**Experiment handmatig hypertunen Convolutional Neural Networks**

Stephan Eikenhorst & Floris Verheijen

**Github Repository:** *Willem03…*

1. **Aanpak**

**Stap 1**

**Variëren met het aantal Convolutional Layers (1-3)**

* Trainen met 3 epochs; kiezen hoeveel lagen zinvol is
* Aantal units in de Linear Layers is gerelateerd aan het aantal Convolutional Layers
* Alle overige parameters constant houden we (o.a. *kernel size*: 3, *filters*: 32, *loss fie*: CrossEntropyLoss, *learning rate*: 0,001, *Optimizer*: Adam)
* Trainen met 100 epochs

**Stap 2**

**Variëren met het aantal Linear Layers (2-4)**

**Stap 3**

**Variëren met de learning rate & de optimizer**

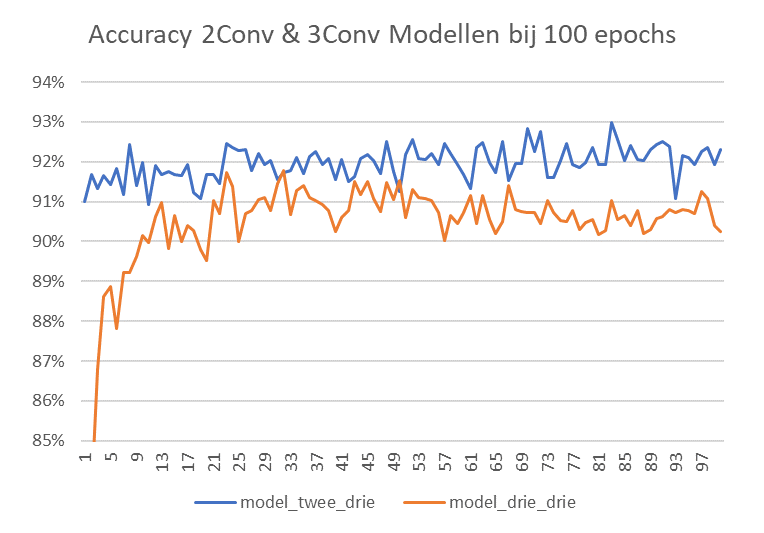
1. **Resultaten**

**Stap 1:**

* Een model met meer dan 3 Conv Layers en een kernel size van 3x3 is niet mogelijk, omdat de plaatjes dan te klein worden;
* Het model met 1 Conv Layer & 3 linear layers presteert okay (89%), maar 3 epochs trainen kost onevenredig veel meer tijd dan de modellen met 2 of 3 Conv Layers door het grote aantal features/parameters. Dat model hebben we niet verder uitgewerkt;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Aant. Epochs |  | Model | | |
|  | **Een\_drie** | **Twee\_drie** | **Drie\_drie** |
| 3 | Gem. Acc. | 89% | 88% | 83% |
| Tijd | .. | .. | .. |
| 100 | Gem. Acc. |  | 92% | 90% |
| Tijd | - | .. | 55 min |

* Het model met 2 Conv Layers (gem 92%) lijkt net wat beter te presteren dan het model met 3 Conv Layers (gem 90%);
* De accuracy stijgt nauwelijks gedurende 100 epochs trainen. Daarom in stap 2 met 50 epochs trainen.



**Stap 2:**