

DISTRIBUCION BINOMIAL

Programa desarrollado en lenguaje de programación c para calcular la distribución binomial.

La formula dada para conocer la probabilidad de la distribución binomial es la siguiente

Probabilidad de obtener k-éxitos

$$P(X = k) = \binom{n}{k} \cdot p^k q^{n-k}$$

n es el número de pruebas.

k es el número de éxitos.

p es la probabilidad de éxito.

q es la probabilidad de fracaso.

El programa se compone por 4 funciones

```
double factorial(int );  
double comb(int, int);  
double potencia(float, int);  
double distBin(int, int, float);
```

La función **distBin** es la función principal en ella se lleva a cabo el procedimiento para obtener la distribución.

A continuación se detalla cada una de las funciones empleadas:

FACTORIAL

```
// funcion para calcular el factorial  
double factorial (int numero)  
{  
    if (numero <= 1)  
        return 1;  
    else  
        //se hace el calculo de forma recursiva  
        return numero * factorial (numero - 1);  
}
```

En esta función calcula números factoriales de forma recursiva que es empleada posteriormente para calcular combinaciones.

COMBINACIONES

//funcion para calcular el numero de combinaciones

```
double comb(int n, int k){  
    int combinaciones;  
  
    combinaciones = factorial(n) / (factorial(n-k) * factorial(k));  
  
    return combinaciones;  
  
}
```

Esta función se utiliza para calcular el numero de combinaciones dadas por el numero de eventos (n) tomados por el numero de casos de exito (k).

POTENCIA

//funcion para calcular potencias

```
double potencia(float base, int exponente){  
    int i;  
    float acu;  
  
    acu = base;  
  
    if(exponente >= 2)  
        for(i = 0; i < exponente - 1; i++){  
            acu *= base;  
        }  
  
    return acu;  
  
}
```

Esta función se utiliza para calcular potencias que son utilizadas como se describe en la formula.

DISTRIBUCION BINOMIAL

```
double distBin(int x, int n, float p){  
    float pmf;  
  
    pmf = (comb(n, x) * potencia(p, x)) * potencia(1 - p, n - x);  
  
    return pmf;  
  
}
```

Por ultimo se tiene la función de la distribución binomial que lo que hace prácticamente es llevar a cabo el procedimiento dado por la formula.

```
// Guillermo Islas Arreola
```

```
//calculo de la probabilidad binomial
```

```
/*Definicion de variables  
n = es el numero de eventos  
x= es el numero de casos de exito  
p = es el numero de proobabilidades  
*/
```

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

```
double factorial(int );  
double comb(int, int);  
double potencia(float, int);  
double distBin(int, int, float);
```

```
int main(){
```

```
printf("%f\n",distBin(*,*,*));
```

```
return 0;  
}
```

```
// funcion para calcular el factorial
```

```
double factorial (int numero)  
{  
    if (numero <= 1)  
        return 1;  
    else  
        //se hace el calculo de forma recursiva  
        return numero * factorial (numero - 1);  
}
```

```
//funcion para calcular el numero de combinaciones
```

```
double comb(int n, int k){  
  
    int combinaciones;  
  
    combinaciones = factorial(n) / (factorial(n-k) * factorial(k));  
  
    return combinaciones;  
  
}
```

//funcion para calcular potencias

double potencia(float base, int exponente){

int i;

float acu;

acu = base;

if(exponente >= 2)

for(i = 0; i < exponente - 1; i++){

acu *= base;

}

return acu;

}

double distBin(int x, int n, float p){

float pmf;

pmf = (comb(n, x) * potencia(p, x)) * potencia(1 - p, n - x);

return pmf;

}