Ecole Polytechnique Universitaire de Lille
Département de Génie Statistique et Informatique
Université des Sciences et Technologies de Lille

TP1 Régression linéaire (GIS3)

Preda/Loingeville

Note : Il est important de lire le cours avant le TP

Simulation. On s'intéresse à l'estimation du modèle linéaire

$$Y = \beta_0 + \beta_1 X + \varepsilon,$$

dont l'intérêt est de pouvoir expliquer la variable Y en fonction de X. Si l'on considère que ε est l'erreur de l'ajustement de Y par $\hat{Y} = \beta_0 + \beta_1 X$, telle que $\mathbb{E}(\varepsilon|X = x) = 0$ et $\mathbb{V}ar(\varepsilon|X = x) = \sigma_\varepsilon^2$, dans ce cas on a

$$\mathbb{E}(Y|X = x) = \beta_0 + \beta_1 x.$$

C'est précisément le modèle de régression linéaire.

On va se placer dans le cas où $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$ et indépendante de X, ce qui implique que la variable

$$(Y|X = x) \sim \mathcal{N}(\beta_0 + \beta_1 x, \sigma_\varepsilon^2).$$

Si X est telle que $\mathbb{V}ar(X) = \sigma_X^2$, alors on a (ANOVA de la régression) :

$$\mathbb{V}ar(Y) = \mathbb{V}ar(\hat{Y} + \varepsilon) = \beta_1^2 \sigma_X^2 + \sigma_\varepsilon^2.$$

L'objectif de cette étude de simulation est d'observer la qualité de l'estimation du modèle en fonction de la taille (variance) de l'erreur. On fixera $\sigma_X^2 = 1$ et on suppose que le vrai modèle reliant Y a X est donnée par ($\beta_0 = 1$ et $\beta_1 = 2$) :

$$Y = 1 + 2X + \varepsilon$$

1. Initialiser le générateur de nombre aléatoires (set.seed(1234)).
2. Générer un échantillon de taille $n = 100$, $\{x_i\}_{i=1,\ldots,n}$ de la variable $X \sim \mathcal{N}(3, 1)$. On stockera l'échantillon dans le vecteur `x`
3. Pour chaque valeur de $\sigma_\varepsilon^2 \in \{0.1, 0.5, 2, 6\}$ générer

   $$y_i = 1 + 2x_i + e_i,$$

   avec $\{e_i\}_{i=1,\ldots,n}$ un échantillon i.i.d de $\mathcal{N}(0, \sigma_\varepsilon^2)$. On stockera les 4 échantillons de Y dans les vecteurs `y1, y2, y3 et y4`. Tracer graphiquement les 4 nuages de points.
4. Dans chaque cas, calculer "à la main" (voir cours) les valeurs estimées de β_0 et β_1 ,σ_ε^2 puis vérifier vos calculs à l'aide de la fonction `lm` de R :

   ```
   > #...x et y1, y2, y3 et y4 contiennent les deux échantillons x_i, y_i
   > d1=data.frame(x=x, y=y1)  #creation d'un data frame
   > m1 = lm(y~x, data=d1)     # la fonction lm retourne dans m1 l'estimation du modele
   > summary(m1)
   > ....
   ```

   Tracer les droites de régression obtenues sur les graphiques précédentes (fonction `abline`).
5. Explorez les propriétés de l'objet retourné par la fonction `lm` : `str(m1)`. On regardera notamment : `coefficients, residuals, fitted.values`
6. Dans chaque cas, observer la signification des tests vérifiant l'hypothèse $H_0 : \beta_1 = 0$. Interpréter. Observer que ce test est equivalent au test $H_0 : \rho(X, Y) = 0$.
7. Calculez dans chaque cas le coefficient de correlation empirique, $r(x, y)$ (fonction `cor`). Comparez-les avec le coefficient de corrélation théorique $\rho(X, Y)$ (à vous de le calculer sur papier cette fois !). Interprétez le r^2 en termes de qualité du modèle.
8. Faisons un peu de prédiction avec notre modèle linéaire. On dispose de nouvelles données uniquement sur la variable X.

   $$X_{new} = \{2.5, 3.5, 4.5\}$$

   On souhaiterais prévoir les valeurs Y à partir du modele linéaire. Mettez-vous dans le cas ou $\sigma_\varepsilon^2 = 2$, donc le modèle construit avec `y3`.
   Faites les prédictions "à la main" puis à l'aide de la fonction `predict` :

```
> m3 = lm(y~x, data=d3)     # la fonction lm retourne dans m1 l'estimation du modele
> x_new=c(2.5,3.5,4,5)
> d_new=data.frame(x=xnew)
> predictons =predict(m3, d_new)
> print(predictions)
> ...
```