Exercise B

We define G as the number of neurons in the second hidden layer. Hence we have:

Exercise C

1. E

Exercise E

Is the derivation good enough?

In order to show that with the two above assumptions, we can derive a loss function that containsas a term, we begin by deriving likelihood function. Given assumption (1), we know that the likelihood of the data is the product of each data points individual likelihood, i.e.

From assumption (2) we have that:

And using hint (1) we can expand the targets as follows:

Hence we have the likelihood .

The above Is simply the likelihood function right?

We now use hint (2) and take the log of the likelihood function as follows:

Using the product rule “log(ab)=log(a)+log(b)” we have:

Using the product rule again we have:

By using logarithm rules for fractions, the first term can be manipulated as follows:

By using logarithm rules for exponential, the second term can be manipulated as follows:

Hence we now have the following log-likelihood function:

We can simplify this function by taking the sums of each individual term:

Replacing the sum of squared errors term with

Using hint (2) which states that multiply the likelihood objective with -1 to turn maximum likelihood into a loss/error function:

Hence the loss is

The reason why applying the log and multiplying by minus one is the right thing to do in order to get a loss function is first because taking the log simplifies calculations as the products are turned into sums. Second taking the log doesn’t change the optimum as the logarithm is a monotonically increasing function. The reason for multiplying with -1 is because maximizing the likelihood is equivalent to minimizing the negative log-likelihood.

Is this good enough ?

Exercise G

We have been given the following likelihood function for each individual datapoint:

We now take the logarithm of the likelihood function:

We know that the likelihood of the data is the product of each data points individual likelihood, i.e.

Now applying the product logarithm rule:

Applying the power logarithm rule:

We can then negate the log-likelihood because we want to minimize the loss rather than maximize the likelihood. Hence the cross-entropy loss is:

Is this good enough needs more text or fine?

DL\_AutoDiff

a)

I b er dette den funktion som skal skrives eller er det resultatet af den:   
f=(a×b)+(a×d)

b)

1.

f=(a×b)+(a×d)

2

c = a \* b

c = a \* b

e = a \* d

e = a \* d

f = c + e

3