BIO4558 Biostatistiques appliquées avec R

Manuel de Laboratoire

Julien Martin

20-08-2020

Contents

Note

Version en cours de développement pour le cours de l'automne 2020. Les chapitres vont apparaitre au cours de la session.

Préface

Les exercices de laboratoire que vous retrouverez dans les pages qui suivent sont conçus de manière à vous permettre de développer une expérience pratique en analyse de données à l'aide d'un logiciel (R). R est un logiciel très puissant, mais comme tous les logiciels, il a des limites. En particulier il ne peut réfléchir à votre place, vous dire si l'analyse que vous tentez d'effectuer est appropriée ou sensée, ou interpréter biologiquement les résultats.

Quelques points importants à retenir

- Avant de commencer une analyse statistique, il faut d'abord vous familiariser son fonctionnement. Cela ne veut pas dire que vous devez connaître les outils mathématiques qui la sous-tendent, mais vous devriez au moins comprendre les principes utilisés lors de cette analyse. Avant de faire un exercice de laboratoire, lisez donc la section correspondante dans les notes de cours. Sans cette lecture préalable, il est très probable que les résultats produits par le logiciel, même si l'analyse a été effectuée correctement, seront indéchiffrables.

- Les laboratoires sont conçus pour compléter les cours théoriques et vice versa. À cause des contraintes d'horaires, il se pourrait que le cours et le laboratoire ne soient pas parfaitement synchronisés. N'hésitez donc pas à poser des questions sur le labo en classe ou des questions théoriques au laboratoire.

- Travaillez sur les exercices de laboratoire à votre propre rythme. Certains exercices prennent beaucoup moins de temps que d'autres et il n'est pas nécessaire de compléter un exercice par séance de laboratoire. En fait deux séances de laboratoire sont prévues pour certains des exercices. Même si vous n'êtes pas notés sur les exercices de laboratoire, soyez conscient que ces exercices sont essentiels. Si vous ne les faites pas, il est très peu probable que vous serez capable de compléter les devoirs et le projet de session. Prenez donc ces exercices de laboratoire au sérieux !

- Les 2 premier laboratoires sont conçu pour vous permettre d'acquérir ou

de réviser le minimum de connaissances requises pour vous permettre de réaliser les exercices de laboratoires avec R. Il y a presque toujours de multiples façons de faire les choses avec R et vous ne trouverez ici que des méthodes simples. Ceux et celles d'entre vous qui y sont enclins pourront trouver en ligne des instructions plus détaillées et complexes. En particulier, je vous conseille :

- R pour les débutants http://cran.r-project.org/doc/contrib/Paradis-rdebuts_fr.pdf
- Using R for psychological research: A simple guide to an elegant package http://www.personality-project.org/r/
- An introduction to R http://cran.r-project.org/doc/manuals/R-intro.html
- Si vous préférez des manuels, le site web de CRAN en garde une liste commentée à : http://www.r-project.org/doc/bib/R-books.html
- Finalement, comme aide-mémoire à garder sous la main, je vous recommande R reference card par Tom Short http://cran.r-project.org/doc/contrib/Short-refcard.pdf

Qu'est-ce que R et pourquoi l'utiliser dans ce cours?

R est un logiciel libre et multiplateforme formant un système statistique et graphique. R est également un langage de programmation spécialisé pour les statistiques. C'est un dialecte du langage S. S-Plus est un autre dialecte, très semblable, et forme un produit commercial qui a un interface graphique que certains trouvent plus convivial.

R a deux très grands avantages pour ce cours, et un inconvénient embêtant initialement mais qui vous forcera à acquérir des excellentes habitudes de travail. Le premier avantage est que vous pouvez tous l'installer sur votre (ou vos) ordinateurs personnel gratuitement. C'est important parce que c'est à l'usage que vous apprendrez et maîtriserez réellement les biostatistiques et cela implique que vous devez avoir un accès facile et illimité à un logiciel statistique. Le deuxième avantage est que R peut tout faire en statistiques. R est conçu pour être extensible et est devenu l'outil de prédilection des statisticiens mondialement. La question n'est plus : " Est-ce que R peut faire ceci? ", mais devient" Comment faire ceci avec R ". Et la recherche internet est votre ami. Aucun autre logiciel n'offre ces deux avantages.

L'inconvénient embêtant initialement est que l'on doit opérer R en tapant des instructions (ou en copiant des sections de code) plutôt qu'en utilisant des menus et en cliquant sur différentes options. Si on ne sait pas quelle commande taper, rien ne se passe. Ce n'est donc pas facile d'utilisation à priori. Cependant, il est possible d'apprendre rapidement à faire certaines des opérations de base (ouvrir un fichier de données, faire un graphique pour examiner ces données,

effectuer un test statistique simple). Et une fois que l'on comprend le principe de la chose, on peut assez facilement trouver sur le web des exemples d'analyses ou de graphiques plus complexes et adapter le code à nos propres besoins. C'est ce que vous ferez dans le premier laboratoire pour vous familiariser avec R.

Pourquoi cet inconvénient est-il d'une certaine façon un avantage? Parce que vous allez sauver du temps en fin de compte. Garanti. Croyez-moi, on ne fait jamais une analyse une seule fois. En cours de route, on découvre des erreurs d'entrée de données, ou que l'on doit faire l'analyse séparément pour des sous-groupes, ou on obtient des données supplémentaires, ou on fait une erreur. On doit alors recommencer l'analyse. Avec une interface graphique et des menus, cela implique recommencer à cliquer ici, entre des paramètres dans des boîtes et sélectionner des boutons. Chaque fois avec possibilité d'erreur. Avec une série de commandes écrites, il suffit de corriger ce qui doit l'être puis de copier-coller l'ensemble pour répéter instantanément. Et vous avez la possibilité de parfaitement documenter ce que vous avez fait. C'est comme cela que les professionnels travaillent et offrent une assurance de qualité de leurs résultats.

Installation

Pour installer R sur un nouvel ordinateur, allez au site http://cran.r-project.org/. Vous y trouverez des versions compilées (binaries) ou non (sources) pour votre système d'exploitation de prédilection (Windows, MacOS, Linux).

Note : R a déjà été installé sur les ordinateurs du laboratoire (la version pourrait être un peu plus ancienne, mais cela devrait être sans conséquences).

Instructions générales pour les laboratoires

- Apporter une clé USB ou son équivalent à chaque séance de laboratoire pour sauvegarder votre travail.
- Lire l'exercice de laboratoire AVANT la séance, lire le code R correspondant et préparer vos questions sur le code.
- Durant les pré-labs, écouter les instructions et posez vos questions au moment approprié.
- Faites les exercices du manuel de laboratoire à votre rythme, en équipe, puis je vous recommande de commencer (compléter?) le devoir. Profitez de la présence du démonstrateur et du prof...
- Pendant vos analyses, copiez-collez des fragments de sorties de R dans un document (par exemple dans votre traitement de texte favori) et annotez abondamment.
- Ne tapez pas directement vos commandes dans R mais plutôt dans un script. Vous pourrez ainsi refaire le labo instantanément, récupérer des fragments de code, ou plus facilement identifier les erreurs dans vos analyses.

- Créez votre propre librairie de fragments de codes (snippets). Annotez-là abondamment. Vous vous en féliciterez plus tard.

Chapter 1

Introduction à R

Après avoir complété cet exercice de laboratoire, vous pourrez : - Ouvrir des fichiers de données R déjà existants - Importer des ensembles de données rectangulaires - Exporter des donnes de R vers un fichier texte - Vérifier si les données ont été correctement importées - Examiner la distribution des observations d'une variable - Examiner visuellement et tester la normalité d'une variable - Calculer des statistiques descriptives d'une variable - Effectuer des transformations de données

1.1 Importer et exporter des données

Il existe de multiple format four sauvegarder les données, les 2 plus utiles sont `.csv` et `.Rdata`. Les fichiers `.csv` sont utilisés pour stocker des données. Ils sont ouvrables par les éditeurs de texte (e.g. Word, Writer, atom, ...) et les tableurs (e.g. MS Excel, LO Calc). Les fichiers `.Rdata` sont utilisés pour stocker n'importe quelle objet R pas uniquement des données. Cependant, ces fichiers ne peuvent être lus et utilisés que par R.

Les données pour les exercices de laboratoire et pour les devoirs vous sont fournies déjà en format `.csv`.

1.1.1 Ouvrir un fichier de données en format `.Rdata`

Pour ouvrir ces fichiers, vous pouvez cliquer dessus et laisser votre système d'exploitation démarrer une nouvelle session de R ou encore, à partir de la console de R, taper sur une ligne de commande :

```
load(file.choose())
```

ce qui ouvrira une boîte de dialogue vous permettant d'aller choisir un fichier sur votre ordinateur. Si cette option semble très attirante de part sa simplicité, je

ne recommande pas de s'en servir car elle ne permet pas de reproduire l'analyse facilement. En effet, elle nécessite de choisir le document chaque que l'on souhaite l'utiliser.

Vous pouvez aussi directement taper le nom du fichier entre guillemet `"nom du fichier avec l'extension"`. Par exemple,

```
load("ErablesGatineau.Rdata")
```

1.1.2 Ouvrir un fichier de données en format `.csv`

Pour importer ces données en format `.csv` dans R, il faut utiliser la commande `read.csv()`. Par exemple, pour créer un objet R `erables` qui contient les données du fichier `ErablesGatineau.csv`, il faut utiliser la commande suivant.

```
erables <- read.csv("data/ErablesGatineau.csv")
```

Attrape : Attention si vous travaillez dans une langue utilisant la virgule au lieu du point décimal. Par défaut, R utilise le point décimal et vous n'obtiendrez pas le résultat escompté. Il existe une version modifiée de `read.csv()` appelée `read.csv2()` qui règle ce problème. Googlez-la si vous en avez besoin.

Pour vérifier si les données ont bel et bien été lues, vous pouvez lister les objets en mémoire avec la fonction `ls()` ou en obtenir une liste avec une description plus détaillée avec `ls.str()`

```
ls()
```

```
##  [1] "bs"                    "erables"               "mygraph"
##  [4] "RegModel.1"            "RegModel.2"            "RegModel.3"
##  [7] "results"               "salmonella"            "sturgeon"
## [10] "sturgeon.female.1978" "sturgeon.male"         "sturgeon.male.subset"
```

```
ls.str()
```

```
## bs : function (formula, data, indices)
## erables : 'data.frame': 100 obs. of  3 variables:
##  $ station: chr  "A" "A" "A" "A" ...
##  $ diam   : num  22.4 36.1 44.4 24.6 17.7 ...
##  $ biom   : num  732 1171 673 1552 504 ...
## mygraph : List of 9
##  $ data       :'data.frame': 80 obs. of  11 variables:
##  $ layers     :List of 3
##  $ scales     :Classes 'ScalesList', 'ggproto', 'gg' <ggproto object: Class ScalesLi
##     add: function
##     clone: function
##     find: function
##     get_scales: function
##     has_scale: function
##     input: function
```

```
##     n: function
##     non_position_scales: function
##     scales: list
##     super:  <ggproto object: Class ScalesList, gg>
##  $ mapping    :List of 2
##  $ theme      : list()
##  $ coordinates:Classes 'CoordCartesian', 'Coord', 'ggproto', 'gg' <ggproto object: Class Coord
##     aspect: function
##     backtransform_range: function
##     clip: on
##     default: TRUE
##     distance: function
##     expand: TRUE
##     is_free: function
##     is_linear: function
##     labels: function
##     limits: list
##     modify_scales: function
##     range: function
##     render_axis_h: function
##     render_axis_v: function
##     render_bg: function
##     render_fg: function
##     setup_data: function
##     setup_layout: function
##     setup_panel_guides: function
##     setup_panel_params: function
##     setup_params: function
##     train_panel_guides: function
##     transform: function
##     super:  <ggproto object: Class CoordCartesian, Coord, gg>
##  $ facet      :Classes 'FacetNull', 'Facet', 'ggproto', 'gg' <ggproto object: Class FacetNull,
##     compute_layout: function
##     draw_back: function
##     draw_front: function
##     draw_labels: function
##     draw_panels: function
##     finish_data: function
##     init_scales: function
##     map_data: function
##     params: list
##     setup_data: function
##     setup_params: function
##     shrink: TRUE
##     train_scales: function
##     vars: function
```

```
##     super:  <ggproto object: Class FacetNull, Facet, gg>
##  $ plot_env   :<environment: R_GlobalEnv>
##  $ labels     :List of 2
## RegModel.1 : List of 13
##  $ coefficients : Named num [1:2] 28.504 0.707
##  $ residuals    : Named num [1:75] 0.725 -4.517 2.903 -8.494 0.978 ...
##  $ effects      : Named num [1:75] -361.75 39.717 2.688 -7.819 0.558 ...
##  $ rank         : int 2
##  $ fitted.values: Named num [1:75] 36.3 33.5 42.6 33.5 44.8 ...
##  $ assign       : int [1:2] 0 1
##  $ qr           :List of 5
##  $ df.residual  : int 73
##  $ na.action    : 'omit' Named int [1:5] 63 64 66 68 70
##  $ xlevels      : Named list()
##  $ call         : language lm(formula = fklngth ~ age, data = sturgeon.male)
##  $ terms        :Classes 'terms', 'formula'  language fklngth ~ age
##  $ model        :'data.frame': 75 obs. of  2 variables:
## RegModel.2 : List of 13
##  $ coefficients : Named num [1:2] 1.192 0.341
##  $ residuals    : Named num [1:75] 0.02134 -0.0186 0.02304 -0.08279 0.00423 ...
##  $ effects      : Named num [1:75] -1.40e+01 4.74e-01 1.99e-02 -8.05e-02 4.07e-04 .
##  $ rank         : int 2
##  $ fitted.values: Named num [1:75] 1.55 1.48 1.64 1.48 1.66 ...
##  $ assign       : int [1:2] 0 1
##  $ qr           :List of 5
##  $ df.residual  : int 73
##  $ na.action    : 'omit' Named int [1:5] 63 64 66 68 70
##  $ xlevels      : Named list()
##  $ call         : language lm(formula = log10(fklngth) ~ log10(age), data = sturgeon
##  $ terms        :Classes 'terms', 'formula'  language log10(fklngth) ~ log10(age)
##  $ model        :'data.frame': 75 obs. of  2 variables:
## RegModel.3 : List of 13
##  $ coefficients : Named num [1:2] 1.227 0.312
##  $ residuals    : Named num [1:72] 0.01642 -0.02914 0.02557 0.00849 -0.02009 ...
##  $ effects      : Named num [1:72] -13.71657 0.41261 0.02249 0.00431 -0.02276 ...
##  $ rank         : int 2
##  $ fitted.values: Named num [1:72] 1.55 1.49 1.63 1.65 1.63 ...
##  $ assign       : int [1:2] 0 1
##  $ qr           :List of 5
##  $ df.residual  : int 70
##  $ na.action    : 'omit' Named int [1:5] 60 61 63 65 67
##  $ xlevels      : Named list()
##  $ call         : language lm(formula = log10(fklngth) ~ log10(age), data = sturgeon
##  $ terms        :Classes 'terms', 'formula'  language log10(fklngth) ~ log10(age)
##  $ model        :'data.frame': 72 obs. of  2 variables:
## results : List of 11
```

```
##  $ t0       : Named num [1:2] 1.192 0.341
##  $ t        : num [1:1000, 1:2] 1.23 1.21 1.2 1.22 1.2 ...
##  $ R        : num 1000
##  $ data     :'data.frame': 80 obs. of  11 variables:
##  $ seed     : int [1:626] 10403 392 -75673670 -1563284964 2045919631 1021160683 547669122 -160
##  $ statistic:function (formula, data, indices)
##  $ sim      : chr "ordinary"
##  $ call     : language boot(data = sturgeon.male, statistic = bs, R = 1000, formula = log10(fk
##  $ stype    : chr "i"
##  $ strata   : num [1:80] 1 1 1 1 1 1 1 1 1 1 ...
##  $ weights  : num [1:80] 0.0125 0.0125 0.0125 0.0125 0.0125 0.0125 0.0125 0.0125 0.0125 0.0125
## salmonella : 'data.frame': 60 obs. of  4 variables:
##  $ ratio : num  0.006322 0.00281 0.000111 0.007533 2.440643 ...
##  $ souche: chr  "SMR" "SMR" "SMR" "SMR" ...
##  $ milieu: chr  "IN VITRO" "IN VITRO" "IN VITRO" "IN VITRO" ...
##  $ labo  : int  1 2 3 4 5 6 7 8 9 10 ...
## sturgeon : 'data.frame': 186 obs. of  11 variables:
##  $ fklngth : num  37 50.2 28.9 50.2 45.6 ...
##  $ totlngth: num  40.7 54.1 31.3 53.1 49.5 ...
##  $ drlngth : num  23.6 31.5 17.3 32.3 32.1 ...
##  $ rdwght  : num  15.95 NA 6.49 NA 29.92 ...
##  $ age     : int  11 24 7 23 20 23 20 7 23 19 ...
##  $ girth   : num  40.5 53.5 31 52.5 50 54.2 48 28.5 44 39 ...
##  $ sex     : chr  "MALE" "FEMALE" "MALE" "FEMALE" ...
##  $ location: chr  "THE_PAS" "THE_PAS" "THE_PAS" "THE_PAS" ...
##  $ year    : int  1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 ...
##  $ lfklngth: num  1.57 1.7 1.46 1.7 1.66 ...
##  $ lrdwght : num  1.203 NA 0.812 NA 1.476 ...
## sturgeon.female.1978 : 'data.frame': 15 obs. of  9 variables:
##  $ fklngth : num  50.2 50.2 49.6 47.7 48.9 ...
##  $ totlngth: num  54.1 53.1 53.9 51.4 53.9 ...
##  $ drlngth : num  31.5 32.3 31.1 34 29.9 ...
##  $ rdwght  : num  NA NA 35.9 33.9 35.9 ...
##  $ age     : int  24 23 23 20 23 24 18 21 19 20 ...
##  $ girth   : num  53.5 52.5 54.2 48 52.5 NA NA NA NA NA ...
##  $ sex     : chr  "FEMALE" "FEMALE" "FEMALE" "FEMALE" ...
##  $ location: chr  "THE_PAS" "THE_PAS" "THE_PAS" "THE_PAS" ...
##  $ year    : int  1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 ...
## sturgeon.male : 'data.frame': 80 obs. of  11 variables:
##  $ fklngth : num  37 28.9 45.6 25 45.7 ...
##  $ totlngth: num  40.7 31.3 49.5 28.1 50.4 ...
##  $ drlngth : num  23.6 17.3 32.1 14.3 29.3 ...
##  $ rdwght  : num  15.95 6.49 29.92 4.73 29.48 ...
##  $ age     : int  11 7 20 7 23 19 17 14 21 29 ...
##  $ girth   : num  40.5 31 50 28.5 44 39 41 47 43.5 49 ...
##  $ sex     : chr  "MALE" "MALE" "MALE" "MALE" ...
```

```
##  $ location: chr  "THE_PAS" "THE_PAS" "THE_PAS" "THE_PAS" ...
##  $ year    : int  1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 ...
##  $ lfklngth: num  1.57 1.46 1.66 1.4 1.66 ...
##  $ lrdwght : num  1.203 0.812 1.476 0.675 1.47 ...
## sturgeon.male.subset : 'data.frame': 178 obs. of  11 variables:
##  $ fklngth : num  37 50.2 28.9 45.6 49.6 ...
##  $ totlngth: num  40.7 54.1 31.3 49.5 53.9 ...
##  $ drlngth : num  23.6 31.5 17.3 32.1 31.1 ...
##  $ rdwght  : num  15.95 NA 6.49 29.92 35.86 ...
##  $ age     : int  11 24 7 20 23 20 7 23 19 17 ...
##  $ girth   : num  40.5 53.5 31 50 54.2 48 28.5 44 39 41 ...
##  $ sex     : chr  "MALE" "FEMALE" "MALE" "MALE" ...
##  $ location: chr  "THE_PAS" "THE_PAS" "THE_PAS" "THE_PAS" ...
##  $ year    : int  1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 ...
##  $ lfklngth: num  1.57 1.7 1.46 1.66 1.7 ...
##  $ lrdwght : num  1.203 NA 0.812 1.476 1.555 ...
```

R confirme avoir en mémoire l'objet `ErablesGatineau`. `ErableGatineau` est un tableau de données rectangulaire (data.frame) contenant 100 observations (lignes) de 3 variables (colonnes): `station`, une variable de type Facteur avec 2 niveaux, et `diam` et `biom` qui sont 2 variables numériques.

1.1.3 Entrer des données

R n'est pas un environnement idéal pour entrer des données. C'est possible, mais la syntaxe est lourde et peut inciter à s'arracher les cheveux. Utilisez votre chiffrier préféré pour faire l'entrée de données. Ce sera plus efficace et moins frustrant.

1.1.4 Nettoyer/corriger des données

Une autre opération qui peut être frustrante en R. Mon conseil : ne le faites pas là. Retournez au fichier original, faites la correction, puis re-exportez les données vers R. Il est finalement plus simple de refaire exécuter les quelques lignes de code par la machine. Vous aurez à la fin une seule version (corrigée) de vos données et un code qui vous permet de refaire votre analyse.

1.1.5 Exporter des données à partir de R.

Vous pouvez utiliser la fonction, `{r write, eval =FALSE}  write.csv(mydata, file = "outfilename.csv", row.names = FALSE)` où `mydata` est le nom du base de données à exporter et `outfilename.csv` est le nom du fichier à produire. Notez que ce fichier sera créé dans le répertoire de travail (qui peut être changé par le menu à `File>Change dir`, ou par la commande `setwd()`)

1.2 Examen préliminaire des données

La première étape de toute analyse est l'examen des données. Elle nous permet de découvrir si on a bien importé les données, si les nombres enregistrés sont possibles, si toutes les données ont bien été lues, etc. L'examen préliminaire des données permet souvent aussi d'identifier des observations suspectes, possiblement dûes à des erreurs d'entrée de donnée. Finalement, l'examen graphique préliminaire permet en général de visualiser les tendances principales qui seront confirmées par l'analyse statistique en tant que telle. Le fichier `sturgeon.csv` contient les données d'une étude effectuée sur les esturgeons de la rivière Saskatchewan. Ces données ont été récoltées, entre autres, pour examiner comment la taille des esturgeons varie entre les sexes (sex), les sites (location), et les années (year).

- Pour recommencer avec une ardoise vide, videz la mémoire de R de tout son contenu en tapant la commande `rm(list=ls())`
- Chargez les données du fichier `sturgeon.csv` dans un objet `sturgeon`.
- Pour obtenir un aperçu des éléments du fichier qui ont été chargés en mémoire, taper la commande `str(sturgeon)`.

```
sturgeon <- read.csv("data/sturgeon.csv")
str(sturgeon)
```

```
## 'data.frame': 186 obs. of  9 variables:
##  $ fklngth : num  37 50.2 28.9 50.2 45.6 ...
##  $ totlngth: num  40.7 54.1 31.3 53.1 49.5 ...
##  $ drlngth : num  23.6 31.5 17.3 32.3 32.1 ...
##  $ rdwght  : num  15.95 NA 6.49 NA 29.92 ...
##  $ age     : int  11 24 7 23 20 23 20 7 23 19 ...
##  $ girth   : num  40.5 53.5 31 52.5 50 54.2 48 28.5 44 39 ...
##  $ sex     : chr  "MALE" "FEMALE" "MALE" "FEMALE" ...
##  $ location: chr  "THE_PAS" "THE_PAS" "THE_PAS" "THE_PAS" ...
##  $ year    : int  1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 ...
```

1.2.1 Sommaire statistique

Pour un sommaire du contenu du base de données appelé sturgeon qui est en mémoire, taper la commande

```
summary(sturgeon)
```

```
##     fklngth         totlngth        drlngth          rdwght      
##  Min.   :24.96   Min.   :28.15   Min.   :14.33   Min.   : 4.73  
##  1st Qu.:41.00   1st Qu.:43.66   1st Qu.:25.00   1st Qu.:18.09  
##  Median :44.06   Median :47.32   Median :27.00   Median :23.10  
##  Mean   :44.15   Mean   :47.45   Mean   :27.29   Mean   :24.87  
##  3rd Qu.:48.00   3rd Qu.:51.97   3rd Qu.:29.72   3rd Qu.:30.27  
##  Max.   :66.85   Max.   :72.05   Max.   :41.93   Max.   :93.72  
```

```
##                 NA's   :85      NA's   :13      NA's   :4
##       age           girth           sex              location
##  Min.   : 7.00   Min.   :11.50   Length:186         Length:186
##  1st Qu.:17.00   1st Qu.:40.00   Class :character   Class :character
##  Median :20.00   Median :44.00   Mode  :character   Mode  :character
##  Mean   :20.24   Mean   :44.33
##  3rd Qu.:23.50   3rd Qu.:48.80
##  Max.   :55.00   Max.   :73.70
##  NA's   :11      NA's   :85
##       year
##  Min.   :1978
##  1st Qu.:1979
##  Median :1979
##  Mean   :1979
##  3rd Qu.:1980
##  Max.   :1980
##
```

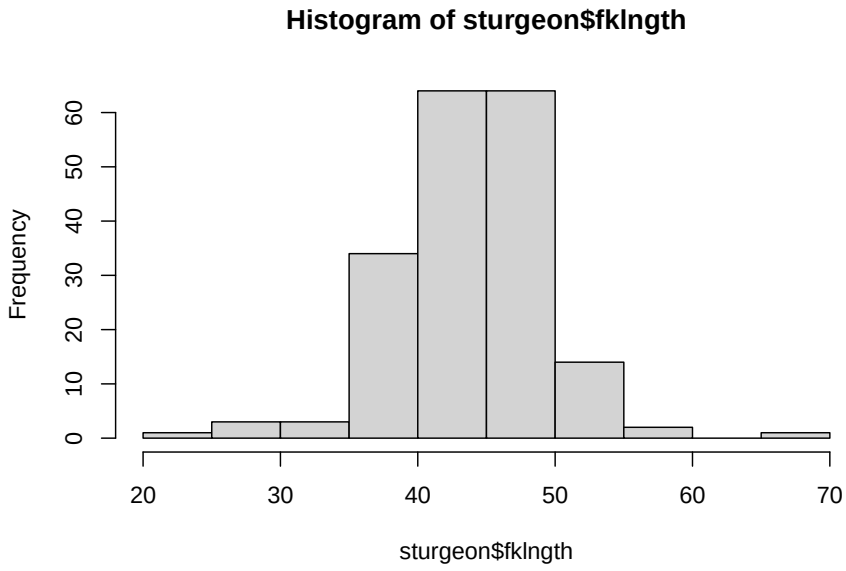
Pour chaque variable, R donne le minimum, le maximum, la médiane qui est la valeur au milieu de la liste des observations ordonnées (appelée le 50 ième percentile), ici, la 93 ième valeur des 186 observations, les valeurs au premier (25%) et troisième quartile (75%), et si il y a des valeurs manquantes dans la colonne. Notez que plusieurs des variables ont des observations manquantes (NA). Donc, seules les variables fklngth (longueur à la fourche), sex, location et year ont 186 observations.

Attrape : Attention aux valeurs manquantes. Plusieurs fonctions de R y réagissent mal et on doit souvent faire les analyses sur des sous- ensembles sans valeur manquante, par des commandes ou des options dans les commandes. On y reviendra, mais prenez l'habitude de noter mentalement si il y a des données manquantes et de vous en rappeler en faisant l'analyse.

1.2.2 Histogramme, densité de probabilité empirique, boxplot et examen visuel de la normalité

Examinons maintenant de plus près la distribution de fklngth. La commande `hist()` permet de tracer un histogramme de la variable fklngth dans le base de données sturgeon.

```
hist(sturgeon$fklngth)
```



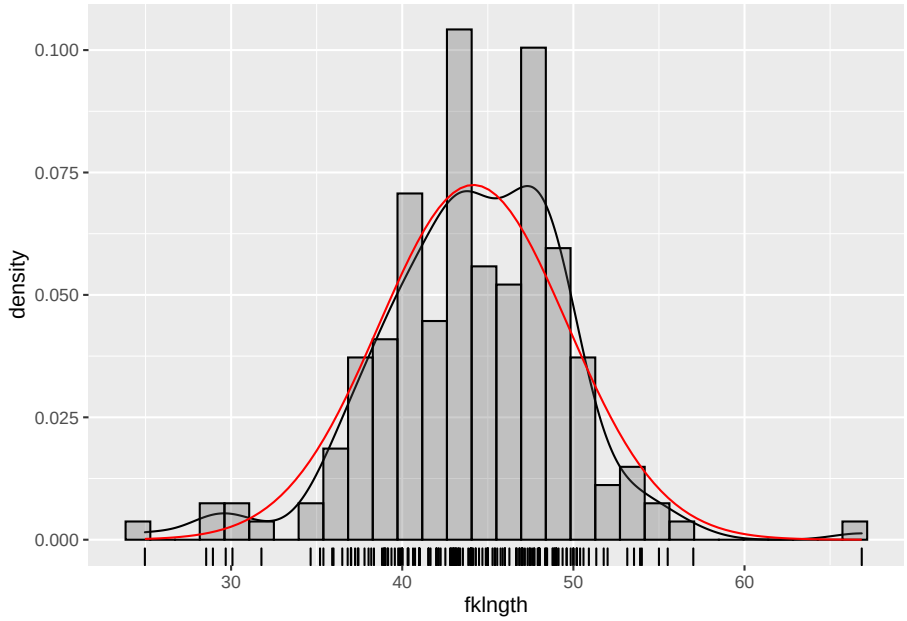
Les données semblent suivre approximativement une distribution normale. C'est bon à savoir. Cette syntaxe est un peu lourde puisqu'on doit ajouter le préfixe `sturgeon$` devant chaque nom de variable. On pourrait se faciliter la tâche en utilisant la commande `attach()` **mais cela est fortement déconseillé** et jamais utilisé dans ce document.

Cet histogramme est la représentation classique. Mais les histogrammes ne sont pas parfaits. Leur forme dépend en partie du nombre de catégories utilisées, surtout pour les petits échantillons. On peut faire mieux, particulièrement si on est intéressé à comparer visuellement la distribution des observations à une distribution normale. Mais il faut programmer un peu (ou savoir copier-coller. . .)

- Copiez-collez le code suivant dans une nouvelle fenêtre script (File->New script, ou Ctrl-n dans Windows), puis exécutez le.

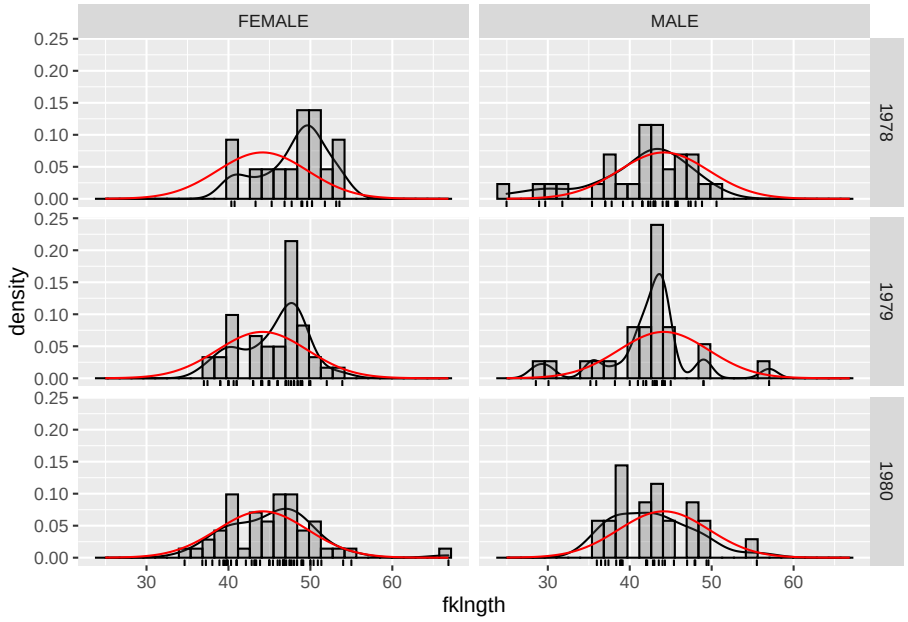
```
library(ggplot2)
#  use "sturgeon" dataframe to make plot called mygraph
#  and define x axis as representing fklngth
mygraph <- ggplot(sturgeon, aes(x = fklngth))
#  add data to the mygraph ggplot
mygraph <- mygraph +
#  add data density smooth
    geom_density() +
#  add rug (bars at the bottom of the plot)
    geom_rug() +
#  add black semitransparent histogram
```

```
    geom_histogram(aes(y = ..density..), bins = 30, color = "black",    alpha = 0.3) +
#  add normal curve in red, with mean and sd from fklength
    stat_function(fun = dnorm,
                  args = list(
                    mean = mean(sturgeon$fklngth),
                    sd = sd(sturgeon$fklngth) ),
                  color = "red")
# display graph
    mygraph
```



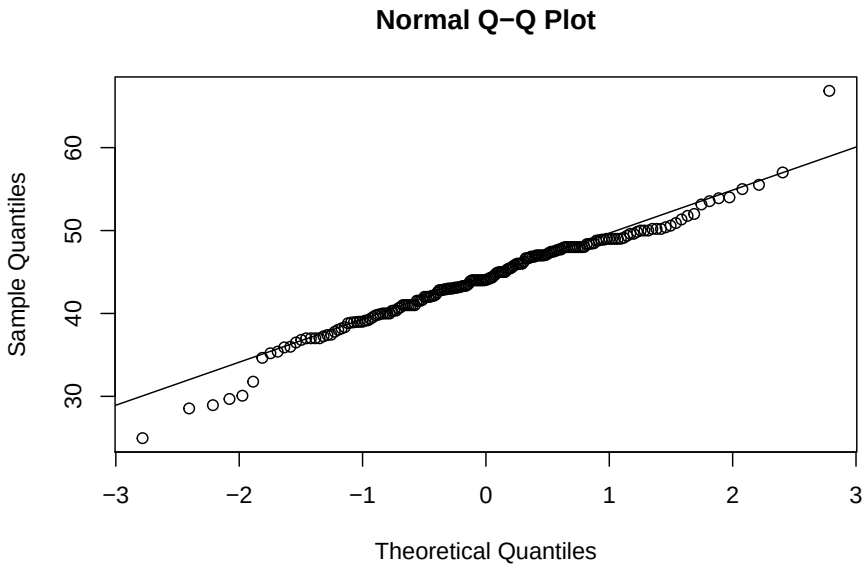
Chaque observation est représentée par une barre sous l'axe des x (rug). En rouge est la distribution normale de données avec la même moyenne et écart-type que les observations. Et l'autre ligne est la densité de probabilité empirique, « lissée » à partir des observations. Si vous êtes plus aventureux, vous pouvez examiner la distribution des observations de fklngth par sous-groupes (par exemple sex et year) avec :

```
mygraph + facet_grid(year ~ sex)
```



Chaque panneau illustre la distribution pour un sexe cette année-là, et la courbe en rouge récurrente représente la distribution normale pour l'ensemble des données. Cette courbe peut servir à mieux évaluer visuellement les différences entre les panneaux. Une autre façon d'évaluer la normalité de données visuellement est de faire un QQ plot avec la paire de commandes `qqnorm()` et `qqline()`.

```
qqnorm(sturgeon$fklngth)
qqline(sturgeon$fklngth)
```



Des données parfaitement normales suivraient la ligne droite diagonale. Ici, il y a des déviations dans les queues de la distribution, et un peu à droite du centre. Comparez cette représentation à celle des deux graphiques précédents. Vous conviendrez sans doute avec moi qu'il est plus facile de visualiser comment la distribution dévie de la normalité sur les histogrammes et les graphiques de la densité empirique de probabilité que sur les QQ plots. Ceci dit, les QQ plots sont souvent utilisés et vous devriez être capable de les interpréter. De plus, on peut facilement éprouver statistiquement l'hypothèse que les données sont distribuées normalement avec R par la commande `shapiro.test()` qui calcule une statistique (W) qui est une mesure de la tendance des points d'un QQ plot à former une ligne parfaite. Si oui, alors W=1. Si W s'éloigne de 1 (vers 0), alors les données s'éloignent de la normalité. Ici,

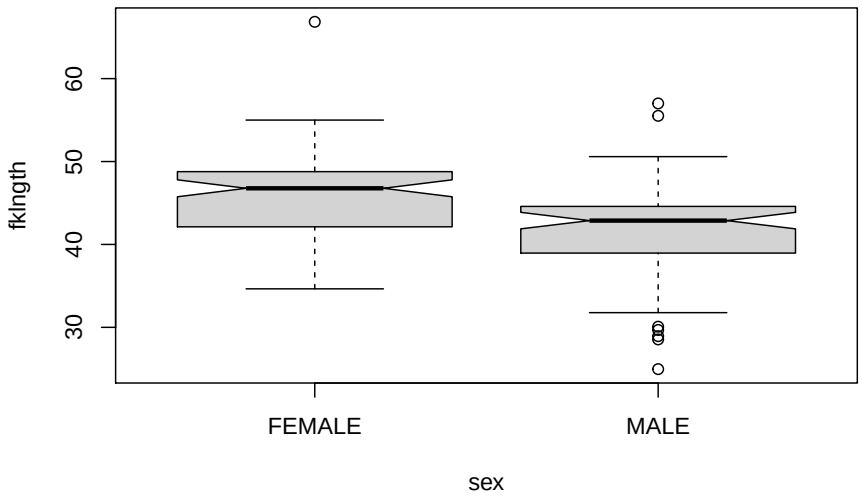
```
shapiro.test(sturgeon$fklngth)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sturgeon$fklngth
## W = 0.97225, p-value = 0.0009285
```

W n'est pas très loin de 1, mais suffisamment pour que la différence soit significative. L'examen visuel des grands échantillons est souvent compliqué par le fait que plusieurs points se superposent et qu'il devient plus difficile de bien visualiser la tendance centrale. Les boxplots avec "moustaches" (box and whiskers plots) offrent une alternative intéressante. La commande `boxplot()` peut produire un

boxplot de fklngth pour chaque niveau de sex, et ajoute les coches.

```
boxplot(fklngth ~ sex, data = sturgeon, notch = TRUE)
```



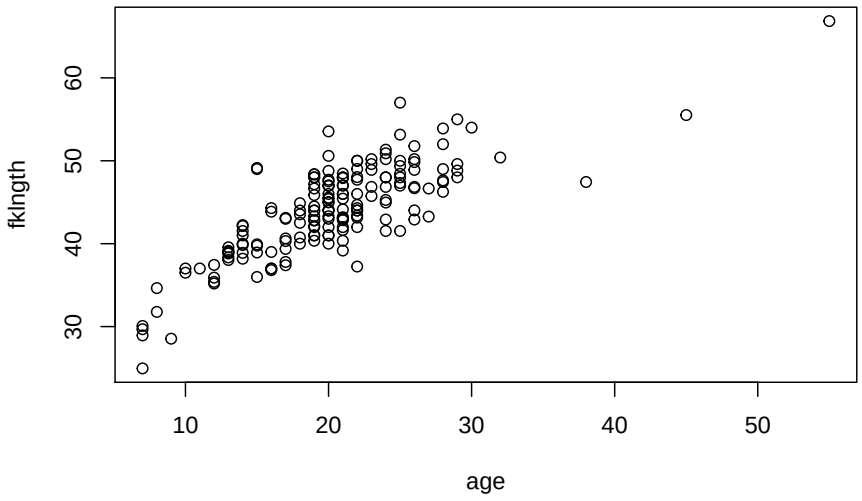
La ligne un peu plus épaisse dans la boîte de la Fig.7 indique la médiane. La coche est proportionnelle à l'incertitude quant à la position de la médiane. On peut visuellement interpréter approximativement les différences entre médianes en examinant si il y a chevauchement entre les coches (ici, il n'y a pas chevauchement, et on conclurait provisoirement que la médiane de fklngth pour les femelles est supérieure à celle des mâles). Les boîtes s'étendent du premier au troisième quartile (du 25ième au 75ième percentile si vous préférez), Les barres (moustaches ou whiskers) au-dessus et en dessous des boîtes s'étendent soit de la valeur minimum à la valeur maximum, ou, si il y a des valeurs extrêmes, de la plus petite à la plus grande valeur à l'intérieur de 1.5x la largeur de l'étendue interquartile . Enfin, les observations qui excèdent les limites des moustaches (donc à plus de 1.5x l'étendue interquartile de chaque côté de la médiane) sont indiquées par des symboles.Ce sont des valeurs qui pourraient être considérées comme extrêmes et possiblement aberrantes.

1.2.3 Diagrammes de dispersion bivariés

En plus des graphiques pour chacune des variables séparément, il est très souvent intéressant de jeter un coup d'oeil aux diagrammes de dispersion . La commande `plot(y~x)` permet de faire le graphique de y sur l'axe vertical (l'ordonnée) en fonction de x sur l'axe horizontal (l'abscisse).

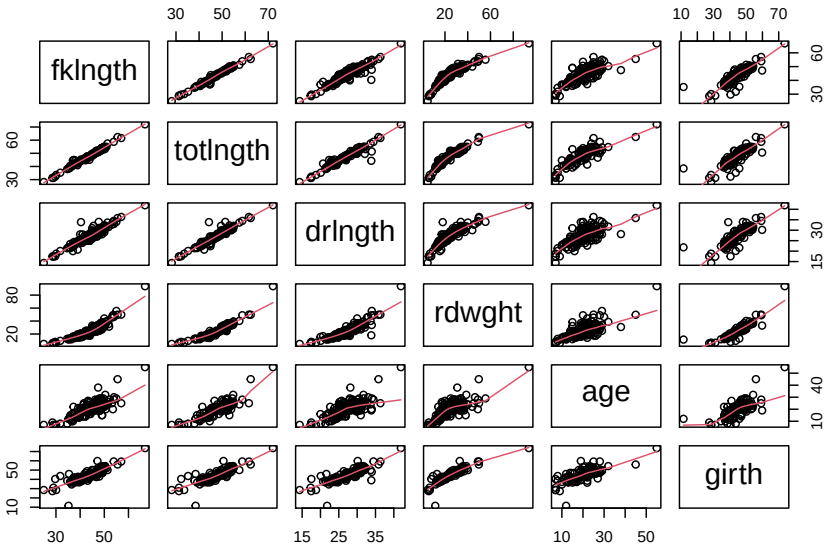
- Faites un graphique de fklngth en fonction de age avec la commande plot. Vous devriez obtenir:

```
plot(fklngth ~ age, data = sturgeon)
```



R a une fonction qui permet la création des graphiques de dispersion de toutes les paires de variables (`pairs()`). Une des option de ¬ est l'ajout d'une trace lowess qui indique la tendance de la relation entre les variables. Pour obtenir la matrice de ces graphiques avec la trace lowess pour toutes les variable dans sturgeon, entrer la commande `pairs(sturgeon[,1:6], panel=panel.smooth)` et vous devriez obtenir

```
pairs(sturgeon[,1:6], panel = panel.smooth)
```



1.3 Créer des sous-ensembles de cas

Il arrive fréquemment qu'une analyse se concentre sur un sous-ensemble des observations contenues dans un fichier de données. Les cas sont d'habitude sélectionnés selon un critère en particulier. Pour utiliser un sous-ensemble de vos données en créant un graphique ou en performant une analyse, on peut utiliser la commande `subset()`. Par exemple, pour créer un sous ensemble des données du tableau sturgeon qui ne contient que les femelles capturées en 1978, on peut écrire :

```
sturgeon.female.1978 <- subset(sturgeon, sex == "FEMALE" & year == "1978")
sturgeon.female.1978
```

```
##      fklngth totlngth  drlngth rdwght age girth    sex   location year
## 2   50.19685 54.13386 31.49606     NA  24  53.5 FEMALE    THE_PAS 1978
## 4   50.19685 53.14961 32.28346     NA  23  52.5 FEMALE    THE_PAS 1978
## 6   49.60630 53.93701 31.10236  35.86  23  54.2 FEMALE    THE_PAS 1978
## 7   47.71654 51.37795 33.97638  33.88  20  48.0 FEMALE    THE_PAS 1978
## 15  48.89764 53.93701 29.92126  35.86  23  52.5 FEMALE    THE_PAS 1978
## 105 46.85039       NA 28.34646  23.90  24    NA FEMALE CUMBERLAND 1978
## 106 40.74803       NA 24.80315  17.50  18    NA FEMALE CUMBERLAND 1978
## 107 40.35433       NA 25.59055  20.90  21    NA FEMALE CUMBERLAND 1978
## 109 43.30709       NA 27.95276  24.10  19    NA FEMALE CUMBERLAND 1978
## 113 53.54331       NA 33.85827  48.90  20    NA FEMALE CUMBERLAND 1978
## 114 51.77165       NA 31.49606  35.30  26    NA FEMALE CUMBERLAND 1978
```

```
## 116 45.27559       NA 26.57480  23.70  24    NA FEMALE CUMBERLAND 1978
## 118 53.14961       NA 32.67717  45.30  25    NA FEMALE CUMBERLAND 1978
## 119 50.19685       NA 32.08661  33.90  26    NA FEMALE CUMBERLAND 1978
## 123 49.01575       NA 29.13386  37.50  22    NA FEMALE CUMBERLAND 1978
```

Attrape: Dans ces comparaisons, il faut toujours utiliser `==` pour égal à. Dans ce contexte, si vous utilisez `=` seulement, vous n'obtiendrez pas ce que vous désirez. Dans le tableau qui suit se trouve une liste de commandes communes que vous allez probablement utiliser pour créer des expressions en R.

Operateur	Explication	Operateur	Explication
==	Égal à	!=	Pas égal à
>	Plus que	<	Moins que
>=	Plus que ou égal à	<=	Moins que ou égal à
&	Et vectorisé	|	Ou vectorisé
&&	Et contrôle	||	Ou contrôle
!	Pas		

- En utilisant les commandes `subset()` et `hist()`, essayez de faire un histogramme pour le sous-ensemble de cas correspondant aux femelles capturées en 1979 et 1980 (donc `sex == "FEMALE" & (year == "1979" | year == "1980")`)

1.4 Transformations de données

Il est très fréquemment nécessaire d'effectuer des transformations mathématiques sur les données brutes pour mieux satisfaire aux conditions d'application de tests statistiques. R étant aussi un langage de programmation complet, il peut donc effectuer les transformations désirées. Les fonctions les plus fréquemment utilisées sont:

- `log()`
- `sqrt()`
- `ifelse()`

On peut employer ces fonctions directement dans les lignes de commandes, ou encore créer de nouvelles variables orphelines ou faisant partie d'un data.frame. Par exemple, pour faire un graphique du logarithme décimal de fklngth en fonction de l'âge, on peut écrire

```
plot(log(fklngth)~age, data = sturgeon)
```

Pour créer une variable orpheline (i.e. non incluse dans le data.frame) appelée logfklngth et contenant le logarithme décimal de fklngth, on peut écrire

```
logfklngth <- log10(sturgeon$fklngth)
```

Si on veut ajouter cette variable transformée à un tableau de données (data.frame), alors, on doit préfixer le nom de la variable par le nom du base de données et du symbole `$`, par exemple, pour ajouter une variable nommée `lfkl` contenant le log10 de fklngth au tableau sturgeon, on peut écrire:

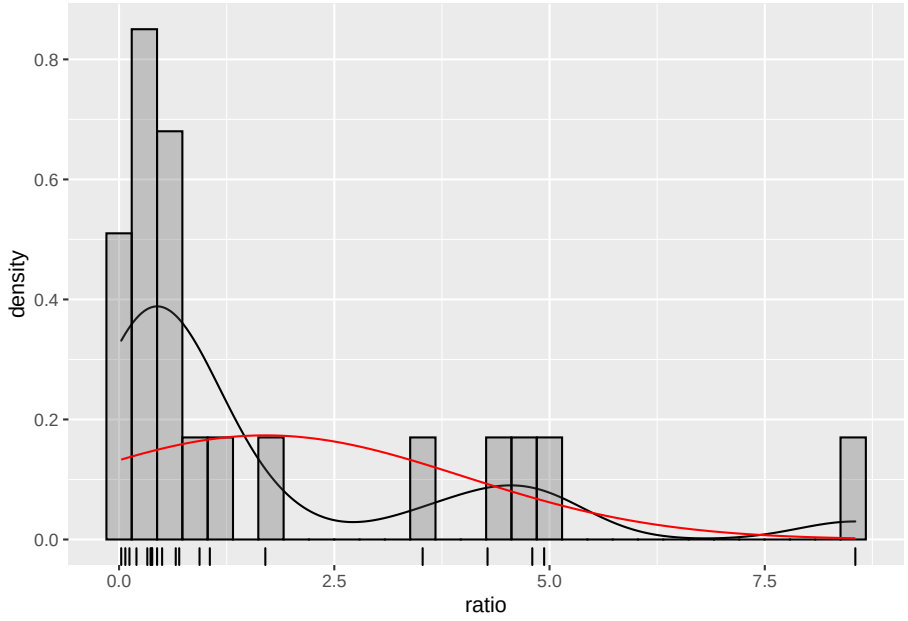
```
sturgeon$logfkl <- log10(sturgeonfklngth)
```

N'oubliez pas de sauvegarder ce tableau modifié si vous voulez avoir accès à cette nouvelle variable dans le futur. Pour les transformations conditionnelles, on peut utiliser la fonction `ifelse()`. Par exemple, pour créer une nouvelle variable appelée dummy qui sera égale à 1 pour les mâles et 0 pour les femelles, on peut écrire:

```
sturgeon$dummy <- ifelse(sturgeon$sex == "MALE", 1, 0)
```

1.5 Exercice sur R

Vous trouverez dans le fichier salmonella, des valeurs numériques du ratio pour deux milieux (IN VITRO et IN VIVO) pour trois souches. Examinez les données pour ratio et faites des graphiques pour évaluer la normalité de la distribution des ratios pour la souche SAUVAGE.



Chapter 2

Analyse de puissance avec R et G*Power

Après avoir complété cet exercice de laboratoire, vous devriez :

- Pouvoir calculer la puissance d'un test de t avec R et G*Power
- Pouvoir calculer l'effectif requis pour obtenir la puissance désirée avec un test de t
- Pouvoir calculer la taille de l'effet détectable par un test de t étant donné l'effectif, la puissance et α
- Comprendre comment la puissance change lorsque l'effectif augmente, la taille de l'effet change, ou lorsque α diminue
- Comprendre comment la puissance est affectée lorsque l'on passe d'un test bilatéral à un test unilatéral

2.1 La théorie

2.1.1 Qu'est-ce que la puissance?

La puissance est la probabilité de rejeter l'hypothèse nulle quand elle est fausse.

2.1.2 Pourquoi faire une analyse de puissance?

Évaluer l'évidence

L'analyse de puissance effectuée après avoir accepté une hypothèse nulle permet de calculer la probabilité que l'hypothèse nulle soit rejetée si elle était fausse et que la taille de l'effet était d'une valeur donnée. Ce type d'analyse a posteriori est très commun.

Planifier de meilleures expériences

L'analyse de puissance effectuée avant de réaliser une expérience (le plus souvent après une expérience préliminaire cependant), permet de déterminer le nombre d'observations nécessaires pour détecter un effet d'une taille donnée à un niveau fixe de probabilité (la puissance). Ce type d'analyse a priori devrait être réalisé plus souvent.

Estimer la "limite de détection" statistique

L'effort d'échantillonnage est souvent déterminé à l'avance (par exemple lorsque vous héritez de données récoltées par quelqu'un d'autre), ou très sévèrement limité (lorsque les contraintes logistiques prévalent). Que ce soit a priori ou a posteriori l'analyse de puissance vous permet d'estimer, pour un effort d'échantillonnage donné et un niveau de puissance fixe, quelle est la taille minimale de l'effet qui peut être détecté (comme étant statistiquement significatif).

2.1.3 Facteurs qui affectent la puissance

Il y a 3 facteurs qui affectent la puissance d'un test statistique.

Le critère de décision

La puissance dépend de α, le seuil de probabilité auquel on rejette l'hypothèse nulle. Si ce seuil est très strict (*i.e.* si α est fixé à une valeur très basse, comme 0.1% ou p = 0.001), alors la puissance sera plus faible que si le seuil était moins strict.

La taille de l'échantillon

Plus l'échantillon est grand, plus la puissance est élevée. La capacité d'un test à détecter de petites différences comme étant statistiquement significatives augmente avec une augmentation du nombre d'observations.

La taille de l'effet

Plus la taille de l'effet est grande, plus un test a de puissance. Pour un échantillon de taille fixe, la capacité d'un test à détecter un effet comme étant statistiquement significatif est plus élevée si l'effet est grand que s'il est petit. La taille de l'effet est en fait une mesure du degré de fausseté de l'hypothèse nulle.

2.2 Qu'est ce que G*Power?

G*Power est un programme gratuit, développé par des psychologues de l'Université de Dusseldorf en Allemagne. Le programme existe en version Mac et Windows. Il peut cependant être utilisé sous linux via Wine. G*Power vous permettra d'effectuer une analyse de puissance pour la majorité des tests que

nous verrons au cours de la session sans avoir à effectuer des calculs complexes ou farfouiller dans des tableaux ou des figures décrivant des distributions ou des courbes de puissance. Il est possible de faire tous les analyses de G*power avec R, mais cela est nettement plus complexes, car il faut tous coder à la main. Dans les cas les plus simple le code R est aussi fourni. G*power est vraiment un outil très utile que vous devrez maîtriser.

- Téléchargez le programme **ici** et installez-le sur votre ordi et votre station de travail au laboratoire (si ce n'est déjà fait).*

2.3 Comment utiliser G*Power

2.3.1 Principe général

L'utilisation de G*Power implique généralement en trois étapes:

1. Choisir le test approprié
2. Choisir l'un des 5 types d'analyses de puissance disponibles
3. Inscrire les valeurs des paramètres requis et cliquer sur Calculate

2.3.2 Types d'analyses de puissance disponibles

A priori

Calcule l'effectif requis pour une valeur de α, β et de taille d'effet donnée. Ce type d'analyse est utile à l'étape de planification des expériences.

Compromis

Calcule α et β pour un rapport β/α donné, un effectif fixe, et une taille d'effet donnée. Ce type d'analyse est plus rarement utilisé (je ne l'ai jamais fait), mais peut être utile lorsque le rapport β/α est d'intérêt, par exemple lorsque le coût d'une erreur de type I et de type II peut être quantifié.

Critère

Calcule α pour β, effectif et taille d'effet donné. En pratique, je vois peu d'utilité pour ce type de calcul. Contactez-moi si vous en voyez une!

Post-hoc

Calcule la puissance (1 - β) pour α, une taille d'effet et un effectif donné. Très utilisée pour interpréter les résultats d'une analyse statistique non-significative, mais seulement si l'on utilise une taille d'effet biologiquement significative (et non la taille d'effet observée). Peu pertinente lorsque le test est significatif.

Sensitivité

Calcule la taille d'effet détectable pour une valeur d'α, β et un effectif donné. Très utile également au stade de planification des expériences.

2.3.3 Comment calculer la taille de l'effet G*Power permet de faire une analyse de puissance pour de nombreux tests statistiques

L'indice de la taille de l'effet qui est utilisé par G*Power pour les calculs dépend du test. Notez que d'autres logiciels peuvent utiliser des indices différents et il est important de vérifier que l'indice que l'on utilise est celui qui convient. G*Power vous facilite la tâche et permet de calculer la taille de l'effet en inscrivant seulement les valeurs pertinentes dans la fenêtre de calcul. Le tableau suivant donne les indices utilisés par G*Power pour les différents tests.

Test	Taille d'effet	Formule
test de t sur des moyennes	d	$d = \frac{\lvert\mu_1 - \mu_2\rvert}{\sqrt{({s_1}^2 + {s_2}^2)/2}}$
test de t pour des corrélations	r	
autres tests de t	f	$f = \frac{\mu_1}{\sigma}$
test F (ANOVA)	f	$f = \frac{\sqrt{\sum_{i=1}^{k} {(\mu_i - \mu)}^2}}{\sigma}$
autres test F	f^2	$f^2 = \frac{{R_p}^2}{1 - {R_p}^2}$ R_p est le coefficient de corrélation partiel
test Chi-carré	w	$w = \sqrt{\sum_{i=1}^{m} \frac{{(p_{0i} - p_{1i})}^2}{p_{0i}}}$ p_{0i} p_{1i} sont les proportions de la catégorie i prédites par l'hypothèse nulle et alternative respectivement

2.4 Puissance pour un test de t comparant deux moyennes

L'objectif de cette séance de laboratoire est de vous familiariser avec G*Power et de vous aider à comprendre comment les quatre paramètres des analyses de puissance (α, β, effectif et taille de l'effet) sont reliés entre eux. On examinera seulement un des nombreux tests, le test de t permettant de comparer deux moyennes indépendantes. C'est le test le plus communément utilisé par les biologistes, vous l'avez tous déjà utilisé, et il conviendra très bien pour les besoins de la cause. Ce que vous apprendrez aujourd'hui s'appliquera à toutes les autres analyses de puissance que vous effectuerez à l'avenir.

Jaynie Stephenson a étudié la productivité des ruisseaux de la région d'Ottawa. Elle a, entre autres, quantifié la biomasse des poissons dans 18 ruisseaux sur le Bouclier Canadien d'une part, et dans 18 autres ruisseaux de la vallée de la rivière des Outaouais et de la rivière Rideau d'autre part. Elle a observé une biomasse plus faible dans les ruisseaux de la vallée (2.64 g/m^2 , écart-type=3.28) que dans ceux du Bouclier (3.31 g/m^2 , écart-type=2.79). En faisant un test de t pour éprouver l'hypothèse nulle que la biomasse des poissons est la même dans les deux régions, elle obtient:

```
Pooled-Variance Two-Sample t-Test
t = -0.5746, df = 34, p-value = 0.5693
```

Elle accepte l'hypothèse nulle (puisque p est plus élevé que 0.05) conclue donc que la biomasse moyenne des poissons est la même dans ces deux régions.

2.4.1 Analyse post-hoc

Compte tenu des valeurs des moyennes observées et de leur écart- type, on peut utiliser G*Power pour calculer la puissance du test de t bilatéral pour deux moyennes indépendantes et pour la taille d'effet (i.e. la différence entre la biomasse entre les deux régions, pondérée par les écarts-type) à $\alpha = 0.05$.

Démarrer G*Power.

1. À **Test family**, choisir: t tests
2. À **Statistical test**, choisir: Means: Difference between two inde- pendent means (two groups)
3. À **Type of power analysis**, choisir: Post hoc: Compute achieved power - given α, sample size, and effect size
4. Dans **Input Parameters**,

- à la boîte **Tail(s)**, choisir: Two,
- vérifier que α **err prob** est égal à 0.05
- inscrire 18 pour **Sample size group** 1 et 2
- pour calculer la taille d'effet (Effect size d), cliquer sur le bouton **Determine =>**

5. Dans la fenêtre qui s'ouvre à droite, sélectionner **n1 = n2**

- entrer les moyennes (**Mean group** 1 et 2)
- entrer les écarts types (**SDs group** 1 et 2)
- cliquer sur le bouton **Calculate and transfer to main window**

6. Cliquer sur le bouton Calculate dans la fenêtre principale et vous devriez obtenir ceci:

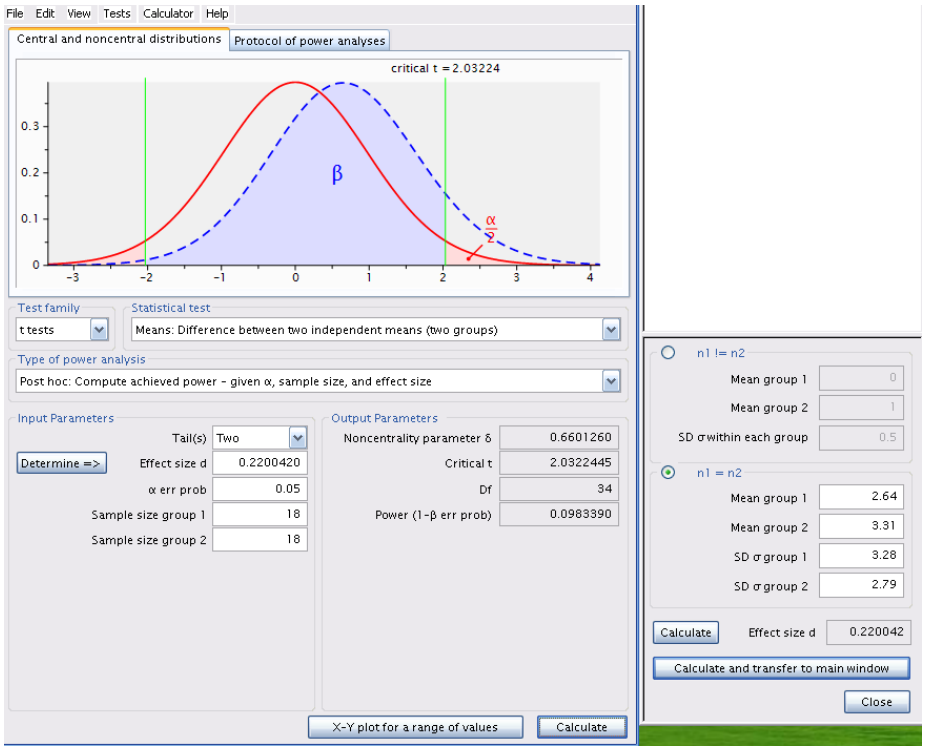


Figure 2.1: Analyse post-hoc avec la taille d'effet estimée

Étudions un peu ce graphique.

- La courbe de gauche, en rouge, correspond à la distribution de la statistique t si H_0 est vraie (i.e si les deux moyennes étaient égales) compte tenu de l'effectif (18 dans chaque région) et des écarts- types observés.
- Les lignes verticales vertes correspondent aux valeurs critiques de t pour une valeur $\alpha = 0.05$ et un effectif total de 36 (2x18).
- Les régions ombrées en rose correspondent aux zones de rejet de H_0. Si Jaynie avait obtenu une valeur de t en dehors de l'intervalle délimité par les valeurs critiques allant de -2.03224 à 2.03224, alors elle aurait rejeté H_0 , l'hypothèse nulle d'égalité des deux moyennes. En fait, elle a obtenu une valeur de t égale à -0.5746 et conclu que la biomasse est la même dans

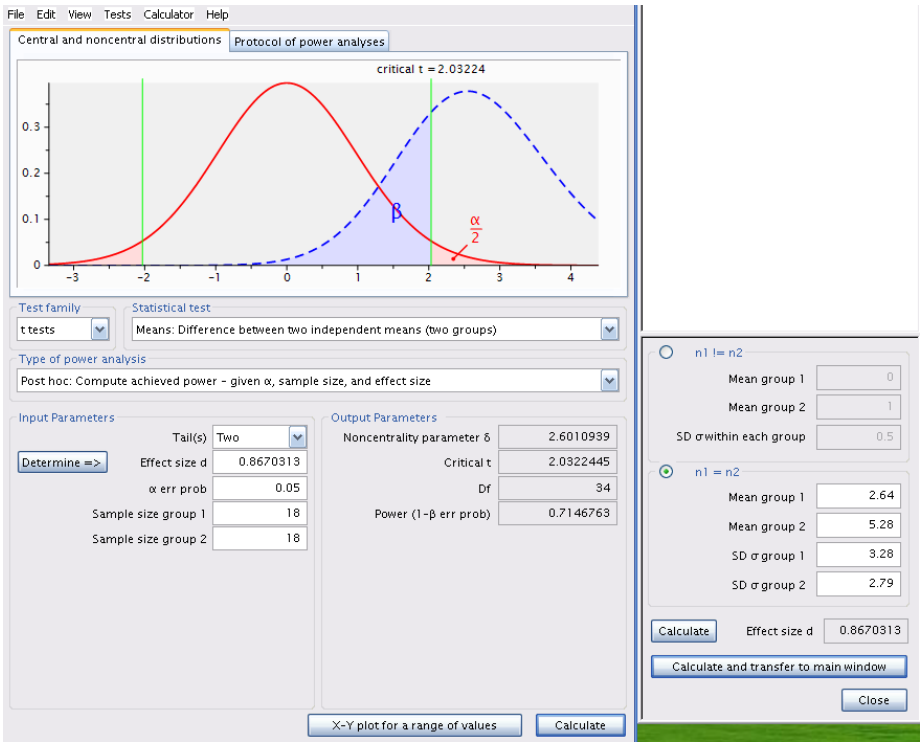
les deux régions.

- La courbe de droite, en bleu, correspond à la distribution de la sta- tistique t si H_1 est vraie (ici H_1 correspond à une différence de biomasse entre les deux régions de $3.33 - 2.64 = 0.69 g/m^2$, compte tenu des écarts-types observés). Cette distribution correspond à ce qu'on devrait s'attendre à observer si H_1 était vraie et que l'on répétait un grand nombre de fois les mesures dans des échantillons aléatoires de 18 ruisseaux des deux régions en calculant la statistique t à chaque fois. En moyenne, on observerait une valeur de t d'environ 0.6.
- Notez que la distribution de droite chevauche considérablement celle de gauche, et une bonne partie de la surface sous la courbe de droite se retrouve à l'intérieur de l'intervalle d'acceptation de H_0, délimité par les deux lignes vertes et allant de -2.03224 à 2.03224. Cette proportion, correspondant à la partie ombrée en bleu sous la courbe de droite et dénoté par β correspond au risque d'erreur de type II qui est d'accepter H_0 quand H_1 est vraie.
- La puissance est simplement $1 - \beta$, et est ici de 0.098339. Donc, si la biomasse différait de $0.69 g/m^2$ entre les deux régions, Jaynie n'avait que 9.8% des chances d'être capable de détecter une différence statistiquement significative à $\alpha = 5$ en échantillonnant 18 ruisseaux de chaque région.

Récapitulons: La différence de biomasse entre les deux régions n'est pas statistiquement significative d'après le test de t. C'est donc que cette différence est relativement petite compte tenu de la précision des mesures. Il n'est donc pas très surprenant que la puissance, i.e. la probabilité de détecter une différence significative, soit faible. Toute cette analyse ne nous informe pas beaucoup.

Une analyse de puissance post hoc avec la taille de l'effet observé n'est pas très utile. On la fera plutôt pour une taille d'effet autre que celle observée quand H 0 est acceptée. Quelle taille d'effet utiliser? C'est la biologie du système étudié qui peut nous guider. Par exemple, en ce qui concerne la biomasse des poissons, on pourrait s'attendre à ce qu'une différence de biomasse du simple au double (disons de 2.64 à 5.28 g/m^2) ait des conséquences écologiques. On voudrait s'assurer que Jaynie avait de bonnes chances de détecter une différence aussi grande que celle-là avant d'accepter ses conclusions que la biomasse est la même entre les deux régions. Quelles étaient les chances de Jaynie de détecter une différence de 2.64 g/m^2 entre les deux régions? G*Power peut nous le dire.

- Changer la moyenne du groupe 2 à 5.28, recalculer la taille d'effet, et cliquer sur Calculate pour obtenir :



La puissance est de 0.71, donc Jaynie avait une chance raisonnable de détecter une différence du simple au double avec 18 ruisseaux dans chaque région.

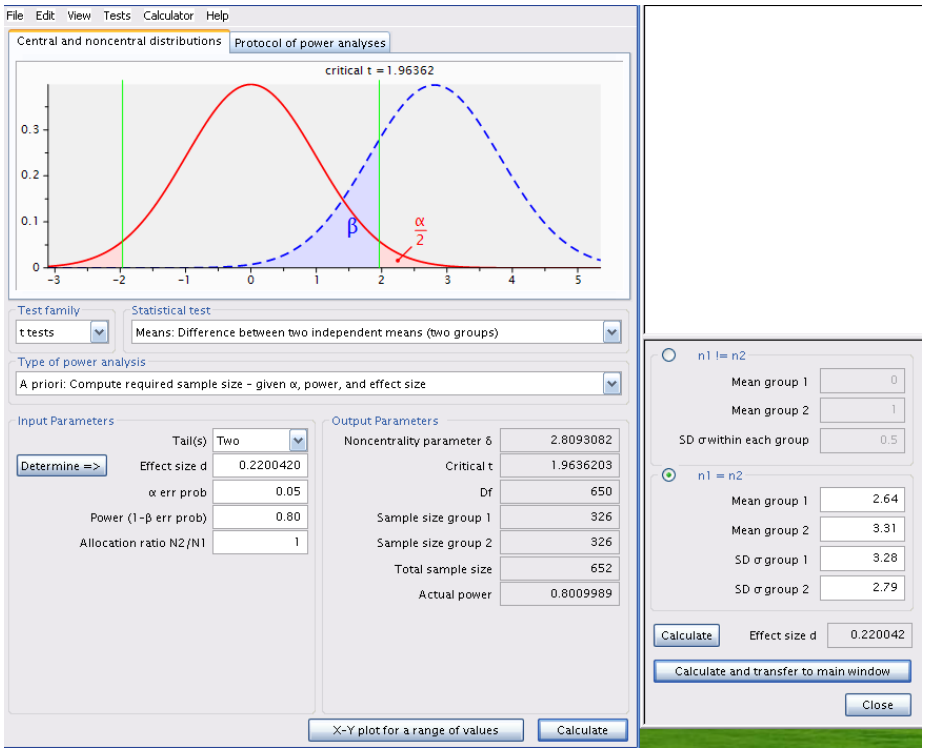
Notez que cette analyse de puissance post hoc pour une taille d'effet jugée biologiquement significative est bien plus informative que l'analyse précédente pour la taille d'effet observée (qui est celle effectuée par défaut par bien des néophytes et de trop nombreux logiciels qui essaient de penser pour nous). En effet, Jaynie n'a pu détecter de différences significatives entre les deux régions. Cela pourrait être pour deux raisons: soit qu'il n'y a pas de différences entre les régions, ou soit parce que la précision des mesures est si faible et l'effort d'échantillonnage était si limité qu'il était très peu probable de détecter même d'énormes différences. La deuxième analyse de puissance permet d'éliminer cette seconde possibilité puisque Jaynie avait 71% des chances de détecter une différence du simple au double.

2.4.2 Analyse à priori

Supposons qu'on puisse défendre la position qu'une différence de biomasse observée par Jaynie entre les deux régions, $3.31 - 2.64 = 0.67 g/m^2$, soit écologiquement signifiante. On devrait donc planifier la prochaine saison d'échantillonnage de manière à avoir de bonnes chances de détecter une différence de cette taille. Combien de ruisseaux Jaynie devrait-elle échantillonner pour avoir 80% des

chances de la détecter (compte tenu de la variabilité observée)?

- Changer le type d'analyse de puissance dans G*Power à **A priori**: **Compute sample size - given α , power, and effect size.** Assurez-vous que les valeurs pour les moyennes et les écarts-type soient celles qu'a obtenu Jaynie, recalculez la taille de l'effet, et inscrivez 0.8 pour la puissance et vous obtiendrez:



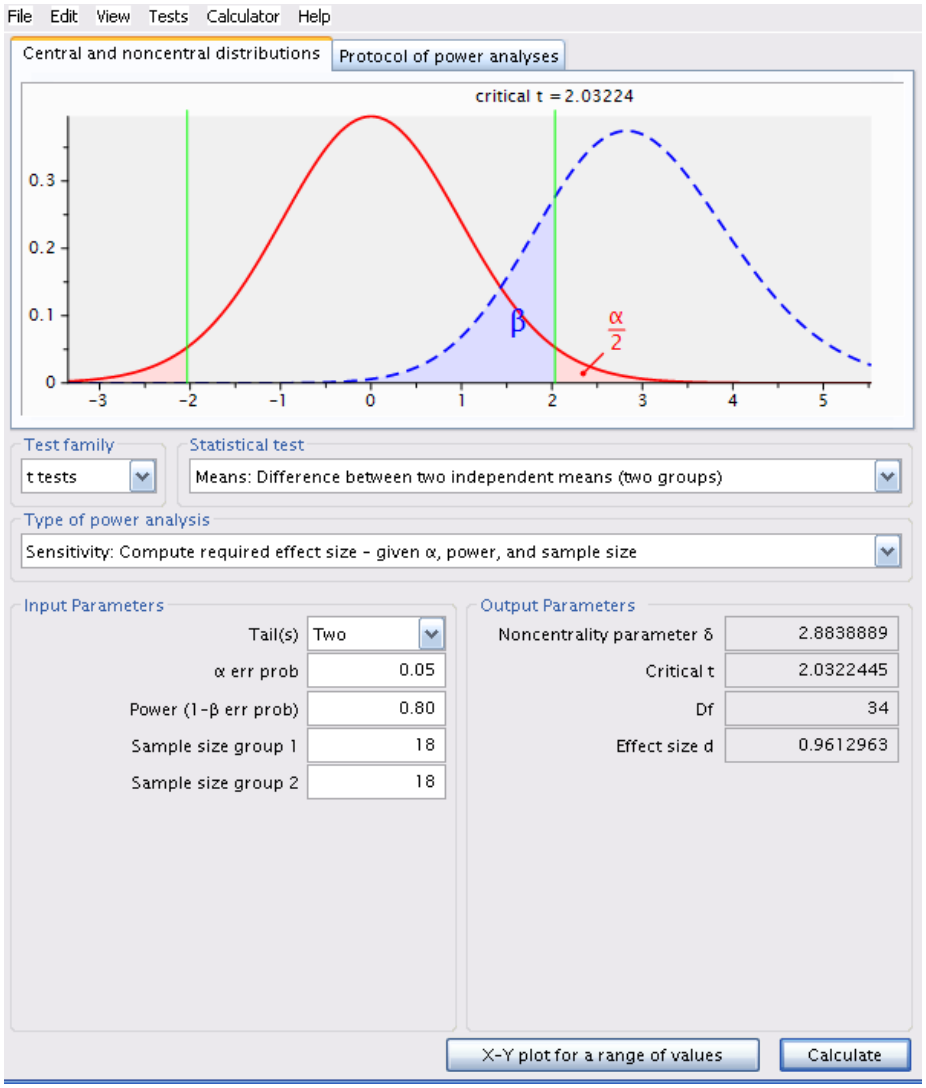
Ouch! Il faudrait échantillonner 326 ruisseaux dans chaque région! Cela coûterait une fortune et exigerait de nombreuses équipes de travail. Sans cela, on ne pourrait échantillonner que quelques dizaines de ruisseaux, et il serait peu probable que l'on puisse détecter une si faible différence de biomasse entre les deux régions. Ce serait vraisemblablement en vain et pourrait être considéré comme une perte de temps: pourquoi tant d'efforts et de dépenses si les chances de succès sont si faibles.

Si on refait le même calcul pour une puissance de 95%, on obtient 538 ruisseaux par région. Augmenter la puissance ça demande plus d'effort.

2.4.3 Analyse de sensitivité - Calculer la taille d'effet détectable

Compte tenu de la variabilité observée, d'un effort d'échantillonnage de 18 ruisseaux par région, et en conservant $\alpha = 0.05$, quelle est la taille d'effet que Jaynie pouvait détecter avec 80% de chances $\beta = 0.2$)?

- Changez le type d'analyse dans G*Power à **Sensitivity**: **Compute required effect size - given α , power, and sample size** et assurez-vous que la taille des échantillons est de 18 dans chaque région. Vous obtiendrez:



La taille d'effet détectable pour cette taille d'échantillon, $\alpha = 0.05$ et $\beta = 0.2$

(ou une puissance de 80%) est de 0.961296. **Attention**, cette valeur est l'indice `d` de la taille de l'effet et est pondérée par la variabilité des mesures. Dans ce cas ci, `d` est approximativement égal à

$$d = \frac{|\bar{X_1}\bar{X_2}|}{\sqrt{\frac{{s_1}^2+{s_2}^2}{2}}}$$

Pour convertir cette valeur de d sans unités en une valeur de différence de biomasse détectable (i.e $|\bar{X_1}\bar{X_2}|$), il suffit de multiplier d par le dénominateur de l'équation.

$$|\bar{X_1}\bar{X_2}| = d * \sqrt{\frac{{s_1}^2 + {s_2}^2}{2}}$$

Donc, avec 18 ruisseaux dans chaque région, pour $\alpha = 0.05$ et $\beta = 0.2$ (une puissance de 80%), Jaynie pouvait détecter une différence de biomasse de $2.93 g/m^2$ entre les régions, un peu plus que du simple au double.

2.5 Points à retenir

- L'analyse de puissance post hoc n'est pertinente que lorsque l'on a accepté l'hypothèse nulle. Il est en effet impossible de faire une erreur de type II quand on rejette H_0.
- Avec de très grands échantillons, on a une puissance quasi infinie et on peut détecter statistiquement de très petites différences qui ne sont pas nécessairement biologiquement significatives.
- En utilisant un critère de signification plus strict (α<0.05) on diminue notre puissance.
- En voulant maximiser la puissance, on augmente l'effort requis, à moins d'utiliser une valeur critique plus libérale ($\alpha > 0.05$)
- Le choix de β est quelque peu arbitraire. On considère que $\beta = 0.2$ (puissance de 80%) est relativement élevé.

2.6 Exercice sur la puissance

Les larves de mouches noires (Diptera: Simuliidae) ont été échantillonnées en février à l'émissaire de deux lacs des Cantons de l'Est (lacs Orford et Lovering). La longueur de chaque larve a été mesurée. Les données sont dans le fichier simulies.RData La relation entre la longueur (L, en mm) et la masse (M, en μg) pour l'espèce dominante (*P. mixtum/fuscum*) est:

$$M = 1.36L^3.05$$

1. Calculer la masse moyenne et l'écart-type à chaque site.
2. En utilisant la masse moyenne de *P. mixtum/fuscum* à Lovering comme référence et les écarts-types observés aux 2 sites, calculer la puissance d'un test de t bilatéral pour moyennes indépendantes

   a) si la différence de masse est de 5 μg, $\alpha = 0.05$, et qu'on échantillonnait 100 larves à chaque site
   b) si la différence de masse est de 20 μg, $\alpha = 0.05$, et qu'on échantillonnait 100 larves à chaque site
   c) si la différence de masse est de 50μg, $\alpha = 0.05$, et qu'on échantillonnait 100 larves à chaque site
   d) Comment est-ce que la puissance varie avec la taille de l'effet?
3. Calculer la taille d'échantillon requis pour pouvoir détecter, par un test de t bilatéral pour moyennes indépendantes en tenant compte des écarts-types observés, une différence de 50 μg entre les moyennes
   a) avec une puissance de 80% et $\alpha = 0.05$
   b) avec une puissance de 80% et $\alpha = 0.001$
   c) avec une puissance de 95% et $\alpha = 0.05$
   d) Comment est-ce que la taille d'échantillon requise varie avec α et β?
4. Calculer la taille d'effet détectable (d) par un test de t bilatéral pour moyennes indépendantes, compte tenu des écarts-types observés
   a) avec une puissance de 80%, $\alpha = 0.05$, et des mesures sur 10 larves de chaque site
   b) avec une puissance de 80%, $\alpha = 0.05$, et des mesures sur 200 larves à chaque site
   c) avec une puissance de 80%, α =0.05, et des mesures sur 20 larves d'un site et sur 380 larves au second site
   d) Comment est-ce que la taille d'effet détectable dépend de la taille d'échantillon dans les 2 groupes?
5. Calculer la différence de masse, en μg, qui est détectable d'après vos estimés de la taille minimale d'effet détectable à 4a, b, et c.