TP Logiciels statistique (R)

10/01/2025

BAC Results Over the last Decade

Répondez à ce TP en produisant un PDF via R Markdown, à partir des données disponibles à cette URL :

`https://github.com/binorassocies/rimdata/raw/refs/heads/main/data/bac_results_2015-2024.csv`

Préparation des données

1. Quelle est l'erreur avec la candidate dont la ligne est 430933 ? Affectez à la colonne erronée 2005.
2. Pour gérer les données manquantes dans les champs `"birth_year", "birth_place", "wilaya"` pour l'année 2020, nous allons imputer de manière aléatoire (hot-deck) à partir des autres années.

```
set.seed(2025)
x = bac_results %>% filter(year != 2020) # isoler 2020
bac_results <- bac_results %>%
  mutate(across(all_of(c("birth_year", "birth_place", "wilaya")),
         ~ sample(na.omit(x[[cur_column()]]), n(),
     replace = TRUE))) # hotdeck
```

3. Ajouter une colonne age et remplacer les valeurs inférieurs à 7 par la médiane de l'age.

Analyse descriptive

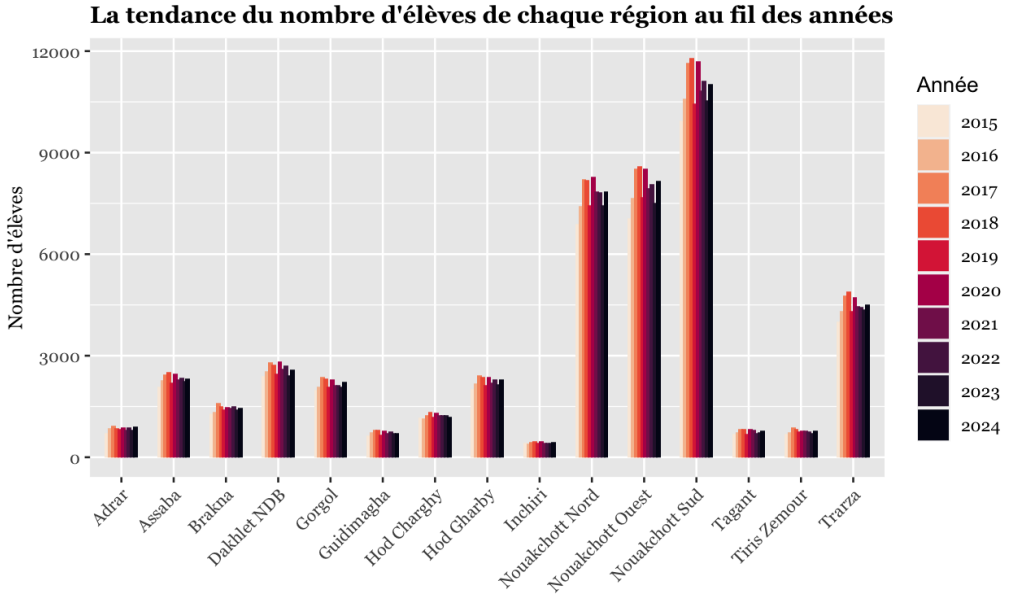
1. Quel est le nombre total d'élèves par année ?
2. Reproduire ce tibble :

A tibble: 10 × 6

year <int>	LM <int>	LO <int>	M <int>	SN <int>	TM <int>
2015	9000	9609	2806	20465	106
2016	9539	9181	3311	22888	107
2017	11147	9524	3505	25730	87
2018	11758	8816	2976	26403	92
2019	10044	8209	1941	24321	202
2020	10813	9021	1862	27881	178
2021	9517	7376	1383	28211	100
2022	8776	7084	1367	29803	221
2023	7579	7404	1402	28223	253
2024	7322	9069	1427	29071	328

1–10 of 10 rows

3. Créer un tableau croisé montrant la moyenne des notes par année et par région.
4. Identifier la note minimale, la note maximale et la note médiane pour chaque année.
5. Afficher les 10 lieux de naissance les moins fréquents.
6. (a) Tracer un histogramme du nombre d'élèves par série, colorié en fonction de la décision.
   (b) Comme le nombre de candidats n'est pas proportionnel entre les séries, reproduire le même graphique mais en proportion.
7. Créer un graphique en camembert, avec ggplot, montrant la répartition des candidats pour chaque série.
8. Faire un tableau des élèves ayant la meilleure note pour chaque année et chaque série. Identifier le plus jeune et le plus âgé des majors.
9. Comparer le taux de succès des candidats de moins de 19 ans et des candidats plus âgés au fil des années.
10. Comment la distribution des âges des candidats a-t-elle changé au fil des années ? (Utilisez `ggridges`)
11. Reproduire ce graphique :



12. Créer une carte affichant le groupe d'âge dominant parmi les admis dans chaque wilaya. Ajouter les groupes d'âges suivants : moins de 19, [19,22[, [22,25[, [25,28[, plus de 28.
13. Reproduire ce tibble :

A tibble: 2 × 4

region <chr>	total_candidates <int>	total_admitted <int>	pass_rate <dbl>
Interior	200599	22800	11.36596
Nouakchott	266839	30420	11.40013

2 rows

Analyse des noms

1. Créer une fonction pour compter les occurrences des prénoms dans le dataset et l'essayer avec le votre.
2. Compter les prénoms qui apparaissent une seule fois dans le dataset.
3. Générer un nuage de mots pour les prénoms, utilisez `library(wordcloud)` et fixez ces paramètres : `ax.words = 100, random.order = FALSE`.
4. Dans quelle wilaya les prénoms ont-ils, en moyenne, le plus grand nombre de caractères ?
5. Créer une fonction qui trace la popularité d'un prénom au fil des années de naissance, ayant le prénom comme unique argument. Testez-la avec le prénom Mariem.

Analyse de probabilité

1. Calculer la probabilité globale de réussir le Bac sur les 10 dernières années.
2. Calculer la probabilité de réussir le Bac en fonction de l'âge des élèves.
3. Identifier la série ayant le plus grand écart entre les taux de réussite et d'échec.
4. Analyser la probabilité de réussite pour la série M avant et après la pandémie.
5. Comparer les probabilités de réussite des élèves ayant des prénoms similaires (exemple pour tester : tous les Abdellahi puis toutes les Binta).