

11 MATLAB 脚本

编写和保存自己的 MATLAB 程序。

在命令行窗口中依次输入命令，可以非常便捷地进行尝试并查看结果。但是，当你开始从多个命令构建任务时，修改命令会变得有些麻烦，因为这样需要重新运行所有后续命令。

要将所有命令及其输出存放在一处，可以创建一个**实时脚本**，这是一种交互式的 MATLAB 文档，其中包含命令、输出、描述性文本、公式、图像等内容。

依次点击“新建 > 实时脚本”，打开一个空白实时脚本。现在可以输入命令，并切换到文本模式以添加一些说明。要对这些命令进行计算，请点击“运行所有 (Run All)”按钮，所有命令会按顺序进行计算。这种交互式文档，可以将代码和结果都保存下来，便于分享。

```
Sine Curve
This live script creates data points on a sine curve and plots them.

Create data
dx = 0.1;
x = 0:dx:2*pi;
y = sin(x);

Plot Data
plot(x,y)
ylabel('sine')
```

任务

所显示的实时脚本可绘制从 0 到 2π 的正弦波形。试着修改该实时脚本，以便绘制 0 到 4π 范围内的正弦波形。

现在，使用 `xlabel` 函数将标签 'radians' 添加到绘图的 x 轴。

12 逻辑数组

12.1 逻辑运算和变量

任务 1

关系运算符（例如 `>`、`<`、`==` 和 `~=`）执行两个值之间的比较。相等或不相等比较的结果为 1 (true) 或 0 (false)。

任务

使用关系运算符 > 测试 π 是否大于 3。

请注意，可以在右侧的工作区窗口中看到新的逻辑变量。逻辑变量只能是以下两个可能值之一：1 (true) 和 0 (false)。

再次测试 π 是否大于 3，但这一次将输出赋给一个名为 `test` 的变量。

任务 2

我们可以使用关系运算符将某个向量或矩阵与单个标量值进行比较。结果是原始数组相同大小的逻辑数组。

```
>> [5 10 15] > 12
ans =
    0     0     1
```

任务 3

试着创建一个表示测试结果的变量x，测试v1的每个元素是否大于5的输出。

可以使用关系运算符对两个数组的对应元素进行比较。这两个数组的大小必须相同，其比较结果是这两个数组具有相同大小的逻辑数组。

```
>> [5 10 15] > [6 9 20]
ans =
    0     1     0
```

任务

试着创建一个表示测试结果的变量y的变量，测试 v1的每个元素是否大于 v2中的对应元素。

12.2 组合逻辑条件

任务 1

MATLAB 包含 AND (&) 和 OR (|) 等逻辑运算符，可将多个逻辑条件组合在一起。如果两个元素都为true，& 运算符将返回 true (1)，否则返回false(0)。例如：

```
>> x = (pi > 5) & (0 < 6)
x =
    0
```

任务

试着创建一个表示测试结果的变量test，测试 π 大于 3 和 x大于0.9是否同时成立。

附加资源：逻辑运算符：<https://www.mathworks.com/help/matlab/logical-operations.html>

12.3 逻辑索引

```
v1 = [4.0753
      6.6678
      1.5177
      3.6375
      4.7243
      9.0698
      5.3002];
v2 = [0.5000
      2.1328
      3.6852
      8.5389
      10.1570]
```

```
2.8739
4.4508];
```

任务 1

我们可以使用逻辑数组作为数组索引，在这种情况下，MATLAB 会提取索引为true的数组元素。以下示例将会提取v1中大于 6 的所有元素。

```
>> v = v1(v1 > 6)
v =
    6.6678
    9.0698
```

任务

试着创建一个名为v的变量，其中包含v1中所有小于 4 的元素。试着创建一个名为 s 的变量，其中包 v1 中小于 4 的元素所在位置对应的元素。

任务2

我们可以使用逻辑索引在数组中重新赋值。例如，如果我们要将数组 x 中等于 999 的所有值都替换为 0，请使用以下语法。

```
x(x==999) = 0
```

试着修改 v1，将其中大于 5 的所有值都替换 为10。

13 编程

13.1 决策分支

任务 1

有时，我们可能想在仅满足某一条件时才执行某个代码段，可以使用 if 语句完成该操作。每个 if 语句都必须包含一个 if 关键字和一个 end 关键字，只有满足条件时，才会执行 if 和 end 关键字之间的代码。

```
x = rand;
if x > 0.5
    y = 3; %只有 x > 0.5 才会执行
end
```

任务 试着修改 decisionBranching.mlx 实时脚本，在 A 大于 0 时才会执行 B = sqrt(A) 行。

任务 2

通常，我们可能还希望在所设条件不成立时执行其他代码。为此，可以使用 else 关键字，如下所示。

```
x = rand;  
if x > 0.5  
    y = 3;  
else  
    y = 4;  
end  
任务
```

试着修改 decisionBranching.mlx 实时脚本，在 if 条件不成立时，实时脚本会将变量 B 设置为 0。

附加资源：if 语句: <http://www.mathworks.com/help/matlab/ref/if.html> switch 语句: <http://www.mathworks.com/help/matlab/ref/switch.html>

13.2 for 循环

常见的编程任务是重复执行某个代码段。在 MATLAB 中，我们可以使用 for 循环完成该操作。

```
for i = 1:3  
    disp(i)  
end
```

请注意，for 循环包含单个 end 关键字，与 if 语句类似。

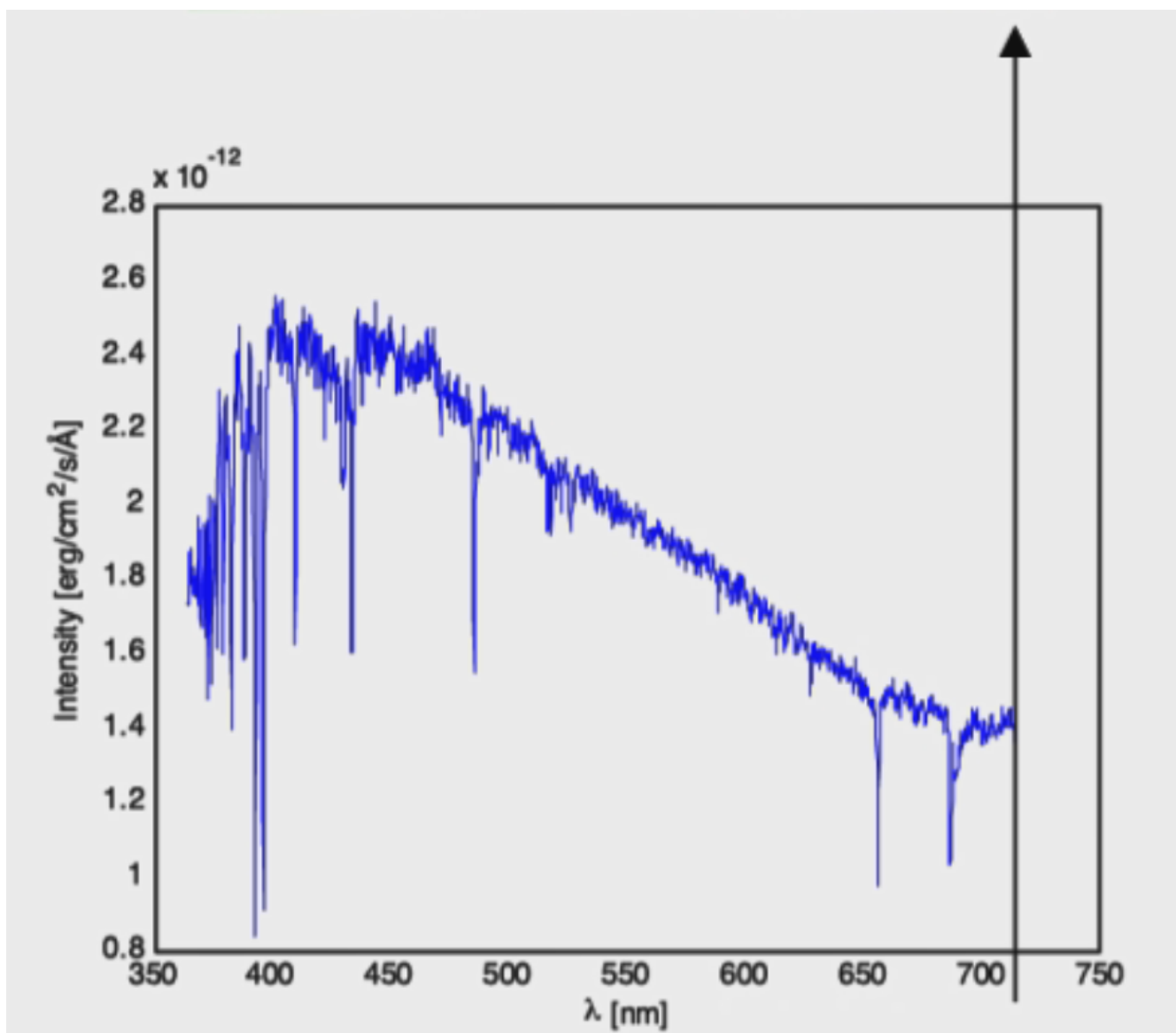
运行此代码时，for 和 end 关键字之间的代码在该示例中将被执行三次，因为循环计数器 (i) 通过 1:3 (1、2 和 3) 进行计数。任务 以循环方式包装实时脚本 forLoop.mlx 中的 disp 函数，执行 5 次。

首次循环时，idx 的值应为 1，并且每次迭代时递增 1。

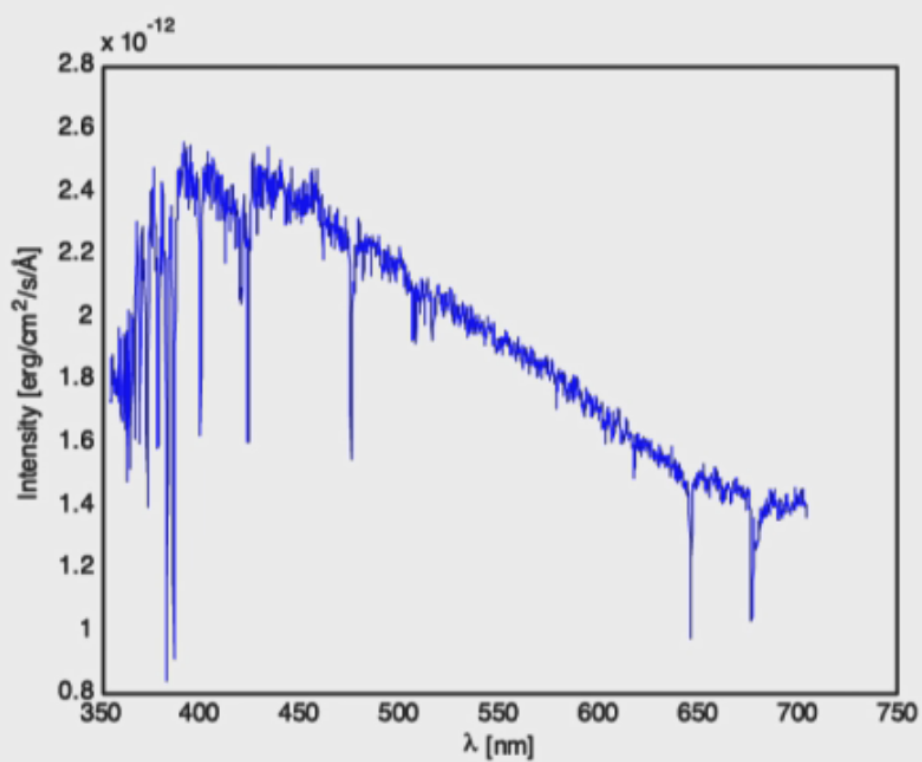
附加资源：for 循环: <http://www.mathworks.com/help/matlab/ref/for.html> while 循环: <http://www.mathworks.com/help/matlab/ref/while.html>

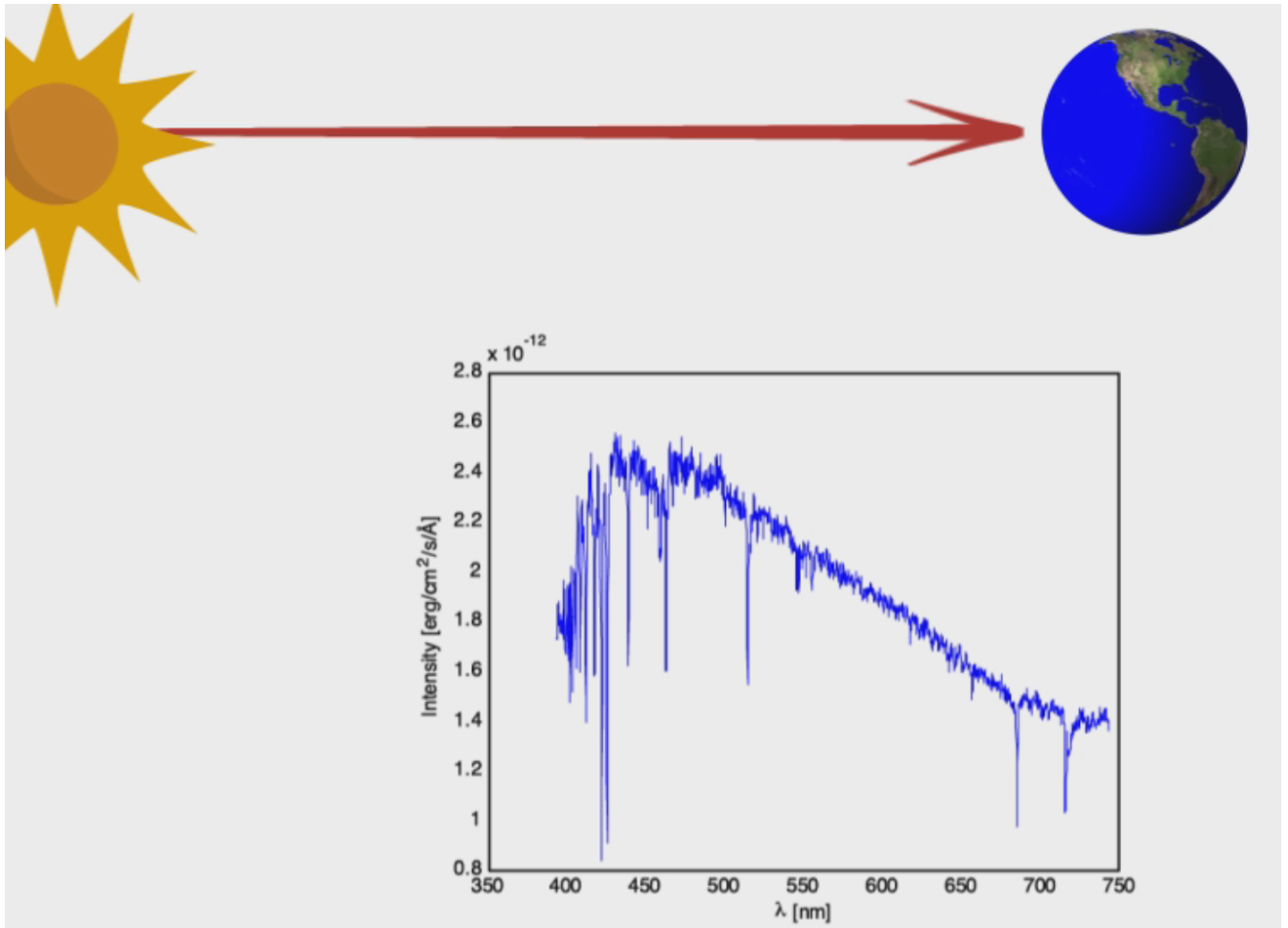
14 综合项目练习

我们可以通过观察分析恒星发出的光来了解它的许多信息，哪怕它位于1000光年以外。使用衍射光栅，可以分离光束中的不同波长，就像水滴将阳光衍射成彩虹一样。测量在每个波长上的发光强度，即可得到恒星的特征光谱。



这是HD94028的光谱，它是狮子座的一颗暗星。注意光谱中的这一低谷，它意味着这一波长，即650nm附近发光较少，这是许多恒星所共有的特征。因为大多数恒星主要由氢组成，而氢原子吸收波长为 656.3nm 的光。但是我们可以在项目中看到，HD94028光谱中的这一低谷即氢- α 谱线，出现在波长略大于656.3纳米的位置。这是因为恒星在逐渐远离地球，就像警笛声在接近我们时音调变高，在远离我们时变得低沉一样。





光谱发生蓝移，即向较低波长移动。恒星远离你时，光谱发生红移，即向较长波长移动。在此项目中，你将使用 MATLAB 找到 HD94028 的氢- α 谱线波长观测值，并将其与已知的氢原子吸收波长比较，从而确定恒星远离地球的快慢程度。天文学家精确测量一段时间内的光谱，从中发现恒星发生轻微震荡，由此证明附近存在行星。

任务 1

`spectra` 数据是在均匀间隔的波长上收集的，起始波长值 (λ_{start})、间隔大小 (λ_{delta}) 和观测值个数是已知的。

任务

创建一个名为 `lambdaEnd` (λ_{end}) 的变量，表示所记录光谱中的最后一个波长值。我们可以通过方程

$\lambda_{end} = \lambda_{start} + (n_{Obs}-1)\lambda_{delta}$ 来计算。

冒号 (`:`) 运算符可创建具有固定步长的行向量。使用转置运算符 (`'`) 将行变成列。

任务2

创建一个名为 `lambda` (λ) 的列向量，表示频谱中的波长，范围从 λ_{start} 到 λ_{end} ，步长为 λ_{delta} 。

任务3

`spectra` 的每一列分别对应不同恒星的光谱。第六列是恒星 HD 94028 的光谱。将 `spectra` 的第六列提取到一个名为 `s` 的向量。

任务4

使用 `loglog` 函数（用法同 `plot` 函数），在两个坐标轴上使用对数刻度。将光谱 (`s`) 作为波长 (`lambda`) 的函数进行绘图，在两个坐标轴上使用对数刻度。使用点标记 (`.`) 并用实线 (`-`) 连接各点。

任务5

`min` 函数可以带有两个输出，其中第二个输出为最小值的索引。该索引与氢- α 谱线的位置对应。创建两个变量 `sHa` 和 `idx`，分别表示 `s` 的最小值和最小值的位置索引。

任务6

使用 `idx` 对 `lambda` 进行索引，找到氢- α 谱线的波长。将结果存储为 `lambdaHa` (λ_{Ha})

完成！ 我们已确定 HD 94028 的氢- α 谱线的波长为 656.62 nm，该波长略长于实验值 656.28 nm。接下来，我们将使用该波长来确定恒星对地球的相对速度。

任务7

实时脚本 `findRedShift` 会执行上一节中的步骤来计算氢- α 波长。在 `findRedShift.mlx` 中添加代码，以在光谱绘图中标示氢- α 谱线的位置。在脚本相应位置进行编辑，将 `x = lambdaHa`、`y = sHa` 处的单个点绘制成一个大小 ('MarkerSize') 为 8 的红色方框 ('rs')，并添加到现有图。

任务8

实时脚本 `findRedShift` 会找到恒星的氢- α 波长。在该脚本中，您可以使用公式 $z = (\lambda_{\text{Ha}} / 656.28) - 1$ 计算红移量。然后，只需将红移量与光速 (299792.458 km/s) 相乘，就可以计算速度。任务在 `findRedShift.mlx` 中添加代码，以计算红移量以及恒星远离地球的速度 (km/s)。将红移量保存为变量 `z` 的变量中，将速度保存为变量 `speed`。

任务9

创建用于计算红移的实时脚本之后，您可以非常方便地修改脚本，并对 `spectra` 矩阵中的任何恒星执行同样的计算。修改 `findRedShift.mlx` 中的代码，计算 `spectra` 中的第二个恒星（而不是第六个）的红移。

完成！ 使用 MATLAB 脚本，您可以很方便地计算数据集中任何恒星的红移。您可以对收集到的数据集中的所有恒星执行同样的操作。

现在，您已经通过计算光谱强度 (`s`) 的最小值得到了恒星的氢- α 波长。不过，对于某些恒星，氢- α 谱线会出现在最大强度值处，而不是最小强度值处。因此，更可靠的方法是计算与异常值的最大偏差，其中异常值定义为 `s` 和 `mean(s)` 差的绝对值。