

线性回归于逻辑回归面试题

1. 简单介绍一下线性回归。

**线性回归(Linear Regression)是利用称为线性回归方程的最小平方函数对一个或多个自变量和因变量之间关系进行建模的一种回归分析。这种函数是一个或多个称为回归系数的模型参数的线性组合。只有一个自变量的情况称为简单回归,大于一个自变量情况的叫做多元回归。

- 线性: 两个变量之间的关系**是**一次函数关系的——图象**是直线**, 叫做线性。
- 非线性: 两个变量之间的关系**不是**一次函数关系的——图象**不是直线**, 叫做非线性。
- 回归: 人们在测量事物的时候因为客观条件所限, 求得的都是测量值, 而不是事物真实的值, 为了能够得到真实值, 无限次的进行测量, 最后通过这些测量数据计算**回归到真实值**, 这就是回归的由来。

线性回归就是利用的样本 $D = (x_i, y_i)$, $i = 1, 2, 3 \dots N$, x_i 是特征数据, 可能是一个, 也可能是多个, 通过有监督的学习, 学习到由 x 到 y 的映射 h , 利用该映射关系对未知的数据进行预估, 因为 y 为连续值, 所以是回归问题。

2. 线性回归的假设函数是什么形式?

线性回归的假设函数 (θ_0 表示截距项, $x_0 = 1$, 方便矩阵表达):

$$f(x) = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 \dots + \theta_n x_n = \theta^T X$$

其中 θ, X 都是列向量

3. 线性回归的损失函数是什么形式?

一般使用**最小二乘法**, 真实值 y_i , 预测值 $h_\theta(x)$, 则误差平方为 $(y_i - h_\theta(x_i))^2$ 找到合适的参数, 使得误差平方和最小

$$MSE: J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (y_i - h_\theta(x_i))^2$$

其中共有 m 个样本点, 乘以 $1/2$ 是为了方便计算。

参考资料: <https://www.cnblogs.com/xym4869/p/11309134.html>

4. 简述岭回归与Lasso回归以及使用场景。

- 目的:
 - 解决线性回归出现的过拟合的情况。
 - 解决在通过正规方程方法求解 θ 的过程中出现的 $X^T X$ 不可逆的情况。
- 本质:
 - 约束(限制)要优化的参数

这两种回归均通过在损失函数中引入**正则化项**来达到目的:

线性回归的损失函数:

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2$$

- **岭回归**
 - 损失函数:

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^n \theta_j^2$$

- **Lasso回归**

- 损失函数

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m \left(h_{\theta}(x^{(i)}) - y^{(i)} \right)^2 + \lambda \sum_{j=1}^n |\theta_j|$$

- 补充

- ElasticNet 回归**：线性回归 + L1正则化 + L2 正则化。

- ElasticNet在我们发现用Lasso回归太过(太多特征被稀疏为0),而岭回归也正则化的不够(回归系数衰减太慢)的时候,可以考虑使用ElasticNet回归来综合,得到比较好的结果。
 - 损失函数

$$J(\theta) = \frac{1}{2} \sum_i \left(y^{(i)} - \theta^T x^{(i)} \right)^2 + \lambda \left(\rho \sum_j |\theta_j| + (1 - \rho) \sum_j \theta_j^2 \right)$$

- LWR(局部加权)回归**:

- 局部加权线性回归是在线性回归的基础上对每一个测试样本(训练的时候就是每一个训练样本)在其已有的样本进行一个加权拟合, **权重的确定**可以通过一个核来计算,常用的有**高斯核**(离测试样本越近,权重越大,反之越小),这样对每一个测试样本就得到了不一样的权重向量,所以最后得出的拟合曲线不再是线性的了,这样就增加的模型的复杂度来更好的拟合非线性数据。
 - 损失函数

$$J(\theta) = \frac{1}{2} \sum_{i=1}^m w^{(i)} \left(h_{\theta}(x^{(i)}) - y^{(i)} \right)^2$$

5. 线性回归要求因变量服从正态分布吗?

线性回归的假设前提是噪声服从正态分布,即因变量服从正态分布。但实际上难以达到,因变量服从正态分布时模型拟合效果更好。

参考资料: http://www.julyedu.com/question/big/kp_id/23/ques_id/2914

6. 简单介绍一下逻辑回归

逻辑回归主要用来解决**分类**问题,线性回归的结果 Y 带入一个非线性变换的**Sigmoid函数**中,得到 $[0, 1]$ 之间取值范围的数 S , S 可以把它看成是一个概率值,如果我们设置概率阈值为0.5,那么 S 大于0.5可以看成是正样本,小于0.5看成是负样本,就可以进行分类了。

7. 简单介绍一下Sigmoid函数

sigmoid函数取值范围为 $(0, 1)$, 它可以将一个实数映射到 $(0, 1)$ 的区间,可以用来做二分类。函数公式如下:

$$S(t) = \frac{1}{1 + e^{-t}}$$

函数中 t 无论取什么值,其结果都在 $[0, 1]$ 的区间内,我们假设分类的**阈值**是0.5,那么超过0.5的归为1分类,低于0.5的归为0分类,阈值是可以自己设定的。

下来我们把线性回归的输出 $\theta^T X + b$ 带入 t 中就得到了逻辑回归的**假设函数**:

$$H(\theta, b) = \frac{1}{1 + e^{-(\theta^T X + b)}}$$

结果也可以理解为概率,换句话说概率大于0.5的属于1分类,概率小于0.5的属于0分类,这就达到了分类的目的。

8. 逻辑回归的损失函数是什么

逻辑回归的损失函数是**交叉熵损失函数**:

$$-\operatorname{cost}(\theta, y) = -\frac{1}{n} \sum_{i=1}^n \left[-y^{(i)} \ln h_{\theta}(x^{(i)}) - (1 - y^{(i)}) \ln (1 - h_{\theta}(x^{(i)})) \right]$$

两式合并得到概率分布表达式：

$$(P(y|x, \theta) = h_{\theta}(x)^y (1 - h_{\theta}(x))^{1-y})$$

对数似然函数最大化得到似然函数的代数表达式为：

$$L(\theta) = \prod_{i=1}^m (h_{\theta}(x^{(i)})^{y^{(i)}} (1 - h_{\theta}(x^{(i)}))^{1-y^{(i)}})$$

取反得到损失函数表达式——(负对数损失函数)：

$$J(\theta) = -\ln L(\theta) = -\sum_{i=1}^m (y^{(i)} \log(h_{\theta}(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)})))$$

9. 逻辑回归如何进行多分类？

多分类问题一般将二分类推广到多分类的方式有三种，一对一，一对多，多对多。

- 一对一：
 - 将 N 个类别两两配对，产生 $N(N-1)/2$ 个二分类任务，测试阶段新样本同时交给所有的分类器，最终结果通过投票产生。
- 一对多：
 - 每一次将一个例作为正例，其他的作为反例，训练 N 个分类器，测试时如果只有一个分类器预测为正类，则对应类别为最终结果，如果有多个，则一般选择置信度最大的。从分类器角度一对一更多，但是每一次都只用了 2 个类别，因此当类别数很多的时候一对一开销通常更小(只要训练复杂度高于 $O(N)$ 即可得到此结果)。
- 多对多：
 - 若干各类作为正类，若干个类作为反类。注意正反类必须特殊的设计。

10. 逻辑回归的优缺点

- 优点
 - LR 的可解释性强、可控度高、训练速度快
- 缺点
 - 对模型中自变量多重共线性较为敏感

例如两个高度相关自变量同时放入模型，可能导致较弱的自变量回归符号不符合预期，符号被扭转。需要利用因子分析或者变量聚类分析等手段来选择代表性的自变量，以减少候选变量之间的相关性；
 - 预测结果呈 S 型，因此从 $\log(odds)$ 向概率转化的过程是非线性的，在两端随着 $\log(odds)$ 值的变化，概率变化很小，边际值太小，slope 太小，而中间概率的变化很大，很敏感。导致很多区间的变量变化对目标概率的影响没有区分度，无法确定阈值。

11. 逻辑斯特回归为什么要对特征进行离散化。

- 逻辑回归属于广义线性模型，表达能力受限；单变量离散化为 N 个后，每个变量有单独的权重，相当于为模型引入了非线性，能够提升模型表达能力，加大拟合；离散特征的增加和减少都很容易，易于模型的快速迭代；
- 稀疏向量内积乘法运算速度快，计算结果方便存储，容易扩展；
- 方便交叉与特征组合：离散化后可以进行特征交叉，由 $M + N$ 个变量变为 $M * N$ 个变量，进一步引入非线性，提升表达能力；
- 简化模型：特征离散化以后，起到了简化了逻辑回归模型的作用，降低了模型过拟合的风险。

- 稳定性：特征离散化后，模型会更稳定，比如如果对用户年龄离散化，20-30作为一个区间，不会因为一个用户年龄长了一岁就变成一个完全不同的人；
- 离散化后的特征对异常数据有很强的鲁棒性：比如一个特征是年龄>30是1，否则0。如果特征没有离散化，一个异常数据“年龄300岁”会给模型造成很大的干扰。

参考资料：<https://blog.csdn.net/qq1195365047/article/details/88638809>

12. 线性回归与逻辑回归的区别

- 线性回归主要解决回归任务，逻辑回归主要解决分类问题。
- 线性回归的输出一半是连续的，逻辑回归的输出一般是离散的。
- 逻辑回归的输入是线性回归的输出，将Sigmoid函数作用于线性回归的输出得到输出结果。
- 线性回归的损失函数是MSE,逻辑回归中，采用的是负对数损失函数

参考资料：<https://blog.csdn.net/ddydavie/article/details/82668141>

13. 为什么逻辑回归比线性回归要好？

逻辑回归和线性回归首先都是广义的线性回归，其次模型的优化目标函数是最小二乘，而逻辑回归则是似然函数，逻辑回归在线性回归的基础上，在特征到结果的映射中加入了一层sigmoid函数（非线性）映射，即先把特征线性求和，然后使用sigmoid函数来预测。另外线性回归在整个实数域范围内进行预测，敏感度一致，而分类范围，需要在0,1间的一种回归模型，因而对于这类问题来说，逻辑回归的鲁棒性比线性回归的要好。

参考资料：<https://www.deeplearn.me/1788.html>

14. 逻辑回归有哪些应用

- CTR预估/推荐系统的learning to rank/各种分类场景。
- 某搜索引擎厂的广告CTR预估基线版是LR。
- 某电商搜索排序/广告CTR预估基线版是LR。
- 某电商的购物搭配推荐用了大量LR。
- 某现在一天广告赚1000w+的新闻app排序基线是LR。

15. 如果label={-1, +1}，给出LR的损失函数？

假设label={-1, +1},则

$$p(y = 1|x) = h_{\omega}(x)$$

$$p(y = -1|x) = 1 - h_{\omega}(x)$$

对于sigmoid函数，有以下特性，

$$h(-x) = 1 - h(x)$$

$$p(y|x) = h_{\omega}(yx)$$

同样，我们使用MLE作估计，

$$L(\omega) = \prod_{i=1}^m p(y_i|x_i; \omega) = \prod_{i=1}^m h_{\omega}(y_i x_i) = \prod_{i=1}^m \frac{1}{1 + e^{-y_i \omega x_i}}$$

对上式取对数及负值，得到损失为：

$$-\log L(\omega) = -\log \prod_{i=1}^m p(y_i|x_i; \omega) = -\sum_{i=1}^m \log p(y_i|x_i; \omega) = -\sum_{i=1}^m \log \frac{1}{1 + e^{-y_i \omega x_i}} = \sum_{i=1}^m \log(1 + e^{-y_i \omega x_i})$$

即对于每一个样本，损失函数为：

$$L(\omega) = \log(1 + e^{-y_i \omega x_i})$$

16. 逻辑回归在训练的过程当中，如果有很多的特征高度相关或者说有一个特征重复了100遍，会造成怎样的影响？

如果在损失函数最终收敛的情况下，有很多特征高度相关也不会影响分类器的效果。对特征本身来说的话，假设只有一个特征，在不考虑采样的情况下，你现在将它重复100遍。训练以后完以后，数据还是这么多，但是这个特征本身重复了100遍，实质上将原来的特征分成了100份，每一个特征都是原来特征权重值的百分之一。如果在随机采样的情况下，其实训练收敛完以后，还是可以认为这100个特征和原来那一个特征扮演的效果一样，只是可能中间很多特征的值正负相消了。

17. LR为什么使用sigmoid函数？

- 线性模型的输出都是在 $[-\infty, +\infty]$ 之间的，而Sigmoid能够把它映射到 $[0, 1]$ 之间。正好这个是概率的范围。
- Sigmoid是连续光滑的。
- 根据Sigmoid函数，最后推导下来逻辑回归其实就是最大熵模型，根据最大似然估计得到的模型的损失函数就是logloss。这让整个逻辑回归都有理可据。
- Sigmoid也让逻辑回归的损失函数成为凸函数，这也是很好的性质。
- 逻辑回归的损失函数是二元分类的良好代理函数，这个也是Sigmoid的功劳。

参考资料：http://sofasofa.io/forum_main_post.php?postid=1004244

18. LR如何进行并行计算？

参考资料：http://blog.sina.com.cn/s/blog_6cb8e53d0101oetv.html

19. LR和SVM有什么不同吗

- LR和SVM都可以处理分类问题，且一般都用于处理线性二分类问题（在改进的情况下可以处理多分类问题）
- 两个方法都可以增加不同的正则化项，如 l_1 、 l_2 等等。所以在很多实验中，两种算法的结果是很接近的。区别：
- LR是参数模型，SVM是非参数模型。
- 从目标函数来看，区别在于逻辑回归采用的是交叉熵损失函数，SVM采用的是hinge loss，这两个损失函数的目的都是增加对分类影响较大的数据点的权重，减少与分类关系较小的数据点的权重。
- SVM的处理方法是只考虑support vectors，也就是和分类最相关的少数点，去学习分类器。而逻辑回归通过非线性映射，大大减小了离分类平面较远的点的权重，相对提升了与分类最相关的数据点的权重。
- 逻辑回归相对来说模型更简单，好理解，特别是大规模线性分类时比较方便。而SVM的理解和优化相对来说复杂一些，SVM转化为对偶问题后，分类只需要计算与少数几个支持向量的距离，这个在进行复杂核函数计算时优势很明显，能够大大简化模型和计算。
- LR能做的SVM能做，但可能在准确率上有问题，SVM能做的LR有的做不了。