Exercises INF 5860

Exercise 1 Linear regression

a) What is the loss function for linear regression?

b) Why would we want an iterative algorithm for the linear regression problem?

c) How does gradient descent update the estimate, give the general formulae?

d) Given x=
   Plot x,y as points in a plot.

e) If we start with Θ_0=0 and Θ_1=0, compute the value of the initial loss function

f) If we start with Θ_0=0 and Θ_1=0, compute the estimate after one iteration if the learning rate is 1.

Exercise 2 – Logistic classification

a) Given a trained logistic classifier for a single feature and 2 classes. What is the equation for the decision boundary if W=2 and b=1?

b) With logistic classification, how is a new sample classified?

c) How can we generalize logistic classification to more than 2 classes?

Exercise 3: Basic neural networks

a) What are the problems with using a sigmoid activation function?

b) In what way does the tanh activation function share the same drawbacks?

c) Discuss briefly why feature scaling of the input features is important.

d) Why is initializing all the weights to zero problematic?

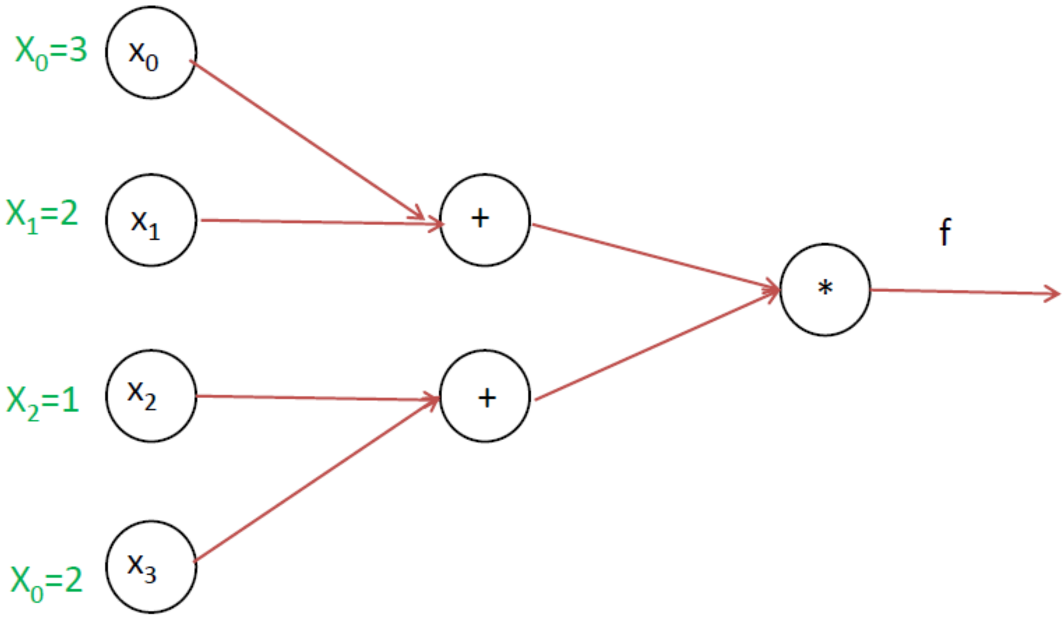
e) Assume that we have a 2-layer net (one hidden layer) with weights $W^{(1)}$, $b^{(1)}$, $W^{(2)}$ and $b^{(2)}$. Assume that we use RELU-activations in the hidden layer, and no activation on the output layer. Write down an equation for the output of the j'th node in the hidden layer, $a^{(j)}$.

f) Explain shortly how maxnorm regularization works.

g) When using dropout, if you do not consider any scaling during training, how should you then compensate during prediction of new data/test samples?

h) Explain briefly how momentum gradient descent works, and why this can be more robust than regular gradient descent.

Exercise 3: A simple network



a) Perform backpropagation on this graph

Exercise 4: Generalization

1. Why can testing out multiple models on your test data be a problem and when is it problematic?

2. How does searching through more hypotheses affect the probability of searching through a solution close to the correct solution?

3. Give an example of a way to measure model complexity.

4. What are the implications of the “No free lunch” theorem mean for machine learning?

5. Give three examples of common assumptions (priors) machine learning models make.

Exercise 5: Representations

1. What do we mean when we refer to the image manifold?
2. Explain why working with image gradients can be better than working with raw pixels. What additional effect can you achieve by scaling the gradients based on the gradient magnitude?
3. How can multiple layers of discriminators/classifiers reduce the need for training examples in image analysis?

Exercise 6: Convolutional nets

1. You have a 32x32x5 image and filter it with a 5x5x5 kernel, the way most convolutional neural networks are implemented. If you use no padding, what will be the output size of the activation map?

2. What do we mean by dilated convolutions and how are they used?

3. Why is the effective field-of-view usually smaller than the theoretical field-of-view? *By theoretical field-of-view we mean the size of the image patch that can influence each of the output values in the activation map. Practical field-of-view is the size of the patch of pixels influencing the results of a given output value.*

4. In deep learning frameworks, you usually operate on 4D tensors, when working with 2D convolutions. If you want to use such a framework to do a average (blur) filtering of images, how would you have to construct the kernel for the convolution? *You should treat each of the color channels (RGB) independently.*

Exercise 7: Training deep networks

1. Gradient flow
    a. Why is *gradient flow* important when training deep neural networks?
    b. Give some common methods that help to ensure good gradient flow.

2. How does batch size relate to learning rate? Explain.

3. Why is it a problem to optimize accuracy directly with a deep neural network?

Exercise 8: Deep learning architectures

1. Give two possible explanations to why residual networks work better than standard feed forward networks.

2. You want to find bounding-boxes for cars in an image. You don't know how many cars there will be in each image, but you can safely assume it's between 0 – 100. Describe how you can construct and train a deep neural network for this task.

3. What does it mean to use a "Fully-convolutional" architecture for image segmentation?

4. What is the reasoning behind the concatenation operations in U-Net for image segmentation?

Exercise 9: Visualization and Adversarial training

1. You have a convolutional neural network trained for image classification. Describe a simple way of detecting what parts of an image are responsible for a certain classification result, without using the image gradients.

2. How can you get a simple estimate of how changing a set of pixel-values will affect the final class probabilities?

3. For some visualization techniques, you apply a lowpass (blurring) filter between each iteration of optimization. Why may this be a reasonable approach?

4. You have lots of training images for one application, but no labelled images for a similar application. How can you use Adversarial domain adaption, to improve your results on the new data?

Exercise 10: Recurrent Neural networks (RNN)

1. Why is vanishing gradients and outputs a more common problem in basic RNNs compared to feed forward networks?

2. Why is vanishing gradients and outputs in RNN less problematic than for feed forward neural networks?

3. Why can you only do gradient descent for a certain number iterations of an RNN and when is this a problem? Explain and provide an example.

4. Give an overview of some common solutions to using deep learning for video data.

Exercise 11: Reinforcement learning

1. Is Reinforcement learning usually training faster or slower than standard supervised learning?

2. In what kind of situations is it common to use Reinforcement learning? Explain why.

3. In what type of situation does Policy learning require a lot of memory?

4. How could you implement hard attention for image analysis in a fully supervised way, without using reinforcement learning?

Exercise 12: Unsupervised learning

1. Draw and explain an example where t-SNE work better than PCA.

2. When you do a PCA of a dataset, you can easily transform new points with the same transform. Why is it more difficult to transform new points with t-SNE?

3. Give basic overview of what an autencoder based on neural networks is.

4. Explain a typical situation where first learning an embedding unsupervised and then using the embedding for supervised learning, can fail.