

survivalGPU : Analyses de survie sur cartes graphiques

Alexis van STRAATEN¹ Jean FEYDY² Anne-Sophie JANNOT^{2,3}

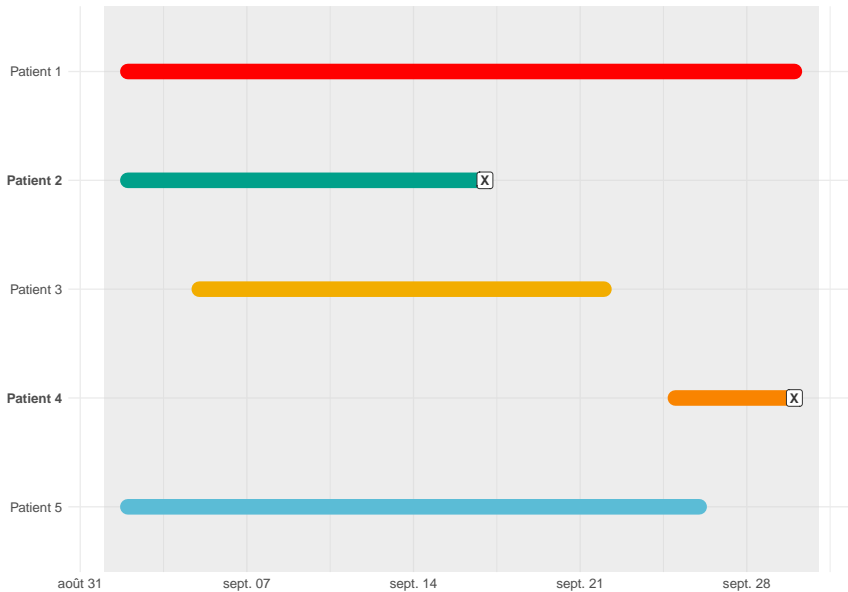
¹Assistance Publique-Hôpitaux de Paris (AP-HP), Service d'informatique Médicale, Biostatistiques Et Santé Publique, Hôpital Européen Georges Pompidou, Paris

²HeKA, Centre de Recherche des Cordeliers, INSERM, INRIA, Paris

³Cellule opérationnelle de la Banque Nationale de Données Maladies Rares, AP-HP, Paris



L'analyse de survie



L'analyse de survie

Modèle : Cox (1972)

Implémentation sur R : package `survival` de Terry Therneau, avec la fonction `coxph()`

- ▶ Etude sur la survenue d'un évènement.

Actuellement dans la recherche médicale:

- ▶ Accès à de larges bases de données de santé & développement d'entrepôts.
- ▶ Les méthodes sont acceptables pour des petites études, mais non utilisables pour des études à grande échelle.

Solution : utiliser les ressources des cartes graphiques.

Construction de survivalGPU

Etapes de construction du package :

1. Profilage du package `survival`
2. Réécriture d'un nouveau solveur de Cox en python avec `pytorch` pour la compatibilité avec les cartes graphiques Nvidia
3. Construction du package R avec les fonctions python en utilisant `reticulate`
4. Tests unitaires : reprise des tests de `survival` et résultats identiques entre `survival` et `survivalGPU`

Implémentation de 2 modèles :

- ▶ Cox
- ▶ WCE (Weighted Cumulative Exposure) pour la pharmacovigilance

Plus-values de survivalGPU :

- ▶ Scalable
- ▶ Rapide
- ▶ Gestion des bootstraps

```
head(WCE::drugdata)
```

	Id	Event	Start	Stop	sex	age	dose
1	1	0	0	1	1	40	0
2	1	0	1	2	1	40	0
3	1	0	2	3	1	40	0
4	1	0	3	4	1	40	0
5	1	0	4	5	1	40	0
6	1	0	5	6	1	40	0

survival et survivalGPU

```
library(survival)
coxph_model <- coxph(Surv(Start, Stop, Event) ~ dose,
                     data = WCE::drugdata)
summary(coxph_model)

Call:
coxph(formula = Surv(Start, Stop, Event) ~ dose, data = WCE::drugdata)

n= 77038, number of events= 383

              coef exp(coef) se(coef)      z Pr(>|z|)
dose 0.21956    1.24553   0.04729 4.643 3.44e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

              exp(coef) exp(-coef) lower .95 upper .95
dose           1.246      0.8029    1.135    1.366

Concordance= 0.556 (se = 0.015 )
Likelihood ratio test= 20.54  on 1 df,   p=6e-06
Wald test              = 21.55  on 1 df,   p=3e-06
Score (logrank) test = 21.89  on 1 df,   p=3e-06
```

```
library(survivalGPU)
coxphGPU_model <- coxphGPU(Surv(Start, Stop, Event) ~ dose,
                           data = WCE::drugdata)
summary(coxphGPU_model)

Call:
coxphGPU.default(formula = Surv(Start, Stop, Event) ~ dose, data = WCE::drugdata)

n= 77038, number of events= 383

              coef exp(coef) se(coef)      z Pr(>|z|)
dose 0.21956    1.24553   0.04729 4.643 3.44e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

              exp(coef) exp(-coef) lower .95 upper .95
dose           1.246      0.8029    1.135    1.366

Concordance= 0.556 (se = 0.015 )
Rsquare= 0 (max possible= 0.049 )
Likelihood ratio test= 20.54  on 1 df,   p=6e-06
Wald test              = 21.55  on 1 df,   p=3e-06
Score (logrank) test = 21.89  on 1 df,   p=3e-06
```

Benchmark

Discussion

- ▶ Implémentation de nouvelles options tels que les tests de permutation
- ▶ Validation du package à travers des simulations

Package disponible sur Github : <https://github.com/jeanfeydy/survivalGPU>