**信息可视化project 1**

Summary

In this project, you will have to create the simples and clear visualisation for a given dataset. You have to follow the best practices taught in the class to use proper chart types and implement them in a way that is easy to read and understand by a general audience.

摘要

在这个项目中，你必须为一个给定的数据集创建简单而清晰的可视化效果。你必须遵循课堂上教授的最佳做法，使用适当的图表类型，并以易于阅读和理解的方式实现它们。

Submission

Deadline: September 30.

Group work in teams of 4 students.

The submission file needs to have the following file name format: group\_<group\_id>\_<student1\_id>\_<student2\_id>\_<student3\_id>.zip.

(Alternatively, you can simply submit the link to a Github repository.)

提交资料

截止日期。9月30日。

以4名学生为一组的小组工作。

提交的文件需要有以下文件名格式：group\_<group\_id>\_<student1\_id>\_<student2\_id>\_<student3\_id>.zip。

(另外，你也可以简单地提交Github资源库的链接)。

Dataset

In this project you will have to work with an existing dataset, available here. The dataset was collected from a study that compares the energy consumption of training a machine learning model across different training algorithms and different dataset properties (i.e., number of features, number of data points, and type of data).

数据集

在这个项目中，你将不得不使用一个现有的数据集，可在此获得。该数据集是从一项研究中收集的，该研究比较了不同训练算法和不同数据集属性（即特征数量、数据点数量和数据类型）下训练机器学习模型的能耗情况。

It contains 17 features:

* algorithm the machine learning algorithm (SVM, Decision Tree, Multinomial NB, KNN, Random Forest, AdaBoost, Bagging Classifier)
* RQ. Research question being study.
  + RQ2.1 refers to what is the impact of the size of the training data set (no\_datapoints) in train\_energy(J).
  + RQ2.2 refers to the impact of the number of the features in the training dataset (no\_features) in train\_energy(J).
  + all these research questions were tested acroos all the different machine learning algorithms.
* experiment\_id. Unique identifier of the experiment.
* iteration. Repetition number. Each experiment is repeated 30 times.
* no\_datapoints. Number of rows in the dataset used to train the machine learning model.
* no\_features. Number of features in the dataset used to train the machine learning model.
* preprocessing\_energy(J). Energy consumption of the preprocessing stage of the machine learning model.
* preprocessing\_time(s). Duration of the preprocessing stage of the machine learning model.
* train\_energy(J). Energy consumption of training the machine learning model.
* train\_time(s). Duration of training the machine learning model.
* predict\_energy(J). Energy consumption of making predictions in the machine learning model.
* predict\_time(s). Duration of making predictions in the machine learning model.
* datatype. Data type used to store the training dataset
* accuracy. Accuracy achieved by the machine learning model.
* precision. Precision achieved by the machine learning model.
* recall. Recall achieved by the machine learning model.
* f1. F1-score achieved by the machine learning model.

Although the dataset has 17 features. For this project, you will focus on 4 main features. 3 independent variables:  no\_datapoints, no\_features, and algorithm. 1 dependent variable: train\_energy(J). In sum, we want to make visualizations that show how the independent variables affect energy consumption — i.e., the dependant variable.

它包含17个特征。  
  
算法 机器学习算法（SVM, Decision Tree, Multinomial NB, KNN, Random Forest, AdaBoost, Bagging Classifier）。  
RQ。正在研究的问题。  
RQ2.1指的是训练数据集的大小（no\_datapoints）对train\_energy(J)有什么影响。  
RQ2.2指的是训练数据集中的特征数量（no\_features）对train\_energy(J)的影响。  
所有这些研究问题都是通过所有不同的机器学习算法来测试的。  
experiment\_id。实验的唯一标识符。  
迭代。重复次数。每个实验重复30次。  
no\_datapoints。用于训练机器学习模型的数据集中的行数。  
no\_features。用于训练机器学习模型的数据集中的特征数量。  
preprocessing\_energy(J)。机器学习模型预处理阶段的能量消耗。  
preprocessing\_time(s)。机器学习模型预处理阶段的时间。  
train\_energy(J)。训练机器学习模型的能量消耗。  
train\_time(s). 训练机器学习模型的时间。  
predict\_energy(J)。在机器学习模型中进行预测的能量消耗。  
predict\_time(s). 在机器学习模型中进行预测的时间。  
datatype。用于存储训练数据集的数据类型。  
准确性。机器学习模型取得的准确率。  
精度。机器学习模型实现的精度。  
召回率。机器学习模型实现的召回率。  
f1. 机器学习模型取得的F1分数。  
虽然数据集有17个特征。对于这个项目，你将专注于4个主要特征。3个自变量：no\_datapoints, no\_features, 和算法。1个因变量：train\_energy(J)。总而言之，我们要做一些可视化的东西，显示自变量是如何影响能源消耗的--也就是依赖变量。  
  
[Translated with DeepL](https://www.deepl.com/translator?utm_source=windows&utm_medium=app&utm_campaign=windows-share)

Requirements

1- Perform an exploratory analysis of the dataset. Remember that exploratory analyses do not need to be refined or clear. It is just a draft of several visualizations that help get familiar with the data.

* Hence, in this step you should have several visualizations that help getting an idea of the datasets and will serve as a starting point to the **explanatory visualizations** in this project. (Don’t forget the difference between exploratory and explanatory).
* **Hint:** Python notebooks are usually useful for exploratory analyses because you can combine python code, visualizations and markdown text. Use all of these elements.

2- Create a visualization that shows, for the algorithm SVM, how no\_features affects energy consumption (train\_energy(J)). This visualization should 1) choose the most suitable chart type, 2) follow the visualization guidelines taught in the class, and 3) be as simple as possible. If necessary, there should be a visual element highlighting how no\_features correlates with train\_energy(J).

* Note that each experiment is repeated 30 times (as denoted by the feature iteration). You may want to use the average of these 30-sized samples and maybe its standard deviation.

3- As you can imagine, creating a simple visualization to show all results is far from trivial. There are 3 main variables that are compared against train\_energy(J): no\_datapoints, no\_features, and algorithm. Create a visualization using small multiples that shows, **for each machine learning algorithm**, how no\_datapoints and no\_features affect energy consumption (train\_energy(J)).

4- Create a single plot that is able to capture most of the insights of the visualization in requirement 3. You wont be able to capture all the insights, but the idea is the capture as much as possible while **keeping the visualization simple and interesting**.

5- Create a visualization that shows an interesting insight in the data that was not unveiled by the visualizations of requirements 2, 3, and 4.

要求  
1- 对数据集进行探索性分析。记住，探索性分析不需要精炼或清晰。它只是几个可视化的草稿，有助于熟悉数据。  
  
因此，在这一步中，你应该有几个有助于了解数据集的可视化，并将作为本项目中解释性可视化的起点。(不要忘记探索性和解释性的区别）。  
提示：Python笔记本通常对探索性分析很有用，因为你可以把Python代码、可视化和markdown文本结合起来。使用所有这些元素。  
2- 创建一个可视化，对于SVM算法，显示no\_features如何影响能量消耗（train\_energy(J)）。这个可视化应该：1）选择最合适的图表类型；2）遵循课堂上教授的可视化准则；3）尽可能简单。如果有必要，应该有一个视觉元素，突出no\_features与train\_energy(J)的相关性。  
  
请注意，每个实验都要重复30次（如用特征迭代表示）。你可能想使用这30个大小样本的平均值，也许还有它的标准偏差。  
3- 正如你所想象的，创建一个简单的可视化来显示所有的结果远非易事。有3个主要的变量与train\_energy(J)进行比较：no\_datapoints, no\_features, 和 algorithm。使用小倍数创建一个可视化，对于每个机器学习算法，显示no\_datapoints和no\_features如何影响能量消耗（train\_energy(J)）。  
  
4- 创建一个能够捕捉到需求3中可视化的大部分洞察力的单一图表。你不可能捕捉到所有的洞察力，但我们的想法是在保持可视化简单和有趣的同时，尽可能多地捕捉到这些洞察力。  
  
5- 创建一个可视化，显示出需求2、3和4的可视化没有揭示的数据中有趣的洞察力。  
  
[Translated with DeepL](https://www.deepl.com/translator?utm_source=windows&utm_medium=app&utm_campaign=windows-share)

### Important Notes

* Requirements 2, 3, 4 should be modular. I.e., it should be easy to reutilize the graph in a different project. To achieve it, you can for example implement part of the graph as a generic method that receives data (the same way we did with the skinny plots of [assignment 6](http://210.26.48.15/mod/assign/view.php?id=874)).
* Use the potential of Jupyter Notebooks to make the submission appealing and easy to read. For example, use the markdown cells to structure the notebook.
* To import the dataset you can use Pandas or the standard libraries of Python.

重要说明

要求2、3、4应该是模块化的。也就是说，应该很容易在不同的项目中重新使用该图。为了实现这一点，你可以把图形的一部分作为一个接收数据的通用方法来实现（就像我们在作业6的瘦身图上所做的那样）。

利用Jupyter笔记本的潜力，使提交的文件有吸引力，易于阅读。例如，使用markdown单元来组织笔记本。为了导入数据集，你可以使用Pandas或Python的标准库。

Grading

This project affects 35% of the final grade. The following rubric items will be considered:

* Overall cohesiveness of the project.
  + To what extent is the team work cohesive?
  + Does the report completely describe the plots and how to read them?
* Quality of the submission.
  + Does the code follow code conventions?
  + Does the code execute without any issues?
* Clarity
  + Do visualizations follow best practices?
  + Are visualizations easy to read?
  + Did the students make an effort to present data in simple way?
* Relevance.
  + Are the visualizations showing relevant patterns of the data.
* Creativity.
  + Did the authors any unusual, yet interesting visualization?
* Robustness.
  + Does the plot work under different testing settings?
* Customization.
  + Is it possible to customize different aspects of the plot? E.g., color, transparency, etc.
* Organization.
  + Are the submission artifacts sound and clear?
  + Is the report easy to read?
  + 成绩
  + 这个项目影响到最终成绩的35%。以下的评分项目将被考虑。
  + - 项目的整体凝聚力。
  + o 团队工作的凝聚力有多大？
  + o 报告是否完全描述了情节和如何阅读它们？
  + - 提交的质量。
  + o 代码是否遵循代码惯例？
  + o 代码的执行是否没有任何问题？
  + - 清晰度
  + o 可视化是否遵循最佳实践？
  + o 可视化是否容易阅读？
  + o 学生们是否努力以简单的方式展示数据？
  + - 相关性。
  + o 可视化是否显示了数据的相关模式。
  + - 创造性。
  + o 作者是否有任何不寻常的，但有趣的可视化？
  + - 稳健性。
  + o 该图在不同的测试设置下是否有效？
  + - 自定义。
  + o 是否有可能对图的不同方面进行定制？例如，颜色、透明度等。
  + - 组织性。
  + o 提交的工件是否健全和清晰？
  + o 报告是否易于阅读？