

Notes : « CRCV – Prosocour » – Versailles

C. Parrinello, J. Grenier, J. Duhesme, M. Houllier, P. Chaumet-Riffaud

8 janvier 2026

1. Documents produits (Livrables)

- **Code source** : Notebooks Python contenant l'intégralité des scripts de traitement, de calcul et de génération de graphiques.
- **Visualisations** : Pages web interactives (format HTML) prêtes à l'emploi et graphiques exportés en images (PNG/JPG).
- **Données** : Tableaux récapitulatifs (format CSV) structurés pour faciliter les analyses ultérieures.

2. État des données (Dataset)

- **Un déséquilibre à corriger** : Les données actuelles sont fortement centrées sur la « Maison du Roi », rendant les autres services moins représentés.
- **Des lacunes identifiées** : Une part importante des fiches présente des manques concernant les dates précises et le détail des liens interpersonnels.
- **Solution** : Un croisement avec les données du projet Girophare (BnF) est souhaitable pour compléter ces manques et renforcer la fiabilité scientifique de l'étude.

3. Ce que l'étude permet de voir (Résultats)

- **Statistiques générales** : Tableaux de bord sur la répartition par sexe, le statut nobiliaire et l'évolution des effectifs par institution.
- **Cartes interactives** : Visualisation dynamique des réseaux d'influence des grandes dynasties et des interactions entre les services.
- **Suivi des carrières** : Analyse détaillée du parcours des individus et des mécanismes de transmission des charges au sein des familles.
- **Frise chronologique** : Outil de synthèse permettant d'identifier les périodes de coexistence des individus et la durée d'influence.

4. Conclusion et intérêt du projet

Les outils développés permettent une compréhension immédiate de l'organisation du pouvoir des familles à Versailles. En rendant les données visuelles et interactives, le projet met en lumière l'influence de certaines familles et leur évolution au sein des institutions de l'État.

5. Outils techniques

L'ensemble des traitements a été réalisé en Python, en s'appuyant sur l'écosystème suivant :

- **networkx** : Pour la création, la manipulation et l'étude de la structure et du dynamisme des réseaux complexes.
- **matplotlib** : Pour la production de graphiques statiques, de diagrammes et de visualisations de données en 2D.
- **plotly** : Pour la génération de visualisations interactives et dynamiques intégrables dans des pages web.
- **python-louvain** : Pour la détection de communautés et le clustering au sein des graphes de réseaux sociaux.
- **tqdm** : Pour l'intégration de barres de progression lors du traitement de jeux de données JSON volumineux.
- **numpy & pandas** : Pour la manipulation de DataFrames, le nettoyage de données et les calculs numériques intensifs.