

# Rapport - Assignment 2 : Visualisation Interactive de Données Multivariées

Author: LIU Yao

Projet : Analyse de la Criminalité aux États-Unis (1,994 communautés)

Technologies : React 18, D3.js 7.9, Redux Toolkit

## 1. Introduction et Architecture

Ce projet vise à explorer les relations entre les facteurs socio-économiques et les taux de criminalité pour identifier les zones les plus sûres. L'architecture repose sur une séparation claire : React gère l'état global et le layout, D3.js gère le rendu visuel (pattern Enter-Update-Exit), et Redux assure la synchronisation de l'état (sélections, brush) entre les différentes vues.

Variable cible : **ViolentCrimesPerPop** | Variable explicative principale : **medIncome**.

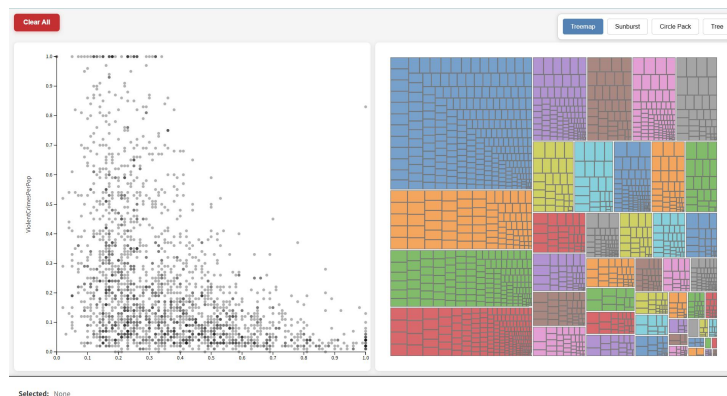


Figure 1: Interface principale avec scatterplot et treemap synchronisés.

## 2. Visualisation 1 : Scatterplot avec Brush 2D

Le scatterplot croise le revenu médian (X) et le taux de criminalité (Y). L'encodage visuel utilise une opacité de 0.3 pour éviter le surplot.

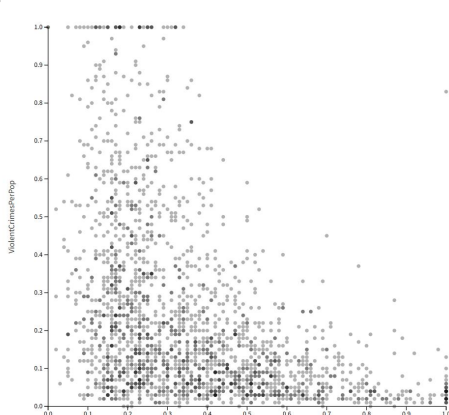


Figure 2: Scatterplot montrant la corrélation négative (↑revenu = ↓criminalité).

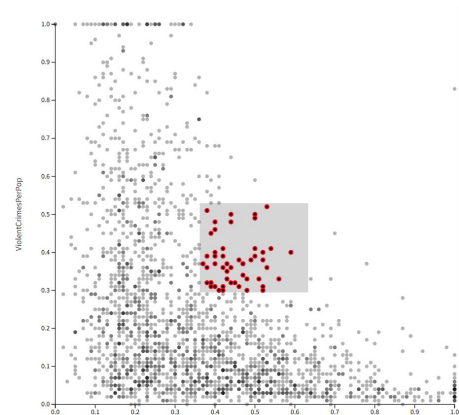


Figure 3: Sélection par brush ciblant les zones à haut revenu et faible criminalité.

Une interaction par Brush (d3-brush) permet de sélectionner une zone rectangulaire. Les points sélectionnés passent à une opacité de 1.0 avec bordure rouge, et l'action est envoyée à Redux.

## 3. Visualisation 2 : Hiérarchies (État → Communauté)

Quatre layouts hiérarchiques ont été implémentés pour visualiser la répartition géographique.

Critère	Treemap ★	Sunburst	Circle Pack	Tree
Densité	Excellente	Bonne	Moyenne	Faible
Comparaison	Très facile (Aires)	Difficile (Angles)	Moyenne	Difficile
Scalabilité	Très Haute	Moyenne	Faible	Nulle

Le Treemap est le layout recommandé car il offre la meilleure densité d'information, permettant d'identifier d'un coup d'œil les états les plus sûrs (ceux avec une majorité de petites tuiles).

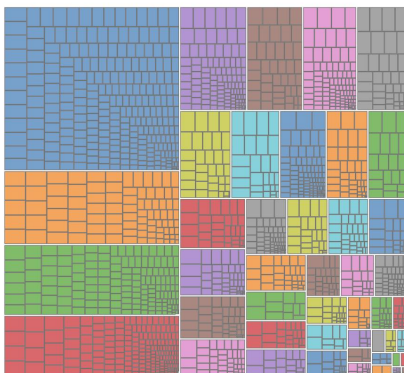


Figure 4: Layout Treemap optimal pour la comparaison des taux de criminalité par état.

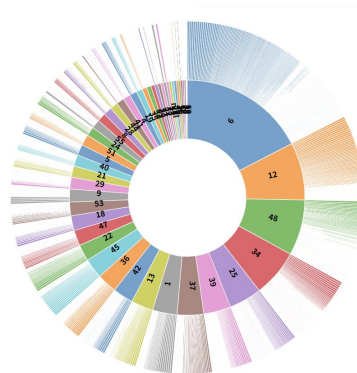


Figure 5: Layout Sunburst, plus esthétique mais sujet à la distorsion visuelle.

## 4. Interactions Synchronisées via Redux

La force de l'application réside dans la communication bidirectionnelle entre les vues.

Brush → Hiérarchie : Encadrer des points dans le scatterplot met en évidence les communautés correspondantes dans le Treemap.

Click État → Scatterplot : Cliquer sur un numéro d'état sélectionne toutes ses communautés et les affiche dans le scatterplot.

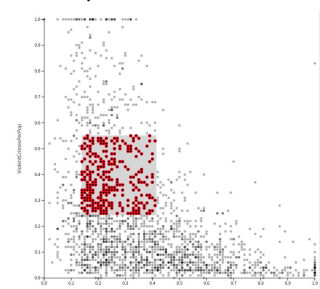


Figure 6: Les communautés encadrées dans le scatterplot s'illuminent dans la hiérarchie.

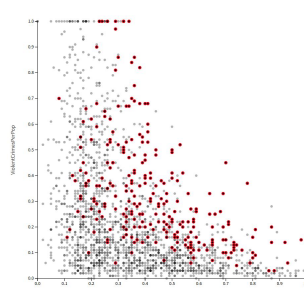
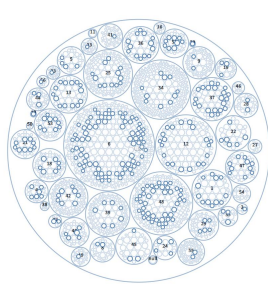


Figure 7: Un clic sur un état révèle la dispersion de ses communautés dans le scatterplot.

## 5. Conclusion

L'association d'une vue basée sur les attributs (Scatterplot) et d'une vue géographique (Treemap) offre un outil puissant. L'interaction fluide (brushing & linking) permet de répondre efficacement au problème posé : trouver les zones combinant sécurité et niveau de vie élevé.