

## DAILY ASSESSMENT FORMAT

<b>Date:</b>	26-05-2020	<b>Name:</b>	BINDUSHRI
<b>Course:</b>	dsp	<b>USN:</b>	4AL17EC011
<b>Topic:</b>	<p>1.fourier series&amp;gibbs phenomena using python.</p> <p>2.fourier transform derivative</p> <p>3.fourier transform and convolution.</p> <p>4.inition of fourier transform&amp;laplace transform</p> <p>5.laplace transform of firstorder</p> <p>6.implementation of fourier transform using matlab</p> <p>7.application of z transform</p> <p>8.z transform of sequence using matlab.</p> <p>9.fourier transform</p>	<b>Semester &amp; Section:</b>	6 <sup>th</sup> A
<b>Github Repository:</b>	Bindushri		

### FORENOON SESSION DETAILS

The screenshot shows a YouTube video player with the URL [https://www.youtube.com/watch?v=sSvj1nCC6\\_o](https://www.youtube.com/watch?v=sSvj1nCC6_o). The video is titled "Fourier Series and Gibbs Phenomena [Python]" and is part of a series by Steve Brunton. The Jupyter Notebook cell displays Python code for calculating coefficients  $a_k$  and  $b_k$  for a square wave function  $f(x)$  over an interval  $[0, L]$ , and plotting the resulting Fourier series approximation. The video feed shows a man in a dark shirt and glasses speaking while writing mathematical formulas on a chalkboard. The formulas include the Fourier series expansion and the definitions of  $a_k$  and  $b_k$ .

```

In [2]: import numpy as np
        import matplotlib.pyplot as plt
        plt.rcParams['figure.figsize'] = [8, 8]
        plt.rcParams.update({'font.size': 16})

        dx = 0.01
        L = 2*np.pi
        x = np.arange(0,L,dx)
        n = len(x)
        nquarter = int(np.floor(n/4))

        f = np.zeros_like(x)
        f[nquarter:nquarter] = 1

        A0 = np.sum(f + np.ones_like(x)) * dx * 2 / L
        fFS = A0/2 + np.ones_like(f)

        for k in range(1,n):
            Ak = np.sum(f * np.cos(2*np.pi*k*x/L)) * dx * 2 / L
            Bk = np.sum(f * np.sin(2*np.pi*k*x/L)) * dx * 2 / L
            fFS = fFS + Ak*np.cos(2*k*np.pi*x/L) + Bk*np.sin(2*k*np.pi*x/L)

        plt.plot(x,f,color='k',LineWidth=2)
        plt.plot(x,fFS, -,color='c',LineWidth=1.5)
        plt.show()
    
```

Up next

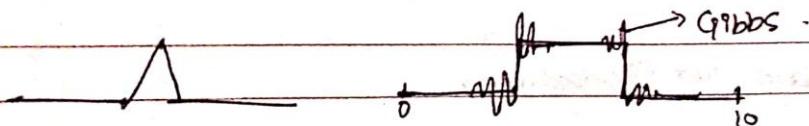
- Denoising Data with FFT [Python] Steve Brunton 3.7K views · 1 month ago
- The Fast Fourier Transform Algorithm Steve Brunton 6.8K views · 1 month ago
- Randomized SVD Code [Python] Steve Brunton 2K views · 3 months ago
- Solving the Heat Equation with the Fourier Transform Steve Brunton 5.6K views · 1 month ago
- Fourier Analysis Steve Brunton
- The Fast Fourier Transform (FFT) DATA-DRIVEN SCIENCE AND ENGINEERING WITH MATLAB

~~Day 2 : 26-05-2020 / FOURIER SERIES & GIBBS Phenomena using Python~~

$$f(x) \cong \sum_{k=0}^{100} a_k \cos\left(k \frac{2\pi x}{L}\right) + b_k \sin\left(k \frac{2\pi x}{L}\right)$$

$$a_k = \langle f(x), \cos\left(k \frac{2\pi x}{L}\right) \rangle$$

$$b_k = \langle f(x), \sin\left(k \frac{2\pi x}{L}\right) \rangle$$



\* Fourier transform :-

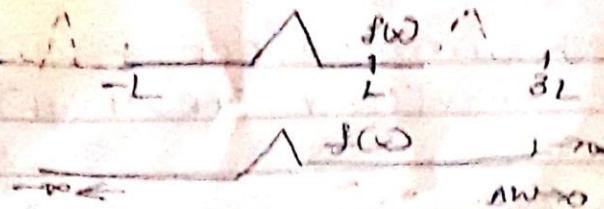
$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{ik\pi x/L}$$

$$c_k = \frac{1}{2\pi} \langle f(x), \psi_k \rangle = \frac{1}{2\pi} \int_{-L}^L f(x) e^{-ik\pi x/L} dx$$

Page \_\_\_\_\_  
Date \_\_\_\_\_

$$w = \frac{2\pi}{L}$$

$$\Delta w = \pi/L$$



$$f(x) = \sum_{n=-\infty}^{\infty} \frac{1}{2\pi n} \int_{-\pi/\Delta w}^{\pi/\Delta w} f(\xi) e^{-i\omega_n \xi} d\xi e^{i\omega_n x}$$

$$= \sum_{n=-\infty}^{\infty} \frac{1}{2\pi n} \int_{-\infty}^{\infty} f(\xi) e^{i\omega_n \xi} d\xi e^{i\omega_n x} dw$$

$$\hat{f}(w) = \mathcal{F}(f(x)) = \int_{-\infty}^{\infty} f(x) e^{iwx} dx$$

$$f(x) = \mathcal{F}^{-1}(\hat{f}(w)) = \sum_{n=-\infty}^{\infty} \hat{f}(w) e^{i\omega_n x} dw$$

\* Fourier transform derivative.

$$\mathcal{F}\left(\frac{df}{dx} f(x)\right) = \int_{-\infty}^{\infty} \frac{df}{dx} e^{iwx} dx$$

$$= \underbrace{\int_{-\infty}^{\infty} f(x) e^{iwx} dx}_{w=0} - \int_{-\infty}^{\infty} f(x) \left( \frac{d}{dx} e^{iwx} \right) dx$$

$$= \underbrace{\int_{-\infty}^{\infty} f(x) e^{iwx} dx}_{\text{circ}} - \underbrace{\mathcal{F}\left(\frac{df}{dx}\right)}_{\mathcal{F}(f'(x))}$$

$$w \cdot f' = (w \cdot f) \xrightarrow{\text{PDE}} \mathcal{F}(f') = -w^2 \hat{f}$$

\* Fourier transform and convolution.

$$(f * g) = \int_{-\infty}^{\infty} f(x-\xi) g(\xi) d\xi$$

$$\mathcal{F}(f * g) = \mathcal{F}(f) \mathcal{F}(g) = \hat{f} \hat{g}$$

$$\mathcal{F}(f * g)(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(w) \hat{g}(w) e^{iwx} dw$$

$$\begin{aligned}
 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} f(\omega) \left( \int_{-\infty}^{\infty} g(y) e^{-i\omega y} dy \right) e^{i\omega x} d\omega \\
 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\omega) g(y) e^{i\omega(x-y)} d\omega dy \\
 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} g(y) \underbrace{\int_{-\infty}^{\infty} f(\omega) e^{i\omega(x-y)} d\omega}_{\int_{-\infty}^{\infty} f(x-y) dy} dy \\
 &= f * g
 \end{aligned}$$

\* Intuition of Fourier Transform & Laplace transform

$$! D = - \int_{-\infty}^{\infty} f(t) \cos(\omega t) dt - i \int_{-\infty}^{\infty} f(t) \sin(\omega t) dt.$$

$$\begin{aligned}
 \mathcal{F}[\cos(n\pi x)] &\rightarrow (\cos(n\pi)) [\cos(\omega x)] \\
 &\rightarrow (\cos(n\pi)) [\sin(\omega x)]
 \end{aligned}$$

$$\mathcal{F}[f(\cos(n\pi x))] = \cos(n\pi) [\cos(\pi n x)] \text{ area} = 0$$



$$\int_{-\infty}^{\infty} (\cos(n\pi)) [\cos(n\pi x)] dt = 0 \quad \text{area} = 0.$$

$$\text{Fourier: } F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$$

$$\text{Laplace: } F(s) = \int_{-\infty}^{\infty} f(t) e^{(s+i\omega)t} dt.$$

$$F(s) = \int_{-\infty}^{\infty} f(t) e^{(s+i\omega)t} e^{-st} dt$$

\* Laplace transformation of first order

~~is~~ 1st order equation. two transforms of  $f(t)$  and  $y(t)$  are  $F(s)$  and  $\bar{y}(s)$

$$\text{definition: } F(s) = \int_{-\infty}^{\infty} f(t) e^{-st} dt$$

example

$$\begin{aligned}
 f(t) &= e^{at} \\
 F(s) &= \int_{-\infty}^{\infty} e^{at} e^{-st} dt = \left[ \frac{e^{(a-s)t}}{a-s} \right]_{t=-\infty}^{t=\infty}
 \end{aligned}$$

$$s \frac{1}{s-a} = e^{-at}$$

$$\frac{dy}{dt} - ay = 0 \quad \int \frac{dy}{dt} e^{at} dt = \int y(a) + C e^{at} dt$$

$$\text{Transform } \frac{dy}{dt} = sY(s) - y(0)$$

$$Y(s) = \frac{y(0)}{s-a} \xrightarrow{\text{INVERSE LT}} y(t) = \frac{y(0)}{s-a} e^{at}$$

Implementation of Laplace transform using MATLAB

BASIC

1. clc;

2. clear all;

4. syms t f;

5. f=5\*t

6. L=laplace(f)

$$\text{out} 2 = 5/s^2.$$

complex

1. clc;

2. clear all;

4. syms L f t;

$$5. f = (\exp(-3*t) * \sin(2*t))/t$$

6. L = laplace(f).

$$\text{out} 2 = \text{atan}(2/(s+3))$$

BASIC

1. clc;

2. clear all;

3. syms t f;

$$4. f = \exp(-5*t);$$

$$5. F = \text{laplace}(f)$$

$$\text{out} 2 = 1/(s+5).$$

Matlab

1. clc;

2. clear all;

3. syms s;

$$4. F = 1/(s^2 + 1)$$

5. g = laplace(F)

$$\text{out} 2 = \sin(t)$$

Applications of Z-transform

$$1. f(\infty) = f(x+1) - f(x)$$

$$2. y_{n+1} = y_{n+2} - y_{n+1}$$

hence

$$\Delta y_{n+1} + y_n = 2$$

$$\Rightarrow y_{n+2} - y_{n+1} + y_n = 2$$

$$\Delta y_{n+1} + \Delta^2 y_{n+1} = 1$$

$$y_{n+2} - y_{n+1} + \Delta(\Delta y_{n+1}) = 1$$

Now,  $\Delta y_{n+1} = y_{n+1} - y_n$

so:

$$\Delta(\Delta y_{n+1}) = \Delta(y_{n+1} - y_n)$$

$$\Delta y_{n+1} - \Delta y_n$$

$$y_{n+1} - y_n - (y_n - y_{n-1})$$

$$= \underline{y_{n+1} - 2y_n + y_{n-1}}$$

$$\Rightarrow y_{n+2} - y_{n+1} + y_n = 2$$

$$y_{n+2} = \underline{2y_n + y_{n+1}} = 1$$

Find the Z-transform of sequence using Matlab.

\* command window

$$\rightarrow \text{syms } n;$$

>>>

$$>>> a = n+1;$$

~~$\rightarrow b = ztrans(a);$~~

$$>>> disp(b)$$

$$z/(z^2 - 1) + z/(z-1)^2$$

$$>>> pretty(b)$$

$$\frac{z}{z-1} + \frac{2}{(z-1)^2}$$

2)

$$>>> a = \omega^n n;$$

$$>>> b = ztrans(a);$$

>>> b

$$\underline{\underline{\omega z = b = z/(z^2 - 1)}}$$

3) >>> syms n w;

$$\Rightarrow a = \sin(w^n n);$$

$$\Rightarrow b = ztrans(a);$$

~~$\Rightarrow$~~  
$$\Rightarrow disp(b)$$

$$\Rightarrow (\sin(\omega)/(z^{n/2}) - 2\cos(\omega)z^{n/2+1})$$

$$\underline{\underline{\frac{\sin(\omega)}{z^{n/2}} = \frac{z \sin(\omega)}{z^2 - 2\cos(\omega)z^{n/2} + 1}}}$$

**Date:26may2020**

**Course: python**

**Topic:**

**Basics:sec20-22**

**Name:Bindushri**

**USN:4AL17EC011**

**Sem&Sec:6<sup>th</sup> A**

## AFTERNOON SESSION DETAILS

### Image of session

The screenshot shows a web browser window displaying a Udemy course page. The title of the course is 'The Python Mega Course: Build 10 Real World Applications'. On the right side of the screen, there is a sidebar titled 'Course content' which lists several video lectures. The first lecture, '168. Congratulations!', is checked off and has a duration of 1 minute. Below the course content, there are two sections: 'Section 21: Graphical User Interfaces with Tkinter' and 'Section 22: Interacting with Databases'. The 'About this course' section is also visible at the bottom left.

**Congratulations!**

Hey, I just wanted to congratulate you and tell you that it's awesome that you made it this far in the course. You have completed around 50% of the course and I know it takes patience and commitment to go this far without quitting. Since you made it this far you probably know a lot about Python by now. I promise that if you go on completing the rest of the course you will be on the right path to becoming a real Python programmer. There's a ton of fun and useful content and apps awaiting for you in the next sections.

Kudos and I'll keep in touch with you!

Course content

- 168. Congratulations! 1min

Section 21: Graphical User Interfaces with Tkinter

- 169. Introduction to Tkinter 3min
- 170. Setting up a GUI with Widgets 9min
- 171. Connecting GUI Widgets with Callback Functions 10min
- 172. Create a Multi-widget GUI (Practice) 1min
- 173. Solution 1min

Section 22: Interacting with Databases

- 0 / 6 | 45min

Overview Q&A Notes Announcements

About this course

A complete Python course for both beginners and intermediate learners. Doctor Python 3 for making 10 real world applications.

26-05-2020

## Python Script - Py

Task

Ques 20 ✓ Personal website - how to o/p user's click to

→ <https://pythonflasktutorial.herokuapp.com>.

→ PPP install flask

demo → script.py

1. from flask import Flask

2. app = Flask(\_\_name\_\_)

3.

4. @app.route('/')

5. def home():

6. return "website content goes here!"

7.

8. if \_\_name\_\_ == "\_\_main\_\_":

9. app.run(debug=True)

10.

→ to click o/p open chrome → localhost:5000

→ HTML template:

1. from flask import Flask, render\_template

2. app = Flask(\_\_name\_\_)

3. @app.route('/')

4. def home():

5. return render\_template("home.html")

6. @app.route('/about/')

7. def about():

8. return "About content goes here!"

9.

10. if \_\_name\_\_ == "\_\_main\_\_":

11. app.run(debug=True)

Notes

[Learn "home.html" should reflect your html code - that has to be created  
right click on demo (folder) → New folder  
→ (named as) temperatures → (inside the temperatures create a new file) right click on temperatures → temperatures/home.html → cutter]

# write a html file inside home.html

1. `<!DOCTYPE html>`
2. `<html>`
3. `<body>`
4.     `<h1>My homepage </h1>`
5.     `<p>This is a test website</p>`
6. `</body>`
7. `</html>`

Output # (check in Oscar host:5000)

My homepage.

-this is a test website.

[Learn - to create another html page right click on home.html → Duplicate → create about.html]

# Start writing ~~new~~ html code at about.html

1. `<!DOCTYPE html>`
2. `<html>`
3. `<body>`
4.     `<h1>my about page </h1>`
5.     `<p>This is a test website again</p>`
6. `</body>`
7. `</html>`

[open script.py] now automatically changes the code

# def about():

return render\_template("about.htm")

B)

Output: click localhost:5000/about

My about page.

This is a test website again

[Now we create one more html folder under templates i.e. "layout.html" & start writing code here]

<!DOCTYPE html>

<html>

<body>

<header>

<div class = "container">

<h1 class = "logo"> Adopt's Web App</h1>

<strong><nav>

<ul class = "menu">

<li><a href = "?url\_for('home')??">

(Home)</a>

</li>

<li><a href = "?url\_for('about')??">

(About)</a></li>

</ul>

</nav></strong>

</div>

<header>

</header>

<div>

<div class = "container">

{% block content %}

+ {% endblock %}

</div>

</body>

<html>

## Script.py (index)

```
1. from flask import Flask, render_template  
2.  
3. app = Flask(__name__)  
4.  
5. @app.route('/')  
6. def home():  
7.     return render_template("home.html")  
8.  
9. @app.route('/about')  
10. def about():  
11.     return render_template("about.html")  
12.  
13. if __name__ == "__main__":  
14.     app.run(debug=True)
```

→ [make changes as need by home.html]

```
1. % extends "layout.html" %  
2. %block content %  
3. <div class="home">  
4.     <h1> my homepage </h1>  
5.     <p> this is a test website </p>  
6. </div>  
7. %endblock %
```

→ [also in about.html]

```
1. % extends "layout.html" %  
2. %block content %  
3. <div class="about">  
4.     <h1> my about page </h1>  
5.     <p> this is a test website again </p>  
6. </div>  
7. %endblock %
```

css styling (Demo → (create static) → (create CSS) → (page main.css))

[here changes in layout.html]

```
<!DOCTYPE html>
<html>
  <head>
    <title>APP</title>
    <link rel="stylesheet" href="{{url_for('static',
      filename='css/main.css')}}">
```

```
<head>
```

```
<body>
```

```
<header>
```

```
<div class="container">
```

# rest part of code remains same as before.

O/P → # can be checked in localhost:5000.

→ Creating Python Virtual Environment

\* pip install virtualenv

[Carue → Demo (here you has all the files (script.py))

→ Create another new folder near to Demo. →

Mysite. → and add paste Demo inside mysite.

→ Now open right VSC in Demo because venv  
has to be created here]

open terminal:-

→ Python -m venv virtual  
this creates folder].

→ virtual\scripts\python.

→ 12 (ctrl+5)

→ virtual\scripts\pp install flask

→ virtual\scripts\python Demo\script1.py

## HTTP & Git - SCM from Downloads

### Deploying the website to a live server

1<sup>st</sup> Step:

~~https://heroku.com~~ https://www.heroku.com  
 Signup → Create an account  
 + then login

2<sup>nd</sup> Step: heroku marketplace [Download]

3<sup>rd</sup> Step: Open your folder Mysite → Demo →  
 → [static, templates, script.py]. Right click →  
 cmd window

[login here to heroku]

4<sup>th</sup> Step: > heroku login

Enter your heroku credentials:

Email:

Password:

5<sup>th</sup> Step: Finder to create your app

→ → create 'Any name'

[this will create a app in heroku]

6<sup>th</sup> Step:

Pre upload files need a grif.

→ pip freeze

[this will point all the packages]

7<sup>th</sup> Step: & ~~Requirements.txt~~

.. \ virtual\ scripts\ pip freeze

8<sup>th</sup> Step: pip install gunicorn

[this is http server]

9<sup>th</sup> Step: .. \ virtual\ scripts\ pip freeze > requirements.txt

[this creates first file in our directory]

10<sup>th</sup> Step: Create new file [profile] Procfile [pushed to Demo]

Open VSC in profile.

1. Web: gunicorn Script: app

Step 1: Create one more file besides Democ  
i.e runtime.txt

and open in VSC or note pad

and write text

Python 3.5.1      60 - 2.7.11

Step 2 cmd window > git init

Step 3 git add .

Step 4 git commit -m "first"

Step 5 heroku git:remote --app Augano

Step 6 git push heroku master

# If it fails then recognise the repository then

Step 7 git config --global user.name "Rudra@gmail.com"

Step 8 git config --global user.name "Bhaduri"

Step 9 git push heroku master

~~File~~ → ~~Run~~ → ~~Script1.py~~

~~File~~ → ~~Run~~ → ~~Script1.py~~

CLASSWORK

Date \_\_\_\_\_  
Page \_\_\_\_\_

## Section 21: Graphical user interfaces (using Tkinter)

1. Import Tkinter from Tkinter import \*
2. window = Tk()
- 3.
4. def km\_to\_miles():  
 Print(e1\_value.get())  
 miles = float(e1\_value.get()) \* 1.6  
 t1.insert(END, miles)
- 5.
- 6.
- 7.
8. b1 = Button(window, text="Execute", command=km\_to\_miles)
9. b1.grid(row=0, column=0)
- 10.
11. e1\_value = StringVar()
12. e1 = Entry(window, textvariable=e1\_value)
13. e1.grid(row=0, column=1)
- 14.
15. t1 = Text(window, height=1, width=20)
16. t1.grid(row=0, column=2)
- 17.
18. window.mainloop()

Database → script2.py.

## Section 22: Interacting with Databases

\* Here require 2 libraries → sqlite3 → psycopg2

1. Import sqlite3.  
 conn = sqlite3.connect("lite.db")
2. cur = conn.cursor()  
 cur.execute("CREATE TABLE IF NOT EXISTS  
 items TEXT, quantity INTEGER,  
 price REAL")
3. cur.execute("SELECT \* FROM items")  
 print(cur.fetchall())
4. cur.execute("UPDATE items SET quantity = 100 WHERE item = 'apple'")  
 conn.commit()
5. cur.execute("DELETE FROM items WHERE item = 'apple'")  
 conn.commit()
6. cur.execute("SELECT \* FROM items")  
 print(cur.fetchall())

(on commit)

(on close)

def insert(item, quantity, price):

conn = sqlite3.connect("lite.db")

cur = conn.cursor()

cur.execute("INSERT INTO store VALUES (?, ?, ?)",

(item, quantity, price))

conn.commit()

conn.close()

insert("coffee cup", 10, 5)

def view():

conn = sqlite3.connect("lite.db")

cur = conn.cursor()

cur.execute("SELECT \* from store")

rows = cur.fetchall()

conn.close()

return rows

.print(view())







