

10:15 – Opgave 3

1. Explain how the "pixel similarity" approach to classifying digits works.

- Opstil matrix, der repræsenterer billedet, hvert koordinat repræsenterer et pixel.
- Man kan nu sammenligne denne matrix med en idéel udgave af det, som man skal klassificere mod, ved at se på, hvor stor forskel der er på den idéelle matrix og billedmatrixen. Forskellen udregnes typisk med root mean square (L2 norm) eller mean of absolute differences(L1 norm).

2. What is a list comprehension? Create one now that selects odd numbers from a list and doubles them.

- `list = [f(x) for x in collection]`

```
In [2]: numberList = [1, 2, 3, 4, 7, 8, 9]
        evenNumbers = [number * 2 for number in numberList if number % 2 != 0]
        evenNumbers
```

```
Out[2]: [2, 6, 14, 18]
```

3. What is a "rank-3 tensor"?

- En tensors rank beskriver hvor mange unikke datatyper den tensor indeholder. Dette har ikke noget med størrelsen eller mængden af data at gøre, men hvad dataen beskriver.

4. What are RMSE and L1 norm?

- RMSE er hvor man beregner forskellen på to datasæt, det gøres ved at tage forskellen altså $(a_1 - b_1)^2$ og sætter den i anden, så tager man gennemsnittet af det altså af alle værdierne $(a_n - b_n)^2$ og så tager du kvadratroden af gennemsnittet for ikke at have forskellen repræsenteres som en kvadrat værdi.
- L1 Norm er hvor man tager den absolutte forskel fra to værdier, og tager gennemsnittet $|a_n - b_n|$. Tallet vil altid blive positivt, selvom forskellen kan være -150 så vil selve forskellen være 150. Hvis begge tal er negative, ganger man de to tal og tager den positive kvadratrod.

5. Create a 3×3 tensor or array containing the numbers from 1 to 9. Double it. Select the bottom-right four numbers.

```
[1] !pip install -Uqq fastbook
import fastbook as fb
fb.setup_book()
```

```
| 727kB 5.4MB/s
| 194kB 8.6MB/s
| 51kB 5.5MB/s
| 1.2MB 11.8MB/s
| 61kB 7.2MB/s
Mounted at /content/gdrive
```

```
[2] import torch as torch
from fastai.vision.all import *
from fastbook import *
```

```
[3] data = [[1,2,3],[4,5,6],[7,8,9]]
```

```
[5] tensordata = tensor(data)
```

```
[6] tensordata = tensordata *2
```

```
▶ tensordata[1:, 1:3]
tensor([[10, 12],
        [16, 18]])
```

6. What is broadcasting?

Hvis to tensor er af forskellige ranks, kan vi bruge broadcasting til at expande den mindre ranket tensor til samme rank, som den større tensor. Det er **vigtigt** at forstå at PyTorch ikke faktisk kopier billederne til at opnå ranken, men *lader* som om at tensoren er af den størrelse. Derved undgås der at allokeres hukommelse til det. Dette muliggør, at PyTorch kan bruge matematiske funktioner på de 2 tensors / array f.eks. at gange dem eller tage gennemsnittet.

7. Are metrics generally calculated using the training set, or the validation set? Why?

- Validation, fordi man er nødt til at vide hvor god den er til at predikte på ikke set data.

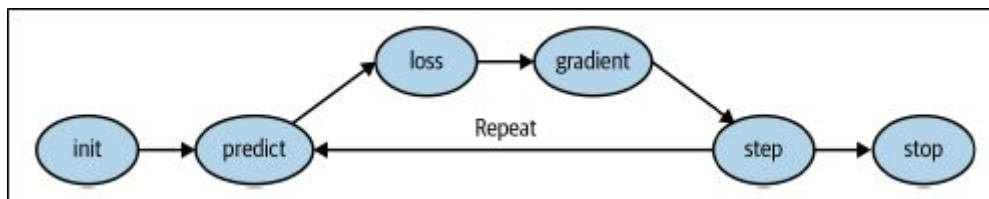
8. What is SGD?

Stochastic Gradient Descent

Det er en iterativ måde, at optimere vores model via gradients. Afhængig af vores Learning Rate, tager vi små skridt mod det bedste fit.

9. What are the seven steps in SGD for machine learning?

Initialize the parameters
Calculate the predictions
Calculate the loss
Calculate the gradients
Step the weights
Repeat the Process
Stop



10. How do we initialize the weights in a model?

- Random eller fra en pretrained model

11. What is "loss"?

- Mængden af fejl, numeriske værdier for hvor god ens model er

12. Why can't we always use a high learning rate?

- Fordi man vil overshoot ens minimale loss eller nogen gange blive være, da den tager for store skridt.

13. What is a "gradient"?

- Differentialet (Hældningen), på et specifikt punkt

14. Why can't we use accuracy as a loss function?

- Fordi accuracy er ikke godt nok til en model. Loss er en bedre repræsentation matematisk for fejl.
-

15. What is the difference between a loss function and a metric?

- Loss function bruges til at justere vægte mens metrics er det vi selv måler på for at se kvaliteten af vores model.

16. What is the function to calculate new weights using a learning rate?

- Backpropagation

17. What does the backward method do?

- metoden går baglæns, starter i slutningen, og tilpasser med ændringerne

- udregner gradienten for hver vægt i hvert lag