

204789: Machine Learning and Neural Network

2/59

The logo for the Julia programming language. It features the word "julia" in a dark grey, lowercase, sans-serif font. The letter 'j' has a blue circle above it. The letter 'i' has a red circle above it. The letter 'l' has a green circle above it. The letter 'i' has a purple circle above it. The letter 'a' has a green circle above it.

julia

Defining a function

- สามารถนิยามโดย

```
function name(arg1, arg2, ... )
```

```
    # function body
```

```
    return ret1, ret2, ...           # สามารถ return ได้มากกว่า 1 ค่า
```

```
end
```

- Function call

```
[a, b, c ... ] = name(arg1, arg2, ...)
```

Julia Machine Learning Kernel Package

- สามารถใช้ได้กับ v0.4.x

- ติดตั้ง

```
Pkg.add("MLKernels")
```

- เรียกใช้งาน

```
using MLKernels
```

- สร้าง kernel function

```
\psi = GaussianKernel(2.0) # สร้าง GaussianKernel (a.k.a RBF kernel) ที่มี width=2.0
```

- สร้าง Gram's matrix (kernel matrix)

```
X = rand(3,2)
```

```
K = kernelmatrix(\psi, X)
```

Julia Machine Learning Kernel Package

- Kernel ที่รองรับ
 - Exponential Kernel, Squared Exponential Kernel, Gamma Exponential Kernel
 - Rational-Quadratic Kernel, Gamma-Rational Kernel, Matern Kernel
 - Linear Kernel, Polynomial Kernel, Exponentiated Kernel
 - Periodic Kernel, Sigmoid Kernel

- For more info see

<http://mlkernels.readthedocs.io/en/latest/>

Exercise 1

- ทดลองใช้งาน GaussianKernel ใน package MLKernels
- สร้างฟังก์ชันเพื่อคำนวณ GaussianKernel
- เปรียบเทียบผลลัพธ์ของ Kernel Matrix ที่ได้จากฟังก์ชันของตน กับฟังก์ชันที่มีมากับ package

SVM + Kernel

- Dual SVM + Kernel
 - Very natural
- Primal SVM + Kernel
 - Also possible
 - เริ่มจากการหา Kernel matrix
 - แล้วใช้แต่ละแถวของ Kernel matrix แทนข้อมูลแต่ละตัว

Example

```
srand(123)  
x = rand(5,2)
```

5x2 Array{Float64,2}:

0.768448	0.662555
0.940515	0.586022
0.673959	0.0521332
0.395453	0.26864
0.313244	0.108871

X_1

```
k1 = kernelmatrix(K,x)
```

5x5 Array{Float64,2}:

1.0	0.931528	0.466226	0.55511	0.357882
0.931528	1.0	0.490574	0.451289	0.288724
0.466226	0.490574	1.0	0.779673	0.765927
0.55511	0.451289	0.779673	1.0	0.937472
0.357882	0.288724	0.765927	0.937472	1.0

Exercise 2

- ลองสร้าง Kernel matrix แล้ว classify ด้วย SVM ดู
- ให้ลองใช้ข้อมูล banana.mat
- สามารถอ้างอิงการอ่าน MATLAB file ได้จากสไลด์เก่า

Naive Bayes

- Sentiment Analysis
- จำแนกความคิดเห็นที่มี ต่อสินค้าหรือบริการ (Positive vs Negative)
- เรียกใช้ Package TextAnalysis เพื่อช่วยจัดการข้อมูล
 - `Pkg.add("TextAnalysis")` and `using TextAnalysis`

Naive Bayes

- สร้าง Corpus (text dataset) ของ ความเห็นลบ และความเห็นบวก
- โดยอ้างอิงข้อมูลจากไฟล์
 - negative.txt และ positive.txt
- nCrps = Corpus(Any[FileDocument("./negative.txt")])
- pCrps = Corpus(Any[FileDocument("./positive.txt")])

Some preprocessing

- Remove numbers
- Convert to lowercase
- Remove preposition [in, on, at]

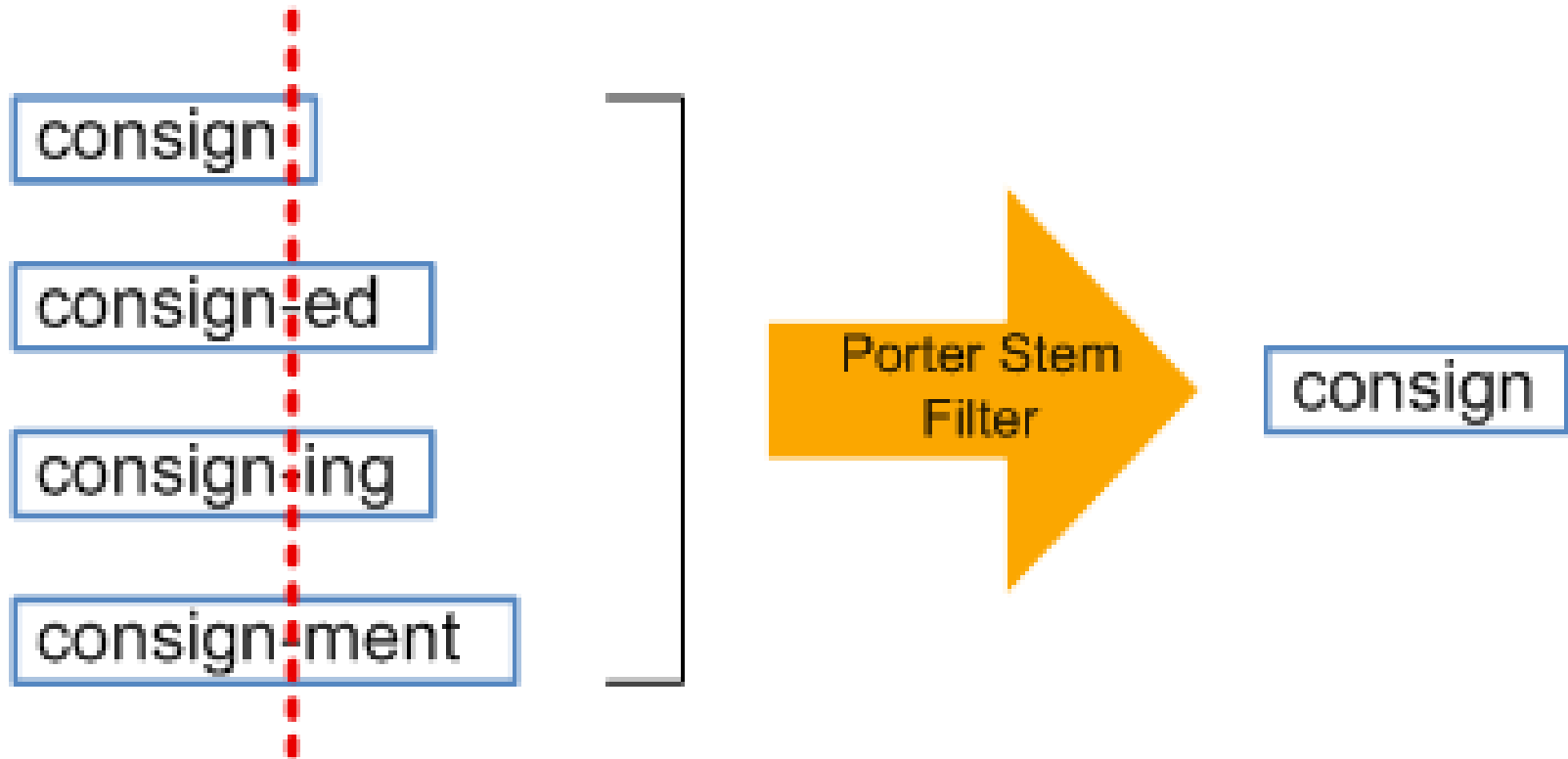
Some preprocessing

- Remove stop words

Begins with...	Stop words
A	about, after, all, also, another, any, are, as, at
B	be, because, been, before, being, between, but, both, by
C	came, can, come, could
D	did, do, does
E	each, else
F	for, from
G	get, got
H	has, had, he, have, her, here, him, himself, his, how
I	if, in, into, is, it, its
J	just
L	like
M	make, many, me, might, more, most, much, must, my
N	never, no, now
O	of, on, only, other, our, out
S	said, same, see, should, since, so, some, still, such
T	take, than, that, the, their, them, then, there, these, they, this, those, through, to, too
U	under, up, use
V	very
W	want, was, way, we, well, were, what, when, where, which, while, who, will, with, would
Y	you, your

Some preprocessing

- Stemming using Porter's stemmer



All the steps

- # preprocessing nCrps
- remove_numbers!(nCrps)
- remove_case!(nCrps)
- remove_prepositions!(nCrps)
- remove_stop_words!(nCrps)
- remove_punctuation!(nCrps)
- stem!(nCrps)
- update_lexicon!(nCrps)

The processed corpus

- เราสามารถหาจำนวนครั้งที่เจอคำนั้นๆได้

`lexicon(nCrps)[“word”]` # `lexicon(nCrps)` คือดิกชันนารี [] ใช้เข้าถึงดิกชันนารี

- For example หากต้องการดูว่า ความน่าจะเป็นที่จะพบคำว่า happy ใน class positive เป็นเท่าไร

`lexicon(pCrps)[“happy”]`

- แต่ถ้าไม่มีคำนั้นอยู่ใน corpus จะเกิด error
- เราอาจต้องเชคก่อนว่ามี key นั้นอยู่หรือไม่โดยใช้ `haskey(lexicon(pCrp),key)`

Exercise 3

- ทำนายว่า รีวิวต่อไปนี้ เป็นรีวิวด้านบวก หรือด้านลบ

“This movie is kinda slow at first but then it starts to get interesting. My heart was litterly beating outa my chest watching it. If you like sick and twisted movies this one is just for you.”

- โดยใช้ Naive Bayes classifier
- อย่าลืม preprocess ประโยคข้างต้นให้เหมือนกับ corpus

Distributions Package

- ติดตั้ง Package

```
Pkg.add("Distributions")
```

- สร้าง distribution ขึ้นมา

```
using Distributions
```

```
myDist = Normal()
```

- สุ่มค่า 100 ค่า จาก distribution ที่สร้างขึ้นมา

```
samples = rand(myDist, 100)
```

- ประมาณค่าพารามิเตอร์ `result = fit(Distribution, data)`

- `Result = fit(Normal, samples)`

Distributions Package

- คำนวณ probability density ณ ตำแหน่ง x

`pdf(Distribution, x)`

- Example

`pdf(myDist, x)`

- จะคืนค่า likelihood กลับมา

Linear Discriminant Analysis

- Binary (boston dataset)
 - Calculate likelihood using Gaussian distribution in Distributions package.
 - Estimate parameters of the Gaussian.
 - Calculate class posterior probability
 - Make prediction
- Multiclass (wine dataset)
 - Same but there are more than 2 class posteriors

Exercise 4

- Implement LDA for the given binary and multiclass classification problems.