**Analýza signálů II**

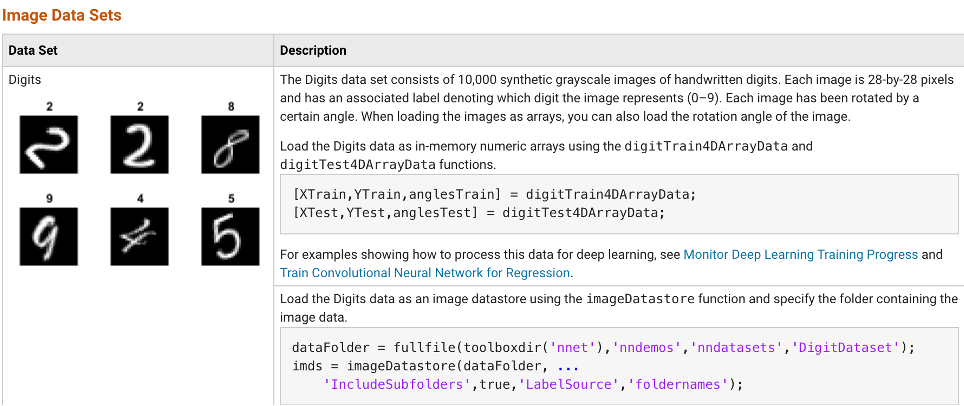
**Úvod do teorie neuronových sítí**

**Zadání:**

* 1. Načtěte volně dostupný dataset
  2. Vykreslete si pár obrázků z datasetu
  3. Podívejte se na počet daných obrázků v každé třídě
  4. Zkontrolujte velikost obrázků
  5. Rozdělte dataset na trénovací, validační a testovací
  6. Navrhněte architekturu konvoluční neuronové sítě (počet a typy vrstev)
  7. Specifikujte trénování konvoluční neuronové sítě (počet epoch, optimalizační algoritmus)
  8. Natrénujte a validujte neuronovou síť
  9. Vyzkoušejte měnit hodnoty parametrů neuronové sítě
  10. Otestujte neuronovou síť

**Data**

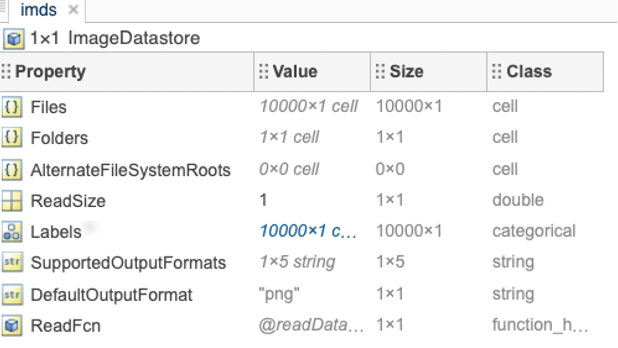
Jsou volně dostupné na internetu. Stáhnout si je můžete podle návodu na mathworks.com, kde je přímo dostupný kód. Data obsahují obrázky číslic od 0-9, které se budete snažit posléze klasifikovat.



**Realizace**

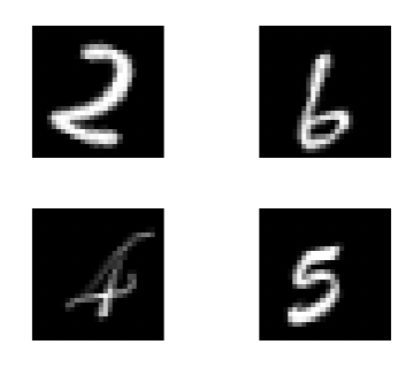
**1. Načtěte volně dostupný dataset**

Načíst dataset můžete podle návodu uvedený nahoře. Doporučuji použít funkci *imageDatastore*. Měli byste pak vidět následující strukturu:



**2. Vykreslete si pár obrázků z datasetu**

Vykreslete si libovolný počet obrázků z datasetu, abyste se s používaným datasetem seznámili a prohlédli si obrázky, které se budete učit klasifikovat do tříd. Měli byste vidět podobné obrázky:

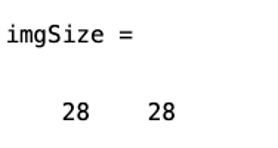


**3. Podívejte se na počet daných obrázků v každé třídě**

Zobrazte si, kolik obrázků v dané třídě je a kolik je tříd celkem. Budete tak mít přehled, jestli je dataset vybalancovaný, nebo jestli nějaká třída je v datasetu málo-hodně zastoupená. Tenhle parametr má vliv na učení neuronové sítě. **Do komentáře ke kódu (nebo do word dokumentu, který budete odevzdávat) napište, proč to má vliv na trénování.**

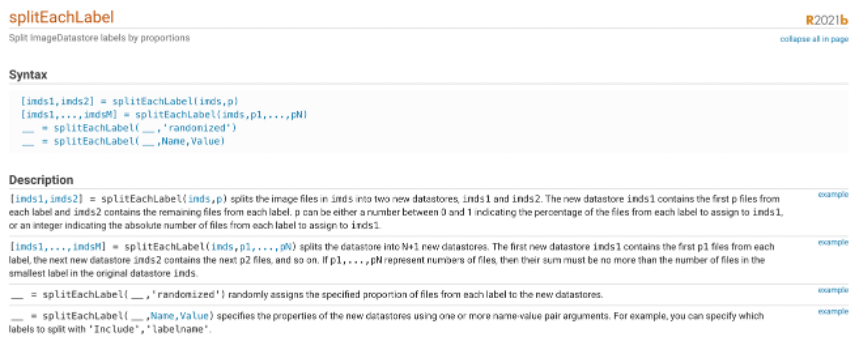
**4. Zkontrolujte velikost obrázků**

Podívejte se, jaké rozměry mají dané obrázky. Obrázky by měly mít stejné rozměry. Pokud bude potřeba rozměry změnit, změňte je pomocí funkce *imresize* na velikost 28x28.



**5. Rozdělte dataset na trénovací, validační a testovací**

Dataset musí být rozdělený na datasety testovací, validační a trénovací v různých poměrech. Dataset rozdělte a **do komentářů v kódu (nebo do word dokumentu, který budete odevzdávat) napište, proč jste vybrali zrovna daný poměr**. Použijte funkci *splitEachLabel*, popis funkce najdete na mathworks.com. Pokud si dataset chcete rozdělit jiným způsobem, nechám to na Vás.



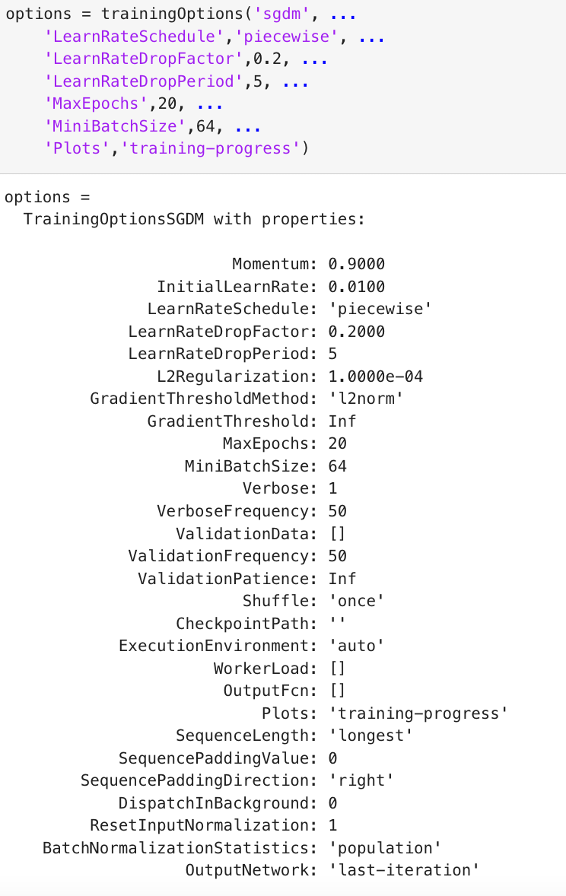
**6. Navrhněte architekturu konvoluční neuronové sítě (počet a typ vrstev)**

Konvoluční síť se skládá z různých bloků jako například konvoluční vrstva, ReLU vrstva, Max-pooling vrstva. Navrhněte si svoji vlastní konvoluční neuronovou síť. Síť nenavrhujte extra hlubokou, ať má maximálně 25 vrstev. **Do komentářů v kódu uveďte (nebo do word dokumentu, který budete odevzdávat), proč jste se rozhodli zrovna takhle neuronovou síť navrhnout** (proč zrovna první vrstva navazuje na druhou atd.). První vrstva v rámci neuronové sítě obsahuje vždy data a poslední vrstva je vždy klasifikační. Jak vytvořit vrstvy neuronové sítě v Matlabu můžete vidět na následujícím obrázku. Více příkladů jak vytvořit v Matlabu vrstvy najdete na odkazu:<https://www.mathworks.com/help/deeplearning/ref/nnet.cnn.layer.layer.html>



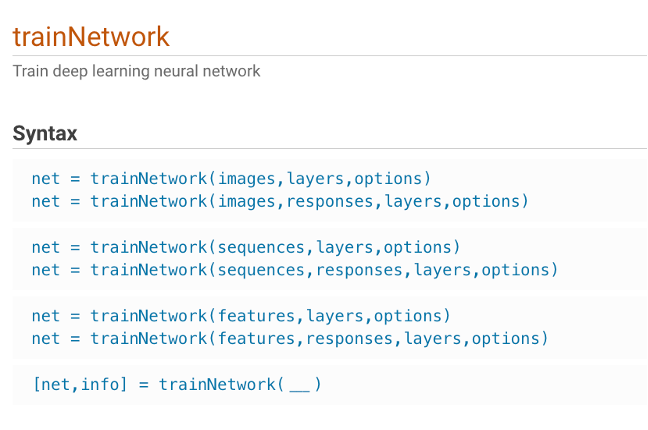
**7. Specifikujte trénování konvoluční neuronové sítě (počet epoch, optimalizační algoritmus,…)**

Každá neuronová síť má svoje určitě parametry, podle kterých se trénování neuronové sítě řídí. Rozhodněte se jaké všechny parametry použijete a **do komentářů v kódu (nebo do word dokumentu, který budete odevzdávat) napište, proč jste se tak rozhodli**. Použijte funkci *trainingOptions.* Posílám příklad neuronové sítě s různými parametry. Více příkladů najdete na následujícím odkazu:<https://www.mathworks.com/help/deeplearning/ref/trainingoptions.html#d123e135897>

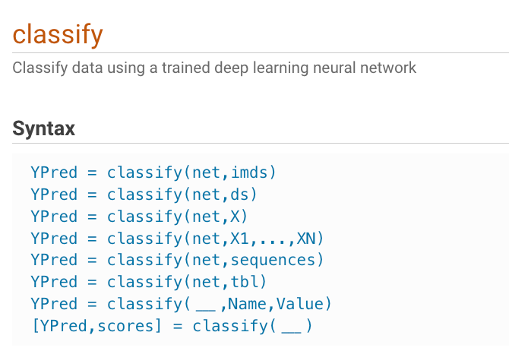


**8. Natrénujte a validujte neuronovou síť**

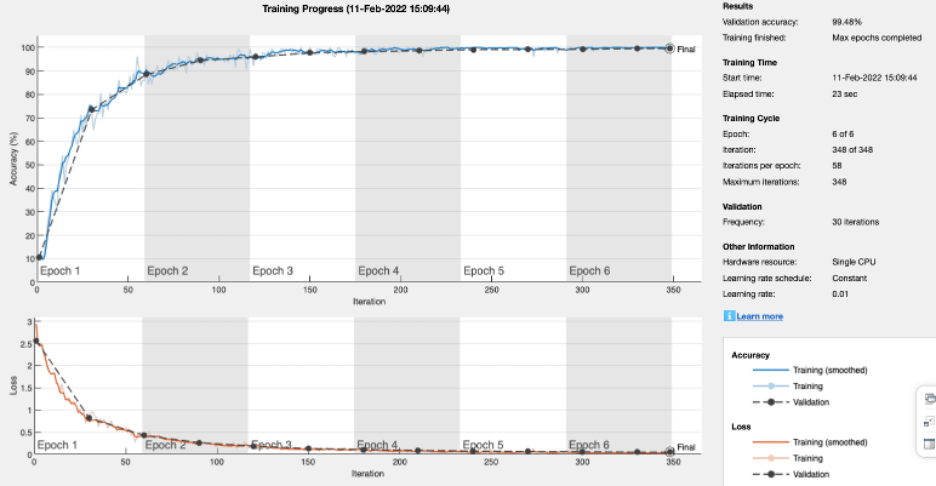
Teď Vás čeká trénovaní a validace Vaší neuronové sítě. Použijte k tomu funkci v Matlabu *trainNetwork*, kde zadáváte dataset, architekturu sítě a parametry sítě, které jste vytvořili.



Poté použijte funkci *classify* ke klasifikaci dat do tříd.



Funkce *trainNetwork a classify* Vám zobrazí trénovaní sítě v real-time. Měli byste získat podobný výstup:

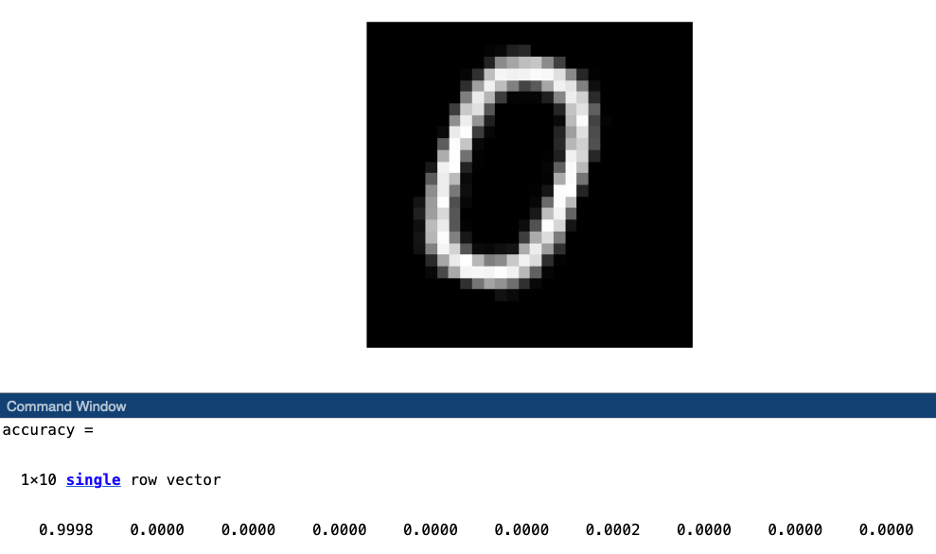
****

**9. Vyzkoušejte měnit hodnoty parametrů neuronové sítě**

Optimalizujte svoji neuronovou síť, aby měla co největší přesnost (validační přesnost najdete vpravo nahoře). Zkuste si pohrát s následujícími parametry: *MaxEpochs, InitialLearnRate, L2Regularization, MiniBatchSize*. U každého z vyjmenovaných parametrů, vyzkoušejte tři jiné hodnoty **a v kódu do komentáře popište (nebo do word dokumentu, který budete odevzdávat), proč s danými hodnotami je trénovaní neuronové sítě přesnější**. Dále si vyberte ještě *jeden jiný parameter* a udělejte to samé. **V souboru word k odevzdání pošlete screeshot jednoho vybraného parametru, jak se mění, když měníte hodnoty parametru.**

**10. Otestujte neuronovou síť**

Otestujte Vaši neuronovou síť na testovacím datasetu. Na následujícím obrázku vidíte, jak se neuronová síť poradí s číslicí O. Správně s přesnosti 0.9998 ji přiřadila k první třídě, což byla právě třída 0. **Výsledky testování jak náhodného obrázku, tak zprůměrované pošlete v dokumentu word s Vašim kódem.** K testování použijte fuknci *predict*.



**Odevzdávání**

Váš kód s dokumentem word, kde budou odpovědi na otázky a screenshoty, posílejte na moji emailovou adresu **kamila.lepkova@cvut.cz.** Do předmětu emailu napište Vaše jméno a zkratku předmětu. Celkem můžete získat 10 bodů.