Rcpp包开发：Rcppist

目录

CONTENTS

第一部分

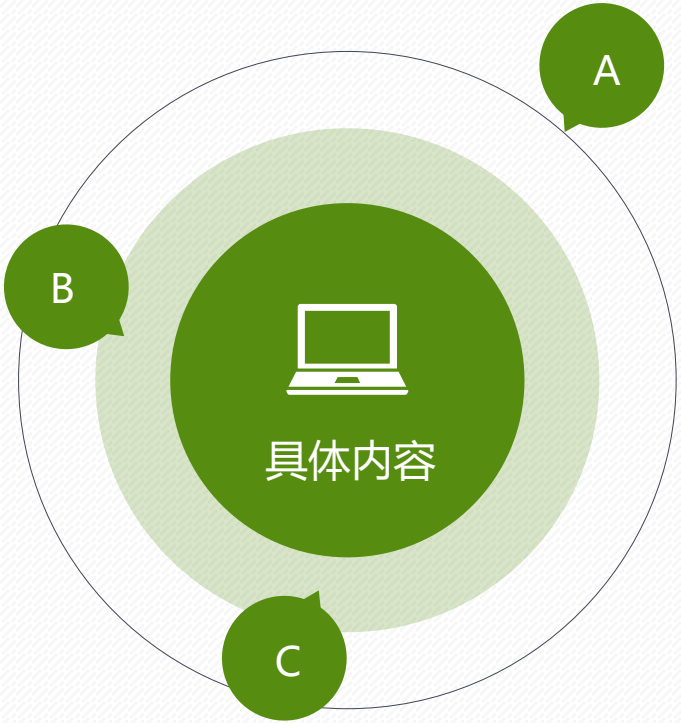
INTRODUCTION 包的说明

包的说明

选题背景及意义

R语言虽然在数据统计和绘图方面有着很大的优势，但是运行速度慢是它最大的缺点，特别是针对数据量大的情况。利用C++底层运行速度优势，使用R语言调用C++函数可以有效的弥补R的这一个缺陷。

3个标准化



Z标准化

Z-Score标准化方法

L2标准化

使得Xi归一化到范数l2

Softmax变换

Softmax函数是logistic函数的一种泛化。Softmax - 用于多分类神经网络输出。

Z-Score标准化方法

标准正态分布

当 $\mu = 0, \sigma = 1$ 时，正态分布就成为**标准正态分布**

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{\left(-\frac{x^2}{2}\right)}$$

若 $X \sim N(\mu, \sigma^2), Y = \frac{X - \mu}{\sigma} \sim N(0,1)$

给予原始数据的均值（mean）和标准差（standard deviation）进行数据的标准化(x-u)/std。经过处理的数据符合标准正态分布，即均值为0，标准差为1。

L2标准化

L2范数

L2的公式：

欧几里德距离之和

$$S=\sum_{i=1}^{n}(y_i-f(x_i))^2$$

就是样本和标签之差的平方之和

向量$x(x_1,x_2,...,x_n)$的L^2范数定义为：

$$\text{norm}(x)=\sqrt{x_1^2+x_2^2+...+x_n^2}$$

要使得x归一化到单位L^2范数，即建立一个从x到x'的映射，使得x'的L^2范数为1

则：

$$1=\text{norm}(x')=\frac{\sqrt{x_1^2+x_2^2+...+x_n^2}}{\text{norm}(x)}$$

$$=\sqrt{\frac{x_1^2+x_2^2+...+x_n^2}{\text{norm}(x)^2}}$$

$$=\sqrt{\left(\frac{x_1}{\text{norm}(x)}\right)^2+\left(\frac{x_2}{\text{norm}(x)}\right)^2+...+\left(\frac{x_n}{\text{norm}(x)}\right)^2}$$

$$=\sqrt{x_1'^2+x_2'^2+...+x_n'^2}$$

即：$x_i'=\frac{x_i}{\text{norm}(x)}$

softmax函数

假设我们有一个数组，V，Vi表示V中的第i个元素，那么这个元素的Softmax值就是

$$S_j = \frac{e^{a_j}}{\sum_{k=1}^{T} e^{a_k}}$$

该元素的指数，与所有元素指数和的比值。
总和为1，表示该样本属于各个类的概率。

第二部分

DEMONSTRATION 包的演示

Rcppist演示

安装Rcppist

#下载安装Rtools https://cran.r-project.org/bin/windows/Rtools/
#Rstudio中运行
system('where make') #显示路径则安装正确
library("Rcpp") #install.packages("Rcpp") #若无则安装
library("devtools") #install.packages("devtools") #若无则安装
install_github("laohur/Rcppist")

使用Rcppist zlise为例

X<-runif(100)*100
X
Y<-Rcppist::zlise(X)
Y
#Y即X的z分数

~/Rcppist-master/

```
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> system('g++ -v')
> system('where make')
C:\Rtools\bin\make.exe
> > library("Rcpp")
Error: unexpected '>' in ">"
>
> library("Rcpp")
>
> library("devtools")
> install_github("laohur/Rcppist")
Downloading GitHub repo laohur/Rcppist@master
from URL https://api.github.com/repos/laohur/Rcppist/zipball/master
Installing Rcppist
"C:/PROGRA~1/R/R-35~1.0/bin/x64/R" --no-site-file --no-environ --no-save --no-restore --quiet CMD  \
  INSTALL "C:/Windows/Temp/RtmpikAS4T/devtools205c13336204/laohur-Rcppist-b57f757"  \
  --library="C:/Users/zen/Documents/R/win-library/3.5" --install-tests

* installing *source* package 'Rcppist' ...
** libs
c:/Rtools/mingw_64/bin/g++ -shared -s -static-libgcc -o Rcppist.dll tmp.def RcppExports.o cosineSimilarity
.o csort.o l2lise.o rcpp_hello_world.o rcpp_zlise.o relu.o softmax.o tanh.o zlise.o -LC:/PROGRA~1/R/R-35~1
.0/bin/x64 -lR
installing to C:/Users/zen/Documents/R/win-library/3.5/Rcppist/libs/x64
** R
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
  converting help for package 'Rcppist'
    finding HTML links ... 好了
    Rcppist-package                         html
    rcpp_hello_world                        html
** building package indices
** testing if installed package can be loaded
* DONE (Rcppist)
In R CMD INSTALL
> Rcppist::zlise(c(1,3,6,9))
[1] -0.6185896 -0.2886751  0.2061965  0.7010682
>
```

正确加载

自动补全

To Console | To Source

```
library("devtools")
install_github("laohur/Rcppist")
library("Rtools")
system('g++ -v')
system('where make')
system('where make')
system('g++ -v')
system('where make')
> library("Rcpp")
library("Rcpp")
library("devtools")
install_github("laohur/Rcppist")
Rcppist::zlise(c(1,3,6,9))
```

Files | Plots | Packages | Help | Viewer

softmax.Rd | Find in Topic

Rcppsoftmax {Rcppist} R Documen

Rcppsoftmax

帮助手册

Description

The Softmax function is a generalization of the logistic function. Softmax - for multi classification neural net output. The two type of classification can also be used, but if you export only one neuron, you can use the SIGMOD function. That is to say, the ratio of the index of the element to the index sum of all elements.

Details

It is called the softmax function, because it represents a smooth version of the "Max" function.

Through the softmax function, the range can be [0,1]. In regression and classification problems, it is usually parameter to be sought, and the maximum parameter can be found as the best parameter.

In addition, the Softmax function can also be used for nonlinear estimation, when parameters can be replac other column vectors in real sense.

The Softmax function gets the value of a [0,1], and the probability that the softmax is calculated is the true probability; in other words, the probability is equal to the expectation.

Examples

```
Rcppsoftmax()
```

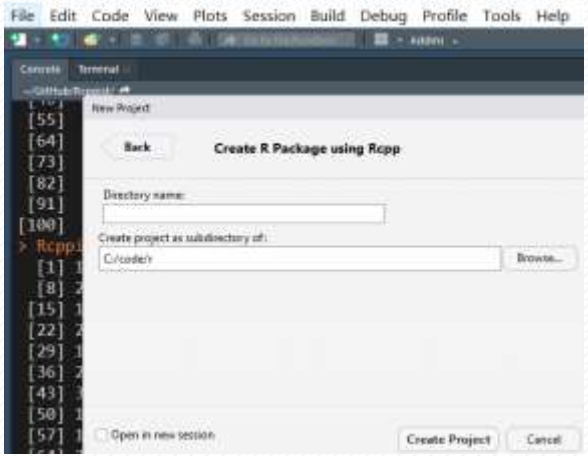
[Package *Rcppist* version 1.0]

第三部分

DEVELOPMENT

包的开发

Rcpp包的开发

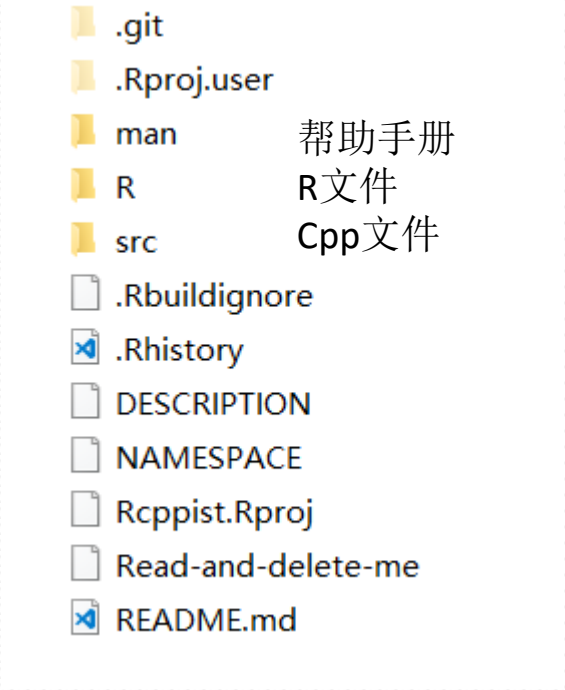


1.安装依赖

2.Rstudio新建包 编译

3. 编写其中的cpp文件 编译

4.发布到github



Cpp编写

install.packages(“Rcpp”)

Rstudio new zlise.cpp
Rstudio source()

x=c(46,8,79,324,12,98,-5,-34,23)
y=zlise(x)

print(y)

zlise.cpp

```cpp
#include <Rcpp.h>
#include <cmath>

using namespace Rcpp;

// [[Rcpp::export]]
NumericVector zlise (const NumericVector & X) {
    int n=X.size();
    NumericVector Y(n);
    double sum=0;
    for(int i=0; i<n; i++){
        sum+=X[i];
    }
    double avg=sum/n;
    double variance=0;
    for(int i=0; i<n; i++){
      variance+=pow((X[i]-avg),2);
    }
    variance=sqrt(variance);
    for(int i=0; i<n; i++){
        Y[i]=(X[i]-avg)/variance;
    }
    return Y;
}
```

第四部分

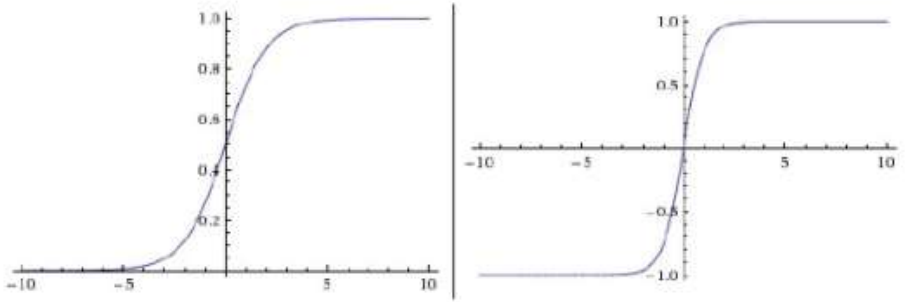
PRACTICE 练习

练习1： tanh（ ）

$$f(z) = \frac{1}{1 + \exp(-z)}.$$

$$f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}},$$

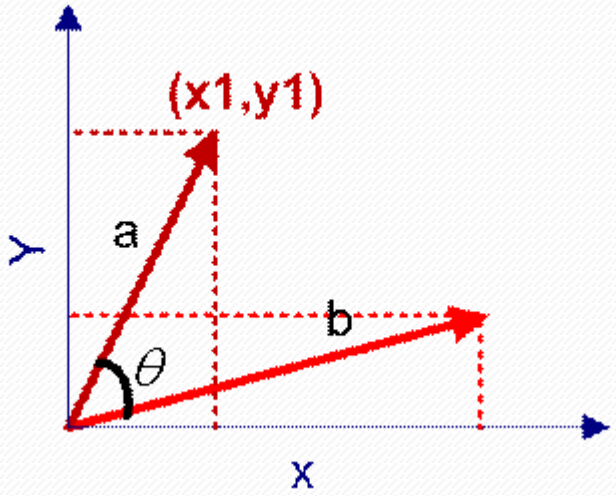
双切正切函数，取值范围为[-1,1]。tanh在特征相差明显时的效果会很好，在循环过程中会不断扩大特征效果。与 sigmoid 的区别是，tanh 是 0 均值的，因此实际应用中 tanh 会比 sigmoid 更好。
提示：#include <cmath> 自带tanh()



Left: Sigmoid non-linearity squashes real numbers to range between [0,1] **Right:** The tanh non-linearity squashes real numbers to range between [-1,1].

练习2:cosineSimilarit

向量余弦相似度量
两组向量，通过计算方向夹角的余弦值，判断相似关系



$$\cos(\theta) = \frac{\sum_{i=1}^{n}(x_i \times y_i)}{\sqrt{\sum_{i=1}^{n}(x_i)^2} \times \sqrt{\sum_{i=1}^{n}(y_i)^2}}$$

$$= \frac{a \bullet b}{||a|| \times ||b||}$$

谢谢观看