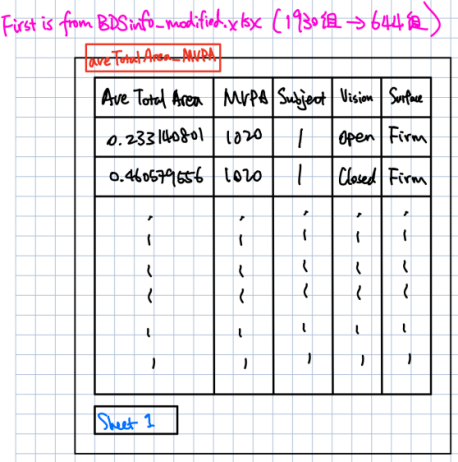
**Posture Project (updated by 04/04/21)**

【Goal】不是用data判断这个人身体好不好，而是用data去创建出一个model，然后把这个patient的数据代入去看ta能够如何improve（找到最适合这个patient的数据去让他根据数据去锻炼）。

【Steps】

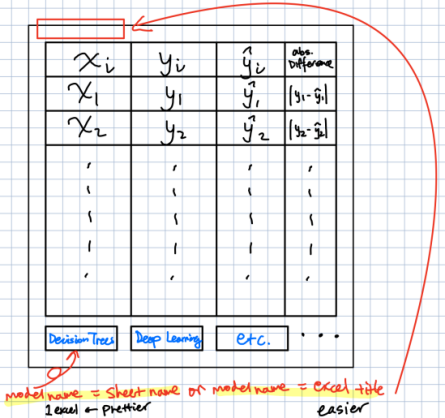
1. 确定x和y分别是什么（建立Average Total Area与其variable的form）。

\* 我脑海里的form长这样子：



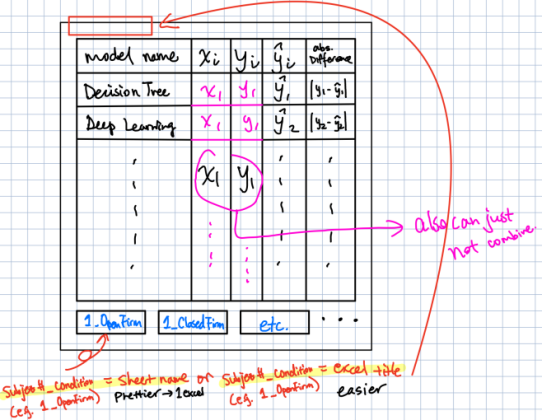
1. [Pre-processing 数据预处理](#数据预处理)
2. 建模：选一行为test data然后其他的行自然成为training data，用training data去建模（确认一种model）。
3. 将得出的结果导出excel，每个sheet为一个model。

\* 我脑海里的form长这样子：



1. 通过step 3的表格进行整理，变成以test data那一行为sheet名（例如X1），然后把每个model得出的数据整理至sheet里。

\* 我脑海里的form长这样子：



**Temporary Schedule**

具体任务内容详见

← github README

(command+click)

**橙 = YOUR DUE!!!**

**紫 = Scalzo**

**绿 = 组聊**

**红 = Presentation**

**灰 = completed**

(<https://github.com/chengmeowz/project-posture/blob/main/README.md>)

***April***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sunday 日 | Monday 一 | Tuesday 二 | Wednesday 三 | Thursday 四 | Friday 五 | Saturday 六 |
|  |  |  | **3/31**   1. No doc → N/A | **4/1**   1. Scalzo doc → 整理理解&问题(write: C) 2. No doc → Email Scalzo → a | 4/2 | 4/3 |
| **4/4**  a. Sunny Due @11:55pm | **4/5**  a. Discuss code & form @8am | 4/6 | 4/7 | **4/8**  a. Cheng/Sunny Due @11:55pm | 4/9 | 4/10 |
| 4/11 | 4/12 | **4/13**  a. Cheng Due @11:55pm | **4/14**   1. Discuss code & form @8am 2. Email Scalzo form/progress/理解/告知4/15 meet (write: S) | **4/15**  a. Meet w/ scalzo @6am | 4/16 | 4/17 |
| 4/18 | 4/19 | **4/20**  a. Sunny Due @11:55pm | **4/21**  a. Discuss code & form @8am  b. Email Scalzo form/progress/理解/告知4/22 meet (write: C) | **4/22**   1. Meet w/ scalzo @6am   **Congratulations! Stage 1 clear (/≧▽≦)/** | 4/23 | **4/24**  a. Cheng Due @11:55pm |
| **4/25** | **4/26** | **4/27**   1. Sunny Due @11:55pm | **4/28**   1. Discuss code & form @8am 2. Email Scalzo form/progress/理解/告知4/29 meet (write: S) | **4/29**   1. Meet w/ scalzo @6am & dice for next part   **Congratulations! Goodbye, MVPA\_minutes.week \(^o^)/** | 4/30 | 5/1 |

***May***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sunday 日 | Monday 一 | Tuesday 二 | Wednesday 三 | Thursday 四 | Friday 五 | Saturday 六 |
| 5/2 | 5/3 | 5/4 | 5/5 | 5/6 | **5/7**   1. Cheng & Sunny Due @12pm 2. Sum Up @下午   c. Email Scalzo form/progress/理解/告知4/28 meet (write: S) | **5/8**   1. Meet w/ scalzo @6am   b. 确定展示内容&分配时间 @上午  **Project Finished (๑•̀ㅂ•́)و✧** |
| 5/9 | 5/10 | **5/11**  a. Rehearsal @8am | **5/12 The Day**  Natural Science Seminar @3am |  |  |  |

**Supplement**

<https://towardsdatascience.com/data-preprocessing-in-python-b52b652e37d5> 就follow这个步骤ok

1. 【Scikit-learn: Preprocessing data 数据预处理】

* Why pre-processing?
  + 数据质量不高，再好的模型也没用。会有些差别很大的值影响数据准确性。
* How? （[实例](https://blog.csdn.net/weixin_39732316/article/details/110870330?utm_medium=distribute.pc_relevant_t0.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-1.baidujs&dist_request_id=&depth_1-utm_source=distribute.pc_relevant_t0.none-task-bl)）
  + Standardization数据标准化: 去除均值和方差进行缩放(mean removal and variance scaling)  
    <https://blog.csdn.net/qq_33472765/article/details/85944256>
    - 「Def」当单个特征的样本取值相差甚大或明显不遵从高斯正态分布时，标准化表现的效果较差。实际操作中，经常忽略特征数据的分布形状，移除每个特征均值，划分离散特征的标准差，从而等级化，进而实现数据中心化。
    - 「好处」提升模型精度、提升收敛速度
    - 「方法」

1. class sklearn.preprocessing.[StandardScaler](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html" \l "sklearn.preprocessing.StandardScaler)(\*, copy=True, with\_mean=True, with\_std=True)

* + - * **标准化数据通过减去均值然后除以方差（或标准差）**，这种数据标准化方法经过处理后数据符合标准正态分布，即均值为0，标准差为1
      * 「适用于」数据的分布本身就服从正态分布
      * 「函数」z = (x - u) / s

1. class sklearn.preprocessing.[MinMaxScaler](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html" \l "sklearn.preprocessing.MinMaxScaler)(feature\_range=0, 1, \*, copy=True, clip=False)

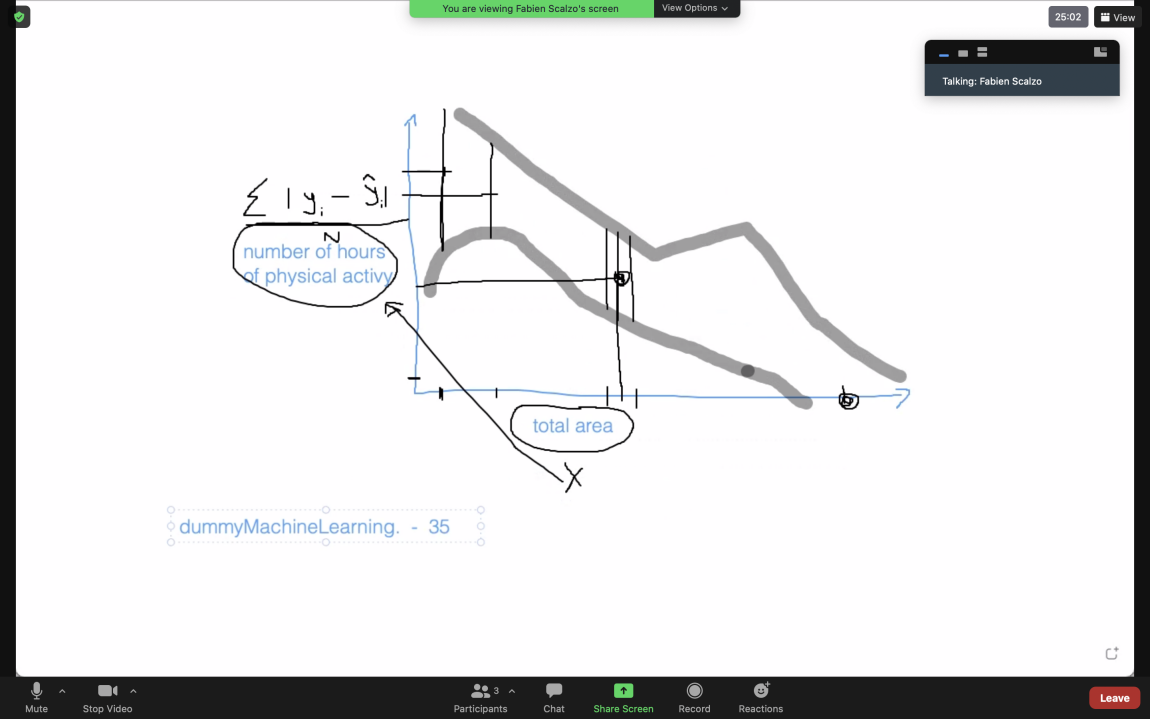
* 将特征缩放至特定区间,将特征缩放到给定的最小值和最大值之间，或者也可以将每个特征的最大绝对值转换至单位大小。这种方法是对原始数据的线性变换，将数据归一到[0,1]中间。
* 「函数」x = (x-min)/(max-min)
* 「缺陷」当有新数据加入时，可能导致max和min的变化，需要重新定义。这种方法对于outlier非常敏感，因为outlier影响了max或min值，所以这种方法只适用于数据在一个范围内分布的情况。  
  or class sklearn.preprocessing.[MaxAbsScaler](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MaxAbsScaler.html" \l "sklearn.preprocessing.MaxAbsScaler)(\*, copy=True)
* Same as MinMaxScaler, but MaxAbsScaler scales in a way that the training data lies within the range [-1, 1] by dividing through the largest maximum value in each feature. It is meant for data that is already centered at zero or sparse data.

1. class sklearn.preprocessing.[RobustScaler](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.RobustScaler.html" \l "sklearn.preprocessing.RobustScaler)(\*, with\_centering=True, with\_scaling=True, quantile\_range=25.0, 75.0, copy=True, unit\_variance=False)

* 它们对你的数据的中心和范围使用更有Robustness (在异常和危险情况下系统生存的能力)的估计。
* 「适用于」数据包含许多**异常值**

1.【Function】

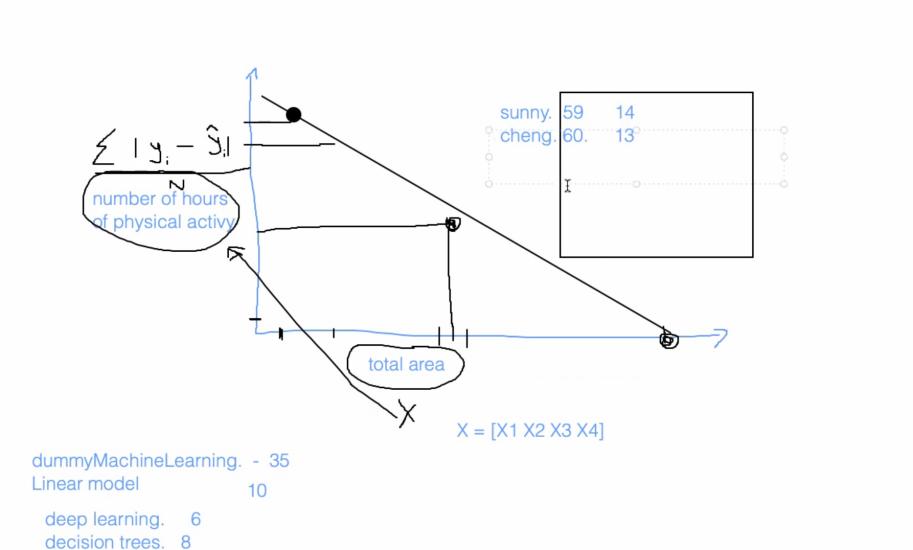
How to use Machine Learning to find correlation? (estimated not a line but an area两条灰色线中间区域)



* scikit-learn -> function fit

2.【找差异/错误率, aka 回归分析Regression Analysis】

* compare y and y hat -> /n to get average error ->



<https://blog.csdn.net/tonydz0523/article/details/84591954> {python数据分析：RA，有example}

- 「Def」确定两种或两种以上变量间相互依赖的定量关系的一种统计分析方法（研究的是因变量（目标）和自变量（预测器）之间的关系，来选择拟合效果最好的model）。

- 「分类」按照涉及的变量的多少，分为**一元回归**和**多元回归分析**；按照因变量的多少，可分为**简单回归分析**和**多重回归分析**；按照自变量和因变量之间的关系类型，可分为**线性回归分析**和**非线性回归分析**。

- 「适用情况」用于预测分析，时间序列模型以及发现变量之间的因果关系。例如，司机的鲁莽驾驶与道路交通事故数量之间的关系，最好的研究方法就是回归。主要应用场景是进行预测和控制，例如计划制定、KPI制定、目标制定等方面；也可以基于预测的数据与实际数据进行比对和分析，确定事件发展程度并给未来行动提供方向性指导。

1. 【交叉验证cross-validation】

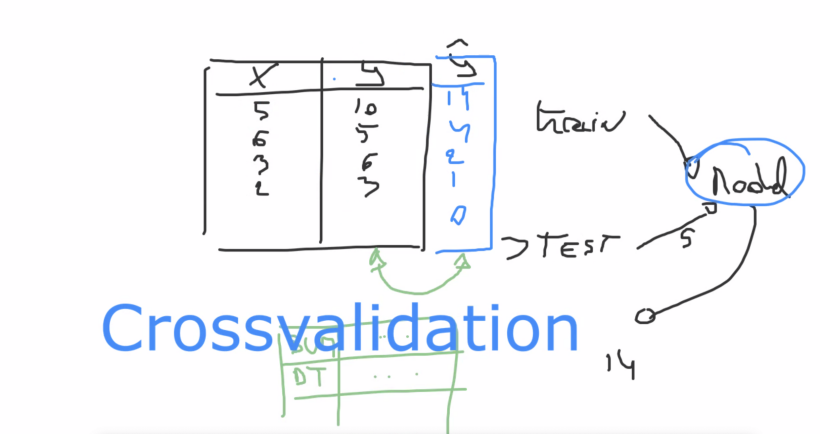
<https://www.youtube.com/watch?v=OoUX-nOEjG0> {斯坦福大学李飞飞教授的计算机视觉、深度学习神课}

- 「作用」在数据集很小的时候防止over-fitting，可以“充分利用”有限的数据找到合适的模型参数，防止过度拟合。

- 「适用情况」数据集较小时，一般做深度学习跑标准数据集的时候用不到。（因为如果不做c-v，只需训练一次，现在等于训练了N次，训练时间变成了N倍）

- 「方法」将数据集平均分成N份， 选其中的N-1份作为训练集（training set），剩余的1份作为验证集(validation set)，用以上N种情况的训练集训练得到模型的参数。

- 「结果」可以选择平均N次训练的模型参数 or 用N次结果中validation error最小的那个模型参数（不常用）。



* leave one out -> 选一行的数据作为test，其他用于training the model，然后再将test的x代入model中得出y hat，| y - hat y | 就是差异/这个model的偏差值。
  + 如果你有更多的数据你可以分成若干份，每次用一份做测试。

机器学习训练时主要需要学两样东西。一样是模型weights，比如coefficients and biases。另外一样是模型的超参数，比如学习率，regularisation strength什么的。cross validation 主要是为了找到最好的超参数。然后用最好的超参数训练模型，得到最终的模型weights。