TP 04 :

Explication des Algorithmes et Tests :

Uniformity :

Vérifie si une séquence de nombres aléatoires suit une distribution uniforme

Chi-Deux :

`testUniformityWithChiSquare` (Test du Khi-Deux pour l'Uniformité) :

Étapes Clés :

Calculer les Observations :

- Divise l'intervalle [0, 1] en k sous-intervalles égaux.
- Compte le nombre de valeurs dans chaque intervalle.

Calculer le Khi-Deux :

- expected: Nombre de valeurs attendues dans chaque intervalle pour une distribution uniforme (n*pj).
- Statistique $\chi 2$: Somme des carrés des écarts normalisés entre observed et expected.

Valeur Critique :

- Appelle `findTheCriticalValueChiDeuxMatrix` pour obtenir la valeur critique basée sur les degrés de liberté (df=k−1df = k-1df=k−1) et α\alphaα.

Décision :

- Si $\chi 2$>valeur critique, rejeter H0 (la distribution n'est pas uniforme).
- Sinon, ne pas rejeter H0 (la distribution est uniforme).

Kolmogorov:

`testUniformityWithKolmogorov` (Test de Kolmogorov-Smirnov pour l'Uniformité):

Étapes Clés :

- **Calculer les Différences** :

  - Trie les nombres aléatoires.
  - Calcule deux différences principales pour chaque point :
    - D1 : Différence entre ri (valeur cumulée empirique) et ri précédent .

- D2 : Différence absolue entre i/n (valeur cumulée théorique) et ri .

- **Trouver la Différence Maximale** :
  - Prend le maximum entre D1 et D2.
- **Valeur Critique** :
  - Appelle `findTheCriticalValueKsMatrix` pour récupérer la valeur critique basée sur n et α.
- **Décision** :
  - Si max(D1,D2)>valeur critique> valeur critique, rejeter H0H_0H0 (la distribution n'est pas uniforme).
  - Sinon, ne pas rejeter H0 (la distribution est uniforme).

Independence :

`testRunAndTestUp` (Test des Runs pour l'Indépendance) :

Vérifie l'indépendance d'une séquence de nombres aléatoires.

Étapes Clés :

Compter les Runs (R).

Calculer la Statistique Z :

- E : Nombre attendu de runs pour une séquence indépendante.
- V: Variance du nombre de runs.
- Z: Statistique Z basée sur R, E et V.

Valeur Critique : Appelle `findTheCriticalValuNormalDistrubtionMatrix` pour récupérer la valeur critique correspondant à α\alphaα dans la loi normale.

Décision :

- Si |Z|> valeur critique, rejeter H0 (les valeurs sont indépendantes).
- Sinon, ne pas rejeter H0 (les valeurs ne sont pas indépendantes).

AFFICHAGE :

Briefly :

```
> testUniformityWithChiSquare(random_numbers,alpha)
La distribution est uniforme (ne pas rejeter H0) - Chi-Square.
>
> # En utilisant la méthode de Kolmogorov
> testUniformityWithKolmogorov(random_numbers,alpha)
La distribution est uniforme (ne pas rejeter H0) - Kolmogorov-Smirnov.
>
> # commencant par l'independence
> testRunAndTestUp(random_numbers,alpha)
Les valeurs ne sont pas indépendantes (ne pas rejeter H0) - Test des runs.
>
```

Nombres Aleatoires & Alpha:

```
random_numbers <- c( 0.2883475, 0.4089769, 0.8830175 ,0.5706998,
0.3329115, 0.2519897 ,0.9005089 ,0.3790737 ,0.5047260, 0.8135167)
alpha <- 0.05
```