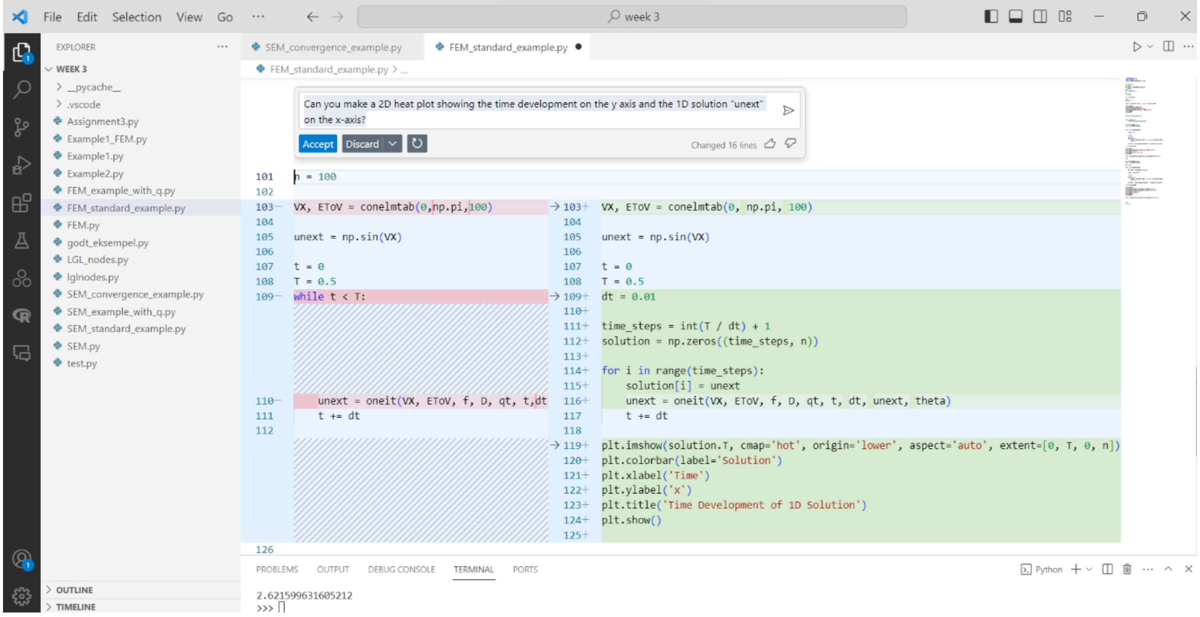
Chatbot examples

In our project we have mainly used github copilot in order to make changes in our code e.g. quickly set loops or plots with mixed success. We have also used ChatGPT 4 in order to create a solution example for the heat equation.

Example 1:

We ask github copilot to create a heatmap plot for our time dependent heat equation based on our code that already creates a solution for each timestep.



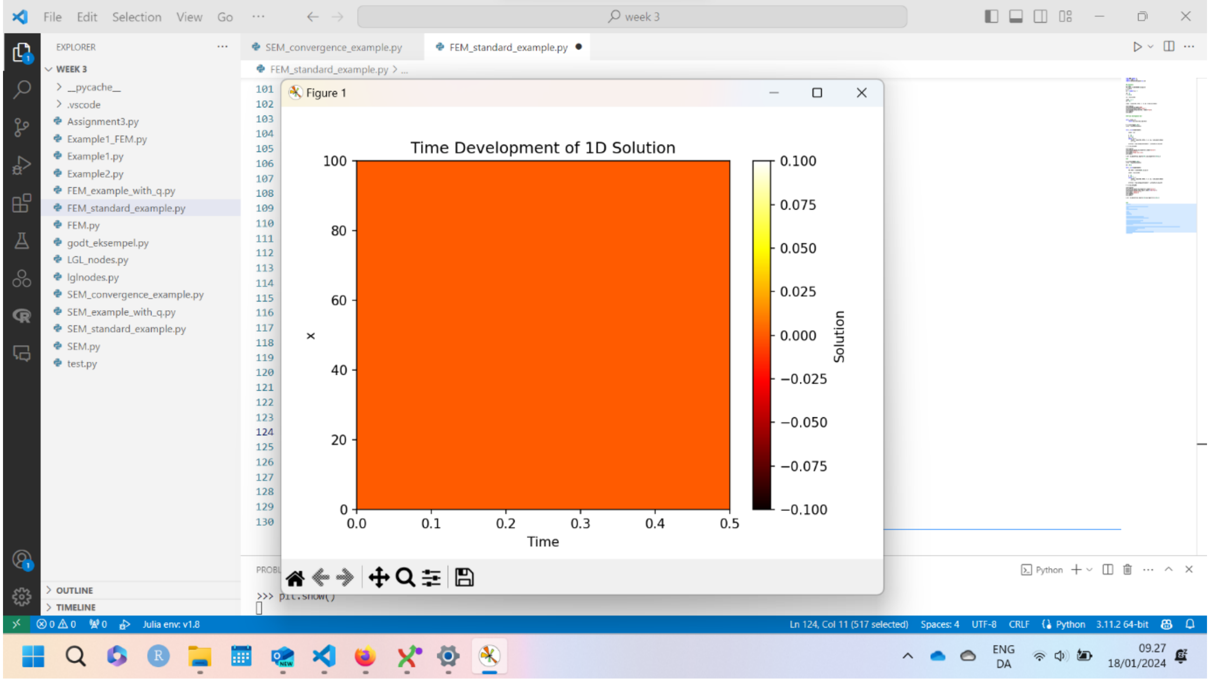
Can you make a 2D heat plot showing the time development on the y axis and the 1D solution "unext" on the x-axis?

```python
n = 100

VX, EToV = conelmtab(0,np.pi,100)

unext = np.sin(VX)

t = 0
T = 0.5
while t < T:
    unext = oneit(VX, EToV, f, D, qt, t,dt
    t += dt

```

```python
n = 100

VX, EToV = conelmtab(0, np.pi, 100)

unext = np.sin(VX)

t = 0
T = 0.5
dt = 0.01

time_steps = int(T / dt) + 1
solution = np.zeros((time_steps, n))

for i in range(time_steps):
    solution[i] = unext
    unext = oneit(VX, EToV, f, D, qt, t, dt, unext, theta)
    t += dt

plt.imshow(solution.T, cmap='hot', origin='lower', aspect='auto', extent=[0, T, 0, n])
plt.colorbar(label='Solution')
plt.xlabel('Time')
plt.ylabel('x')
plt.title('Time Development of 1D Solution')
plt.show()

```

```
2.621599631605212
>>> 
```

The result

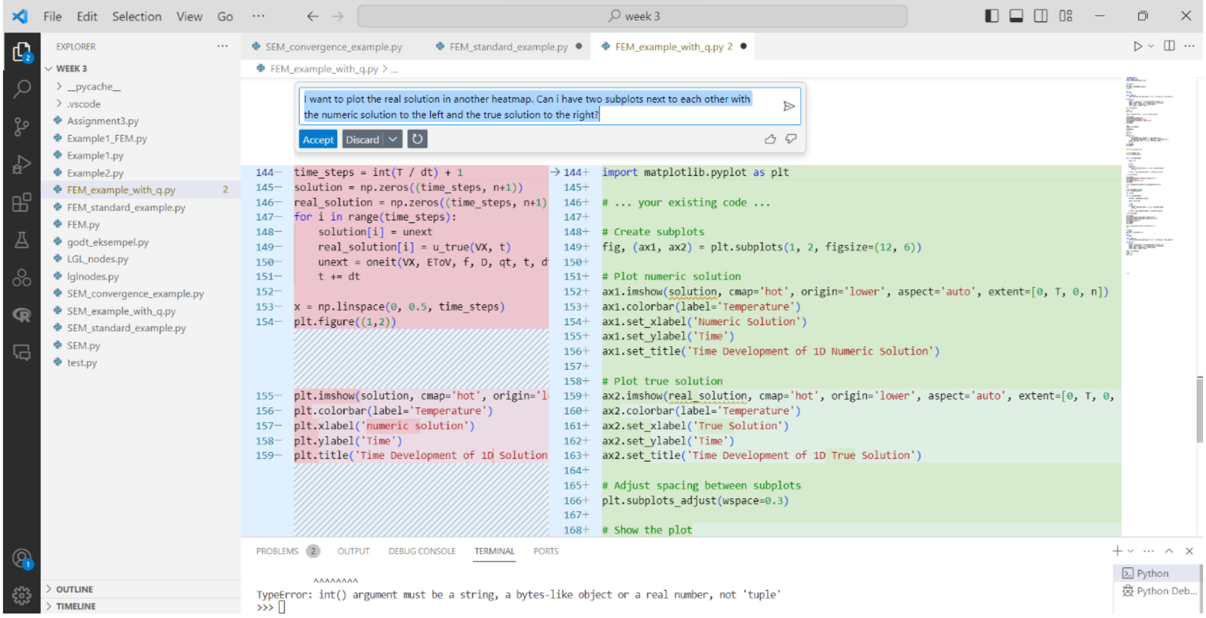


```
>>> plt.show()
```

Obviously not what we desired. It has flipped the axis opposite to what was prompted, however with some small changes it was easy to generate the desired plot.

Example 2:

After generating the correct and desired heatmap we ask the github to generate a similar plot with the exact solution next to the numerical plot.



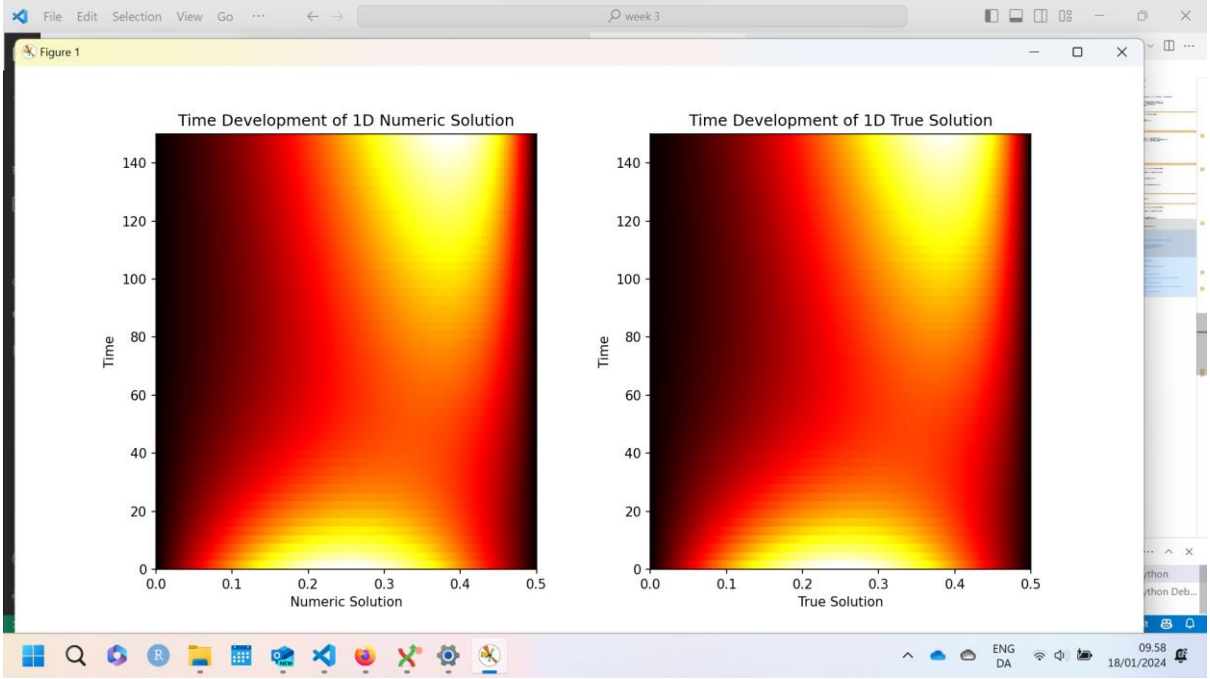
I want to plot the real solution in another heatmap. Can i have two subplots next to each other with the numeric solution to the left and the true solution to the right?

```
time_steps = int(T / dt) + 1
solution = np.zeros((time_steps, n+1))
real_solution = np.zeros((time_steps, n+1)
for i in range(time_steps):
    solution[i] = unext
    real_solution[i] = u_true(VX, t)
    unext = oneit(VX, EToV, f, D, qt, t, d
    t += dt

x = np.linspace(0, 0.5, time_steps)
plt.figure((1,2))
plt.imshow(solution, cmap='hot', origin='l
plt.colorbar(label='Temperature')
plt.xlabel('numeric solution')
plt.ylabel('Time')
plt.title('Time Development of 1D Solution
```

```
import matplotlib.pyplot as plt

# ... your existing code ...

# Create subplots
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 6))

# Plot numeric solution
ax1.imshow(solution, cmap='hot', origin='lower', aspect='auto', extent=[0, T, 0, n])
ax1.colorbar(label='Temperature')
ax1.set_xlabel('Numeric Solution')
ax1.set_ylabel('Time')
ax1.set_title('Time Development of 1D Numeric Solution')

# Plot true solution
ax2.imshow(real_solution, cmap='hot', origin='lower', aspect='auto', extent=[0, T, 0,
ax2.colorbar(label='Temperature')
ax2.set_xlabel('True Solution')
ax2.set_ylabel('Time')
ax2.set_title('Time Development of 1D True Solution')

# Adjust spacing between subplots
plt.subplots_adjust(wspace=0.3)

# Show the plot
```

```
        ^^^^^^^^
TypeError: int() argument must be a string, a bytes-like object or a real number, not 'tuple'
>>> 
```

Which gives us the following desired result. The code probably made it easy for the chatbot to understand our prompt.

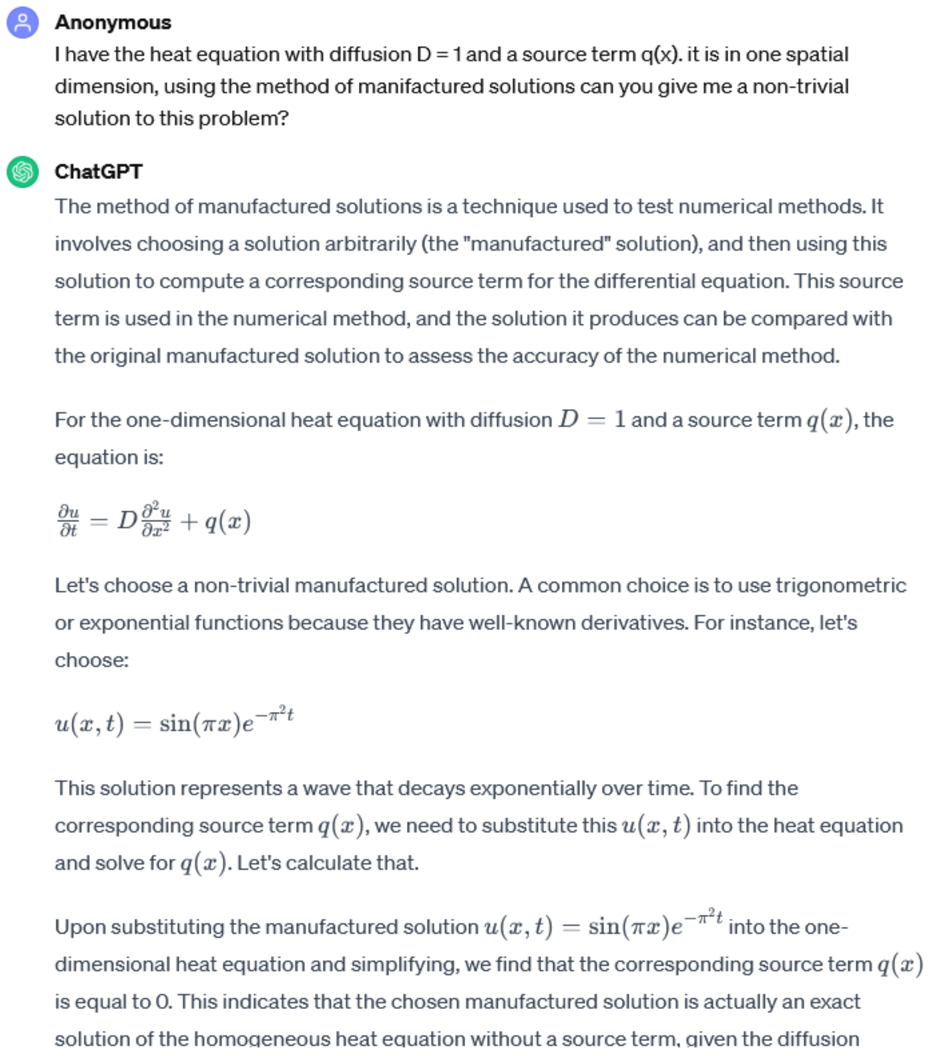


Example 3:

In this example we ask ChatGPT 4 to generate an example for a solution for the heat equation in the desired settings. It gives us a solution $u(x,t) = \sin(\pi x)\, e^{-\pi^2 t} + x(1-x)t$.

and we only needed to rescale it. However, in order to make the source term a bit more interesting we also multiplied by e^x and $k = 0.1$ in order to give a less symmetric heat source and control the effect of the heat source. Therefor we ended up having $u(x,t) = \sin(x)\, e^{-\pi^2 t} + 0.1x(\pi - x)e^{x}t$, which was inspired by the input from ChatGPT 4 as seen below.

This conversation was followed by some error checking to make sure that the solution and corresponding $\tilde{q}$ was correct.



Anonymous

I have the heat equation with diffusion D = 1 and a source term q(x). it is in one spatial dimension, using the method of manifactured solutions can you give me a non-trivial solution to this problem?

ChatGPT

The method of manufactured solutions is a technique used to test numerical methods. It involves choosing a solution arbitrarily (the "manufactured" solution), and then using this solution to compute a corresponding source term for the differential equation. This source term is used in the numerical method, and the solution it produces can be compared with the original manufactured solution to assess the accuracy of the numerical method.

For the one-dimensional heat equation with diffusion $D = 1$ and a source term $q(x)$, the equation is:

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} + q(x)$$

Let's choose a non-trivial manufactured solution. A common choice is to use trigonometric or exponential functions because they have well-known derivatives. For instance, let's choose:

$$u(x,t) = \sin(\pi x)e^{-\pi^2 t}$$

This solution represents a wave that decays exponentially over time. To find the corresponding source term $q(x)$, we need to substitute this $u(x,t)$ into the heat equation and solve for $q(x)$. Let's calculate that.

Upon substituting the manufactured solution $u(x,t) = \sin(\pi x)e^{-\pi^2 t}$ into the one-dimensional heat equation and simplifying, we find that the corresponding source term $q(x)$ is equal to 0. This indicates that the chosen manufactured solution is actually an exact solution of the homogeneous heat equation without a source term, given the diffusion coefficient $D = 1$.

To create a non-trivial source term, we should choose a different manufactured solution that does not already satisfy the homogeneous heat equation. Let's try another manufactured solution:

$$u(x,t) = \sin(\pi x)e^{-\pi^2 t} + x(1-x)t$$

This solution combines a sinusoidal wave that decays over time with a polynomial in x and linear in t. Let's calculate the corresponding source term $q(x)$ for this new solution. [>_]

The corresponding source term $q(x)$ for the new manufactured solution $u(x,t) = \sin(\pi x)e^{-\pi^2 t} + x(1-x)t$ in the one-dimensional heat equation with diffusion $D = 1$ is calculated as $2t - x(x-1)$.

This source term is non-trivial and varies with both x and t. It ensures that when substituted into the heat equation, it results in the manufactured solution $u(x,t)$. This provides a more interesting and complex test case for assessing the accuracy of numerical methods applied to solve the heat equation with a source term. [>_]

Anonymous

Can we find another source term?

ChatGPT

Certainly, we can find another source term by choosing a different manufactured solution for the one-dimensional heat equation with diffusion $D = 1$. Let's select a new manufactured solution that is more complex and ensures a non-trivial source term.

A good approach is to combine different types of functions, such as trigonometric, polynomial, and exponential functions. For example, let's consider:

$$u(x,t) = x^2 \sin(\pi x)e^{-t} + \cos(2\pi x)t^2$$

This solution includes both sinusoidal and polynomial components in x and t, which should lead to an interesting source term $q(x)$. Let's calculate the corresponding source term $q(x)$ for this manufactured solution.

Error analyzing

It appears I forgot to import the `cos` function from the `sympy` library. Let me correct that and calculate the source term $q(x)$ again for the new manufactured solution $u(x,t) = x^2 \sin(\pi x)e^{-t} + \cos(2\pi x)t^2$. [>_]

The corresponding source term $q(x)$ for the manufactured solution $u(x,t) = x^2 \sin(\pi x)e^{-t} + \cos(2\pi x)t^2$ in the one-dimensional heat equation with diffusion $D = 1$ is:

$$q(x) = (2t(2\pi^2 t + 1)\exp(t)\cos(2\pi x) - x^2\sin(\pi x) + \pi^2 x^2 \sin(\pi x) - 4\pi x\cos(\pi x) - 2\sin(\pi x))\exp(-t)$$

This source term is more complex and varies with both x and t. It represents a non-trivial case for numerical methods solving the heat equation with a source term. [>_]