Séance 2 : Régression linéaire avec R - courbe d'étalonnage

1 Courbes d'étalonnage

Exercice 1.1 *Etalonnage pour le dosage du glucose*
Pour pouvoir déterminer la concentration du Glucose à partir de la mesure de l'absorbance, on effectue une calibration préliminaire: une courbe d'étalonnage du dosage du Glucose. Le principe du dosage est basé sur les réactions suivantes

$$D - glucose + ATP \rightarrow glucose - 6 - P + ADP$$
$$Glucose - 6 - P + NAD^+ \rightarrow gluconate - 6 - P + NADH + H^+$$

On suppose que la formation du Glucose est donc proportionnelle à celle du NADH dont on va mesurer l'apparition avec le spectrophotomètre. Il nous faut donc connaître le coefficient de proportionnalité qui permet à partir de l'absorbance de déterminer la concentration en Glucose.

Une courbe étalon est donc réalisée à partir d'une solution mère que l'on dilue et dont on mesure l'absorbance: on obtient ainsi un nuage de points ($C_{Glc,i}$,$absorbance_i$)

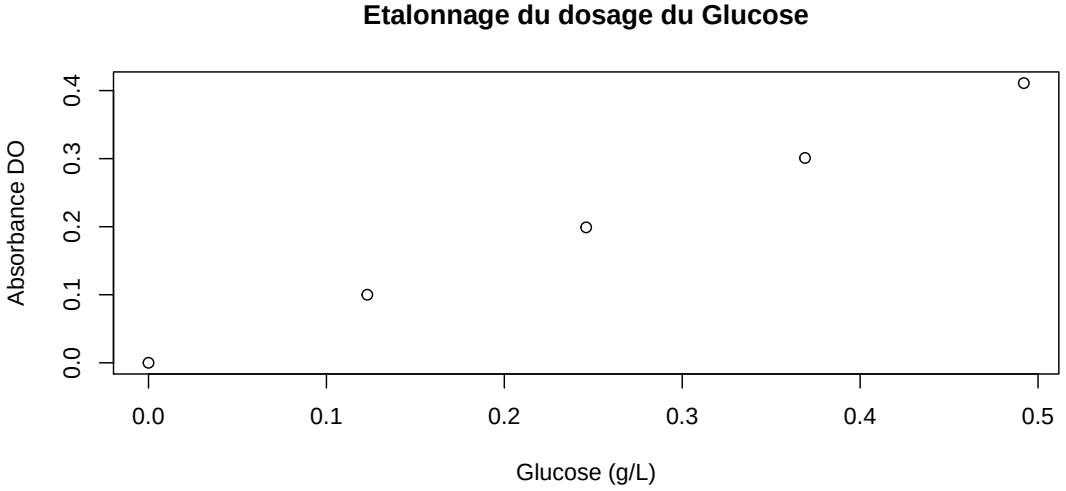
Solution	*[glucose] (g/L)*	*Absorbance*
Blanc	*0*	*0*
Dilution1/4	*0.123*	*0.1*
Dilution 1/2	*0.246*	*0.199*
Dilution 3/4	*0.369*	*0.301*
Non diluée	*0.492*	*0.411*

1. *Lire le fichier Etalon.csv à l'aide de l'instruction read.csv2. On appellera deta la data frame et on fera un attach(deta) pour simplifier le nom des variables.*

```
> deta <- read.csv2("Etalon.csv")
> attach(deta)# pour simplifier les noms des variables
```

2. *Afficher le nuage de points ($glucose_i$,$absorbance_i$).*

```
> titre="Etalonnage du dosage du Glucose"
> plot(glucose,absorbance,main=titre,xlab="Glucose (g/L)",ylab="Absorbance DO")
```



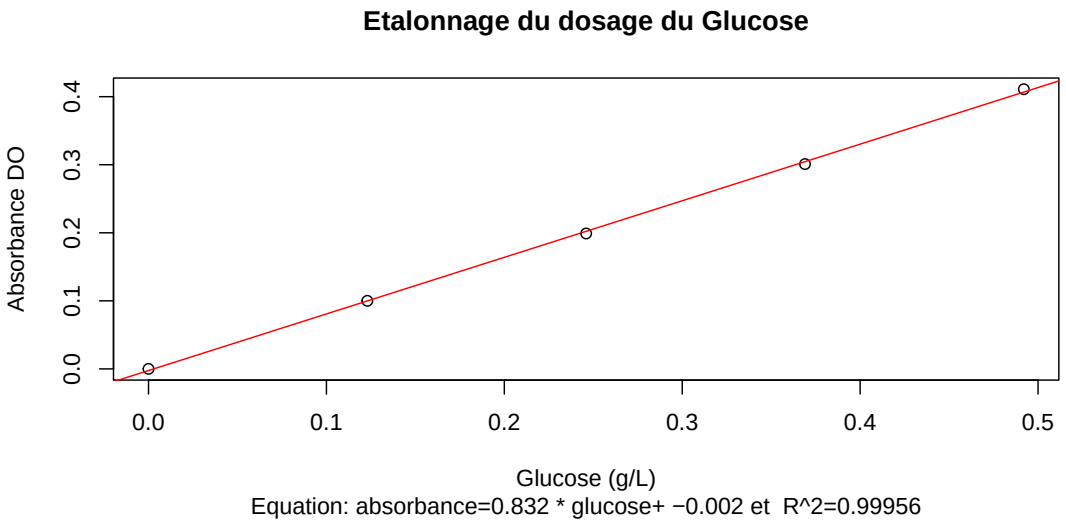
3. *Déterminer l'équation de la droite de régression entre concentration de Glucose et absorbance à l'aide de la fonction "lm". On appellera reg le résultat de lm. A quoi correspond l'instruction coef(reg) ? Donner l'équation de la droite de régression*

```
> reg<- lm(absorbance~glucose,data=deta)# détermination de la droite de régression
> coef(reg)

(Intercept)     glucose
 -0.0024000   0.8317073

> a<- coef(reg)[2]# pente de la droite de regression
> b<- coef(reg)[1]# ordonnée à l'origine
```

L'instruction coef(reg) permet d'afficher l'équation de la droite de régression qui est absorbance=0.832glucose+ -0.002 (valeurs arrondies à 10^{-3}).

4. *Afficher le nuage de points, la droite de régression sur un même graphe. Mettre à ce graphique un titre principal "Etalonnage Glucose vs absorbance" puis l'équation de la droite de régression et le coefficient de détermination R^2 en sous-titre.*

```
> R2=summary(reg)$r.squared # coefficient de détermination
> eq=paste0("Equation: absorbance=",round(a,3)," * glucose+ ",round(b,3)," et  R^2=",round(R2,5))
> plot(glucose,absorbance,main=titre,sub=eq,xlab="Glucose (g/L)",ylab="Absorbance DO")
> abline(b,a,col="red")# droite de régression
```



Le R^2 (la variance expliquée par le modèle divisée par la variance totale) étant très proche de 1, cela signifie que le pouvoir prédictif du modèle est fort.

5. *Etude de la qualité de la régression.*

   (a) *A partir de la commande summary(reg), remplir le tableau suivant*

Coefficients	*Estimate*	*Std. Error*	*t value*	*Pr(>|t|)*
(Intercept)				
x				

```
> summary(reg)

Call:
lm(formula = absorbance ~ glucose, data = deta)

Residuals:
      1       2       3       4       5
 0.0024  0.0001 -0.0032 -0.0035  0.0042

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
```

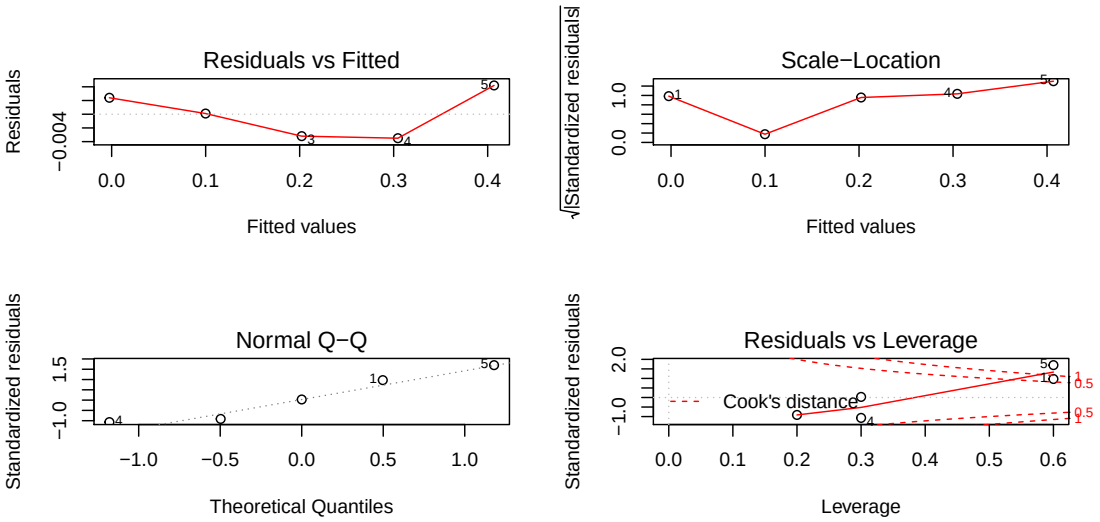
```
(Intercept) -0.00240    0.00303  -0.792    0.486
glucose      0.83171    0.01006  82.705  3.9e-06 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.003912 on 3 degrees of freedom
Multiple R-squared:  0.9996,        Adjusted R-squared:  0.9994
F-statistic:  6840 on 1 and 3 DF,  p-value: 3.896e-06
```

(b) *Quels sont les estimations des écarts types des coefficients de la droite ?*
Pour chaque paramètre, la table donne la statistique observée (colonne t value) ainsi que la pvalue (colonne Pr(>|t|)) associée au test d'hypothèse H_0 : "le paramètre est nul" contre H_1 : "le paramètre n'est pas nul". Peut-on accepter l'hypothèse "la courbe passe par l'origine" ? Si oui, tester un autre modèle plus adapté.

```
> reg2<- lm(absorbance~-1+glucose,data=deta)# détermination de la droite de régression
> summary(reg2)

Call:
lm(formula = absorbance ~ -1 + glucose, data = deta)

Residuals:
         1          2          3          4          5
-9.628e-17 -1.500e-03 -4.000e-03 -3.500e-03  5.000e-03

Coefficients:
        Estimate Std. Error t value Pr(>|t|)
glucose 0.825203   0.005529   149.2 1.21e-08 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.003725 on 4 degrees of freedom
Multiple R-squared:  0.9998,        Adjusted R-squared:  0.9998
F-statistic: 2.228e+04 on 1 and 4 DF,  p-value: 1.209e-08
```

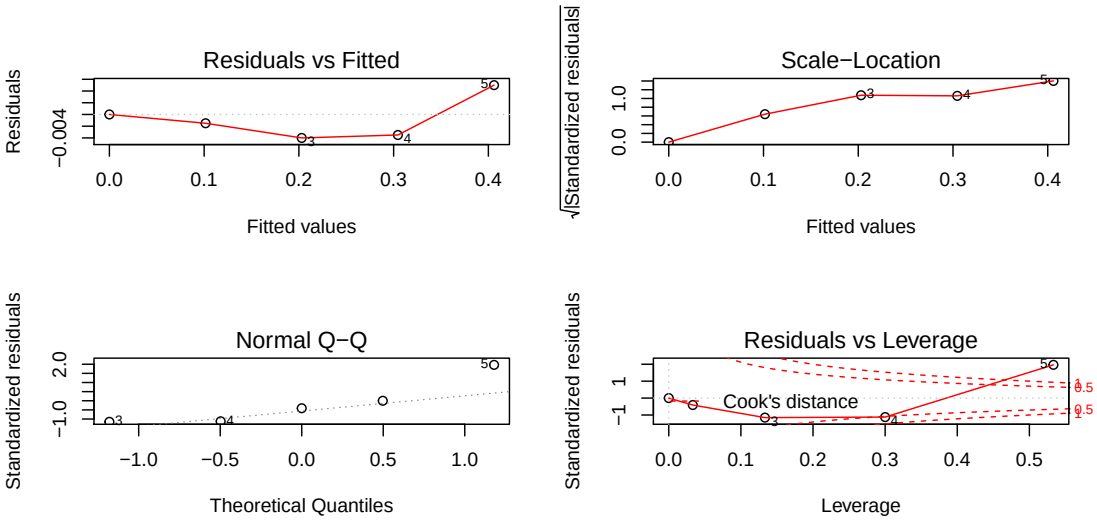
6. *Etudier les résidus*
Modèle initial

```
> layout(matrix(1:4,2,2))# fenetre graphique coupée en 4
> plot(reg) # 4 graphiques
```



Modèle: droite passant par l'origine

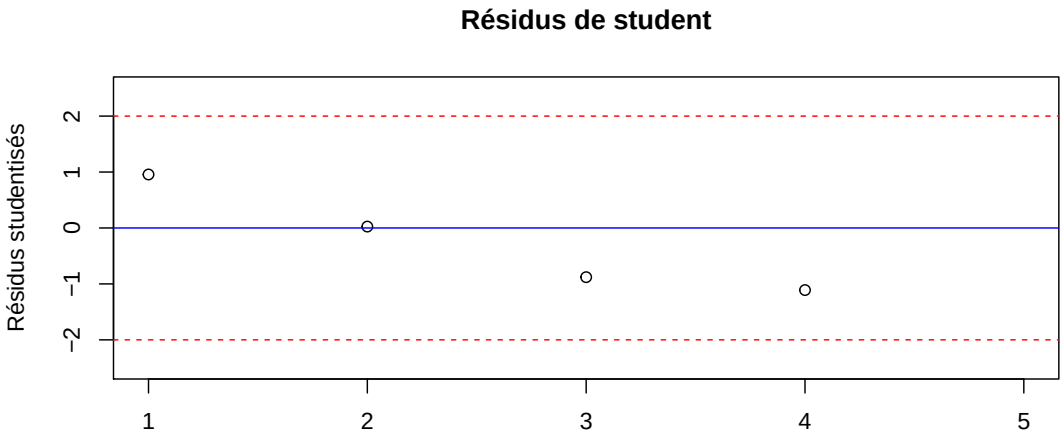
```
> layout(matrix(1:4,2,2))# fenetre graphique coupée en 4
> plot(reg2) # 4 graphiques
```



Les résidus ne semblent pas aléatoirement distribués mais il faudrait davantage de points pour pouvoir en juger. Un point ayant une distance de Cook supérieure à 1 semble particulièrement suspect.

```
> # résidus studentisés
> res <- rstudent(reg)
> # tracé des résidus studentisés (ylim échelle pour y)
> plot(res,ylab="Résidus studentisés",xlab="",main="Résidus de student",ylim=c(-2.5,2.5)) # tracés des d
> # h= contient équation y=(-2,0,2) et lty le type de lignes
> abline(h=c(-2,0,2),lty=c(2,1,2),col=c("red","blue","red"))
> print(res)
```

```
         1          2          3          4          5
0.95618289 0.02495326 -0.87943954 -1.11013259 7.00000000
```



On retrouve qu'un point est particulièrement suspect (il n'apparaît pas sur le graphique puisque des bornes sur les ordonnées ont été mises avec la commande ylim).

On retiendra qu'il aurait été judicieux de dupliquer toutes les mesures: pour une même valeur de glucose, deux ou trois valeurs d'absorbance permettraient de mieux étudier les résidus (augmentent-ils avec la valeur du glucose ?). Quel échantillonnage ?: si le caractère linéaire du phénomène étudié est certain, mieux vaut répartir les points aux extrémités du domaine de validité. Ainsi on prédit mieux la pente. Mais si on veut vérifier le caractère linéaire, il faut les répartir régulièrement sur le domaine.

7. *Prédictions*

   (a) *Prédire les valeurs suivantes et leur intervalle de prédiction*

[glucose] (g/L)	*Absorbance*	*Absorbance min*	*Absorbance max*
0.05			
0.1			
0.15			
0.2			
0.25			
0.3			
0.35			
0.4			

```
> new=seq(0.05,0.4,0.05);
> pc=predict(reg2,data.frame(glucose= new), level = 0.95, interval = "confidence")
> print(pc)# valeur prédite et intervalle de confiance
```

```
         fit        lwr        upr
1 0.04126016 0.04049261 0.042027772
2 0.08252033 0.08098521 0.08405544
3 0.12378049 0.12147782 0.12608315
4 0.16504065 0.16197043 0.16811087
5 0.20630081 0.20246303 0.21013859
6 0.24756098 0.24295564 0.25216631
7 0.28882114 0.28344825 0.29419403
8 0.33008130 0.32394086 0.33622175
```

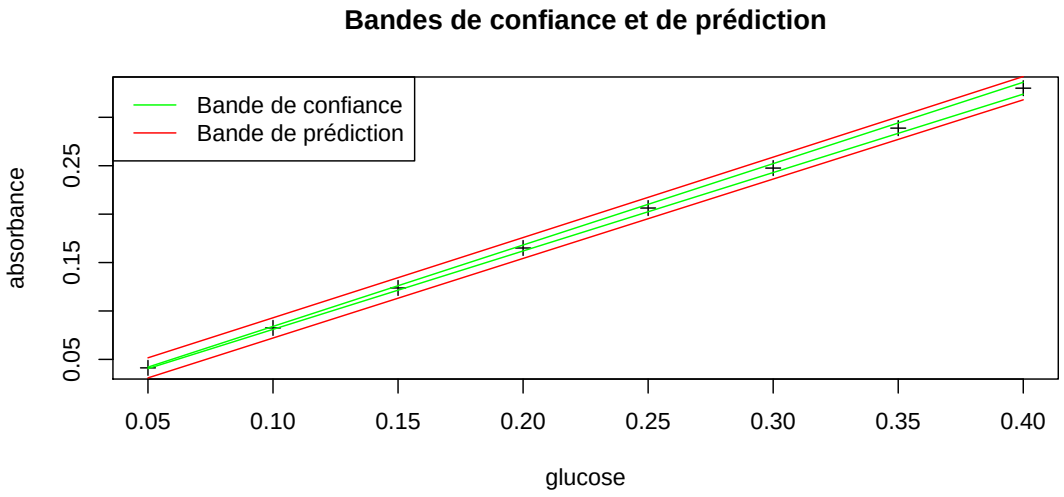
```
> pp=predict(reg2,data.frame(glucose= new), level = 0.95, interval = "prediction")
> print(pp)# valeur prédite et intervalle de prédiction
```

```
         fit        lwr        upr
1 0.04126016 0.03088969 0.05163063
2 0.08252033 0.07206499 0.09297566
3 0.12378049 0.11318522 0.13437576
4 0.16504065 0.15425252 0.17582878
5 0.20630081 0.19526967 0.21733195
6 0.24756098 0.23623991 0.25888204
7 0.28882114 0.27716672 0.30047555
8 0.33008130 0.31805373 0.34210888
```

La fonction predict renvoie une matrice de trois colonnes contenant la valeur prédite, le borne inférieure et la borne supérieure de l'intervalle demandé (confiance ou prédiction).

   (b) *Tracer la droite de régression ainsi que les intervalles de confiance et de prédiction (bornes min et max).*

```
> layout(1)
> plot( pp[,1] ~ new, type='p',pch=3,xlab="glucose",ylab="absorbance" )
> points( pc[,2] ~ new, type='l', col="green" )
> points( pc[,3] ~ new, type='l', col="green" )
> points( pp[,2] ~ new, type='l', col="red" )
> points( pp[,3] ~ new, type='l', col="red" )
> title(main="Bandes de confiance et de prédiction")
> legend("topleft", c("Bande de confiance", "Bande de prédiction"),lwd=1, lty=1, col=c("green", "red
```



8. *Calibration: courbe d'étalonnage*

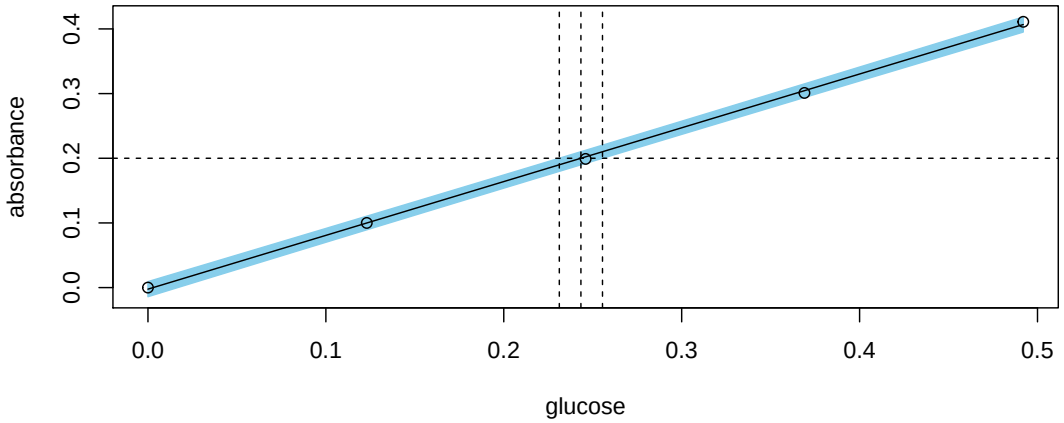
On a fait une régression linéaire glucose (en abscisse) en fonction de l'absorbance (en ordonnée) puisque c'est la concentration de glucose qui est connue (variable explicative supposée certaine), l'absorbance mesurée étant l'estimation d'une variable aléatoire. Mais ensuite, c'est la fonction réciproque qui nous intéresse: connaissant la mesure de l'absorbance, on voudra en déduire la concentration (inconnue) en glucose d'un soluté qu'on passera au spectromètre. On ne va pas donc pas utiliser la fonction predict mais la fonction calibrate du package investr. Cette fonction nous donnera à partir d'une absorbance mesurée la concentration de glucose correspondante et un intervalle de prédiction.

(a) *Charger le package investr. On a obtenu une absorbance de 0.2. Quelle est la concentration de glucose correspondante ?*

```
> library(investr)
> # absorbance de 0.20 donne concentration glc et un IC dans res
> res <- calibrate(reg, y0 =0.2, interval = "inversion", level = 0.9)
> print(res)
```

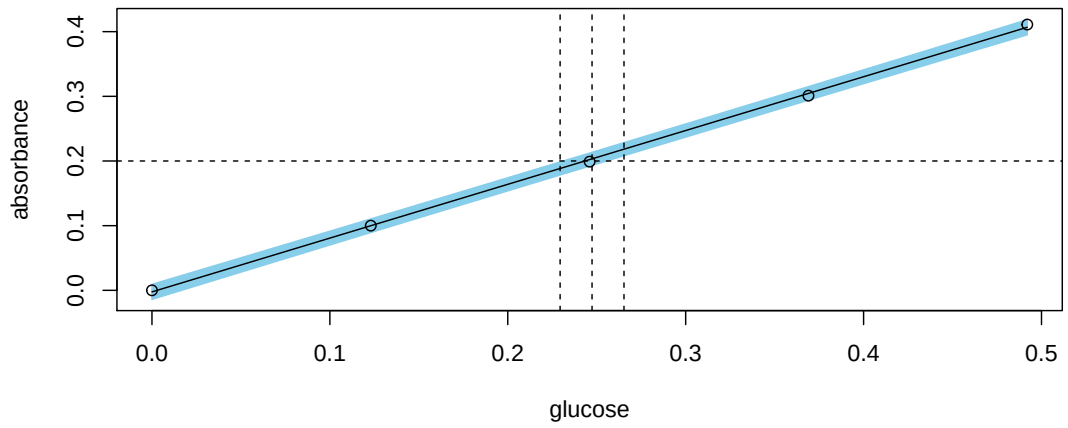
```
 estimate     lower     upper
0.2433548 0.2312233 0.2554821
```

```
> plotFit(reg, interval = "prediction", level = 0.9, shade = TRUE, col.pred = "skyblue")
> abline(h = 0.2, v = c(res$lower, res$estimate, res$upper), lty = 2)
```



(b) *On a répété trois fois la mesure et on a obtenu: 0.2; 0.22 et 0.19. Quelle est la concentration de glucose correspondante ?*

```
> # si on dispose de 3 mesures de l'absorbance
> res=calibrate(reg, y0 = c(0.2,0.22,0.19), interval = "inversion", level = 0.9)
> print(res)
 estimate     lower     upper
0.2473627 0.2294177 0.2653185
> plotFit(reg, interval = "prediction", level = 0.9, shade = TRUE, col.pred = "skyblue")
> abline(h = 0.2, v = c(res$lower, res$estimate, res$upper), lty = 2)
```



9. *Remarque : on cherche à minimiser les écarts horizontaux au carré entre la droite et les points expérimentaux (glucose, absorbance): on fait la régression sur le glucose (comme fonction de l'absorbance), on obtient des résultats similaires. La droite de régression est ici presque symétrique (ce n'est pas vrai en général).*

```
> reginv<- lm(glucose~absorbance,data=deta)# détermination de la droite de régression sur X
> d=1/coef(reginv)[2]
> c=-coef(reginv)[1]/coef(reginv)[2]

> plot(glucose,absorbance,main=titre,sub=eq,xlab="Glucose (g/L)",ylab="Absorbance DO")
> abline(b,a,col="red")# droite de régression sur y
> abline(c,d,col="green")# droite de régression sur x
```