MATH 407 FINAL PROJECT
PROFESSOR LOTOTSKY

Nathanael Jo

In collaboration with Benjamin Weltman and Madison Kadesh

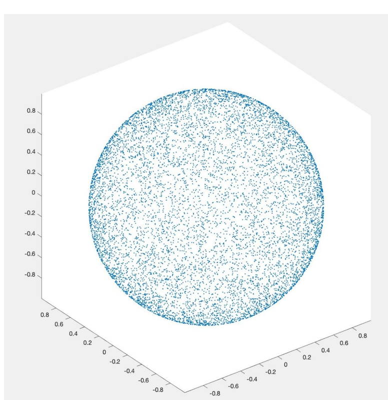
NOVEMBER 29, 2018

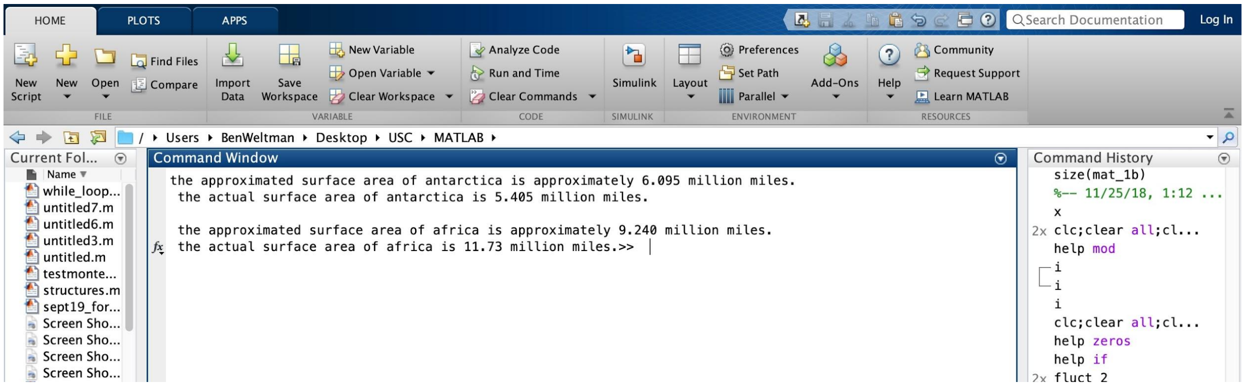
PART I

Code

```
clc;clear all;close all;

theta=linspace(0,2*pi,40);
phi=linspace(0,pi,40);
[theta,phi]=meshgrid(theta,phi);
rho=1;
x=rho*sin(phi).*cos(theta);
y=rho*sin(phi).*sin(theta);
z=rho*cos(phi);
mesh(x,y,z);

i = 1;
ant = 0;
af = 0;

for i = 1:1:1000
    TH = 2*pi*rand(1,1e4);
    PH = asin(-1+2*rand(1,1e4));
    [X,Y,Z] = sph2cart(TH,PH,1);
    plot3(X,Y,Z,'.','markersize',1)
    axis equal vis3d
    i = i+1;

    if PH(1,i)> 1.20428 %radians (@ 79 degrees)
        ant = ant + 1; %counter for antarctica
    else
    end

    if PH(1,i) > 1.1033 %radians (from 0 (equator) to 12.55 degrees)
        af = af + 1; %counter for africa
    else
    end

end

ant;
af;

sa_ant = (ant/1000)*196.6;
sa_af = (af/1000)*196.6;

fprintf('the approximated surface area of antarctica is approximately
%.3f million miles. \n the actual surface area of antarctica is 5.405
million miles. \n', sa_ant)

fprintf('the approximated surface area of africa is approximately
%.3f million miles. \n the actual surface area of africa is 11.73
million miles.', sa_af)
```

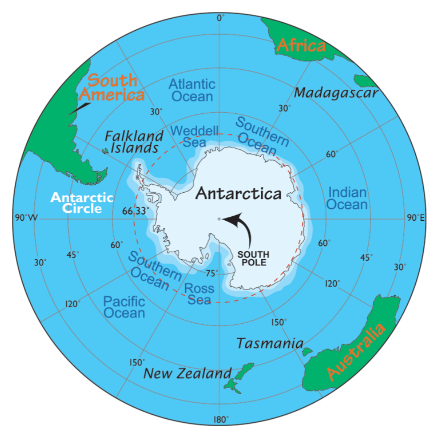
Findings





Pseudo-code/Explanation:

1. Plotted 1,000 uniform random points on a unit sphere. Points are in spherical coordinates, but the conventions for phi are different, ranging from $\frac{-\Pi}{2}$ to $\frac{\Pi}{2}$
2. Antarctica: Used the latitude coordinates for the Antarctic circle (66.33) and approximated the entire area could fit inside the 69° circle
    a. Count all points where $\phi > 1.20428\, rad\ (69^o)$
    b. Surface area (in mi^2) = $(\frac{count}{1000})\ *\ 196.6\ =\ 6.095\ mill.\ mi^2$, where 196.6 is the surface area of the earth



3. Africa: Treat the equator as the bottom of the circle (i.e. tilt the globe as if Africa is in Antarctica's positions). This allows us to repeat the code used for Antarctica.
    a. Finding the latitude
        i. Photoshopped Africa's land area into a circle centered at the equator
        ii. Picked a point B on the outline of the circle that is directly below the center (point A)
        iii. Find the difference in latitudes between points A and B (26.78-0)
        iv. Since we are treating Africa to be at the bottom of the globe, the entire area could fit inside the circle $90\ -\ 26.56\ =\ 63.2144$(1.1033 rad)
    b. Count all points where $\phi > 1.1033\, rad$
    c. Surface area (in mi^2) = $(\frac{count}{1000})\ *\ 196.6$, where 196.6 is the surface area of the earth

Analysis

From our calculation we found Antarctica to be approximately 6.095 million square miles, with the actual surface area being 5.405 million square miles.

For Africa, our calculation of survale area came to be approximately 9.240 million square miles, the actual surface area is 11.73 million square miles.

The relative errors seem to be larger for Africa then for Antarctica. This can be due to the fact that Africa is a much larger surface area to approximate. Additionally, Africa is not a perfect circle, so there were a number of assumptions made, one of which being that we fit Africa into a circle to approximate the latitude lines which we used to calculate the surface area. Finally, as you move further away from the poles, the change in surface area is more radical. So, because Africa is larger, our surface area approximation is much more sensitive to small errors in our latitude approximation.

PART II

I am using the **Wald-Wolfowitz runs test** as described below, which tests for non-randomness.

Test Statistic:	The test statistic is $$Z = \frac{R - \bar{R}}{s_R}$$ where R is the observed number of runs, $\bar{R}$, is the expected number of runs, and s_R is the standard deviation of the number of runs. The values of $\bar{R}$ and s_R are computed as follows: $$\bar{R} = \frac{2n_1 n_2}{n_1 + n_2} + 1$$ $$s_R^2 = \frac{2n_1 n_2(2n_1 n_2 - n_1 - n_2)}{(n_1 + n_2)^2(n_1 + n_2 - 1)}$$ with n_1 and n_2 denoting the number of positive and negative values in the series.
Significance Level:	α
Critical Region:	The runs test rejects the null hypothesis if $$|\mathbf{Z}| > \mathbf{Z}_{1-\alpha/2}$$ For a large-sample runs test (where $n_1 > 10$ and $n_2 > 10$), the test statistic is compared to a standard normal table. That is, at the 5 % significance level, a test statistic with an absolute value greater than 1.96 indicates non-randomness. For a small-sample runs test, there are tables to determine critical values that depend on values of n_1 and n_2 (Mendenhall, 1982).

Code

```
clc;clear all;close all;

% ALL NUMBERS
concat = '';

for i = 1:1000
    newnum = dec2bin(i);
    concat = strcat(concat,newnum);
end

counter = 0;
counter0 = 0;
counter1 = 0;

for i = 1:8986
    if concat(1,i) ~= concat(1,i+1)
        counter = counter + 1;
    end

    if concat(1,i) == '0'
        counter0 = counter0 + 1;
    end

    if concat(1,i) == '1'
        counter1 = counter1 + 1;
    end
end

r = counter;
rbar = ((2*(counter0)*(counter1))/(counter0 + counter1)) + 1;
sr = sqrt((2*counter0*counter1*(2*counter0*counter1-counter0-
counter1))/(((counter0+counter1)^2)*(counter0+counter1-1)));

z = (r-rbar)/sr;

fprintf('the champernowne z score is %.5f \n',z)
```

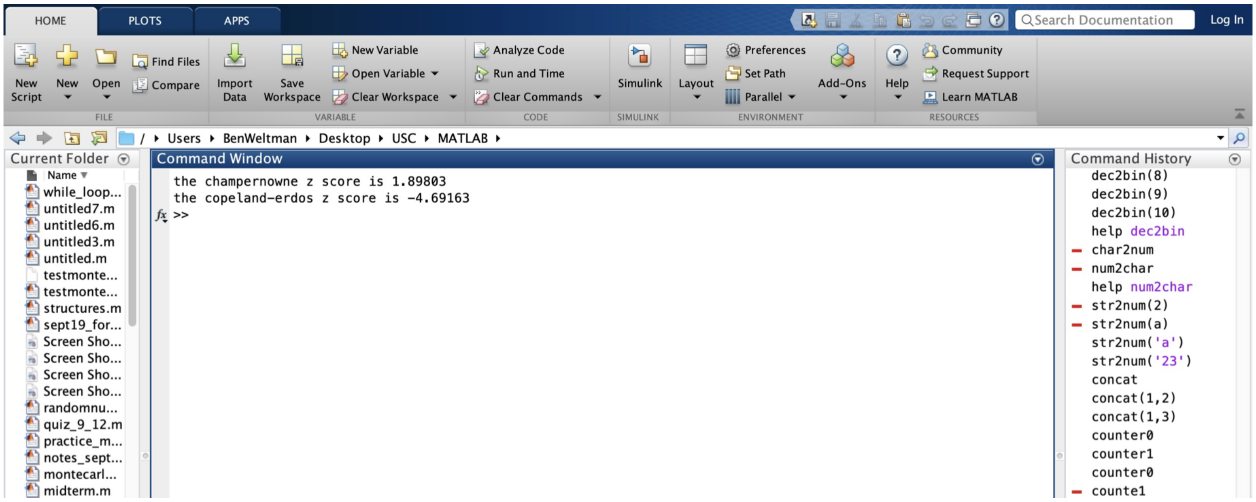
```
% PRIME NUMBERS

prime = [2, 3,  5,  7,  11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61,
67, 71, 73, 79, 83, 89, 97] % array has been shortened from the original 1000
prime numbers for the sake of simplicity.

concat_prime = '';

for i = 1:1000
    newnum = dec2bin(prime(1,i));
    concat_prime = strcat(concat_prime,newnum);
end

counter = 0;
counter0 = 0;
counter1 = 0;

for i = 1:11731
    if concat_prime(1,i) ~= concat_prime(1,i+1)
        counter = counter + 1;
    end

    if concat_prime(1,i) == '0'
        counter0 = counter0 + 1;
    end

    if concat_prime(1,i) == '1'
        counter1 = counter1 + 1;
    end
end

r = counter;
rbar = ((2*(counter0)*(counter1))/(counter0 + counter1)) + 1;
sr = sqrt((2*counter0*counter1*(2*counter0*counter1-counter0-
counter1))/(((counter0+counter1)^2)*(counter0+counter1-1)));

z = (r-rbar)/sr;

fprintf('the copeland-erdos z score is %.5f \n',z)
```

<u>Pseudo-code/Explanation</u> (explanation of process for both Champernowne number and Copeland-Erdös constant):

1. Convert the sequence into a continuous string of binary numbers (the first 1,000 numbers were used)
2. Count the total amount of runs in the string
3. Count also the frequency of 1s and 0s
4. Find the z-score by putting into the formula:

$$Z = \frac{R-\bar{R}}{s_R}$$

Findings



Analysis

- Defining the critical region:

The runs test rejects the null hypothesis if

$$|Z| > Z_{1-\alpha/2}$$

And let there be a 5% significance level. So, if

$$|Z| > 1.96$$

then the sequence is not produced in a random manner.

- Champernowne constant: With a z-score of 1.89803, it's reasonable to conclude that this number was generated randomly. This is interesting because all the Champernowne number is just a sequence of numbers going up by one, which is not random at all.
- Copeland-Erdös constant: With a z-score of -4.69163, it's reasonable to conclude, even without the confidence interval, that this is not a random sequence. This result is very peculiar because the sequence of prime numbers is widely considered as random.

Both results defied our initial assumptions about their respective definitions of randomness. There are a few things that could explain this anomaly:

1. We interpreted the test incorrectly
2. However, it is more likely that the Wald-Wolfowitz runs test simply defines randomness in a different way than we do
3. Randomness relies on the conversion system we used (base 10, or in our case, binary)