ML learning notes

Jack(Zhou Xinyu)

September 22, 2019

Contents

1 Basic Introduction of ML

1.1 General Steps of ML, supervised learning

监督学习有哪些步骤?
监督学习是使用已知正确答案的示例来训练网络，每组训练数据有一个明确的标识或结果。想象一下，我们可以训练一个网络，让其从照片库中（其中包含气球的照片）识别出气球的照片。以下就是我们在这个假设场景中所要采取的步骤。

1. 数据集的创建和分类 ** 首先，浏览你的照片（数据集），确定所有包含气球的照片，并对其进行标注。然后，将所有照片分为训练集和验证集。
2. 数据增强（Data Augmentation）当原始数据搜集和标注完毕，一般搜集的数据并不一定包含目标在各种扰动下的信息。数据的好坏对于机器学习模型的预测能力至关重要，因此一般会进行数据增强。对于图像数据来说，数据增强一般包括，图像旋转，平移，颜色变换，裁剪，仿射变换等。
3. 特征工程（Feature Engineering）一般来讲，特征工程包含特征提取和特征选择。常见的手工特征 (Hand-Crafted Feature) 有尺度不变特征变换 (Scale-Invariant Feature Transform, SIFT)，方向梯度直方图 (Histogram of Oriented Gradient, HOG) 等。由于手工特征是启发式的，其算法设计背后的出发点不同，将这些特征组合在一起的时候有可能会产生冲突，如何将组合特征的效能发挥出来，使原始数据在特征空间中的判别性最大化，就需要用到特征选择的方法。在深度学习方法大获成功之后，人们很大一部分不再关注特征工程本身。因为，最常用到的卷积神经网络 (Convolutional Neural Networks, CNNs) 本身就是一种特征提取和选择的引擎。研究者提出的不同的网络结构、正则化、归一化方法实际上就是深度学习背景下的特征工程。
4. 构建预测模型和损失将原始数据映射到特征空间之后，也就意味着我们得到了比较合理的输入。下一步就是构建合适的预测模型得到对应输入的输出。而如何保证模型的输出和输入标签的一致性，就需要构建模型预测和标签之间的损失函数，常见的损失函数 (Loss Function) 有交叉熵、均方差等。通过优化方法不断迭代，使模型从最初的初始化状态一步步变化为有预测能力的模型的过程，实际上就是学习的过程。
5. 训练选择合适的模型和超参数进行初始化，其中超参数比如支持向量机中核函数、误差项惩罚权重等。当模型初始化参数设定好后，将制作好的特征数据输入到模型，通过合适的优化方法不断缩小输出与标签之间的差距，当迭代过程到了截止条件，就可以得到训练好的模型。优化方法最常见的就是梯度下降法及其变种，使用梯度下降法的前提是优化目标函数对于模型是可导的。
6. 验证和模型选择训练完训练集图片后，需要进行模型测试。利用验证集来验证模型是否可以准确地挑选出含有气球在内的照片。在此过程中，通常会通过调整和模型相关的各种事物（超参数）来重复步骤 2 和 3，诸如里面有多少个节点，有多少层，使用怎样的激活函数和损失函数，如何在反向传播阶段积极有效地训练权值等等。
7. 测试及应用当有了一个准确的模型，就可以将该模型部署到你的应用程序中。你可以将预测功能发布为 API（Application Programming Interface, 应用程序编程接口）调用，并且你可以从软件中调用该 API，从而进行推理并给出相应的结果。

2 Supervised and unsupervised learning

Supervised learning: we have taught the computer about the ”true, false” result for each element in dataset
Unsupervised learning: the computers are not told about the ”true, false” result for each element in the dataset, it only knows that there exists a dataset

3 Linear regression and classification

Classification:logistic regression Linear regression: solve continuous case and output continuous result (e.g. the price prediction of house) classification: consider data as discrete, output a ”0 or 1” result
假设函数（hypothesis function）

损失函数，代价函数（loss function and cost function）

4 Gradient descent and feature scaling

** gradient descent: when the data set is large, approximately 10000*10000 matrix, it is more efficient but it needs to choose an appropriate å (学习速率), avoid the possibility of missing the global min or diverging ** technique of gradient descent: to make this process faster, if two feature's range is quite different, we need to change the scala, make two range relatively close to each other, say -1 to 1 (actual do not need to be precise, say -3 to 3, -1/3 to 1/3 is also good)

* 利用线代知识向量化 x_j^i j_{th} feature in i_{th} data set ** why we need bias unit in the formula? since in the hypothesis function, there is a θ_0, for convenience of computation, just assume $x_0 = 1$, then we have $\theta_0 * x_0$

* 高级优化（advanced optimization）公式求偏导，最小二乘法（Linear Algebra）当 data set 10000*10000，考虑梯度下降局部梯度下降，每次计算不用所有 data set ** Norm equation: when the data set is "small", it is fast and more efficient than the gradient descent, since it can compute the thetas but when the data set is quite large, it is time-consuming to computing, say 10000*10000 matrix's inverse. But it works pretty good when the data is relatively small

5 Under fitting and over fitting problem

6 Neural network

神经元，神经网络（neuron,neuron network）* 可利用线代知识向量化

激活函数（activation function/ actually just our sigmoid function）

向前传播（forward propagation）"非" 逻辑运算（negation）在预期得到非结果的变量前加一个很大的负的权重（weight/just theta in neural network）e.g. $-20 * x_1$ 对 x_1 的非运算二元分类（binary classification）0 or 1

多分类问题（in neural network）（multi-class classification）单分类的拓展，就是将输出层改成多个输出的

代价函数(in neural network) is the same as the regularization form of the cost function in logistic regression

反向传播算法（back propagation）用于计算导数项反向传播算法与 PHY1002 误差分析中的 error propagation 如出一辙，参考 PHY1002 attached 资料

$delta_j^l$ error term of node j in layer l

梯度检测（不是梯度下降），用于防止反向传播中可能出现的错误。参数 $\epsilon\ 10^{-4}$

随机初始化

7 More details about NNs

details about NNs

Table 1: Parameters of NNs

Value 1	**Value 2**	**Value 3**
α	β	γ
1	1110.1	a
2	10.1	b
3	23.113231	c

8 How to evaluate a model

* 训练集，测试集三七分，元素要随机取 model selection problem 训练集（training set），验证集（cross validation set）cv，测试集（test set）6,2,2 分

9 SVM choose parameter

$C(=\frac{1}{\lambda})$
Large C: lower bias, higher variance
Small C: Higher bias, lower variance
σ^2: large σ, feature f_i more smoothly, Higher bias, lower variance
σ^2: small σ, feature f_i less smoothly, lower bias, higher variance

10 How to use SVM

** No kernal(Linear kernal), when n is large, m is small, just use $\theta^T * x$ to avoid overfitting ** Gaussian kernal, we need to choose σ^2 $f_i = exp(-\frac{||x_i - l_i||^2}{2\sigma^2})$ perform feature scaling before using Guassian kernal ** logistic regression vs SVM $n >> m, sayn = 10000, m = 10...1000$ Use logistic refression or SVM without kernel (linear kernal) n small, m intermediate, say n =10-1000, m = 100-10000 Use SVM with Guassian kernal $n << m, sayn = 1-1000, m = 500000+$ add/create more features, then use logistic refression or SVM without kernel (linear kernal)

NN likely to work well for mst of the settings, but maybe slower to train

11 Unsupervised learning (clustering)

Feature: No label

1. Randomly generate n clustering centroids, cluster assign part, loop through every data point, according to the distance to the centroids and assign them to centroids
2. Move the class centroids according to the mean of the current clusters

input: K, Training set, $x^{(1)}, x^{(2)}, ...x^{(n)}, x^{(i)} \in R^{(n)}$ drop $x^(0) = 1$ convention
if no point in one cluster, then just move that centroid, or randomly reassign that centroid for seperated cluster, market segment

12 K-means algorithm

1. Optimization objective or say, (cost function) to minimize
2. Parameters
   (a) $C^{(i)}$:index of cluster $(1, 2, 3...K)$ to which example $x^{(i)}$ is currently assigned
   (b) μ_k: cluster centroid k, $\mu_k \in R^n$
   (c) $\mu_C(i)$: cluster centroid of cluster to which example $x^{(i)}$ has been assigned
3. $min_{c^{(1)},...,c^{(m)},\mu_1,...,\mu_K}\ J(c^{(1)}, ...c^{(m)}, \mu_1, ..., \mu_K)$
   $J(c^{(1)}, ...c^{(m)}, \mu_1, ..., \mu_K) = \frac{1}{m}\sum_{i=1}^{m} ||x^{(i)} - \mu_c(i)||^2$
   This is also called distortion function

4. Randomly initialize K cluster centroid $\mu_1, \mu_2, ..., \mu_K \in R^n$
   Repeat{
   step1: cluster assignment
   for $i = 1$ *to* m
   $C^{(i)}$:index of cluster centroid $(1, 2, 3...K)$ closest to $x^{(i)}$
   step2: move centroid
   for $k = 1$ *to* K
   μ_k:average(mean)of points assigned to cluster k}

13 Random Initialize of K-means

Randomly pick K training examples and set $\mu_1, ..., \mu_K$ equal to these K examples. Notice $K < m$

The converging result (local optima) could be different owing to different initializations. Solution, initialize it many times.

Say, K is relatively small 2-10, then multi-random initializations can make sure we get the global min

13.1 Choose num of clusters for K-means

Elbow Method

Choose according to practical usage