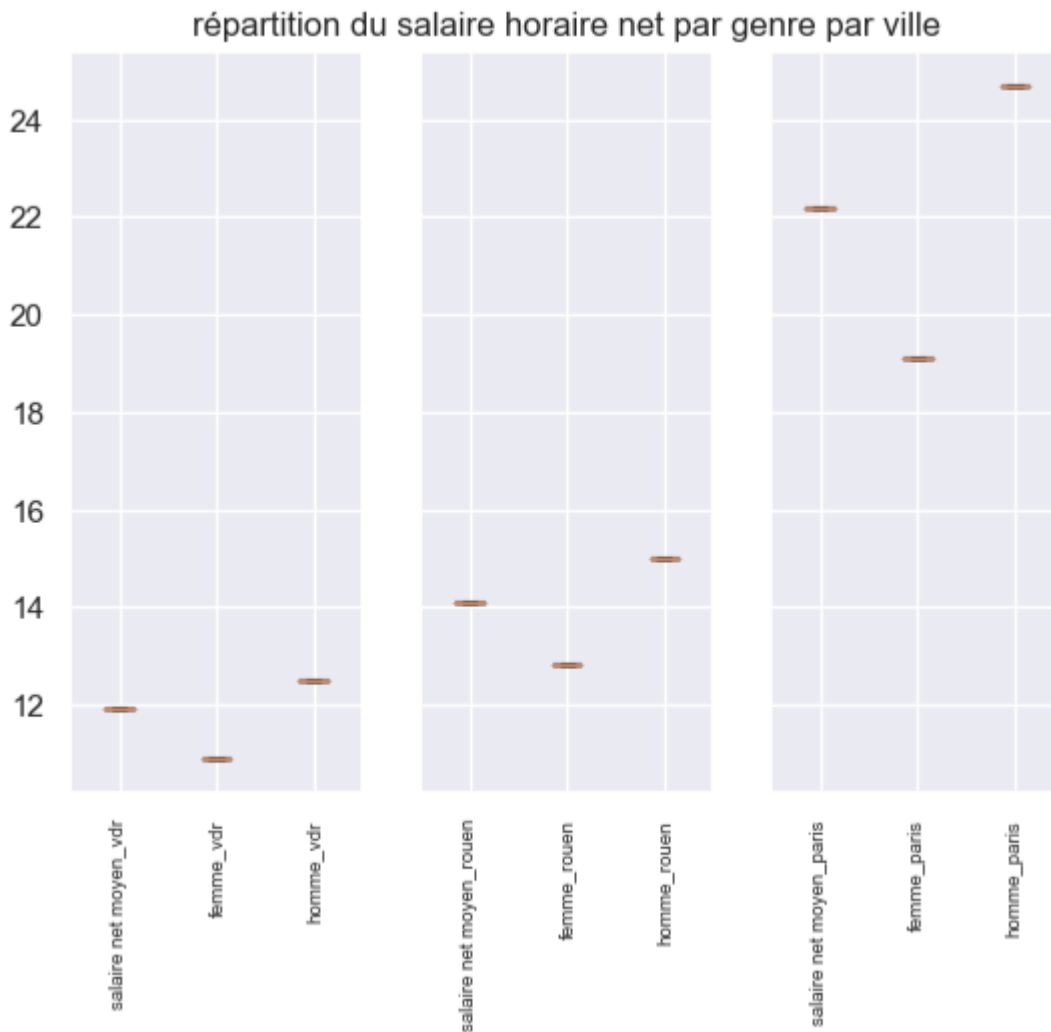




Si nous observons les entreprises de cette ville, nous constatons une même tendance qu'au niveau national pour les petites entreprises, c'est à dire un nombre conséquent d'entreprise de 0/ND employé, mais en écartant cette catégorie nous observons également des entreprises plus conséquentes de plus de 100, 200, 500 employés.



Enfin, une représentation graphique du salaire par comparaison avec Rouen et Paris, nous montre une très forte différence quant au salaire moyen net horaire sans distinction de genre.



Conclusion et difficultés rencontrées lors du projet

Le projet French Industry est un projet très complet avec 4 databases à étudier et des multitudes d'analyses à réaliser. Il s'agit de maîtriser les outils d'exploration des données (tel que la bibliothèque pandas), les outils de visualisation (tel que matplotlib ou seaborn) et réussir à faire des jointures de tableaux pour croiser les données.

Nous avons pu constater à travers ce rapport, que les analyses sont nombreuses et peuvent être réalisées à l'infini au vu du nombre de variables présentes.

Il en ressort spécifiquement sur ce sujet que la répartition des industries est inégale à travers le territoire, également la mesure des salaires net horaire et très dépendante du genre de la personne, de son type d'emploi et de sa situation géographique

Pour aller plus loin dans le projet avec davantage de temps pour la réalisation, nous aurions pu creuser notamment en croisant davantage les tables, en allant chercher des données (web scraping ou table en open data sur data.gouv) telles que les secteurs des entreprises

listées dans de df_entreprise et en réalisant un tableau de bord via power Bi ce qui aurait permis une analyse dynamique des données à travers l'ensemble du territoire