

Historie kybernetiky a umělé inteligence

07. Historie neuronových sítí

Miloš Železný

Katedra kybernetiky
Fakulta aplikovaných věd
Západočeská univerzita v Plzni

24. listopadu 2025

Historie neuronových sítí

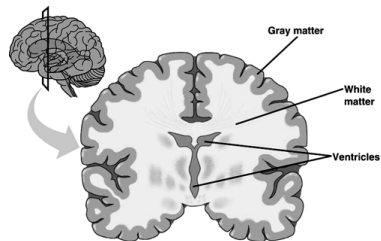
Téma přednášky

- ▶ Stručný přehled vývoje neuronových sítí od 40. let 20. století po současnost
- ▶ Od biologické inspirace k matematickému modelování
- ▶ Klíčové přístupy, metody a myšlenky v různých obdobích vývoje

Co je to neuron?

Základní struktura lidského mozku

- ▶ Lidé se snažili pochopit funkci mozku tisíce let
- ▶ Nejmenší jednotka nervového systému se nazývá **neuron**
- ▶ Mozek člověka obsahuje přibližně **100 miliard neuronů** (\pm několik miliard)
- ▶ Neuron je speciální typ buňky schopné vést, zpracovávat a vysílat signál
- ▶ Poprvé popsán roku 1835 (Jan Evangelista Purkyně)



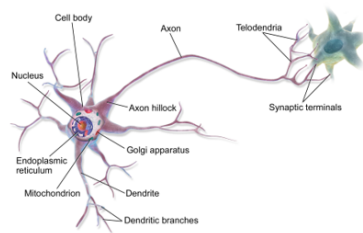
Neuron – základní části

Hlavní části neuronu

- ▶ **Jádro** – zpracování informací
- ▶ **Dendrity** – příjem signálů
- ▶ **Axon** – výstupní signál
- ▶ **Synapse** – propojení neuronů

Shrnutí

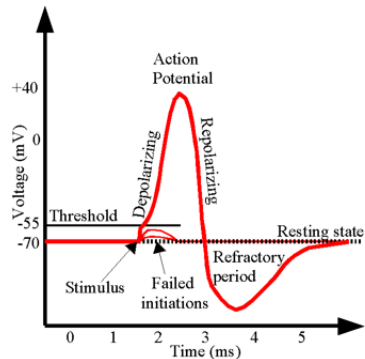
Neuron je buňka schopná elektricko-chemického přenosu signálu a tvoří základní stavební prvek mozku.



Jak funguje biologický neuron?

Princip činnosti

- ▶ **Dendrity** slouží jako příjemce vzruchu
- ▶ Signály se **akumulují v těle neuronu**
- ▶ Po překročení určité prahové hodnoty neuron **vysílá impuls axonem** do dalších neuronů
- ▶ Neurony jsou propojeny tzv. **synapsemi** – axon jednoho neuronu propojuje na dendrit dalšího
- ▶ Chování neuronu odpovídá principu: „**všechno nebo nic**“



Co je to umělá neuronová síť?

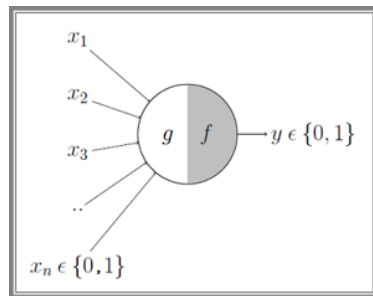
Základní charakteristika

- ▶ Umělá neuronová síť je **matematický model** inspirovaný funkcí biologických neuronů
- ▶ Topologie inspirovaná lidským/zvířecím (obecně biologickým) mozkem
- ▶ Skládá se ze silně propojených výpočetních jednotek – **umělých neuronů**
- ▶ Hlavní rozdíl – funguje synchronně
- ▶ Neurony jsou uspořádány do vrstev: vstupní, skryté a výstupní
- ▶ Učení probíhá pomocí úpravy vah spojení mezi neurony
- ▶ Jedná se o jeden z klíčových přístupů moderní umělé inteligence

McCulloch a Pitts (1943)

První formální model neuronu

- ▶ Warren McCulloch a Walter Pitts v roce 1943 publikují model binárního neuronu
- ▶ Neuron je popsán jako **prahová logická jednotka**
- ▶ Vstupy vážené, výstup aktivován při překročení prahu
- ▶ Ukázali, že neuronové sítě dokáží implementovat **logické funkce** i konečné automaty
- ▶ Šlo o první zásadní krok směrem k teoretickým základům AI



Hebbovo pravidlo (1949)

„Neurons that fire together, wire together“

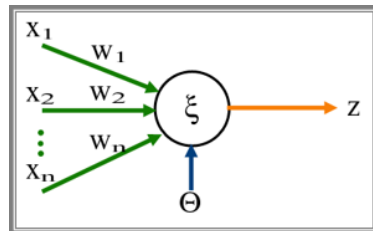
- ▶ Donald Hebb formuloval biologicky inspirované pravidlo učení
- ▶ Spojení mezi dvěma neurony se posiluje, pokud jsou aktivní současně
- ▶ Hebbovké učení – učení na základě plasticity biologických neuronů – tedy na základě Hebbova pravidla
- ▶ Prahová logika – převod spojitého vstupu na diskrétní výstup – Základ mnoha typů učení v raných neuronových sítích
- ▶ Ještě nešlo o dnešní složité metody gradientního učení, ale šlo o zásadní inspiraci pro další práci



Rosenblattův perceptron (1957)

Perceptron

- ▶ Vytvořil Frank Rosenblatt v roce 1957, inspirován biologickým zrakovým systémem (pozoroval, jak oko mouchy ovládá únikovou reakci)
- ▶ Schopen rozlišovat jednoduché vzory
- ▶ Zaveden **učení na základě chyb** (error-driven learning)
- ▶ Umožnil řešit lineárně separovatelné problémy



Pravidlo učení perceptronu

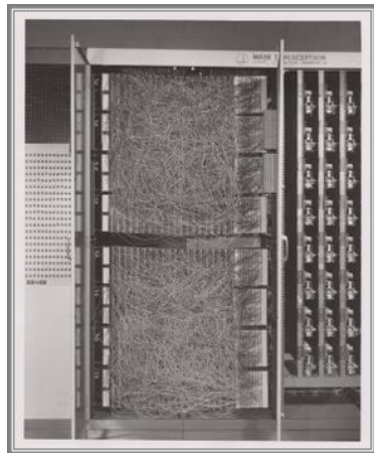
Jak perceptron upravuje váhy

- ▶ Perceptron – počítá váženou sumu vstupů, odečítá práh a na výstupu má jednu ze dvou možných hodnot – 0;1
- ▶ Pokud perceptron klasifikuje vzor správně, váhy se nemění
- ▶ Pokud je výsledek chybný:
 - ▶ váhy se upraví směrem k požadovanému výstupu
- ▶ Jedná se o formu **učení učitelem (supervised learning)**
- ▶ Pravidlo je podobné gradientnímu sestupu, ale jednodušší

Mark I Perceptron

První strojová implementace

- ▶ První prakticky implementovaná neuronová síť — stroj *Mark I Perceptron*
- ▶ Připojena na kameru 20x20 pixelů
- ▶ Váhy implementovány pomocí potenciometrů
- ▶ Změna váhy probíhala změnou odporu potenciometru pomocí elektromotoru



První použití pro reálný problém

ADALINE a MADALINE (1959)

- ▶ 1959 – ADALINE a MADALINE – Stanford University – Widrow a Hoff (*(Multiple) ADaptive LINear Elements*, dříve *ADaptive LInear NEuron*)
- ▶ Založeno na modelu neuronu od McCullocha a Pittse
- ▶ Úprava vah je při trénování na rozdíl od perceptronu McCullocha a Pittse založena na vážené sumě ještě před průchodem aktivační funkcí
- ▶ ADALINE jako adaptivní filtr sloužil k odstranění ozvěny v telefonních hovorech
- ▶ MADALINE byla první neuronová síť aplikovaná na reálný problém
- ▶ Tento typ neuronové sítě se stále komerčně využívá

Minsky a Papert (1969): Kritika perceptronu

Kniha *Perceptrons*

- ▶ Marvin Minsky a Seymour Papert ukázali omezení perceptronu
- ▶ Perceptron nedokáže řešit problémy, které nejsou **lineárně separovatelné**
- ▶ Neumí ani jednoduchou úlohu jako **XOR**
- ▶ Tento „neúspěch“ způsobil stagnaci vývoje neuronových sítí až do začátku osmdesátých let
- ▶ Tzv. první „AI winter“

70. léta: stagnace ve výzkumu neuronových sítí

Důvody stagnace

- ▶ Přílišná očekávání – vědci doufali v rychlý pokrok, který nepřišel
- ▶ Výpočetní technika byla příliš pomalá
- ▶ Finanční podpora byla výrazně snížena
- ▶ Lineární perceptrony byly nedostatečné a nebyly známy efektivní metody učení hlubších sítí

80. léta: renesance neuronových sítí

Důvody nového zájmu

- ▶ Objev backpropagation umožnil trénovat hlubší sítě
- ▶ Zvýšení výpočetního výkonu počítačů
- ▶ Zájem o modely s pamětí (Hopfield) a organizací (Kohonen)
- ▶ Zlepšila se matematická teorie neuronových sítí
- ▶ Nové aplikace: rozpoznávání řeči, predikce, klasifikace

Backpropagation (Rumelhart, Hinton, Williams – 1986)

Zpětné šíření chyby

- ▶ Metoda umožnila trénovat vícevrstvé neuronové sítě (MLP)
- ▶ Spočítá chybu na výstupu a šíří ji zpět skrz síť
- ▶ Váhy se upravují pomocí gradientního sestupu
- ▶ Gradientní sestup (angl. „gradient descent“) je optimalizační metoda, která upravuje váhy směrem, kde chyba modelu nejrychleji klesá.
- ▶ Poprvé umožnila praktické učení sítí se skrytými vrstvami
- ▶ Výrazně oživila výzkum neuronových sítí v 80. letech

Vícevrstvý perceptron (MLP)

Architektura a princip

- ▶ Vícevrstvý perceptron (MLP) je neuronová síť složená ze:
 - ▶ **vstupní vrstvy**
 - ▶ **jedné nebo více skrytých vrstev**
 - ▶ **výstupní vrstvy**
- ▶ Každá vrstva obsahuje sadu neuronů, které jsou plně propojeny s vrstvou následující
- ▶ MLP využívá **nelineární aktivační funkce** (např. sigmoid, tanh, ReLU), což mu umožňuje modelovat i nelineární vztahy
- ▶ Na rozdíl od jednoduchého perceptronu dokáže řešit i problémy, které **nejsou lineárně separovatelné** (např. XOR)
- ▶ Učení probíhá pomocí algoritmu **backpropagation**
- ▶ MLP je jedním z nejrozšířenějších a nejklassičtějších typů neuronových sítí a tvoří stavební základ moderních hlubokých sítí



Hopfieldovy síť (1982)

John Hopfield

- ▶ Zavedl tzv. **asociativní paměťové síť**
- ▶ Síť se chová jako dynamický systém – hledá minimum energetické funkce
- ▶ Schopná dokončovat vzory z neúplných nebo poškozených dat
- ▶ První hlubší propojení neuronových sítí s fyzikou (Isingův model)
- ▶ Důležitý krok směrem k moderním rekurentním sítím

Kohonenovy samoorganizující mapy (SOM, 1982)

Teuvo Kohonen

- ▶ Zavádí koncept **samoorganizujících se map** (SOM)
- ▶ Modeluje organizaci sensorických oblastí mozku
- ▶ Neřízené učení – síť se sama přizpůsobuje vstupním datům
- ▶ Vytváří 2D mapu, kde podobné vzory jsou umístěny blízko sebe
- ▶ Významná metoda pro vizualizaci dat a clustering

Boltzmannovy stroje (Hinton, 1985)

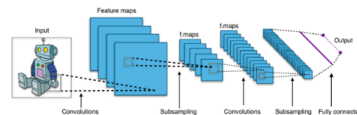
Stochastický neuronový model

- ▶ Geoffrey Hinton zavádí **Boltzmannovy stroje** – stochastické neuronové sítě
- ▶ Učení pomocí energetické funkce (podobné Hopfieldovým sítím)
- ▶ Schopnost modelovat složité distribuované reprezentace
- ▶ Varianta: **RBM (Restricted Boltzmann Machine)** – později klíčová pro hluboké učení

Konvoluční neuronové sítě (CNN, 1989)

Yann LeCun – průlom pro vizuální rozpoznávání

- ▶ Inspirace vizuální kůrou savců (Hubel Wiesel)
- ▶ Zavedení konvolučních filtrů, pooling vrstev a backpropagation
- ▶ Úspěch na úloze rozpoznávání číslic (dataset MNIST)
- ▶ První praktické „deep learning“ architektury



Druhá zima UI (90. léta)

Stagnace a pomalý pokrok v neuronových sítích

- ▶ Druhá vlna oslabení zájmu a financování výzkumu neuronových sítí.
- ▶ Způsobena především:
 - ▶ rychlým rozvojem teorie, kterou tehdejší hardware nedokázal prakticky využít,
 - ▶ přesunem zájmu k novým metodám, zejména **Support Vector Machines (SVM)**, **Hidden Markov Models (HMM)** a dalším statistickým algoritmům.
- ▶ Přesto některé výzkumné skupiny pokračovaly ve vývoji neuronových sítí.
- ▶ 1997 – vznik **Long Short-Term Memory (LSTM)**, nového typu rekurentní sítě od Hochreitera a Schmidhubera.
- ▶ 1998 – článek *Gradient-Based Learning Applied to Document Recognition* (LeCun et al.) – zásadní milník pro konvoluční sítě.

Deep Belief Networks (Hinton, 2006)

Znovuobjevení hlubokého učení

- ▶ Geoffrey Hinton v roce 2006 ukazuje, že hluboké neuronové sítě lze trénovat postupným **předtrénováním vrstev**.
- ▶ Každá vrstva se nejprve naučí **zachytit strukturu vstupních dat** bez učitele (generativní přístup).
- ▶ Teprve poté se celá síť společně dotrénuje pomocí **gradientního sestupu**.
- ▶ Tento postup odstranil dřívější problém mizejícího gradientu a umožnil stabilní trénink hlubších sítí.
- ▶ Výsledkem byl první výrazný skok v přesnosti po desetiletích stagnace — začátek moderní éry hlubokého učení.

Revoluce ImageNet

Revoluce v hlubokém učení

- ▶ 2012 – první významné umístění neuronové sítě v soutěži **ImageNet** (Krizhevsky, Sutskever a Hinton)
 - ▶ **AlexNet** využívá konvoluční vrstvy v úloze klasifikace obrázků do 1000 tříd
 - ▶ umístí uje se na druhém místě, ale s výrazně nižší chybovostí než ostatní metody
 - ▶ od té doby neuronové sítě v soutěži ImageNet dominují každý rok
- ▶ Chyba snížena o téměř 40 oproti tradičním metodám
- ▶ Od tohoto roku začíná dramatický nárůst zájmu o deep learning
- ▶ 2017 – Článek *Attention Is All You Need* – představení architektury **transformeru**
- ▶ 2020 – ImageNet přechází z klasifikace do 1000 tříd na přibližně **25 000 tříd**

Dvě základní úlohy neuronových sítí

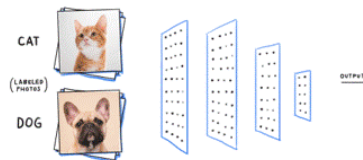
Přehled

- ▶ Neuronové sítě se v praxi zaměřují především na dvě hlavní úlohy:
 - ▶ **Klasifikace** – cílem je zařadit vstup do jedné z předem definovaných tříd.
 - ▶ **Regrese** – cílem je předpovědět spojitou (numerickou) hodnotu.
- ▶ Obě úlohy využívají stejné principy učení:
 - ▶ úprava vah na základě chyby (gradientní sestup),
 - ▶ trénovací dataset obsahující vstupy a požadované výstupy,
 - ▶ minimalizace ztrátové funkce.
- ▶ Liší se však typem výstupní vrstvy, volbou aktivační funkce a ztrátové funkce.
- ▶ Rozlišování těchto dvou úloh je základním krokem při návrhu architektury sítě.

Klasifikace

Co je klasifikace?

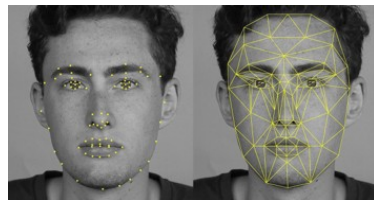
- ▶ Přiřazuje vstup do jedné z předem daných tříd.
- ▶ Využívá nejčastěji **softmax** (více tříd) nebo **sigmoid** (binární).
- ▶ Ztrátová funkce: **cross-entropy**.
- ▶ Využití: rozpoznávání obrázků, detekce spamu, klasifikace zvuku.



Regrese

Co je regrese?

- ▶ Předpovídá spojitou numerickou hodnotu.
- ▶ Výstupní aktivace zpravidla **lineární**.
- ▶ Ztrátová funkce: **Mean Squared Error (MSE)**.
- ▶ Příklady: predikce ceny, teploty, vzdálenosti, spotřeby.



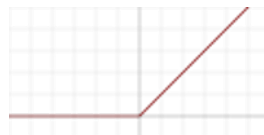
Současný model používaný pro výpočty

Moderní pojetí perceptronu

- ▶ Současné neuronové modely vycházejí z perceptronu, ale využívají různé **aktivační funkce**, které neuronu dodávají nelineární chování.

Aktivační funkce

- ▶ **Lineární** – používá se zejména v regresi.
- ▶ **Binární** – historický model, dnes převážně výukový.
- ▶ **Sigmoidální** – plynulá nelinearita, dříve standard.
- ▶ **ReLU** – současný standard, rychlé a stabilní učení.



Typy neuronových sítí – přehled

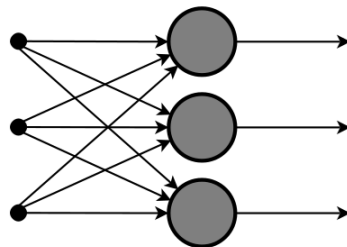
Hlavní architektury neuronových sítí

- ▶ **Dopředná neuronová síť (Feedforward Network)**
- ▶ **Vícevrstvý perceptron (MLP)**
- ▶ **Konvoluční neuronová síť (CNN)**
- ▶ **Rekurentní neuronová síť (RNN)**
- ▶ **Transformer**
- ▶ **Hluboké neuronové sítě (DNN)**
- ▶ **Generativní sítě (VAE, GAN)**

Dopředná neuronová síť

Charakteristika

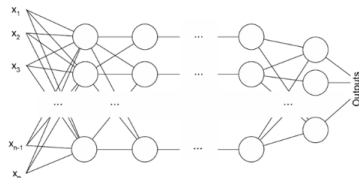
- ▶ Informace se šíří **směrem od vstupu k výstupu**.
- ▶ Jde o nejjednodušší a základní typ neuronové sítě.
- ▶ Typické použití: **klasifikační úlohy**.
- ▶ Relativně snadná na trénování.



Vícevrstvý perceptron (MLP)

Vlastnosti

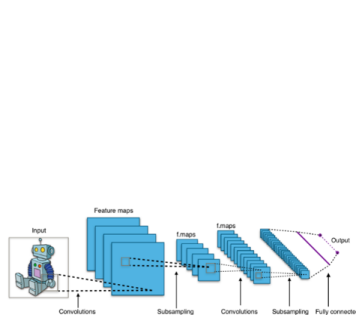
- ▶ Obsahuje tři a více vrstev (vstupní, skryté, výstupní).
- ▶ Umožňuje zpracovávat **nelineární klasifikaci**.
- ▶ Využívá nelineární aktivační funkce.
- ▶ Typické použití: rozpoznávání řeči, překlad textu.



Konvoluční neuronová síť (CNN)

Charakteristika

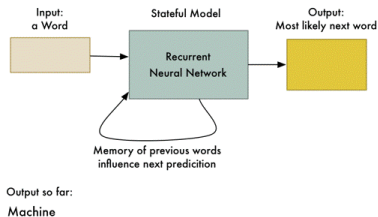
- ▶ Zpracovává vstup pomocí **konvoluce** s filtry (jádry).
- ▶ Nejčastěji používaná pro:
 - ▶ zpracování obrazů,
 - ▶ video,
 - ▶ některé NLP úlohy.
- ▶ Umí extrahovat hierarchické rysy (hrany → tvary → objekty).



Rekurentní neuronová síť (RNN)

Charakteristika

- ▶ Obsahuje **smyčky**, které umožňují uchovávat informaci mezi kroky.
- ▶ Vhodná pro sekvenční data:
 - ▶ text,
 - ▶ řeč,
 - ▶ časové řady.
- ▶ Umožňuje modelovat závislosti v čase.
- ▶ Nevýhodou je problém **mizejícího gradientu**, který komplikuje učení.



Transformery

Charakteristika

- ▶ Zavedeny v roce 2017 v článku *Attention Is All You Need*.
- ▶ Nepoužívají rekurenci – pracují pouze s **mechanismem pozornosti (self-attention)**.
- ▶ Umožňují paralelní zpracování celé sekvence najednou.
- ▶ Dominantní architektura pro:
 - ▶ NLP (BERT, GPT),
 - ▶ překlad,
 - ▶ shrnování textu,
 - ▶ generování textu.
- ▶ Vysoká škálovatelnost → základ současné generativní AI.



Hluboké neuronové sítě (DNN)

Charakteristika

- ▶ Hluboké neuronové sítě obsahují **větší počet skrytých vrstev** (obvykle 5 a více).
- ▶ Umožňují modelovat složité vztahy a struktury v datech díky postupnému učení reprezentací.
- ▶ Využívají různé typy vrstev:
 - ▶ plně propojené vrstvy,
 - ▶ konvoluční vrstvy,
 - ▶ rekurentní vrstvy,
 - ▶ normalizační a dropout vrstvy.
- ▶ Trénink hlubokých sítí byl dříve obtížný kvůli mizejícím gradientům – moderní architektury (ReLU, normalizace) tento problém výrazně omezují.
- ▶ DNN se uplatňují v mnoha oblastech: klasifikace obrazu, detekce objektů, zpracování řeči, generování textu, predikce časových řad.

Generativní modely

Co jsou generativní modely?

- ▶ Generativní modely se učí **pravděpodobnostní rozložení dat** a umí z něj generovat nové vzorky.
- ▶ Nejde tedy jen o klasifikaci – cílem je **vytvořit nová data** podobná těm trénovacím.
- ▶ Používají se v úlohách: generování obrázků, syntéza zvuku a řeči, generování textu, rekonstrukce a doplňování chybějících dat.

Typické architektury

- ▶ **VAE** – variational autoencoder, produkuje hladké latentní prostory.
- ▶ **GAN** – generativní adversariální síť, soutěž dvou sítí (generátor vs. diskriminátor).
- ▶ **Flow-based modely** – umožňují přesný výpočet pravděpodobností.
- ▶ **Diffusion modely** – nejúspěšnější pro generování obrázků (např. Stable Diffusion).



Jak se neuronová síť učí?

Princip učení

- ▶ Učení probíhá úpravou **vah** na základě chyby mezi skutečným a očekávaným výstupem.
- ▶ Síť prochází procesem:
 1. **Dopředný průchod** – výpočet výstupu s aktuálními váhami.
 2. **Vyčíslení ztráty** – kolik se síť „mýlí“.
 3. **Zpětný průchod** – výpočet gradientů (jak změnit váhy).
 4. **Aktualizace vah** – obvykle pomocí gradientního sestupu.
- ▶ Učení probíhá na mnoha příkladech, často ve formě minibatchí.
- ▶ Cílem je minimalizovat ztrátovou funkci, typicky MSE (regrese) nebo cross-entropy (klasifikace).

Backpropagation

Zpětné šíření chyby

- ▶ Jde o algoritmus, který umožňuje efektivně spočítat gradienty ztrátové funkce vůči všem vahám v síti.
- ▶ Funguje na principu **řetězového pravidla** z infinitesimálního počtu.
- ▶ Probíhá postupně od výstupní vrstvy zpět ke vstupní.
- ▶ Pro každou vrstvu se vypočítá, jak moc přispěla k celkové chybě.
- ▶ Tyto gradienty se poté použijí k úpravě vah směrem, který chybu nejvíce snižuje (gradientní sestup).
- ▶ Backpropagation byl klíčový průlom umožňující trénink vícevrstvých perceptronů.

Adversariální trénování

Co je adversariální trénování?

- ▶ Adversariální trénování je technika, kdy se síť učí odolávat **úmyslně pozměněným vstupům**, které mají síť zmást.
- ▶ Adversariální příklady jsou malé, často nepozorovatelné změny vstupu, které ale způsobí chybu modelu.
- ▶ Cílem je zvýšit robustnost modelu vůči útokům či nečekaným variacím dat.

Pro GAN však znamená něco jiného

- ▶ U **Generative Adversarial Networks** jde o „soutěž“ dvou sítí:
 - ▶ **Generátor** vytváří falešná data.
 - ▶ **Discriminátor** se je snaží odhalit.
- ▶ Obě sítě se zlepšují navzájem v tzv. **adversariálním procesu**.
- ▶ Výsledkem jsou velmi realistické generované vzorky (obrázky, zvuk, text).



Frameworky – jak se programují neuronové sítě

Frameworky pro neuronové sítě

- ▶ Neuronové sítě se dnes většinou neprogramují „ručně“, ale pomocí **specializovaných frameworků**:
 - ▶ TensorFlow, Keras
 - ▶ PyTorch
 - ▶ případně další (JAX, MXNet aj.)
- ▶ Frameworky zajišťují:
 - ▶ automatické počítání derivací (**automatic differentiation**),
 - ▶ efektivní výpočty na **GPU/TPU**,
 - ▶ připravené vrstvy (dense, konvoluční, RNN, attention, ...),
 - ▶ standardní optimalizační algoritmy (SGD, Adam, ...).

Frameworky – jak se programují neuronové sítě

Typický postup při programování sítě

- ▶ Definice architektury sítě (vrstvy, aktivační funkce).
- ▶ Volba ztrátové funkce a optimalizátoru.
- ▶ Příprava trénovacího a testovacího datasetu.
- ▶ Spuštění trénování (např. `model.fit(...)` nebo trénovací smyčka).

Neuronové sítě – interaktivní playgroundy

Vyzkoušejte si neuronové sítě online

- ▶ **TensorFlow Playground** <https://playground.tensorflow.org>
- ▶ **Teachable machine** <https://teachablemachine.withgoogle.com/>

Děkuji za pozornost ...
Dotazy?