机器学习视屏学习笔记

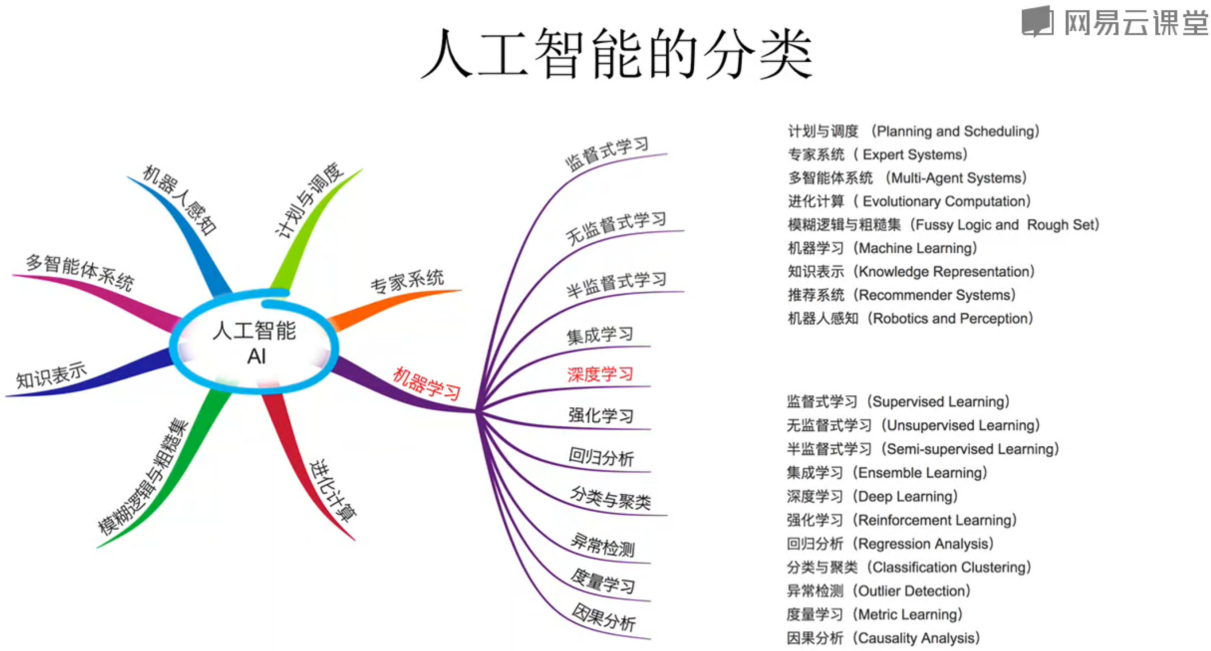
Machine Learning 2020-10-21 Daolin Sheng 机器学习

01. 人工智能，机器学习和深度学习关系

- 人工智能 = 数据 + 算法 + 算力
- 大数据时代（出行数据，消费数据，搜索数据等）海量 and 多维度
- 高性能计算机的崛起（CPU + GPU + TPU），分布式计算

- 弱人工智能（现代的人工智能： 数据+算法+算力）

- 强人工智能（认知智能）
- 超人工智能（人类自己要先成为超人，才能创造出来）



监督学习 - 数据打标签
训练集 - 测试集
提取特征值

- 图像，三维数组 200 * 100 * 3 （RGB）

有监督学习（Classification Algorithm） 和 无监督学习（Clustering Algorithm）

是否有标签

回归（Regression）分析与预测

人工智能的标志性事件

- 2016 - 李世石

传统机器学习算法 - 深度学习
ImageNet - 图像集合 - 机器学习算法评估
应用

- 翻译
- 文本描述
- 医疗领域 - 医疗诊断
- 自动驾驶

02. 机器学习工具

- Python （Anaconda）
- IDE (Pycharm)
- Jupyter notebook
- Scikit-Learn (sklearn) , 是一个通用的机器学习库。
    - TensorFlow（Google), PyTorch（Facebook) 深度学习库
    - 功能不同
    - 使用的自由度不同
    - 针对的群体，项目不同

03. Jupyter Notebook 简介与安装

- 交互性笔记本，用于编辑和运行各种编程语言，前身 IPython Notebook
- 好的界面，可视化
- 方便远程访问
- pip install jupyter
- jupyter notebook &

04. 使用Jupyter notebook

- 相关命令
-

05. 远程访问Jupyter notebook

- token / password
- 如何生成密码 jupyter generate config

06. 机器学习

- 人均GDP与幸福指数的关系
- 线性回归

07. 训练线性回归， 并预测幸福指数

- csv 数据
- numpy
- pandas
- sklearn

代码步骤：

- 装载数据
- 整理数据
- 连接两个数据集
- 绘制一个散点图
- 选择模型（线性回归模型）， 开始训练
- 开始预测

透视表: 行列交叉

- pandas 函数 pivot()
- pd.merge(left=, right=, left_index, right_index=)
- model = sklearn.linear_model.LinearRegression() 线性回归
- model = sklearn.neighbors.KNeighborRegression(n_neighbors=3) K-近
- model.fit(X, Y)
- model.predict(newX)

步骤：

- 准备训练数据
- 选择模型
- 开始训练
- 预测

08. 机器学习的主要挑战

- 训练数据不足
- 训练数据没有代表性
- 数据质量差
- 无关特征
- 训练数据过度拟合
- 训练数据拟合不足

09. 准备训练数据

- 预测房价
- import os, tarfile, urllib

10. 查看和可视化数据集

- pandas as pd
- pd.read_csv()
- 分组函数 value_counts()

11. 准备训练集和测试集

- 测试比例
- np.ramdom.permutation((len(data))) 随机取数据集
- 获取测试集和训练集
- return data.iloc[train_set, test_set]

12. 用更完美的方式获取训练集和测试集

因为每次产生的数据不一样

- 1，保存测试集和训练集数据
- 2， 将随机种子固定 np.random.seed(315)
- 3， from zlib import crc32

13. 用sklearn API 产生训练集和测试集

- from sklearn.model_selection import train_test_split
- train_test_split(data, test_size=0.2, random_state = 315)

14. 分层抽样

- from sklearn.model_selection import StratifiedShuffleSplit
- 符合整体的特征比例
- 随机抽样

15. 通过可视化地理数据寻找模式

- 加上透明度
- 颜色，半径

16. 用两种办法检测属性之间的相关度

- 正相关
- 负相关
- 1，相关系数
- 2， pandas的 scatter_matrix 函数
- 3， corr (皮尔逊相关系数)

17. 为房屋数据添加新属性，并计算与房屋均价的相关度

- 清除无效的数据
- 选取属性相关度高的属性

18. 清理数据： 用转换器填充缺失值

- 数据清理，填补缺失值
- from sklearn,impute import SimpleImputer
- 平均值（mean），中间值(median)，众数(most_frequent)，常量(constant)
- fit
- transform
- fit_transform

19. 将文本类型属性转为数值

- one-hot 编码 独热编码
- from sklearn.preprocessing import OneHotEncoder
- 稀疏矩阵 SciPy
- from sklearn.preprocessing import lable_binarize 二值化

20. 自定义转换器

- from sklearn.base import BaseEstimator, TransformerMixin

21. 数据转换通道

- from sklearn.pipeline import Pipeline

- from sklearn.preprocessing import StandardScaler
- 多个转换器， 按照顺序执行
- FeatureUnion 数据转换 - 稀疏矩阵

22. 选择，训练模型以及预测房价

- 特征标签
- 选择模型（线性回归模型）
- 训练数据
- 选择模型（决策树）
- from skliearn.tree import DecisionTreeRegressor

23. 评估模型的性能

- 回归算法的模型评估方法
- 均方根误差（RMSE）root-mean-square-error
- 平均绝对误差（MAE） mean absolute error
- from sklearn,metrics import mean_squared_error
- from sklearn.metrics import mean_solute_error

24. 用交叉验证评估和选择模型

- 大数集划分为小的数据集
- cross-validation
- from sklearn.model_selection import cross_val_score
- 效用函数

25. 项目概述

- 获取图像数据
- 读取图像数据
- 训练二元分类器
- 精度和召回率
- ROC曲线
- 多类别分类器
- 多标签分类器
- 多输出分类

26. 使用sklearn 内置的图像数据

- from sklearn import datasets
- digits = datasets.load_digits()
- reshape() 一维数组变为二维
- plt.imshow()

27. 使用fetch_mldata 函数获取MNIST图像数据集

- fetch_mldata 7万张数据
- from sklearn,datasets import fectch_mldata
- .mat

28. 直接读取mat 格式的MNIST 图像数据集

- 读取mat 文件, SciPy 处理科学计算
- import scipy.io as sio
- sio.loadmat('xxx.mat')

29. 将多张图像合成一个图像

- 多张图片合成一张图

30. 对数字图像进行二元分类

- 训练二元分类器
- from sklearn,linear_model SGDClassifier
- sgd.fit(X, Y)
- sgd.predict(X_test)

31. 使用K-fold 交叉验证法评估分类器模型的性能

- from sklearn.model_selection import cross_val_score
- K-fold 评估不准确

32. 使用混淆矩阵评估分类器模型的性能

- 真正类 假负类
- 真负类 假正类
- from skliearn.model_selection import cross_val_predict
- from sklearn.metrics import confusion_matrix

33. 用精度，召回率和F1分数评估分类模型

- 精度
- 召回率
- F1分数
- from sklearn.metrics import percision_score, recall _score

34. 调整阈值得到不同的精度和召回率

- 精度和召回率的 相交点
- 精度和召回率权衡
- threshold = 0， 阈值
- from sklearn.metrics import precision_recall_curve
- 显示中文
- 显示负号

35. ROC 曲线与模型评估

- ROC （Receiver Operating Characteristic） 曲线，接收者操作特征曲线
- TPR（真正率） FPR (假正率)
- 计算面积
- from sklearn.metrics import roc_curve
- from sklearn.metrcs import roc_auc_score
- AUC 接近1越好

36. 比较随机森林分类器和梯度下降分类器的ROC曲线

- 随机森林和SGD 分类器比较
- predict_proba
- 预测概率
- 逻辑回归 - LogisticRegression()

37. 多类别分类器

- 支持多类别分类，随机森林，朴素贝叶斯 分类器
- 组合多个二元分类，不断调用多次，建立多个检测器
- OvA 一对多策略
- OvO N *(N-1) 个检测器
- 支持向量机分类器 OvO
- from sklearn.multiclass import OneVOneClassifier

38. 通过对特征值进行转换提高分类效果

- 均值为0，方差为1
- StandardScaler

39. 通过分析错误类型改进分类模型

- 错误分析
- 混淆矩阵，并可视化，plt.matshow()
- 按照比例
- 学会如何分析混淆矩阵

40. 多标签分类

- 二元分类和多元分类是单标签分类
- 属于哪一类数字，是否属于奇数、偶数
- 打上多个标签
- KNN 支持多标签分类

41. 去除图像噪音

- 属于分类问题
- 生成噪声

42. KNN- 实现原理

- 有监督的，分类和预测
- k 个点，最近的k个点
- 准确性高，

- 计算量大
- K值越大，模型的偏差越大，过拟合
- K值越少，方差越大，欠拟合
- 算法变种
    - 增加临近的权重
    - 圆心之内的

43. 用k-临近算法进行分类

- 产生训练集
- 计算距离 N * M
- 挑出距离最近的K个点
- 统计所有分类在k个点占有的席位
- KNeighborClassifier

44. 用k-临近算法进行预测

- 找到k个距离最近的点
- 计算k个点的平均值（x, y 值), 预测点的预测值
- 添加一些噪声
- 训练模型
- KNeighborsRegressor

45. 绘制拟合曲线

- 绘制拟合曲线

46. 准备训练数据和测试数据

- k临近来预测糖尿病
-

47. 比较和选择分类模型

- None

48. 训练模型与预测糖尿病

- None

49. 绘制学习曲线

- 学习曲线， 训练样本数量与学习效果（分数）的函数
- sklearn.model_selection import learning_curve
- ShuffleSplit

50. 选择相关特征与数据可视化

- 选择相关的特征
- from sklearn.feature_selection import SelectBest
- K 临近算法不是和糖尿病的预测

51. 线性回归都讲了什么

- KNN，随机森林， SGD，决策树
- scikit-learn API
- 阅读API的源码
- 超参数（模型外部的参数，例如k），模型参数(从模型中得到)
- 选择合适模型，模型参数，成本函数
- 计算（闭式 方程）
- 迭代方法（GD）- 批量梯度下降，小批量梯度下降。

52. 线性回归模型概述

- 均方差最小
- 线性回归预测模型 特征数量，特征值，模型参数
- 矩阵乘法
- RMSE 均方根误差
- MSE 均方误差

53. 使用标准方程进行线性回归拟合

- 标准方程 求 theta
- 成本函数（MSE），使最小
- 求最小theta
- 拟合，有噪音的线性方程

- 使用标准方程进行线性拟合
- 绘制模型的预测结果

54. 梯度下降算法原理

- 通用的优化算法
- 迭代方法，成本函数最小
- 学习率，是一个超参数
- 梯度为零，斜率为零，切线，导数
- 初始值，步长（学习率）
- 局部最小，全局最小，plateau
- 凸函数
- 特征值缩放和无缩放的梯度下降，圆和椭圆，保证每个特征值规模差不多
- SGDRegressor

55. 批量梯度下降

- 求成本函数的偏导数
- 下一个theta 的值 的等于上一个theta - 学习率*梯度向量
- 可以批量计算theta 值
- 迭代次数
- 基于梯度下降公式

56. 比较不同学习率的迭代效果

- 学习率的学习效果
- 绘制学习率
- 迭代结果发散

57. 随机梯度下降

- 随机梯度下降
- SGD
- 在迭代过程中让学习率变小
- 学习计划 t0 / (t+ t1)， 是的学习率变小
- 随机选取样本进行训练
- from sklearn.linear_model import SGDRegressor

58. 小批量梯度下降

- 随机选择一批

59. 比较四种线性回归算法

- 标准方程
- 批量梯度下降
- 随机梯度下降
- 小批量梯度下降

60. 用线性模型拟合非线性数据

- 线性拟合非线性
- 一元二次方程变成为二元一次方程
- 降阶，扩展X

61. 线性SVM分类

- 支持向量机
- 线性SVM分类，大间隔分类
- from sklearn 有一个超参数， C 越小街道越载
- from sklearn import datasets
- iris = datasets.load_iris()
- SVC

62. 添加特征使数据集线性可分离

- 添加特征使数据集线性可分
- 线性不可分变为线性可分
- 加一个平方特征

63. 基于多项式核的SVM分类器

- 添加多项式特征
- 基于多项式核- 高阶的
- from sklearn.preprocessing import StandardScaler
- from sklearn.svm import LinearSVC

- from sklearn.pipeline import Pipeline
- from sklearn.preprocessing import PolynomiaFeatures
- d=3, r=1, c=5

64. 高斯RNF的相似特征

- BRF (高斯经向基函数)
- -相似特征，使用RBF作为相似函数
- 地标（landmark），方程
- 一维坐标转为二维坐标

65. 基于高斯BRF核函数的SVM 分类器

- 基于高斯RBF核函数的SVM分类器
- 也支持线性和非线性分类

66. SVM 线性回归

- 支持线性和非线性回归
- 位于街道上