TP2 : Modèles paramétriques pour les durées de vie avec ou sans covariables

Vous rédigerez un compte-rendu de ce TP dans son intégralité.

1 Modèle exponentiel

Soit X une variable aléatoire représentant une durée de vie (durée s'écoulant jusqu'à la survenue d'un événement d'intérêt), de loi exponentielle de paramètre $\lambda = 1$.

1.1 Simulation d'un échantillon avec contrôle du taux d'observations censurées

► Simuler à l'aide de la fonction `rexp()` un échantillon de taille $n = 50$ de couples (X_i, C_i), $i = 1, \cdots, n$, avec X_i indépendant de C_i, de lois exponentielles de paramètres respectifs $\lambda = 1$ et $\mu = 0.5$.

► Définir un vecteur $T = \min(X, C)$ et un vecteur $\delta = \mathbb{I}(X \leq C)$. On pourra utiliser `as.numeric(X==T)`.

► Déterminer $\mathbb{P}(X > C)$ par un calcul exact.

► Calculer la proportion observée de censure sur l'échantillon simulé c'est-à-dire

$$1 - \frac{1}{n}\sum_{i=1}^{n} \delta_i.$$

Répéter plusieurs fois ce calcul pour plusieurs réalisations de l'échantillon et observer la fluctuation de la proportion de censure.

1.2 Estimation du paramètre λ

► Calculer une estimation ponctuelle du paramètre λ pour l'échantillon simulé.

► Reprendre la simulation de l'échantillon censuré de la question **1)** et faire varier la taille n de l'échantillon.

Pour n allant de 10 à 5000, calculer les estimateurs $\hat{\lambda}_n$ et les représenter graphiquement en fonction de n. On représentera aussi sur le même graphique la vraie valeur du paramètre $\lambda = 1$. Observer la convergence de $\hat{\lambda}_n$.

► Démontrer que :

$$\sqrt{n}\left(\hat{\lambda}_n - \lambda\right) \xrightarrow{\mathscr{L}} \mathscr{N}\left(0, \frac{\lambda^2}{\mathbb{P}(\delta_1 = 1)}\right)$$

En déduire un intervalle de confiance pour le paramètre λ.

► Pour n allant de 10 à 5000, calculer les bornes de l'intervalle de confiance, pour chaque valeur de n et les ajouter au graphique.

► Recommencer avec un échantillon simulé dont le taux de censure est de 50 %.

1.3 Estimation paramétrique de la fonction de survie

On construit un estimateur paramétrique $\hat{S}_n(t) = \exp(-\hat{\lambda}_n t)$ de la fonction de survie $S(t)$ dans le modèle exponentiel.

► Pour un échantillon de taille $n = 30, 50, 100, 500$, et un taux de censure correspondant à la simulation de 1.1, calculer les valeurs de l'estimateur $\hat{S}_n(t) = \exp(-\hat{\lambda}_n t)$ pour t un vecteur de pas 0.01 allant de 0 à 3.

► Tracer la courbe représentative de $\hat{S}_n(t)$ et superposer la courbe théorique $S(t)$ pour les différentes tailles d'échantillons.

► On admet que

$$\sup_{t\geq 0} |\hat{S}_n(t) - S(t)| \leq \frac{|\hat{\lambda}_n - \lambda|}{\max(\lambda; \hat{\lambda}_n)}$$

en déduire une bande de confiance pour $S(t)$ (uniforme en t).

► Comparer graphiquement l'estimateur de Kaplan-Meier de la fonction de survie et l'estimateur $\hat{S}_n(t)$ pour $n = 30, 50, 100, 500$.

2 Modèle Weibull

On s'intéresse maintenant à une variable aléatoire positive qui suit la loi de Weibull $\mathscr{W}(a, b)$ de densité :

$$f(x) = (a/b)(x/b)^{a-1} \exp(-(x/b)^a, x > 0$$

où $a, b > 0$ paramètres de forme et d'échelle (paramétrage de R)

2.1 Estimation des paramètres avec la fonction "survreg"

► Simuler à l'aide de la fonction `rweibull()` un échantillon de X_i, $i = 1, \cdots, n$, de loi de Weibull $\mathscr{W}(a, b)$ de paramètres $a = 2$ et $b = 5$ et $n = 50$.

► On pose $Y = \log(X) = \mu + \sigma W$ où W admet pour fonction de répartition $F_W(x) = 1 - e^{-e^x}$, $x \in \mathbb{R}$. Exprimer les paramètres a et b de la loi de X en fonction de μ et σ.

► Utiliser la fonction de R `survreg` du package `survival` pour estimer μ et σ. Si `X` est le vecteur qui contient l'échantillon de loi de $\mathscr{W}(a, b)$:

```
model<-survreg(Surv(X)~1, dist="weibull")
summary(model)
```

Observez ce que contiennent :

– `model$coefficients`

– `model$scale`

– `model$var`

Donner des intervalles de confiance asymptotiques de μ et σ pour un niveau de confiance de 95%.

► Déduire une estimation ponctuelle de a et b de la loi $\mathscr{W}(a, b)$.

► La fonction `deltamethod` du package *msm* permet d'estimer la matrice de variance-covariance d'un vecteur $g(U_n)$ à partir de celle de U_n pour g une fonction convenablement choisie. La syntaxe est la suivante si $g : (x_1, x_2) \mapsto g(x_1, x_2) = (g_1(x_1, x_2), g_2(x_1, x_2))$:

```
deltamethod(list(~(g1(x1,x2),~g2(x1,x2)),Un,Var(Un),ses=FALSE)
```

où `Var(Un)` est la matrice de variance-covariance estimée du vecteur `Un`. L'option `ses=FALSE` renvoie la matrice de variance-covariance, si `ses=TRUE` les écarts-types estimés.

► Donner des intervalles de confiance asymptotiques de a et b pour un niveau de confiance de 95% obtenus par la δ-méthode.

2.2 Estimation des paramètres avec la fonction "flexsurvreg"

On reprend la même simulation (fixer la graine!) et on utilise une autre fonction `flexreg` du package `flexsurv` qui est un peu plus facile d'utilisation, mais davantage "boîte noire". De plus, elle permet de faire des graphiques facilement. La syntaxe est la suivante :

```
model2  <- flexsurvreg(Surv(X)~1, dist="weibull")
model2
plot(model2,type="survival",est=TRUE,ci=TRUE)
```

► Comparer les résultats obtenus avec `model2` et ceux précédemment obtenus. Expliquer l'utilité pratique de superposer l'estimateur de Kaplan-Meier sur la fonction de survie estimée dans le modèle de Weibull.

► Reprendre la simulation avec un échantillon censuré. On pourra conserver la même loi $\mathcal{W}(a, b)$ pour l'échantillon $(X_i)_{i=1,\cdots,n}$ et simulera l'échantillon $(C_i)_{i=1,\cdots,n}$ selon une loi exponentielle de paramètre μ calibré de façon à obtenir environ 25% de censure pour $n = 50$, puis pour $n = 100$.

3 Autres modèles paramétriques

Reprendre rapidement le paragraphe précédent pour une autre loi de X de votre choix (log-logistique, log-normale, gamma, etc).

4 Ajout d'une covariable

On s'intéresse maintenant à un modèle de régression avec une covariable Z.

$$Y = \log(X) = \mu + \gamma Z + \sigma W$$

4.1 Simulation et estimation des paramètres

► Simulation.

- Générer des variables $(W_i)_{i=1,\cdots,n}$ de fonction de répartition $F_W(x) = 1 - \exp(-\exp(x))$, $x \in \mathbb{R}$, pour n=100.
- Puis générer les $(X_i)_{i=1,\cdots,n}$ selon le modèle de régression log-linéaire $\ln(X_i) = \mu + \gamma Z_i + W_i$ avec $\mu = 2$, $\gamma = 3$ et $\sigma = 0.5$ et pour une covariable binaire Z telle que $Z_i = 0$ pour $i = 1, \cdots, n/2$ et $Z_i = 1$ pour $i = n/2+1, \cdots, n$ (prendre n pair).

► Générer un échantillon censuré avec l'échantillon $(C_i)_{i=1,\cdots,n}$ selon une loi exponentielle de paramètre $\mu = 0.01$.

► Utiliser la fonction `survreg` pour donner une estimation de μ, γ et σ.

► Modifier la valeur de $n = 100, 200, 500$ et le taux de censure (avec $\mu = 0.01, 0.003$) et faire une étude de la sensibilité (en examinant par exemple les écarts-types des estimateurs).

4.2 Examen des résidus de Cox-Snell

► Déterminer les résidus de Cox-Snell $(R_i)_{i=1,\cdots,n}$ associés à l'hypothèse du modèle de Weibull où $R_i = \hat{\Lambda}_{\hat{\mu},\hat{\gamma},\hat{\sigma}}(T_i)$.

► A l'aide de la fonction `survfit`, représenter graphiquement l'estimateur de Nelson-Aalen de la fonction de risque cumulé de l'échantillon censuré (R_i, δ_i) et superposer la droite d'équation $y = x$ correspondant à la fonction de risque cumulé d'une loi exponentielle de paramètre 1.

► Examiner l'adéquation du modèle de Weibull à l'aide de ce graphique.

5 Application au jeu de données réelles "pharmacoSmoking".

► Proposer une analyse statistique de ces données en ajustant plusieurs modèles de régression (avec hypothèse de loi Weibull, log-logistique et log-normale) pour une ou plusieurs covariables de votre choix. En particulier :

► Examiner les valeurs des AIC pour le choix de modèle

► Superposer les courbes de survie ajustées avec les trois modèles paramétriques et avec l'estimateur de Kaplan-Meier (on pourra aussi choisir de comparer les courbes de risques cumulés). Utiliser `flexsurvreg` pour les graphiques et convertir la covariable en facteur afin de pouvoir superposer les courbes des deux groupes avec la fonction `plot`.

► Examiner les résidus de Cox-Snell pour les trois modèles paramétriques.

► Examiner éventuellement d'autres modèles de lois disponibles dans `flexsurvreg` (visualisation graphique seulement ...).

► Conclure en faisant le lien avec le TP1.