Mini-projets de R

Ivan Kojadinovic
Université de Pau et des Pays de l'Adour

Septembre 2024

Chaque section ci-dessous correspond à un mini-projet R . Des éléments de correction vous seront donnés à la fin du temps imparti pour chaque mini-projet. Ces mini-projets ne seront pas notés mais vous aurez probablement des questions les concernant dans l'évaluation finale de cette matière. Afin d'acquérir le plus de compétences possibles, il vous est très fortement conseillé de faire ces mini-projets individuellement sans tenter de trouver des solutions complètes sur le web et sans vous servir de systèmes d'IA générative pour générer du code.

1 Illustration empirique de résultats théoriques clés

Soit Z une variable aléatoire telle que $\mathbb{V}[Z] < \infty$ et soient $Z_1, \ldots, Z_n$ n copies indépendantes et identiquement distribuées (i.i.d.) de Z. On dit aussi que $Z_1, \ldots, Z_n$ est un *échantillon aléatoire* de Z. On notera $z_1, \ldots, z_n$ une réalisation de l'échantillon aléatoire $Z_1, \ldots, Z_n$

1. Dans le cadre de l'estimation de $\mathbb{E}[Z]$, rappeler brièvement les notions de paramètre, d'estimateur et d'estimation. Déterminer l'espérance et la variance de l'estimateur usuel $\bar{Z}_n$ de $\mathbb{E}[Z]$.

2. Qu'appelle-t-on l'*erreur standard* de $\bar{Z}_n$? Quel est l'estimateur usuel de cette quantité ?

3. Si Z suit par exemple une loi exponentielle de paramètre $\lambda = 2$, une réalisation d'un échantillon aléatoire de Z de taille $n = 10$ peut être obtenue en tapant `rexp(10, rate = 2)`. Chaque exécution de cette instruction donne un résultat différent [1]. Cela permet par exemple de simuler l'observation d'un grand nombre de réalisations de l'échantillon aléatoire $Z_1, \ldots, Z_n$.
   (a) Si Z suit une loi normale (resp. une loi gamma) de paramètres fixés, quelle fonction faut-il utiliser pour obtenir des réalisations d'un échantillon aléatoire de Z. Donner des exemples.

1. Sauf si elle est précédée d'un appel à la fonction `set.seed()` avec le même argument.

(b) Étant donnée une réalisation de l'échantillon aléatoire $Z_1, \ldots, Z_n$ de Z (d'une loi exponentielle, normale, gamma, etc, de paramètres fixés), comment obtenir la réalisation correspondante de $\bar{Z}_n$ ainsi que de l'estimateur usuel de son erreur standard ? Donner des exemples.

(c) Expliquer ce qu'illustre le code suivant :

```
par(mfrow = c(1, 2))
n <- 1000
z <- rexp(n, rate = 2)
hist(z, freq = FALSE, breaks = 30)
curve(dexp(x, rate = 2), add = TRUE, col = "red")
curve(qexp(x, rate = 2), from = 0, to = 5, col = "blue")
curve(pexp(x, rate = 2), add = TRUE, col = "brown")
abline(0, 1, lty=2)
```

En particulier, expliquer ce que permettent d'évaluer les fonctions R du type `d + abrev`, `q + abrev` et `p + abrev`, où `abrev` est l'abréviation R de la loi manipulée (par exemple `exp`, `norm`, `gamma`, etc) et `+` représente la concaténation de chaînes de caractères. Donner des exemples.

(d) Relativement au code précédent, de quelle valeur est-ce que `mean(z <= 1.2)` (resp. `quantile(z, probs = 0.9)` devrait s'approcher si on augmente `n`.

4. Dans le cadre de l'estimation de $\mathbb{E}[Z]$, rappeler la loi forte des grands nombres, le théorème central limite et leurs conséquences sur $\bar{Z}_n$.

5. Dans le cas où Z suit une loi exponentielle de paramètre $\lambda = 2$, écrire du code R permettant d'illustrer empiriquement :

   (a) le fait que $\bar{Z}_n$ soit un estimateur sans biais pour $\mathbb{E}[Z]$ et que $\mathbb{V}[\bar{Z}_n] = \mathbb{V}[Z]/n$;

   (b) le fait que, lorsque "n est grand", $\bar{Z}_n$ suit approximativement une loi normale de paramètres $\mathbb{E}[Z]$ et $\mathbb{V}[Z]/n$.

6. Reprendre le code du (b) de la question précédente et en faire une fonction `illusTcl()` prenant en argument, entre autres, une fonction R permettant de générer des réalisations d'échantillons aléatoires d'une loi choisie. Cet argument pourra par exemple prendre la valeur :

   ```
   rMaLoiExp4 <- function(n) rexp(n, rate = 4)
   ```

   Appelez la fonction `illusTcl` pour illustrer empiriquement le théorème central limite lorsque Z suit une loi gamma que vous aurez choisie pour `n` égal à 3, 5, 30 et 100. Que constatez-vous ?

2 Les k plus proches voisins en régression

On considère un problème de régression théorique univarié dans lequel la variable *à expliquer* Y est donnée en fonction de la variable *explicative* X par l'expression :

$$Y = \sin(X) + \ln(X) + \epsilon, \qquad \text{presque sûrement,}$$

où X est une variable aléatoire uniformément distribuée sur $]0, 10[$ et ϵ est une variable aléatoire normale centrée réduite indépendante de X. La variable ϵ dans l'expression précédente représente une incertitude sur la relation entre Y et X.

1. Écrire une fonction R permettant de générer n réalisations (x_1, y_1), ..., (x_n, y_n) du vecteur aléatoire (X, Y). Ces réalisations seront stockées en ligne dans un `data.frame` de colonnes `x` et `y`.

2. Générer $n = 100$ réalisations de (X, Y) avec la fonction R précédente et donner du code R permettant de représenter le nuage de points correspondant. Proposer à la fois du code R "historique" et du code basé sur `ggplot`.

3. Pour $x \in]0, 10[$, donner l'expression de l'espérance conditionnelle $\mathbb{E}[Y|X = x]$. Modifier ensuite le code de la question précédente afin de superposer le graphe de la fonction f donnée par $f(x) = \mathbb{E}[Y|X = x]$, $x \in]0, 10[$, sur les nuages de points.

4. Dans un problème de régression pratique, la fonction f est inconnue et on ne dispose que des n réalisations (x_1, y_1), ..., (x_n, y_n) du vecteur aléatoire (X, Y). L'objectif est alors d'utiliser les données disponibles afin d'obtenir un modèle pour prédire "au mieux" Y lorsque que $X = x$. Un modèle de régression conceptuellement simple repose sur la méthode des k *plus proches voisins*. Soit $k \in \{1, \ldots, n\}$ fixé et, pour tout x et $i \in \{1, \ldots, n\}$, soit $d_i^x = |x - x_i|$. Notons $d_{(1)}^x, \ldots, d_{(n)}^x$ un ré-arrangement de $d_1^x, \ldots, d_n^x$ tel que $d_{(1)}^x \leq \cdots \leq d_{(n)}^x$, et soit $\mathcal{N}_k(x) = \{i : d_{(i)}^x \leq d_{(k)}^x\}$. La prédiction de Y lorsque $X = x$ par la méthode des k plus proches voisins est alors donnée par

$$\hat{f}_k(x) = \frac{1}{k} \sum_{i \in \mathcal{N}_k(x)} y_i. \tag{1}$$

   (a) Écrire une fonction R prenant en entrée les données disponibles, une valeur de $k \in \{1, \ldots, n\}$ et une valeur x de X, et renvoyant la prédiction de Y lorsque $X = x$ par la méthode des k plus proches voisins, c-à-d, $\hat{f}_k(x)$ dans (1). Notez que, comme il n'y a qu'une variable explicative, vous pourriez être tentés de pré-traiter les données en triant les réalisations disponibles de (X, Y) selon les valeurs de X. Afin de pouvoir facilement généraliser le code au cas de plusieurs variables explicatives, il vous est suggéré de plutôt utiliser la fonction R `sort()` avec l'argument `index.return` à `TRUE`.

   (b) Afin de tester la fonction implémentée pour $k = 10$, créer un vecteur R contenant les valeurs 0.1, 0.2, ..., 9.9 et calculer la prédiction de Y pour ces valeurs de

X. En partant des éléments précédents, donner ensuite une représentation graphique approximative de $\hat{f}_k$ en utilisant `plot(..., type = "l")` (R "historique") et `geom_line(...)` (ggplot). Superposer enfin les graphes approximatifs précédents aux nuages de points avec graphes de f existants.

(c) La fonction R précédente n'est pas "vectorisée" dans le sens où on ne peut pas directement l'appeler pour obtenir les prédictions de Y pour un vecteur de valeurs de X. En donner une version vectorisée la plus élégante possible (en utilisant par exemple `sapply()`). Le code pourra tester que k est bien dans $\{1, \ldots, n\}$ (en utilisant par exemple `stopifnot()`).

(d) Le fait de disposer d'une fonction vectorisée permet notamment de l'utiliser directement avec la fonction `curve()`. Par exemple, en supposant que le nom de la fonction vectorisée soit `f.hat.vect` :

```
plot(d)
curve(sin(x) + log(x), add = TRUE, col = "red")
curve(f.hat.vect(d = d, k = 10, x), add = TRUE, col = "blue")
```

Représenter alors, dans une matrice de graphiques 2 x 2, des graphiques similaires pour $k = 1$, 5, 10 et 50. Quelle valeur de k suggérez-vous ? Quelle(s) situation(s) qualifieriez-vous de "sur-apprentissage"/"sous-apprentissage" ? Donner enfin du code R permettant de réaliser une matrice similaire de graphiques en utilisant `ggpplot`.