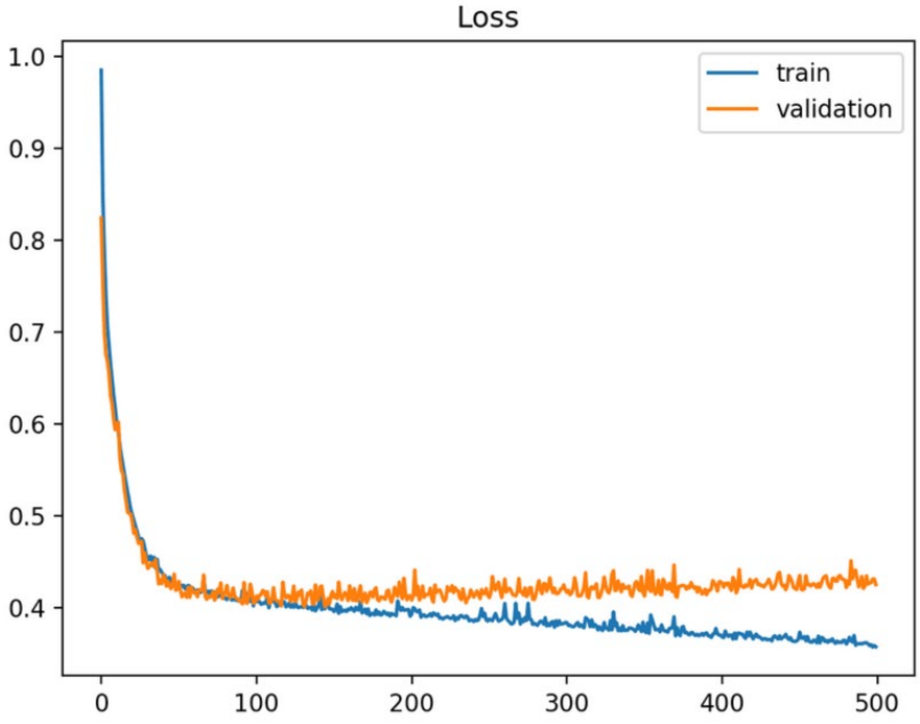
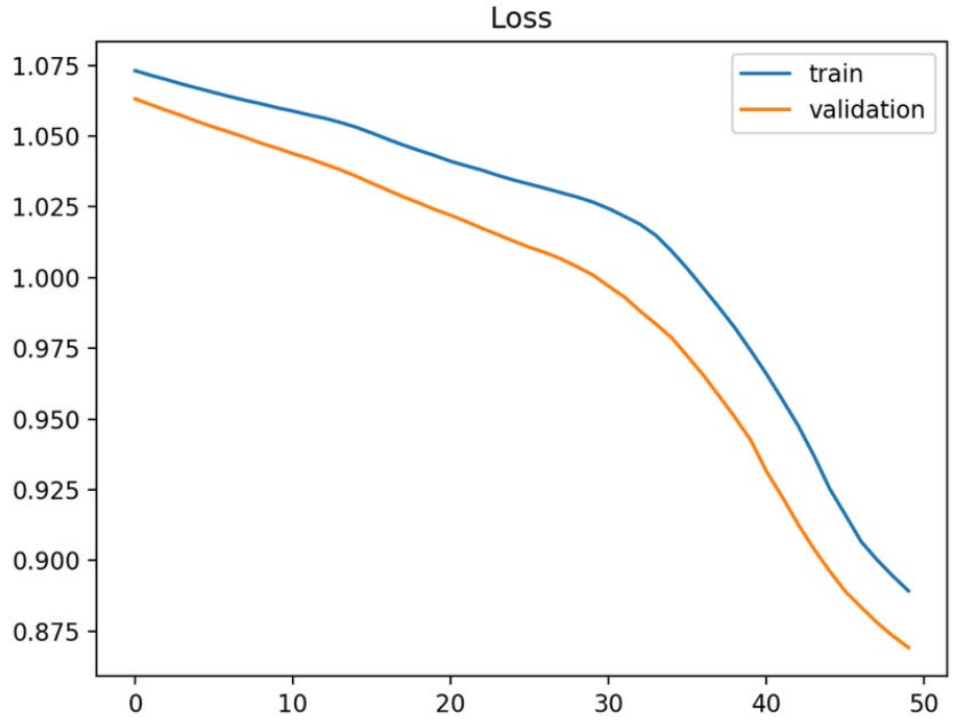
Lecture 11: *Summary*

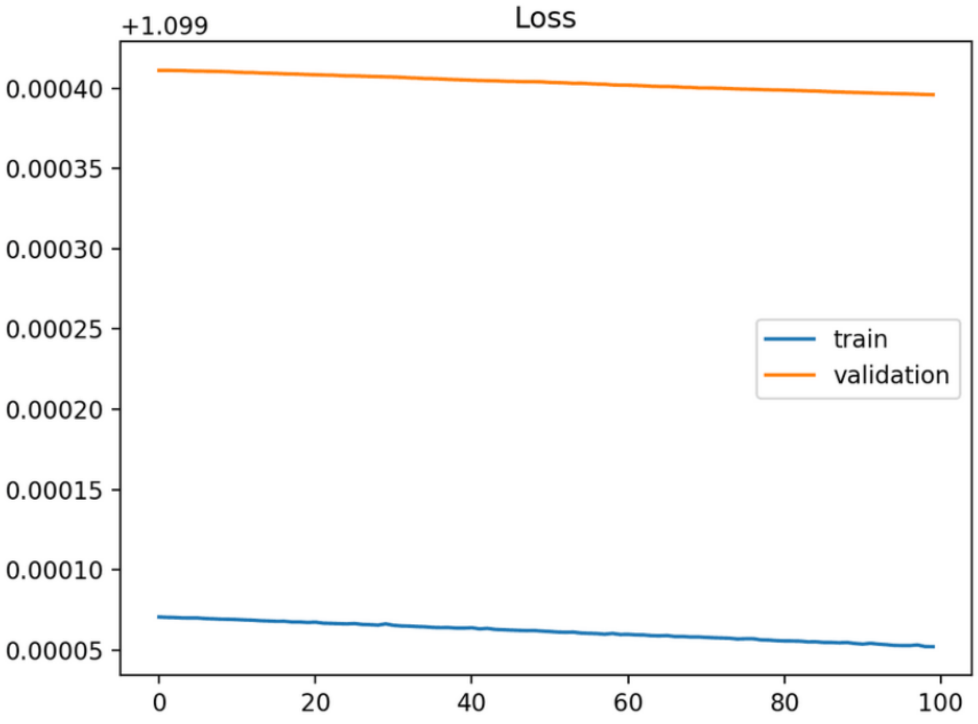
We now recap the topic we talked about in the whole semester.

Questions

1. List the key steps to be taken in a Machine Learning project.
2. Draw a picture structuring the different main components of machine learning and the algorithms and performance metrics for them.
3. Which performance measures exist in classification? For each measure, give an example of when it makes sense to optimize the model on it.
4. What can be done when a classification model is not accurate enough?
5. What is overfitting? How can we measure overfitting?
6. When do we need unsupervised learning?
7. How does the kmeans algorithm work?
8. How can we apply semi-supervised learning?
9. Describe the architecture of a feed foward network for regression.
10. Describe the basic procedure for the backpropagation algorithm.
11. Analyse the following learning curves. Which situation do they describe?



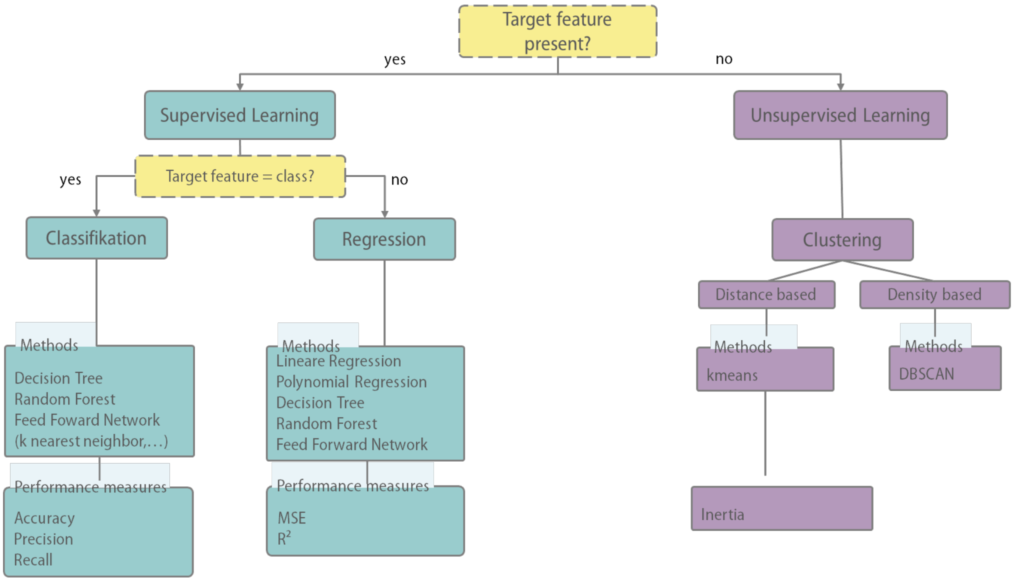




All figures from: https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/

Answers

1. Define the task, Get the data, Visualize and prepare the data, Select and train a model, fine-tune the model, deploy it (you can iterate back if the model is not good)
2.



3. The measures are accuracy, precision and recall. There exist also the f1-score. Accuracy is used, when we want a model that predictions for all classes are as good as possible. The classes should be balanced.
   If we optimize to the precision the false positives are minimized. For example if we want to find a filter for safe videos for children, then we want to allow only the safe ones and would take into account a high rejection of false ones (high recall).
   The recall minimizes the false negatives. If we would optimize the video filter to this, then only videos where the model is sure that they are not for children will be rejected, but many doubts will be allowed.
4. Hyperparameter tuning, try another algorithm, then: increase the data set, integrate or create new meaningful features
5. Overfitting means that the models is too much adapted to the data, but do not learn the general patterns. We can measures this by dividing the data into training and test data. We train the model just with the training data, so an independent test of generalization with the test data is possible.
6. We do unsupervised learning when the data is not labeled.
7. In semi-supervised learning first all data are clustered (with a big k). If the labels in one cluster are the same the rest of unlabeled data in this cluster gets this label.

If we can label samples after clustering it is a good idea to cluster the data with the number of classes and then label all the cluster centres, they are good representatives.

8. We can cluster the data and then analyse if in one cluster are only the samples of one class. Then we assign all the samples of that cluster to that class.
9. A feed foward network has

- one input layer with number of neurons = number of input features, no activation function
- several hidden layers, these are dense layers, the choice of the number of neurons is free, activation function = ReLU (or SELU etc.)
- one output layer and for binary classification or regression one output layer, for multiclass classification as many neurons as classes and activation function: sigmoid for binary classification, softmax for multiclass and none for regression (if bounded tanh, if only positive ReLU)

We have to define the loss function, this is the mse for regression, also a learning rate and the number of epochs.

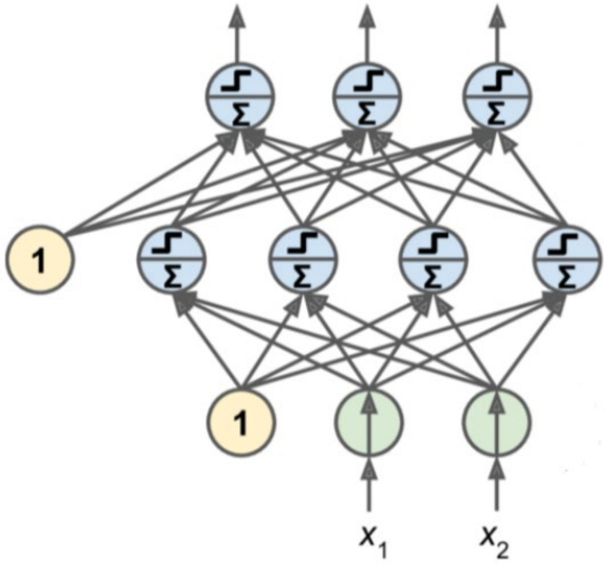
10. In the training one sample is given into the neural network. With the actual weigths an output is computed, the values in each neuron are stored. For the output the error to the labeled values is computed. This error is divided back to all the weights which caused it. Layer by layer the error is distributed to each neuron and his weights. The weights are corrected according to the amount of error they contributed. It is backpropagated back to the first layer.
11.
    - Overfitting, 2. Underfitting, more training could help, 3. Underfitting, model too simple

Questions during the meeting

1. Fill out the following table:

Problem Type	Input layer		Hidden layer		Output layer		
	Number of neurons	**Activation function**	**Number of neurons**	**Activation function**	**Number of neurons**	**Activation function**	**Loss function**
Binary Classification							
Multi-class Classification							
Regression with unbounded output							
Regression with bounded output							

1. Describe the task of an optimizer and of his learning rate.
2. What is an epoch?
3. What properties should a good initialization of the weights of a neural network have?
4. Explain the following figure:



Taken from: Geron: Hands-on Machine Learning, 2nd edition, p. 289

Answers to the questions during the meeting

1.

Problem Type	Input layer		Hidden layer		Output layer		
	Number of neurons	Activation function	Number of neurons	Activation function	Number of neurons	Activation function	Loss function
Binary Classification	as many as input features	no	free	ReLU	1	sigmoid	binary cross entropy
Multiclass Classification	as many as input features	no	free	ReLU	as many as classes	softmax	categorical cross entropy
Regression with unbounded output	as many as input features	no	free	ReLU	1	none	mse or mae
Regression with bounded output	as many as input features	no	free	ReLU	1	sigmoid (or ReLU for positve values)	mse or mae

1. The optimizer computes the error and in which direction the smallest error is. The learning rate gives the "stepsize" of learning.
2. epoch = iteration over the whole training data, the size and the number of iterations is always the same.
3. The weights should be initialized randomly. If we have the same values in one layer they would be corrected the same every time. (p. 333 The variance of the output of each layer should be the same as the variance of the inputs.)
4. It is a feed foward network with one hidden layer. It has two inputs x1 and x2 and three outputs. The second layer has 4 neurons, the last three. The yellow 1s are the bias term (in general it is not drawn). The arrows are the weigths going from one layer to the next, from each neuron of the one layer to each neuron of the next layer, so the network is fully connected. The sum symbol means that all weights are summed up and the next symbol stands for the activation function which is applied to th sum of weights in each neuron.