

pro

August 23, 2023

```
[153]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import re
import seaborn as sns
import missingno as msno
from statsmodels.formula.api import ols
from statsmodels.api import qqplot
from distinctipy import distinctipy
```

```
[154]: # Chargement des données
ins_prof_dut = pd.read_csv("../csv/
↳fr-esr-insertion_professionnelle-dut_donnees_nationales.csv", sep=";")
ins_prof_lp = pd.read_csv("../csv//
↳fr-esr-insertion_professionnelle-lp_donnees_nationales.csv", sep=";")
ins_prof_master = pd.read_csv("../csv/
↳fr-esr-insertion_professionnelle-master_donnees_nationales.csv", sep=";")
```

```
[155]: ins_prof_dut.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1128 entries, 0 to 1127
Data columns (total 75 columns):
 #   Column                                     Non-
Null Count  Dtype
---  -
-----
 0   Année                                     1128
non-null    object
 1   Diplôme                                   1128
non-null    object
 2   situation                                1128
non-null    object
 3   Genre                                    1128
non-null    object
 4   Disciplines                             1128
non-null    object
 5   Code du domaine                         1128
```

non-null	object	
6	Domaine	1128
non-null	object	
7	Code de la discipline	1128
non-null	object	
8	Discipline	1128
non-null	object	
9	Sigle	932
non-null	object	
10	Spécialité	1128
non-null	object	
11	Nombre de réponses	1128
non-null	int64	
12	Taux d'insertion	1128
non-null	object	
13	Part des emplois de niveau cadre ou profession intermédiaire	1128
non-null	object	
14	Part des emplois de niveau cadre	1112
non-null	object	
15	% emplois extérieurs à la région de l'université	1128
non-null	object	
16	Part des emplois stables	1128
non-null	object	
17	Part des emplois à temps plein	1128
non-null	object	
18	Salaire net mensuel médian des emplois à temps plein	1128
non-null	object	
19	Salaire brut annuel estimé	1128
non-null	object	
20	Premier quartile des salaires nets mensuels des emplois à temps plein	1128
non-null	object	
21	Dernier quartile des salaires nets mensuels des emplois à temps plein	1128
non-null	object	
22	Part des diplômés boursiers dans la discipline	1128
non-null	object	
23	Part des femmes	412
non-null	object	
24	Taux de chômage national	1128
non-null	float64	
25	Salaire net mensuel médian national	1128
non-null	int64	
26	Code du secteur disciplinaire SISE	20
non-null	object	
27	Salaire net mensuel national 1er quartile	1128
non-null	int64	
28	Salaire net mensuel national 3ème quartile	1128
non-null	int64	
29	cle_DISC	1128

non-null	object	
30	Prof. libérale, indépendant,' chef d'entreprise	343
non-null	object	
31	Fonctionnaire	343
non-null	object	
32	CDI	343
non-null	object	
33	CDI de chantier ou CDI de mission	216
non-null	object	
34	Contrat spécifique au doctorat	0
non-null	float64	
35	CDD	343
non-null	object	
36	Vacataire	343
non-null	object	
37	Intérimaire	343
non-null	object	
38	Intermittent du spectacle	343
non-null	object	
39	Contrat de professionnalisa tion	343
non-null	object	
40	Emplois aidés (Contrat Initiative Emploi...)	343
non-null	object	
41	Volontariat international	330
non-null	object	
42	Vous-même	175
non-null	object	
43	La fonction publique (d''etat',' territoriale ou hospitalière)	175
non-null	object	
44	Une entreprise privée	175
non-null	object	
45	Une entreprise publique	175
non-null	object	
46	Une association	175
non-null	object	
47	Une personne exerçant une profession libérale ou un indépendant	175
non-null	object	
48	Organisation internationale ou une institution de l''Union européenne	111
non-null	object	
49	Société d''économie mixte	124
non-null	object	
50	Un particulier	175
non-null	object	
51	Agriculture', sylviculture et pêche	174
non-null	object	
52	Industries (manufacturières, extractives et autres)	174
non-null	object	
53	Construction	174

```

non-null    object
  54 Activités immobilières                                124
non-null    object
  55 Commerce, transports, héberg-ement et restauration    174
non-null    object
  56 Information et communication                          174
non-null    object
  57 Activités financières et d'assurance                  174
non-null    object
  58 Activités spécialisées, scientifiques et techniques    174
non-null    object
  59 Activités de services administratifs et de soutien    174
non-null    object
  60 Enseignement                                          174
non-null    object
  61 Administra tion publique (hors ens.)                  174
non-null    object
  62 Santé humaine et action sociale                      174
non-null    object
  63 Arts, spectacles et activités récréatives            174
non-null    object
  64 Autres activités de service                          174
non-null    object
  65 Agriculteur                                           229
non-null    object
  66 Artisan, commerçant,' chef d''entreprise             229
non-null    object
  67 Profession libérale                                  242
non-null    object
  68 Personnel de catégorie A de la fonction publique      343
non-null    object
  69 Ingénieur', cadre, prof. libérales, prof. intellectuelles sup 343
non-null    object
  70 Personnel de catégorie B de la fonction publique      343
non-null    object
  71 Emploi de niveau intermédiaire : technicien, agent de maîtrise... 343
non-null    object
  72 Personnel de catégorie C de la fonction publique      343
non-null    object
  73 Manœuvre, ouvrier                                    343
non-null    object
  74 Employé de bureau, de commerce, personnel de service 1128
non-null    object
dtypes: float64(2), int64(4), object(69)
memory usage: 661.1+ KB

```

```
[156]: ins_prof_lp.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1867 entries, 0 to 1866
Data columns (total 75 columns):
 #   Column                                     Non-
Null Count  Dtype
---  -
-----
 0   Année                                     1867
non-null    object
 1   Diplôme                                 1867
non-null    object
 2   situation                             1867
non-null    object
 3   Genre                                 1867
non-null    object
 4   Disciplines                           1867
non-null    object
 5   Code du domaine                       1867
non-null    object
 6   Domaine                              1867
non-null    object
 7   Code de la discipline                 1867
non-null    object
 8   Discipline                           1867
non-null    object
 9   Code du secteur disciplinaire         1867
non-null    object
10   Secteur disciplinaire                 1867
non-null    object
11   Nombre de réponses                   1747
non-null    float64
12   Taux d'insertion                     1851
non-null    object
13   Part des emplois de niveau cadre ou profession intermédiaire 1851
non-null    object
14   Part des emplois de niveau cadre     1845
non-null    object
15   % emplois extérieurs à la région de l'université 1851
non-null    object
16   Part des emplois stables              1851
non-null    object
17   Part des emplois à temps plein       1851
non-null    object
18   Salaire net mensuel médian des emplois à temps plein 1851
non-null    object
19   Salaire brut annuel estimé           1851
non-null    object
20   Premier quartile des salaires nets mensuels des emplois à temps plein 1851

```

non-null	object	
21	Dernier quartile des salaires nets mensuels des emplois à temps plein	1851
non-null	object	
22	Part des diplômés boursiers dans la discipline	1851
non-null	object	
23	Part des femmes	675
non-null	object	
24	Taux de chômage national	1867
non-null	float64	
25	Salaire net mensuel médian national	1867
non-null	int64	
26	Code du secteur disciplinaire SISE	1502
non-null	object	
27	Salaire net mensuel national 1er quartile	1867
non-null	int64	
28	Salaire net mensuel national 3ème quartile	1867
non-null	int64	
29	cle_DISC	1867
non-null	object	
30	Prof. libérale, indépendant,' chef d'entreprise	517
non-null	object	
31	Fonctionnaire	517
non-null	object	
32	CDI	517
non-null	object	
33	CDI de chantier ou CDI de mission	373
non-null	object	
34	Contrat spécifique au doctorat	39
non-null	object	
35	CDD	517
non-null	object	
36	Vacataire	517
non-null	object	
37	Intérimaire	517
non-null	object	
38	Intermittent du spectacle	517
non-null	object	
39	Contrat de professionnalisa tion	517
non-null	object	
40	Emplois aidés (Contrat Initiative Emploi...)	517
non-null	object	
41	Volontariat international	517
non-null	object	
42	Vous-même	258
non-null	object	
43	La fonction publique (d'etat',' territoriale ou hospitalière)	258
non-null	object	
44	Une entreprise privée	258

non-null	object	
45	Une entreprise publique	258
non-null	object	
46	Une association	258
non-null	object	
47	Une personne exerçant une profession libérale ou un indépendant	258
non-null	object	
48	Organisation internationale ou une institution de l'Union européenne	186
non-null	object	
49	Société d'économie mixte	186
non-null	object	
50	Un particulier	258
non-null	object	
51	Agriculture', sylviculture et pêche	258
non-null	object	
52	Industries (manufacturières, extractives et autres)	258
non-null	object	
53	Construction	258
non-null	object	
54	Activités immobilières	186
non-null	object	
55	Commerce, transports, hébergement et restauration	258
non-null	object	
56	Information et communication	258
non-null	object	
57	Activités financières et d'assurance	258
non-null	object	
58	Activités spécialisées, scientifiques et techniques	258
non-null	object	
59	Activités de services administratifs et de soutien	258
non-null	object	
60	Enseignement	258
non-null	object	
61	Administration publique (hors ens.)	258
non-null	object	
62	Santé humaine et action sociale	258
non-null	object	
63	Arts, spectacles et activités récréatives	258
non-null	object	
64	Autres activités de service	258
non-null	object	
65	Agriculteur	373
non-null	object	
66	Artisan, commerçant, chef d'entreprise	373
non-null	object	
67	Profession libérale	373
non-null	object	
68	Personnel de catégorie A de la fonction publique	517

```

non-null    object
 69 Ingénieur', cadre, prof. libérales, prof. intellectuelles sup      517
non-null    object
 70 Personnel de catégorie B de la fonction publique                    517
non-null    object
 71 Emploi de niveau intermédiaire : technicien, agent de maîtrise...  517
non-null    object
 72 Personnel de catégorie C de la fonction publique                    517
non-null    object
 73 Manœuvre, ouvrier                                                  517
non-null    object
 74 Employé de bureau, de commerce, personnel de service              1867
non-null    object
dtypes: float64(2), int64(3), object(70)
memory usage: 1.1+ MB

```

```
[157]: ins_prof_master.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2925 entries, 0 to 2924
Data columns (total 75 columns):
 #   Column                                     Non-
Null Count  Dtype
---  -
-----
 0   Année                                     2925
non-null    int64
 1   Diplôme                                   2925
non-null    object
 2   situation                               2925
non-null    object
 3   Genre                                    2925
non-null    object
 4   Disciplines                             2925
non-null    object
 5   Code du domaine                         2925
non-null    object
 6   Domaine                                 2925
non-null    object
 7   Code de la discipline                   2925
non-null    object
 8   Discipline                             2925
non-null    object
 9   Code du secteur disciplinaire           2925
non-null    object
10   Secteur disciplinaire                   2925
non-null    object
11   Nombre de réponses                     2609

```


non-null	float64	
12	Taux d'insertion	2911
non-null	object	
13	Part des emplois de niveau cadre ou profession intermédiaire	2911
non-null	object	
14	Part des emplois de niveau cadre	2804
non-null	object	
15	% emplois extérieurs à la région de l'université	2911
non-null	object	
16	Part des emplois stables	2911
non-null	object	
17	Part des emplois à temps plein	2911
non-null	object	
18	Salaire net mensuel médian des emplois à temps plein	2911
non-null	object	
19	Salaire brut annuel estimé	2911
non-null	object	
20	Premier quartile des salaires nets mensuels des emplois à temps plein	2911
non-null	object	
21	Dernier quartile des salaires nets mensuels des emplois à temps plein	2911
non-null	object	
22	Part des diplômés boursiers dans la discipline	2697
non-null	object	
23	Part des femmes	1067
non-null	object	
24	Taux de chômage national	2925
non-null	float64	
25	Salaire net mensuel médian national	2925
non-null	int64	
26	Code du secteur disciplinaire SISE	2491
non-null	object	
27	Salaire net mensuel national 1er quartile	2866
non-null	float64	
28	Salaire net mensuel national 3ème quartile	2866
non-null	float64	
29	cle_DISC	2925
non-null	object	
30	Prof. libérale, indépendant, chef d'entreprise	642
non-null	object	
31	Fonctionnaire	642
non-null	object	
32	CDI	642
non-null	object	
33	CDI de chantier ou CDI de mission	490
non-null	object	
34	Contrat spécifique au doctorat	144
non-null	float64	
35	CDD	642

non-null	object	
36	Vacataire	642
non-null	object	
37	Intérimaire	642
non-null	object	
38	Intermittent du spectacle	640
non-null	object	
39	Contrat de professionnalisation	640
non-null	object	
40	Emplois aidés (Contrat Initiative Emploi...)	624
non-null	object	
41	Volontariat international	618
non-null	object	
42	Vous-même	312
non-null	object	
43	La fonction publique (d'etat, territoriale ou hospitalière)	312
non-null	object	
44	Une entreprise privée	312
non-null	object	
45	Une entreprise publique	312
non-null	object	
46	Une association	312
non-null	object	
47	Une personne exerçant une profession libérale ou un indépendant	312
non-null	object	
48	Organisation internationale ou une institution de l'Union européenne	236
non-null	object	
49	Société d'économie mixte	236
non-null	object	
50	Un particulier	312
non-null	object	
51	Agriculture, sylviculture et pêche	311
non-null	object	
52	Industries (manufacturières, extractives et autres)	312
non-null	object	
53	Construction	312
non-null	object	
54	Activités immobilières	236
non-null	object	
55	Commerce, transports, héberg-ement et restauration	312
non-null	object	
56	Information et communication	312
non-null	object	
57	Activités financières et d'assurance	312
non-null	object	
58	Activités spécialisées, scientifiques et techniques	312
non-null	object	
59	Activités de services administratifs et de soutien	312

```

non-null    object
60 Enseignement                                     312
non-null    object
61 Administration publique (hors ens.)              312
non-null    object
62 Santé humaine et action sociale                  312
non-null    object
63 Arts, spectacles et activités récréatives        312
non-null    object
64 Autres activités de service                       312
non-null    object
65 Agriculteur                                      434
non-null    object
66 Artisan, commerçant, chef d'entreprise           472
non-null    object
67 Profession libérale                              472
non-null    object
68 Personnel de catégorie A de la fonction publique 624
non-null    object
69 Ingénieur, cadre, prof. libérales, prof. intellectuelles sup 624
non-null    object
70 Personnel de catégorie B de la fonction publique 624
non-null    object
71 Emploi de niveau intermédiaire : technicien, agent de maîtrise... 624
non-null    object
72 Personnel de catégorie C de la fonction publique 624
non-null    object
73 Manœuvre, ouvrier                               624
non-null    object
74 Employé de bureau, de commerce, personnel de service 624
non-null    object
dtypes: float64(5), int64(2), object(68)
memory usage: 1.7+ MB

```

Etude de l'insertion professionnelle des DUT, licences pro et Master

```

[158]: # Agrégation de toutes les formations (DUT, LP, MASTER)
ins_prof = pd.concat([ins_prof_dut, ins_prof_lp, ins_prof_master], keys=["D", "L", "M"])

```

```

[159]: ins_prof.head()

```

```

[159]:   Année Diplôme      situation      Genre \
D 0  2013      DUT  30 mois après le diplôme      hommes
1  2017      DUT  18 mois après le diplôme  femmes et hommes
2  2017      DUT  18 mois après le diplôme      hommes
3  2017      DUT  18 mois après le diplôme  femmes et hommes
4  2017      DUT  18 mois après le diplôme      femmes

```

	Disciplines	Code du domaine	\
D 0	Droit, économie et gestion>Droit>Carrières jur...		DEG
1	Sciences, technologies et santé>Informatique>I...		STS
2	Sciences, technologies et santé>Informatique>I...		STS
3	Sciences humaines et sociales>Information comm...		SHS
4	Sciences humaines et sociales>Information comm...		SHS

	Domaine	Code de la discipline	\
D 0	Droit, économie et gestion	disc02	
1	Sciences, technologies et santé	disc16	
2	Sciences, technologies et santé	disc16	
3	Sciences humaines et sociales	disc10	
4	Sciences humaines et sociales	disc10	

	Discipline	Sigle	...	Secteur disciplinaire	\
D 0	Droit	NaN	...	NaN	
1	Informatique	NaN	...	NaN	
2	Informatique	NaN	...	NaN	
3	Information communication	NaN	...	NaN	
4	Information communication	NaN	...	NaN	

	Administration publique (hors ens.)	\
D 0	NaN	
1	NaN	
2	NaN	
3	NaN	
4	NaN	

	Prof. libérale, indépendant, chef d'entreprise	\
D 0	NaN	
1	NaN	
2	NaN	
3	NaN	
4	NaN	

	Contrat de professionnalisation	\
D 0	NaN	
1	NaN	
2	NaN	
3	NaN	
4	NaN	

	La fonction publique (d'etat, territoriale ou hospitalière)	\
D 0	NaN	
1	NaN	
2	NaN	

3		NaN
4		NaN

Organisation internationale ou une institution de l'Union européenne \

D 0		NaN
1		NaN
2		NaN
3		NaN
4		NaN

Société d'économie mixte Agriculture, sylviculture et pêche \

D 0	NaN	NaN
1	NaN	NaN
2	NaN	NaN
3	NaN	NaN
4	NaN	NaN

Artisan, commerçant, chef d'entreprise \

D 0	NaN
1	NaN
2	NaN
3	NaN
4	NaN

Ingénieur, cadre, prof. libérales, prof. intellectuelles sup

D 0	NaN
1	NaN
2	NaN
3	NaN
4	NaN

[5 rows x 86 columns]

```
[160]: ins_prof.info()
```

```
<class 'pandas.core.frame.DataFrame'>
MultiIndex: 5920 entries, ('D', 0) to ('M', 2924)
Data columns (total 86 columns):
 #   Column                                     Non-
Null Count  Dtype
---  ---
0    Année                                     5920
non-null    object
1    Diplôme                                   5920
non-null    object
2    situation                                5920
non-null    object
```

3	Genre	5920
	non-null object	
4	Disciplines	5920
	non-null object	
5	Code du domaine	5920
	non-null object	
6	Domaine	5920
	non-null object	
7	Code de la discipline	5920
	non-null object	
8	Discipline	5920
	non-null object	
9	Sigle	932
	non-null object	
10	Spécialité	1128
	non-null object	
11	Nombre de réponses	5484
	non-null float64	
12	Taux d'insertion	5890
	non-null object	
13	Part des emplois de niveau cadre ou profession intermédiaire	5890
	non-null object	
14	Part des emplois de niveau cadre	5761
	non-null object	
15	% emplois extérieurs à la région de l'université	5890
	non-null object	
16	Part des emplois stables	5890
	non-null object	
17	Part des emplois à temps plein	5890
	non-null object	
18	Salaire net mensuel médian des emplois à temps plein	5890
	non-null object	
19	Salaire brut annuel estimé	5890
	non-null object	
20	Premier quartile des salaires nets mensuels des emplois à temps plein	5890
	non-null object	
21	Dernier quartile des salaires nets mensuels des emplois à temps plein	5890
	non-null object	
22	Part des diplômés boursiers dans la discipline	5676
	non-null object	
23	Part des femmes	2154
	non-null object	
24	Taux de chômage national	5920
	non-null float64	
25	Salaire net mensuel médian national	5920
	non-null int64	
26	Code du secteur disciplinaire SISE	4013
	non-null object	

27 Salaire net mensuel national 1er quartile	5861
non-null float64	
28 Salaire net mensuel national 3ème quartile	5861
non-null float64	
29 cle_DISC	5920
non-null object	
30 Prof. libérale, indépendant,' chef d'entreprise	860
non-null object	
31 Fonctionnaire	1502
non-null object	
32 CDI	1502
non-null object	
33 CDI de chantier ou CDI de mission	1079
non-null object	
34 Contrat spécifique au doctorat	183
non-null object	
35 CDD	1502
non-null object	
36 Vacataire	1502
non-null object	
37 Intérimaire	1502
non-null object	
38 Intermittent du spectacle	1500
non-null object	
39 Contrat de professionnalisat ion	860
non-null object	
40 Emplois aidés (Contrat Initiative Emploi...)	1484
non-null object	
41 Volontariat international	1465
non-null object	
42 Vous-même	745
non-null object	
43 La fonction publique (d'etat',' territoriale ou hospitalière)	433
non-null object	
44 Une entreprise privée	745
non-null object	
45 Une entreprise publique	745
non-null object	
46 Une association	745
non-null object	
47 Une personne exerçant une profession libérale ou un indépendant	745
non-null object	
48 Organisation internationale ou une institution de l'Union européenne	297
non-null object	
49 Société d'économie mixte	310
non-null object	
50 Un particulier	745
non-null object	

51	Agriculture', sylviculture et pêche	432
non-null	object	
52	Industries (manufacturières, extractives et autres)	744
non-null	object	
53	Construction	744
non-null	object	
54	Activités immobilières	546
non-null	object	
55	Commerce, transports, héberg-ement et restauration	744
non-null	object	
56	Information et communication	744
non-null	object	
57	Activités financières et d'assurance	744
non-null	object	
58	Activités spécialisées, scientifiques et techniques	744
non-null	object	
59	Activités de services administratifs et de soutien	744
non-null	object	
60	Enseignement	744
non-null	object	
61	Administra tion publique (hors ens.)	174
non-null	object	
62	Santé humaine et action sociale	744
non-null	object	
63	Arts, spectacles et activités récréatives	744
non-null	object	
64	Autres activités de service	744
non-null	object	
65	Agriculteur	1036
non-null	object	
66	Artisan, commerçant,' chef d''entreprise	602
non-null	object	
67	Profession libérale	1087
non-null	object	
68	Personnel de catégorie A de la fonction publique	1484
non-null	object	
69	Ingénieur', cadre, prof. libérales, prof. intellectuelles sup	860
non-null	object	
70	Personnel de catégorie B de la fonction publique	1484
non-null	object	
71	Emploi de niveau intermédiaire : technicien, agent de maîtrise...	1484
non-null	object	
72	Personnel de catégorie C de la fonction publique	1484
non-null	object	
73	Manœuvre, ouvrier	1484
non-null	object	
74	Employé de bureau, de commerce, personnel de service	3619
non-null	object	

75	Code du secteur disciplinaire	4792
non-null	object	
76	Secteur disciplinaire	4792
non-null	object	
77	Administration publique (hors ens.)	570
non-null	object	
78	Prof. libérale, indépendant, chef d'entreprise	642
non-null	object	
79	Contrat de professionnalisation	640
non-null	object	
80	La fonction publique (d'etat, territoriale ou hospitalière)	312
non-null	object	
81	Organisation internationale ou une institution de l'Union européenne	236
non-null	object	
82	Société d'économie mixte	236
non-null	object	
83	Agriculture, sylviculture et pêche	311
non-null	object	
84	Artisan, commerçant, chef d'entreprise	472
non-null	object	
85	Ingénieur, cadre, prof. libérales, prof. intellectuelles sup	624
non-null	object	

dtypes: float64(4), int64(1), object(81)
memory usage: 4.0+ MB

Les variables de 75 à 85 sont à corriger car elles sont en supplément des 3 datasets d'origine.

```
[161]: # Correction des noms des variables en DUT et LP en comparaison des Masters
ins_prof_dut.rename(columns=
{
    "Sigle": "Code du secteur disciplinaire",
    "Spécialité": "Secteur disciplinaire",
    "Ingénieur', cadre, prof. libérales, prof. intellectuelles sup":
↪ "Ingénieur, cadre, prof. libérales, prof. intellectuelles sup",
    "Artisan, commerçant,' chef d''entreprise": "Artisan, commerçant, chef_
↪ d'entreprise",
    "Administra tion publique (hors ens.)": "Administration publique (hors_
↪ ens.)",
    "Prof. libérale, indépendant,' chef d'entreprise": "Prof. libérale,_
↪ indépendant, chef d'entreprise",
    "Contrat de professionnalisa tion": "Contrat de professionnalisation",
    "La fonction publique (d'etat',' territoriale ou hospitalière)": "La_
↪ fonction publique (d'etat, territoriale ou hospitalière)",
    "Organisation internationale ou une institution de l''Union européenne":
↪ "Organisation internationale ou une institution de l'Union européenne",
    "Société d' 'économie mixte": "Société d'économie mixte",
```

```

        "Agriculture', sylviculture et pêche": "Agriculture, sylviculture et
        ↳pêche"
    }, inplace=True
)

ins_prof_lp.rename(columns=
    {
        "Ingénieur', cadre, prof. libérales, prof. intellectuelles sup":
        ↳"Ingénieur, cadre, prof. libérales, prof. intellectuelles sup",
        "Artisan, commerçant,' chef d'entreprise": "Artisan, commerçant, chef
        ↳d'entreprise",
        "Administra tion publique (hors ens.)": "Administration publique (hors
        ↳ens.)",
        "Prof. libérale, indépendant,' chef d'entreprise": "Prof. libérale,
        ↳indépendant, chef d'entreprise",
        "Contrat de professionnalisa tion": "Contrat de professionnalisation",
        "La fonction publique (d'etat',' territoriale ou hospitalière)": "La
        ↳fonction publique (d'etat, territoriale ou hospitalière)",
        "Organisation internationale ou une institution de l'Union européenne":
        ↳ "Organisation internationale ou une institution de l'Union européenne",
        "Société d'économie mixte": "Société d'économie mixte",
        "Agriculture', sylviculture et pêche": "Agriculture, sylviculture et
        ↳pêche"
    }, inplace = True
)

```

```

[162]: # Nouvelle concaténation avec les datasets corrigés
ins_prof = pd.concat([ins_prof_dut, ins_prof_lp, ins_prof_master], keys=["D",
        ↳"L", "M"])

```

```

[163]: ins_prof.info()

```

```

<class 'pandas.core.frame.DataFrame'>
MultiIndex: 5920 entries, ('D', 0) to ('M', 2924)
Data columns (total 75 columns):
 #   Column                                     Non-
Null Count  Dtype
---  -
-----
 0   Année                                     5920
non-null    object
 1   Diplôme                                   5920
non-null    object
 2   situation                                5920
non-null    object
 3   Genre                                    5920
non-null    object

```

4	Disciplines	5920
	non-null object	
5	Code du domaine	5920
	non-null object	
6	Domaine	5920
	non-null object	
7	Code de la discipline	5920
	non-null object	
8	Discipline	5920
	non-null object	
9	Code du secteur disciplinaire	5724
	non-null object	
10	Secteur disciplinaire	5920
	non-null object	
11	Nombre de réponses	5484
	non-null float64	
12	Taux d'insertion	5890
	non-null object	
13	Part des emplois de niveau cadre ou profession intermédiaire	5890
	non-null object	
14	Part des emplois de niveau cadre	5761
	non-null object	
15	% emplois extérieurs à la région de l'université	5890
	non-null object	
16	Part des emplois stables	5890
	non-null object	
17	Part des emplois à temps plein	5890
	non-null object	
18	Salaire net mensuel médian des emplois à temps plein	5890
	non-null object	
19	Salaire brut annuel estimé	5890
	non-null object	
20	Premier quartile des salaires nets mensuels des emplois à temps plein	5890
	non-null object	
21	Dernier quartile des salaires nets mensuels des emplois à temps plein	5890
	non-null object	
22	Part des diplômés boursiers dans la discipline	5676
	non-null object	
23	Part des femmes	2154
	non-null object	
24	Taux de chômage national	5920
	non-null float64	
25	Salaire net mensuel médian national	5920
	non-null int64	
26	Code du secteur disciplinaire SISE	4013
	non-null object	
27	Salaire net mensuel national 1er quartile	5861
	non-null float64	

28 Salaire net mensuel national 3ème quartile	5861
non-null float64	
29 cle_DISC	5920
non-null object	
30 Prof. libérale, indépendant, chef d'entreprise	1502
non-null object	
31 Fonctionnaire	1502
non-null object	
32 CDI	1502
non-null object	
33 CDI de chantier ou CDI de mission	1079
non-null object	
34 Contrat spécifique au doctorat	183
non-null object	
35 CDD	1502
non-null object	
36 Vacataire	1502
non-null object	
37 Intérimaire	1502
non-null object	
38 Intermittent du spectacle	1500
non-null object	
39 Contrat de professionnalisation	1500
non-null object	
40 Emplois aidés (Contrat Initiative Emploi...)	1484
non-null object	
41 Volontariat international	1465
non-null object	
42 Vous-même	745
non-null object	
43 La fonction publique (d'etat, territoriale ou hospitalière)	745
non-null object	
44 Une entreprise privée	745
non-null object	
45 Une entreprise publique	745
non-null object	
46 Une association	745
non-null object	
47 Une personne exerçant une profession libérale ou un indépendant	745
non-null object	
48 Organisation internationale ou une institution de l'Union européenne	533
non-null object	
49 Société d'économie mixte	546
non-null object	
50 Un particulier	745
non-null object	
51 Agriculture, sylviculture et pêche	743
non-null object	

52 Industries (manufacturières, extractives et autres)	744
non-null object	
53 Construction	744
non-null object	
54 Activités immobilières	546
non-null object	
55 Commerce, transports, hébergement et restauration	744
non-null object	
56 Information et communication	744
non-null object	
57 Activités financières et d'assurance	744
non-null object	
58 Activités spécialisées, scientifiques et techniques	744
non-null object	
59 Activités de services administratifs et de soutien	744
non-null object	
60 Enseignement	744
non-null object	
61 Administration publique (hors ens.)	744
non-null object	
62 Santé humaine et action sociale	744
non-null object	
63 Arts, spectacles et activités récréatives	744
non-null object	
64 Autres activités de service	744
non-null object	
65 Agriculteur	1036
non-null object	
66 Artisan, commerçant, chef d'entreprise	1074
non-null object	
67 Profession libérale	1087
non-null object	
68 Personnel de catégorie A de la fonction publique	1484
non-null object	
69 Ingénieur, cadre, prof. libérales, prof. intellectuelles sup	1484
non-null object	
70 Personnel de catégorie B de la fonction publique	1484
non-null object	
71 Emploi de niveau intermédiaire : technicien, agent de maîtrise...	1484
non-null object	
72 Personnel de catégorie C de la fonction publique	1484
non-null object	
73 Manœuvre, ouvrier	1484
non-null object	
74 Employé de bureau, de commerce, personnel de service	3619
non-null object	

dtypes: float64(4), int64(1), object(70)

memory usage: 3.5+ MB

```
[164]: ins_prof["Discipline"].unique()
```

```
[164]: array(['Droit', 'Informatique', 'Information communication',  
        'Sciences fondamentales', 'Gestion',  
        'Autres formations juridiques, économiques et de gestion',  
        'Autres sciences humaines et sociales', 'Sciences de l''ingénieur',  
        'Ensemble des départements d''IUT',  
        'Ensemble formations juridiques, économiques et de gestion',  
        'Ensemble sciences humaines et sociales',  
        'Ensemble sciences, technologies et santé',  
        'Sciences de la vie et de la terre', 'Lettres, langues, arts',  
        'Histoire-géographie', 'Psychologie',  
        'Autres sciences, technologies et santé',  
        'Autres sciences, technologies et santé',  
        'Ensemble Licence professionnelle', 'Économie',  
        'Ensemble Masters LMD (hors Masters enseignement et hors Dauphine et  
        Antilles-Guyane)',  
        'Sciences de l''ingénieur', 'Masters enseignement',  
        'Ensemble Masters LMD (hors Masters enseignement)',  
        'Masters enseignement : premier degré',  
        'Masters enseignement : second degré, CPE...'], dtype=object)
```

```
[165]: # Renommage de certaines disciplines  
ins_prof["Discipline"] = ins_prof["Discipline"].replace(  
    ["Sciences de l''ingénieur", "Sciences de l''ingénieur"],  
    ["Ensemble des départements d''IUT", "Ensemble des départements d'IUT"])
```

```
[166]: # Recherche des valeurs représentant l'année  
ins_prof['Année'].unique()
```

```
[166]: array(['2013', '2017', '2014', '2019', "'2019", '2015', '2016', "'2013",  
        "'2014", "'2015", "'2017", "'2016", '2018', "'2018", 2013, 2014,  
        2012, 2011, 2019, 2016, 2015, 2018, 2017], dtype=object)
```

```
[167]: # Extraction des années afin de supprimer les cotes  
ins_prof["Année"] = ins_prof["Année"].astype(str).str.extract(r"([0-9]{4})")
```

```
[168]: # Liste des contrats  
liste_contrats = ins_prof.columns[30:42].to_list()  
  
# Liste des types d'entreprises  
liste_entreprises = ins_prof.columns[42:51].to_list()  
  
# Liste des secteurs d'activités  
liste_secteurs = ins_prof.columns[51:65].to_list()  
  
# Liste des professions
```

```
liste_professions = ins_prof.columns[65:].to_list()
```

```
[169]: # Liste de l'ensemble des variables sélectionnées pour l'étude des valeurs
↳manquantes
categories = liste_contrats + liste_professions + liste_secteurs +
↳liste_entreprises
df_categories = ins_prof[["Année", "Nombre de réponses"] + categories]
```

```
[170]: # Total de valeurs manquantes des modalités par catégories (contrat, secteur,
↳profession, entreprises)
total_mod_contrats = ins_prof[liste_contrats].isna().sum()
total_mod_secteurs = ins_prof[liste_secteurs].isna().sum()
total_mod_professions = ins_prof[liste_professions].isna().sum()
total_mod_entreprises = ins_prof[liste_entreprises].isna().sum()

# Total des valeurs manquantes pour chaque catégorie
total_na_contrats = total_mod_contrats.sum()
total_na_secteurs = total_mod_secteurs.sum()
total_na_professions = total_mod_professions.sum()
total_na_entreprises = total_mod_entreprises.sum()

# Calcul du pourcentage de valeurs manquantes des modalités pour chaque
↳catégorie
percent_na_contrats = round(total_mod_contrats / total_na_contrats * 100, 1)
percent_na_secteurs = round(total_mod_secteurs / total_na_secteurs * 100, 1)
percent_na_professions = round(total_mod_professions / total_na_professions *
↳100, 1)
percent_na_entreprises = round(total_mod_entreprises / total_na_professions *
↳100, 1)
#
```

```
[171]: df_percent_na_contrats = pd.DataFrame(
    {
        "contrat": percent_na_contrats.index,
        "%": percent_na_contrats.values
    }
)
df_percent_na_secteurs = pd.DataFrame(
    {
        "contrat": percent_na_secteurs.index,
        "%": percent_na_secteurs.values
    }
)
df_percent_na_professions = pd.DataFrame(
    {
```

```

        "contrat": percent_na_professions.index,
        "%": percent_na_professions.values
    }
)
df_percent_na_entreprises= pd.DataFrame(
    {
        "contrat": percent_na_entreprises.index,
        "%": percent_na_entreprises.values
    }
)

```

```
[172]: df_percent_na_professions
```

```
[172]:
```

	contrat	%
0	Agriculteur	11.2
1	Artisan, commerçant, chef d'entreprise	11.1
2	Profession libérale	11.1
3	Personnel de catégorie A de la fonction publique	10.2
4	Ingénieur, cadre, prof. libérales, prof. intel...	10.2
5	Personnel de catégorie B de la fonction publique	10.2
6	Emploi de niveau intermédiaire : technicien, a...	10.2
7	Personnel de catégorie C de la fonction publique	10.2
8	Manœuvre, ouvrier	10.2
9	Employé de bureau, de commerce, personnel de s...	5.3

```
[173]: ins_prof.reset_index(inplace=True)
```

```
[174]: # Version des données avec valeurs manquantes
ins_prof_na = ins_prof.copy()
```

```
[175]: ins_prof.drop("level_1", axis=1, inplace=True)
ins_prof.rename(columns={"level_0": "label"}, inplace=True)
```

```
[176]: # Remplacement des valeurs manquantes, non significatives ou non communiquées,
↳ par la valeur 0
ins_prof = ins_prof.replace([pd.NA, "ns", "nd"], '0')
```

```
[177]: # Sélection des variables à convertir en numérique
colonnes = ins_prof.columns[12:].to_list()
```

```
[178]: # Suppression des variables à ne pas convertir de la liste sélectionnée
colonnes.remove("Nombre de réponses")
colonnes.remove("Taux de chômage national")
colonnes.remove("cle_DISC")
colonnes.remove("Code du secteur disciplinaire SISE")
```



```
colonnes.remove("Salaire net mensuel médian national")
colonnes.remove("Salaire net mensuel national 1er quartile")
colonnes.remove("Salaire net mensuel national 3ème quartile")
```

```
[179]: # suppression des valeurs non numériques pour chaque variable de la liste
for col in colonnes:
    ins_prof[col] = ins_prof[col].str.extract(r"(\d+)")
```

```
[180]: # Changement de type pour les variables sélectionnées
ins_prof[colonnes] = ins_prof[colonnes].astype(float)
```

```
[181]: ins_prof.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 5920 entries, 0 to 5919
```

```
Data columns (total 76 columns):
```

#	Column	Non-
Null Count	Dtype	
---	-----	
0	label	5920
non-null	object	
1	Année	5920
non-null	object	
2	Diplôme	5920
non-null	object	
3	situation	5920
non-null	object	
4	Genre	5920
non-null	object	
5	Disciplines	5920
non-null	object	
6	Code du domaine	5920
non-null	object	
7	Domaine	5920
non-null	object	
8	Code de la discipline	5920
non-null	object	
9	Discipline	5920
non-null	object	
10	Code du secteur disciplinaire	5920
non-null	object	
11	Secteur disciplinaire	5920
non-null	object	
12	Nombre de réponses	5484
non-null	float64	
13	Taux d'insertion	5920
non-null	float64	

14	Part des emplois de niveau cadre ou profession intermédiaire	5920
non-null	float64	
15	Part des emplois de niveau cadre	5920
non-null	float64	
16	% emplois extérieurs à la région de l'université	5920
non-null	float64	
17	Part des emplois stables	5920
non-null	float64	
18	Part des emplois à temps plein	5920
non-null	float64	
19	Salaire net mensuel médian des emplois à temps plein	5920
non-null	float64	
20	Salaire brut annuel estimé	5920
non-null	float64	
21	Premier quartile des salaires nets mensuels des emplois à temps plein	5920
non-null	float64	
22	Dernier quartile des salaires nets mensuels des emplois à temps plein	5920
non-null	float64	
23	Part des diplômés boursiers dans la discipline	5920
non-null	float64	
24	Part des femmes	5920
non-null	float64	
25	Taux de chômage national	5920
non-null	float64	
26	Salaire net mensuel médian national	5920
non-null	int64	
27	Code du secteur disciplinaire SISE	5920
non-null	object	
28	Salaire net mensuel national 1er quartile	5861
non-null	float64	
29	Salaire net mensuel national 3ème quartile	5861
non-null	float64	
30	cle_DISC	5920
non-null	object	
31	Prof. libérale, indépendant, chef d'entreprise	5920
non-null	float64	
32	Fonctionnaire	5920
non-null	float64	
33	CDI	5920
non-null	float64	
34	CDI de chantier ou CDI de mission	5920
non-null	float64	
35	Contrat spécifique au doctorat	5776
non-null	float64	
36	CDD	5920
non-null	float64	
37	Vacataire	5920
non-null	float64	

38	Intérimaire	5920
non-null	float64	
39	Intermittent du spectacle	5920
non-null	float64	
40	Contrat de professionnalisation	5920
non-null	float64	
41	Emplois aidés (Contrat Initiative Emploi...)	5920
non-null	float64	
42	Volontariat international	5920
non-null	float64	
43	Vous-même	5920
non-null	float64	
44	La fonction publique (d'etat, territoriale ou hospitalière)	5920
non-null	float64	
45	Une entreprise privée	5920
non-null	float64	
46	Une entreprise publique	5920
non-null	float64	
47	Une association	5920
non-null	float64	
48	Une personne exerçant une profession libérale ou un indépendant	5920
non-null	float64	
49	Organisation internationale ou une institution de l'Union européenne	5920
non-null	float64	
50	Société d'économie mixte	5920
non-null	float64	
51	Un particulier	5920
non-null	float64	
52	Agriculture, sylviculture et pêche	5920
non-null	float64	
53	Industries (manufacturières, extractives et autres)	5920
non-null	float64	
54	Construction	5920
non-null	float64	
55	Activités immobilières	5920
non-null	float64	
56	Commerce, transports, héberg-ement et restauration	5920
non-null	float64	
57	Information et communication	5920
non-null	float64	
58	Activités financières et d'assurance	5920
non-null	float64	
59	Activités spécialisées, scientifiques et techniques	5920
non-null	float64	
60	Activités de services administratifs et de soutien	5920
non-null	float64	
61	Enseignement	5920
non-null	float64	

```

    62 Administration publique (hors ens.) 5920
non-null float64
    63 Santé humaine et action sociale 5920
non-null float64
    64 Arts, spectacles et activités récréatives 5920
non-null float64
    65 Autres activités de service 5920
non-null float64
    66 Agriculteur 5920
non-null float64
    67 Artisan, commerçant, chef d'entreprise 5920
non-null float64
    68 Profession libérale 5920
non-null float64
    69 Personnel de catégorie A de la fonction publique 5920
non-null float64
    70 Ingénieur, cadre, prof. libérales, prof. intellectuelles sup 5920
non-null float64
    71 Personnel de catégorie B de la fonction publique 5920
non-null float64
    72 Emploi de niveau intermédiaire : technicien, agent de maîtrise... 5920
non-null float64
    73 Personnel de catégorie C de la fonction publique 5920
non-null float64
    74 Manœuvre, ouvrier 5920
non-null float64
    75 Employé de bureau, de commerce, personnel de service 3630
non-null float64
dtypes: float64(61), int64(1), object(14)
memory usage: 3.4+ MB

```

```

[182]: # Ajout de l'indication de l'année, Genre, Diplôme et la situation
df = ins_prof[["Année", "Genre", "Diplôme", "Discipline", "situation", "Nombre_
↳de réponses"]].join(ins_prof[liste_contrats + liste_secteurs +_
↳liste_professions])

```

```

[183]: df.info()

```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5920 entries, 0 to 5919
Data columns (total 42 columns):
#   Column                                     Non-Null
Count  Dtype
---  ---
-----
0   Année                                     5920 non-
null  object
1   Genre                                     5920 non-

```

2	Diplôme	5920 non-
3	Discipline	5920 non-
4	situation	5920 non-
5	Nombre de réponses	5484 non-
6	Prof. libérale, indépendant, chef d'entreprise	5920 non-
7	Fonctionnaire	5920 non-
8	CDI	5920 non-
9	CDI de chantier ou CDI de mission	5920 non-
10	Contrat spécifique au doctorat	5776 non-
11	CDD	5920 non-
12	Vacataire	5920 non-
13	Intérimaire	5920 non-
14	Intermittent du spectacle	5920 non-
15	Contrat de professionnalisation	5920 non-
16	Emplois aidés (Contrat Initiative Emploi...)	5920 non-
17	Volontariat international	5920 non-
18	Agriculture, sylviculture et pêche	5920 non-
19	Industries (manufacturières, extractives et autres)	5920 non-
20	Construction	5920 non-
21	Activités immobilières	5920 non-
22	Commerce, transports, hébergement et restauration	5920 non-
23	Information et communication	5920 non-
24	Activités financières et d'assurance	5920 non-
25	Activités spécialisées, scientifiques et techniques	5920 non-

```

null    float64
 26 Activités de services administratifs et de soutien          5920 non-
null    float64
 27 Enseignement                                                5920 non-
null    float64
 28 Administration publique (hors ens.)                        5920 non-
null    float64
 29 Santé humaine et action sociale                             5920 non-
null    float64
 30 Arts, spectacles et activités récréatives                 5920 non-
null    float64
 31 Autres activités de service                                5920 non-
null    float64
 32 Agriculteur                                                  5920 non-
null    float64
 33 Artisan, commerçant, chef d'entreprise                     5920 non-
null    float64
 34 Profession libérale                                         5920 non-
null    float64
 35 Personnel de catégorie A de la fonction publique           5920 non-
null    float64
 36 Ingénieur, cadre, prof. libérales, prof. intellectuelles sup 5920 non-
null    float64
 37 Personnel de catégorie B de la fonction publique           5920 non-
null    float64
 38 Emploi de niveau intermédiaire : technicien, agent de maîtrise... 5920 non-
null    float64
 39 Personnel de catégorie C de la fonction publique           5920 non-
null    float64
 40 Manœuvre, ouvrier                                           5920 non-
null    float64
 41 Employé de bureau, de commerce, personnel de service       3630 non-
null    float64
dtypes: float64(37), object(5)
memory usage: 1.9+ MB

```

```

[184]: # Sélection des lignes à somme nulle contrats + professions
rows_na = df[df.iloc[:, 6:].sum(axis=1, numeric_only=True) == 0].index

```

```

[185]: # Suppression des lignes contrats + professions à somme nulle.
df = df.drop(rows_na)

```

```

[186]: df.sort_values("Année", inplace=True)

```

```

[187]: # Sélection des contrats
df_contrat = df[["Année", "Diplôme", "Discipline", "Genre", "situation", "
↪Nombre de réponses"] + liste_contrats].sort_values("Année")

```

```
[188]: df_contrat.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1289 entries, 2948 to 5916
Data columns (total 18 columns):
#   Column                                          Non-Null Count  Dtype
---  -
0   Année                                          1289 non-null   object
1   Diplôme                                       1289 non-null   object
2   Discipline                                    1289 non-null   object
3   Genre                                         1289 non-null   object
4   situation                                    1289 non-null   object
5   Nombre de réponses                          1113 non-null   float64
6   Prof. libérale, indépendant, chef d'entreprise 1289 non-null   float64
7   Fonctionnaire                               1289 non-null   float64
8   CDI                                           1289 non-null   float64
9   CDI de chantier ou CDI de mission            1289 non-null   float64
10  Contrat spécifique au doctorat               1145 non-null   float64
11  CDD                                           1289 non-null   float64
12  Vacataire                                   1289 non-null   float64
13  Intérimaire                                  1289 non-null   float64
14  Intermittent du spectacle                    1289 non-null   float64
15  Contrat de professionnalisation              1289 non-null   float64
16  Emplois aidés (Contrat Initiative Emploi...) 1289 non-null   float64
17  Volontariat international                    1289 non-null   float64
dtypes: float64(13), object(5)
memory usage: 191.3+ KB
```

Proportion des catégories de contrats par année selon le niveau de diplôme et la situation

```
[189]: # Sélection de l'ensemble des hommes et des femmes
contrat_full_genre = df_contrat[(df_contrat["Genre"] == "femmes et hommes")]

# Sélection de l'ensemble des départements pour les différents diplômes
df_departements = contrat_full_genre[
    contrat_full_genre["Discipline"].str.contains(
        r"Ensemble des départements|Masters enseignement|Ensemble Masters_
↳LMD|Ensemble Licence professionnelle")
]
```

```
[190]: df_departements.head()
```

```
[190]:
```

	Année	Diplôme	Discipline	Genre \
2948	2013	LICENCE PRO	Ensemble Licence professionnelle	femmes et hommes
56	2013	DUT	Ensemble des départements d'IUT	femmes et hommes
769	2013	DUT	Ensemble des départements d'IUT	femmes et hommes
770	2013	DUT	Ensemble des départements d'IUT	femmes et hommes
1423	2013	LICENCE PRO	Ensemble Licence professionnelle	femmes et hommes

	situation	Nombre de réponses	\
2948	30 mois après le diplôme	16625.0	
56	18 mois après le diplôme	2054.0	
769	18 mois après le diplôme	456.0	
770	30 mois après le diplôme	456.0	
1423	18 mois après le diplôme	16625.0	

	Prof. libérale, indépendant, chef d'entreprise	Fonctionnaire	CDI	\
2948	2.0	3.0	73.0	
56	3.0	4.0	57.0	
769	2.0	3.0	64.0	
770	2.0	4.0	70.0	
1423	2.0	2.0	63.0	

	CDI de chantier ou CDI de mission	Contrat spécifique au doctorat	CDD	\
2948	0.0	0.0	16.0	
56	0.0	0.0	26.0	
769	0.0	0.0	17.0	
770	0.0	0.0	14.0	
1423	0.0	0.0	24.0	

	Vacataire	Intérimaire	Intermittent du spectacle	\
2948	0.0	4.0	0.0	
56	0.0	7.0	0.0	
769	0.0	12.0	0.0	
770	0.0	9.0	0.0	
1423	0.0	7.0	0.0	

	Contrat de professionnalisation	\
2948	1.0	
56	0.0	
769	1.0	
770	1.0	
1423	0.0	

	Emplois aidés (Contrat Initiative Emploi...)	Volontariat international
2948	1.0	0.0
56	1.0	0.0
769	0.0	0.0
770	0.0	0.0
1423	1.0	0.0

```
[191]: # Pourcentage de contrats par année, selon le diplôme, la discipline et le type
        ↪ de contrat
df_departements = df_departements.melt(
    id_vars=["Année", "Diplôme", "Discipline", "Genre", "situation"],
```



```
value_vars=liste_contrats, var_name="contrat", value_name="%")
```

```
[192]: df_graph = df_departements.pivot_table(
    index=["Année", "Diplôme"], columns=["situation", "contrat"],
    values="%"
)
```

```
[193]: df_graph.head()
```

```
[193]: situation      18 mois après le diplôme \
contrat              CDD   CDI
Année Diplôme
2013  DUT              21.5  60.5
      LICENCE PRO        20.5  66.0
      MASTER ENS         10.0   6.0
      MASTER LMD         29.0  55.0
2014  DUT              23.5  55.0
```

```
situation \
contrat      CDI de chantier ou CDI de mission
Année Diplôme
2013  DUT              0.0
      LICENCE PRO        0.0
      MASTER ENS         0.0
      MASTER LMD         0.0
2014  DUT              0.0
```

```
situation \
contrat      Contrat de professionnalisation
Année Diplôme
2013  DUT              0.5
      LICENCE PRO        0.0
      MASTER ENS         0.0
      MASTER LMD         0.0
2014  DUT              0.5
```

```
situation \
contrat      Contrat spécifique au doctorat
Année Diplôme
2013  DUT              0.0
      LICENCE PRO        0.0
      MASTER ENS         0.0
      MASTER LMD         NaN
2014  DUT              0.0
```

```
situation
```

contrat	Emplois aidés (Contrat Initiative Emploi...) Fonctionnaire		
Année Diplôme			
2013	DUT	0.5	3.5
	LICENCE PRO	0.5	2.0
	MASTER ENS	0.0	81.0
	MASTER LMD	1.0	5.0
2014	DUT	1.5	4.0

situation	\		
contrat	Intermittent du spectacle Intérimaire		
Année Diplôme			
2013	DUT	0.0	9.5
	LICENCE PRO	0.0	8.5
	MASTER ENS	0.0	0.0
	MASTER LMD	1.0	3.0
2014	DUT	0.0	12.5

situation	...			\
contrat	Prof. libérale, indépendant, chef d'entreprise			...
Année Diplôme				...
2013	DUT	2.5	...	
	LICENCE PRO	1.5	...	
	MASTER ENS	1.0	...	
	MASTER LMD	4.0	...	
2014	DUT	1.5	...	

situation	30 mois après le diplôme			\
contrat	CDI de chantier ou CDI de mission			
Année Diplôme				
2013	DUT	0.0		
	LICENCE PRO	0.0		
	MASTER ENS	0.0		
	MASTER LMD	0.0		
2014	DUT	0.0		

situation	\		
contrat	Contrat de professionnalisation		
Année Diplôme			
2013	DUT	1.0	
	LICENCE PRO	0.5	
	MASTER ENS	0.0	
	MASTER LMD	0.0	
2014	DUT	1.5	

situation	\		
contrat	Contrat spécifique au doctorat		
Année Diplôme			

2013	DUT	0.0
	LICENCE PRO	0.0
	MASTER ENS	0.0
	MASTER LMD	NaN
2014	DUT	0.0

situation \

contrat Emplois aidés (Contrat Initiative Emploi...) Fonctionnaire

Année Diplôme

2013	DUT	0.5	4.5
	LICENCE PRO	0.5	2.5
	MASTER ENS	1.0	83.0
	MASTER LMD	1.0	6.0
2014	DUT	0.5	5.0

situation \

contrat Intermittent du spectacle Intérimaire

Année Diplôme

2013	DUT	0.0	7.0
	LICENCE PRO	0.0	5.0
	MASTER ENS	0.0	0.0
	MASTER LMD	1.0	2.0
2014	DUT	0.0	8.0

situation \

contrat Prof. libérale, indépendant, chef d'entreprise Vacataire

Année Diplôme

2013	DUT	2.5	0.5
	LICENCE PRO	1.5	0.0
	MASTER ENS	1.0	1.0
	MASTER LMD	4.0	1.0
2014	DUT	2.0	0.0

situation

contrat Volontariat international

Année Diplôme

2013	DUT	0.0
	LICENCE PRO	0.0
	MASTER ENS	0.0
	MASTER LMD	1.0
2014	DUT	0.0

[5 rows x 24 columns]

```
[194]: # Définition de la palette des couleurs pour les catégories de contrat
colors = [
    "cornflowerblue", "mediumslateblue", "gold",
```

```

    "orange", "brown", "darksalmon",
    "peru", "mediumseagreen", "teal",
    "slategray", "green", "tan"
]

# Définition des catégories de contrat
categories = df_graph.columns.get_level_values("contrat").tolist()

# Map des couleurs selon les catégories de contrat
map_colors = dict(zip(categories, colors))

```

```

[195]: # Sélection des DUT
df_graph_dut = df_graph[df_graph.index.get_level_values("Diplôme") == "DUT"]

# Réordonnement des colonnes par rapport à leur valeur maximale
# valeur maximale du nombre de diplômé par situation
df_max_dut = df_graph_dut.max()
# liste des contrats triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_dut.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_dut.loc["30 mois après le diplôme"].sort_values().index
# réordonnement par situation des contrats dans la dataframe
df_graph_dut_18 = df_graph_dut.loc[:, "18 mois après le diplôme"][trie_18]
df_graph_dut_30 = df_graph_dut.loc[:, "30 mois après le diplôme"][trie_30]
#
df_graph_dut_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_dut_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les DUT
colors_dut_18 = [map_colors[categorie] for categorie in trie_18]
colors_dut_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[196]: df_graph_dut_18

```

```

[196]: contrat  CDI de chantier ou CDI de mission  Contrat spécifique au doctorat  \
Année
2013                0.0                0.0
2014                0.0                0.0
2015                0.0                0.0
2016                0.0                0.0
2017                0.0                0.0
2019                0.0                0.0

contrat  Intermittent du spectacle  Volontariat international  Vacataire  \
Année
2013                0.0                0.0                0.0
2014                0.0                0.0                0.0

```

2015	0.0	0.5	1.0
2016	0.0	0.0	0.0
2017	0.0	0.0	0.5
2019	0.0	0.2	0.2

```
contrat  Emplois aidés (Contrat Initiative Emploi...) \
Année
2013          0.5
2014          1.5
2015          0.5
2016          0.5
2017          0.0
2019          1.0
```

```
contrat  Contrat de professionnalisation \
Année
2013          0.5
2014          0.5
2015          1.0
2016          2.0
2017          0.5
2019          0.0
```

```
contrat  Prof. libérale, indépendant, chef d'entreprise  Fonctionnaire \
Année
2013          2.5          3.5
2014          1.5          4.0
2015          1.5          5.5
2016          1.5          3.5
2017          2.0          6.0
2019          1.8          4.8
```

```
contrat  Intérimaire    CDD    CDI
Année
2013          9.5    21.5    60.5
2014          12.5    23.5    55.0
2015          14.0    21.5    54.0
2016          13.0    21.0    57.5
2017          8.0     20.0    63.0
2019          9.6     18.2    63.0
```

```
[197]: # Sélection des licences professionnelles
df_graph_lp = df_graph[df_graph.index.get_level_values("Diplôme") == "LICENCE_
↳PRO"]

## Réordonnancement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation
```

```

df_max_lp = df_graph_lp.max()
# liste des contrats triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_lp.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_lp.loc["30 mois après le diplôme"].sort_values().index
# réordonnement par situation des contrats dans la dataframe
df_graph_lp_18 = df_graph_lp.loc[:, "18 mois après le diplôme"][trie_18]
df_graph_lp_30 = df_graph_lp.loc[:, "30 mois après le diplôme"][trie_30]
#
df_graph_lp_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_lp_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les licences
colors_lp_18 = [map_colors[categorie] for categorie in trie_18]
colors_lp_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[198]: # Sélection des masters ENS
df_graph_master_ens = df_graph[df_graph.index.get_level_values("Diplôme") == "MASTER ENS"]

## Réordonnement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation
df_max_master_ens = df_graph_master_ens.max()
# liste des contrats triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_master_ens.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_master_ens.loc["30 mois après le diplôme"].sort_values().index
# réordonnement par situation des contrats dans la dataframe
df_graph_master_ens_18 = df_graph_master_ens.loc[:, "18 mois après le_
↳diplôme"][trie_18]
df_graph_master_ens_30 = df_graph_master_ens.loc[:, "30 mois après le_
↳diplôme"][trie_30]
#
df_graph_master_ens_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_master_ens_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les master ENS
colors_master_ens_18 = [map_colors[categorie] for categorie in trie_18]
colors_master_ens_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[199]: # Sélection des masters LMD
df_graph_master_lmd = df_graph[df_graph.index.get_level_values("Diplôme") == "MASTER LMD"]

## Réordonnement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation
df_max_master_lmd = df_graph_master_lmd.max()
# liste des contrats triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_master_lmd.loc["18 mois après le diplôme"].sort_values().index

```

```

trie_30 = df_max_master_lmd.loc["30 mois après le diplôme"].sort_values().index
# réordonnement par situation des contrats dans la dataframe
df_graph_master_lmd_18 = df_graph_master_lmd.loc[:, "18 mois après le_
↳diplôme"][trie_18]
df_graph_master_lmd_30 = df_graph_master_lmd.loc[:, "30 mois après le_
↳diplôme"][trie_30]
#
df_graph_master_lmd_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_master_lmd_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les master LMD
colors_master_lmd_18 = [map_colors[categorie] for categorie in trie_18]
colors_master_lmd_30 = [map_colors[categorie] for categorie in trie_30]

```

```

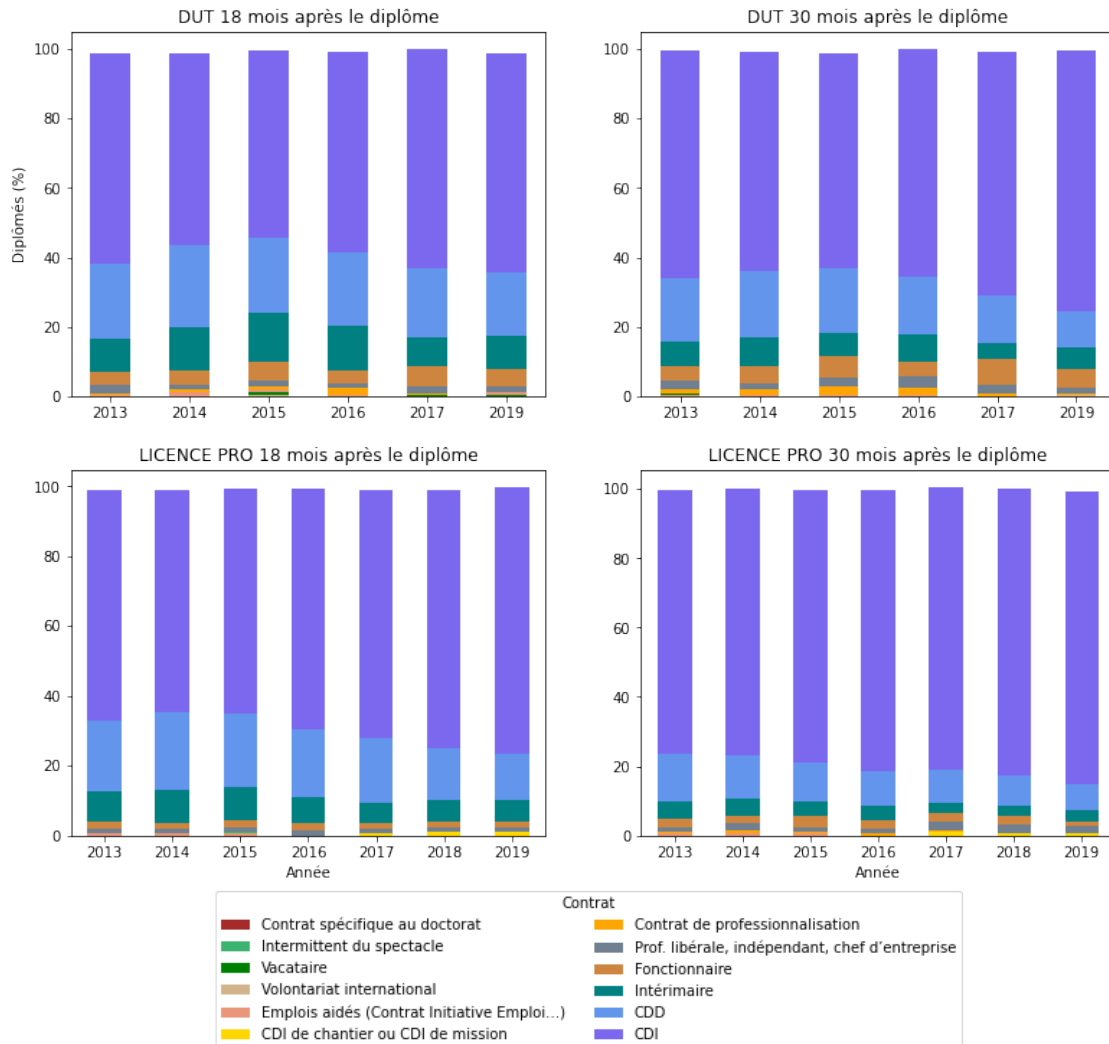
[200]: # Graphique du nombre de diplômés par type de contrat selon l'année
fig, (
    (ax1, ax2),
    (ax3, ax4),
    ) = plt.subplots(2, 2, figsize=(13, 10), sharex=False, sharey=False)
fig.suptitle("Types de contrat", fontsize=15)
df_graph_dut_18.plot(kind="bar", stacked=True, ax=ax1, color=colors_dut_18,
↳legend=False)
df_graph_dut_30.plot(kind="bar", stacked=True, ax=ax2, color=colors_dut_30,
↳legend=False)
df_graph_lp_18.plot(kind="bar", stacked=True, ax=ax3, color=colors_lp_18,
↳legend=False)
df_graph_lp_30.plot(kind="bar", stacked=True, ax=ax4, color=colors_lp_30,
↳legend=False)
# ajustement de l'espace entre le titre général et le 1er graph
plt.subplots_adjust(top=0.90)

ax1.set_title("DUT 18 mois après le diplôme")
ax2.set_title("DUT 30 mois après le diplôme")
ax3.set_title("LICENCE PRO 18 mois après le diplôme")
ax4.set_title("LICENCE PRO 30 mois après le diplôme")

ax1.set_xticklabels(df_graph_dut_18.index, rotation=0)
ax2.set_xticklabels(df_graph_dut_30.index, rotation=0)
ax3.set_xticklabels(df_graph_lp_18.index, rotation=0)
ax4.set_xticklabels(df_graph_lp_30.index, rotation=0)
ax1.set_ylabel("Diplômés (%)")
ax1.set_xlabel(None)
ax2.set_xlabel(None)
plt.legend(loc='lower center', title='Contrat', bbox_to_anchor=(-0.11, -0.6),
↳ncol=2)
plt.show()

```

Types de contrat



```
[201]: # Graphique du nombre de diplômés par type de contrat selon l'année
fig, (
    (ax1, ax2),
    (ax3, ax4)
) = plt.subplots(2, 2, figsize=(13, 10), sharex=False, sharey=False)
fig.suptitle("Types de contrat", fontsize=15)
df_graph_master_lmd_18.plot(kind="bar", stacked=True, ax=ax1,
    color=colors_master_lmd_18, legend=False)
df_graph_master_lmd_30.plot(kind="bar", stacked=True, ax=ax2,
    color=colors_master_lmd_30, legend=False)
df_graph_master_ens_18.plot(kind="bar", stacked=True, ax=ax3,
    color=colors_master_ens_18, legend=False)
```



```

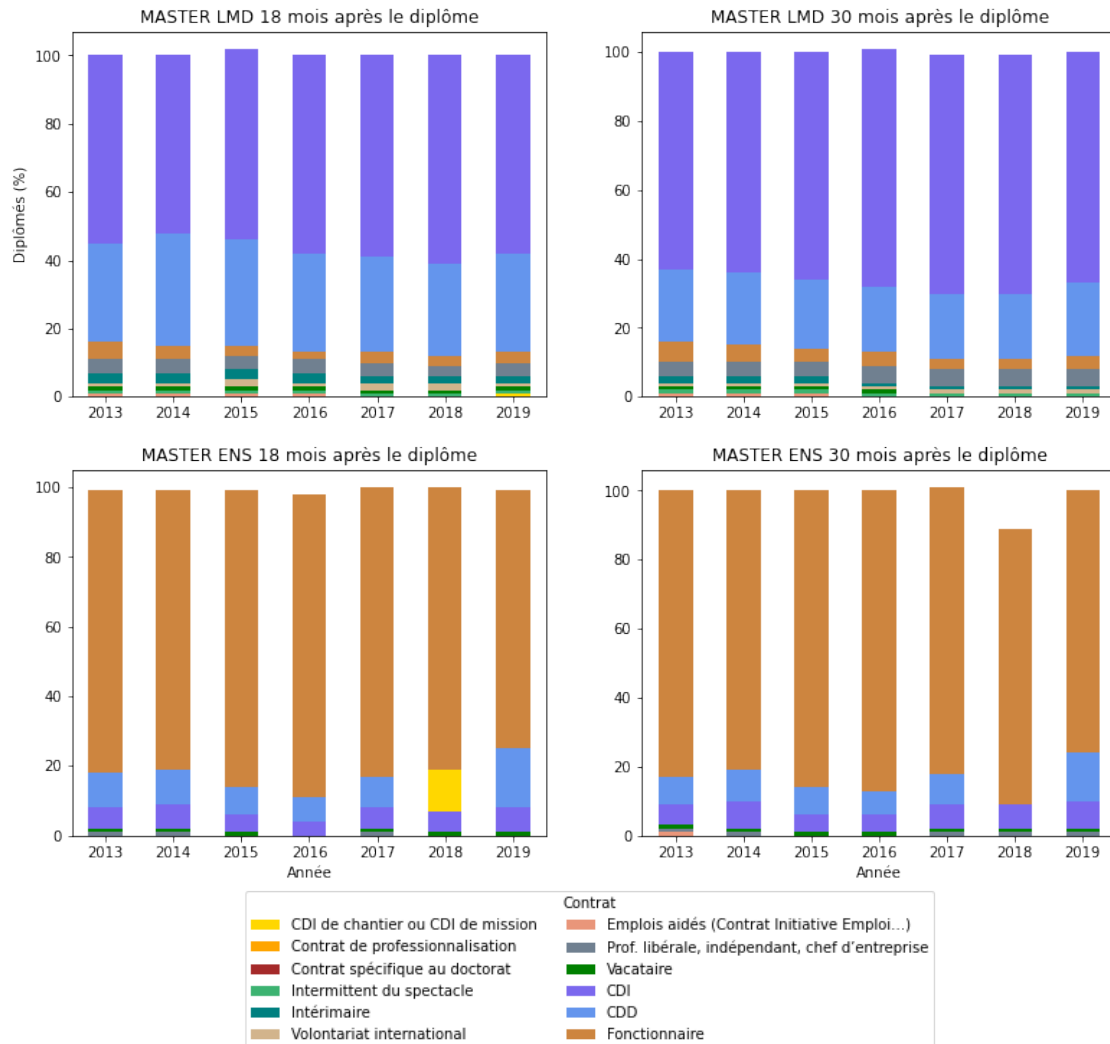
df_graph_master_ens_30.plot(kind="bar", stacked=True, ax=ax4,
    ↪color=colors_master_ens_30, legend=False)
# ajustement de l'espace entre le titre général et le 1er graph
plt.subplots_adjust(top=0.90)

ax1.set_title("MASTER LMD 18 mois après le diplôme")
ax2.set_title("MASTER LMD 30 mois après le diplôme")
ax3.set_title("MASTER ENS 18 mois après le diplôme")
ax4.set_title("MASTER ENS 30 mois après le diplôme")

ax1.set_xticklabels(df_graph_master_lmd_18.index, rotation=0)
ax2.set_xticklabels(df_graph_master_lmd_30.index, rotation=0)
ax3.set_xticklabels(df_graph_master_ens_18.index, rotation=0)
ax4.set_xticklabels(df_graph_master_ens_30.index, rotation=0)
ax1.set_ylabel("Diplômés (%)")
ax1.set_xlabel(None)
ax2.set_xlabel(None)
plt.legend(loc='lower center', title='Contrat', bbox_to_anchor=(-0.11, -0.6),
    ↪ncol=2)
plt.show()

```

Types de contrat



Proportion des professions d'activités par année selon le niveau de diplôme et la situation

```
[202]: # Sélection des professions
df_profession = df[
    ["Année", "Diplôme",
     "Discipline", "Genre",
     "situation", "Nombre de réponses"] + liste_professions].sort_values("Année")
```

```
[203]: df_profession.head()
```

```
[203]:
```

	Année	Diplôme	Discipline \
2948	2013	LICENCE PRO	Ensemble Licence professionnelle
47	2013	DUT	Autres sciences humaines et sociales

5594	2013	MASTER LMD	Autres sciences humaines et sociales
3052	2013	MASTER LMD	Autres formations juridiques, économiques et d..
3354	2013	MASTER LMD	Sciences de la vie et de la terre

	Genre	situation	Nombre de réponses \
2948	femmes et hommes	30 mois après le diplôme	16625.0
47	femmes et hommes	30 mois après le diplôme	126.0
5594	femmes et hommes	18 mois après le diplôme	NaN
3052	femmes et hommes	30 mois après le diplôme	NaN
3354	femmes et hommes	18 mois après le diplôme	NaN

	Agriculteur	Artisan, commerçant, chef d'entreprise \
2948	0.0	0.0
47	0.0	0.0
5594	0.0	0.0
3052	0.0	0.0
3354	0.0	0.0

	Profession libérale	Personnel de catégorie A de la fonction publique \
2948	0.0	1.0
47	0.0	4.0
5594	0.0	14.0
3052	0.0	10.0
3354	0.0	10.0

	Ingénieur, cadre, prof. libérales, prof. intellectuelles sup \
2948	10.0
47	5.0
5594	30.0
3052	49.0
3354	49.0

	Personnel de catégorie B de la fonction publique \
2948	3.0
47	15.0
5594	8.0
3052	5.0
3354	5.0

	Emploi de niveau intermédiaire : technicien, agent de maîtrise... \
2948	56.0
47	26.0
5594	23.0
3052	22.0
3354	25.0

	Personnel de catégorie C de la fonction publique	Manœuvre, ouvrier \
--	--	---------------------

2948	3.0	3.0
47	13.0	1.0
5594	5.0	1.0
3052	4.0	0.0
3354	2.0	2.0

	Employé de bureau, de commerce, personnel de service
2948	23.0
47	36.0
5594	18.0
3052	10.0
3354	8.0

```
[204]: # Sélection de l'ensemble des hommes et des femmes
profession_full_genre = df_profession[(df_profession["Genre"] == "femmes et_
↳hommes")]
```

```
[205]: # Sélection de l'ensemble des départements pour les différents diplômes
df_ensemble = profession_full_genre[
    profession_full_genre["Discipline"].str.contains(
        r"Ensemble des départements|Masters enseignement|Ensemble Masters_
↳LMD|Ensemble Licence professionnelle")]
```

```
[206]: # Pourcentage de contrats par année, selon le diplôme, la discipline et le type_
↳de contrat
df_ensemble = df_ensemble.melt(
    id_vars=["Année", "Diplôme", "Discipline", "Genre", "situation"],
    value_vars=liste_professions,
    var_name="profession",
    value_name="%")
```

```
[207]: df_ensemble.head()
```

```
[207]:  Année      Diplôme      Discipline      Genre \
0  2013  LICENCE PRO  Ensemble Licence professionnelle  femmes et hommes
1  2013              DUT  Ensemble des départements d'IUT  femmes et hommes
2  2013              DUT  Ensemble des départements d'IUT  femmes et hommes
3  2013              DUT  Ensemble des départements d'IUT  femmes et hommes
4  2013  LICENCE PRO  Ensemble Licence professionnelle  femmes et hommes
```

		situation	profession	%
0	30 mois après le diplôme	Agriculteur	0.0	
1	18 mois après le diplôme	Agriculteur	0.0	
2	18 mois après le diplôme	Agriculteur	0.0	
3	30 mois après le diplôme	Agriculteur	0.0	
4	18 mois après le diplôme	Agriculteur	0.0	

```
[208]: # Pourcentage de diplômés par type de profession pour chaque année
df_profession = df_ensemble.groupby(["Année", "profession"])["%"].mean()
df_profession.reset_index(inplace=True)
```

```
[209]: df_profession.head()
```

```
[209]:
```

	Année	profession	%
0	2013	Agriculteur	0.000000
1	2013	Artisan, commerçant, chef d'entreprise	0.000000
2	2013	Emploi de niveau intermédiaire : technicien, a...	44.083333
3	2013	Employé de bureau, de commerce, personnel de s...	14.916667
4	2013	Ingénieur, cadre, prof. libérales, prof. intel...	14.000000

```
[210]: df_graph = df_ensemble.pivot_table(
    index=["Année", "Diplôme"],
    columns=["situation", "profession"],
    values="%"
)
```

```
[211]: df_graph.head()
```

```
[211]:
```

	situation	18 mois après le diplôme \
	profession	Agriculteur
Année Diplôme		
2013	DUT	0.0
	LICENCE PRO	0.0
	MASTER ENS	0.0
	MASTER LMD	0.0
2014	DUT	0.0

	situation	\
	profession	Artisan, commerçant, chef d'entreprise
Année Diplôme		
2013	DUT	0.0
	LICENCE PRO	0.0
	MASTER ENS	0.0
	MASTER LMD	0.0
2014	DUT	0.0

	situation	\
	profession	Emploi de niveau intermédiaire : technicien, agent de maîtrise...
Année Diplôme		
2013	DUT	54.5
	LICENCE PRO	66.0
	MASTER ENS	2.0

	MASTER LMD	22.0
2014	DUT	55.5

situation \

profession Employé de bureau, de commerce, personnel de service

Année Diplôme

2013	DUT	22.5
------	-----	------

	LICENCE PRO	15.5
--	-------------	------

	MASTER ENS	4.0
--	------------	-----

	MASTER LMD	14.0
--	------------	------

2014	DUT	21.0
------	-----	------

situation

\

profession Ingénieur, cadre, prof. libérales, prof. intellectuelles sup

Année Diplôme

2013	DUT	6.5
------	-----	-----

	LICENCE PRO	7.0
--	-------------	-----

	MASTER ENS	4.0
--	------------	-----

	MASTER LMD	48.0
--	------------	------

2014	DUT	4.0
------	-----	-----

situation

\

profession Manœuvre, ouvrier

Année Diplôme

2013	DUT	8.5
------	-----	-----

	LICENCE PRO	4.5
--	-------------	-----

	MASTER ENS	0.0
--	------------	-----

	MASTER LMD	1.0
--	------------	-----

2014	DUT	11.0
------	-----	------

situation

\

profession Personnel de catégorie A de la fonction publique

Année Diplôme

2013	DUT	1.5
------	-----	-----

	LICENCE PRO	1.0
--	-------------	-----

	MASTER ENS	81.0
--	------------	------

	MASTER LMD	9.0
--	------------	-----

2014	DUT	1.0
------	-----	-----

situation

\

profession Personnel de catégorie B de la fonction publique

Année Diplôme

2013	DUT	4.0
------	-----	-----

	LICENCE PRO	3.5
--	-------------	-----

	MASTER ENS	5.0
--	------------	-----

	MASTER LMD	3.0
--	------------	-----

2014	DUT	3.5
------	-----	-----

situation		\
-----------	--	---

profession	Personnel de catégorie C de la fonction publique	
------------	--	--

Année Diplôme	
---------------	--

2013	DUT	3.5
------	-----	-----

LICENCE PRO	2.0
-------------	-----

MASTER ENS	3.0
------------	-----

MASTER LMD	2.0
------------	-----

2014	DUT	4.0
------	-----	-----

situation	30 mois après le diplôme	\
-----------	--------------------------	---

profession	Profession libérale	Agriculteur
------------	---------------------	-------------

Année Diplôme	
---------------	--

2013	DUT	0.0	0.0
------	-----	-----	-----

LICENCE PRO	0.0	0.0
-------------	-----	-----

MASTER ENS	0.0	0.0
------------	-----	-----

MASTER LMD	0.0	0.0
------------	-----	-----

2014	DUT	0.0	0.0
------	-----	-----	-----

situation		\
-----------	--	---

profession	Artisan, commerçant, chef d'entreprise
------------	--

Année Diplôme	
---------------	--

2013	DUT	0.0
------	-----	-----

LICENCE PRO	0.0
-------------	-----

MASTER ENS	0.0
------------	-----

MASTER LMD	0.0
------------	-----

2014	DUT	0.0
------	-----	-----

situation	
-----------	--

\

profession	Emploi de niveau intermédiaire : technicien, agent de maîtrise...
------------	---

Année Diplôme	
---------------	--

2013	DUT	54.0
------	-----	------

LICENCE PRO	66.0
-------------	------

MASTER ENS	3.0
------------	-----

MASTER LMD	21.0
------------	------

2014	DUT	60.0
------	-----	------

situation		\
-----------	--	---

profession	Employé de bureau, de commerce, personnel de service
------------	--

Année Diplôme	
---------------	--

2013	DUT	20.5
------	-----	------

LICENCE PRO	14.5
-------------	------

MASTER ENS	3.0
------------	-----

MASTER LMD	12.0
------------	------

2014	DUT	19.0
------	-----	------

situation \

profession	Ingénieur, cadre, prof. libérales, prof. intellectuelles sup
Année Diplôme	
2013	DUT 7.5
	LICENCE PRO 9.5
	MASTER ENS 4.0
	MASTER LMD 51.0
2014	DUT 5.5

situation \

profession	Manœuvre, ouvrier
Année Diplôme	
2013	DUT 8.5
	LICENCE PRO 3.5
	MASTER ENS 0.0
	MASTER LMD 1.0
2014	DUT 6.5

situation \

profession	Personnel de catégorie A de la fonction publique
Année Diplôme	
2013	DUT 1.0
	LICENCE PRO 1.0
	MASTER ENS 82.0
	MASTER LMD 9.0
2014	DUT 1.0

situation \

profession	Personnel de catégorie B de la fonction publique
Année Diplôme	
2013	DUT 5.0
	LICENCE PRO 2.5
	MASTER ENS 5.0
	MASTER LMD 4.0
2014	DUT 3.5

situation \

profession	Personnel de catégorie C de la fonction publique
Année Diplôme	
2013	DUT 3.5
	LICENCE PRO 2.0
	MASTER ENS 3.0
	MASTER LMD 2.0
2014	DUT 4.0

situation	profession	Profession libérale
Année	Diplôme	
2013	DUT	0.0
	LICENCE PRO	0.0
	MASTER ENS	0.0
	MASTER LMD	0.0
2014	DUT	0.0

```
[212]: # Définition de la palette des couleurs pour les catégories de profession
colors = [
    "cornflowerblue", "mediumslateblue", "gold",
    "orange", "brown", "darksalmon",
    "peru", "mediumseagreen", "teal",
    "slategray"
]

# Définition des catégories de profession
categories = df_graph.columns.get_level_values("profession").tolist()

# Map des couleurs selon les catégories de profession
map_colors = dict(zip(categories, colors))
```

```
[213]: # Sélection des DUT
df_graph_dut = df_graph[df_graph.index.get_level_values("Diplôme") == "DUT"]

# Réordonnement des colonnes par rapport à leur valeur maximale
# valeur maximale du nombre de diplômé par situation
df_max_dut = df_graph_dut.max()
# liste des contrats triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_dut.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_dut.loc["30 mois après le diplôme"].sort_values().index
# réordonnement par situation des contrats dans la dataframe
df_graph_dut_18 = df_graph_dut.loc[:, "18 mois après le diplôme"][trie_18]
df_graph_dut_30 = df_graph_dut.loc[:, "30 mois après le diplôme"][trie_30]
#
df_graph_dut_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_dut_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les DUT
colors_dut_18 = [map_colors[categorie] for categorie in trie_18]
colors_dut_30 = [map_colors[categorie] for categorie in trie_30]
```

```
[214]: # Sélection des licences professionnelles
df_graph_lp = df_graph[df_graph.index.get_level_values("Diplôme") == "LICENCE_
↳PRO"]
```

```

## Réordonnement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation
df_max_lp = df_graph_lp.max()
# liste des contrats triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_lp.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_lp.loc["30 mois après le diplôme"].sort_values().index
# réordonnement par situation des contrats dans la dataframe
df_graph_lp_18 = df_graph_lp.loc[:, "18 mois après le diplôme"][trie_18]
df_graph_lp_30 = df_graph_lp.loc[:, "30 mois après le diplôme"][trie_30]
#
df_graph_lp_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_lp_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les licences
colors_lp_18 = [map_colors[categorie] for categorie in trie_18]
colors_lp_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[215]: # Sélection des masters ENS
df_graph_master_ens = df_graph[df_graph.index.get_level_values("Diplôme") == "MASTER ENS"]

## Réordonnement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation
df_max_master_ens = df_graph_master_ens.max()
# liste des contrats triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_master_ens.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_master_ens.loc["30 mois après le diplôme"].sort_values().index
# réordonnement par situation des contrats dans la dataframe
df_graph_master_ens_18 = df_graph_master_ens.loc[:, "18 mois après le_
↳diplôme"][trie_18]
df_graph_master_ens_30 = df_graph_master_ens.loc[:, "30 mois après le_
↳diplôme"][trie_30]
#
df_graph_master_ens_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_master_ens_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les master ENS
colors_master_ens_18 = [map_colors[categorie] for categorie in trie_18]
colors_master_ens_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[216]: # Sélection des masters LMD
df_graph_master_lmd = df_graph[df_graph.index.get_level_values("Diplôme") == "MASTER LMD"]

## Réordonnement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation

```

```

df_max_master_lmd = df_graph_master_lmd.max()
# liste des contrats triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_master_lmd.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_master_lmd.loc["30 mois après le diplôme"].sort_values().index
# réordonnancement par situation des contrats dans la dataframe
df_graph_master_lmd_18 = df_graph_master_lmd.loc[:, "18 mois après le_
↳diplôme"][trie_18]
df_graph_master_lmd_30 = df_graph_master_lmd.loc[:, "30 mois après le_
↳diplôme"][trie_30]
#
df_graph_master_lmd_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_master_lmd_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnancement du map des couleurs pour les master LMD
colors_master_lmd_18 = [map_colors[categorie] for categorie in trie_18]
colors_master_lmd_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[217]: # Graphique du nombre de diplômés par type de contrat selon l'année
fig, (
    (ax1, ax2),
    (ax3, ax4),
    ) = plt.subplots(2, 2, figsize=(13, 10), sharex=False, sharey=False)
fig.suptitle("Professions", fontsize=15)
df_graph_dut_18.plot(kind="bar", stacked=True, ax=ax1, color=colors_dut_18,
↳legend=False)
df_graph_dut_30.plot(kind="bar", stacked=True, ax=ax2, color=colors_dut_30,
↳legend=False)
df_graph_lp_18.plot(kind="bar", stacked=True, ax=ax3, color=colors_lp_18,
↳legend=False)
df_graph_lp_30.plot(kind="bar", stacked=True, ax=ax4, color=colors_lp_30,
↳legend=False)
# ajustement de l'espace entre le titre général et le 1er graph
plt.subplots_adjust(top=0.90)

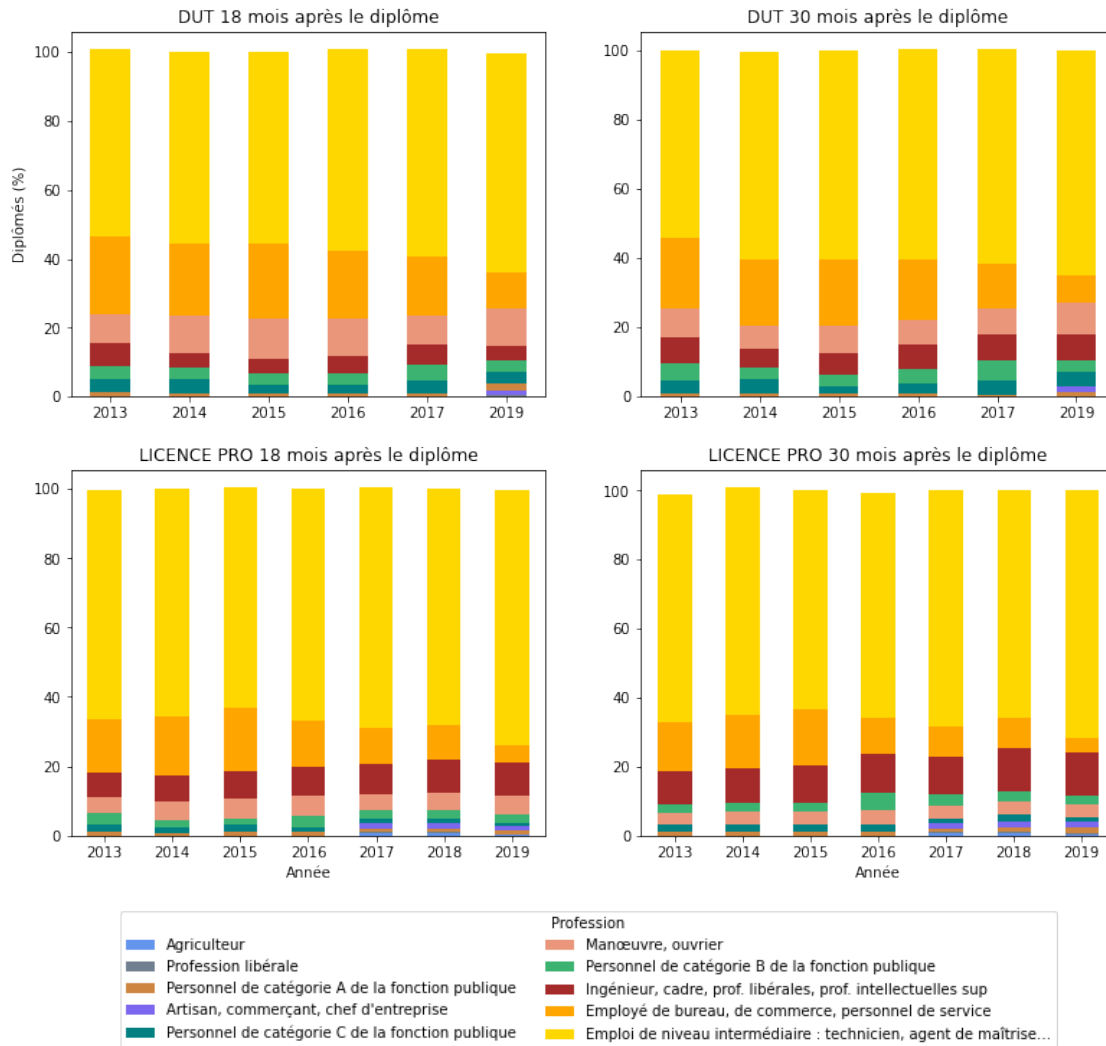
ax1.set_title("DUT 18 mois après le diplôme")
ax2.set_title("DUT 30 mois après le diplôme")
ax3.set_title("LICENCE PRO 18 mois après le diplôme")
ax4.set_title("LICENCE PRO 30 mois après le diplôme")

ax1.set_xticklabels(df_graph_dut_18.index, rotation=0)
ax2.set_xticklabels(df_graph_dut_30.index, rotation=0)
ax3.set_xticklabels(df_graph_lp_18.index, rotation=0)
ax4.set_xticklabels(df_graph_lp_30.index, rotation=0)
ax1.set_ylabel("Diplômés (%)")
ax1.set_xlabel(None)
ax2.set_xlabel(None)

```

```
plt.legend(loc='lower center', title='Profession', bbox_to_anchor=(-0.11, -0.
↪6), ncol=2)
plt.show()
```

Professions



```
[218]: # Graphique du nombre de diplômés par type de contrat selon l'année
fig, (
    (ax1, ax2),
    (ax3, ax4)
) = plt.subplots(2, 2, figsize=(13, 10), sharex=False, sharey=False)
fig.suptitle("Professions", fontsize=15)
df_graph_master_lmd_18.plot(kind="bar", stacked=True, ax=ax1,
↪color=colors_master_lmd_18, legend=False)
```

```

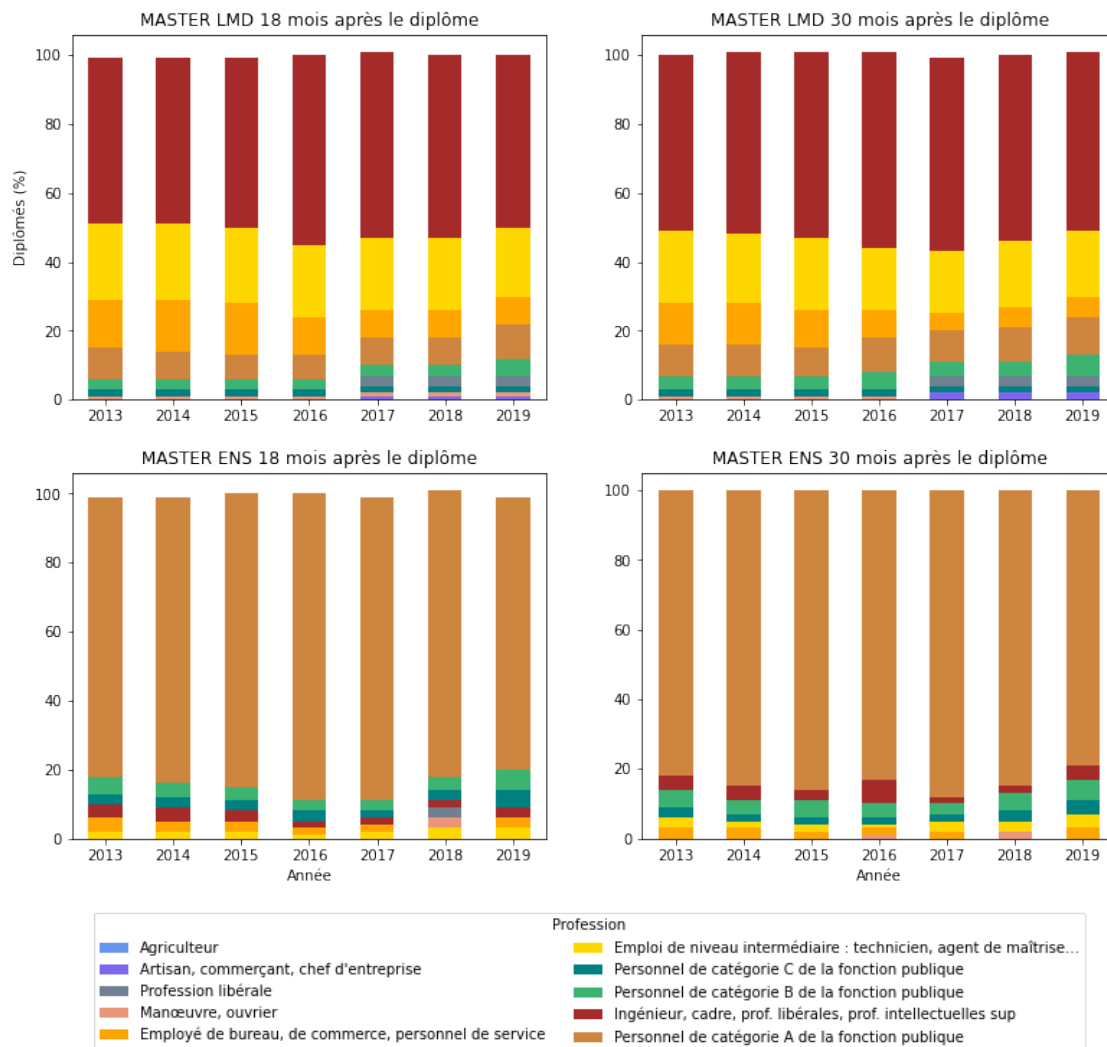
df_graph_master_lmd_30.plot(kind="bar", stacked=True, ax=ax2,
    ↪color=colors_master_lmd_30, legend=False)
df_graph_master_ens_18.plot(kind="bar", stacked=True, ax=ax3,
    ↪color=colors_master_ens_18, legend=False)
df_graph_master_ens_30.plot(kind="bar", stacked=True, ax=ax4,
    ↪color=colors_master_ens_30, legend=False)
# ajustement de l'espace entre le titre général et le 1er graph
plt.subplots_adjust(top=0.90)

#ax2.legend(title="Profession", bbox_to_anchor=(1, 1.02))
ax1.set_title("MASTER LMD 18 mois après le diplôme")
ax2.set_title("MASTER LMD 30 mois après le diplôme")
ax3.set_title("MASTER ENS 18 mois après le diplôme")
ax4.set_title("MASTER ENS 30 mois après le diplôme")

ax1.set_xticklabels(df_graph_master_lmd_18.index, rotation=0)
ax2.set_xticklabels(df_graph_master_lmd_30.index, rotation=0)
ax3.set_xticklabels(df_graph_master_ens_18.index, rotation=0)
ax4.set_xticklabels(df_graph_master_ens_30.index, rotation=0)
ax1.set_ylabel("Diplômés (%)")
ax1.set_xlabel(None)
ax2.set_xlabel(None)
plt.legend(loc='lower center', title='Profession', bbox_to_anchor=(-0.11, -0.
    ↪6), ncol=2)
plt.show()

```

Professions



Répartition des différents secteurs par année selon le niveau de diplôme et la situation

```
[219]: df_secteur = df[
    ["Année", "Diplôme",
     "Discipline", "Genre",
     "situation", "Nombre de réponses"] + liste_secteurs].sort_values("Année")
```

```
[220]: # Sélection de l'ensemble des hommes et des femmes
secteur_full_genre = df_secteur[(df_secteur["Genre"] == "femmes et hommes")]

# Sélection de l'ensemble des départements pour les différents diplômes
df_ensemble = secteur_full_genre[
    profession_full_genre["Discipline"].str.contains(
```

```
r"Ensemble des départements|Masters enseignement|Ensemble Masters_
↳LMD|Ensemble Licence professionnelle"]]
```

```
[221]: df_ensemble.head()
```

```
[221]:
```

	Année	Diplôme	Discipline	Genre \
2948	2013	LICENCE PRO	Ensemble Licence professionnelle	femmes et hommes
56	2013	DUT	Ensemble des départements d'IUT	femmes et hommes
769	2013	DUT	Ensemble des départements d'IUT	femmes et hommes
770	2013	DUT	Ensemble des départements d'IUT	femmes et hommes
1423	2013	LICENCE PRO	Ensemble Licence professionnelle	femmes et hommes

	situation	Nombre de réponses \
2948	30 mois après le diplôme	16625.0
56	18 mois après le diplôme	2054.0
769	18 mois après le diplôme	456.0
770	30 mois après le diplôme	456.0
1423	18 mois après le diplôme	16625.0

	Agriculture, sylviculture et pêche \
2948	4.0
56	0.0
769	0.0
770	2.0
1423	0.0

	Industries (manufacturières, extractives et autres)	Construction \
2948	20.0	7.0
56	0.0	0.0
769	0.0	0.0
770	47.0	14.0
1423	0.0	0.0

	Activités immobilières \
2948	0.0
56	0.0
769	0.0
770	0.0
1423	0.0

	Commerce, transports, héberg-ement et restauration \
2948	17.0
56	0.0
769	0.0
770	10.0
1423	0.0

	Information et communication	Activités financières et d'assurance	\
2948	6.0	10.0	
56	0.0	0.0	
769	0.0	0.0	
770	3.0	0.0	
1423	0.0	0.0	

	Activités spécialisées, scientifiques et techniques	\
2948	11.0	
56	0.0	
769	0.0	
770	8.0	
1423	0.0	

	Activités de services administratifs et de soutien	Enseignement	\
2948	3.0	2.0	
56	0.0	0.0	
769	0.0	0.0	
770	2.0	2.0	
1423	0.0	0.0	

	Administration publique (hors ens.)	Santé humaine et action sociale	\
2948	5.0	5.0	
56	0.0	0.0	
769	0.0	0.0	
770	4.0	0.0	
1423	0.0	0.0	

	Arts, spectacles et activités récréatives	Autres activités de service
2948	2.0	8.0
56	0.0	0.0
769	0.0	0.0
770	2.0	6.0
1423	0.0	0.0

```
[222]: df_ensemble = df_ensemble.melt(id_vars=[
    "Année",
    "Diplôme",
    "Discipline",
    "Genre",
    "situation"
], value_vars=liste_secteurs, var_name="secteur", value_name="%")
```

```
[223]: df_graph = df_ensemble.pivot_table(index=[
    "Année",
    "Diplôme"], columns=[
    "situation",
```



```
"secteur"
], values="%")
```

```
[224]: df_graph.head()
```

```
[224]: situation                                18 mois après le diplôme \
secteur                Activités de services administratifs et de soutien
Année Diplôme
2013  DUT                                                    0.0
      LICENCE PRO                                              0.0
      MASTER ENS                                               0.0
      MASTER LMD                                               0.0
2014  DUT                                                    0.0

situation                                \
secteur                Activités financières et d'assurance Activités immobilières
Année Diplôme
2013  DUT                                                    0.0
      LICENCE PRO                                              0.0
      MASTER ENS                                               0.0
      MASTER LMD                                               0.0
2014  DUT                                                    0.0

situation                                \
secteur                Activités spécialisées, scientifiques et techniques
Année Diplôme
2013  DUT                                                    0.0
      LICENCE PRO                                              0.0
      MASTER ENS                                               0.0
      MASTER LMD                                               0.0
2014  DUT                                                    0.0

situation                                \
secteur                Administration publique (hors ens.)
Année Diplôme
2013  DUT                                                    0.0
      LICENCE PRO                                              0.0
      MASTER ENS                                               0.0
      MASTER LMD                                               0.0
2014  DUT                                                    0.0

situation                                \
secteur                Agriculture, sylviculture et pêche
Année Diplôme
2013  DUT                                                    0.0
      LICENCE PRO                                              0.0
      MASTER ENS                                               0.0
```

	MASTER LMD	0.0
2014	DUT	0.0

situation			\
secteur	Arts, spectacles et activités récréatives		
Année Diplôme			
2013	DUT	0.0	
	LICENCE PRO	0.0	
	MASTER ENS	0.0	
	MASTER LMD	0.0	
2014	DUT	0.0	

situation			\
secteur	Autres activités de service		
Année Diplôme			
2013	DUT	0.0	
	LICENCE PRO	0.0	
	MASTER ENS	0.0	
	MASTER LMD	0.0	
2014	DUT	0.0	

situation			\
secteur	Commerce, transports, héberg-ement et restauration		
Année Diplôme			
2013	DUT	0.0	
	LICENCE PRO	0.0	
	MASTER ENS	0.0	
	MASTER LMD	0.0	
2014	DUT	0.0	

situation		...	30 mois après le diplôme	\
secteur	Construction	...	Administration publique (hors ens.)	
Année Diplôme				
2013	DUT	0.0	...	6.0
	LICENCE PRO	0.0	...	3.5
	MASTER ENS	0.0	...	1.0
	MASTER LMD	0.0	...	8.0
2014	DUT	0.0	...	9.0

situation			\
secteur	Agriculture, sylviculture et pêche		
Année Diplôme			
2013	DUT	2.0	
	LICENCE PRO	3.0	
	MASTER ENS	0.0	
	MASTER LMD	2.0	
2014	DUT	1.0	

situation				\
secteur	Arts, spectacles et activités récréatives			
Année Diplôme				
2013	DUT		2.0	
	LICENCE PRO		1.0	
	MASTER ENS		1.0	
	MASTER LMD		4.0	
2014	DUT		1.0	

situation				\
secteur	Autres activités de service			
Année Diplôme				
2013	DUT	7.0		
	LICENCE PRO	7.0		
	MASTER ENS	1.0		
	MASTER LMD	9.0		
2014	DUT	6.5		

situation				\
secteur	Commerce, transports, héberg-ement et restauration			
Année Diplôme				
2013	DUT		16.5	
	LICENCE PRO		13.0	
	MASTER ENS		1.0	
	MASTER LMD		11.0	
2014	DUT		17.5	

situation				\
secteur	Construction Enseignement			
Année Diplôme				
2013	DUT	9.0	2.5	
	LICENCE PRO	12.5	1.5	
	MASTER ENS	0.0	93.0	
	MASTER LMD	3.0	8.0	
2014	DUT	9.5	2.5	

situation				\
secteur	Industries (manufacturières, extractives et autres)			
Année Diplôme				
2013	DUT		33.5	
	LICENCE PRO		29.5	
	MASTER ENS		0.0	
	MASTER LMD		11.0	
2014	DUT		29.0	

situation

secteur	Information et communication Santé humaine et action sociale	
Année Diplôme		
2013 DUT	6.0	3.5
LICENCE PRO	3.5	3.5
MASTER ENS	0.0	1.0
MASTER LMD	9.0	9.0
2014 DUT	7.0	4.5

[5 rows x 28 columns]

```
[225]: # Définition de la palette des couleurs pour les secteurs
colors = distinctipy.get_colors(len(liste_secteurs))

# Définition des catégories des secteurs
categories = df_graph.columns.get_level_values("secteur").tolist()

# Map des couleurs selon les secteurs
map_colors = dict(zip(categories, colors))

[226]: # Sélection des DUT
df_graph_dut = df_graph[df_graph.index.get_level_values("Diplôme") == "DUT"]

# Réordonnement des colonnes par rapport à leur valeur maximale
# valeur maximale du nombre de diplômé par situation
df_max_dut = df_graph_dut.max()
# liste des secteurs triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_dut.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_dut.loc["30 mois après le diplôme"].sort_values().index
# réordonnement par situation des contrats dans la dataframe
df_graph_dut_18 = df_graph_dut.loc[:, "18 mois après le diplôme"][trie_18]
df_graph_dut_30 = df_graph_dut.loc[:, "30 mois après le diplôme"][trie_30]
#
df_graph_dut_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_dut_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les DUT
colors_dut_18 = [map_colors[categorie] for categorie in trie_18]
colors_dut_30 = [map_colors[categorie] for categorie in trie_30]

[227]: # Sélection des licences professionnelles
df_graph_lp = df_graph[df_graph.index.get_level_values("Diplôme") == "LICENCE_
↳PRO"]

## Réordonnement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation
df_max_lp = df_graph_lp.max()
# liste des secteurs triée 18 et 30 mois après l'obtention du diplôme
```

```

trie_18 = df_max_lp.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_lp.loc["30 mois après le diplôme"].sort_values().index
# réordonnancement par situation des secteurs dans la dataframe
df_graph_lp_18 = df_graph_lp.loc[:, "18 mois après le diplôme"][trie_18]
df_graph_lp_30 = df_graph_lp.loc[:, "30 mois après le diplôme"][trie_30]
#
df_graph_lp_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_lp_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnancement du map des couleurs pour les licences
colors_lp_18 = [map_colors[categorie] for categorie in trie_18]
colors_lp_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[228]: # Sélection des masters ENS
df_graph_master_ens = df_graph[df_graph.index.get_level_values("Diplôme") == ↵
↵ "MASTER ENS"]

## Réordonnancement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation
df_max_master_ens = df_graph_master_ens.max()
# liste des secteurs triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_master_ens.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_master_ens.loc["30 mois après le diplôme"].sort_values().index
# réordonnancement par situation des secteurs dans la dataframe
df_graph_master_ens_18 = df_graph_master_ens.loc[:, "18 mois après le ↵
↵ diplôme"][trie_18]
df_graph_master_ens_30 = df_graph_master_ens.loc[:, "30 mois après le ↵
↵ diplôme"][trie_30]
#
df_graph_master_ens_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_master_ens_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnancement du map des couleurs pour les master ENS
colors_master_ens_18 = [map_colors[categorie] for categorie in trie_18]
colors_master_ens_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[229]: # Sélection des masters LMD
df_graph_master_lmd = df_graph[df_graph.index.get_level_values("Diplôme") == ↵
↵ "MASTER LMD"]

## Réordonnancement des colonnes par rapport à leur valeur maximale ##
# valeur maximale du nombre de diplômé par situation
df_max_master_lmd = df_graph_master_lmd.max()
# liste des secteurs triée 18 et 30 mois après l'obtention du diplôme
trie_18 = df_max_master_lmd.loc["18 mois après le diplôme"].sort_values().index
trie_30 = df_max_master_lmd.loc["30 mois après le diplôme"].sort_values().index
# réordonnancement par situation des secteurs dans la dataframe

```

```

df_graph_master_lmd_18 = df_graph_master_lmd.loc[:, "18 mois après le_
↳diplôme"][trie_18]
df_graph_master_lmd_30 = df_graph_master_lmd.loc[:, "30 mois après le_
↳diplôme"][trie_30]
#
df_graph_master_lmd_18.reset_index("Diplôme", drop=True, inplace=True)
df_graph_master_lmd_30.reset_index("Diplôme", drop=True, inplace=True)

# Réordonnement du map des couleurs pour les master LMD
colors_master_lmd_18 = [map_colors[categorie] for categorie in trie_18]
colors_master_lmd_30 = [map_colors[categorie] for categorie in trie_30]

```

```

[230]: # Graphique du nombre de diplômés par secteur selon l'année
# Les données sont manquantes après 18 mois
fig, (
    (ax1, ax2),
    (ax3, ax4),
    ) = plt.subplots(2, 2, figsize=(13, 10), sharex=False, sharey=False)
fig.suptitle("Secteurs d'activité", fontsize=15)
df_graph_dut_30.plot(kind="bar", stacked=True, ax=ax1, color=colors_dut_30,
↳legend=False)
df_graph_lp_30.plot(kind="bar", stacked=True, ax=ax2, color=colors_lp_30,
↳legend=False)
df_graph_master_lmd_30.plot(kind="bar", stacked=True, ax=ax3,
↳color=colors_master_lmd_30, legend=False)
df_graph_master_ens_30.plot(kind="bar", stacked=True, ax=ax4,
↳color=colors_master_ens_30, legend=False)
plt.subplots_adjust(top=0.90)

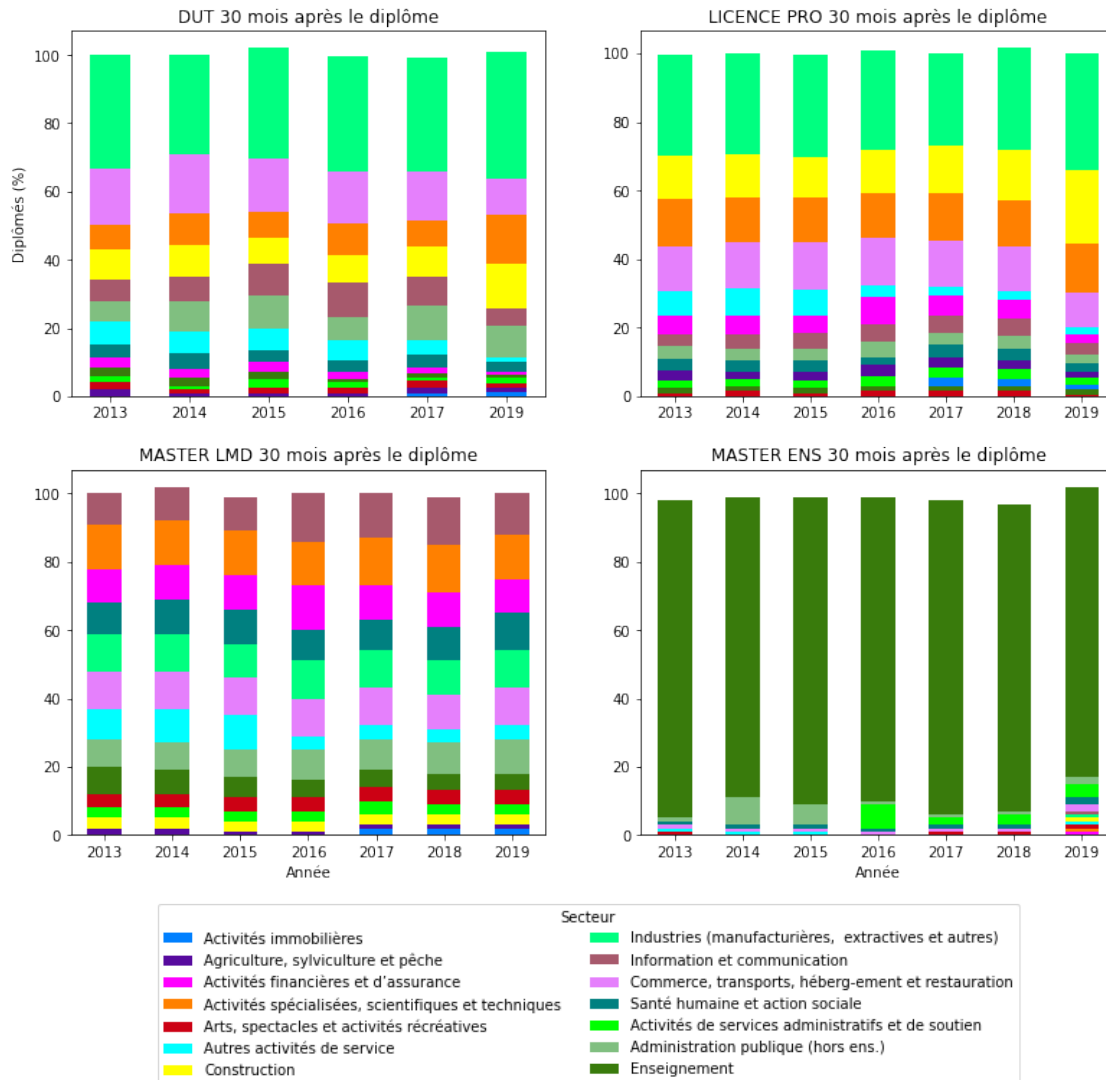
ax1.set_title("DUT 30 mois après le diplôme")
ax2.set_title("LICENCE PRO 30 mois après le diplôme")
ax3.set_title("MASTER LMD 30 mois après le diplôme")
ax4.set_title("MASTER ENS 30 mois après le diplôme")

#ax2.legend(title="Secteur", bbox_to_anchor=(1, 1.02))
ax1.set_xticklabels(df_graph_dut_30.index, rotation=0)
ax2.set_xticklabels(df_graph_lp_30.index, rotation=0)
ax3.set_xticklabels(df_graph_master_lmd_30.index, rotation=0)
ax4.set_xticklabels(df_graph_master_ens_30.index, rotation=0)
ax1.set_ylabel("Diplômés (%)")

ax1.set_xlabel(None)
ax2.set_xlabel(None)
plt.legend(loc='lower center', title='Secteur', bbox_to_anchor=(-0.11, -0.7),
↳ncol=2)
plt.show()

```

Secteurs d'activité



Statistiques descriptives

```
[231]: # Selection des variables à analyser
selection = [
    "Année", "Genre",
    "Nombre de réponses", "Diplôme",
    "situation", "Discipline",
    "Salaire net mensuel médian des emplois à temps plein",
    "Premier quartile des salaires nets mensuels des emplois à temps plein",
    "Dernier quartile des salaires nets mensuels des emplois à temps plein",
    "Salaire net mensuel médian national",
    "Taux d'insertion"]
```

```
[232]: # Initialisation des valeurs manquantes
df_stat_na = ins_prof_na[selection]
df_stat_na = df_stat_na.replace(["ns", "nd"], [np.NaN, np.NaN])
# Taux de valeur manquantes
taux_na = df_stat_na.isna().sum()
taux_na.name = "NA"
taux_na = pd.DataFrame(taux_na)
# Nombre total de valeur par variable
total = df_stat_na.shape[0]
taux_na["% NA"] = round(taux_na["NA"] / total * 100)
```

```
[233]: # Suppression des ensembles par diplôme et selection par genre
df_stat = ins_prof[selection]
df_stat = df_stat[
    (~df_stat.loc[:, "Discipline"].str.contains(r"^[Ensemble|Autres]")) &
    (~df_stat.loc[:, "Genre"].str.contains(r"^femmes et hommes")) &
    (df_stat["Année"] > '2012')]
df_stat.sort_values(["Année", "Diplôme"], inplace=True)
```

```
[234]: # liste des disciplines
df_stat["Discipline"].unique()
```

```
[234]: array(['Droit', 'Information communication',
        'Sciences de la vie et de la terre', 'Gestion',
        'Sciences fondamentales', 'Informatique', 'Lettres, langues, arts',
        'Histoire-géographie', 'Psychologie', 'Économie',
        'Masters enseignement', "Sciences de l'ingénieur"], dtype=object)
```

```
[235]: # Renommage des sciences de l'ingénieur
df_stat.loc[df_stat.loc[:, "Discipline"] == "Sciences de l'ingénieur",
    ↪ "Discipline"] = "Sciences de l'ingénieur"
```

```
[236]: df_stat.head()
```

```
[236]:
```

	Année	Genre	Nombre de réponses	Diplôme	situation \
0	2013	hommes	5.0	DUT	30 mois après le diplôme
10	2013	hommes	22.0	DUT	18 mois après le diplôme
113	2013	femmes	130.0	DUT	18 mois après le diplôme
163	2013	hommes	58.0	DUT	18 mois après le diplôme
187	2013	femmes	7.0	DUT	30 mois après le diplôme

	Discipline \
0	Droit
10	Information communication
113	Sciences de la vie et de la terre
163	Gestion
187	Sciences de la vie et de la terre

	Salaire net mensuel médian des emplois à temps plein \
0	0.0
10	0.0
113	1400.0
163	1500.0
187	0.0

	Premier quartile des salaires nets mensuels des emplois à temps plein \
0	0.0
10	0.0
113	1300.0
163	0.0
187	0.0

	Dernier quartile des salaires nets mensuels des emplois à temps plein \
0	0.0
10	0.0
113	1510.0
163	0.0
187	0.0

	Salaire net mensuel médian national	Taux d'insertion
0	1870	0.0
10	1870	0.0
113	1870	88.0
163	1870	70.0
187	1870	0.0

```
[237]: # Nombre de diplômés selon le genre
df_distrib_genre = df_stat.groupby(["Genre"])[["Nombre de réponses"]].sum()

# Proportion en %
total = df_distrib_genre["Nombre de réponses"].sum()
df_distrib_genre["%"] = df_distrib_genre["Nombre de réponses"] / total * 100

df_distrib_genre.reset_index(inplace=True)
```

```
[238]: df_distrib_genre.head()
```

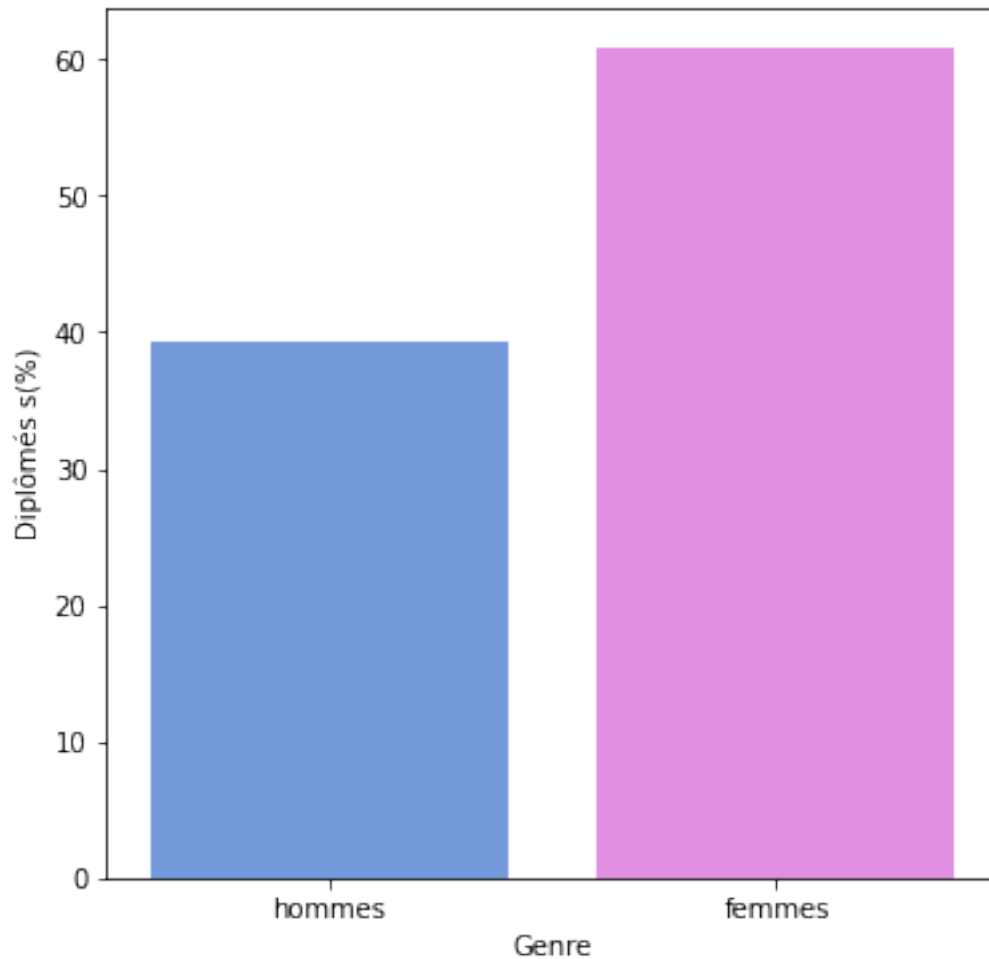
```
[238]:      Genre  Nombre de réponses      %
0  femmes      360364.0  60.796302
1  hommes      232376.0  39.203698
```

```
[239]: # Nombre d'hommes et de femmes diplômés sur toute la période
fig, ax = plt.subplots(figsize=(6,6))
sns.barplot(
```

```

data=df_distrib_genre,
x="Genre",
y="%",
order=["hommes", "femmes"],
palette=["cornflowerblue", "violet"],
ax=ax)
ax.set_ylabel("Diplômés s(%)")
plt.show()

```



```

[240]: # Nombre de diplômés par diplôme selon le genre sur l'ensemble de la période
↳ (2013-2019)
df_diplome_genre = df_stat.groupby(["Diplôme", "Genre"])["Nombre de
↳ réponses"].sum()
df_diplome_genre.reset_index(inplace=True)

```

```

[241]: df_diplome_genre

```

```
[241]:
```

	Diplôme	Genre	Nombre de réponses
0	DUT	femmes	8560.0
1	DUT	hommes	9328.0
2	LICENCE PRO	femmes	72910.0
3	LICENCE PRO	hommes	56474.0
4	MASTER ENS	femmes	56912.0
5	MASTER ENS	hommes	16400.0
6	MASTER LMD	femmes	221982.0
7	MASTER LMD	hommes	150174.0

```
[242]: # Nombre total de réponses
total = df_diplome_genre["Nombre de réponses"].sum()

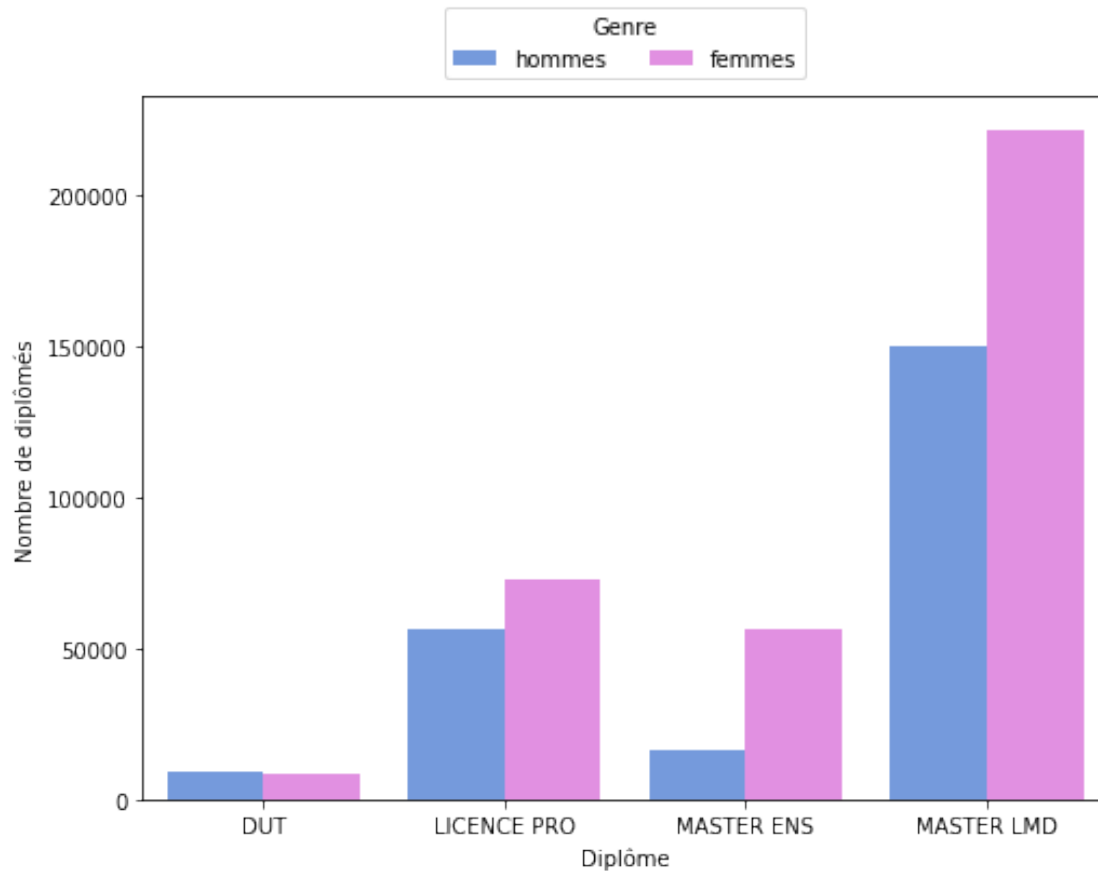
# Pourcentage de réponses
df_diplome_genre["%"] = round(df_diplome_genre["Nombre de réponses"] / total * 100, 1)
```

```
[243]: df_diplome_genre
```

```
[243]:
```

	Diplôme	Genre	Nombre de réponses	%
0	DUT	femmes	8560.0	1.4
1	DUT	hommes	9328.0	1.6
2	LICENCE PRO	femmes	72910.0	12.3
3	LICENCE PRO	hommes	56474.0	9.5
4	MASTER ENS	femmes	56912.0	9.6
5	MASTER ENS	hommes	16400.0	2.8
6	MASTER LMD	femmes	221982.0	37.5
7	MASTER LMD	hommes	150174.0	25.3

```
[244]: # Représentation graphique du nombre de diplômés par diplôme selon le genre
fig, ax = plt.subplots(figsize=(8,6))
sns.barplot(
    data=df_diplome_genre,
    x="Diplôme",
    y="Nombre de réponses",
    hue="Genre",
    hue_order=["hommes", "femmes"],
    palette=["cornflowerblue", "violet"],
    ax=ax
)
ax.set_ylabel("Nombre de diplômés")
plt.legend(loc='upper center', title='Genre', bbox_to_anchor=(0.5, 1.14),
    ncol=2)
plt.show()
```



SALAIRES

```
[245]: # Ensembles des disciplines par diplôme et année selon le genre
df_stat = ins_prof[selection]
df_stat = df_stat[
    df_stat["Discipline"].str.contains(r"Ensemble des départements|Ensemble_
↳Licence professionnelle|Ensemble Masters LMD|Masters enseignement") &
    ~df_stat["Genre"].str.contains(r"femmes et hommes") &
    #(df_stat["situation"] == "30 mois après le diplôme") &
    (df_stat["Année"] > "2012")
]
```

```
[246]: df_stat.sort_values(["Année", "Diplôme"]).head()
```

```
[246]:
```

	Année	Genre	Nombre de réponses	Diplôme	situation \
45	2013	femmes	5.0	DUT	30 mois après le diplôme
57	2013	hommes	1135.0	DUT	30 mois après le diplôme
122	2013	hommes	70.0	DUT	30 mois après le diplôme
130	2013	femmes	2.0	DUT	30 mois après le diplôme
131	2013	femmes	2.0	DUT	18 mois après le diplôme

	Discipline \	
45	Ensemble des départements d'IUT	
57	Ensemble des départements d'IUT	
122	Ensemble des départements d'IUT	
130	Ensemble des départements d'IUT	
131	Ensemble des départements d'IUT	
	Salaire net mensuel médian des emplois à temps plein \	
45		0.0
57		1600.0
122		1650.0
130		0.0
131		0.0
	Premier quartile des salaires nets mensuels des emplois à temps plein \	
45		0.0
57		1350.0
122		1440.0
130		0.0
131		0.0
	Dernier quartile des salaires nets mensuels des emplois à temps plein \	
45		0.0
57		1850.0
122		1920.0
130		0.0
131		0.0
	Salaire net mensuel médian national	Taux d'insertion
45	1870	0.0
57	1870	90.0
122	1870	99.0
130	1870	0.0
131	1870	0.0

```
[247]: # Suppression des salaires médians nuls
df_stat.drop(df_stat[df_stat["Salaire net mensuel médian des emplois à temps_
plein"] == 0].index, inplace=True)
```

```
[248]: # Somme du nombre de réponses par salaires médians
df_median = df_stat[[
    "Année", "Genre",
    "Diplôme", "Salaire net mensuel médian des emplois à temps plein",
    "Nombre de réponses"
]]
```

```
[249]: df_median["Salaire net mensuel médian des emplois à temps plein"].sort_values().
        ↪unique()
```

```
[249]: array([ 55., 1350., 1400., 1410., 1430., 1440., 1450., 1460.,
            1480., 1500., 1510., 1520., 1530., 1540., 1550., 1560.,
            1570., 1580., 1590., 1600., 1610., 1620., 1630., 1640.,
            1650., 1660., 1670., 1680., 1690., 1700., 1710., 1720.,
            1730., 1740., 1750., 1760., 1770., 1780., 1790., 1800.,
            1810., 1830., 1840., 1850., 1870., 1880., 1890., 1900.,
            1920., 1930., 1950., 1960., 1970., 1980., 2000., 2020.,
            2050., 2070., 2080., 2100., 2160., 2200., 2210., 2220.,
            23100.] )
```

1 Salaire médian selon le genre et le diplôme

```
[250]: # Salaire par genre selon le diplôme
df_median_genre = df_median.groupby(
    ["Genre", "Diplôme"]
).agg(
    {
        "Salaire net mensuel médian des emplois à temps plein": "median",
        "Nombre de réponses": "sum"
    }
)
df_median_genre.reset_index(inplace=True)
```

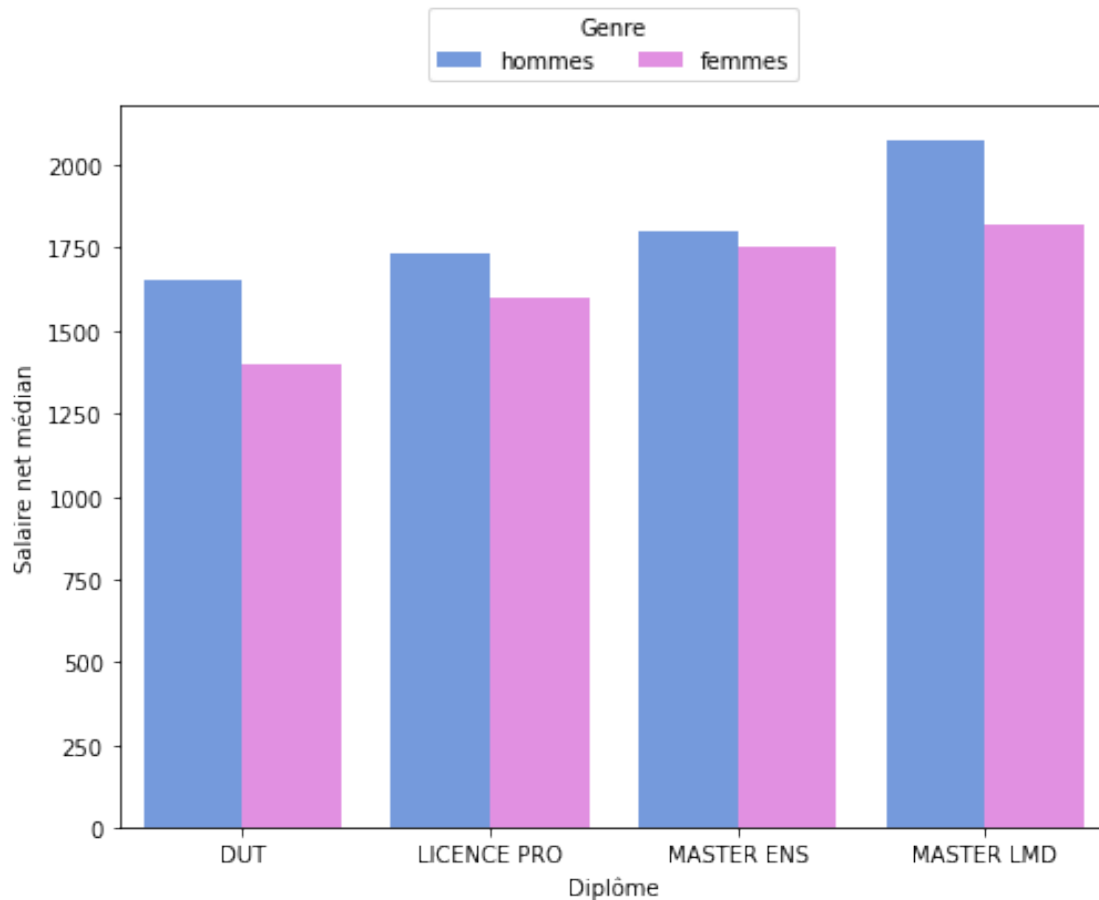
```
[251]: df_median_genre
```

```
[251]:
```

	Genre	Diplôme	Salaire net mensuel médian des emplois à temps plein \
0	femmes	DUT	1400.0
1	femmes	LICENCE PRO	1600.0
2	femmes	MASTER ENS	1750.0
3	femmes	MASTER LMD	1820.0
4	hommes	DUT	1650.0
5	hommes	LICENCE PRO	1730.0
6	hommes	MASTER ENS	1800.0
7	hommes	MASTER LMD	2075.0

	Nombre de réponses
0	9512.0
1	93804.0
2	56912.0
3	208954.0
4	18962.0
5	157746.0
6	16400.0
7	149682.0

```
[252]: fig, ax = plt.subplots(figsize=(8,6))
sns.barplot(
    data= df_median_genre,
    x="Diplôme",
    y="Salaire net mensuel médian des emplois à temps plein",
    hue="Genre",
    hue_order=["hommes", "femmes"],
    palette=["cornflowerblue", "violet"],
    ax=ax
)
ax.set_ylabel("Salaire net médian")
plt.legend(loc='upper center', title='Genre', bbox_to_anchor=(0.5, 1.15),
ncol=2)
plt.show()
```



2 Distribution des quartiles des salaires sur toute la période

```
[253]: # Selection de tous les quartiles
select = df_median.columns.to_list() + [
    "Premier quartile des salaires nets mensuels des emplois à temps plein",
    "Dernier quartile des salaires nets mensuels des emplois à temps plein"
]
df_quartiles = df_stat[select]

# groupement de tous les quartiles par genre
df_quartiles_genre = df_quartiles.melt(
    id_vars=["Année", "Genre"],
    value_vars=[
        "Salaire net mensuel médian des emplois à temps plein",
        "Premier quartile des salaires nets mensuels des emplois à temps plein",
        "Dernier quartile des salaires nets mensuels des emplois à temps plein"
    ],
    var_name="quantile",
    value_name="salaire"
)
```

```
[254]: df_quartiles_genre.head()
```

```
[254]:
```

	Année	Genre		quantile	salaire
0	2013	hommes	Salaire net mensuel médian des emplois à temps...		1600.0
1	2019	hommes	Salaire net mensuel médian des emplois à temps...		1740.0
2	2015	femmes	Salaire net mensuel médian des emplois à temps...		1400.0
3	2015	femmes	Salaire net mensuel médian des emplois à temps...		1350.0
4	2017	hommes	Salaire net mensuel médian des emplois à temps...		1700.0

```
[255]: df_quartiles_genre.groupby("Genre")[["salaire"]].quantile([0.25, 0.5, 0.75])
```

```
[255]:
```

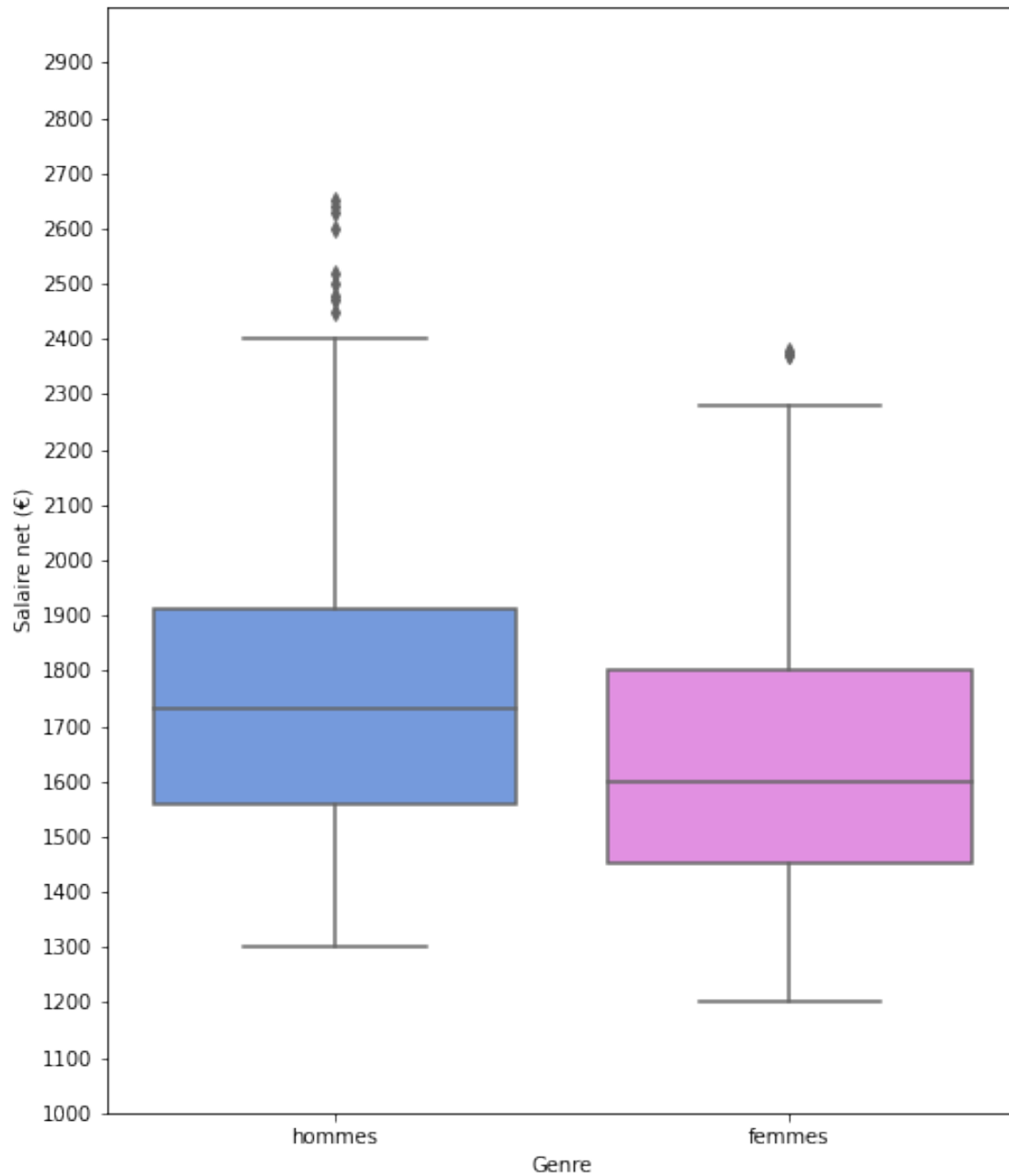
		salaire
Genre		
femmes	0.25	1450.0
	0.50	1600.0
	0.75	1800.0
hommes	0.25	1560.0
	0.50	1730.0
	0.75	1910.0

```
[256]: df_quartiles_genre.groupby("Genre")[["salaire"]].median()
```

```
[256]:
```

	salaire
Genre	
femmes	1600.0
hommes	1730.0


```
[257]: # Définition de la graduation des salaires
yticks = range(1000, 3000, 100)
plt.figure(figsize=(8,10))
sns.boxplot(
    data=df_quartiles_genre,
    x="Genre",
    y="salaire",
    order=["hommes", "femmes"],
    palette=["cornflowerblue", "violet"],
)
plt.yticks(yticks, [f"{int(y)}" for y in yticks])
# On limite la tranche des salaires entre 1000 et 4000 €
plt.ylim(1000, 3000)
plt.ylabel("Salaire net (€)")
plt.show()
```



```
[258]: df_quartiles_diplomes = df_quartiles.melt(  
    id_vars=["Année", "Genre", "Diplôme"],  
    value_vars=[  
        "Salaire net mensuel médian des emplois à temps plein",  
        "Premier quartile des salaires nets mensuels des emplois à temps plein",  
        "Dernier quartile des salaires nets mensuels des emplois à temps plein"  
    ],  
    var_name="quantile",
```

```

        value_name="salaire"
    )

```

```

[259]: #
df_quartiles_diplomes.groupby(["Genre", "Diplôme"])[["salaire"]].quantile([0.
↪25, 0.5, 0.75])

```

```

[259]:
Genre  Diplôme      salaire
femmes DUT      0.25    1290.0
          0.50    1400.0
          0.75    1585.0
          LICENCE PRO 0.25    1430.0
          0.50    1585.0
          0.75    1737.5
          MASTER ENS  0.25    1700.0
          0.50    1750.0
          0.75    1842.5
          MASTER LMD  0.25    1607.5
          0.50    1820.0
          0.75    2087.5
hommes DUT      0.25    1400.0
          0.50    1600.0
          0.75    1800.0
          LICENCE PRO 0.25    1600.0
          0.50    1730.0
          0.75    1950.0
          MASTER ENS  0.25    1700.0
          0.50    1800.0
          0.75    1965.0
          MASTER LMD  0.25    1817.5
          0.50    2075.0
          0.75    2345.0

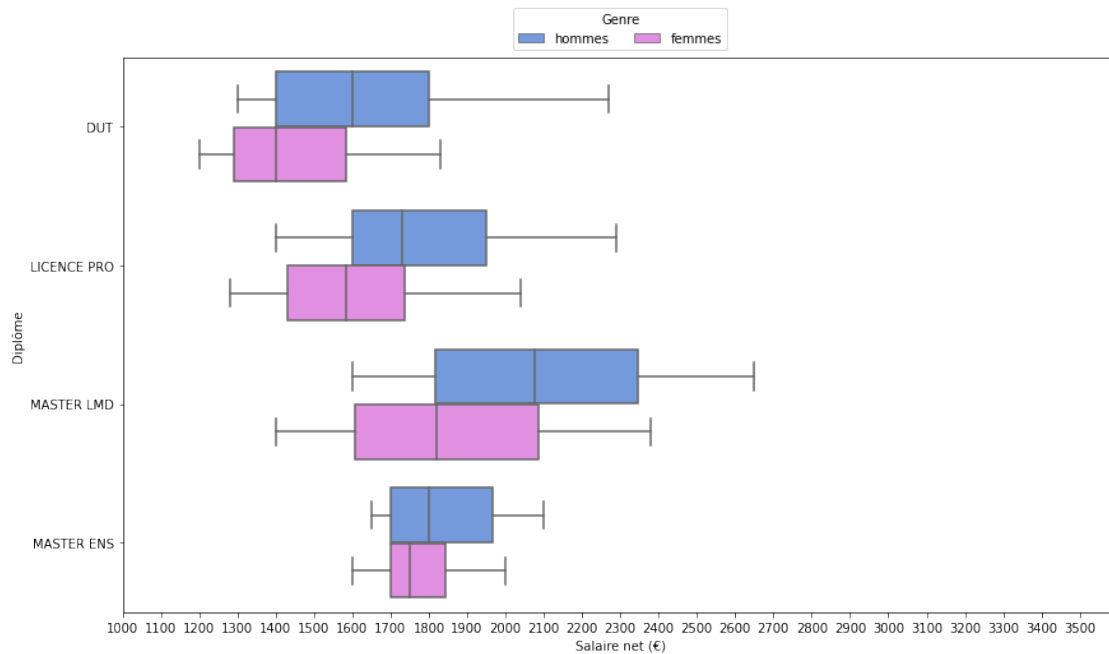
```

```

[260]: # Définition de la graduation des salaires
xticks = range(1000, 3600, 100)
# Création du boxplot des quartiles de salaires par diplôme selon le genre
plt.figure(figsize=(14,8))
sns.boxplot(
    data=df_quartiles_diplomes,
    y="Diplôme",
    x="salaire",
    hue_order=["hommes", "femmes"],
    hue="Genre",
    palette=["cornflowerblue", "violet"]
)
plt.xticks(xticks, [f"{int(x)}" for x in xticks])

```

```
# On limite la tranche des salaires entre 1000 et 4000 €
plt.xlim(1000, 3600)
plt.xlabel("Salaire net (€)")
plt.legend(loc='upper center', title='Genre', bbox_to_anchor=(0.5, 1.1), ncol=2)
plt.show()
```



```
[261]: # Suppression des ensembles pour ne garder que les disciplines propres
df_disciplines = ins_prof[select + ["Discipline", "situation"]].copy()
df_disciplines = df_disciplines[
    ~df_disciplines["Discipline"].str.contains(r"Ensemble|Autres", case=False) &
    ~df_disciplines["Genre"].str.contains(r"femmes et hommes") &
    (df_disciplines["situation"] == "18 mois après le diplôme") &
    (df_disciplines["Année"] > "2012")
]
```

```
[262]: df_disciplines["Discipline"].unique()
```

```
[262]: array(['Informatique', 'Information communication',
            'Sciences fondamentales', 'Sciences de la vie et de la terre',
            'Gestion', 'Droit', 'Lettres, langues, arts',
            'Histoire-géographie', 'Psychologie', 'Économie',
            'Sciences de l'ingénieur', 'Masters enseignement'], dtype=object)
```

```
[263]: df_quartiles_disciplines = df_disciplines.melt(
    id_vars=["Année", "Genre", "Diplôme", "Discipline"],
    value_vars=[
```

```

        "Salaire net mensuel médian des emplois à temps plein",
        "Premier quartile des salaires nets mensuels des emplois à temps plein",
        "Dernier quartile des salaires nets mensuels des emplois à temps plein"
    ],
    var_name="quartile",
    value_name="salaire"
)

```

```
[264]: df_quartiles_disciplines
```

```

[264]:
   Année  Genre  Diplôme  Discipline \
0    2017  hommes    DUT    Informatique
1    2017  femmes    DUT  Information communication
2    2013  hommes    DUT  Information communication
3    2019  hommes    DUT  Information communication
4    2014  hommes    DUT    Sciences fondamentales
...    ...    ...    ...    ...
2860  2018  hommes  MASTER LMD    Sciences de l'ingénieur
2861  2019  hommes  MASTER LMD    Sciences de l'ingénieur
2862  2017  hommes  MASTER ENS    Masters enseignement
2863  2013  femmes  MASTER ENS    Masters enseignement
2864  2015  hommes  MASTER ENS    Masters enseignement

                                quartile  salaire
0    Salaire net mensuel médian des emplois à temps...  1570.0
1    Salaire net mensuel médian des emplois à temps... 20300.0
2    Salaire net mensuel médian des emplois à temps...    0.0
3    Salaire net mensuel médian des emplois à temps...  1400.0
4    Salaire net mensuel médian des emplois à temps...    0.0
...    ...    ...    ...    ...
2860  Dernier quartile des salaires nets mensuels de...  2350.0
2861  Dernier quartile des salaires nets mensuels de...  2450.0
2862  Dernier quartile des salaires nets mensuels de...  1950.0
2863  Dernier quartile des salaires nets mensuels de...  1780.0
2864  Dernier quartile des salaires nets mensuels de...  1970.0

```

```
[2865 rows x 6 columns]
```

```

[265]: # Suppression des salaires nuls
df_quartiles_disciplines = df_quartiles_disciplines.drop(
    df_quartiles_disciplines[df_quartiles_disciplines.loc[:, "salaire"] == 0].
    ↪index
)

```

```

[266]: # Tri par salaires médians
df_test = df_quartiles_disciplines.groupby(["Genre",
    ↪"Discipline"])[["salaire"]].quantile([0.25, 0.5, 0.75])

```

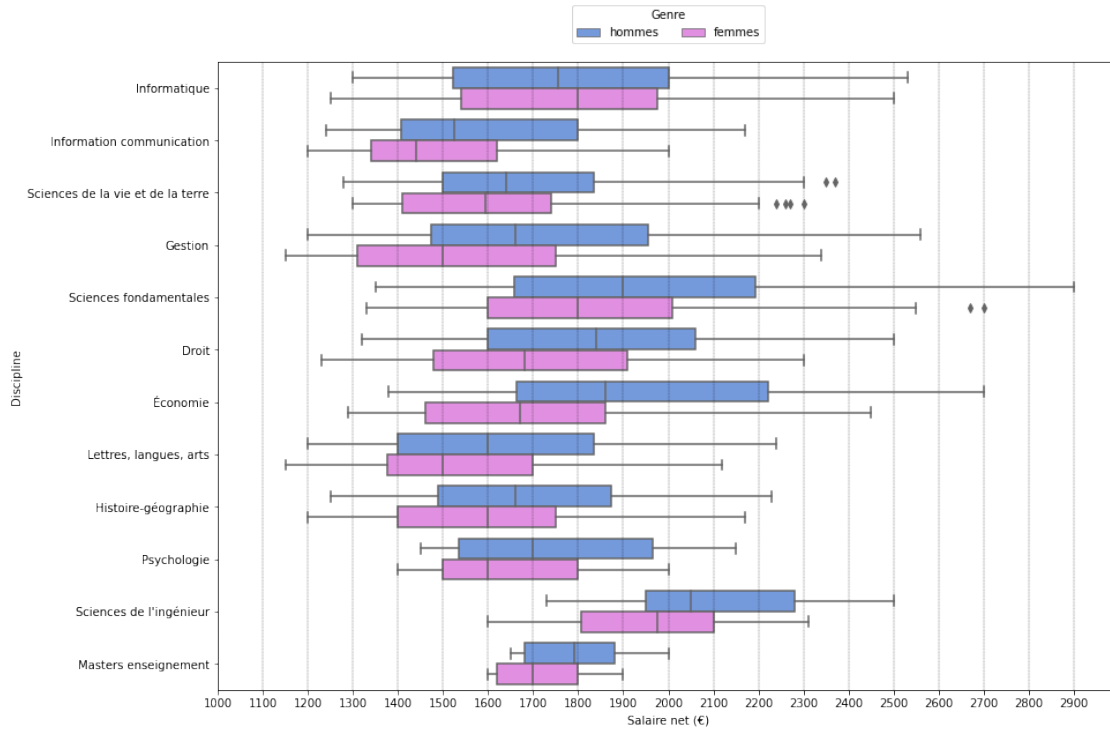
```
[267]: median_femmes = df_test.loc["femmes", :, 0.50]
       median_hommes = df_test.loc["hommes", :, 0.50]
```

```
[268]: # différence de salaire entre les hommes et les femmes dans chaque discipline
       median_hommes - median_femmes
```

```
[268]:
```

Discipline	salaire
Droit	160.0
Gestion	160.0
Histoire-géographie	60.0
Information communication	85.0
Informatique	-45.0
Lettres, langues, arts	100.0
Masters enseignement	90.0
Psychologie	100.0
Sciences de l'ingénieur	75.0
Sciences de la vie et de la terre	45.0
Sciences fondamentales	100.0
Économie	190.0

```
[269]: # Définition de la graduation des salaires
xticks = range(1000, 3000, 100)
# Création du boxplot des quartiles de salaires par discipline selon le genre
plt.figure(figsize=(14,10))
sns.boxplot(
    data=df_quartiles_disciplines,
    y="Discipline",
    x="salaire",
    hue_order=["hommes", "femmes"],
    hue="Genre",
    palette=["cornflowerblue", "violet"]
)
plt.gca().xaxis.grid(True, linestyle='--', which='both', linewidth=0.3,
    color='black')
plt.xticks(xticks, [f"{int(x)}" for x in xticks])
# On limite la tranche des salaires entre 1000 et 4000 €
plt.xlim(1000, 3000)
plt.xlabel("Salaire net (€)")
plt.legend(loc='upper center', title='Genre', bbox_to_anchor=(0.5, 1.1), ncol=2)
plt.show()
```



3 Distribution des salaires médians

```
[270]: # Distribution des salaires
df_salaires_distribution = df_stat.groupby(["Salaire net mensuel médian des_
↳emplois à temps plein"])[["Nombre de réponses"]].sum()

[271]: df_salaires_distribution.sort_values("Salaire net mensuel médian des emplois à_
↳temps plein", ascending=False, inplace=True)
df_salaires_distribution.reset_index(inplace=True)

[272]: df_salaires_distribution.head()

[272]:   Salaire net mensuel médian des emplois à temps plein  Nombre de réponses
0                23100.0                826.0
1                2220.0               12995.0
2                2210.0                 0.0
3                2200.0               12469.0
4                2160.0               12732.0

[273]: # Correction des salaires trop élevés
rows = df_salaires_distribution[
    df_salaires_distribution["Salaire net mensuel médian des emplois à temps_
↳plein"] > 20000].index
```

```
df_salaires_distribution.iloc[rows, 0] = df_salaires_distribution.iloc[rows, 0] / 10

# Suppression des salaires trop faibles
rows = df_salaires_distribution[
    df_salaires_distribution["Salaire net mensuel médian des emplois à temps plein"] < 1000].index
df_salaires_distribution.drop(rows, inplace=True)
```

```
[274]: df_salaires_distribution.head()
```

```
[274]:
```

	Salaire net mensuel médian des emplois à temps plein	Nombre de réponses
0	2310.0	826.0
1	2220.0	12995.0
2	2210.0	0.0
3	2200.0	12469.0
4	2160.0	12732.0

```
[275]: # Proportion des salaires
total = df_salaires_distribution["Nombre de réponses"].sum()
df_salaires_distribution["%"] = df_salaires_distribution["Nombre de réponses"] /
    total * 100
```

```
[276]: df_salaires_distribution.head()
```

```
[276]:
```

	Salaire net mensuel médian des emplois à temps plein	Nombre de réponses \
0	2310.0	826.0
1	2220.0	12995.0
2	2210.0	0.0
3	2200.0	12469.0
4	2160.0	12732.0

	%
0	0.116151
1	1.827332
2	0.000000
3	1.753367
4	1.790350

```
[277]: # Découpage par tranche de salaires
salaires_range = pd.cut(
    df_salaires_distribution["Salaire net mensuel médian des emplois à temps plein"],
    bins=range(1200,3000,100), right=False)

df_salaires_distribution["Fourchette salaire"] = salaries_range
```



```
[278]: df_salaires_distribution.head()
```

```
[278]:   Salaire net mensuel médian des emplois à temps plein  Nombre de réponses  \
0                2310.0                826.0
1                2220.0            12995.0
2                2210.0                0.0
3                2200.0            12469.0
4                2160.0            12732.0

      % Fourchette salaire
0  0.116151    [2300, 2400)
1  1.827332    [2200, 2300)
2  0.000000    [2200, 2300)
3  1.753367    [2200, 2300)
4  1.790350    [2100, 2200)
```

```
[279]: # Calcul du nombre de diplômé par fourchette de salaires
df_graph_salaires = df_salaires_distribution.groupby("Fourchette_
↪salaire")["Nombre de réponses"].sum()

# Renommage des fourchettes

df_graph_salaires.reset_index(inplace=True)
```

```
[280]: df_graph_salaires
```

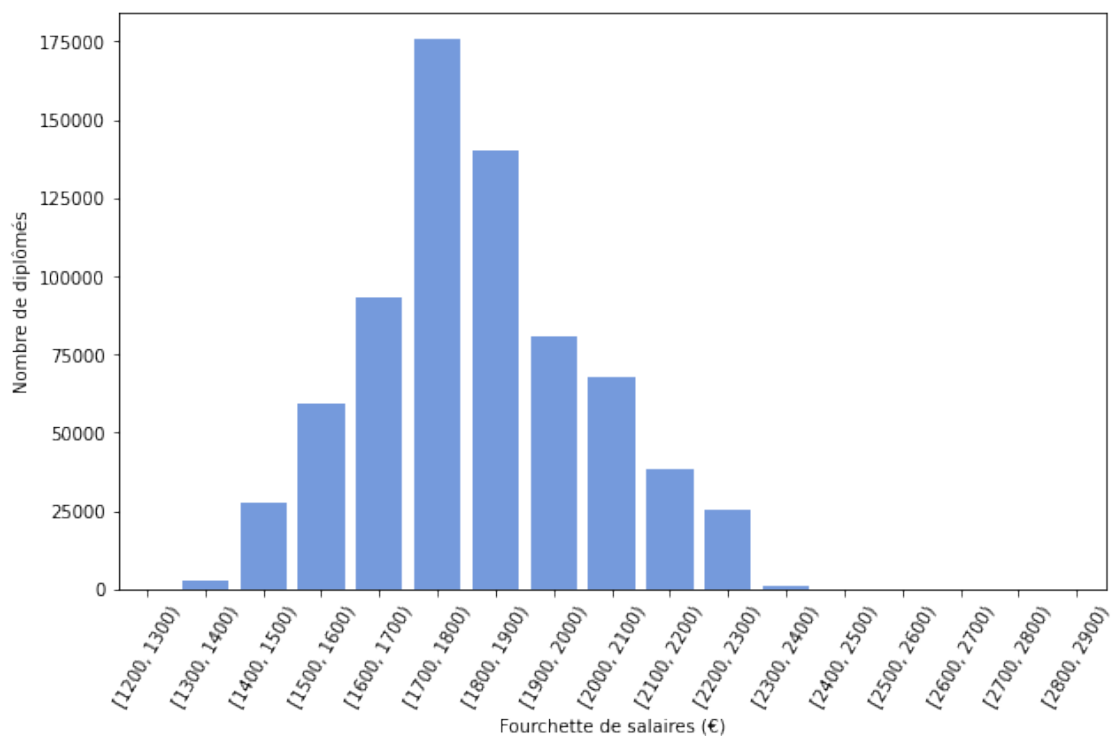
```
[280]:   Fourchette salaire  Nombre de réponses
0    [1200, 1300)          0.0
1    [1300, 1400)        2499.0
2    [1400, 1500)       27349.0
3    [1500, 1600)      59139.0
4    [1600, 1700)      93221.0
5    [1700, 1800)     175637.0
6    [1800, 1900)     140276.0
7    [1900, 2000)     80556.0
8    [2000, 2100)     67913.0
9    [2100, 2200)     38266.0
10   [2200, 2300)     25464.0
11   [2300, 2400)       826.0
12   [2400, 2500)        0.0
13   [2500, 2600)        0.0
14   [2600, 2700)        0.0
15   [2700, 2800)        0.0
16   [2800, 2900)        0.0
```

```
[281]: # Distribution des salaires par trange de 100 €
fig, ax = plt.subplots(figsize=(10,6))
```

```

sns.barplot(
    data=df_graph_salaires,
    x="Fourchette salaire",
    y="Nombre de réponses",
    errorbar=None,
    palette=["cornflowerblue"],
    ax=ax
)
ax.set_xlabel("Fourchette de salaires (€)")
ax.set_ylabel("Nombre de diplômés")
ax.set_xticklabels(ax.get_xticklabels(),rotation=60)
plt.show()

```



4 Proportion des disciplines

```

[282]: df_distrib_disciplines = ins_prof[select + ["Discipline", "situation"]]
df_distrib_disciplines = df_distrib_disciplines[
    ~df_distrib_disciplines["Discipline"].str.contains(r"Ensemble|Autres",
    ↪case=False) &
    ~df_distrib_disciplines["Genre"].str.contains(r"femmes et hommes") &
    #(df_distrib_disciplines["situation"] == "30 mois après le diplôme") &
    (df_distrib_disciplines["Année"] > "2012")

```

```
]
```

```
[283]: # Nombre de diplômés par disciplines selon le genre
df_distrib_discipline = df_distrib_disciplines.groupby(["Genre", "Diplôme",
↳ "Discipline"])[["Nombre de réponses"]].sum()
total = df_distrib_discipline["Nombre de réponses"].sum()
df_distrib_discipline["%"] = df_distrib_discipline["Nombre de réponses"] /
↳ total * 100
df_distrib_discipline.sort_values("%", ascending=False, inplace=True)
df_distrib_discipline.reset_index(inplace=True)
```

```
[284]: df_distrib_discipline.head()
```

```
[284]:
```

	Genre	Diplôme	Discipline	Nombre de réponses	%
0	femmes	MASTER ENS	Masters enseignement	56912.0	9.601512
1	femmes	MASTER LMD	Lettres, langues, arts	53536.0	9.031953
2	femmes	MASTER LMD	Gestion	46924.0	7.916456
3	hommes	MASTER LMD	Gestion	35956.0	6.066066
4	femmes	LICENCE PRO	Gestion	31946.0	5.389547

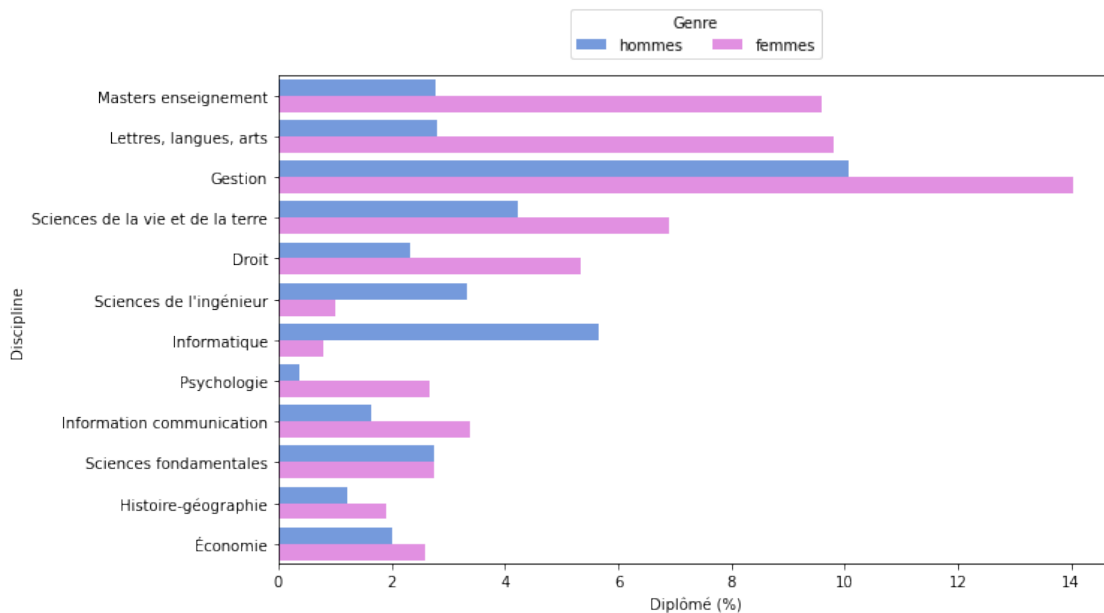
```
[285]: df_distrib_discipline.groupby(["Genre", "Discipline"])[["Nombre de réponses",
↳ "%"]].sum().sort_values(["Genre", "%"], ascending=False)
```

```
[285]:
```

		Nombre de réponses	%
Genre	Discipline		
hommes	Gestion	59674.0	10.067483
	Informatique	33584.0	5.665891
	Sciences de la vie et de la terre	25126.0	4.238958
	Sciences de l'ingénieur	19770.0	3.335358
	Lettres, langues, arts	16652.0	2.809326
	Masters enseignement	16400.0	2.766812
	Sciences fondamentales	16338.0	2.756352
	Droit	13818.0	2.331208
	Économie	11840.0	1.997503
	Information communication	9758.0	1.646253
	Histoire-géographie	7180.0	1.211324
	Psychologie	2236.0	0.377231
femmes	Gestion	83264.0	14.047306
	Lettres, langues, arts	58158.0	9.811722
	Masters enseignement	56912.0	9.601512
	Sciences de la vie et de la terre	40958.0	6.909944
	Droit	31676.0	5.343996
	Information communication	20006.0	3.375173
	Sciences fondamentales	16264.0	2.743867
	Psychologie	15898.0	2.682120
	Économie	15334.0	2.586969
	Histoire-géographie	11228.0	1.894254

Sciences de l'ingénieur	5928.0	1.000101
Informatique	4738.0	0.799339

```
[286]: fig, ax = plt.subplots(figsize=(10, 6))
sns.barplot(
    data=df_distrib_discipline,
    y="Discipline",
    x="%",
    hue_order=["hommes", 'femmes'],
    hue="Genre",
    errorbar=None,
    palette=["cornflowerblue", "violet"],
    estimator="sum",
    ax=ax
)
ax.set_xlabel("Diplômé (%)")
plt.legend(loc='upper center', title='Genre', bbox_to_anchor=(0.5, 1.15),
ncol=2)
plt.show()
```



4.1 Taux d'insertion par diplôme

```
[287]: # Nombre de valeurs manquantes du taux d'insertion par genre
test_na = df_stat_na["Taux d'insertion"].isna()
df_na = df_stat_na[test_na][["Genre", "Diplôme", "Taux d'insertion"]]
df_na["Taux d'insertion"] = True
```

```
df_taux_na = df_na.groupby(["Genre", "Diplôme"])["Taux d'insertion"].count()
df_taux_na.name= "NA"
df_taux_na = pd.DataFrame(df_taux_na)
```

```
[288]: # Taux de valeurs manquantes du taux d'insertion par genre
total = df_na.shape[0]
df_taux_na["% NA"] = round(df_taux_na["NA"] / total * 100)
```

```
[289]: df_taux_na
```

```
[289]:
```

		NA	% NA
Genre	Diplôme		
femmes	DUT	227	16.0
	LICENCE PRO	172	12.0
	MASTER LMD	151	11.0
femmes et hommes	DUT	79	6.0
	LICENCE PRO	130	9.0
	MASTER LMD	80	6.0
hommes	DUT	128	9.0
	LICENCE PRO	239	17.0
	MASTER LMD	198	14.0

```
[290]: # Selection par diplôme selon la situation
df_stat = ins_prof[selection]
df_stat = df_stat[
    (df_stat.loc[:, "Discipline"].str.contains(r"Ensemble des_
↳départements|Ensemble Licence professionnelle|Ensemble Masters LMD|^Masters_
↳enseignement")) &
    (df_stat.loc[:, "Genre"].str.contains(r"^femmes et hommes")) &
    (df_stat["Année"] > '2012')]
df_stat.sort_values(["Année", "Diplôme"], inplace=True)
```

```
[291]: # Suppression des taux d'insertions non significatifs
df_insertion = df_stat.drop(df_stat[df_stat["Taux d'insertion"] == 0].index)

# Nombre de diplômés insérés par rapport au taux d'insertion
df_insertion["Diplômé inséré"] = round(df_insertion["Nombre de réponses"] *
↳df_insertion["Taux d'insertion"] / 100)

# Nombre de diplômés insérés par année et diplôme
df_insertion = df_insertion.groupby(["Année", "Diplôme", "Discipline",
↳"situation"])[["Nombre de réponses", "Diplômé inséré"]].sum()

# Pourcentage de diplômés insérés par année selon le diplôme
df_insertion["% diplômé inséré"] = round(df_insertion["Diplômé inséré"] /
↳df_insertion["Nombre de réponses"] * 100)
```

[292]: df_insertion

```
[292]:
```

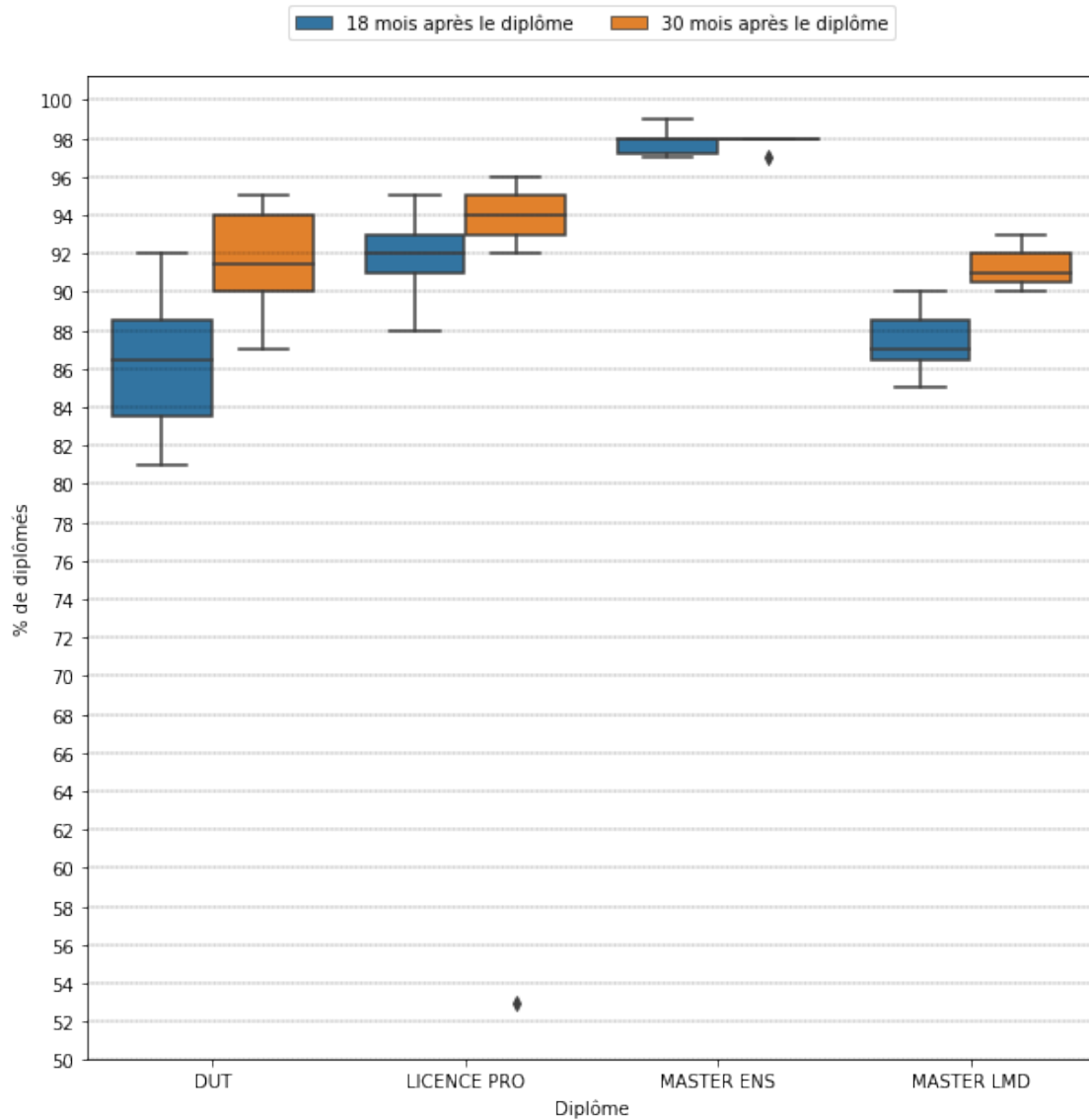
		Nombre de réponses \	
Année	Diplôme	Discipline	situation
2013	DUT	Ensemble des départements d'IUT	18 mois après
	le diplôme	2054.0	30 mois après
	le diplôme	2054.0	18 mois après
		Ensemble des départements d'IUT	30 mois après
	le diplôme	850.0	18 mois après
		850.0	30 mois après
	LICENCE PRO	Ensemble Licence professionnelle	18 mois après
	le diplôme	16625.0	
	...		
	...		
2019	LICENCE PRO	Ensemble des départements d'IUT	30 mois après
	le diplôme	7544.0	18 mois après
	MASTER ENS	Masters enseignement	30 mois après
	le diplôme	5912.0	18 mois après
		5912.0	30 mois après
	MASTER LMD	Ensemble Masters LMD (hors Masters enseignement)	18 mois après
	le diplôme	31616.0	30 mois après
	le diplôme	31616.0	

		Diplômé inséré \	
Année	Diplôme	Discipline	situation
2013	DUT	Ensemble des départements d'IUT	18 mois après
	le diplôme	1664.0	30 mois après
	le diplôme	1849.0	18 mois après
		Ensemble des départements d'IUT	30 mois après
	le diplôme	714.0	18 mois après
		781.0	30 mois après
	LICENCE PRO	Ensemble Licence professionnelle	18 mois après
	le diplôme	14630.0	
	...		
	...		
2019	LICENCE PRO	Ensemble des départements d'IUT	30 mois après
	le diplôme	7183.0	18 mois après
	MASTER ENS	Masters enseignement	30 mois après
	le diplôme	5794.0	
	le diplôme	5794.0	

MASTER LMD	Ensemble Masters LMD (hors Masters enseignement)	18 mois après
le diplôme	27506.0	
		30 mois après
le diplôme	29403.0	
	% diplômé inséré	
Année Diplôme	Discipline	situation
2013 DUT	Ensemble des départements d'IUT	18 mois après
le diplôme	81.0	
		30 mois après
le diplôme	90.0	
	Ensemble des départements d'IUT	18 mois après
le diplôme	84.0	
		30 mois après
le diplôme	92.0	
LICENCE PRO	Ensemble Licence professionnelle	18 mois après
le diplôme	88.0	
...		
...		
2019 LICENCE PRO	Ensemble des départements d'IUT	30 mois après
le diplôme	95.0	
MASTER ENS	Masters enseignement	18 mois après
le diplôme	98.0	
		30 mois après
le diplôme	98.0	
MASTER LMD	Ensemble Masters LMD (hors Masters enseignement)	18 mois après
le diplôme	87.0	
		30 mois après
le diplôme	93.0	

[80 rows x 3 columns]

```
[293]: fig, ax = plt.subplots(figsize=(10,10))
yticks = range(50, 101, 2)
sns.boxplot(
    data=df_insertion.reset_index(),
    x="Diplôme",
    y="% diplômé inséré",
    hue="situation",
    ax=ax
)
plt.gca().yaxis.grid(True, linestyle='--', which='both', linewidth=0.3,
    color='black')
plt.ylabel("% de diplômés")
plt.yticks(yticks, [f"{int(x)}" for x in yticks])
plt.legend(loc='upper center', title='', bbox_to_anchor=(0.5, 1.08), ncol=3)
plt.show()
```



```
[294]: # Selection par diplôme selon la situation
df_stat = ins_prof[selection]
df_stat = df_stat[
    (~df_stat.loc[:, "Discipline"].str.contains(r"Ensemble|^Masters|^
    enseignement|Autres")) &
    (df_stat.loc[:, "Genre"].str.contains(r"^femmes et hommes")) &
    (df_stat["Année"] > '2012')]
df_stat.sort_values(["Année", "Diplôme"], inplace=True)

[295]: # Suppression des taux d'insertions non significatifs
df_insertion = df_stat.drop(df_stat[df_stat["Taux d'insertion"] == 0].index)
```



```

# Nombre de diplômés insérés par rapport au taux d'insertion
df_insertion["Diplômé inséré"] = round(df_insertion["Nombre de réponses"] *
↳ df_insertion["Taux d'insertion"] / 100)

# Nombre de diplômés insérés par année et diplôme
df_insertion = df_insertion.groupby(["Année", "Diplôme", "Discipline",
↳ "situation"])[["Nombre de réponses", "Diplômé inséré"]].sum()

# Pourcentage de diplômés insérés par année selon le diplôme
df_insertion["% diplômé inséré"] = round(df_insertion["Diplômé inséré"] /
↳ df_insertion["Nombre de réponses"] * 100)

```

```
[296]: df_insertion.head(20)
```

```

[296]: Nombre de réponses \
Année Diplôme      Discipline      situation
2013 DUT          Gestion          18 mois après le diplôme
1387.0
1387.0          30 mois après le diplôme
1387.0
440.0          Information communication 18 mois après le diplôme
440.0          30 mois après le diplôme
440.0
308.0          Informatique          18 mois après le diplôme
308.0          30 mois après le diplôme
308.0
406.0          Sciences de la vie et de la terre 18 mois après le diplôme
406.0          30 mois après le diplôme
406.0
272.0          Sciences fondamentales 18 mois après le diplôme
272.0          30 mois après le diplôme
272.0
406.0 LICENCE PRO Droit          18 mois après le diplôme
406.0          30 mois après le diplôme
406.0
4113.0          Gestion          18 mois après le diplôme
4113.0          30 mois après le diplôme
4113.0
43.0          Histoire-géographie 18 mois après le diplôme
43.0          30 mois après le diplôme

```

43.0			
	Information communication		18 mois après le diplôme
855.0			30 mois après le diplôme
855.0			
	Informatique		18 mois après le diplôme
1548.0			30 mois après le diplôme
1548.0			
Diplômé inséré \			
Année Diplôme	Discipline		situation
2013 DUT	Gestion		18 mois après le diplôme
1095.0			30 mois après le diplôme
1250.0			
	Information communication		18 mois après le diplôme
347.0			30 mois après le diplôme
379.0			
	Informatique		18 mois après le diplôme
259.0			30 mois après le diplôme
276.0			
	Sciences de la vie et de la terre		18 mois après le diplôme
357.0			30 mois après le diplôme
382.0			
	Sciences fondamentales		18 mois après le diplôme
214.0			30 mois après le diplôme
230.0			
LICENCE PRO Droit			18 mois après le diplôme
361.0			30 mois après le diplôme
378.0			
	Gestion		18 mois après le diplôme
3702.0			30 mois après le diplôme
3825.0			
	Histoire-géographie		18 mois après le diplôme
37.0			30 mois après le diplôme
40.0			
	Information communication		18 mois après le diplôme
693.0			30 mois après le diplôme

752.0			
	Informatique		18 mois après le diplôme
1378.0			
			30 mois après le diplôme
1440.0			
			%
diplômé inséré			
Année Diplôme	Discipline		situation
2013 DUT	Gestion		18 mois après le diplôme
79.0			
			30 mois après le diplôme
90.0			
	Information communication		18 mois après le diplôme
79.0			
			30 mois après le diplôme
86.0			
	Informatique		18 mois après le diplôme
84.0			
			30 mois après le diplôme
90.0			
	Sciences de la vie et de la terre		18 mois après le diplôme
88.0			
			30 mois après le diplôme
94.0			
	Sciences fondamentales		18 mois après le diplôme
79.0			
			30 mois après le diplôme
85.0			
LICENCE PRO Droit			18 mois après le diplôme
89.0			
			30 mois après le diplôme
93.0			
	Gestion		18 mois après le diplôme
90.0			
			30 mois après le diplôme
93.0			
	Histoire-géographie		18 mois après le diplôme
86.0			
			30 mois après le diplôme
93.0			
	Information communication		18 mois après le diplôme
81.0			
			30 mois après le diplôme
88.0			
	Informatique		18 mois après le diplôme
89.0			

93.0

```
[297]: fig, ax = plt.subplots(figsize=(10,10))
xticks = range(0, 101, 5)
sns.boxplot(
    data=df_insertion.reset_index(),
    x="% diplômé inséré",
    y="Discipline",
    hue="situation",
    ax=ax
)
plt.gca().xaxis.grid(True, linestyle='--', which='both', linewidth=0.3,
    color='black')
plt.ylabel("% de diplômés")
plt.xticks(xticks, [f"{int(x)}" for x in xticks])
plt.legend(loc='upper center', title='', bbox_to_anchor=(0.5, 1.08), ncol=3)
plt.show()
```

