Tutorial 1 Dane, overfitting i przekleństwo wymiarowości

1. W dostępnym dla kolegów miejscu umieścić dane, którymi będziemy się zajmować: MNIST, FMNIST, CIFAR 10, SmallNorb, TNG. Opisać w skrócie te dane i pokazać studentom jak wyglądają.
2. Opisać Pythonowe procedury normalizujące dane, a także procedury wybierające dane ze zbioru danych. Pokazać, jak działają na wybranym zbiorze z UCI dataset (nie może być mały, raczej M~1000, N~100).
3. Opisać algorytm k-NN oraz procedury Pythonowe służące do różnych jego realizacji. Opisać najbardziej efektywne obliczeniowo procedury do znajdowania k-najbliższych sąsiadów. Biblioteka FAISS dla obliczeń na GPU. Pokazać, jak działają na wybranym zbiorze z UCI dataset.
4. Opracować miary jakości klasyfikatora: cross-validation (procedura scikit-learn), leave-one-out. Pokazać, jak działają na wybranym zbiorze z UCI dataset.
5. Pokazać, jak zmieniają się miary jakości klasyfikatora, gdy zmniejsza się zbiór uczący a wielkość zbioru treningowego pozostaje taka sama. Czy można poprawić tą jakość zwiększając lub zmniejszając k, lub zmieniając kernel, lub ograniczając poszukiwania NN do zadanego promienia?
6. Pokazać jak można regularyzować system z małą ilością danych dokonując augumentacji (sztucznego powiększania zboru danych przez np. aproksymację. Znaleźć lub napisac procedurę). Jak zachowują się miary jakości w trakcie augumentacji.
7. Pokazać, jak działają procedury augumentacyjne z biblioteki imgaug.
8. ZADANIE: Wykonać punkty 6 i 7 na zbiorach MNIST, FMNIST przy pomocy imgaug oraz tworząc obrazki dodatkowe poprzez zaburzenie danych (np. dla x% pikseli losować liczbę 0,1 (MNIST), i odpowiednią – stopień szarości - dla zbioru FMNIST)