biostat

camille mathilde

22/12/2020

Introduction

De nos jours, les maladies sont de plus en plus étudiées et de mieux en mieux comprises. De nombreux organismes et intituts recherchent des solutions pour vaincre ces maladies en trouvant des traitements. De nombreuses diciplines sont impliquées, notamment la biostatistique, qui permet d'étudier différentes situations avec des données. Dans notre cas, nous allons étudier l'occurence des maladies coronarienne du coeur. Une étude préalable à déja été faite sur une cohorte d'individus. Ces individus ont répondu à une date donnée à une enquête sur les habitudes alimentaires. Les réponses ont été recueilli dans une base de donnée que l'on nomme **Coeur**.
L'objectif de cette étude sera de connaître les bonnes habitudes alimentaires à adopter pour se prevenir d'une maladie coronarienne.

Présentation de la base de donnée

Notre table de données contient 337 individus et 15 variables qui sont

- **Id** : l'identifiant du sujet.
- **DateEntrée** et **Date de sortie** : les dates d'entrée et de sortie de l'étude.
- **Date Naissance** : la date de naissance.
- **Statut** : si la sortie de l'enquête est due à une maladie coronarienne de cœur, alors le type de maladie est indiqué (la signification du code n'est pas précisée ici). Si l'individu est sain à la sortie de l'enquête, alors le code vaut 0.
- **Emploi** : le type d'emploi.
- **MoisEnquête** : le mois (1= Janvier, 12= Décembre) où l'individu a répondu à l'enquête sur ses pratiques alimentaires.
- **Taille/Poids** : la taille et le poids de l'individu (en cm et en kg).
- **Graisse** : la quantité moyenne de graisse ingérée par jour (g/jour).
- **Fibres** : la quantité moyenne de fibres ingérée par jour (g/jour).
- **Consommation** : la quantité de calories(/100) ingérée par jour.
- **hauteConsomation** : une variable binaire, recodage de la variable consommation.
- **MCC** : une variable binaire, recodage de la variable statut (1=MCC, 0=pas de MCC).

Nous avons également rajouté la variable **IMC** en divisant le poids par la taille au carrée pour faire un lien entre la condition physique et la maladie du coeur car le poids ou la taille tout seul ne suffisent pas pour savoir si une personne est en bonne santé ou en surpoids.

Nous nous sommes également rendu compte qu'il y avait des erreurs dans le recodage de la variables **statut**. En effet, certains individus avaient contracté une maladie du coeur, mais la variable recodage **MCC** ne l'avait pas pris en compte nous avons donc rectifié ça. Nous avons aussi remarqué que certaines variables qualitatives étaient en **numeric**, ce qui posera problème pour notre étude. Nous les recodons donc en **factor**. Nous observons enfin que la table de données contient des valeurs manquantes. Nous enlevons donc chaque ligne qui contient au moins une valeur manquante. Nous passons donc de 337 à 328 individus.

Regardons les 5 premières lignes de notre table de données :

```
coeur <- readRDS('data/my_data_frame.rds')
coeur <-coeur%>%drop_na()
coeur <-coeur%>%dplyr::select(-X1)
```

```r
coeur$statut<-as.factor(coeur$statut)
coeur$emploi<-as.factor(coeur$emploi)
coeur$moisEnqu_e<-as.factor(coeur$moisEnqu_e)
coeur$hauteConsomation<-as.factor(coeur$hauteConsomation)

coeur<-mutate(coeur,imc =poids /(taille/100)^2)
coeur<-coeur %>% dplyr::select(-MCC)
coeur <-mutate(coeur,MCC =case_when(
  statut!=0~1,
  TRUE~0))

pander(head(coeur))
```

Table 1: Table continues below

id	dateEntree	dateSortie	dateNaissance	statut	emploi
102	17/01/76	02/12/86	02/03/39	0	Driver
59	16/07/73	05/07/82	05/07/12	0	Driver
126	17/03/70	20/03/84	24/12/19	13	Conductor
16	16/05/69	31/12/69	17/09/06	3	Driver
247	16/03/68	25/06/79	10/07/18	13	Bank worker
272	16/03/69	13/12/73	06/03/20	3	Bank worker

Table 2: Table continues below

moisEnqu_e	consommation	taille	poids	graisse	fibre
1	22.86	181.6	88.18	9.168	1.4
7	23.88	166	58.74	9.651	0.935
3	24.95	152.4	49.9	11.25	1.248
5	22.24	171.2	89.4	7.578	1.557
3	18.54	177.8	97.07	9.147	0.991
3	20.31	175.3	61.01	8.536	0.765

hauteConsomation	imc	MCC
<=2750 KCals	26.74	0
<=2750 KCals	21.32	0
<=2750 KCals	21.48	1
<=2750 KCals	30.51	1
<=2750 KCals	30.71	1
<=2750 KCals	19.86	1

Remarquons que cet ensemble d'individu est bien une cohorte car nous avons relevé certaines covariables et les trois données fondamentales qui sont, la date d'entrée dans l'étude, la date de sortie dans l'étude et le cause de sortie dans l'étude. Nous pouvons ajouter que les covariables utilisées dans l'étude sont fixe.

Statistique descriptive

Notre jeu de donnée présente 5 variables quantitatives et 9 variables qualitatives.

Faisons un sommaire des variables quantitatives :

consommation	fibre	graisse	taille	poids
Min. :17.48	Min. :0.605	Min. : 7.26	Min. :152.4	Min. : 46.72
1st Qu.:25.46	1st Qu.:1.367	1st Qu.:11.15	1st Qu.:168.9	1st Qu.: 64.64
Median :28.11	Median :1.679	Median :12.60	Median :173.0	Median : 72.80
Mean :28.35	Mean :1.723	Mean :12.76	Mean :173.4	Mean : 72.40
3rd Qu.:31.10	3rd Qu.:1.939	3rd Qu.:14.02	3rd Qu.:177.8	3rd Qu.: 79.44
Max. :43.96	Max. :5.351	Max. :21.63	Max. :190.5	Max. :106.14

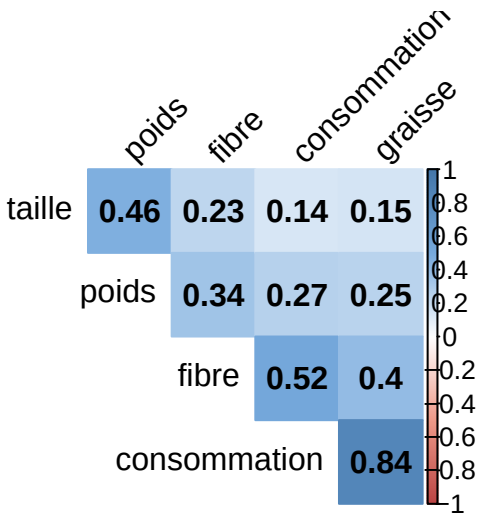
Les individus mangent en moyenne 2835 calories par jours. Ils ingèrent de plus 12.76 grammes de gras par jour, mesurent 173 cm et pèsent 72.40 kilo-gramme en moyenne.

Faisons maintenant un sommaire des variables quantitatives :

statut	emploi	moisEnqu_e	hauteConsomation
0 :252	Bank worker:147	11 : 39	<=2750 KCals:149
3 : 18	Conductor : 83	1 : 37	>2750 KCals :179
13 : 18	Driver : 98	3 : 37	NA
12 : 12	NA	2 : 34	NA
5 : 10	NA	5 : 34	NA
1 : 8	NA	12 : 33	NA
(Other): 10	NA	(Other):114	NA

Il y a 2 fois plus de **Bank worker** que de **Conductor** ou de **Driver**. Il y a 149 personnes qui mangent moins de 2750 calories par jours et 179 personnes qui en mangent plus.

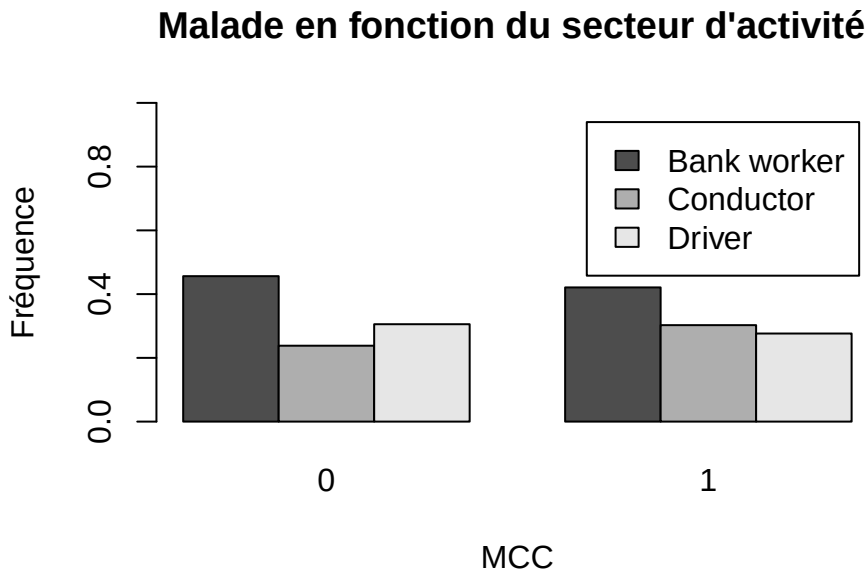
Regardons désormais la corrélation entre les variables quantitatives :



Nous observons que toutes les variables sont corrélées positivement.

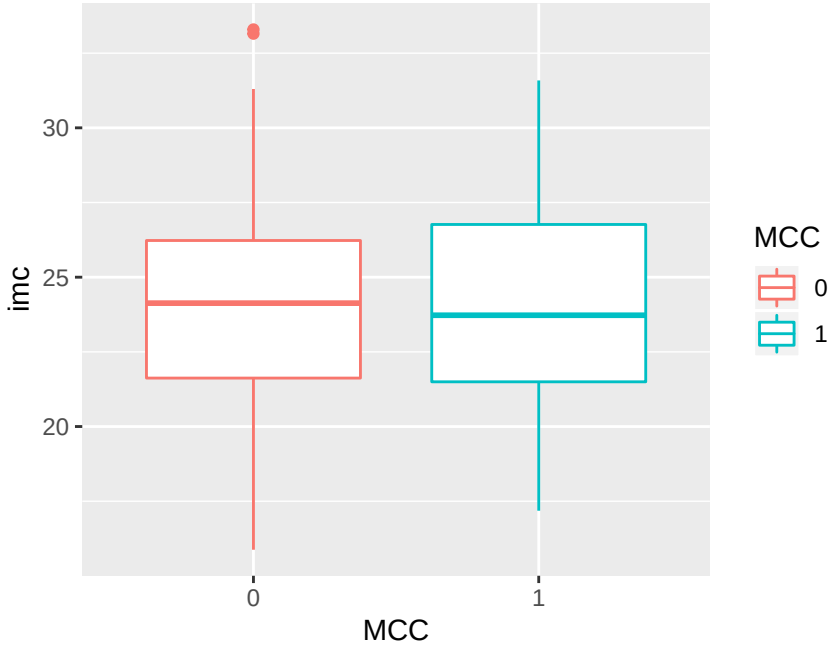
La variable **consommation** est très corrélée avec la variable **graisse** mais elle est très peu corrélée avec la variable **taille**. La variable **taille** est peu corrélée avec la variable **graisse**.

Nous allons maintenant regarder le lien entre la maladie et le secteur d'activité.



Nous observons que la proportion de conducteur est plus élevée chez les malades que chez les non malades.

Nous allons maintenant regarder le lien entre l'IMC et les maladies coronariennes.



Nous remarquons que la mediane de l'IMC du groupe de personnes malades est à peine plus basse que la médiane pour le groupe de personnes non malade. Une simple analyse descriptive ne suffit pas pour obtenir des résultats bien concluants, nous allons continuer avec des méthodes plus poussées. Nous commencerons par des régréssions logistiques.

MCC expliqué par le poids et l'IMC

Nous allons commencer par une régréssion logistique qui nous permettera d'expliquer la variable **MCC** en fonction de certaines covariables. Rappelons que dans la régréssion logistique ce n'est pas la réponse binaire qui est modélisé mais la probabilité de réalisation d'une des deux modalités (avoir une maladie coronarienne ou non).

Nous allons commencer par regarder le meilleur modèle en comparant les AIC puis nous étudierons ce modèle.

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
coeurlogbin<-coeur%>%dplyr::select(-id,-dateEntree,-dateSortie,-dateNaissance,-statut)

res<-glm(MCC~.,family = binomial(logit),data=coeurlogbin)
stepAIC(res)
```

```
## Start:  AIC=379.62
## MCC ~ emploi + moisEnqu_e + consommation + taille + poids + graisse +
##     fibre + hauteConsomation + imc
##
##                     Df Deviance    AIC
## - moisEnqu_e        11   342.66 362.66
## - emploi             2   338.12 376.12
## - consommation       1   337.62 377.62
## - taille             1   337.66 377.66
## - hauteConsomation   1   337.80 377.80
## - imc                1   337.94 377.94
## - poids              1   337.95 377.95
## - graisse            1   338.06 378.06
## - fibre              1   338.59 378.59
## <none>                   337.62 379.62
##
## Step:  AIC=362.66
## MCC ~ emploi + consommation + taille + poids + graisse + fibre +
##     hauteConsomation + imc
##
##                     Df Deviance    AIC
## - emploi             2   343.57 359.57
## - taille             1   342.66 360.66
## - consommation       1   342.69 360.69
## - poids              1   342.81 360.81
## - imc                1   342.82 360.82
```

```
## - graisse           1   342.97 360.97
## - hauteConsomation  1   342.97 360.97
## - fibre             1   343.69 361.69
## <none>                  342.66 362.66
##
## Step:  AIC=359.57
## MCC ~ consommation + taille + poids + graisse + fibre + hauteConsomation +
##     imc
##
##                    Df Deviance    AIC
## - taille            1   343.58 357.58
## - consommation      1   343.65 357.65
## - poids             1   343.75 357.75
## - imc               1   343.76 357.76
## - graisse           1   343.81 357.81
## - hauteConsomation  1   343.90 357.90
## - fibre             1   344.38 358.38
## <none>                  343.57 359.57
##
## Step:  AIC=357.58
## MCC ~ consommation + poids + graisse + fibre + hauteConsomation +
##     imc
##
##                    Df Deviance    AIC
## - consommation      1   343.66 355.66
## - graisse           1   343.82 355.82
## - hauteConsomation  1   343.91 355.91
## - fibre             1   344.39 356.39
## <none>                  343.58 357.58
## - imc               1   348.82 360.82
## - poids             1   349.72 361.72
##
## Step:  AIC=355.66
## MCC ~ poids + graisse + fibre + hauteConsomation + imc
##
##                    Df Deviance    AIC
## - hauteConsomation  1   343.91 353.91
## - graisse           1   344.51 354.51
## - fibre             1   344.72 354.72
## <none>                  343.66 355.66
## - imc               1   348.88 358.88
## - poids             1   349.77 359.77
##
## Step:  AIC=353.91
## MCC ~ poids + graisse + fibre + imc
##
##           Df Deviance    AIC
## - graisse  1   344.52 352.52
## - fibre    1   344.79 352.79
## <none>         343.91 353.91
## - imc      1   349.27 357.27
## - poids    1   350.12 358.12
##
## Step:  AIC=352.52
```

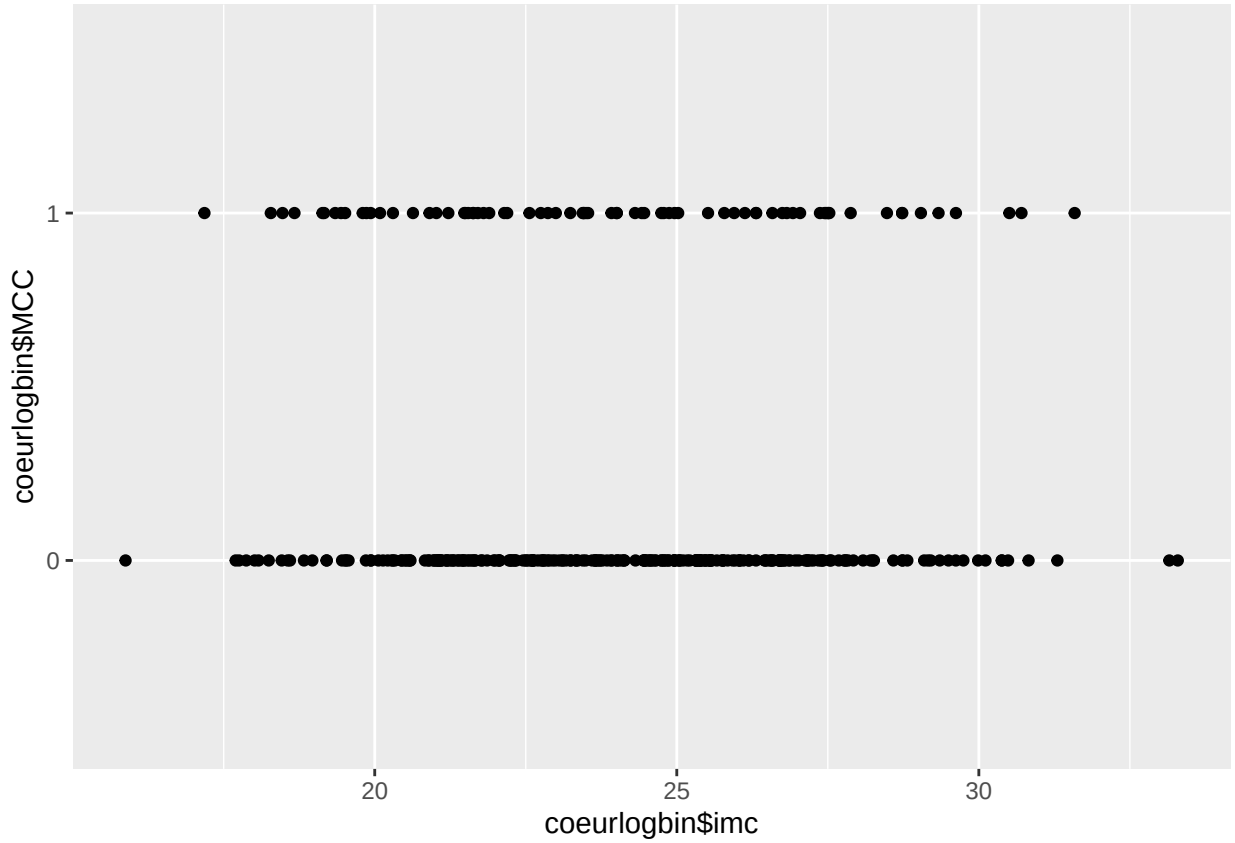
```
## MCC ~ poids + fibre + imc
##
##         Df Deviance    AIC
## - fibre  1   346.17 352.17
## <none>       344.52 352.52
## - imc    1   349.90 355.90
## - poids  1   351.05 357.05
##
## Step:  AIC=352.17
## MCC ~ poids + imc
##
##         Df Deviance    AIC
## <none>       346.17 352.17
## - imc    1   351.99 355.99
## - poids  1   355.01 359.01
##
## Call:  glm(formula = MCC ~ poids + imc, family = binomial(logit), data = coeurlogbin)
##
## Coefficients:
## (Intercept)        poids          imc
##    -0.68843     -0.07396      0.19954
##
## Degrees of Freedom: 327 Total (i.e. Null);  325 Residual
## Null Deviance:       355.1
## Residual Deviance: 346.2     AIC: 352.2
```

Le modèle ayant le plus faible AIC et que nous allons donc etudier est le suivant :

$$y = \mu + \beta_1 poids + \beta_2 imc$$

Regardons tout d'abord les relations entre les maladies coronariennes et l'imc.

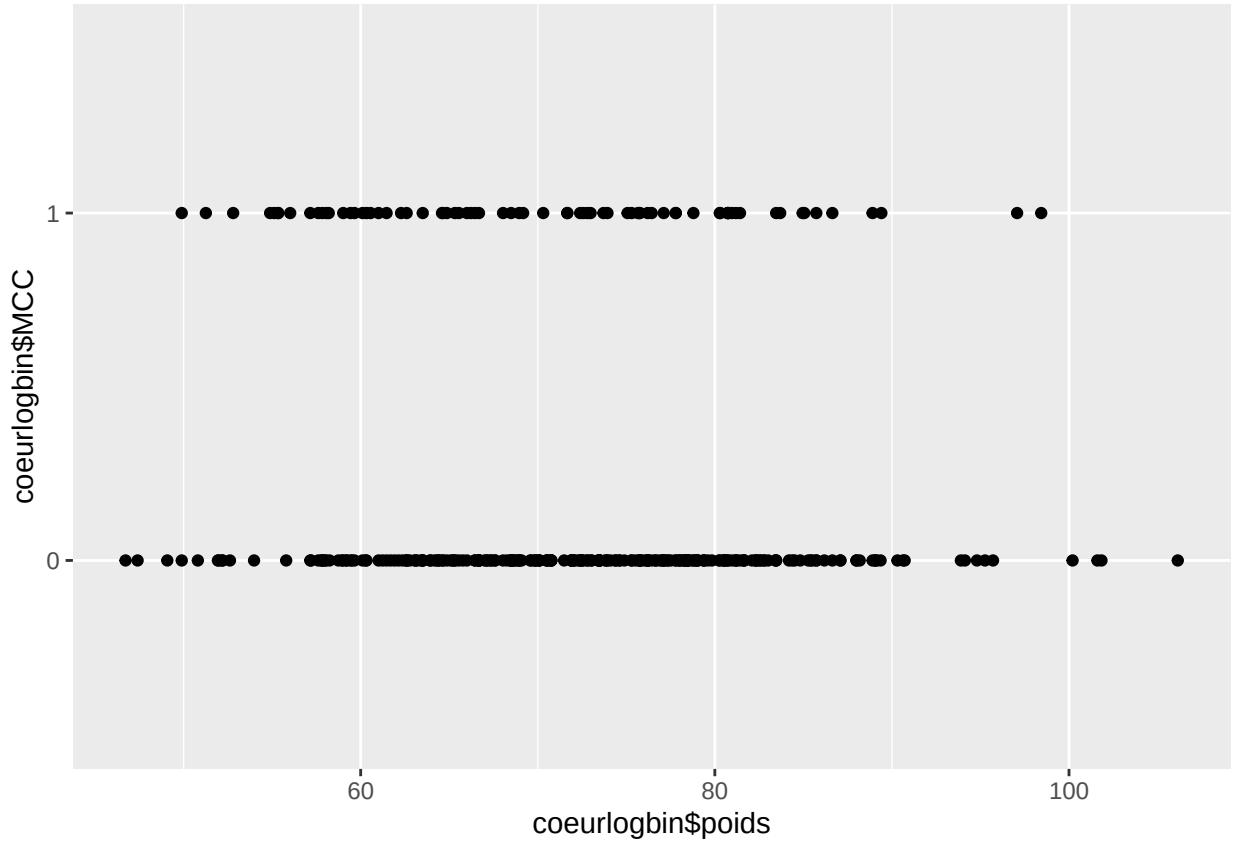
```
ggplot(coeurlogbin,aes(x=coeurlogbin$imc,y=coeurlogbin$MCC))+geom_point()
```



Nous osbervons qu

Regardons les relations entre les maladies coronariennes et le poids.

```
ggplot(coeurlogbin,aes(x=coeurlogbin$poids,y=coeurlogbin$MCC))+geom_point()
```



Faisons une régression logisitique :

```
reslog<-glm(MCC~imc+poids,family = binomial(logit),data=coeurlogbin)
summary(reslog)
```

```
##
## Call:
## glm(formula = MCC ~ imc + poids, family = binomial(logit), data = coeurlogbin)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0621  -0.7686  -0.6527  -0.4912   2.0452
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.68843    1.00829  -0.683  0.49475
## imc          0.19954    0.08317   2.399  0.01643 *
## poids       -0.07396    0.02540  -2.912  0.00359 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 355.11  on 327  degrees of freedom
## Residual deviance: 346.17  on 325  degrees of freedom
## AIC: 352.17
```

```
##
## Number of Fisher Scoring iterations: 4
```

Nous vérifions tout d'abord les conditions d'application de la régréssion logistique. Il est recommandé d'avoir en pratique 10 fois plus d'évenements que de paramètres dans le modèle. Nous utilisons ici 3 paramétres nous devrions donc avoir au moins 30 malades.

```
table(coeurlogbin$MCC)
```

```
##
##   0   1
## 252  76
```

Nous avons 76 malades nous pouvons donc continuer.

Il faut maintenant vérifier que nous ne sommes pas dans le cas de surdipersion c'est à dire qu'il ne faut pas que la dispersion réelle des données soit supérieure à celle prévue par la théorie car dans ce cas l'erreur standard des paramètres est sous-estimée ce qui peut conduire à des p-valeurs très faible et donner des conclusions erronnées. Evaluons donc s'il y a ou non une surdispersion :

$$\frac{devianceresiduelle}{nddl} = \frac{346.17}{325} = 1.06$$

nous pouvons ainsi considérer qu'il n'y a pas surdispersion.

Nous pouvons maintenant passer à l'interprétation des résultats de la régréssion logistique.

```
reslog<-glm(MCC~imc+poids,family = binomial(logit),data=coeurlogbin)
summary(reslog)
```

```
##
## Call:
## glm(formula = MCC ~ imc + poids, family = binomial(logit), data = coeurlogbin)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0621  -0.7686  -0.6527  -0.4912   2.0452
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.68843    1.00829  -0.683  0.49475
## imc          0.19954    0.08317   2.399  0.01643 *
## poids       -0.07396    0.02540  -2.912  0.00359 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 355.11  on 327  degrees of freedom
## Residual deviance: 346.17  on 325  degrees of freedom
## AIC: 352.17
##
## Number of Fisher Scoring iterations: 4
```

Notons tout d'abord que les p-values de imc et poids sont inférieurs au seuil de 5%, les effets des variables explicatives sont donc significatifs au seuil de 5%. L'intercept n'est pas significatif nous le retirons donc du

modèle.

```
reslog<-glm(MCC~imc+poids-1,family = binomial(logit),data=coeurlogbin)
reslog
```

```
##
## Call:  glm(formula = MCC ~ imc + poids - 1, family = binomial(logit),
##     data = coeurlogbin)
##
## Coefficients:
##      imc     poids
##  0.17541  -0.07536
##
## Degrees of Freedom: 328 Total (i.e. Null);  326 Residual
## Null Deviance:       454.7
## Residual Deviance: 346.6     AIC: 350.6
```

```
summary(reslog)
```

```
##
## Call:
## glm(formula = MCC ~ imc + poids - 1, family = binomial(logit),
##     data = coeurlogbin)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0618  -0.7744  -0.6543  -0.4683   2.0785
##
## Coefficients:
##       Estimate Std. Error z value Pr(>|z|)
## imc    0.17541    0.07544   2.325  0.02006 *
## poids -0.07536    0.02541  -2.966  0.00302 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 454.70  on 328  degrees of freedom
## Residual deviance: 346.64  on 326  degrees of freedom
## AIC: 350.64
##
## Number of Fisher Scoring iterations: 4
```

Nous observons que les p-valeurs sont toutes inférieures au seuil de 5%. De plus $\hat{\beta}_1 = 0.17541$ et $\hat{\beta}_2 = -0.07536$.

Nous ajoutons de plus les intervalles de confiances qui sont primordiaux pour effectuer une analyse et une interprétation. Nous pouvons faire des intervalles de confiance car nous disposons d'un assez grand jeu de donnés contenant 328 observations.

```
confint(reslog)
```

```
## Waiting for profiling to be done...
```

```
##             2.5 %      97.5 %
## imc    0.02898397  0.32575504
```

```
## poids -0.12623701 -0.02628422
```

Notons que les intervalles de confiance sont interprétable car ils ne contiennt pas 0.

Nous observons au premier abord (sans regarder les intervalles de confiance) que plus l'imc est élevé plus le risque d'être malade augmente car le signe du coefficient (0.17541) est positif. Au contraire plus le poids est élevé, plus le risque d'être malade diminue car le signe du coefficient (-0.07536) est négatif. Ainsi nous pouvons dire plus simplement que si l'imc augmente, le risque d'être malade augmente.
Adaptons une analyse plus précise en quantifiant :

Nous augmentons l'imc de 1 ($x_1 + 1$) et nous laissons le poids fixé à x_2.
Si l'imc augemente de 1, l'odds d'être malade est $exp(\beta_1)exp(x_1\beta_1 + x_2\beta_2)$

Nous pouvons donc voir qu'a poids fixé, augmenter l'imc de 1 va multiplier l'odds d'avoir une MCC par au moins $\exp(0.02898397) = 1.03$ et au plus $\exp(0.32575504) = 1.39$. Nous pouvons de plus voir qu'a imc fixé, augmenter le poids de 1 va multiplier l'odds d'avoir une MCC par au moins $\exp(-0.12623701) = 0.88$ et au plus $\exp(-0.02628422) = 0.97$.

Fixer une imc et augmenter le poids signifie que nous considèrons que la personne prend en taille ce qui paraît un peu hors réalite. En effet, si nous prenons une personne de 60 kg pour 1m65, l'imc est de 22.04. Si nous fixons l'imc à 22.04 et que cette personne prend 1kg, pour ne pas changer d'imc elle devra avoir pris également 1.4 cm en plus.

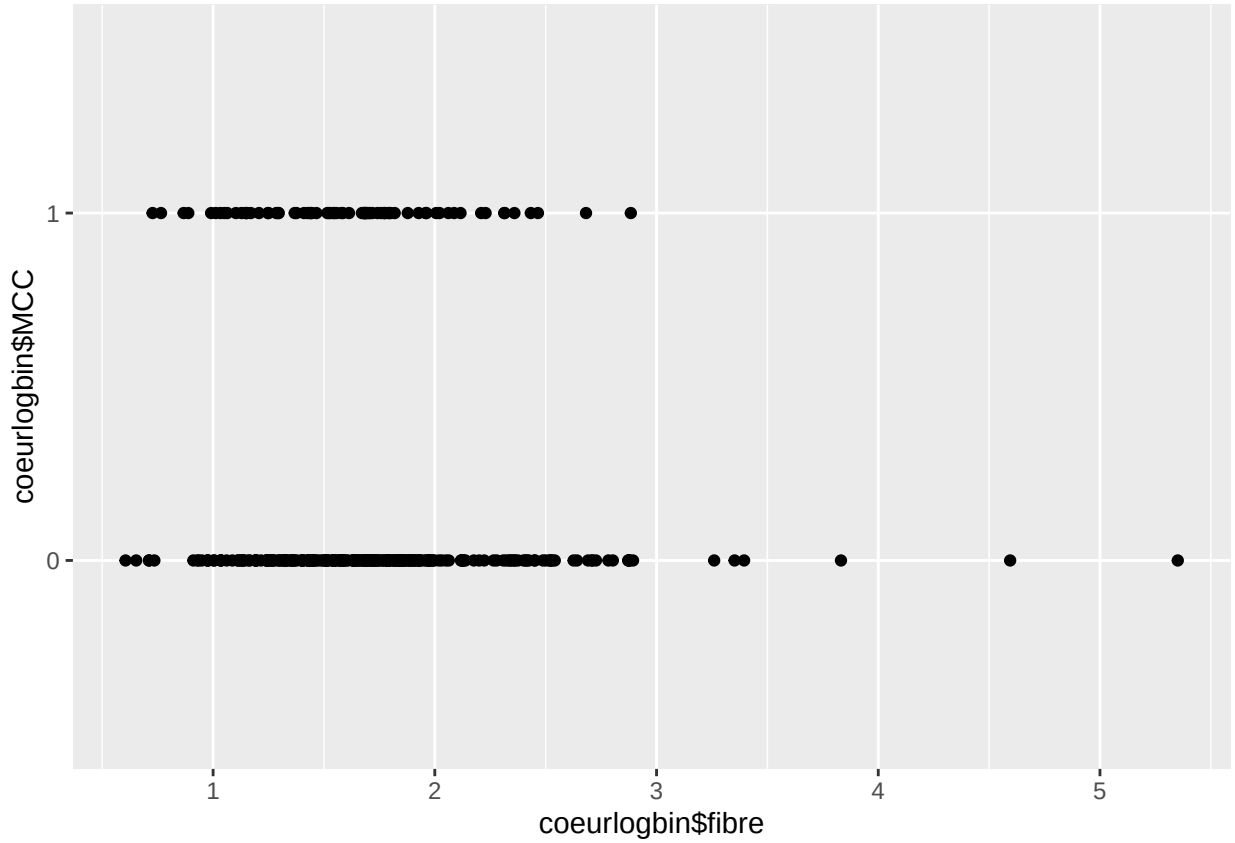
En conclusion de cette partie, il faut retenir que plus l'imc d'une personne augmente, plus le risque q'elle tombe malade d'une maladie coronarienne diminue. Avoir un imc stable et assez bas est un facteur protecteur du risque de maladie coronarienne.

MCC expliqué par la consommation de fibre

Nous trouvons également intéressant de regarder le lien entre la consommation de fibre et les maladies coronariennes.

Regardons la relation entre les maladies coronariennes et la consommation de fibre.

```
ggplot(coeurlogbin,aes(x=coeurlogbin$fibre,y=coeurlogbin$MCC))+geom_point()
```



Nous observons que les personnes mangeant plus de 2g de fibre par jour sont moins malades.

Effectuer désormais la régréssion logistique.

```
reslog<-glm(MCC~fibre-1,family = binomial(logit),data=coeurlogbin) #nous enelevons l'intercept qui n'es
summary(reslog)
```

```
##
## Call:
## glm(formula = MCC ~ fibre - 1, family = binomial(logit), data = coeurlogbin)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0030  -0.7823  -0.6959  -0.4816   2.0741
##
## Coefficients:
##       Estimate Std. Error z value Pr(>|z|)
## fibre -0.70288    0.07799  -9.013   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 454.70  on 328  degrees of freedom
## Residual deviance: 351.66  on 327  degrees of freedom
```

```
## AIC: 353.66
##
## Number of Fisher Scoring iterations: 4
```

```
confint(reslog)
```

```
## Waiting for profiling to be done...
```

```
##      2.5 %     97.5 %
## -0.8604690 -0.5542772
```

Augmenter la consommation de fibre de 1g par jour va multiplier l'odds d'avoir une MCC par au moins $\exp(-0.8604690) = 0.43$ et au plus $\exp(-0.5542772) = 0.57$. Autrement dit, si nous augmentons notre consommation de fibre de 1g par jour nous divisons par au moins (1/0.43)=2.32 l'odds de contracter une maladie coronarienne.

régréssion polytomique ordonné

```
coeur <-mutate(coeur,consom_dec =case_when(
  consommation<23~"peu",
  consommation>22 & consommation<30~"moyen",
  consommation>29 ~"beaucoup"))
coeur$consom_dec<-as.factor(coeur$consom_dec)
```

```
coeurlogbin1<-coeur%>%dplyr::select(-id,-dateEntree,-dateSortie,-dateNaissance,-statut,-hauteConsomatio
modele<-polr(consom_dec~.,data=coeurlogbin1)
stepAIC(modele)
```

```
## Start:  AIC=325.09
## consom_dec ~ emploi + moisEnqu_e + taille + poids + graisse +
##     fibre + imc + MCC
##
##              Df    AIC
## - moisEnqu_e 11 316.92
## - emploi      2 322.04
## - MCC         1 323.09
## - poids       1 324.30
## - taille      1 324.43
## - imc         1 324.47
## <none>          325.09
## - fibre       1 343.66
## - graisse     1 532.55
##
## Step:  AIC=316.92
## consom_dec ~ emploi + taille + poids + graisse + fibre + imc +
##     MCC
##
##           Df    AIC
## - emploi   2 313.65
## - MCC      1 314.95
## - poids    1 316.07
## - taille   1 316.08
## - imc      1 316.20
## <none>       316.92
```

```
## - fibre    1 337.12
## - graisse  1 520.53
##
## Step:  AIC=313.65
## consom_dec ~ taille + poids + graisse + fibre + imc + MCC
##
##           Df    AIC
## - MCC      1 311.71
## - taille   1 312.55
## - poids    1 312.62
## - imc      1 312.71
## <none>       313.65
## - fibre    1 334.19
## - graisse  1 518.76
##
## Step:  AIC=311.71
## consom_dec ~ taille + poids + graisse + fibre + imc
##
##           Df    AIC
## - taille   1 310.59
## - poids    1 310.66
## - imc      1 310.74
## <none>       311.71
## - fibre    1 332.24
## - graisse  1 517.42
##
## Step:  AIC=310.59
## consom_dec ~ poids + graisse + fibre + imc
##
##           Df    AIC
## - poids    1 308.69
## - imc      1 308.94
## <none>       310.59
## - fibre    1 332.25
## - graisse  1 515.77
##
## Step:  AIC=308.69
## consom_dec ~ graisse + fibre + imc
##
##           Df    AIC
## - imc      1 307.08
## <none>       308.69
## - fibre    1 330.53
## - graisse  1 514.32
##
## Step:  AIC=307.08
## consom_dec ~ graisse + fibre
##
##           Df    AIC
## <none>       307.08
## - fibre    1 331.70
## - graisse  1 515.49

## Call:
```

```
## polr(formula = consom_dec ~ graisse + fibre, data = coeurlogbin1)
##
## Coefficients:
##   graisse     fibre
## -1.289518 -1.808077
##
## Intercepts:
## beaucoup|moyen      moyen|peu
##      -20.93183      -14.60442
##
## Residual Deviance: 299.079
## AIC: 307.079
```

```
modele2<-polr(consom_dec~graisse+fibre,data=coeurlogbin1)
modele2
```

```
## Call:
## polr(formula = consom_dec ~ graisse + fibre, data = coeurlogbin1)
##
## Coefficients:
##   graisse     fibre
## -1.289518 -1.808077
##
## Intercepts:
## beaucoup|moyen      moyen|peu
##      -20.93183      -14.60442
##
## Residual Deviance: 299.079
## AIC: 307.079
```

```
confint(modele2)
```

```
## Waiting for profiling to be done...
```

```
##
## Re-fitting to get Hessian
```

```
##             2.5 %    97.5 %
## graisse -1.566429 -1.051114
## fibre   -2.599604 -1.074953
```

La commande polyr utilisée renvoie l'opposé du coefficient β considéré donc nous obtenons les résultats suivants pour nos coefficients:

QUESTION SUR LES MOINS

$$ln\left(\frac{odds(Y \leq moyen|graisse = x_1 + 1, fibre = x_2)}{odds(Y \leq moyen|graisse = x_1, fibre = x_2)}\right) = \beta_{graisse} = 1.29$$

A consommation de fibre fixée, augmenter la consommation de graisse de 1 g/jour va diviser $odds(Y \leq moyen)$ par au moins $exp(-1.56)$ et au plus $exp(-1.05)$. QUEST

$$ln\left(\frac{odds(Y \leq moyen|fibre = x_1 + 1, graisse = x_2)}{odds(Y \leq moyen|fibre = x_1, graisse = x_2)}\right) = \beta_{fibre} = 1.81$$

A consommation de graisse fixée, augmenter la consommation de fibre de 1 g/jour va diviser $odds(Y \leq moyen)$ par au moins $exp(-2.599)$ et au plus $exp(-1.07)$. QUEST

régréssion polytomique non-ordonné

```
library(nnet)
modele3<-multinom(emploi~.,data=coeurlogbin1)
stepAIC(modele3)
```

```
modele4<-multinom(emploi~poids+taille,data=coeurlogbin1)
```

```
## # weights:  12 (6 variable)
## initial  value 360.344831
## iter  10 value 302.773746
## iter  20 value 301.538107
## final  value 301.526738
## converged
```

```
modele4
```

```
## Call:
## multinom(formula = emploi ~ poids + taille, data = coeurlogbin1)
##
## Coefficients:
##           (Intercept)       poids     taille
## Conductor    34.13104 -0.07620753 -0.1694273
## Driver       22.11401 -0.01424113 -0.1231291
##
## Residual Deviance: 603.0535
## AIC: 615.0535
```

```
confint(modele4)
```

```
## , , Conductor
##
##                  2.5 %      97.5 %
## (Intercept) 30.3445677 37.91750699
## poids       -0.1114239 -0.04099117
## taille      -0.1959546 -0.14290007
##
## , , Driver
##
##                   2.5 %      97.5 %
## (Intercept) 16.95239383 27.27563010
## poids       -0.04357888  0.01509662
## taille      -0.15673368 -0.08952445
```

Bank worker est la modalité de référence pour emploi.

$$\beta_{poids|conductor} = ln\left(\frac{\frac{P(conductor|poids=x_1+1,taille=x_2)}{P(Bankworker|poids=x_1+1,taille=x_2)}}{\frac{P(conductor|poids=x_1,taille=x_2)}{P(Bankworker|poids=x_1,taille=x_2)}}\right) = -0.07620753$$

QUESTION POUR LES MO DE REF X1 X2

Toutes les autres covariables fixées à leurs modalités de référencee, un 1kg en plus va multiplier par au moins $exp(-0.1114239)$ = la préférence de conductor par rapport à Bank worker.

$$\beta_{taille|conductor} = ln\left(\frac{\frac{P(conductor|taille=x_1+1,poids=x_2)}{P(Bankworker|taille=x_1+1,poids=x_2)}}{\frac{P(conductor|taille=x_1,poids=x_2)}{P(Bankworker|taille=x_1,poids=x_2)}}\right) = -0.1694273$$

QUESTION POUR LES MO DE REF X1 X2

Toutes les autres covariables fixées à leurs modalités de référencee, un 1cm en plus va multiplier par au moins $exp(-0.1959546)$ = la préférence de conductor par rapport à Bank worker.

Modèle de Cox

Dans cette partie, nous allons chercher à répondre à la problématique suivante : à une date donnée, quelle sera le taux de nouveaux malades dans la population étudiée ?

Nous allons désormais, netoyer la base de données et faire des transformation de format des variables pour pouvoir utiliser le model de Cox. Les variables qui contiennent une date (date entrée, date sortie et date de naissance) sont de type charactère donc dans un premier temps, nous allons convertir toutes ces variable en type Date.

```
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 3.6.2
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
coeur$dateEntree<-as.Date(coeur$dateEntree,format="%d/%m/%y")
year(coeur$dateEntree)<-1900+year(coeur$dateEntree) %% 100
coeur$dateSortie<-as.Date(coeur$dateSortie,format="%d/%m/%y")
year(coeur$dateSortie)<-1900+year(coeur$dateSortie) %% 100

coeur<-coeur%>% mutate(time = coeur$dateSortie - coeur$dateEntree )
```

Ensuite, la fonction qui execute le modèle de cox à besoin de valeurs numériques représentant les dates. Sous R, chaque date est repésenté par un nombre de jour à partir d'une date d'origine : le 1 janvier 1970. Nous allons donc créer une variable que récupérera ce nombre pour chaque date correspondant à chaque individus.

```
coeur2<-coeur
year(coeur2$dateEntree)<-1900+year(coeur2$dateEntree) %% 100
coeur2<-coeur2%>%mutate(date_entree_num=as.numeric(as.Date(coeur2$dateEntree)))
year(coeur2$dateSortie)<-1900+year(coeur2$dateSortie) %% 100
coeur2<-coeur2%>%mutate(date_sortie_num=as.numeric(as.Date(coeur2$dateSortie)))
```

Nous pouvons désormais faire nos analyse avec le modèle de Cox. Pour une prèmière analyse, nous estimerons notre odèle en prenant en compte toute les covariables possibles, présentes dans la table de données. On obtient les résultats suivant :

```
coeurcox<-coeur%>%dplyr::select(-id,-dateEntree,-dateSortie,-dateNaissance,-statut,-hauteConsomation,-m
 library(survival)
```

```
## Warning: package 'survival' was built under R version 3.6.3
```

```
  survie=Surv(coeur2$date_entree_num,coeur2$date_sortie_num,coeur$MCC)
 res=coxph(survie~coeur2$fibre+coeur2$taille+coeur2$poids+coeur2$consommation+coeur2$emploi+coeur2$grais
```

Avant d'interpréter plus avant le modèle, il est utile de voir si l'hypothèse que les formes multiplicatives et lescovariables sont indépendantes du temps est vérifiée . Cette vérification est à faire sur le modèle global, avant même d'interpréter les tests (qui ne sont pas valables lorsque l'hypothèse n'est pas vérifiée), et avant de sélectionner des variables : il se peut qu'une covariable ait un effet non significatif lorsque cet effet est moyenné dans le temps mais qu'elle ait une interaction significative avec le temps. C'est pourquoi il vaut mieux tester l'hypothèse de HP avant d'interpréter la significativité des effets

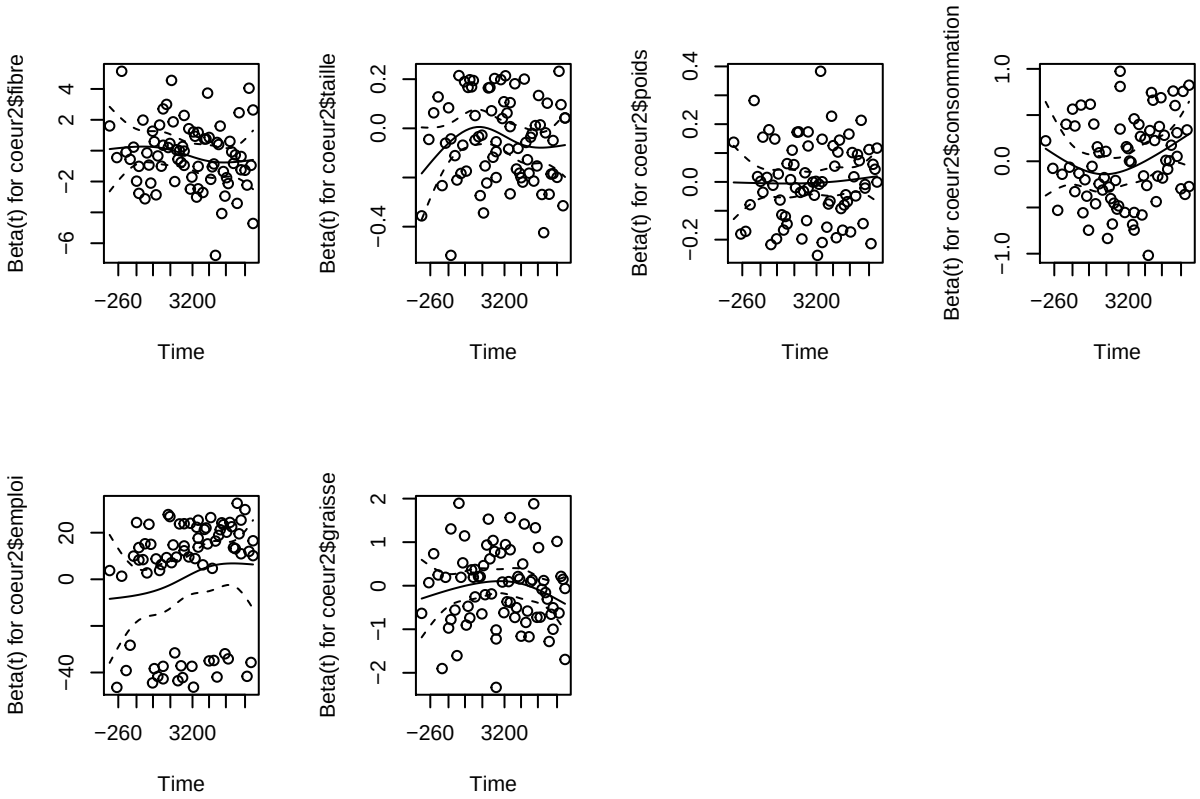
```
res.c=cox.zph(res)
res.c
```

```
##                        chisq df     p
## coeur2$fibre         0.00153  1 0.969
## coeur2$taille        0.01130  1 0.915
## coeur2$poids         0.07142  1 0.789
## coeur2$consommation 2.96334  1 0.085
## coeur2$emploi        4.23311  2 0.120
## coeur2$graisse       1.53309  1 0.216
## GLOBAL               9.09338  7 0.246
```

Le test global de validité de l'hypothèse de HP conduit à ne pas rejeter cette hypothèse au seuil de 5% : aucune covariable n'a un effet dépendant du temps.

Il est possible de générer des graphiques de résidus pour des prédicteurs individuels. Dans les graphiques, une pente non nulle est une preuve contre la proportionnalité. Remarque : R trace le graphe des résidus en incluant par défaut un lissage par des splines (trait plein), et des IC à 95%.

```
par(mfrow=c(2,4))
plot(res.c)
```



A l'oeil nu, on observe que a peu près que toute les droites sont à l'horizontales. Peut etre que la droites dans le graphiques de et emploie est a preine incliné mais comme la p_{valeur} restait supérieur à 5 pourcent, on va considérer que cette covariable reste indépendante du temps. En revanche la droite de consommation est aussi un peu uplus inclinée mais sa $p_{valeur} = 0.085$ sup´rieur à 5 pourcent mais supérieur à 10 pourcent quand meme. On va considérer que cette covariable dépend du temps. Il ne serait pas judicieux de l'étudier avec le modèle de cox à risques proportionnelles.

```
res=coxph(survie~coeur2$fibre+coeur2$taille+coeur2$poids+coeur2$emploi+coeur2$graisse,id=coeur$id)
summary(res)
```

```
## Call:
## coxph(formula = survie ~ coeur2$fibre + coeur2$taille + coeur2$poids +
##     coeur2$emploi + coeur2$graisse, id = coeur$id)
##
##   n= 328, number of events= 76
##
##                              coef  exp(coef)   se(coef)      z Pr(>|z|)
## coeur2$fibre           -0.2921471  0.7466587  0.2681083 -1.090   0.2759
## coeur2$taille          -0.0477640  0.9533588  0.0207428 -2.303   0.0213 *
## coeur2$poids            0.0006888  1.0006890  0.0137675  0.050   0.9601
## coeur2$emploiConductor  0.0221884  1.0224364  0.3210355  0.069   0.9449
## coeur2$emploiDriver    -0.0779503  0.9250104  0.2987611 -0.261   0.7942
## coeur2$graisse         -0.0453903  0.9556244  0.0567640 -0.800   0.4239
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                        exp(coef) exp(-coef) lower .95 upper .95
```

```
## coeur2$fibre              0.7467     1.3393    0.4415    1.2628
## coeur2$taille             0.9534     1.0489    0.9154    0.9929
## coeur2$poids              1.0007     0.9993    0.9740    1.0281
## coeur2$emploiConductor    1.0224     0.9781    0.5450    1.9182
## coeur2$emploiDriver       0.9250     1.0811    0.5150    1.6613
## coeur2$graisse            0.9556     1.0464    0.8550    1.0681
##
## Concordance= 0.617  (se = 0.031 )
## Likelihood ratio test= 13.81  on 6 df,   p=0.03
## Wald test            = 13.88  on 6 df,   p=0.03
## Score (logrank) test = 13.95  on 6 df,   p=0.03
```

Le test de Wald teste l'effet d'une covariable, les autres étant dans le modèle. S'il n'est pas significatif, cela ne veut pas dire qu'il ne le serait pas dans le modèle constitué uniquement de cette covariable.

Le test de Wald pour la covariable « taille » montre que le coefficients correspondants est fortement significatifs au seuil 5% $p_{values} < 0.05$: cette covariable a un effet important sur l'incidence instentanée à un moment t. Les autres covariables ne modifient pas significativement cette incidence lorsque taille est dans le modèle.

On sélectionne les variables pas à pas c'est à dire, on enlève celle dont la statistique de Wald est la plus faible avec p_{valeur} la plus élevée. Ensuite on refait tourner le modèle et on recommence jusqu'à obtention de toutes les variables significatives.

On obtiens le modèle final suivant :

```
res=coxph(survie~coeur2$taille,id=coeur$id)
summary(res)
```

```
## Call:
## coxph(formula = survie ~ coeur2$taille, id = coeur$id)
##
##   n= 328, number of events= 76
##
##                    coef exp(coef) se(coef)      z Pr(>|z|)
## coeur2$taille -0.05589   0.94565  0.01712 -3.264   0.0011 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##               exp(coef) exp(-coef) lower .95 upper .95
## coeur2$taille    0.9456      1.057    0.9144    0.9779
##
## Concordance= 0.588  (se = 0.03 )
## Likelihood ratio test= 10.51  on 1 df,   p=0.001
## Wald test            = 10.66  on 1 df,   p=0.001
## Score (logrank) test = 10.68  on 1 df,   p=0.001
```

```
confint(res)
```

```
##                       2.5 %      97.5 %
## coeur2$taille -0.08944234 -0.02233322
```

Les p-values des 3 tests sont très inférieures à 5%, ce qui montre que l'ajustement d'un modèle de Cox est très pertinent au seuil 5%.

Seul la co-variable « taille » a un coefficients fortement significatifs au seuil 5% $p_{values} < 0.05$.

L'exponentielle des coefficients mesure l'effet multiplicatif d'une augmentation de la covariable d'une unité sur le taux, toutes choses égales par ailleurs.

En supposant que toutes les autres covariables restent constantes, la taille a un effet positif sur le taux de nouveaux malades à un instant donnée, l'effet d'une augmentation de taille de 1 unité réduit le taux de nouveau malade à l'instant t par un facteur de en moyenne, au moins, $exp(-0.08944234) = 0.9144$ c'est à dire par $100\% - 91,44\% = 8,56\%$.

Auparavant, nous avont constater que dans le modèle global, tout les autres coefficients n'étaient pas significatifs au seuil de 5 pourcent. Mais qu'en est t-il si on test avec un modèle avec seulement 1 covariable. Regardons ce qui se passe pour fibre :

```
res=coxph(survie~coeur$fibre,id=coeur$id)
summary(res)
```

```
## Call:
## coxph(formula = survie ~ coeur$fibre, id = coeur$id)
##
##   n= 328, number of events= 76
##
##                coef exp(coef) se(coef)      z Pr(>|z|)
## coeur$fibre -0.5315    0.5878   0.2401 -2.213   0.0269 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##             exp(coef) exp(-coef) lower .95 upper .95
## coeur$fibre    0.5878      1.701    0.3671     0.941
##
## Concordance= 0.569  (se = 0.032 )
## Likelihood ratio test= 5.47  on 1 df,   p=0.02
## Wald test            = 4.9  on 1 df,   p=0.03
## Score (logrank) test = 4.74  on 1 df,   p=0.03
```

```
confint(res)
```

```
##                 2.5 %      97.5 %
## coeur$fibre -1.002119 -0.06078271
```

« fibre » a un coefficients significatifs au seuil 5% $p_{values} < 0.05$.

En supposant que toutes les autres covariables restent constantes, fibre a un effet positif sur le taux de nouveaux malades à un instant donnée, l'effet d'une augmentation de taille de 1 unité réduit le taux de nouveau malade à l'instant t par un facteur de, au moins, $exp(-0.5315) = 0.5877227$ c'est à dire par $100\% - 58.77\% = 41,33\%$.

Testons maintenant pour la variable graisse

```
 res=coxph(survie~coeur$graisse,id=coeur$id)
summary(res)
```

```
## Call:
## coxph(formula = survie ~ coeur$graisse, id = coeur$id)
##
##   n= 328, number of events= 76
##
##                   coef exp(coef) se(coef)      z Pr(>|z|)
## coeur$graisse -0.09396   0.91032  0.05125 -1.833   0.0668 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##               exp(coef) exp(-coef) lower .95 upper .95
```

```
## coeur$graisse    0.9103      1.099    0.8233     1.007
##
## Concordance= 0.564  (se = 0.033 )
## Likelihood ratio test= 3.5  on 1 df,   p=0.06
## Wald test            = 3.36  on 1 df,   p=0.07
## Score (logrank) test = 3.35  on 1 df,   p=0.07
```

```
confint(res)
```

```
##                    2.5 %      97.5 %
## coeur$graisse -0.1944183 0.006496239
```

« graisse » a une $p_{values} > 0.05$. De plus, l'intervalle de confiance n'est pas interpretable car il contient 0. Nous ne pouvons pas faire d'estimation avec ce modèle.

```
res=coxph(survie~coeur2$emploi,id=coeur2$id)
summary(res)
```

```
## Call:
## coxph(formula = survie ~ coeur2$emploi, id = coeur2$id)
##
##   n= 328, number of events= 76
##
##                           coef exp(coef) se(coef)     z Pr(>|z|)
## coeur2$emploiConductor 0.4637    1.5899   0.2786 1.664   0.0961 .
## coeur2$emploiDriver    0.2240    1.2511   0.2841 0.788   0.4305
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                        exp(coef) exp(-coef) lower .95 upper .95
## coeur2$emploiConductor     1.590     0.6290    0.9209     2.745
## coeur2$emploiDriver        1.251     0.7993    0.7168     2.183
##
## Concordance= 0.548  (se = 0.031 )
## Likelihood ratio test= 2.73  on 2 df,   p=0.3
## Wald test            = 2.78  on 2 df,   p=0.2
## Score (logrank) test = 2.82  on 2 df,   p=0.2
```

```
confint(res)
```

```
##                              2.5 %    97.5 %
## coeur2$emploiConductor -0.08243578 1.0097674
## coeur2$emploiDriver    -0.33288997 0.7808997
```

L'intervalle de confiance comprend 0, rien n'est interpretable

Conclusion : Comment diminuer les rirsques d'avoir une maladie coronarienne.

Manger des fibres