

文章

[Frank Ma](#) · 六月 27, 2022 阅读大约需 3 分钟

[Open Exchange](#)

使用Python和仪表盘制作的一个欧洲水状况应用程序

比较不同的商业智能技术是非常有趣的。我很好奇它们在功能、开发工具、速度和可用性方面有什么不同。

在这个应用程序中，我选择了一个有欧洲各国水状况的数据集。这是一个开源的数据集，包含1991年到2017年的观测数据。

团队和我决定使用IRIS BI、Tableau、PowerBI和InterSystems Reports（由Logi Reports驱动）在这个BI数据集的基础上制作一个模型

对于前端，我们通过Embedded Python在PythonFlask中制作了一个网页界面。

顺便说一下，其结果可以在这个网页上看到：<http://atscale.teccod.com:8080/> 你可以看看demo stand (演示台)，因为从资源库部署一个容器可能需要多至20分钟的时间。大量的python包，后面会有更多的原因。

The screenshot shows a web browser at the URL `atscale.teccod.com:8080`. The page is titled "water-conditions-Europe" and describes an application for data visualization using Python ML. It includes a sidebar menu with links to Home, Timeline, IRIS BI, Logi report, PowerBI report, and Tableau report. The main content area provides installation instructions for Docker, including a terminal command to clone the repository and another to start the container. Below the instructions, there is a preview of the application's user interface, which matches the sidebar seen in the top-left of the screenshot.

主页面

数据

事实上，数据似乎很小，期间只有17年：)

因此，在现有的基础上，我想延续数据集，为此使用了一个神经网络。使用同样的嵌入式Python，使用了Tensorflow，这个包下载后占据了511MB，不要惊讶

实际上，这也是容器部署时间长的原因--为神经网络下载了很多包，相当多的相关包，安装时间很长。不过会有一篇关于神经网络和Integrated ML（一体化机器学习）的单独文章，我很快会发表。

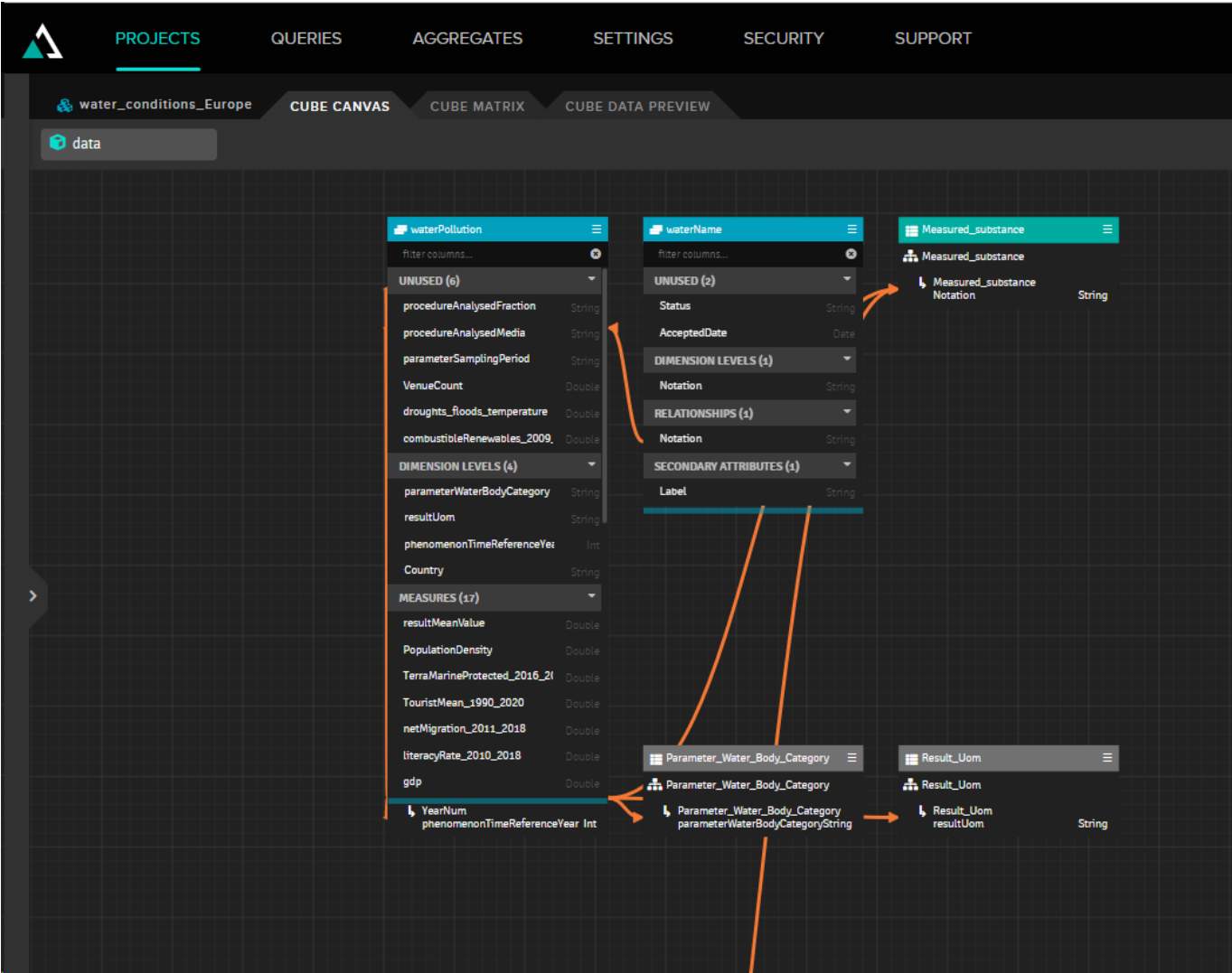
我还要说的是，预测的结果被输入到同一个数据库，所以您可以通过BI工具看到数据集。但是预测只针对法国的一条河，请仔细看一下。因为我们只有足够的力量计算一件事。完整的预测会花很长的时间。

Rows		Columns											
phenomenonTimeReferenceYe... Drop row here		resultMeanValue Country Drop column here											
phenomenonTimeReferenceYear		resultMeanValue	Austria	Belarus	Belgium	Bulgaria	Croatia	Czech Republic	Denmark	Finland	France	Ge	
2029		17.46									24		
2030		17.47									24		
2031		17.50									24		
2032		17.53									24		
2033		17.57									24		
2034		17.60									24		
2035		17.64									24		
2036		17.66									24		
2037		17.68									24		
2038		17.68									24		
2039		17.67									24		
2040		17.66									24		
2041		17.64									24		
2042		17.62									24		
2043		17.59									24		
2044		17.58									24		
2045		17.57									24		
2046		17.57									24		
2047		17.58									24		
2048		17.60									24		
2049		17.62									24		
2050		17.63									24		

Cells: 1,680 Total: 23,493.82

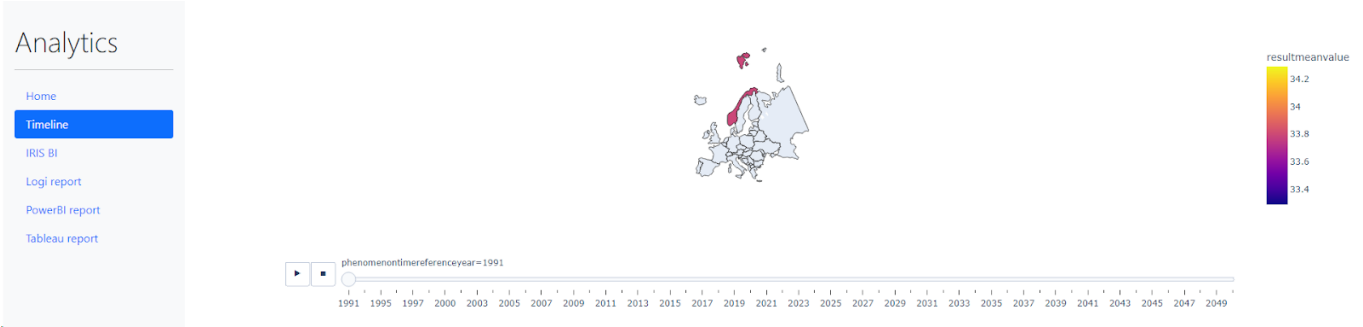
数据立方体

数据立方体是在IRIS中制作的，同时立方体也是在Adaptive Analytics（由AtScale驱动）中制作的。因此，IRIS的BI仪表盘是在IRIS上建立的，其余的工具（Logi、PowerBI、Tableau）从AtScale上获取数据。

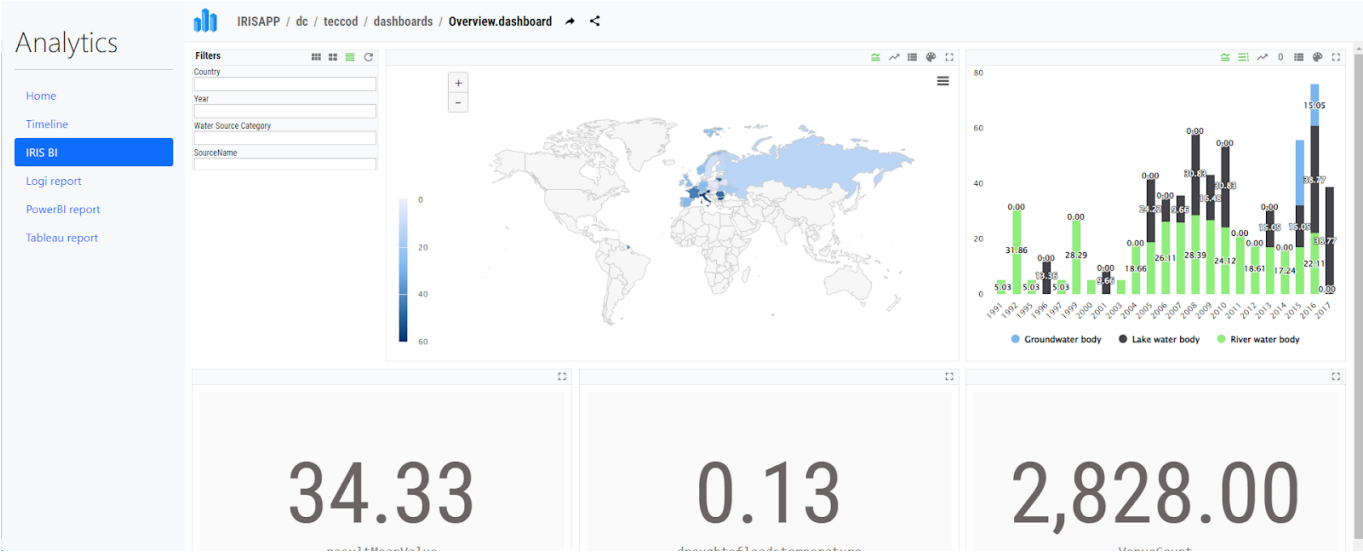


仪表盘

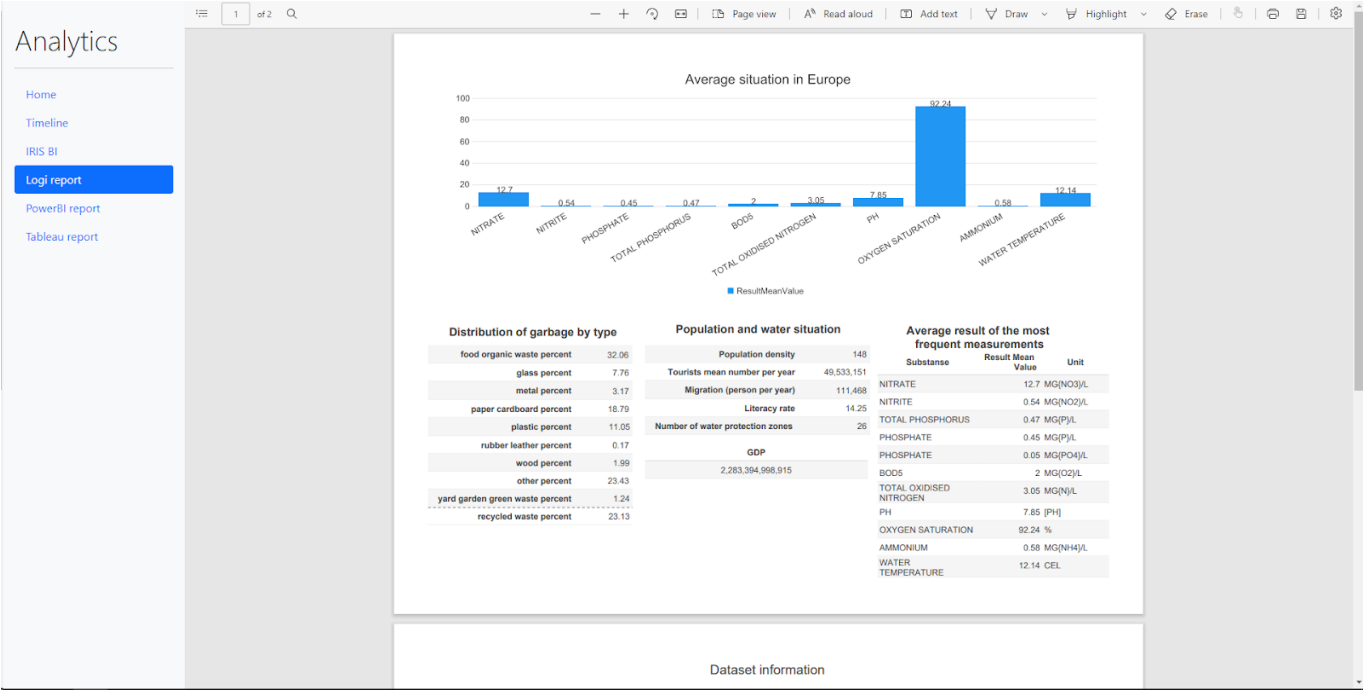
实际上商业智能系统是以多种形式呈现的。



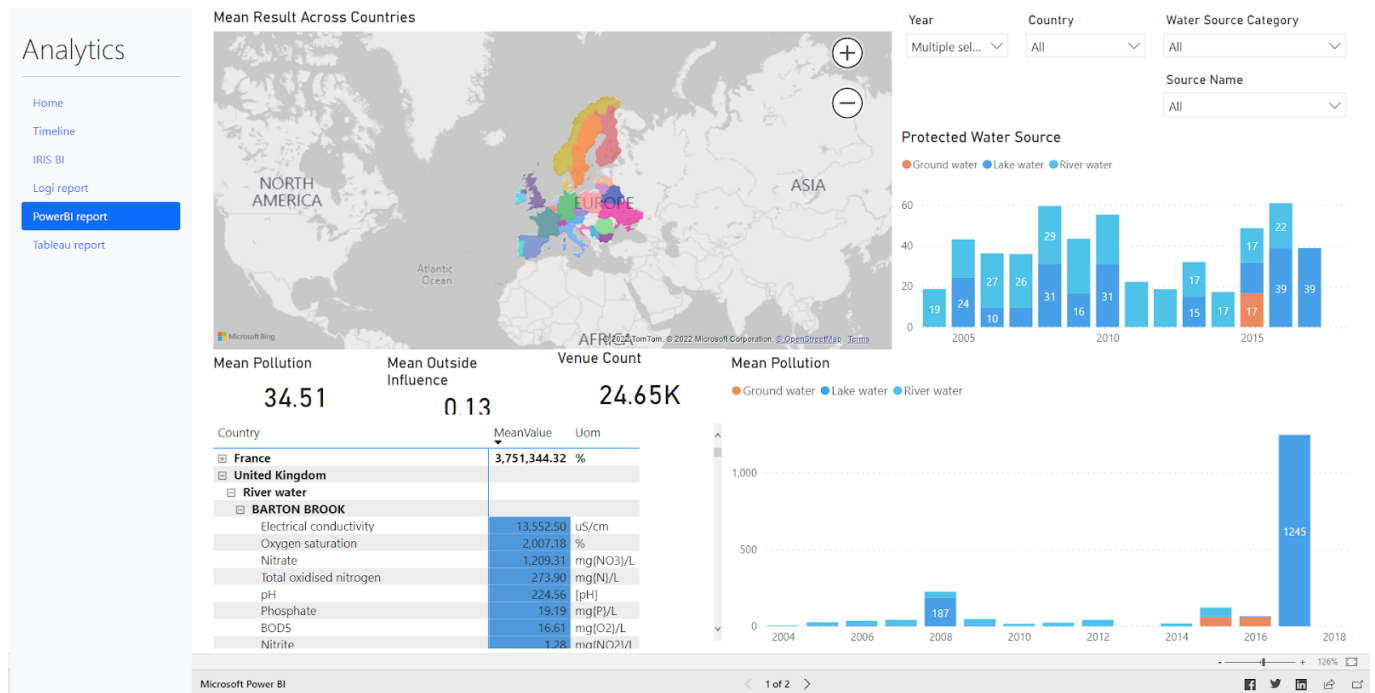
这是在python中的Dash。



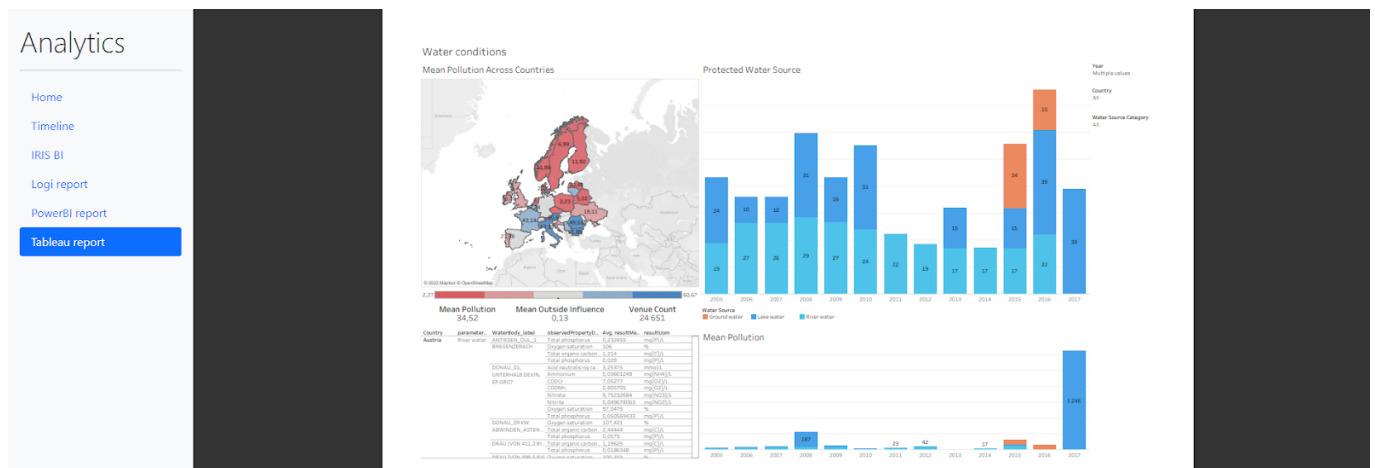
我们喜欢的IRIS BI



InterSystems报告（由Logi报告提供）。



PowerBI



[Europe/blob/main/iris/src/PythonFlask/pages/ml/mlrun.py](#) 文件夹中。

谢谢你阅读这篇文章，我在等待评论和反馈。很快就会有一篇关于在嵌入式Python和IntegratedML（一体化机器学习）中比较神经网络的文章，我将把它附在这篇文章中。

[##嵌入式 Python](#) [#自适应分析（Adaptive Analytics）](#) [#IntegratedML](#) [#InterSystems报表](#) [#仪表板](#) [#InterSystems IRIS](#)

[在 InterSystems Open Exchange 上检查相关应用程序](#)

源

URL:

<https://cn.community.intersystems.com/post/%E4%BD%BF%E7%94%A8python%E5%92%8C%E4%BB%AA%E8%A1%A8%E7%9B%98%E5%88%B6%E4%BD%9C%E7%9A%84%E4%B8%80%E4%B8%AA%E6%AC%A7%E6%B4%B2%E6%B0%B4%E7%8A%B6%E5%86%B5%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F>