11 Important Model Evaluation Metrics for Machine Learning Everyone should know

BEGINNER LISTICLE MACHINE LEARNING PYTHON STATISTICS

Overview

- Evaluating a model is a core part of building an effective machine learning model
  一些评价指标 混淆矩阵 交叉验证 AUC-ROC曲线
- There are several evaluation metrics, like confusion matrix, cross-validation, AUC-ROC curve, etc.
- Different evaluation metrics are used for different kinds of problems

This article was originally published in February 2016 and updated in August 2019. with four new evaluation metrics.

Introduction

建设性的反馈原则

The idea of building machine learning models works on a constructive feedback principle. You build a model, get feedback from metrics, make improvements and continue until you achieve a desirable accuracy. Evaluation metrics explain the performance of a model. An important aspect of evaluation metrics is their capability to discriminate among model results.

度量 评估指标 区分模型结果的能力 评估指标的一个重要方面是它们能够区分模型结果的能力

健壮

I have seen plenty of analysts and aspiring data scientists not even bothering to check how robust their model is. Once they are finished building a model, they hurriedly map predicted values on unseen data. This is an incorrect approach.

他们匆忙地把预测值映射到看不见的数据上

Simply building a predictive model is not your motive. It's about creating and selecting a model which gives high accuracy on out of sample data. Hence, it is crucial to check the accuracy of your model prior to computing predicted values. 在计算预测值之前，检查模型的准确性是至关重要的。

In our industry, we consider different kinds of metrics to evaluate our models. The choice of metric
指标
度量的选择完全取决于模型的类型和模型的实施计划
completely depends on the type of model and the implementation plan of the model.

11个指标将帮助您评估模型的准确性
After you are finished building your model, these 11 metrics will help you in evaluating your model's accuracy. Considering the rising popularity and importance of cross-validation, I've also mentioned its
考虑到交叉验证的日益普及和重要性
principles in this article.

综合的；广泛的
And if you're starting out your machine learning journey, you should check out the comprehensive and popular ['Applied Machine Learning' cours]e which covers this concept in a lot of detail along with the various algorithms and components of machine learning.

Table of Contents

Warming up: Types of Predictive models

回归模型
When we talk about predictive models, we are talking either about a regression model (continuous output)
分类模型
or a classification model (nominal or binary output). The evaluation metrics used in each of these models are different.

在分类问题中
In classification problems, we use two types of algorithms (dependent on the kind of output it creates):

1. 输出类别 **Class output**: Algorithms like SVM and KNN create a class output. For instance, in a binary classification problem, the outputs will be either 0 or 1. However, today we have algorithms which can convert these class outputs to probability. But these algorithms are not well accepted by the statistics 统计界 community.
2. **Probability output**: Algorithms like 逻辑回归 Logistic Regression, Random Forest, 损失函数，算法 Gradient Boosting, Adaboost etc. give probability outputs. 转换概率输出 Converting probability outputs to class output is just a matter of creating a 阈值概率 threshold probability.

在回归问题中，我们在输出中没有这种不一致性。输出在本质上总是连续的，不需要进一步处理。
In regression problems, we do not have such inconsistencies in output. The output is always continuous in nature and requires no further treatment.

说明的；解释性的
Illustrative Example

有关分类模型评估指标的讨论

For a classification model evaluation metric discussion, I have used my predictions for the problem BCI challenge on Kaggle. The solution of the problem is out of the scope of our discussion here. However the final predictions on the training set have been used for this article. The predictions made for this problem were probability outputs which have been converted to class outputs assuming a threshold of 0.5.

混淆矩阵 类输出模型一起使用

1. Confusion Matrix

A confusion matrix is an N X N matrix, where N is the number of classes being predicted. For the problem in hand, we have N=2, and hence we get a 2 X 2 matrix. Here are a few definitions, you need to remember for a confusion matrix :

预测正确的总数所占的比例。

- **Accuracy** : the proportion of the total number of predictions that were correct.
- **Positive Predictive Value or Precision** : the proportion of positive cases that were correctly identified. 比例
- **Negative Predictive Value** : the proportion of negative cases that were correctly identified.
- 敏感度 **Sensitivity or Recall** : the proportion of actual positive cases which are correctly identified.
- 明确性；具体性 **Specificity** : the proportion of actual negative cases which are correctly identified.

Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		

Count of ID	Target			
Model	**1**	**0**	**Grand Total**	
1	3,834	639	4,473	85.7%
0	16	951	967	1.7%
Grand Total	**3,850**	**1,590**	**5,440**	
	99.6%	40.19%		88.0%

The accuracy for the problem in hand comes out to be 88%. As you can see from the above two tables, the Positive predictive Value is high, but negative predictive value is quite low. Same holds for Sensitivity and Specificity. This is primarily driven by the threshold 阈值 value we have chosen. If we decrease our threshold value, the two pairs of starkly 截然不同的 different numbers will come closer.

制药公司

In general we are concerned with one of the above defined metric. For instance, in a pharmaceutical company, they will be more concerned with minimal wrong positive diagnosis 最小错误阳性诊断. Hence, they will be more concerned about high Specificity. On the other hand an attrition model will be more concerned with Sensitivity. Confusion matrix are generally used only with class output models.

混淆矩阵通常仅与类输出模型一起使用。

2. F1 Score

精度

In the last section, we discussed precision and recall for classification problems and also highlighted the importance of choosing precision/recall 选择精度/召回率是我们的用例 basis our use case. What if for a use case, we are trying to get the best precision and recall at the same time? F1-Score is the harmonic mean of precision and recall values F1-Score是一个分类问题的准确率和召回率的调和平均值 for a classification problem. The formula for F1-Score is as follows:

$$F_1 = \left(\frac{\mathrm{recall}^{-1} + \mathrm{precision}^{-1}}{2}\right)^{-1} = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}.$$

为什么采用调和均值而不是算术均值

这是因为HM对极端值的惩罚更多

Now, an obvious question that comes to mind is why are taking a harmonic mean and not an arithmetic mean. This is because HM punishes extreme values more. Let us understand this with an example. We have a binary classification model with the following results:

精度 召回率

Precision: 0, Recall: 1

算术平均数；算术

Here, if we take the arithmetic mean, we get 0.5. It is clear that the above result comes from a dumb classifier which just ignores the input and just predicts one of the classes as output. Now, if we were to take HM, we will get 0 which is accurate as this model is useless for all purposes.

This seems simple. There are situations however for which a data scientist would like to give a percentage more importance/weight to either precision or recall. Altering the above expression a bit such that we can include an adjustable parameter beta for this purpose, we get:

$$F_\beta = (1 + \beta^2) \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{(\beta^2 \cdot \mathrm{precision}) + \mathrm{recall}}.$$

Fbeta衡量的是一个模型的有效性，该模型的用户对回忆的重视程度与回忆的精确度一样高

Fbeta measures the effectiveness of a model with respect to a user who attaches β times as much importance to recall as precision.

增益和升力图

3. Gain and Lift charts

检查概率的等级顺序

Gain and Lift chart are mainly concerned to check the rank ordering of the probabilities. Here are the steps to build a Lift/Gain chart:

Step 1 : Calculate probability for each observation

Step 2 : Rank these probabilities in decreasing order.

建立十分位数，每组几乎有10%的观测值

Step 3 : Build deciles with each group having almost 10% of the observations.

计算好（响应者），差（无响应）和总计的每十分位数的响应率。

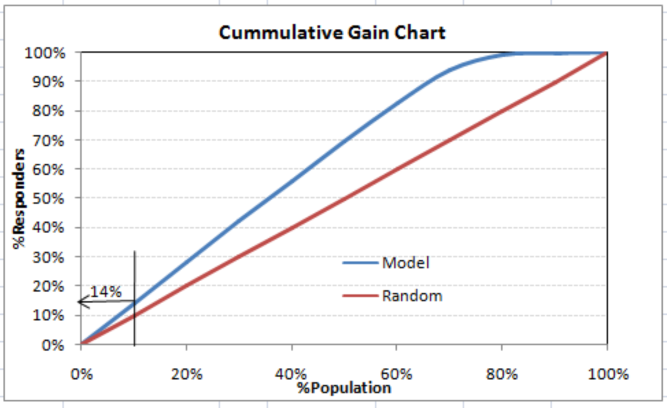
Step 4 : Calculate the response rate at each deciles for Good (Responders) ,Bad (Non-responders) and total.

You will get following table from which you need to plot Gain/Lift charts:

Lift/Gain	Column Labels			%Rights	%Wrongs	%Population	Cum %Right	Cum %Pop	Lift @decile	Total Lift
Row Labels	0	1	Grand Total	0%	0%	0%	0%	0%		
1		543	543	14%	0%	10%	14%	10%	141%	141%
2	2	542	544	14%	0%	10%	28%	20%	141%	141%
3	7	537	544	14%	0%	10%	42%	30%	139%	141%
4	15	529	544	14%	1%	10%	56%	40%	137%	140%
5	20	524	544	14%	1%	10%	69%	50%	136%	139%
6	42	502	544	13%	3%	10%	83%	60%	130%	138%
7	104	440	544	11%	7%	10%	94%	70%	114%	134%
8	345	199	544	5%	22%	10%	99%	80%	52%	124%
9	515	29	544	1%	32%	10%	100%	90%	8%	111%
10	540	5	545	0%	34%	10%	100%	100%	1%	100%
Grand Total	1590	3850	5440							

This is a very informative table. Cumulative Gain chart is the graph between Cumulative %Right and Cummulative %Population. For the case in hand here is the graph :

这是一个非常有用的表格。 累积增益图是累积权利百分比与累积人口百分比之间的关系图。 对于这里的情况，这是图形



你的模型在区分反应者和非反应者方面做得有多好

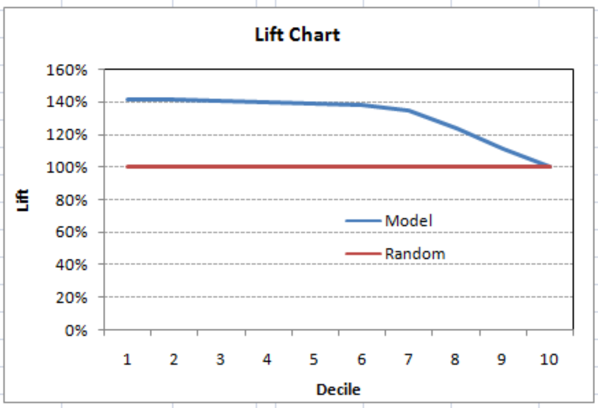
This graph tells you how well is your model segregating responders from non-responders. For example, the first decile however has 10% of the population, has 14% of responders. This means we have a 140% lift at first decile.

反映者 这意味着我们的十分位提升了140%

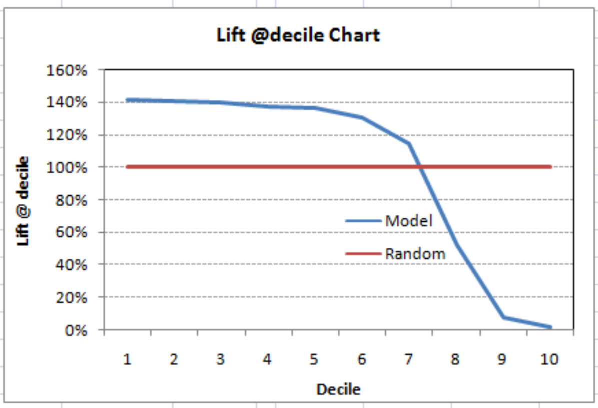
What is the maximum lift we could have reached in first decile? From the first table of this article, we know that the total number of responders are 3850. Also the first decile will contains 543 observations. Hence, the maximum lift at first decile could have been 543/3850 ~ 14.1%. Hence, we are quite close to perfection with this model.

升力曲线是总升力与人口百分比之间的曲线

Let's now plot the lift curve. Lift curve is the plot between total lift and %population. Note that for a random model, this always stays flat at 100%. Here is the plot for the case in hand :



You can also plot decile wise lift with decile number :



What does this graph tell you? It tells you that our model does well till the 7th decile. Post which every decile will be skewed towards non-responders. Any model with lift @ decile above 100% till minimum 3rd decile and maximum 7th decile is a good model. Else you might consider over sampling first.

每个十分位将偏向无响应者的帖子

Lift / Gain charts are widely used in campaign targeting problems. This tells us till which decile can we target customers for an specific campaign. Also, it tells you how much response do you expect from the new target base. 提升/收益图表广泛用于广告系列定位问题。 这可以告诉我们针对特定广告系列的目标客户。 此外，它还告诉您您希望新目标群体带来多少响应。

4. Kolomogorov Smirnov chart

更精确地

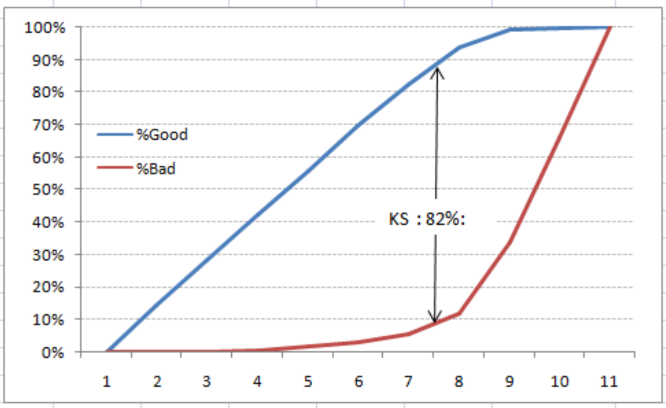
K-S or Kolmogorov-Smirnov chart measures performance of classification models. More accurately, K-S is a measure of the degree of separation between the positive and negative distributions. The K-S is 100, if the scores partition the population into two separate groups in which one group contains all the positives and the other all the negatives.

On the other hand, If the model cannot differentiate between positives and negatives, then it is as if the model selects cases randomly from the population. The K-S would be 0. In most classification models the K-S will fall between 0 and 100, and that the higher the value the better the model is at separating the positive from negative cases. 在大多数分类模型中，K-S将落在0到100之间，并且值越大，模型在区分阳性和阴性案例时就越好

For the case in hand, following is the table :

						Cummulative			
Lift/Gain Column				%Rights	%Wrongs	Cum %Rig	Cum %Wrong	K-S	
Row La	0	1	Grand Tot	0%	0%	0%	0%	0%	
1		543	543	14%	0%	14%	0%	14%	
2	2	542	544	14%	0%	28%	0%	28%	
3	7	537	544	14%	0%	42%	1%	42%	
4	15	529	544	14%	1%	56%	2%	54%	
5	20	524	544	14%	1%	69%	3%	67%	
6	42	502	544	13%	3%	83%	5%	77%	
7	104	440	544	11%	7%	94%	12%	82%	K-S
8	345	199	544	5%	22%	99%	34%	65%	
9	515	29	544	1%	32%	100%	66%	34%	
10	540	5	545	0%	34%	100%	100%	0%	
Grand Tot	1590	3850	5440						

We can also plot the %Cumulative Good and Bad to see the maximum separation. Following is a sample plot : 我们还可以绘制%累计好和坏以查看最大分离



The metrics covered till here are mostly used in classification problems. Till here, we learnt about confusion matrix, lift and gain chart and kolmogorov-smirnov chart. Let's proceed and learn few more important metrics.

到目前为止，所涉及的度量标准主要用于分类问题。到这里，我们学习了混淆矩阵，升力增益图和kolmogorov-smirnov图

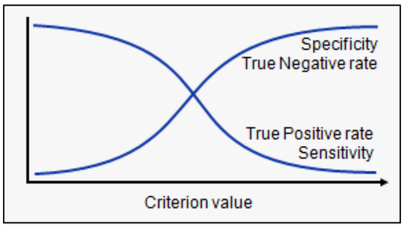
面积
5. Area Under the ROC curve (AUC – ROC)

This is again one of the popular metrics used in the industry. The biggest advantage of using ROC curve is that it is independent of the change in proportion of responders. This statement will get clearer in the following sections. 与反应者比例的变化无关

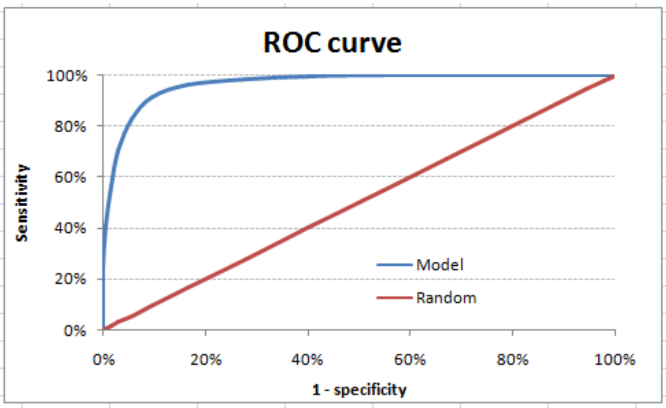
接收器工作特性
Let's first try to understand what is ROC (Receiver operating characteristic) curve. If we look at the confusion matrix below, we observe that for a probabilistic model, we get different value for each metric.

Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		

对于每种敏感性，我们得到不同的特异性
Hence, for each sensitivity, we get a different specificity.The two vary as follows:



敏感度 特异性
The ROC curve is the plot between sensitivity and (1- specificity). (1- specificity) is also known as false positive rate and sensitivity is also known as True Positive rate. Following is the ROC curve for the case in hand. （1-特异性）也称为假阳性率，灵敏度也称为真阳性率



Let's take an example of threshold = 0.5 (refer to confusion matrix). Here is the confusion matrix :

Target		
1	0	Grand
3,834	639	
16	951	
3,850	1,590	
99.6%	40.19%	

As you can see, the sensitivity at this threshold is 99.6% and the (1-specificity) is ~60%. This coordinate becomes on point in our ROC curve. To bring this curve down to a single number, we find the area under this curve (AUC).

因此AUC本身就是曲线下面积与总面积的比值

Note that the area of entire square is 1*1 = 1. Hence AUC itself is the ratio under the curve and the total area. For the case in hand, we get AUC ROC as 96.4%. Following are a few thumb rules:

- .90-1 = excellent (A)
- .80-.90 = good (B)
- .70-.80 = fair (C)
- .60-.70 = poor (D)
- .50-.60 = fail (F)

属于当前模型的出色范围。但这可能只是过度拟合，在这种情况下，及时和不定期的验证就变得非常重要。

We see that we fall under the excellent band for the current model. But this might simply be over-fitting. In such cases it becomes very important to to in-time and out-of-time validations.

Points to Remember:

对于以类作为输出的模型，将在ROC图中表示为单点

1. For a model which gives class as output, will be represented as a single point in ROC plot.

此类模型无法相互比较，因为需要根据单个度量标准而不是使用多个度量标准进行判断。 例如，具有参数（0.2,0.8）的模型和具有参数（0.8,0.2）的模型可能来自同一模型，因此不应直接比较这些指标

2. Such models cannot be compared with each other as the judgement needs to be taken on a single metric and not using multiple metrics. For instance, model with parameters (0.2,0.8) and model with parameter (0.8,0.2) can be coming out of the same model, hence these metrics should not be directly compared.

3. In case of probabilistic model, we were fortunate enough to get a single number which was AUC-ROC. But still, we need to look at the entire curve to make conclusive decisions. It is also possible that one model performs better in some region and other performs better in other.

在概率模型的情况下，我们很幸运地得到了一个单数，即AUC-ROC。 但是，我们仍然需要查看整个曲线以做出决定性的决定。 一个模型在某些区域中表现更好，而其他模型在其他区域中表现更好，这也是可能的。

Advantages of using ROC

提升取决于人群的总反应率。 因此，如果总体的响应率发生变化，则相同的模型将给出不同的提升图。 解决这个问题的方法可以是真实的升力图（在每个十分位上找到升力与理想模型升力的比率）。 但是这样的比例对企业来说几乎没有意义

Why should you use ROC and not metrics like lift curve?

Lift is dependent on total response rate of the population. Hence, if the response rate of the population changes, the same model will give a different lift chart. A solution to this concern can be true lift chart (finding the ratio of lift and perfect model lift at each decile). But such ratio rarely makes sense for the business.

ROC curve on the other hand is almost independent of the response rate. This is because it has the two axis coming out from columnar calculations of confusion matrix. The numerator and denominator of both x and y axis will change on similar scale in case of response rate shift.

另一方面，ROC曲线几乎与反应率无关。这是因为它有两个轴从圆柱计算混淆矩阵。当响应应率发生变化时，x轴和y轴的分子和分母都会发生相似的变化。

6. Log Loss

决定我们模型性能的预测概率

AUC ROC considers the predicted probabilities for determining our model's performance. However, there is an issue with AUC ROC, it only takes into account the order of probabilities and hence it does not take into account the model's capability to predict higher probability for samples more likely to be positive. In that case, we could us the log loss which is nothing but negative average of the log of corrected predicted probabilities for each instance. 然而，AUC ROC存在一个问题，它只考虑了概率的顺序，因此没有考虑模型的能力，预测更高概率的样本更有可能是正的。在这种情况下，我们可以得到对数损失也就是每个实例中修正后的预测概率的对数的负平均值。

$$-\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1-y_i)\cdot log(1-p(y_i))$$

正类的预测概率

- p(yi) is predicted probability of positive class
- 1-p(yi) is predicted probability of negative class 负类的预测概率
- yi = 1 for positive class and 0 for negative class (actual values)

让我们计算几个随机值的对数损失，以得到上述数学函数的要点

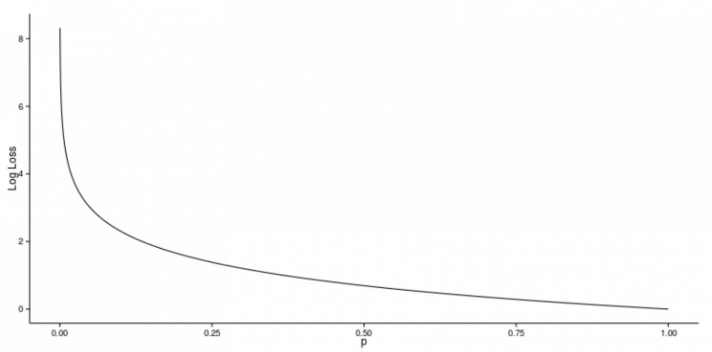
Let us calculate log loss for a few random values to get the gist of the above mathematical function:

Logloss(1, 0.1) = 2.303

Logloss(1, 0.5) = 0.693

Logloss(1, 0.9) = 0.105

If we plot this relationship, we will get a curve as follows:



明显的从向右平缓的向下斜率可以看出，随着预测概率的提高，对数损失逐渐减小。从相反的方向来看，当预测概率接近0时，对数损失会迅速增加

It's apparent from the gentle downward slope towards the right that the Log Loss gradually declines as the predicted probability improves. Moving in the opposite direction though, the Log Loss ramps up very rapidly as the predicted probability approaches 0.

So, lower the log loss, better the model. However, there is no absolute measure on a good log loss and it is use-case/application dependent. 它取决于用例/应用程序

尽管针对具有不同决策阈值的二进制分类计算了AUC，但对数损失实际上是将分类的“确定性”考虑在内。

Whereas the AUC is computed with regards to binary classification with a varying decision threshold, log loss actually takes "certainty" of classification into account.

基尼系数

7. Gini Coefficient

Gini coefficient is sometimes used in classification problems. Gini coefficient can be straigh away derived from the AUC ROC number. Gini is nothing but ratio between area between the ROC curve and the diagnol line & the area of the above triangle. Following is the formulae used :

Gini就是ROC曲线与诊断线之间的面积比，即上述三角形的面积

Gini = 2*AUC – 1

Gini above 60% is a good model. For the case in hand we get Gini as 92.7%.

和谐 – 不和谐的比例

8. Concordant – Discordant ratio

分类预测问题

This is again one of the most important metric for any classification predictions problem. To understand this let's assume we have 3 students who have some likelihood to pass this year. Following are our predictions :

A – 0.9

B – 0.5

C – 0.3

Now picture this. if we were to fetch pairs of two from these three student, how many pairs will we have? We will have 3 pairs : AB , BC, CA. Now, after the year ends we saw that A and C passed this year while B failed. No, we choose all the pairs where we will find one responder and other non-responder. How many such pairs do we have?

一对

We have two pairs AB and BC. Now for each of the 2 pairs, the concordant pair is where the probability of responder was higher than non-responder. Whereas discordant pair is where the vice-versa holds true. In case both the probabilities were equal, we say its a tie. Let's see what happens in our case :

一致的

AB – Concordant

不一致的

BC – Discordant

一致率超过60%被认为是一个很好的模型

Hence, we have 50% of concordant cases in this example. Concordant ratio of more than 60% is considered to be a good model. This metric generally is not used when deciding how many customer to target etc. It is primarily used to access the model's predictive power. For decisions like how many to target are again taken by KS / Lift charts.

它主要用于评价模型的预测能力

KS /提升图表再次决定要定位的目标

均方根误差

9. Root Mean Squared Error (RMSE)

回归问题

RMSE is the most popular evaluation metric used in regression problems. It follows an assumption that error are unbiased and follow a normal distribution. Here are the key points to consider on RMSE:

它遵循一个假设，即误差是无偏的并且服从正态分布

平方根''的力量使该指标能够显示大量偏差

1. The power of 'square root' empowers this metric to show large number deviations.

该度量的平方特性有助于提供更健壮的结果，从而防止正负误差值的抵消。换句话说，这个度量恰当地显示了误差项的可能大小

2. The 'squared' nature of this metric helps to deliver more robust results which prevents cancelling the positive and negative error values. In other words, this metric aptly displays the plausible magnitude of error term.

它避免了绝对误差值的使用，这在数学计算中是非常不可取的

3. It avoids the use of absolute error values which is highly undesirable in mathematical calculations.

当我们有更多样本时，使用RMSE重构误差分布被认为更可靠

4. When we have more samples, reconstructing the error distribution using RMSE is considered to be more reliable.

RMSE受异常值的影响很大。 因此，在使用此指标之前，请确保已从数据集中删除了异常值

5. RMSE is highly affected by outlier values. Hence, make sure you've removed outliers from your data set prior to using this metric.

与平均绝对误差相比，RMSE的权重更高，惩罚较大的误差

6. As compared to mean absolute error, RMSE gives higher weightage and punishes large errors.

RMSE metric is given by:

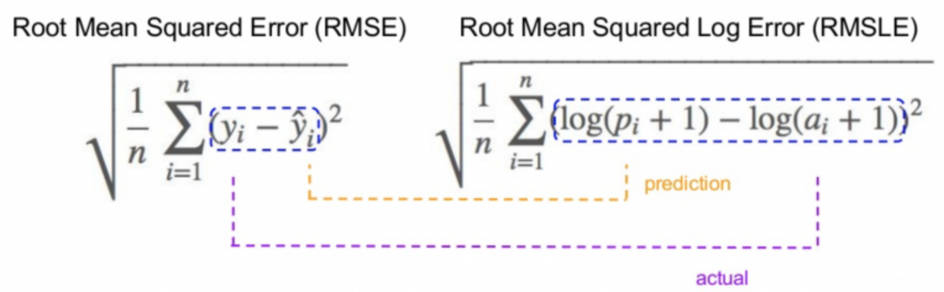
$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

where, N is Total Number of Observations.

均方根对数误差

10. Root Mean Squared Logarithmic Error

In case of Root mean squared logarithmic error, we take the log of the predictions and actual values. So
所以基本上，什么变化是我们正在测量的方差
basically, what changes are the variance that we are measuring. RMSLE is usually used when we don't want to penalize huge differences in the predicted and the actual values when both predicted and true values are huge numbers. 当我们不希望对预测值和实际值都是巨大的数值进行巨大的补偿时，通常使用RMSLE。



1. If both predicted and actual values are small: RMSE and RMSLE are same.
2. If either predicted or the actual value is big: RMSE > RMSLE
3. If both predicted and actual values are big: RMSE > RMSLE (RMSLE becomes almost negligible)

就几乎可以忽略不计

R平方/调整后的R平方

11. R-Squared/Adjusted R-Squared

我们了解到，当RMSE降低时，模型的性能将会提高。 但是，这些值本身并不直观

We learned that when the RMSE decreases, the model's performance will improve. But these values alone are not intuitive.

估计，判定

In the case of a classification problem, if the model has an accuracy of 0.8, we could gauge how good our model is against a random model, which has an accuracy of 0.5. So the random model can be treated as a
标杆，基准点
benchmark. But when we talk about the RMSE metrics, we do not *have* a benchmark to compare.

This is where we can use R-Squared metric. The formula for R-Squared is as follows:

$$R^2 = 1 - \frac{\text{MSE(model)}}{\text{MSE(baseline)}}$$

$$\frac{\text{MSE(model)}}{\text{MSE(baseline)}} \qquad \frac{\sum_{i=1}^{N}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{N}(\overline{y}_i-\hat{y}_i)^2}$$

预测值与实际值的均方误差

MSE(model): Mean Squared Error of the predictions against the actual values

相对于实际值的均值预测的均方误差

MSE(baseline): Mean Squared Error of mean prediction against the actual values

换句话说，我们的回归模型和一个非常简单的模型相比有多好这个简单的模型只是预测了从训练集合中得到的目标的平均值

In other words how good our regression model as compared to a very simple model that just predicts the mean value of target from the train set as predictions.

调整后R平方

Adjusted R-Squared

A model performing equal to baseline would give R-Squared as 0. Better the model, higher the r2 value. The best model with all correct predictions would give R-Squared as 1. However, on adding new features to the model, the R-Squared value either increases or remains the same. R-Squared does not penalize for adding features that add no value to the model. So an improved version over the R-Squared is the adjusted R-Squared. The formula for adjusted R-Squared is given by:

如果模型的性能等于基线，则R-Squared为0。模型越好，r2值越高。 具有所有正确预测的最佳模型将R-Squared设置为1。但是，在向模型添加新功能时，R-Squared值将增加或保持不变。 R-Squared不会因添加不会为模型增加任何价值的特征而受到不利影响。 因此，相对于R平方的改进版本是调整后的R平方。

$$\bar{R}^2 = 1-\left(1-R^2\right)\left[\frac{n-1}{n-(k+1)}\right]$$

k: number of features

n: number of samples

As you can see, this metric takes the number of features into account. When we add more features, the term in the denominator n-(k +1) decreases, so the whole expression increases.

If R-Squared does not increase, that means the feature added isn't valuable for our model. So overall we subtract a greater value from 1 and adjusted r2, in turn, would decrease.

Beyond these 11 metrics, there is another method to check the model performance. These 7 methods are statistically prominent in data science. But, with arrival of machine learning, we are now blessed with more robust methods of model selection. **Yes! I'm talking about Cross Validation**. 交叉验证

这7种方法在数据科学中具有统计学上的显著性 但是，随着机器学习的到来，我们现在有幸拥有了更健壮的模型选择方法

Though, cross validation isn't a really an evaluation metric which is used openly to communicate model accuracy. But, the result of cross validation provides good enough intuitive result to generalize the performance of a model. 虽然，交叉验证并不是真正用于评估模型准确性的评估指标。 但是，交叉验证的结果提供了足够好的直观结果，可以概括模型的性能

Let's now understand cross validation in detail.

12. Cross Validation

Let's first understand the importance of cross validation. Due to busy schedules, these days I don't get much time to participate in data science competitions. Long time back, I participated in TFI Competition on Kaggle. Without delving into my competition performance, I would like to show you the dissimilarity between my public and private leaderboard score. 我的公共排行榜和私人排行榜得分之间的差异

Here is an example of scoring on Kaggle!

For TFI competition, following were three of my solution and scores (Lesser the better) :

Submission	Files	Public Score	Private Score	Selected?
Mon, 04 May 2015 12:59:31 Edit description	submission_all_with_sai3.csv	1649776.86428	1809956.02878	☑
Mon, 04 May 2015 11:48:54 Edit description	submission_all_wit	1651071.47287	1802503.24607	☐
Mon, 13 Apr 2015 13:28:08 Edit description	submission_all.csv	1677138.71291	1795007.23155	☑

You will notice that the third entry which has the worst Public score turned to be the best model on Private ranking. There were more than 20 models above the "submission_all.csv", but I still chose "submission_all.csv" as my final entry (which really worked out well). What caused this phenomenon ? The 我的公共排行榜和私人排行榜的差异是过度拟合造成的 dissimilarity in my public and private leaderboard is caused by over-fitting.

高度复杂 产生噪音

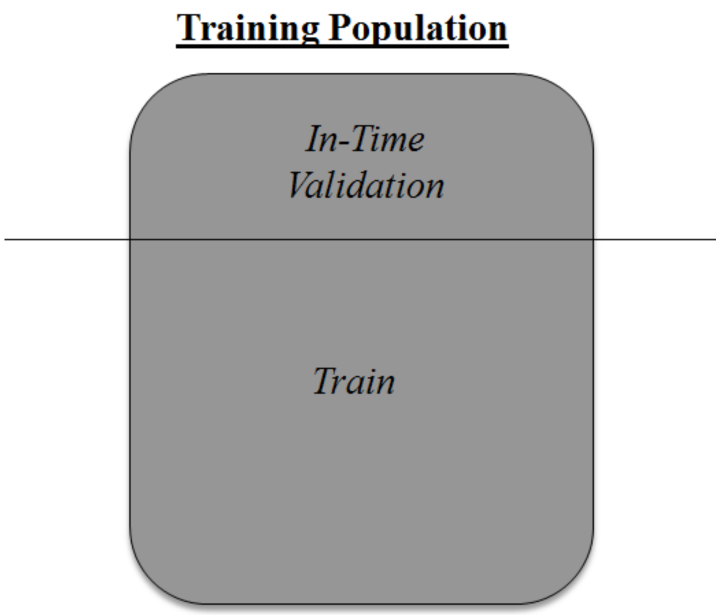
Over-fitting is nothing but when you model become highly complex that it starts capturing noise also. This 'noise' adds no value to model, but only inaccuracy. 不精确

In the following section, I will discuss how you can know if a solution is an over-fit or not before we actually know the test results.

The concept : Cross Validation

Cross Validation is one of the most important concepts in any type of data modelling. It simply says, try to leave a sample on which you do not train the model and test the model on this sample before finalizing the model.

交叉验证是任何类型的数据建模中最重要的概念之一。 简而言之，在最终确定模型之前，请尝试保留一个您不训练模型的样本并在该样本上测试模型



Above diagram shows how to validate model with in-time sample. We simply divide the population into 2 samples, and build model on one sample. Rest of the population is used for in-time validation.

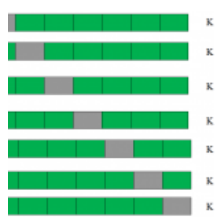
Could there be a negative side of the above approach?

I believe, a negative side of this approach is that we loose a good amount of data from training the model. Hence, the model is very high bias. And this won't give best estimate for the coefficients. So what's the next best option?

What if, we make a 50:50 split of training population and the train on first 50 and validate on rest 50. Then, we train on the other 50, test on first 50. This way we train the model on the entire population, however on 50% in one go. This reduces bias because of sample selection to some extent but gives a smaller sample to train the model on. This approach is known as 2-fold cross validation.

k-fold Cross validation

Let's extrapolate the last example to k-fold from 2-fold cross validation. Now, we will try to visualize how does a k-fold validation work.



This is a 7-fold cross validation.

Here's what goes on behind the scene : we divide the entire population into 7 equal samples. Now we train models on 6 samples (Green boxes) and validate on 1 sample (grey box). Then, at the second iteration we train the model with a different sample held as validation. In 7 iterations, we have basically built model on each sample and held each of them as validation. This is a way to reduce the selection bias and reduce the variance in prediction power. Once we have all the 7 models, we take average of the error terms to find which of the models is best.

How does this help to find best (non over-fit) model?

k折交叉验证广泛用于检查模型是否过拟合。 如果在每k次建模中的性能指标都彼此接近，并且指标的平均值最高。

k-fold cross validation is widely used to check whether a model is an overfit or not. If the performance metrics at each of the k times modelling are close to each other and the mean of metric is highest. In a Kaggle competition, you might rely more on the cross validation score and not on the Kaggle public score. This way you will be sure that the Public score is not just by chance.

How do we implement k-fold with any model?

Coding k-fold in R and Python are very similar. Here is how you code a k-fold in Python :

```
from sklearn import cross_validation model = RandomForestClassifier(n_estimators=100) #Simple K-Fold cross validation. 5 folds. #(Note: in older scikit-learn versions the "n_folds" argument is named "k".) cv = cross_validation.KFold(len(train), n_folds=5, indices=False) results = [] # "model" can be replaced by your model object # "Error_function" can be replaced by the error function of your analysis for traincv, testcv in cv:           probas = model.fit(train[traincv], target[traincv]).predict_proba(train[testcv])
results.append( Error_function ) #print out the mean of the cross-validated results print "Results: " + str( np.array(results).mean() )
```

But how do we choose k?

我们有一个权衡选择k

This is the tricky part. We have a trade off to choose k.

对于一个小k，我们有一个较高的选择偏差，但在性能的低方差

For a small k, we have a higher selection bias but low variance in the performances.

对于较大的k，我们有一个小的选择偏差，但在性能上有高的方差

For a large k, we have a small selection bias but high variance in the performances.

Think of extreme cases :

k = 2 : We have only 2 samples similar to our 50-50 example. Here we build model only on 50% of the population each time. But as the validation is a significant population, the variance of validation performance is minimal.

k = number of observations (n) : This is also known as "Leave one out". We have n samples and modelling repeated n number of times leaving only one observation out for cross validation. Hence, the selection bias is minimal but the variance of validation performance is very large.

k =观察数（n）：这也被称为“遗漏”。 我们有n个样本并进行了n次重复的建模，只留下了一个观察值供交叉验证。 因此，选择偏差很小，但验证性能的差异很大。

Generally a value of k = 10 is recommended for most purpose.

通常，建议将k = 10的值用于大多数用途。

End Notes

Measuring the performance on training sample is point less. And leaving a in-time validation batch aside is a waste of data. K-Fold gives us a way to use every singe datapoint which can reduce this selection bias to a good extent. Also, K-fold cross validation can be used with any modelling technique.

In addition, the metrics covered in this article are some of the most used metrics of evaluation in a classification and regression problems.

Which metric do you often use in classification and regression problem ? Have you used k-fold cross validation before for any kind of analysis? Did you see any significant benefits against using a batch validation? Do let us know your thoughts about this guide in the comments section below.

You want to apply your analytical skills and test your potential?
Then participate in our Hackathons and compete with Top Data Scientists from all over the world.

Article Url - https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/

Tavish Srivastava

Tavish Srivastava, co-founder and Chief Strategy Officer of Analytics Vidhya, is an IIT Madras graduate and a passionate data-science professional with 8+ years of diverse experience in markets including the US, India and Singapore, domains including Digital Acquisitions, Customer Servicing and Customer Management, and industry including Retail Banking, Credit Cards and Insurance. He is fascinated by the idea of artificial intelligence inspired by human intelligence and enjoys every discussion, theory or even movie related to this idea.