

TP 2 : Analyse Factorielle des Correspondances (AFC)

Zaineb Smida

Vous disposez d'une base de données intitulée `assurance.txt`, qui provient d'une compagnie d'assurance et concerne 1776 jeunes conducteurs. Les variables d'intérêt sont les suivantes :

- **FORMULE** : formule du contrat d'assurance (A = "minimum", B = "moyenne", C = "maximum")
- **VALEUR** : valeur marchande du véhicule assuré, codée en 4 classes : (1 = "faible valeur", 2 = "valeur moyenne", 3 = "valeur élevée", 4 = "valeur très élevée")

Packages nécessaires :

```
library("FactoMineR")  
library("Factoshiny")  
library("RColorBrewer")
```

1 Question

Importer le fichier `assurance.txt` sous **R**. Vous pouvez l'appeler `conducteurs`.

Conseils

Utiliser la fonction `read.table()`. Si la première ligne du fichier `.txt` contient le nom des variables, ajouter l'option `header = T` pour le préciser. Vous pouvez vérifier que l'importation a été correctement réalisée en utilisant `View(conducteurs)`.

Solution

2 Question

Transformer la variable qualitative `valeur` en `factor` et mettre des labels.

Conseils

Lorsque vous utilisez `table(conducteurs$valeur)`, vous constatez que cette variable est codée avec des entiers allant de 1 à 4. En réalité, 1 correspond à la modalité **faible valeur**, 2 à **valeur moyenne**, 3 à **valeur élevée** et 4 à **valeur très élevée**. La fonction `factor()` permet d'assigner des étiquettes à ces entiers pour la variable `valeur`. Pour ce faire, utilisez l'option `labels =` suivie d'un vecteur contenant les étiquettes ordonnées selon les entiers.

Solution

3 Question

Construire la table de contingence des deux variables qualitatives.

Conseils

Vous pouvez utiliser la fonction `table()` sur deux variables qualitatives afin d'obtenir la table de contingence des effectifs.

Solution

4 Question

Afficher les profils lignes et les profils colonnes.

Conseils

Pour obtenir les profils lignes (resp. colonnes), divisez les effectifs (la table créée précédemment) par la somme des lignes (resp. colonnes). Pour cela, utilisez la fonction `prop.table()` avec l'argument `margin = 1` (resp. `margin = 2`).

Solution

5 Question

Statistique bivariée : apprécier le lien entre les variables à l'aide d'un outil graphique vu en cours.

Conseils

Pour représenter graphiquement les profils lignes ou colonnes, utilisez la fonction `barplot()`, appliquée sur les profils créés précédemment. Il existe plusieurs options pour personnaliser le graphique. Par exemple l'option `beside = T` permet de représenter les barres de manière juxtaposée.

Solution

6 Question

Les deux variables sont-elles liées ? Utiliser le test statistique vu en cours permettant de répondre à cette question. Interpréter les sorties de R.

Conseils

Vous pouvez utiliser la fonction `chisq.test()`

Solution

7 Question

Réaliser l'étude des écarts et des contributions au χ^2 .

Conseils

Vous pouvez extraire les écarts de l'objet retourné par la fonction `chisq.test()` en utilisant la syntaxe `$residuals`.

Solution

8 Question

Réaliser l'AFC sur les deux variables. Combien d'axes sont à retenir ? Justifier votre réponse.

Conseils

Pour réaliser une **AFC** sous **R**, chargez le package **FactoMineR** avec la commande `library(FactoMineR)` et utilisez la fonction `CA()`. Pour le moment, nous ne nous intéressons pas à afficher les résultats de l'AFC. Toutefois, vous pouvez utiliser l'argument `graph =` pour activer l'affichage graphique si nécessaire.

Solution

9 Question

Quels sont les profils ayant fortement contribué à l'apparition du (des) axe(s) retenu(s) ? Justifier votre réponse.

Conseils

On accède aux résultats de l'AFC en utilisant les syntaxes `rowcoord` et `colcoord` pour les coordonnées, ainsi que `rowcontrib` et `colcontrib` pour les contributions.

Solution

10 Question

Quels sont les profils bien représentés ? Justifier votre réponse.

Conseils

On accède aux cosinus 2 en faisant `rowcos2` et `colcos2` pour étudier la qualité de la représentation.

Solution

11 Question

Réaliser le graphique simultané et commenter l'AFC.

Conseils

Pour réaliser le graphique, vous pouvez modifier l'argument `graph =` de la fonction `CA()`.

Solution

12 Question

Refaire l'analyse en utilisant la fonction `CAshiny()` du package **Factoshiny**:

Solution