

UNIREMINGTON

# CONSULTA/TALLER EN CLASE

MODELOS DE SIMULACION

Santiago Restrepo Zuluaga//Victor Manuel Espinosa  
3-10-2023

# Taller:

## 1. CONSULTA

- regresión lineal múltiple  
<https://www.youtube.com/watch?v=7j0qMSnyG9M>
- regresión lineal multivariante  
<https://www.youtube.com/watch?v=GJebP9a3duo>
- machine learning  
<https://www.youtube.com/watch?v=QN72Nw6LDU8>
- Deep learning  
<https://www.youtube.com/watch?v=xuSnmWZagN8>

### CONSULTA 1:

#### Regresión Lineal Múltiple:

La regresión lineal múltiple es un enfoque en estadísticas y aprendizaje automático que nos ayuda a entender cómo varias variables pueden influir en una cosa o resultado. Piensa en ello como tratar de entender qué factores pueden afectar el precio de una casa. No solo miramos una sola cosa, como el tamaño de la casa, sino que también consideramos otros factores como el número de habitaciones, la ubicación, el año de construcción, etc.

Imagina que quieres predecir el precio de una casa. En lugar de usar solo una característica, como el tamaño de la casa (como en la regresión lineal simple), utilizamos múltiples características o variables, como el tamaño de la casa, el número de habitaciones, la ubicación, la edad de la casa y cualquier otro factor que pueda influir en el precio.

La regresión lineal múltiple busca una relación entre todas estas variables y el precio de la casa. A través de esta relación, podemos hacer predicciones sobre cuánto podría valer una casa basándonos en sus características. Es como juntar todas las piezas del rompecabezas para obtener una imagen más completa de cómo estas variables afectan el resultado que estamos tratando de predecir.

En resumen, la regresión lineal múltiple es una herramienta útil cuando deseamos comprender cómo múltiples factores pueden estar relacionados con un resultado particular y cómo podemos utilizar esa relación para hacer predicciones. Es especialmente útil en el campo del aprendizaje automático y la estadística para tomar decisiones informadas en una amplia variedad de aplicaciones, desde la predicción de precios de casas hasta la estimación de ventas de productos.

#### Regresión lineal multivariante:

Este término se refiere a un enfoque más amplio en el que se analiza la relación entre una variable dependiente y múltiples variables independientes, pero también se consideran múltiples variables dependientes a la vez. En otras palabras, estamos tratando de entender cómo un conjunto de variables independientes puede explicar o predecir múltiples variables dependientes simultáneamente. Esto es común en investigaciones y análisis de datos en los que se estudian múltiples aspectos o resultados relacionados.

En ambos casos, el objetivo principal es entender cómo las variables independientes se relacionan con las variables dependientes y cómo se pueden utilizar estas relaciones para hacer predicciones, tomar decisiones o comprender mejor un fenómeno.

En resumen, la "regresión lineal multivariante" se refiere principalmente a la regresión lineal múltiple cuando se trabajan con varias variables independientes para predecir una variable dependiente o, en un contexto más amplio, al análisis de regresión multivariante cuando se consideran múltiples variables dependientes y variables independientes a la vez.

### **Machine learning:**

El machine learning es como enseñar a las computadoras a aprender por sí mismas. En lugar de programar instrucciones específicas para cada tarea, les damos datos y algoritmos para que la computadora aprenda patrones y tome decisiones por sí misma. Es como enseñar a una computadora a reconocer gatos en fotos mostrándole muchas imágenes de gatos en lugar de decirle cómo buscar bigotes y orejas de gato específicos.

Imagina que quieres que una computadora sea capaz de predecir si lloverá o no mañana. En lugar de decirle cómo hacerlo, le das muchos datos históricos del clima y le permites descubrir por sí misma las señales que indican si es probable que llueva o no. Con el tiempo, la computadora se vuelve mejor en esta tarea y puede ayudarte a tomar decisiones basadas en sus predicciones.

En resumen, el machine learning es enseñar a las computadoras a aprender de la experiencia y los datos para hacer tareas útiles sin necesidad de una programación detallada para cada tarea.

## Deep learning:

El deep learning (aprendizaje profundo) es una forma avanzada de enseñar a las computadoras a pensar y tomar decisiones de manera similar a como lo hacemos los seres humanos. Imagina que quieres que una computadora entienda lo que hay en una fotografía, como si hay un gato o un perro. Con el deep learning, en lugar de decirle cómo identificar ojos, orejas y patas, le damos muchas fotos de gatos y perros para que aprenda automáticamente por sí misma qué características son importantes para distinguirlos.

La idea principal es simular cómo funciona nuestro cerebro, con capas de neuronas artificiales que procesan la información en diferentes niveles de abstracción. Es como si estuviéramos enseñando a la computadora a descubrir patrones en los datos por sí misma y a tomar decisiones basadas en esos patrones. Esto se utiliza en muchas aplicaciones, desde reconocimiento de voz hasta traducción de idiomas y conducción autónoma de automóviles.

En resumen, el deep learning es como enseñar a las computadoras a pensar de manera profunda y aprender de manera similar a como lo hacemos nosotros, utilizando capas de procesamiento para entender mejor el mundo a partir de los datos.

## 2. Que es un Dataset

Descargar 1 dataset opcional del siguiente link

<https://bigdatamagazine.es/10-sitios-donde-encontrar-data-sets-gratuitos>

Revisarlo y analizar que posible información valiosa podría generar.

Un dataset es esencialmente una colección de información relacionada que puede ser utilizada para diversos fines, como análisis, estadísticas o cualquier otra operación que requiera datos estructurados.

La API Graph de Instagram es una interfaz de programación de aplicaciones que permite a profesionales de Instagram, como empresas y creadores, utilizar una aplicación para gestionar su presencia en Instagram de manera más avanzada. Esta API ofrece varias funcionalidades, incluyendo:

**Obtener y publicar contenido multimedia:** Los profesionales pueden utilizar la API para acceder a su propio contenido multimedia y publicarlo en Instagram o recuperar contenido multimedia publicado anteriormente.

**Administrar y responder comentarios:** Permite gestionar los comentarios que se hacen en su contenido y responder a ellos de manera automatizada o personalizada.

Identificar contenido mencionado: La API permite identificar contenido multimedia en el que otros usuarios de Instagram los mencionan utilizando el "@".

Búsqueda de contenido por hashtags: Los usuarios pueden buscar contenido multimedia específico utilizando hashtags.

Obtener metadatos y métricas básicas: Proporciona información sobre otros creadores y empresas de Instagram, lo que puede ser útil para realizar análisis y obtener estadísticas sobre su audiencia y desempeño.

### 3. Realizar en Matlab los ejercicios según el siguiente link

[https://la.mathworks.com/help/matlab/data\\_analysis/linear-regression.html](https://la.mathworks.com/help/matlab/data_analysis/linear-regression.html)

Realizar el ejercicio Matlab con la siguiente tabla

[https://la.mathworks.com/help/matlab/data\\_analysis/linear-regression.html](https://la.mathworks.com/help/matlab/data_analysis/linear-regression.html)

x	y
1	0,3
2	3,2
3	2,9

Simulación – Regresión Lineal  
Plinio Neira - Uniremington

---

MODELOS DE SIMULACIÓN – UNIREMINGTON

4	4
5	3,4
6	9
7	4,8

### CODIGO DE FUNCION:

```
function [a,Y,T]= Taller2Clase(x,y,X)

n = length(x);

sigmaX = sum(x);
sigmaY = sum(y);

sigmaX2 = sum(x.^2);
sigmaXY = sum(x.*y);

A = [n,sigmaX;sigmaX,sigmaX2];
B = [sigmaY;sigmaXY];

a = A\B;

Sr = sum((y-a(1)-a(2).*x).^2)
St = sum((y-sigmaY/n).^2)
es = sqrt(Sr/(n-2))
cdet = (St - Sr)/St
ccor = sqrt(cdet)

E1={'Sr';'St';'Error estandar del estimado';...
    'Coeficiente de determinación';'Coeficiente de correlacion'};
E2={Sr;St;es;cdet;ccor};
T = table (E1,E2,'VariableNames',{ 'Valores Calculados','Resultados'});

scatter(x,y); hold on
grid on
modelo = poly2sym(flipud(a));
f= matlabFunction(modelo);
fplot(modelo,[x(1),x(end)]); hold on

if sign(a(2)) == -1
    title(['Y = ',num2str(a(1)),num2str(a(2)), 'x']);
else
    title(['Y = ',num2str(a(1)), ' + ',num2str(a(2)), 'x'])
end

maxX = max(x);
maxY = max(y);minY = min(y);
axis([0,maxX + 2, minY - 2, maxY + 2]);

if nargin < 3
    Y= NaN;
    return
elseif nargin == 3
    Y= f(x);
    plot(X,Y,'ro','MarkerFaceColor','r');
end
```

### ENTRADAS DE LA CONSOLA:

```
x=[1,2,3,4,5,6,7]
y=[0.3,3.2,2.9,4,3.4,9,4.8]
[a,Y,T]=Taller2Clase(x,y,5)
```

x							=
	1	2	3	4	5	6	7

y							=
	0.3000	3.2000	2.9000	4.0000	3.4000	9.0000	4.8000

Sr							=
	18.1114						

St							=
	41.5171						

es							=
	1.9032						

cdet							=
	0.5638						

ccor							=
	0.7508						

a							=
	0.2857						
	0.9143						

Y							=
	1.2000	2.1143	3.0286	3.9429	4.8571	5.7714	6.6857

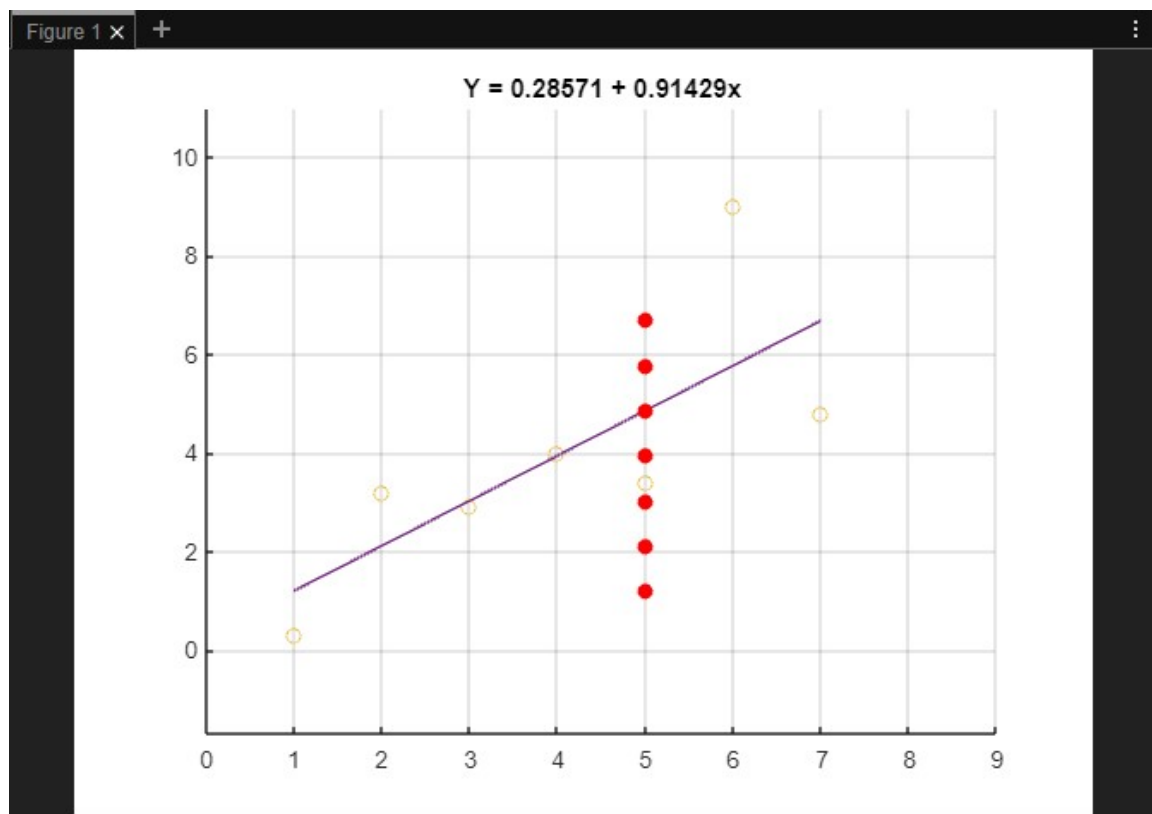
T							=
---	--	--	--	--	--	--	---

5x2

[table](#)

Valores	Calculados	Resultados
{'Sr'	}	{[18.1114]}
{'St'	}	{[41.5171]}
{'Error estandar del estimado' }		{[ 1.9032]}
{'Coeficiente de determinación'}		{[ 0.5638]}
{'Coeficiente de correlacion' }	{[ 0.7508]}	

GRAFICA:





5. Como importar un dataset en Matlab
  - a. Cargar el dataset consultado en Matlab
  - b. Contar el numero de registros

`A = importdata(filename)`

`A = importdata('-pastespecial')`

`A = importdata(___,delimiterIn)`

`A = importdata(___,delimiterIn,headerlinesIn)`

`[A,delimiterOut,headerlinesOut] = importdata(___)`

Descripción

ejemplo

`A = importdata(filename)` carga los datos en el arreglo A.

ejemplo

`A = importdata('-pastespecial')` carga los datos desde el portapapeles del sistema en lugar de hacerlo desde un archivo.

`A = importdata(___,delimiterIn)` interpreta `delimiterIn` como el separador de columna en el archivo ASCII, `filename`, o los datos del portapapeles. Puede utilizar `delimiterIn` con cualquiera de los argumentos de entrada de las sintaxis anteriores.

ejemplo

`A = importdata(___,delimiterIn,headerlinesIn)` carga los datos del archivo ASCII, `filename`, o del portapapeles, leyendo los datos numéricos a partir de la línea `headerlinesIn+1`.

ejemplo

`[A,delimiterOut,headerlinesOut] = importdata(___)` devuelve también el carácter delimitador detectado en el archivo ASCII de entrada en `delimiterOut` y el número de líneas de encabezado detectado en `headerlinesOut`, utilizando cualquiera de los argumentos de entrada de las sintaxis anteriores.

## 6. Consultar sobre teoría de colas

### a. Como funcionan las estructuras:

- i. LIFO – y 2 ejemplos
- ii. FIFO- y 2 ejemplos

### b. Leer sobre los parámetros de una cola (clientes, colas, servidores)

### c. Consultar herramienta online para simulación de colas

- i. Realizar pruebas y tomar evidencia.

La teoría de colas es un campo de estudio que se enfoca en el análisis y la modelización matemática de sistemas de espera o colas. Estos sistemas son comunes en una variedad de aplicaciones, como centros de llamadas, sistemas de atención médica, líneas de producción y más, donde los elementos o clientes deben esperar para ser atendidos por un servidor. A continuación, responderé a tus preguntas sobre la teoría de colas:

#### a. Estructuras de cola:

i. LIFO (Last-In, First-Out): En una estructura LIFO, el último elemento en entrar a la cola es el primero en ser atendido. Es decir, el elemento que llega más recientemente es el próximo en ser procesado. Dos ejemplos de situaciones en las que se utiliza LIFO son:

1. Pila de libros: Imagina una pila de libros donde siempre tomas el libro que se colocó en la parte superior más recientemente. El libro que se puso en último lugar será el primero que saques.
2. Gestión de llamadas telefónicas en una centralita: Cuando una llamada llega a una centralita telefónica, se atiende la llamada más reciente antes que las que llegaron antes.

ii. FIFO (First-In, First-Out): En una estructura FIFO, el primer elemento en entrar a la cola es el primero en ser atendido. El orden de llegada determina el orden de servicio. Dos ejemplos de situaciones en las que se utiliza FIFO son:

1. Cola de impresión: Cuando envías múltiples trabajos de impresión a una impresora, se imprimirán en el orden en que los enviaste, es decir, el primero en ser enviado será el primero en imprimirse.
2. Línea de espera en un supermercado: Los clientes que llegan primero a una fila de caja serán los primeros en ser atendidos por el cajero.

#### b. Parámetros de una cola:

Los principales parámetros de una cola incluyen:

Clientes: El número de elementos (clientes) en la cola que esperan ser atendidos.

Colas: El número de colas independientes que pueden existir en un sistema. Algunos sistemas tienen una única cola, mientras que otros pueden tener múltiples colas.

**Servidores:** El número de servidores disponibles para atender a los clientes en la cola. Puede ser uno o más, dependiendo del sistema.

c. Herramienta de simulación de colas:

Una herramienta en línea popular para la simulación de colas es "Arena Simulation Software". Esta herramienta permite crear modelos de sistemas de colas y simular su comportamiento. Puedes realizar pruebas con diferentes configuraciones de clientes, colas y servidores para analizar el rendimiento del sistema y tomar decisiones informadas. Asegúrate de visitar el sitio web oficial de Arena Simulation Software para obtener más información y acceder a la herramienta.

## Conclusiones

### Dataset

### Teoría de colas

Data Set (Conjunto de Datos):

Un "data set" (conjunto de datos) es una colección estructurada de datos organizados en filas y columnas. Cada fila representa una entrada o un registro de datos, y cada columna representa un atributo o una característica específica de esos datos.

Los "data sets" se utilizan en una variedad de aplicaciones, como análisis de datos, machine learning, estadísticas y más, para extraer información, patrones o conocimientos de los datos.

Pueden ser de diferentes tamaños y formatos, desde pequeños conjuntos de datos en hojas de cálculo hasta grandes "data sets" almacenados en bases de datos o sistemas de almacenamiento distribuido.

Teoría de Colas:

La teoría de colas es un campo de estudio que se enfoca en el análisis y la modelización matemática de sistemas de espera o colas. Estos sistemas son comunes en situaciones donde los elementos (clientes, tareas, solicitudes, etc.) deben esperar su turno para ser atendidos por un servidor.

La teoría de colas se utiliza para entender y optimizar el rendimiento de sistemas de espera, como centros de llamadas, sistemas de atención médica, líneas de producción y más.

Los conceptos clave de la teoría de colas incluyen la llegada de clientes, el servicio por parte de servidores, la capacidad del sistema, las tasas de llegada y servicio, y la longitud de la cola, entre otros.

El objetivo principal de la teoría de colas es analizar y predecir el comportamiento de un sistema de cola dado, teniendo en cuenta diversos parámetros y métricas de rendimiento.

Un "data set" es una colección de datos estructurados, mientras que la teoría de colas se centra en el análisis y la modelización de sistemas de espera o colas, que son comunes en una variedad de aplicaciones donde los elementos esperan ser atendidos en un orden específico.