



VO2015 (如月) 講習会 AGNと銀河のクラスタリング解析

国立天文台 天文データセンター 白崎裕治

本課題の目的

- VO を活用する利点
 - GUI により世界中の天文データベースから簡単に自分が必要なデータを探し出すことができる。
 - データアクセスを自動化することにより、大量の様々なデータを利用した研究が可能になる。
- 本課題では後者の研究を目指す
 - 必要となるプログラミング技術
 - プロット作成、フィッティング
 - 比較的上級者向きだが、初心者でも実施できるようサンプルプログラムを用意

プログラミング向け VO ツール

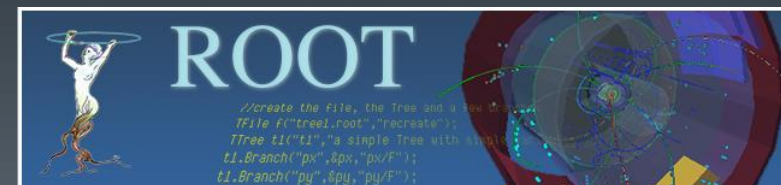
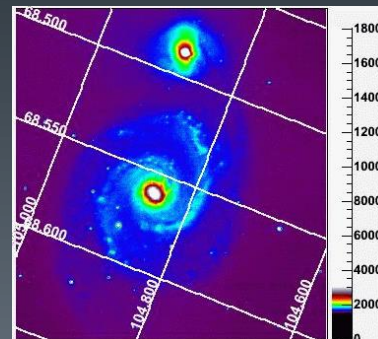
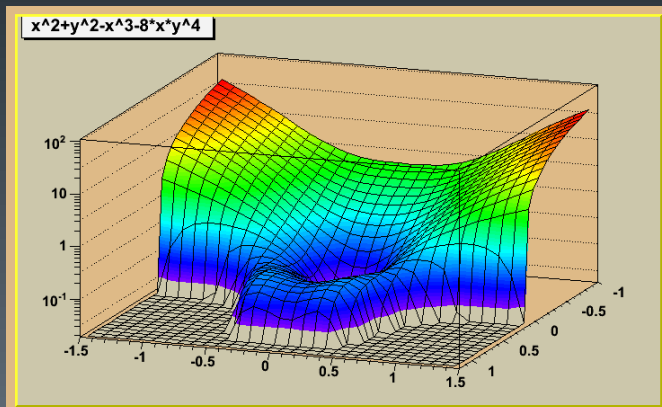
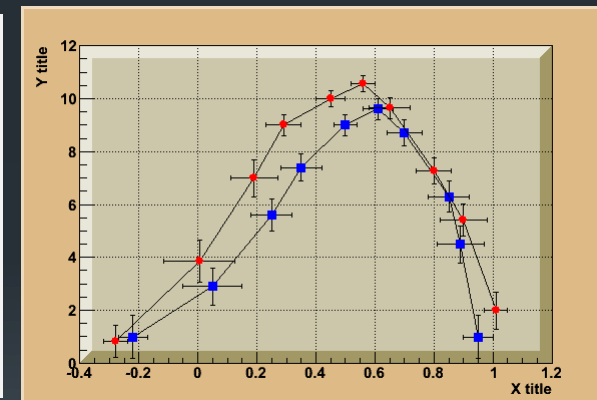
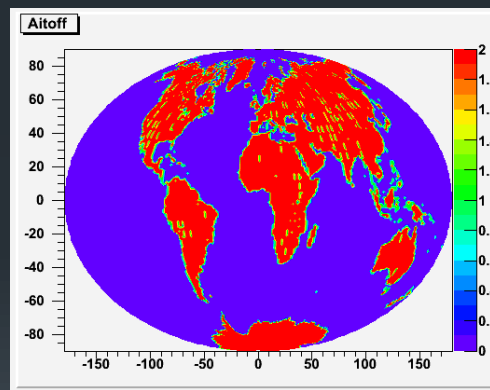
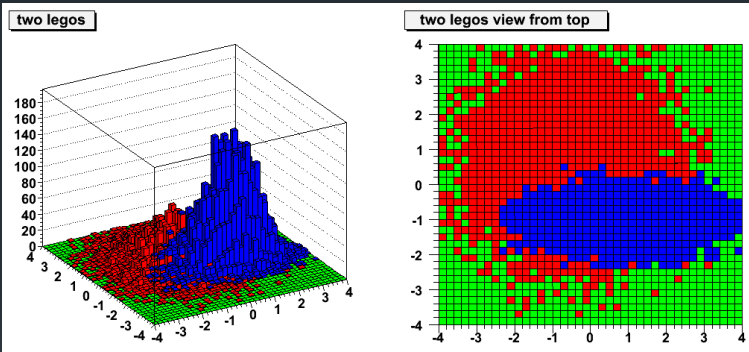
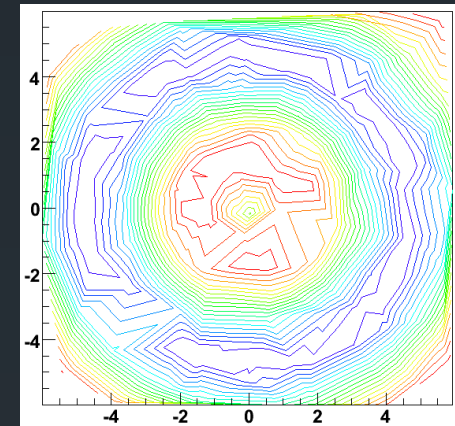
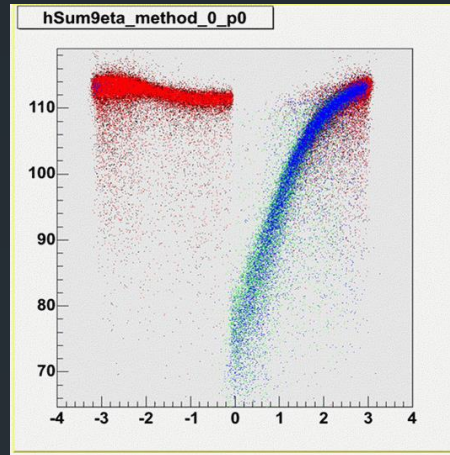
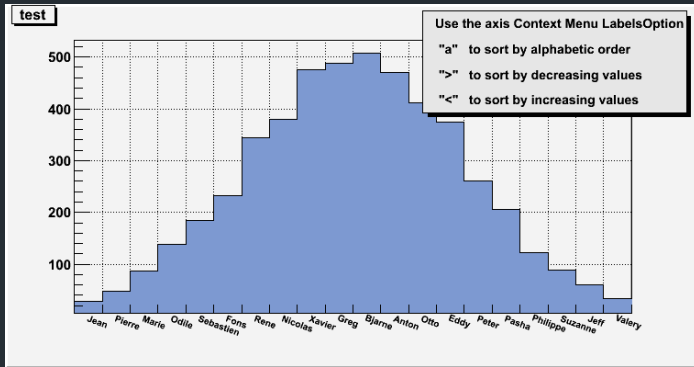
- jc client
JVO portal の機能をコマンドラインから使うツール
- STILTS
Command-line tools for table/VOTable manipulation
- VO-CLI
Command-line Tools for the VO
- AR Commandline
Python commandline VO tools
- Astro Runtime
Middleware that makes it simple to call VO services from programs and science scripts
- VORuby
RUBY language libraries for using VO Services

<http://www.ivoa.net/cgi-bin/twiki/bin/view/IVOA/IvoaApplications>

獲得目標

- JVO データ検索言語 (JVOQL) の使い方
 - パラメータを変えながら何度も検索するのに便利
- jc client によるデータ検索方法
 - JVO ポータルの検索機能をコマンドラインから利用
- シェルスクリプト (bash) の書き方
 - jc client による JVOQL 検索の自動化
- ROOT によるプロット、フィッティング
 - 大量のデータを扱ううえで、データの可視化は重要
 - ROOT はコマンドラインからプロット・フィッティングが可能。しかもフリーウエアである。
 - 使い方がちょっと覚えにくい

ROOTで描けるグラフ



研究テーマ

- 超巨大ブラックホール ($>10^6 M_{\odot}$)
 - ほぼすべての銀河の中心に存在
 - どのようにして形成されるのかは未解決
 - 銀河の合体がブラックホール同士の合体を誘発？
- 活動銀河中心核 (AGN)
 - 銀河中心の超巨大ブラックホールへガス降着
 - 太陽系程の領域から銀河全体を上回る放射
 - 大量のガスが短時間でブラックホールに落ち込む必要
 - 銀河の合体のような激しい相互作用
- AGN は銀河が密集した場所で発生？
 - AGN と銀河のクラスタリング解析



AGN と銀河の相互相関関数

■ 相関関数の定義

- 距離 r における銀河数密度の平均数密度に対する超過比

$$\xi(r) = \frac{\rho(r)}{\rho_0} - 1$$

■ 視線方向に積分した射影相関関数： $\omega(r_p)$

- 銀河の距離データはないので使わない
- 銀河の自己相関関数はべき関数

$$\xi(r) = (r_0/r)^\gamma$$

$$\omega(r_p) = 2 \int_0^\infty \xi(r_p, \pi) d\pi = 2 \int_{r_p}^\infty r dr \xi(r) (r^2 - r_p^2)^{-0.5} = r_p \left(\frac{r_0}{r_p} \right)^\gamma \frac{\Gamma(1/2)\Gamma((\gamma-1)/2)}{\Gamma(\gamma/2)},$$

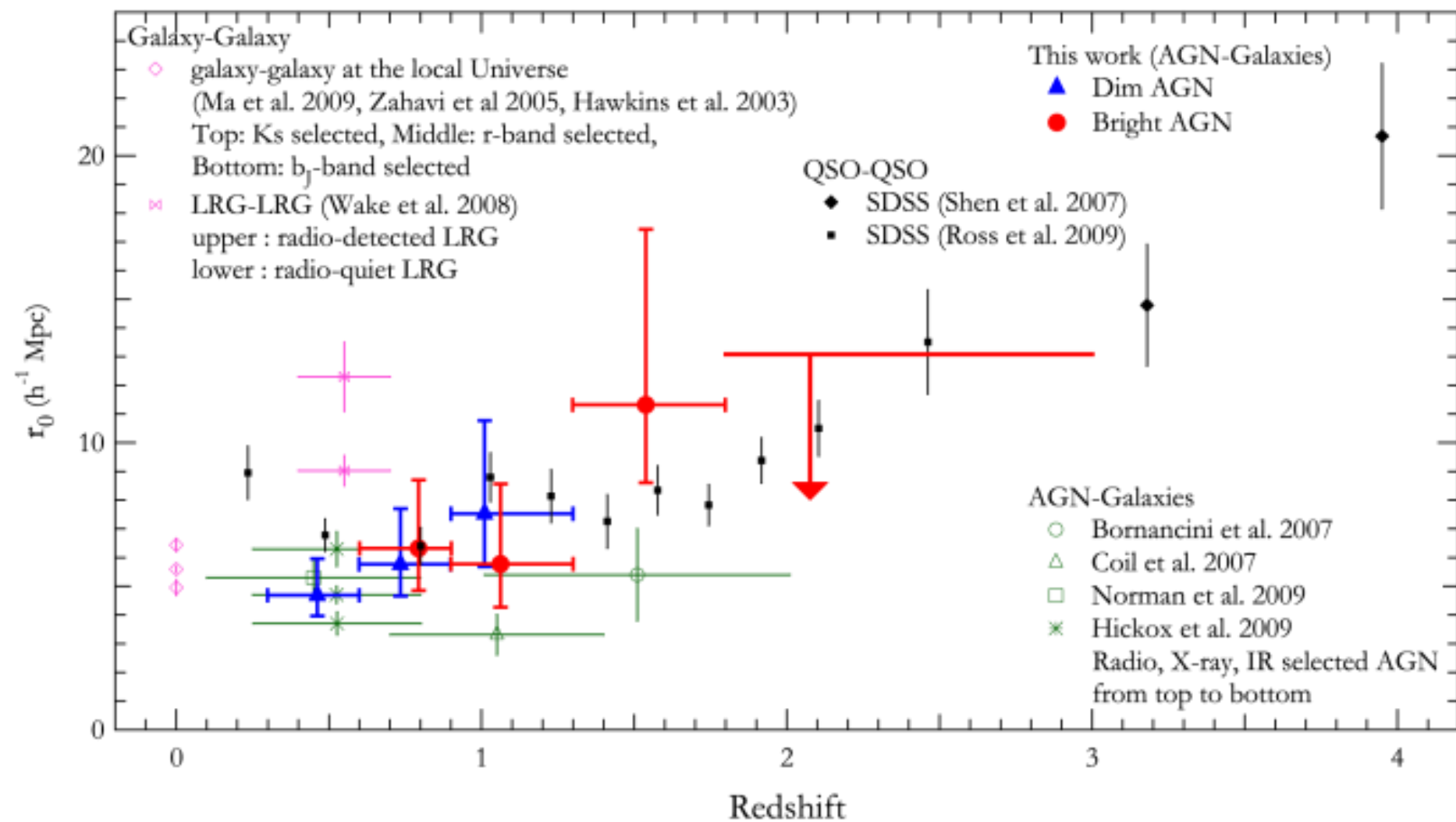
$$\omega(r_p) = \frac{1}{\rho_0} \int_{-\infty}^\infty (\rho(r) - \rho_0) d\pi = \frac{n(r_p) - n_{\text{bg}}}{\rho_0},$$

← Fitting で求める

← 1.8に固定

← LFから求める

先行研究のまとめ



本課題で利用するデータ

- Veron-Cetty and Veron AGN/QSO catalog (Ed.13)
 - `ivo://jvo/agn`
- UKIDSS DR8 カタログ (LAS サーベイ)
 - `ivo://jvo/ukidss`
- JVO のデータサービスを利用する
 - テーブルアップロード機能があり、AGN カタログと UKIDSS のカタログとのクロスマッチ検索が可能

ソフトウェアのインストール

- jc client のインストール
 - http://jvo.nao.ac.jp/jc_client からダウンロード
 - 展開して、install.sh を実行。
 - ~/jvo_tools/ にインストールされる
- VO2015a 講習会用パッケージのインストール
 - <http://jvo.nao.ac.jp/vos2015a/A/vo2015a.tgz> からダウンロード
 - 展開して、INSTALL に書かれている方法でインストール
 - ~/jvo_tools/ にインストールされる

VM イメージの OS 上には
インストール済み

環境設定とサンプルファイル

- VO2015a 講習会用パッケージに環境設定ファイルがあります。
 - 以下のコマンドを実行してください。

```
$ . ~/jvo_tools/init/jvorc.sh
```
 - ~/jc_client/bin が PATH に設定されるほか、ライブラリの検索パス設定などが行われます。
- サンプルファイルが以下のディレクトリにあります。
 - ~/install/vo2015a/clustering/samples
 - ROOT のマクロやシェルクスクリプトのサンプルです。

jc client の使い方

- JVO portal のユーザアカウントが必要です。
- 環境変数の設定を行ってください。
- まずパスワードファイルの作成を行います。

```
1 $ jc passwd
2 user: <ユーザ名>
3 password: <パスワード>
4 retype password: <パスワード>
5 access information has been updated.
```

- 以下のコマンドを実行してみます。

```
1 $ jc --help
2 usage: jc [GLOBAL_OPT] ... COMMAND [COMMAND_OPT] ... COMMAND_ARG ...
3 GLOBAL_OPT:
4     --help show this help
5     -v,--verbose increase verbosity
6 COMMAND:
7     abort abort executing tool
8     conf configure account
9     copy2l download files from VOSpace
10    copy2v upload files to VOSpace
11    delete alias of remove
12    dummy print informations for debug
13    get alias of copy2l
```

実習方法

- 実習テキストを読みながら各自で進めてください。
- わからないことがありましたらお気軽に質問してください。
- 課題 1～5までを順番に実施してください。
- シェルスクリプトを書く課題がありますが、どうしてもできない場合は samples ディレクトリにあるスクリプトを使っても OK です。
- その場合はスクリプトを読んでみてください。

課題1 (AGNと銀河データを取得)

課題1. jc コマンドを用いて、赤方偏移が 0.3 から 0.4 までの AGN のうち、UKIDSS データが存在する AGN について、その座標と赤方偏移のデータを取得する。また、数個の AGN について、その周辺 0.2 度以内の UKIDSS データを取得する。

- 一度に多数のデータを取得することはできない（ことが多い）。
- 検索を分割して実行。まずは AGN のデータのみ取得。
- 下記の JVOQL をファイルに保存、jc client で検索実行

```
1 SELECT DISTINCT agn.name, agn.z, agn.ra, agn.dec
2 FROM ivo://jvo/agn:veron_2010 agn,
3      ivo://jvo/ukidss:catalog_dr8 AS ir
4 WHERE agn.z >= 0.3 and agn.z < 0.4
5      AND
6      distance((ir.ra2000, ir.de2000), (agn.ra, agn.dec)) < 0.01
```

座標の近さで二つのカタログを結合

```
1 $ jc search -i agn.sql -o agn.xml 検索実行 → VOTable
```

```
1 $ jc vot2xsv -F, -o agn.csv agn.xml VOTable → CSV (カンマ区切りのテーブルデータ)
```

課題1 (続き)

- 個々の AGN についてその周辺の銀河カタログを取得。
- 下記の JVOQL をファイルに保存、jc client で検索実行

```
1 SELECT ID, SOURCE_ID, RA2000, DE2000, MAG_K
2 FROM ivo://jvo/ukidss:catalog_dr8 AS t
3 WHERE REGION('CIRCLE 42.861667 1.252222 0.2')
4         AND (mergedClass = 1 OR mergedClass = -3)
5         AND MAG_K > 0 AND kppErrBits <= 255
6         AND SURVEY = 'las'
```

AGN の座標を中心とする半径 0.2 度の範囲

- jc client で検索実行、VOTable → CSV 変換

```
1 $ jc search -i search-ukidss.sql -o ukidss.xml
2 $ jc vot2xsv -F, -o ukidss.csv ukidss.xml
```

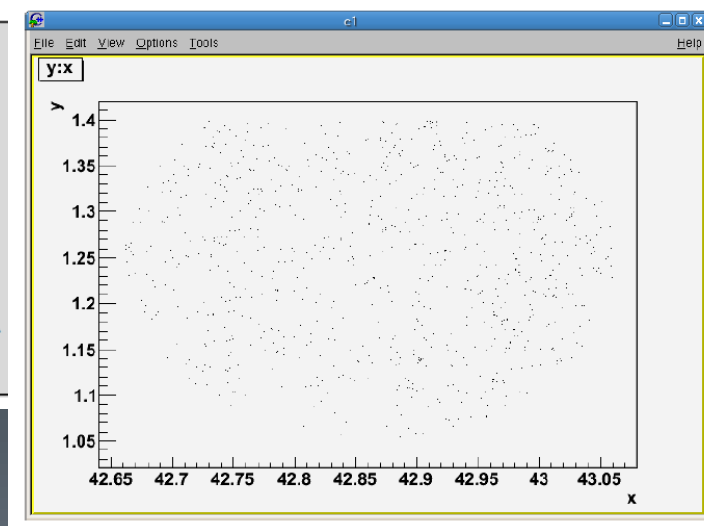
- 以上の検索を数個の AGN について実行してみる。

課題2 (データ欠損がないかを確認)

課題2. 課題1で取得した UKIDSS データをプロットし、欠損領域があるかを確認すること。また、CatalogEffectiveArea を用いて欠損率を求めてみること。

- 取得した銀河データの空間分布をプロットして、欠損領域がないかを確認します。
- ROOT を利用します。以下のコマンドを実行してみます。

```
1 $ cat ukidss.csv | grep -v "#" | awk -F, '{print $3,$4}' > radec.dat
2 $ root -l
3 root [0] double x,y;
4 root [1] TTuple *nt = new TTuple("", "", "x:y");
5 root [2] ifstream data("radec.dat");
6 root [3] while (data>>x>>y) { nt->Fill(x, y);}
7 root [4] nt->Draw("y:x");
8 Info in <TCanvas::MakeDefCanvas>: created default TCanvas with name c1
9 root [5]
```



課題2 (続き)

- サンプルディレクトリにある plot-map.C を実行するともう少し見栄えのよいプロットが作成できます。
- 欠損領域の割合と銀河の数密度を AGN からの距離毎に求めます。配布したプログラムを利用します。

AGN の座標

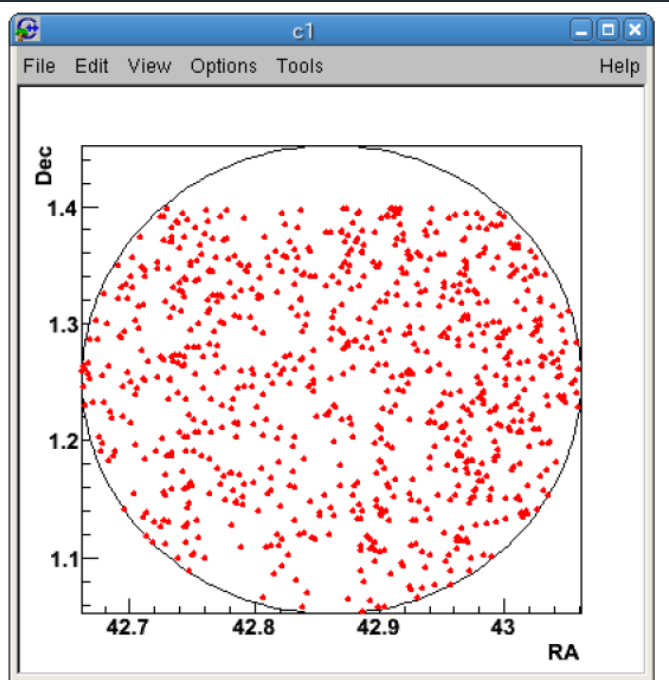


図 6: 銀河の分布。円はデータ検索範囲。

```
1 $ java jp.ac.nao.jvo.research.clustering.CatalogEffectiveArea 42.861667 1.252222  
   radecl.dat 4.0 17.3 10
```

```
1 # bin rmid earea badFrac count density maskFrac  
2 # --- arcsec arcmin2 --- --- arcmin-2 ---  
3 0 8.650 2.47217e-01 0.0000 1 4.04503e+00 0.0000  
4 1 25.950 7.83539e-01 0.0000 0 0.00000e+00 0.0000  
5 2 43.250 1.30590e+00 0.0000 2 1.53151e+00 0.0000  
6 3 60.550 1.82826e+00 0.0000 4 2.18787e+00 0.0000  
7 4 77.850 2.35062e+00 0.0000 3 1.27626e+00 0.0000  
8 5 95.150 2.87298e+00 0.0000 3 1.04421e+00 0.0000  
9 6 112.450 3.39534e+00 0.0000 13 3.82878e+00 0.0000  
10 7 129.750 3.91770e+00 0.0000 9 2.29727e+00 0.0000
```

```
.....  
31 28 493.050 1.48872e+01 0.0000 32 2.14949e+00 0.0000  
32 29 510.350 1.54096e+01 0.0000 39 2.53089e+00 0.0000  
33 30 527.650 1.59160e+01 0.0010 50 3.14149e+00 0.0000  
34 31 544.950 1.55329e+01 0.0560 34 2.18890e+00 0.0000  
35 32 562.250 1.53299e+01 0.0970 26 1.69603e+00 0.0000  
36 33 579.550 1.48042e+01 0.1540 19 1.28342e+00 0.0000  
37 34 596.850 1.53723e+01 0.1470 33 2.14672e+00 0.0000  
38 35 614.150 1.52244e+01 0.1790 39 2.56167e+00 0.0000  
39 36 631.450 1.46809e+01 0.2300 26 1.77101e+00 0.0000  
40 37 648.750 1.42996e+01 0.2700 32 2.23783e+00 0.0000  
41 38 666.050 1.46407e+01 0.2720 39 2.66381e+00 0.0000
```

- AGN からの距離のビン幅 (arcsec)
- 0.1 Mpc に対応する角度
- 配布プログラム cosmic-distance を利用

課題3 (AGN毎に銀河分布を計算する)

課題3. 次の作業を行うスクリプトを作成する。

1. 引数により AGN の座標と赤方偏移、作業ディレクトリ名をスクリプトに変数として与えられること。
2. スクリプト実行開始時に引数で与えられた作業ディレクトリを作成し、そこに移動しそこで検索などの作業が行われること。
3. AGN から半径 0.14 度の範囲の UKIDSS データを取得すること。取得した UKIDSS データのうち、赤経、赤緯、等級について抜き出し radecmag.dat というファイルにスペース区切りで天体毎に一行ずつ書き出すこと。
4. 取得した UKIDSS データから欠損率を求め、earea.dat というファイル名で保存すること。
5. UKIDSS データから数密度分布を AGN からの射影距離の関数として求め、hist.dat というファイル名で保存すること。
6. UKIDSS データの天球分布と数密度分布をグラフ化し、png 形式で保存すること。
7. AGN の座標、赤方偏移、観測バンド (今回は K) を summary.dat に出力する。フォーマットは一行ごとに<KEYWORD>: <VALUE> 形式で記述すること。それぞれ、次のキーワードを使用すること: RA, DEC, REDSHIFT, BAND。
課題1で検索された AGN のうち、100 個程度について、このスクリプトを実行し、数密度分布を求めるスクリプトを作成し実行する。

- 100個の AGN について銀河数密度分布を求めるために、上記の処理を行うスクリプトを作成し、実行します。

- 課題 2で求めた欠損率で補正した銀河数密度分布を AGN からの距離の関数として求めます。
- 配布プログラム catalog-density を利用します。

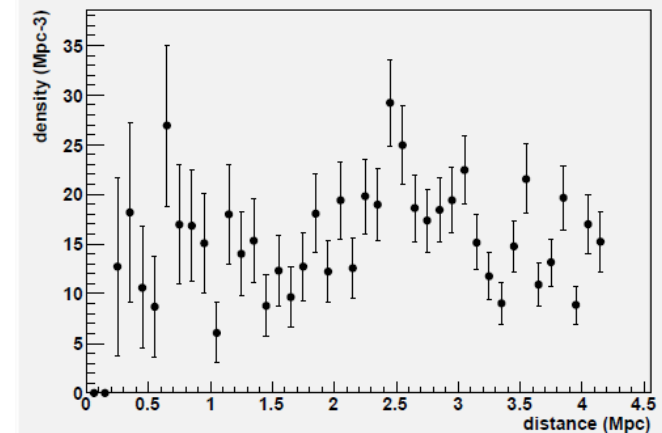


図 7: AGN からの射影距離に対する銀河数密度分布

課題4 (平均化された銀河分布を求める)

課題4. 課題3で求められた数密度分布を `agn-density-add.sh` を用いて足し合わせ平均化する。

- 課題3で作成された100個のディレクトリのリストをファイルに書き出します。

```
1 $ ls -d agn_??? > list.dat
```

- 配布スクリプト `agn-density-add.sh` にディレクトリリストを引数として与えて実行する。

```
1 $ agn-density-add.sh -l list.dat
```

- 足し合わされた結果が `hist-add.dat` というファイルに作成されます。

課題5 (相関距離を求める)

課題5. 課題4で求めた平均化された数密度分布をプロットする。また、相関関数フィッティングによって相関距離を求める。

- サンプルディレクトリにある fit-density.C を利用します。
- hist-add.dat に書かれている AVE_DEN_60 の値 (左から2番目) を第一引数に、第二引数にはフィットする範囲 (AGN からの距離の最大値)

```
1 root [1] .x fit-density.C(0.00329598, 2.6);
2 num = 26
3 FCN=33.6957 FROM MIGRAD STATUS=CONVERGED 118 CALLS 119 TOTAL
4 EDM=1.75127e-10 STRATEGY= 1 ERROR MATRIX ACCURATE
5 EXT PARAMETER STEP FIRST
6 NO. NAME VALUE ERROR SIZE DERIVATIVE
7 1 p0 9.41010e+00 5.83351e-01 3.01021e-05 1.24625e-03
8 2 p1 7.75979e+00 4.67590e-02 1.40621e-06 3.73505e-02
9 3 p2 3.29598e-03 fixed
```

```
1 # AVE_DEN_PK: 0.398565 0.00237241 168
2 # AVE_DEN_10: 0.417478 0.00248499 168
3 # AVE_DEN_60: 0.553724 0.00329598 168
4 # AVE_DEN_80: 0.636316 0.00378759 168
```

相関距離 ($h^{-1}\text{Mpc}$)

