



湖北大学  
HUBEI UNIVERSITY

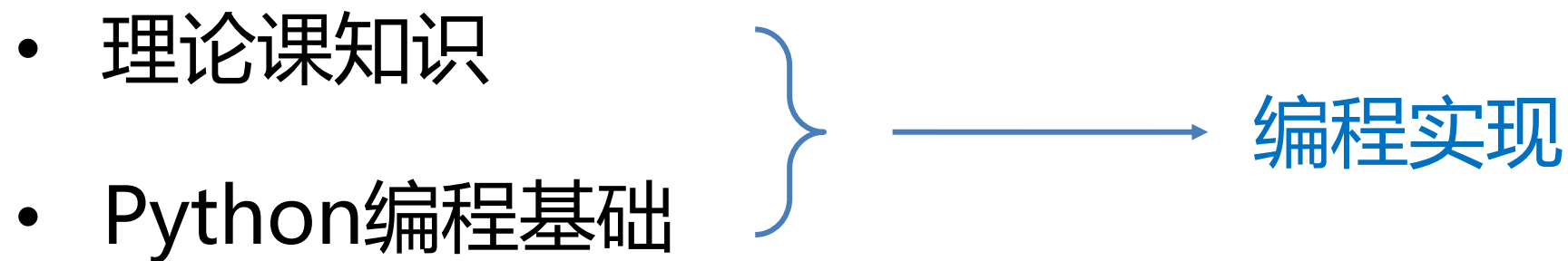
## 实验二 规则演绎系统



# 实验课补充要求



湖北大学  
HUBEI UNIVERSITY





# 实验课补充要求

- 时间允许的情况下，后续的实验会手把手带大家完成
- **要求：**
  - ✓ 完成实验前不允许做和实验无关的事
  - ✓ 下课前进行程序验收
  - ✓ 成绩由以下两点决定：1. 程序是否可以通过验收；2. 实验报告的认真程度（包括但不限于提交的文件是否为pdf格式，排版是否美观，字体是否统一，实验小结是否详细等）



## 本周实验内容

- 根据知识库编写简单的规则演绎系统
- 第一节课编写程序，第二节课自愿完成思考题
- 完成并提交实验报告（4月2日上课前交给班长）



# 实验要求



湖北大学  
HUBEI UNIVERSITY

- 知识库15条规则如下:

1. 有毛发 -> 哺乳类
2. 产奶 -> 哺乳类
3. 有羽毛 -> 鸟类
4. 不会飞, 会下蛋 -> 鸟类
5. 食肉 -> 哺乳类
6. 有犬齿, 有爪, 眼盯前方 -> 食肉类
7. 有蹄, 哺乳类 -> 蹄类
8. 反刍, 哺乳类 -> 蹄类
9. 黄褐色, 有斑点, 哺乳类, 食肉类 -> 金钱豹
10. 黄褐色, 有黑色条纹, 哺乳类, 食肉类 -> 虎
11. 有黑色条纹, 蹄类 -> 斑马
12. 有斑点, 有黑色条纹, 长脖, 蹄类 -> 长颈鹿
13. 善飞, 鸟类 -> 信天翁
14. 不会飞, 长脖, 长腿, 鸟类 -> 鸵鸟
15. 不会飞, 会游泳, 黑白二色, 鸟类 -> 企鹅



# 实验要求



湖北大学  
HUBEI UNIVERSITY

## 可输入特征如下:

- |        |        |        |          |       |
|--------|--------|--------|----------|-------|
| 1:有毛发  | 2:产奶   | 3:有羽毛  | 4:不会飞    | 5:会下蛋 |
| 6:食肉   | 7:有犬齿  | 8:有爪   | 9:眼盯前方   | 10:有蹄 |
| 11:反刍  | 12:黄褐色 | 13:有斑点 | 14:有黑色条纹 | 15:长脖 |
| 16:长腿  | 17:不会飞 | 18:会游泳 | 19:黑白二色  | 20:善飞 |
| 21:哺乳类 | 22:鸟类  | 23:食肉类 | 24:蹄类    |       |

## 需要预测结果如下:

- 25:金钱豹 26:虎 27:长颈鹿 28:斑马 29:鸵鸟 30:企鹅 31:信天翁



# 实验要求

## • 结果示例:

D:\Software\anaconda3\envs\matplotlib\python.exe D:/PycharmProjects/lesson1/

可输入的前提条件如下:

```
*****
*   1:有毛发    2:产奶    3:有羽毛    4:不会飞    5:会下蛋    *
*   6:食肉      7:有犬齿    8:有爪    9:眼盯前方    10:有蹄    *
*   11:反刍     12:黄褐色   13:有斑点   14:有黑色条纹   15:长脖    *
*   16:长腿     17:不会飞   18:会游泳   19:黑白二色   20:善飞    *
*   21:哺乳类   22:鸟类     23:食肉类   24:蹄类      *
*****
*****输入数字0开始推理*****
```

请输入前提条件的编号: 1

请输入前提条件的编号: 6

请输入前提条件的编号: 12

请输入前提条件的编号: 14

请输入前提条件的编号: 23

请输入前提条件的编号: 0

前提条件为:

食肉类 黄褐色 吃肉 有毛发 有黑色条纹

推理过程如下:

食肉 -> 哺乳类

有毛发 -> 哺乳类

黄褐色, 有黑色条纹, 哺乳类, 食肉类 -> 虎

结论:

所识别的动物为虎

Process finished with exit code 0



湖北大学  
HUBEI UNIVERSITY





- 使用制表符 `\t` 对齐输出的文本

`print('''可输入的前提条件如下:`

`*****`

`*\t1:有毛发\t2:产奶\t\t3:有羽毛\t4:不会飞\t\t5:会下蛋\t*`

`*\t6:食肉\t\t7:有犬齿\t8:有爪\t\t9:眼盯前方\t\t10:有蹄\t\t*`

`*\t11:反刍\t\t12:黄褐色\t13:有斑点\t14:有黑色条纹\t15:长脖\t\t*`

`*\t16:长腿\t\t17:不会飞\t18:会游泳\t19:黑白二色\t\t20:善飞\t\t*`

`*\t21:哺乳类\t22:鸟类\t\t23:食肉类\t24:蹄类\t\t\t\t\t*`

`*****`

`*****输入数字0开始推理*****`

`''')`





# 编程思路



湖北大学  
HUBEI UNIVERSITY

- 使用哈希表（字典）将编号和条件一一对应
- 注意python的编程规范：一行代码不能过长

```
1 hash_table = {'1': '有毛发', '2': '产奶', '3': '有羽毛', '4': '不会飞', '5': '会下蛋', '6': '吃肉', '7': '有犬齿',  
2             '8': '有爪', '9': '眼盯前方', '10': '有蹄', '11': '反刍', '12': '黄褐色', '13': '有斑点', '14': '有黑色条纹',  
3             '15': '长脖', '16': '长腿', '17': '不会飞', '18': '会游泳', '19': '黑白二色', '20': '善飞', '21': '哺乳类',  
4             '22': '鸟类', '23': '食肉类', '24': '蹄类', '25': '金钱豹', '26': '虎', '27': '长颈鹿', '28': '斑马',  
5             '29': '鸵鸟', '30': '企鹅', '31': '信天翁'}
```

```
hash_table = {'1': '有毛发', '2': '产奶', '3': '有羽毛', '4': '不会飞', '5': '会下蛋', '6': '食肉', '7': '有犬齿',  
             '8': '有爪', '9': '眼盯前方', '10': '有蹄', '11': '反刍', '12': '黄褐色', '13': '有斑点', '14': '有黑色条纹',  
             '15': '长脖', '16': '长腿', '17': '不会飞', '18': '会游泳', '19': '黑白二色', '20': '善飞', '21': '哺乳类',  
             '22': '鸟类', '23': '食肉类', '24': '蹄类', '25': '金钱豹', '26': '虎', '27': '长颈鹿', '28': '斑马',  
             '29': '鸵鸟', '30': '企鹅', '31': '信天翁'}
```



# 编程思路



湖北大学  
HUBEI UNIVERSITY

- 使用集合`set()`避免知识库里出现重复的条件

# 知识库

```
knowledge_base = set()
```

# 键盘输入

```
while(True):
```

```
    num = input('请输入前提条件的编号: ')
```

```
    if num == '0':
```

```
        break
```

```
    knowledge_base.add(num)
```

```
print('\n前提条件为: ')
```

```
for knowledge in knowledge_base:
```

```
    print(hash_table[knowledge], end="  ")
```



# 编程思路



湖北大学  
HUBEI UNIVERSITY

```
print("\n\n推理过程如下：")

# 遍历集合的过程中不能对集合进行修改
# 先将推理出的知识保存在一个临时的集合里
temp_set = set()
for knowledge in knowledge_base:
    if knowledge == '1':
        temp_set.add('21')
        print('有毛发 -> 哺乳类')
    elif knowledge == '2':
        temp_set.add('21')
        print('产奶 -> 哺乳类')
    elif knowledge == '3':
        temp_set.add('22')
        print('有羽毛 -> 鸟类')
    elif knowledge == '6':
        temp_set.add('21')
        print('食肉 -> 哺乳类')
knowledge_base = knowledge_base.union(temp_set)
```



# 学会使用python的保留字in

```
if '4' in knowledge_base and '5' in knowledge_base:  
    knowledge_base.add('22')  
    print('不会飞，会下蛋 -> 鸟类')
```

```
if '7' in knowledge_base and '8' in knowledge_base and '9' in knowledge_base:  
    knowledge_base.add('23')  
    print('有犬齿，有爪，眼盯前方 -> 食肉类')
```

```
if '10' in knowledge_base and '21' in knowledge_base:  
    knowledge_base.add('24')  
    print('有蹄，哺乳类 -> 蹄类')
```

```
if '11' in knowledge_base and '21' in knowledge_base:  
    knowledge_base.add('22')  
    print('反刍，哺乳类 -> 蹄类')
```



# flag为True代表推理失败

flag = True

```
if '23' in knowledge_base and '12' in knowledge_base and '21' in knowledge_base and '13' in knowledge_base:
```

```
    print('黄褐色, 有斑点, 哺乳类, 食肉类 -> 金钱豹')
```

```
    print('\n结论: \n所识别的动物为金钱豹')
```

```
    flag = False
```

```
if '12' in knowledge_base and '14' in knowledge_base and '21' in knowledge_base and '23' in knowledge_base:
```

```
    print('黄褐色, 有黑色条纹, 哺乳类, 食肉类 -> 虎')
```

```
    print('\n结论: \n所识别的动物为虎')
```

```
    flag = False
```

```
if '14' in knowledge_base and '24' in knowledge_base:
```

```
    print('有黑色条纹, 蹄类 -> 斑马')
```

```
    print('\n结论: \n所识别的动物为斑马')
```

```
    flag = False
```

```
if '13' in knowledge_base and '15' in knowledge_base and '16' in knowledge_base and '24' in knowledge_base:
```

```
    print('有斑点, 有黑色条纹, 长脖, 蹄类 -> 长颈鹿')
```

```
    print('\n结论: \n所识别的动物为长颈鹿')
```

```
    flag = False
```



```
if '20' in knowledge_base and '22' in knowledge_base:
    print('善飞, 鸟类 -> 信天翁')
    print('\n结论: \n所识别的动物为信天翁')
    flag = False

if '4' in knowledge_base and '5' in knowledge_base and '16' in knowledge_base and '22' in knowledge_base:
    print('不会飞, 长脖, 长腿, 鸟类 -> 鸵鸟')
    print('\n结论: \n所识别的动物为鸵鸟')
    flag = False

if '4' in knowledge_base and '18' in knowledge_base and '19' in knowledge_base and '22' in knowledge_base:
    print('不会飞, 会游泳, 黑白二色, 鸟类 -> 企鹅')
    print('\n结论: \n所识别的动物为企鹅')
    flag = False

# 若推理失败, 则输出如下信息
if flag:
    print('\n结论: \n根据现有条件无法推理出动物类别')
```



## 思考题



湖北大学  
HUBEI UNIVERSITY

- 可否把知识库的数据结构由set改为dict（哈希表/字典）？

```
knowledge_base = {'1': False, '2': False, '3': False, '4': False, '5': False, '6': False, '7': False,
                  '8': False, '9': False, '10': False, '11': False, '12': False, '13': False, '14': False,
                  '15': False, '16': False, '17': False, '18': False, '19': False, '20': False, '21': False,
                  '22': False, '23': False, '24': False, '25': False, '26': False, '27': False, '28': False,
                  '29': False, '30': False, '31': False}
```

- 输入前提条件:

```
while(True):
    num = input('请输入前提条件的编号: ')
    if num == '0':
        break
    knowledge_base[num] = True
```

- 推理:

```
if knowledge_base['4'] and knowledge_base['5']:
    knowledge_base['22'] = True
    print('不会飞, 会下蛋 -> 鸟类')
```

(空间换时间, 优化时间复杂度)





# 作业

- 安装机器学习库scikit-learn



湖北大学  
HUBEI UNIVERSITY

Not installed

Channels

Update index...

scikit-learn X

Name	T	Description	Version
<input type="checkbox"/> dask-searchcv		Tools for doing hyperparameter search with scikit-learn and dask	0.2.0
<input checked="" type="checkbox"/> scikit-learn		A set of python modules for machine learning and data mining	1.2.2
<input type="checkbox"/> scikit-learn-intelex		Intel(r) extension for scikit-learn is a seamless way to speed up your scikit-learn application.	2023.1.1

Create

Clone

Import

Backup

Remove

3 packages available matching "scikit-learn" 1 package selected

Apply

Clear



# 作业



湖北大学  
HUBEI UNIVERSITY

- 下载学习通里：资料/实验课ppt/20news-bydate.tar.gz（无需解压）
- 新建一个python文件，输入以下代码：

```
from sklearn.datasets import fetch_20newsgroups  
news = fetch_20newsgroups(data_home='./', subset='all')  
print(news.data[0])
```

- 在python文件的同目录下，新建文件夹命名为20news\_home（文件夹名称不可随意修改！），将20news-bydate.tar.gz放在该文件夹内
- 运行上述python代码

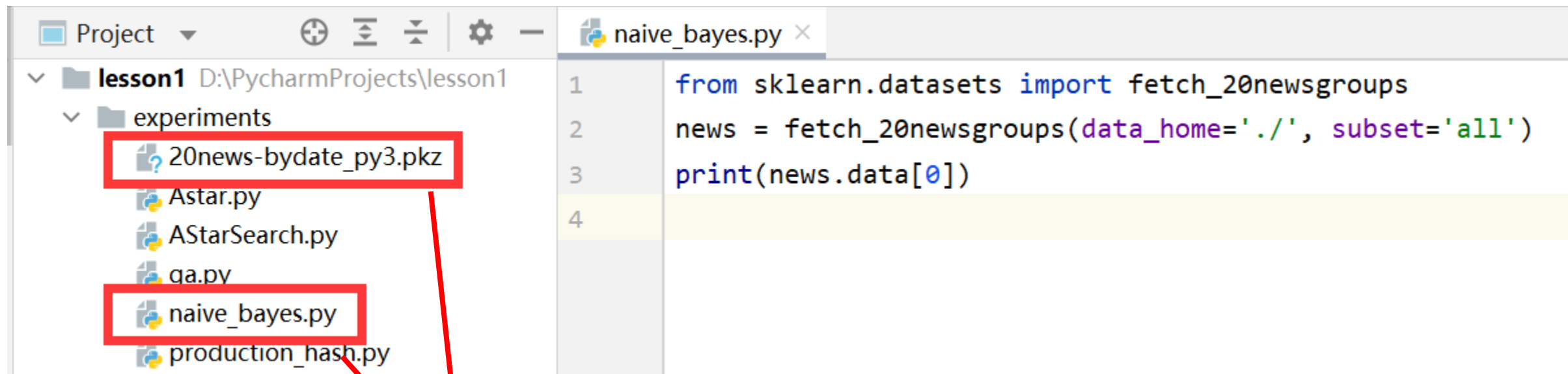


# 作业



湖北大学  
HUBEI UNIVERSITY

- 例:



相同目录下!



# 作业



湖北大学  
HUBEI UNIVERSITY

- 成功示例:

```
naive_bayes x
D:\Software\anaconda3\envs\matplotlib\python.exe D:/PycharmProjects/lesson1/experiments/naive_bayes.py
From: Mamatha Devineni Ratnam <mr47+@andrew.cmu.edu>
Subject: Pens fans reactions
Organization: Post Office, Carnegie Mellon, Pittsburgh, PA
Lines: 12
NNTP-Posting-Host: po4.andrew.cmu.edu

I am sure some bashers of Pens fans are pretty confused about the lack
of any kind of posts about the recent Pens massacre of the Devils. Actually,
I am bit puzzled too and a bit relieved. However, I am going to put an end
to non-Pittsburghers' relief with a bit of praise for the Pens. Man, they
are killing those Devils worse than I thought. Jagr just showed you why
he is much better than his regular season stats. He is also a lot
fo fun to watch in the playoffs. Bowman should let JAgr have a lot of
fun in the next couple of games since the Pens are going to beat the pulp out of Jersey anyway. I was
very disappointed not to see the Islanders lose the final
regular season game.          PENS RULE!!!

Process finished with exit code 0
```

# 结束语



# 谢谢!