Weekly Report of Research Work

WR-ABS-TEMP-2015A-No.001

汤吉(Ji TANG)

Number: WR-ABS-TEMP-2015A, E-mail: tangji08@hotmail.com

Date: 15/11/2015 - 22/11/2015

November 22, 2015

Contents

1 Work

This week, I have read the two version (in EN and CN) of the book "Machine Learning In Action" about the fundamental concepts of k-Nearest Neighbors method. And we have collected the basis information about "Machine Learning" and "Big Data", then have make a big list of them of 6 aspects "Journals","Person","Directions", "Open Source Software","Classic Review" and "Reference". Because these things are so difficult for us to search by using "Baidu", so we have studied how to get a VPS and Open a VPN, it's not really easy. And finally, I have rewrite my C language code about the Integral Method "Gauss-Legendre" refer to your code.

2 The basis concept of kNN

2.1 Description

k-Nearest Neighbors (kNN) works like this: we have an existing set of example data, our training set. We have labels for all of this data-we know what class each piece of the data should fall into. When we are given a new piece of data without a label, we compare that new piece of data to the existing data, every piece of existing data. We then take the most similar pieces of data (the nearest neighbors) and look at their labels. We look at the top k most similar pieces of data from our known dataset, this is where the k comes from. (k is an integer and it's usually less than 20.)Lastly, we take a majority vote from the k most similar pieces of data, and the majority is the new class we assign to the data we were asked to classify.

2.2 General Approach

1. Collect:Any method.
2. Prepare:Numeric values are needed for a distance calculation.A structured data format is best.
3. Analyze:Any method.
4. Train:Does not apply to the kNN algorithm.
5. Test:Calculate the error rate.
6. Use:This application needs to get some input data and output structured numeric values.Next, the application runs the kNN algorithm on this input data and determines which class the input data should belong to. The application then takes some action on the calculated class.

3 The basis concept of BigData

Big data is a broad term for data sets so large or complex that traditional data processing applications are inadequate. Challenges include analysis, capture, data curation, search, sharing, storage, transfer, visualization, and information privacy. The term often refers simply to the use of predictive analytics or other certain advanced methods to extract value from data, and seldom to a particular size of data set. Accuracy in big data may lead to more confident decision making. And better decisions can mean greater operational efficiency, cost reduction and reduced risk.

Analysis of data sets can find new correlations, to "spot business trends, prevent diseases, combat crime and so on." Scientists, business executives, practitioners of media, and advertising and governments alike regularly meet difficulties with large data sets in areas including Internet search, finance and business informatics. Scientists encounter limitations in e-Science work, including meteorology, genomics,connectomics, complex physics simulations,and biological and environmental research. [1]

4 Set up VPN on VPS

At first, we have purchased a VPS with configuration of "10 GB SSD RAID-10 DISK Space, 256 RAM, 500 Transfer". And then it took us a little long time to understand how to open the VPN of PPTP service. But finally, we have succeed !

5 Gauss-Legendre

After the guidance of Pro. H.Y.Zhang, I have rewritten my C language code:

```
 /*
============================================================================
Name        : Intergral_Legendre.c
Author      : 汤吉 Tangji
Version     : 2.0
Copyright   : 2015 Ji Tang <tangji08@hotmail.com>
Description : Gauss Legendre integral method function
Function defined: double Gauss_legendre()
Function usage:
   1.You should input 4 parameters:double f(double),double a,double b,int n
     Then the function will return a double result (the value of this integral)
   2.Used by double Gauss_legendre()
```

```
 ================================================================================
 */

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#include "Polynome_Legendre.h"

double Gauss_legendre(double a, double b, int m, double f(double))
{
    double x, Root_x, Half_range, Average, sum = 0;
    Half_range = (b-a) /2;
    //Define the amplitude of a and b
    Average = (a+b) /2;
    //Define the half value between a and b
    for ( int i = 1; i <= m; i++)
    {
      x = cos(M_PI*(double)(i-0.25)/(double)(m+0.5));
      //To find the root by using the method of Newton
      if (x != Root_x){
        Root_x = x;
        x -= Poly(x,m) / DPoly(x,m);
          }
  Root_x = DPoly(x,m);
    sum+= f(Half_range*x+Average) * 2.0 / ( ( 1- x*x ) * Root_x * Root_x );
    }
    return sum*Half_range;
}

//Define the integral function "f = X^2"
double f(double x){
  return x*x;
}

int main(){
  double result,a,b;
  int p;
  printf("You want to integral \"X^2\" from _ to _ by _ ranks?\n");
  scanf("%lf %lf %d",&a,&b,&p);
  result = Gauss_legendre(a,b,p,f);
  printf("The integral is  %lf  by the method of Gauss Legendre\n",result);
```

```
  printf("Do you want to continue?(Y/N)\n");
  char C = ' ';
  while((C<'a'|C>'z') & (C<'A'|C>'Z'))
    {
      C = getchar();
    }
    //Get a char until the user input a letter from "a" to "z" or "A" to "Z"
    if(C == 'Y'|C == 'y'){
      main();
    }else{ return 0;}
    //If the user choose "Y" or "y",repeat this program.Otherwise,the program is
        over!
}
```

```
 /*
 ============================================================================
 Name        : Polynome_Legendre.h
 Author      : 汤吉 Tangji
 Version     : 2.0
 Copyright   : 2015 Ji Tang <tangji08@hotmail.com>
 Description : Gauss Legendre integral method function
 Function defined: 1.double Poly()
                   2.double DPoly()
 Function usage:
       You can use this head file to obtain the polynome and the derivpolynome
          of Gauss-Legendre.
 ============================================================================
 */

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

double Poly( double x, int n)
{
  double P0 = 1.0;    // for P0(x)
  double P1 = x;      // for P1(x)
  double Pn;
    switch(n) {
    case 0:Pn = P0;break;
    case 1:Pn = P1;break;
```

```
    default:
      for(int i = 1; i < n; i++){
        // iterating via the formula
        Pn = ((2*i -1.0)*x*P1 - (i -1.0)*P0 )/i;
        P0 = P1;    // Renew P0
        P1 = Pn;    // Renew P1
        }
      break;
    }
  return Pn;
}

double DPoly( double x, int n)
{
  //Calculate the derivative of Lengdre Polynomial
  return (double)n*(x*Poly(x,n)-Poly(x,n-1.0))/(x*x-1.0);
}
```

Although there are some bugs, the next week I'll continue to debug them.

But the most important thing is that I can conclude from your code that I should firstly make all the variances clearly, and try my best to let each function or formula distinct. For example, it's better for me to write the expression

```
x=cos(M\_PI*(i+0.75)/(m+0.25));
```

In this way

```
x = cos( M\_PI * ( i + 0.75 ) / ( m + 0.25 ) );
```

Which could make me feel more comfortable.

References

[1] Wikipedia. International conference on machine learning — wikipedia, the free encyclopedia, 2015. [Online; accessed 18-November-2015].