SURP Homework (topic: CMB)

Z. Huang

May 5, 2015

1 In python you can put a "j" *after a number* to make it imaginary. Compare print (1+j)*(2-3j) and print (1+1j)*(2-3j) in python interactive mode. What is the difference? (2pt)

Write a function convert2complex(x, y) that converts two real inputs x and y to a complex output (function return value) $x + iy$. (8pt)

2 The python function random.gauss(μ, σ) returns a random Gaussian variable $x \sim N(\mu, \sigma)$, where μ is the mean and σ the standard deviation. Write a script to generate a list of 100 independent random Gaussian variables $x_1, x_2, \ldots, x_{100} \sim N(0, 1)$ and calculate the mean $\mu_{100} = \frac{1}{100}\sum_{i=1}^{100} x_i$. (10pt) Expand the list to 10,000 independent random Gaussian variables $\sim N(0, 1)$ and calculate the mean μ_{10000}. (5pt) Compare μ_{100} and μ_{10000}: which one is closer to zero? (5pt) Why? (5pt)

3 Write a function ComplexGaussian(P) that, for any input "power" $P \geq 0$, returns a complex random Gaussian variable $z = \frac{x+iy}{\sqrt{2}}$, where $x \sim N(0, \sqrt{P})$, $y \sim N(0, \sqrt{P})$, and x and y are independent. (10pt) Choose a $P > 0$ and generate a list of 100 ComplexGaussian(P) outputs $z_1, z_2, \ldots, z_{100}$. Compute the "observed power" $P_{\rm obs} \equiv \frac{1}{100}\sum_{i=1}^{100} |z_i|^2$ and compare it to the "true power" P. (5pt) Repeat the calculation for 10,000 ComplexGaussian(P) outputs. Does $P_{\rm obs}$ become closer to P? (5pt) Why? (5pt)

4 The cosmic microwave background (CMB) temperature fluctuation ΔT is a function of direction $\mathbf{n}$. In spherical coordinates it can be written as $\Delta T(\theta, \phi)$ and decomposed into

$$\Delta T(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} a_{lm} Y_{lm}(\theta, \phi)\,, \tag{1}$$

where Y_{lm}'s are the spherical harmonic functions. The simplest Inflation models predict that each a_{lm} is a independent complex random Gaussian variable with "power" C_ℓ. (See question 3 for the definition.) Statistical isotropy guarantees that the power does not depend on m. C_ℓ as a function of ℓ is called "power spectrum".

Import a Git repository from the terminal (linux or MacOS):
git clone https://github.com/zqhuang/SURP2015/
In the repository you will find a file sample_cmb_map.fits, a pixelized map of $\Delta T(\theta, \phi)$. Use more sample_cmb_map.fits to read the header of the file. What is the NSIDE parameter of the map? (10pt)

Install python package "Healpy" from https://healpy.readthedocs.org/en/latest/. Read the documentation. Write a script to compute the power spectrum C_ℓ ($0 \leq \ell \leq 300$) for the map. (25pt)

Mask out the "northern sphere" ($0 \leq \theta < \pi/2$) by filling pixels with zeros. Compute the "pseudo power spectrum" (power spectrum of the masked sky) $\tilde{C}_\ell$ ($0 \leq \ell \leq 300$). (5pt)