

# INTRODUCTION AU BIG DATA

Synthèse des travaux effectués

LEFEVRE Clément

Encadrant : M. SOHIER

# Table des matières

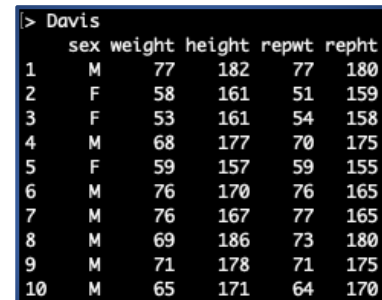
<b>TABLE DES MATIERES .....</b>	<b>1</b>
<b>REGRESSION LINEAIRE – DAVIS, POIDS ET TAILLE .....</b>	<b>2</b>
<b>1. CONTEXTE.....</b>	<b>2</b>
<b>2. PROBLEMATIQUE.....</b>	<b>2</b>
<b>3. TRAVAUX.....</b>	<b>2</b>
A. PREMIERE APPROCHE.....	2
B. CORRECTION DE LA VALEUR ERRONEE .....	3
C. AJUSTEMENT DE LA REGRESSION .....	5
D. DIFFERENCIATION DU SEXE.....	6
<b>ANALYSE EN COMPOSANTES PRINCIPALES - ALLOCINE.....</b>	<b>8</b>
<b>1. CONTEXTE.....</b>	<b>8</b>
<b>2. PROBLEMATIQUE.....</b>	<b>10</b>
<b>3. TRAVAUX.....</b>	<b>10</b>
A. Analyse en composantes principales – 1 <sup>ère</sup> approche .....	10
B. Expliquer le vote du public .....	12
C. ANALYSE EN COMPOSANTES PRINCIPALES - OUTILS AVANCES.....	12
<b>CONCLUSION .....</b>	<b>17</b>

# Régression linéaire — Davis, poids et taille

## 1. Contexte

Les données manipulées sont issues de la librairie *car*, et sont accessibles via la commande *Davis*. Ces données représentent 200 individus (lignes), et 3 variables : le **sexe** (*sex*), le **poids** (*weight*), la **taille** (*height*), le **poids reporté** (*repwt*), et la **taille reportée** (*repht*).

Dans un premier temps, nous avons découvert quelques fonctions de manipulation de *R*, parmi lesquelles la fonction *lm*, qui permet de construire un modèle linéaire.



	sex	weight	height	repwt	repht
1	M	77	182	77	180
2	F	58	161	51	159
3	F	53	161	54	158
4	M	68	177	70	175
5	F	59	157	59	155
6	M	76	170	76	165
7	M	76	167	77	165
8	M	69	186	73	180
9	M	71	178	71	175
10	M	65	171	64	170

Figure 1 : 10 premières valeurs du jeu de données

## 2. Problématique

On cherche à déterminer une **relation entre le poids et la taille** d'une personne. En raisonnant de manière logique, il apparaît évident de penser que **la taille permet d'expliquer le poids**, plus que l'inverse.

On veut donc **expliquer le poids d'individus par leur taille**, ce qui signifie que la **variable explicative** est la **taille** et la **variable expliquée** est le **poids**.

## 3. Travaux

### A. Première approche

On peut donc utiliser la fonction *lm*, mentionnée précédemment, de la manière suivante :

```
> lm(Davis$weight ~ Davis$height)

Call:
lm(formula = Davis$weight ~ Davis$height)

Coefficients:
(Intercept)  Davis$height
    25.2662         0.2384
```

Le caractère ~ signifie ici simplement 'est expliqué par'.

On obtient donc deux coefficients :

- Intercept, qui correspond à la partie constante de la régression.
- Davis\$height, qui correspond au coefficient entre le poids et la taille.

Ainsi, si l'on explique le poids par la taille, à ce stade, on a :

$$Davis\$weight = 0,2384 * Davis\$height + 25,2662 + \epsilon$$

On remarquera également que les valeurs correspondant au  $\epsilon$  de la régression sont bien contenus dans l'objet obtenu, et correspondent au champ *residuals*.

La fonction *summary* nous permet d'obtenir plus d'informations au sujet de cette régression, et notamment le coefficient de détermination ( $R^2$ ), qui mesure la qualité de prédiction d'une régression linéaire.

Ici, pour cette première régression, on a donc :  $R^2 = 0,03597$ , ce qui est relativement faible.

En effet, cela signifie que **l'équation de la droite de régression ne détermine qu'environ 3,6% de la distribution de points**.

Pour mieux observer cela, on peut dans un premier temps afficher nos points, à l'aide de la fonction *plot*.

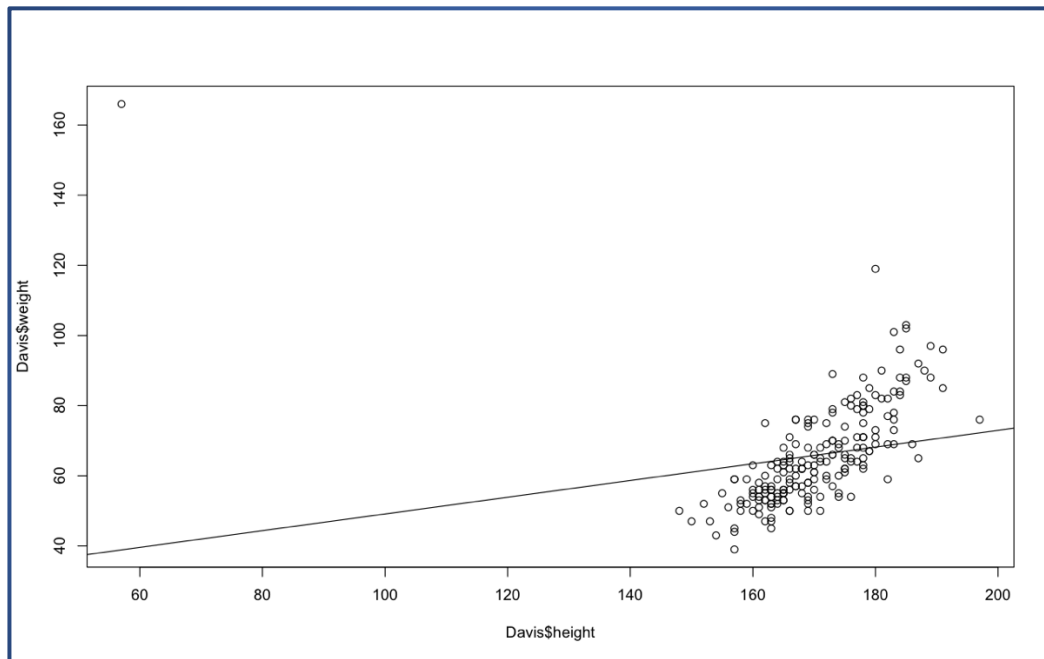


Figure 2 : Graphique représentant la régression linéaire du poids en fonction de la taille

Rapidement, on remarque un **individu isolé**, dont les **données semblent erronées** (moins de 60cm pour plus de 160kg).

En recherchant dans la liste de données, on trouve l'individu correspondant : il s'agit du **12<sup>ème</sup>**.

### B. Correction de la valeur erronée

Si l'on analyse les autres données fournies par cet individu (notamment son poids et sa taille estimée), on remarque que celui-ci a **inversé la taille et le poids**.

Ici, on modifie cela manuellement (voir ci-contre), mais de manière générale, il conviendrait de **prévenir ce genre d'erreur**, et de **mettre en place des algorithmes/programmes de correction**, du moins pour les cas les plus courants.

```
> Davis$weight<-57
> Davis$height<-166
```

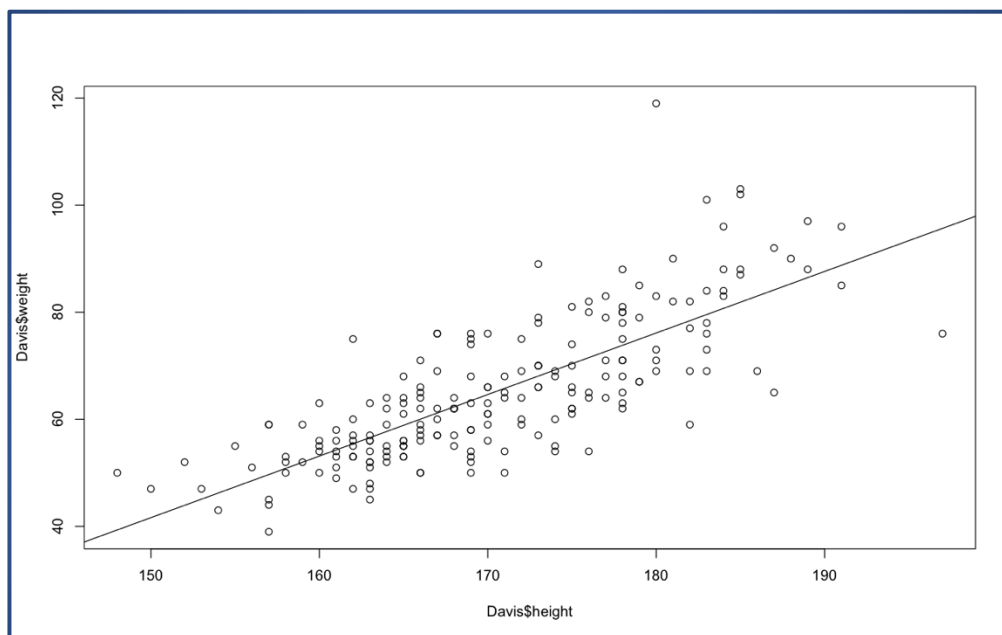


Figure 3 : Graphique représentant la régression linéaire du poids en fonction de la taille, après correction

Les données sont désormais plus cohérentes, on peut donc relancer la régression linéaire, et on obtient :

$$Davis\$weight = 1,15 * Davis\$height - 130,91 + \epsilon$$

La qualité de la régression se trouve logiquement améliorée, on a désormais  $R^2 = 0,5946$ , ce qui signifie que l'équation de la droite de régression détermine environ 59,5% de la distribution de points.

On constate que l'augmentation du coefficient de détermination apportée par la correction est considérable, il faut donc toujours **veiller à la cohérence des données**, avant d'effectuer une régression linéaire.

Si on s'intéresse aux résidus (i.e.  $\epsilon$ , informations que l'on ne peut expliquer par la régression, dont on sait qu'ils sont stockés dans le champ `reg$residuals`), il peut être intéressant de les recalculer à partir de la formule de base :  $Y = aX + b + \epsilon$ , d'où il vient  $\epsilon = Y - aX - b$ .

Or ici, on a :

- $a = \text{reg\$coefficients}[2]$
- $b = \text{reg\$coefficients}[1]$
- $X = \text{Davis\$height}$
- $Y = \text{Davis\$weight}$

En remplaçant dans la formule, il vient la commande suivante :

```
Davis$weight-reg$coefficients[2]*Davis$height-reg$coefficients[1]
```

Pour vérifier que cette commande est équivalente aux résidus, on peut les comparer, en calculant l'écart maximal, de la manière suivante :

```
> max(abs(reg$residuals - (Davis$weight-reg$coefficients[2]*Davis$height-reg$coefficients[1])))
[1] 1.228129e-12
```

Cet écart est de l'ordre de  $10^{-12}$ , ce qui est très faible, et négligeable devant `Davis$weight`, les résultats sont équivalents.

De la même manière que pour les résidus, on peut également recalculer de manière formelle le coefficient de détermination  $R^2$ .

$$R^2 = \frac{V(aX + b)}{V(Y)} = a^2 \frac{V(X)}{V(Y)} = 1 - \frac{V(\epsilon)}{V(Y)}$$

En remplaçant par les variables correspondantes dans  $R$ , il vient :

```
> 1 - var(reg$residuals)/var(Davis$weight)
[1] 0.5945555
```

Cette valeur est équivalente au *Multiple R-squared* de la commande *summary*, toutefois, cette dernière peut être erronée, c'est pourquoi il est préférable de la recalculer manuellement, comme on vient de le faire.

Si l'on revient à notre régression linéaire, du poids expliqué par la taille, on avait obtenu la formule suivante :  $Davis\$weight = 1,15 * Davis\$height - 130,91 + \epsilon$ , qui nous semblait relativement satisfaisante, au vu du coefficient  $R^2$  (d'environ 59,46%).

Néanmoins, le coefficient  $b$  (ou (Intercept)) dans  $R$  est trop élevé, et est surtout négatif, ce qui est insensé, notamment pour des personnes de petite taille, qui ne sont donc pas couvertes par la régression (une personne de moins de 110 cm posséderait un poids négatif...).

### C. Ajustement de la régression

Lors de la première régression, nous avons voulu expliquer le poids par la taille. Cependant, on s'aperçoit que le poids est plus susceptible d'être expliqué par un volume (en  $cm^3$  donc), que par une hauteur (en  $cm$ ).

On va donc mettre en place une nouvelle régression, dans laquelle on tentera d'expliquer le poids (Davis\$weight) par la taille au cube (Davis\$height<sup>3</sup>).

On obtient :

$$Davis\$weight = 1,3 \times 10^{-5} * Davis\$height^3 + 0,185 + \epsilon$$

On remarque immédiatement que le coefficient  $b$ , constant, est désormais bien plus faible.

On a également  $R^2 = 0,5979$ , ce qui est légèrement mieux que la régression précédente.

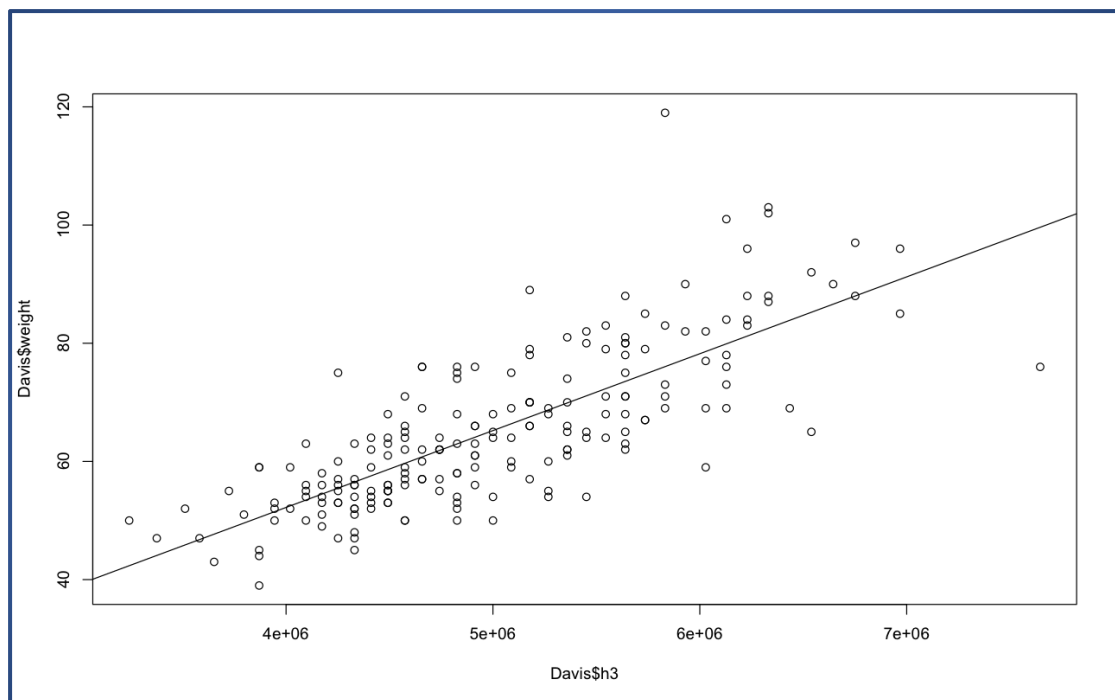


Figure 4 : Graphique représentant la régression linéaire du poids en fonction de la taille au cube

En poursuivant dans la même logique, de réduire l'impact du coefficient constant  $b$ , on peut également fixer sa valeur, via la commande `lm`, de la manière suivante :

```
> reg3=lm(Davis$height~Davis$h3+0)
```

On obtient alors :

$$Davis\$weight = 3,354 \times 10^{-5} * Davis\$height^3 + \epsilon$$

$$R^2 = 0,5979$$

### D. Différenciation du sexe

Si on reprend la courbe précédente, en mettant en évidence le sexe (les hommes correspondent aux points rouges, et les femmes aux points noirs), on remarque que ce facteur joue un rôle dans la relation entre le poids et la taille au cube d'une personne.

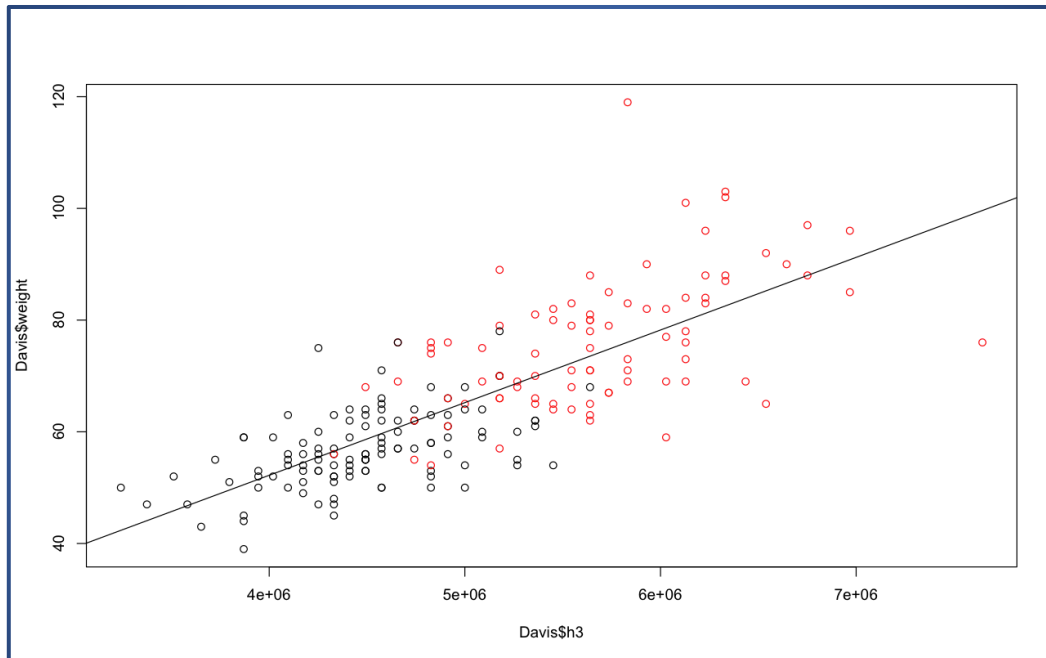


Figure 5 : Poids en fonction de la taille au cube, avec mise en évidence du sexe

D'après cette courbe, il apparaît que les femmes sont en moyenne plus petite et légères que les hommes.

Afin d'améliorer la régression, il paraît donc judicieux de **séparer les hommes des femmes**.

Il serait alors possible de créer deux tableaux de données distincts *Dh* pour les hommes et *Df* pour les femmes, de la manière suivante :

```
Dh<-Davis[Davis$sex == "M",]
Df<-Davis[Davis$sex == "F",]
```

Ainsi, on pourrait par la suite réitérer les manipulations précédentes, pour obtenir les deux régressions distinctes.

Néanmoins, cette séparation homme/femme peut s'effectuer d'une autre manière, plus simple à appréhender :

```
reg3 = lm(Davis$weight~Davis$h3:Davis$sex+0)
```

Le fait de rajouter les ' : ' permet de calculer la régression expliquant le poids par la taille au cube, **suivant le sexe**.

En appelant la fonction *summary*, on obtient le résultat suivant :

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
Davis$h3:Davis$sexF 1.264e-05  1.732e-07  72.93  <2e-16 ***
Davis$h3:Davis$sexM 1.337e-05  1.547e-07  86.42  <2e-16 ***
```

- Pour les hommes, on a donc :

$$\text{Davis\$weight} = 1,264 \times 10^{-5} * \text{Davis\$height}^3 + \epsilon$$

- Et pour les femmes :

$$Davis\$weight = 1,337 \times 10^{-5} * Davis\$height^3 + \epsilon$$

On trouve également  $R^2 \approx 0.617$ , ce qui montre que cette régression explique un plus fort pourcentage de valeurs, et donc qu'elle est meilleure.

```
> 1 - (var(reg3$residuals) / var(Davis$weight))
[1] 0.6173972
```

A l'aide de la fonction *lines*, on peut tracer la régression, en prenant en abscisses uniquement les points que l'on veut joindre par une ligne polygonale (ici entre 140 et 200cm).

```
> lines(c(140:200), reg3$coefficients[2]*c(140:200)^3, col="red")
> lines(c(140:200), reg3$coefficients[1]*c(140:200)^3)
```

On obtient le résultat suivant :

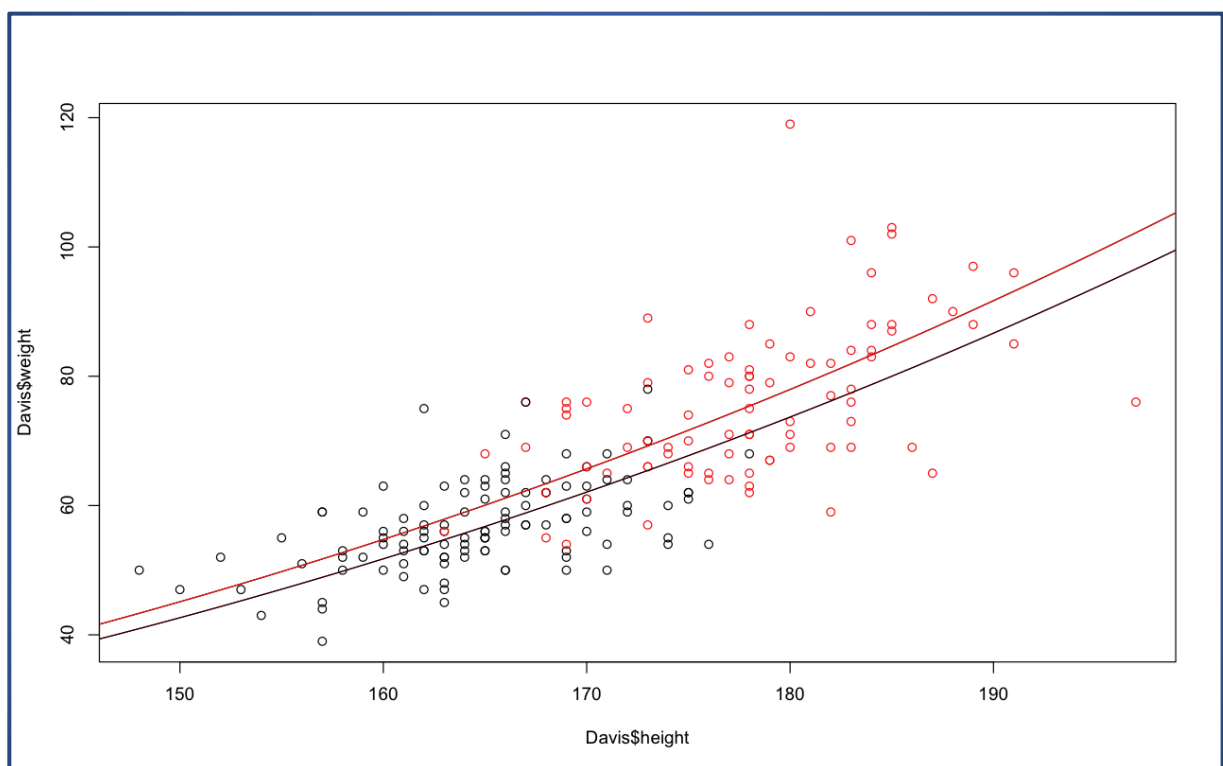


Figure 6 : Poids en fonction de la taille au cube, selon le sexe



# Analyse en composantes principales - Allociné

## 1. Contexte

Les données manipulées sont issues du site *allociné.fr*, fournissant des informations cinématographiques en ligne, appartenant à la société Webmedia.

On s'intéresse ici aux notes de différents films, données par différents médias, ainsi que le public. Les données ont été extraites vers un fichier CSV, à partir duquel nous allons importer les données dans R, via le script suivant :

```
a<-scan("allocine2019f.csv", what=character(), sep="\n")
lfilm <- lapply(a, strsplit, ";")
dfilm <- data.frame()
for(f in lfilm){
  film <- f[[1]];
  print(film[2])
  for(i in c(3:length(film))){
    if(i%%2==1){
      if(is.na(as.numeric(film[i+1]))){
        if(is.null(dfilm[film[2], film[i]])||is.na(dfilm[film[2], film[i]])){
          dfilm[film[2], film[i]] <- film[i+1]
        }
      }
    } else{
      dfilm[film[2], film[i]] <- as.numeric(film[i+1])
    }
  }
}
```

Figure 7 : Script d'import des données depuis le fichier CSV dans R

Une fois l'import effectué, on obtient nos données sous la forme suivante :

	h	min	jour	mois	annee	genre	nationalite
La Valle des loups	1	30	4	janvier	2017	Documentaire	franais
Faut pas lui dire	1	36	4	janvier	2017	Comdie	belge
Nocturnal Animals	1	57	4	janvier	2017	Drame	amricain
Quelques minutes aprs minuit	1	48	4	janvier	2017	Fantastique	amricain
Primaire	1	45	4	janvier	2017	Drame	franais
Mes Trsors	1	35	4	janvier	2017	Comdie	franais
	aVoir-aLire.com La.Croix Le.Dauphin.Lib Le.Figaro						
La Valle des loups		40			40		40
Faut pas lui dire		NA			NA		20
Nocturnal Animals		40			20		10
Quelques minutes aprs minuit		40			NA		20
Primaire		40			30		40
Mes Trsors		NA			NA		30
	Le.Nouvel.Observateur Le.Parisien Ouest.France						
La Valle des loups					40		40
Faut pas lui dire					NA		30
Nocturnal Animals					20		10
Quelques minutes aprs minuit					40		20
Primaire					40		40
Mes Trsors					NA		20
	TLrama VSD Le.Monde Les.Fiches.du.Cinma						
La Valle des loups		40	40		30		30
Faut pas lui dire		NA	NA		NA		NA
Nocturnal Animals		30	10		20		30
Quelques minutes aprs minuit		30	20		30		30
Primaire		30	NA		20		30
Mes Trsors		NA	NA		NA		20
	Studio.Cin.Live L.Express n5 n4 n3 n2 n1 n0						
La Valle des loups			30		20	19	13
Faut pas lui dire			20		NA	42	24
Nocturnal Animals			30		10	42	142
Quelques minutes aprs minuit			30		20	41	89
Primaire			40		40	14	35
Mes Trsors			NA		NA	4	7
						24	11
						14	5

Initialement, les notes du public étaient présentées de la manière suivante : une colonne n5, correspondant à une note de 5 étoiles, une colonne n4 correspondant à une note de 4 étoiles, etc.

Ceci est **assez contraignant**, il serait en effet préférable de n'avoir **qu'une colonne nommée public**, contenant **la moyenne des notes**.

On commence donc par effectuer cette manipulation :

```
> dfilm$public<-(dfilm$n5*5+dfilm$n4*4+dfilm$n3*3+dfilm$n2*2+dfilm$n1*1+dfilm$n0*0)/
(dfilm$n5+dfilm$n4+dfilm$n3+dfilm$n2+dfilm$n1+dfilm$n0)
> dfilm$public
 [1] 4.046512 3.151724 3.258856 3.757732 3.333333 2.400000 2.822222 3.818182
 [9] 2.941176 4.063158 3.333333 2.666667 2.875000 4.020833      NA 2.808399
[17] 3.619048 3.053476 2.552083 2.514706 3.129496 3.065217 3.225000 3.741935
[25] 3.458333 2.428571 2.760000 3.000000 3.500000 5.000000 3.000000 3.000000
[33] 3.714286      NA 2.283951 3.740506 3.262500 2.973451 3.000000 3.355263
[41] 2.833333 4.526316 2.166667      NA      NA 3.588235 3.500000      NA
[49] 3.738376 3.792105 2.257732 3.763975 4.084772 2.722222 2.449438 3.490196
[57] 4.104167 3.200000 4.215909 2.571429 3.600000 3.666667 4.400000 4.117647
[65] 3.886364 3.902439 4.454545 4.366667 4.500000 3.000000 3.869565 2.625000
[73] 3.600000      NA 2.759582 3.309302 3.200000 4.619048 1.395522 2.722807
[81] 3.051948 3.388889 3.500000 4.117647 3.066667 2.500000 3.375000 4.000000
[89]      NA 2.451306 2.428135 3.019417 3.234177 3.482143 3.000000 3.887500
[97] 3.250000 3.068966 2.600000      NA 3.000000 3.714286      NA 3.400000
[105] 3.595923 2.668823 2.841935 3.242775 2.183333 2.523490 4.631579 3.204545
```

Il serait également intéressant d'avoir une colonne recensant la variance des notes du public. Or, on sait que par définition, la variance est la somme des carrés des écarts à la moyenne, il vient donc :

```
> dfilm$varpub<-((5-dfilm$public)^2*dfilm$n5 + (4-dfilm$public)^2*dfilm$n4 + (3-dfilm$
public)^2*dfilm$n3 + (2-dfilm$public)^2*dfilm$n2 + (1-dfilm$public)^2*dfilm$n1 + (0-df
ilm$public)^2*dfilm$n0)/(dfilm$n5+dfilm$n4+dfilm$n3+dfilm$n2+dfilm$n1+dfilm$n0)
> dfilm$varpub
 [1] 1.30016225 2.59766944 1.50520087 0.91553300 1.37373737 1.71692308
 [7] 1.36839506 1.42148760 1.58477509 0.92232687 0.22222222 4.22222222
[13] 1.35937500 1.18706597      NA 1.52496883 1.31916100 1.07735423
[19] 2.10145399 2.30860727 1.47963356 1.23487713 0.87437500 2.06243496
[25] 1.28993056 1.31632653 1.14240000 0.00000000 0.25000000 0.00000000
[31] 1.62500000 1.00000000 0.20408163      NA 2.37616217 1.36937190
[37] 1.24984375 0.90195004 2.29166667 1.91326177 1.23412698 0.35457064
[43] 1.13888889      NA      NA 0.71280277 2.25000000      NA
[49] 1.88644539 0.93309557 2.26347114 0.92565873 1.49359857 1.31172840
[55] 1.68564575 1.19108035 0.63498264 0.56000000 0.91929236 1.67346939
[61] 1.94000000 0.88888889 0.81777778 0.80968858 0.73708678 1.06365259
[67] 1.42975207 0.56555556 0.25000000 1.20000000 0.98298677 0.73437500
```

Autre particularité des données fournies, celles-ci sont **nombreuses à posséder des valeurs non-renseignées (NA)**, il serait donc intéressant de faire en sorte de trier les données selon leur taux de renseignement (i.e. **avoir les lignes et les colonnes les plus remplies d'abord**).

Pour cela, on utilise la fonction `is.na`, qui indique si une valeur vaut NA ou non (sous la forme d'un tableau de booléens).

On souhaite donc ici **trier les films par ordre de NA décroissant** (c'est à dire par **ordre de critiques croissant**).

A noter que l'on s'intéresse uniquement aux films les plus récents (sortis après 2015).

Ceci nous amène donc à utiliser la formule suivante :

```
> sort(apply(is.na(dfilm[dfilm$annee>2015,]),1,sum))
                                Les Freres Sisters
                                                73
                                La La Land
                                                74
                                Loving
                                                74
                                Valrian et la Cit des mille plantes
                                                77
                                Blade Runner 2049
                                                77
                                The Square
                                                78
                                Star Wars - Les Derniers Jedi
                                                78
                                Le Redoutable
                                                79
                                Detroit
                                                79
                                Le Grand Bain
                                                79
                                Amanda
                                                79
                                Silence
                                                80
                                Pentagon Papers
                                                80
                                La Forme de l'eau - The Shape of Water
                                                80
```

Nos données sont donc désormais relativement mieux organisées et présentées.

## 2. Problématique

On cherche ici à effectuer une analyse en composantes principales, qui permettrait de comprendre quels sont les principaux facteurs orientant la critique cinématographique.

## 3. Travaux

### A. Analyse en composantes principales – 1<sup>ère</sup> approche

En R, la fonction *prcomp* permet de réaliser une analyse en composantes principales. Avant de commencer à la réaliser, il convient de distinguer deux colonnes particulières, qui doivent être considérées comme facteurs de l'analyse en composante principale : le genre et la nationalité.

```
dfs$nationalite<-factor(dfs$nationalite)
dfs$genre<-factor(dfs$genre)
```

On peut maintenant définir l'analyse en composante principale :

```
acp1<-prcomp(~.,dfs[,c(17:22),(25:33)])
```

On prend seulement une dizaine de variables, car en prendre plus pourrait nuire aux observations.

L'objet obtenu possède les champs suivants :

```
> acp1$
acp1$call      acp1$na.action  acp1$scale      acp1$x
acp1$center    acp1$rotation  acp1$sdev
```

Parmi eux, on retrouve les trois champs principaux de l'analyse en composante principale :

- La matrice de rotation (**acp1\$rotation**)
- L'écart type (**acp1\$sdev**)
- Film et leur score, en composantes principales (**acp1\$x**)

Grâce à la fonction *biplot*, on peut obtenir une représentation graphique de cette analyse en composantes principales :

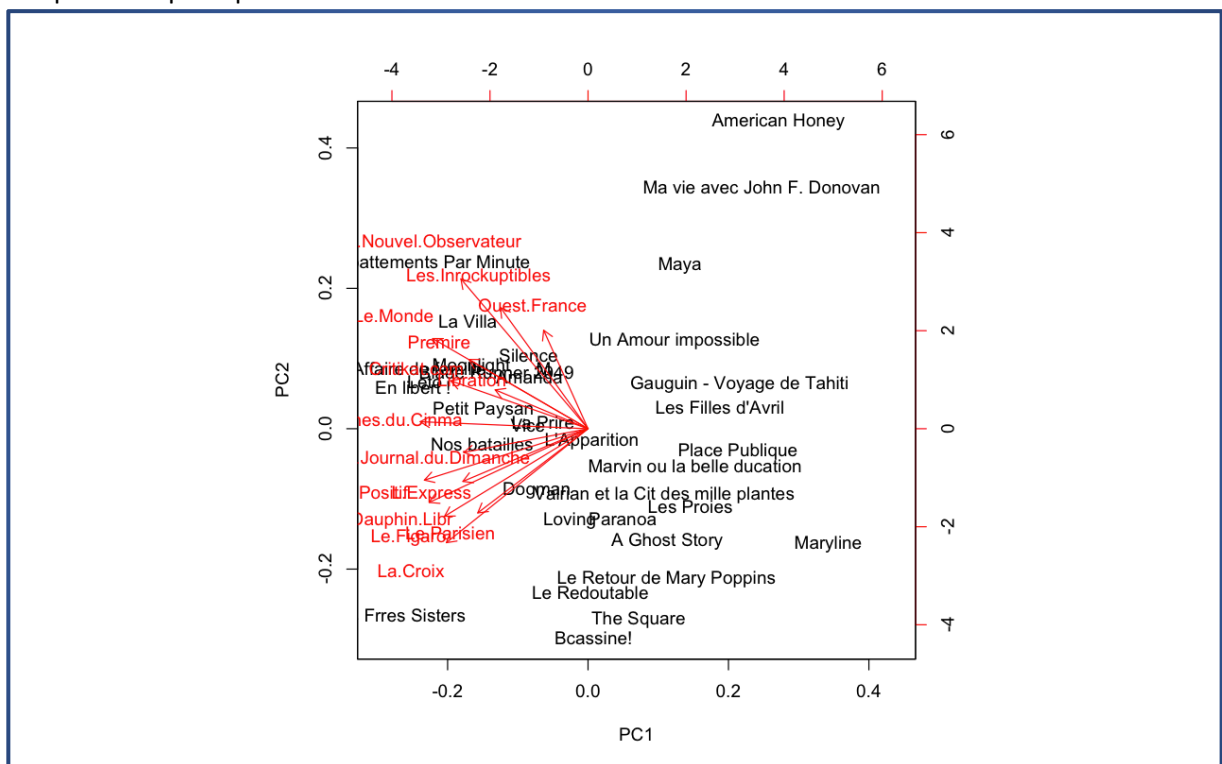


Figure 8 : Représentation graphique de l'analyse en composantes principales

On remarque donc **deux axes, PC1 et PC2** (pour Composante Principale 1, et 2) :

- Sur le premier axe (PC1), on **remarque que les critiques sont relativement en accord**, puisqu'elles sont **orientées dans la même direction** (i.e. vers la gauche). On peut donc penser que les différents médias sont en **accord sur un point de vue artistique, qualitatif**.
- En revanche, **le second axe divise les médias**. On remarque que les **médias aux opinions politiques tendant à gauche sont orientés vers le haut** (Le Nouvel Observateur, Ouest France, etc.) et ceux ayant des **opinions politiques tendant à droite sont orientés vers le bas** (le Journal du Dimanche, l'Express, etc.).

## B. Expliquer le vote du public

On souhaite désormais s'intéresser au **comportement du public, vis à vis des critiques des médias**. Pour calculer cela, on va utiliser la fonction *cor*, qui permet de calculer une corrélation.

```
> acp1<-prcomp(~,dfs[,c(17:22,25:33)], scale=T)
>
> cor(dfs[rownames(acp1$x),"public"],acp1$x)
      PC1      PC2      PC3      PC4      PC5      PC6      PC7
[1,] -0.4231483 0.3844521 0.1529491 -0.1418836 0.2899771 -0.1620988 0.3065005
      PC8      PC9      PC10     PC11     PC12     PC13     PC14
[1,] 0.06226567 0.1615494 0.1091012 -0.0914881 0.07182766 0.09243643 0.05488824
      PC15
[1,] 0.2407221
```

Pour la première composante, on observe un coefficient de corrélation d'environ  $-0,42$ , ce qui montre que la note du public est relativement peu expliquée par les critiques effectuées par les médias.

La deuxième composante nous indique, elle, que le public semble en accord avec les journaux partageant des opinions politiques plutôt de droite.

## C. Analyse en composantes principales - avancée

Pour la suite des travaux pratiques, nous avons utilisés des **outils plus poussés** afin de mettre en œuvre notre analyse en composantes principale.

Une des limites de la précédente analyse en composantes principales résidait dans les valeurs 'NA', relativement nombreuses, qui la faussaient légèrement.

Pour pallier à cela, nous allons utiliser deux nouvelles librairies :

- *FactoMineR*, qui est dédiée à **l'analyse exploratoire des données**, et
- *MissMDA*, qui vient comme un complément de la précédente librairie, et qui permet de **gérer les données manquantes pour les méthodes d'analyses factorielles** (telles que l'analyse en composantes principales).

De cette manière, les valeurs 'NA' seront remplacées par des valeurs 'plausibles', ce qui permet de travailler sur une plus grande quantité d'information, et donc d'être plus pertinent.

Durant l'analyse de nos données, nous avons pu remarquer que les **films les plus représentés**, et donc, **possédant le plus d'avis**, étaient les **films français et américains**.

Il nous a donc semblé judicieux de **distinguer les nationalités des films selon trois catégories** : **Français, Américain, ou Autre**, pour toute autre nationalité.

La commande suivante nous permet de réaliser cela de manière assez simple, sous la forme d'une expression conditionnelle *if else*, qui affectera donc « autre », dans la colonne *nat2*, pour chaque film de nationalité ni française, ni américaine.

```
> dfs$nat2<-ifelse(dfs$nationalite == "franais" | dfs$nationalite == "amricain",dfs$nationalite,"autre")
```

On obtient ainsi le résultat suivant :

```
dfs$nat2
[1] "29" "4" "4" "29" "4" "autre" "4" "29" "4"
[10] "29" "29" "4" "4" "4" "autre" "29" "autre" "4"
[19] "29" "autre" "4" "4" "4" "29" "4" "4" "4"
[28] "4" "29" "autre" "4" "autre" "29" "4" "4" "autre"
[37] "autre" "4" "autre" "29" "4" "29" "4" "29" "4"
[46] "29" "29" "4" "29" "autre" "29" "4" "4" "4"
[55] "autre" "29" "29" "autre" "4" "4" "29" "29" "autre"
[64] "4" "29" "29" "4" "29" "autre" "29" "29" "autre"
[73] "4" "autre" "autre" "4" "29" "29" "29" "4" "29"
```

Les valeurs **4** et **29** correspondent respectivement aux **nationalités américaines et françaises**. Ces deux nationalités ont été représentées sous cette forme car nous avons défini la colonne nationalité comme facteur (factor).

Désormais, comme mentionné précédemment, nous allons **remplacer les valeurs 'NA' par des valeurs 'pertinentes'**, grâce à la fonction `impute_FAMD`.

On va donc sélectionner quelques colonnes, en prenant des **titres représentatifs** :

- **Presse régionale** (e.g. Ouest France, La Voix du Nord, ...),
- **Spécialistes du cinéma** (e.g. Les fiches du cinéma, Cahiers du cinéma, ...),
- **Sites internet** (e.g. Culturopoing.com, Critikat.com, ...),
- **Magazines** (Les Inrockuptibles, Femme Actuelle, ...),
- **Presse nationale** (e.g. Libération, La Croix, ...),
- Etc.

```
res.comp<-imputeFAMD(dfs[,c(17,38,42,52,28,19,29,22,20,33,26,48,27,41,54,46,58,13,24,136)])
```

On peut ensuite refaire notre analyse en composante principales, cette fois grâce à la fonction `PCA`.

```
res2<-PCA(tab,quali.sup=c(19,20),quanti.sup=18)
```

En plus de notre tableau de données, on a passé deux autres paramètres à la fonction :

- `quali.sup`, qui correspond aux **variables qualitatives** (ici le genre et la nationalité), ces variables doivent être distinguées des autres.
- `quanti.sup`, qui correspond à la **variable quantitative**, ici le vote du public.

On obtient deux représentations graphiques : un **cercle de corrélation** et un **nuage de points**.

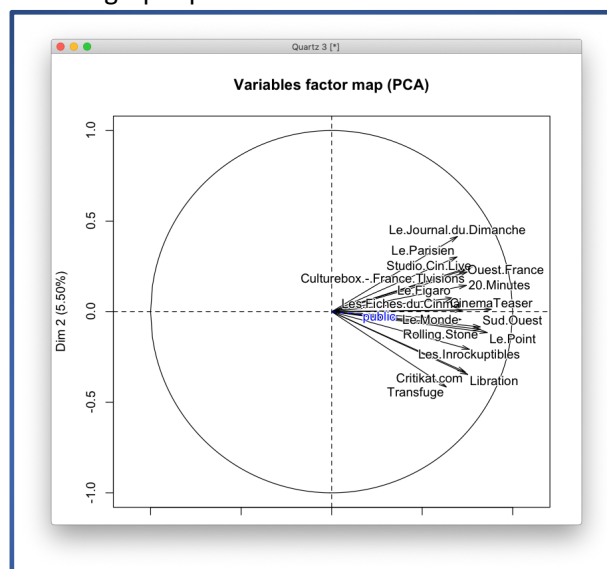


Figure 9 : Cercle de corrélation obtenu



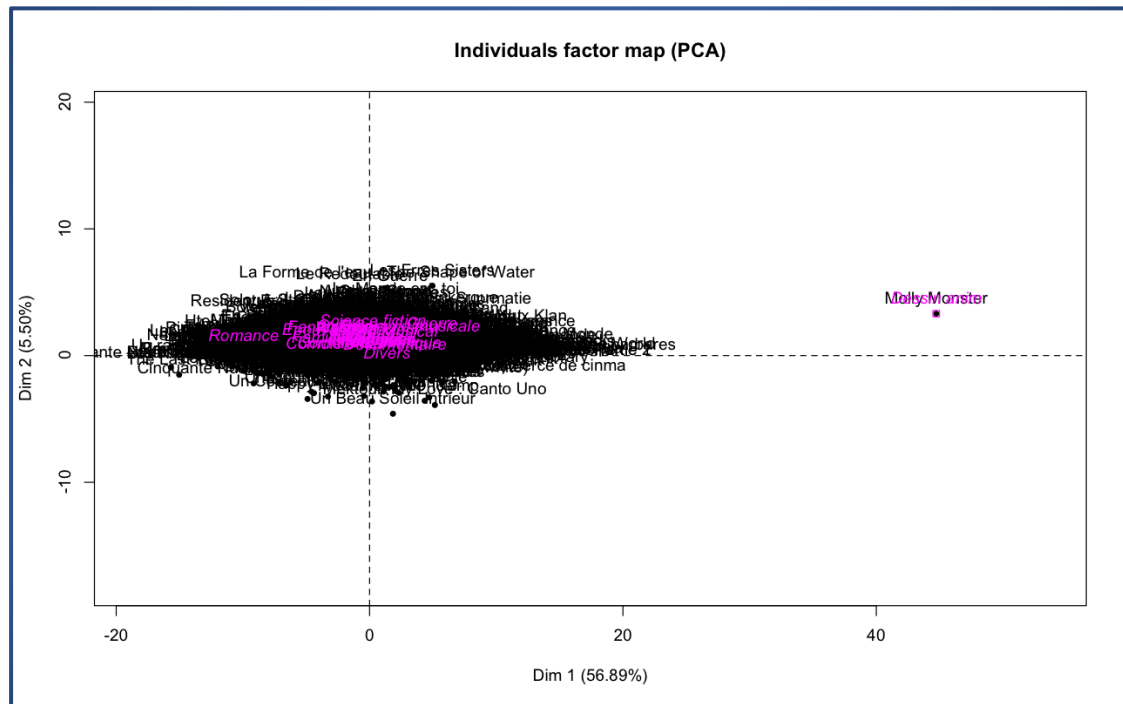


Figure 10 : Nuage de points obtenu

Rapidement, on remarque un cas similaire à celui observé dans notre précédente étude (explication du poids par la taille), puisqu'un individu est anormalement positionné sur le graphique.

Il s'agit ici du film Molly Monster, et, comme dans le cas précédent, afin de ne pas fausser notre analyse en composantes principales, nous allons supprimer ce film, et le remettre après notre analyse.

Avec la fonction `PCA`, on peut réaliser cela en fournissant le paramètre `ind.sup`. On cherche donc le numéro de ligne correspondant, et on relance l'analyse en composantes principales.

```
> grep("Molly Monster",rownames(tab))
[1] 1277
> res2<-PCA(tab,quali.sup=c(19,20),quanti.sup=18,ind.sup=(1277))
```

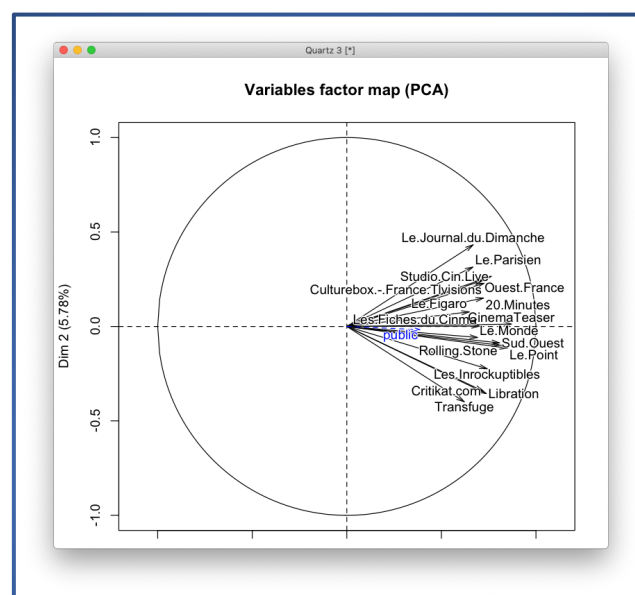
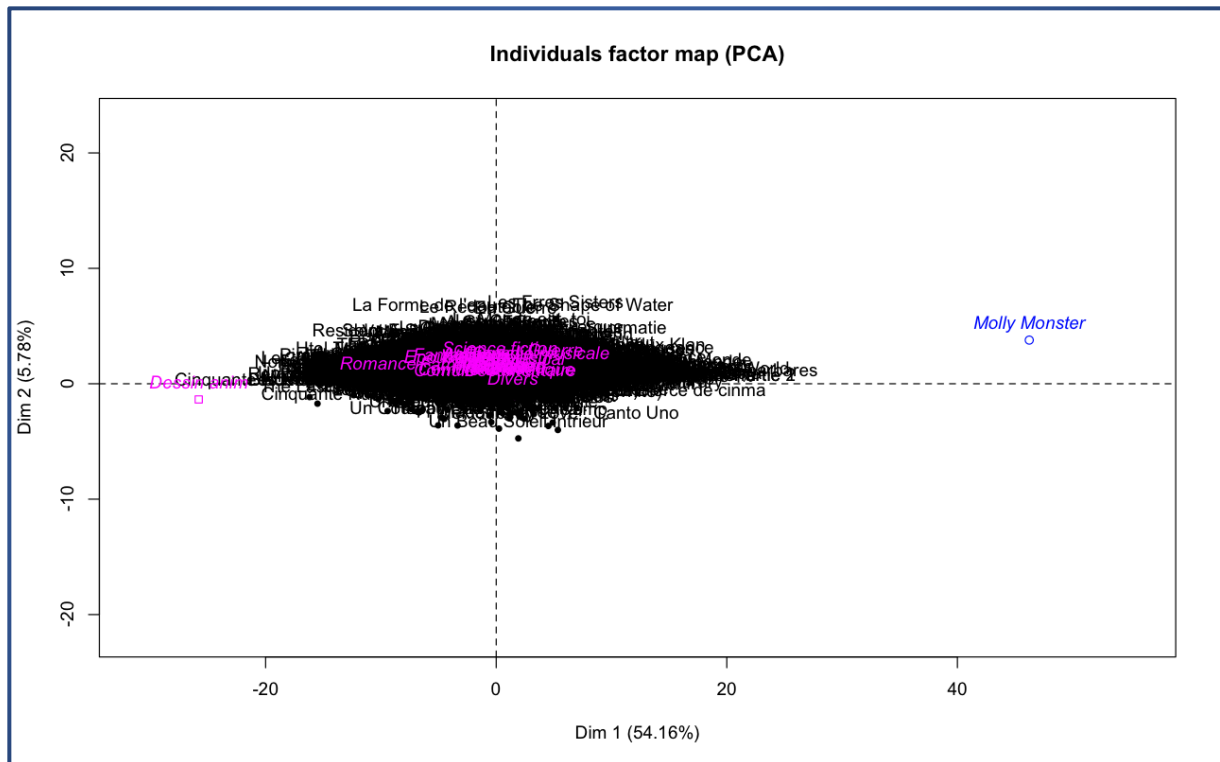


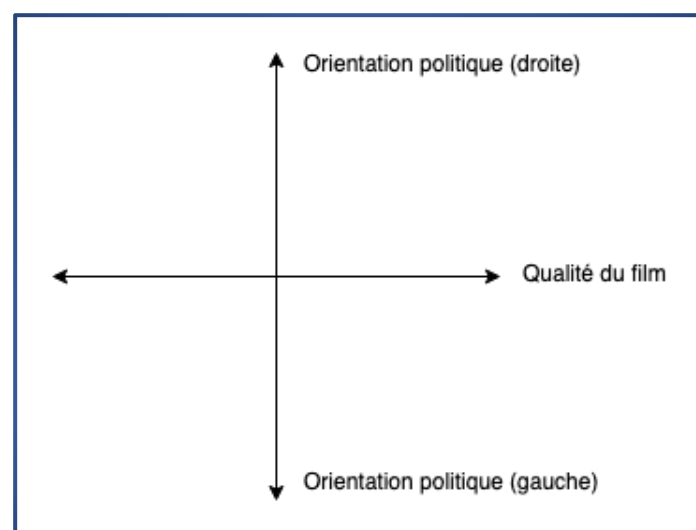
Figure 11 : Nouveau cercle de corrélations



On remarque que le film Molly Monster apparaît désormais en bleu, ce qui signifie qu'il n'a pas été pris en compte pour l'analyse.

Si l'on regarde le cercle de corrélations et le nuage de points, on remarque également que les résultats sont semblables à ceux observés lors de l'analyse précédente, puisque l'on retrouve deux axes :

- Le **premier (abscisses)**, sur lequel l'ensemble des critiques sont en accord (la **qualité du film**)
- Le **second (ordonnées)**, qui témoigne de l'**orientation politiques des critiques**, avec, vers le haut, des médias orientés à droite, et vers le bas, des médias orientés à gauche.



On remarque également que des **coefficients** sont associés aux deux axes : **54.16% pour le premier**, et **5.78% pour le second**.



On peut **ensuite déterminer les meilleurs, et pires films selon la critique**. En effet, comme on a déterminé que la qualité du film correspondait au premier axe, on peut simplement effectuer les commandes suivantes :

```
> head(sort(res2$ind$coord[,1]),n=10)
Cinquante Nuances plus sombres          The Last Face
-16.187339                               -15.498908
      Seule la vie...      Un raccourci dans le temps
-12.356997                               -11.245353
      # Pire soire          All Inclusive
-11.174635                               -10.713047
      Un homme press       Les Petits Flocons
-10.604688                               -10.216366
      Eiffel I'm in Love 2 Cinquante Nuances plus claires
-9.779250                                -9.415969
```

Figure 12 : Les 10 pires films, selon la critique

```
> tail(sort(res2$ind$coord[,1]),n=10)
      Une Affaire de famille
      7.738445
      Lumière ! L'aventure commence
      7.739062
      La La Land
      7.976587
      120 battements Par Minute
      7.982879
      Flicit
      8.076910
      Makala
      8.188999
      Les mes mortes - Partie 1
      8.352272
      Les mes mortes - Partie 2
      8.384294
      Les mes mortes - Partie 3
      8.384294
      Rumble: The Indians Who Rocked The World
      9.795607
```

Figure 13 : Les 10 meilleurs films, selon la critique

On pourrait également effectuer des classements plus poussés, en regardant par exemple, les films appréciés par la critique de droite, en se basant sur la deuxième dimension cette fois.

On peut aussi **évaluer les variables qualitatives**, notamment les **genres/nationalités de films les plus qualitatifs**, selon la critique :

```
> sort(res2$quali.sup$coord[,1])
Dessin anim      Romance      Comdie      Famille
-2.580064e+01    -1.024276e+01    -4.499483e+00    -4.152869e+00
Fantastique      Action      Western      Epouvante-horreur
-3.159808e+00    -2.541860e+00    -1.901942e+00    -1.489678e+00
      nat2 4      Thriller      Biopic      Aventure
-1.175673e+00    -1.127522e+00    -5.724568e-01    -5.599552e-01
Historique      nat2 29      Comdie dramatique      Science fiction
-2.728197e-01    -5.353784e-02    1.996065e-04    2.914195e-01
      Drame      nat2 autre      Animation      Divers
3.704164e-01    7.226048e-01    7.756041e-01    1.432915e+00
      Policier      Documentaire      Musical      Comdie musicale
1.438141e+00    2.098601e+00    3.313492e+00    3.858722e+00
      Guerre
5.105570e+00
```

Figure 14 : Meilleurs genre/nationalités, selon la critique

# Conclusion

Au cours de ces Travaux Pratiques, nous nous sommes **familiarisés avec le langage/logiciel R**, et principalement deux méthodes statistiques : **la régression linéaire et l'analyse en composantes principales**.

L'analyse de données massives, constitue un **enjeu stratégique, sociétal et économique majeur**, pour les années à venir, tant **l'activité commerciale des entreprises génère de plus en plus de données**.

Ainsi, leur exploitation et leur analyse dépend de plusieurs paramètres, dont **l'exactitude**, l'un des plus importants, comme nous avons pu le constater au sein de ces Travaux Pratiques.

De plus, bien que cela ne soit pas l'objet de ce module, ces volumes de données posent également des **questions de sécurité**, comme l'ont montré certaines actualités (piratage de Yahoo en 2013, piratage de Sony en 2014, etc.).

Il est donc primordial pour les entreprises de mettre en place des **solutions de Data-Gouvernance**, et de respecter les **mesures fournies par le RGPD** en la matière.