Segundo examen parcial

Jorge Alejandro Rodriguez Aldana

Escuela de Ciencias físicas y matemáticas

4 de mayo de 2021

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.
  - Plotear los datos con Gnuplot

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.
  - Plotear los datos con Gnuplot
- Un ajuste de una recta que mejor se aproximara

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.
  - Plotear los datos con Gnuplot
- Un ajuste de una recta que mejor se aproximara
  - Agregar una función a Gnuplot y hacerle un *fit*

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.
  - Plotear los datos con Gnuplot
- Un ajuste de una recta que mejor se aproximara
  - Agregar una función a Gnuplot y hacerle un *fit*
  - Programar el método de mínimos cuadrados

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.
  - Plotear los datos con Gnuplot
- Un ajuste de una recta que mejor se aproximara
  - Agregar una función a Gnuplot y hacerle un *fit*
  - Programar el método de mínimos cuadrados
- Obtener la aceleración aproximada

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.
  - Plotear los datos con Gnuplot
- Un ajuste de una recta que mejor se aproximara
  - Agregar una función a Gnuplot y hacerle un *fit*
  - Programar el método de mínimos cuadrados
- Obtener la aceleración aproximada
  - La aceleración es igual a la pendiente estimada

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.
  - Plotear los datos con Gnuplot
- Un ajuste de una recta que mejor se aproximara
  - Agregar una función a Gnuplot y hacerle un *fit*
  - Programar el método de mínimos cuadrados
- Obtener la aceleración aproximada
  - La aceleración es igual a la pendiente estimada
- Estimar la velocidad en $t = 15s$

Planteamiento

Dada una tabla de datos de velocidad y tiempo de un objeto se nos pedía:

- Una gráfica que comparara los valores
  - Tabular los datos en un archivo de texto, y agregar una columna con el error de tiempo.
  - Plotear los datos con Gnuplot
- Un ajuste de una recta que mejor se aproximara
  - Agregar una función a Gnuplot y hacerle un *fit*
  - Programar el método de mínimos cuadrados
- Obtener la aceleración aproximada
  - La aceleración es igual a la pendiente estimada
- Estimar la velocidad en $t = 15s$
  - La velocidad dependiente del tiempo está dada por la ecuación de la recta, solo hace falta valuarla en $t = 15s$

Mínimos cuadrados

Se busca aproximar las constantes m y b para la ecuación de la recta $y(x) = mx + b$

Mínimos cuadrados

Se busca aproximar las constantes m y b para la ecuación de la recta $y(x) = mx + b$

$$m = \frac{n\sum_{k=1}^{n}(x_k y_k) - \sum_{k=1}^{n} x_k \sum_{k=1}^{n} y_k}{n\sum_{k=1}^{n} x_k^2 - \left(\sum_{k=1}^{n} x_k\right)^2} \tag{1}$$

Mínimos cuadrados

Se busca aproximar las constantes m y b para la ecuación de la recta $y(x) = mx + b$

$$m = \frac{n\sum_{k=1}^{n}(x_k y_k) - \sum_{k=1}^{n} x_k \sum_{k=1}^{n} y_k}{n\sum_{k=1}^{n} x_k^2 - \left(\sum_{k=1}^{n} x_k\right)^2} \tag{1}$$

$$b = \frac{n\sum_{k=1}^{n} y_k - m\sum_{k=1}^{n} x_k}{n} \tag{2}$$

Programación mínimos cuadrados

Funciones principales

- main(): Imprime los return de las funciones m() y b(m)

Funciones secundarias

Programación mínimos cuadrados

Funciones principales

- main(): Imprime los return de las funciones m() y b(m)
- m(): Calcula y entrega la pendiente con el método de mínimos cuadrados

Funciones secundarias

Programación mínimos cuadrados

Funciones principales

- main(): Imprime los return de las funciones m() y b(m)
- m(): Calcula y entrega la pendiente con el método de mínimos cuadrados
- b(m): Calcula y entrega el desplazamiento de la recta, en otras palabras, la velocidad inicial

Funciones secundarias

Programación mínimos cuadrados

Funciones principales

- main(): Imprime los return de las funciones m() y b(m)
- m(): Calcula y entrega la pendiente con el método de mínimos cuadrados
- b(m): Calcula y entrega el desplazamiento de la recta, en otras palabras, la velocidad inicial

Funciones secundarias

- SumDProd(a,b): Dadas dos listas de n elementos, calcula el producto en i-esimo elemento de cada lista, y luego suma los productos

Programación mínimos cuadrados

Funciones principales

- main(): Imprime los return de las funciones m() y b(m)
- m(): Calcula y entrega la pendiente con el método de mínimos cuadrados
- b(m): Calcula y entrega el desplazamiento de la recta, en otras palabras, la velocidad inicial

Funciones secundarias

- SumDProd(a,b): Dadas dos listas de n elementos, calcula el producto en i-esimo elemento de cada lista, y luego suma los productos
- ProdDSum(a,b): Dadas dos listas de n elementos, calcula la suma de cada lista y luego realiza el producto de estas sumas

Programación mínimos cuadrados

Funciones principales

- main(): Imprime los return de las funciones m() y b(m)
- m(): Calcula y entrega la pendiente con el método de mínimos cuadrados
- b(m): Calcula y entrega el desplazamiento de la recta, en otras palabras, la velocidad inicial

Funciones secundarias

- SumDProd(a,b): Dadas dos listas de n elementos, calcula el producto en i-esimo elemento de cada lista, y luego suma los productos
- ProdDSum(a,b): Dadas dos listas de n elementos, calcula la suma de cada lista y luego realiza el producto de estas sumas
- Sum(a): Dada una lista, realiza la suma de los elementos en esta

Errores

El proceso para calcular los errores de m y b, Δm y Δb respectivamente, fue uno muy similar. Se realizaron las funciones $Dm()$ y $Db()$ que realizaban el cálculo matemático.

Automatización

Para automatizar todo el proceso programé un código simple en Bash que realiza las siguientes acciones:

- Compila los archivos de C

Automatización

Para automatizar todo el proceso programé un código simple en Bash que realiza las siguientes acciones:

- Compila los archivos de C
- Almacena la salida en una variable

Automatización

Para automatizar todo el proceso programé un código simple en Bash que realiza las siguientes acciones:

- Compila los archivos de C
- Almacena la salida en una variable
- Duplica un archivo .gp pre programado

Automatización

Para automatizar todo el proceso programé un código simple en Bash que realiza las siguientes acciones:

- Compila los archivos de C
- Almacena la salida en una variable
- Duplica un archivo .gp pre programado
- Agrega la función de la recta a este archivo

Automatización

Para automatizar todo el proceso programé un código simple en Bash que realiza las siguientes acciones:

- Compila los archivos de C
- Almacena la salida en una variable
- Duplica un archivo .gp pre programado
- Agrega la función de la recta a este archivo
- Compila la gráfica

Automatización

Para automatizar todo el proceso programé un código simple en Bash que realiza las siguientes acciones:

- Compila los archivos de C
- Almacena la salida en una variable
- Duplica un archivo .gp pre programado
- Agrega la función de la recta a este archivo
- Compila la gráfica
- Imprime los valores de la aceleración y la velocidad a los 15s

Automatización

Para automatizar todo el proceso programé un código simple en Bash que realiza las siguientes acciones:

- Compila los archivos de C
- Almacena la salida en una variable
- Duplica un archivo .gp pre programado
- Agrega la función de la recta a este archivo
- Compila la gráfica
- Imprime los valores de la aceleración y la velocidad a los 15s
- Abre el pdf (válido solo para gnome)

MAY THE 4TH BE WITH YOU