

# Apuntes de Clase

Semana 9

David Alberto Guevara Sánchez  
Instituto Tecnológico de Costa Rica, Maestría en Computación  
Diseño de Experimentos

2023-04-13

## Tabla de contenidos

<b>1</b>	<b>Exposiciones</b>	<b>2</b>
1.1	<i>Playing Super Mario induces structural brain plasticity: gray matter changes resulting from training with a commercial video game</i> expuesto por Jorge Luis Gonzalez Rodríguez . . . . .	2
1.1.1	¿Quiénes son los autores? . . . . .	3
1.1.2	¿Cómo se consiguió la muestra? . . . . .	3
1.1.3	¿Cómo se dividió la muestra? . . . . .	3
1.1.4	¿Cómo se hizo el escaneo? . . . . .	3
1.1.5	¿Cómo se le hizo pruebas a los participantes? . . . . .	3
1.1.6	¿Cómo se hizo el análisis de datos? . . . . .	3
1.1.7	¿Cuáles fueron los resultados? . . . . .	4
1.1.8	¿Cuáles fueron las conclusiones? . . . . .	4
1.1.9	¿Cuáles son las limitaciones? . . . . .	4
1.2	<i>Does Video Gaming Have Impacts on the Brain Evidence from a Systematic Review</i> expuesto por David Alberto Guevara Sánchez	4
1.2.1	¿Qué es una revisión sistemática? . . . . .	4
1.2.2	¿Quiénes son los autores? . . . . .	4
1.2.3	¿Cuántos Artículos Científicos fueron analizados? . . . . .	5
1.2.4	¿Cómo se buscaron los Artículos Científicos? . . . . .	5
1.2.5	¿Cuáles son los requisitos para los Artículos Científicos? . . . . .	5
1.2.6	¿Cuántos Artículos Científicos fueron excluidos por no cumplir los requisitos? . . . . .	5
1.2.7	¿Cuáles son los requisitos secundarios para los Artículos Científicos? . . . . .	5
1.2.8	¿Cuántos Artículos Científicos fueron excluidos por no cumplir los requisitos secundarios? . . . . .	5
1.2.9	¿Cuántos y cuáles artículos fueron incluidos? . . . . .	6
1.2.10	¿Cuáles son los efectos benéficos de los videojuegos? . . . . .	6

1.2.11	¿Qué áreas son afectadas por los videojuegos? . . . . .	6
1.2.12	¿Qué puntos de mejora hay para el Artículo Científico y cuáles limitaciones este encontró? . . . . .	6
1.2.13	¿A qué conclusiones se llegaron? . . . . .	7
1.3	<i>Predicting protein structures with a multiplayer online game</i> expuesto por Aarón Josué Sibaja Villalobos . . . . .	7
1.3.1	¿Quiénes son los autores? . . . . .	7
1.3.2	¿Qué es un aminoácido? . . . . .	7
1.3.3	¿Cuál es la estructura de las proteínas? . . . . .	7
1.3.4	¿Qué es la desnaturalización de las proteínas? . . . . .	7
1.3.5	¿Por qué es importante esto? . . . . .	7
1.3.6	¿Por qué es este un problema complejo? . . . . .	7
1.3.7	¿Qué formas hay de encontrar dicha conformación? . . . .	8
1.3.8	¿Qué es FoldIt? . . . . .	8
1.3.9	¿Qué resultados dio el experimento? . . . . .	9
1.3.10	¿A qué conclusiones se llegó? . . . . .	9
<b>2</b>	<b>“Experimento” que ejecutó el profesor</b>	<b>10</b>
<b>3</b>	<b>Análisis con datos reales del examen</b>	<b>10</b>
<b>4</b>	<b>Diseño monofactorial con bloques</b>	<b>17</b>
4.1	¿Qué es <i>Factor Bloque</i> ? . . . . .	17
4.2	¿Qué es Diseño de Bloques? . . . . .	17
4.3	Basado en el último experimento que se hizo... . . . . .	17
<b>5</b>	<b>Anova Monofactorial con Bloques</b>	<b>18</b>
5.1	Ejemplo . . . . .	18
<b>6</b>	<b>Información sobre la tarea corta de la siguiente semana</b>	<b>32</b>

## 1 Exposiciones

La clase comenzó con los compañeros David Guevara (el autor de este resumen), Jorge Gonzalez y Aarón Sibaja exponiendo los *papers* que tenían asignados.

### 1.1 *Playing Super Mario induces structural brain plasticity: gray matter changes resulting from training with a commercial video game* expuesto por Jorge Luis Gonzalez Rodríguez

Es un artículo que trata sobre estudiar el impacto positivo que pueden tener los videojuegos sobre el cerebro. Esto debido a que los videojuegos se han vuelto cada vez más omnipresentes en la vida de las personas y hay evidencia de que jugarlos ayuda a mejorar ciertas habilidades perceptivas.

### **1.1.1 ¿Quiénes son los autores?**

**Simone Kühn**

Max Planck Institute for Human Development

**Ulman Lindenberger**

Max Planck Institute for Human Development

**Robert Lorenz**

Max Planck Institute for Human Development

**Tobias Gleich**

Charité Universitätsmedizin Berlin

**Jürgen Gallinat**

Universitätsklinikum Hamburg-Eppendorf

### **1.1.2 ¿Cómo se consiguió la muestra?**

Los 48 participantes se consiguieron mediante anuncios en periódicos e internet, entrevistándolos para evaluar trastornos mentales y dándoles una compensación económica.

### **1.1.3 ¿Cómo se dividió la muestra?**

La muestra se dividió en dos grupos, uno de entrenamiento con 23 personas (17 mujeres y 6 hombres), y uno de control con 25 personas (17 mujeres y 8 hombres).

El grupo de entrenamiento tenía que jugar al Super Mario 64 en un Nintendo DS durante 30 minutos al día por dos meses.

El grupo de control hizo lo mismo que el grupo de entrenamiento, pero empezó dos meses después.

### **1.1.4 ¿Cómo se hizo el escaneo?**

Para la recopilación de imágenes se hizo uso de un escáner Siemens Trim Trio 3T con una bobina de cabeza de 12 canales.

Para la obtención de imágenes se hizo uso de una secuencia eco-gradiente.

### **1.1.5 ¿Cómo se le hizo pruebas a los participantes?**

Se les hizo un cuestionario con un reporte de horas y puntuación (estrellas recolectadas) y pruebas de rendimiento cognitivas.

### **1.1.6 ¿Cómo se hizo el análisis de datos?**

Se hizo uso de las VBM8 y SPM8 para el procesamiento de imágenes estructurales. Se hizo uso de las plantillas DARTEL para la segmentación de materia blanca y gris y se hizo un análisis estadístico.

### 1.1.7 ¿Cuáles fueron los resultados?

Los participantes en promedio jugaron 50.2 horas por día y obtuvieron 74.7 estrellas.

Se encontró una interacción significativa entre el hipocampo derecho, la corteza prefrontal dorsolateral derecha y el cerebelo.

Se encontró una correlación negativa entre el cambio de estrategia de orientación y el cambio del hipocampo.

Se mostró una asociación positiva con el deseo promedio para continuar jugando.

### 1.1.8 ¿Cuáles fueron las conclusiones?

Las conclusiones fueron que el cerebro tiene efectos positivos al jugar videojuegos. Adicionalmente, los videojuegos podrían ayudar con las enfermedades mentales.

Otra conclusión es que el deseo de jugar influencia en el crecimiento de DLPFC.

### 1.1.9 ¿Cuáles son las limitaciones?

- Nuevos dispositivos técnicos.
- Juegos de otros géneros.
- Éxito de los videojuegos
- Medidas de transferencia.

## 1.2 *Does Video Gaming Have Impacts on the Brain Evidence from a Systematic Review* expuesto por David Alberto Guevara Sánchez

Esta presentación habla sobre el *paper* “*Does Video Gaming Have Impacts on the Brain Evidence from a Systematic Review*” publicado por Denilson Brilliant T, Rui Nouchi y Ryuta Kawashima en el año 2019.

### 1.2.1 ¿Qué es una revisión sistemática?

El *paper* sigue el formato de una revisión sistemática, donde se hace un análisis de una bibliografía heterogénea.

### 1.2.2 ¿Quiénes son los autores?

El primero autor del *paper*, Denilson Brilliant T, posee un bachillerato y 3 *papers* publicados.

El segundo autor, Rui Nouchi, posee un Ph.D. y 128 *papers* publicados.

El tercer autor, Ryuta Kawashima, es una eminencia en el área del efecto de los videojuegos en el cerebro y aparece en varios juegos de Nintendo titulados Brain Training Video-games.

### 1.2.3 ¿Cuántos Artículos Científicos fueron analizados?

Se analizaron 140 artículos científicos, de los cuales 1 estaba repetido.

### 1.2.4 ¿Cómo se buscaron los Artículos Científicos?

Se utilizaron las palabras clave:

- “*video game*”
- “*video gaming*”
- “*game*”
- “*action video game*”
- “*video game training*”
- “*training*”
- “*play*”
- “*playing*”
- “*MRI*”
- “*cognitive*”
- “*cognition*”
- “*executive function*”
- “*randomized control trial*”

Estas palabras clave fueron utilizadas para buscar artículos en plataformas como Google Scholar.

### 1.2.5 ¿Cuáles son los requisitos para los Artículos Científicos?

- Utilizar únicamente MRI.
- Los participantes deben estar sanos y sin adicciones.
- Los estudios deben ser experimentales (no correlacionales).

### 1.2.6 ¿Cuántos Artículos Científicos fueron excluidos por no cumplir los requisitos?

121 artículos científicos fueron excluidos por no cumplir los requisitos.

### 1.2.7 ¿Cuáles son los requisitos secundarios para los Artículos Científicos?

- Haber sido publicados luego del año 2000.
- Haber sido publicados en Inglés.
- Tener una duración especificada y mayor a 4 semanas.
- No haber estudiado juegos de cognición.

### 1.2.8 ¿Cuántos Artículos Científicos fueron excluidos por no cumplir los requisitos secundarios?

9 artículos fueron excluidos por no cumplir los requisitos secundarios.

### 1.2.9 ¿Cuántos y cuáles artículos fueron incluidos?

Al final solamente 9 artículos fueron incluidos, y se pueden ver en Tabla 1.

Tabla 1: Artículos incluidos

Autor	Año	Edades de los participantes	Género de Videojuego	Duración
Gleich et al.	2017	18-36	3D adventure	8 semanas
Haier et al.	2009	12-15	puzzle	3 meses
Kuhn et al.	2014	19-29	3D adventure	8 semanas
Lee et al.	2012	18-30	strategy	8-10 semanas 8-11 semanas
Lorenz et al.	2015	19-27	3D adventure	8 semanas
Martinez et al.	2013	16-21	puzzle	4 semanas
Roush	2013	50-65	rhythm dance	24 semanas
West et al.	2017	55-75	3D adventure	24 semanas
West et al.	2018	18-29	FPS	8 semanas

### 1.2.10 ¿Cuáles son los efectos benéficos de los videojuegos?

- Cambios en la materia gris
- Cambios en la actividad del cerebro
- Cambios en la conectividad funcional

### 1.2.11 ¿Qué áreas son afectadas por los videojuegos?

- Hipocampo (Memoria y Reconocimiento de Escenas)
- Corteza Prefrontal Dorsolateral (Solución de Problemas y Memoria)
- Cerebelo (Solución de Problemas y Memoria)

### 1.2.12 ¿Qué puntos de mejora hay para el Artículo Científico y cuáles limitaciones este encontró?

- Hubo mala calidad bajo el criterio de Delphi.
- No es fácil hacer estudios ciegos sobre videojuegos.
- La muestra era pequeña para cada estudio.
- No hay estudios a largo plazo.
- Los géneros de videojuegos utilizados eran heterogéneos.
- No hubieron pruebas cognitivas.

### **1.2.13 ¿A qué conclusiones se llegaron?**

Se llegó a la conclusión que los videojuegos sí tienen efectos positivos sobre el cerebro humano.

## **1.3 *Predicting protein structures with a multiplayer online game* expuesto por Aarón Josué Sibaja Villalobos**

Este artículo trata sobre el videojuego FoldIt.

### **1.3.1 ¿Quiénes son los autores?**

Seth Cooper fue el líder del proyecto. Tiene un Ph.D. en ciencias de la computación u es un profesor asociado de Khoury College.

Firas Khatib tiene un Ph.D. en bioinformática y es profesor asistente de la Universidad de Massachusetts Dartmouth.

Adrien Treuille es el co-fundador de Streamlit y es profesor de la universidad de Carnegie Mellon, lead en el proyecto Google X y vicepresidente de Zoox.

### **1.3.2 ¿Qué es un aminoácido?**

Es la unidad base de la estructura fundamental de las proteínas. Hay 20 tipos distintos. Cada proteína tiene dos o más cadenas de aminoácidos (polipéptidos) cuya secuencia codifica un gen.

### **1.3.3 ¿Cuál es la estructura de las proteínas?**

La secuencia de una proteína se determina con el ADN del gen que la codifica. Cambiar un solo aminoácido puede afectar la estructura y la función de la proteína.

### **1.3.4 ¿Qué es la desnaturalización de las proteínas?**

Proteínas que cambian de estado por alguna razón, como el calor.

### **1.3.5 ¿Por qué es importante esto?**

Estudiar la estructura 3D de las proteínas es fundamental para comprender su función y desarrollar nuevos fármacos y tratamientos.

### **1.3.6 ¿Por qué es este un problema complejo?**

Porque las proteínas se pueden plegar en una variedad de formas y pequeños cambios en la secuencia de aminoácidos pueden conducir a grandes cambios en la estructura 3D de la proteína.

Encontrar la conformación nativa biológicamente relevante de una proteína es un desafío computacional formidable dado el gran tamaño del espacio de búsqueda.

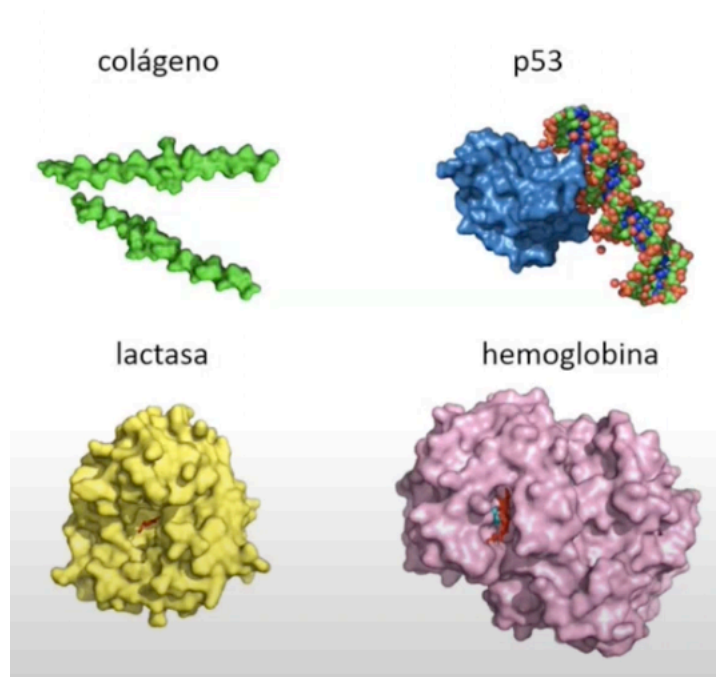


Figura 1: Algunas formas que pueden tener las proteínas

En la Figura 1 se pueden ver ejemplos.

### 1.3.7 ¿Qué formas hay de encontrar dicha conformación?

Hay una metodología de predicción de estructuras llamada Rosetta. Esta utiliza una simulación Monte Carlo para muestrear el espacio conformacional de las proteínas. Busca conformaciones de baja energía, lo cual es muy probable que sea el estado nativo de la proteína.

El problema de esta herramienta es que tenía poca interacción.

### 1.3.8 ¿Qué es FoldIt?

FoldIt es un videojuego que trata sobre encontrar proteínas nativas mediante pliegues haciendo uso de la astucia de los jugadores.

Es un juego multijugador en línea y no está enfocado a jugadores con experiencia en biología molecular.

Es un híbrido de *crowdsourcing* y *computación distribuida*.



Tiene modalidades de competencia y colaboración.

Está para las tres plataformas principales: Linux, Windows y MacOS.

En la Figura 2 se puede ver un ejemplo del juego en progreso.

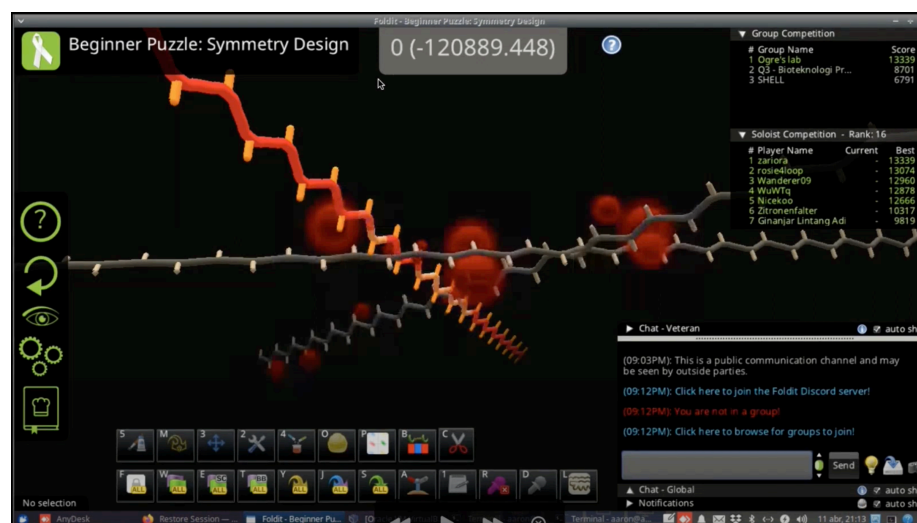


Figura 2: Ejemplo de Gameplay de FoldIt

### 1.3.9 ¿Qué resultados dio el experimento?

- Los jugadores pudieron identificar el modelo más cercano a la estructura nativa y mejorarlo aún más.
- Los jugadores tuvieron un desempeño similar al protocolo de reconstrucción y refinamiento de Rosetta para 3 de 10 acertijos.
- Los humanos usan una mayor gama de métodos de exploración que las computadoras.
- Para problemas particularmente difíciles los jugadores solo reciben una cadena de proteína extendida al inicio. Aunque las herramientas permiten alcanzar conformación nativa desde el inicio los jugadores presentaban problemas para alcanzarla, revelando la necesidad de encontrar el equilibrio adecuado entre los humanos y los métodos computacionales.
- Unos jugadores lograron encontrar una encima del VIH.

### 1.3.10 ¿A qué conclusiones se llegó?

Las computadoras no son buenas para realizar *puzzles* visuales. Es útil combinar el *gaming* con el *crowdsourcing* y se puede utilizar en otros ámbitos también.

## 2 “*Experimento*” que ejecutó el profesor

El profesor empieza la clase enseñando un “*experimento*” que él ejecutó. El “*experimento*” era un ejemplo de lo que pueden llegar a hacer los autores de *papers* para presentar su *paper* como significativo, a pesar de que no lo sea.

Un profesor del TEC del curso de Diseño de Experimentos publica un artículo científico en el cual expone los resultados de un método infalible que encontró para mejorar el rendimiento académico de sus estudiantes.

Los resultados que se muestran son del promedio obtenido en la nota del primer examen, los contenidos del curso, tipo de estudiantes, dificultad de la prueba y condiciones en que se realizó la prueba se mantuvo homogéneo durante el experimento.

Bajo el análisis se tienen tres grupos, de 13, 19 y 25 estudiantes, que fueron expuestos a mejoras progresivas del método del profesor en ese mismo orden.

En la Figura 3 se puede ver la nota promedio de cada grupo. Cabe recalcar que esta Figura 3 está manipulando los datos para que la mejoría parezca más grande de lo que realmente es. La diferencia entre grupos es de 0.3 entre el grupo IS-2022 y el grupo IS-2023 y de 3.0 entre IIS-2021 y IS-2022, pero en la Figura 3 parece que la distancia entre notas es prácticamente la misma, haciendo parecer que hubo un crecimiento lineal (esto se puede ver ilustrado en la Figura 4).

Cabe recalcar también que el profesor dice que la mejora entre cada grupo fue de un 7% aproximadamente (entre IIS-2021 y IS-2022 un 6.7% y entre IIS-2021 (el cual ni siquiera aparece en el gráfico de la Figura 3) y IS-2023 un 7.3%), lo cual no es una mentira, pero fue hecho para ser lo más confuso posible para el lector. Adicionalmente, el rango de los datos que presenta el profesor es de 45. Este rango es importante porque el profesor dice que aplicando una regla de 3, haciendo,  $\frac{3}{45} : \frac{x}{100}$  donde 3 es la diferencia entre IS-2022 y IS-2023. Esto nos da el 6.7% de mejoría.

En lo que corresponde a los resultados el profesor mencionó lo siguiente:

- Por su increíble aporte a la ciencia, el profesor recibió aplausos y elogios en la conferencia científica donde publicó su trabajo.
- Ganó un premio Nobel.
- Los resultados obtenidos son irrefutables.
- Ninguno de los resultados mostrados faltan a la verdad.

## 3 Análisis con datos reales del examen

No hay nada más bonito que realizar el análisis con datos reales .

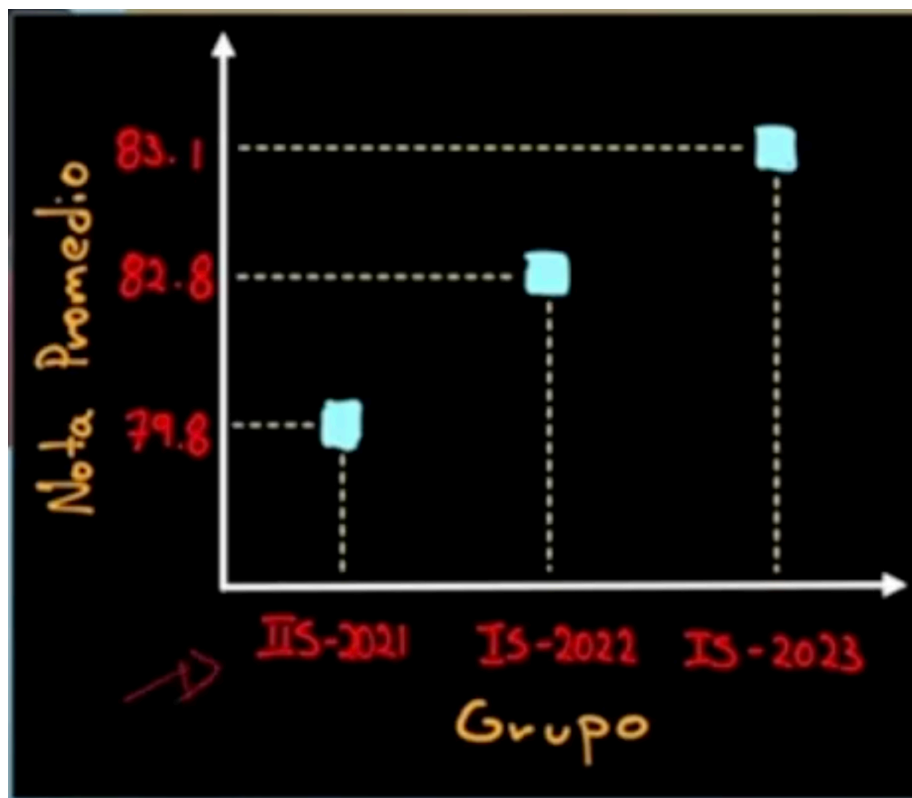


Figura 3: Nota promedio para cada grupo

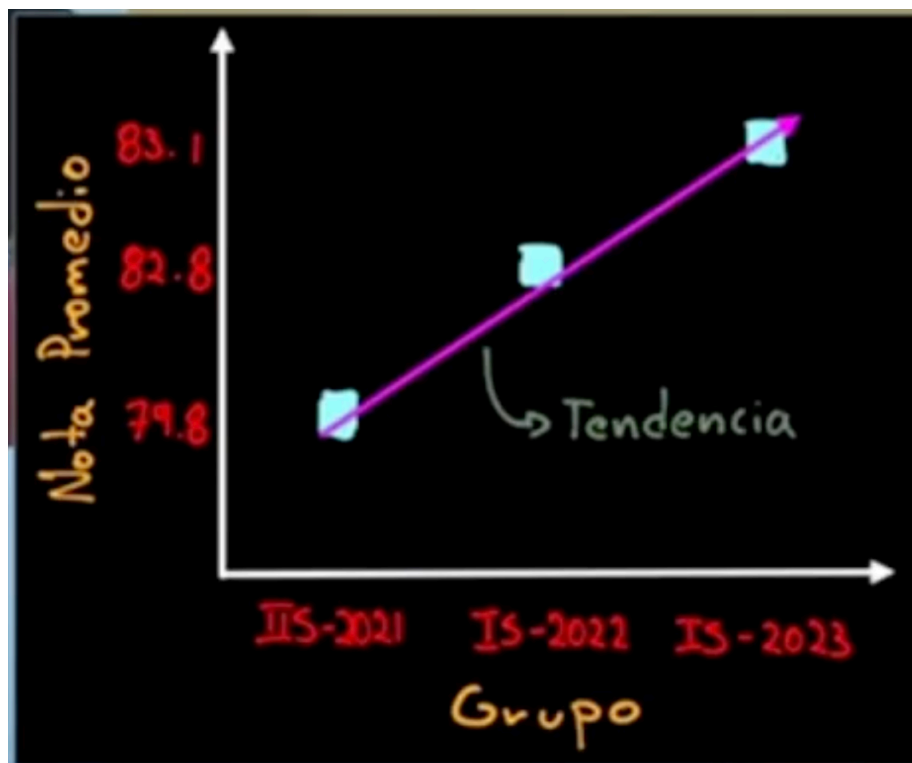


Figura 4: Nota promedio para cada grupo con línea de tendencia

### Hipótesis Nula

Los promedios de la variable medida en cada grupo son iguales.

### Hipótesis Alternativa

Los promedios de las variables entre grupos **NO** son iguales.

Básicamente es si se encontraron o no diferencias significativas entre los promedios de los grupos.

Tabla 2: Tabla de factores para el Anova Monofactorial

	Factores
	Grupo
Niveles	IIS-2021
	IS-2022
	IS-2023

En la Tabla 2 se puede ver la tabla que usualmente se hace para el Anova Monofactorial.

Grupo	n	mean	sd	min	Q1	median	Q3	max
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
IIS-2021	13	79.77692	10.71558	53.2	75.8	80.6	83.9	95.2
IS-2022	19	82.77895	8.06071	65.5	79.1	83.6	88.2	94.5
IS-2023	25	83.06000	7.58183	65.5	80.0	83.6	87.3	98.2

Figura 5: Datos resumidos por grupo

En la Figura 5 se pueden ver los datos resumidos por grupo. Un dato importante es que en la mediana entre el grupo IS-2022 y el grupo ISS-2023 no se puede ver ningún cambio.

En la Figura 6 se puede ver el gráfico de cajas de la Figura 5. Con este diagrama de cajas no se puede notar una diferencia estadísticamente significativa, al menos según lo que se nos fue enseñado hasta el momento en el curso. Un dato importante es que los *outliers* fueron sacados del gráfico mas no del análisis.

Luego, en la Figura 7 se puede ver prácticamente lo mismo, no hay diferencia estadísticamente significativa a simple vista.

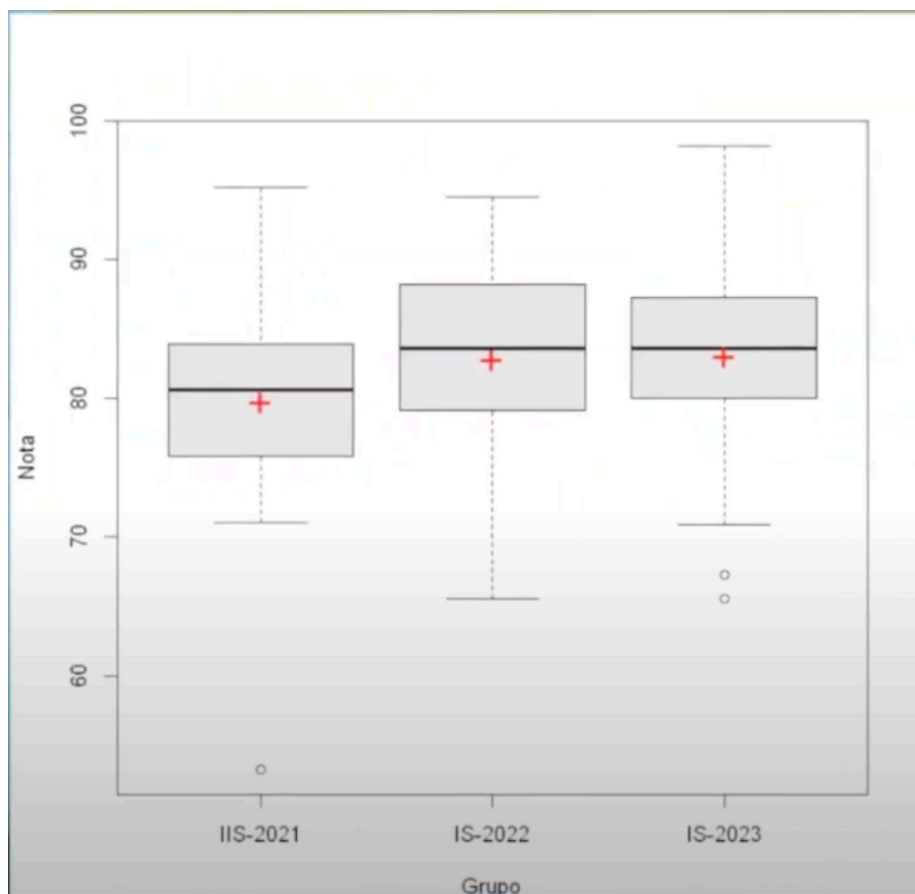


Figura 6: Gráfico de cajas

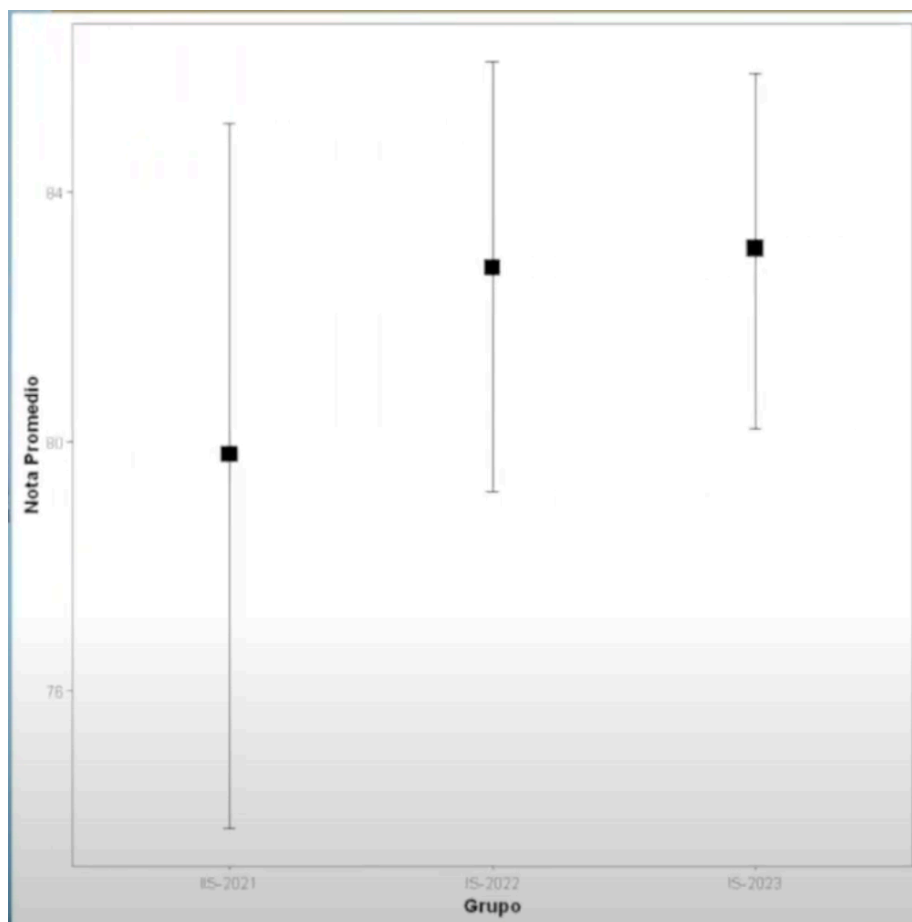


Figura 7: Gráfico de promedios e intervalos de confianza

💡 Tip

La idea de verlo a simple vista es que si se traslapan se puede decir que no son diferentes, pero si **NO** se traslapan es necesario hacer alguna prueba estadística.

💡 Tip

Un dato nuevo importante es que con el diagrama de cajas y bigotes hay una regla de dedo: si las cajas **NO** se traslapan los grupos son distintos.

⚠ Advertencia

Esta es una regla informal, así que no se puede usar en un *paper*.

	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Grupo	101.1678	2	0.6955672	0.5032045
Residuals	3927.0547	54	NA	NA

Figura 8: Resultados del Anova Monofactorial

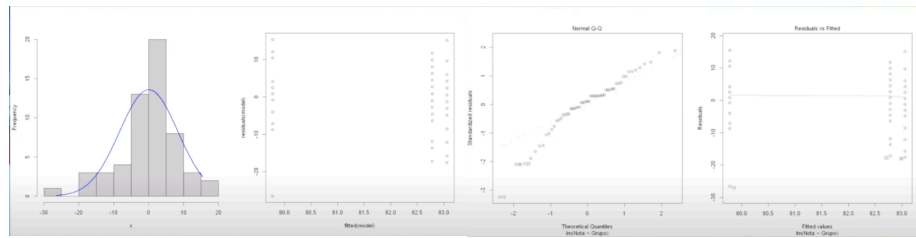
En la Figura 8 se puede ver el resultado del Anova Monofactorial. El *P value* que se puede ver en la Figura 8 es mayor a 0.05 por bastante, lo que indica que no hay una diferencia entre grupos estadísticamente significativa.

La conclusión de esto es que hay que ser críticos y preferiblemente aplicar las pruebas estadísticas.

Para finalizar, en la Figura 9 se pueden ver el histograma y el diagrama de residuos y homocedasticidad. El histograma sigue una distribución normal, porque aunque hay desviaciones estas no son muy grandes y la prueba del Anova Monofactorial acepta incluso desviaciones bastante grandes. En lo que corresponde al diagrama de residuos se puede ver que la varianza está bastante bien.

El profesor da un ejemplo sobre un estudiante que dice que no está contento con la valoración del supuesto de normalidad y quiere hacer transformación de datos. Esto lo da el profesor porque más adelante se verá cómo hacerlo en el





(a) Histograma (b) Diagramas de residuos y homocedasticidad

Figura 9: Histograma y diagramas de residuos y homocedasticidad

curso. Igualmente el resultado no va a cambiar. La transformación de datos hace la prueba más estricta.

## 4 Diseño monofactorial con bloques

### 4.1 ¿Qué es *Factor Bloque*?

Un factor bloque es un factor que tiene influencia en nuestra variable de estudio, y, aunque no es el objeto de estudio, se incluye dentro del experimento porque se desea llegar a conclusiones estadísticamente válidas a pesar de la influencia de este.

### 4.2 ¿Qué es Diseño de Bloques?

- Es utilizado cuando se desea hacer un análisis de varianza.
- Es utilizado para considerar el efecto de factores más allá de la variable independiente bajo estudio.
- Dependiendo del experimento, se utiliza para factores que no pueden ser controlados.
- La idea no es enfocarse en los efectos de los bloques como tales y tampoco en los factores detrás de asignar esos bloques.

La idea es ecualizar el error que estos factores bloque puedan tener en nuestro experimento.

El mecanismo de bloque nos permite controlar la variación introducida por el factor bloque.

### 4.3 Basado en el último experimento que se hizo...

La idea del último experimento es conocer el rendimiento del **Algoritmo C**, no si la variable de respuesta depende de la computadora en que se ejecuta. Debido a eso, se considera la computadora como un factor bloque.

#### Advertencia

Un factor bloque debe tener 2 o más niveles.

Debido a eso se va a correr el Algoritmo A, el Algoritmo B y el Algoritmo C en las dos máquinas.

## 5 Anova Monofactorial con Bloques

Tabla 3: Tabla de factores para el Anova Monofactorial con Bloques

	Factores	
	Método de entrenamiento	Computadora
Niveles	Algoritmo A	Computadora 1
	Algoritmo B	Computadora 2
	Algoritmo C	

En Tabla 3 **Computadora** es nuestro factor bloque.

#### Advertencia

Idealmente si sabemos que existe un factor bloque tenemos que incorporarlo en el documento. Si los recursos no son suficientes para incorporarlo se debe especificar en el *paper*.

#### Tip

Una justificación sólida sobre por qué se incluyó el factor bloque es que uno tiene referencias bibliográficas o resultados donde el factor bloque afecta la variable resultado.

### 5.1 Ejemplo

Este ejemplo se debe entregar como Tarea Corta #5 antes de la semana 10.

En el apartado de **Documentos** del TEC Digital están los datos. El nombre del archivo es `Datos_monofactorial_bloques.txt`.

Las bibliotecas a utilizar son las siguientes:

```
if (!require(psych)) install.packages("psych")
if (!require(FSA)) install.packages("FSA")
```

```

if (!require(ggplot2)) install.packages("ggplot2")
if (!require(car)) install.packages("car")
if (!require(multcompView)) install.packages("multcompView")
if (!require(lsmeans)) install.packages("lsmeans")
if (!require(rcompanion)) install.packages("rcompanion")

```

Utilizamos el método clásico para ingresar los datos:

```
Data <- read.table(textConnection(In), header = TRUE)
```

Organizamos los datos según son ingresados (para que R no los ordene alfabéticamente):

```

Data$Algoritmo = factor(Data$Algoritmo, levels = unique(Data$Algoritmo))
Data$Computadora = factor(Data$Computadora, levels = unique(Data$Computadora))

```

Verificamos que todo esté correcto y liberamos la tabla:

```

library(psych)
headTail(Data)

```

```

      Algoritmo  Computadora Tiempo
1  Algoritmo A Computadora 1  12976
2  Algoritmo A Computadora 1  14854
3  Algoritmo A Computadora 1  13627
4  Algoritmo A Computadora 1   9850
...      <NA>      <NA>      ...
117 Algoritmo C Computadora 2  10991
118 Algoritmo C Computadora 2  13048
119 Algoritmo C Computadora 2  15078
120 Algoritmo C Computadora 2  13443

```

```
str(Data)
```

```

'data.frame':  120 obs. of  3 variables:
 $ Algoritmo  : Factor w/ 3 levels "Algoritmo A",...: 1 1 1 1 1 1 1 1 1 1 ...
 $ Computadora: Factor w/ 2 levels "Computadora 1",...: 1 1 1 1 1 1 1 1 1 1 ...
 $ Tiempo     : int  12976 14854 13627 9850 14466 11598 13184 13096 14895 15986 ...

```

```
summary(Data)
```

```

      Algoritmo      Computadora      Tiempo
Algoritmo A:40  Computadora 1:60  Min.    : 9018
Algoritmo B:40  Computadora 2:60  1st Qu.:11258

```

```

Algoritmo C:40
Median :12288
Mean :12382
3rd Qu.:13546
Max. :17000

```

```
rm(In)
```

Y resumimos los datos por grupo:

```
Summarize(Tiempo ~ Algoritmo, data = Data, digits = 3)
```

```

Algoritmo  n      mean      sd  min      Q1  median      Q3  max
1 Algoritmo A 40 12280.35 1940.001 9018 11130.00 12288.5 13597.0 16033
2 Algoritmo B 40 12692.25 1683.902 9506 11514.00 12568.0 13661.5 17000
3 Algoritmo C 40 12173.65 1792.109 9148 11209.25 12112.0 13275.0 16367

```

Un dato importante es que en ese resumen de los datos no se muestran los datos a como el profesor los mostró durante la clase. Esto se puede ver en Figura 10.

Algoritmo	Computadora	n	mean	sd	min	Q1	median	Q3	max
<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	Computadora 1	20	13093.65	1903.929	9850	11673.0	13196.0	14563.00	16033
Algoritmo B	Computadora 1	20	12505.60	1414.667	10067	11510.5	12154.5	13544.00	15297
Algoritmo C	Computadora 1	20	11385.60	1420.394	9148	10585.5	11606.5	12189.50	14155
Algoritmo A	Computadora 2	20	11467.05	1645.540	9018	10266.5	11551.0	12435.75	14670
Algoritmo B	Computadora 2	20	12878.90	1935.371	9506	11514.0	13016.5	14006.50	17000
Algoritmo C	Computadora 2	20	12961.70	1807.597	9364	11519.5	13090.5	14222.00	16367

Figura 10: Resumen de los datos a como los mostró el profesor

Para verlos a como el profesor los mostró durante la clase se debe usar el siguiente código:

```
Summarize(Tiempo ~ Algoritmo + Computadora, data = Data, digits = 3)
```

```

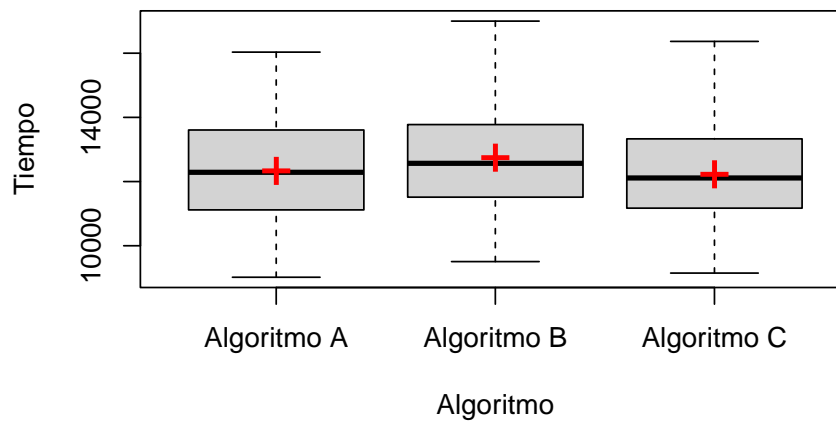
Algoritmo  Computadora  n      mean      sd  min      Q1  median      Q3
1 Algoritmo A Computadora 1 20 13093.65 1903.929 9850 11673.0 13196.0 14563.00
2 Algoritmo B Computadora 1 20 12505.60 1414.667 10067 11510.5 12154.5 13544.00
3 Algoritmo C Computadora 1 20 11385.60 1420.394 9148 10585.5 11606.5 12189.50
4 Algoritmo A Computadora 2 20 11467.05 1645.540 9018 10266.5 11551.0 12435.75
5 Algoritmo B Computadora 2 20 12878.90 1935.371 9506 11514.0 13016.5 14006.50
6 Algoritmo C Computadora 2 20 12961.70 1807.597 9364 11519.5 13090.5 14222.00
max
1 16033

```

```
2 15297
3 14155
4 14670
5 17000
6 16367
```

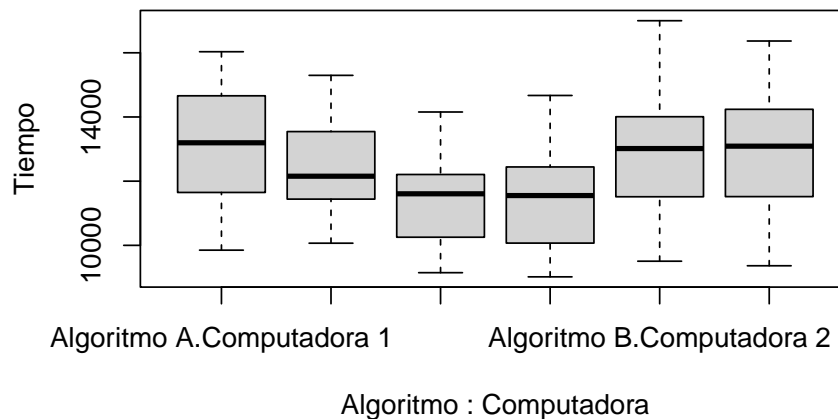
Ahora procedemos a generar el diagrama de cajas:

```
M = tapply(Data$Tiempo, INDEX = Data$Algoritmo, FUN = mean)
boxplot(Tiempo ~ Algoritmo, data = Data)
points(M, col = "red", pch = "+", cex = 2)
```



A como lo hemos desarrollado en el curso no hay diferencia entre los grupos. Ahora hagamos otro gráfico de cajas pero esta vez separados por computadora:

```
boxplot(Tiempo ~ Algoritmo + Computadora, data = Data)
```



A simple vista parece haber diferencia entre el Algoritmo A en la Computadora 1 y el Algoritmo B en la Computadora 2, pero según la regla informal introducida, como las cajas no se traslapan entonces no hay diferencia.

Ahora procedemos a crear el gráfico de promedios e intervalos de confianza:

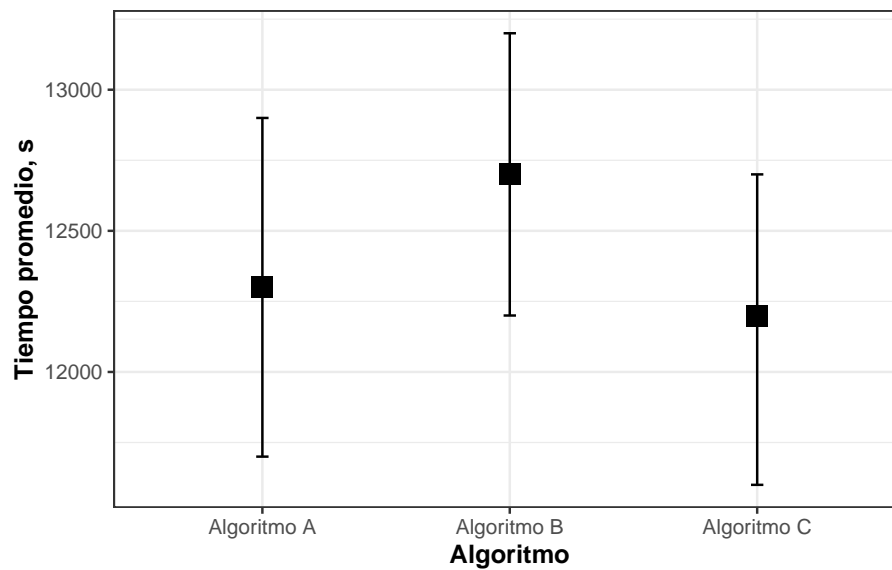
```
Sum = groupwiseMean(Tiempo ~ Algoritmo,
  data = Data, conf = 0.95, digits = 3,
  traditional = FALSE, percentile = TRUE
)
Sum
```

	Algoritmo	n	Mean	Conf.level	Percentile.lower	Percentile.upper
1	Algoritmo A	40	12300	0.95	11700	12900
2	Algoritmo B	40	12700	0.95	12200	13200
3	Algoritmo C	40	12200	0.95	11600	12700

```
library(ggplot2)

ggplot(Sum, aes(x = Algoritmo, y = Mean)) +
  geom_errorbar(aes(ymin = Percentile.lower, ymax = Percentile.upper),
    width = 0.05,
    size = 0.5
  ) +
  geom_point(shape = 15, size = 4) +
```

```
theme_bw() +
theme(axis.title = element_text(face = "bold")) +
ylab("Tiempo promedio, s")
```



Con eso se puede llegar a la conclusión de que los grupos no son distintos. Pero, si cambiamos el código para considerar a la computadora el resultado cambia:

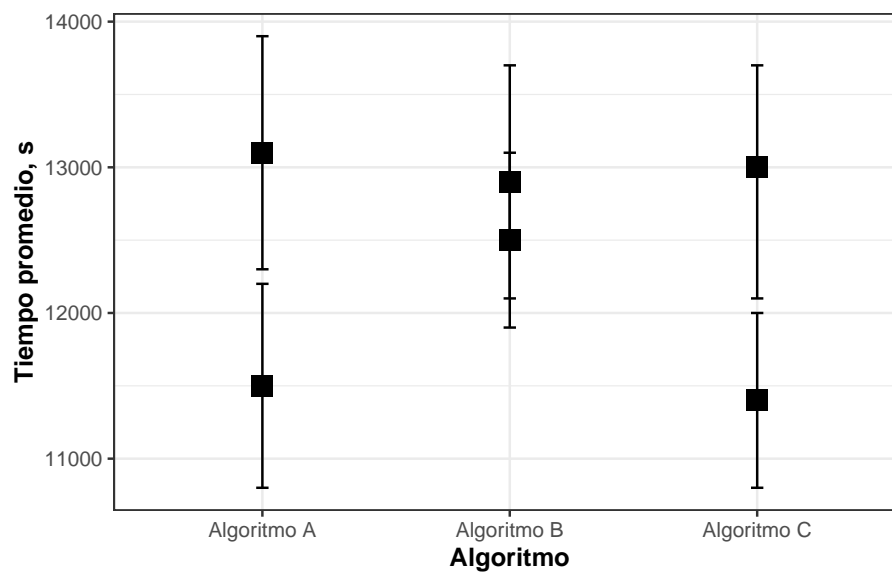
```
Sum = groupwiseMean(Tiempo ~ Algoritmo + Computadora,
  data = Data, conf = 0.95, digits = 3,
  traditional = FALSE, percentile = TRUE
)
Sum
```

	Algoritmo	Computadora	n	Mean	Conf.level	Percentile.lower
1	Algoritmo A	Computadora 1	20	13100	0.95	12300
2	Algoritmo A	Computadora 2	20	11500	0.95	10800
3	Algoritmo B	Computadora 1	20	12500	0.95	11900
4	Algoritmo B	Computadora 2	20	12900	0.95	12100
5	Algoritmo C	Computadora 1	20	11400	0.95	10800
6	Algoritmo C	Computadora 2	20	13000	0.95	12100
	Percentile.upper					
1						13900
2						12200
3						13100
4						13700

```
5         12000
6         13700
```

```
library(ggplot2)

ggplot(Sum, aes(x = Algoritmo, y = Mean)) +
  geom_errorbar(aes(ymin = Percentile.lower, ymax = Percentile.upper),
    width = 0.05,
    size = 0.5
  ) +
  geom_point(shape = 15, size = 4) +
  theme_bw() +
  theme(axis.title = element_text(face = "bold")) +
  ylab("Tiempo promedio, s")
```



Al parecer los dos resultados del Algoritmo A y Algoritmo C son distintos.

Ahora procedemos a hacer el modelo lineal:

```
model = lm(Tiempo ~ Algoritmo + Computadora, data = Data)
summary(model)
```

Call:

```
lm(formula = Tiempo ~ Algoritmo + Computadora, data = Data)
```



Residuals:

Min	1Q	Median	3Q	Max
-3316.1	-1193.8	-53.7	1226.4	4253.9

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	12226.6	331.4	36.890	<2e-16 ***
AlgoritmoAlgoritmo B	411.9	405.9	1.015	0.312
AlgoritmoAlgoritmo C	-106.7	405.9	-0.263	0.793
ComputadoraComputadora 2	107.6	331.4	0.325	0.746

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1815 on 116 degrees of freedom

Multiple R-squared: 0.01633, Adjusted R-squared: -0.009107

F-statistic: 0.642 on 3 and 116 DF, p-value: 0.5895

#### Nota

Tomar cuenta que hacer factor de bloque es solo agregar el +.

Como el *R-squared* es bajo el modelo lineal no explica muy bien la variabilidad. Esto debido a que los datos no están cerca de la línea.

#### Importante

El *P-value* aquí nos dice es qué tanto nuestros factores bloque impactan la variable de respuesta. Como es mayor a 0.05 nuestra variable **NO** se ve afectada.

Ahora procedemos a hacer el análisis de varianza con el Anova Monofactorial:

```
library(car)

Anova(model, # Tipo 2 es el por defecto
       type = "II") # Suma de cuadrados
```

Anova Table (Type II tests)

Response: Tiempo

	Sum Sq	Df	F value	Pr(>F)
Algoritmo	5999899	2	0.9103	0.4052
Computadora	347333	1	0.1054	0.7460
Residuals	382273165	116		

El *P-value* de **Algoritmo** nos dice que no hay diferencia estadísticamente significativa entre los promedios de los grupos. Esto es lo mismo que nos dice el gráfico de cajas.

El *P-value* de **Computadora** nos dice que no hay diferencia estadísticamente significativa entre los promedios de las computadoras. Como no es la variable de estudio el profesor no lo graficó.

**i** Nota

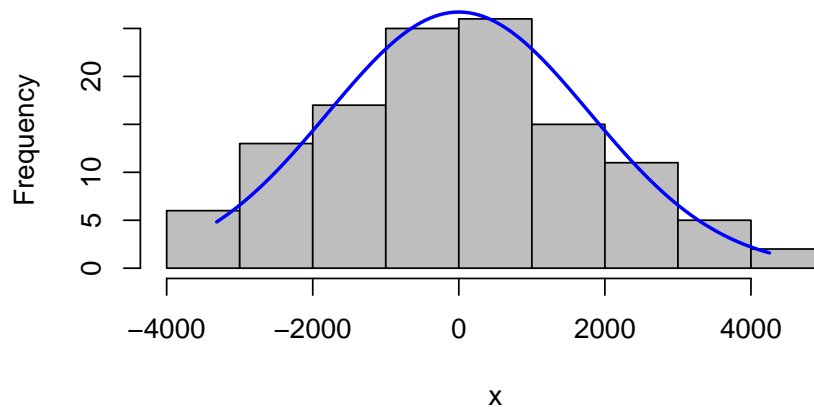
Si estuviéramos haciendo un análisis de varianza común y corriente no haríamos *post-hoc* porque el Anova Monofactorial no está diciendo que no hay diferencia.

Si tuviéramos que continuar con el análisis haríamos el análisis de los residuos, a como se puede ver a continuación:

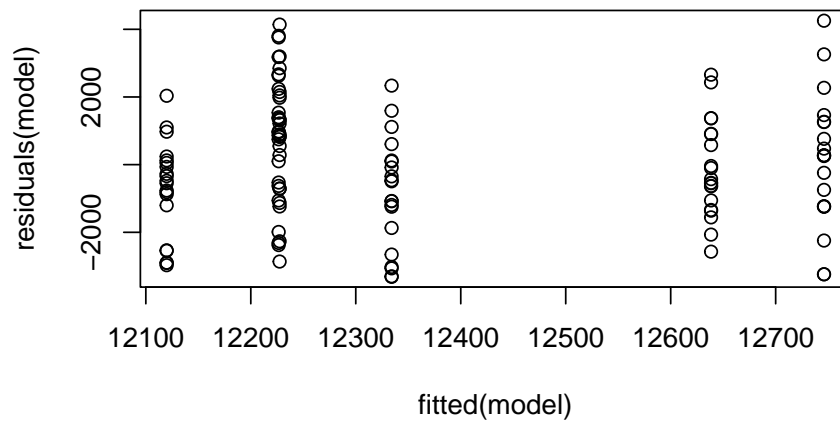
```
x = residuals(model)

library(rcompanion)

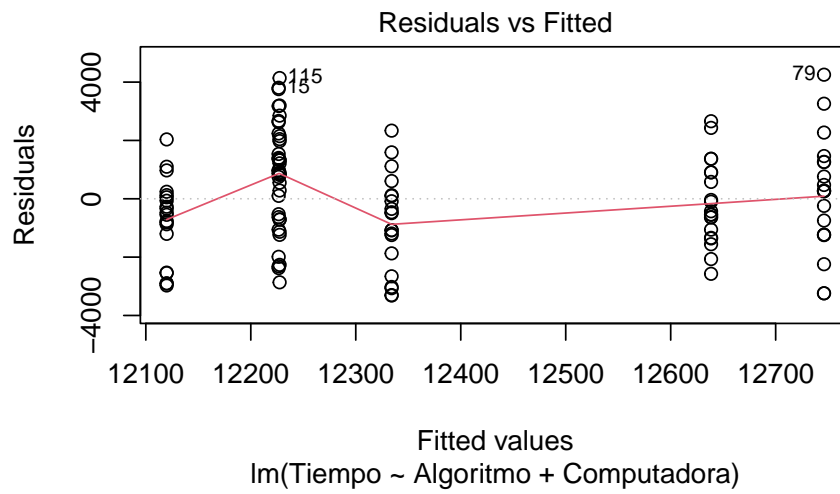
plotNormalHistogram(x)
```

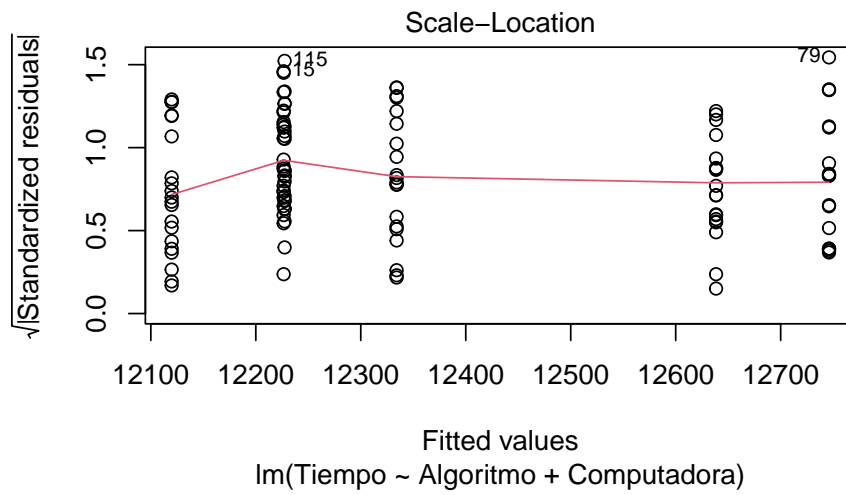
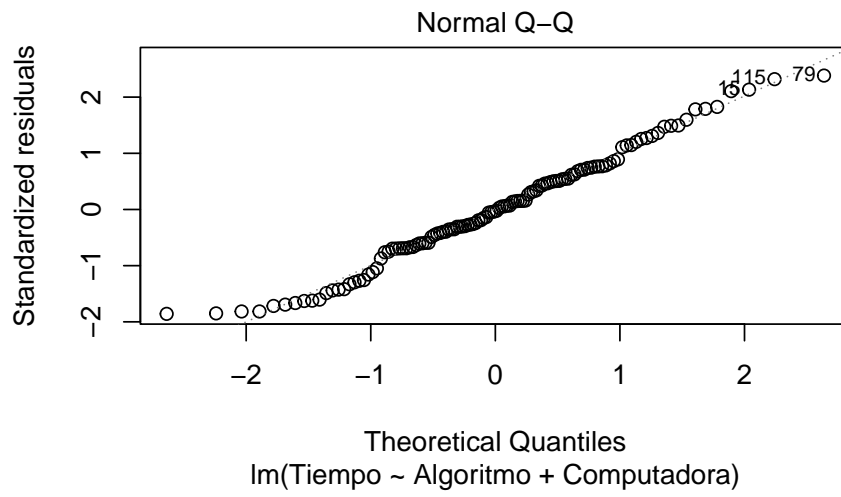


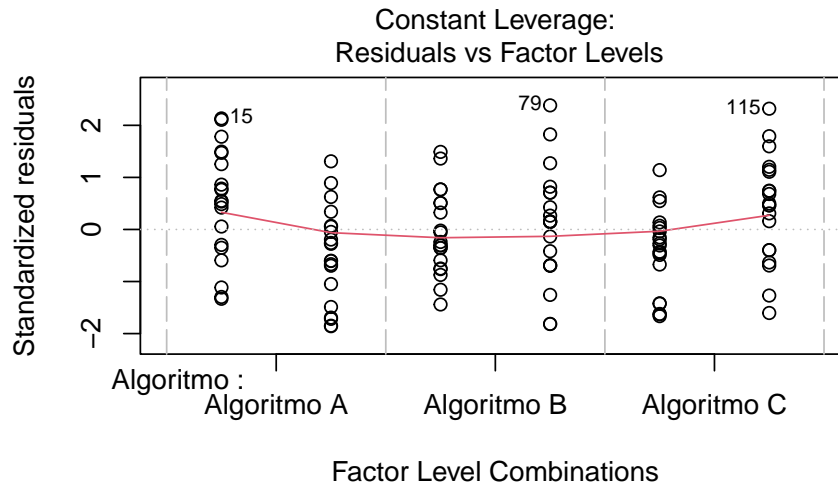
```
plot(fitted(model), residuals(model))
```



```
plot(model)
```







En base a esto se puede ver que se cumplen los supuestos de normalidad y homocedasticidad.

#### 💡 Tip

La regla de homocedasticidad dice que se cumple si no se puede encontrar un patrón en la varianza. Si se tiene que rebuscar el patrón entonces el supuesto de homocedasticidad se cumple, porque es super evidente cuando no se cumple.

Finalmente, hacemos análisis post-hoc de promedio de mínimos cuadrados:

```
library(multcompView)
library(lsmeans)

marginal = lsmeans(model, ~ Algoritmo)
pairs(marginal, adjust = "tukey")
```

contrast	estimate	SE	df	t.ratio	p.value
Algoritmo A - Algoritmo B	-412	406	116	-1.015	0.5691
Algoritmo A - Algoritmo C	107	406	116	0.263	0.9626
Algoritmo B - Algoritmo C	519	406	116	1.278	0.4107

Results are averaged over the levels of: Computadora

P value adjustment: tukey method for comparing a family of 3 estimates

Esto nos dice exactamente lo mismo que el Anova Monofactorial, que no hay diferencia.

Luego obtenemos las letras de los grupos:

```
library(multcomp)
```

```
Loading required package: mvtnorm
```

```
Loading required package: survival
```

```
Loading required package: TH.data
```

```
Loading required package: MASS
```

```
Attaching package: 'MASS'
```

```
The following object is masked from 'package:dplyr':
```

```
select
```

```
Attaching package: 'TH.data'
```

```
The following object is masked from 'package:MASS':
```

```
geyser
```

```
CLD = cld(marginal, alpha = 0.05, Letters = letters, adjust = "tukey")
```

```
Note: adjust = "tukey" was changed to "sidak"
```

```
because "tukey" is only appropriate for one set of pairwise comparisons
```

```
CLD
```

Algoritmo	lsmean	SE	df	lower.CL	upper.CL	.group
Algoritmo C	12174	287	116	11478	12869	a
Algoritmo A	12280	287	116	11585	12976	a
Algoritmo B	12692	287	116	11997	13388	a

```
Results are averaged over the levels of: Computadora
```

```
Confidence level used: 0.95
```

```
Conf-level adjustment: sidak method for 3 estimates
```

```
P value adjustment: tukey method for comparing a family of 3 estimates
```

```
significance level used: alpha = 0.05
```

```
NOTE: If two or more means share the same grouping symbol,  
then we cannot show them to be different.
```

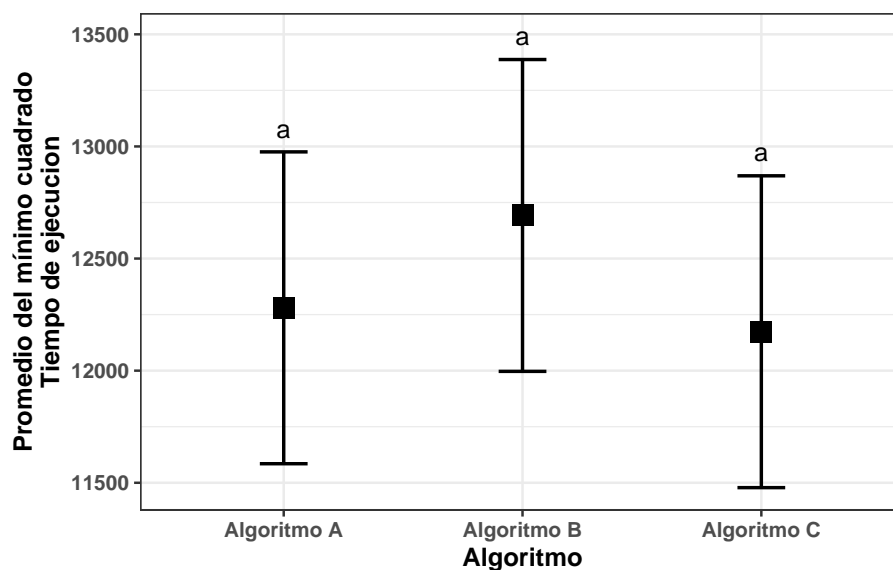
```
But we also did not show them to be the same.
```

Como todos los grupos tienen las mismas letras, no hay diferencia.

Y, finalmente, hacemos el gráfico de intervalos de confianza y letras de separación entre grupos:

```
# Ordenamos los niveles para imprimirlos
CLD$Algoritmo <- factor(
  CLD$Algoritmo,
  levels = c("Algoritmo A", "Algoritmo B", "Algoritmo C")
)
# Y removemos los espacios en blanco
CLD$.group = gsub(" ", "", CLD$.group)
# Y creamos el gráfico
library(ggplot2)

ggplot(CLD, aes(x = Algoritmo, y = lsmean, label = .group)) +
  geom_point(shape = 15, size = 4) +
  geom_errorbar(aes(ymin = lower.CL,
                    ymax = upper.CL),
               width = 0.2,
               size = 0.7) +
  theme_bw() +
  theme(axis.title = element_text(face = "bold"),
        axis.text = element_text(face = "bold"),
        plot.caption = element_text(hjust = 0)) +
  ylab("Promedio del mínimo cuadrado\nTiempo de ejecucion") +
  geom_text(nudge_x = c(0, 0, 0),
            nudge_y = c(800, 800, 800),
            color = "black")
```



Como todos tienen la misma letra no hay diferencia entre grupos.

#### ! Importante

Tomar en cuenta que el promedio de mínimo cuadrados es más estricta. La distancia se minimiza.

Aquí tenemos la leyenda:

Tiempo de entrenamiento de algoritmos de redes neuronales. Las cajas representan el promedio del mínimo cuadrado para las tres clases seguido por anova monofactorial. Las barras de error indican los intervalos de confianza de los promedios del mínimo cuadrado al 95%. Promedios que comparten letra no son significativamente diferentes ( $\alpha = 0.05$ , ajustados por Tukey).

## 6 Información sobre la tarea corta de la siguiente semana

Para la semana 10 hay que subir dos archivos. El primero de esos archivos es el ejemplo hecho hoy. El segundo archivo va a contener las siguientes diferencias:

- Se debe eliminar de los datos la **Computadora 1**. Hay que ver si hay diferencia estadística si no se hubiera considerado el factor de bloque.
- También se debe hacer otro ejemplo (conservando las dos computadoras) modificando el modelo lineal. Esto implica utilizar : en lugar de + para



crear el modelo lineal. Solamente eso, el resto del código queda idéntico.  
Con esa segunda modificación se va a medir si la interacción de factores influye estadísticamente. Esto porque en un gráfico pasado vimos que sí afectan.

**! Importante**

Cada archivo se sube como una tarea corta aparte.

**! Observación**

Un factor bloque tiene el supuesto de que no hay interacción con otros factores. Si este supuesto no se cumple, esto no debió ser analizado por factor bloque sino que debió ser un análisis multifactorial.  
Un dato curioso es que el cambiar el + por el : es un análisis multifactorial.