

CNN síť:

<https://poloclub.github.io/cnn-explainer/>

<https://colab.research.google.com/>

V Colabu zkuste tyto prompty pro Gemini (generovat kód s AI), které vám vytvoří spustitelný kód vy pythonu pro Colab:

- 1) Načti mi CIFAR-10 dataset a zobraz mi nějaké obrázky z tohoto datasetu
- 2) Neuronová síť z <https://poloclub.github.io/cnn-explainer/>, stejná architektura

Prompt: Nyní mi napiš kód na tento model:

2 konvoluční vrstvy (každá 10 filtrů 3×3 , ReLU)

MaxPooling (2×2)

2 konvoluční vrstvy (každá 10 filtrů 3×3 , ReLU)

MaxPooling (2×2)

Flatten

Dense vrstva (10 neuronů, softmax = klasifikace do 10 kategorií)

a spusť trénink na pět epoch na CIFAR-10. Před spuštěním zobraz `model.summary()`.

Vždy po každé epoše mi zobraz validation accuracy

- 3) Nyní mi napiš kód na tento model:

Konvoluční vrstva (Conv2D): Má **32 filtrů** velikosti **3×3** , aktivační funkce **ReLU**.

Max Pooling vrstva (MaxPooling2D): **2×2** px

Konvoluční vrstva (Conv2D): **64 filtrů** velikosti **3×3** , opět **ReLU**.

Max Pooling vrstva 2×2 .

Konvoluční vrstva (Conv2D): **128 filtrů** velikosti **3×3** , aktivační funkce **ReLU**.

Flatten vrstva

Dense vrstva (Fully-connected): **10 neuronů**, aktivační funkce **softmax**

a spusť trénink na 5 epoch na CIFAR-10. Před spuštěním zobraz `model.summary()`

Vždy po každé epoše mi zobraz validation accuracy

- 4) Napiš mi kód, který bude vycházet z předchozí buňky a který mi ukáže náhodně 5 obrázků kde se můj model trefil a náhodně 5 kde se můj model netrefil. Vždycky napiš predikovanou třídu a ground truth třídu. Převeď čísla tříd na text.
- 5) Bonus: Model jen s jednou konvoluční vrstvou, který bude mít hodně parametrů v porovnání z předchozími modely (díky vysokém rozlišení obrazu, které vstupuje do flatten vrstvy), ale nebude fungovat lépe. Prompt pro Gemini:

Napiš kód na deep learningový model s jednou konvoluční vrstvou 3x3 px a aktivační funkcí ReLU, jednou flatten vrstvou a jednou dense vrstvou. A spusť trénink na tom CIFAR-10 datasetu. Před spuštěním zobraz `model.summary()`