# 给初学者和初级数据科学家的建议 给初学者和初级数据科学家的建议 原文链接： 我会在几年前告诉自己的 img Image credit: Alice Truong 动机 两年前，我分享了我在数据科学领域内的行业。最初写这本书的目的是为了对我自己进行自我反思，以庆祝我在Twitter上的两年推特活动，但我最后将其发表在Medium上，因为我认为它对许多有抱负的数据科学家来说非常有用。 快进到2017年，我在Airbnb工作了不到两年，最近晋升为一名高级数据科学家。这是一个行业头衔，用来表示某些人技术上已经达到了一定的高度。当我回顾迄今为止的旅程并想象接下来会发生什么时，我再次写下了一些我希望在职业生涯的早些时候就知道的课程。

原文链接：[Advice For New and Junior Data Scientists](https://medium.com/@rchang/advice-for-new-and-junior-data-scientists-2ab02396cf5b?from=hackcv&hmsr=hackcv.com&utm_medium=hackcv.com&utm_source=hackcv.com)

## 我会在几年前告诉自己的



img

Image credit: Alice Truong

### 动机

两年前，我分享了我在数据科学领域内的行业[经验](https://medium.com/@rchang/my-two-year-journey-as-a-data-scientist-at-twitter-f0c13298aee6)。最初写这本书的目的是为了对我自己进行自我反思，以庆祝我在Twitter上的两年推特活动，但我最后将其发表在Medium上，因为我认为它对许多有抱负的数据科学家来说非常有用。

快进到2017年，我在Airbnb工作了不到两年，最近晋升为一名高级数据科学家。这是一个行业头衔，用来表示某些人技术上已经达到了一定的高度。当我回顾迄今为止的旅程并想象接下来会发生什么时，我再次写下了一些我希望在职业生涯的早些时候就知道的课程。

如果我之前文章的目标读者是有抱负的数据科学家和该领域的新手，那么本文适用于已经在该领域但刚刚起步的人。我的目标不仅是希望通过这篇文章来提醒自己我学到的重要知识，而且还可以激励其他人踏上DS生涯！  
如果我之前文章的目标读者是有抱负的数据科学家和该领域的新手，那么本文适用于已经在该领域但刚刚起步的人。我的目标不仅是希望通过这篇文章来提醒自己我学到的重要知识，而且还可以激励其他人踏上DS生涯！  
你在谁的关键路径上？  
郭飞（Philip Guo）是一位杰出的学术和多才多艺的博客作者，他回顾了自己在学生，实习生和研究员期间与各种导师互动的经验。 在他的污点文章“ ”中，他进行了以下观察：  
如果我处于导师的关键道路[为了职业发展或成就]，那么他们会尽力提供我成功的所需。 相反，如果我不在导师的关键路线上，那么我只能自供自给。 […]如果你在某人的关键道路，那么你会迫使他们将你的成功与他们的成功联系起来，这将激励他们尽最大努力帮助你。

### 你在谁的关键路径上？

郭飞（Philip Guo）是一位杰出的学术和多才多艺的博客作者，他回顾了自己在学生，实习生和研究员期间与各种导师互动的经验。 在他的污点文章“ [你的关键路径在哪里？](http://www.pgbovine.net/critical-path.htm)”中，他进行了以下观察：

如果我处于导师的**关键道路**[为了职业发展或成就]，那么他们会尽力提供我成功的所需。 相反，如果我不在导师的关键路线上，那么我只能自供自给。 […]如果你在某人的关键道路，那么你会迫使他们将你的成功与他们的成功联系起来，这将激励他们尽最大努力帮助你。



img  
img  
Image credit: The Icefields Parkway // Daniel Han  
这种工作动态非常直观，我希望我可以在职业生涯的早期将其内部化，这时可以选择项目，选择团队，甚至可以选择要为哪些导师或公司工作。  
例如，在Twitter上，我一直想了解有关机器学习的更多信息，尽管我的团队非常依赖数据驱动，但仍非常需要数据科学家专注于实验设计和产品分析。 虽然我尽了最大的努力，但我发现很难将这种理想的愿望与团队的关键项目相结合。  
结果，当我到Airbnb时，我做出了一个明智的决定，即专注于加入ML对于其成功至关重要的项目/团队。 我与我的经理一起确定了一些有前景的机会，其中之一是对Airbnb上房源的终身价值（LTV）进行建模。

Image credit: The Icefields Parkway // Daniel Han

这种工作动态非常直观，我希望我可以在职业生涯的早期将其内部化，这时可以选择项目，选择团队，甚至可以选择要为哪些导师或公司工作。

例如，在Twitter上，我一直想了解有关机器学习的更多信息，尽管我的团队非常依赖数据驱动，但仍非常需要数据科学家专注于实验设计和产品分析。 虽然我尽了最大的努力，但我发现很难将这种理想的愿望与团队的关键项目相结合。

结果，当我到Airbnb时，我做出了一个明智的决定，即专注于加入ML对于其成功至关重要的项目/团队。 我与我的经理一起确定了一些有前景的机会，其中之一是对Airbnb上房源的终身价值（LTV）进行建模。

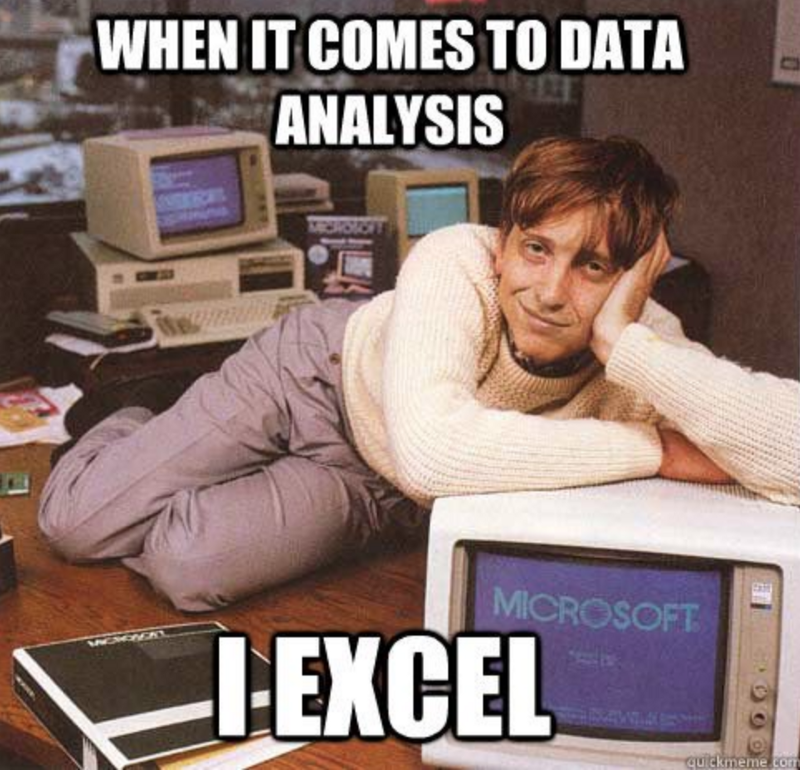
这个项目不仅对我们业务的成功至关重要，而且对我的职业发展也至关重要。我了解到很多的工作流程，除了在解决具体业务问题的背景下进行学习之外，没有更好的学习方法。  
这个项目不仅对我们业务的成功至关重要，而且对我的职业发展也至关重要。我了解到很多的工作流程，除了在解决具体业务问题的背景下进行学习之外，没有更好的学习方法。  
毫无疑问，我很幸运地找到了一个与我的志向以及我想在其中培养技能的项目。我相信，在导师的关键路径上挑选项目的框架可以使我们随着时间的流逝，将自己的理想与正确的项目相匹配，从而使我们越来越“幸运”。  
我学到的原则：我们所有人都有我们想发展的技能和我们追求的知识兴趣。评估我们的理想与我们所处环境的关键路径的契合度非常重要。找到关键路径与你的关键路径最相符的项目，团队和公司。

毫无疑问，我很幸运地找到了一个与我的志向以及我想在其中培养技能的项目。我相信，在导师的关键路径上挑选项目的框架可以使我们随着时间的流逝，将自己的理想与正确的项目相匹配，从而使我们越来越“幸运”。

**我学到的原则**：*我们所有人都有我们想发展的技能和我们追求的知识兴趣。评估我们的理想与我们所处环境的关键路径的契合度非常重要。找到关键路径与你的关键路径最相符的项目，团队和公司。*

### 选择正确的工具解决问题 选择正确的工具解决问题 在加入Airbnb之前，我一生的大部分时间都在使用R和。 在开始LTV项目之后，我很快意识到可交付成果不是一段分析代码，而是一条生产机器学习流水线。 鉴于使用Python在中构建复杂的流水线要容易得多， 我面临两难的境地— —我应该从R切换到Python吗？ img Image source: quickmeme.com (besides R or Python, Excel is also a serious contender 👊) 事实证明，这是数据科学家中一个非常普遍的问题，因为许多人都在选择哪种语言上纠结。对我来说，一旦承诺一个或另一个，显然会有转换成本。我通过分析利弊来了解折衷，但是我对它的思考越多，我越容易陷入两难之地。 （这里是一个有趣的（展示了这个概念）。最终，我在上阅读以下回复后，摆脱了这种纠结：

在加入Airbnb之前，我一生的大部分时间都在使用R和[dplyr](https://github.com/tidyverse/dplyr)。 在开始LTV项目之后，我很快意识到可交付成果不是一段分析代码，而是一条生产机器学习流水线。 鉴于使用Python在[Airflow](https://medium.com/the-astronomer-journey/airflow-and-the-future-of-data-engineering-aqa-266f68d956a9)中构建复杂的流水线要容易得多， 我面临两难的境地— —我应该从R切换到Python吗？



img

Image source: quickmeme.com (besides R or Python, Excel is also a serious contender 👊)

事实证明，这是数据科学家中一个非常普遍的问题，因为许多人都在选择哪种语言上纠结。对我来说，一旦承诺一个或另一个，显然会有转换成本。我通过分析利弊来了解折衷，但是我对它的思考越多，我越容易陷入两难之地。 （这里是一个有趣的[talk](https://blog.dominodatalab.com/video-huge-debate-r-vs-python-data-science/)（展示了这个概念）。最终，我在[Reddit](https://www.reddit.com/r/Python/comments/2tkkxd/considering_putting_my_efforts_into_python/)上阅读以下回复后，摆脱了这种纠结：

不用考虑学习哪种编程语言，而是考虑哪种语言可以为你提供适合你问题的正确的领域特定语言（DSL）。  
不用考虑学习哪种编程语言，而是考虑哪种语言可以为你提供适合你问题的正确的领域特定语言（DSL）。  
工具的适当性始终取决于上下文和具体问题。这与我是否应该学习Python无关，而与Python是否适合这项工作无关。为了详细说明这一点，下面是一些示例：  
如果您的目标是最新应用，最先进的统计方法，则R可能是更好的选择。 为什么？ 因为R是由统计学家为统计学家建立的。 如今，学者们不仅在论文中发表研究成果，还在R包中发表研究成果。 每周，上都会提供许多有趣的新R软件包，例如。

工具的适当性始终取决于上下文和具体问题。这与我是否应该学习Python无关，而与Python是否适合这项工作无关。为了详细说明这一点，下面是一些示例：

* 如果您的目标是最新应用，最先进的统计方法，则R可能是更好的选择。 为什么？ 因为R是由统计学家为统计学家建立的。 如今，学者们不仅在论文中发表研究成果，还在R包中发表研究成果。 每周，[CRAN](https://cran.r-project.org/mirrors.html)上都会提供许多有趣的新R软件包，例如[这个](https://github.com/susanathey/%20causalTree)。
* 另一方面，Python是构建生产数据管道的理想选择，因为它是一种通用的编程语言。 例如，可以使用轻松包装模型 在Hive中进行分布式计算，使用复杂的逻辑编排Airflow DAG，或编写Flask Web应用程序以在浏览器中展示模型的输出。  
  另一方面，Python是构建生产数据管道的理想选择，因为它是一种通用的编程语言。 例如，可以使用轻松包装模型 在Hive中进行分布式计算，使用复杂的逻辑编排Airflow DAG，或编写Flask Web应用程序以在浏览器中展示模型的输出。  
  对于我的特定项目，我需要构建一个生产机器学习管道，如果我使用Python进行工作，我的生活会容易得多。 最终，我卷起袖子迎接了这个新挑战！  
  我学到的原则: 不问自己只使用一种技术或编程语言，而是问自己，什么是最能帮助你解决问题的工具或技术？ 专注于解决问题，使用工具就变成很自然地。

对于我的特定项目，我需要构建一个生产机器学习管道，如果我使用Python进行工作，我的生活会容易得多。 最终，我卷起袖子迎接了这个新挑战！

**我学到的原则**: *不问自己只使用一种技术或编程语言，而是问自己，什么是最能帮助你解决问题的工具或技术？ 专注于解决问题，使用工具就变成很自然地*。

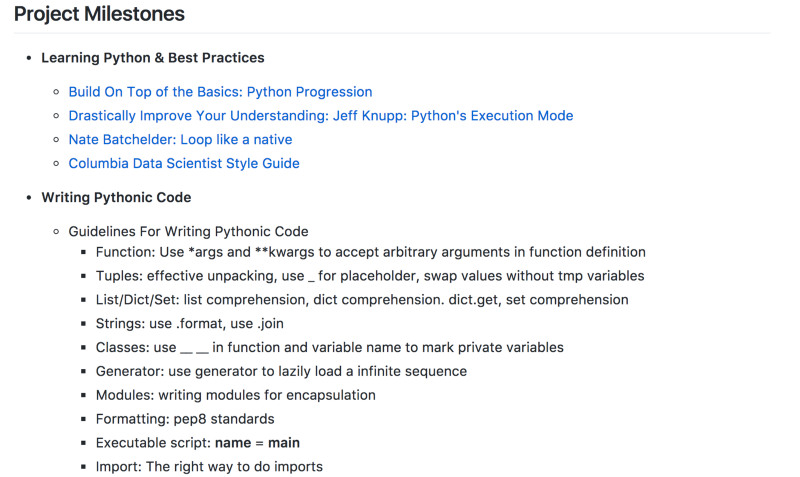
### 建立学习项目 建立学习项目 即使我以前从未使用过Python从事数据科学工作，但我还是以使用了该语言。 但是，我从来没有真正正确地学习过Python基础知识。 结果，当代码组织成 时，我感到恐惧，我一直想知道 init.py 的用途 To really learn the fundamentals properly this time, I took inspiration from Anders Ericsson’s research on : Deliberate Practice is activities designed, typically by a teacher, for the sole purpose of effectively improving specific aspects of an individual’s performance. Given that I was my own teacher, insights from Dr. Ericsson were very helpful. For example, I kicked off my “learning project” by curating a set of materials that were most relevant for doing ML in Python. This process took me a few weeks until I settled on a personalized . I stress tested this curriculum by asking experienced Pythonistas to review my plan. All of this pre-work was meant to ensure I would be on the right learning path.

即使我以前从未使用过Python从事数据科学工作，但我还是以[不同身份](https://medium.com/@rchang/learning-how-to-build-a-web-application-c5499bd15c8f)使用了该语言。 但是，我从来没有真正正确地学习过Python基础知识。 结果，当代码组织成 [classes](https://jeffknupp.com/blog/2014/06/18/improve-your-python-python-classes-and-object-oriented-programming/)时，我感到恐惧，我一直想知道 ***init****.py* 的用途

To really learn the fundamentals properly this time, I took inspiration from Anders Ericsson’s research on [**Deliberate Practice**](https://www.amazon.com/Peak-Secrets-New-Science-Expertise/dp/1531864880):

**Deliberate Practice** is activities designed, typically by a teacher, for the sole purpose of effectively improving specific aspects of an individual’s performance.

Given that I was my own teacher, insights from Dr. Ericsson were very helpful. For example, I kicked off my “learning project” by curating a set of materials that were most relevant for doing ML in Python. This process took me a few weeks until I settled on a personalized [curriculum](https://github.com/robert8138/python-deliberate-practice). I stress tested this curriculum by asking experienced Pythonistas to review my plan. All of this pre-work was meant to ensure I would be on the right learning path.



img  
img公司

Here is a glimpse of my personalized curriculum  
以下是我的个性化课程

Once I had a clearly defined curriculum, I used the following strategies to deliberately practice on the job:  
一旦我有了明确的课程设置，我就用以下策略在工作中刻意练习：

* Practice Repeatedly: I forced myself to carry out mundane, non mission-critical analyses in Python instead of in R. This dragged down my productivity initially, but it forced me to get familiar with the basic API of , without the burden of needing to meet an urgent deadline.  
  反复练习：我强迫自己用Python而不是R来执行平凡的、非任务关键型的分析。这一点最初拖累了我的工作效率，但它迫使我熟悉的基本API，而不需要满足紧迫的最后期限。
* Create Feedback Loop: I found opportunities to review other people’s code and fix small bugs when appropriate. For example, I tried to understand how our internal Python libraries were designed before using them. When writing my own code, I also tried to refactor it several times and make it more readable for everyone.  
  创建反馈循环：我找到了机会来审查其他人的代码，并在适当的时候修复小错误。例如，在使用内部Python库之前，我试图了解它们是如何设计的。在编写自己的代码时，我还尝试了多次重构，使其对每个人都更具可读性。
* Learn By Chunking and Recalling: By the end of each week, I wrote down my , which included the important resources I studied in that week, concepts I learned, and any major takeaways during that week. By recalling the materials I learned, I was able to internalize the concepts better.  
  通过分块和回忆来学习：每周末，我写下我的，其中包括我在那一周学习的重要资源，我学到的概念，以及在那一周的任何主要收获。通过回忆我所学的材料，我能够更好地将这些概念内化。

Slowly and gradually, I got better each week. It certainly wasn’t easy though: there were times when I had to look up basic syntax in both R and Python because I was switching back and forth between the two languages. That said, I kept in mind that this is a long term investment, and dividends will be paid as I dived into the ML project.  
慢慢地，渐渐地，我每星期都好起来了。不过，这当然不容易：有时我不得不在R和Python中查找基本语法，因为我在这两种语言之间来回切换。也就是说，我牢记这是一项长期投资，当我投入到ML项目中时，将支付股息。

Principle I learned: As supported by many , before diving into a project, planning ahead helps you to practice more deliberately. Repeating, chunking, recalling, and getting feedback are among the most useful activities to reinforce learning.  
我学到的原则是：在许多人的支持下，在进入一个项目之前，提前计划可以帮助你更加谨慎地练习。重复、分块、回忆和获得反馈是强化学习最有用的活动之一。

### Partnering With Experienced Data Scientists 与经验丰富的数据科学家合作

One of the key ingredients of deliberate practice is to receive timely and actionable feedback. No great athletes, musicians, or mathematicians are able to achieve greatness without coaching or targeted feedback.  
深思熟虑的练习的一个关键要素是获得及时和可行的反馈。没有优秀的运动员、音乐家或数学家能够在没有指导或有针对性的反馈的情况下取得成功。

One common trait I have observed from people who have a strong is that they are generally not ashamed of acknowledging what they don’t know and they constantly ask for feedback.  
我从一个强壮的人身上观察到的一个共同特点是，他们通常不羞于承认自己不知道的事情，并且不断要求反馈。

Looking back at my own academic and professional career so far, many times in the past I self-censored my questions because I did not want to appear incapable. However, over time I realized that this attitude was rather detrimental — in the long run, most instances of self-censorship are missed opportunities for learning rather than shame.  
回顾到目前为止我自己的学术和职业生涯，过去很多时候我都会自我审查我的问题，因为我不想显得无能。然而，随着时间的推移，我意识到这种态度是相当有害的 - 从长远来看，大多数自我审查的例子都是错过了学习的机会，而不是羞耻。



img  
img公司

Image source: edutopia — It’s important to have a growth mindset!  
图片来源：edutopia - 重要的是要有一个成长的心态！

Before this project, I had very little experience putting machine learning models into . Of the many decisions that I made for the project, one of the best decisions was to declare early and shamelessly to my collaborators that I know very little about ML infrastructure, but that I wanted to learn. I promised them, however, as I got more knowledgeable, I would make myself useful for the team.  
在这个项目之前，我很少有将机器学习模型应用到其中的经验。在我为这个项目所做的众多决定中，最好的决定之一是尽早地、无耻地向我的合作者宣布，我对ML基础设施知之甚少，但我想学习。不过，我向他们保证，随着我的知识越来越丰富，我会让自己对球队有用。

This turned out to be a pretty good strategy, because people generally love to share their knowledge, especially when they know their mentorship will benefit themselves eventually. Below are a few examples that I would not have learned so quickly without the guidance of my partners:  
这是一个很好的策略，因为人们通常喜欢分享他们的知识，特别是当他们知道他们的导师关系最终会对自己有利的时候。以下是一些没有我的合作伙伴的指导，我不会那么快学会的例子：

* : My collaborator suggested to me that I can make my code more modular by adopting Sklearn’s pipeline construct. Essentially, pipelines define a series of data transformation that are consistent across training and scoring. This tool made my code cleaner, more reusable, and more easily compatible with production models.  
  ：我的合作者建议我可以通过采用Sklearn的管道结构使代码更模块化。本质上，管道定义了一系列的数据转换，这些转换在训练和评分方面是一致的。这个工具使我的代码更干净，更可重用，更容易与生产模型兼容。
* Model Diagnostics: Given that our prediction problem involves time, my collaborator taught me that typical cross validation will not work, as we could run into the risk of predicting the past using future data. Instead, a better method would be to use . I also learned different diagnostic techniques such as and various other evaluation metrics such as .  
  模型诊断：鉴于我们的预测问题涉及时间，我的合作者告诉我，典型的交叉验证是行不通的，因为我们可能会遇到使用未来数据预测过去的风险。相反，更好的方法是使用。我还学习了不同的诊断技术（如）和各种其他评估指标（如）。
* Machine Learning Infrastructure: With the help from ML infra engineers, I learned about managing package dependency via virtualenvs, how to serialize models using , and how to make the model available at scoring time using . All these are data engineering skills that I didn’t know before.  
  机器学习基础设施：在ML infra工程师的帮助下，我学习了如何通过virtualenvs管理包依赖性，如何使用序列化模型，以及如何在评分时使用使模型可用。所有这些都是我以前不知道的数据工程技能。

As I learned more new concepts, not only was I able to apply them for my own project, I was able to drive engaging discussions with the machine learning infrastructure team so they can build better ML tools for data scientists. This creates a virtuous cycle because the knowledge that was shared with me made me a better partner and collaborator.  
随着我学到了更多的新概念，我不仅能够将它们应用到我自己的项目中，还能够推动与机器学习基础设施团队的讨论，以便他们能够为数据科学家构建更好的ML工具。这创造了一个良性循环，因为与我分享的知识使我成为更好的合作伙伴和合作者。

Principle I learned: In the long run, most instances of self-censorship are missed opportunities for learning rather than shame. Declare early and shamelessly your desire to learn, and make yourself useful as you become better.  
我学到的原则是：从长远来看，大多数自我审查的例子都是错失了学习的机会，而不是羞耻。早而无耻地宣布你的求知欲，当你变得更好时使自己变得有用。

### Teaching And Evangelizing 传道授业

As I got closer to putting my model into production, I noticed that a lot of the skills that I picked up could be very valuable for other data scientists on our team. Having been a graduate student instructor for years, I always knew I had a passion for teaching, and I always learned more about the subject when I became the teacher. Richard Feynman, the late Nobel Laureate in Physics and a , spoke about his view on teaching:  
随着我的模型即将投入生产，我注意到我学到的许多技能对我们团队中的其他数据科学家来说可能非常有价值。作为一名多年的研究生导师，我一直知道我对教学充满热情，当我成为老师时，我对这门学科的了解也越来越多已故诺贝尔物理学奖获得者理查德·费曼（Richard Feynman）谈到了他对教学的看法：

Richard Feynman was once asked by a Caltech faculty member to explain why spin one-half particles obey Fermi Dirac statistics. Rising to the challenge, he said, “I’ll prepare a freshman lecture on it.” But a few days later he told the faculty member, “You know, I couldn’t do it. I couldn’t reduce it to the freshman level. That means we really don’t understand it.”  
理查德·费曼曾经被加州理工学院的一名教员要求解释为什么自旋一半的粒子服从费米-狄拉克统计。面对挑战，他说：“我会准备一个关于它的新生讲座。”但几天后，他告诉老师，“你知道，我做不到。我不能把它降到一年级。这意味着我们真的不明白。”

This was really inspiring — if you can’t reduce the subject to its core and make it accessible for others, that means you don’t really understand it. Knowing that teaching these skills can improve my understanding, I seek opportunities to carefully document my model implementations, give learning lunches, and encourage others to try out the tools. This was a win-win because evangelization raises awareness, which in tern helps to drive tool adoption across the team.  
这真的很鼓舞人心——如果你不能把主题缩小到核心，让其他人也能理解，那就意味着你并不真正理解它。我知道教授这些技能可以提高我的理解力，所以我寻找机会仔细记录我的模型实现，提供学习午餐，并鼓励其他人试用这些工具。这是一个双赢的过程，因为传播提高了人们的意识，这有助于推动整个团队采用工具。

As of late September, I have started collaborating with our internal team to prepare a series of classes on our internal ML tools. I am not exactly sure where this will go, but I am very excited about driving more ML education at Airbnb.  
截至9月底，我已经开始与我们的内部团队合作，在我们的内部ML工具上准备一系列的类。我不确定这会去哪里，但我非常兴奋驾驶更多的毫升教育在Airbnb。

Finally, I would end this section with a tweet from :  
最后，我将在这一节的结尾处加上一条来自：

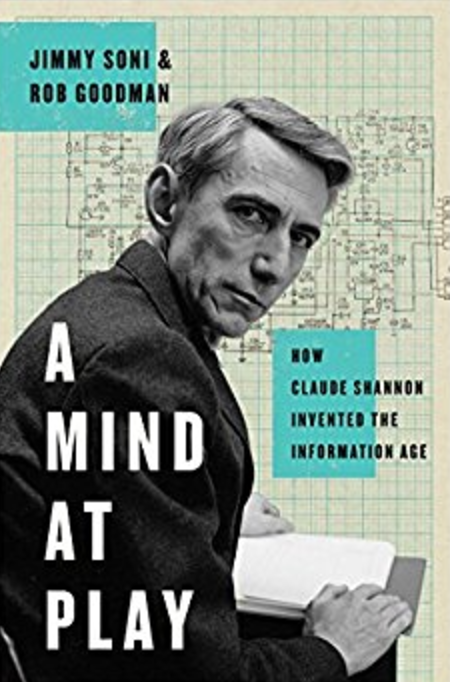
<https://twitter.com/hadleywickham/status/890107458219368448>

Principle I learned: Teaching is the best way to test your understanding of the subject and the best way to improve your skills. When you learn something valuable, share it with others. You don’t always have to create new software, explaining how existing tools work can also be super valuable.  
我学到的原则是：教学是检验你对学科理解的最好方法，也是提高你技能的最好方法。当你学到有价值的东西时，与他人分享。你不一定要创造新的软件，解释现有的工具如何工作也可以是非常有价值的。

### At Step K, Think About Your Step K+1 在步骤K，想想你的步骤K+1

From focusing on my own deliverables, to partnering with the ML infrastructure team, to finally teaching and enabling other data scientists to learn more about ML tools, I am really happy that the scope of my original project was much larger than it was a few months ago. Yet, admittedly, I never anticipated this in the first place.  
从专注于我自己的可交付成果，到与ML基础架构团队合作，最后教给其他数据科学家并使他们能够更多地了解ML工具，我真的很高兴我最初项目的范围比几个月前大得多。然而，不可否认的是，我从来没有预料到这一点。

As I reflected on the evolution of this project, one thing that was different from my previous projects was that I always had a slight dissatisfaction with the current state of things, and I always wanted to make it a little bit better. The most eloquent way to characterize this is from :  
当我反思这个项目的发展时，有一点不同于我以前的项目，那就是我总是对目前的状况有一点不满，我总是想把它做得更好一点。最能说明这一点的方法是：



img  
img公司

Image source: Book cover from “A Mind at Play: How Claude Shannon Invented the Information Page” by Jimmy Soni, Rob Goodman  
图片来源：Jimmy Soni，Rob Goodman撰写的“游戏中的大脑：Claude Shannon如何发明信息页”一书的封面

“There’s the idea of dissatisfaction. By this I don’t mean a pessimistic dissatisfaction of the world — we don’t like the way things are — I mean a constructive dissatisfaction. The idea could be expressed in the words, This is OK, but I think things could be done better. I think there is a neater way to do this. I think things could be improved a little. In other words, there is continually a slight irritation when things don’t look quite right; and I think that dissatisfaction in present days is a key driving force in good scientists.”  
“有不满的想法。我指的不是对世界的悲观不满，我们不喜欢事情的发展方式，我指的是建设性的不满。这个想法可以用文字来表达，这没关系，但我认为事情可以做得更好。我认为有一个更整洁的方法来做这件事。我认为情况会有所改善。换言之，当事情看起来不太对劲时，人们会持续感到些许恼怒；我认为，现在的不满是优秀科学家的一个关键推动力。”

By no means I am a qualified scientist (even though that is somehow in my job title), but I do think the characterization of slight dissatisfaction is quite telling for whether you will be able to extend the impact of your project. Throughout my project, whenever I am at step K, I naturally would start thinking about what to do for step K+1 and beyond:  
我决不是一个合格的科学家（尽管这在我的职称上有所体现），但我确实认为，对于你是否能够扩大你项目的影响，轻微不满的描述是很有说服力的。在我的整个项目中，每当我处于步骤K时，我自然会开始考虑如何处理步骤K+1及其后的问题：

* From “I don’t know how to build a production model, let me figure out how”to “I think the tools can be improved, here are my pain points, suggestion and feedback for how to make the tools better”, I reframed myself from a customer to a partner with ML infrastructure team.  
  从“我不知道如何构建生产模型，让我想想如何”到“我认为工具可以改进，以下是我的痛点、如何改进工具的建议和反馈”，我将自己从一个客户变成了ML基础架构团队的合作伙伴。
* From “let me learn the tools so I can be good at it” to “let’s make these tools more accessible for all the other Data Scientists interested in ML”, I reframed myself from a partner to an evangelizer.  
  从“让我学习这些工具，这样我就能擅长它”到“让所有对ML感兴趣的数据科学家都能更容易地使用这些工具”，我将自己从一个合作伙伴变成了一个传教士。

I think this mindset is extremely helpful — use your good taste and slight dissatisfaction to fuel your progress with persistence. That said, I do think that this dissatisfaction cannot be manufactured, and can only come from working on a problem you care about, which brings to my last point.  
我认为这种心态是非常有帮助的，用你的好品味和轻微的不满来激励你坚持不懈地前进。也就是说，我确实认为这种不满是不能制造的，只能来自于解决你关心的问题，这就引出了我的最后一点。

Principle I learned: Pay attention to your inner dissatisfaction when working on a project. These are clues to how you can improve and scale your project to the next level.  
我学到的原则是：在做一个项目时要注意你内心的不满。这些是如何改进项目并将其扩展到下一个级别的线索。

### Parting Thoughts: You And Your Work 离别的思绪：你和你的工作

Recently, I came across a lecture from , who is an American Mathematician well known for many of his scientific contributions, including Hamming code and Hamming distance. The lecture was titled , where Dr. Hamming said it can very well be renamed as “You And Your Career”.  
最近，我遇到了一位美国数学家的演讲，他以许多科学贡献而闻名，包括哈明码和哈明距离。这堂课的题目是，哈明博士说，它可以很好地改名为“你和你的事业”。

<https://www.youtube.com/watch?v=a1zDuOPkMSw>

As he shared his stories, a few important points stood out for me.  
当他讲述他的故事时，有几点对我很重要。

If what you are doing is not important, not likely to be important, why are you doing it? You must work on important problems. I spent Friday afternoon for years thinking about the important problems in my field [that’s 10% of my working time].  
如果你正在做的事情不重要，也不太可能重要，那你为什么要这么做呢？你必须处理重要的问题。我花了很多年的时间在周五下午思考我所在领域的重要问题（这是我工作时间的10%）。

Let me warn you about important problems, importance is not the consequence, some problems are not important because you haven’t gotten an attack. The importance of problem, to a great extent, depends on if you got a way of attacking the problem.  
让我警告你一些重要的问题，重要不是结果，有些问题不重要是因为你没有受到攻击。问题的重要性在很大程度上取决于你是否有办法解决问题。

This whole course, I am trying to teach you something about style and taste, so you’ll be able to have some hunch on when the problem is right, what problem is right, how to go about it. The right problem at the right time at the right way counts, and nothing else counts. Nothing.  
整个过程中，我试图教你一些关于风格和品味的东西，这样你就能有一些预感什么时候问题是对的，什么问题是对的，如何去做。在正确的时间以正确的方式解决正确的问题是有价值的，其他的都不重要。没有什么。

When Dr. Hamming speaks about importance, he means problems that are important to you. For him, it was scientific problems, and for many of us, it might be something different. He also talked about the importance of having a plan of attack. If you don’t have a plan, the problem does not matter, however big the consequences. Lastly, he mentioned doing it with your own unique style and taste.  
当哈明博士谈到重要性时，他指的是对你来说很重要的问题。对他来说，这是科学问题，对我们许多人来说，这可能是不同的。他还谈到制定进攻计划的重要性。如果你没有计划，不管后果有多大，问题都无关紧要。最后，他提到用你自己独特的风格和品味来做。

His bar for doing great work is extremely high, but it’s one worth pursuing. When you find your important problem, you will naturally try to make it better and make it more impactful; you will find ways to teach other about its significance; you will spend time to learn from other great people and build your craft.

**What’s a problem that is important to you that is on your critical path?**

*I would like to thank* [*Jason Goodman*](https://medium.com/@jasonkgoodman) *and Tim Kwan for reviewing my post and giving me feedback*