

Desafío de código 06



Aviso Legal

Al leer este problema, acepto no copiar código de fuente alguna (incluidos sitios web, libros o amigos y colegas) para completar este desafío. También acepto no compartir este desafío de código por ningún medio. Sin embargo, puedo hacer referencia a la documentación del lenguaje de programación o usar un IDE que ofrezca funciones de auto-completar.

Desafío de código

Evaluación de riesgos

Algunas estadísticas y datos son necesarios para evaluar el riesgo de incumplimiento de nuevos clientes; estos son la desviación estándar de pagos atrasados, pagos tardíos inusuales y probabilidad de pagos tardíos.

Un pago tardío inusual se detecta cuando se presenta un pago con un numero de días de atraso muy alto (*numero de días después de la fecha limite de pago*), y el numero de días de atraso en el periodo *anterior* y *siguiente* es **menor**; gráficamente, se vería un pico. Para este ejercicio necesitamos la anomalía con el pico más alto.

Adicionalmente se debe calcular la probabilidad de pagos tardíos basado en los pagos tardíos de otros productos que ya posee el cliente para un numero de periodos de tiempo.

Entrada

Tanto para el cálculo de desviación estándar como para el pago tardío inusual, recibirás un arreglo de enteros con el numero de días de retraso en los pagos.

Cada posición del arreglo representa un periodo de pago, usualmente un mes, significa que no hubo retraso en el pago, significa un día de retraso en el pago, y así sucesivamente.

Para calcular la probabilidad de pagos tardíos, recibirás un arreglo de dos dimensiones, donde cada fila representa los pagos tardíos de un producto para distintos periodos de tiempo (*columnas*).

Salida esperada

Salida esperada para la desviación estándar

La salida esperada para la desviación estándar es un valor tipo `double`, el cual representa el resultado del cálculo de la desviación estándar. (*La desviación estándar poblacional*).

Salida esperada para pago tardío inusual

La salida esperada es un dato tipo `int` que representa el índice en el arreglo de entrada donde se encuentra el pago anómalo con el pico mas alto.

Salida esperada para cálculo de probabilidad de pagos tardíos

La salida esperada para la probabilidad de pagos tardíos es un arreglo de `double` donde cada posición representa la probabilidad de pago tardío para el periodo (*índice*) de los diferentes productos (*filas del arreglo*)

Ejemplo

Ejemplo de entrada

```
arreglo con retrasos en el pago:
{0, 15, 2, 0, 1, 3}
[0 1 2 3 4 5] //índice
```

Ejemplo de entrada para cálculo de probabilidad de pago tardío

```
arreglo de dos dimensiones, cada fila es un producto, cada columna es un periodo de tiempo.
{
    {0, 3, 6, 1, 0, 5, 0, 0}, //producto 1
    {0, 3, 0, 2, 0, 0, 0, 2}, //producto 2
    {0, 0, 1, 0, 3, 0, 2, 0}, //producto 3
    {0, 4, 0, 2, 0, 1, 1, 0}  //producto 4
    [0 1 2 3 4 5 6 7] //índice (periodo)
}
```

Ejemplo de salida

Desviación estándar

```
5.251983752
```

Pago anómalo

```
1 //índice del arreglo donde se encuentra el pago anómalo con el pico mas alto.
```

Probabilidad de pago tardío

```
{0.0, 0.75, 0.5, 0.75, 0.25, 0.5, 0.5, 0.25}
```

Explicación

La desviación estándar para estos números `{0, 15, 2, 0, 1, 3}` es `5.251983752`.

La anomalía con el pico mas alto para el arreglo se encuentra en el índice `1` debido a que `15 > 0` y `15 > 2` el índice 5 contiene un pico pero `3` es menor que `15`.

Para el cálculo de probabilidad de pago tardío, como explicación veamos el cálculo para los dos primeros periodos (*índices 0 y 1 del arreglo de dos dimensiones*). Para el primer periodo (*índice 0*) vemos que no hubo pagos tardíos en ninguno de los `4` productos, entonces el resultado es `0.0` $0/4 = 0.0$. Para el segundo periodo (*índice 1*) vemos tres pagos tardíos (`3,3,4`), lo que nos da una probabilidad de `0.75` $3/4 = 0.75$.

Como completar este desafío

Para completar este desafío, debes completar los métodos `standardDeviation`, `paymentDelayMaxPeakIndex` y `latePaymentProbabilityByPeriod` dentro de la clase `CreditRiskAssessment`. Una vez completados los métodos, las pruebas unitarias de la clase `CreditRiskAssessmentTest` deben pasar.

Nota: Documenta tu código usando Javadoc para obtener puntos extra.

Buena suerte!