

```
In [2]: import pandas as pd
import numpy as np
import matplotlib as plt

In [4]: pd.read_csv("/Users/matthieulambert/Desktop/Portfolio Data analysis/Declarations préalables à l'embauche/dpae-mensuelles-france-entiere.csv")

Out[4]:
```

	annee	trimestre	dernier_jour_du_mois	duree_de_contrat	nature_de_contrat	dpae_brut	dpae_cvs
0	2000	1	2000-02-29	Contrat de plus d'un mois	CDI	220983	230017
1	2000	2	2000-04-30	Contrat de plus d'un mois	CDI	275215	283662
2	2000	3	2000-09-30	Contrat de plus d'un mois	CDI	361544	264729
3	2000	4	2000-10-31	Contrat de plus d'un mois	CDD de plus d'un mois	339174	315891
4	2000	4	2000-11-30	Contrat de plus d'un mois	CDI	298850	296905
...
573	2023	3	2023-09-30	Contrat de plus d'un mois	CDI		
574	2023	3	2023-09-30	Contrat de plus d'un mois	CDI		
575	2023	4	2023-12-31	Contrat de plus d'un mois	CDI		
576	2024	1	2024-01-31	Contrat de plus d'un mois	CDI		
577	2022	1	2022-02-28	Contrat de plus d'un mois	CDI		

578 rows × 1 columns

```
In [22]: data = pd.read_json("/Users/matthieulambert/Desktop/Portfolio Data analysis/Declarations préalables à l'embauche/dpae-mensuelles-france-entiere.js")
```

```
In [23]: data.head()
```

```
Out[23]:
```

	annee	trimestre	dernier_jour_du_mois	duree_de_contrat	nature_de_contrat	dpae_brut	dpae_cvs
0	2000	1	2000-02-29	Contrat de plus d'un mois	CDI	220983	230017
1	2000	2	2000-04-30	Contrat de plus d'un mois	CDI	275215	283662
2	2000	3	2000-09-30	Contrat de plus d'un mois	CDI	361544	264729
3	2000	4	2000-10-31	Contrat de plus d'un mois	CDD de plus d'un mois	339174	315891
4	2000	4	2000-11-30	Contrat de plus d'un mois	CDI	298850	296905

```
In [24]: data.columns.tolist()
```

```
Out[24]: ['annee',
'trimestre',
'dernier_jour_du_mois',
'duree_de_contrat',
'nature_de_contrat',
'dpae_brut',
'dpae_cvs']
```

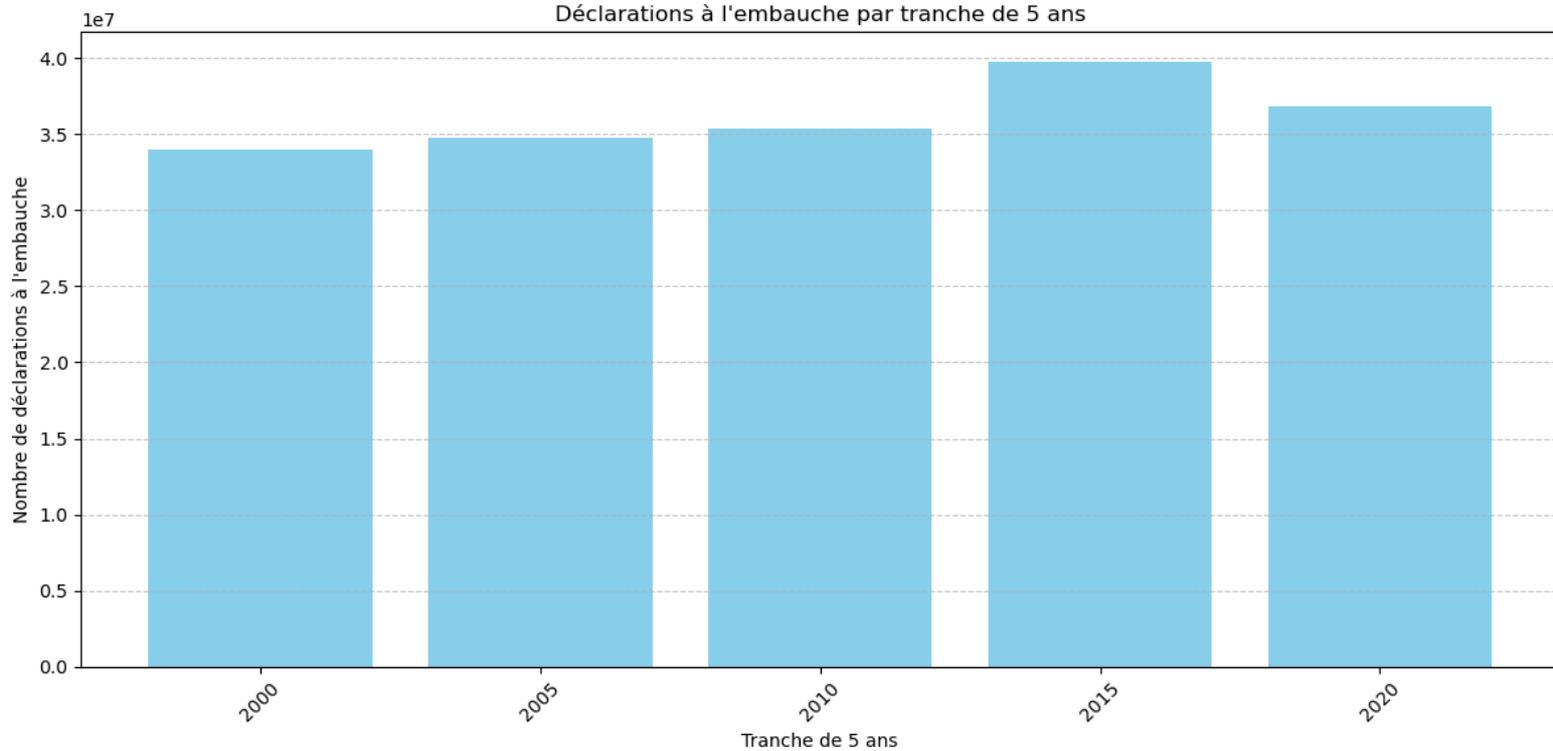
```
In [31]: import matplotlib.pyplot as plt
import numpy as np
```

```
# Créer une nouvelle colonne pour les tranches de 5 ans
data['Tranche_5_ans'] = (np.floor(data['annee'] / 5) * 5).astype(int)
```

```
# Grouper les données par tranche de 5 ans et sommer les déclarations à l'embauche
grouped_data = data.groupby('Tranche_5_ans')['dpae_brut'].sum().reset_index()
```

```
# Création du graphique
plt.figure(figsize=(12, 6))
plt.bar(grouped_data['Tranche_5_ans'].astype(str), grouped_data['dpae_brut'], color='skyblue')
plt.title('Déclarations à l'embauche par tranche de 5 ans')
plt.xlabel('Tranche de 5 ans')
plt.ylabel('Nombre de déclarations à l'embauche')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```

Déclarations à l'embauche par tranche de 5 ans



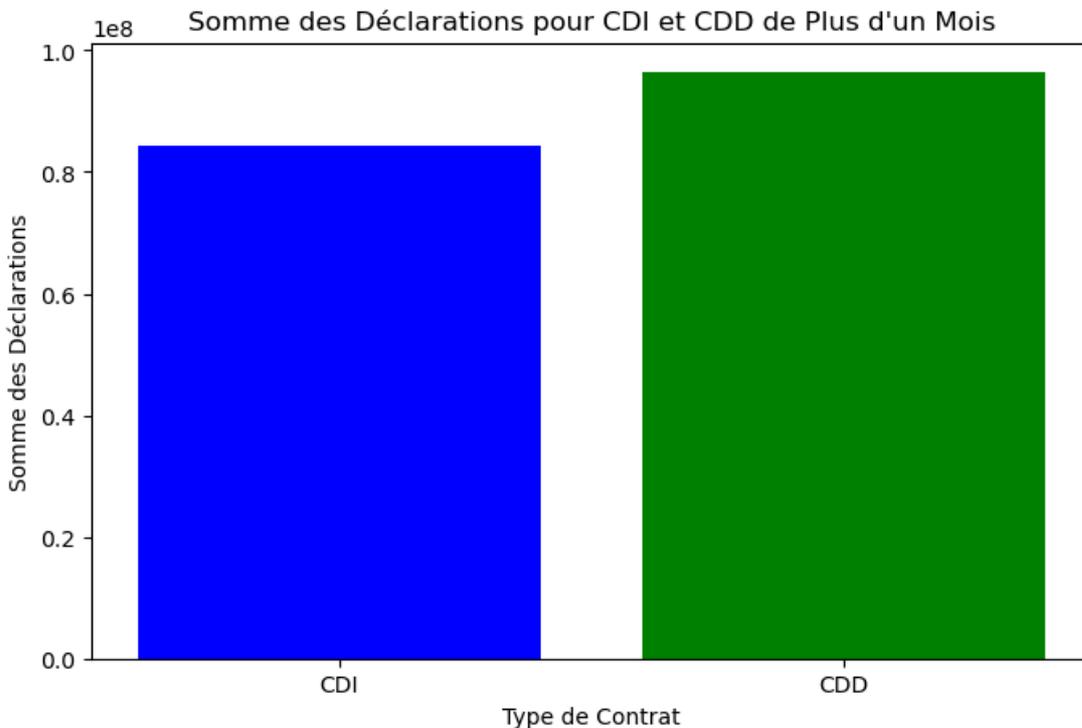
```
In [26]: cdi_data = data[data['nature_de_contrat'] == 'CDI']
```

```
In [32]: cdd_data = data[data['nature_de_contrat'] == "CDD de plus d'un mois"]
```

```
In [33]: cdi_sum = cdi_data['dpae_brut'].sum()
cdd_sum = cdd_data['dpae_brut'].sum()
```

```
In [35]: # Préparer les données pour le graphique
sum_data = pd.DataFrame({'Type de Contrat': ['CDI', 'CDD'], 'Somme des Déclarations': [cdi_sum, cdd_sum]})
```

```
In [36]: # Créer le graphique
plt.figure(figsize=(8, 5))
plt.bar(sum_data['Type de Contrat'], sum_data['Somme des Déclarations'], color=['blue', 'green'])
plt.title('Somme des Déclarations pour CDI et CDD de Plus d'un Mois')
plt.xlabel('Type de Contrat')
plt.ylabel('Somme des Déclarations')
plt.show()
```



```
In []:
```

```
In [39]: unique_contracts = data['nature_de_contrat'].unique()
```

```
results = {} # Pour stocker les résultats de la régression pour chaque type de contrat
```

```
for contract_type in unique_contracts:
    # Filtrer les données pour le type de contrat actuel
```

```
contract_data = data[data['nature_de_contrat'] == contract_type]

# Agréger les données par année
yearly_data = contract_data.groupby('annee')['dpae_brut'].sum().reset_index()

# Préparer les variables X et y pour la régression linéaire
X = yearly_data[['annee']] # Variable indépendante
y = yearly_data['dpae_brut'] # Variable dépendante

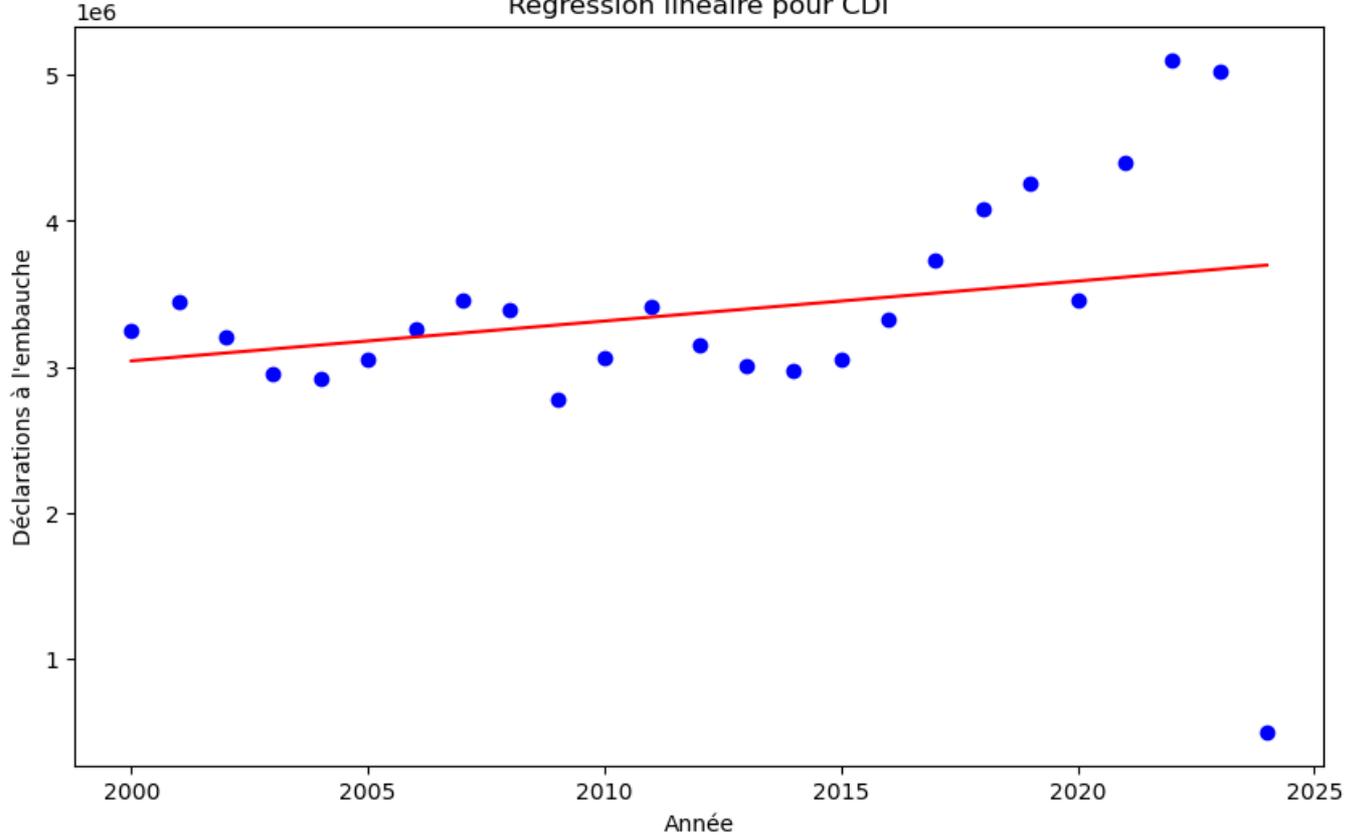
# Créer et entraîner le modèle de régression linéaire
model = LinearRegression()
model.fit(X, y)

# Stocker le modèle et les données pour le graphique
results[contract_type] = {'model': model, 'data': yearly_data}

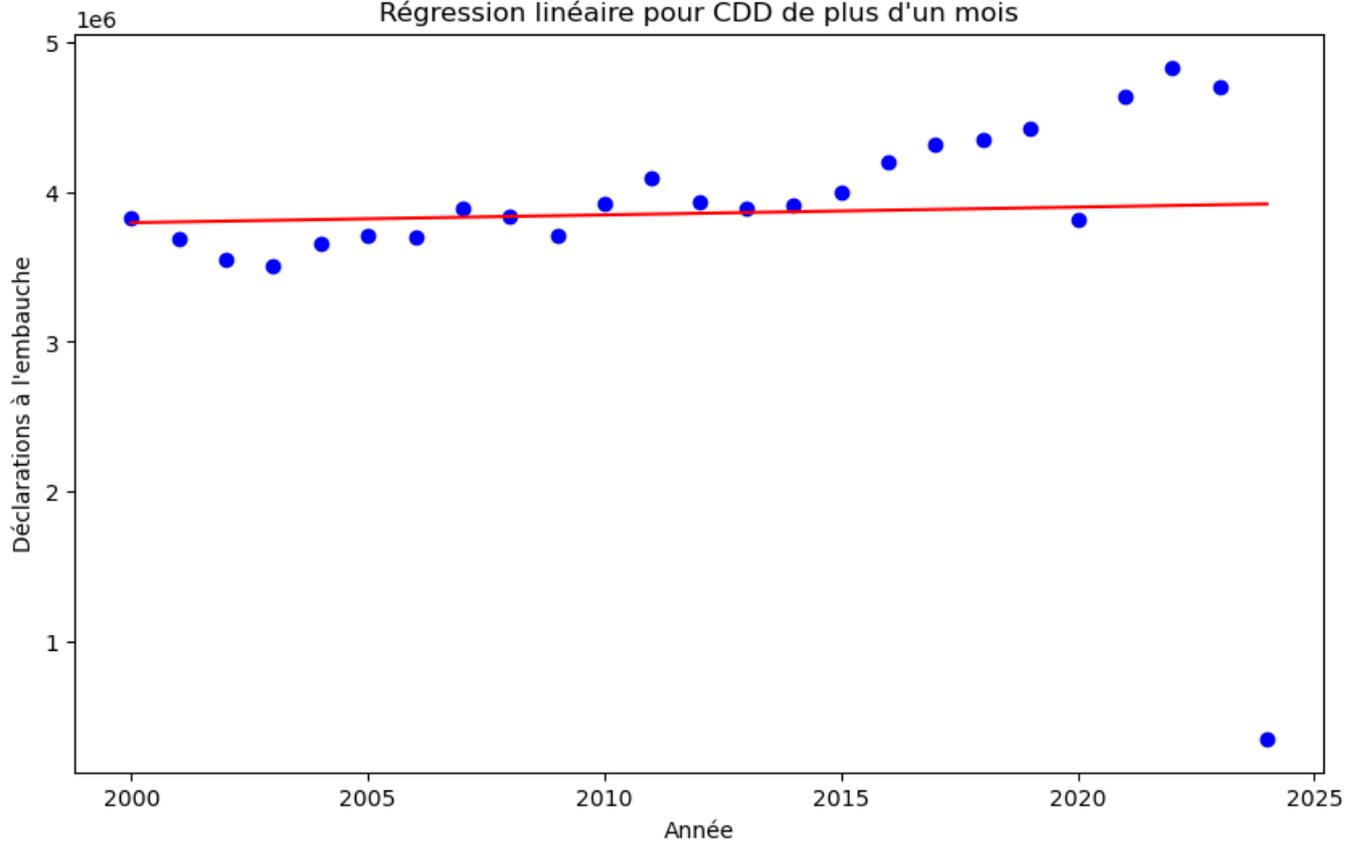
# Créer les graphiques pour chaque type de contrat
for contract_type, info in results.items():
    model = info['model']
    yearly_data = info['data']

    plt.figure(figsize=(10, 6))
    plt.scatter(yearly_data['annee'], yearly_data['dpae_brut'], color='blue')
    plt.plot(yearly_data['annee'], model.predict(yearly_data[['annee']]), color='red')
    plt.title(f'Régression linéaire pour {contract_type}')
    plt.xlabel('Année')
    plt.ylabel('Déclarations à l'embauche')
    plt.show()
```

Régression linéaire pour CDI



Régression linéaire pour CDD de plus d'un mois



In []:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js