

# Travaux Pratiques RECH 601

David Sanchez, Mathieu de Bony

5 mai 2020

**But du TP :** Le but du TP est d'étudier les données que vous venez de recevoir d'un collègue astronome ayant été prises par l'observatoire Cherenkov CTA. Pour ce faire vous allez passer par les principales étapes effectuées par les astrophysiciens travaillant dans le domaine des gammas de haute énergie, lors d'une analyse de données. Bien évidemment, le travail ici est simplifié et certains des scripts sont déjà prêts à être utilisés. Nous ne nous concentrerons pas sur les aspects techniques mais sur les concepts mathématiques et physiques mises en jeux. Il faudra donc apporter un soin tout particulier aux explications apportées lors des réponses aux questions.

**Rendu :** Ce module sera évalué par le compte rendu du TP

Un rapport décrivant le travail effectué lors des séances de TPs. Il devra contenir les points suivants :

- Un contexte introductif décrivant l'astronomie gamma et Cherenkov
- Les réponses aux questions posées dans l'énoncé de TP
- Une explication des notions utilisées (spectre, courbe de lumière...)
- Une présentation des résultats avec des discussions physiques
- Une conclusion récapitulant les principales informations vues et utilisées
- Tous les fichiers de résultats (.fits). La liste est fournie à la fin de l'énoncé

Le rapport de TP ne doit pas juste être une liste de réponses aux questions mais permettre au lecteur de comprendre votre travail, les étapes et les problèmes que vous avez rencontrés.

**Informations complémentaires :**

- Les premières lignes des encadrements sont les commandes à effectuer. Les lignes suivantes sont les entrées que l'on doit donner pour faire fonctionner la commande.
- Les [ ] dans les encadrés sont les valeurs que le code prend en compte si rien n'est écrit à côté. Pour le TP, les informations à écrire sont celles entre [ ].

L'outil de visualisation des cartes est DS9. C'est un outils de la NASA dont le fonctionnement est décrit en annexe.

L'analyse des données que nous vous proposons se fera avec des outils de la collaboration CTA nommé `ctools`. `ctools` est un logiciel développé pour l'analyse scientifique des données de CTA. Dans ce TP, le logiciel sera utilisé en ligne de commande et permet de sélectionner les données, calculer les paramètres de l'observation (exposition, temps, etc..) et d'extraire les caractéristiques des objets étudiés (flux, etc...).

# Chapitre 1

## Première partie

### Étape 1. Se Familiariser avec l'astronomie Cherenkov

Le but de cette partie est de comprendre le fonctionnement des instruments qui ont pris les données. Vous avez suivi une formation préliminaire qui avait pour objectif de vous initier à l'astronomie gamma et les méthodes d'analyse associées. Les prochaines parties seront les applications de ce que vous avez vu durant cette formation. N'hésitez pas à poser des questions sur les points que vous n'avez pas compris.

**Décrire le fonctionnement de l'astronomie gamma et Cherenkov ainsi que la méthode d'analyse (avantages et inconvénients) (15 lignes maximum).**

### Étape 2. Commencer une analyse

Les données que vous avez reçues sont situées dans un fichier appelé **events\_1.fits**. Ce fichier contient les événements de la prise de données et sera utilisé dans la suite du projet pour effectuer les analyses.

La première étape consiste à sélectionner les événements qui proviennent d'une région du ciel qui nous intéresse, dans la gamme en énergie désirée ainsi que dans un temps d'observation voulu. On va ensuite construire une carte du ciel avec les événements pour voir s'il y a une accumulation à une certaine position, signant la présence d'une source.

La région qui nous intéresse ici est celle de la radio galaxie Centaurus A dont on parlera dans les cours. Ses coordonnées dans le ciel sont (en céleste) :

- RA : 201.36
- Dec : -43.0

Nous utiliserons ses coordonnées pour le centre des cartes.

### Carte des événements et positions des sources

Il est possible de voir rapidement les événements mesurés par notre instrument avec des logiciels de visualisation. Cependant il est nécessaire de transformer la carte brute des événements avant de la visualiser avec DS9. Pour créer une carte du ciel contenant les positions des événements, il faut exécuter la commande `ctskymap`. Il vous suffit d'écrire la commande `ctskymap` dans votre terminal. Les autres lignes seront automatiquement demandées par la console. Remplacez-les de la façon suivante.

```

ctskymap
Input event list or observation definition XML file [events.fits] events_1.fits
Coordinate system (CEL -> celestial, GAL -> galactic) (CEL|GAL) [CEL]
Projection method (AIT|AZP|CAR|GLS|MER|MOL|SFL|SIN|STG|TAN) [CAR]
First coordinate of image center in degrees (RA or galactic l) (0-360) [83.63] 201.36
Second coordinate of image center in degrees (DEC or galactic b) (-90-90) [22.01] -43.0
Image scale (in degrees/pixel) [0.02]
Size of the X axis in pixels [200] 300
Size of the Y axis in pixels [200] 300
Lower energy limit (TeV) [0,1] 0.05
Upper energy limit (TeV) [100,0] 20.0
Background subtraction method (NONE|IRF|RING) [NONE]
Output skymap file [skymap.fits]

```

Dans cette commande, on crée une carte en coordonnées célestes et une projection cartésienne. La carte étant relativement petite ceci ne pose pas de problème. Le centre de la carte est définie par First coordinate et Second coordinate. Ensuite, on défini la pixellisation de la carte et la gamme en énergie. Notre carte fera ainsi 300 pixels par 300 pixel. Chaque pixel fera 0.02 degrés. Seul les événements entre 0.05 TeV et 20 TeV sont gardés.

Le nouveau fichier va s'appeler *skymap.fits*. Il est maintenant possible d'ouvrir le fichier avec DS9 en suivant la procédure décrite en Annexe A.

#### **Décrire rapidement la carte et la présenter dans votre rapport.**

La source présente au centre de la carte est Centaurus A, un noyau actif de galaxie bien connu. Elle servira de vérification des résultats tout au long des travaux pratiques. . Comme vous pouvez le voir sur l'image de DS9, une autre source est présente dans le champ de vue. Celle-ci est nouvelle, elle n'avait encore jamais été détectée dans le domaine d'énergie de CTA. Son nom dans la suite du TP sera source2. Vous décidez donc d'étudier cette source.

La première étape consiste à trouver les positions des sources. Pour ce faire vous allez devoir à la main déterminer la position de cette source à l'aide de DS9.

Une autre façon plus précise et correcte donnant les erreurs d'estimations. Le script intitulé *FindPosition.py* permet de trouver les positions des sources ainsi que les erreurs associées. Ces ajustements se font en assumant une forme de gaussienne et cherchant la meilleure position et taille de ce point pour représenter les données (se reporter à la littérature sur les ajustements par minimisation de chi2 pour plus d'informations).

#### **Donner les coordonnées trouvées pour les deux sources en sexagésimal et en degrés. La conversion pourra se faire grâce à l'utilisation du site internet :**

<http://www.robertmartinayers.org/tools/coordinates.html>

**Vous pouvez aussi donner le résultat en coordonnées galactiques.**

**Décrire pourquoi l'hypothèse de source de type point est favorisée.**

**Tracer la carte du ciel et représenter la position de l'ajustement du script (avec une région DS9). Le rayon de la source sera assimilé à l'écart type de la gaussienne donnée par l'ajustement. Une explication de la procédure de création de région dans DS9 est donnée en annexe.**

Afin d'aller un peu plus loin, nous allons utiliser DS9 pour quelques opérations. Dans l'onglet *Analysis*, allez dans *Smooth Paramters* et réglez les paramètres comme dans la figure 1.1. Enfin dans *Analysis*, cliquez sur *Smooth* et *Contours* (dans cet ordre, figure 1.2) .

#### **Que notez vous ?**

La carte est "floutée" avec un noyau gaussien qui permet de mieux visualiser les structures. Les contours donnent sont des isocontours comme sur une carte géographique.

**Le fait que le fond (matérialisé par les grands cercles), ne soit pas centré sur la source ne vous interroge pas ?**

**Charger la région DS9 "obs.reg". elle représente le centre de pointé de notre instrument. Commentez en vous aidant de votre cours.**

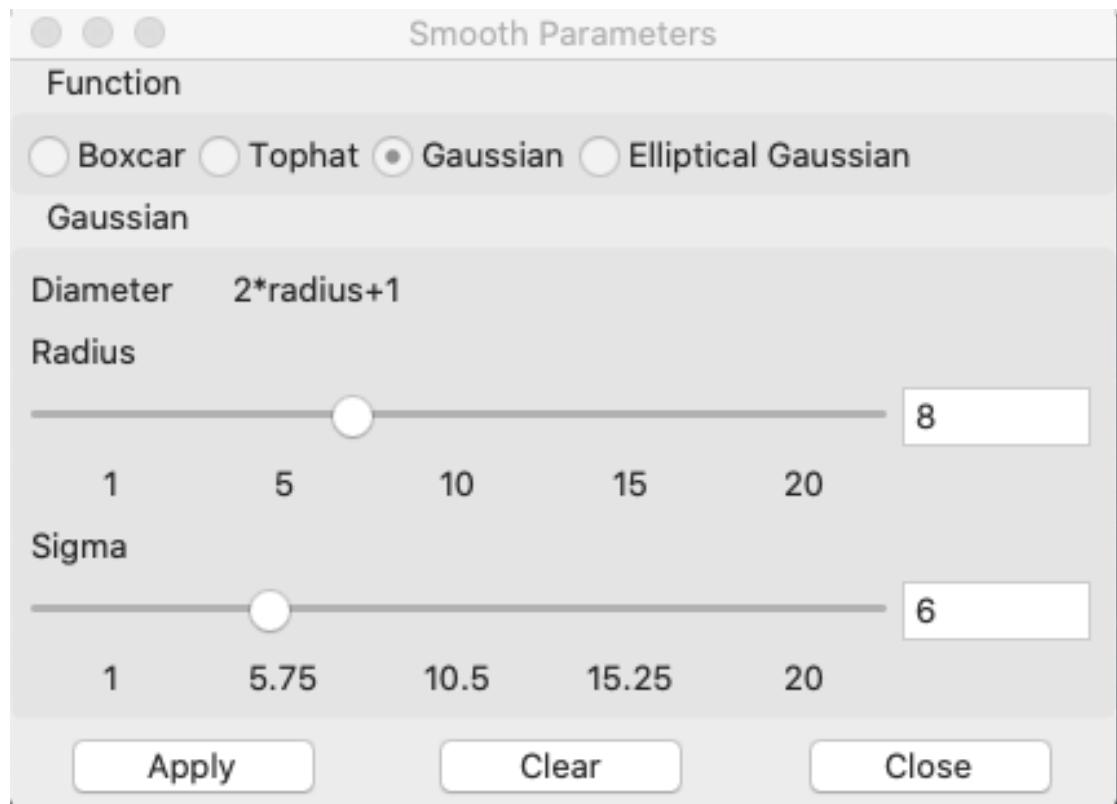


FIGURE 1.1 – *Smooth Paramters* à utiliser et configuration

### Évolution de la carte d'événements en fonction de l'énergie

Connaître l'évolution de la carte en changeant l'énergie est aussi un moyen intéressant d'obtenir des informations sur la source. Pour tracer les cartes d'événements à différentes énergies il faut utiliser la fonction ctbin de ctools. Nous allons séparer l'observation en 4 d'intervalles (bins) en de façon logarithmique. Vous devriez obtenir les mêmes énergies que celles du Tableau 1.1 en prenant 4 bins en énergie dans ctbin. Pour utiliser cette fonction vous devez utiliser les commandes suivantes :

```
ctbin
Input event list or observation definition XML file [events.fits] events_1.fits
Coordinate system (CEL - celestial, GAL - galactic) (CEL|GAL) [CEL]
Projection method (AIT|AZP|CAR|GLS|MER|MOL|SFL|SIN|STG|TAN) [CAR]
First coordinate of image center in degrees (RA or galactic l) (0-360) [83.63] 201.36
Second coordinate of image center in degrees (DEC or galactic b) (-90-90) [22.51] -43.0
Image scale (in degrees/pixel) [0.02]
```

Limite basse en énergie (TeV)	Limite haute en énergie (TeV)
0.05	0.223
0.223	1
1	4.47
4.47	20

TABLE 1.1 – bins en énergie pour tracer le spectre en énergie

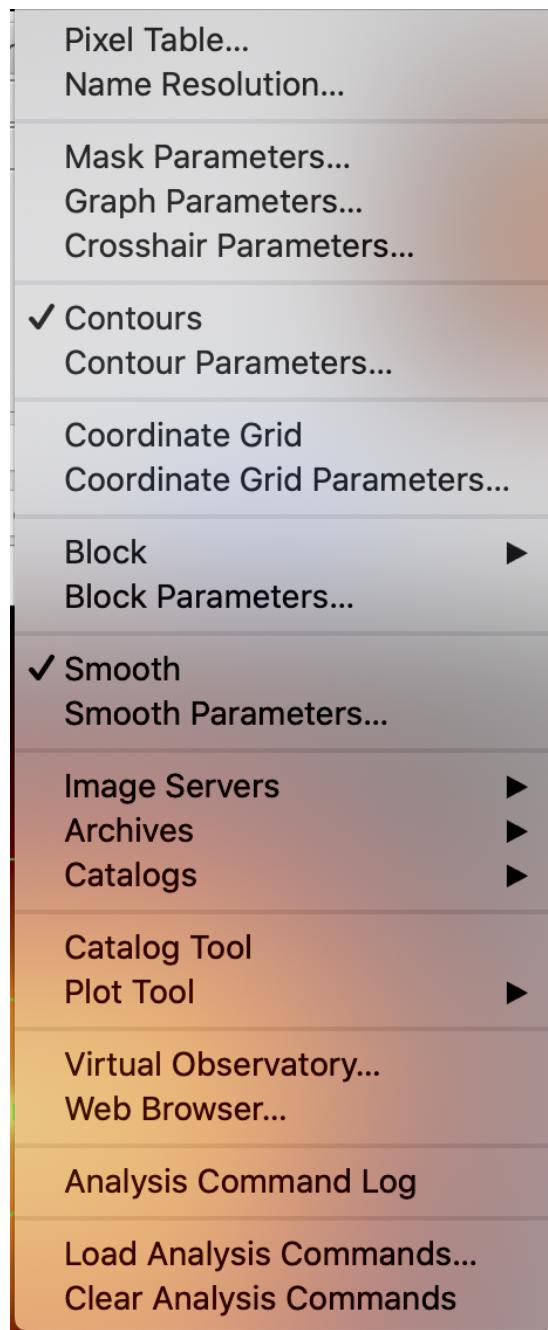


FIGURE 1.2 – Configuration à utiliser

```
Size of the X axis in pixels [200] 300
Size of the Y axis in pixels [200] 300
Algorithm for defining energy bins (FILE|LIN|LOG) [LOG]
Lower energy limit (TeV) [0.1] 0.05
Upper energy limit (TeV) [100.0] 20.
Number of energy bins (1–200) [20] 4
```

Cette carte est peu différente de celle d'avant, a ceci près que les données ont été découpées en 4 bin en énergies, logarithmiquement espacé.

Il est possible de voir la nouvelle carte d'événements dans chacun des bins en énergie. La Figure 1.3 montre un exemple de fenêtre qui sera utilisée pour afficher les cartes d'événements à différentes énergies.

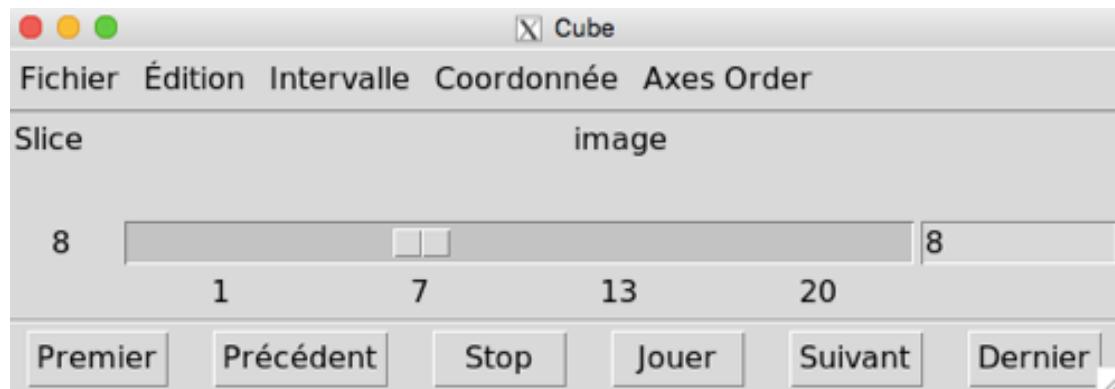


FIGURE 1.3 – Fenêtre de choix du bin en énergie.

**Tracer les cartes d'événements pour les différents bins en énergie donnés dans le tableau 1.1.**

**Comment interpréter ces observations ?**

**A partir de quelle énergie, les sources disparaissent ?**

**Comment évolue les tailles des sources ?**

**Comment change le fond ?**

## Carte de significativité

Tracer des cartes d'événements permet d'obtenir des informations. Mais celles-ci ne sont pas suffisantes pour savoir si une source est détectée. Car au-delà du nombre d'événements dans le ciel, la réponse de l'instrument et les conditions de prise de données doivent aussi être pris en compte. De plus, il est nécessaire de calculer la probabilité que le signal détecté soit un vrai signal. Toutes ces considérations sont prises en compte dans le calcul de la significativité de détection (voir les formations théoriques dédiées pour des explications supplémentaires).

Nous n'entrerons pas pour l'instant dans le détail de l'explication de la significativité dans ce document. Il est simplement important de définir ici la significativité de détection comme la probabilité que le signal obtenu soit un vrai signal et non une fluctuation statistique. La référence seuil, c'est-à-dire la valeur pour laquelle on parle de détection est  $5\sigma$ .

La commande pour tracer la carte de détection pour une liste d'événement est `ctskymap`. Il s'agit de la même commande que précédemment, mais ici la valeur du paramètre *Background subtraction method* est maintenant *IRF*. Ainsi, les réponses de fonction de l'instrument (qui décrivent le comportement de l'instrument) seront utilisées pour évaluer le fond et calculer la significativité à différentes positions de la carte. Un exemple de carte de significativité est donnée en figure 1.4.

```
ctskymap
Input event list or observation definition XML file [events_1.fits]
Coordinate system (CEL - celestial, GAL - galactic) (CEL|GAL) [CEL]
Projection method (AIT|AZP|CAR|GLS|MER|MOL|SFL|SIN|STG|TAN) [CAR]
First coordinate of image center in degrees (RA or galactic l) (0-360) [201.36]
```

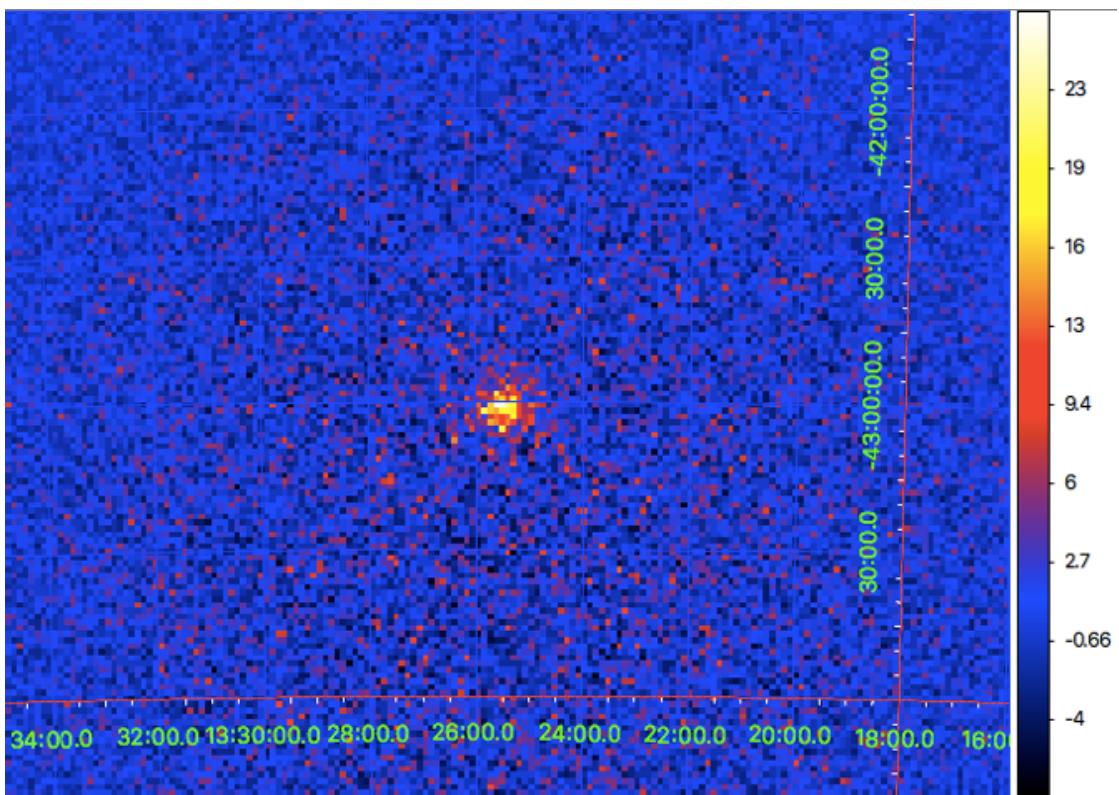


FIGURE 1.4 – Carte de significativité de détection du Crabe.

```

Second coordinate of image center in degrees (DEC or galactic b) (-90–90) [-43.0]
Image scale (in degrees/pixel) [0.02]
Size of the X axis in pixels [300]
Size of the Y axis in pixels [300]
Lower energy limit (TeV) [0.05]
Upper energy limit (TeV) [20.0]
Background subtraction method (NONE|IRF|RING) [NONE] IRF
Calibration database [prod2]
Instrument response function [South_0.5h]
Output skymap file [skymap.fits] skymap_significance.fits

```

La seule différence ici est le paramètre *Background subtraction method*. Celui-ci est mis à *IRF* pour Instrumental Response Function. *ctools* utilise une description de l'instrument obtenue par simulation pour estimer le fond de rayons cosmiques et le soustraire. Il ne reste plus alors que les sources pour calculer leur significativité (voir la suite).

**Donner la carte de significativité de détection de la région. Quelles sont les valeurs de significativité de détection des 2 sources ? (On assimilera dans cette question, la significativité au maximum).**

## Chapitre 2

# Calcul de significativité de la source

### Déterminer une significativité de détection à la main

Durant le premier TP, une carte de significativité de détection a été calculée. Nous allons maintenant voir comment calculer la significativité à la position de la source. L'idée de base est de compter le nombre de photon d'une source et aussi le nombre d'événements de fond. c'est une technique plutôt standard et appliquée souvent en physique. La façon de faire a été expliquée en cours lors de la formation théorique préliminaire. Il est nécessaire de compter les événements dans la région ON et ceux dans les régions OFF.

Nous allons utiliser le logiciel DS9 pour dans un premier temps estimer à la main le nombre d'événements dans ces régions ON et OFF. Puis vous allez utiliser la fonction `lima()` qui va vous donner la significativité de détection. Les nombres d'événements ON et OFF peuvent être compilés à partir du fichier `skymap.fits`.

#### Source 1 : Centaurus A

Pour cela, il faut déterminer une région ON qui est centrée sur la source. Définissez une région dans DS9 de taille 0.1 centrée sur la source (voir annexe). **Déterminer le nombre  $N_{on}$ .**

Il faut ensuite déterminer des régions OFF, symétriques par rapport au pointé d'observation (donné dans le fichier `région obs.reg`). Placer 7 régions OFF comme sur la figure ???. **Déterminer le nombre  $N_{off}$  et présenter dans votre rapport une images avec les régions utilisées.**

L'excès est donné par la relation  $N_{excess} = N_{on} - \alpha \times N_{off}$ .  $\alpha$  est dans notre cas (simplifié) le rapport du nombre de région ON sur le nombre de régions OFF choisies.

**Déterminer le nombre  $N_{excess}$ .**

**Puis utilisez ces valeurs dans le script `SimpleSignificance` pour calculer la valeur de significativité de la source Centaurus A.**

Nous vous conseillons de sauvegarder les régions dans un fichier texte pour la suite du TP.

#### Source 2

Faites la même chose pour la source 2. *Attention, les régions doivent être placées de façon symétrique par rapport au pointé d'observation.* Nous vous conseillons de sauvegarder les régions dans un fichier texte pour la suite du TP.

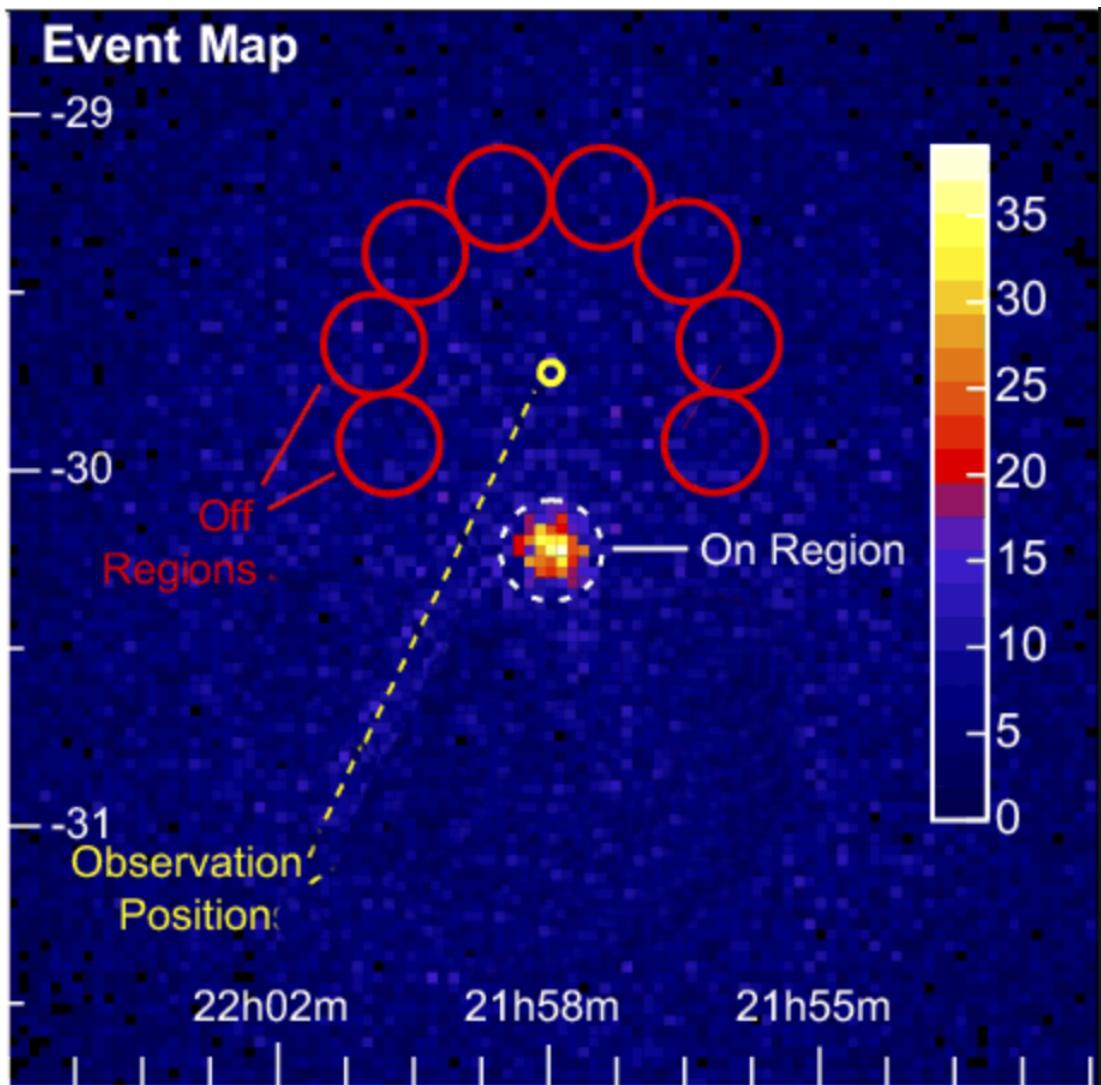


FIGURE 2.1 – Schéma des régions pour estimer la significativité d'une source.

## Estimation de la significativité de détection grâce à un script

Pour ce faire utiliser la commande `cspagen`. Cette ligne de commande va faire une analyse ON-OFF des événements de la même façon que ce que vous avez vu durant la formation préliminaire

```
cspagen
Input event list or observation definition XML file [events_1.fits]
Calibration database [prod2]
Instrument response function [South_0.5h]
Input model definition XML file (if NONE, use point source) [NONE]
Algorithm for defining energy bins (FILE|LIN|LOG) [LOG]
Start value for first energy bin in TeV [0.1] 0.05
Stop value for last energy bin in TeV [100.0] 20.
Number of energy bins [120] 30
Stack multiple observations into single PHA, ARF and RMF files? [no]
Output observation definition XML file [onoff_obs.xml]
Output model definition XML file [onoff_model.xml]
```

```

Method for background estimation (REFLECTED|CUSTOM) [REFLECTED]
Coordinate system (CEL - celestial, GAL - galactic) (CEL|GAL) [CEL]
Right Ascension of source region centre (deg) (0-360) [83.63] 201.36
Declination of source region centre (deg) (-90-90) [22.01] -43.0
Radius of source region circle (deg) (0-180) [0.2] 0.1

```

En effectuant les commandes précédentes vous obtiendrez un fichier appelé *onoff\_obs.xml*. Ce fichier contient les noms des fichiers nouvellement créés. Un exemple du contenu de ce fichier est montré dans le cadre suivant.

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<observation_list title="observation_list">
  <observation name="" id="" instrument="CTAOnOff" statistic="cstat">
    <parameter name="Pha_on" file="onoff_pha_on.fits"/>
    <parameter name="Pha_Off" file="onoff_pha_off.fits"/>
    <parameter name="Arf" file="onoff_arf.fits"/>
    <parameter name="Rmf" file="onoff_rmf.fits"/>
  </observation>
</observation_list>

```

Le fichier *onoff\_pha\_on.fits* contient les événements de tous les bins en énergie pour la région ON.

Idem pour les régions OFF avec le fichier *onoff\_pha\_off.fits*.

*onof\_arf.fits* contient la réponse spectrale des instruments (ces informations sont extraites des IRF). Les données les plus importantes ici sont la surface effective pour la détection de gamma et les taux de fond.

*onoff\_rmf.fits* contient la partie restante des IRFs (ces informations ne sont pas importantes pour notre étude).

L'outil crée deux fichiers région *onoff\_on.reg* et *onoff\_off.reg* utilisé pour les régions ON et OFF. **Chargez les avec DS9 sur votre carte du ciel et comparez les avec les vôtres. Commenter.**

Après avoir produit les résultats de l'analyse, il est possible de compter les événements ON, OFF et alpha. Le script *Significance\_cspagen.py* va vous donner la valeur de la significativité de détection.

**Noter la valeur obtenue et la comparer à celle obtenue précédemment. Quelle méthode vous semble la plus précise ? Et pourquoi ?**

## Chapitre 3

# Analyse Spectrale et temporelle

Vous allez maintenant trouver les caractéristiques intrinsèques de la source à savoir son spectre en énergie et sa courbe de lumière. Ces deux informations sont les principales utilisées pour définir et étudier les sources.

### Calcul des points de flux à la main

Après les cartes d'événements et de significativité, l'étude des émissions des sources en fonction de l'énergie permet d'obtenir des informations sur les mécanismes d'émission présents.

Pour calculer les points d'un spectre, il faut différentes grandeurs physiques dépendantes de l'instrument et des conditions d'observation (vues durant la formation) : la surface efficace, le temps d'exposition, l'énergie du bin et l'excès.

- Le temps d'exposition est le temps de la simulation, dans vos données il est de 7200s.
- La surface efficace est donnée par les instruments eux-mêmes elle est donnée en  $m^2$ .  
La Figure ?? est la surface efficace en fonction de l'énergie des particules incidentes. Les données ont été simulées avec un temps d'observation d'environ 2h. La courbe à prendre en compte est la plus proche. Il s'agit donc de celle de 5h.
- Les énergies moyennes sont les moyennes géométriques des bins du tableau 1.1.
- L'excès dans chaque bin est à calculer comme précédemment.

**Pour les bins en énergie présentés dans le tableau 1.1, calculez les points des spectres pour les 2 sources grâce à la formule suivante :**

$$F = \frac{N_{excess}}{T_{exp}A_{eff}} [\text{photons/temps/surface}]$$

Pour compter les événements ON et OFF, vous devrez utiliser les cartes d'événements obtenues précédemment pour ces bins en énergie. Les erreurs en flux associées à ces points doivent aussi être calculées.

**Nous ne prendrons ici que les erreurs provenant de l'énergie pour le calcul des erreurs des points de spectre. Quelle autre source d'erreur peut exister ?**

Les résultats seront donnés sous forme d'un tableau compilant les nombres d'événements ON, OFF, les excès, les surfaces efficaces, les énergies moyennes, les flux et erreurs sur le flux dans les bonnes unités

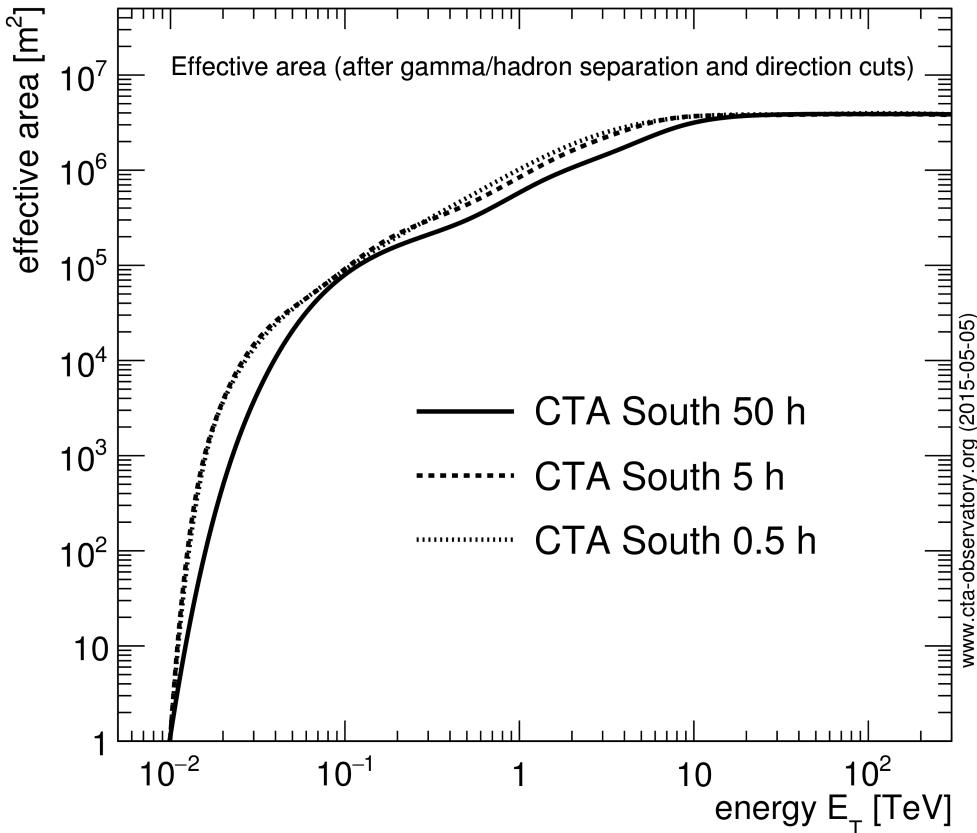


FIGURE 3.1 – Surface efficace en fonction de l'énergie pour le réseau CTA sud.

## Calcul des points de spectre avec un script

Comme vous pouvez vous en douter, il existe déjà des outils fonctionnant pour tracer directement des spectres. Une fonction appelée *csspec* est présente dans ctools pour effectuer ceci. Utiliser cette commande pour tracer les spectres des 2 sources.

```
csspec
Input event list, counts cube, or observation definition XML file [events_1.fits]
Calibration database [prod2]
Instrument response function [South_Sh]
Input model definition XML file [SteadyFoV.xml]
Source name [GenA ou 1ES1312 -423]
Spectrum generation method (SLICE|NODES|AUTO) [AUTO]
Binning algorithm (LIN|LOG|FILE) [LOG]
Lower energy limit (TeV) [0.05]
Upper energy limit (TeV) [20.0]
Number of energy bins [4]
Output spectrum file [spectrum.fits]
```

On utilise ici un fichier xml qui donne le modèle de notre ciel. On y trouve 2 sources (Centaurus A et source2) et aussi un model pour le fond (CTABackgroundModel).

Cette commande donne un fichier *spectrum.fits* qui contient toutes les informations sur le spectre obtenu.

Tracez le résultat avec le script *show\_spectrum.py* en utilisant la ligne de commande suivante :

```
python show_spectrum.py
```

Un exemple de spectre est donné dans la figure ?? . En plus des points des valeurs, les incertitudes de ces points en énergie et flux sont aussi tracées.

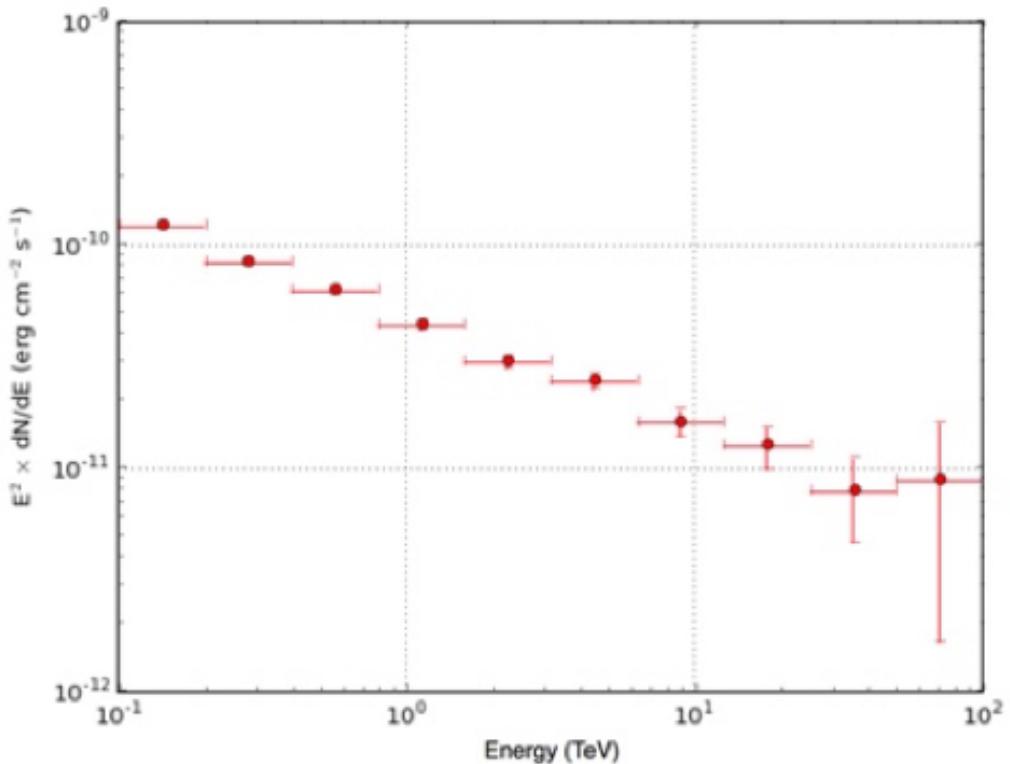


FIGURE 3.2 – Spectre pour une observation du Crabe.

**Ces résultats sont en accord avec ceux provenant des calculs à la main ? Si oui pourquoi ? Si non pourquoi ?**

### Calcul des points de la courbe de lumière à la main

D'autres informations importantes à propos des sources concernent la variabilité de celles-ci. L'évolution des émissions au cours du temps permet aussi d'obtenir de nombreux indices sur les mécanismes dans les sources. En ce qui concerne la courbe de lumière, la façon de procédé est à peu près la même. Vous devez dans un premier temps tracer les points de cette courbe de lumière pour différents moments (voir tableau ?? ).

Avant de pouvoir calculer les points de la courbe de lumière il faut produire les fichiers d'évènements dans les bins en temps. Pour faire cette sélection la commande ctselect va être utilisée. Le *Start time* et le *Stop time* doivent être sous la forme suivante « 2021-01-01T00 :10 :00 ». Les noms des fichiers produits devront être sous la forme `selected_events_début_fin.fits`. Avec *début* et *fin* les valeurs des bins en temps.

Début	Fin
662774400 (T0+0s)	662776800.0 (T0+2400s)
662776800.0 (T0+2400s)	662779200.0 (T0+4800s)
662779200.0 (T0+4800s)	662781600 (T0+7200s)

TABLE 3.1 – Intervalles de temps des bins de la courbe de lumière. T0 représente le début de la prise de données.

---

```

ctselect
Input event list or observation definition XML file [events_1.fits]
Radius of ROI around pointing or specified RA/DEC (degrees) (0–180) [15]
Start time (UTC string, JD, MJD or MET in seconds) [2021-01-01T11:58:51] 662774400
Stop time (UTC string, JD, MJD or MET in seconds) [2021-01-01T12:38:51] 662781600
Lower energy limit (TeV) [0.05]
Upper energy limit (TeV) [20]
Output event list or observation definition XML file [selected_events.fits]

```

---

Les événements ON et OFF seront comptés de la même façon que présentée durant la séance précédente.

**Utiliser la même formule du calcul des points que pour les spectres. Comment caractériser le flux de source en fonction du temps ?**

## Calcul des points de la courbe de lumière avec un script

De la même façon que pour le spectre, des outils déjà existant peuvent être utilisé pour tracer plus facilement une courbe de lumière. Le script `cslightcrv` va prendre en entrée le fichier `events_1.fits` et calcule la courbe de lumière pour chaque bin.

---

```

cslightcrv
Input event list or observation definition XML file [events_1.fits]
Calibration database [prod2]
Instrument response function [South_0.5h]
Input model definition XML file [SteadyFoV.xml]
Algorithm for defining time bins (FILE|LIN|GTI) [LIN]
Lightcurve start time (UTC string, JD, MJD or MET in seconds) [MET 662774400]
Lightcurve stop time (UTC string, JD, MJD or MET in seconds) [MET 662781600]
Number of time bins (1–10000) [3]
Analysis method (3D|ONOFF) [ONOFF] 3D
Source name [CenA]
Lower energy limit of events (TeV) [0.05]
Upper energy limit of events (TeV) [20]
Number of energy bins for binned (0=unbinned for 3D analysis only) (0–100) [20]
Coordinate System (CEL|GAL) [CEL]
Projection method (AIT|AZP|CAR|GLS|MER|MOL|SFL|SIN|STG|TAN) [CAR]
First coordinate of image/source region center in degrees (RA or galactic l) (0–360) [201.365063]
Second coordinate of image/source region center in degrees (DEC or galactic b) (-90–90) [-43.019113]
Size of the X axis in pixels [200]
Size of the Y axis in pixels [200]
Pixel size (deg/pixel) [0.02]
Output light curve file [lightcurve.fits]

```

---

Ici on fait une analyse un peu particulière et on entrera pas dans les détails. La courbe de lumière peut être tracée avec le script `lightcurve.py`.

Un exemple de courbe de lumière est donné dans la figure ??.

**Comparer avec vos résultat. Commentaires sur les sources et les méthodes.**

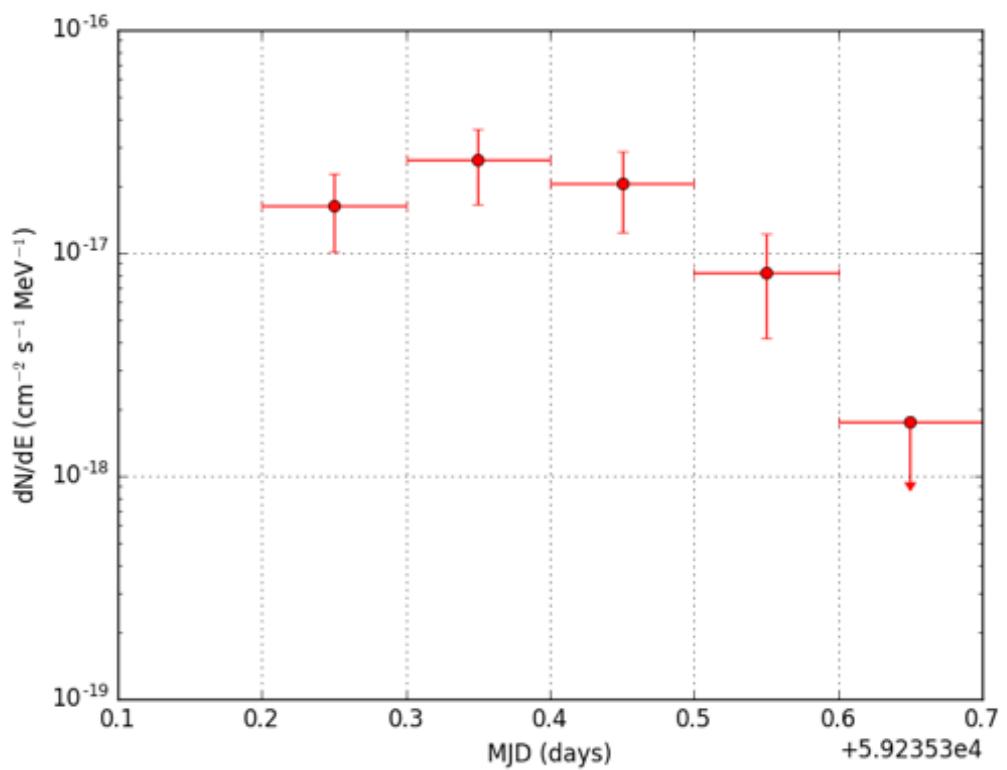


FIGURE 3.3 – Courbe de lumière pour une source quelconque.

## Chapitre 4

# Aller plus loin

Cette section n'est pas obligatoire et donné seulement si vous souhaitez allez plus loin.

### 4.1 Ajuster les points du spectre

**En reprenant les points obtenus par le script précédemment, faire un ajustement des données et donner les paramètres du fit. L'ajustement des données se fera avec une loi de puissance définie par l'équation suivante :**

$$F = F_0 \left(\frac{E}{E_0}\right)^{-g}$$

ou  $g$  est l'indice spectral de la source. Le script intitulé *FitData.py* permet de faire un ajustement des données. Il nécessaire de remplir les variables  $x$ ,  $y$  et  $dy$  avec les valeurs obtenues précédemment.

**Donner les valeurs des ajustements provenant du script.** Un source dure possède un indice spectral plus petit que 2, sinon elle est dite molle. **Qu'en est-il pour nos deux sources ?**

### 4.2 Déterminer la source inconnu

Afin de savoir quelle source inconnue a été détectée, rendez vous sur le site <http://simbad.u-strasbg.fr/simbad/sim-fcoo>. C'est le portail SIMBAD de l'université de Strasbourg qui accueille une base de données des objets du ciel.

**Entrer les coordonnées de la source inconnue. Quelle source est proposée ? quelle type de source est-ce ?**

Vous avez de la chance de nombreux instruments ont répondu aux alertes de positions envoyées durant la nuit d'observation. Vous avez donc une large couverture multi-longueur d'onde de la source. Toutes les informations relatives à ces observations sont répertoriées dans les fichiers *source2\_fermi.txt* et *source2\_lowenergy.txt*.

Ce fichier est constitué de plusieurs colonnes, la première est le logarithme de l'énergie en TeV, la seconde est l'erreur associée, la troisième est le logarithme du flux en erg cm<sup>-2</sup> s<sup>-1</sup>, la quatrième est l'erreur sur le flux, la cinquième et la sixième sont les début et fin des observations en MJD.

**Utiliser le script *source2* et modifidier le afin de déterminer les caractéristiques de la source 2**

Les sources AGN sont caractérisées par 2 pics d'émission à des énergies différentes. Afin de pouvoir classer la source 2, nous allons estimer la position de cette source dans la figure ?? grâce à l'énergie du pic synchrotron et les indices des lois de puissances dans la gamme Fermi-LAT

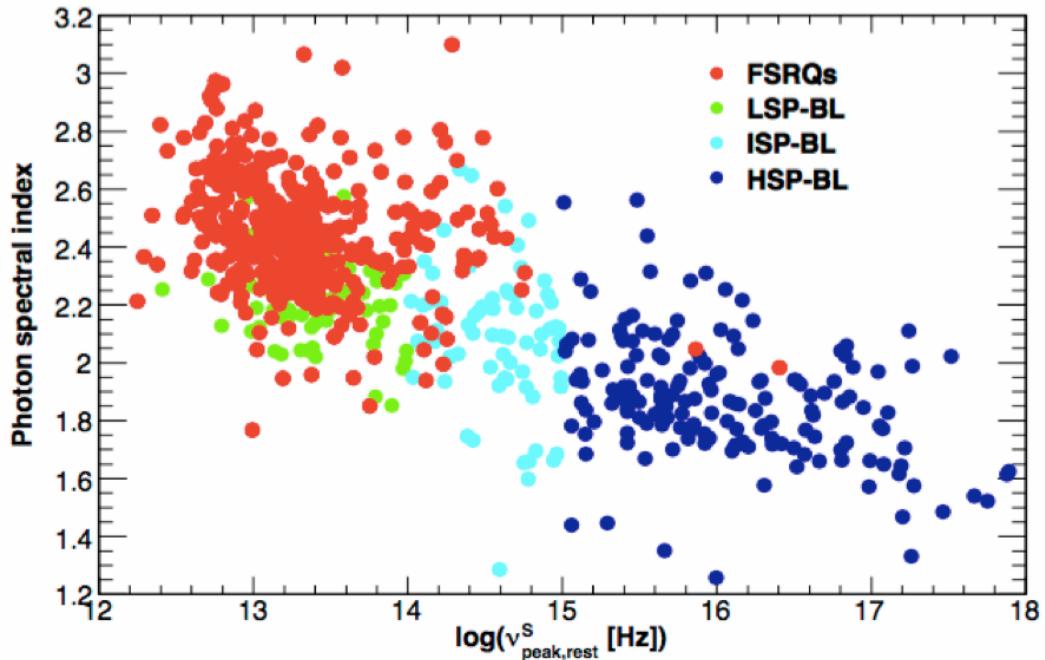


FIGURE 4.1 – Classification des Noyaux actifs de galaxies et disposition en fonction de l'indice spectral et la position du pic synchrotron.

**Déterminer l'indice spectral de Fermi et classer la source 2. Est-ce cohérent avec les données Simbad.**

## Annexe A

# Utilisation du package matplotlib de python

Cette annexe montre les bases de la création d'un script utilisant matplotlib. Dans le script `test.py`.

```
import matplotlib.pyplot as plt
x = [1,2,3,4]
y = [10,20,30,40]
plt.xlabel("Axe_des_X")
plt.ylabel("Axe_des_Y")
plt.plot(x,y)
plt.show() #permet de tracer
```

## Annexe B

# Utilisation de DS9

## Ouvrir un fichier avec DS9

DS9 est un outils de la NASA permettant de visualiser le contenu de fichier fits (type de fichier image utilisé en astrophysique). DS9 permet certaines choses :

- lire des fichiers .fits (figure ??)
- changer l'échelle d'une carte (scale) (figure ??)
- donner une couleur aux cartes (figure ??)
- changer aussi le système de coordonnées (figure ??)

### B.1 Crédation de régions dans DS9 et comptage des événements

Pour compter les événements, aller dans l'onglet édition/région puis tracer sur la carte un cercle soit sur la source (région ON) soit hors de la source. La taille du cercle doit contenir la source (figure ??).

Il est aussi possible de sauvegarder les régions dans un fichier texte. Un exemple est donné ensuite :

```
global color=green dashlist=8 3 width=1 font="helvetica_10_normal_roman" select=1 highlight=1 dash=0 fixed=0 edit=1 move=1 delete=1 include=1 source=1
fk5
circle(201.36506, -43.019113,0.2) # color=green text={source1}
```

- circle(X,Y,R) definir un cerle en X, Y de taille R
- color=green text=source1 donne des infos a afficher

Vous pouvez définir vos régions et leur taille dans un fichier et ensuite le charger dans DS9 pour un résultat plus précis.

Pour compter le nombre d'événements, sélectionner une ou plusieurs région, faites un clic droit et ensuite Analysis/Statistics pour avoir le nombre d'événement (figure??).

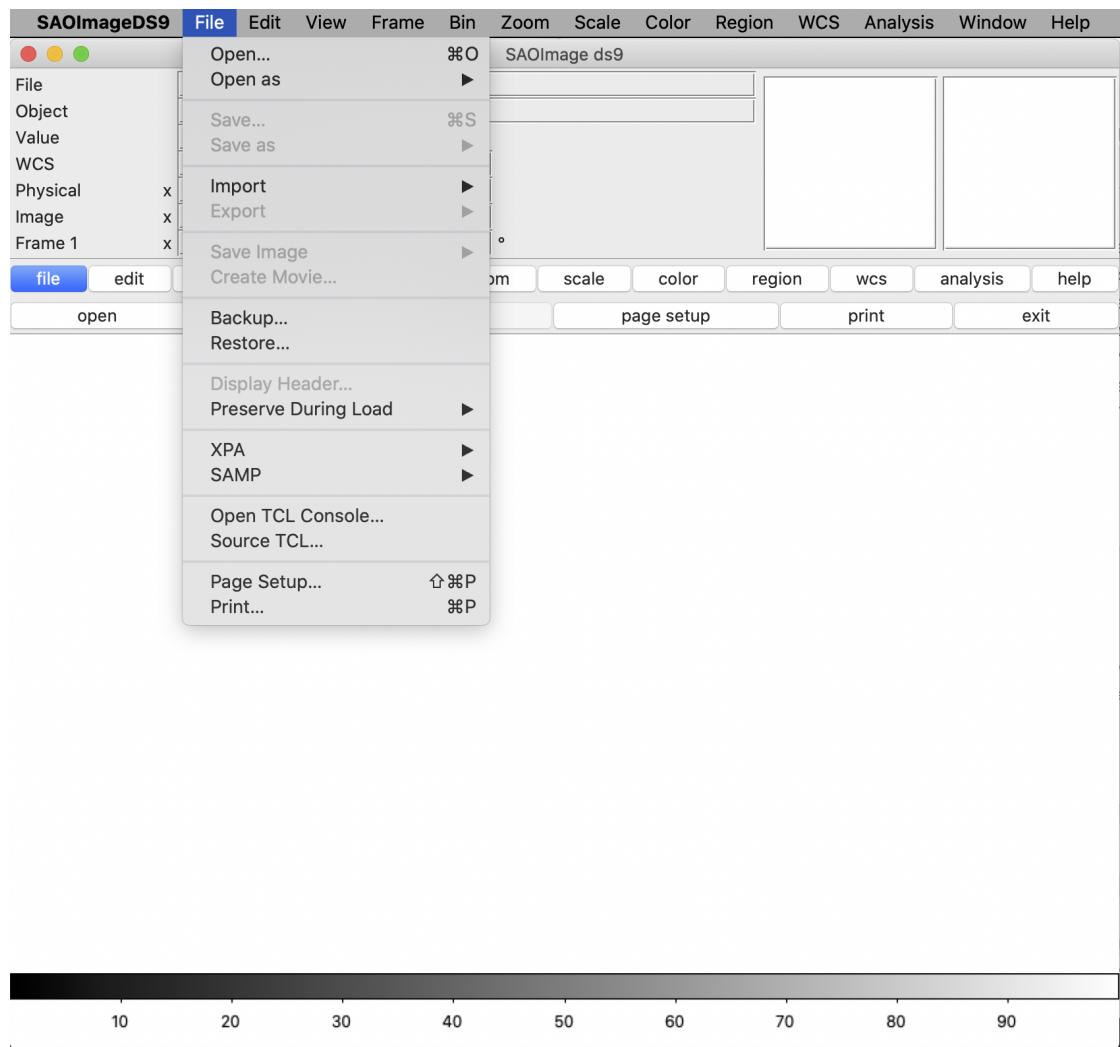


FIGURE B.1 – ouvrir un fichier.

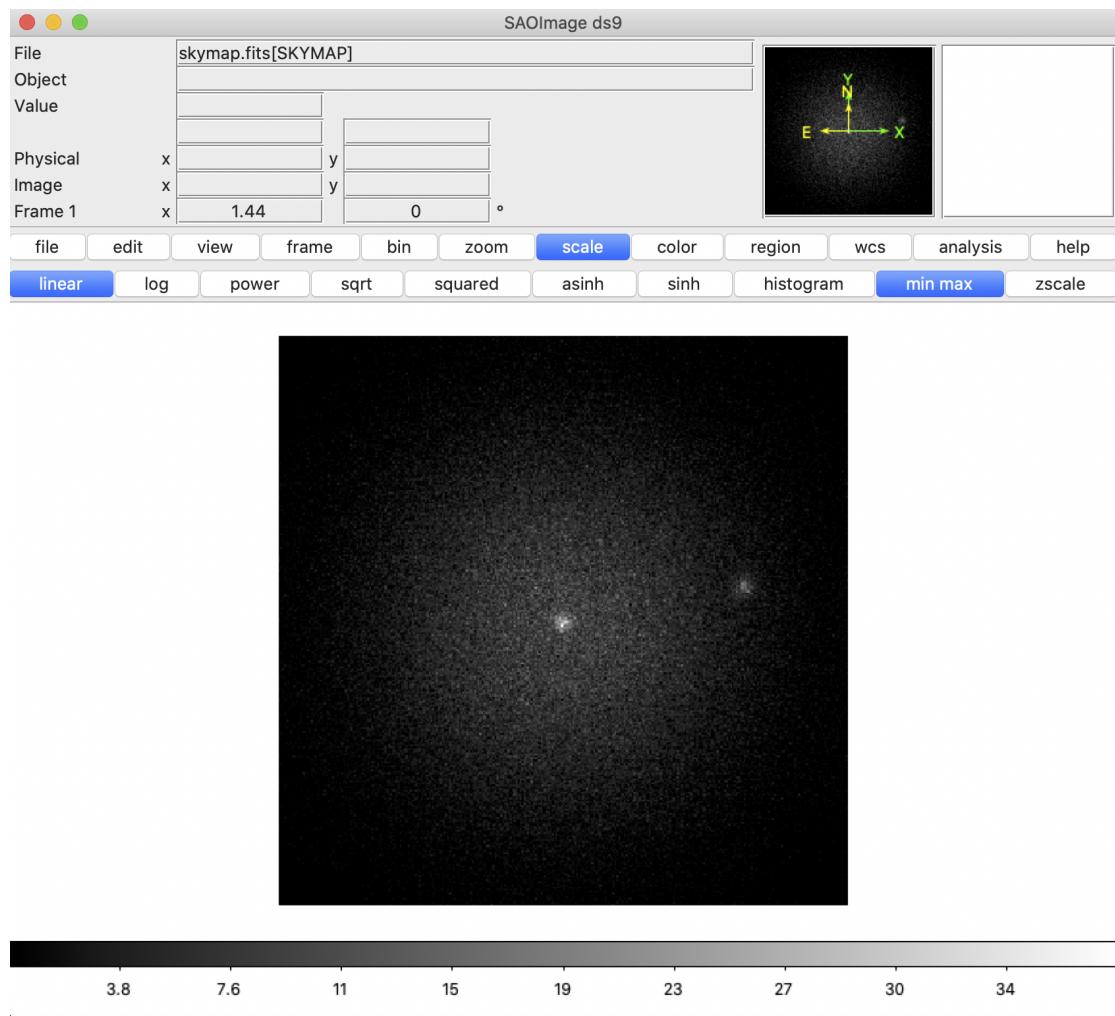


FIGURE B.2 – Les différentes fonctions pour changer l'échelle.

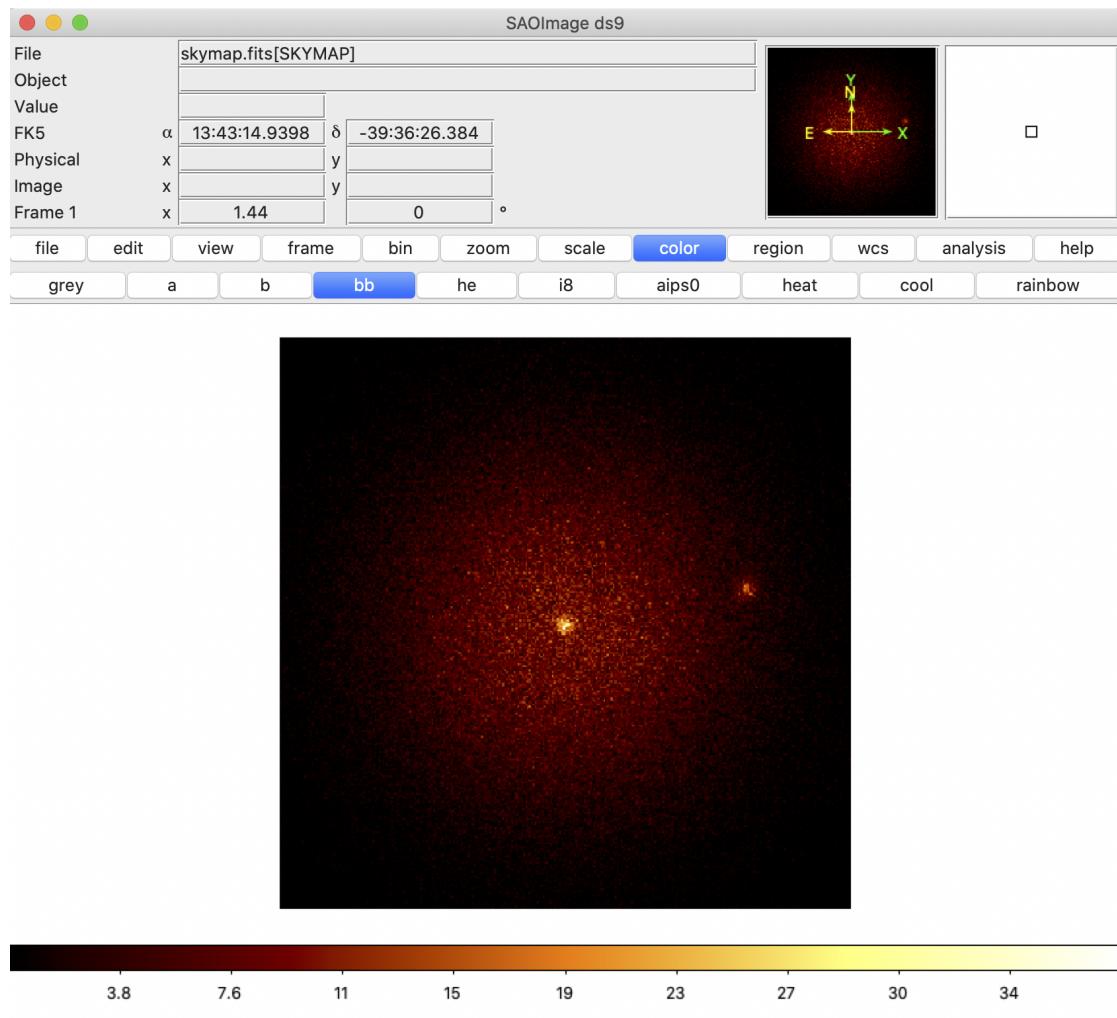


FIGURE B.3 – Les différentes fonctions pour changer la couleur.

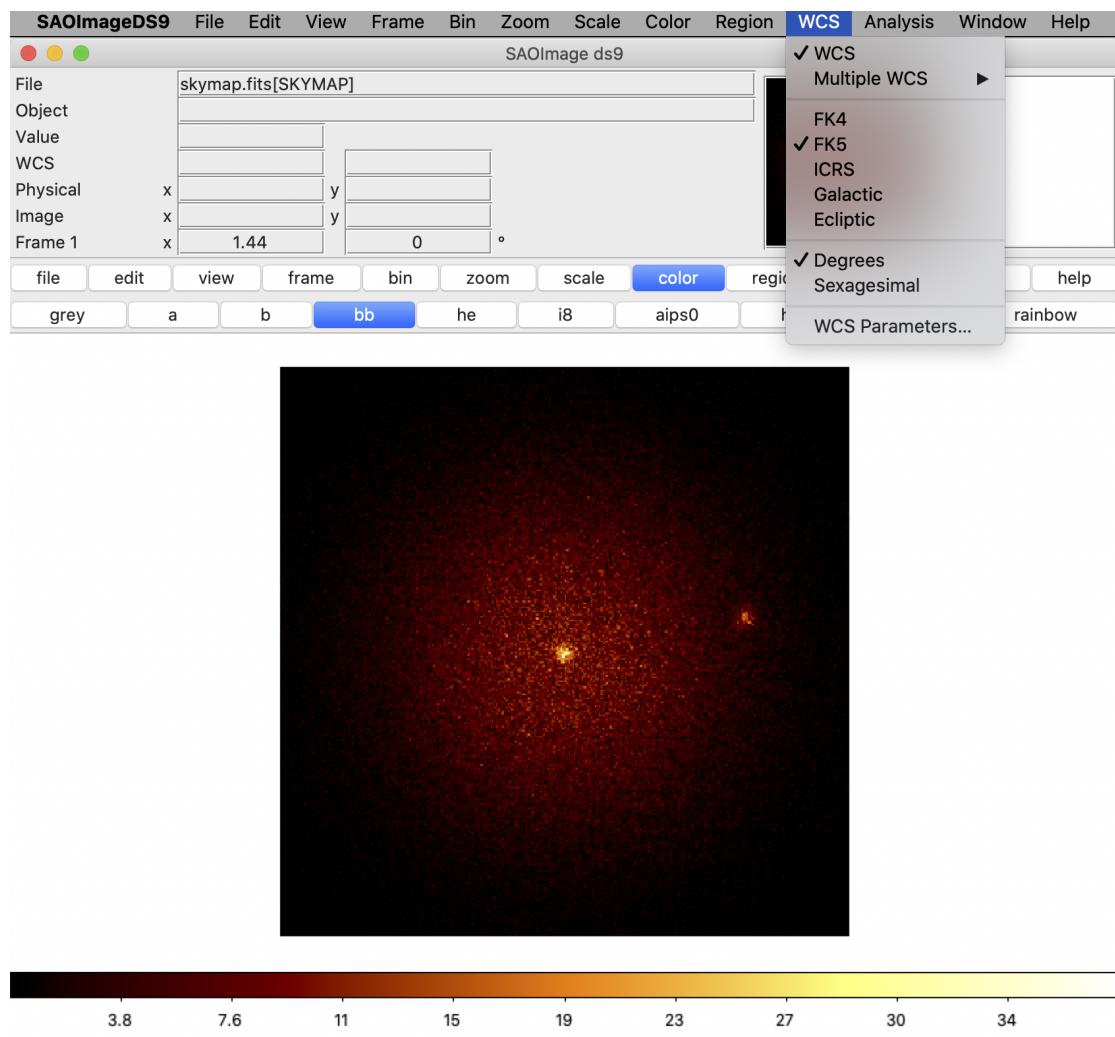


FIGURE B.4 – Les différentes fonctions changer le système de coordonnées.

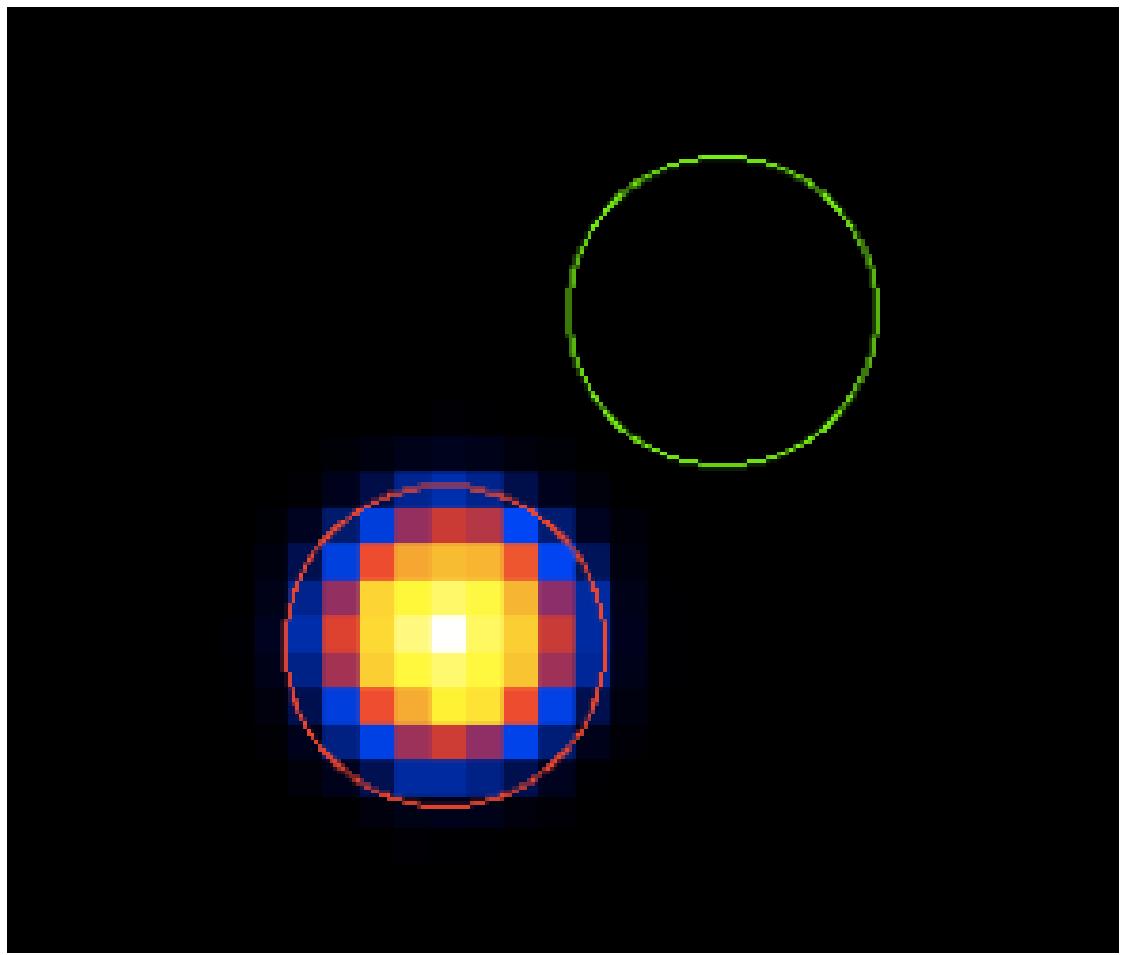


FIGURE B.5 – Exemple de positionnement des régions ON (rouge) et OFF (vert).

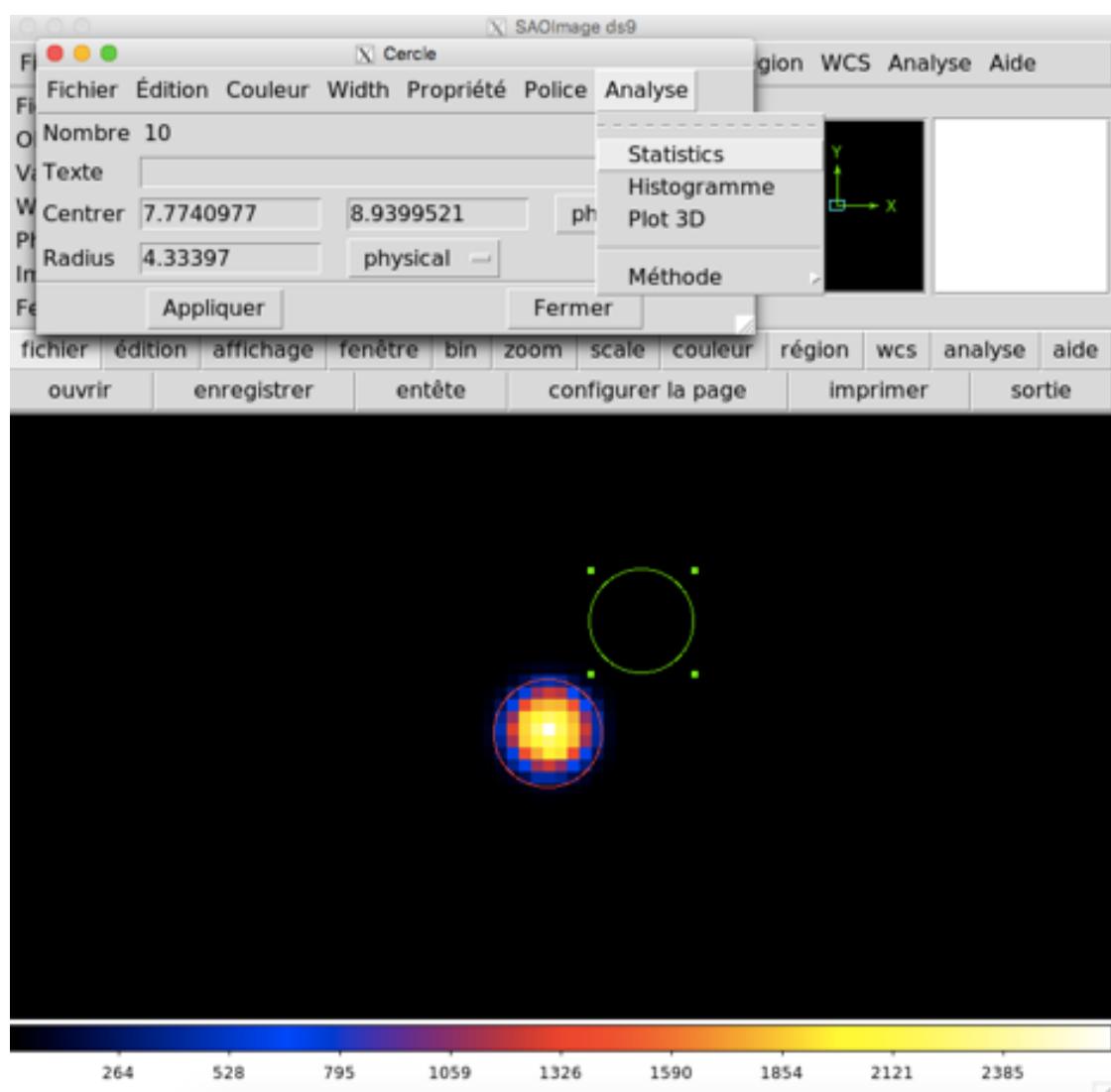


FIGURE B.6 – Positions des boutons de commande de DS9

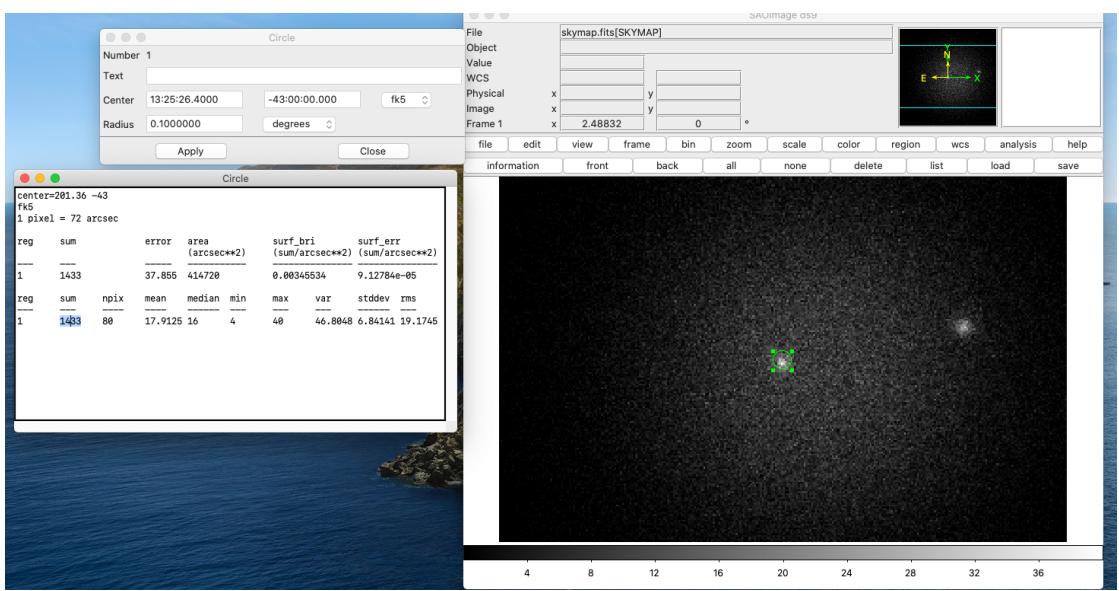


FIGURE B.7 – Lecture du nombre d'événement.