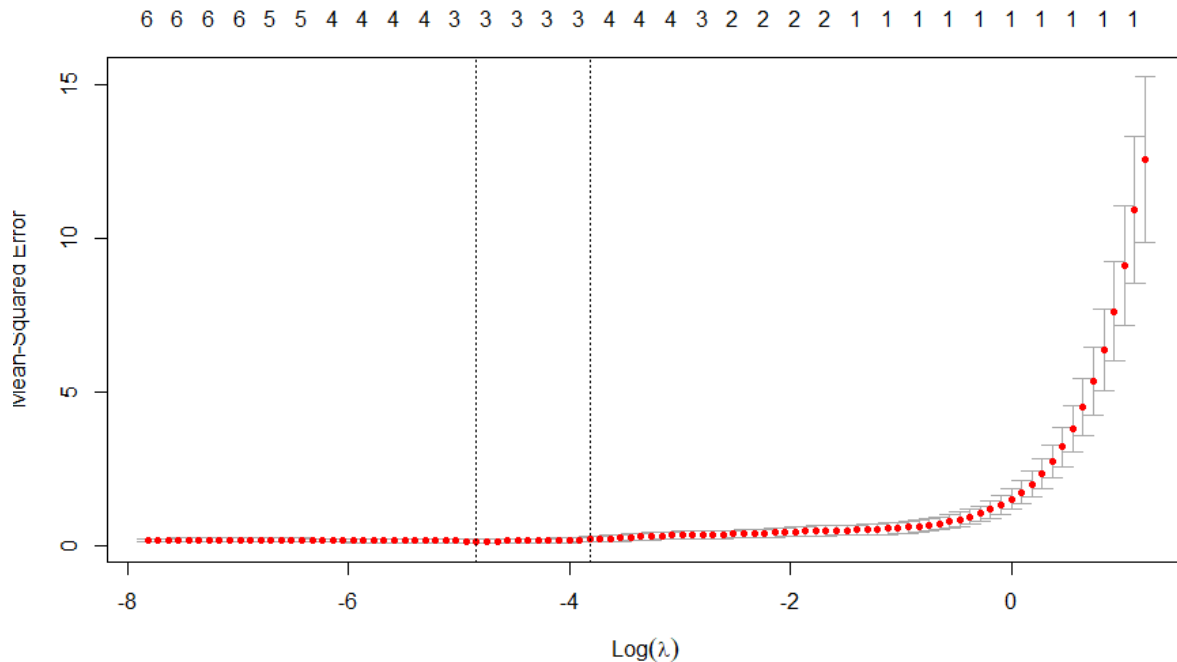


2 - régressions pénalisées

On définit Employed comme variable dépendante et GNP.deflator, GNP, Unemployed, Armed.Forces, Populationet Year comme explicatives.

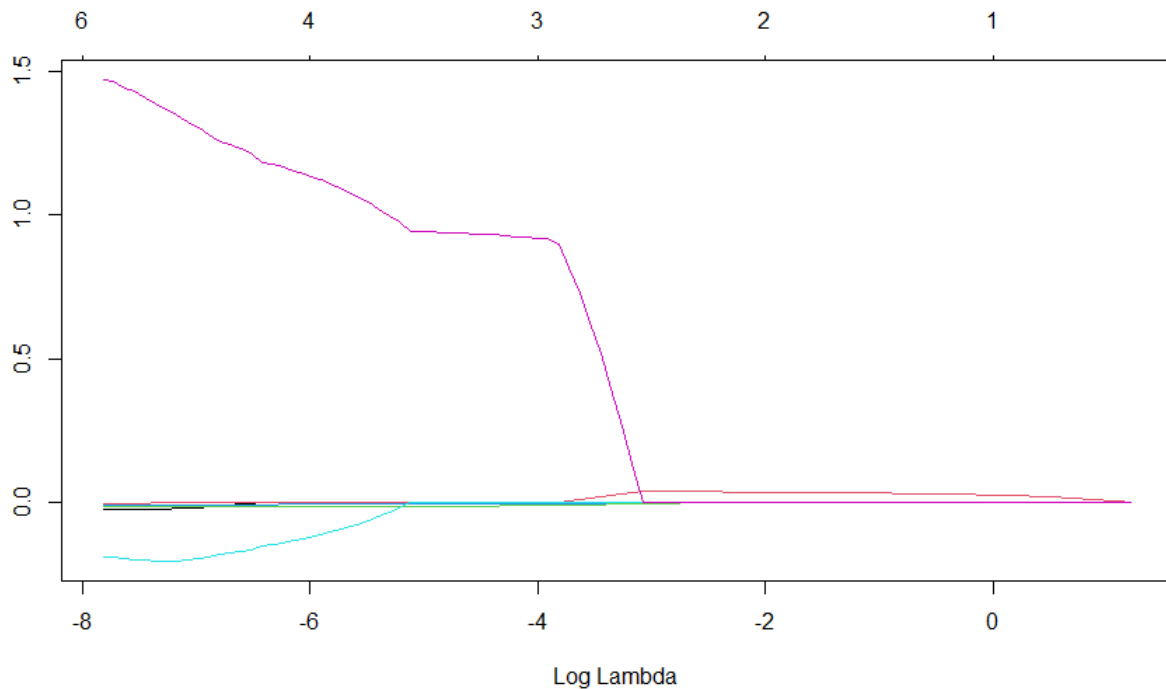


On visualise les erreurs quadratiques calculées par validation croisée en fonction de lambda. Deux traits verticaux sont représentés :

- celui de gauche correspond à la valeur de λ qui minimise l'erreur quadratique
- celui de droite correspond à la plus grande valeur de λ telle que l'erreur ne dépasse pas l'erreur minimale + 1 écart-type estimé de cette erreur.

D'un point de vu pratique, cela signifie que l'utilisateur peut choisir n'importe quelle valeur de lambda entre les deux traits verticaux.

on peut ainsi sortir le lambda minimal qui est 0.007205543. Cette valeur varie à chaque exécution du code en raison de la nature aléatoire de la validation croisée. C'est pour cela que l'on rajoute `set.seed(123)` pour avoir des résultats reproductibles



Le graphique montre comment les coefficients du modèle évoluent en fonction de la valeur de lambda. Lorsque lambda augmente, la pénalité pour les coefficients non nuls augmente également, ce qui entraîne une diminution du nombre de coefficients non nuls dans le modèle.

```

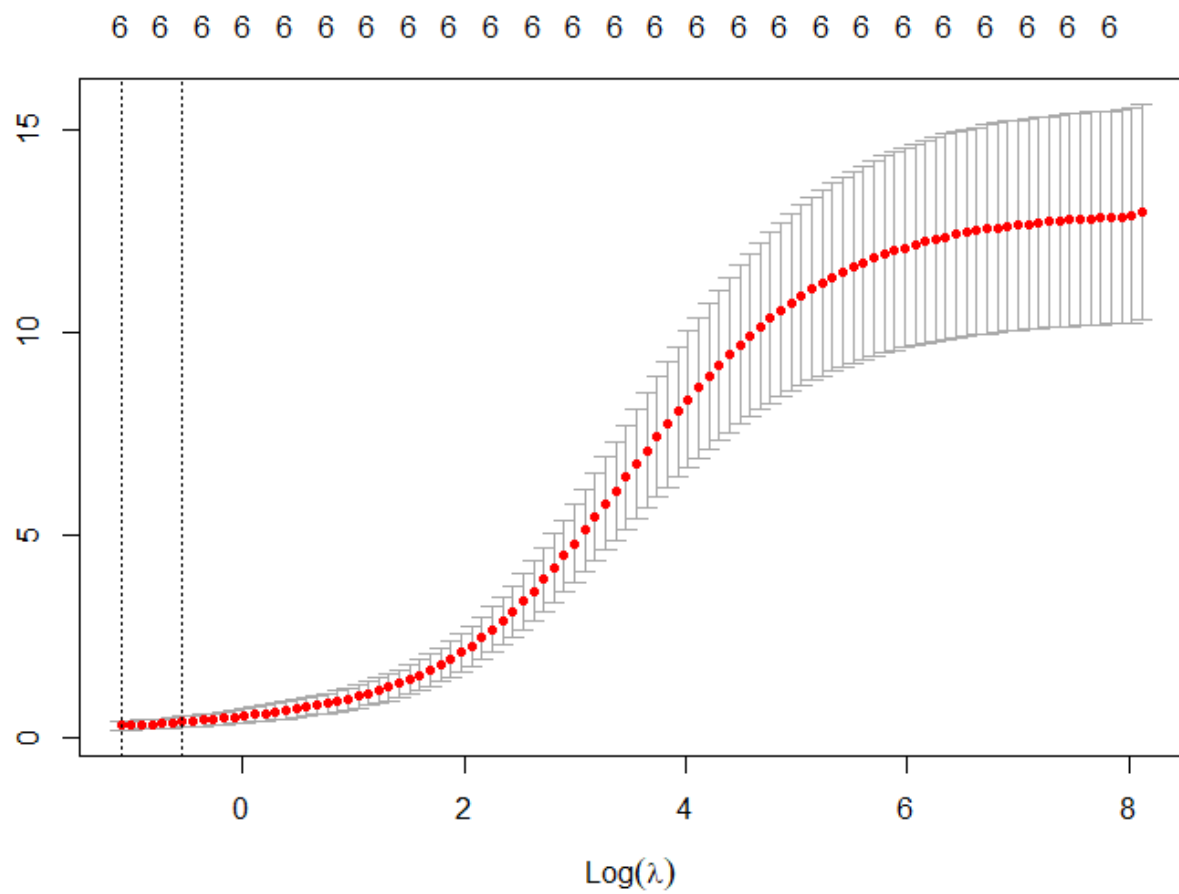
                    50
(Intercept) -1.769070e+03
GNP.deflator .
GNP .
Unemployed -1.402264e-02
Armed.Forces -7.034056e-03
Population .
Year 9.417747e-01

```

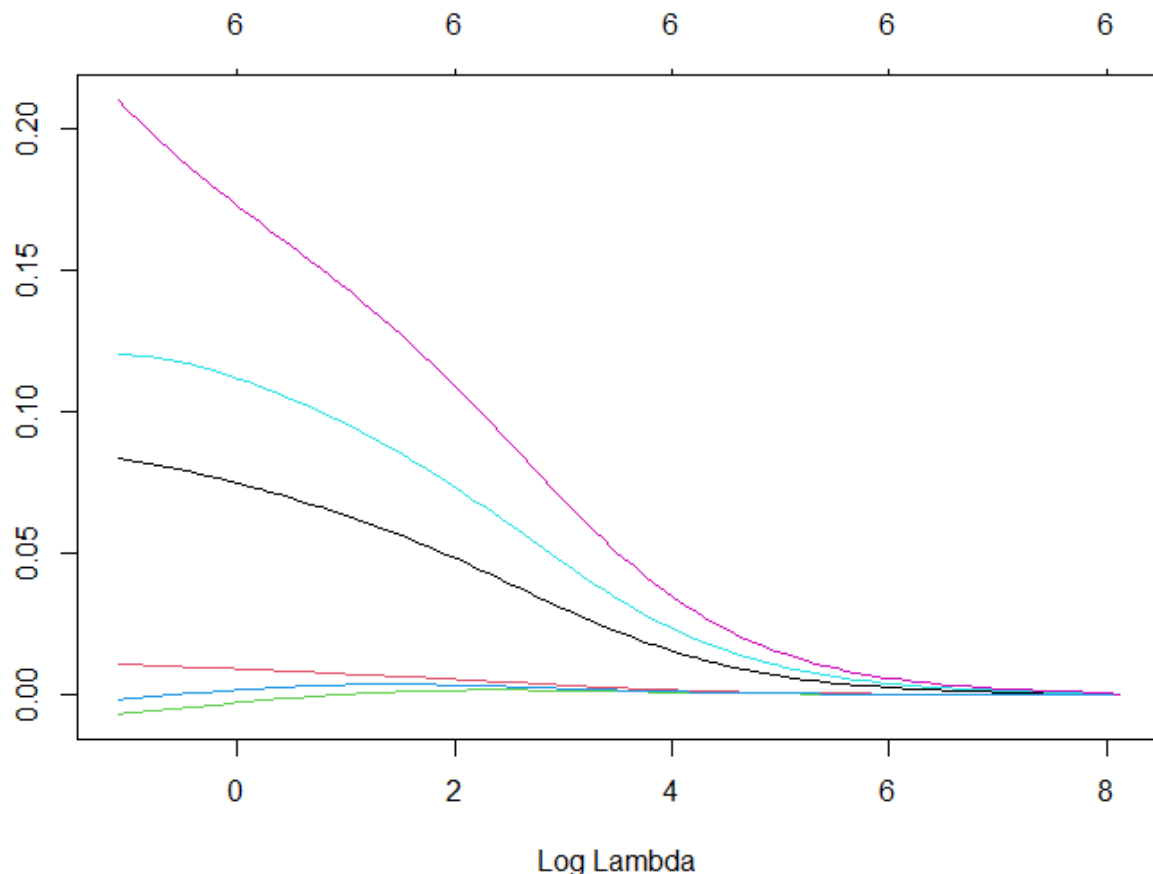
Au vu des résultats, les variables GNP.deflator, GNP et Population ont été exclues du modèle. Les autres coefficients représentent l'effet de chaque variable sur la variable de réponse Y: L'ordonnée à l'origine est de -1.769070e+03 et le coefficient pour la variable Unemployed est de -1.402264e-02, ce qui signifie que pour chaque augmentation d'une unité de la variable Unemployed, la variable réponse y diminue en moyenne de 0.01402264 unités.

On trouve dans notre cas un $R^2 = 0.9927339$ ceci est très proche de 1 signifiant que le modèle explique bien les données.

On fait pareil pour RIDGE ($\alpha = 0$)



on trouve un Lambda optimal de 0.3344517



l'allure des courbes est typique d'une régression Ridge puisque les coefficients ne sont jamais nuls mais convergent vers 0.

```

                                s0
(Intercept) -3.709052e+02
GNP.deflator  8.428486e-02
GNP          1.073490e-02
Unemployed   -6.866973e-03
Armed.Forces -1.670780e-03
Population   1.189529e-01
Year         2.108726e-01

```

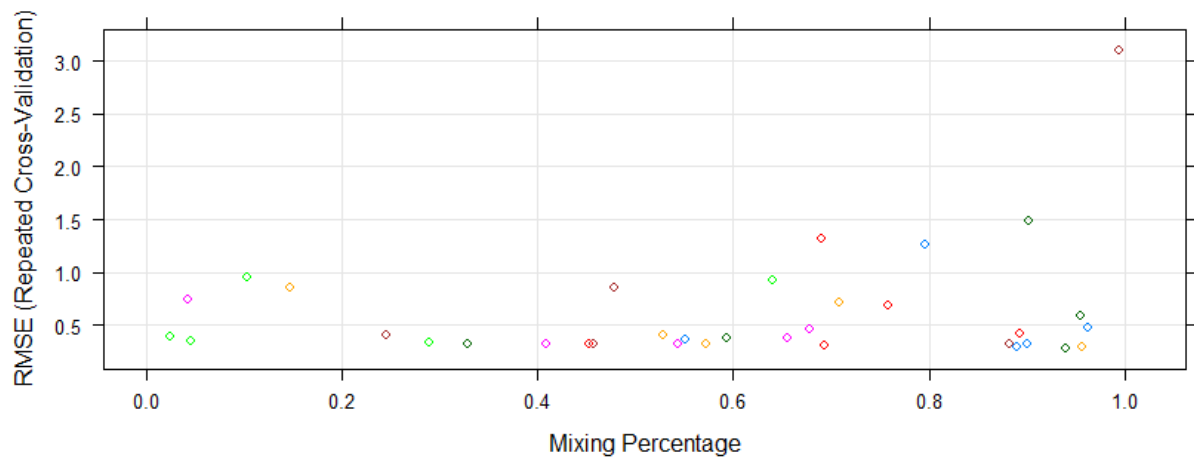
On a donc aucune variables excluent. Il est intéressant de noter que la différence de pénalisation entre RIDGE et LASSO montre des variables avec des plus fort coef dans RIDGE alors qu'il sont nul dans LASSO ex: Population

On obtient un R^2 de 0.9798747 la variable est bien expliqué mais légèrement moins bien que le modèle LASSO.

ElasticNet

ElasticNet est une méthode de régression régularisée qui combine les pénalités LASSO et Ridge pour ajuster des modèles linéaires en présence de colinéarité entre les variables explicatives

Le code va explorer plusieurs combinaison de alpha et Lambda afin de trouver celle optimal pour notre modèle.



Le graphique montre comment les performances du modèle (mesurées par l'erreur de validation croisée) varient en fonction des valeurs des hyperparamètres du modèle.

La meilleure combinaison retenue ici est : `alpha` 0.9404673 `lambda` 0.007873033

on obtient à la fin un $R^2 = 0.9927284$

Les résultats sont très satisfaisants mais on peut se questionner sur le fait qu'il soit inférieur au LASSO ce qui ne devrait pas être le cas.