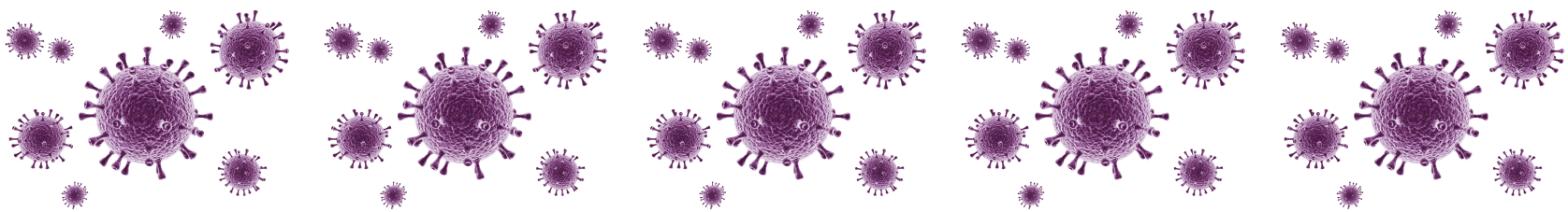
Modèles Linéaires Appliqués / Régression
Analyse Discriminante & Courbe ROC

Arthur Charpentier

UQAM

Hiver 2020 - COVID-19 # 6



Analyse Discriminante de Fisher

Supposons que le but soit *simplement* de prédire une classe, $\widehat{y} \in \{0, 1\}$ (ou plus généralement $\widehat{y} \in \{a_1, a_2, \cdots, a_J\}$)

$$m^\star(\boldsymbol{x}) = \underset{y \in \{0,1\}}{\text{argmin}} \{\mathbb{P}[Y = y | \boldsymbol{X} = \boldsymbol{x}]\}$$

soit

$$m^\star(\boldsymbol{x}) = \underset{y \in \{0,1\}}{\text{argmin}} \left\{ \frac{\mathbb{P}[\boldsymbol{X} = \boldsymbol{x} | Y = y]}{\mathbb{P}[\boldsymbol{X} = \boldsymbol{x}]} \right\}$$

(où $\mathbb{P}[\boldsymbol{X} = \boldsymbol{x}]$ devient $f(\boldsymbol{x})$ dans le cas continu).
Si y prend deux valeurs – i.e. $\{0, 1\}$

$$m^\star(\boldsymbol{x}) = \begin{cases} 1 \text{ si } \mathbb{E}(Y | \boldsymbol{X} = \boldsymbol{x}) > \dfrac{1}{2} \\ 0 \text{ sinon} \end{cases}$$

Analyse Discriminante de Fisher

L'ensemble

$$\mathcal{D}_S = \left\{ \boldsymbol{x} : \ \mathbb{E}(Y|\boldsymbol{X} = \boldsymbol{x}) = \frac{1}{2} \right\}$$

est appelé frontière de décision.

Supposons que $\boldsymbol{X}|Y = 0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ et $\boldsymbol{X}|Y = 1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, alors, si r_y^2 est la distance de Manalahobis de $\boldsymbol{x}$ à $\boldsymbol{\mu}_y$

$$r_y^2 = [\boldsymbol{x} - \boldsymbol{\mu}_y]^\top \boldsymbol{\Sigma}_y^{-1} [\boldsymbol{x} - \boldsymbol{\mu}_y] \text{ pour } y \in \{0, 1\},$$

$$m^\star(\boldsymbol{x}) = \begin{cases} 1 \text{ si } r_1^2 < r_0^2 + 2\log \dfrac{\mathbb{P}(Y=1)}{\mathbb{P}(Y=0)} + \log \dfrac{|\boldsymbol{\Sigma}_0|}{|\boldsymbol{\Sigma}_1|} \\ 0 \text{ sinon} \end{cases}$$

Analyse Discriminante de Fisher

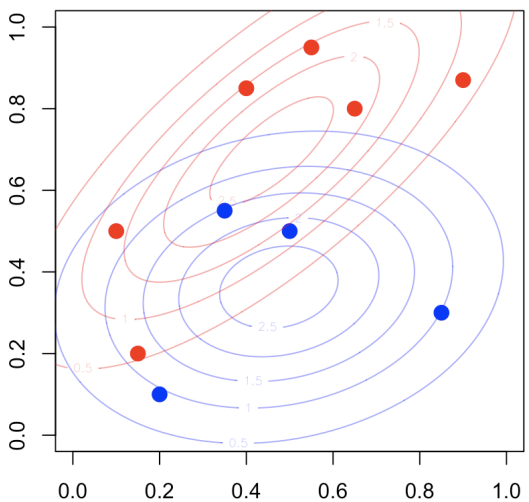
Soit δ_y la fonction définie (pour $y \in \{0, 1\}$) par

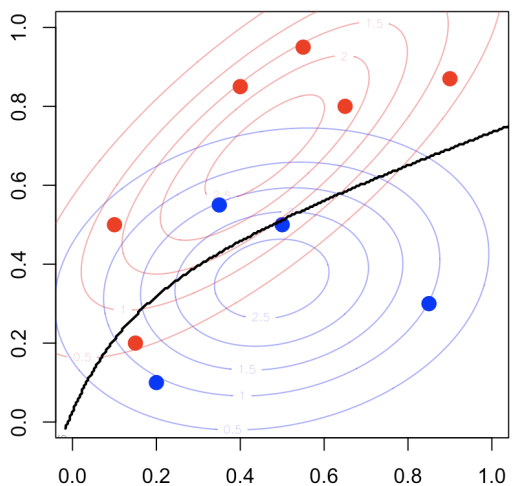
$$\delta_y(\boldsymbol{x}) = -\frac{1}{2}\log|\boldsymbol{\Sigma}_y| - \frac{1}{2}[\boldsymbol{x} - \boldsymbol{\mu}_y]^\top \boldsymbol{\Sigma}_y^{-1}[\boldsymbol{x} - \boldsymbol{\mu}_y] + \log \mathbb{P}(Y = y)$$

telle que la frontière de décision soit

$$\{\boldsymbol{x} \text{ tel que } \delta_0(\boldsymbol{x}) = \delta_1(\boldsymbol{x})\}$$

qui est quadratique en $\boldsymbol{x}$



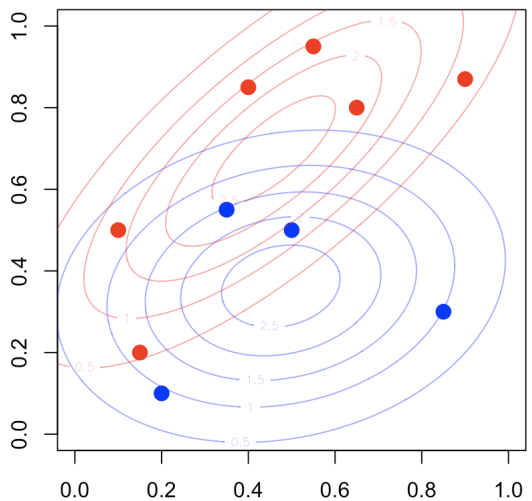


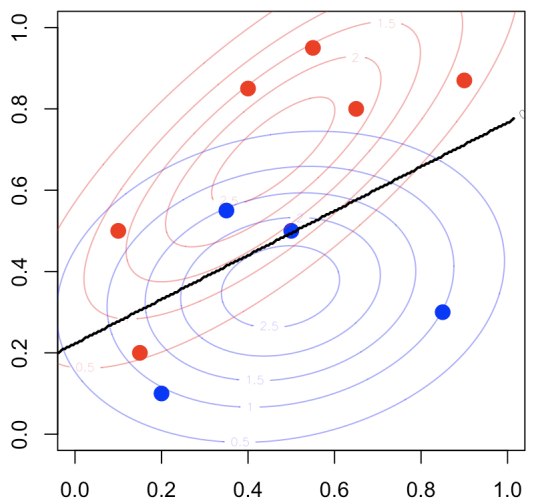
Analyse Discriminante de Fisher

Fisher (1936) a rajouté hypothèse $\boldsymbol{\Sigma}_0 = \boldsymbol{\Sigma}_1$. Alors

$$\delta_y(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_y - \frac{1}{2} \boldsymbol{\mu}_y^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_y + \log \mathbb{P}(Y = y)$$

et la frontière de décision est linéaire en $\boldsymbol{x}$





Analyse Discriminante de Fisher

Si $\boldsymbol{X}|Y=0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma})$ et $\boldsymbol{X}|Y=1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$ alors

$$\log \frac{\mathbb{P}(Y=1|\boldsymbol{X}=\boldsymbol{x})}{\mathbb{P}(Y=0|\boldsymbol{X}=\boldsymbol{x})}$$

est égal à

$$\boldsymbol{x}^\top \boldsymbol{\Sigma}^{-1}[\boldsymbol{\mu}_y] - \frac{1}{2}[\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0]^\top \boldsymbol{\Sigma}^{-1}[\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0] + \log \frac{\mathbb{P}(Y=1)}{\mathbb{P}(Y=0)}$$

qui est linéaire en $\boldsymbol{x}$, autrement dit

$$\log \frac{\mathbb{P}(Y=1|\boldsymbol{X}=\boldsymbol{x})}{\mathbb{P}(Y=0|\boldsymbol{X}=\boldsymbol{x})} = \boldsymbol{x}^\top \beta$$

ce qui rappelle la régression logistique...

Test et Décision

$y \in \{0, 1\}$ et on va construire $\widehat{y} \in \{0, 1\}$ – via $\widehat{y} = \mathbf{1}(\widehat{p} > s)$

décision \ vérité	−	+
−	true negative	false negative
+	false positive	true positive

décision \ vérité	−	+
−	bonne décision	type 2 error
+	type 1 error	bonne décision

On a un tradeoff entre les types d'erreurs, cf base rate fallacy.

Cf théorie des tests, où on teste H_0 – qui peut être valide (ou pas) – et on doit prendre une décision : rejeter H_0 ou accepter H_0.

Test et Décision

$$\text{Prevalence } \frac{200}{10,000} = 2\%$$

$$\text{Specificity } \frac{9,751}{9,800} = 99.5\%$$

$$\text{Sensitivity } \frac{100}{200} = 50\%$$

$$\text{Positive Predictive Value } \frac{100}{149} \sim 67\%$$

$$\text{Specificity } \frac{9,310}{9,800} = 95\%$$

$$\text{Positive Predictive Value } \frac{100}{590} \sim 17\%$$

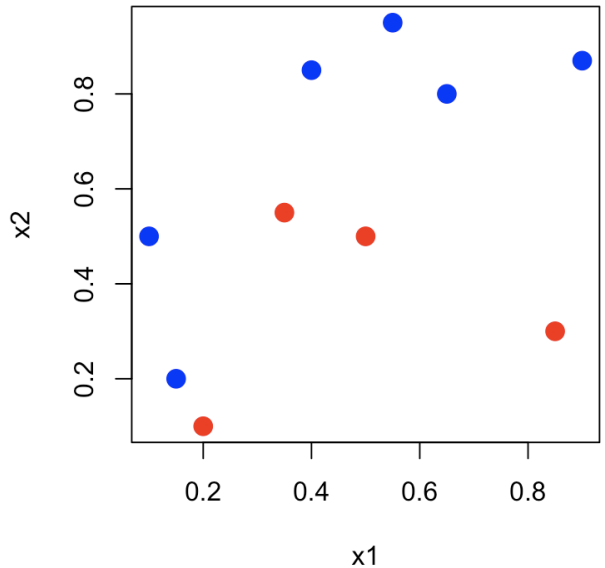
	− well	+ disease	
−	9,751	100	9,851
+	49	100	149
	9,800	200	10,000

	well	disease	
−	9,310	100	9,410
+	490	100	590
	9,800	200	10,000

Cf Wainer & Savage (2008) (until proven guilty: False positives and the war on terror)

Courbe ROC

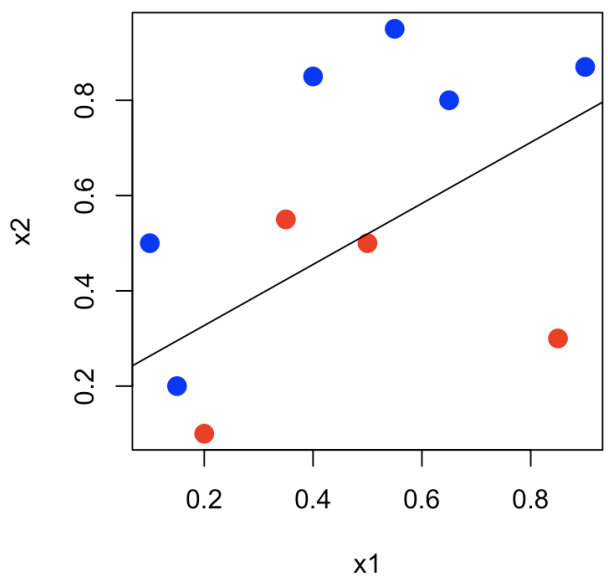
```
> x1 = c(.4,.55,.65,.9,.1,.35,
.5,.15,.2,.85)
> x2 = c(.85,.95,.8,.87,.5,.55,
.5,.2,.1,.3)
> y = c(1,1,1,1,1,0,0,1,0,0)
> df = data.frame(x1=x1,x2=x2,
y=as.factor(y))
> plot(x1,x2,col=1+y)
> reg = glm(y~x1+x2,data=df,
family=binomial(link = "logit"))
> b = coefficients(reg)
> abline(a=-b[1]/b[3],b=-b[2]/b[3])
```



$$\mathbb{P}[Y = 1|\boldsymbol{X} = \boldsymbol{x}] = \mathbb{P}[Y = 0|\boldsymbol{X} = \boldsymbol{x}] \text{ si}$$

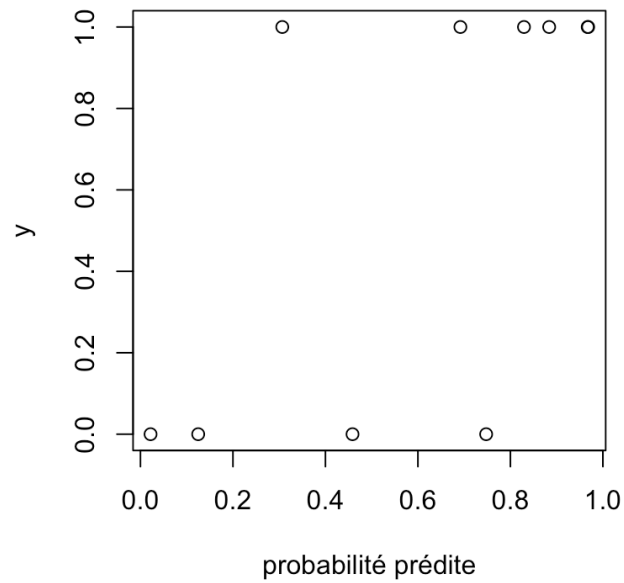
$$e^{\boldsymbol{x}^\top \beta} = 1 \text{ ou } \boldsymbol{x}^\top \beta = 0$$

soit ici $\beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0$.



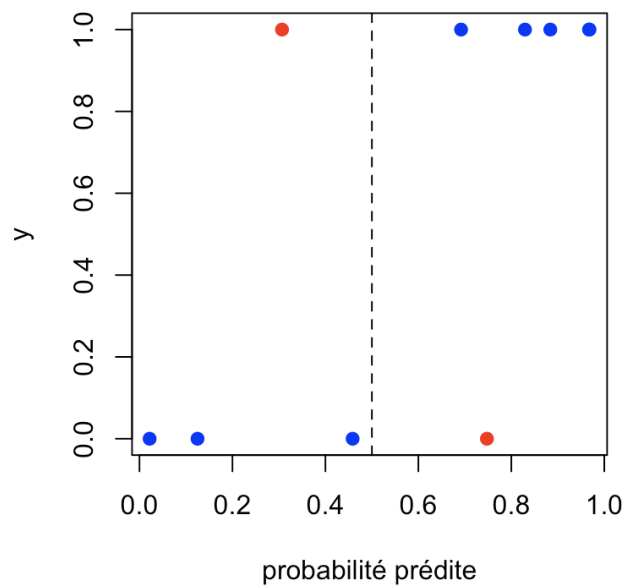
Courbe ROC

```
> Y = df$y
> S = predict(reg,type="response")
> plot(S,y)
> seuil = .5
> Yhat = (S>seuil)*1
> plot(S,y,col=1+(y==Yhat))
> abline(v=seuil,lty=2)
```



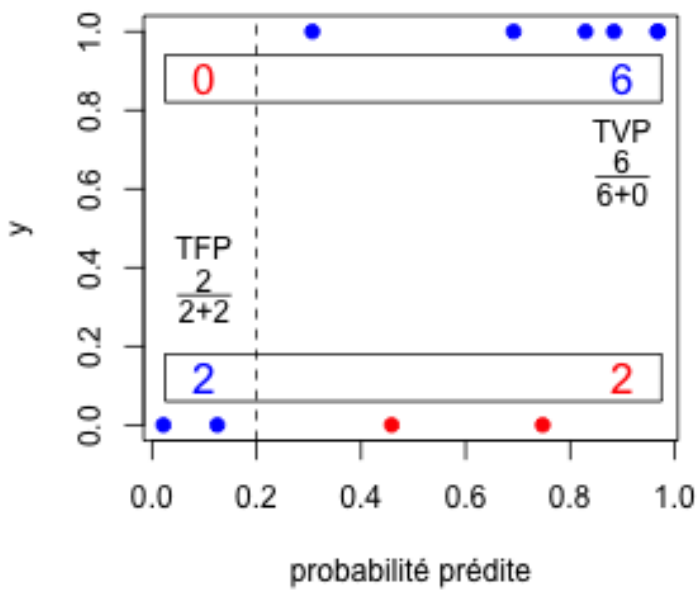
$$\widehat{y}_i = \begin{cases} 1 & \text{si } \widehat{p}_i > \text{seuil} \\ 0 & \text{si } \widehat{p}_i \leq \text{seuil} \end{cases}$$

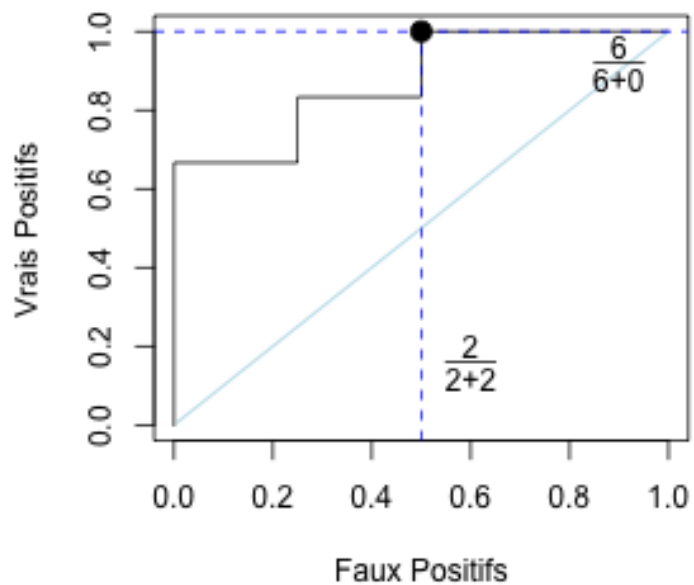
```
> table(Yhat,Y)
     Y
Yhat 0 1
   0 3 1
   1 1 5
```



Courbe ROC

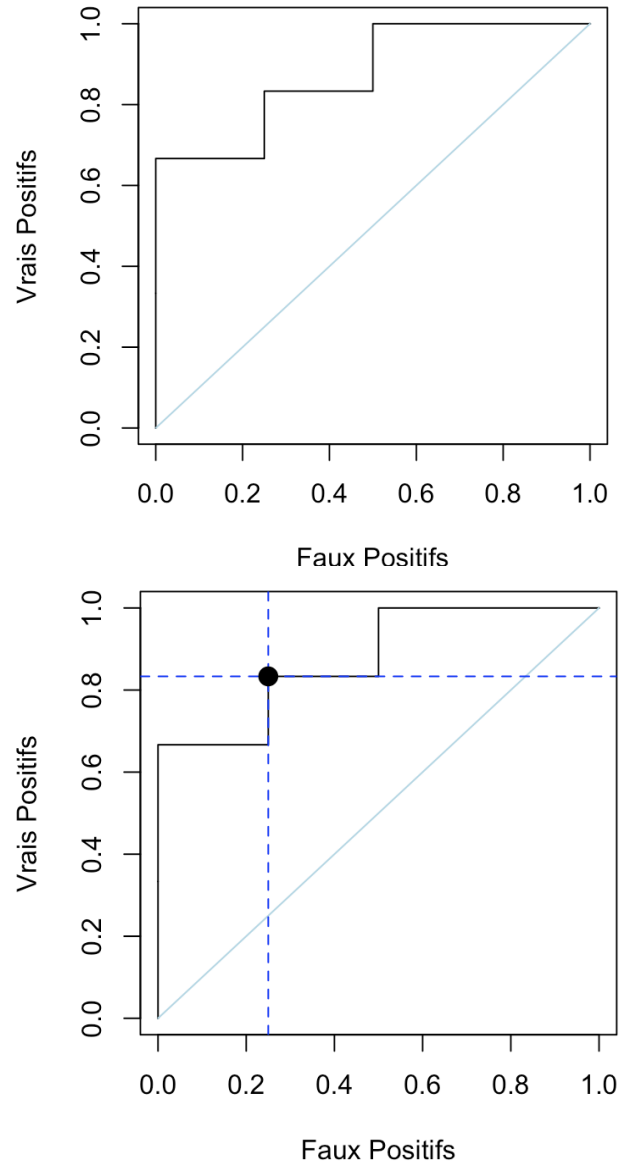
$$FPR = \frac{\mathbb{P}[y = 0, \widehat{y} = 1]}{\mathbb{P}[y = 0]} \text{ et } TPR = \frac{\mathbb{P}[y = 1, \widehat{y} = 1]}{\mathbb{P}[y = 1]}$$





Courbe ROC

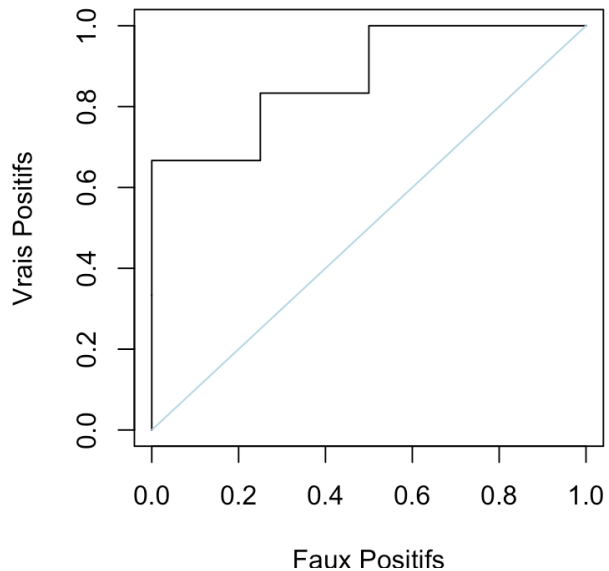
```
> roc.curve=function(s){
  Ps=(S>s)*1
  FP=sum((Ps==1)*(Y==0))/sum(Y==0)
  TP=sum((Ps==1)*(Y==1))/sum(Y==1)
  vect=c(FP,TP)
  names(vect)=c("FPR","TPR")
  return(vect) }
> u = seq(0,1,length=251)
> V = Vectorize(roc.curve)(u)
> plot(t(V),type="s")
> table(Yhat,Y)
     Y
Yhat  0  1
   0  3  1
   1  1  5
> sum((Yhat)*(Y==0))/sum(Y==0)
[1] 0.25
> sum((Yhat==1)*(Y==1))/sum(Y==1)
[1] 0.8333333
```



Courbe ROC

De nombreux packages permettent de tracer des courbes ROC, dont `ROCR` (ou `pROC`, `plotROC`)

```
> library(ROCR)
> pred = prediction(S,Y)
> plot(performance(pred,"tpr","fpr
    "))
> auc.perf = performance(pred,
    measure = "auc")
> auc.perf@y.values[[1]]
[1] 0.875
```

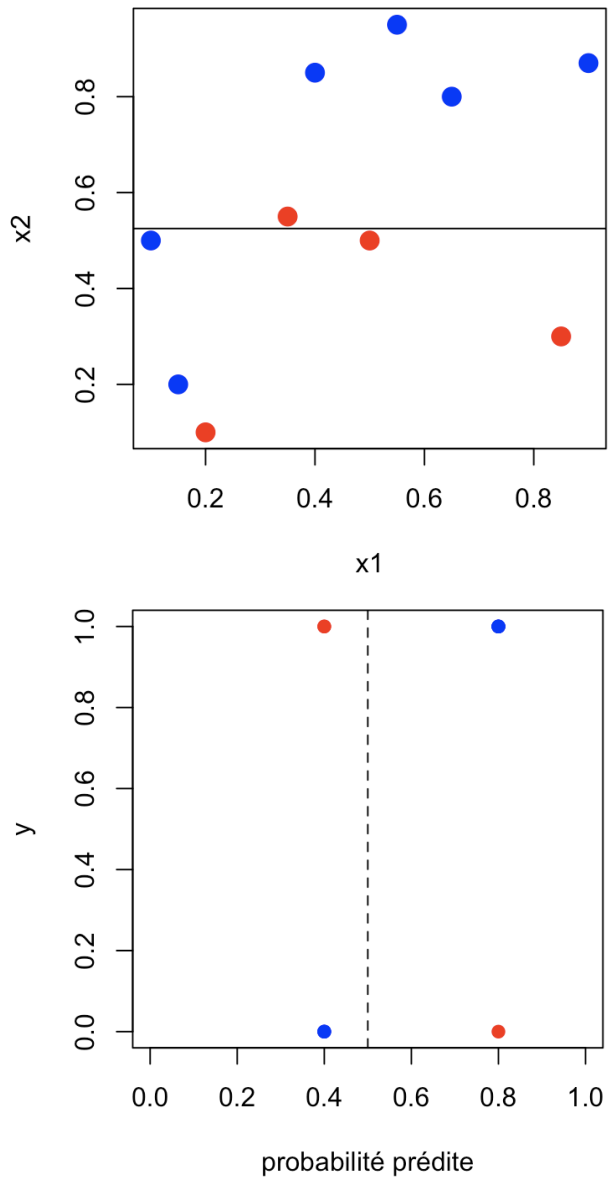


L'AUC – aire sous la courbe – donne une idée de la qualité de la classification.

Courbe ROC

Régression de y sur $\mathbf{1}_{[.525,\infty)}(x_2)$

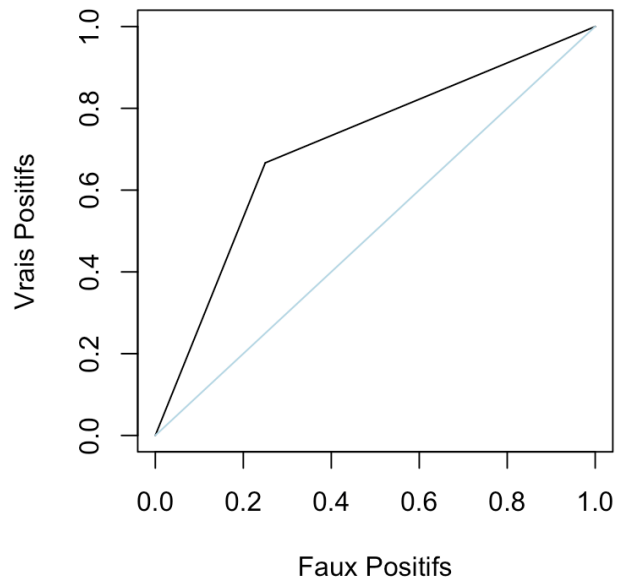
```
> reg = glm(y~I(x2> .525),data=df,
family=binomial(link = "logit"))
> abline(h=.525)
> Y = df$y
> S = predict(reg,type="response")
> plot(S,y,xlim=0:1)
> seuil = .5
> Yhat = (S>seuil)*1
> table(Yhat,Y)
     Y
Yhat 0 1
   0 3 |\aftergroup\rcred|2|\
    aftergroup\blackcolor|
   1 |\aftergroup\rcred|4|\
    aftergroup\blackcolor| 5
```

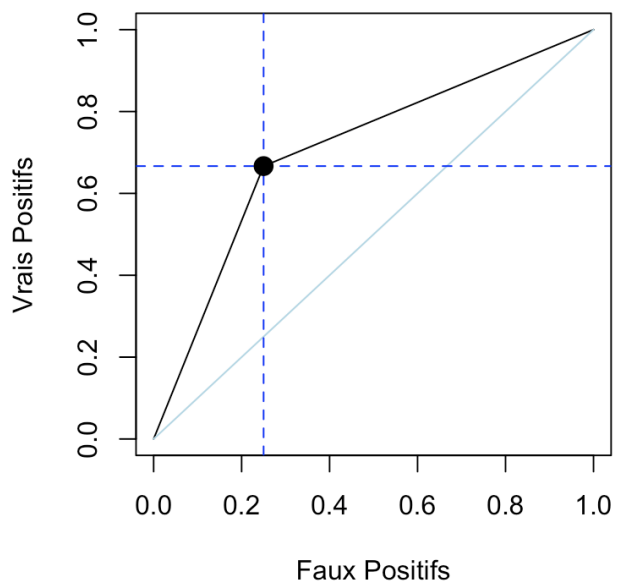


Courbe ROC

La courbe ROC est linéaire par morceaux

```
> pred = prediction(S,Y)
> plot(performance(pred,"tpr","fpr"))
> table(Yhat,Y)
     Y
Yhat 0 1
   0 3 2
   1 1 4
> auc.perf = performance(pred,
    measure = "auc")
> auc.perf@y.values[[1]]
[1] 0.7083333
```



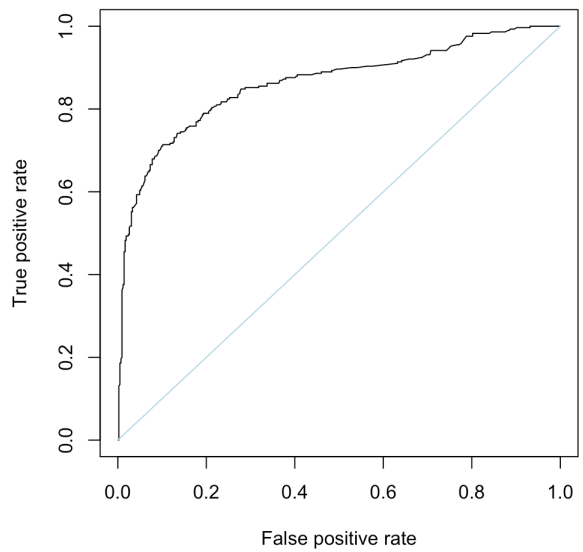


Survie des Passagers du Titanic

y : indicatrice de survie d'un passager du Titanic

```
> loc = "http://freakonometrics.free.fr/titanic.RData"
> download.file(loc, "titanic.RData")
> load("titanic.RData")
> base = base[!is.na(base$Age),1:7]
> reg = glm(Survived ~ Sex+poly(Age,3)+Pclass+SibSp,
  family = "binomial", data = base)
```

```
> library(ROCR)
> Y = base$Survived
> S = predict(reg,type="response")
> pred = prediction(S,Y)
> plot(performance(pred,"tpr","fpr"))
> performance(pred, measure = "auc")
  @y.values[[1]]
[1] 0.8627358
```



Survie des Passagers du Titanic

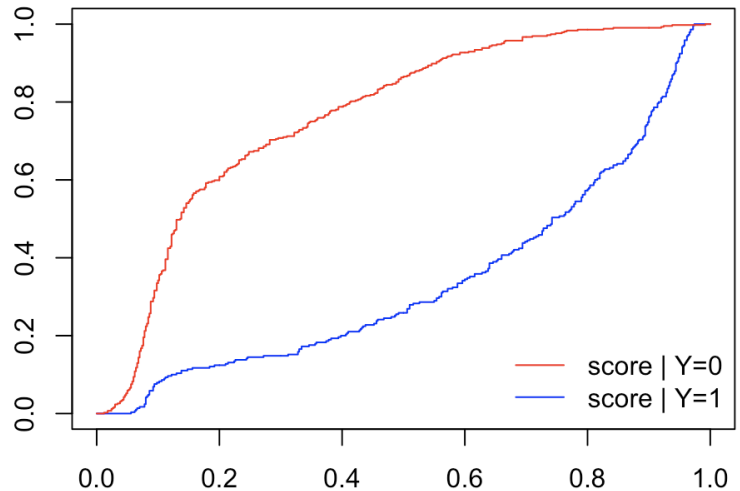
Kolmogorov-Smirnov (KS) :
Comparer les distributions de
$(S|Y=0)$ et $(S|Y=1)$

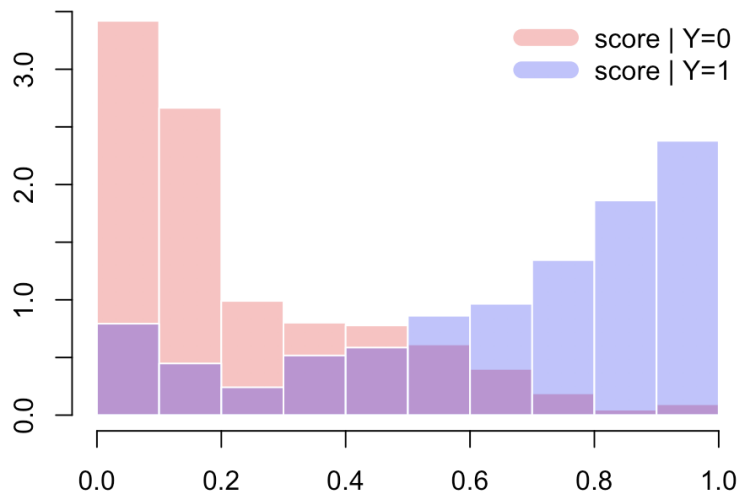
$$d = \sup_{x \in [0,1]} \{|\widehat{F}_1(x) - \widehat{F}_0(x)|\}$$

où

$$\widehat{F}_1(x) = \frac{1}{n_1} \sum_{i:y_i=1} \mathbf{1}(s_i \leq x)$$

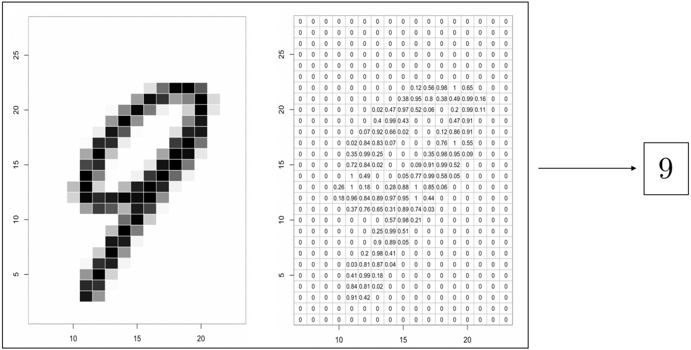
$$\widehat{F}_0(x) = \frac{1}{n_0} \sum_{i:y_i=0} \mathbf{1}(s_i \leq x)$$





Classification... sur des images

$(y_i, \boldsymbol{x}_i)$, où $y \in \{0, 1, 2, 3, \cdots, 9\}$ et $\boldsymbol{x}_i \in \mathcal{M}_{28,28} = [0, 1]^{28 \times 28}$

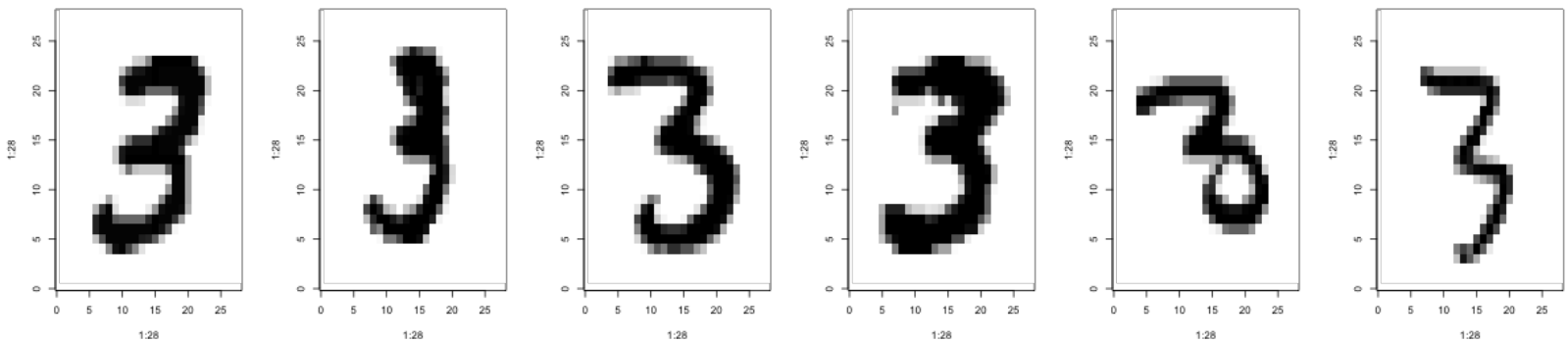


$\boldsymbol{x}_i \in \mathcal{M}_{28,28}$ $\qquad\qquad$ $y_i \in \{0, 1, \cdots, 9\}$

```
> library(keras)
> mnist = dataset_mnist()
```

Classification... sur des images

Here $\{(y_i, \boldsymbol{x}_i)\}$ with $y_i =$ "3" and $\boldsymbol{x}_i \in [0,1]^{28\times 28}$



```
> library(keras)
> mnist = dataset_mnist()
> n = 1000
> V = mnist$train$x[1:n,,]
> MV = NULL
for(i in 1:n) MV=cbind(MV,as.vector(V[i,,]))
> MV = t(MV)
> df = data.frame(y=mnist$train$y[1:n],x=MV)
```

Classification... sur des images

Peut-on reconnaître les '1' ?

```
> reg=glm((y==1)~.,data=df,family=binomial)
Warning messages:
1: glm.fit: algorithm did not converge
2: glm.fit: fitted probabilities numerically 0 or 1
    occurred
```

Problème numérique !

On a $k = 784$ variables explicatives... il faut réduire la dimension !

```
> library(factoextra)
> pca=prcomp(MV)
> res.ind = get_pca_ind(pca)
> PTS = res.ind$coord
> k=3
> dfpca = data.frame(y=mnist$train$y[1:n],x=PTS[,1:k])
> reg1 = glm((y==1)~.,data=dfpca,family=binomial)
> reg8 = glm((y==8)~.,data=dfpca,family=binomial)
```

Classification... sur des images

On essaye de reconnaître les '1' et les '8'

```
> library(ROCR)
> Y1 = as.numeric((df$y == 1)*1)
> S1 = predict(reg1,type="response")
> pred1 = prediction(S1,Y1)
> plot(performance(pred1,"tpr","fpr"))
> Y8 = as.numeric((df$y == 8)*1)
> S8 = predict(reg8,type="response")
> pred8 = prediction(S8,Y8)
> plot(performance(pred8,"tpr","fpr"))
```

On utilise ici seulement
les 3 premières composantes principales