Seminar 6 questions

1. **A Boltzmann neuron uses simulated annealing, explain what that means (Verkligen dåligt förklarat, fattar hälften av svaret)**Simulated annealing means that it is chance based and does changes that could be increasing the error in the beginning, but as a function of time the are decreases, the system “hardens”. This is often used to avoid getting stuck in a non-optimal solution which could happen if we only move to “better weights”.
2. **Explain how training is done in feedforward networks.**Training is done with an algorithm, back propagation, which works by inputting data to which you know the solution and then comparing the output to the desired output. This error is then used to change weights in the previous layer and this process is repeated for each layer. By modifying the weights we change when which neuron will fire and therefore achieve a better result. The weights change only by a small amount as if we would change the weight to get 0 error on one example we would overtrain and the network wouldn’t be able to generalize. This is why training is done.
3. **What is special about recurrent neural networks?**Recurrent neural networks are NN’s where there are feedback paths back into the network. This allows them to use their internal state as a sort of memory to process a sequence of inputs. This is very useful when trying to for example recognize human speech.
4. **Name some activation functions**Logistic Sigmoid, Hyperbolic Tangent, Heaviside Step, ReLU..
5. **Describe a feedforward neural networks internal structure.**The connections never form a cycle, every neuron has an input and an output and data flows from the input to the output (feedforward).
6. **What is the commonly used model of a neuron and how does it work?**The commonly used type of neuron is called a McCulloch-Pitts neuron. This model has a neuron which has inputs and outputs, and each of the inputs has a weight. The neuron takes the sum of the inputs\*weights, add to it the bias of the neuron and this is the activation level upon which the neuron will in turn turn on its own outputs.
7. **Name two algorithms used to train neural networks and what they require**Back-propagation, needs to know how much can be changed in one step and resistance to change.   
   Genetic algorithms, need population size, mutation rate, a number of the maximum generations and a selection probability which decides which instances survive to the next generation.
8. **What is overfitting?**Overfitting is is a term used when a model models the training data too well. This means that the model picks up random fluctuations in the training data and accepts is as a part of the model. Overfitting is a reason many machine learning algorithms also include parameters and techniques that constrain the detail the model is using.
9. **What is underfitting?**When a model can not model the training data or generalize new data, the model is underfitted. An underfitted model is not a suitable model and have poor performance on the training data.
10. **Explain what components exist in an Artificial Neural Network and how they interact.**The ANN is usually structured with an input layer, some number of hidden layers followed by an output layer. Each layer contains one or more neurons that in most cases has one input from each of the previous neurons and an output to all the neurons in the next layer. Each connection between two neurons has a weight. Each neuron can have a bias and an activation function. The output of a neuron is then the dot product “inputs \* weights” added with the bias, applied in the activation function. This output is then part of the input to the next layer. These weights and biases are changed in the training process by an algorithm (usually backpropagation).
11. **What are the main components of a neural network and explain what they do. (Possible duplicate of above)***Artificial neurons(perceptrons):*  A perceptron is a mathematical function that simulates how biological neurons work. A artificial neuron receives one or more inputs and sums them to produce an output.

*Connections, weights and biases:* Connections are used for transferring the perceptrons outputs to other perceptrons as input. Each connection is assigned its own weight which decides how “good” it is for the network to use that particular path. Bias is a type of offset that adjusts your range of your activation functions.   
*Activation function:* The activation function computes the input from a perceptron and gives an output to another perceptron as input.   
*Learning rule:* The modification of the parameters of a neural network is done according to the learning rule. It is an algorithm or “rule” that decides for a given input what the desired output is.

1. **Where in the Artificial Neural Network is the knowledge stored? i.e. what it has learned**It is stored in the weights and biases. There is a weight between each connected neuron, and a bias connected to each neuron which is independent from previous layers. The weights and biases are changed to fit the correct result better during training.
2. **Explain how Backpropagation is used to train Artificial Neural Networks.**It essentially takes the ANN output, compares it to the desired value and changes the weights and biases so that the output better reflects the desired values. It does this by taking the gradient of the cost/performance function i.e. the difference between desired value and actual value (often to the power of 2, and divided by 2, for mathematical convenience). The gradient then tells us which direction will increase the performance (of that particular data result), then we take a step in that direction (of magnitude proportional to the gradient). This way we do not jump too far, missing our local maximum or take forever to reach it. This is much like climbing a hill (where you would climb faster the steeper it is). Usually, we do not backpropagate for each data point but rather in batches, calculating the mean squared error and applying gradient descent.
3. **What does it mean to dropout neurons? Why do we do it?**To dropout neurons means that we drop a random set of neurons from the hidden layers of the neural network during training. We do this every training example/batch to effectively reduce the dependence between neurons and force them to work more independently. The purpose of dropouts is to reduce the problem of overfitting.
4. **Explain the essentials of Genetic Algorithms.**Genetic algorithms are inspired by biology genes. It uses the concepts of selection, cross-over and mutation, to find optimal solutions. We start by generating a random population of “chromosomes”. Evaluate if a good enough solution is found. (Probably not since it is so far randomized). Determine “fitness” of chromosomes. Apply crossovers to combine two “parents” into a new chromosome “child”. Apply mutations to existing chromosomes to get new chromosomes. Select subset of population to survive. Then go back to evaluation step and repeat whole procedure from there if solution is not found.
5. **What are they good for? (neural networks)**Neural networks are good at tasks involving pattern recognition. These include character recognition, machine translation, social network filtering and medical diagnosis.
6. **Compare different activation functions and their impact on the firing rates of neurons**Relu leads to a sparse firing, since half the time the neuron will output 0. Sigmoid and tanh functions will always fire. This means that in a dense network most neurons will produce a signal, increasing the amount of floating point math that must be performed. Leaky Relu leads to dense firing but often has simpler math which can improve speed.
7. **Explain what the dying ReLu problem is and explore possible solutions**Suppose a neuron using ReLu receives feedback that shifts its bias towards the negative. There is a real chance no further input will produce an output with a gradient. This means back propagation will fail to alter its weights any further, ensuring the neuron never fires and never updates. This is called a ‘dead’ neuron. One solution, called leaky ReLu is to alter the function to max(0.00x,Z) in order to allow a slight gradient for adjustment. The main idea is to ensure the gradient is not zero. We can also occasionally search for and re randomize the weights of dead neurons.
8. **Compare the features of Sigmoid, tanh and Relu**Sigmoid and softmax works well for a classifier since different features will activate neurons with different strengths. This means the final output layer will have a varying magnitude based on the likelihood of an input belonging to a certain category. Sigmoid functions lock their outputs within [0,1] and have gradients that tend to push outputs towards either extreme, meaning they are good for classification. However, the gradient shrinks at extreme inputs, meaning learning can be slow until a value closer to the middle is reached. This is called ‘saturation’. Sigmoid is also non-zero centered. This can produce zig zagging learning behaviour. Tanh has similar problems. Its gradient is stronger but it still suffers from the vanishing gradient problem. ReLu is non-linear and any function can be approximated with combinations of ReLu. Its values are unbound between [0,inf] so it can blow up the activation. ReLu in a dense network cause less activation the sigmoid and tanh, improving speed. However, ReLu can produce the dying neuron problem due to its lack of a gradient for half of the possible inputs.

Relu works often as a general approximator.

1. **Why should we not use the linear function as an activation function?**The derivative of A = cx is always a constant c. This means that the gradient has no relationship with the value of x. A constant gradient will result in back propagation changing the weight on a neuron’s connections by a constant amount each time rather then refining the weights based on the magnitude of the error. In addition, each layer and each neuron will use a linear function of its inputs. Thus, no matter how many layers there are, we can never build a nonlinear function. Thus our final output is just a linear combination of other layers, and we can merge every layer into a single layer without changing the result.
2. **In gaming, behaviours of NPCs is often controlled through decision trees. When might we consider using a neural network instead?**A finite decision tree produces set responses in response to set situations. When an unforeseen situation arises a decision tree may produce behaviour that appears bizarre to the player. A neural network might not produce as refined behaviour as a decision tree but it can often handle grey areas more reasonably. This means that in games where situations are not easily evaluated for input to a decision tree, a neural network may produce better results. In addition, it can be difficult to provide utility evaluations that compare apples to oranges. For example, how do we compare the value of increased long range DPS on our team’s side vs a deeper reserve of hit points on the other team’s side? With a trained neural network, appropriate functions for comparing these two things emerge naturally from the training.
3. **What are Artificial Neural Networks?**Artificial neural network (ANN) is algorithms that find relationships in a dataset similar to how humans think. ANN are based on the same principle as a human brain, as people learn through their surroundings and then save the information. In order for an artificial neural network to find relationships, it must first be trained. They learn by training them with examples where the correct result is known.
4. **What is ANN limited by?**ANN are limited by the hardware it uses and the time takes to train it. If the ANN needs to compute more complex tasks, it requires a considerable amount of compute power and the longer time to produce good results.
5. **Explain the three common layers in artificial neural network.**A typical ANN consists of differents layers called input layer, hidden layer (or layers) and output layer.   
   ● Input Layer: Receives data from the outside world and sends it to first hidden layer. ● Hidden Layers: Computes the data with algorithms sends the output data to the output layer if the computation is finished. The more computation is needed, the output data is sent to the next hidden layer.   
   ● Output Layer: Produces output data received from the last hidden layer to the outside world.
6. **In which application can ANN be used?**They can be trained to interpret handwritten text or to recognize faces. ANN is also good at other areas, for example stock market prediction and image compression.
7. **When was the first computational model for neural networks created?**The first computational model was proposed by McCulloch and Pitts in 1943.
8. **What is NEAT?**NEAT, or Neuro-Evolution of Augmenting Topologies, is a method for evolving artificial neural networks with a genetic algorithm, introduced by Kenneth O'Stanley and Risto Miikkulainen. The method is to start with a simple neural network and then allow it to become more complex with each new generation.
9. **What is the difference between single layer and multilayer artificial neural networks?**A Multilayer ANN contains one or more hidden layers between the input and output layer while singler layer does not contain any hidden layer. Single layer only contains an input and output layer.
10. **What is a threshold/activation level in an artificial neuron?**It is the number of inputs required for the neuron to create an output at all. Then the output could possibly be different based on the number of inputs it has.
11. **What are two ways of preventing overfitting in neural networks?**Overfitting is when the model adapts to data that does not necessarily represent what we are looking for, leading to the system not really working as intended. This could be prevented by pruning for example, where we remove data that strays too far from the other data points. Another method is to insert noise into the data, this makes the data more even as all data is not that exact anymore.
12. **What is a drawback to genetic algorithms compared to backpropagation?**Genetic algorithms will just try different values for all the nodes until we get a satisfactory result, this is a brute-force kind of way of going about it and it is not as efficient as it takes a lot of time. Whereas backpropagation will go back adjusting the weights of neurons that gave a wrong or correct answer to make the system better.
13. **Describe how a Feed forward neural network differs from a Recurrent neural network?**They differ in what kind of connections are allowed where in a feed forward network only forward directed signals/impulses are allowed. They differ also in and how backpropagation is done where one in a recurrent neural networks needs to choose appropriate time slice to use for back propagation training.
14. **Explain why one needs to interconnect neurons?**A single neuron cannot “simulate” complex functions, this can be proven using math. Therefore one needs to combine neurons to be able to express more complicated functions such as XOR.
15. **Explain a McCulloch Pitts neuron(MCP) differs from today’s artificial neurons?**  
    A MCP neuron does not output any signal when it receives an inhibitory signal compared to artificial neurons used today which calculate the sum of the negative “inhibitionary” with the positive “excitatory“ signals to decide whether or not to fire.
16. **What is deep learning? (Might be other definitions used for this? This one is not the best)**Deep learning is when one combines several “pre-trained” ordinary neural networks and then adjusts the whole network using ordinary back-propagation.
17. **Discuss the differences between the scalability in ANN and traditional ML models**Neural Networks scales better. Neural Networks scales well with big amounts of data, since it's always possible to increase the network size. In contrast more traditional models don't scale as well.
18. **Explain what underfitting is**The training data when the model performs poorly on the training data. This is

because the model is unable to capture the relationship between the input examples

and the target values.

1. **Give some examples where ANN is good**To infer a function from observations. Particularly useful in applications where the complexity of the data or task makes the design by hand, impractical. Function approximation, regression analysis including time series prediction, fitness approximation and modeling. Classification, including pattern and sequence recognition, novelty detection and sequential decision making. Data processing, including filtering, clustering, blind source separation and compression.
2. **What is a activation function?**The activation function of a node defines the output of that node, or "neuron,"

given an input or set of inputs. This output is then used as input for the next node

and so on until a desired solution to the original problem is found.

1. **Explain Forward Propagation in Neural Networks**Forward Propagation is when actual input data is fed to the neural network, interpreted and then used to estimate an output. For a neural network that tries to recognize handwritten characters, the input would be all the pixel data. Then the hidden layer would take this data as input, using certain biases, applying the activation function to it, and the hidden layer output would similarly be used by the output layer. The output layer would have one neuron for each character, and the end results would represent how likely it is that it's that specific character.
2. **Explain Weights and Backward Propagation in Neural Networks**Backward Propagation can be used to train a neural network. It's a form of supervised training, since it requires the ideal output to be known in every specific case. A neural network will use weights to give each neuron their unique characteristics. At first they are random, but the purpose of a neural network is to connect specific multilayered input patterns to specific output results. By adjusting the weights, you're basically telling the neuron to care less, or more about the input from a specific input neuron. Backward propagation is the process where you evaluate the output, compares it to what it should have been (margin of error), and then updates the weights so that they become more accurate. Then eventually, by fine tuning the weights via backward propagation, the neural network will get better at predicting the correct output.
3. **What does the Deep part of a Deep Neural Network mean?**

Deep Neural Networks have multilayered hidden layers. The purpose of this is that it ideally enables the neural network to recognize more complex patterns, the first layer picking up single features, and the following layers hopefully combining them into specific more complex combinations.

1. **Explain Normalization (I think there are more to normalization though)**

Normalization, or normalizing the data is about scaling all numeric values in range [0,1]. The point is to adjust numbers measured on certain different scales, to a certain common scale.

1. **Explain the term Convolutional Neural Network(CNN) (bad answer, but important question, so I left it here)**Convolutional Neural Network(CNN) is a type of ANN where the connection between units is inspired by the connection between neurons in the brain. Moreover, CNN consists of multiple layers. The use of multilayering also reduces the need for preprocessing.
2. **What are the differences between Back-Propagation and Genetic algorithms in the training of neural networks?**

Genetic algorithms are algorithm used for optimisation problems and therefore can find an optimal NN design, with the smallest possible training error. By defining a population size, mutation rate and max generations, different network designs will be tested. Since the entire given domain is taken into account, GAs will probably do better in the long run by avoiding local optimas, which cause troubles for back-propagated NN. Back-Propagation is used to find optimal weights and bias values, which can get stuck in local minima and therefore be not optimal. Furthermore, the layout of the NN remains unchanged which restricts the quality of the final NN in comparison to NN trained with GAs. However, the training process based on GAs takes much longer.

1. **How does the training of neural networks work for unsupervised problems?**

Since the input data are unlabelled, backpropagation can not be applied as a training algorithm. Instead the learning is based on the Hebbian law, which states that neurons that fire together should have a stronger connection. One example of the real world are auto-encoders, which transform the input in a lower dimension. The retransformation is used to calculate a reconstruction error (distance from the ID function) which is then used to adjust the weights. For clustering problems, self organizing maps are most likely be used. A SOM is a NN that has a set of neurons connected to form a low dimensional topology preserving grid of a high dimensional input space. They apply competitive learning as opposed to error-correction learning (such as backpropagation with gradient descent), and in the sense that they use a neighbourhood function to preserve the.

1. **Why is a normalization of the input data necessary?**In theory, it's not necessary to normalize numeric x-data (also called independent data). However, practice has shown that when numeric x-data values are normalized, neural network training is often more efficient, which leads to a better predictor. Basically, if numeric data is not normalized, and the magnitudes of two predictors are far apart, then a change in the value of a neural network weight has far more relative influence on the x-value with larger magnitudes. For example, for the first line of raw data, a neural network weight change of 0.1 will change magnitude of the age factor by (0.1 \* 30) = 3, but will change the income factor by (0.1 \* 38,000) = 3,800.
2. **What is overfitting and how can it be mitigated?**Overfitting is an undesirable effect due to poor generalisation (off bias-variance balance) since the network only memorizing the data it has seen which prevents it from working properly on unseen data. Overfitting usually manifests itself by large test error (unseen data), especially when compared to training error. Regularization techniques, such as pruning (dropout), early stopping, adding noise,.. can be used to mitigate overfitting.
3. **Explain the difference between online and offline learning.**With online learning the model learns during runtime while with offline learning the model can only learn from the training sessions.
4. **What makes a neural network a deep neural network? (Tror inte det här stämmer), finns olika definitioner antagligen, men denna är ganska vanlig**A neural network with 2 or more hidden layers is considered a deep neural network.
5. **A feedforward neural network can consist of three types of nodes. Please describe the three nodes.***Input Nodes –* The Input nodes provide information from the outside world to the

network and are together referred to as the “Input Layer”. No computation is performed in any of the Input nodes – they just pass on the information to the hidden nodes.

*Hidden Nodes –* The Hidden nodes have no direct connection with the outside world

(hence the name “hidden”). They perform computations and transfer information from

the input nodes to the output nodes. A collection of hidden nodes forms a “Hidden Layer”. While a feedforward network will only have a single input layer and a single output layer, it can have zero or multiple Hidden Layers.   
*Output Nodes –* The Output nodes are collectively referred to as the “Output Layer” and are responsible for computations and transferring information from the network to the outside world.

1. **Did Warren McCulloch and Walter Pitts create a neural network called S N A R C?**No, the creators of S N A R C was Marvin Minsky and Dean Edmonds.
2. **Activation functions introduce a nonlinearity into the learning model, what**

**does that allow us to do?**

It allows us to make complex predictions, without it, predictions will always be

linear.

1. **Explain what is a neural network?**A neural network is a system based on three steps: input, calculations and output. The input triggers perceptrons in the hidden layer that will either fire or not fire depending on the activation function. The hidden layers are connected to the output, if enough hidden perceptrons are fired then an output is fired. The perceptrons in the neural network forms a connected graph. All perceptrons in the input are connected to all perceptrons in the hidden layer(assuming we only have one hidden layer) and all perceptrons in the hidden layer are connected to the perceptrons in the output layer. The connections between perceptrons are weighted, this is because the perceptrons should have different amount of impact on each other.
2. **Discuss, why do we randomise the weights in the beginning?**We randomize the values of the weights to “try” to avoid a system where all nodes have the same activation value. If we just set all the weights to 0 we will have no perceptrons firing to anything. If we have all weights to the same value, the we might get stuck in a pattern were the solution is not optimal. To get the best result we randomise the weights to increase our chance to of finding the optimal solution. With other words, it uses heuristics.
3. **What is the bias used for?**The bias is a value that adjust the sum of an activation function. You move the value in to the “critical zone” of the activation function.
4. **Explain: what is Fire/Not Fire?**Fire or not fire are the states of a perceptron. If the perceptron is activated it’s called fire and if it’s not activated it’s called not fire. See more on question below.
5. **Explain: what is activation level?**Activation level is the calculated value that is compared to the threshold. If the activation level is equal or higher than the threshold, the perceptron Fires. The activation level is calculated by adding each input multiplied with its weight and then adding the bias.
6. **What is a perceptron?**The nodes/neurons in a neural network are called perceptrons. Perceptrons are linear classifiers that uses the activation level and threshold to either activate or not.
7. **Which algorithm do you know to train a neural network? Briefly describe one.**From machine learning: Supervised learning, unsupervised learning, reinforcement learning, Back-propagation algorithm

Genetic algorithms: Generate a number of random networks and compute their fitness based on their outcome (classification accuracy). Keep the best networks and breed children out of them. Randomly pick networks and change their weights and check again how good they fit. Repeating this several times is time consuming. A slow learning technique which requires the population size, mutation rate, max generations, selection probability.

1. **Explain the similarities between a human neuron and a software perceptron?**

The similarities consists of the way a perceptron or a neuron computes information. Both got a number of inputs, compute and then decide if they should fire or not.

1. **Describe a situation where a neural network would compute good results but should nevertheless not be used.**When for example a customer needs to know why a decision is good. ANN’s can usually not explain it’s decisions.
2. **Explain why the weights in a perceptron or a neural network are important.**The weights of the inputs of perceptrons and neural networks are the key factor for the ability to learn of these systems. The weights are adjusted every time a training dataset is processed. This mechanism adjusts the system until it makes good predictions.
3. **Why are hidden layers called hidden layers?**

Hidden layers are the layers between the input and the output layer. These layers are called hidden layers because they are not visible from the outside. This is a key factor of the appearance of a neural network as a black box. The user does not exactly know what happens inside.

1. **Are neural networks always the best choice? Describe a situation where it is not.**Of course neural networks are not always the best choice. In general neural networks are able to make the best decisions but only if sufficient resources are available. This means a neural network needs far more training data, computing power and time to be trained well compared with an easier algorithm like naïve bayes. This means if you have limited resources it may be better to choose another algorithm.