继2024年10月开始学习单细胞数据整合（integration）方法一来，对于单细胞数据（rds和h5ad文件）结构认识进一步深入，知道了一些整合方法（Python: scVI, harmony; R: RPCA, CCA, RLIGE, BBKNNR, harmony），当然整合方法远不只是如此，一些文章或者网址整理收录了不少整合方法，面向不同的数据内容采取不同的方法。方法很多，如何在众多的方法中选取最好最适合的方法呢？或者说尽可能的好的方法，方法终究是服务于目的即我们的科学问题，一个方法绝不能干扰或者致误科学问题。所以，言归正传，如何评价不同整合去批次方法进而获得高质量的数据整合？

第一步，我们应该阅读一些专业文档，去了解一下对于整合方法评价现有资源/尺度。我将从以下渠道获得相关知识内容: 微信公众号、浏览器、Google外网、PubMed

@整合去批次效果评价指标

生物保留性评价 (Biological Conservation)：整合后数据能否保持生物学上的一致性和信息。这部分的评估通过以下几个指标来完成：

平均轮廓宽度（ASW, Average Silhouette Width）：ASW 是一种衡量聚类效果的指标，用来评估每个细胞与其所属簇的相似度。细胞离自己所属簇越近，与其他簇的距离越远，ASW 就越高。ASW 评估了在整合后的空间中，生物学上相似的细胞是否被正确归类到相同的簇中。孤立标签得分（ASW, Isolated Label Score）：孤立标签得分也是基于ASW计算的，但重点是评估那些本应具有明显区分性的生物亚群，是否在整合后依然能够清晰分离。高的孤立标签得分意味着生物学上的不同亚群在整合后仍然是独立的。

cLISI：cLISI（cell Labeling In Silico Integration Score）是用来评估整合后生物保留性的一种指标。cLISI的高分值表明生物学相似的细胞在整合后的空间中聚集在一起，而非随机散布。

批次效应消除 (Batch Correction)

batch ASW：batch ASW 是基于 ASW 的一个衍生指标，用来专门评估批次效应。通过测量同一批次中的细胞是否过度聚类，batch ASW 可以评估批次效应是否有效地被消除。理想情况下，不同批次的细胞应均匀分布在各个簇中，而不会因为批次效应而分组。

iLISI：iLISI（Integration Labeling In Silico Integration Score）用于评估批次效应消除的效果。如果整合成功，不同批次的细胞会均匀混合，iLISI的分数越高，说明批次效应越低。

图连接性（Graph Connectivity）：这是用来评估细胞在整合后的邻近关系网络中的连通性。高的图连接性表示不同批次的细胞在整合后能够形成紧密的、连续的图结构，而不是分散的、孤立的。

kBET分数：kBET（k-nearest neighbor Batch Effect Test）是一个专门用于评估批次效应消除的指标。kBET测试邻近的细胞是否来自同一批次，如果不同批次的细胞能够混合均匀，kBET分数就会较高，表示批次效应得到了有效的消除。

簇同质性评估：为了确保在整合过程中，簇内的细胞仍然具有一致的生物学特性，研究使用了以下评估方法：

ROGUE指标：ROGUE（Relative Overlap of Gene Expression）是一种基于熵的指标，用来衡量簇内基因表达的一致性。ROGUE值越接近1，表示簇内的异质性越小，即簇内细胞的基因表达模式越一致，簇的纯度越高。ROGUE常用于评估簇是否需要进一步细分。（这个就是张院士课题组开发的，我也经常用，效果不错,可以看看我的结果）

SCCAF（Single Cell Clustering Assessment Framework）：SCCAF 是用来评估聚类准确性的一种工具。通过评估聚类模型的训练和预测能力，SCCAF可以衡量聚类结果是否反映了数据中的真实生物学特征。

<https://mp.weixin.qq.com/s/BA78Q6PO2FZ9CEbSnFymCg>

Batch correction

Average silhouette width (ASW): 平均轮廓宽度。这是一种用于评估聚类质量的指标，用于衡量细胞在不同批次中是否能够被正确地分开

Graph integration local inverse Simpson’s Index (graph iLISI): 图集成局部倒数Simpson指数。这似乎是一种用于比较不同图谱（可能是从不同批次收集的）之间差异的方法，可能是通过比较细胞类型或样本之间的相似性来完成的。

k-nearest-neighbor batch effect test (kBET): k最近邻批次效应检验。这是一种用于衡量数据集中不同批次之间是否存在批次效应的统计方法。

k-nearest-neighbor (kNN) graph connectivity: k最近邻图连通性。这可能涉及构建一个基于细胞之间相似性的图，以衡量不同批次之间的连接性。

Bio conservation

Isolated label scores: 孤立标签分数。用于评估罕见细胞身份标签的指标。Normalized Mutual Information (NMI): 归一化互信息。用于测量两个随机变量之间关联度的度量。在这里我们测量不同batch校正后细胞类型的一致性

Adjusted Rand Index (ARI): 调整兰德指数。一种用于比较两个数据分区的相似度的指标。Average silhouette width (ASW): 平均轮廓宽度。这是一种用于评估聚类质量的指标，用于衡量不同细胞类型中是否能够被正确地分开

graph cLISI: cLISI。可能是一种用于衡量标签保守性的统计方法，可能与细胞之间的相似性有关。

@数据整合的实现本质？<https://mp.weixin.qq.com/s/BA78Q6PO2FZ9CEbSnFymCg>

数据整合模型的类别在scRNA-seq中，消除批次效应往往由以下三个步骤构成：①降维；②建模并消除批次效应；③嵌入降维

主成分分析法PCA，运用PCA可以降低我们数据的信噪比，提取出细胞的差异信息。而建模并消除批次效应则是核心步骤，绝大多数的批次效应校正算法都包括了如何构建一个稳定的数学模型。我们按照模型的发展顺序，可以分为四类：全局模型，线性嵌入模型，基于图的模型，深度学习模型。

①全局模型: 源自bulk RNA-seq，将批次效应建模为所有细胞中存在的（加法/或乘法）效应。一个常见的例子是 ComBat

②线性嵌入模型: 是第一个单细胞特异性批量去除方法。这些方法通常使用奇异值分解 (SVD) 的变体来嵌入数据，然后在嵌入中跨批次查找相似单元的局部邻域，并使用它们以局部自适应（非线性）方式校正批次效应。常见的例子包括最近邻MNN，Seurat，Harmony，Scanorama，FastMNN等

③基于图的模型: 通常是运行速度最快的方法。使用最近邻域图来表示每个批次的数据。通过强制连接不同批次的细胞，然后修建细胞类型组成的差异的图的边缘，可以纠正批次效应。一个常见的例子是BBKNN

④深度学习模型: 大多数深度学习批次效应校正方法都基于自动编码器网络，并且要么在条件变分自动编码器（CVAE）中对批量协变量进行降维，要么在嵌入空间中拟合局部线性校正。

当消除样本的批次效应时，方法可能会过度校正并消除除批次效应之外的有意义的生物变异。因此，除了消除批次效应外，方法是否保留了潜在的生物学意义，也是我们需要去着重考虑的一个点。

对于主成分回归（PCR）指标，值的高低是否表示“越好”取决于具体的指标和上下文。在单细胞数据集成的评估中，PCR 指标通常用于评估批次效应的去除效果。具体来说，PCR 指标衡量的是批次信息在主成分中的解释力。以下是一些常见的解释：

PCR 指标解释

高 PCR 值：批次效应未去除：如果 PCR 值较高，表示批次信息在主成分中占据了较大的解释力，这意味着批次效应可能没有被有效去除。这通常是一个不好的信号，因为批次效应可能会干扰生物学信号的分析。示例：PCR 值为 0.8 表示批次信息解释了 80% 的主成分变异，这表明批次效应仍然很强。

低 PCR 值：批次效应去除效果好：如果 PCR 值较低，表示批次信息在主成分中的解释力较小，这意味着批次效应已经被有效去除。这通常是一个好的信号，因为去除批次效应后，生物学信号更容易被检测到。

示例：PCR 值为 0.1 表示批次信息仅解释了 10% 的主成分变异，这表明批次效应已经被有效去除。

Silhouette Batch 值（Batch ASW）是用于评估单细胞数据中批次效应的一个重要指标。它通过计算每个细胞在其所属批次内的轮廓系数（Silhouette Coefficient）来衡量批次之间的混合程度。具体来说，Batch ASW 的值范围在 [0, 1] 之间，其意义如下：

值接近 0：表示批次之间有很好的重叠，即批次效应被有效去除。这通常是一个好的信号，表明数据集成效果良好。

值接近 1：表示批次之间分离度很高，即批次效应仍然存在。这通常是一个不好的信号，表明数据集成效果不佳。

Graph Connectivity 值的意义

在单细胞数据分析中，Graph Connectivity（图连通性）是一个用于评估数据中细胞聚类质量的指标。它衡量的是聚类结果中细胞之间的连接程度，具体来说，它评估的是每个细胞在其所属聚类中与其他细胞的连接情况。Graph Connectivity 值的范围通常在 [0, 1] 之间，其意义如下：

值接近 1：表示聚类中的细胞之间有很好的连接性，即聚类结果中的细胞形成了紧密的网络。这通常是一个好的信号，表明聚类结果具有较高的内部一致性。

值接近 0：表示聚类中的细胞之间连接性较差，即聚类结果中的细胞没有形成紧密的网络。这通常是一个不好的信号，表明聚类结果可能不够理想。

在单细胞数据分析中，使用 Leiden 聚类算法时，resolution 参数控制聚类的粒度。resolution 参数值越高，生成的聚类越细（即聚类数量越多），反之则越粗（即聚类数量越少）。nmi（Normalized Mutual Information，归一化互信息）是一个用于评估聚类结果与真实标签一致性的指标，其值范围在 [0, 1] 之间，值越接近 1 表示聚类结果与真实标签的一致性越高。

关于如何提升代码执行效率，节约资源？

对于脚本而言，取消掉那些无用的变量，对于无必要的处理进行取消，对于耗时比较长的参数进行传参选择进行执行

对于workflow而言，使用

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Batch Correction | PCR(Principal component regression) | 主成分回归指标 | [0, 1] | 小 |
| bASW(batch Average Silhouette Width) | 所属批次内的轮廓系数 | [0, 1] | 小 |
| kBET(k-Nearest Neighbor Batch Effect Test) | k最近邻批次效应检验 | [0, 1] | 小 |
| GC(Graph Connectivity) | 图连通性 | [0, 1] | 大 |
| Biological Conservation | NMI(Normalized Mutual Information) | 归一化互信息 | [0, 1] | 大 |
| ARI(Adjusted Rand Index) | 调整兰德指数 | [0, 1] | 大 |
| cASW(cell Average Silhouette Width) | 细胞轮廓系数。 | [-1, 1] | 大 |
| iso\_F1(Isolated label F1) | 孤立标签在聚类结果中的分离程度 | [0, 1] | 大 |

2025/1/16

batch\_key是不同的样本；label\_key是给的生物学标签、注意是生物学标签，就是有注释信息的细胞信息，而不是数字聚类结果。

回过头来看一下scIB评估的指标，biological conservation和batch correction，注意到生物学性状保留如何去评估，百思不得其解，冰冷的数据怎么表示其生物学性状，原来关键就在于这个label\_key，label\_key赋予了各个细胞属于那种细胞的特征，在这里我们默认完全信任这个注释，即这就是这个细胞正确的注释，暂且不考虑错误注释。那么基于这样的考虑，就有了在整合前生物学性状的情况。

通过整合之后，再次pca得到降维的信息，这是数据信息，也是新的位置信息，通过对比之前生物学性状信息，最后得出其计算。

按照这个思维，那么label\_key肯定不是降维之后聚类得到的key，所以今天使用到的cell\_type肯定是错的，这也是为什么基于LIGER和harmony为first文件，其结果显示巨大的差异，而first文件总是最好得分的原因就在于此。

关于pca维度的统一问题，或者说聚类前统一的问题。在聚类的过程，根据示例数据，要求hvg基因是一样，降维也是统一，这样也有好处吧，就是差异完全由于方法本身决定了，当然最后确定最好的方法，最好的降维pca维度应该不断优化选择。

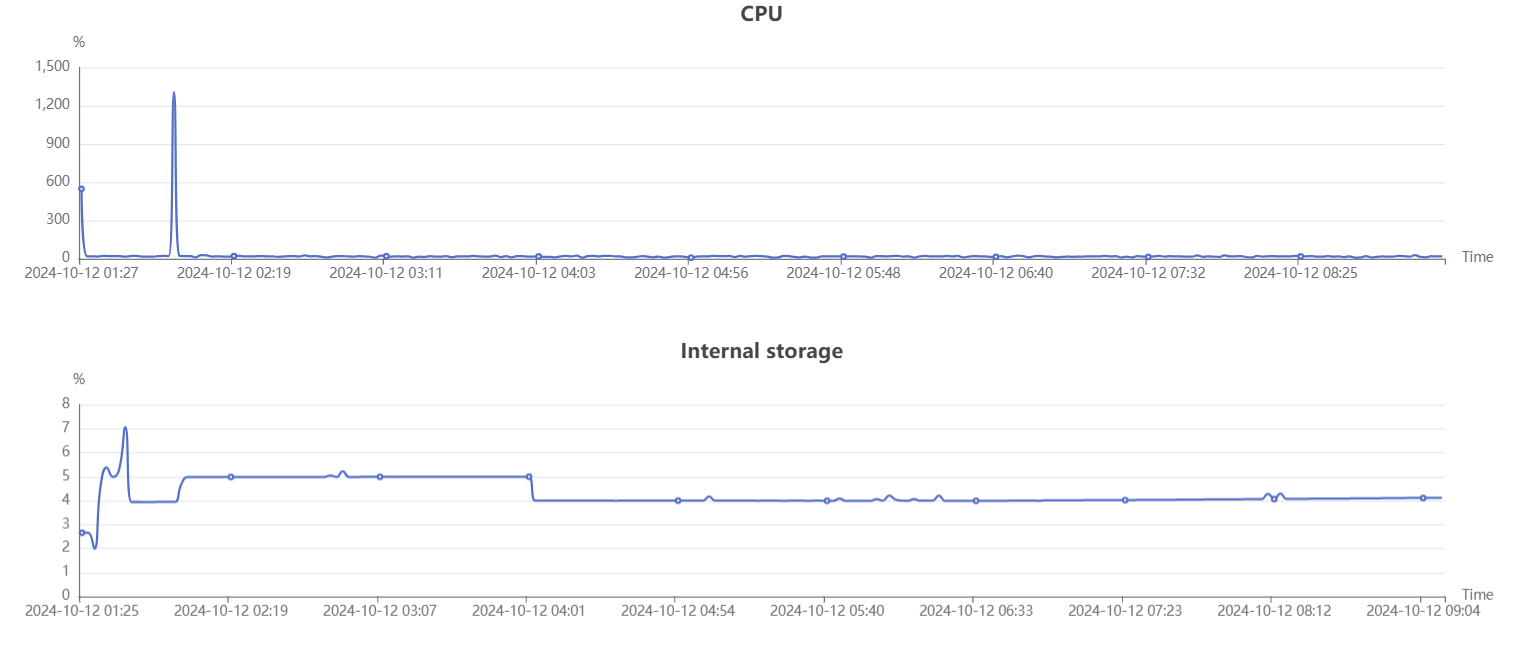
所以说，这也是为什么没有每个整合文件都用其obs的cell\_type，因为这是生物学性状，在整合前就是完全确定的。

那对于现有测试数据如何去做个事情呢？首先我们的数据的生物学性状没有统一，之前已经验证过了，batch和cell\_type完全独立，这一点要解决，必须统一一下cell\_type的命名。而是对于整合前hvg和pca的统一，则会个应该不是特别严谨，可以忽略不计。

最后总结。生物性状的信息是在一开始就确定了，这样具有生物学性状的数据如何获得了，一般多从公共数据库获得，做公共数据库的整合，可以很好的用起来。另外就是对于多个批次，各个批次先注释一次，默认完全正确，然后在整合，看情况。这样也可以起到这样的流程。所以先整合再注释，还是先注释后整合，都是基于个性分析和实际需要进行的。对于scIB而言，需要先注释后整合，要不然没有生物学性状的label。

2025/1/17

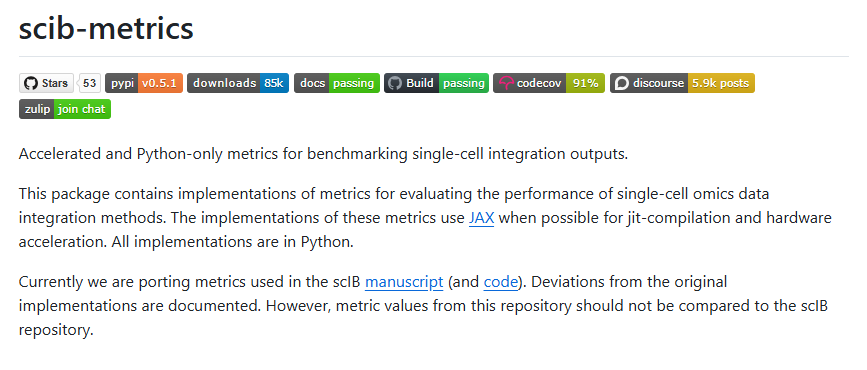
scIB学习感悟:

接受benchmarker测试的工作应该要追溯到去年10月，最初版本的scIB-metrics-test是针对单一整合方法处理得到的h5ad文件进行测试，其中测试的指标有8个，四个biological conservation，4个batch correction。当时测试的输入文件是Cer的整合文件，输入的label\_key就是整合时聚类得到的cluster\_key。当时的测试工作就非常困难，我使用到了6种整合方法（算少的了），6个处理文件跑scIB非常的慢，而且莫名资源消耗的特别多，资源利用率非常低，跑一个文件差不多一天，那时候一直好奇为什么会这么久，但未找到原因。

当时为了解决资源的问题，就是让资源利用率更高，我对不同的整合方法测试进行了个性化的资源配置，这花了很多的时间，但仍然未解决其根本运算慢的问题。

到25年1月再次工作，与老师聊到scIB\_metrics测试工作，老师的文件有18个h5ad文件，运行了整整一个月仍然无结果，这样肯定是有问题的，资源配置也比较高，16个CPU，200G的memory，这么高的消耗仍然无法满足需求，肯定是script和code有问题。带着这个问题，我先看了一下workflow流程，workflow的task通过scatter进行外围循环是并行计算（看中文文档知），所以我想着还是走老路线，优化workflow的流程。另外就是对于脚本进行优化，尽可能的精简codes，设置可选参数来确定要用那些指标。讲到评价指标，其实我一直不清楚其生物学意义，已经如何解答其运算结果，只感觉是一群从0到1的数字罢了。所以我去阅读了相关的中文文档，基本了解了指标意义，整理了【指标】。

整体而言我对之前的代码是不满意的，运算时间超级长，代码冗余，资源利用效率低。这样的方式计算真的对吗，于是我去GitHub寻找其tutorial，最后找到两个文档【】，一开始想的仍然是用老代码，是不是参数的原因，想着改改参数，但效率仍然不如意，一天测试一次都为难，整体工作效率极低。后面我注意到作者有打包指标的函数，将多个指标用一个函数计算，这样似乎好一些了（代码好看点）。本着试试的心理，发现import scib不等于import scib\_metrics，原来这是两个独立的package。



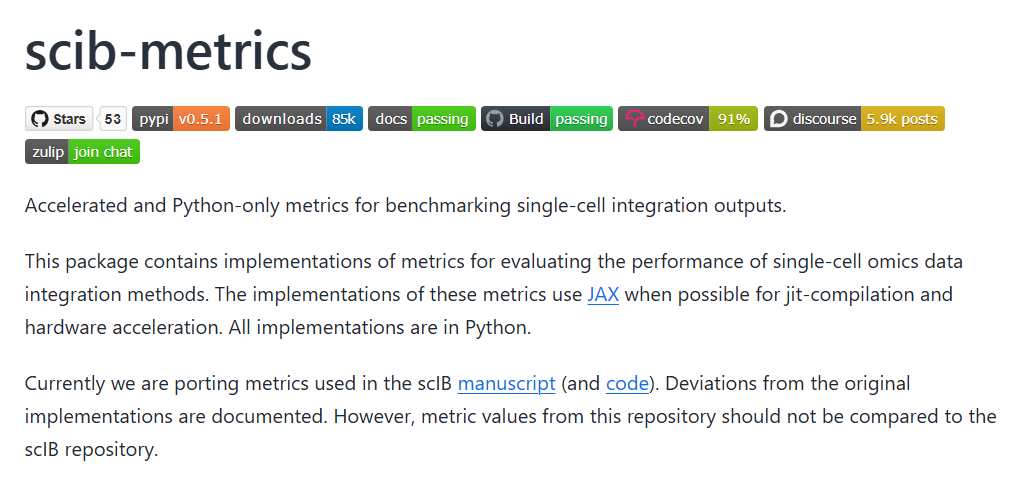
pip install scib-metrics

# Install the latest development version

pip install git+https://github.com/yoseflab/scib-metrics.git@main

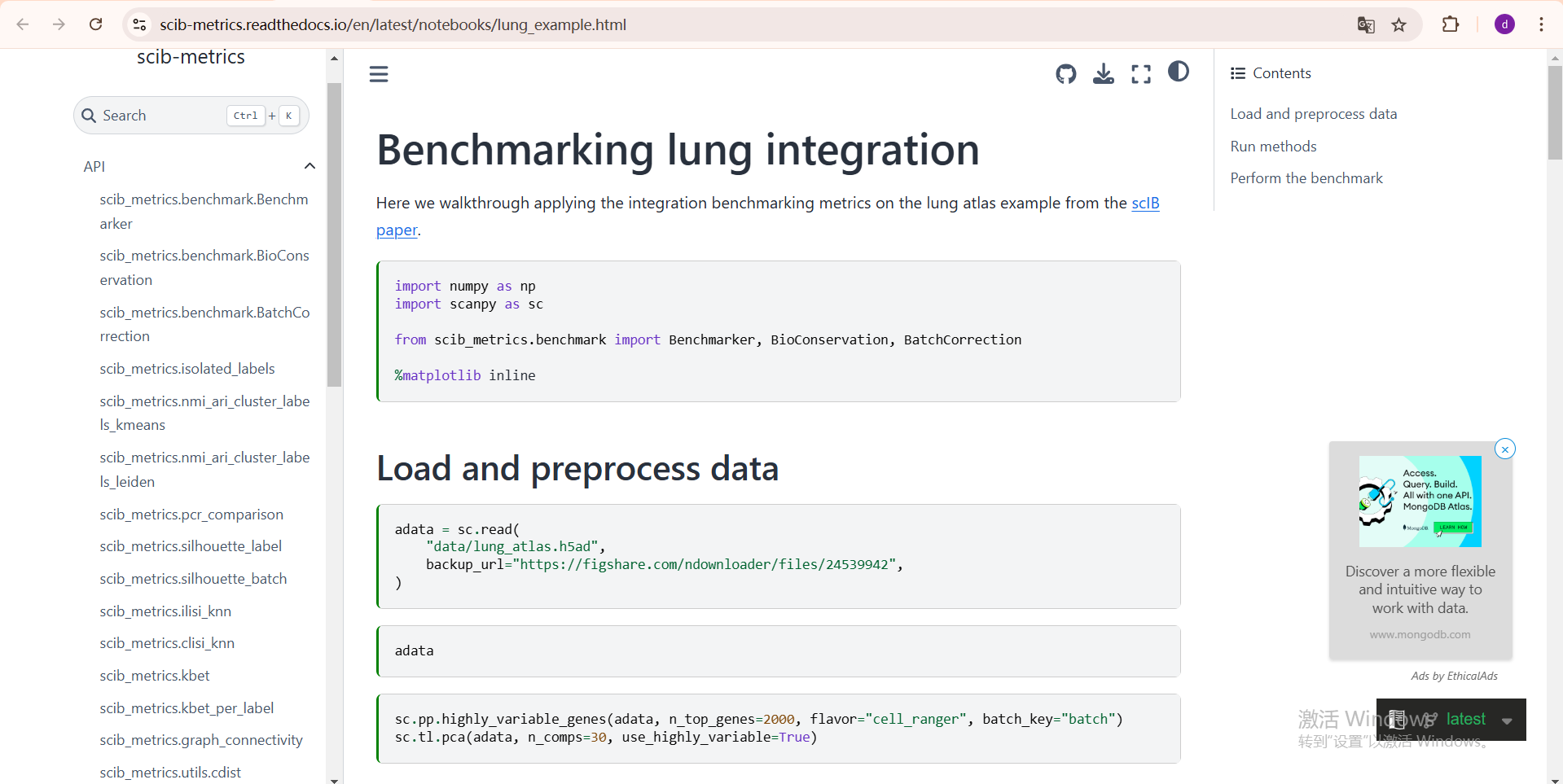
到现在还不太清楚这两者的关系，但显然scib-metrics提供了更加便易的方法，而且我注意到它提到了jax加速，后面我根据这个方法又了解了什么是jax和CPU、GPU资源的配置，最终jax的方法没有得到实现，嗯，因为这个似乎不能简单的使用加速，还要适配于计算逻辑，后面再看怎么搞。

后面就是阅读scib-metrics的文档学习<https://scib-metrics.readthedocs.io/en/latest/index.html#>

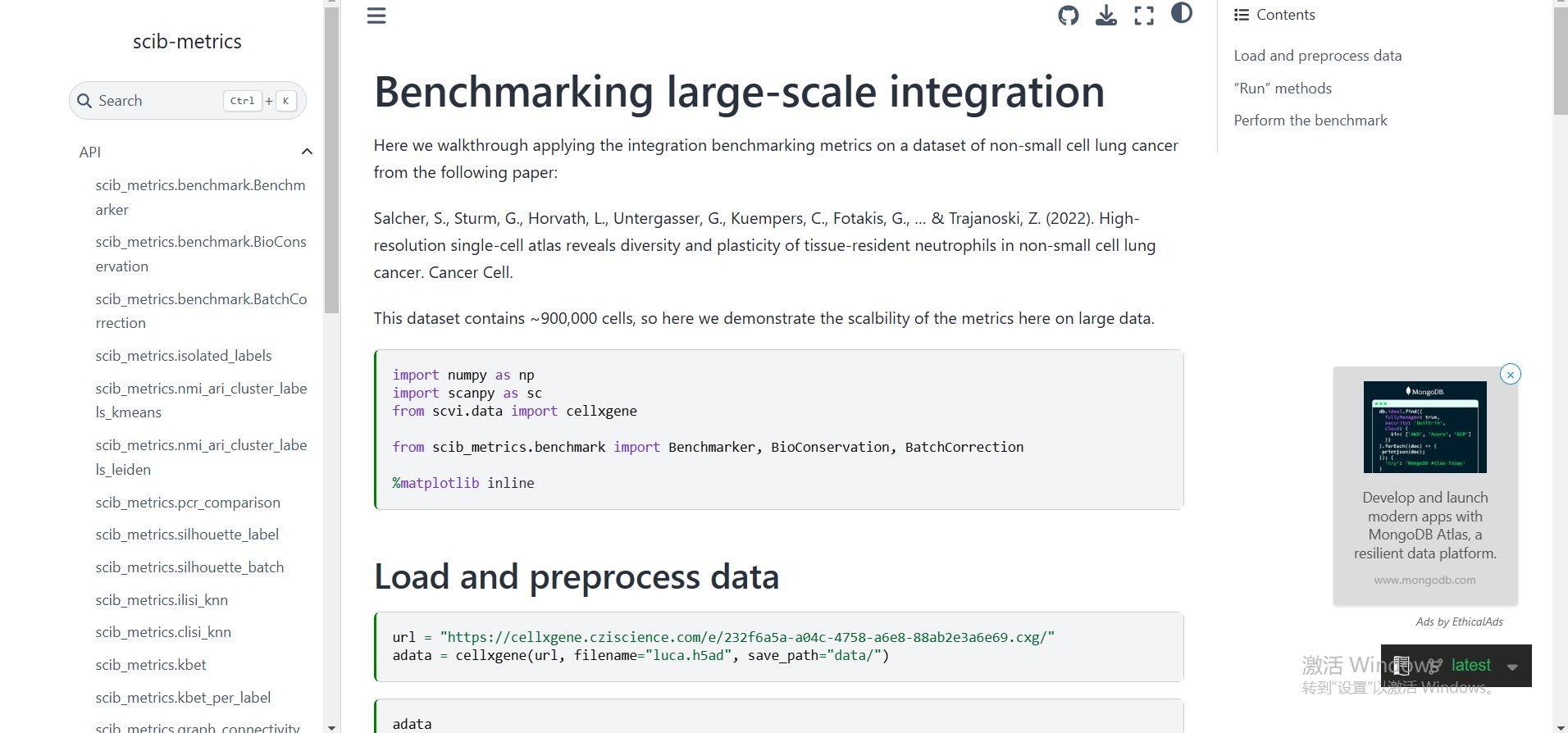


Accelerated这个词赫然醒目，所以我想我们的基本脚本就太老，应该换新的脚本，之前一直改老脚本的参数，改来改去其核心没有改变，终究会没有改变。找到好的方法，就学习这个文档组织新的脚本。学习方法最快的方法莫过于复现作者的测试过程，感谢作者提供的可复现的资源，学习作者的复现资源，主要关注的是其输入数据的结构，这样才可以直接搬运作者的代码(doge)。

<https://scib-metrics.readthedocs.io/en/latest/notebooks/lung_example.html>



<https://scib-metrics.readthedocs.io/en/latest/notebooks/large_scale.html>

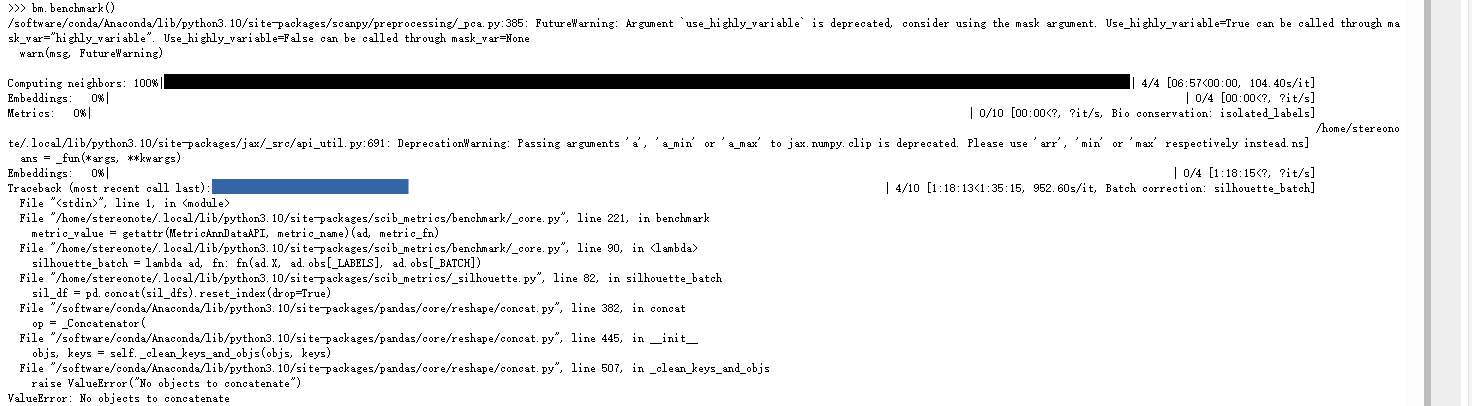


我主要关注的第一个复现，资源也合理10W左右的细胞量，作者使用了几种方法进行整合。看到作者的脚本，我产生了极大的疑惑（因为跟我之前的脚本方法完全不一样）。

1输入的h5ad文件只有一个，只取了每种整合方法的obsm[pca]文件，相比输入每种整合方法的整个h5ad确实更节约memory。马上我就产生了好奇：计算scib-metrics只用obsm信息吗？obs信息不看吗？

2文件输入的label\_key是cell\_type，cell\_type信息整合前后完全没变，如果是基于obs信息来看，如何算出整合前后的差别，毕竟cell\_type自始至终没有改变

带着这些个疑问，下载好测试数据，复现了一下作者流程，确实非常丝滑，差不多1个小时就运算好了。按着依葫芦画瓢运行了自己的数据，几个小时就得到了结果，相比之前效率提高了很多倍。但我仍然好奇label\_key的设置，为什么自始至终都不变了。因为我的数据使用的键是celltypes，但是因为celltypes加了species信息，其实celltypes是celltype + ’ \_’ + species的结构，这就导致所有的label都是独立的，就是说不同的batch不可能有相同的celltypes，这样就给计算造成了错误，无法完成计算。



在找这个错误的时候进一步阅读了scib-metric的内置脚本，通过测试\_silhouette.py发现是因为键值的问题，这也给我了不少启发，面对问题，可以追根溯源的去解决，虽然效率很低，但不失为一种方法，同时也会加深对于其脚本逻辑的理解。

面对键值不一样的问题，我想的是先人为创造一个看似符合要求的cell\_type，我以聚类后得到的数字信息作为cell\_type作为label\_key，果然就可以正常运行了，并且效率大大提高，测试六种方法只需要6个小时就完成了。这个时候就有疑问了，六种方法的obsm在一个h5ad文件里面，显然测试的就是obsm文件，但是obs的label\_key只有一种呀，而且我用的label\_key是harmony聚类后的数字信息，不同的数字信息会不会造成影响。果不出意料，换做用LIGER的聚类后数字信息，其测试结果完全不一样。这究竟是为什么呢？我发现只要谁用了obs的label\_key其测试得分就高，而且联想到作者给出的示例测试中其cell\_type是自始自纵未变的，说明其不应该是某一方法的聚类信息，而是实实在在的生物学性状注释信息。再联想到其计算的主要两大指标是biological conservation和batch correction，一下子就想通了，label\_key就是储存生物学性状的键，而我把cluster的信息误当作label\_key，就相当于完全信任其信息作为生物学意义的注释，而显然不是这样的。基于这样的想法，应该是在整合前有一个注释，整合之后，使用整合前的注释作为label\_key，即当下完全信任其生物学性质，当然整合后肯定会调整注释，这个过程就比较弹性，根据实际情况调整，其注释本身就是一个不断优化的过程。对于公共数据库的整合，其天生的注释结果可以直接帮助我们做scib-metrics的测评，而对于新测数据的测试，就需要劳费一些心神，好好做好注释，如何做好整合前的注释呢？这个问题还要多多思考