SPRAWOZDANIE

Zajęcia: Matematyka Konkretna

Prowadzący: prof. dr hab. inż. Vasyl Martsenyuk

|  |  |
| --- | --- |
| Laboratorium Nr 6  Data 27.04.2024  Temat: „Funkcje aktywacji w sieciach neuronowych"  Wariant 6 | Adam Kubliński  Informatyka  II stopień, niestacjonarne, zaoczne,  II semestr, gr. 1A |

Repozytorium GitHub: https://github.com/Adamadacho/MK\_REP

# Polecenie: wariant 6 zadania

# Poprowadzić badania funkcji aktywacji zgodnie z wariantem zadania z użyciem Python. Obliczyć gradient funkcji. Wyświetlić funkcję wraz z gradientem na jednym wykresie. We wniosku opisać zagadnienia w których używają daną funkcję aktywacji. Funkcja Swish.

# Opis opracowanego programu

Zastosowany kod:

import numpy as np

import matplotlib.pyplot as plt

def sigmoid(x):

return 1 / (1 + np.exp(-x))

def swish(x, beta=1):

return x \* sigmoid(beta \* x)

def swish\_derivative(x, beta=1):

sig = sigmoid(beta \* x)

return sig + beta \* x \* sig \* (1 - sig)

# Zakres danych dla wykresu

x = np.linspace(-10, 10, 400)

# Obliczanie wartości funkcji i pochodnej

y\_swish = swish(x)

y\_swish\_derivative = swish\_derivative(x)

# Rysowanie wykresów

plt.figure(figsize=(10, 6))

plt.plot(x, y\_swish, label='Swish Function')

plt.plot(x, y\_swish\_derivative, label='Gradient of Swish', linestyle='--')

plt.title('Swish Function and its Gradient')

plt.xlabel('x')

plt.ylabel('f(x) and f\'(x)')

plt.legend()

plt.grid(True)

plt.show()

# Wyniki

A graph with a blue line

Description automatically generated