

目录 2

第一章 数学基础 1

- 1.1标量、向量、张量之间的联系 1
- 1.2张量与矩阵的区别？ 1
- 1.3矩阵和向量相乘结果 1
- 1.4向量和矩阵的范数归纳 1
- 1.5如何判断一个矩阵为正定？ 2
- 1.6导数偏导计算 3
- 1.7导数和偏导数有什么区别？ 3
- 1.8特征值分解与特征向量 3
- 1.9奇异值与特征值有什么关系？ 4
- 1.10机器学习为什么要使用概率？ 4
- 1.11变量与随机变量有什么区别？ 4
- 1.12常见概率分布？ 5
- 1.13举例理解条件概率 9
- 1.14联合概率与边缘概率联系区别？ 10
- 1.15条件概率的链式法则 10
- 1.16独立性和条件独立性 11
- 1.17期望、方差、协方差、相关系数总结 11

第二章 机器学习基础 14

- 2.1 各种常见算法图示 14
- 2.2监督学习、非监督学习、半监督学习、弱监督学习？ 15
- 2.3 监督学习有哪些步骤 16
- 2.4 多实例学习？ 17
- 2.5 分类网络和回归的区别？ 17
- 2.6 什么是神经网络？ 17

2.7 常用分类算法的优缺点？	18
2.8 正确率能很好的评估分类算法吗？	20
2.9 分类算法的评估方法？	20
2.10 什么样的分类器是最好的？	22
2.11 大数据与深度学习的关系	22
2.12 理解局部最优与全局最优	23
2.13 理解逻辑回归	24
2.14 逻辑回归与朴素贝叶斯有什么区别？	24
2.15 为什么需要代价函数？	25
2.16 代价函数作用原理	25
2.17 为什么代价函数要非负？	26
2.18 常见代价函数？	26
2.19 为什么用交叉熵代替二次代价函数	28
2.20 什么是损失函数？	28
2.21 常见的损失函数	28
2.22 逻辑回归为什么使用对数损失函数？	30
0.00 对数损失函数是如何度量损失的？	31
2.23 机器学习中为什么需要梯度下降？	32
2.24 梯度下降法缺点？	32
2.25 梯度下降法直观理解？	32
2.23 梯度下降法算法描述？	33
2.24 如何对梯度下降法进行调优？	35
2.25 随机梯度和批量梯度区别？	35
2.26 各种梯度下降法性能比较	37
2.27 计算图的导数计算图解？	37
2.28 线性判别分析 (LDA) 思想总结	39
2.29 图解LDA核心思想	39
2.30 二类LDA算法原理？	40
2.30 LDA算法流程总结？	41
2.31 LDA和PCA区别？	41
2.32 LDA优缺点？	41
2.33 主成分分析 (PCA) 思想总结	42

2.34 图解PCA核心思想	42
2.35 PCA算法推理	43
2.36 PCA算法流程总结	44
2.37 PCA算法主要优缺点	45
2.38 降维的必要性及目的	45
2.39 KPCA与PCA的区别?	46
2.40模型评估	47
2.40.1模型评估常用方法?	47
2.40.2 经验误差与泛化误差	47
2.40.3 图解欠拟合、过拟合	48
2.40.4 如何解决过拟合与欠拟合?	49
2.40.5 交叉验证的主要作用?	50
2.40.6 k折交叉验证?	50
2.40.7 混淆矩阵	50
2.40.8 错误率及精度	51
2.40.9 查准率与查全率	51
2.40.10 ROC与AUC	52
2.40.11如何画ROC曲线?	53
2.40.12如何计算TPR, FPR?	54
2.40.13如何计算Auc?	56
2.40.14为什么使用Roc和Auc评价分类器?	56
2.40.15 直观理解AUC	56
2.40.16 代价敏感错误率与代价曲线	57
2.40.17 模型有哪些比较检验方法	59
2.40.18 偏差与方差	59
2.40.19为什么使用标准差?	60
2.40.20 点估计思想	61
2.40.21 点估计优良性原则?	61
2.40.22点估计、区间估计、中心极限定理之间的联系?	62
2.40.23 类别不平衡产生原因?	62
2.40.24 常见的类别不平衡问题解决方法	62
2.41 决策树	64

- 2.41.1 决策树的基本原理 64
- 2.41.2 决策树的三要素? 64
- 2.41.3 决策树学习基本算法 65
- 2.41.4 决策树算法优缺点 65
- 2.40.5 熵的概念以及理解 66
- 2.40.6 信息增益的理解 66
- 2.40.7 剪枝处理的作用及策略? 67
- 2.41 支持向量机 67
- 2.41.1 什么是支持向量机 67
- 2.25.2 支持向量机解决的问题? 68
- 2.25.2 核函数作用? 69
- 2.25.3 对偶问题 69
- 2.25.4 理解支持向量回归 69
- 2.25.5 理解SVM (核函数) 69
- 2.25.6 常见的核函数有哪些? 69
- 2.25.6 软间隔与正则化 73
- 2.25.7 SVM主要特点及缺点? 73
- 2.26 贝叶斯 74
- 2.26.1 图解极大似然估计 74
- 2.26.2 朴素贝叶斯分类器和一般的贝叶斯分类器有什么区别? 76
- 2.26.4 朴素与半朴素贝叶斯分类器 76
- 2.26.5 贝叶斯网三种典型结构 76
- 2.26.6 什么是贝叶斯错误率 76
- 2.26.7 什么是贝叶斯最优错误率 76
- 2.27 EM算法解决问题及实现流程 76
- 2.28 为什么会产生维数灾难? 78
- 2.29 怎样避免维数灾难 82
- 2.30 聚类和降维有什么区别与联系? 82
- 2.31 GBDT和随机森林的区别 83
- 2.32 四种聚类方法之比较 84

第三章 深度学习基础 88

3.1基本概念 88

3.1.1神经网络组成？ 88

3.1.2神经网络有哪些常用模型结构？ 90

3.1.3如何选择深度学习开发平台？ 92

3.1.4为什么使用深层表示 92

3.1.5为什么深层神经网络难以训练？ 93

3.1.6深度学习和机器学习有什么不同 94

3.2 网络操作与计算 95

3.2.1前向传播与反向传播？ 95

3.2.2如何计算神经网络的输出？ 97

3.2.3如何计算卷积神经网络输出值？ 98

3.2.4如何计算Pooling层输出值输出值？ 101

3.2.5实例理解反向传播 102

3.3超参数 105

3.3.1什么是超参数？ 105

3.3.2如何寻找超参数的最优值？ 105

3.3.3超参数搜索一般过程？ 106

3.4激活函数 106

3.4.1为什么需要非线性激活函数？ 106

3.4.2常见的激活函数及图像 107

3.4.3 常见激活函数的导数计算？ 109

3.4.4激活函数有哪些性质？ 110

3.4.5 如何选择激活函数？ 110

3.4.6使用ReLu激活函数的优点？ 111

3.4.7什么时候可以用线性激活函数？ 111

3.4.8怎样理解Relu (<0 时) 是非线性激活函数？ 111

3.4.9 Softmax函数如何应用于多分类？ 112

3.5 Batch_Size 113

3.5.1为什么需要Batch_Size？ 113

3.5.2 Batch_Size值的选择	114
3.5.3在合理范围内, 增大 Batch_Size 有何好处?	114
3.5.4盲目增大 Batch_Size 有何坏处?	114
3.5.5调节 Batch_Size 对训练效果影响到底如何?	114
3.6 归一化	115
3.6.1归一化含义?	115
3.6.2为什么要归一化	115
3.6.3为什么归一化能提高求解最优解速度?	115
3.6.4 3D图解未归一化	116
3.6.5归一化有哪些类型?	117
3.6.6局部响应归一化作用	117
3.6.7理解局部响应归一化公式	117
3.6.8什么是批归一化 (Batch Normalization)	118
3.6.9批归一化 (BN) 算法的优点	119
3.6.10批归一化 (BN) 算法流程	119
3.6.11批归一化和群组归一化	120
3.6.12 Weight Normalization和Batch Normalization	120
3.7 预训练与微调(fine tuning)	121
3.7.1为什么无监督预训练可以帮助深度学习?	121
3.7.2什么是模型微调fine tuning	121
3.7.3微调时候网络参数是否更新?	122
3.7.4 fine-tuning模型的三种状态	122
3.8权重偏差初始化	122
3.8.1 全都初始化为0	122
3.8.2 全都初始化为同样的值	123
3.8.3 初始化为小的随机数	124
3.8.4用 $1/\sqrt{n}$ 校准方差	125
3.8.5稀疏初始化(Sparse Initialazation)	125
3.8.6初始化偏差	125
3.9 Softmax	126
3.9.1 Softmax定义及作用	126
3.9.2 Softmax推导	126

- 3.10 理解One Hot Encodeing原理及作用? 126
- 3.11 常用的优化器有哪些 127
- 3.12 Dropout 系列问题 128
 - 3.12.1 dropout率的选择 128
- 3.27 Padding 系列问题 128

第四章 经典网络 129

- 4.1LetNet5 129
 - 4.1.1模型结构 129
 - 4.1.2模型结构 129
 - 4.1.3 模型特性 131
- 4.2 AlexNet 131
 - 4.2.1 模型结构 131
 - 4.2.2模型解读 131
 - 4.2.3模型特性 135
- 4.3 可视化ZFNet-解卷积 135
 - 4.3.1 基本的思想及其过程 135
 - 4.3.2 卷积与解卷积 136
 - 4.3.3卷积可视化 137
 - 4.3.4 ZFNe和AlexNet比较 139
- 4.4 VGG 140
 - 4.1.1 模型结构 140
 - 4.1.2 模型特点 140
- 4.5 Network in Network 141
 - 4.5.1 模型结构 141
 - 4.5.2 模型创新点 141
- 4.6 GoogleNet 143
 - 4.6.1 模型结构 143
 - 4.6.2 Inception 结构 145
 - 4.6.3 模型层次关系 146

4.7 Inception 系列	148
4.7.1 Inception v1	148
4.7.2 Inception v2	150
4.7.3 Inception v3	153
4.7.4 Inception V4	155
4.7.5 Inception-ResNet-v2	157
4.8 ResNet及其变体	158
4.8.1重新审视ResNet	159
4.8.2残差块	160
4.8.3 ResNet架构	162
4.8.4残差块的变体	162
4.8.5 ResNeXt	162
4.8.6 Densely Connected CNN	164
4.8.7 ResNet作为小型网络的组合	165
4.8.8 ResNet中路径的特点	166
4.9为什么现在的CNN模型都是在GoogleNet、VGGNet或者AlexNet上调整的？	167

第五章 卷积神经网络(CNN) 170

5.1 卷积神经网络的组成层	170
5.2 卷积如何检测边缘信息？	171
5.2 卷积的几个基本定义？	174
5.2.1卷积核大小	174
5.2.2卷积核的步长	174
5.2.3边缘填充	174
5.2.4输入和输出通道	174
5.3 卷积网络类型分类？	174
5.3.1普通卷积	174
5.3.2扩张卷积	175
5.3.3转置卷积	176
5.3.4可分离卷积	177

5.3 图解12种不同类型的2D卷积?	178
5.4 2D卷积与3D卷积有什么区别?	181
5.4.1 2D 卷积	181
5.4.2 3D卷积	182
5.5 有哪些池化方法?	183
5.5.1一般池化 (General Pooling)	183
5.5.2重叠池化 (Overlapping Pooling)	184
5.5.3空金字塔池化 (Spatial Pyramid Pooling)	184
5.6 1x1卷积作用?	186
5.7卷积层和池化层有什么区别?	187
5.8卷积核一定越大越好?	189
5.9每层卷积只能用一种尺寸的卷积核?	189
5.10怎样才能减少卷积层参数量?	190
5.11卷积操作时必须同时考虑通道和区域吗?	191
5.12采用宽卷积的好处有什么?	192
5.12.1窄卷积和宽卷积	192
5.12.2 为什么采用宽卷积?	192
5.13卷积层输出的深度与哪个部件的个数相同?	192
5.14 如何得到卷积层输出的深度?	193
5.15激活函数通常放在卷积神经网络的那个操作之后?	194
5.16 如何理解最大池化层有几分缩小?	194
5.17理解图像卷积与反卷积	194
5.17.1图像卷积	194
5.17.2图像反卷积	196
5.18不同卷积后图像大小计算?	198
5.18.1 类型划分	198
5.18.2 计算公式	199
5.19 步长、填充大小与输入输出关系总结?	199
5.19.1没有0填充, 单位步长	200
5.19.2零填充, 单位步长	200
5.19.3不填充, 非单位步长	202
5.19.4零填充, 非单位步长	202

5.20 理解反卷积和棋盘效应	204
5.20.1为什么出现棋盘现象?	204
5.20.2 有哪些方法可以避免棋盘效应?	205
5.21 CNN主要的计算瓶颈?	207
5.22 CNN的参数经验设置	207
5.23 提高泛化能力的方法总结	208
5.23.1 主要方法	208
5.23.2 实验证明	208
5.24 CNN在CV与NLP领域运用的联系与区别?	213
5.24.1联系	213
5.24.2区别	213
5.25 CNN凸显共性的手段?	213
5.25.1 局部连接	213
5.25.2 权值共享	214
5.25.3 池化操作	215
5.26 全卷积与Local-Conv的异同点	215
5.27 举例理解Local-Conv的作用	215
5.28 简述卷积神经网络进化史	216

第六章 循环神经网络(RNN) 218

6.1 RNNs和FNNs有什么区别?	218
6.2 RNNs典型特点?	218
6.3 RNNs能干什么?	219
6.4 RNNs在NLP中典型应用?	220
6.5 RNNs训练和传统ANN训练异同点?	220
6.6常见的RNNs扩展和改进模型	221
6.6.1 Simple RNNs(SRNs)	221
6.6.2 Bidirectional RNNs	221
6.6.3 Deep(Bidirectional) RNNs	222
6.6.4 Echo State Networks (ESNs)	222

6.6.5 Gated Recurrent Unit Recurrent Neural Networks 224

6.6.6 LSTM Networks 224

6.6.7 Clockwork RNNs(CW-RNNs) 225

第七章 目标检测 228

7.1 基于候选区域的目标检测器 228

7.1.1 滑动窗口检测器 228

7.1.2 选择性搜索 229

7.1.3 R-CNN 230

7.1.4 边界框回归器 230

7.1.5 Fast R-CNN 231

7.1.6 ROI 池化 233

7.1.7 Faster R-CNN 233

7.1.8 候选区域网络 234

7.1.9 R-CNN 方法的性能 236

7.2 基于区域的全卷积神经网络 (R-FCN) 237

7.3 单次目标检测器 240

7.3.1 单次检测器 241

7.3.2 滑动窗口进行预测 241

7.3.3 SSD 243

7.4 YOLO系列 244

7.4.1 YOLOv1介绍 244

7.4.2 YOLOv1模型优缺点? 252

7.4.3 YOLOv2 253

7.4.4 YOLOv2改进策略 254

7.4.5 YOLOv2的训练 261

7.4.6 YOLO9000 261

7.4.7 YOLOv3 263

7.4.8 YOLOv3改进 264

第八章 图像分割 269

- 8.1 传统的基于CNN的分割方法缺点? 269
 - 8.1 FCN 269
 - 8.1.1 FCN改变了什么? 269
 - 8.1.2 FCN网络结构? 270
 - 8.1.3全卷积网络举例? 271
 - 8.1.4为什么CNN对像素级别的分类很难? 271
 - 8.1.5全连接层和卷积层如何相互转化? 272
 - 8.1.6 FCN的输入图片为什么可以是任意大小? 272
 - 8.1.7把全连接层的权重 W 重塑成卷积层的滤波器有什么好处? 273
 - 8.1.8反卷积层理解 275
 - 8.1.9跳级(skip)结构 276
 - 8.1.10模型训练 277
 - 8.1.11 FCN缺点 280
 - 8.2 U-Net 280
 - 8.3 SegNet 282
 - 8.4空洞卷积(Dilated Convolutions) 283
 - 8.4 RefineNet 285
 - 8.5 PSPNet 286
 - 8.6 DeepLab系列 288
 - 8.6.1 DeepLabv1 288
 - 8.6.2 DeepLabv2 289
 - 8.6.3 DeepLabv3 289
 - 8.6.4 DeepLabv3+ 290
 - 8.7 Mask-RCNN 293
 - 8.7.1 Mask-RCNN 的网络结构示意图 293
 - 8.7.2 RCNN行人检测框架 293
 - 8.7.3 Mask-RCNN 技术要点 294
 - 8.8 CNN在基于弱监督学习的图像分割中的应用 295
 - 8.8.1 Scribble标记 295

- 8.8.2 图像级别标记 297
- 8.8.3 DeepLab+bounding box+image-level labels 298
- 8.8.4统一的框架 299

第九章 强化学习 301

- 9.1强化学习的主要特点？ 301
- 9.2强化学习应用实例 302
- 9.3强化学习和监督式学习、非监督式学习的区别 303
- 9.4 强化学习主要有哪些算法？ 305
- 9.5深度迁移强化学习算法 305
- 9.6分层深度强化学习算法 306
- 9.7深度记忆强化学习算法 306
- 9.8 多智能体深度强化学习算法 307
- 9.9深度强化学习算法小结 307

第十章 迁移学习 309

- 10.1 什么是迁移学习？ 309
- 10.2 什么是多任务学习？ 309
- 10.3 多任务学习有什么意义？ 309
- 10.4 什么是端到端的深度学习？ 311
- 10.5 端到端的深度学习举例？ 311
- 10.6 端到端的深度学习有什么挑战？ 311
- 10.7 端到端的深度学习优缺点？ 312

第十三章 优化算法 314

- 13.1 CPU和GPU 的区别？ 314
- 13.2如何解决训练样本少的问题 315
- 13.3 什么样的样本集不适合用深度学习? 315

13.4 有没有可能找到比已知算法更好的算法? 316

13.5 何为共线性, 跟过拟合有啥关联? 316

13.6 广义线性模型是怎被应用在深度学习中? 316

13.7 造成梯度消失的原因? 317

13.8 权值初始化方法有哪些 317

13.9 启发式优化算法中, 如何避免陷入局部最优解? 318

13.10 凸优化中如何改进GD方法以防止陷入局部最优解 319

13.11 常见的损失函数? 319

13.14 如何进行特征选择 (feature selection) ? 321

13.14.1 如何考虑特征选择 321

13.14.2 特征选择方法分类 321

13.14.3 特征选择目的 322

13.15 梯度消失/梯度爆炸原因, 以及解决方法 322

13.15.1 为什么要使用梯度更新规则? 322

13.15.2 梯度消失、爆炸原因? 323

13.15.3 梯度消失、爆炸的解决方案 324

13.16 深度学习为什么不用二阶优化 325

13.17 怎样优化你的深度学习系统? 326

13.18为什么要设置单一数字评估指标? 326

13.19满足和优化指标 (Satisficing and optimizing metrics) 327

13.20 怎样划分训练/开发/测试集 328

13.21如何划分开发/测试集大小 329

13.22什么时候该改变开发/测试集和指标? 329

13.23 设置评估指标的意义? 330

13.24 什么是可避免偏差? 331

13.25 什么是TOP5错误率? 331

13.26 什么是人类水平错误率? 332

13.27 可避免偏差、几大错误率之间的关系? 332

13.28 怎样选取可避免偏差及贝叶斯错误率? 332

13.29 怎样减少方差? 333

13.30贝叶斯错误率的最佳估计 333

13.31举机器学习超过单个人类表现几个例子? 334

- 13.32如何改善你的模型? 334
- 13.33 理解误差分析 335
- 13.34 为什么值得花时间查看错误标记数据? 336
- 13.35 快速搭建初始系统的意义? 336
- 13.36 为什么要在不同的划分上训练及测试? 337
- 13.37 如何解决数据不匹配问题? 338
- 13.38 梯度检验注意事项? 340
- 13.39什么是随机梯度下降? 341
- 13.40什么是批量梯度下降? 341
- 13.41什么是小批量梯度下降? 341
- 13.42怎么配置mini-batch梯度下降 342
- 13.43 局部最优的问题 343
- 13.44提升算法性能思路 346

第十四章 超参数调整 358

- 14.1 调试处理 358
- 14.2 有哪些超参数 359
- 14.3 如何选择调试值? 359
- 14.4 为超参数选择合适的范围 359
- 14.5 如何搜索超参数? 359

第十五章 正则化 361

- 15.1 什么是正则化? 361
- 15.2 正则化原理? 361
- 15.3 为什么要正则化? 361
- 15.4 为什么正则化有利于预防过拟合? 361
- 15.5 为什么正则化可以减少方差? 362
- 15.6 L2正则化的理解? 362
- 15.7 理解dropout 正则化 362

15.8 有哪些dropout 正则化方法? 362
15.8 如何实施dropout 正则化 363
15.9 Python 实现dropout 正则化 363
15.10 L2正则化和dropout 有什么不同? 363
15.11 dropout有什么缺点? 363
15.12 其他正则化方法? 364

参考文献 366
