

# Régression linéaire

Florent Benaych-Georges

Université Paris Descartes

Janvier 2015

*Séries temporelles pratiques*

```
1 require(caschrono); require(fSeries)
2
3 data(popfr) #on importe les donnees de popfr.
4 temps1=time(popfr); temps2=temps1^2; temps3=temps1^3;
5 temps4=temps1^4; temps5=temps1^5; temps6=temps1^6
6
7 reg=lm(popfr~temps1+temps2+temps3+temps4+temps5+temps6) #peu
8
9 str(reg)
10 cont_temps1=seq(min(temps1),max(temps1),.1)
11 cont_temps0=rep(1,length(cont_temps1)); cont_temps2=cont_temps1^2;
12 cont_temps4=cont_temps1^4; cont_temps5=cont_temps1^5; cont_temps6=cont_temps1^6
13
14 reg$coeff[is.na(reg$coeff)]=0;
15 plot.ts(popfr,xlab='annee',ylab='population',main="Populatio")
16
17 pol_popfr=reg$coeff[1]*cont_temps0+reg$coeff[2]*cont_temps1+
18 reg$coeff[4]*cont_temps3+reg$coeff[5]*cont_temps4+reg$coeff[6]*cont_temps6
19 lines(cont_temps1,pol_popfr,col="red")
```

(file: LinReg1.R)

```

1 Ad=read.csv(file="Advertising.csv",header=T)
2 str(Ad)
3 AdTraining=Ad[1:170,]
4 AdTest=Ad[171:200,]
5 lin=lm(Sales~TV+Radio+News,AdTraining)
6 lin$coefficients
7 summary(lin)
8 lin$fitted.values
9 lin$res #res=residuals
10 mean_yTr=mean(AdTraining$Sales)
11 (R2Tr=1-sum((lin$res)^2)/sum((AdTraining$Sa-mean_yTr)^2))
12
13 lin2=lm(Sales~TV*Radio,AdTraining)
14 prediction=predict(lin2,newdata=AdTest)
15 class(prediction)
16 myTe=mean(AdTest$Sales)
17 (R2Te=1-
18     sum((prediction-AdTest$Sa)^2)/sum((AdTest$Sa-myTe)^2))

```

(file: Linear\_Regression\_Example.R)