**第一门课 神经网络和深度学习(Neural Networks and Deep Learning)**

**第一周：深度学习引言(Introduction to Deep Learning)**

1.1 欢迎(Welcome) 1.2 什么是神经网络？(What is a Neural Network)

1.3 神经网络的监督学习(Supervised Learning with Neural Networks)

1.4 为什么神经网络会流行？(Why is Deep Learning taking off?)

1.5 关于本课程(About this Course) 1.6 课程资源(Course Resources)

**第二周：神经网络的编程基础(Basics of Neural Network programming)**

2.1 二分类(Binary Classification)

2.2 逻辑回归(Logistic Regression)

2.3 逻辑回归的代价函数（Logistic Regression Cost Function）

2.4 梯度下降（Gradient Descent）

2.5 导数（Derivatives）

2.6 更多的导数例子（More Derivative Examples）

2.7 计算图（Computation Graph）

2.8 计算图导数（Derivatives with a Computation Graph）

2.9 逻辑回归的梯度下降（Logistic Regression Gradient Descent）

2.10 梯度下降的例子(Gradient Descent on m Examples)

2.11 向量化(Vectorization)

2.12 更多的向量化例子（More Examples of Vectorization）

2.13 向量化逻辑回归(Vectorizing Logistic Regression)

2.14 向量化逻辑回归的梯度计算（Vectorizing Logistic Regression's Gradient）

2.15 Python中的广播机制（Broadcasting in Python）

2.16 关于 Python与numpy向量的使用（A note on python or numpy vectors）

2.17 Jupyter/iPython Notebooks快速入门（Quick tour of Jupyter/iPython Notebooks）

2.18 逻辑回归损失函数详解（Explanation of logistic regression cost function）

**第三周：浅层神经网络(Shallow neural networks)**

3.1 神经网络概述（Neural Network Overview）

3.2 神经网络的表示（Neural Network Representation）

3.3 计算一个神经网络的输出（Computing a Neural Network's output）

3.4 多样本向量化（Vectorizing across multiple examples）

3.5 向量化实现的解释（Justification for vectorized implementation）

3.6 激活函数（Activation functions）

3.7 为什么需要非线性激活函数？（why need a nonlinear activation function?）

3.8 激活函数的导数（Derivatives of activation functions）

3.9 神经网络的梯度下降（Gradient descent for neural networks）

3.10（选修）直观理解反向传播（Backpropagation intuition）

3.11 随机初始化（Random+Initialization）

**第四周：深层神经网络(Deep Neural Networks)**

4.1 深层神经网络（Deep L-layer neural network）

4.2 前向传播和反向传播（Forward and backward propagation）

4.3 深层网络中的前向和反向传播（Forward propagation in a Deep Network）

4.4 核对矩阵的维数（Getting your matrix dimensions right）

4.5 为什么使用深层表示？（Why deep representations?）

4.6 搭建神经网络块（Building blocks of deep neural networks）

4.7 参数VS超参数（Parameters vs Hyperparameters）

4.8 深度学习和大脑的关联性（What does this have to do with the brain?）

**第二门课 改善深层神经网络：超参数调试、正则化以及优化(Improving Deep Neural Networks:Hyperparameter tuning, Regularization and Optimization)**

**第一周：深度学习的实用层面(Practical aspects of Deep Learning)**

1.1 训练，验证，测试集（Train / Dev / Test sets）

1.2 偏差，方差（Bias /Variance）

1.3 机器学习基础（Basic Recipe for Machine Learning）

1.4 正则化（Regularization）

1.5 为什么正则化有利于预防过拟合呢？（Why regularization reduces overfitting?）

1.6 dropout 正则化（Dropout Regularization）

1.7 理解 dropout（Understanding Dropout）

1.8 其他正则化方法（Other regularization methods）

1.9 标准化输入（Normalizing inputs）

1.10 梯度消失/梯度爆炸（Vanishing / Exploding gradients）

1.11 神经网络的权重初始化（Weight Initialization for Deep NetworksVanishing /Exploding gradients）

1.12 梯度的数值逼近（Numerical approximation of gradients）

1.13 梯度检验（Gradient checking）

1.14 梯度检验应用的注意事项（Gradient Checking Implementation Notes）

**第二周：优化算法 (Optimization algorithms)**

2.1 Mini-batch 梯度下降（Mini-batch gradient descent）

2.2 理解Mini-batch 梯度下降（Understanding Mini-batch gradient descent）

2.3 指数加权平均（Exponentially weighted averages）

2.4 理解指数加权平均（Understanding Exponentially weighted averages）

2.5 指数加权平均的偏差修正（Bias correction in exponentially weighted averages）

2.6 momentum梯度下降（Gradient descent with momentum）

2.7 RMSprop——root mean square prop（RMSprop）

2.8 Adam优化算法（Adam optimization algorithm）

2.9 学习率衰减（Learning rate decay）

2.10 局部最优问题（The problem of local optima）

**第三周超参数调试，batch正则化和程序框架（Hyperparameter tuning, Batch Normalization and Programming Frameworks)**

3.1 调试处理（Tuning process）

3.2 为超参数选择和适合范围（Using an appropriate scale to pick hyperparameters）

3.3 超参数训练的实践：Pandas vs. Caviar（Hyperparameters tuning in practice: Pandas vs. Caviar）

3.4 网络中的正则化激活函数（Normalizing activations in a network）

3.5 将 Batch Norm拟合进神经网络（Fitting Batch Norm into a neural network）

3.6 为什么Batch Norm奏效？（Why does Batch Norm work?）

3.7 测试时的Batch Norm（Batch Norm at test time）

3.8 Softmax 回归（Softmax Regression）

3.9 训练一个Softmax 分类器（Training a softmax classifier）

3.10 深度学习框架（Deep learning frameworks）

3.11 TensorFlow（TensorFlow）

**第三门课 结构化机器学习项目 (Structuring Machine Learning Projects)**

**第一周：机器学习策略（1）(ML Strategy (1))**

1.1 为什么是ML策略？ (Why ML Strategy)

1.2 正交化(Orthogonalization)

1.3 单一数字评估指标(Single number evaluation metric)

1.4 满足和优化指标 (Satisficing and Optimizing metric)

1.5 训练集、开发集、测试集的划分(Train/dev/test distributions)

1.6 开发集和测试集的大小 (Size of the dev and test sets)

1.7 什么时候改变开发集/测试集和评估指标(When to change dev/test sets and metrics)

1.8 为什么是人的表现 (Why human-level performance?)

1.9 可避免偏差(Avoidable bias)

1.10 理解人类的表现 (Understanding human-level performance)

1.11 超过人类的表现(Surpassing human-level performance)

1.12 改善你的模型表现 (Improving your model performance)

**第二周：机器学习策略（2）(ML Strategy (2))**

2.1 误差分析 (Carrying out error analysis)

2.2 清除标注错误的数据(Cleaning up incorrectly labeled data)

2.3 快速搭建你的第一个系统，并进行迭代(Build your first system quickly, then iterate)

2.4 在不同的分布上的训练集和测试集 (Training and testing on different distributions)

2.5 数据分布不匹配的偏差与方差分析 (Bias and Variance with mismatched data distributions)

2.6 处理数据不匹配问题(Addressing data mismatch)

2.7 迁移学习 (Transfer learning)

2.8 多任务学习(Multi-task learning)

2.9 什么是端到端的深度学习？ (What is end-to-end deep learning?)

2.10 是否使用端到端的深度学习方法 (Whether to use end-to-end deep learning)

**第四门课 卷积神经网络（Convolutional Neural Networks）**

**第一周 卷积神经网络(Foundations of Convolutional Neural Networks)**

2.1.1 计算机视觉（Computer vision）

1.2 边缘检测示例（Edge detection example）

1.3 更多边缘检测内容（More edge detection）

1.4 Padding

1.5 卷积步长（Strided convolutions）

1.6 三维卷积（Convolutions over volumes）

1.7 单层卷积网络（One layer of a convolutional network）

1.8 简单卷积网络示例（A simple convolution network example）

1.9 池化层（Pooling layers）

1.10 卷积神经网络示例（Convolutional neural network example）

1.11 为什么使用卷积？（Why convolutions?）

**第二周 深度卷积网络：实例探究(Deep convolutional models: case studies)**

2.1 为什么要进行实例探究？（Why look at case studies?）

2.2 经典网络（Classic networks）

2.3 残差网络（Residual Networks (ResNets)）

2.4 残差网络为什么有用？（Why ResNets work?）

2.5 网络中的网络以及 1×1 卷积（Network in Network and 1×1 convolutions）

2.6 谷歌 Inception 网络简介（Inception network motivation）

2.7 Inception 网络（Inception network）

2.8 使用开源的实现方案（Using open-source implementations）

2.9 迁移学习（Transfer Learning）

2.10 数据扩充（Data augmentation）

2.11 计算机视觉现状（The state of computer vision）

**第三周 目标检测（Object detection）**

3.1 目标定位（Object localization）

3.2 特征点检测（Landmark detection）

3.3 目标检测（Object detection）

3.4 卷积的滑动窗口实现（Convolutional implementation of sliding windows）

3.5 Bounding Box预测（Bounding box predictions）

3.6 交并比（Intersection over union）

3.7 非极大值抑制（Non-max suppression）

3.8 Anchor Boxes

3.9 YOLO 算法（Putting it together: YOLO algorithm）

3.10 候选区域（选修）（Region proposals (Optional)）

**第四周 特殊应用：人脸识别和神经风格转换（Special applications: Face recognition &Neural style transfer）**

4.1 什么是人脸识别？(What is face recognition?)

4.2 One-Shot学习（One-shot learning）

4.3 Siamese 网络（Siamese network）

4.4 Triplet 损失（Triplet 损失）

4.5 面部验证与二分类（Face verification and binary classification）

4.6 什么是神经风格转换？（What is neural style transfer?）

4.7 什么是深度卷积网络？（What are deep ConvNets learning?）

4.8 代价函数（Cost function）

4.9 内容代价函数（Content cost function）

4.10 风格代价函数（Style cost function）

4.11 一维到三维推广（1D and 3D generalizations of models）

**第五门课 序列模型(Sequence Models)**

**第一周 循环序列模型（Recurrent Neural Networks）**

1.1 为什么选择序列模型？（Why Sequence Models?）

1.2 数学符号（Notation）

1.3 循环神经网络模型（Recurrent Neural Network Model）

1.4 通过时间的反向传播（Backpropagation through time）

1.5 不同类型的循环神经网络（Different types of RNNs）

1.6 语言模型和序列生成（Language model and sequence generation）

1.7 对新序列采样（Sampling novel sequences）

1.8 循环神经网络的梯度消失（Vanishing gradients with RNNs）

1.9 GRU单元（Gated Recurrent Unit（GRU））

1.10 长短期记忆（LSTM（long short term memory）unit）

1.11 双向循环神经网络（Bidirectional RNN）

1.12 深层循环神经网络（Deep RNNs）

**第二周 自然语言处理与词嵌入（Natural Language Processing and Word Embeddings）**

2.1 词汇表征（Word Representation）

2.2 使用词嵌入（Using Word Embeddings）

2.3 词嵌入的特性（Properties of Word Embeddings）

2.4 嵌入矩阵（Embedding Matrix）

2.5 学习词嵌入（Learning Word Embeddings）

2.6 Word2Vec

2.7 负采样（Negative Sampling）

2.8 GloVe 词向量（GloVe Word Vectors）

2.9 情绪分类（Sentiment Classification）

2.10 词嵌入除偏（Debiasing Word Embeddings）

**第三周 序列模型和注意力机制（Sequence models & Attention mechanism**

3.1 基础模型（Basic Models）

3.2 选择最可能的句子（Picking the most likely sentence）

3.3 集束搜索（Beam Search）

3.4 改进集束搜索（Refinements to Beam Search）

3.5 集束搜索的误差分析（Error analysis in beam search）

3.6 Bleu 得分（选修）（Bleu Score (optional)）

3.7 注意力模型直观理解（Attention Model Intuition）

3.8注意力模型（Attention Model）

3.9语音识别（Speech recognition）

3.10触发字检测（Trigger Word Detection）

3.11结论和致谢（Conclusion and thank you）

吴恩达采访人工智能大师访谈（interview）:

Geoffery Hinton / Ian Goodfellow / Ruslan Salakhutdinov / Yoshua Bengio / 林元庆 / Pieter Abbeel / Andrej Karpathy

深度学习符号指南（原课程翻译）/课程简介和致谢