Gliwice, 27.11.2024

Mathematical modelling and computer simulations in theory and practice

Documentation of laboratory task no 3

Title: **Function Approximation**

Author: Aleksandra Stachecka
Field of studies: Informatics sem.V

1 Project Objective

The goal of the project is to calculate the approximation of the given function using three mathematical techniques: Fourier series, Taylor series and Pade approximant, then visualize the results through plots comparing the accuracy of these methods.

2 Description

At the beginning we will define the functions that will be used later to evaluate the accuracy of the approximations:

- $f(x) = Sin(x) + Cos(2x)$
- $g(x) = e^x$
- $h(x) = Log(1 + x)$
- $y(x) = \begin{cases} 1 & \text{if } -\pi \leq x \leq 0, \\ 0 & \text{if } 0 < x \leq \pi. \end{cases}$
- $z(x) = 1/(1 + x^3)$

The program consists of three modules, each dedicated to a specific type of approximation, returning a plot that compares the original function with its approximation.

2.1 Fourier Series

The first section is the `AnimateFourier` function, which takes three inputs: the function to approximate *func*, the period of the function *period* (which will in most cases be 2 pi, but in this function the user can try another period), and the number of terms in the Fourier series to include *maxRank*. The function computes Fourier series approximations up to the specified number of terms. These approximations are stored in a table.

```
fourierTerms =
Table[Normal[FourierSeries[func, x, n]], {n, 1, maxRank}];
```

For each approximation there will be generated a plot showing the original function in purple alongside the current Fourier approximation in green, covering the interval form $-T$ to T ($T = period$).

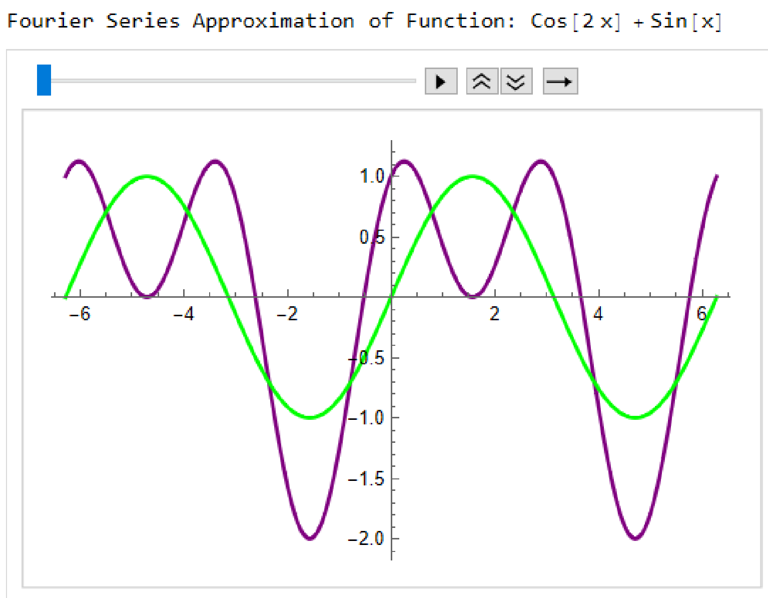
```
frames = Table[Show[
  Plot[func, {x, -T, T},
   PlotStyle -> Purple,
   PlotRange -> All],
  Plot[fourierTerms[[n]], {x, -T, T},
   PlotStyle -> Green]],
 {n, 1, maxRank}
 ];
```

These plots are combined into a sequence of frames, which are then displayed as an animation.

```
ListAnimate[frames]
```

2.1.1 Observations

After testing the `AnimateFourier[func, period, maxRank]` function we can observe that the Fourier series is the most effective at approximating periodic and trigonometrical functions, because as we can see the first frame of the animation is already quite accurate approximation of the function comparing to the first frames of other approximations.



2.2 Taylor Series

In the second section we will be using Taylor series to approximate functions. We have `AnimateTaylor` function, which takes two inputs: the function to approximate *func*, and the number of terms to include in the Taylor series *maxRank*. The approximations are computed for the given function around the point $x = 0$ and up to the specified number of terms. These approximations are stored in table.

```
taylorTerms = Table[Normal[Series[func, {x, 0, n}]], {n,
    1, maxRank}];
```

For each approximation, a plot is generated. The original function (shown in purple) is plotted alongside the current Taylor series approximation (shown in green) over the interval $[-5, 5]$.

```
frames = Table[Show[
Plot[func, {x, -5, 5},
 PlotStyle -> Purple,
 PlotRange -> All],
Plot[taylorTerms[[n]], {x, -5, 5},
 PlotStyle -> Green]],
{n, 1, maxRank}];
```

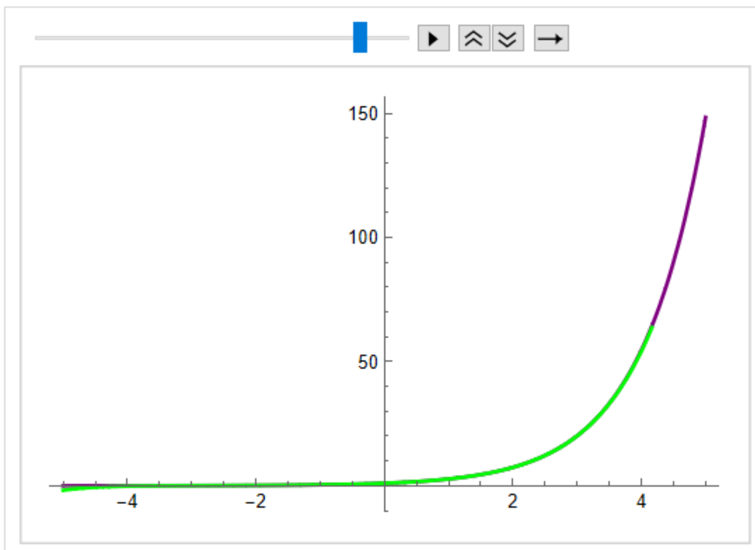
The generated frames are combined into an animation.

```
ListAnimate[frames]
```

2.2.1 Observations

After testing the `AnimateTaylor[func, maxRank]` function, we can see that Taylor series is the most effective at approximating exponential functions.

Taylor Series Approximation of Function: e^x



2.3 Pade Approximant

In the third section we will use Pade approximations to approximate functions. We have `AnimatePade` function which takes two inputs: the function to approximate $func$, and the maximum order of the Pade approximation $maxOrder$. The Pade approximations are computed for the given function around the point $x = 0$ up to the specified order. These approximations are stored in a table.

```
    padeApprox s =
  Table[With[{approx = PadeApproximant[func, {x, 0, n}]},
     approx],
   {n, 1, maxOrder}];
```

For each approximation, a plot is generated. The original function (shown in purple) is plotted alongside the current Pade approximation (show in green) over the interval $[-5, 5]$ - similar to Taylor series approximation.

```
    frames = Table[Show[
    Plot[func, {x, -5, 5},
     PlotStyle -> Purple,
     PlotRange -> All],
    Plot[padeApproxs[[n]], {x, -5, 5},
```

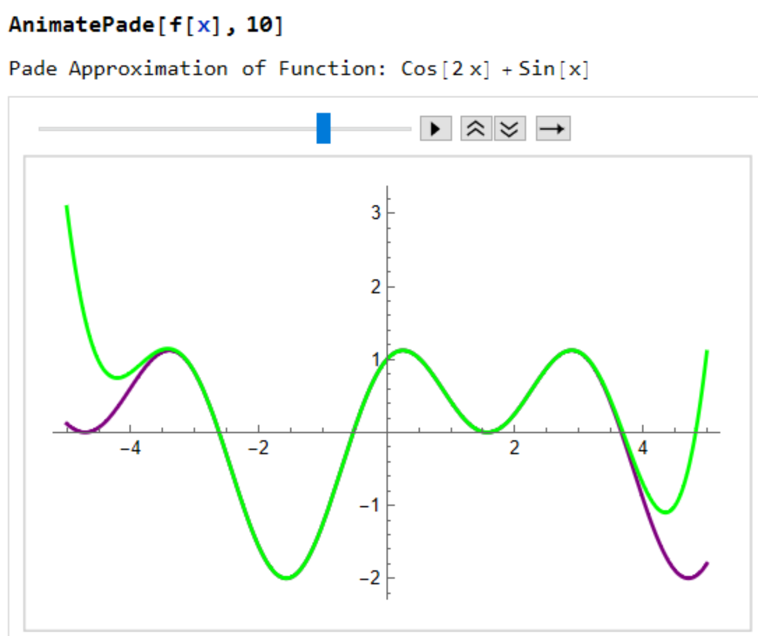
```
    PlotStyle -> Green]],
  {n, 1, maxOrder}];
```

These plots are combined into a sequence of frames, which are then displayed as an animation.

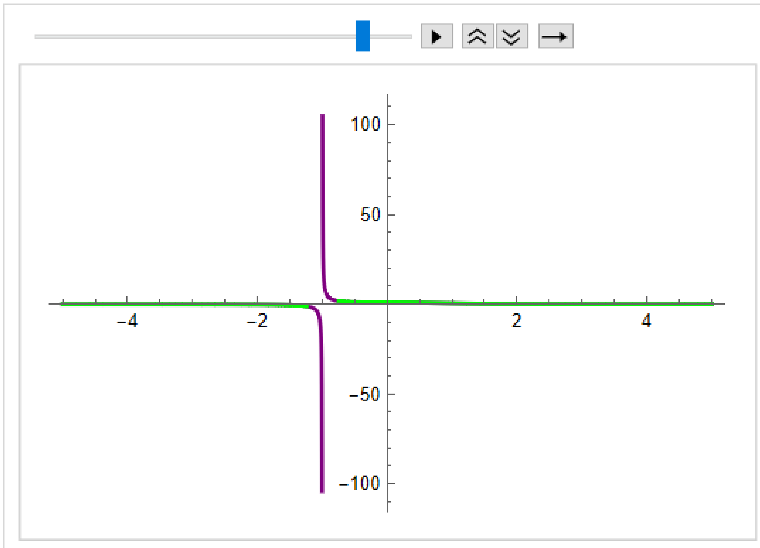
```
ListAnimate[frames]
```

2.3.1 Observations

After testing the `AnimatePade[func, maxOrder]` function on our set of functions, we can observe that Pade approximations are quite effective for trigonometrical functions or exponential functions, also functions with singularities. However based on our observations it is better to approximate these functions using Taylor series, especially when higher accuracy is required. Also Pade approximant seems to have the biggest problem approximating piecewise function.



Pade Approximation of Function: $\frac{1}{1+x^3}$



3 Enclosures

"STACHECKA Aleksandra Project 3.nb"