



Instituto Tecnológico de Durango

Programa No. 5. Nombre del Proyecto: Manual de prácticas de la materia de
Probabilidad y Estadística

Carrera: Ingeniería en Sistemas Computacionales

Docente: RUBÉN PIZARRO GURROLA

Academia de Sistemas Computacionales

Departamento Académico. Sistemas y Computación.

Periodo Sabático: Agosto 2018-Agosto 2019

Dictamen AS-2-097/2018

Fecha: Febrero 2019

Tabla de contenido

Introducción	7
1 Instalación del entorno de desarrollo RStudio y manipulación de una población de datos.	11
1.1 Conceptos.....	11
1.2 Instalar R.....	12
1.3 Instalar R Studio	12
1.4 Primeros pasos con R y R Studio.	15
1.5 Crear programa hola mundo.....	16
1.6 Organizando el trabajo en R Studio	17
1.7 Mis primeros datos	17
2 Creación de prácticas para utilización de RPubs como servicio en la nube.....	21
2.1 Conceptos.....	21
2.2 Crear el primer archivo markdown	21
2.3 Publicar en el servicio RPubs de R Studio	27
2.4 Datos, población y muestra aleatoria	31
3 Generar datos agrupados, no agrupados, y aspectos de frecuencia	34
3.1 Conceptos.....	34
3.2 Frecuencias, frecuencias relativas, histograma, gráfica de frecuencia.....	35
3.3 Programa en R. Frecuencias Caso Refrescos.....	35
3.3.1 Los datos.....	35
3.3.2 La frecuencia absoluta.....	36
3.3.3 Visualización de datos. Barras.....	36
3.3.4 Visualización de datos. Pastel	37
3.4 Programa en R. Frecuencias Caso Simular alumnos	37
3.4.1 Los datos iniciales.....	38
3.4.2 Funciones generadoras de datos	38
3.4.3 Generando la población	39
3.4.4 Tabla de frecuencias de Genero.....	40
3.4.5 Tabla de frecuencias de Carrera.....	40
3.4.6 Tabla de frecuencias de Edades	41
3.4.7 Frecuencias relativa y acumuladas de Edades	42
3.4.8 Puntos medios, máximos y mínimos de edad con summary().....	43
3.4.9 Agrupación de alumnos por edades.....	43

3.4.10	Diagrama de tallo y hoja con edades	46
4	Generar operaciones y valores usando comandos en R para determinar medidas de tendencia central	48
4.1	Conceptos.....	48
4.2	Caso de media, mediana y moda de alumnos	49
4.2.1	Importar los datos	49
4.2.2	Factorizando genero y carreras de alumnos	49
4.2.3	Estructura de alumnos	49
4.2.4	Estadísticos de genero con summary()	50
4.2.5	Estadísticos de carrera con summary()	50
4.2.6	Estadísticos de edades con summary()	50
4.2.7	Estadísticos de promedios con summary()	50
4.2.8	Estadísticos de pesos con summary()	50
4.2.9	Estadísticos de alturas con summary().....	50
4.2.10	Media, Mediana y Moda de promedios de todos.....	51
4.2.11	Promedio de alumnos de dos carreras	51
5	Generar operaciones y valores usando comandos para determinar medidas de dispersión...	53
5.1	Conceptos.....	53
5.2	Medidas de dispersión	54
5.2.1	Importar los datos	55
5.2.2	Factorizando genero y carreras de alumnos	55
5.2.3	Estructura de alumnos	55
5.2.4	Resumen de los alumnos con summary()	56
5.2.5	Determinando población Hombres y Mujeres y su media en Peso y Altura	56
5.2.6	Determinando cuartiles de Pesos en Hombre y Mujeres	57
5.2.7	Determinando percentiles de alturas del hombres y mujeres al 10% y 90%	58
5.2.8	Determinando las media de tres carreras.....	59
5.2.9	Determinando rango de promedio de calificaciones de tres carreras.....	59
5.2.10	Determinando la varianza del promedio de calificaciones de tres carreas	61
5.2.11	Determinando las desviaciones estándar de tres carreras	61
5.2.12	Determinando coeficiente de variación CV del promedio de calificaciones de tres carreras	62
5.3	Datos agrupados	63
5.3.1	Proceso:.....	63

5.4	Identificar datos de la muestra	63
5.5	Ordenar datos y mostrar.....	63
5.6	Encontrar número de elementos n, valores máximos y mínimos, rango y amplitud del rango de la muestra	64
5.7	Agrupando datos de manera habitual	64
5.8	Tabla de frecuencia de datos agrupados	64
5.9	Plot o visualizar tabla de frecuencia.....	65
5.10	Regla de Sturges.....	65
6	Crear distribución y graficación de fequencias en R.....	68
6.1	Clases de los promedios de los alumnos del ITD	68
6.2	Los datos.....	69
6.2.1	ARQUITECTURA.....	70
6.2.2	BIOQUIMICA.....	77
6.2.3	CIVIL.....	84
6.2.4	ELECTRICA.....	91
6.2.5	ELECTRONICA	98
6.2.6	INDUSTRIAL	104
6.2.7	MECANICA.....	111
6.2.8	MECATRONICA	118
6.2.9	QUIMICA.....	125
6.2.10	SISTEMAS.....	132
6.2.11	TIC.....	140
6.2.12	GESTION	145
6.2.13	INFORMATICA	153
6.2.14	ADMINISTRACION	158
6.2.15	Todas las CARRERAS en un solo data.frame	166
7	Realizar técnicas de conteo en R.....	174
7.1	Reglas de conteo, combinaciones y permutaciones Al asignar probabilidades es necesario saber identificar y contar los resultados experimentales. A continuación tres reglas de conteo que son muy utilizadas.....	174
7.1.1	Experimentos de pasos múltiples	174
7.2	Simular lanzamiento de una moneda	174
8	Realizar técnicas de conteo mediante principio aditivo y multiplicativo en R	176
9	Utilizar el factorial como técnica de conteo en R	177

10	Crear permutaciones en R.....	178
10.1	Combinaciones y permutaciones	178
10.2	La libreria que necesita cargar	178
10.3	Los datos y los datos ordenados	178
10.4	Combinaciones con grupos de 2	178
10.5	permutaciones con grupos de 2.....	179
10.6	Combinaciones con grupos de 4	179
10.7	permutaciones con grupos de 4.....	180
10.8	permutaciones con grupos de 6.....	181
10.9	Analizando datos con COMBINACIONES de grupos de 6	181
10.9.1	¿en cuántas combinaciones de grupos seis está ‘Adriana’ en la primera columna? 181	
10.9.2	¿en cuántas combinaciones de grupos seis está ‘Adriana’ en la primera o en la segunda o en la tercera columna?	184
10.9.3	¿en cuántas combinaciones de grupos seis está ‘Paty’ en la cuarta o en la quinta o en la sexta columna?.....	187
10.9.4	¿en cuántas combinaciones de grupos seis está ‘Olga’ en la cuarta o en la quinta o en la sexta columna?.....	190
10.9.5	¿en cuántas combinaciones de grupos seis está ‘Olga’ en cualquier columna de la 1 a la 6? 192	
10.9.6	¿en cuántas combinaciones de grupos seis están ‘Adriana’, ‘Mary’, ‘Olga’ y ‘Paty’ en cualquier columna de la 1 a la 6.....	195
10.10	Analizando datos con PERMUTACIONES de grupos de 6	196
10.10.1	¿en cuántas permutaciones de grupos seis está ‘Adriana’ en la primera columna 196	
10.10.2	¿en cuántas permutaciones de grupos seis está ‘Adriana’ en la primera o en la segunda o en la tercera columna?	196
10.10.3	¿en cuántas permutaciones de grupos seis está ‘Paty’ en la cuarta o en la quinta o en la sexta columna?.....	197
10.10.4	¿en cuántas permutaciones de grupos seis está ‘Olga’ en la cuarta o en la quinta o en la sexta columna?.....	197
10.10.5	¿en cuántas permutaciones de grupos seis está ‘Olga’ en cualquier columna de la 1 a la 6? 198	
10.10.6	¿en cuántas permutaciones de grupos seis están ‘Adriana’, ‘Mary’, ‘Olga’ y ‘Paty’ en cualquier columna de la 1 a la 6.....	198
11	Crear combinaciones en R.....	202

11.1	Combinaciones y permutaciones con números	202
11.2	Combinaciones con números.....	202
11.3	Permutaciones con números	203
11.4	con grupos de 3. Combinaciones de 50 alumnos.....	204
11.5	con grupos de 3. Permutaciones de 50 alumnos	204
12	Desarrollar diagramas de árbol en R	205
13	Elaborar práctica mediante teorema del binomio en R.....	205
14	Elaborar prácticas en R con los temas de Teoría de probabilidad.....	206
14.1	Conceptos.....	206
14.2	Una introducción a la teoría de probabilidad	207
14.2.1	La población	207
14.2.2	Determinar subconjuntos de una población.....	207
14.2.3	Operaciones con conjuntos y complementos	207
14.2.4	Probabilidades por la frecuencia.....	208
14.3	Simulando. Teorema de Bayes	209
14.3.1	Un conjunto de personas de una empresa X	209
15	Elaborar prácticas para determinar valores de variables aleatorias discretas y continuas	211
15.1	Conceptos.....	211
15.2	Distribucion de probabilidad uniforme. Caso licitaciones	211
15.3	Probabilidad de que la baja licitación de embarque del próximo contrato nacional:	212
15.4	a) Sea inferior a 22 000 dólares.....	212
15.5	b) Rebase los 24 000 dólares.....	213
15.6	c) $P(X < x) = 1/5$	214
15.7	d) $P(X > x) = 2/5$	214
15.8	Generando números aleatorios conforme distribución de probabilidad uniforme con <i>runif()</i>	214
16	Prácticas de distribuciones de probabilidad en R	216
16.1	Distribución binomial	216
16.2	Probabilidad binomial. Caso lectores de novela	216
16.3	Valores de los parámetros	217
16.3.1	1 ¿Cuál es la probabilidad de que en el grupo, hayan leido la novela 2 personas?. $x = 2$	217
16.3.2	2 ¿y cómo máximo 2?. Acumulado hasta 2, incluye: 0, 1 y 2.....	217
16.3.3	Plot la probabilidades desde 0 hasta 4.....	217

16.4	Distribución de probabilidad de Poisson.....	218
16.5	Probabilidad de Poisson. Caso fallas de componentes electrónicos	219
16.5.1	¿cuál es la probabilidad de que falle un componente en 25 horas?.....	219
16.5.2	?y de que fallen no m?s de 2 componentes en 50 horas?	220
16.5.3	¿cuál es la probabilidad de que fallen por lo menos 10 en 125 horas?	221
16.5.4	Graficando las probabilidad $p(x \leq 10)$	222
16.6	Distribución de probabilidad normal	223
16.7	Probabilidad Normal Bolsa de New York	224
16.7.1	Generando la muestra.....	225
16.7.2	a)Calcule la media y la desviación estándar.....	225
16.7.3	b. ¿Cuál es la probabilidad de que, en un día elegido al azar, el volumen de negociaciones en la mañana temprano sea superior a 180 millones de acciones?	225
16.7.4	c) ¿Cuál es la probabilidad de que, en un día elegido al azar, el volumen de negociaciones en la mañana temprano sea superior a 230 millones de acciones?	225
16.7.5	d. ¿Cuántas acciones deberán ser negociadas para que el volumen de negociaciones en la mañana temprano de un día determinado pertenezca al 5% de los días de mayor movimiento?	226
16.8	Probabilidad normal. Caso alumnos de sistemas.....	226
16.8.1	Probabilidades conforme a Distribución Normal.....	226
16.8.2	Los datos, la media y la desviación estándar.	227
16.8.3	Densidad de los datos conforme la media y la desviación estándar del promedio	229
16.8.4	Gráfica de gauss conforme probabilidad normal de acuerdo a su densidad y a la variable promedio	231
16.8.5	* ¿Qué probabilidad existe de que el promedio de los alumnos de sistemas al final del semestre sea de 85 ?.....	233
16.8.6	* ¿Cual es la probabilidad de que el promedio de los alumnos de sistemas esté entre 86 y 90?.....	233
16.8.7	¿Será posible afirmar que el 20% o más, los alumnos de sistemas obtendrán un promedio superior a 90 ?	234
16.8.8	¿Cuál será el promedio del 10% de los alumnos de sistemas ?	234
17	Prácticas de regresión lineal	235
18	Prácticas en R para muestreo, Estimaciones y pruebas de hipótesis	235
	Conclusiones	236

Introducción.

Este manual de prácticas, se refiere al resultado de trabajo académico del periodo sabático de Agosto 2018-Agosto 2019.

Este manual de prácticas se relaciona con la materia de probabilidad y estadística de la carrera de Ingeniería en Sistemas Computacionales.

Se verá un conjunto de 19 prácticas en R, utilizando para ello R Studio como entorno de trabajo, cada práctica se relaciona con un objetivo especificado, además cada una se asocia a un tema de alguna de las unidades de la materia de probabilidad y estadística.

La herramienta sugerida como estrategia didáctica es el uso del lenguaje de programación “R”. Este lenguaje permite el tratamiento de los datos por medio del análisis descriptivo y visualización gráfica de datos, estos elementos ayudan al análisis, la interpretación y toma de decisiones en los estudiantes. La computación en la nube que se sugiere como estrategia didáctica es el servicio RPubs que vienen incluido en la instalación de “R” y su entorno de trabajo (IDE) RStudio. (Pizarro, Calzada, & Valdez, 2018)

Cada práctica viene acompañada de un conjunto de datos que servirá de fuente para aplicar las técnicas estadísticas y probabilísticas según los temas de la materia.

En la tabla siguiente, muestra el objetivo general del manual y el desglose de los temas conforme se presentaron al principio del periodo sabático, indicando en tres columnas nombre, objetivo y temas con que se relaciona.

Objetivo General del Manual:

Realizar prácticas de R utilizando el IDE RStudio creando un portafolio de evidencias en la nube mediante servicio RPubs que permita asociar los temas de la clase de probabilidad y estadística para las carreras de Ingeniería en Sistemas Computacionales.

Todas las prácticas deberán ser incorporadas en R, integrando elementos en formato Markdown tales como datos del autor, comentarios, código, descripción e interpretación de la práctica.

Núm	Nombre de la Práctica	Objetivo Específico	Temas con los que se relacionan
1	Instalación del entorno de desarrollo RStudio y manipulación de una población de datos	Instalar RStudio, manipular los primeros datos. Aspectos básicos de R	1.1 Conceptos básicos de estadística
2	Creación de prácticas para utilización de RPubs como servicio en la nube	RPubs como servicio en la nube para creación de evidencia de prácticas del manual.	1.1. Población, Muestra aleatoria, parámetros aleatorios.
3	Generar datos agrupados, no agrupados, y aspectos de frecuencia	Mediante datos almacenados a variables de R generar data.frame para agrupar y determinar frecuencias. Importar datos y determinar agrupación y frecuencias	1.2. Descripción de datos: Datos agrupados y no agrupados, Frecuencia de clase, Frecuencia relativa, Punto medio, Límites.
4	Generar operaciones y valores usando comandos en R para determinar medidas de tendencia central	Utilizar comandos en R para determinar medidas de tendencia central. Mean, median, mod. Con datos importados y con la finalidad de crear su correspondiente interpretación.	1.3 Medidas de tendencia central: Media aritmética, , Mediana, Moda.
5	Generar operaciones y valores usando comandos para determinar medidas de dispersión.	Utilizar comandos en R para determinar medidas de dispersión con datos importados y con la	1.3. Medidas de dispersión, Varianza, Desviación estándar,

		finalidad de crear su correspondiente interpretación.	Desviación media, Desviación mediana, Rango
6	Crear agrupación de datos con diversos parámetros en R	Crear agrupación de datos con diversos parámetros en R. Crear evidencia en RPubs	1.4 Parámetros para datos agrupados.
7	Crear distribución y graficación de fecuencias en R	Crear distribución y graficación de fecuencias en R. Crear evidencia en RPubs	1.5 Distribución de frecuencias.
8	Realizar técnicas de conteo en R	Mediante R efectuar técnicas de conteo. Crear evidencia en RPubs	2.1 Técnicas de Conteo
9	Realizar técnicas de conteo mediante principio aditivo y multiplicativo en R	Realizar técnicas de conteo mediante principio aditivo y multiplicativo en R. Crear evidencia Rpuds	2.1.1 Principio aditivo. 2.1.2 Principio multiplicativo.
10	Utilizar el factorial como técnica de conteo en R	Utilizar el factorial como técnica de conteo en R. Crear evidencian RPubs	2.1.3 Notación Factorial.
11	Crear permutaciones en R	Crear permutaciones en R	2.1.4 Permutaciones.
12	Crear combinaciones en R	Crear combinaciones en R	2.1.5 Combinaciones.
13	Desarrollar diagramas de árbol en R	Desarrollar diagramas de árbol en R	2.1.6 Diagrama de Árbol.
14	Elaborar práctica mediante teorema del binomio en R	Desarrollar teorema del binomio en R. Crear evidencia RPubs	2.1.7 Teorema del Binomio.
15	Elaborar prácticas en R con los temas de Teoría de probabilidad.	Elaborar prácticas en R con los temas de Teoría de probabilidad, incorporando evidencia en RPubs	2.2 Teoría elemental de probabilidad. 2.3 Probabilidad de Eventos: Definición de espacio muestral, definición de evento, simbología, unión, intersección, diagramas de Venn. 2.4 Probabilidad con Técnicas de Conteo: Axiomas,

			<p>Teoremas.</p> <p>2.5 Probabilidad condicional: Dependiente, Independiente.</p> <p>2.6 Ley multiplicativa.</p> <p>2.7 Eventos independientes: Regla de Bayes.</p>
16	Elaborar prácticas para determinar valores de variables aleatorias discretas y continuas	Prácticas incorporadas en RPubs	<p>3.1. Variables aleatorias discretas</p> <p>3.2. Variables aleatorias continuas</p>
17	Prácticas de distribuciones de probabilidad en R	Crear prácticas mediante RStudio y lenguaje de programación e R incorporando las mismas en el servicio en la nube RPubs como portafolio de evidencia	<p>4.1 Función de probabilidad.</p> <p>4.2 Distribución binomial.</p> <p>4.3 Distribución hipergeométrica.</p> <p>4.4 Distribución de Poisson.</p> <p>4.5 Distribución normal.</p> <p>4.6 Distribución T-student.</p> <p>4.7 Distribución Chi cuadrada.</p> <p>4.8 Distribución F.</p>
18	Prácticas de regresión lineal	Prácticas en R y Rpubs asociando los temas de regresión lineal	<p>5.1 Regresión y correlación.</p>
19	Prácticas en R para muestreo, Estimaciones y pruebas de hipótesis	Prácticas en R para muestreo, Estimaciones y pruebas de hipótesis	<p>6.1 Muestreo</p> <p>6.2 Estimación</p> <p>6.3 Prueba de hipótesis</p>

1 Instalación del entorno de desarrollo RStudio y manipulación de una población de datos.

Esta práctica tiene por objetivo Instalar RStudio, manipular los primeros datos. Aspectos básicos de R.

1.1 Conceptos

R es un conjunto integrado de instalaciones de software para la manipulación de datos, cálculo y visualización gráfica de datos.

Entre otras cosas tiene

- una instalación eficaz de manejo y almacenamiento de datos,
- un conjunto de operadores para cálculos en matrices,
- una colección grande, coherente e integrada de herramientas intermedias para el análisis de dato
- facilidades gráficas para el análisis y visualización de datos directamente en la computadora o en papel, y
- un lenguaje de programación bien desarrollado, simple y efectivo (llamado "S") que incluye: condicionales, bucles, funciones recursivas definidas por el usuario e instalaciones de entrada y salida. (En efecto la mayoría de las funciones proporcionadas por el sistema están escritas en el lenguaje S.)

R es en gran medida un vehículo para desarrollar nuevos métodos de análisis de datos interactivos, ha sido muy extendido por una gran colección de paquetes.

R es ideal para el aprendizaje estadístico. Sin embargo, en R se han implementado técnicas estadísticas modernas. Algunos de estos están integrados en la base el entorno R, muchas de estas técnicas se suministran como paquetes. Hay alrededor de 25 paquetes suministrados con R (llamados paquetes "estándar" y "recomendados") y muchos más están disponibles a través de Familia CRAN de sitios de Internet (a través de <https://CRAN.R-project.org>) y en otros lugares.

1.2 Instalar R

Para descargar R se debe dirigir al enlace siguiente: <https://www.r-project.org/>

Aquí reconocer debe recoger la versión de R de acuerdo a las características de su equipo. Buscar una instalación como la versión de R 3.6.1 for Windows (81 megabytes, 32/64 bit).

The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To [download R](#), please choose your preferred CRAN mirror.

If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.

News

- **R version 3.3.3 (Another Canoe) prerelease versions** will appear starting Friday 2017-02-24. Final release is scheduled for Monday 2017-03-06.
- **useR! 2017** (July 4 - 7 in Brussels) has opened registration and more at <http://user2017.brussels/>
- Tomas Kalibera has joined the R core team.
- The R Foundation welcomes five new ordinary members: Jennifer Bryan, Dianne Cook, Julie Josse, Tomas Kalibera, and Balasubramanian Narasimhan.
- **R version 3.3.2 (Sincere Pumpkin Patch)** has been released on Monday 2016-10-31.
- **The R Journal Volume 8/1** is available.
- The **useR! 2017** conference will take place in Brussels, July 4 - 7, 2017.
- **R version 3.3.1 (Bug in Your Hair)** has been released on Tuesday 2016-06-21.
- **R version 3.2.5 (Very, Very Secure Dishes)** has been released on 2016-04-14. This is a rebranding of the quick-fix release 3.2.4-revised.
- **Notice XQuartz users (Mac OS X)** A security issue has been detected with the Sparkle update

Para mayor referencia se sugiere este enlace: <https://r-spanish.com/2017/06/19/como-instalar-r-como-instalar-rstudio/>

1.3 Instalar R Studio

Una vez hecho esto, se puede proceder a instalar **RStudio**. R studio es un entorno de desarrollo, es un programa mucho más avanzado y con una interfaz más intuitiva para programar en lenguaje **R**.

Para ello, se debe ingresar en la web de **RStudio**, <https://www.rstudio.com/>. Allí, navegando un poco hacia abajo, se encuentra el enlace llamado “Download”.

Descargar la versión escritorio gratuita (*RStudio Desktop FREE*). Con esto es más que suficiente para programar en **R** y ejecutar tareas bastante avanzadas.

Choose Your Version of RStudio

RStudio is a set of integrated tools designed to help you be more productive with R. It includes a console, syntax-highlighting editor that supports direct code execution, and a variety of robust RStudio features.

RStudio Desktop Open Source License FREE	RStudio Desktop Commercial License \$995 per year	RStudio Server Open Source License FREE	RStudio Server Pro Commercial License \$9,995 per year	
Integrated Tools for R	●	●	●	
Priority Support		●	●	
Access via Web Browser		●	●	
Enterprise Security			●	
Project Sharing			●	
Manage Multiple R Sessions & Versions			●	
Admin Dashboard			●	
Load Balancing			●	
License	AGPL FREE	Commercial \$995/yr	AGPL FREE	Commercial \$9,995/yr
Pricing	DOWNLOAD Learn More	BUY NOW Learn More	DOWNLOAD Learn More	DOWNLOAD Learn More

Have Questions?

[TALK TO THE SALES TEAM](#)

RStudio Desktop 1.0.136 — Release Notes

RStudio requires R 2.11.1+. If you don't already have R, download it [here](#).

Installers for Supported Platforms

Installers	Size	Date	MD5
RStudio 1.0.136 - Windows Vista/7/8/10	61.9 MB	2016-12-21	93d4f39777e2317a0a14899c1c1266f3adea03a3fcfcffefed3e05c1ae
RStudio 1.0.136 - Mac OS X (64-bit)	71.1 MB	2016-12-21	0a20fb89d8aaeb3b9b5229ad48ddadd2c5
RStudio 1.0.136 - Ubuntu 12.04+ (Debian 8) (64-bit)	65.5 MB	2016-12-21	2e73bb8a32a9fbaf96251cecf8d41340
RStudio 1.0.136 - Ubuntu 12.04+ (Debian 8) (64-bit)	92.1 MB	2016-12-21	fcd179a785bf70f939a34169daef5d
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	64.7 MB	2016-12-21	2b3a1486ed380704e45849d0bf155245
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	88.7 MB	2016-12-21	ea2a262f65de02f568f48edc1c099298

Zip/Tarballs

Zip/tar archives	Size	Date	MD5
RStudio 1.0.136 - Windows Vista/7/8/10	117.6 MB	2016-12-21	f41599b9f5d12caab127c7cfbc9668be
RStudio 1.0.136 - Ubuntu 12.04+ (Debian 8) (32-bit)	66.2 MB	2016-12-21	fc475f953d4d42569a177d4535b122165
RStudio 1.0.136 - Ubuntu 12.04+ (Debian 8) (64-bit)	93.2 MB	2016-12-21	7cf0092653aa44fc76325a8f1325fb1f
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	68.4 MB	2016-12-21	39c89299d3be03b38098e51e99f49b8
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	88.6 MB	2016-12-21	ea2a262f65de02f568f48edc1c099298

Source Code

A tarball containing source code for RStudio v1.0.136 can be downloaded from [here](#).

250 Northern Ave, Boston, MA 02210
844-448-1212
info@rstudio.com

[R Studio](#)

DMCA
Support
ECCN

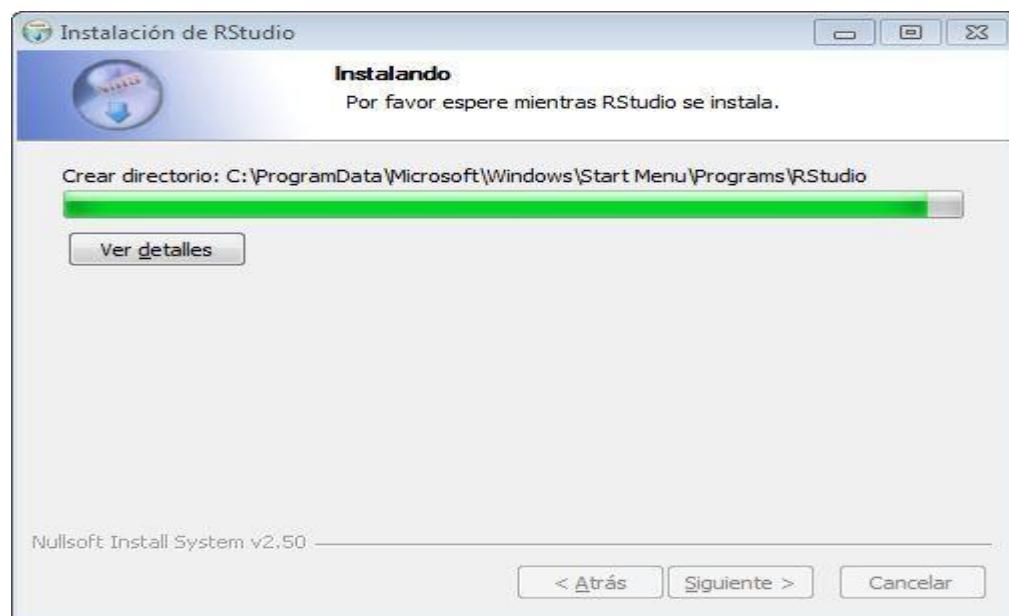
Twitter Missed #rstudioconf? Here are some tips from IDE engineer @kevin_ushey! Slides from all the forthcoming #rstats twitter sessions bhaskar_vk/sta... 4 weeks ago

Copyright 2016 RStudio | All Rights Reserved | Legal Terms

[Twitter](#) [LinkedIn](#) [Facebook](#) [RSS](#) [Email](#) [Print](#)

Una vez descargado, al igual que se hizo el instalador de **R**, se ejecuta y comienza a pulsar “Siguiente”. En este caso sí hace falta algo más de espacio en disco que para **R**, aunque sigue siendo extraño ver un equipo actual sin esta cantidad de espacio (unos 500MB).

Se selecciona la carpeta de instalación (por lo general es válida la opción por defecto, C:\Program Files\RStudio), y la carpeta del menú inicio correspondiente (también vale la opción por defecto, “RStudio”):



Y con esto hemos finalizado la instalación de R y RStudio.

1.4 Primeros pasos con R y R Studio.

Al tener instalado R y R studio es necesario comenzar a utilizarlo.

Abrir el ejecutable de R Studio que se encuentra normalmente en C:\Program Files\RStudio\bin\R Studio

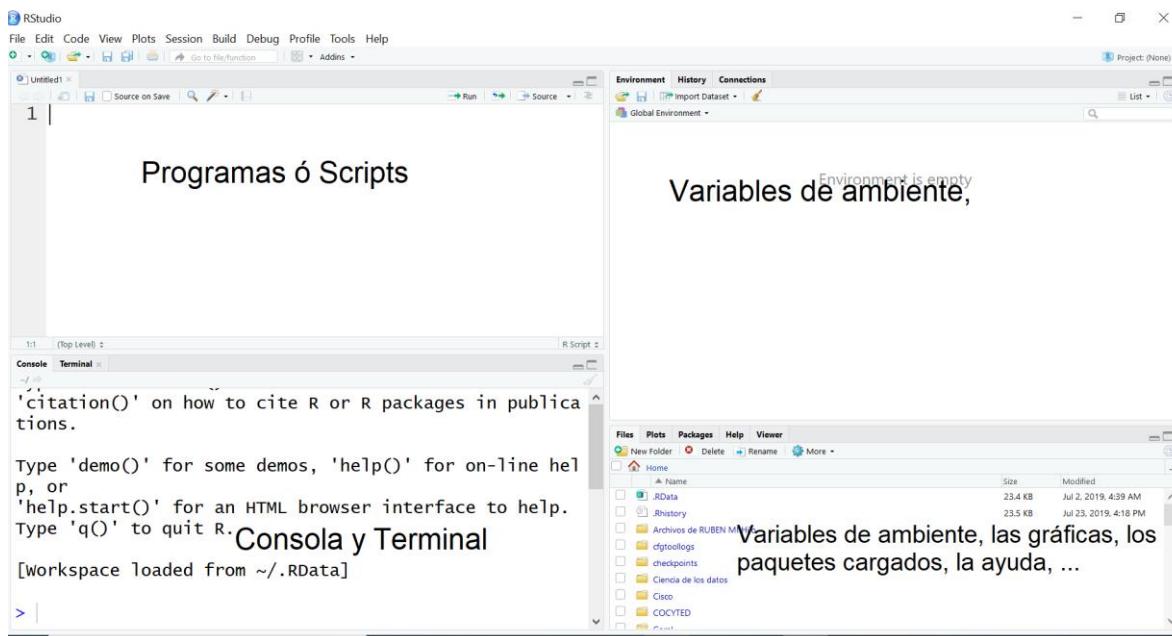


Al ingresar a R Studio, se visualizan las principales ventanas, el menú principal y otras opciones:

El menú principal en el entorno del idioma inglés, hay opciones muy tradicionales que las de cualquier ambiente de desarrollo,: File para abrir, guardar, cerrar, entre otras cosas; le siguen: Edit, Code View que son opciones que se van utilizando conforme el usuario se familiarice con el entorno. Luego una opción muy utilizada la de Session que permite entre otras cosas utilizar espacios de trabajo o sesiones de trabajo, es decir lo que el usuario realizó cuando entró y cuando cerró, esto se recomienda para guardar los comandos ejecutados, ahí mismo la opción de definir carpetas de trabajo con *Set Working Directory* que es una opción muy sugerida. De las opciones del menú principal una opción también utilizada y sugerida es la de Tools para descargar paquetes y finalmente Help para ayuda y versiones.

Cabe hacer mención que muchas de estas opciones se pueden utilizar mediante comandos en la ventana de comandos abajo izquierda.

Hay cuatro ventanas: describiendo de arriba abajo y de izquierda a derecha: Arriba izquierda es una ventana que se abre cuando se está editando programas (scripts) de R; a la derecha arriba es la ventana que permite entre otras cosas ver las variables que se van creando y los datos que se van generando.; abajo izquierda, la ventana de comandos y terminal que son para ejecutar comandos directamente y para visualizar resultados; a la derecha abajo un ventana que permite entre otras cosas conocer el directorio visualmente de trabajo, otra ventana plot para graficar datos; la ventana de packages para visualizar paquetes descargados además de descargar paquetes, la ventana de help entre otras.



1.5 Crear programa hola mundo

En la parte izquierda File/Abrir nuevo script y escribir lo siguiente:

print ("Hola Mundo"), marcar y ejecutar. Este código marcado y ejecutado se replica en la ventana de consola y se visualiza el resultado ahí mismo.

```
> print ("Hola mundo")
[1] "Hola mundo"
```

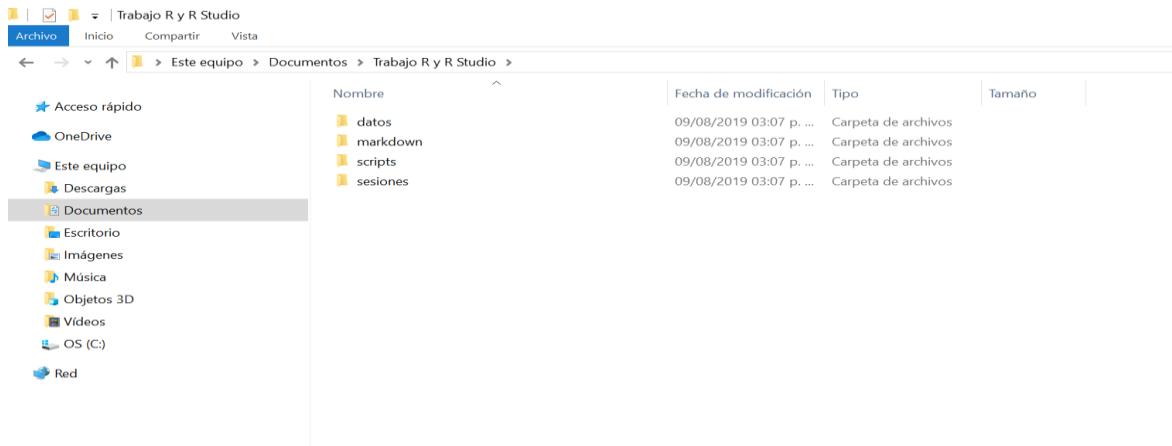
Se puede guardar el script como holamundo y se guarda con extensión .R. Se recomienda guardarlo en una carpeta llamada script en el lugar de su preferencia.

1.6 Organizando el trabajo en R Studio

Al lector de este manual se le recomienda generar una carpeta en c:/Documentos que se llame / Trabajo R y R Studio/ y dentro de ésta, cuatro subcarpetas que se llamen respectivamente /datos; /scripts; /markdown; /sesiones.

- La carpeta datos contendrá fuentes de datos utilizando en las prácticas,
- La carpeta markdown almacenará físicamente los archivos en formato tal, (HTML y código R incrustado) que se pueden subir a la nube, a un espacio de trabajo del usuario,
- La carpeta scripts, son los programas que el usuario realiza,
- La carpeta sesiones contendrá las sesiones del usuario en caso de pretender guardarlas

La siguiente imagen muestra el resultado de lo anterior.

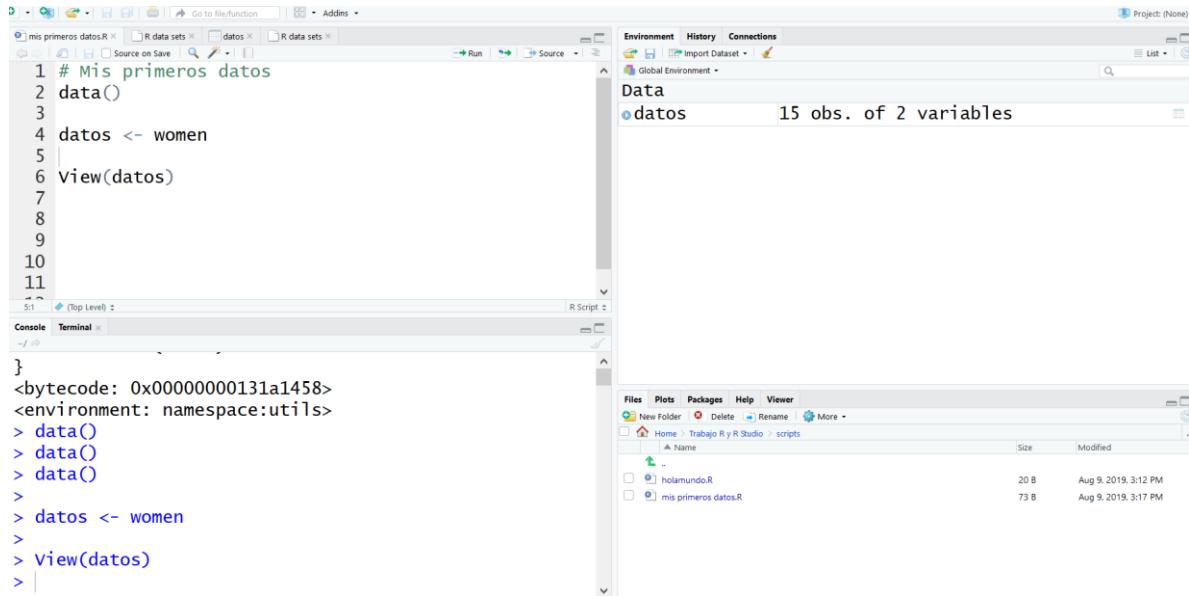


En este momento si no ha guardado su primer programa holamundo.R guardararlo en /Documentos/Trabajo R y RStudio/scripts/holamundo.R

1.7 Mis primeros datos

Antes de hacer el nuevo programa se recomienda indicar un directorio en donde almacenar de manera directa los scripts, utilizando el asistente en la ventana

derecha abajo con la opción File designar la ruta C:\Users\Usuario\Documents\Trabajo R y R Studio\scripts como se muestra en la imagen.



The screenshot shows the RStudio interface. On the left, the 'Console' tab is active, displaying R code and its execution results. The code is as follows:

```
1 # Mis primeros datos
2 data()
3
4 datos <- women
5
6 view(datos)
7
8
9
10
11
```

Below the code, the results of the 'data()' command are shown:

```
<bytecode: 0x00000000131a1458>
<environment: namespace:utils>
> data()
> data()
> data()
>
> datos <- women
>
> view(datos)
>
```

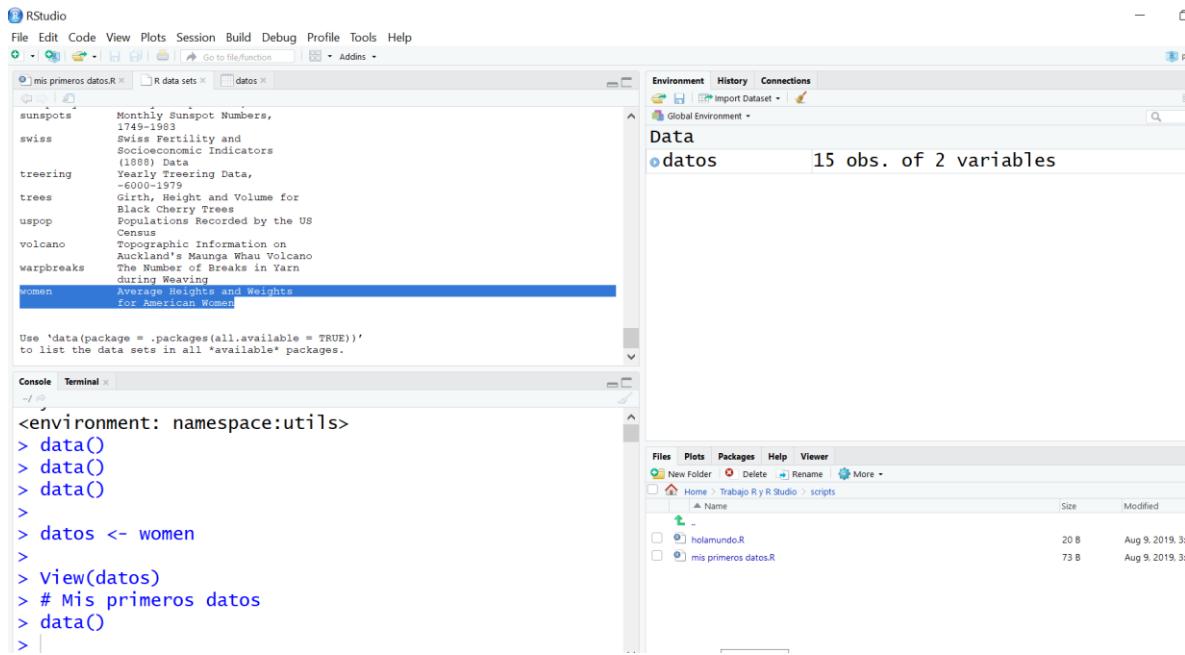
On the right side of the interface, the 'Global Environment' pane is open, showing the variable 'datos' which contains 15 observations of 2 variables. Below the environment pane, the 'Files' tab of the file browser is selected, showing a folder structure under 'Trabajo R y R Studio > scripts'. The folder contains two files: 'holamundo.R' (size 20 B, modified Aug 9, 2019, 3:12 PM) and 'mis primeros datos.R' (size 73 B, modified Aug 9, 2019, 3:17 PM).

Ahora bien, abrir un nuevo script con el siguiente código:

```
# Mis primeros datos
data()
```

Pulsar ejecutar marcando data() y se observarán conjuntos de datos que vienen instalados por default en R y R Studio.

Se abre una ventana en la parte superior izquierda a la altura de los scripts que se llama R data set, deslizar en dicha ventana hasta encontrar el conjunto de datos de women.



Habiendo identificado el conjunto de datos 'women', continuar con el programa y escribir las siguientes líneas:

```

# Mis primeros datos
data()

datos <- women

View(datos)

```

Guardar el programa con el nombre 'mis primeros datos', por default se guarda con extensión R. dentro de la carpeta scripts.

Ejecutar desde datos hasta View(datos) y se visualiza en una nueva ventana arriba izquierda con los datos del conjunto de datos 'women'. También se genera una nueva variable llamada datos que contiene los datos de 'women'. Este conjunto de datos se asocia con datos de, el peso y altura de mujeres americanas.

También se observa del lado derecho que la variable datos contiene 15 observaciones y 2 variables.

The screenshot shows the RStudio interface. In the Environment pane, there is a table named 'datos' with 15 observations and 2 variables: 'height' and 'weight'. The data is as follows:

	height	weight
1	58	115
2	59	117
3	60	120
4	61	123
5	62	126
6	63	129
7	64	132
8	65	135
9	66	139
10	67	142
11	68	146
12	69	150

In the Console pane, the following R code is visible:

```
<environment: namespace:utils>
> data()
> data()
> data()
>
> datos <- women
>
> View(datos)
> # Mis primeros datos
> data()
> |
```

The File Browser pane shows a folder structure under 'Trabajo R y R Studio / scripts':

- ..
- holamundo.R (20 B, Aug 9, 2019, 3:12 PM)
- mis primeros datos.R (73 B, Aug 9, 2019, 3:17 PM)

En caso de no encontrar el conjunto de datos 'women', se puede elegir cualquier otro conjunto de datos y guardarlo en la variable datos y replicar el proceso.

Se recomienda el siguiente enlace para enriquecer el tema: <https://r-spanish.com/2017/07/10/operaciones-basicas-en-r/> y <https://bookdown.org/boscomendoza/r-principiantes4/> (Mendoza Vega, 2019)

2 Creación de prácticas para utilización de RPubs como servicio en la nube

Esta práctica tiene como objetivo generar programas en forma markdown para poder publicar evidencias de programación R en la nube.

Para comenzar es necesario describir algunos conceptos:

2.1 Conceptos

R Markdown provee un marco de referencia para la ciencia de datos, combinando el código, los resultados y comentarios descriptivos. Los documentos de R Markdown son completamente reproducibles y soportan docenas de formatos de salida tales como PDFs, archivos de Word, presentaciones y más.

Los archivos R Markdown están diseñados para ser usados de tres maneras:

Para comunicarse con los tomadores de decisiones, quienes desean enfocarse en las conclusiones, no en el código que subyace al análisis.

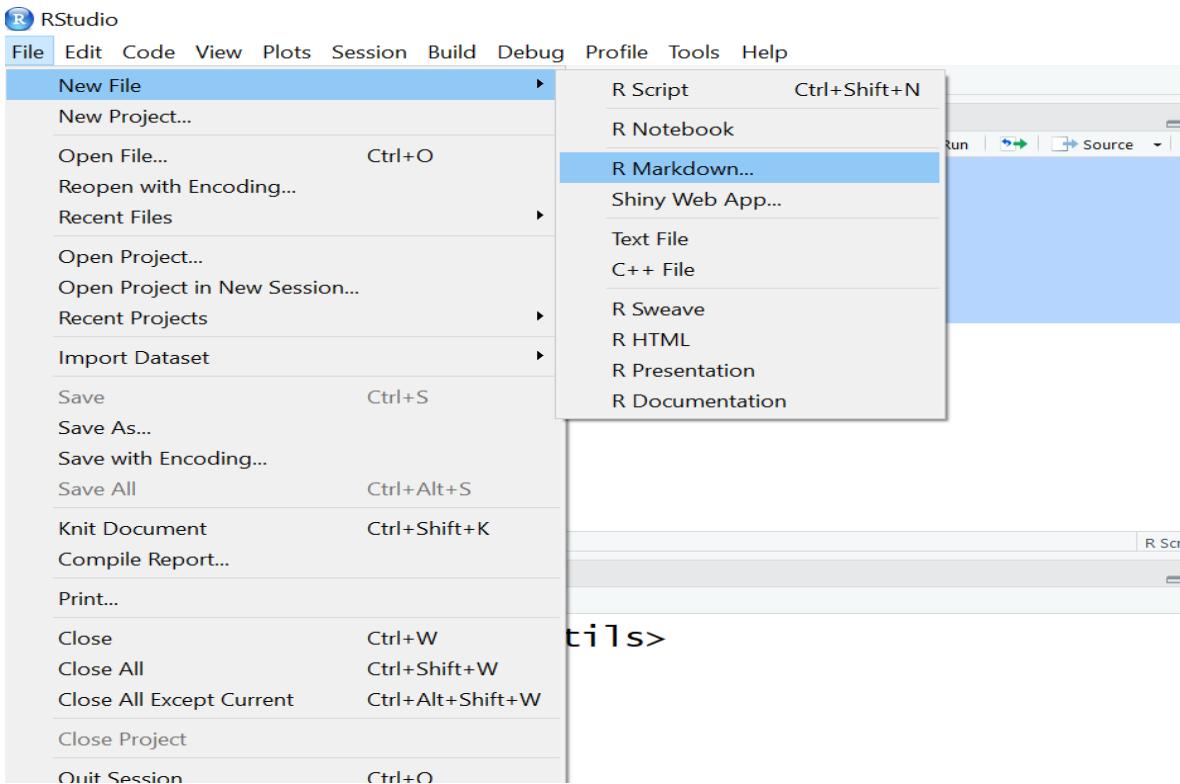
Para colaborar con otros científicos de datos (¡incluyendo a tu yo futuro!), quienes están interesados tanto en tus conclusiones como en el modo en el que llegaste a ellas (es decir, el código).

Por otra parte los archivos Markdown que se podrán publicar en la nube mediante la opción de R Studio llamada R Pubs. RStudio le permite aprovechar el poder de R Markdown para crear documentos que entrelazan su escritura y la salida de su código R.

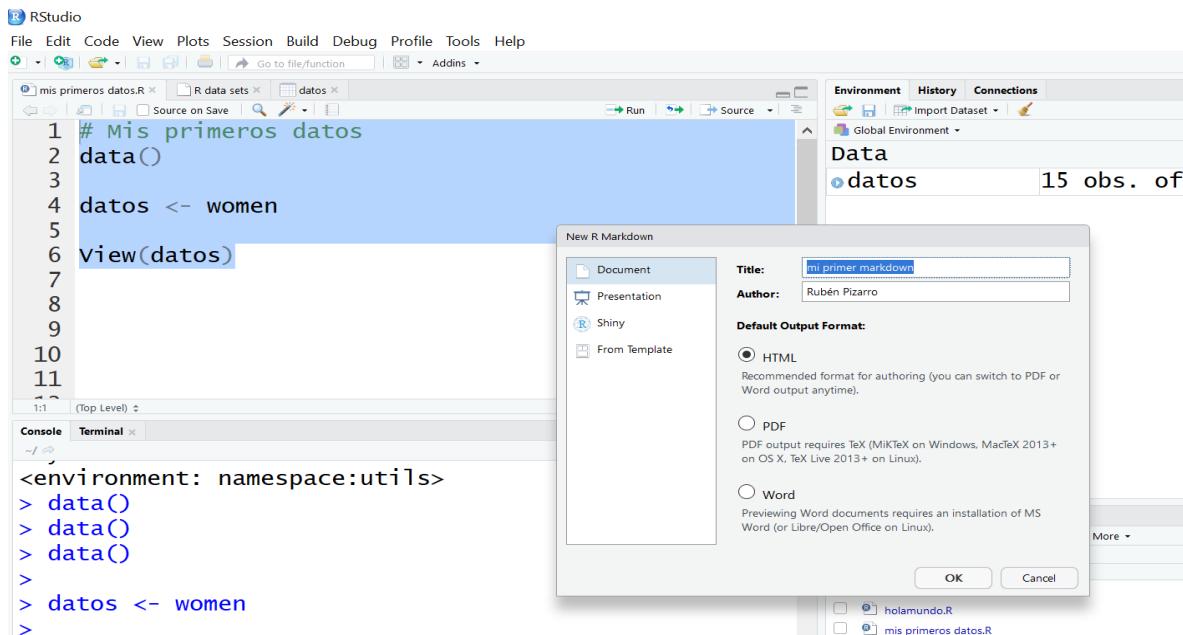
R Pubs es un servicio de alojamiento de archivos HTML Markdown, en donde se requiere para su uso, el registro de una cuenta personalizada de usuario. Correo electrónico e identificador de usuario.

2.2 Crear el primer archivo markdown

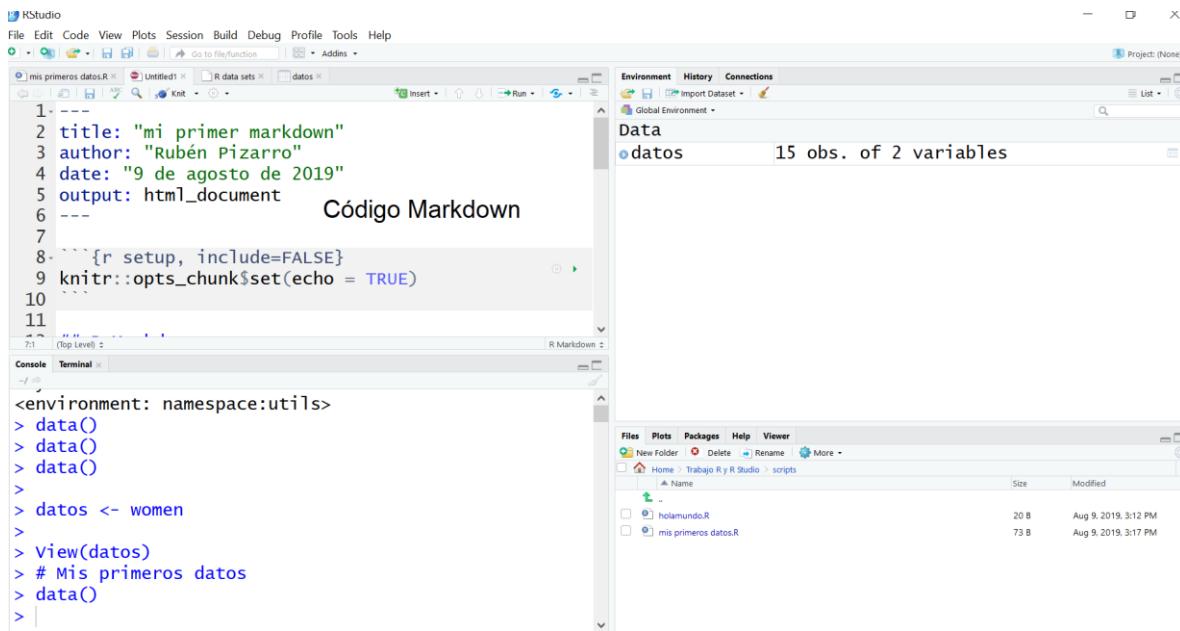
En R Studio abrir nuevo archivo de tipo markdown



Proporcionar el nombre del archivo: mi primer markdown y presionar OK.



Luego Aparece un programa en la ventana arriba izquierda de R Studio con un código parecido a lo que se muestra:



Para efecto de esta práctica marcar a partir de la línea 8 y eliminar todas las líneas.
Esto se hará para hacer un código más entendible para el lector.

Comenzar a escribir estas etiquetas **### Mi primer markdown** y **#### Cargar unos datos** en el archivo markdown solo después de los encabezados. Como sigue y como se muestra en la imagen

```

---
title: "mi primer markdown"
author: "Rubén Pizarro"
date: "9 de agosto de 2019"
output: html_document
---

### Mi primer markdown
#### Cargar unos datos

```

Seguido poner la líneas que ya conocemos del primer programa ‘mis primeros datos.R’. Interesa el código: `datos <- women`.

Como es código R se debe incrustar y esto se hace mediante con el asistente. En el programa markdown, en la línea 10 u 11 incrustar mediante la opción insert R como se muestra en la imagen:

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu is a toolbar with various icons. The code editor window contains the following R code:

```
1 ---  
2 title: "mi primer markdown"  
3 author: "Rubén Pizarro"  
4 date: "9 de agosto de 2019"  
5 output: html_document  
6 ---  
7  
8 ### Mi primer markdown  
9 ##### Cargar unos datos  
10  
11
```

The status bar at the bottom shows "11:1" and "Cargar unos datos". A context menu is open over the line "11", with the "Insert" option selected. A submenu titled "Insert a new R chunk" is displayed, containing options for R, Python, Rcpp, SQL, and Stan.

Ahora bien, se abre un espacio para código R y poner el siguiente código

```
datos <- women  
datos  
  
str(datos)  
summary(datos)
```

Hacer una segunda incrustación de código R aproximadamente en la línea 22 o 23 con el siguiente código

```
datos <- women  
datos  
  
str(datos)  
summary(dtos)
```

El código queda de la siguiente manera:

```

1 #title: "mi primer markdown"
2 #author: "Rubén Pizarro"
3 #date: "9 de agosto de 2019"
4 #output: html_document
5
6 ---
7
8 ### Mi primer markdown
9 #### Cargar unos datos
10
11 ``{r}
12 datos <- women
13 datos
14
15 str(datos)
16 summary(datos)
17
18 ...
19
20
21 ``{r}
22 plot(datos)
23
24

```

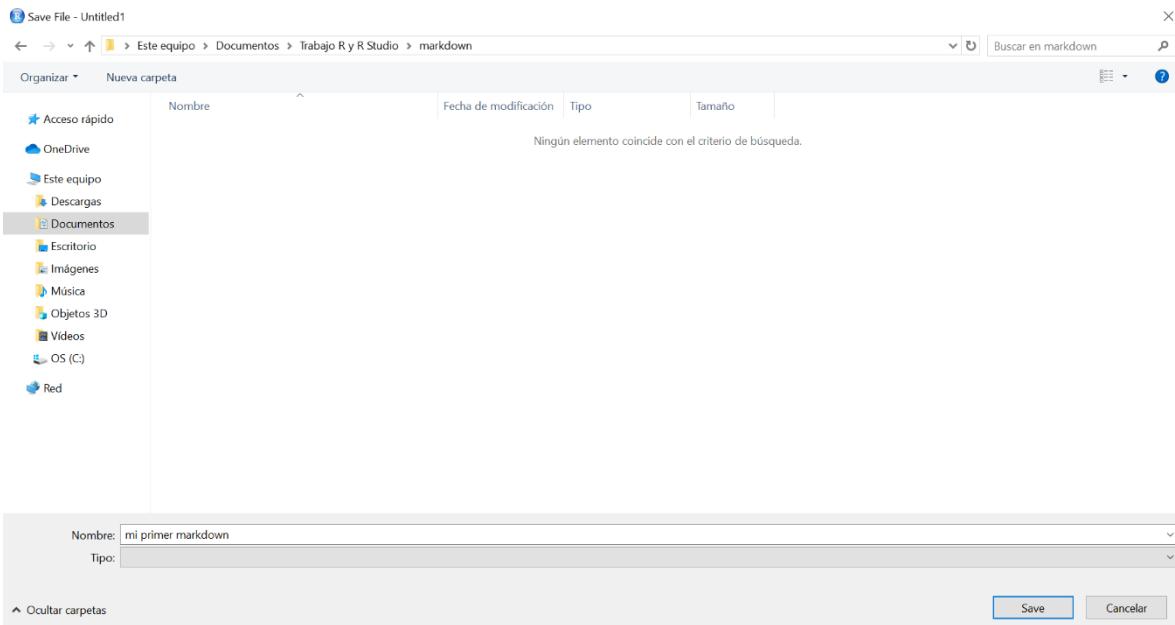
Con lo anterior se recomienda no perder las etiquetas de los caracteres ````{r}`` y ````` que aparecen al principio y al final de cada código R, es precisamente para indicar al documento HTML de markdown que es programación R.

Una vez que se tiene el código R incrustado se ejecuta todo el documento markdown en la opción KNIT de R Studio.

The screenshot shows the 'mis primeros datos.R' file open in RStudio. The 'Knit' option is highlighted in the top navigation bar. A context menu is open over the code, showing options like 'Knit to HTML', 'Knit to PDF', 'Knit to Word', 'Knit with Parameters...', 'Knit Directory', and 'Clear Knitr Cache...'. The rest of the code is identical to the previous screenshot.

Se pide que guarde el documento y se le proporciona el mismo nombre 'mi primer

markdown', se guarda en la carpeta markdown definida inicialmente para ese propósito.



Pulsar **Save** o **salvar el documento** y se genera un archivo como el que parece en pantalla.

```
## height weight
## 1 58 115
## 2 59 117
## 3 60 120
## 4 61 123
## 5 62 126
## 6 63 129
## 7 64 132
## 8 65 135
## 9 66 139
## 10 67 142
## 11 68 146
## 12 69 150
## 13 70 154
## 14 71 159
## 15 72 164

str(datos)

## 'data.frame': 15 obs. of 2 variables:
## $ height: num 58 59 60 61 62 63 64 65 66 67 ...
## $ weight: num 115 117 120 123 126 129 132 135 139 142 ...

summary(datos)
```

En caso que sea la primera vez que se genera un markdown, R Studio solicitará que instale los paquetes adecuados, en este caso, se podrá instalar automáticamente o mediante la instrucción siguiente en la consola de R Studio:

```
install.packages("rmarkdown")
```

Tratar de generar nuevamente el archivo HTML con la opción KNIT de R Studio.

2.3 Publicar en el servicio RPubs de R Studio

Una vez que se ha generado el primer archivo markdown con los datos de ‘women’ en la variable datos, se le indica a R Studio que se desea publicar mediante la opción Publish

The screenshot shows the R Studio interface with a published markdown document. The title is 'mi primer markdown'. The author is 'Rubén Pizarro' and the date is '9 de agosto de 2019'. The document content includes a section titled 'Mi primer markdown' and a section titled 'Cargar unos datos'. Below these sections are several code snippets:

```
datos <- women  
datos
```

```
##   height weight  
## 1    58     115  
## 2    59     117  
## 3    60     120  
## 4    61     123  
## 5    62     126  
## 6    63     129  
## 7    64     132  
## 8    65     135  
## 9    66     139  
## 10   67     142  
## 11   68     146  
## 12   69     150  
## 13   70     154  
## 14   71     159  
## 15   72     164
```

```
str(datos)
```

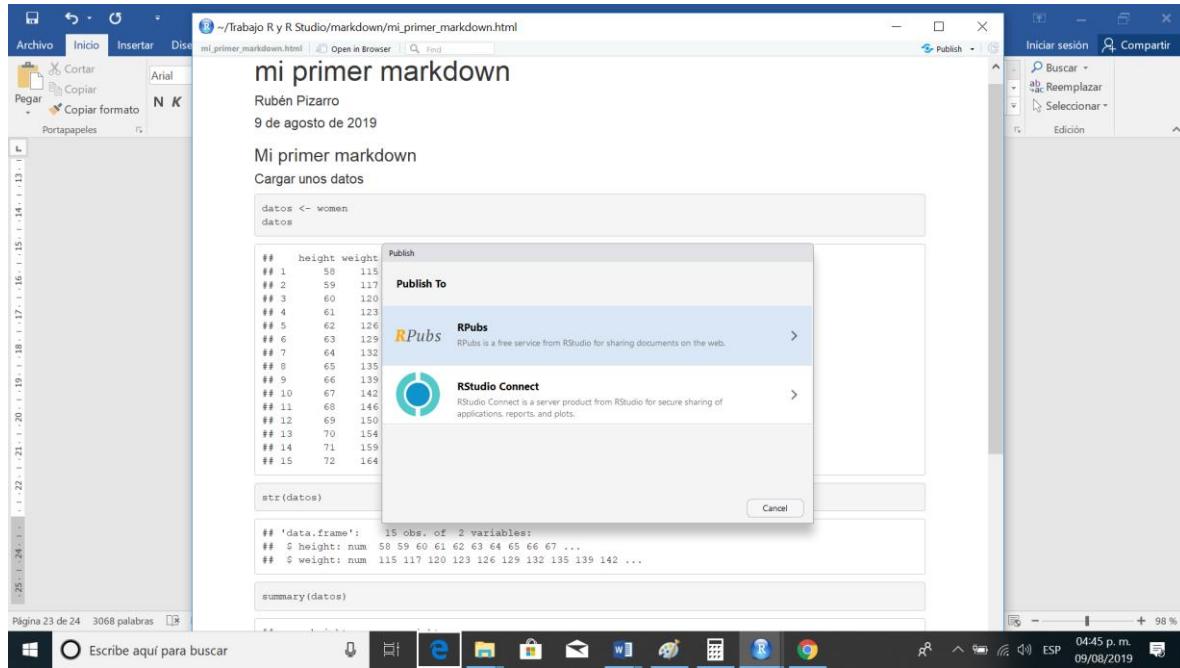
```
## 'data.frame': 15 obs. of 2 variables:  
## $ height: num 58 59 60 61 62 63 64 65 66 67 ...  
## $ weight: num 115 117 120 123 126 129 132 135 139 142 ...
```

```
summary(datos)
```

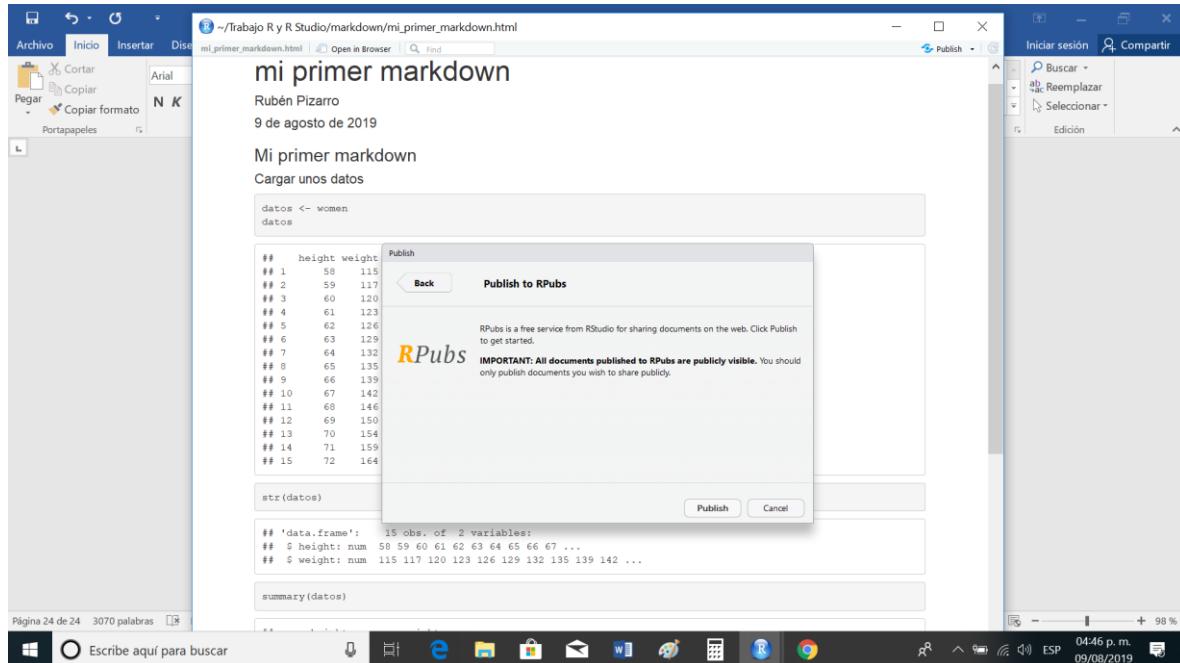
```
##
```

Con ello se presenta la opción de publicar en RPubs, para ello es necesario contar con una cuenta activa, si no la tiene ahí mismo se da la oportunidad de registrarse y continuar con el proceso.

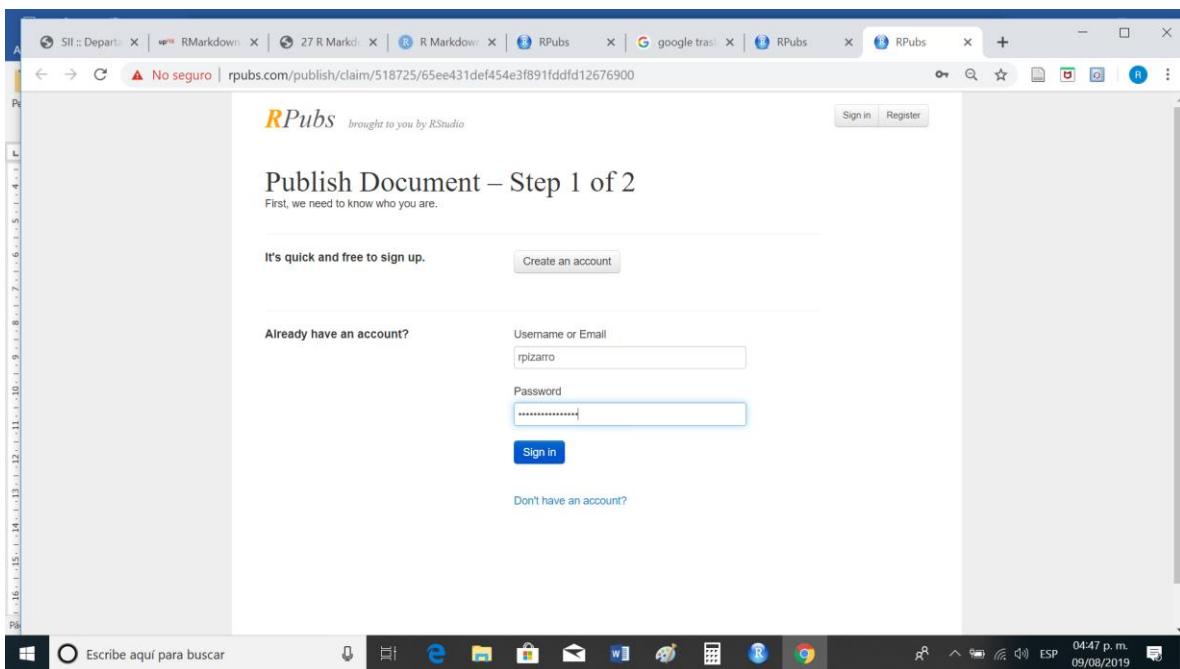
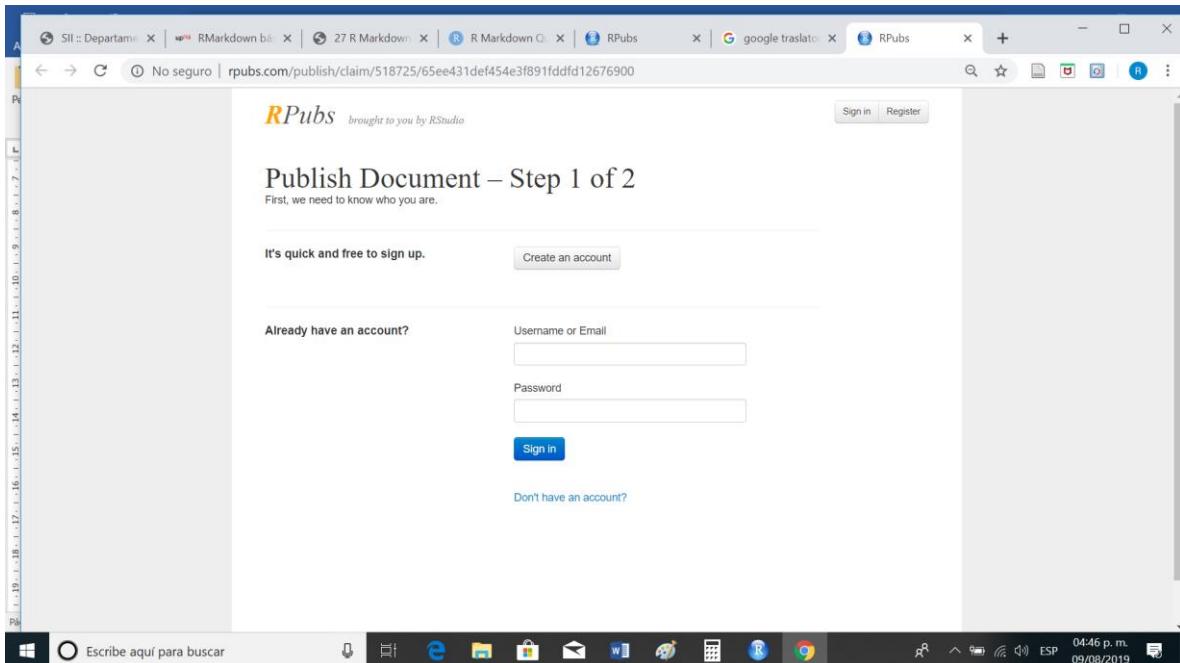
Seleccionar Rpubs



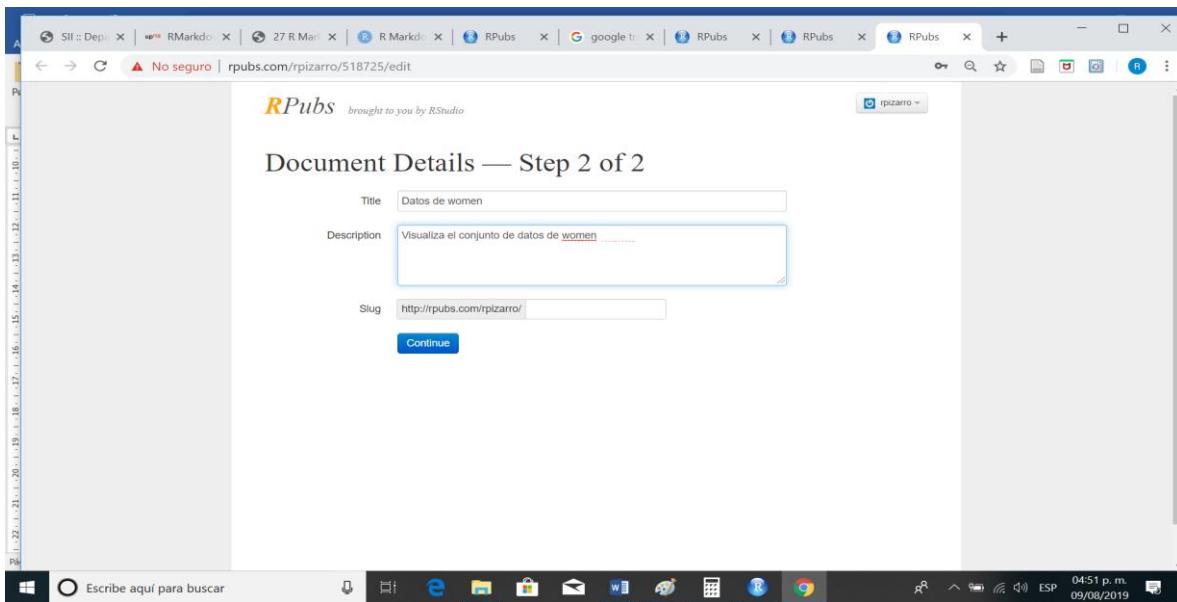
Publicar



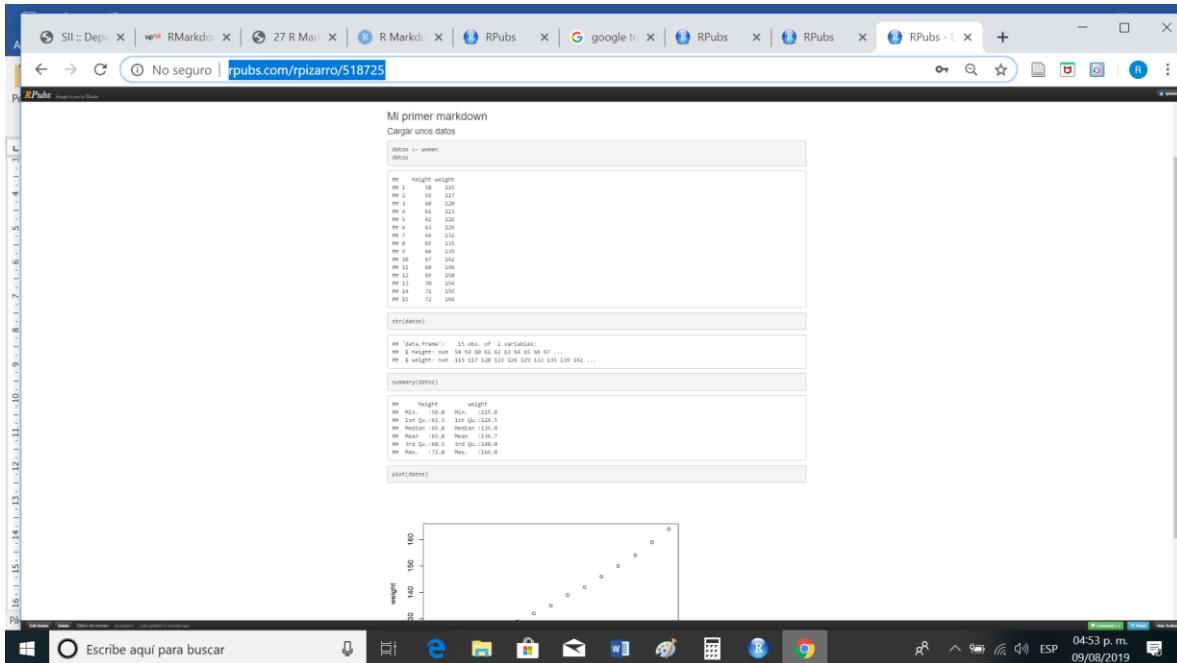
Proporcionar las credenciales de autenticación o registrarse como usuario y luego proporcionar cuenta y contraseña.



Continuar con el siguiente paso que es el título y una descripción del programa, implica como se desea que todas las personas del mundo conozcan el trabajo. Cabe recordar que esto es publicar en la nube o sea que cualquier persona de cualquier parte del mundo lo puede ver.



Presionar continuar y visualizar en la dirección correspondiente del usuario. Para este ejemplo el programa se conoce en la dirección: <http://rpubs.com/rpizarro/518725> y se visualiza de la siguiente manera:



Para finalizar solo falta indicar para que sirven las opciones str() y summary. Como se puede percibir str() indica la estructura de los datos, es decir hay 15 observaciones y 2 variables del tipo numéricas.

El comando summary() determina los valores estadísticos más comunes, la media, mediana, máximos y mínimos y cuartiles que se describirán más adelante en los temas de estadísticos.

Para mayor información de archivo markdown se sugiere el siguiente enlace
<https://rpubs.com/about/getting-started>

Parar una introducción a conceptos de R se sugiere el siguiente enlace:
<https://bookdown.org/boscomendoza/r-principiantes4/datos-mas-comunes.html>

2.4 Datos, población y muestra aleatoria

Datos son hechos/informaciones y cifras que se recogen, analizan y resumen para su presentación e interpretación. A todos los datos reunidos para un determinado estudio se les llama conjunto de datos para el estudio.

Una variable es una característica de los elementos que es de interés del conjunto de datos.

Los datos cualitativos comprenden etiquetas o nombres que se usan para identificar un atributo de cada elemento. Los datos cualitativos emplean la escala nominal o la ordinal y pueden ser numéricos o no. Los datos cuantitativos requieren valores numéricos que indiquen cuánto o cuántos. Los datos cuantitativos se obtienen usando las escalas de medición de intervalo o de razón.

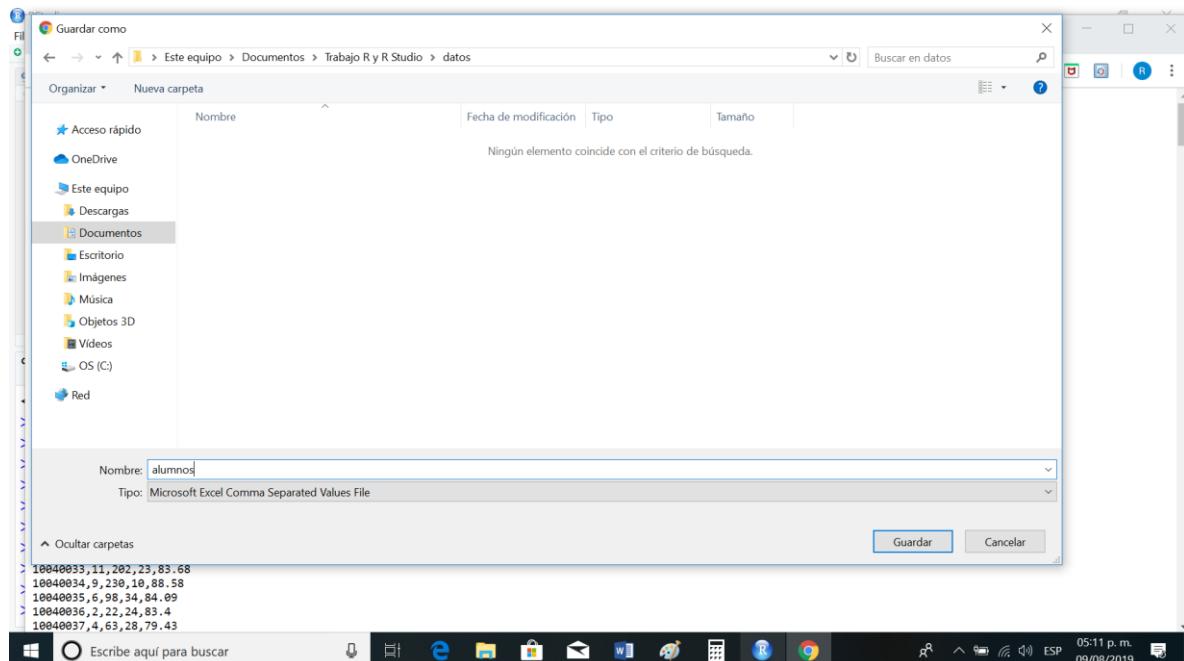
La población es el conjunto de todos los elementos de interés en un estudio determinado.

La muestra es un subconjunto de la población. (Anderson, Sweeney, & Williams, 2008)

Para entender estos conceptos se utiliza el siguiente ejemplo:

Dirigirse al siguiente enlace y descargar los datos en la carpeta de /Documentos/datos/... que fue creada anteriormente:

<https://raw.githubusercontent.com/rpizarroq/probabilidad-y-estadistica/master/practicas%20R/datos/Promedio%20alumnos%20de%202010%20y%2020o%20semestre%20ISC%20del%20ITD/alumnos%20inscritos%20de%2020de%20IS%C20SIN%20nocontrols.csv>



Ponerle al archivo de nombre 'alumnos'

Ahora es necesario importar los datos y se puede cargar los datos locales o ir directamente a la dirección URL de donde provienen.

Pulsar Import Data Set/From Text. Aquí en caso de que no esté instalado el paquete se deberá hacerlo en este momento, puede ser automático dejando que solo lo instale R Studio o puede hacerse mediante:

```
install.packages("readr")
library(readr)
```

Ejecutar para luego cargar los datos mediante este script:

```
# Poblacion y muestra
#install.packages("readr")
library(readr)
datos <- read.csv("C:/Users/Usuario/Documents/Trabajo R y R
Studio/datos/alumnos.csv")
View(datos)

str(datos)

filas.muestra <- sample(1:nrow(datos), 10, replace = FALSE)
filas.muestra
muestra <- datos[filas.muestra,] # Todos las columnas

View(muestra)
```

El script markdown en rpubs que hace lo anterior describe lo anterior se encuentra en <http://rpubs.com/rpizarro/518730>.

Solo falta indicar que la población son datos de alumnos que estuvieron inscritos en Instituto Tecnológico de Durango (ITD) en segundo semestre en un periodo determinado, se saca una muestra de 10 alumnos.

Los datos de número de control han sido modificados y no son reales, aunque existieran esos identificadores de números de control, se informa nuevamente que no son datos reales ni personales de un estudiante del ITD.

El comando sample() determina la muestra, para documentarse acerca de sample(), proporcionar: `? sample()` en la consola de R Studio.

3 Generar datos agrupados, no agrupados, y aspectos de frecuencia

El objetivo de esta práctica es mediante un conjunto de datos determinar agrupaciones y frecuencias.

3.1 Conceptos

- Una distribución de frecuencia es un resumen tabular de datos que muestra el número (frecuencia) de elementos en cada una de las diferentes clases disyuntas (que no se sobreponen).
- Frecuencia ó La frecuencia absoluta es el número de veces que aparece un determinado valor en un estudio estadístico El número de repeticiones de un valor dentro de una muestra o población. Se cuenta el número de veces que aparece.
- La suma de las frecuencias absolutas es igual al número total de datos o sea n
- Frecuencia relativa. La relación de la frecuencia con respecto al número de elementos n . Es el cociente entre la frecuencia absoluta de un determinado valor y el número total de datos. La suma de la Frecuencia relativa es 1.
- Los datos agrupados y no agrupados se les llaman en estadística a la manera de representar y analizar la información que has reunido o que dispones.
- Datos no agrupados es el conjunto de observaciones que se presentan en su forma original tal y como fueron recolectados, para obtener información directamente de ellos. Los datos no agrupados es un conjunto de información si ningún orden que no nos establece relación clara con lo que se pretende desarrollar a lo largo de un problema, esto se soluciona mediante una tabulación que nos conduce a una tabla de frecuencias
- Datos agrupados son datos que se dan en intervalos de clase, en un rango, como cuando se resumen para una distribución de frecuencias. No se tienen los valores de los datos originales.

- Parámetro poblacional Valor numérico que resume una población (por ejemplo, la media poblacional μ , la varianza poblacional, σ^2 y la desviación estándar poblacional, σ). (Anderson, Sweeney, & Williams, 2008)

3.2 Frecuencias, frecuencias relativas, histograma, gráfica de frecuencia

Para comprender y usar estos conceptos se utiliza el conjunto de datos.

Coke Classic	Sprite	Pepsi
Diet Coke	Coke Classic	Coke Classic
Pepsi	Diet Coke	Coke Classic
Diet Coke	Coke Classic	Coke Classic
Coke Classic	Diet Coke	Pepsi
Coke Classic	Coke Classic	Dr. Pepper
Dr. Pepper	Sprite	Coke Classic
Diet Coke	Pepsi	Diet Coke
Pepsi	Coke Classic	Pepsi
Pepsi	Coke Classic	Pepsi
Coke Classic	Coke Classic	Pepsi
Dr. Pepper	Pepsi	Pepsi
Sprite	Coke Classic	Coke Classic
Coke Classic	Sprite	Dr. Pepper
Diet Coke	Dr. Pepper	Pepsi
Coke Classic	Pepsi	Sprite
Coke Classic	Diet Coke	

Los datos de 'refrescos.csv' representa una compra de 50 refrescos por 50 personas.

3.3 Programa en R. Frecuencias Caso Refrescos

3.3.1 Los datos

```
refrescos <- c('Coke Classic','Sprite','Pepsi',
'Diet Coke','Coke Classic','Coke Classic',
'Pepsi','Diet Coke','Coke Classic',
'Diet Coke','Coke Classic','Coke Classic',
'Coke Classic','Diet Coke','Pepsi',
'Coke Classic','Coke Classic','Dr. Pepper',
'Dr. Pepper','Sprite','Coke Classic',
'Diet Coke','Pepsi','Diet Coke',
'Pepsi','Coke Classic','Pepsi',
'Pepsi','Coke Classic','Pepsi',
'Coke Classic','Coke Classic','Pepsi',
'Dr. Pepper','Pepsi','Pepsi',
'Sprite','Coke Classic','Coke Classic',
'Coke Classic','Sprite','Dr. Pepper',
'Diet Coke','Dr. Pepper','Pepsi',
'Coke Classic','Pepsi', 'Sprite',
```

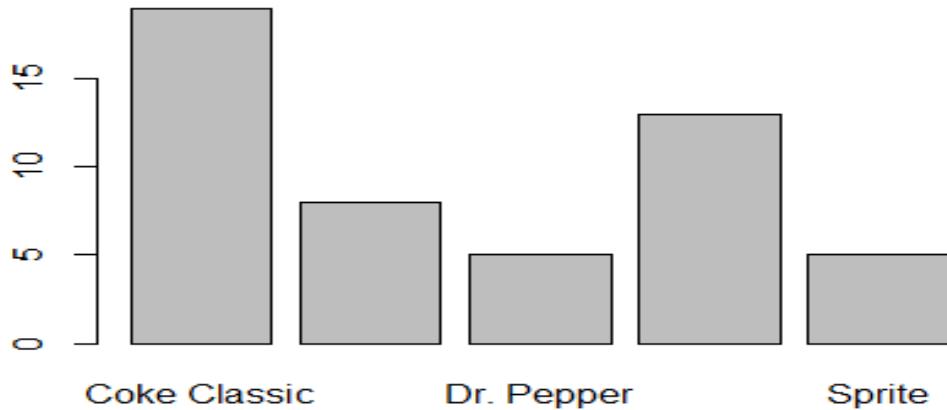
```
    'Coke Classic', 'Diet Coke'  
)  
  
refrescos <- factor(refrescos)
```

3.3.2 La frecuencia absoluta

```
frecuencias <- table(refrescos)  
frecuencias  
  
## refrescos  
## Coke Classic      Diet Coke     Dr. Pepper      Pepsi      Sprite  
##          19            8             5            13            5
```

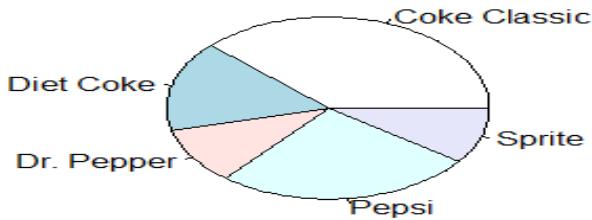
3.3.3 Visualización de datos. Barras

```
barplot(frecuencias)
```



3.3.4 Visualización de datos. Pastel

```
pie(frecuencias)
```



El código en markdown se encuentra en: <http://rpubs.com/rpizarro/518749>

3.4 Programa en R. Frecuencias Caso Simular alumnos

Este script simula la generación de 5000 observaciones que equivale a una población de 5000 alumnos de una Institución educativa de nivel superior.

Las variables que se manejan de cada alumno son las siguientes:

- Una matrícula o identificador, el cual es un valor consecutivo.
- Carrera que puede ser de un conjunto de 12 carreas de nivel superior
- Genero Masculino M o Femenino F
- Promedio escolar cuyo rango está entre 72 y 99. Una media de 85 bajo una distribución normal
- Edad de entre 17 y 24 años, la media 20 años bajo una distribución normal
- Peso. Complejión en Kg de cada alumno Hombres de 60 Kg la media Mujeres 55 Kg la media bajo una distribución normal
- Altura Hombres 1.75 la media y mujeres 1.65 la media.

3.4.1 Los datos iniciales

```
carreras <- c("ARQUITECTURA", "ADMINISTRACION", "BIOQUIMICA",
            "CIVIL", "ELECTRICA", "ELECTRONICA",
            "INDUSTRIAL", "INFORMATICA", "MECANICA",
            "MECATRONICA", "SISTEMAS", "TIC", "QUIMICA")
generos <- c('M', 'F')
```

3.4.2 Funciones generadoras de datos

```
# semilla inicial
set.seed(1)

getMatriculas <- function(n) {
  c(1:n)
}

getCarreras <- function(n, semilla) {
  set.seed(semilla)
  sample(carreras, n, replace = TRUE)
}

getGeneros <- function(n, semilla) {
  set.seed(semilla)
  sample(generos, n, replace = TRUE)
}

getEdades <- function(n, semilla) {
  set.seed(semilla)
  round(rnorm(n, 21, 1))
}

getPromedios <- function(n, semilla) {
  set.seed(semilla)
  round(rnorm(n, 86, sqrt(12)), 2)
}

getPesos <- function(generos, semilla) {
  set.seed(semilla)
  pesos <- NULL
  for (gen in generos) {
    if (gen == 'M') {
      peso <- round(rnorm(1, 80, sqrt(10)), 2) # Hombre
    }
    else {
      peso <- round(rnorm(1, 60, sqrt(5)), 2) # Mujer
    }
    pesos <- rbind(pesos, peso)
  }
  return (pesos)
}
```

```

getAlturas <- function(generos, semilla) {
  set.seed(semilla)
  alturas <- NULL
  for (gen in generos ) {
    if (gen == 'M') {
      altura <- round(rnorm(1, 180, sqrt(10)),2) # Hombre
    }
    else {
      altura <- round(rnorm(1, 160, sqrt(5)),2) # Mujer
    }
    alturas <- rbind(alturas, altura)
  }
  return (alturas)
}

genAlumnos <- function(n, semilla) {
  alumnos <- data.frame(matricula = getMatriculas(n))
  alumnos <- cbind(alumnos, carrera = getCarreras(n,semilla))
  alumnos <- cbind(alumnos, genero = getGeneros(n,semilla))
  alumnos <- cbind(alumnos, promedio = getPromedios(n,semilla))
  alumnos <- cbind(alumnos, edad = getEdades(n,semilla))
  alumnos <- cbind(alumnos, peso = getPesos(getGeneros(n,semilla),semilla
)))
  alumnos <- cbind(alumnos, altura = getAlturas(getGeneros(n,semilla),semilla))

  rownames(alumnos) <- c(1:n)
  return(alumnos)
}

```

3.4.3 Generando la población

```

alumnos <- genAlumnos(5000, 1)
head(alumnos) # Primeras 6 filas

##   matricula      carrera genero promedio edad   peso altura
## 1          1        CIVIL     M    83.83  20 74.29 174.29
## 2          2    ELECTRICA     M    86.64  21 77.86 177.86
## 3          3 INFORMATICA     F    83.11  20 58.94 158.94
## 4          4         TIC     F    91.53  23 62.30 162.30
## 5          5  BIOQUIMICA     M    87.14  21 78.11 178.11
## 6          6         TIC     F    83.16  20 62.59 162.59

tail(alumnos) # Ultimas 6 filas

##      matricula      carrera genero promedio edad   peso altura
## 4995    4995    INFORMATICA     F    81.15  20 58.57 158.57
## 4996    4996    ELECTRONICA     M    86.56  21 79.00 179.00
## 4997    4997 ADMINISTRACION     M    89.40  22 82.37 182.37
## 4998    4998    ELECTRONICA     M    83.60  20 81.20 181.20

```

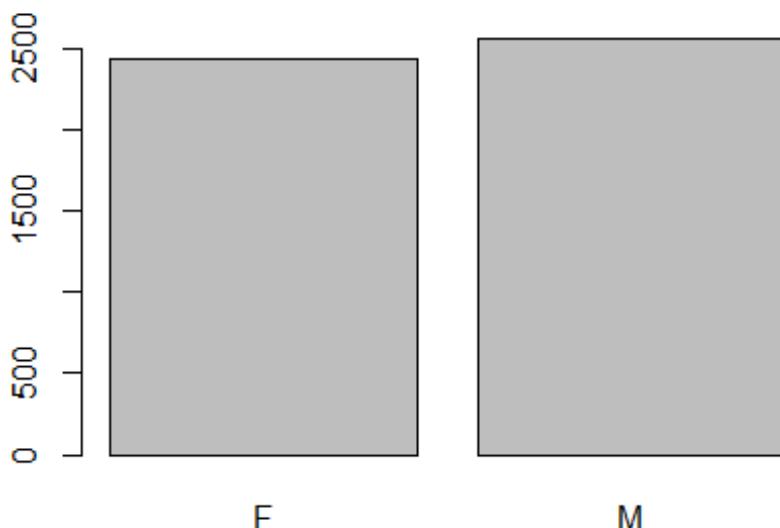
## 4999	4999	ADMINISTRACION	M	85.99	21	81.25	181.25
## 5000	5000	CIVIL	M	86.59	21	83.42	183.42

3.4.4 Tabla de frecuencias de Genero

```
frecuencias <- table(alumnos$genero)
frecuencias

## 
##      F      M
## 2436 2564

barplot(frecuencias)
```



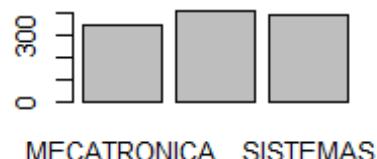
3.4.5 Tabla de frecuencias de Carrera

```
frecuencias <- table(alumnos$carrera)
frecuencias

## 
## ADMINISTRACION      ARQUITECTURA      BIOQUIMICA      CIVIL      ELEC
## TRICA                390                 421                 374                 378
## 407
## ELECTRONICA      INDUSTRIAL      INFORMATICA      MECANICA      MECATR
## ONICA                403                 379                 354                 336
## 345
```

```
##          QUIMICA      SISTEMAS        TIC
##          415            397           401
```

```
par(mfrow=c(2,2))
barplot(frecuencias[1:3])
barplot(frecuencias[4:6])
barplot(frecuencias[7:9])
barplot(frecuencias[10:12])
```



MECATRONICA SISTEMAS

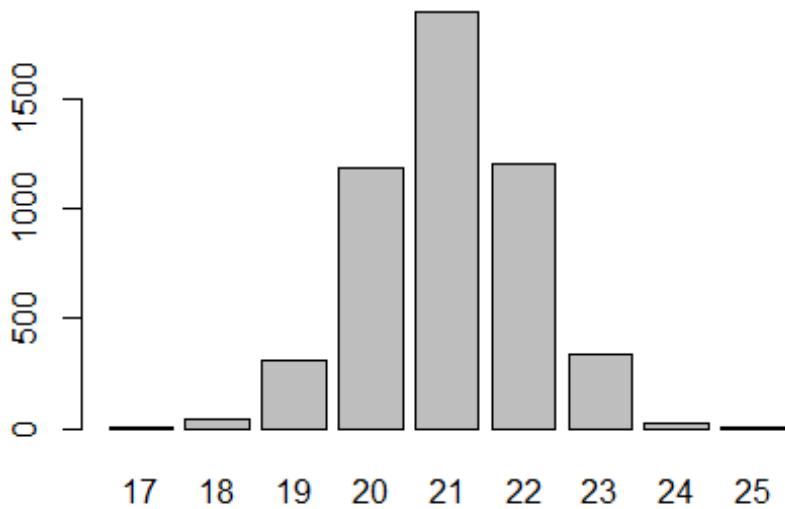
```
par(mfrow=c(1,1))
```

3.4.6 Tabla de frecuencias de Edades

```
frecuencias <- table(alumnos$edad)
frecuencias

##
##   17   18   19   20   21   22   23   24   25
##   2    44   310  1187  1891  1204  339   21    2
```

```
par(mfrow=c(1,1))
barplot(frecuencias)
```



3.4.7 Frecuencias relativa y acumuladas de Edades

```
frecuencias <- data.frame(table(alumnos$edad))
frecuencias

##   Var1 Freq
## 1 17    2
## 2 18   44
## 3 19  310
## 4 20 1187
## 5 21 1891
## 6 22 1204
## 7 23  339
## 8 24   21
## 9 25    2

frecuencias <- cbind(frecuencias, relativa = round(frecuencias$Freq / nro
w(alumnos),2))

frecuencias <- cbind(frecuencias, porcentual = round(frecuencias$Freq / n
row(alumnos) * 100,0))

frecuencias

##   Var1 Freq  relativa  porcentual
## 1 17    2      0.00        0
## 2 18   44      0.01        1
```

```

## 3   19   310    0.06      6
## 4   20  1187    0.24     24
## 5   21  1891    0.38     38
## 6   22  1204    0.24     24
## 7   23   339    0.07      7
## 8   24    21    0.00      0
## 9   25     2    0.00      0

```

3.4.8 Puntos medios, máximos y mínimos de edad con summary()

```
summary(alumnos$edad)
```

```

##   Min. 1st Qu. Median   Mean 3rd Qu. Max.
##     17      20      21      21      22      25

```

3.4.9 Agrupación de alumnos por edades

- Se utiliza la librería library(fdth)
- Se genera una distribución donde>
- f= frecuencia absoluta
- rf= frecuencia relativa
- rf(%) frecuencia relativa porcentual
- cf= frecuencia acumulada
- cf(%)=frecuencia acumulada porcentual

```

library(fdth)

## 
## Attaching package: 'fdth'

## The following objects are masked from 'package:stats':
## 
##     sd, var

distribucion <- fdt(alumnos$edad, breaks="Sturges") # Regla Sturge
distribucion

##   Class limits     f   rf rf(%)   cf   cf(%)
##   [16.8,17.4)     2 0.00  0.04     2  0.04
##   [17.4,18)      44 0.01  0.88    46  0.92
##   [18,18.6)       0 0.00  0.00    46  0.92
##   [18.6,19.2)    310 0.06  6.20   356  7.12
##   [19.2,19.8)     0 0.00  0.00   356  7.12
##   [19.8,20.4)   1187 0.24 23.74  1543 30.86
##   [20.4,21)      1891 0.38 37.82  3434 68.68
##   [21,21.6)       0 0.00  0.00  3434 68.68
##   [21.6,22.2)    1204 0.24 24.08  4638 92.76
##   [22.2,22.8)     0 0.00  0.00  4638 92.76

```

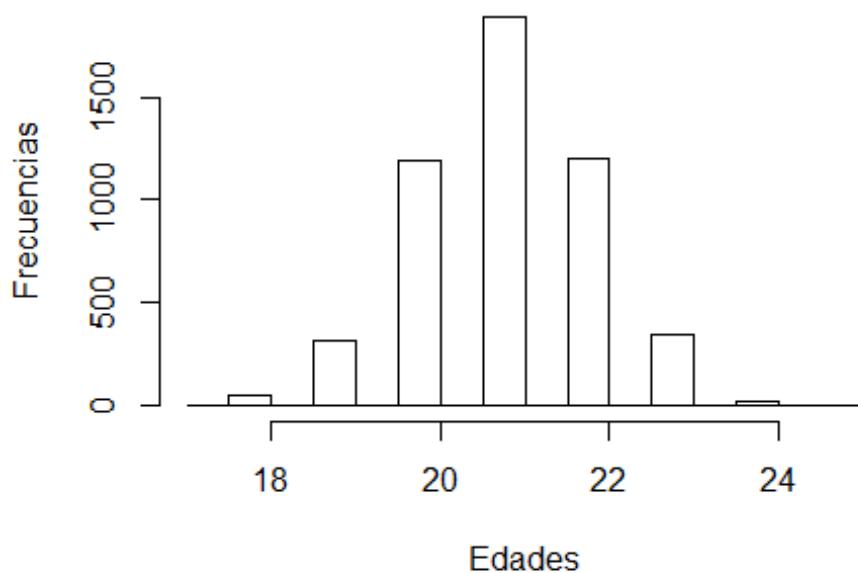
```

## [22.8,23.4) 339 0.07 6.78 4977 99.54
## [23.4,24)   21 0.00 0.42 4998 99.96
## [24,24.6)    0 0.00 0.00 4998 99.96
## [24.6,25.2)   2 0.00 0.04 5000 100.00

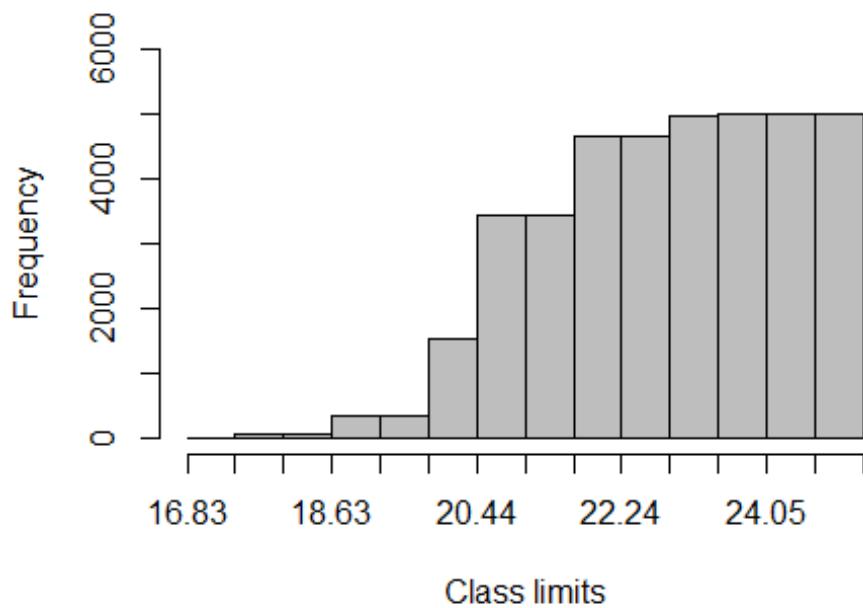
hist(alumnos$edad, breaks = "Sturges",
      main = "Histograma de edades",
      xlab = "Edades", ylab = "Frecuencias") #histograma utilizando el numero de clases según Sturge

```

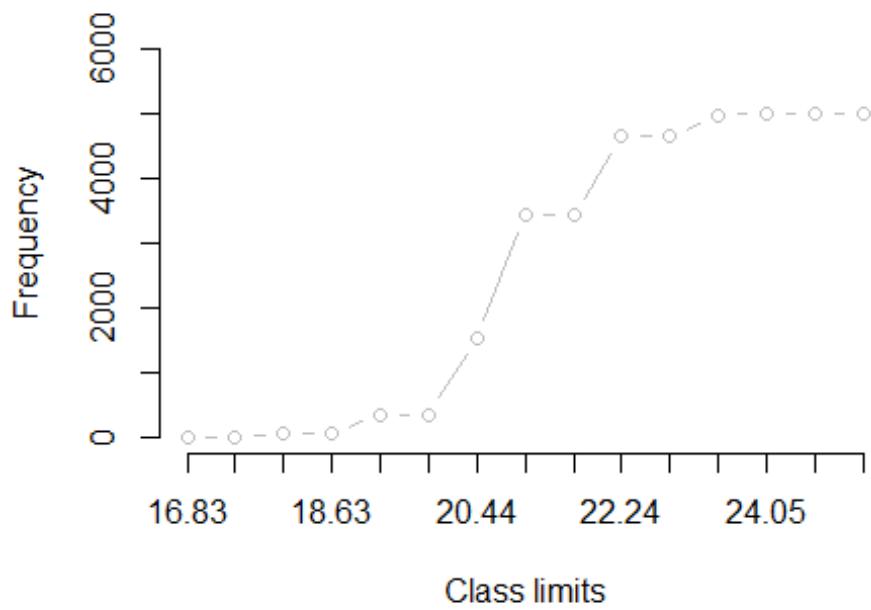
Histograma de edades



```
plot(distribucion, type="cfh") #histograma de frecuencias acumulada
```



```
plot(distribucion, type="cfp") #poligono de frecuencias acumulado
```



3.4.10 Diagrama de tallo y hoja con edades

```
## 24 |
## 25 | +2
```

El script markdown que hace lo anterior, se encuentra en el servicio rpubs que está en el siguiente enlace: <http://rpubs.com/rpizarro/518806>

4 Generar operaciones y valores usando comandos en R para determinar medidas de tendencia central

El objetivo de esta práctica es determinar las medidas de tendencia utilizando R como lo son la media, mediana y moda de un conjunto de datos.

Para realizar esta práctica, se utilizan los mismos datos que la práctica del punto 3.

4.1 Conceptos

Se describen los estadísticos de media, mediana y moda.

Media

La medida de localización más importante es la media, o valor promedio, de una variable. La media proporciona una medida de localización central de los datos. Si los datos son datos de una muestra, la media se denota x barra; si los datos son datos de una población, la media se denota con la letra griega μ .

Mediana

La mediana es otra medida de localización central. Es el valor de en medio en los datos ordenados de menor a mayor (en forma ascendente). Cuando tiene un número impar de observaciones, la mediana es el valor de en medio. Cuando la cantidad de observaciones es par, no hay un número en medio. En este caso, se sigue una convención y la mediana es definida como el promedio de las dos observaciones de en medio. Por conveniencia, la definición de mediana se replantea así:

- Ordenar los datos de menor a mayor (en forma ascendente).
- a. Si el número de observaciones es impar, la mediana es el valor de en medio.
- Si el número de observaciones es par, la mediana es el promedio de las dos observaciones de en medio.

Moda

La moda es el valor que se presenta con mayor frecuencia. (Anderson, Sweeney, & Williams, 2008)

4.2 Caso de media, mediana y moda de alumnos

Este script determina medidas de tendencia central, media, mediana y moda de algunas variables cuantitativas de alumnos. Se utiliza el conjunto de datos que son importados desde el servicio github.

El conjunto de alumnos se encuentra en la dirección:
<https://raw.githubusercontent.com/rpizarro/probabilidad-y-estadistica/master/practicas%20R/datos/alumnos%20simulados.csv>

4.2.1 Importar los datos

```
alumnos <- read.csv("https://raw.githubusercontent.com/rpizarro/probabilidad-y-estadistica/master/practicas%20R/datos/alumnos%20simulados.csv")
head(alumnos)

##   X matricula      carrera genero promedio edad peso altura
## 1 1          1        CIVIL     M    83.83  20 74.29 174.29
## 2 2          2    ELECTRICA     M    86.64  21 77.86 177.86
## 3 3          3 INFORMATICA     F    83.11  20 58.94 158.94
## 4 4          4         TIC     F    91.53  23 62.30 162.30
## 5 5          5 BIOQUIMICA     M    87.14  21 78.11 178.11
## 6 6          6         TIC     F    83.16  20 62.59 162.59

tail(alumnos)

##       X matricula      carrera genero promedio edad peso altura
## 4995 4995      4995    INFORMATICA     F    81.15  20 58.57 158.57
## 4996 4996      4996    ELECTRONICA     M    86.56  21 79.00 179.00
## 4997 4997      4997 ADMINISTRACION     M    89.40  22 82.37 182.37
## 4998 4998      4998    ELECTRONICA     M    83.60  20 81.20 181.20
## 4999 4999      4999 ADMINISTRACION     M    85.99  21 81.25 181.25
## 5000 5000      5000        CIVIL     M    86.59  21 83.42 183.42
```

4.2.2 Factorizando genero y carreras de alumnos

```
alumnos$genero <- factor(alumnos$genero)
alumnos$carrera <- factor(alumnos$carrera)
```

4.2.3 Estructura de alumnos

```
str(alumnos)

## 'data.frame': 5000 obs. of  8 variables:
## $ X      : int  1 2 3 4 5 6 7 8 9 10 ...
## $ matricula: int  1 2 3 4 5 6 7 8 9 10 ...
## $ carrera : Factor w/ 13 levels "ADMINISTRACION",...: 4 5 8 13 3 13 1
## $ genero  : Factor w/ 2 levels "F","M": 2 2 1 1 2 1 1 1 1 2 ...
## $ promedio : num  83.8 86.6 83.1 91.5 87.1 ...
```

```
## $ edad      : int  20 21 20 23 21 20 21 22 22 21 ...
## $ peso       : num  74.3 77.9 58.9 62.3 78.1 ...
## $ altura     : num  174 178 159 162 178 ...
```

4.2.4 Estadísticos de genero con summary()

```
summary(alumnos$genero)
```

```
##      F      M
## 2436 2564
```

4.2.5 Estadísticos de carrera con summary()

```
summary(alumnos$carrera)
```

	ADMINISTRACION	ARQUITECTURA	BIOQUIMICA	CIVIL	ELECTRICA
##	390	421	374	378	
407					
##	ELECTRONICA	INDUSTRIAL	INFORMATICA	MECANICA	MECATRO
NICA	403	379	354	336	
##					
345					
##	QUIMICA	SISTEMAS	TIC		
##	415	397	401		

4.2.6 Estadísticos de edades con summary()

```
summary(alumnos$edad)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	17	20	21	21	22	25

4.2.7 Estadísticos de promedios con summary()

```
summary(alumnos$promedio)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	73.28	83.68	85.95	85.99	88.42	99.20

4.2.8 Estadísticos de pesos con summary()

```
summary(alumnos$peso)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	52.77	60.05	73.69	70.24	80.10	91.46

4.2.9 Estadísticos de alturas con summary()

```
summary(alumnos$altura)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	152.8	160.1	173.7	170.2	180.1	191.5

4.2.10 Media, Mediana y Moda de promedios de todos

```
media <- mean(alumnos$promedio)
paste("La media de la variable edad es", media)

## [1] "La media de la variable edad es 85.98896"

mediana <- median(alumnos$promedio)
paste("La mediana de la variable edad es", mediana)

## [1] "La mediana de la variable edad es 85.95"

frecuencias <- data.frame(table(alumnos$promedio))
moda <- frecuencias[which.max(frecuencias$Freq),1]

paste("La moda de la variable edad es", moda)

## [1] "La moda de la variable edad es 83.73"
```

4.2.11 Promedio de alumnos de dos carreras

```
# install.packages("dplyr") # Instalarlo antes si se necesita no haberlo
hecho
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

sistemas <- filter(alumnos, carrera == "SISTEMAS")
informatica <- filter(alumnos, carrera == "INFORMATICA")
arquitectura <- filter(alumnos, carrera == "ARQUITECTURA")

mean(sistemas$promedio)

## [1] 85.98738

mean(informatica$promedio)

## [1] 85.9652

mean(arquitectura$promedio)

## [1] 85.9043
```

```
if (mean(sistemas$promedio) > mean(informatica$promedio)) {  
  print("Los alumnos de sistemas tienen mejor promedio académico")  
}  
  
## [1] "Los alumnos de sistemas tienen mejor promedio académico"  
  
if (mean(sistemas$promedio) < mean(informatica$promedio)) {  
  print("Los alumnos de informatica tienen mejor promedio académico")  
}  
if (mean(sistemas$promedio) == mean(informatica$promedio)) {  
  print("Los alumnos de informatica tienen promedio académico similar a los alumnos de sistemas")  
}
```

La práctica en markdown se encuentra en el siguiente enlace

<http://rpubs.com/rpizarro/518837>

5 Generar operaciones y valores usando comandos para determinar medidas de dispersión.

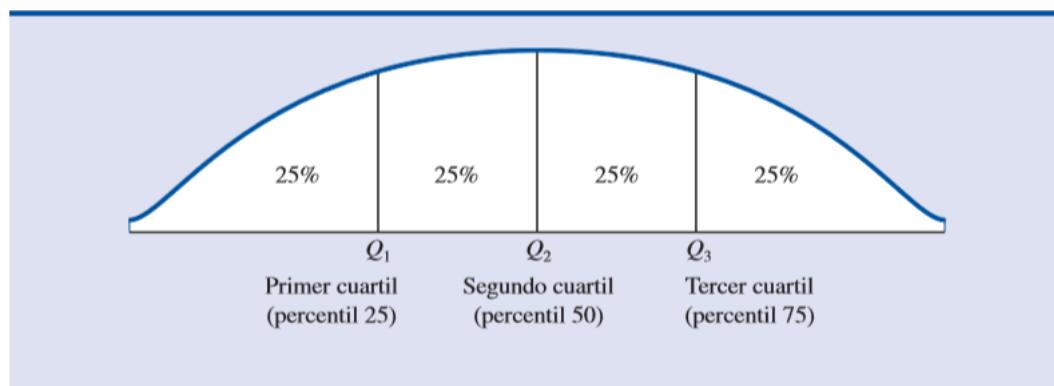
El objetivo de esta práctica es determinar las medidas de tendencia utilizando R como lo son la desviación estándar y la varianza de un conjunto de datos de alumnos

Para realizar esta práctica, se utilizan los mismos datos que la práctica del punto 3 y 4.

5.1 Conceptos

Un percentil aporta información acerca de la dispersión de los datos en el intervalo que va del menor al mayor valor de los datos. En los conjuntos de datos que no tienen muchos valores repetidos, el percentil p divide a los datos en dos partes. Cerca de p por ciento de las observaciones tienen valores menores que el percentil p y aproximadamente $(100 - p)$ por ciento de las observaciones tienen valores mayores que el percentil p . El percentil p se define como sigue: El percentil p es un valor tal que por lo menos p por ciento de las observaciones son menores o iguales que este valor y por lo menos $(100 - p)$ por ciento de las observaciones son mayores o iguales que este valor.

Los cuartiles sólo son percentiles determinados; así que los pasos para calcular los percentiles también se emplean para calcular los cuartiles. Con frecuencia es conveniente dividir los datos en cuatro partes; así, cada parte contiene una cuarta parte o 25% de las observaciones.



Rango. La medida de variabilidad más sencilla es el rango. Resulta de la diferencia entre el valor máximo menos el valor mínimo del valor de una variable de las observaciones.

Una medida que no es afectada por los valores extremos es el rango intercuartílico (RIC). Esta medida de variabilidad es la diferencia entre el tercer cuartil Q3 y el primer cuartil Q1.

La varianza es una medida de variabilidad que utiliza todos los datos. La varianza está basada en la diferencia entre el valor de cada observación (x_i) y la media.

La desviación estándar se define como la raíz cuadrada positiva de la varianza. La desviación estándar se mide en las mismas unidades que los datos originales. Por esta razón es más fácil comparar la desviación estándar con la media y con otros estadísticos que se miden en las mismas unidades que los datos originales.

En algunas ocasiones se requiere un estadístico descriptivo que indique cuán grande es la desviación estándar en relación con la media. Esta medida es el coeficiente de variación y se representa como porcentaje. En general, el coeficiente de variación es un estadístico útil para comparar la variabilidad de variables que tienen desviaciones estándar distintas y medias distintas (Anderson, Sweeney, & Williams, 2008)

La siguiente imagen determina el coeficiente de variación

COEFICIENTE DE VARIACIÓN

$$\left(\frac{\text{Desviación estándar}}{\text{Media}} \times 100 \right) \%$$

En términos porcentuales es la relación que hay entre la desviación estándar y la media.

5.2 Medidas de dispersión

Este script determina medidas de dispersión, rango, cuartiles, percentiles, varianza, desviación estándar y coeficiente de variación. Se utiliza el conjunto de datos que son importados desde el servicio github.

El conjunto de alumnos se encuentra en la dirección:

<https://raw.githubusercontent.com/rpizarro/probabilidad-y-estadistica/master/practicas%20R/datos/alumnos%20simulados.csv>

5.2.1 Importar los datos

```
alumnos <- read.csv("https://raw.githubusercontent.com/rpizarro/probabilidad-y-estadistica/master/practicas%20R/datos/alumnos%20simulados.csv")
head(alumnos)

##   X matricula      carrera genero promedio edad  peso altura
## 1 1           1       CIVIL     M    83.83  20 74.29 174.29
## 2 2           2   ELECTRICA     M    86.64  21 77.86 177.86
## 3 3           3 INFORMATICA     F    83.11  20 58.94 158.94
## 4 4           4        TIC     F    91.53  23 62.30 162.30
## 5 5           5 BIOQUIMICA     M    87.14  21 78.11 178.11
## 6 6           6        TIC     F    83.16  20 62.59 162.59

tail(alumnos)

##          X matricula      carrera genero promedio edad  peso altura
## 4995 4995     4995 INFORMATICA     F    81.15  20 58.57 158.57
## 4996 4996     4996 ELECTRONICA     M    86.56  21 79.00 179.00
## 4997 4997     4997 ADMINISTRACION     M    89.40  22 82.37 182.37
## 4998 4998     4998 ELECTRONICA     M    83.60  20 81.20 181.20
## 4999 4999     4999 ADMINISTRACION     M    85.99  21 81.25 181.25
## 5000 5000     5000       CIVIL     M    86.59  21 83.42 183.42

# aLumnos
```

5.2.2 Factorizando genero y carreras de alumnos

```
alumnos$genero <- factor(alumnos$genero)
alumnos$carrera <- factor(alumnos$carrera)
```

5.2.3 Estructura de alumnos

```
str(alumnos)

## 'data.frame':  5000 obs. of  8 variables:
##   $ X         : int  1 2 3 4 5 6 7 8 9 10 ...
##   $ matricula: int  1 2 3 4 5 6 7 8 9 10 ...
##   $ carrera   : Factor w/ 13 levels "ADMINISTRACION",...: 4 5 8 13 3 13 1
##   $ genero    : Factor w/ 2 levels "F","M": 2 2 1 1 2 1 1 1 2 ...
##   $ promedio   : num  83.8 86.6 83.1 91.5 87.1 ...
##   $ edad       : int  20 21 20 23 21 20 21 22 22 21 ...
##   $ peso       : num  74.3 77.9 58.9 62.3 78.1 ...
##   $ altura     : num  174 178 159 162 178 ...
```

5.2.4 Resumen de los alumnos con summary()

```
summary(alumnos)

##      X      matricula      carrera      genero
## Min.   : 1   Min.   : 1   ARQUITECTURA: 421   F:2436
## 1st Qu.:1251 1st Qu.:1251  QUIMICA      : 415   M:2564
## Median :2500 Median :2500  ELECTRICA    : 407
## Mean   :2500 Mean   :2500 ELECTRONICA : 403
## 3rd Qu.:3750 3rd Qu.:3750    TIC         : 401
## Max.   :5000 Max.   :5000 SISTEMAS    : 397
##                   (Other)     :2556
##      promedio      edad      peso      altura
## Min.   :73.28   Min.   :17   Min.   :52.77   Min.   :152.8
## 1st Qu.:83.68  1st Qu.:20  1st Qu.:60.05  1st Qu.:160.1
## Median :85.95  Median :21  Median :73.69  Median :173.7
## Mean   :85.99  Mean   :21  Mean   :70.24  Mean   :170.2
## 3rd Qu.:88.42 3rd Qu.:22  3rd Qu.:80.10  3rd Qu.:180.1
## Max.   :99.20  Max.   :25  Max.   :91.46  Max.   :191.5
##
```

5.2.5 Determinando población Hombres y Mujeres y su media en Peso y Altura

```
# install.packages("dplyr") # Instalarlo antes si se necesita no haberlo
# hecho
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

hombres <- filter(alumnos, genero == 'M')
mujeres <- filter(alumnos, genero == 'F')

print(paste("Cantidad de Hombres: ", nrow(hombres)))
## [1] "Cantidad de Hombres: 2564"

print(paste("Cantidad de Mujeres: ", nrow(mujeres)))
## [1] "Cantidad de Mujeres: 2436"
```

```

print(paste("Media de Peso en Hombres: ",round(mean(hombres$peso),2)))
## [1] "Media de Peso en Hombres: 79.97"
print(paste("Media de Peso en Mujeres: ",round(mean(mujeres$peso),2)))
## [1] "Media de Peso en Mujeres: 60"
print(paste("Media de Altura en Hombres: ",round(mean(hombres$altura),2)))
## [1] "Media de Altura en Hombres: 179.97"
print(paste("Media de Altura en Mujeres: ",round(mean(mujeres$altura),2)))
## [1] "Media de Altura en Mujeres: 160"

```

5.2.6 Determinando cuartiles de Pesos en Hombre y Mujeres

- Se crean variables de cuartiles al 25%, 50% y 75% del peso hombres
- Se crean variables de cuartiles al 25%, 50% y 75% del peso de las mujeres
- Se visualiza e interpretan los cuartiles del peso de Hombres y Mujeres

```

q25PesoH <- quantile(hombres$peso, 0.25)
q50PesoH <- quantile(hombres$peso, 0.50)
q75PesoH <- quantile(hombres$peso, 0.75)

print(paste("El cuartile 25% del peso de los hombres, ", "con valor de ",
q25PesoH, " significa que a partir de ese valor, está el otro 75%"))
## [1] "El cuartile 25% del peso de los hombres, con valor de 77.9 significa que a partir de ese valor, está el otro 75%"

print(paste("El cuartile 50% del peso de los hombres, ", "con valor de ",
q50PesoH, " significa que a partir de ese valor, está el otro 50%"))
## [1] "El cuartile 50% del peso de los hombres, con valor de 79.99 significa que a partir de ese valor, está el otro 50%"

print(paste("El cuartile 75% del peso de los hombres, ", "con valor de ",
q75PesoH, " significa que a partir de ese valor, está el otro 25%"))
## [1] "El cuartile 75% del peso de los hombres, con valor de 81.99 significa que a partir de ese valor, está el otro 25%"

q25PesoM <- quantile(hombres$peso, 0.25)
q50PesoM <- quantile(hombres$peso, 0.50)
q75PesoM <- quantile(hombres$peso, 0.75)

```

```

print(paste("El cuartile 25% del peso de las mujeres, ", "con valor de ", q25PesoM, " significa que a partir de ese valor, está el otro 75%"))

## [1] "El cuartile 25% del peso de las mujeres, con valor de 77.9 significa que a partir de ese valor, está el otro 75%"

print(paste("El cuartile 50% del peso de las mujeres, ", "con valor de ", q50PesoM, " significa que a partir de ese valor, está el otro 50%"))

## [1] "El cuartile 50% del peso de las mujeres, con valor de 79.99 significa que a partir de ese valor, está el otro 50%"

print(paste("El cuartile 75% del peso de las mujeres, ", "con valor de ", q75PesoM, " significa que a partir de ese valor, está el otro 25%"))

## [1] "El cuartile 75% del peso de las mujeres, con valor de 81.99 significa que a partir de ese valor, está el otro 25%"

```

5.2.7 Determinando percentiles de alturas del hombres y mujeres al 10% y 90%

- Determinar percentiles al 10% y 90% de la altura de los hombres
- Determinar quantile al 10% y 90% de la altura de las mujeres
- Visualizar su valores

```

p10alturaH <- quantile(hombres$altura, 0.10)
p90alturaH <- quantile(hombres$altura, 0.90)

print(paste("El percentil 10% de la altura de los hombres, ", "con valor de ", p10alturaH, " significa que a partir de ese valor, está el otro 90%"))

## [1] "El percentil 10% de la altura de los hombres, con valor de 175.95 significa que a partir de ese valor, está el otro 90%"

print(paste("El percentil 90% de la altura de los hombres, ", "con valor de ", p90alturaH, " significa que a partir de ese valor, está el otro 10%"))

## [1] "El percentil 90% de la altura de los hombres, con valor de 184.03 significa que a partir de ese valor, está el otro 10%"

p10alturaM <- quantile(hombres$altura, 0.10)
p90alturaM <- quantile(hombres$altura, 0.90)

print(paste("El percentil 10% de la altura de las mujeres, ", "con valor de ", p10alturaM, " significa que a partir de ese valor, está el otro 90%"))

## [1] "El percentil 10% de la altura de las mujeres, con valor de 175.95 significa que a partir de ese valor, está el otro 90%"

```

```

print(paste("El percentil 90% de la altura de las mujeres, ", "con valor
de ", p90alturaM, " significa que a partir de ese valor, está el otro 10%
"))

## [1] "El percentil 90% de la altura de las mujeres, con valor de 184.
03 significa que a partir de ese valor, está el otro 10%"

```

5.2.8 Determinando las media de tres carreras

```

# install.packages("dplyr") # Instalarlo antes si se necesita no haberlo
hecho
library(dplyr)

quimica <- filter(alumnos, carrera == "QUIMICA")
bioquimica <- filter(alumnos, carrera == "BIOQUIMICA")
civil <- filter(alumnos, carrera == "CIVIL")

meanquimica <- round(mean(quimica$promedio),2)
meanbioquimica <- round(mean(bioquimica$promedio),2)
meancivil<- round(mean(civil$promedio), 2)

paste("La media del promedio de calificaciones de quimica es: ", meanquimica)
## [1] "La media del promedio de calificaciones de quimica es: 85.92"

paste("La media del promedio de calificaciones de bioquimica es: ", meanbioquimica)
## [1] "La media del promedio de calificaciones de bioquimica es: 86.13"

paste("La media del promedio de calificaciones de civil es: ", meancivil)
## [1] "La media del promedio de calificaciones de civil es: 85.81"

```

5.2.9 Determinando rango de promedio de calificaciones de tres carreras

- El rango en R se determina por la función range() que encuentra límite superior (LS) y límite inferior (LI)
- Genera un arreglo de 2 posiciones [1] y 2[]
- Para determinar el rango entonces encontrar la diferencia entre LS-LI
- Determinar el rango para tres carreras QUIMICA, BIQUIMICA Y CIVIL

```

limites <- range(quimica$promedio)
LI <- limites[1]
LS <- limites[2]

```

```

rango <- LS - LI
print(paste("El rango del promedio de calificaciones de la carrera de QUI
MICA es: ", rango))

## [1] "El rango del promedio de calificaciones de la carrera de QUIMICA
es: 20.16"

limites <- range(bioquimica$promedio)
LI <- limites[1]
LS <- limites[2]
rango <- LS - LI
print(paste("El rango del promedio de calificaciones de la carrera de BIO
QUIMICA es: ", rango))

## [1] "El rango del promedio de calificaciones de la carrera de BIOQUIMI
CA es: 19.63"

limites <- range(civil$promedio)
LI <- limites[1]
LS <- limites[2]
rango <- LS - LI
print(paste("El rango del promedio de calificaciones de la carrera de CIV
IL es: ", rango))

## [1] "El rango del promedio de calificaciones de la carrera de CIVIL es
: 22.85"

```

5.2.9.1 Determinando el rango intercuartílico del promedio de calificaciones de tres carreras

- Primero determinar los Q1 y Q3 de promedio de calificaciones de cada
carrera. BIOQUIMICA, QUIMICA, CIVIL
- Encontrar el rango intercuartílico
- Mostrar el dato

```

q1 <- quantile(bioquimica$promedio, 0.25)
q3 <- quantile(bioquimica$promedio, 0.75)
RIC <- round(q3 - q1, 2)
print(paste("El Rango Inter Cuartílico RIC del promedio de calificaciones
de la carrera de BIOQUIMICA es: ", RIC))

## [1] "El Rango Inter Cuartílico RIC del promedio de calificaciones de l
a carrera de BIOQUIMICA es: 4.8"

q1 <- quantile(quimica$promedio, 0.25)
q3 <- quantile(quimica$promedio, 0.75)
RIC <- round(q3 - q1, 2)
print(paste("El Rango Inter Cuartílico RIC del promedio de calificaciones
de la carrera de QUIMICA es: ", RIC))

```

```

## [1] "El Rango Inter Cuartílico RIC del promedio de calificaciones de 1
a carrera de QUIMICA es: 4.72"

q1 <- quantile(civil$promedio, 0.25)
q3 <- quantile(civil$promedio, 0.75)
RIC <- round(q3 - q1, 2)
print(paste("El Rango Inter Cuartílico RIC del promedio de calificaciones
de la carrera de CIVIL es: ", RIC))

## [1] "El Rango Inter Cuartílico RIC del promedio de calificaciones de 1
a carrera de CIVIL es: 5.04"

```

5.2.10 Determinando la varianza del promedio de calificaciones de tres carreas

```

varBioquimica <- round(var(bioquimica$promedio), 2)
print(paste("La varianza del promedio de calificaciones de la carrera de
BIOQUIMICA es: ", varBioquimica))

## [1] "La varianza del promedio de calificaciones de la carrera de BIOQU
IMICA es: 12.42"

varQuimica <- round(var(quimica$promedio),2)
print(paste("La varianza del promedio de calificaciones de la carrera de
QUIMICA es: ", varQuimica))

## [1] "La varianza del promedio de calificaciones de la carrera de QUIMI
CA es: 13.14"

varCivil <- round(var(civil$promedio),2)
varQuimica <- var(quimica$promedio)
print(paste("La varianza del promedio de calificaciones de la carrera de
CIVIL es: ", varCivil))

## [1] "La varianza del promedio de calificaciones de la carrera de CIVIL
es: 13.76"

```

5.2.11 Determinando las desviaciones estándar de tres carreras

```

dsBioquimica <- round(sd(bioquimica$promedio), 2)
print(paste("La Desviación Estándard del promedio de calificaciones de la
carrera de BIOQUIMICA es: ", dsBioquimica))

## [1] "La Desviación Estándard del promedio de calificaciones de la Carrera
de BIOQUIMICA es: 3.52"

dsQuimica <- round(sd(quimica$promedio), 2)
print(paste("La Desviación Estándard del promedio de calificaciones de la
carrera de QUIMICA es: ", dsQuimica))

## [1] "La Desviación Estándard del promedio de calificaciones de la Carrera
de QUIMICA es: 3.63"

```

```

dsCivil <- round(sd(civil$promedio), 2)
print(paste("La Desviación Estándar del promedio de calificaciones de la
carrera de CIVIL es: ", dsCivil))

## [1] "La Desviación Estándar del promedio de calificaciones de la carrera de CIVIL es: 3.71"

```

5.2.12 Determinando coeficiente de variación CV del promedio de calificaciones de tres carreras

- Se hace la relación en términos porcentuales de la desviación con respecto a la media de la variable promedio de calificaciones por cada carrera
- Se muestra el resultado

```

CVBioquimica <- round(sd(bioquimica$promedio) / mean(bioquimica$promedio)
* 100, 2)
print(paste("El coeficiente de variación CV del promedio de calificaciones de la carrera de BIOQUIMICA es: ", CVBioquimica, "%"))

## [1] "El coeficiente de variación CV del promedio de calificaciones de la carrera de BIOQUIMICA es: 4.09 %"

CVQuimica <- round(sd(quimica$promedio) / mean(quimica$promedio) * 100, 2
)
print(paste("El coeficiente de variación CV del promedio de calificaciones de la carrera de QUIMICA es: ", CVQuimica, "%"))

## [1] "El coeficiente de variación CV del promedio de calificaciones de la carrera de QUIMICA es: 4.22 %"

CVCivil <- round(sd(civil$promedio) / mean(civil$promedio) * 100, 2)
print(paste("El coeficiente de variación CV del promedio de calificaciones de la carrera de CIVIL es: ", CVCivil, "%"))

## [1] "El coeficiente de variación CV del promedio de calificaciones de la carrera de CIVIL es: 4.32 %"

```

Hay mayor Dispersión y variabilidad en los datos del promedio de calificaciones de los alumnos de la carrera de CIVIL con respecto al promedio de calificaciones de los alumnos de las carreras de BIQUIMICA Y QUIMICA. con un valor de:

```
## [1] 4.32
```

El script markdown que hace lo anterior, se encuentra en:
<http://rpubs.com/rpizarro/518882>

Otro enlace del mismo autor con referencia a estos temas se encuentran en
<https://rpubs.com/rpizarro/360665>

Crear agrupación de datos con diversos parámetros en R

La siguiente práctica tiene la finalidad de crear agrupación de datos con diversos parámetros en R además de mostrar la evidencia en RPubs

5.3 Datos agrupados

Objetivo: Generar datos agrupados, tabla de frecuencia y gráfica los datos agrupados

5.3.1 Proceso:

- Identificar y mostrar datos de la muestra
- Ordenar datos y mostrar
- Encontrar número de elementos n, valores mínimos y máximos , rango y amplitud del rango de la muestra
- Determinar número de intervalos igual a 5
- Identificar rango de cada intervalo mediante: VALOR MINIMO / INTERVALOS
- Mostrar tabla de frecuencia de datos agrupados
- Plot o graficar tabla de frecuencia

5.4 Identificar datos de la muestra

```
muestra <- c(18, 19, 20, 24, 26, 28, 18, 19, 23, 16, 21, 30, 23, 27, 29, 15, 18, 26, 34, 45, 23, 46, 53, 23, 38, 46, 34, 13, 46, 87, 46, 62, 56, 27, 22, 29, 65, 54, 76, 86, 74, 34, 45, 54, 65, 76, 23, 45)
```

```
muestra
```

```
## [1] 18 19 20 24 26 28 18 19 23 16 21 30 23 27 29 15 18 26 34 45 23 46 53
```

```
## [24] 23 38 46 34 13 46 87 46 62 56 27 22 29 65 54 76 86 74 34 45 54 65 76
```

```
## [47] 23 45
```

5.5 Ordenar datos y mostrar

```
muestraord <- sort(muestra)
```

```
muestraord
```

```
## [1] 13 15 16 18 18 18 19 19 20 21 22 23 23 23 23 23 24 26 26 27 27 28 29
```

```
## [24] 29 30 34 34 34 38 45 45 45 46 46 46 46 53 54 54 56 62 65 65 74 76
```

```
## [47] 86 87
```

5.6 Encontrar número de elementos n, valores máximos y mínimos, rango y amplitud del rango de la muestra

```
n <- length(muestra)
n

## [1] 48

max(muestra)

## [1] 87

min(muestra)

## [1] 13

rango <- range(muestra) # Valores mínimo y máximo
rango

## [1] 13 87

amplitud <- diff(rango) # amplitud del rango. Tambien es max(muestra) - min(muestra)
amplitud

## [1] 74
```

5.7 Agrupando datos de manera habitual

- ¿Cuántos intervalos se quiere tener? ?Cuántos grupos?
- Determinar número de intervalos igual a 5

```
# Fórmula : valor mínimo / intervalos
nintervalos <- 5 # Número de intervalos que se desea
rangointervalos <- amplitud / nintervalos
rangointervalos

## [1] 14.8

# paste significa concatenar
print(paste("Los valores de cada grupos van ...", "de ", rangointervalos,
" en ", rangointervalos, " a partir de :", min(muestra)))

## [1] "Los valores de cada grupos van ... de 14.8 en 14.8 a partir
de : 13"
```

5.8 Tabla de frecuencia de datos agrupados

Se empieza del valor menor para evitar errores de agrupamiento

```
tabla.intervalos <- transform(table(cut(muestra, breaks = 5)))
tabla.intervalos
```

```

##           Var1 Freq
## 1 (12.9,27.8]    21
## 2 (27.8,42.6]     8
## 3 (42.6,57.4]    11
## 4 (57.4,72.2]     3
## 5 (72.2,87.1]     5

```

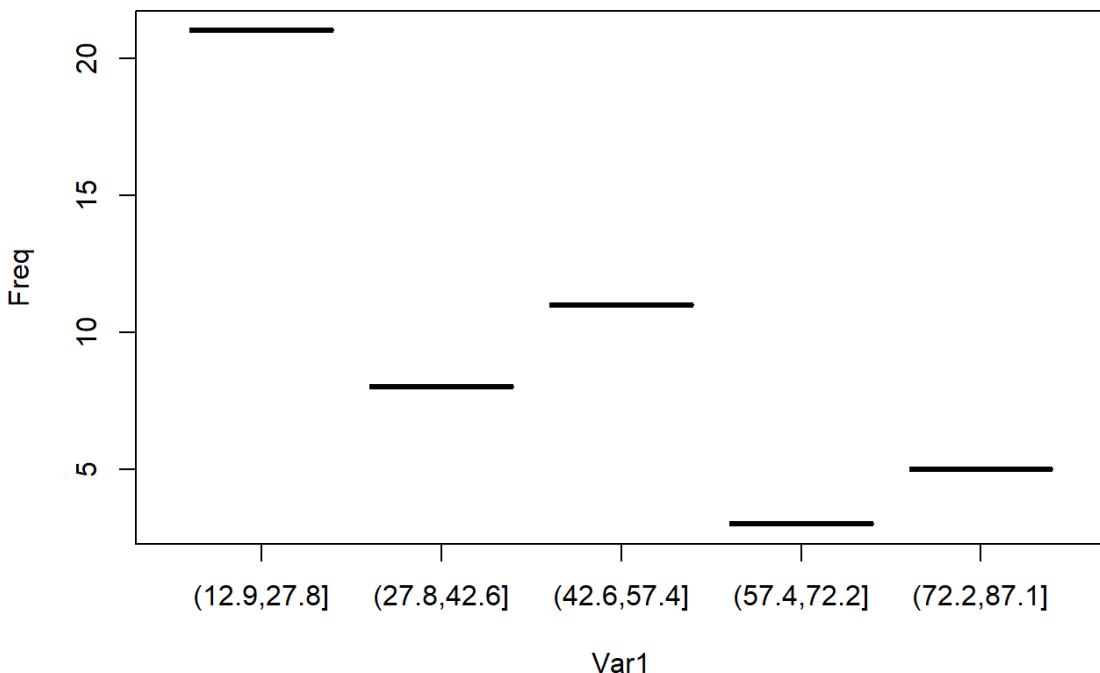
5.9 Plot o visualizar tabla de frecuencia

```

plot(tabla.intervalos, main = "¿De cuál intervalo hay más y menos elementos?")

```

¿De cuál intervalo hay más y menos elementos?



5.10 Regla de Sturges.

- De manera matemática sugiere los intérvalos y las amplitudes de cada intervalo
- ¿Cuál es la amplitud de cada intervalo?
- Fórmula: $K=1+3.322(\log N)$ /* Logaritmo de base 10 */

```

1 + 3.3222 * (log10(n)) ## Redondeado hacia arriba entonces sale 6 igual que siguiente
## [1] 6.58542

```

```

nointervalos <- nclass.Sturges(muestra) # igual al número de intervalos a
qui sale 6
nointervalos

## [1] 7

cut(muestra, breaks = nointervalos) #Cortes de cada intérvalo

## [1] (12.9,23.6] (12.9,23.6] (12.9,23.6] (23.6,34.1] (23.6,34.1]
## [6] (23.6,34.1] (12.9,23.6] (12.9,23.6] (12.9,23.6] (12.9,23.6]
## [11] (12.9,23.6] (23.6,34.1] (12.9,23.6] (23.6,34.1] (23.6,34.1]
## [16] (12.9,23.6] (12.9,23.6] (23.6,34.1] (23.6,34.1] (44.7,55.3]
## [21] (12.9,23.6] (44.7,55.3] (44.7,55.3] (12.9,23.6] (34.1,44.7]
## [26] (44.7,55.3] (23.6,34.1] (12.9,23.6] (44.7,55.3] (76.4,87.1]
## [31] (44.7,55.3] (55.3,65.9] (55.3,65.9] (23.6,34.1] (12.9,23.6]
## [36] (23.6,34.1] (55.3,65.9] (44.7,55.3] (65.9,76.4] (76.4,87.1]
## [41] (65.9,76.4] (23.6,34.1] (44.7,55.3] (44.7,55.3] (55.3,65.9]
## [46] (65.9,76.4] (12.9,23.6] (44.7,55.3]
## 7 Levels: (12.9,23.6] (23.6,34.1] (34.1,44.7] (44.7,55.3] ... (76.4,87
.1]

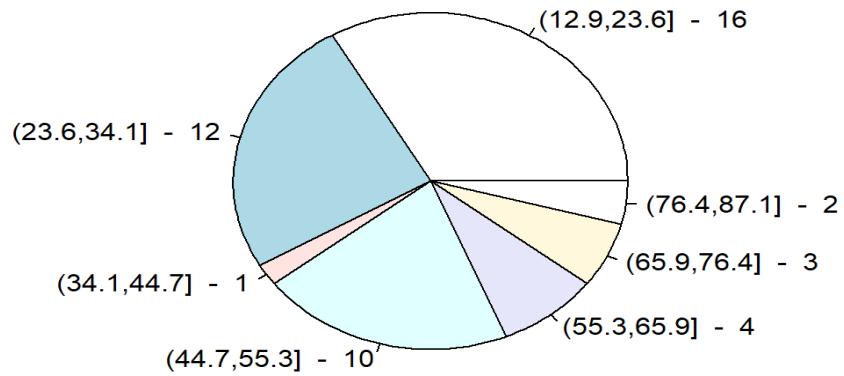
tabla.intervalos <- transform(table(cut(muestra, breaks = nointervalos)))
# son 6
tabla.intervalos

##      Var1 Freq
## 1 (12.9,23.6]   16
## 2 (23.6,34.1]   12
## 3 (34.1,44.7]    1
## 4 (44.7,55.3]   10
## 5 (55.3,65.9]    4
## 6 (65.9,76.4]    3
## 7 (76.4,87.1]    2

pie(tabla.intervalos$Freq, labels = paste(tabla.intervalos$Var1, " - ", t
abla.intervalos$Freq), main = "¿De cuál intervalo hay más y menos element
os?. Sturges")

```

¿De cuál intervalo hay más y menos elementos?. Sturges



El archivo markdown que hace lo anterior y que se encuentra en el servicio rpubs está en el siguiente enlace: <http://rpubs.com/rpizarro/518988>

6 Crear distribución y graficación de frecuencias en R

Esta práctica tiene el objetivo de crear datos, visualizar o graficar datos y determinar frecuencias en R. Crear evidencia en RPubs.

La práctica utiliza datos reales de alumnos del Instituto Tecnológico de Durango, con las siguientes variables: no, semestre, promedio y carrera.

- El no representa un consecutivo, no es un identificador real como persona de un alumno, es solo un identificador para objeto de esta práctica.
- Semestre representa el semestre en que estudia el alumno.
- Promedio es el promedio del alumno
- Carrera es la carrera que estudia un alumno

6.1 Clases de los promedios de los alumnos del ITD

Objetivo: Determinar clases conforme reglas Sturges, Scott y FD (Freedman-Diaconis). Visualizar histograma y pastel.

Proceso:

- Importar datos de promedios de alumnos de todas las carreras
<https://raw.githubusercontent.com/rpizarro/probabilidad-y-estadistica/master/practicas%20R/datos/alumnos%20inscritos%20ene-jun%202018.csv>
- Graficar histograma de alumnos con promedio mayor que cero. Sturges, Scott, FD de las CARRERAS:
 - ARQUITECTURA
 - BIOQUIMICA
 - CIVIL
 - ELECTRICA
 - ELECTRONICA
 - INDUSTRIAL

- MECANICA
- MECATRONICA
- QUIMICA
- SISTEMAS
- TIC
- GESTION
- INFORMATICA
- ADMINISTRACION
- Determinar clases conforme a **Sturges** DE CADA CARRERA
- Graficar pastel
- Determinar clases conforme a **Scott** DE CADA CARRERA
- Graficar pastel
- Determinar clases conforme a **FD** DE CADA CARRERA
- Graficar pastel
- Interpretar salidas de datos

6.2 Los datos

```

datos <- read.csv("https://raw.githubusercontent.com/rpizarrog/probabilidad-y-estadistica/master/practicas%20R/datos/alumnos%20inscritos%20ene-jun%202018.csv", header = TRUE, sep = ",")  
  

# datos # Los datos le ponemos comentario # para que no aparezcan son muchos  
  

head(datos) # solo los primeros  
  

##   no semestre promedio      carrera  

## 1 1          7    82.27 ARQUITECTURA  

## 2 2          11    82.59 ARQUITECTURA  

## 3 3          8    87.47 ARQUITECTURA  

## 4 4          12    83.18 ARQUITECTURA  

## 5 5          11    83.72 ARQUITECTURA  

## 6 6          4    90.11 ARQUITECTURA  
  

carreras <- unique(datos$carrera) # Cuales carreras  

carreras  
  

## [1] ARQUITECTURA    BIOQUIMICA      CIVIL          ELECTRICA  

## [5] ELECTRONICA     INDUSTRIAL      MECANICA      MECATRONICA  

## [9] QUIMICA         SISTEMAS       GESTION       TIC

```

```

## [13] INFORMATICA      ADMINISTRACION
## 14 Levels: ADMINISTRACION ARQUITECTURA BIOQUIMICA CIVIL ... TIC

6.2.1 ARQUITECTURA

6.2.1.1 Histogramas de ARQUITECTURA. Sturges, Scott y FD

arquitectura <- datos$promedio[which(datos$promedio > 0 & datos$semestre =
= 2 & datos$carrera == 'ARQUITECTURA')]

arquitectura

##  [1] 85.60 88.33 89.83 89.67 90.83 87.50 95.00 84.83 80.00 82.80 90.33
## [12] 83.20 85.83 87.20 85.50 90.17 86.83 83.67 87.40 92.83 86.33 82.60
## [23] 84.67 83.67 86.00 88.75 90.00 91.50 84.75 85.60 89.75 78.00 82.00
## [34] 79.83 81.40 84.33 88.00 90.33 86.33 84.17 90.50 86.17 81.40 81.80
## [45] 86.67 87.33 88.17 77.33 89.50 84.50 81.00 84.83 79.33 87.00 81.33
## [56] 84.33 88.17 85.60 78.75 88.33 84.00 79.20 88.00 88.67 84.71 83.50
## [67] 84.50 83.67 83.17 79.20 83.80 85.67

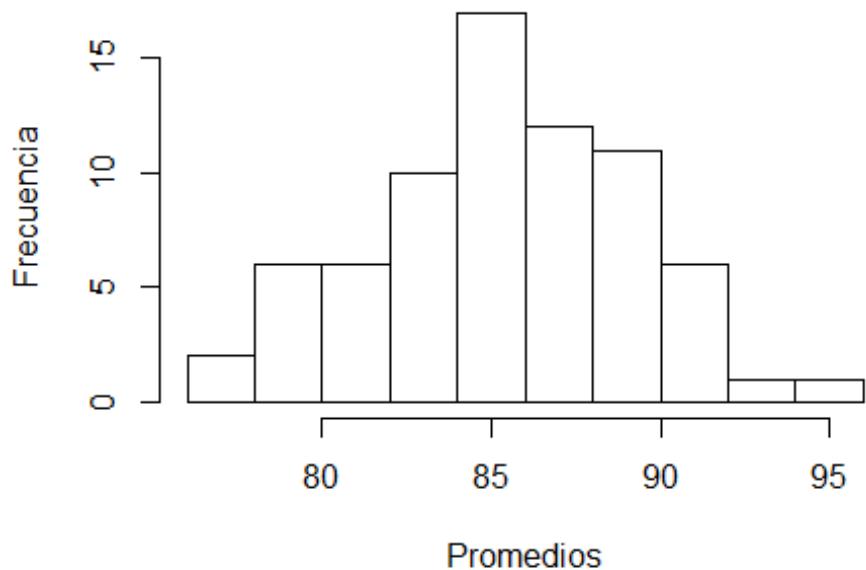
n <- length(arquitectura)

n
## [1] 72

hist(arquitectura, breaks = "Sturges", freq = TRUE, main = "Promedios de
los alumnos ARQUITECTURA del ITD. Sturges", xlab = "Promedios", ylab = "F
recuencia")

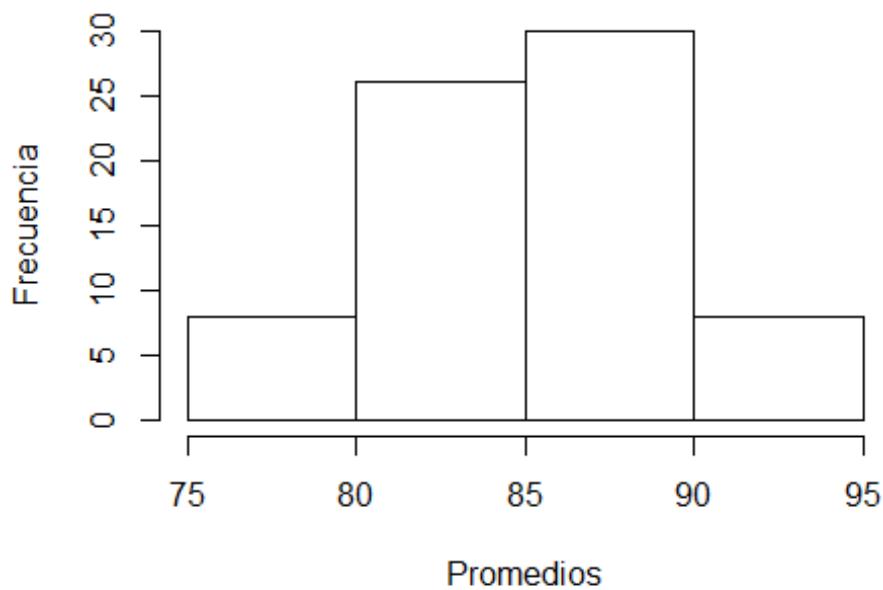
```

Promedios de los alumnos ARQUITECTURA del ITD. Scott



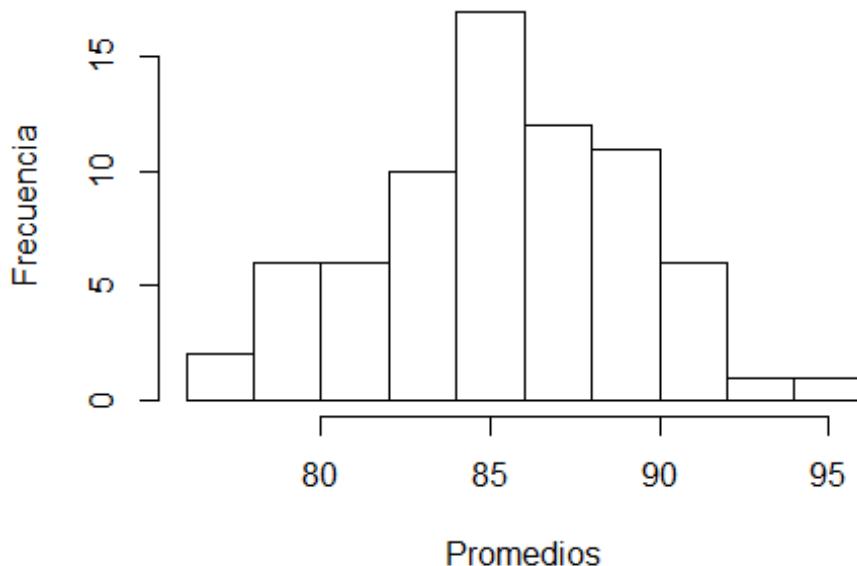
```
hist(arquitectura, breaks = "Scott", freq = TRUE, main = "Promedios de los alumnos ARQUITECTURA del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos ARQUITECTURA del ITD. Freedman-Diaconis



```
hist(arquitectura, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos ARQUITECTURA del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
)
```

Promedios de los alumnos ARQUITECTURA del ITD.



6.2.1.2 Tablas de Frecuencias de ARQUITECTURA. Sturges

```
clases <- nclass.Sturges(arquitectura)
clases

## [1] 8

tabla.intervalos <- transform(table(cut(arquitectura, breaks = clases)))

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (77.3,79.5]    6
## 2 (79.5,81.7]    6
## 3 (81.7,84]    11
## 4 (84,86.2]    18
## 5 (86.2,88.4]   16
## 6 (88.4,90.6]   11
## 7 (90.6,92.8]    2
## 8 (92.8,95]     2

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.inter
```

```

valos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq  Freq.Rel Freq.Porc
## 1 (77.3,79.5]    6 0.08333333  8.333333
## 2 (79.5,81.7]    6 0.08333333  8.333333
## 3 (81.7,84]     11 0.15277778 15.277778
## 4 (84,86.2]      18 0.25000000 25.000000
## 5 (86.2,88.4]    16 0.22222222 22.222222
## 6 (88.4,90.6]    11 0.15277778 15.277778
## 7 (90.6,92.8]     2 0.02777778  2.777778
## 8 (92.8,95]      2 0.02777778  2.777778

# ****
*****  

#Inicializamos carreras.clases <- NULL
carreras.clases <- NULL # Solo una vez inicializar en este script

```

```

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones practicas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRERA"
A" = "ARQUITECTURA"))

```

6.2.1.3 Gráficas de pastel. Arquitectura Sturges

```

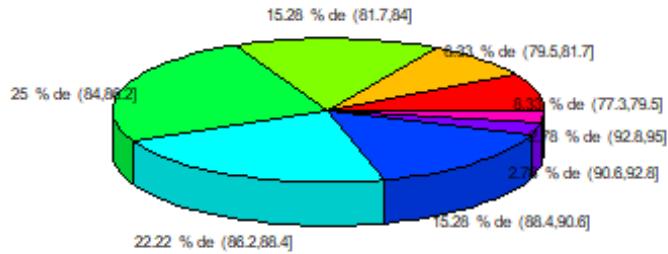
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 72 Observaciones



6.2.1.4 Tablas de Frecuencias de ARQUITECTURA. Scott

```
clases <- nclass.scott(arquitectura)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(arquitectura, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (77.3,80.3]    8
## 2 (80.3,83.2]   10
## 3 (83.2,86.2]   23
## 4 (86.2,89.1]   18
## 5 (89.1,92.1]   11
## 6  (92.1,95]     2

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)
```

```
tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (77.3,80.3]    8 0.11111111 11.111111
## 2 (80.3,83.2]   10 0.13888889 13.888889
## 3 (83.2,86.2]   23 0.31944444 31.944444
## 4 (86.2,89.1]   18 0.25000000 25.000000
## 5 (89.1,92.1]   11 0.15277778 15.277778
## 6 (92.1,95]      2 0.02777778  2.777778
```

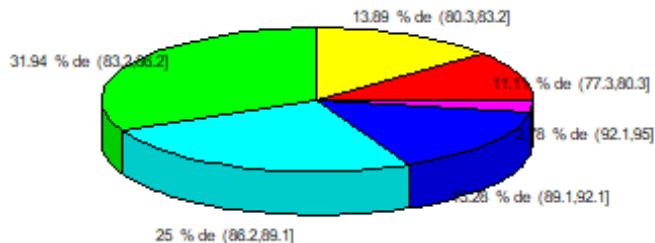
6.2.1.5 Gráficas de pastel. Arquitectura Scott

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
nervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 72 Observaciones



6.2.1.6 Tablas de Frecuencias de ARQUITECTURA. FD

```
clases <- nclass.FD(arquitectura)
clases
```

```

## [1] 8

tabla.intervalos <- transform(table(cut(arquitectura, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1 Freq
## 1 (77.3,79.5]    6
## 2 (79.5,81.7]    6
## 3 (81.7,84]   11
## 4 (84,86.2]   18
## 5 (86.2,88.4]   16
## 6 (88.4,90.6]   11
## 7 (90.6,92.8]    2
## 8 (92.8,95]     2

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq  Freq.Rel Freq.Porc
## 1 (77.3,79.5]    6 0.08333333 8.333333
## 2 (79.5,81.7]    6 0.08333333 8.333333
## 3 (81.7,84]   11 0.15277778 15.277778
## 4 (84,86.2]   18 0.25000000 25.000000
## 5 (86.2,88.4]   16 0.22222222 22.222222
## 6 (88.4,90.6]   11 0.15277778 15.277778
## 7 (90.6,92.8]    2 0.02777778 2.777778
## 8 (92.8,95]     2 0.02777778 2.777778

```

6.2.1.7 Gráficas de pastel. Arquitectura FD

```

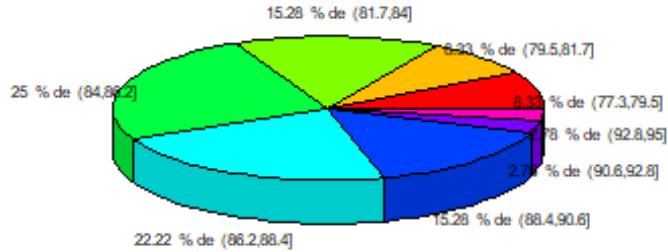
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 72 Observaciones



6.2.2 BIOQUIMICA

6.2.2.1 Histogramas de BIOQUIMICA Sturges, Scott y FD

```
bioquimica <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2 & datos$carrera == 'BIOQUIMICA'))]

bioquimica

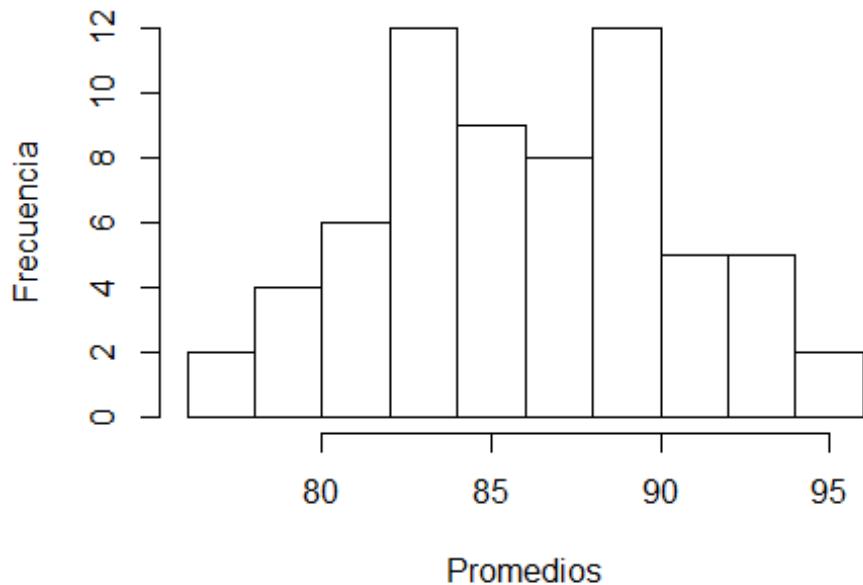
## [1] 81.17 89.50 83.60 88.17 87.75 85.83 91.67 87.50 84.20 83.80 90.25
## [12] 83.67 77.57 88.40 77.50 88.00 94.67 87.00 83.17 79.67 90.00 92.17
## [23] 93.17 84.67 84.00 85.40 88.20 89.00 88.33 86.80 94.00 93.17 91.20
## [34] 85.40 93.33 88.33 89.17 83.80 81.00 82.33 81.40 79.50 80.00 84.83
## [45] 83.40 79.40 82.50 90.83 88.17 88.00 84.40 81.20 83.00 81.80 82.20
## [56] 87.83 86.33 85.33 83.50 89.00 81.67 90.50 85.33 89.50 95.17

n <- length(bioquimica)

n
## [1] 65

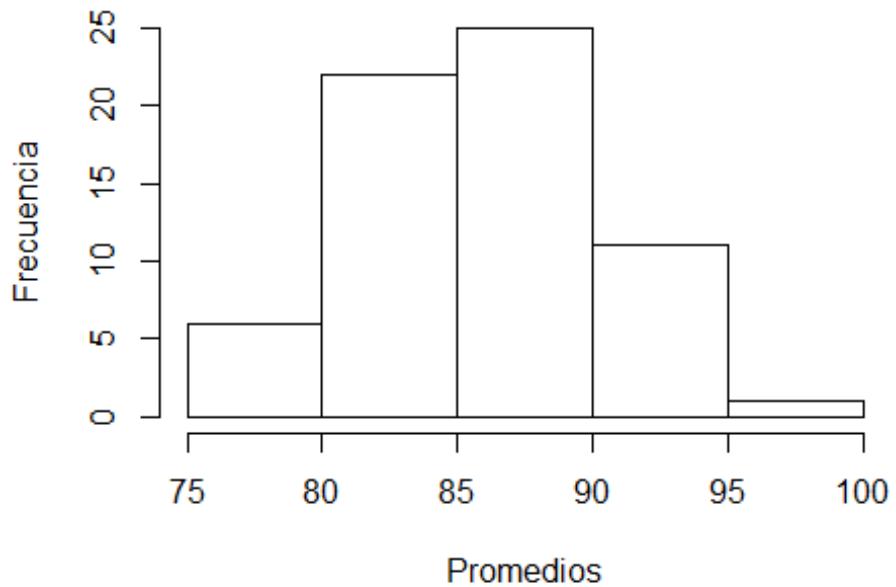
hist(bioquimica, breaks = "Sturges", freq = TRUE, main = "Promedios de los alumnos BIOQUIMICA del ITD. Sturges", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos BIOQUIMICA del ITD. Scott



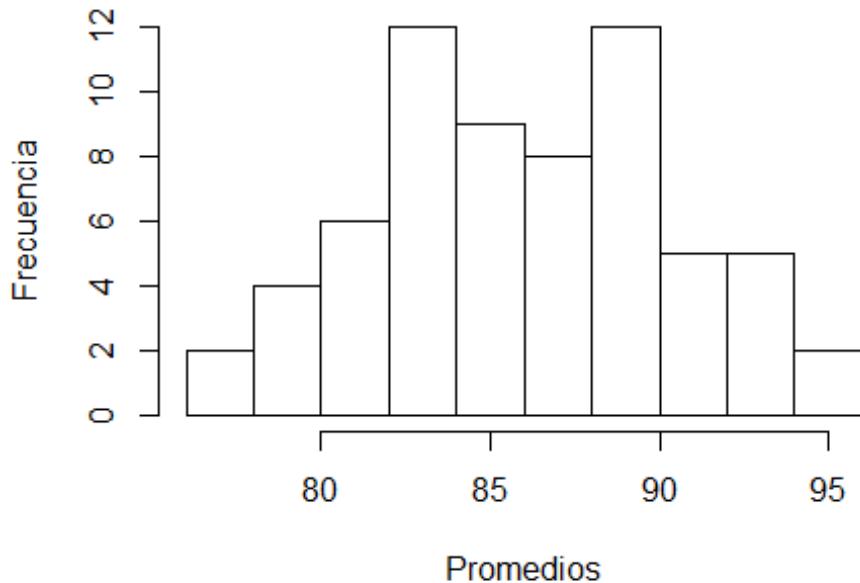
```
hist(bioquimica, breaks = "Scott", freq = TRUE, main = "Promedios de los  
alumnos BIOQUIMICA del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos BIOQUIMICA del ITD. Scott



```
hist(bioquimica, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos BIOQUIMICA del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos BIOQUIMICA del ITD. F



6.2.2.2 Tablas de Frecuencias de BIOQUIMICA Sturges

```
clases <- nclass.Sturges(bioquimica)
clases
## [1] 8

tabla.intervalos <- transform(table(cut(bioquimica, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1 Freq
## 1 (77.5,79.7]    5
## 2 (79.7,81.9]    7
## 3 (81.9,84.1]   12
## 4 (84.1,86.3]   10
## 5 (86.3,88.5]   13
## 6 (88.5,90.8]    8
## 7 (90.8,93]      4
## 8 (93,95.2]      6

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (77.5,79.7]    5 0.07692308  7.692308
## 2 (79.7,81.9]    7 0.10769231 10.769231
## 3 (81.9,84.1]   12 0.18461538 18.461538
## 4 (84.1,86.3]   10 0.15384615 15.384615
## 5 (86.3,88.5]   13 0.20000000 20.000000
## 6 (88.5,90.8]    8 0.12307692 12.307692
## 7 (90.8,93]      4 0.06153846  6.153846
## 8 (93,95.2]      6 0.09230769  9.230769

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones pr?cticas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRER
A" = "BIOQUIMICA"))

```

6.2.2.3 Gráficas de pastel. BIOQUIMICA Sturges

```

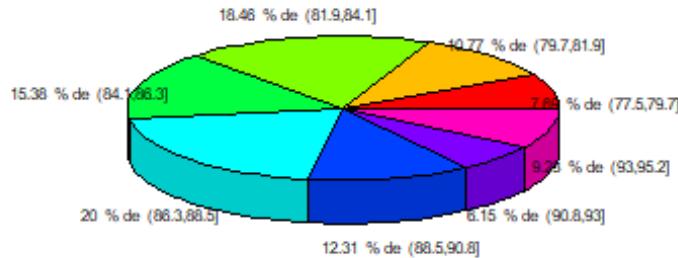
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 65 Observaciones



6.2.2.4 Tablas de Frecuencias de BIOQUIMICA Scott

```
clases <- nclass.scott(bioquimica)
clases

## [1] 5

tabla.intervalos <- transform(table(cut(bioquimica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1  (77.5,81]
## 2  (81,84.6]
## 3 (84.6,88.1]
## 4 (88.1,91.6]
## 5 (91.6,95.2]

## Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos
```

```

##           Var1 Freq  Freq.Rel Freq.Porc
## 1  (77.5,81]    7 0.1076923 10.76923
## 2  (81,84.6]   19 0.2923077 29.23077
## 3 (84.6,88.1]   15 0.2307692 23.07692
## 4 (88.1,91.6]   16 0.2461538 24.61538
## 5 (91.6,95.2]    8 0.1230769 12.30769

```

6.2.2.5 Gráficas de pastel. BIOQUIMICA Scott

```

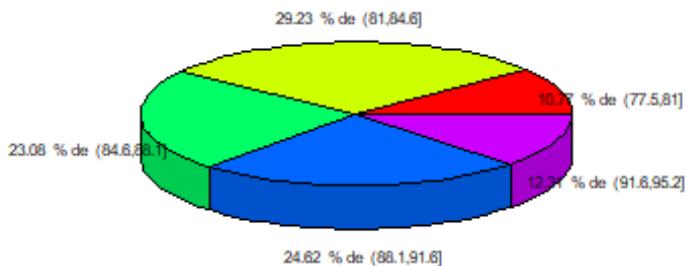
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 65 Observaciones



6.2.2.6 Tablas de Frecuencias de BIOQUIMICA FD

```

clases <- nclass.FD(bioquimica)
clases

## [1] 7

```

```

tabla.intervalos <- transform(table(cut(arquitectura, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (77.3,79.9]    7
## 2 (79.9,82.4]    7
## 3 (82.4,84.9]   20
## 4 (84.9,87.4]   16
## 5 (87.4,90]     13
## 6 (90,92.5]      7
## 7 (92.5,95]      2

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##          Var1 Freq  Freq.Rel Freq.Porc
## 1 (77.3,79.9]    7 0.10769231 10.769231
## 2 (79.9,82.4]    7 0.10769231 10.769231
## 3 (82.4,84.9]   20 0.30769231 30.769231
## 4 (84.9,87.4]   16 0.24615385 24.615385
## 5 (87.4,90]     13 0.20000000 20.000000
## 6 (90,92.5]      7 0.10769231 10.769231
## 7 (92.5,95]      2 0.03076923  3.076923

```

6.2.2.7 Gráficas de pastel. BIOQUIMICA FD

```

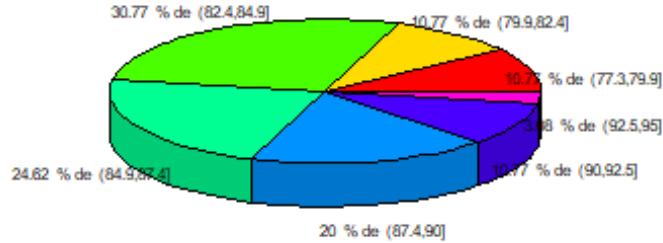
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 65 Observaciones



6.2.3 CIVIL

6.2.3.1 Histogramas de CIVIL Sturges, Scott y FD

```
civil <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2 & datos$carrera == 'CIVIL')]

civil

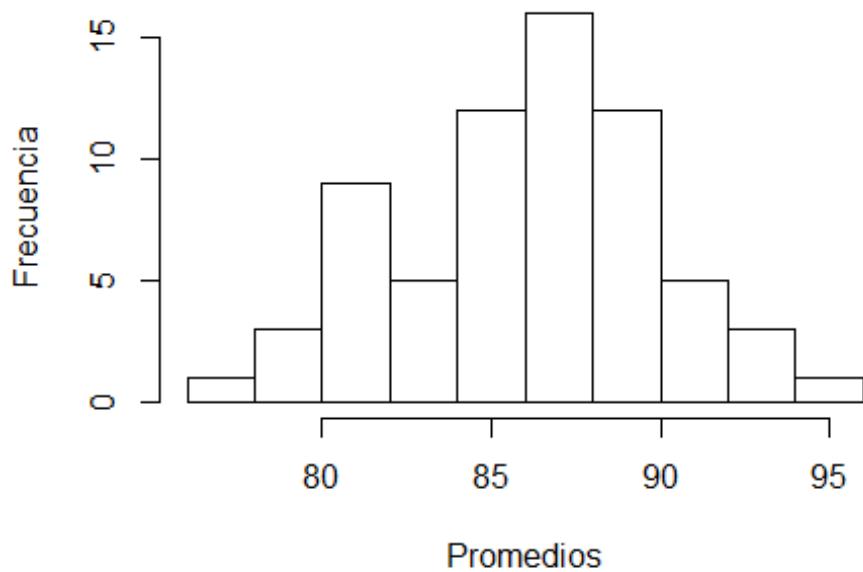
## [1] 90.00 88.33 83.67 85.00 92.50 86.50 87.00 90.00 82.00 81.00 86.20
## [12] 88.60 81.67 86.67 88.00 79.00 85.33 91.50 94.67 85.00 81.60 83.00
## [23] 88.20 87.25 85.67 81.50 79.00 85.60 84.83 89.17 90.25 93.00 82.20
## [34] 88.33 89.17 84.83 89.83 87.17 82.50 85.17 82.75 90.50 78.20 86.33
## [45] 86.00 91.33 85.33 86.33 87.67 92.83 86.40 87.67 77.33 91.17 85.80
## [56] 87.67 87.83 89.83 87.00 82.00 87.40 85.20 89.20 81.50 81.50 81.00
## [67] 88.50

n <- length(civil)

n
## [1] 67

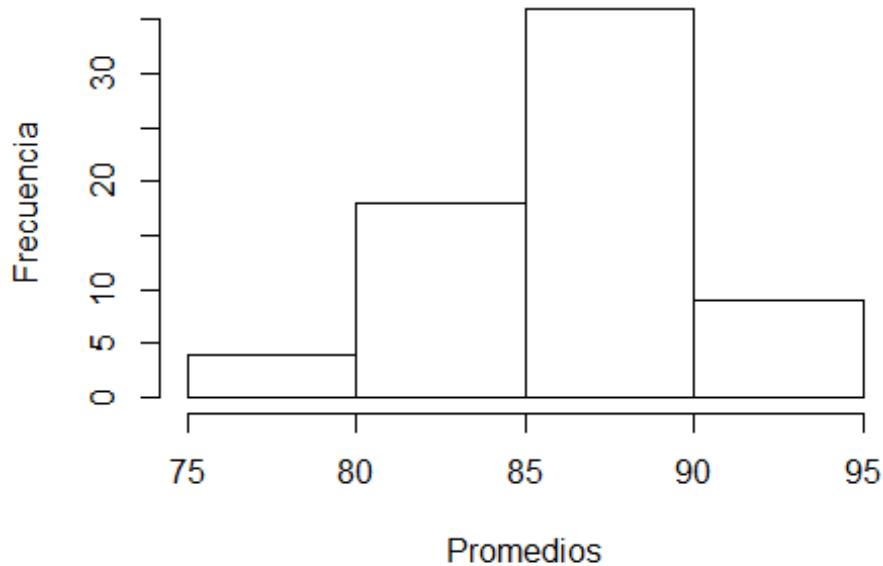
hist(civil, breaks = "Sturges", freq = TRUE, main = "Promedios de los alumnos CIVIL del ITD. Sturges", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos CIVIL del ITD. Sturges



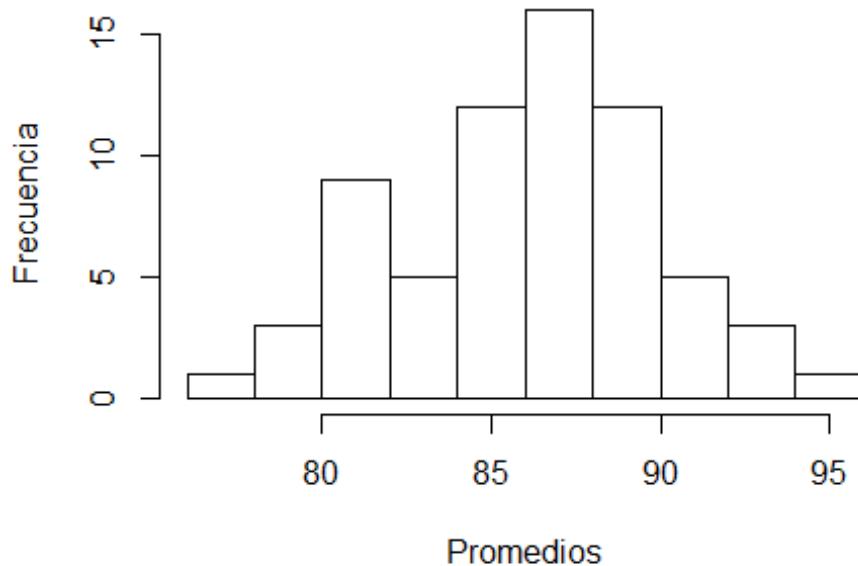
```
hist(civil, breaks = "Scott", freq = TRUE, main = "Promedios de los alumnos CIVIL del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos CIVIL del ITD. Scott



```
hist(civil, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos CIVIL del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos CIVIL del ITD. FD



6.2.3.2 Tablas de Frecuencias de CIVIL Sturges

```
clases <- nclass.Sturges(civil)
clases

## [1] 8

tabla.intervalos <- transform(table(cut(civil, breaks = clases)))

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (77.3,79.5]    4
## 2 (79.5,81.7]    6
## 3 (81.7,83.8]    8
## 4 (83.8,86]   12
## 5 (86,88.2]   16
## 6 (88.2,90.3]   13
## 7 (90.3,92.5]    5
## 8 (92.5,94.7]    3

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (77.3,79.5]    4 0.05970149  5.970149
## 2 (79.5,81.7]    6 0.08955224  8.955224
## 3 (81.7,83.8]    8 0.11940299 11.940299
## 4 (83.8,86]   12 0.17910448 17.910448
## 5 (86,88.2]   16 0.23880597 23.880597
## 6 (88.2,90.3]   13 0.19402985 19.402985
## 7 (90.3,92.5]    5 0.07462687  7.462687
## 8 (92.5,94.7]    3 0.04477612  4.477612

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones pr?cticas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRER
A" = "CIVIL"))

```

6.2.3.3 Gráficas de pastel. CIVIL Sturges

```

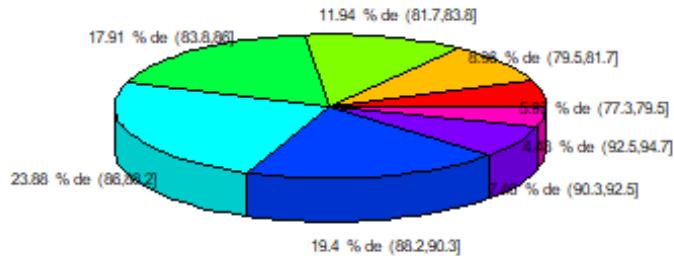
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 67 Observaciones



6.2.3.4 Tablas de Frecuencias de CIVIL Scott

```
clases <- nclass.scott(civil)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(civil, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (77.3,80.2]    4
## 2 (80.2,83.1]   13
## 3 (83.1,86]    13
## 4 (86,88.9]    21
## 5 (88.9,91.8]   12
## 6 (91.8,94.7]    4

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)
```

```

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (77.3,80.2]    4 0.05970149  5.970149
## 2 (80.2,83.1]   13 0.19402985 19.402985
## 3 (83.1,86]     13 0.19402985 19.402985
## 4 (86,88.9]     21 0.31343284 31.343284
## 5 (88.9,91.8]   12 0.17910448 17.910448
## 6 (91.8,94.7]    4 0.05970149  5.970149

```

6.2.3.5 Gráficas de pastel. CIVIL Scott

```

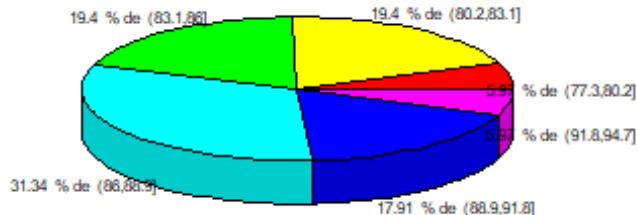
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
nervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 67 Observaciones



6.2.3.6 Tablas de Frecuencias de CIVIL FD

```

clases <- nclass.FD(civil)
clases

```

```

## [1] 7

tabla.intervalos <- transform(table(cut(civil, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (77.3,79.8]    4
## 2 (79.8,82.3]   10
## 3 (82.3,84.8]    4
## 4 (84.8,87.2]   21
## 5 (87.2,89.7]   15
## 6 (89.7,92.2]    9
## 7 (92.2,94.7]    4

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##          Var1 Freq  Freq.Rel Freq.Porc
## 1 (77.3,79.8]    4 0.05970149  5.970149
## 2 (79.8,82.3]   10 0.14925373 14.925373
## 3 (82.3,84.8]    4 0.05970149  5.970149
## 4 (84.8,87.2]   21 0.31343284 31.343284
## 5 (87.2,89.7]   15 0.22388060 22.388060
## 6 (89.7,92.2]    9 0.13432836 13.432836
## 7 (92.2,94.7]    4 0.05970149  5.970149

```

6.2.3.7 Gráficas de pastel. CIVIL FD

```

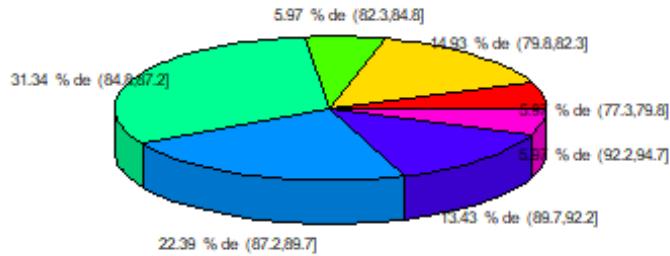
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 67 Observaciones

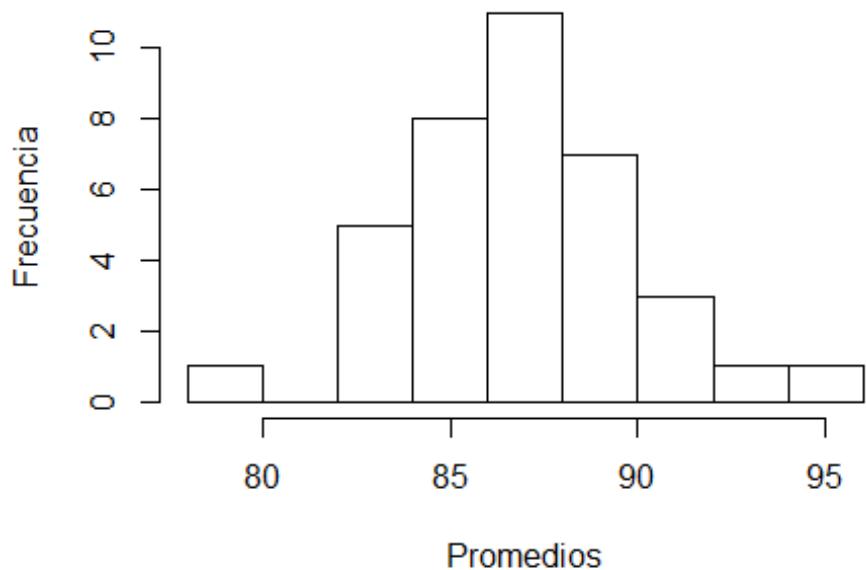


6.2.4 ELECTRICA

6.2.4.1 Histogramas de ELECTRICA Sturges, Scott y FD

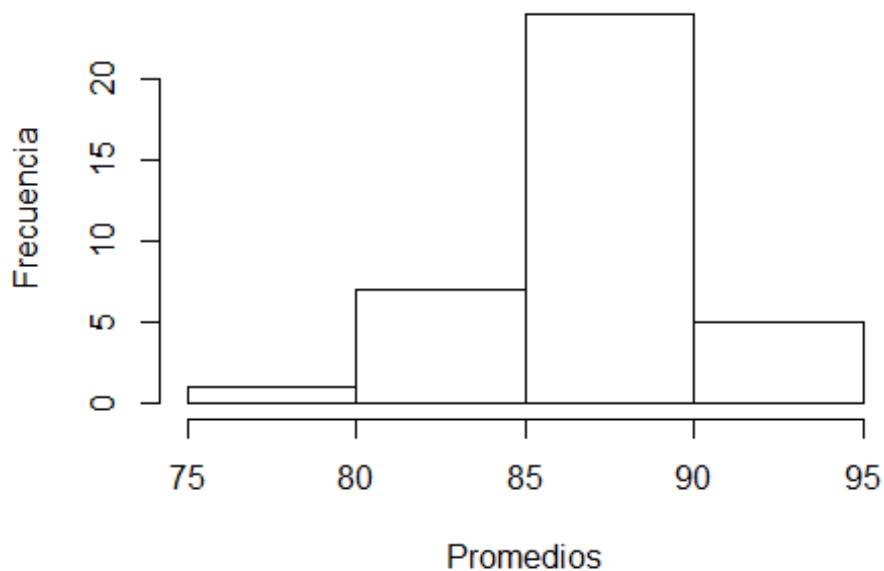
```
electrica <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2  
& datos$carrera == 'ELECTRICA'))]  
  
electrica  
## [1] 86.50 91.00 88.75 88.33 90.83 88.83 87.00 83.50 89.00 86.00 85.67  
## [12] 83.17 87.25 89.75 83.00 85.60 88.00 84.67 87.25 92.83 87.50 84.67  
## [23] 88.33 86.67 87.80 82.33 89.00 94.17 87.00 85.83 83.00 86.67 78.67  
## [34] 85.83 90.17 87.50 85.50  
  
n <- length(electrica)  
  
n  
## [1] 37  
  
hist(electrica, breaks = "Sturges", freq = TRUE, main = "Promedios de los  
alumnos ELECTRICA del ITD. Sturges", xlab = "Promedios", ylab = "Frecuenc  
ia")
```

Promedios de los alumnos ELECTRICA del ITD. Stur



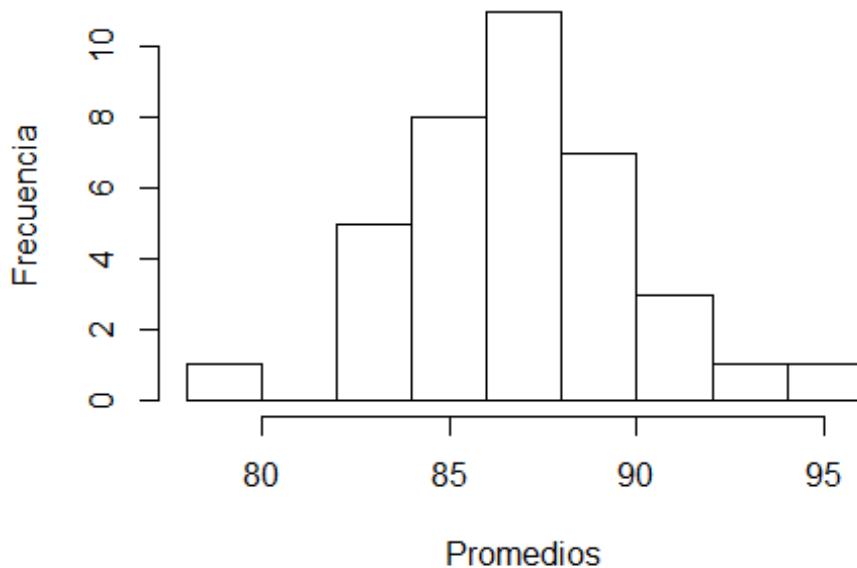
```
hist(electrica, breaks = "Scott", freq = TRUE, main = "Promedios de los alumnos ELECTRICA del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos ELECTRICA del ITD. Sc



```
hist(electrica, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos ELECTRICA del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos ELECTRICA del ITD. F



6.2.4.2 Tablas de Frecuencias de ELECTRICA Sturges

```
clases <- nclass.Sturges(electrica)
clases
## [1] 7

tabla.intervalos <- transform(table(cut(electrica, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1 Freq
## 1 (78.7,80.9]    1
## 2 (80.9,83.1]    3
## 3 (83.1,85.3]    4
## 4 (85.3,87.5]   15
## 5 (87.5,89.7]    8
## 6 (89.7,92]      4
## 7 (92,94.2]      2

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##      Var1 Freq   Freq.Rel Freq.Porc
## 1 (78.7,80.9]    1 0.02702703  2.702703
## 2 (80.9,83.1]    3 0.08108108  8.108108
## 3 (83.1,85.3]    4 0.10810811 10.810811
## 4 (85.3,87.5]   15 0.40540541 40.540541
## 5 (87.5,89.7]    8 0.21621622 21.621622
## 6 (89.7,92]      4 0.10810811 10.810811
## 7 (92,94.2]      2 0.05405405  5.405405

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones prácticas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRERA"
= "ELECTRICA"))

```

6.2.4.3 Gráficas de pastel. ELECTRICA Sturges

```

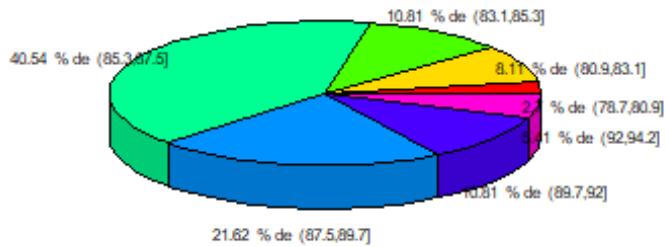
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 37 Observaciones



6.2.4.4 Tablas de Frecuencias de ELECTRICA Scott

```
clases <- nclass.scott(electrica)
clases

## [1] 5

tabla.intervalos <- transform(table(cut(electrica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (78.7,81.8]    1
## 2 (81.8,84.9]    7
## 3 (84.9,88]   16
## 4 (88,91.1]   11
## 5 (91.1,94.2]    2

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos
```

```

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (78.7,81.8]    1 0.02702703  2.702703
## 2 (81.8,84.9]    7 0.18918919 18.918919
## 3 (84.9,88]    16 0.43243243 43.243243
## 4 (88,91.1]    11 0.29729730 29.729730
## 5 (91.1,94.2]    2 0.05405405  5.405405

```

6.2.4.5 Gráficas de pastel. ELECTRICA Scott

```

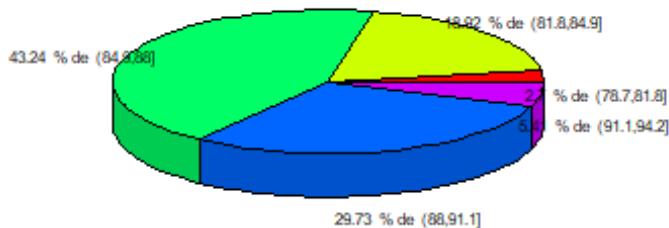
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 37 Observaciones



6.2.4.6 Tablas de Frecuencias de ELECTRICA FD

```

clases <- nclass.FD(electrica)
clases

## [1] 9

```

```

tabla.intervalos <- transform(table(cut(electrica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (78.7,80.4]    1
## 2 (80.4,82.1]    0
## 3 (82.1,83.8]    5
## 4 (83.8,85.6]    3
## 5 (85.6,87.3]   12
## 6 (87.3,89]    10
## 7 (89,90.7]     2
## 8 (90.7,92.4]    2
## 9 (92.4,94.2]    2

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##          Var1 Freq  Freq.Rel Freq.Porc
## 1 (78.7,80.4]    1 0.02702703  2.702703
## 2 (80.4,82.1]    0 0.00000000  0.000000
## 3 (82.1,83.8]    5 0.13513514 13.513514
## 4 (83.8,85.6]    3 0.08108108  8.108108
## 5 (85.6,87.3]   12 0.32432432 32.432432
## 6 (87.3,89]     10 0.27027027 27.027027
## 7 (89,90.7]      2 0.05405405  5.405405
## 8 (90.7,92.4]    2 0.05405405  5.405405
## 9 (92.4,94.2]    2 0.05405405  5.405405

```

6.2.4.7 Gráficas de pastel. ELECTRICA FD

```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

etiquetas

## [1] "2.7 % de (78.7,80.4]"    "0 % de (80.4,82.1]"
## [3] "13.51 % de (82.1,83.8]"  "8.11 % de (83.8,85.6]"
## [5] "32.43 % de (85.6,87.3]"  "27.03 % de (87.3,89]"

```

```

## [7] "5.41 % de (89,90.7]"    "5.41 % de (90.7,92.4]"
## [9] "5.41 % de (92.4,94.2)"

#pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pa
ste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5) # N
o puede hacer pastel porque trae 0 la frecuencia. Pendiente

```

6.2.5 ELECTRONICA

6.2.5.1 Histogramas de ELECTRONICA Sturges, Scott y FD

```

electronica <- datos$promedio[which(datos$promedio > 0 & datos$semestre ==
2 & datos$carrera == 'ELECTRONICA')]

electronica

## [1] 90.17 95.33 94.25 92.67 93.00 92.50 92.00 93.67 95.33 88.33 96.00
## [12] 97.50 95.00 97.00 99.17 96.50 97.50 91.00 97.17 90.60 98.50 93.80
## [23] 91.80 90.50 94.75 91.67 92.00 93.00 96.33 96.17 92.17 92.60 93.40
## [34] 93.33 70.00

n <- length(electronica)

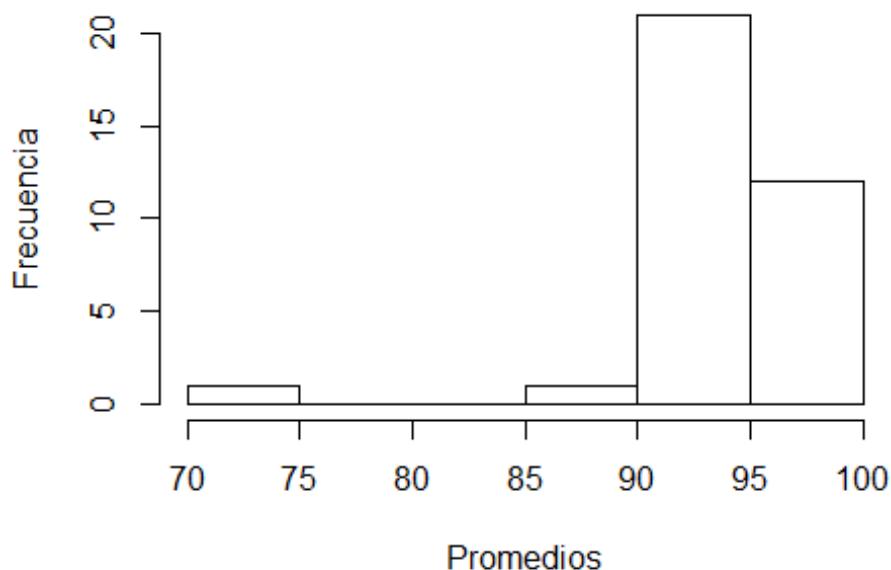
n

## [1] 35

hist(electronica, breaks = "Sturges", freq = TRUE, main = "Promedios de 1
os alumnos ELECTRONICA del ITD. Sturges", xlab = "Promedios", ylab = "Fre
cuencia")

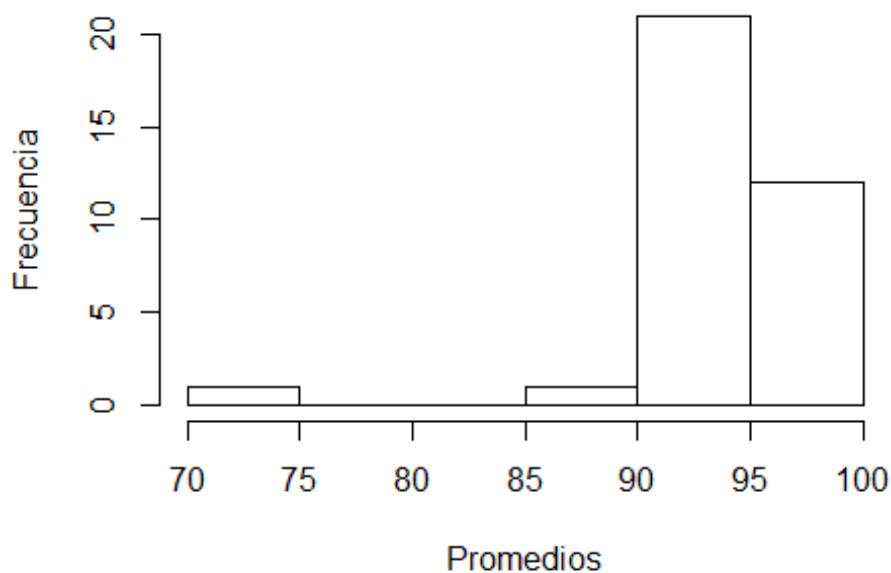
```

Promedios de los alumnos ELECTRONICA del ITD. Scott



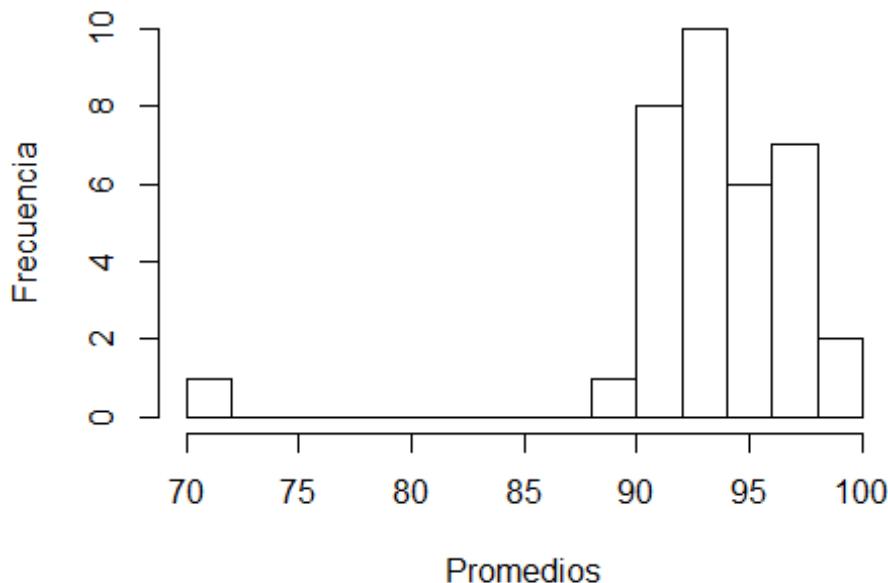
```
hist(electronica, breaks = "Scott", freq = TRUE, main = "Promedios de los
alumnos ELECTRONICA del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos ELECTRONICA del ITD. Sturges



```
hist(electronica, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos ELECTRONICA del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos ELECTRONICA del ITD.



6.2.5.2 Tablas de Frecuencias de ELECTRONICA Sturges

```
clases <- nclass.Sturges(electronica)
clases
## [1] 7

tabla.intervalos <- transform(table(cut(electronica, breaks = clases)))

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (70,74.2]    1
## 2 (74.2,78.3]    0
## 3 (78.3,82.5]    0
## 4 (82.5,86.7]    0
## 5 (86.7,90.8]    4
## 6 (90.8,95]   18
## 7 (95,99.2]   12

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##      Var1 Freq Freq.Rel Freq.Porc
## 1 (70,74.2]    1 0.02857143  2.857143
## 2 (74.2,78.3]   0 0.00000000  0.000000
## 3 (78.3,82.5]   0 0.00000000  0.000000
## 4 (82.5,86.7]   0 0.00000000  0.000000
## 5 (86.7,90.8]   4 0.11428571 11.428571
## 6 (90.8,95]    18 0.51428571 51.428571
## 7 (95,99.2]   12 0.34285714 34.285714

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones pr?cticas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRER
A" = "ELECTRONICA"))

```

6.2.5.3 Gráficas de pastel. ELECTRONICA Sturges

```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), "% de ", tabla.i
ntervalos$Var1)
etiquetas

## [1] "2.86 % de (70,74.2]"     "0 % de (74.2,78.3]"
## [3] "0 % de (78.3,82.5]"     "0 % de (82.5,86.7]"
## [5] "11.43 % de (86.7,90.8]" "51.43 % de (90.8,95]"
## [7] "34.29 % de (95,99.2)"

#pie3D(round(tabla.intervalos$Freq.Porc,2), Labels = etiquetas, main = pa
ste("Frecuencia de clases. % ", n, " Observaciones"), Labelcex = 0.5) # N
o puede hacer pastel porque trae 0 la frecuencia. Pendiente

```

6.2.5.4 Tablas de Frecuencias de ELECTRONICA Scott

```

clases <- nclass.scott(electronica)
clases

## [1] 6

```

```

tabla.intervalos <- transform(table(cut(electronica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1  (70,74.9]    1
## 2 (74.9,79.7]    0
## 3 (79.7,84.6]    0
## 4 (84.6,89.4]    1
## 5 (89.4,94.3]   19
## 6 (94.3,99.2]   14

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##          Var1 Freq  Freq.Rel Freq.Porc
## 1  (70,74.9]    1 0.02857143  2.857143
## 2 (74.9,79.7]    0 0.00000000  0.000000
## 3 (79.7,84.6]    0 0.00000000  0.000000
## 4 (84.6,89.4]    1 0.02857143  2.857143
## 5 (89.4,94.3]   19 0.54285714 54.285714
## 6 (94.3,99.2]   14 0.40000000 40.000000

```

6.2.5.5 Gráficas de pastel. ELECTRONICA Scott

```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)
etiquetas

## [1] "2.86 % de (70,74.9]"     "0 % de (74.9,79.7]"
## [3] "0 % de (79.7,84.6]"     "2.86 % de (84.6,89.4]"
## [5] "54.29 % de (89.4,94.3]" "40 % de (94.3,99.2)"

# pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5) #
# No puede hacer pastel porque trae 0 la frecuencia. Pendiente

```

6.2.5.6 Tablas de Frecuencias de ELECTRONICA FD

```
clases <- nclass.FD(electronica)
clases

## [1] 12

tabla.intervalos <- transform(table(cut(electronica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1  Freq
## 1 (70,72.4]    1
## 2 (72.4,74.9]   0
## 3 (74.9,77.3]   0
## 4 (77.3,79.7]   0
## 5 (79.7,82.2]   0
## 6 (82.2,84.6]   0
## 7 (84.6,87]     0
## 8 (87,89.4]     1
## 9 (89.4,91.9]    6
## 10 (91.9,94.3]   13
## 11 (94.3,96.7]    8
## 12 (96.7,99.2]    6

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##          Var1  Freq  Freq.Rel  Freq.Porc
## 1 (70,72.4]    1  0.02857143  2.857143
## 2 (72.4,74.9]   0  0.00000000  0.000000
## 3 (74.9,77.3]   0  0.00000000  0.000000
## 4 (77.3,79.7]   0  0.00000000  0.000000
## 5 (79.7,82.2]   0  0.00000000  0.000000
## 6 (82.2,84.6]   0  0.00000000  0.000000
## 7 (84.6,87]     0  0.00000000  0.000000
## 8 (87,89.4]     1  0.02857143  2.857143
## 9 (89.4,91.9]    6  0.17142857 17.142857
## 10 (91.9,94.3]   13 0.37142857 37.142857
## 11 (94.3,96.7]   8  0.22857143 22.857143
## 12 (96.7,99.2]   6  0.17142857 17.142857
```

6.2.5.7 Gráficas de pastel. ELECTRONICA FD

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)
etiquetas

## [1] "2.86 % de (70,72.4]"      "0 % de (72.4,74.9]"
## [3] "0 % de (74.9,77.3]"      "0 % de (77.3,79.7]"
## [5] "0 % de (79.7,82.2]"      "0 % de (82.2,84.6]"
## [7] "0 % de (84.6,87]"        "2.86 % de (87,89.4]"
## [9] "17.14 % de (89.4,91.9]"  "37.14 % de (91.9,94.3]"
## [11] "22.86 % de (94.3,96.7]" "17.14 % de (96.7,99.2)"

# pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = p
aste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5) # 
No puede hacer pastel porque trae 0 La frecuencia. Pendiente
```

6.2.6 INDUSTRIAL

6.2.6.1 Histogramas de INDUSTRIAL Sturges, Scott y FD

```
industrial <- datos$promedio[which(datos$promedio > 0 & datos$semestre ==
2 & datos$carrera == 'INDUSTRIAL')]

electronica

## [1] 90.17 95.33 94.25 92.67 93.00 92.50 92.00 93.67 95.33 88.33 96.00
## [12] 97.50 95.00 97.00 99.17 96.50 97.50 91.00 97.17 90.60 98.50 93.80
## [23] 91.80 90.50 94.75 91.67 92.00 93.00 96.33 96.17 92.17 92.60 93.40
## [34] 93.33 70.00

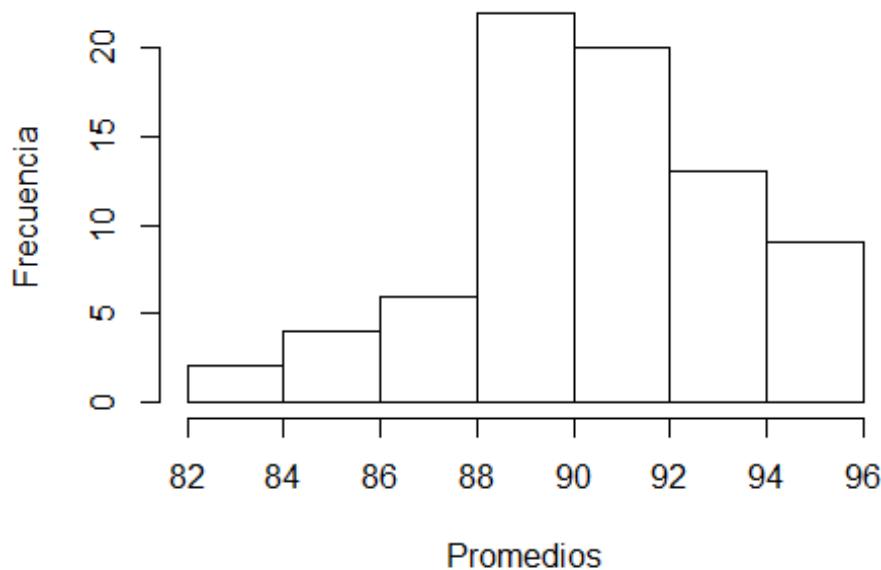
n <- length(industrial)

n

## [1] 76

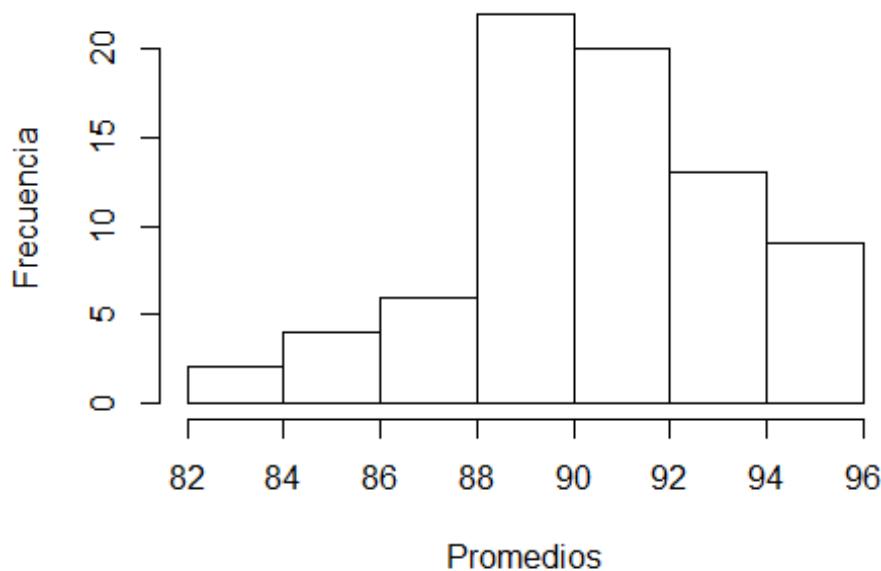
hist(industrial, breaks = "Sturges", freq = TRUE, main = "Promedios de lo
s alumnos INDUSTRIAL del ITD. Sturges", xlab = "Promedios", ylab = "Frecu
encia")
```

Promedios de los alumnos INDUSTRIAL del ITD. Stu



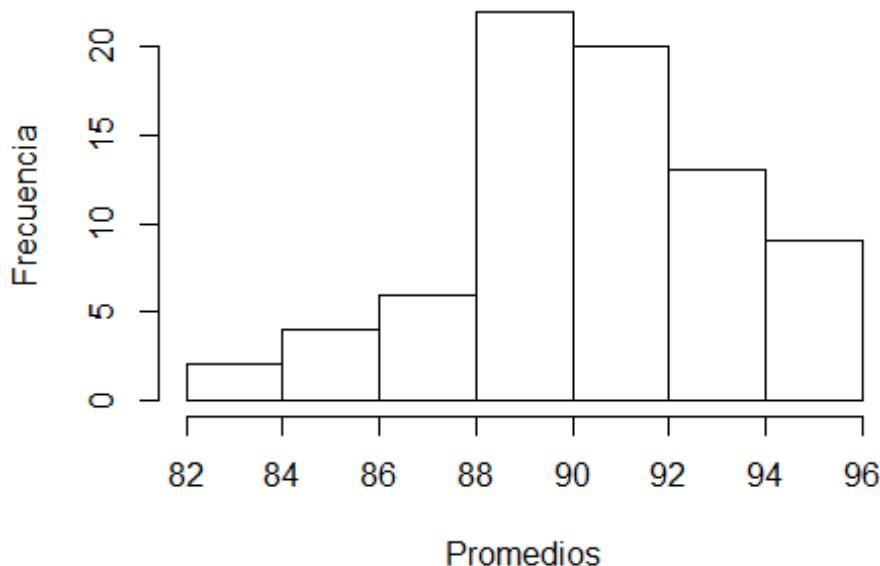
```
hist(industrial, breaks = "Scott", freq = TRUE, main = "Promedios de los
alumnos INDUSTRIAL del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos INDUSTRIAL del ITD. Sc



```
hist(industrial, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos INDUSTRIAL del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos INDUSTRIAL del ITD. F



6.2.6.2 Tablas de Frecuencias de INDUSTRIAL Sturges

```
clases <- nclass.Sturges(industrial)
clases
## [1] 8

tabla.intervalos <- transform(table(cut(industrial, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1 Freq
## 1 (82.5,84.2]    2
## 2 (84.2,85.8]    3
## 3 (85.8,87.5]    4
## 4 (87.5,89.2]   14
## 5 (89.2,90.8]   18
## 6 (90.8,92.5]   18
## 7 (92.5,94.2]    8
## 8 (94.2,95.8]    9

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (82.5,84.2]    2 0.02631579  2.631579
## 2 (84.2,85.8]    3 0.03947368  3.947368
## 3 (85.8,87.5]    4 0.05263158  5.263158
## 4 (87.5,89.2]   14 0.18421053 18.421053
## 5 (89.2,90.8]   18 0.23684211 23.684211
## 6 (90.8,92.5]   18 0.23684211 23.684211
## 7 (92.5,94.2]    8 0.10526316 10.526316
## 8 (94.2,95.8]    9 0.11842105 11.842105

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones pr?cticas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRER
A" = "INDUSTRIAL"))

```

6.2.6.3 Gráficas de pastel. INDUSTRIAL Sturges

```

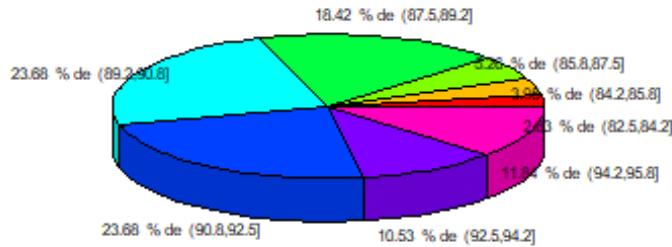
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 76 Observaciones



6.2.6.4 Tablas de Frecuencias de INDUSTRIAL Scott

```
clases <- nclass.scott(industrial)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(industrial, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (82.5,84.7]    3
## 2 (84.7,86.9]    5
## 3 (86.9,89.2]   15
## 4 (89.2,91.4]   22
## 5 (91.4,93.6]   17
## 6 (93.6,95.8]   14

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)
```

```
tabla.intervalos

##           Var1 Freq  Freq.Rel Freq.Porc
## 1 (82.5,84.7]    3 0.03947368  3.947368
## 2 (84.7,86.9]    5 0.06578947  6.578947
## 3 (86.9,89.2]   15 0.19736842 19.736842
## 4 (89.2,91.4]   22 0.28947368 28.947368
## 5 (91.4,93.6]   17 0.22368421 22.368421
## 6 (93.6,95.8]   14 0.18421053 18.421053
```

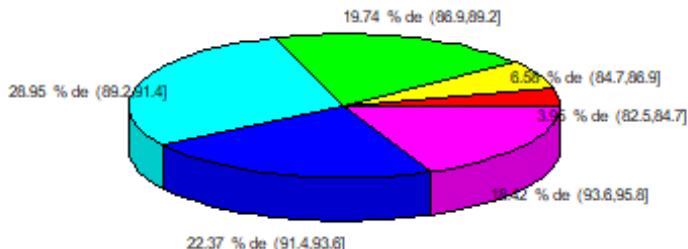
6.2.6.5 Gráficas de pastel. INDUSTRIAL Scott

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
nervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 76 Observaciones



6.2.6.6 Tablas de Frecuencias de INDUSTRIAL FD

```
clases <- nclass.FD(industrial)
clases
```

```

## [1] 9

tabla.intervalos <- transform(table(cut(industrial, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1  (82.5,84]    2
## 2  (84,85.5]   3
## 3 (85.5,86.9]   3
## 4 (86.9,88.4]  10
## 5 (88.4,89.9]  14
## 6 (89.9,91.4]  13
## 7 (91.4,92.9]  16
## 8 (92.9,94.3]   9
## 9 (94.3,95.8]   6

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##          Var1 Freq  Freq.Rel Freq.Porc
## 1  (82.5,84]    2 0.02631579  2.631579
## 2  (84,85.5]   3 0.03947368  3.947368
## 3 (85.5,86.9]   3 0.03947368  3.947368
## 4 (86.9,88.4]  10 0.13157895 13.157895
## 5 (88.4,89.9]  14 0.18421053 18.421053
## 6 (89.9,91.4]  13 0.17105263 17.105263
## 7 (91.4,92.9]  16 0.21052632 21.052632
## 8 (92.9,94.3]   9 0.11842105 11.842105
## 9 (94.3,95.8]   6 0.07894737  7.894737

```

6.2.6.7 Gráficas de pastel. INDUSTRIAL FD

```

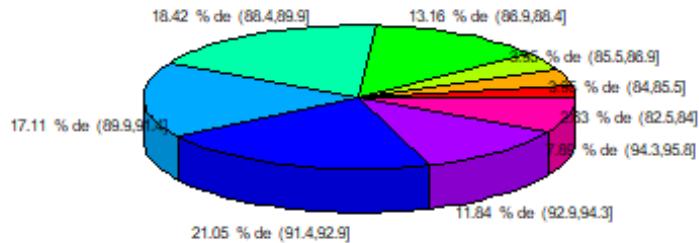
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 76 Observaciones

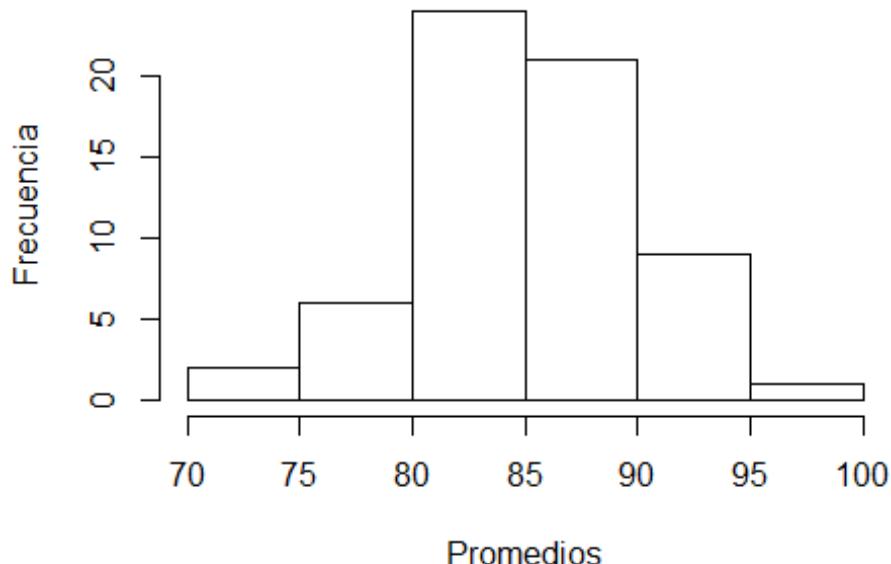


6.2.7 MECANICA

6.2.7.1 Histogramas de MECANICA Sturges, Scott y FD

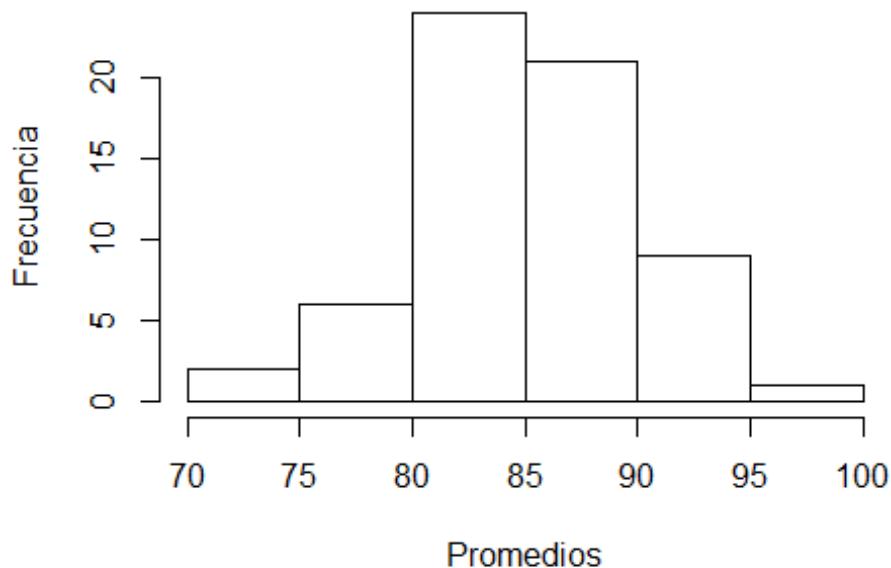
```
mecanica <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2  
& datos$carrera == 'MECANICA')]  
  
mecanica  
  
## [1] 87.67 86.83 84.83 81.67 94.17 93.17 78.00 94.00 88.67 81.40 81.00  
## [12] 84.17 86.67 85.67 82.20 81.67 95.50 81.33 80.60 81.67 78.33 79.25  
## [23] 91.50 83.80 85.17 84.75 84.00 90.50 91.33 79.33 85.00 82.50 86.50  
## [34] 78.67 93.00 81.00 89.17 81.00 89.83 92.83 87.00 85.75 88.00 86.67  
## [45] 87.83 86.33 73.25 88.83 82.67 81.38 78.00 74.20 80.25 92.33 89.50  
## [56] 88.67 83.40 86.50 82.00 89.50 83.83 84.67 85.80  
  
n <- length(mecanica)  
  
n  
## [1] 63  
  
hist(mecanica, breaks = "Sturges", freq = TRUE, main = "Promedios de los  
alumnos MECANICA del ITD. Sturges", xlab = "Promedios", ylab = "Frecuenci  
a")
```

Promedios de los alumnos MECANICA del ITD. Stur



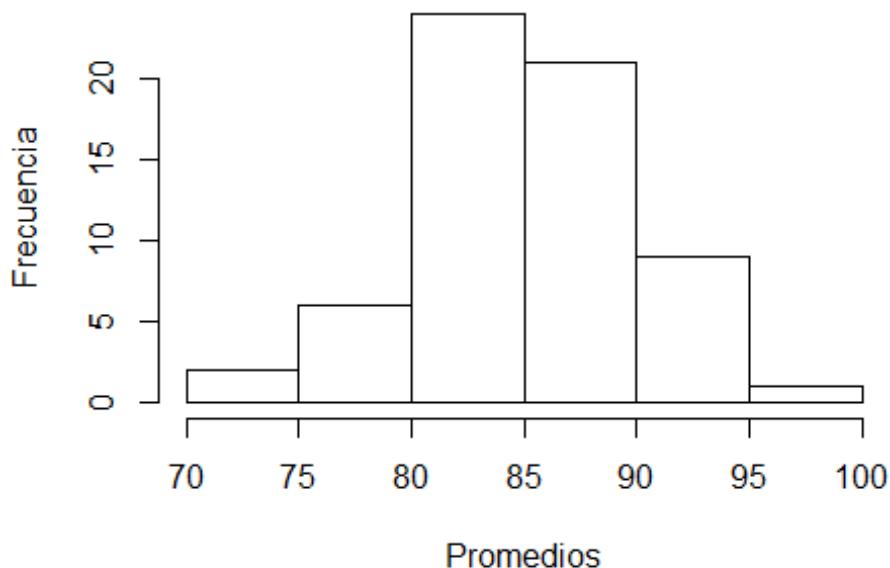
```
hist(mecanica, breaks = "Scott", freq = TRUE, main = "Promedios de los al  
umnos MECANICA del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos MECANICA del ITD. Sc



```
hist(mecanica, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos MECANICA del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos MECANICA del ITD. FI



6.2.7.2 Tablas de Frecuencias de MECANICA Sturges

```
clases <- nclass.Sturges(mecanica)
clases
## [1] 7

tabla.intervalos <- transform(table(cut(mecanica, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1 Freq
## 1 (73.2,76.4]    2
## 2 (76.4,79.6]    6
## 3 (79.6,82.8]   15
## 4 (82.8,86]     13
## 5 (86,89.1]     13
## 6 (89.1,92.3]    7
## 7 (92.3,95.5]    7

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##      Var1 Freq Freq.Rel Freq.Porc
## 1 (73.2,76.4]    2 0.03174603  3.174603
## 2 (76.4,79.6]    6 0.09523810  9.523810
## 3 (79.6,82.8]   15 0.23809524 23.809524
## 4 (82.8,86]    13 0.20634921 20.634921
## 5 (86,89.1]    13 0.20634921 20.634921
## 6 (89.1,92.3]    7 0.11111111 11.111111
## 7 (92.3,95.5]    7 0.11111111 11.111111

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones practicas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRER
A" = "MECANICA"))

```

6.2.7.3 Gráficas de pastel. MECANICA Sturges

```

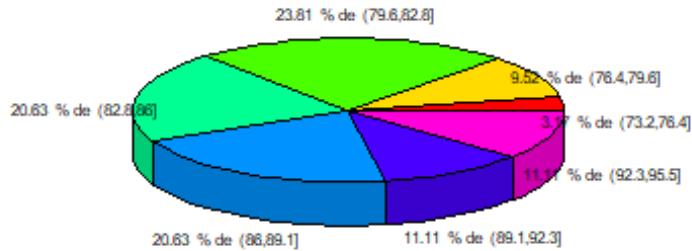
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 63 Observaciones



6.2.7.4 Tablas de Frecuencias de MECANICA Scott

```
clases <- nclass.scott(mecanica)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(mecanica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1  (73.2,77]    2
## 2  (77,80.7]   8
## 3 (80.7,84.4]  18
## 4 (84.4,88.1]  18
## 5 (88.1,91.8]  10
## 6 (91.8,95.5]    7

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)
```

```

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (73.2,77]    2 0.03174603  3.174603
## 2 (77,80.7]    8 0.12698413 12.698413
## 3 (80.7,84.4]  18 0.28571429 28.571429
## 4 (84.4,88.1]  18 0.28571429 28.571429
## 5 (88.1,91.8]  10 0.15873016 15.873016
## 6 (91.8,95.5]   7 0.11111111 11.111111

```

6.2.7.5 Gráficas de pastel. MECANICA Scott

```

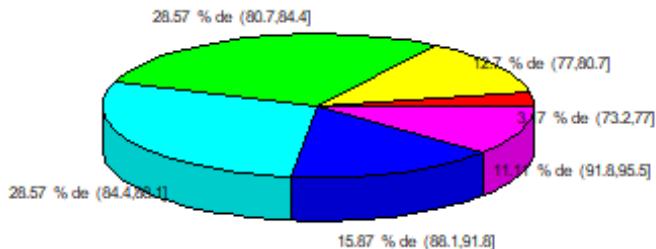
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
nervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 63 Observaciones



6.2.7.6 Tablas de Frecuencias de MECANICA FD

```

clases <- nclass.FD(mecanica)
clases

```

```

## [1] 7

tabla.intervalos <- transform(table(cut(mecanica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (73.2,76.4]    2
## 2 (76.4,79.6]    6
## 3 (79.6,82.8]   15
## 4 (82.8,86]    13
## 5 (86,89.1]    13
## 6 (89.1,92.3]    7
## 7 (92.3,95.5]    7

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##          Var1 Freq  Freq.Rel Freq.Porc
## 1 (73.2,76.4]    2 0.03174603  3.174603
## 2 (76.4,79.6]    6 0.09523810  9.523810
## 3 (79.6,82.8]   15 0.23809524 23.809524
## 4 (82.8,86]    13 0.20634921 20.634921
## 5 (86,89.1]    13 0.20634921 20.634921
## 6 (89.1,92.3]    7 0.11111111 11.111111
## 7 (92.3,95.5]    7 0.11111111 11.111111

```

6.2.7.7 Gráficas de pastel. MECANICA FD

```

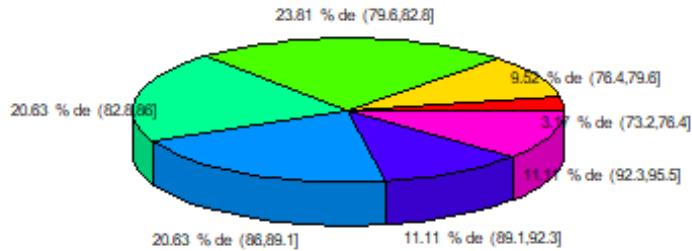
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 63 Observaciones



6.2.8 MECATRONICA

6.2.8.1 Histogramas de MECATRONICA Sturges, Scott y FD

```
mecatronica <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2 & datos$carrera == 'MECATRONICA'))]

mecatronica

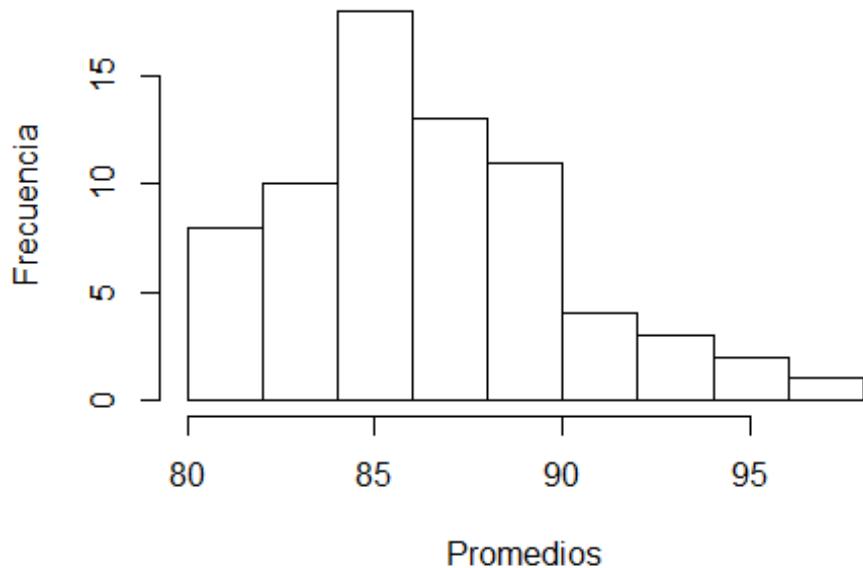
## [1] 88.83 85.67 84.67 85.83 81.60 82.20 83.60 90.00 85.00 88.00 87.33
## [12] 93.17 90.67 81.50 88.00 84.00 86.67 81.00 84.17 88.17 87.33 84.83
## [23] 86.17 85.80 82.17 84.00 84.67 88.00 89.67 89.50 86.00 84.80 89.67
## [34] 83.50 80.25 94.67 89.50 83.40 91.17 86.00 85.00 97.33 84.83 81.60
## [45] 89.33 81.00 83.50 88.17 86.50 86.75 88.33 92.00 81.50 89.33 84.17
## [56] 84.83 80.00 87.33 86.00 87.17 84.33 92.17 83.67 84.33 86.33 92.83
## [67] 94.60 87.00 90.50 82.80

n <- length(mecatronica)

n
## [1] 70

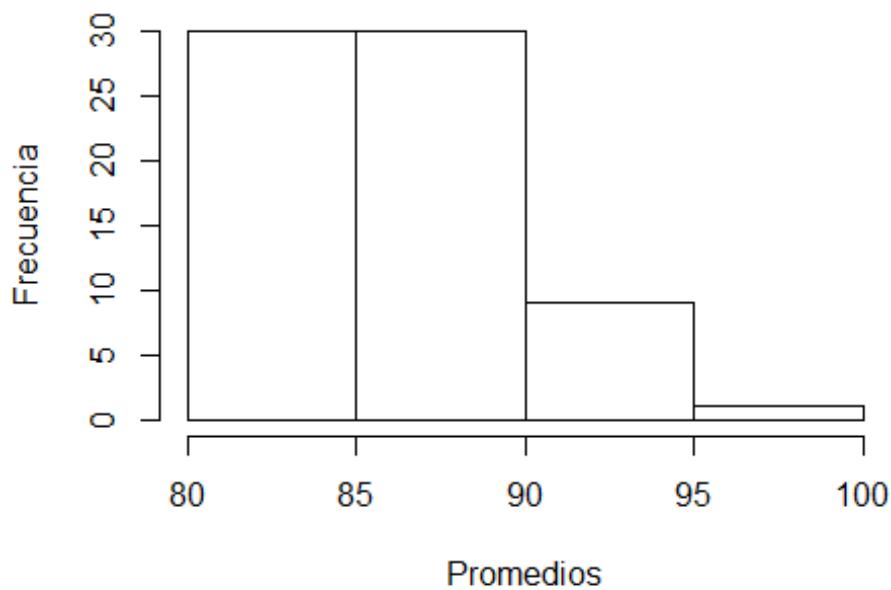
hist(mecatronica, breaks = "Sturges", freq = TRUE, main = "Promedios de los alumnos MECATRONICA del ITD. Sturges", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos MECATRONICA del ITD. St



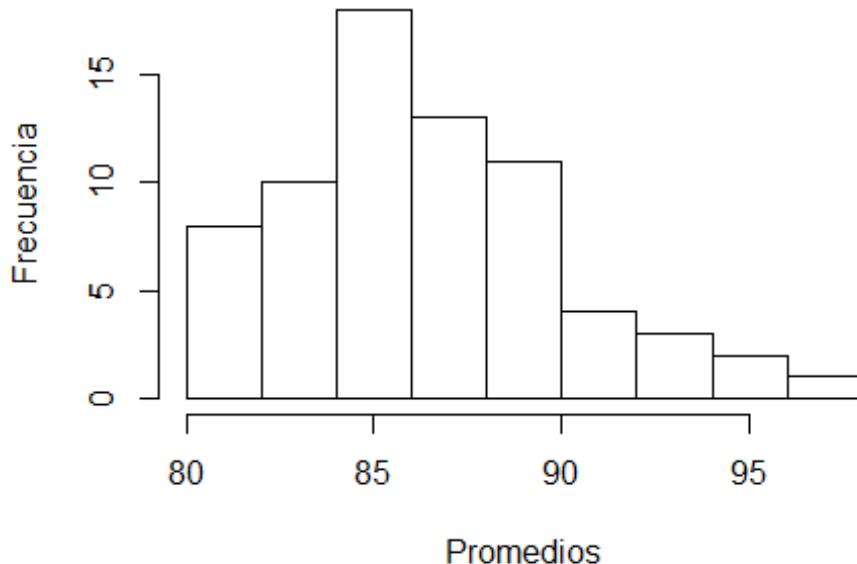
```
hist(mecatronica, breaks = "Scott", freq = TRUE, main = "Promedios de los
alumnos MECATRONICA del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos MECATRONICA del ITD. S



```
hist(mecatronica, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos MECATRONICA del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos MECATRONICA del ITD.



6.2.8.2 Tablas de Frecuencias de MECATRONICA Sturges

```
clases <- nclass.Sturges(mecatronica)
clases
## [1] 8

tabla.intervalos <- transform(table(cut(mecatronica, breaks = clases)))

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (80,82.2]    8
## 2 (82.2,84.3]   14
## 3 (84.3,86.5]   16
## 4 (86.5,88.7]   14
## 5 (88.7,90.8]   10
## 6 (90.8,93]     4
## 7 (93,95.2]     3
## 8 (95.2,97.3]    1

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (80,82.2]    8 0.11428571 11.428571
## 2 (82.2,84.3] 14 0.20000000 20.000000
## 3 (84.3,86.5] 16 0.22857143 22.857143
## 4 (86.5,88.7] 14 0.20000000 20.000000
## 5 (88.7,90.8] 10 0.14285714 14.285714
## 6 (90.8,93]    4 0.05714286  5.714286
## 7 (93,95.2]    3 0.04285714  4.285714
## 8 (95.2,97.3]  1 0.01428571  1.428571

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones pr?cticas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRER
A" = "MECATRONICA"))

```

6.2.8.3 Gráficas de pastel. MECATRONICA Sturges

```

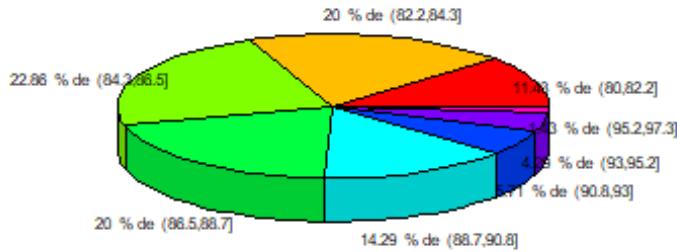
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 70 Observaciones



6.2.8.4 Tablas de Frecuencias de MECATRONICA Scott

```
clases <- nclass.scott(mecatronica)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(mecatronica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1  (80,82.9]    11
## 2  (82.9,85.8]   20
## 3  (85.8,88.7]   21
## 4  (88.7,91.6]   11
## 5  (91.6,94.4]    4
## 6  (94.4,97.3]    3

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)
```

```
tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (80,82.9]    11 0.15714286 15.714286
## 2 (82.9,85.8]   20 0.28571429 28.571429
## 3 (85.8,88.7]   21 0.30000000 30.000000
## 4 (88.7,91.6]   11 0.15714286 15.714286
## 5 (91.6,94.4]    4 0.05714286  5.714286
## 6 (94.4,97.3]    3 0.04285714  4.285714
```

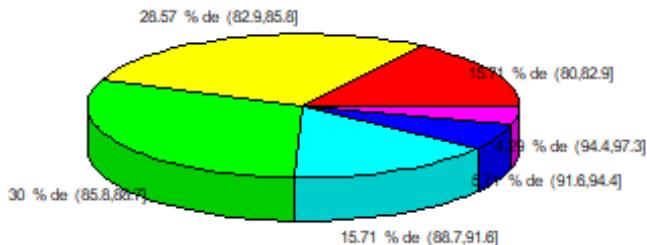
6.2.8.5 Gráficas de pastel. MECATRONICA Scott

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
nervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 70 Observaciones



6.2.8.6 Tablas de Frecuencias de MECATRONICA FD

```
clases <- nclass.FD(mecatronica)
clases
```

```

## [1] 8

tabla.intervalos <- transform(table(cut(mecatronica, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1 Freq
## 1 (80,82.2]    8
## 2 (82.2,84.3] 14
## 3 (84.3,86.5] 16
## 4 (86.5,88.7] 14
## 5 (88.7,90.8] 10
## 6 (90.8,93]    4
## 7 (93,95.2]    3
## 8 (95.2,97.3]   1

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq  Freq.Rel Freq.Porc
## 1 (80,82.2]    8 0.11428571 11.428571
## 2 (82.2,84.3] 14 0.20000000 20.000000
## 3 (84.3,86.5] 16 0.22857143 22.857143
## 4 (86.5,88.7] 14 0.20000000 20.000000
## 5 (88.7,90.8] 10 0.14285714 14.285714
## 6 (90.8,93]    4 0.05714286  5.714286
## 7 (93,95.2]    3 0.04285714  4.285714
## 8 (95.2,97.3]   1 0.01428571  1.428571

```

6.2.8.7 Gráficas de pastel. MECATRONICA FD

```

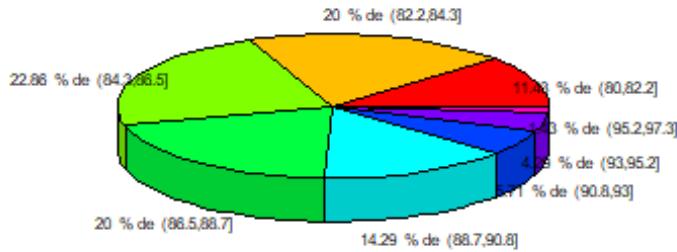
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 70 Observaciones



6.2.9 QUIMICA

6.2.9.1 Histogramas de QUIMICA Sturges, Scott y FD

```
quimica <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2 &
datos$carrera == 'QUIMICA')]

quimica

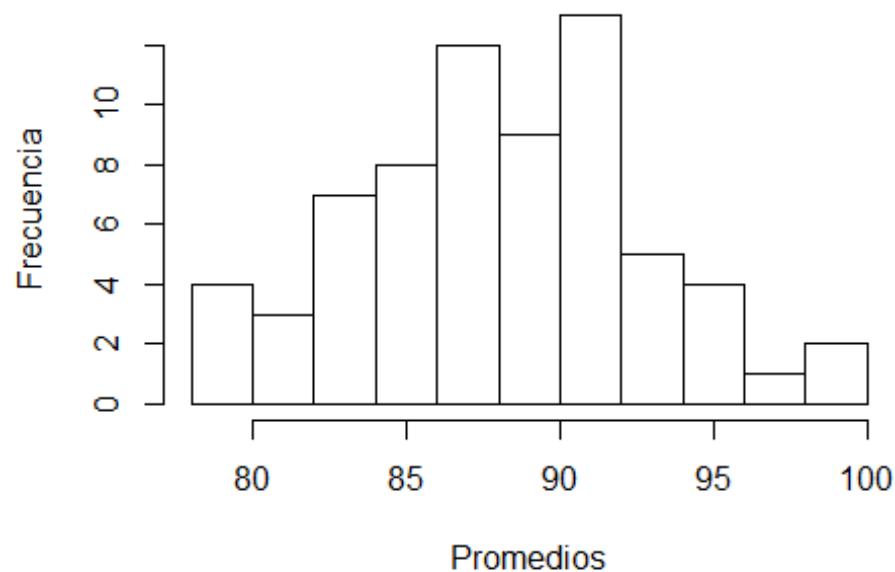
## [1] 89.83 92.00 82.40 85.17 83.00 87.50 86.20 89.67 86.50 82.50 95.17
## [12] 90.67 85.67 91.80 98.17 90.33 79.83 89.00 98.33 90.67 86.83 91.25
## [23] 91.83 83.33 81.17 81.33 82.33 95.50 87.17 91.33 95.50 88.33 88.67
## [34] 97.17 90.17 78.00 86.67 90.60 88.00 85.60 87.83 85.17 87.20 86.33
## [45] 93.33 86.00 81.50 93.00 86.20 84.83 85.17 85.80 80.00 83.60 93.00
## [56] 89.50 88.33 90.83 94.00 93.17 89.00 78.67 83.80 88.00 91.17 91.00
## [67] 95.33 89.83

n <- length(quimica)

n
## [1] 68

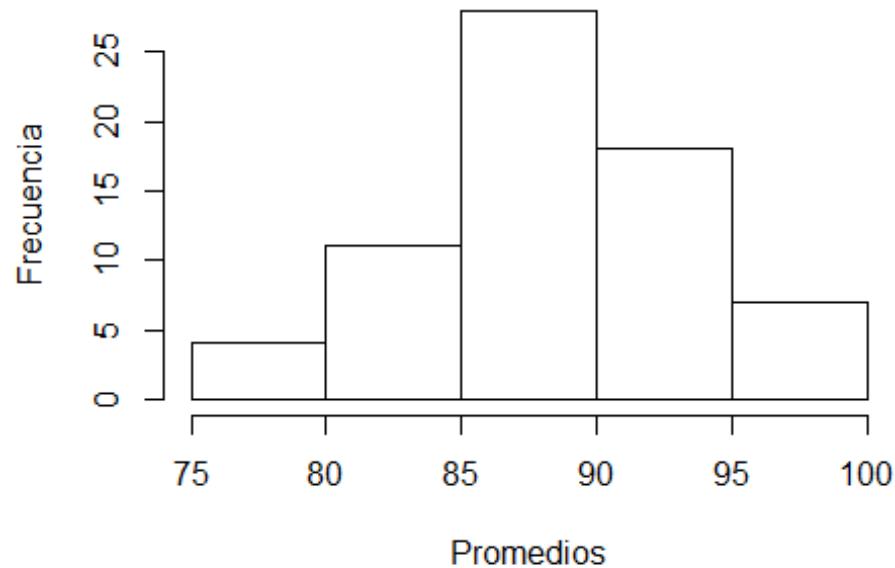
hist(quimica, breaks = "Sturges", freq = TRUE, main = "Promedios de los alumnos QUIMICA del ITD. Sturges", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos QUIMICA del ITD. Sturg



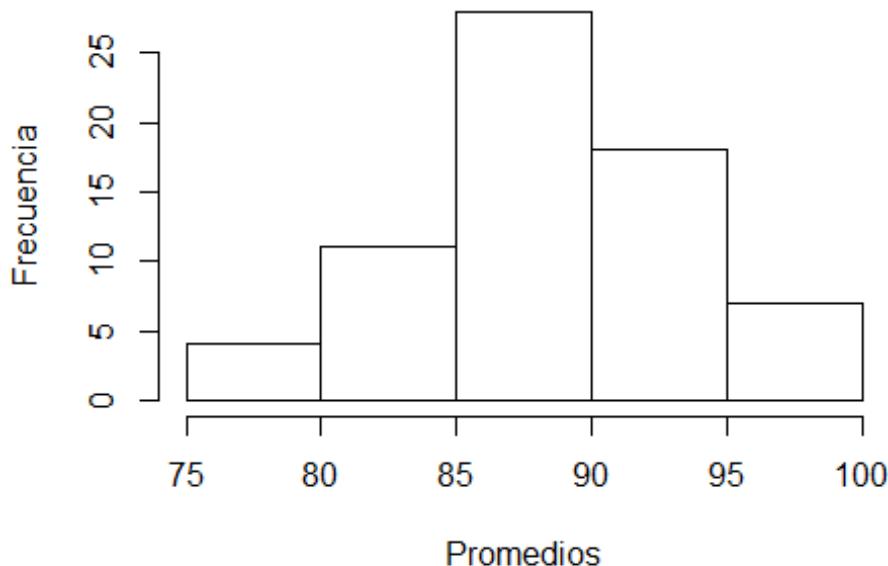
```
hist(quimica, breaks = "Scott", freq = TRUE, main = "Promedios de los alumnos QUIMICA del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos QUIMICA del ITD. Scott



```
hist(quimica, breaks = "FD", freq = TRUE, main = "Promedios de los alumno s QUIMICA del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos QUIMICA del ITD. FD



6.2.9.2 Tablas de Frecuencias de QUIMICA Sturges

```
clases <- nclass.Sturges(quimica)
clases

## [1] 8

tabla.intervalos <- transform(table(cut(quimica, breaks = clases)))

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (78,80.5]    4
## 2 (80.5,83.1]   7
## 3 (83.1,85.6]   8
## 4 (85.6,88.2]  15
## 5 (88.2,90.7]  14
## 6 (90.7,93.2]  11
## 7 (93.2,95.8]   6
## 8 (95.8,98.4]   3

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuenci a Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (78,80.5]    4 0.05882353  5.882353
## 2 (80.5,83.1]   7 0.10294118 10.294118
## 3 (83.1,85.6]   8 0.11764706 11.764706
## 4 (85.6,88.2]  15 0.22058824 22.058824
## 5 (88.2,90.7]  14 0.20588235 20.588235
## 6 (90.7,93.2]  11 0.16176471 16.176471
## 7 (93.2,95.8]   6 0.08823529  8.823529
## 8 (95.8,98.4]   3 0.04411765  4.411765

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones pr?cticas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRER
A" = "QUIMICA"))

```

6.2.9.3 Gráficas de pastel. QUIMICA Sturges

```

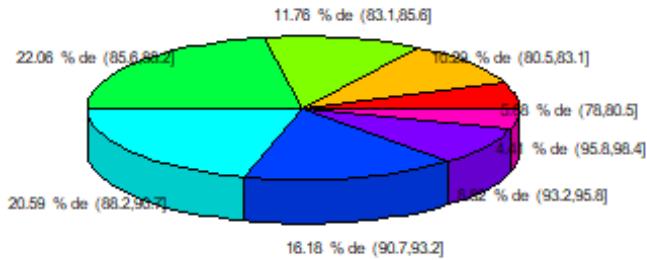
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 68 Observaciones



6.2.9.4 Tablas de Frecuencias de QUIMICA Scott

```
clases <- nclass.scott(quimica)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(quimica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1  (78,81.4]    6
## 2 (81.4,84.8]    8
## 3 (84.8,88.2]   20
## 4 (88.2,91.6]   19
## 5 (91.6,94.9]    8
## 6 (94.9,98.4]    7

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)
```

```
tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (78,81.4]    6 0.08823529  8.823529
## 2 (81.4,84.8]   8 0.11764706 11.764706
## 3 (84.8,88.2]  20 0.29411765 29.411765
## 4 (88.2,91.6]  19 0.27941176 27.941176
## 5 (91.6,94.9]   8 0.11764706 11.764706
## 6 (94.9,98.4]   7 0.10294118 10.294118
```

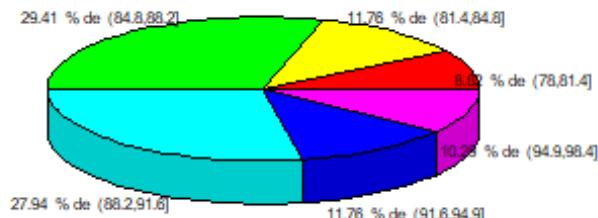
6.2.9.5 Gráficas de pastel. QUIMICA Scott

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
nervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 68 Observaciones



6.2.9.6 Tablas de Frecuencias de QUIMICA FD

```
clases <- nclass.FD(quimica)
clases
```

```

## [1] 7

tabla.intervalos <- transform(table(cut(quimica, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1 Freq
## 1 (78,80.9]    4
## 2 (80.9,83.8] 10
## 3 (83.8,86.7] 13
## 4 (86.7,89.6] 13
## 5 (89.6,92.5] 16
## 6 (92.5,95.4]  7
## 7 (95.4,98.4]  5

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos
```

Var1	Freq	Freq.Rel	Freq.Porc
(78,80.9]	4	0.05882353	5.882353
(80.9,83.8]	10	0.14705882	14.705882
(83.8,86.7]	13	0.19117647	19.117647
(86.7,89.6]	13	0.19117647	19.117647
(89.6,92.5]	16	0.23529412	23.529412
(92.5,95.4]	7	0.10294118	10.294118
(95.4,98.4]	5	0.07352941	7.352941

6.2.9.7 Gráficas de pastel. QUIMICA FD

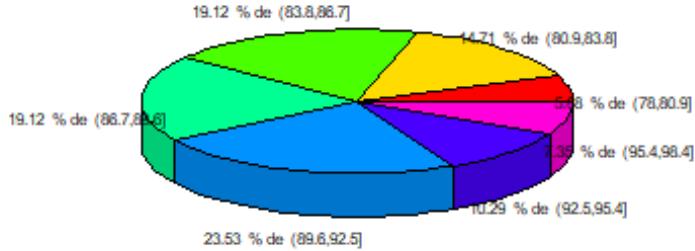
```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 68 Observaciones



6.2.10 SISTEMAS

6.2.10.1 Histogramas de SISTEMAS Sturges, Scott y FD

```
sistemas <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2
& datos$carrera == 'SISTEMAS')]

quimica

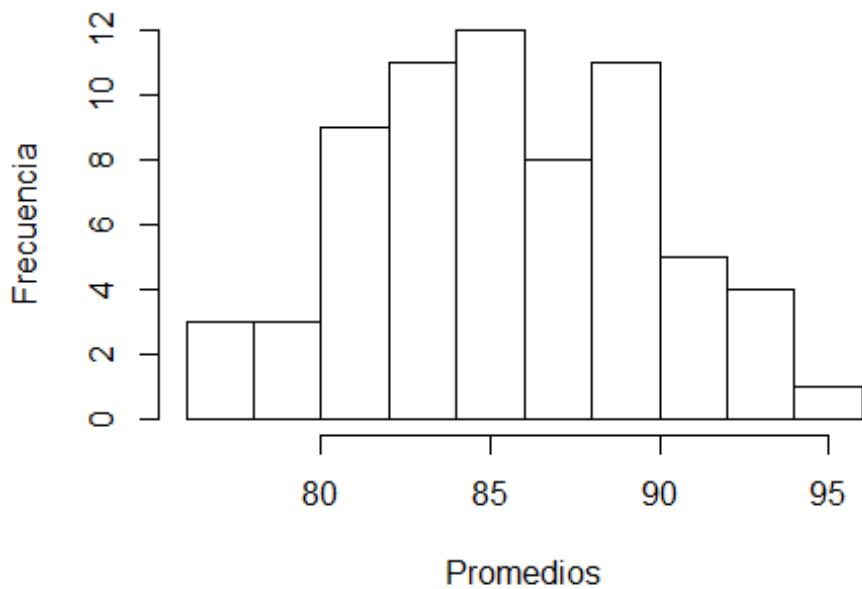
## [1] 89.83 92.00 82.40 85.17 83.00 87.50 86.20 89.67 86.50 82.50 95.17
## [12] 90.67 85.67 91.80 98.17 90.33 79.83 89.00 98.33 90.67 86.83 91.25
## [23] 91.83 83.33 81.17 81.33 82.33 95.50 87.17 91.33 95.50 88.33 88.67
## [34] 97.17 90.17 78.00 86.67 90.60 88.00 85.60 87.83 85.17 87.20 86.33
## [45] 93.33 86.00 81.50 93.00 86.20 84.83 85.17 85.80 80.00 83.60 93.00
## [56] 89.50 88.33 90.83 94.00 93.17 89.00 78.67 83.80 88.00 91.17 91.00
## [67] 95.33 89.83

n <- length(sistemas)

n
## [1] 67

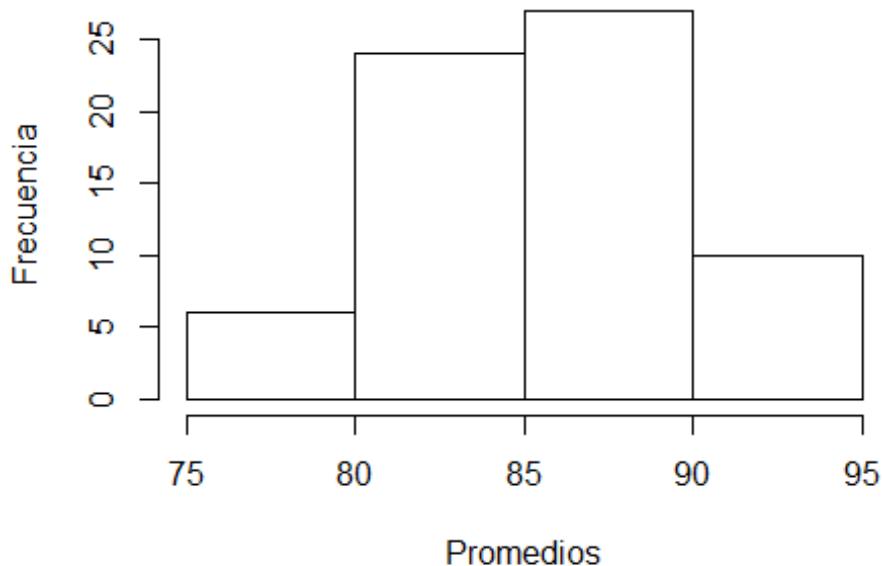
hist(sistemas, breaks = "Sturges", freq = TRUE, main = "Promedios de los
alumnos SISTEMAS del ITD. Sturges", xlab = "Promedios", ylab = "Frecuenci
a")
```

Promedios de los alumnos SISTEMAS del ITD. Sturges



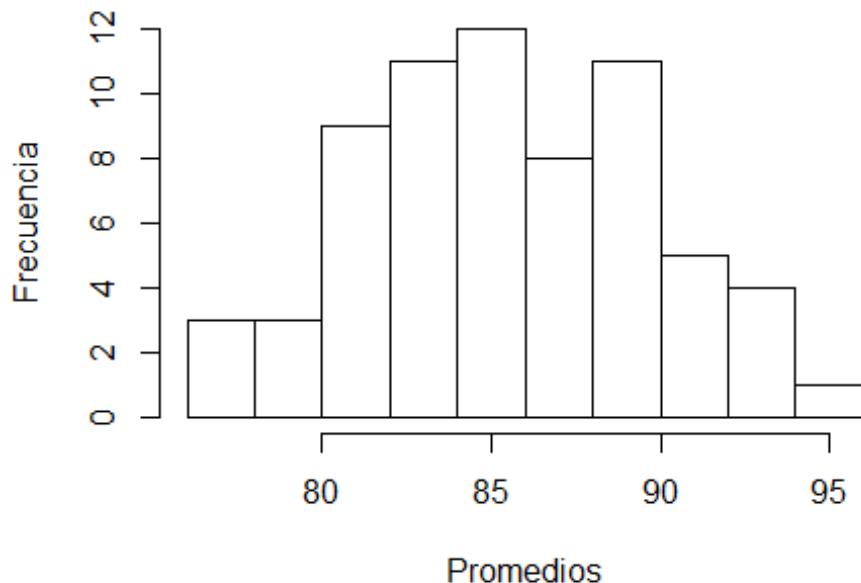
```
hist(sistemas, breaks = "Sturges", freq = TRUE, main = "Promedios de los alumnos SISTEMAS del ITD. Sturges", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos SISTEMAS del ITD. Scott



```
hist(sistemas, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos SISTEMAS del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos SISTEMAS del ITD. FD



6.2.10.2 Tablas de Frecuencias de SISTEMAS Sturges

```
clases <- nclass.Sturges(sistemas)
clases
## [1] 8

tabla.intervalos <- transform(table(cut(sistemas, breaks = clases)))

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (76.7,78.9]    3
## 2 (78.9,81.2]    8
## 3 (81.2,83.5]   14
## 4 (83.5,85.8]   11
## 5 (85.8,88]     10
## 6 (88,90.3]     11
## 7 (90.3,92.6]    7
## 8 (92.6,94.8]    3

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)
```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (76.7,78.9]    3 0.04477612  4.477612
## 2 (78.9,81.2]    8 0.11940299 11.940299
## 3 (81.2,83.5]   14 0.20895522 20.895522
## 4 (83.5,85.8]   11 0.16417910 16.417910
## 5 (85.8,88]     10 0.14925373 14.925373
## 6 (88,90.3]      11 0.16417910 16.417910
## 7 (90.3,92.6]     7 0.10447761 10.447761
## 8 (92.6,94.8]     3 0.04477612  4.477612

```

6.2.10.3 Gráficas de pastel. SISTEMAS Sturges

```

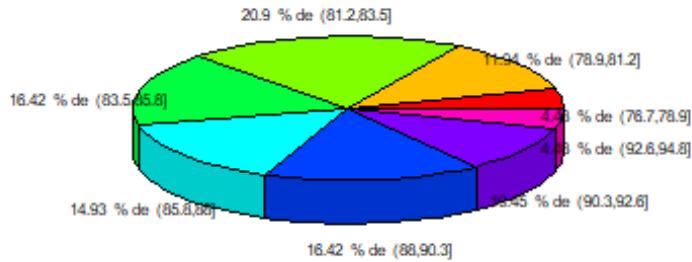
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 67 Observaciones



6.2.10.4 Tablas de Frecuencias de SISTEMAS Scott

```
clases <- nclass.scott(sistemas)
clases

## [1] 5

tabla.intervalos <- transform(table(cut(sistemas, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (76.7,80.3]    6
## 2 (80.3,83.9]   20
## 3 (83.9,87.6]   19
## 4 (87.6,91.2]   14
## 5 (91.2,94.8]    8

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos
```

```

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (76.7,80.3]    6 0.08955224  8.955224
## 2 (80.3,83.9]   20 0.29850746 29.850746
## 3 (83.9,87.6]   19 0.28358209 28.358209
## 4 (87.6,91.2]   14 0.20895522 20.895522
## 5 (91.2,94.8]    8 0.11940299 11.940299

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame  

# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo  

data.frame  

# solo usar la regla Struges para cuestiones pr?cticas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRER  

A" = "SISTEMAS"))

```

6.2.10.5 Gráficas de pastel. SISTEMAS Scott

```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i  

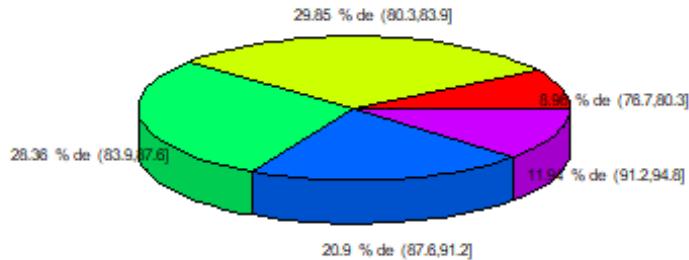
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas  

te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 67 Observaciones



6.2.10.6 Tablas de Frecuencias de SISTEMAS FD

```
clases <- nclass.FD(sistemas)
clases

## [1] 7

tabla.intervalos <- transform(table(cut(sistemas, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (76.7,79.3]    4
## 2 (79.3,81.9]   10
## 3 (81.9,84.5]   12
## 4 (84.5,87]     18
## 5 (87,89.6]     11
## 6 (89.6,92.2]    7
## 7 (92.2,94.8]    5

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)
```

```
tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (76.7,79.3]    4 0.05970149  5.970149
## 2 (79.3,81.9]   10 0.14925373 14.925373
## 3 (81.9,84.5]   12 0.17910448 17.910448
## 4 (84.5,87]     18 0.26865672 26.865672
## 5 (87,89.6]     11 0.16417910 16.417910
## 6 (89.6,92.2]    7 0.10447761 10.447761
## 7 (92.2,94.8]    5 0.07462687  7.462687
```

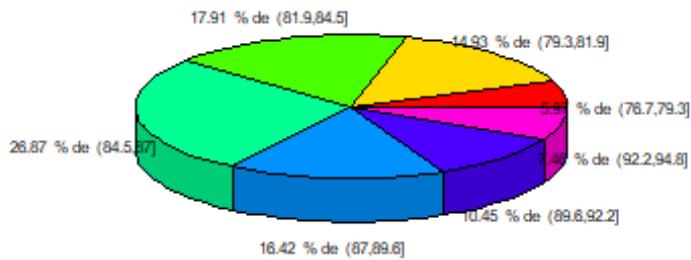
6.2.10.7 Gráficas de pastel. SISTEMAS FD

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 67 Observaciones



6.2.11 TIC

6.2.11.1 Histogramas de TIC Sturges, Scott y FD

```
tic <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2 & datos$carrera == 'TIC')]

tic

## [1] 89.33 92.20 87.40 81.33 84.83 94.00 90.80 86.60 83.00 85.80 91.67
## [12] 71.50 83.80 83.80 82.80 81.80 80.00 89.50 84.40 88.50 89.33 85.00

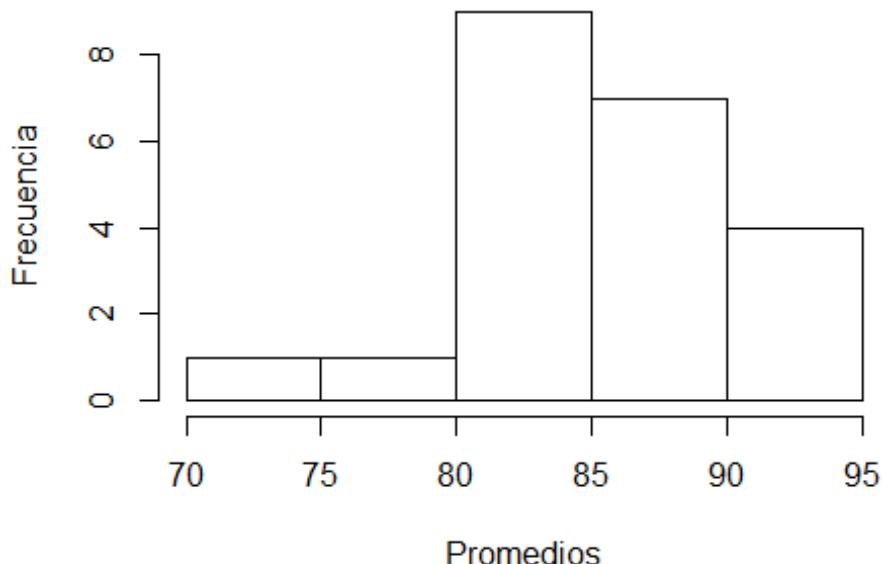
n <- length(tic)

n

## [1] 22

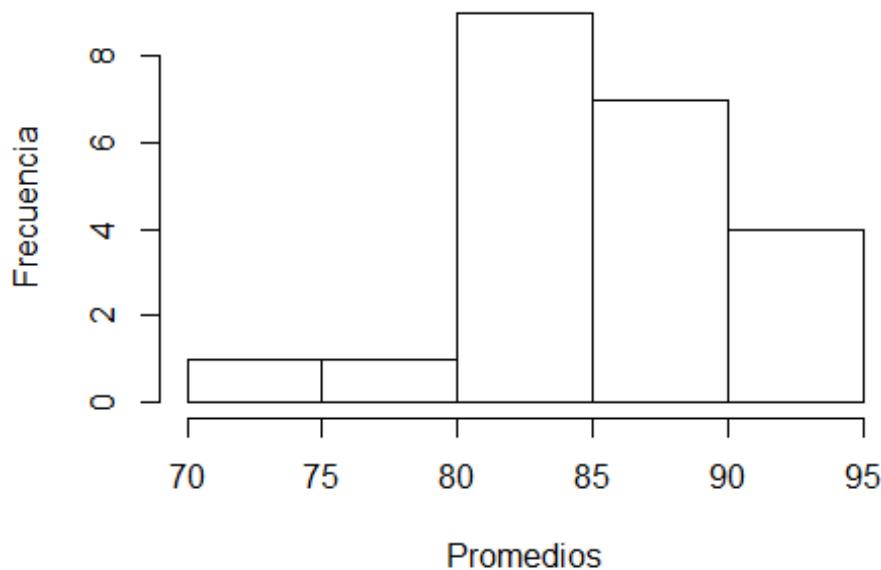
hist(tic, breaks = "Sturges", freq = TRUE, main = "Promedios de los alumnos TIC del ITD. Sturges", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos TIC del ITD. Sturges



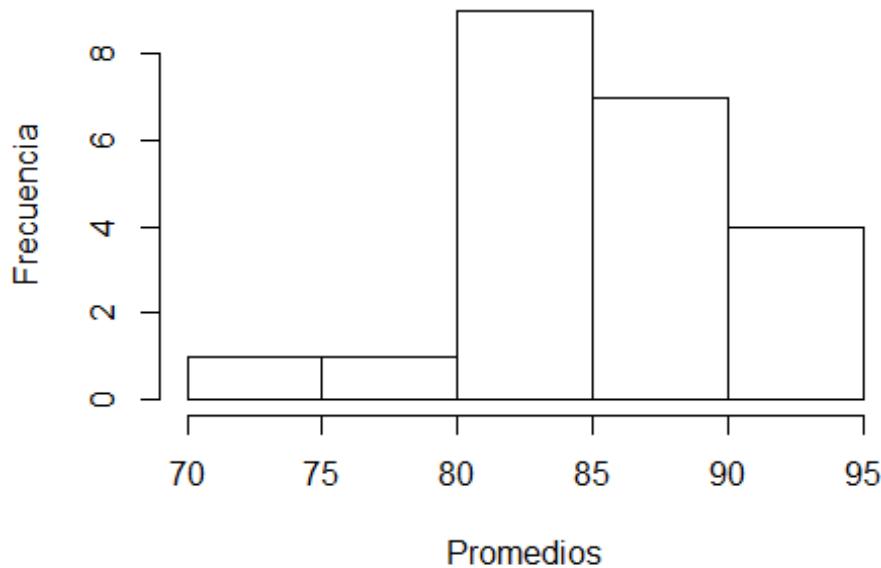
```
hist(tic, breaks = "Scott", freq = TRUE, main = "Promedios de los alumnos TIC del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos TIC del ITD. Scott



```
hist(tic, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos TI  
C del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos TIC del ITD. FD



6.2.11.2 Tablas de Frecuencias de TIC Sturges

```
clases <- nclass.Sturges(tic)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(tic, breaks = clases)))

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (71.5,75.2]    1
## 2 (75.2,79]     0
## 3 (79,82.8]     3
## 4 (82.8,86.5]   8
## 5 (86.5,90.2]   6
## 6 (90.2,94]     4

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##      Var1 Freq  Freq.Rel Freq.Porc
## 1 (71.5,75.2]    1 0.04545455  4.545455
## 2 (75.2,79]     0 0.00000000  0.000000
## 3 (79,82.8]     3 0.13636364 13.636364
## 4 (82.8,86.5]   8 0.36363636 36.363636
## 5 (86.5,90.2]   6 0.27272727 27.272727
## 6 (90.2,94]     4 0.18181818 18.181818

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo
# data.frame
# solo usar la regla Sturges para cuestiones practicas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRERA" = "TIC"))
```

6.2.11.3 Gráficas de pastel. TIC Sturges

```
# install.packages("plotrix")
library(plotrix)
```

```

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

# pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = p
aste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5) # 
No puede hacer pastel porque trae 0 la frecuencia. Pendiente

```

6.2.11.4 Tablas de Frecuencias de TIC Scott

```

clases <- nclass.scott(tic)
clases

## [1] 4

tabla.intervalos <- transform(table(cut(tic, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1  Freq
## 1 (71.5,77.1]    1
## 2 (77.1,82.8]    3
## 3 (82.8,88.4]   10
## 4 (88.4,94]     8

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##           Var1  Freq  Freq.Rel Freq.Porc
## 1 (71.5,77.1]    1  0.04545455  4.545455
## 2 (77.1,82.8]    3  0.13636364 13.636364
## 3 (82.8,88.4]   10  0.45454545 45.454545
## 4 (88.4,94]     8  0.36363636 36.363636

```

6.2.11.5 Gráficas de pastel. TIC Scott

```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i

```

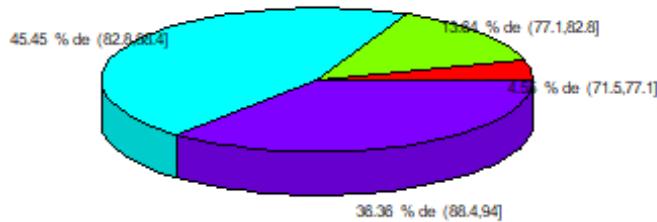
```

intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 22 Observaciones



6.2.11.6 Tablas de Frecuencias de TIC FD

```

clases <- nclass.FD(tic)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(tic, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (71.5,75.2]    1
## 2 (75.2,79]     0
## 3 (79,82.8]     3
## 4 (82.8,86.5]   8
## 5 (86.5,90.2]   6
## 6 (90.2,94]     4

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (71.5,75.2]    1 0.04545455  4.545455
## 2 (75.2,79]     0 0.00000000  0.000000
## 3 (79,82.8]     3 0.13636364 13.636364
## 4 (82.8,86.5]    8 0.36363636 36.363636
## 5 (86.5,90.2]    6 0.27272727 27.272727
## 6 (90.2,94]      4 0.18181818 18.181818

```

6.2.11.7 Gráficas de pastel. TIC FD

```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

# pie3D(round(tabla.intervalos$Freq.Porc,2), Labels = etiquetas, main = p
aste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5) #
No puede hacer pastel porque trae 0 la frecuencia. Pendiente

```

6.2.12 GESTION

6.2.12.1 Histogramas de GESTION Sturges, Scott y FD

```

gestion <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2 &
datos$carrera == 'GESTION')]

gestion

## [1] 77.50 88.00 82.50 93.00 85.80 88.17 89.20 78.67 75.75 82.60 78.33
## [12] 85.00 95.00 82.40 91.83 90.17 83.50 81.80 93.17 92.20 87.50 90.00
## [23] 89.83 82.80 90.33 91.17 80.83 78.83 83.80 90.80 88.17 94.40 85.40
## [34] 85.83 85.83 87.83 94.83 90.80 92.50 80.83 90.00 91.60 87.40 85.80
## [45] 92.00 79.80 91.00 90.80 82.17 78.75 88.50 85.67 92.00 86.80 87.83
## [56] 93.17 78.00 85.57 81.50 87.60 86.00 78.67 92.50 81.50 87.00 73.00
## [67] 83.00 82.67 89.20 80.50

n <- length(gestion)

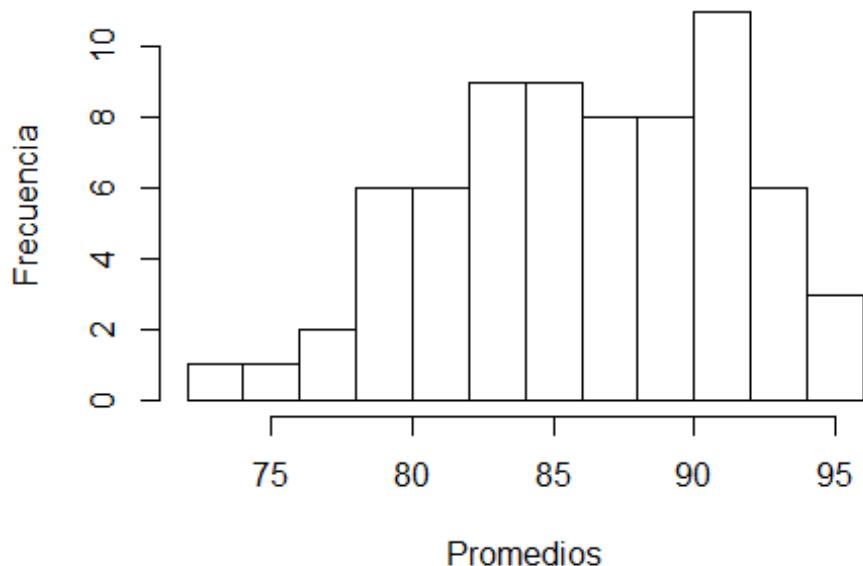
n

## [1] 70

```

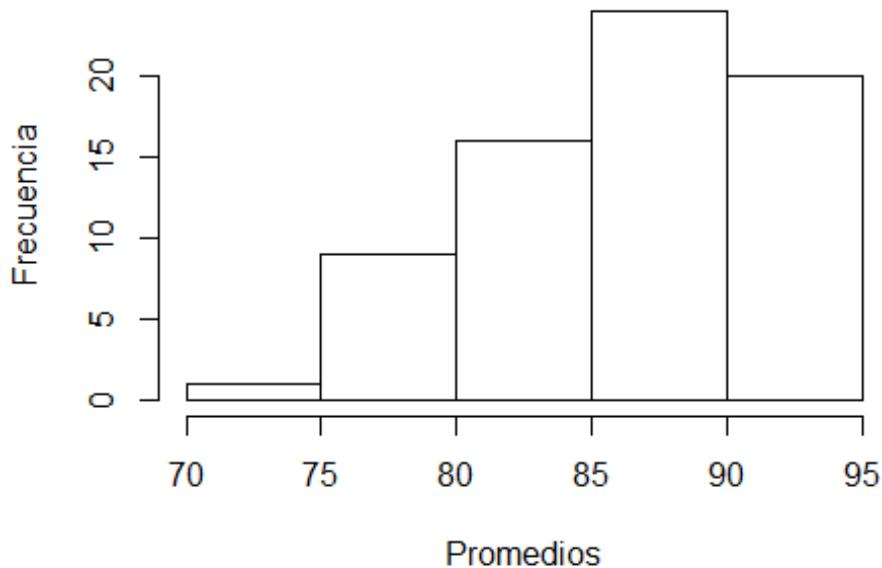
```
hist(gestion, breaks = "Sturges", freq = TRUE, main = "Promedios de los alumnos GESTION del ITD. Sturges", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos GESTION del ITD. Sturg



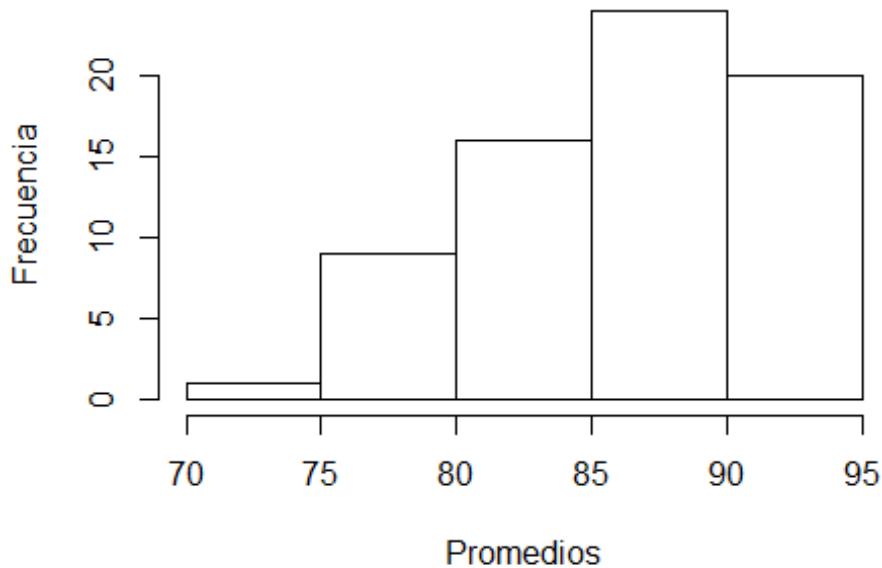
```
hist(gestion, breaks = "Scott", freq = TRUE, main = "Promedios de los alumnos GESTION del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos GESTION del ITD. Sco



```
hist(gestion, breaks = "FD", freq = TRUE, main = "Promedios de los alumnos GESTION del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos GESTION del ITD. FD



6.2.12.2 Tablas de Frecuencias de GESTION Sturges

```
clases <- nclass.Sturges(gestion)
clases

## [1] 8

tabla.intervalos <- transform(table(cut(gestion, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1  Freq
## 1  (73,75.8]     2
## 2 (75.8,78.5]     3
## 3 (78.5,81.2]     8
## 4 (81.2,84]    12
## 5 (84,86.8]     9
## 6 (86.8,89.5]   13
## 7 (89.5,92.2]   15
## 8 (92.2,95]      8

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##          Var1  Freq  Freq.Rel Freq.Porc
## 1  (73,75.8]     2  0.02857143  2.857143
## 2 (75.8,78.5]     3  0.04285714  4.285714
## 3 (78.5,81.2]     8  0.11428571 11.428571
## 4 (81.2,84]    12  0.17142857 17.142857
## 5 (84,86.8]     9  0.12857143 12.857143
## 6 (86.8,89.5]   13  0.18571429 18.571429
## 7 (89.5,92.2]   15  0.21428571 21.428571
## 8 (92.2,95]      8  0.11428571 11.428571

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de cada carrera de un mismo
# data.frame
# solo usar la regla Sturges para cuestiones practicas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRERA" = "GESTION"))
```

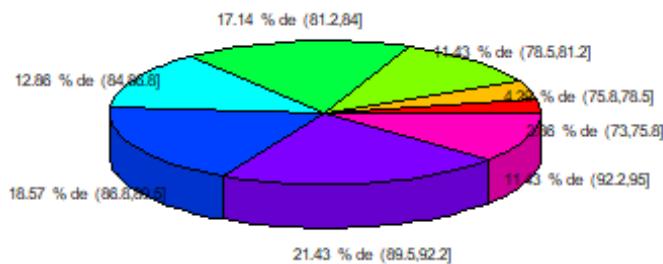
6.2.12.3 Gráficas de pastel. GESTION Sturges

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 70 Observaciones



6.2.12.4 Tablas de Frecuencias de GESTION Scott

```
clases <- nclass.scott(gestion)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(gestion, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1 (73,76.7]    2
## 2 (76.7,80.3]   8
## 3 (80.3,84]   15
```

```

## 4  (84,87.7] 14
## 5  (87.7,91.3] 18
## 6  (91.3,95] 13

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1  (73,76.7]    2 0.02857143  2.857143
## 2 (76.7,80.3]    8 0.11428571 11.428571
## 3  (80.3,84]   15 0.21428571 21.428571
## 4  (84,87.7]   14 0.20000000 20.000000
## 5 (87.7,91.3]   18 0.25714286 25.714286
## 6  (91.3,95]   13 0.18571429 18.571429

```

6.2.12.5 Gráficas de pastel. GESTION Scott

```

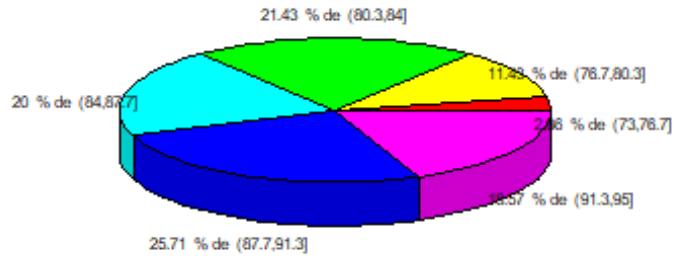
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 70 Observaciones



6.2.12.6 Tablas de Frecuencias de GESTION FD

```
clases <- nclass.FD(gestion)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(gestion, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1  (73,76.7]    2
## 2 (76.7,80.3]    8
## 3  (80.3,84]   15
## 4  (84,87.7]   14
## 5 (87.7,91.3]   18
## 6 (91.3,95]   13

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)
```

```
tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (73,76.7]    2 0.02857143  2.857143
## 2 (76.7,80.3]   8 0.11428571 11.428571
## 3 (80.3,84]   15 0.21428571 21.428571
## 4 (84,87.7]   14 0.20000000 20.000000
## 5 (87.7,91.3]  18 0.25714286 25.714286
## 6 (91.3,95]   13 0.18571429 18.571429
```

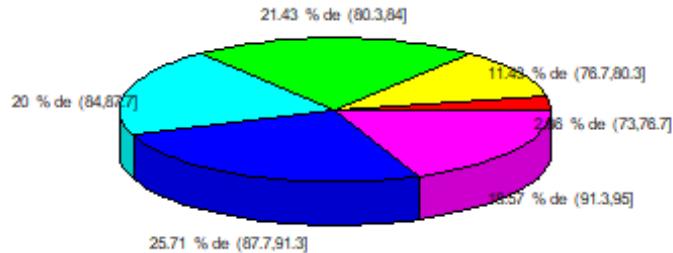
6.2.12.7 Gráficas de pastel. GESTION FD

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
nervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 70 Observaciones



6.2.13 INFORMATICA

6.2.13.1 Histogramas de INFORMATICA Sturges, Scott y FD

```
informatica <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2 & datos$carrera == 'INFORMATICA')]

informatica

## [1] 78.00 81.50 82.00 85.20 86.67 86.60 86.00 84.20 80.33 84.33 87.00
## [12] 84.60 82.00 86.00 87.67 85.67 86.00 85.83 82.50 85.00 86.50 81.25
## [23] 92.67 86.17 85.00 81.83 85.33

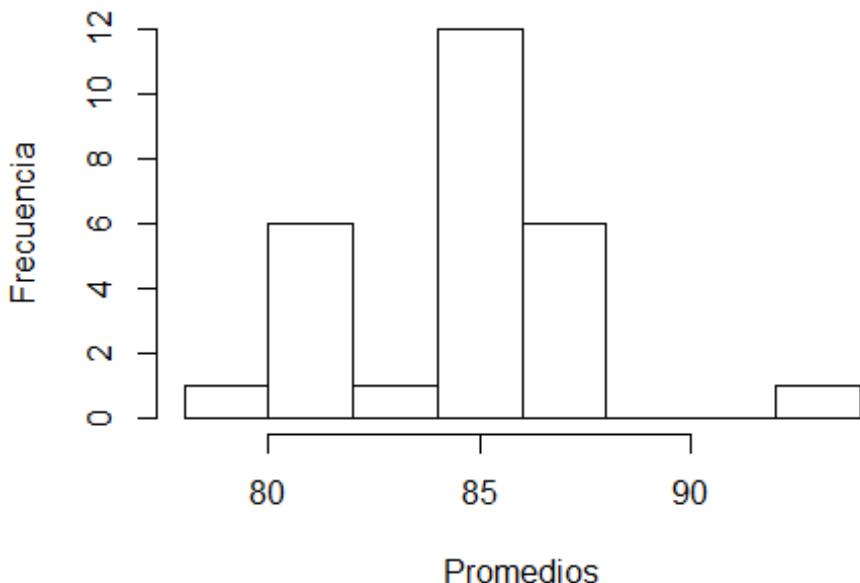
n <- length(informatica)

n

## [1] 27

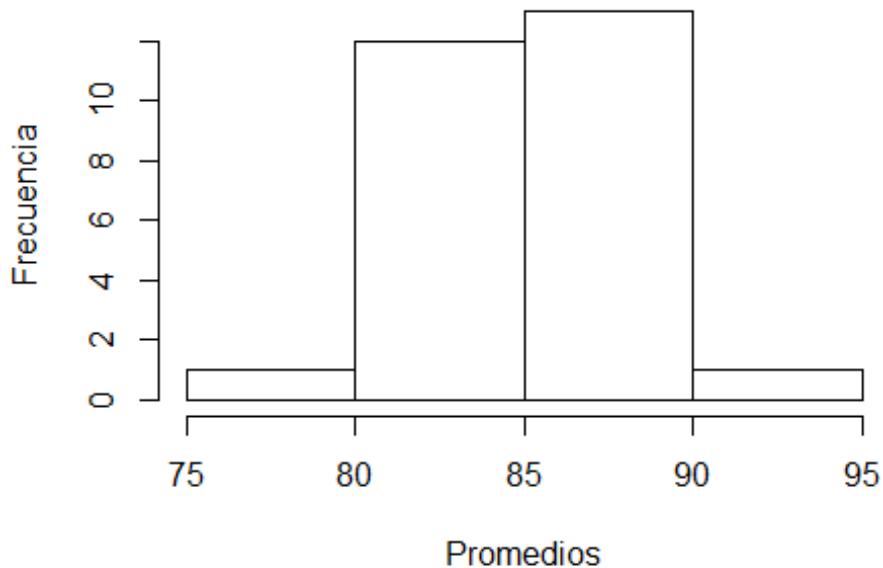
hist(informatica, breaks = "Sturges", freq = TRUE, main = "Promedios de los alumnos INFORMATICA del ITD. Sturges", xlab = "Promedios", ylab = "Frecuencia")
```

'romedios de los alumnos INFORMATICA del ITD. Stu



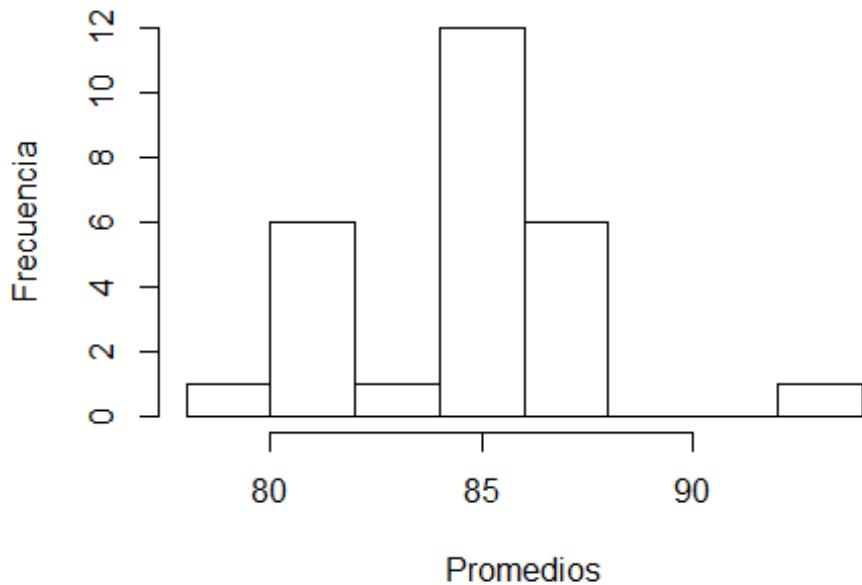
```
hist(informatica, breaks = "Scott", freq = TRUE, main = "Promedios de los alumnos INFORMATICA del ITD. Scott", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos INFORMATICA del ITD. S



```
hist(informatica, breaks = "FD", freq = TRUE, main = "Promedios de los al  
umnos INFORMATICA del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos INFORMATICA del ITD. I



6.2.13.2 Tablas de Frecuencias de INFORMATICA Sturges

```
clases <- nclass.Sturges(informatica)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(informatica, breaks = clases)))

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (78,80.4]    2
## 2 (80.4,82.9]   6
## 3 (82.9,85.3]   7
## 4 (85.3,87.8]  11
## 5 (87.8,90.2]   0
## 6 (90.2,92.7]   1

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##      Var1 Freq  Freq.Rel Freq.Porc
## 1 (78,80.4]    2 0.07407407  7.407407
## 2 (80.4,82.9]   6 0.22222222 22.222222
## 3 (82.9,85.3]   7 0.25925926 25.925926
## 4 (85.3,87.8]  11 0.40740741 40.740741
## 5 (87.8,90.2]   0 0.00000000  0.000000
## 6 (90.2,92.7]   1 0.03703704  3.703704

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de de cada carrera de un mismo
# data.frame
# solo usar la regla Sturges para cuestiones practicas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRERA" = "INFORMATICA"))
```

6.2.13.3 Gráficas de pastel. INFORMATICA Sturges

```
# install.packages("plotrix")
library(plotrix)
```

```

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

# pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = p
aste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5) #
No puede hacer pastel porque trae 0 la frecuencia. Pendiente

```

6.2.13.4 Tablas de Frecuencias de INFORMATICA Scott

```

clases <- nclass.scott(informatica)
clases

## [1] 5

tabla.intervalos <- transform(table(cut(informatica, breaks = clases)))

tabla.intervalos # es un data.frame

##           Var1  Freq
## 1 (78,80.9]    2
## 2 (80.9,83.9]   6
## 3 (83.9,86.8]  16
## 4 (86.8,89.7]   2
## 5 (89.7,92.7]   1

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##           Var1  Freq  Freq.Rel  Freq.Porc
## 1 (78,80.9]    2 0.07407407  7.407407
## 2 (80.9,83.9]   6 0.22222222 22.222222
## 3 (83.9,86.8]  16 0.59259259 59.259259
## 4 (86.8,89.7]   2 0.07407407  7.407407
## 5 (89.7,92.7]   1 0.03703704  3.703704

```

6.2.13.5 Gráficas de pastel. INFORMATICA Scott

```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %

```

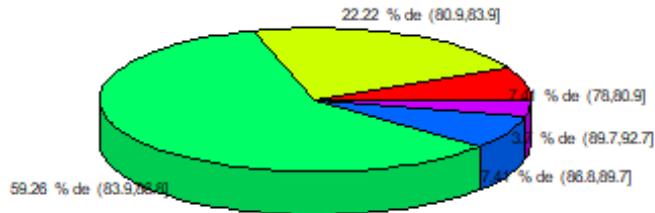
```

# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
ntervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 27 Observaciones



6.2.13.6 Tablas de Frecuencias de INFORMATICA FD

```

clases <- nclass.FD(informatica)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(informatica, breaks = clases)))

tabla.intervalos # es un data.frame

##          Var1 Freq
## 1  (78,80.4]    2
## 2  (80.4,82.9]   6
## 3  (82.9,85.3]   7
## 4  (85.3,87.8]  11
## 5  (87.8,90.2]   0
## 6  (90.2,92.7]   1

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuenci
a Porcentual

```

```

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (78,80.4]    2 0.07407407  7.407407
## 2 (80.4,82.9]   6 0.22222222 22.222222
## 3 (82.9,85.3]   7 0.25925926 25.925926
## 4 (85.3,87.8]  11 0.40740741 40.740741
## 5 (87.8,90.2]    0 0.00000000  0.000000
## 6 (90.2,92.7]    1 0.03703704  3.703704

```

6.2.13.7 Gráficas de pastel. INFORMATICA FD

```

# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

# pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5) # No puede hacer pastel porque trae 0 La frecuencia. Pendiente

```

6.2.14 ADMINISTRACION

6.2.14.1 Histogramas de ADMINISTRACION Sturges, Scott y FD

```

administracion <- datos$promedio[which(datos$promedio > 0 & datos$semestre == 2 & datos$carrera == 'ADMINISTRACION')]

administracion

## [1] 90.14 91.75 87.17 83.83 91.00 84.80 86.50 79.40 79.00 79.50 85.00
## [12] 89.67 83.00 90.00 92.00 79.40 84.17 87.00 95.33 96.17 86.33 85.20
## [23] 88.17 86.75 90.80 89.83 92.17 87.50 88.50 82.75 89.40 91.50 85.50
## [34] 91.25 90.17 95.00 83.60 92.40 86.20 92.67 90.75 90.00 90.67 90.00
## [45] 95.50 85.17 86.20 82.00 87.60 91.00 85.83 89.50 82.60 88.40 89.33
## [56] 90.00 93.00 91.60 77.00 96.17 88.00 92.60 83.00 91.40 94.00 87.17
## [67] 84.25 84.40 88.20 86.75 88.33 97.17 85.40 84.80 88.67 90.67 91.67

n <- length(administracion)

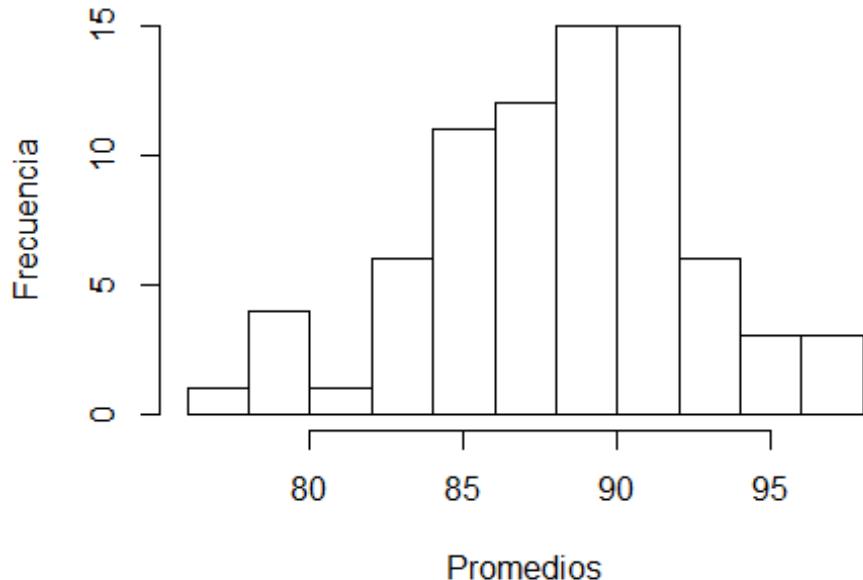
n

```

```
## [1] 77
```

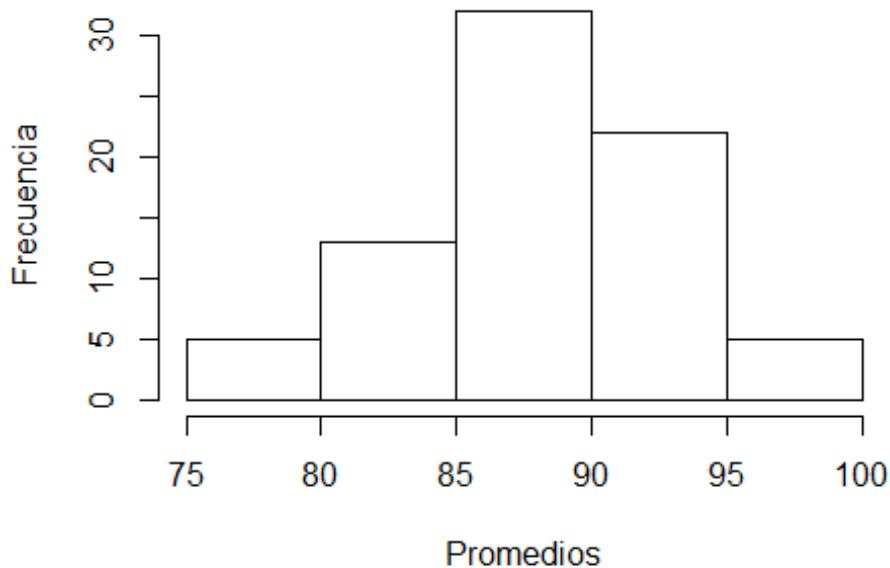
```
hist(administracion, breaks = "Sturges", freq = TRUE, main = "Promedios d  
e los alumnos ADMINISTRACION del ITD. Sturges", xlab = "Promedios", ylab  
= "Frecuencia")
```

Promedios de los alumnos ADMINISTRACION del ITD. Sturges



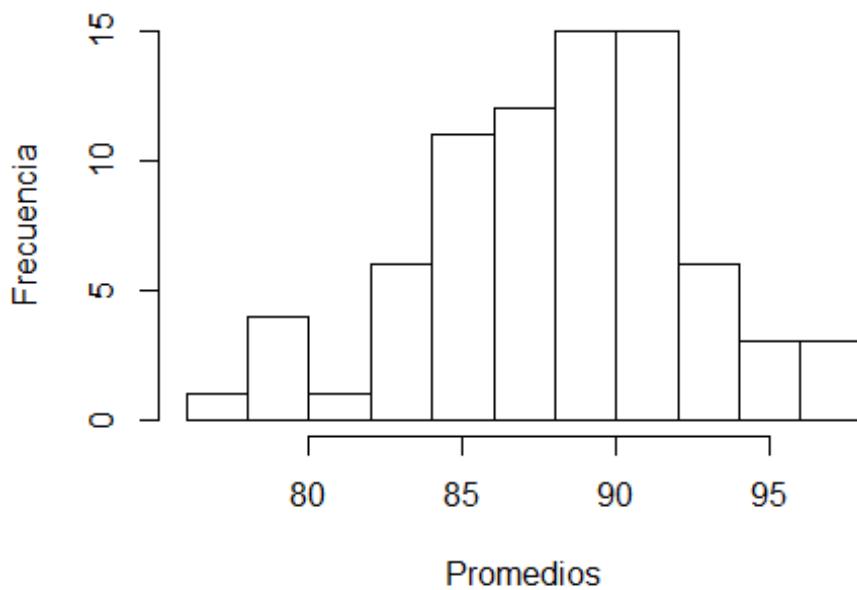
```
hist(administracion, breaks = "Scott", freq = TRUE, main = "Promedios de  
los alumnos ADMINISTRACION del ITD. Scott", xlab = "Promedios", ylab = "F  
recuencia")
```

Promedios de los alumnos ADMINISTRACION del ITD.



```
hist(administracion, breaks = "FD", freq = TRUE, main = "Promedios de los
alumnos ADMINISTRACION del ITD. FD", xlab = "Promedios", ylab = "Frecuencia")
```

Promedios de los alumnos ADMINISTRACION del ITC



6.2.14.2 Tablas de Frecuencias de ADMINISTRACION Sturges

```
clases <- nclass.Sturges(administracion)
clases

## [1] 8

tabla.intervalos <- transform(table(cut(administracion, breaks = clases))
)

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (77,79.5]    5
## 2 (79.5,82]    1
## 3 (82,84.6]    9
## 4 (84.6,87.1]   15
## 5 (87.1,89.6]   14
## 6 (89.6,92.1]   21
## 7 (92.1,94.6]    6
## 8 (94.6,97.2]    6

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##      Var1 Freq  Freq.Rel Freq.Porc
## 1 (77,79.5]    5 0.06493506  6.493506
## 2 (79.5,82]    1 0.01298701  1.298701
## 3 (82,84.6]    9 0.11688312 11.688312
## 4 (84.6,87.1]   15 0.19480519 19.480519
## 5 (87.1,89.6]   14 0.18181818 18.181818
## 6 (89.6,92.1]   21 0.27272727 27.272727
## 7 (92.1,94.6]    6 0.07792208  7.792208
## 8 (94.6,97.2]    6 0.07792208  7.792208

# ****
*****  

# VAMOS A ACUMULAR TODAS LAS CLASES EN UN NUEVO data.frame
# Luego determinar LAS FRECUENCIAS maximas de cada carrera de un mismo
data.frame
# solo usar la regla Sturges para cuestiones practicas

carreras.clases <- rbind(carreras.clases, cbind(tabla.intervalos, "CARRERA" = "ADMINISTRACION"))
```

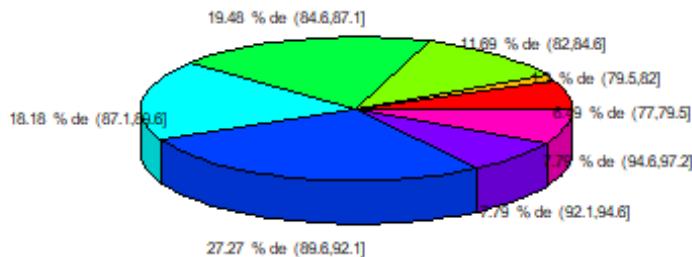
6.2.14.3 Gráficas de pastel. ADMINISTRACION Sturges

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 77 Observaciones



6.2.14.4 Tablas de Frecuencias de ADMINISTRACION Scott

```
clases <- nclass.scott(administracion)
clases

## [1] 6

tabla.intervalos <- transform(table(cut(administracion, breaks = clases)))
)

tabla.intervalos # es un data.frame

##           Var1 Freq
## 1 (77,80.4]    5
## 2 (80.4,83.7]   6
```

```

## 3 (83.7,87.1]    19
## 4 (87.1,90.4]    22
## 5 (90.4,93.8]    18
## 6 (93.8,97.2]     7

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.intervalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1 (77,80.4]    5 0.06493506  6.493506
## 2 (80.4,83.7]   6 0.07792208  7.792208
## 3 (83.7,87.1]  19 0.24675325 24.675325
## 4 (87.1,90.4]  22 0.28571429 28.571429
## 5 (90.4,93.8]  18 0.23376623 23.376623
## 6 (93.8,97.2]   7 0.09090909  9.090909

```

6.2.14.5 Gráficas de pastel. ADMINISTRACION Scott

```

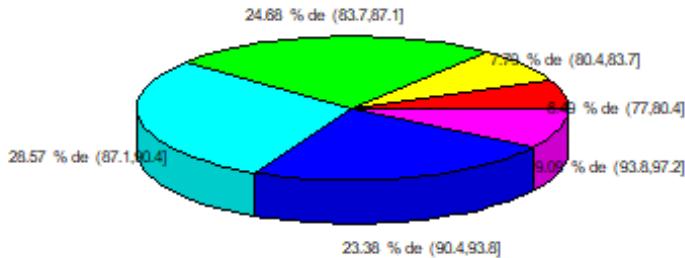
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.intervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = paste("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)

```

Frecuencia de clases. % 77 Observaciones



6.2.14.6 Tablas de Frecuencias de ADMINISTRACION FD

```
clases <- nclass.FD(administracion)
clases

## [1] 8

tabla.intervalos <- transform(table(cut(administracion, breaks = clases)))
)

tabla.intervalos # es un data.frame

##      Var1 Freq
## 1 (77,79.5]    5
## 2 (79.5,82]    1
## 3 (82,84.6]    9
## 4 (84.6,87.1]   15
## 5 (87.1,89.6]   14
## 6 (89.6,92.1]   21
## 7 (92.1,94.6]    6
## 8 (94.6,97.2]    6

# Podemos agregar columnas al Data.Frame. Frecuencia Relativa y Frecuencia Porcentual
tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Rel' = tabla.intervalos$Freq / n)

tabla.intervalos <- data.frame(tabla.intervalos, 'Freq.Porc' = tabla.inte
```

```
rvalos$Freq.Rel * 100)

tabla.intervalos

##           Var1 Freq   Freq.Rel Freq.Porc
## 1  (77,79.5]    5 0.06493506  6.493506
## 2  (79.5,82]    1 0.01298701  1.298701
## 3  (82,84.6]    9 0.11688312 11.688312
## 4 (84.6,87.1]   15 0.19480519 19.480519
## 5 (87.1,89.6]   14 0.18181818 18.181818
## 6 (89.6,92.1]   21 0.27272727 27.272727
## 7 (92.1,94.6]   6 0.07792208  7.792208
## 8 (94.6,97.2]   6 0.07792208  7.792208
```

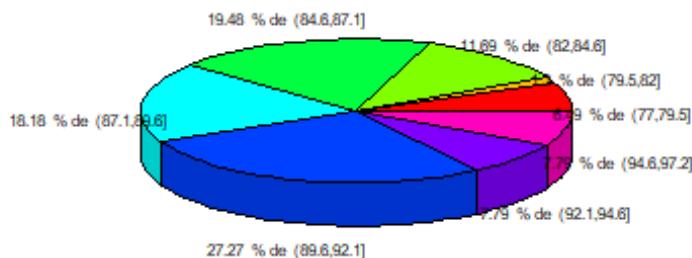
6.2.14.7 Gráficas de pastel. ADMINISTRACION FD

```
# install.packages("plotrix")
library(plotrix)

# En porcentaje %
# paste significa concatenar, unir cadenas de caracteres
etiquetas <- paste(round(tabla.intervalos$Freq.Porc,2), " % de ", tabla.i
nervalos$Var1)

pie3D(round(tabla.intervalos$Freq.Porc,2), labels = etiquetas, main = pas
te("Frecuencia de clases. % ", n, " Observaciones"), labelcex = 0.5)
```

Frecuencia de clases. % 77 Observaciones



6.2.15 Todas las CARRERAS en un solo data.frame

```
# install.packages("dplyr")
library(dplyr) # Para oder usar filter como una funci?n para seleccionar
registos con alguna condici?n

## 
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

carreras.clases

##      Var1 Freq Freq.Rel Freq.Porc      CARRERA
## 1  (77.3,79.5]    6 0.08333333  8.333333 ARQUITECTURA
## 2  (79.5,81.7]    6 0.08333333  8.333333 ARQUITECTURA
## 3  (81.7,84]    11 0.15277778 15.277778 ARQUITECTURA
## 4  (84,86.2]    18 0.25000000 25.000000 ARQUITECTURA
## 5  (86.2,88.4]   16 0.22222222 22.222222 ARQUITECTURA
## 6  (88.4,90.6]   11 0.15277778 15.277778 ARQUITECTURA
## 7  (90.6,92.8]    2 0.02777778  2.777778 ARQUITECTURA
## 8  (92.8,95]     2 0.02777778  2.777778 ARQUITECTURA
## 9  (77.5,79.7]    5 0.07692308  7.692308 BIOQUIMICA
## 10 (79.7,81.9]    7 0.10769231 10.769231 BIOQUIMICA
## 11 (81.9,84.1]   12 0.18461538 18.461538 BIOQUIMICA
## 12 (84.1,86.3]   10 0.15384615 15.384615 BIOQUIMICA
## 13 (86.3,88.5]   13 0.20000000 20.000000 BIOQUIMICA
## 14 (88.5,90.8]    8 0.12307692 12.307692 BIOQUIMICA
## 15 (90.8,93]     4 0.06153846  6.153846 BIOQUIMICA
## 16 (93,95.2]     6 0.09230769  9.230769 BIOQUIMICA
## 17 (77.3,79.5]    4 0.05970149  5.970149 CIVIL
## 18 (79.5,81.7]    6 0.08955224  8.955224 CIVIL
## 19 (81.7,83.8]    8 0.11940299 11.940299 CIVIL
## 20 (83.8,86]     12 0.17910448 17.910448 CIVIL
## 21 (86,88.2]     16 0.23880597 23.880597 CIVIL
## 22 (88.2,90.3]   13 0.19402985 19.402985 CIVIL
## 23 (90.3,92.5]    5 0.07462687  7.462687 CIVIL
## 24 (92.5,94.7]    3 0.04477612  4.477612 CIVIL
## 25 (78.7,80.9]    1 0.02702703  2.702703 ELECTRICA
## 26 (80.9,83.1]    3 0.08108108  8.108108 ELECTRICA
## 27 (83.1,85.3]    4 0.10810811 10.810811 ELECTRICA
## 28 (85.3,87.5]   15 0.40540541 40.540541 ELECTRICA
## 29 (87.5,89.7]    8 0.21621622 21.621622 ELECTRICA
## 30 (89.7,92]     4 0.10810811 10.810811 ELECTRICA
## 31 (92,94.2]     2 0.05405405  5.405405 ELECTRICA
## 32 (70,74.2]     1 0.02857143  2.857143 ELECTRONICA
```

## 33	(74.2,78.3]	0	0.00000000	0.000000	ELECTRONICA
## 34	(78.3,82.5]	0	0.00000000	0.000000	ELECTRONICA
## 35	(82.5,86.7]	0	0.00000000	0.000000	ELECTRONICA
## 36	(86.7,90.8]	4	0.11428571	11.428571	ELECTRONICA
## 37	(90.8,95]	18	0.51428571	51.428571	ELECTRONICA
## 38	(95,99.2]	12	0.34285714	34.285714	ELECTRONICA
## 39	(82.5,84.2]	2	0.02631579	2.631579	INDUSTRIAL
## 40	(84.2,85.8]	3	0.03947368	3.947368	INDUSTRIAL
## 41	(85.8,87.5]	4	0.05263158	5.263158	INDUSTRIAL
## 42	(87.5,89.2]	14	0.18421053	18.421053	INDUSTRIAL
## 43	(89.2,90.8]	18	0.23684211	23.684211	INDUSTRIAL
## 44	(90.8,92.5]	18	0.23684211	23.684211	INDUSTRIAL
## 45	(92.5,94.2]	8	0.10526316	10.526316	INDUSTRIAL
## 46	(94.2,95.8]	9	0.11842105	11.842105	INDUSTRIAL
## 47	(73.2,76.4]	2	0.03174603	3.174603	MECANICA
## 48	(76.4,79.6]	6	0.09523810	9.523810	MECANICA
## 49	(79.6,82.8]	15	0.23809524	23.809524	MECANICA
## 50	(82.8,86]	13	0.20634921	20.634921	MECANICA
## 51	(86,89.1]	13	0.20634921	20.634921	MECANICA
## 52	(89.1,92.3]	7	0.11111111	11.111111	MECANICA
## 53	(92.3,95.5]	7	0.11111111	11.111111	MECANICA
## 54	(80,82.2]	8	0.11428571	11.428571	MECATRONICA
## 55	(82.2,84.3]	14	0.20000000	20.000000	MECATRONICA
## 56	(84.3,86.5]	16	0.22857143	22.857143	MECATRONICA
## 57	(86.5,88.7]	14	0.20000000	20.000000	MECATRONICA
## 58	(88.7,90.8]	10	0.14285714	14.285714	MECATRONICA
## 59	(90.8,93]	4	0.05714286	5.714286	MECATRONICA
## 60	(93,95.2]	3	0.04285714	4.285714	MECATRONICA
## 61	(95.2,97.3]	1	0.01428571	1.428571	MECATRONICA
## 62	(78,80.5]	4	0.05882353	5.882353	QUIMICA
## 63	(80.5,83.1]	7	0.10294118	10.294118	QUIMICA
## 64	(83.1,85.6]	8	0.11764706	11.764706	QUIMICA
## 65	(85.6,88.2]	15	0.22058824	22.058824	QUIMICA
## 66	(88.2,90.7]	14	0.20588235	20.588235	QUIMICA
## 67	(90.7,93.2]	11	0.16176471	16.176471	QUIMICA
## 68	(93.2,95.8]	6	0.08823529	8.823529	QUIMICA
## 69	(95.8,98.4]	3	0.04411765	4.411765	QUIMICA
## 70	(76.7,80.3]	6	0.08955224	8.955224	SISTEMAS
## 71	(80.3,83.9]	20	0.29850746	29.850746	SISTEMAS
## 72	(83.9,87.6]	19	0.28358209	28.358209	SISTEMAS
## 73	(87.6,91.2]	14	0.20895522	20.895522	SISTEMAS
## 74	(91.2,94.8]	8	0.11940299	11.940299	SISTEMAS
## 75	(71.5,75.2]	1	0.04545455	4.545455	TIC
## 76	(75.2,79]	0	0.00000000	0.000000	TIC
## 77	(79,82.8]	3	0.13636364	13.636364	TIC
## 78	(82.8,86.5]	8	0.36363636	36.363636	TIC
## 79	(86.5,90.2]	6	0.27272727	27.272727	TIC
## 80	(90.2,94]	4	0.18181818	18.181818	TIC
## 81	(73,75.8]	2	0.02857143	2.857143	GESTION
## 82	(75.8,78.5]	3	0.04285714	4.285714	GESTION

```

## 83  (78.5,81.2]    8 0.11428571 11.428571      GESTION
## 84  (81.2,84]     12 0.17142857 17.142857      GESTION
## 85  (84,86.8]     9 0.12857143 12.857143      GESTION
## 86  (86.8,89.5]   13 0.18571429 18.571429      GESTION
## 87  (89.5,92.2]   15 0.21428571 21.428571      GESTION
## 88  (92.2,95]     8 0.11428571 11.428571      GESTION
## 89  (78,80.4]     2 0.07407407 7.407407      INFORMATICA
## 90  (80.4,82.9]   6 0.22222222 22.222222      INFORMATICA
## 91  (82.9,85.3]   7 0.25925926 25.925926      INFORMATICA
## 92  (85.3,87.8]   11 0.40740741 40.740741      INFORMATICA
## 93  (87.8,90.2]   0 0.00000000 0.000000      INFORMATICA
## 94  (90.2,92.7]   1 0.03703704 3.703704      INFORMATICA
## 95  (77,79.5]     5 0.06493506 6.493506      ADMINISTRACION
## 96  (79.5,82]     1 0.01298701 1.298701      ADMINISTRACION
## 97  (82,84.6]     9 0.11688312 11.688312      ADMINISTRACION
## 98  (84.6,87.1]   15 0.19480519 19.480519      ADMINISTRACION
## 99  (87.1,89.6]   14 0.18181818 18.181818      ADMINISTRACION
## 100 (89.6,92.1]   21 0.27272727 27.272727      ADMINISTRACION
## 101 (92.1,94.6]   6 0.07792208 7.792208      ADMINISTRACION
## 102 (94.6,97.2]   6 0.07792208 7.792208      ADMINISTRACION

```

Los máximos de cada clase de cada carrera

```

carreras.clases.maximos <- NULL # inicializamos NULL

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "ARQUITECTURA")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "BIOQUIMICA")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "CIVIL")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "ELECTRICA")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "ELECTRONICA")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "INDUSTRIA")

```

```

L")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "MECANICA")
)
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "MECATRONICA")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "QUIMICA")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "SISTEMAS")
)
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "TIC")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "GESTION")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "INFORMATICA")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

temporal <- filter(carreras.clases, carreras.clases$CARRERA == "ADMINISTRACION")
carreras.clases.maximos <- rbind(carreras.clases.maximos, filter(temporal,
, temporal$Freq.Porc == max(temporal$Freq.Porc)))

# MOSTRAR ORDENADOS
arrange(carreras.clases.maximos, CARRERA)

##          Var1 Freq  Freq.Rel Freq.Porc      CARRERA
## 1    (84,86.2]   18 0.2500000  25.00000  ARQUITECTURA
## 2  (86.3,88.5]   13 0.2000000  20.00000  BIOQUIMICA
## 3  (86,88.2]    16 0.2388060  23.88060      CIVIL
## 4  (85.3,87.5]   15 0.4054054  40.54054 ELECTRICA
## 5  (90.8,95]    18 0.5142857  51.42857 ELECTRONICA

```

```

## 6  (89.2,90.8] 18 0.2368421 23.68421 INDUSTRIAL
## 7  (90.8,92.5] 18 0.2368421 23.68421 INDUSTRIAL
## 8  (79.6,82.8] 15 0.2380952 23.80952 MECANICA
## 9  (84.3,86.5] 16 0.2285714 22.85714 MECATRONICA
## 10 (85.6,88.2] 15 0.2205882 22.05882 QUIMICA
## 11 (80.3,83.9] 20 0.2985075 29.85075 SISTEMAS
## 12 (82.8,86.5] 8 0.3636364 36.36364 TIC
## 13 (89.5,92.2] 15 0.2142857 21.42857 GESTION
## 14 (85.3,87.8] 11 0.4074074 40.74074 INFORMATICA
## 15 (89.6,92.1] 21 0.2727273 27.27273 ADMINISTRACION

```

6.2.15.1 Gráfica de barras de los valores máximos de clases de promedio de cada carrera.

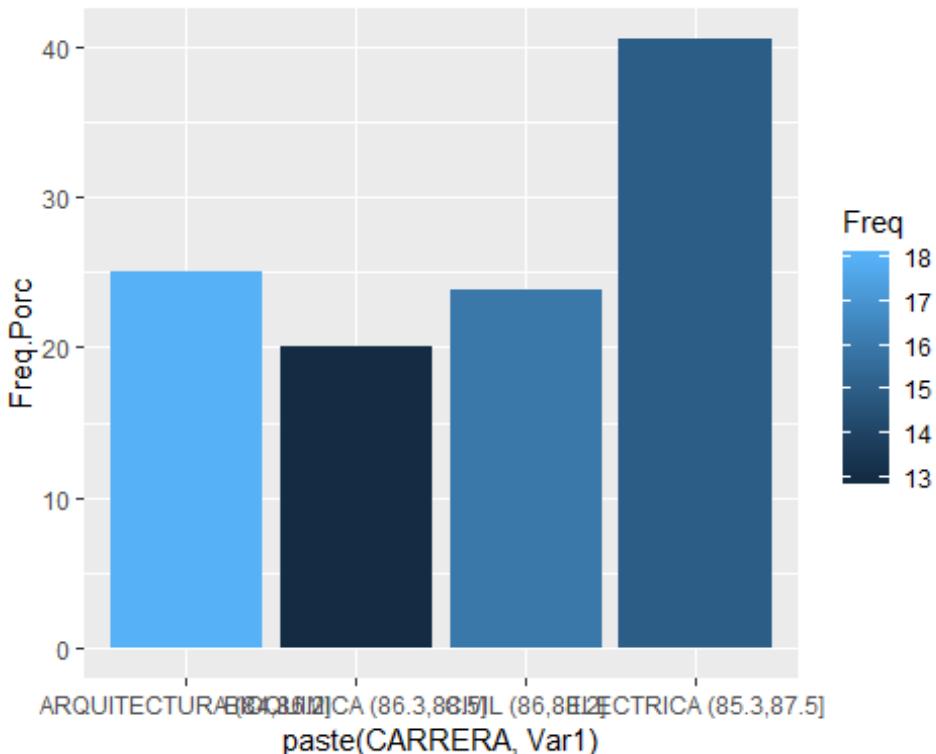
```

# install.packages("ggplot2")

library (ggplot2)

ggplot(data=carreras.clases.maximos[1:4], aes(x=paste(CARRERA,Var1), y =
Freq.Porc, fill=Freq)) + geom_bar(stat="identity")

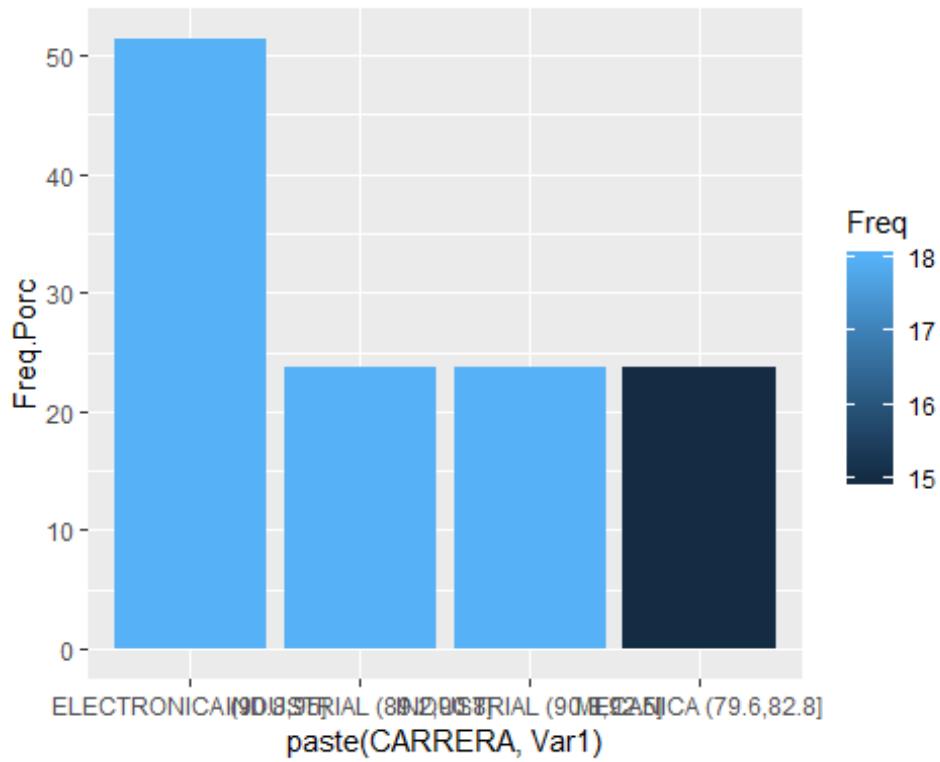
```



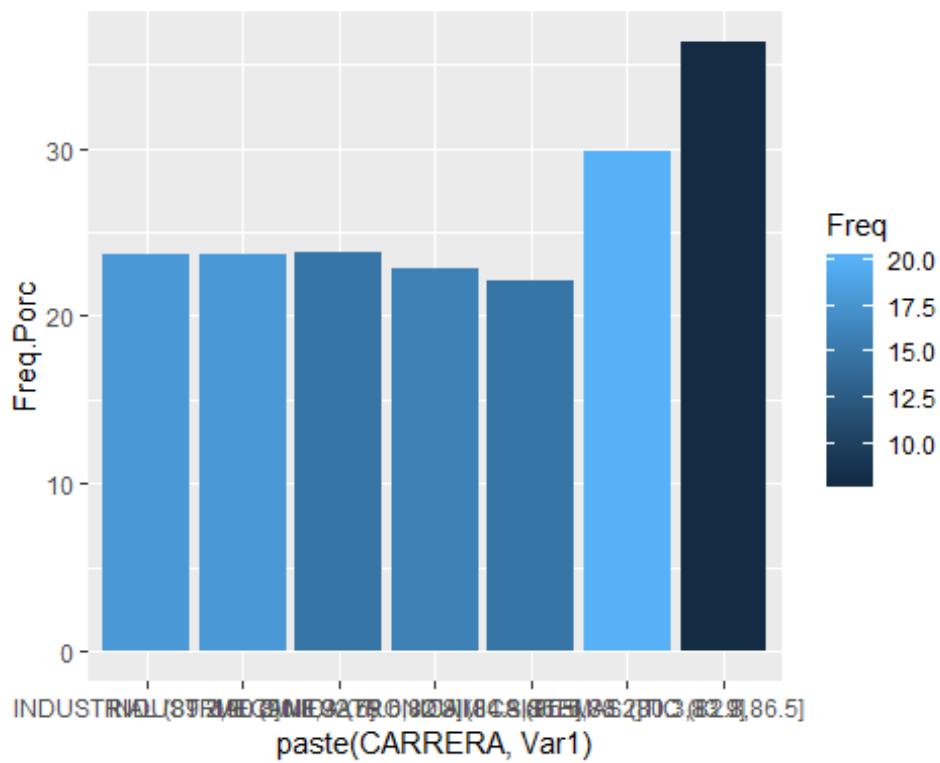
```

ggplot(data=carreras.clases.maximos[5:8], aes(x=paste(CARRERA,Var1), y =
Freq.Porc, fill=Freq)) + geom_bar(stat="identity")

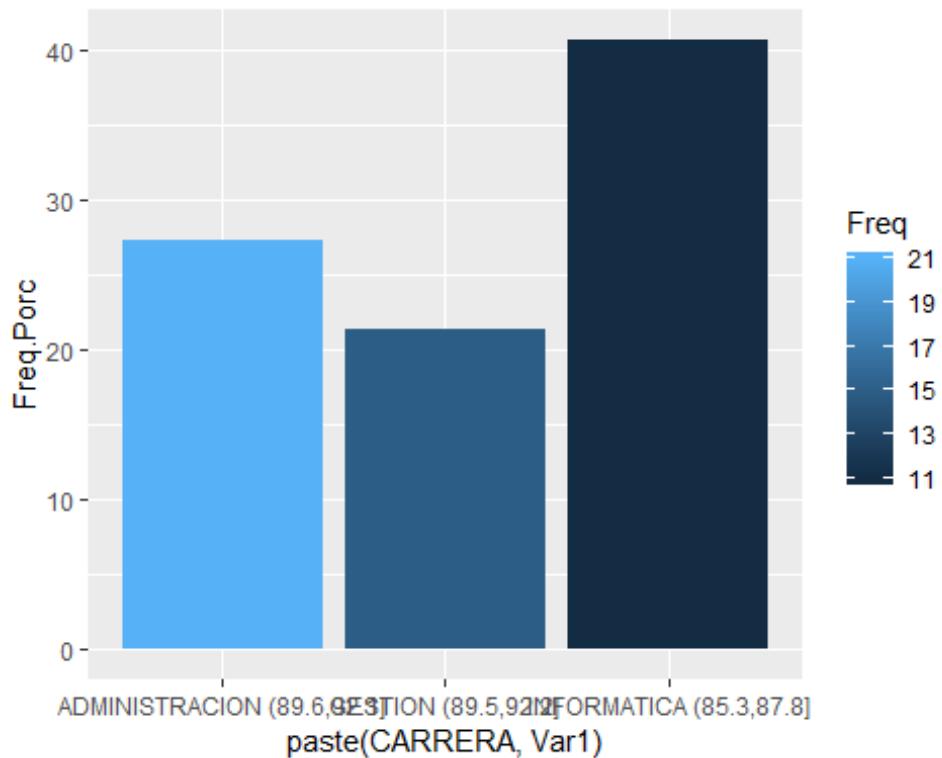
```



```
ggplot(data=carreras.clases.maximos[6:12,], aes(x=paste(CARRERA,Var1), y = Freq.Porc, fill=Freq)) + geom_bar(stat="identity")
```

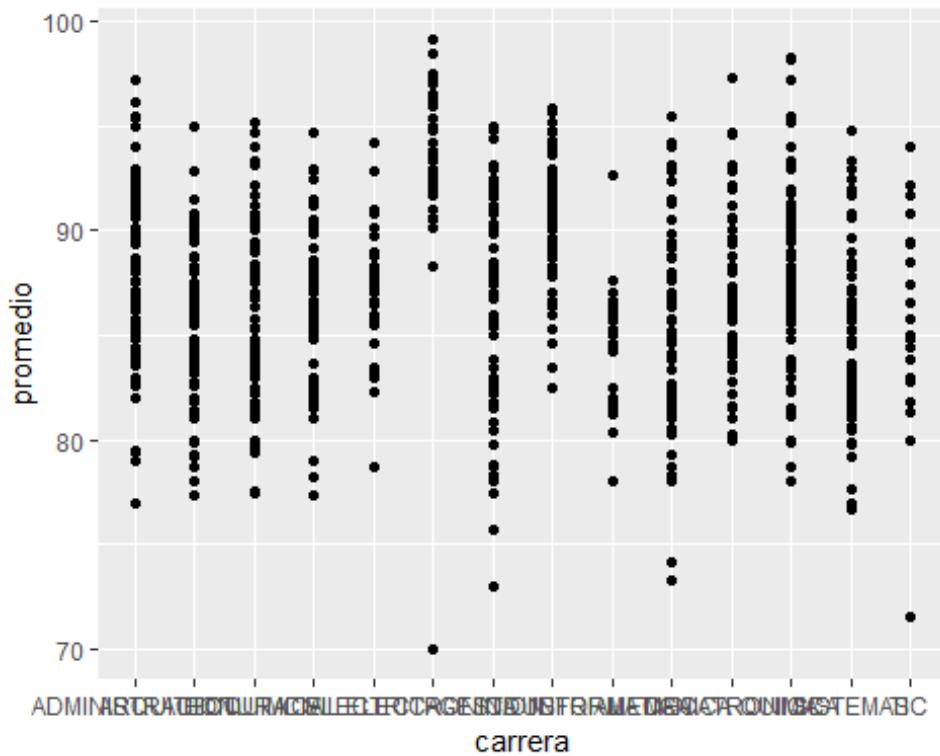


```
ggplot(data=carreras.clases.maximos[13:15,], aes(x=paste(CARRERA,Var1), y = Freq.Porc, fill=Freq)) + geom_bar(stat="identity")
```



6.2.15.2 Gráfica de puntos de todas la carreras promedios de alumnos de segundo semestre

```
ggplot(datos[which(datos$promedio > 0 & datos$semestre == 2),], aes(x = carrera, y = promedio)) + geom_point()
```



El enlace en donde se encuentra el archivo markdown de la práctica anterior es el siguiente:

<http://rpubs.com/rpizarro/518994>

7 Realizar técnicas de conteo en R

7.1 Reglas de conteo, combinaciones y permutaciones Al asignar probabilidades es necesario saber identificar y contar los resultados experimentales. A continuación tres reglas de conteo que son muy utilizadas.

7.1.1 Experimentos de pasos múltiples

La primera regla de conteo sirve para experimentos de pasos múltiples. Considere un experimento que consiste en lanzar dos monedas. Defina los resultados experimentales en términos de las caras y cruces que se observan en las dos monedas. ¿Cuántos resultados experimentales tiene este experimento? El experimento de lanzar dos monedas es un experimento de dos pasos: el paso 1 es lanzar la primera moneda y el paso 2 es lanzar la segunda moneda. Si se emplea H para denotar cara y T para denotar cruz, (H, H) será el resultado experimental en el que se tiene cara en la primera moneda y cara en la segunda moneda. Si continúa con esta notación, el espacio muestral (S) en este experimento del lanzamiento de monedas será el siguiente: Por tanto, hay cuatro resultados experimentales. En este caso es fácil enumerar todos los resultados experimentales. La regla de conteo para experimentos de pasos múltiples permite determinar el número de resultados experimentales sin tener que enumerarlos.

$$S = \{(H, H), (H, T), (T, H), (T, T)\}$$

7.2 Simular lanzamiento de una moneda

- 1 Probabilidad de un evento
- 2 Probabilidad de sello o águila al lanzar una moneda n veces
- 3 que cara cae? ¿sello o águila?

```
# simulando probabilidad con monedas y con datos
moneda <- c("aguila", "sello")
acumular<-NULL
n <- 100 # numero de veces de Laznar moneda

acumular <- sample(moneda, n, replace = TRUE)

acumular

## [1] "sello"  "aguila" "sello"  "sello"  "aguila" "aguila" "aguila"
## [8] "aguila" "sello"  "aguila" "aguila" "sello"  "aguila" "aguila"
```

```

## [15] "sello"  "aguila" "aguila" "sello"   "aguila" "sello"  "aguila"
## [22] "aguila" "sello"  "aguila" "aguila"  "aguila" "sello"  "sello"
## [29] "aguila" "sello"  "sello"  "sello"   "sello"  "aguila" "aguila"
## [36] "sello"   "aguila" "aguila" "aguila"  "sello"  "sello"  "sello"
## [43] "sello"   "aguila" "aguila" "aguila"  "sello"  "aguila" "sello"
## [50] "aguila"  "aguila" "sello"  "sello"   "sello"  "sello"  "aguila"
## [57] "sello"   "sello"  "aguila" "aguila"  "aguila" "aguila" "sello"
## [64] "sello"   "sello"  "sello"  "sello"   "sello"  "aguila" "aguila"
## [71] "sello"   "aguila" "sello"  "sello"   "aguila" "sello"  "aguila"
## [78] "aguila"  "aguila" "sello"  "sello"   "aguila" "aguila" "aguila"
## [85] "sello"   "sello"  "sello"  "sello"   "sello"  "aguila" "sello"
## [92] "sello"   "aguila" "sello"  "aguila"  "sello"  "sello"  "aguila"
## [99] "aguila"  "aguila"

```

table(acumular)/n # Sacar sus frecuencias

```

## acumular
## aguila  sello
##  0.51   0.49

```

El archivo markdown se encuentra en el siguiente enlace:
<http://rpubs.com/rpizarro/519033>

8 Realizar técnicas de conteo mediante principio aditivo y multiplicativo en R

Combinaciones Otra regla de conteo útil le permite contar el número de resultados experimentales cuando el experimento consiste en seleccionar n objetos de un conjunto (usualmente mayor) de N objetos. Ésta es la regla de conteo para combinaciones.

REGLA DE CONTEO PARA COMBINACIONES

El número de combinaciones de N objetos tomados de n en n es

$$C_n^N = \binom{N}{n} = \frac{N!}{n!(N-n)!}$$

donde $N! = N(N-1)(N-2)\cdots(2)(1)$

$$n! = n(n-1)(n-2)\cdots(2)(1)$$

y por definición, $0! = 1$

Ejemplos:

$$C_2^5 = \binom{5}{2} = \frac{5!}{2!(5-2)!} = \frac{(5)(4)(3)(2)(1)}{(2)(1)(3)(2)(1)} = \frac{120}{12} = 10$$

$$\binom{53}{6} = \frac{53!}{6!(53-6)!} = \frac{53!}{6!47!} = \frac{(53)(52)(51)(50)(49)(48)}{(6)(5)(4)(3)(2)(1)} = 22\,957\,480$$

Cuando se hace un muestreo de una población finita de tamaño N , la regla de conteo para combinaciones sirve para hallar el número de muestras de tamaño n que pueden seleccionarse.

La regla de conteo para combinaciones muestra que la probabilidad de ganar en esta lotería es muy pequeña.

9 Utilizar el factorial como técnica de conteo en R

La tercera regla de conteo que suele ser útil, es para permutaciones. Dicha regla permite calcular el número de resultados experimentales cuando se seleccionan n objetos de un conjunto de N objetos y el orden de selección es relevante. Los mismos n objetos seleccionados en orden diferente se consideran un resultado experimental diferente.

REGLA DE CONTEO PARA PERMUTACIONES

El número de permutaciones de N objetos tomados de n en n está dado por

$$P_n^N = n! \binom{N}{n} = \frac{N!}{(N-n)!}$$

Ejemplo:

$$P_2^5 = \frac{5!}{(5-2)!} = \frac{5!}{3!} = \frac{(5)(4)(3)(2)(1)}{(3)(2)(1)} = \frac{120}{6} = 20$$

En los puntos 11 y 12 se identifican prácticas de combinaciones y permutaciones.

10 Crear permutaciones en R

Esta práctica tiene la finalidad de generar permutaciones con un conjunto de nombres simulando nombres de alumnos. Se identifica en enlace en donde se encuentra el archivo markdown que genera las permutaciones.

El archivo markdown en el servicio rpubs en donde se encuentran las permutaciones está en el siguiente enlace: <http://rpubs.com/rpizarro/519014>

10.1 Combinaciones y permutaciones

Objetivo: Realizar combinaciones y permutaciones dado un conjunto de nombres de personas. Se determinan las combinaciones y permutaciones con grupos de 2 4 y 6 identificando la cantidad de opciones generadas.

10.2 La libreria que necesita cargar

```
# install.packages("gtools")  
  
library(gtools)
```

10.3 Los datos y los datos ordenados

```
nombres <- c("Ari", "Erick", "Angel", "Jonhatan", "Augusto", "Daniel", "Paty", "Adriana", "Olga", "Mary")  
  
nombres <- sort(nombres)  
  
nombres  
## [1] "Adriana"   "Angel"      "Ari"        "Augusto"    "Daniel"     "Erick"  
## [7] "Jonhatan"  "Mary"       "Olga"      "Paty"
```

10.4 Combinaciones con grupos de 2

```
N <- length(nombres) # Cantidad de alumnos  
n <- 2 # Grupos de 2  
  
combinaciones <- combinations(N, n, nombres)  
  
head(combinaciones)  
  
##      [,1]      [,2]  
## [1,] "Adriana" "Angel"  
## [2,] "Adriana" "Ari"  
## [3,] "Adriana" "Augusto"  
## [4,] "Adriana" "Daniel"  
## [5,] "Adriana" "Erick"  
## [6,] "Adriana" "Jonhatan"
```

```

# cuantas salen
nrow(combinaciones) # ¿cuántas combinaciones?

## [1] 45

# Comprobar conforme a la fórmula

factorial(N) / (factorial(n) * (factorial (N-n)))
## [1] 45

```

10.5 permutaciones con grupos de 2

```

N <- length(nombres) # Cantidad de alumnos
n <- 2 # Grupos de 2

permutaciones <- permutations(N, n, nombres)

head(permutaciones)

##      [,1]      [,2]
## [1,] "Adriana" "Angel"
## [2,] "Adriana" "Ari"
## [3,] "Adriana" "Augusto"
## [4,] "Adriana" "Daniel"
## [5,] "Adriana" "Erick"
## [6,] "Adriana" "Jonhatan"

# ¿cuántas salen ?
nrow(permutaciones) # ¿cuántas permutaciones?

## [1] 90

# Comprobar conforme a la fórmula

factorial(N) / (factorial (N-n))
## [1] 90

```

10.6 Combinaciones con grupos de 4

```

N <- length(nombres) # Cantidad de alumnos
n <- 4 # Grupos de 4

combinaciones <- combinations(N, n, nombres)

head(combinaciones)

##      [,1]      [,2]      [,3]      [,4]
## [1,] "Adriana" "Angel" "Ari" "Augusto"
## [2,] "Adriana" "Angel" "Ari" "Daniel"
## [3,] "Adriana" "Angel" "Ari" "Erick"
## [4,] "Adriana" "Angel" "Ari" "Jonhatan"

```

```

## [5,] "Adriana" "Angel" "Ari" "Mary"
## [6,] "Adriana" "Angel" "Ari" "Olga"

# cuantas salen
nrow(combinaciones) # ¿cuántas combinaciones?

## [1] 210

# Comprobar conforme a la fórmula

factorial(N) / (factorial(n) * (factorial (N-n)))

```

[1] 210

10.7 permutaciones con grupos de 4

```

N <- length(nombres) # Cantidad de alumnos
n <- 4 # Grupos de 4

permutaciones <- permutations(N, n, nombres)

head(permutaciones)

##      [,1]      [,2]      [,3]      [,4]
## [1,] "Adriana" "Angel" "Ari" "Augusto"
## [2,] "Adriana" "Angel" "Ari" "Daniel"
## [3,] "Adriana" "Angel" "Ari" "Erick"
## [4,] "Adriana" "Angel" "Ari" "Jonhatan"
## [5,] "Adriana" "Angel" "Ari" "Mary"
## [6,] "Adriana" "Angel" "Ari" "Olga"

# cuantas salen
nrow(permutaciones) # ¿cuántas permutaciones?

```

[1] 5040

Comprobar conforme a la fórmula

```

factorial(N) / (factorial (N-n))

```

[1] 5040

10.7.1.1 Combinaciones con grupos de 6

```

N <- length(nombres) # Cantidad de alumnos
n <- 6 # Grupos de 6

```

```

combinaciones <- combinations(N, n, nombres)

```

```

head(combinaciones)

```

```

##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Erick"

```

```

## [2,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Jonhatan"
## [3,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Mary"
## [4,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Olga"
## [5,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Paty"
## [6,] "Adriana" "Angel" "Ari" "Augusto" "Erick" "Jonhatan"

# cuantas salen
nrow(combinaciones) # ¿cuántas combinaciones?

## [1] 210

# Comprobar conforme a la fórmula

factorial(N) / (factorial(n) * (factorial (N-n)))
## [1] 210

```

10.8 permutaciones con grupos de 6

```

N <- length(nombres) # Cantidad de alumnos
n <- 6 # Grupos de 6

permutaciones <- permutations(N, n, nombres)

head(permutaciones)

##      [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## [1,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Erick"
## [2,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Jonhatan"
## [3,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Mary"
## [4,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Olga"
## [5,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Paty"
## [6,] "Adriana" "Angel" "Ari" "Augusto" "Erick" "Daniel"

# cuantas salen
nrow(permutaciones) # ¿cuántas permutaciones?

## [1] 151200

# Comprobar conforme a la fórmula

factorial(N) / (factorial (N-n))
## [1] 151200

```

10.9 Analizando datos con COMBINACIONES de grupos de 6

Preguntas:

10.9.1 ¿en cuántas combinaciones de grupos seis está ‘Adriana’ en la primera columna?

```
combinaciones[which(combinaciones[,1] == 'Adriana'),]
```

```

## [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] "Adriana" "Angel"   "Ari"     "Augusto"  "Daniel"   "Erick"
## [2,] "Adriana" "Angel"   "Ari"     "Augusto"  "Daniel"   "Jonhatan"
## [3,] "Adriana" "Angel"   "Ari"     "Augusto"  "Daniel"   "Mary"
## [4,] "Adriana" "Angel"   "Ari"     "Augusto"  "Daniel"   "Olga"
## [5,] "Adriana" "Angel"   "Ari"     "Augusto"  "Daniel"   "Paty"
## [6,] "Adriana" "Angel"   "Ari"     "Augusto"  "Erick"    "Jonhatan"
## [7,] "Adriana" "Angel"   "Ari"     "Augusto"  "Erick"    "Mary"
## [8,] "Adriana" "Angel"   "Ari"     "Augusto"  "Erick"    "Olga"
## [9,] "Adriana" "Angel"   "Ari"     "Augusto"  "Erick"    "Paty"
## [10,] "Adriana" "Angel"  "Ari"     "Augusto"  "Jonhatan" "Mary"
## [11,] "Adriana" "Angel"  "Ari"     "Augusto"  "Jonhatan" "Olga"
## [12,] "Adriana" "Angel"  "Ari"     "Augusto"  "Jonhatan" "Paty"
## [13,] "Adriana" "Angel"  "Ari"     "Augusto"  "Mary"     "Olga"
## [14,] "Adriana" "Angel"  "Ari"     "Augusto"  "Mary"     "Paty"
## [15,] "Adriana" "Angel"  "Ari"     "Augusto"  "Olga"    "Paty"
## [16,] "Adriana" "Angel"  "Ari"     "Daniel"   "Erick"    "Jonhatan"
## [17,] "Adriana" "Angel"  "Ari"     "Daniel"   "Erick"    "Mary"
## [18,] "Adriana" "Angel"  "Ari"     "Daniel"   "Erick"    "Olga"
## [19,] "Adriana" "Angel"  "Ari"     "Daniel"   "Erick"    "Paty"
## [20,] "Adriana" "Angel"  "Ari"     "Daniel"   "Jonhatan" "Mary"
## [21,] "Adriana" "Angel"  "Ari"     "Daniel"   "Jonhatan" "Olga"
## [22,] "Adriana" "Angel"  "Ari"     "Daniel"   "Jonhatan" "Paty"
## [23,] "Adriana" "Angel"  "Ari"     "Daniel"   "Mary"     "Olga"
## [24,] "Adriana" "Angel"  "Ari"     "Daniel"   "Mary"     "Paty"
## [25,] "Adriana" "Angel"  "Ari"     "Daniel"   "Olga"    "Paty"
## [26,] "Adriana" "Angel"  "Ari"     "Erick"   "Jonhatan" "Mary"
## [27,] "Adriana" "Angel"  "Ari"     "Erick"   "Jonhatan" "Olga"
## [28,] "Adriana" "Angel"  "Ari"     "Erick"   "Jonhatan" "Paty"
## [29,] "Adriana" "Angel"  "Ari"     "Erick"   "Mary"     "Olga"
## [30,] "Adriana" "Angel"  "Ari"     "Erick"   "Mary"     "Paty"
## [31,] "Adriana" "Angel"  "Ari"     "Erick"   "Olga"    "Paty"
## [32,] "Adriana" "Angel"  "Ari"     "Jonhatan" "Mary"     "Olga"
## [33,] "Adriana" "Angel"  "Ari"     "Jonhatan" "Mary"     "Paty"
## [34,] "Adriana" "Angel"  "Ari"     "Jonhatan" "Olga"    "Paty"
## [35,] "Adriana" "Angel"  "Ari"     "Mary"     "Olga"    "Paty"
## [36,] "Adriana" "Angel"  "Augusto" "Daniel"   "Erick"    "Jonhatan"
## [37,] "Adriana" "Angel"  "Augusto" "Daniel"   "Erick"    "Mary"
## [38,] "Adriana" "Angel"  "Augusto" "Daniel"   "Erick"    "Olga"
## [39,] "Adriana" "Angel"  "Augusto" "Daniel"   "Erick"    "Paty"
## [40,] "Adriana" "Angel"  "Augusto" "Daniel"   "Jonhatan" "Mary"
## [41,] "Adriana" "Angel"  "Augusto" "Daniel"   "Jonhatan" "Olga"
## [42,] "Adriana" "Angel"  "Augusto" "Daniel"   "Jonhatan" "Paty"
## [43,] "Adriana" "Angel"  "Augusto" "Daniel"   "Mary"     "Olga"
## [44,] "Adriana" "Angel"  "Augusto" "Daniel"   "Mary"     "Paty"
## [45,] "Adriana" "Angel"  "Augusto" "Daniel"   "Olga"    "Paty"
## [46,] "Adriana" "Angel"  "Augusto" "Erick"   "Jonhatan" "Mary"
## [47,] "Adriana" "Angel"  "Augusto" "Erick"   "Jonhatan" "Olga"
## [48,] "Adriana" "Angel"  "Augusto" "Erick"   "Jonhatan" "Paty"
## [49,] "Adriana" "Angel"  "Augusto" "Erick"   "Mary"     "Olga"

```

```

## [50,] "Adriana" "Angel"    "Augusto"   "Erick"     "Mary"      "Paty"
## [51,] "Adriana" "Angel"    "Augusto"   "Erick"     "Olga"      "Paty"
## [52,] "Adriana" "Angel"    "Augusto"   "Jonhatan"  "Mary"      "Olga"
## [53,] "Adriana" "Angel"    "Augusto"   "Jonhatan"  "Mary"      "Paty"
## [54,] "Adriana" "Angel"    "Augusto"   "Jonhatan"  "Olga"      "Paty"
## [55,] "Adriana" "Angel"    "Augusto"   "Mary"       "Olga"      "Paty"
## [56,] "Adriana" "Angel"    "Daniel"    "Erick"     "Jonhatan"  "Mary"
## [57,] "Adriana" "Angel"    "Daniel"    "Erick"     "Jonhatan"  "Olga"
## [58,] "Adriana" "Angel"    "Daniel"    "Erick"     "Jonhatan"  "Paty"
## [59,] "Adriana" "Angel"    "Daniel"    "Erick"     "Mary"       "Olga"
## [60,] "Adriana" "Angel"    "Daniel"    "Erick"     "Mary"       "Paty"
## [61,] "Adriana" "Angel"    "Daniel"    "Erick"     "Olga"      "Paty"
## [62,] "Adriana" "Angel"    "Daniel"    "Jonhatan"  "Mary"      "Olga"
## [63,] "Adriana" "Angel"    "Daniel"    "Jonhatan"  "Mary"      "Paty"
## [64,] "Adriana" "Angel"    "Daniel"    "Jonhatan"  "Olga"      "Paty"
## [65,] "Adriana" "Angel"    "Daniel"    "Mary"       "Olga"      "Paty"
## [66,] "Adriana" "Angel"    "Erick"     "Jonhatan"  "Mary"      "Olga"
## [67,] "Adriana" "Angel"    "Erick"     "Jonhatan"  "Mary"      "Paty"
## [68,] "Adriana" "Angel"    "Erick"     "Jonhatan"  "Olga"      "Paty"
## [69,] "Adriana" "Angel"    "Erick"     "Mary"       "Olga"      "Paty"
## [70,] "Adriana" "Angel"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [71,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Erick"     "Jonhatan"
## [72,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Erick"     "Mary"
## [73,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Erick"     "Olga"
## [74,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Erick"     "Paty"
## [75,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Jonhatan"  "Mary"
## [76,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Jonhatan"  "Olga"
## [77,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Jonhatan"  "Paty"
## [78,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Mary"       "Olga"
## [79,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Mary"       "Paty"
## [80,] "Adriana" "Ari"     "Augusto"   "Daniel"    "Olga"      "Paty"
## [81,] "Adriana" "Ari"     "Augusto"   "Erick"    "Jonhatan"  "Mary"
## [82,] "Adriana" "Ari"     "Augusto"   "Erick"    "Jonhatan"  "Olga"
## [83,] "Adriana" "Ari"     "Augusto"   "Erick"    "Jonhatan"  "Paty"
## [84,] "Adriana" "Ari"     "Augusto"   "Erick"    "Mary"       "Olga"
## [85,] "Adriana" "Ari"     "Augusto"   "Erick"    "Mary"       "Paty"
## [86,] "Adriana" "Ari"     "Augusto"   "Erick"    "Olga"      "Paty"
## [87,] "Adriana" "Ari"     "Augusto"   "Jonhatan" "Mary"      "Olga"
## [88,] "Adriana" "Ari"     "Augusto"   "Jonhatan" "Mary"      "Paty"
## [89,] "Adriana" "Ari"     "Augusto"   "Jonhatan" "Olga"      "Paty"
## [90,] "Adriana" "Ari"     "Augusto"   "Mary"      "Olga"      "Paty"
## [91,] "Adriana" "Ari"     "Daniel"    "Erick"    "Jonhatan"  "Mary"
## [92,] "Adriana" "Ari"     "Daniel"    "Erick"    "Jonhatan"  "Olga"
## [93,] "Adriana" "Ari"     "Daniel"    "Erick"    "Jonhatan"  "Paty"
## [94,] "Adriana" "Ari"     "Daniel"    "Erick"    "Mary"       "Olga"
## [95,] "Adriana" "Ari"     "Daniel"    "Erick"    "Mary"       "Paty"
## [96,] "Adriana" "Ari"     "Daniel"    "Erick"    "Olga"      "Paty"
## [97,] "Adriana" "Ari"     "Daniel"    "Jonhatan" "Mary"      "Olga"
## [98,] "Adriana" "Ari"     "Daniel"    "Jonhatan" "Mary"      "Paty"
## [99,] "Adriana" "Ari"     "Daniel"    "Jonhatan" "Olga"      "Paty"

```

```

## [100,] "Adriana" "Ari"      "Daniel"    "Mary"      "Olga"      "Paty"
## [101,] "Adriana" "Ari"      "Erick"     "Jonhatan"  "Mary"      "Olga"
## [102,] "Adriana" "Ari"      "Erick"     "Jonhatan"  "Mary"      "Paty"
## [103,] "Adriana" "Ari"      "Erick"     "Jonhatan"  "Olga"      "Paty"
## [104,] "Adriana" "Ari"      "Erick"     "Mary"      "Olga"      "Paty"
## [105,] "Adriana" "Ari"      "Jonhatan"  "Mary"      "Olga"      "Paty"
## [106,] "Adriana" "Augusto"   "Daniel"    "Erick"     "Jonhatan"  "Mary"
## [107,] "Adriana" "Augusto"   "Daniel"    "Erick"     "Jonhatan"  "Olga"
## [108,] "Adriana" "Augusto"   "Daniel"    "Erick"     "Jonhatan"  "Paty"
## [109,] "Adriana" "Augusto"   "Daniel"    "Erick"     "Mary"      "Olga"
## [110,] "Adriana" "Augusto"   "Daniel"    "Erick"     "Mary"      "Paty"
## [111,] "Adriana" "Augusto"   "Daniel"    "Erick"     "Olga"      "Paty"
## [112,] "Adriana" "Augusto"   "Daniel"    "Jonhatan"  "Mary"      "Olga"
## [113,] "Adriana" "Augusto"   "Daniel"    "Jonhatan"  "Mary"      "Paty"
## [114,] "Adriana" "Augusto"   "Daniel"    "Jonhatan"  "Olga"      "Paty"
## [115,] "Adriana" "Augusto"   "Daniel"    "Mary"      "Olga"      "Paty"
## [116,] "Adriana" "Augusto"   "Erick"     "Jonhatan"  "Mary"      "Olga"
## [117,] "Adriana" "Augusto"   "Erick"     "Jonhatan"  "Mary"      "Paty"
## [118,] "Adriana" "Augusto"   "Erick"     "Jonhatan"  "Olga"      "Paty"
## [119,] "Adriana" "Augusto"   "Erick"     "Mary"      "Olga"      "Paty"
## [120,] "Adriana" "Augusto"   "Jonhatan"  "Mary"      "Olga"      "Paty"
## [121,] "Adriana" "Daniel"   "Erick"     "Jonhatan"  "Mary"      "Olga"
## [122,] "Adriana" "Daniel"   "Erick"     "Jonhatan"  "Mary"      "Paty"
## [123,] "Adriana" "Daniel"   "Erick"     "Jonhatan"  "Olga"      "Paty"
## [124,] "Adriana" "Daniel"   "Erick"     "Mary"      "Olga"      "Paty"
## [125,] "Adriana" "Daniel"   "Jonhatan"  "Mary"      "Olga"      "Paty"
## [126,] "Adriana" "Erick"    "Jonhatan"  "Mary"      "Olga"      "Paty"

```

10.9.2 ¿en cuántas combinaciones de grupos seis está 'Adriana' en la primera o en la segunda o en la tercera columna?

```

combinaciones[which(combinaciones[,1] == 'Adriana' | combinaciones[,2] ==
'Adriana' | combinaciones[,3] == 'Adriana'),]

##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] "Adriana" "Angel"   "Ari"    "Augusto"  "Daniel"  "Erick"
## [2,] "Adriana" "Angel"   "Ari"    "Augusto"  "Daniel"  "Jonhatan"
## [3,] "Adriana" "Angel"   "Ari"    "Augusto"  "Daniel"  "Mary"
## [4,] "Adriana" "Angel"   "Ari"    "Augusto"  "Daniel"  "Olga"
## [5,] "Adriana" "Angel"   "Ari"    "Augusto"  "Daniel"  "Paty"
## [6,] "Adriana" "Angel"   "Ari"    "Augusto"  "Erick"   "Jonhatan"
## [7,] "Adriana" "Angel"   "Ari"    "Augusto"  "Erick"   "Mary"
## [8,] "Adriana" "Angel"   "Ari"    "Augusto"  "Erick"   "Olga"
## [9,] "Adriana" "Angel"   "Ari"    "Augusto"  "Erick"   "Paty"
## [10,] "Adriana" "Angel"   "Ari"    "Augusto"  "Jonhatan" "Mary"
## [11,] "Adriana" "Angel"   "Ari"    "Augusto"  "Jonhatan" "Olga"
## [12,] "Adriana" "Angel"   "Ari"    "Augusto"  "Jonhatan" "Paty"
## [13,] "Adriana" "Angel"   "Ari"    "Augusto"  "Mary"    "Olga"
## [14,] "Adriana" "Angel"   "Ari"    "Augusto"  "Mary"    "Paty"
## [15,] "Adriana" "Angel"   "Ari"    "Augusto"  "Olga"    "Paty"

```

```

## [16,] "Adriana" "Angel"   "Ari"      "Daniel"    "Erick"     "Jonhatan"
## [17,] "Adriana" "Angel"   "Ari"      "Daniel"    "Erick"     "Mary"
## [18,] "Adriana" "Angel"   "Ari"      "Daniel"    "Erick"     "Olga"
## [19,] "Adriana" "Angel"   "Ari"      "Daniel"    "Erick"     "Paty"
## [20,] "Adriana" "Angel"   "Ari"      "Daniel"    "Jonhatan"  "Mary"
## [21,] "Adriana" "Angel"   "Ari"      "Daniel"    "Jonhatan"  "Olga"
## [22,] "Adriana" "Angel"   "Ari"      "Daniel"    "Jonhatan"  "Paty"
## [23,] "Adriana" "Angel"   "Ari"      "Daniel"    "Mary"       "Olga"
## [24,] "Adriana" "Angel"   "Ari"      "Daniel"    "Mary"       "Paty"
## [25,] "Adriana" "Angel"   "Ari"      "Daniel"    "Olga"      "Paty"
## [26,] "Adriana" "Angel"   "Ari"      "Erick"     "Jonhatan"  "Mary"
## [27,] "Adriana" "Angel"   "Ari"      "Erick"     "Jonhatan"  "Olga"
## [28,] "Adriana" "Angel"   "Ari"      "Erick"     "Jonhatan"  "Paty"
## [29,] "Adriana" "Angel"   "Ari"      "Erick"     "Mary"       "Olga"
## [30,] "Adriana" "Angel"   "Ari"      "Erick"     "Mary"       "Paty"
## [31,] "Adriana" "Angel"   "Ari"      "Erick"     "Olga"      "Paty"
## [32,] "Adriana" "Angel"   "Ari"      "Jonhatan"  "Mary"       "Olga"
## [33,] "Adriana" "Angel"   "Ari"      "Jonhatan"  "Mary"       "Paty"
## [34,] "Adriana" "Angel"   "Ari"      "Jonhatan"  "Olga"      "Paty"
## [35,] "Adriana" "Angel"   "Ari"      "Mary"       "Olga"      "Paty"
## [36,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Erick"     "Jonhatan"
## [37,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Erick"     "Mary"
## [38,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Erick"     "Olga"
## [39,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Erick"     "Paty"
## [40,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Jonhatan"  "Mary"
## [41,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Jonhatan"  "Olga"
## [42,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Jonhatan"  "Paty"
## [43,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Mary"       "Olga"
## [44,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Mary"       "Paty"
## [45,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Olga"      "Paty"
## [46,] "Adriana" "Angel"   "Augusto"  "Erick"     "Jonhatan"  "Mary"
## [47,] "Adriana" "Angel"   "Augusto"  "Erick"     "Jonhatan"  "Olga"
## [48,] "Adriana" "Angel"   "Augusto"  "Erick"     "Jonhatan"  "Paty"
## [49,] "Adriana" "Angel"   "Augusto"  "Erick"     "Mary"       "Olga"
## [50,] "Adriana" "Angel"   "Augusto"  "Erick"     "Mary"       "Paty"
## [51,] "Adriana" "Angel"   "Augusto"  "Erick"     "Olga"      "Paty"
## [52,] "Adriana" "Angel"   "Augusto"  "Jonhatan"  "Mary"       "Olga"
## [53,] "Adriana" "Angel"   "Augusto"  "Jonhatan"  "Mary"       "Paty"
## [54,] "Adriana" "Angel"   "Augusto"  "Jonhatan"  "Olga"      "Paty"
## [55,] "Adriana" "Angel"   "Augusto"  "Mary"       "Olga"      "Paty"
## [56,] "Adriana" "Angel"   "Daniel"   "Erick"     "Jonhatan"  "Mary"
## [57,] "Adriana" "Angel"   "Daniel"   "Erick"     "Jonhatan"  "Olga"
## [58,] "Adriana" "Angel"   "Daniel"   "Erick"     "Jonhatan"  "Paty"
## [59,] "Adriana" "Angel"   "Daniel"   "Erick"     "Mary"       "Olga"
## [60,] "Adriana" "Angel"   "Daniel"   "Erick"     "Mary"       "Paty"
## [61,] "Adriana" "Angel"   "Daniel"   "Erick"     "Olga"      "Paty"
## [62,] "Adriana" "Angel"   "Daniel"   "Jonhatan"  "Mary"       "Olga"
## [63,] "Adriana" "Angel"   "Daniel"   "Jonhatan"  "Mary"       "Paty"
## [64,] "Adriana" "Angel"   "Daniel"   "Jonhatan"  "Olga"      "Paty"
## [65,] "Adriana" "Angel"   "Daniel"   "Mary"       "Olga"      "Paty"

```

```

## [66,] "Adriana" "Angel"    "Erick"      "Jonhatan"  "Mary"      "Olga"
## [67,] "Adriana" "Angel"    "Erick"      "Jonhatan"  "Mary"      "Paty"
## [68,] "Adriana" "Angel"    "Erick"      "Jonhatan"  "Olga"      "Paty"
## [69,] "Adriana" "Angel"    "Erick"      "Mary"       "Olga"      "Paty"
## [70,] "Adriana" "Angel"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [71,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Erick"      "Jonhatan"
## [72,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Erick"      "Mary"
## [73,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Erick"      "Olga"
## [74,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Erick"      "Paty"
## [75,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Jonhatan"  "Mary"
## [76,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Jonhatan"  "Olga"
## [77,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Jonhatan"  "Paty"
## [78,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Mary"       "Olga"
## [79,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Mary"       "Paty"
## [80,] "Adriana" "Ari"     "Augusto"    "Daniel"    "Olga"      "Paty"
## [81,] "Adriana" "Ari"     "Augusto"    "Erick"     "Jonhatan"  "Mary"
## [82,] "Adriana" "Ari"     "Augusto"    "Erick"     "Jonhatan"  "Olga"
## [83,] "Adriana" "Ari"     "Augusto"    "Erick"     "Jonhatan"  "Paty"
## [84,] "Adriana" "Ari"     "Augusto"    "Erick"     "Mary"       "Olga"
## [85,] "Adriana" "Ari"     "Augusto"    "Erick"     "Mary"       "Paty"
## [86,] "Adriana" "Ari"     "Augusto"    "Erick"     "Olga"      "Paty"
## [87,] "Adriana" "Ari"     "Augusto"    "Jonhatan"  "Mary"       "Olga"
## [88,] "Adriana" "Ari"     "Augusto"    "Jonhatan"  "Mary"       "Paty"
## [89,] "Adriana" "Ari"     "Augusto"    "Jonhatan"  "Olga"      "Paty"
## [90,] "Adriana" "Ari"     "Augusto"    "Mary"       "Olga"      "Paty"
## [91,] "Adriana" "Ari"     "Daniel"     "Erick"     "Jonhatan"  "Mary"
## [92,] "Adriana" "Ari"     "Daniel"     "Erick"     "Jonhatan"  "Olga"
## [93,] "Adriana" "Ari"     "Daniel"     "Erick"     "Jonhatan"  "Paty"
## [94,] "Adriana" "Ari"     "Daniel"     "Erick"     "Mary"       "Olga"
## [95,] "Adriana" "Ari"     "Daniel"     "Erick"     "Mary"       "Paty"
## [96,] "Adriana" "Ari"     "Daniel"     "Erick"     "Olga"      "Paty"
## [97,] "Adriana" "Ari"     "Daniel"     "Jonhatan"  "Mary"       "Olga"
## [98,] "Adriana" "Ari"     "Daniel"     "Jonhatan"  "Mary"       "Paty"
## [99,] "Adriana" "Ari"     "Daniel"     "Jonhatan"  "Olga"      "Paty"
## [100,] "Adriana" "Ari"    "Daniel"     "Mary"       "Olga"      "Paty"
## [101,] "Adriana" "Ari"    "Erick"      "Jonhatan"  "Mary"       "Olga"
## [102,] "Adriana" "Ari"    "Erick"      "Jonhatan"  "Mary"       "Paty"
## [103,] "Adriana" "Ari"    "Erick"      "Jonhatan"  "Olga"      "Paty"
## [104,] "Adriana" "Ari"    "Erick"      "Mary"       "Olga"      "Paty"
## [105,] "Adriana" "Ari"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [106,] "Adriana" "Augusto" "Daniel"    "Erick"     "Jonhatan"  "Mary"
## [107,] "Adriana" "Augusto" "Daniel"    "Erick"     "Jonhatan"  "Olga"
## [108,] "Adriana" "Augusto" "Daniel"    "Erick"     "Jonhatan"  "Paty"
## [109,] "Adriana" "Augusto" "Daniel"    "Erick"     "Mary"       "Olga"
## [110,] "Adriana" "Augusto" "Daniel"    "Erick"     "Mary"       "Paty"
## [111,] "Adriana" "Augusto" "Daniel"    "Erick"     "Olga"      "Paty"
## [112,] "Adriana" "Augusto" "Daniel"    "Jonhatan"  "Mary"       "Olga"
## [113,] "Adriana" "Augusto" "Daniel"    "Jonhatan"  "Mary"       "Paty"
## [114,] "Adriana" "Augusto" "Daniel"    "Jonhatan"  "Olga"      "Paty"
## [115,] "Adriana" "Augusto" "Daniel"    "Mary"       "Olga"      "Paty"

```

```

## [116,] "Adriana" "Augusto" "Erick"      "Jonhatan" "Mary"      "Olga"
## [117,] "Adriana" "Augusto" "Erick"      "Jonhatan" "Mary"      "Paty"
## [118,] "Adriana" "Augusto" "Erick"      "Jonhatan" "Olga"      "Paty"
## [119,] "Adriana" "Augusto" "Erick"      "Mary"       "Olga"      "Paty"
## [120,] "Adriana" "Augusto" "Jonhatan"   "Mary"       "Olga"      "Paty"
## [121,] "Adriana" "Daniel"   "Erick"      "Jonhatan" "Mary"      "Olga"
## [122,] "Adriana" "Daniel"   "Erick"      "Jonhatan" "Mary"      "Paty"
## [123,] "Adriana" "Daniel"   "Erick"      "Jonhatan" "Olga"      "Paty"
## [124,] "Adriana" "Daniel"   "Erick"      "Mary"       "Olga"      "Paty"
## [125,] "Adriana" "Daniel"   "Jonhatan"   "Mary"       "Olga"      "Paty"
## [126,] "Adriana" "Erick"    "Jonhatan"   "Mary"       "Olga"      "Paty"

```

10.9.3 ¿en cuántas combinaciones de grupos seis está 'Paty' en la cuarta o en la quinta o en la sexta columna?

```
combinaciones[which(combinaciones[,4] == 'Paty' | combinaciones[,5] == 'P
aty' | combinaciones[,6] == 'Paty'),]
```

```

##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] "Adriana" "Angel"   "Ari"     "Augusto"  "Daniel"   "Paty"
## [2,] "Adriana" "Angel"   "Ari"     "Augusto"  "Erick"    "Paty"
## [3,] "Adriana" "Angel"   "Ari"     "Augusto"  "Jonhatan" "Paty"
## [4,] "Adriana" "Angel"   "Ari"     "Augusto"  "Mary"     "Paty"
## [5,] "Adriana" "Angel"   "Ari"     "Augusto"  "Olga"     "Paty"
## [6,] "Adriana" "Angel"   "Ari"     "Daniel"   "Erick"    "Paty"
## [7,] "Adriana" "Angel"   "Ari"     "Daniel"   "Jonhatan" "Paty"
## [8,] "Adriana" "Angel"   "Ari"     "Daniel"   "Mary"     "Paty"
## [9,] "Adriana" "Angel"   "Ari"     "Daniel"   "Olga"     "Paty"
## [10,] "Adriana" "Angel"  "Ari"     "Erick"   "Jonhatan" "Paty"
## [11,] "Adriana" "Angel"  "Ari"     "Erick"   "Mary"     "Paty"
## [12,] "Adriana" "Angel"  "Ari"     "Erick"   "Olga"     "Paty"
## [13,] "Adriana" "Angel"  "Ari"     "Jonhatan" "Mary"     "Paty"
## [14,] "Adriana" "Angel"  "Ari"     "Jonhatan" "Olga"     "Paty"
## [15,] "Adriana" "Angel"  "Ari"     "Mary"     "Olga"     "Paty"
## [16,] "Adriana" "Angel"  "Ari"     "Augusto"  "Daniel"   "Erick"   "Paty"
## [17,] "Adriana" "Angel"  "Ari"     "Augusto"  "Daniel"   "Jonhatan" "Paty"
## [18,] "Adriana" "Angel"  "Ari"     "Augusto"  "Daniel"   "Mary"     "Paty"
## [19,] "Adriana" "Angel"  "Ari"     "Augusto"  "Daniel"   "Olga"     "Paty"
## [20,] "Adriana" "Angel"  "Ari"     "Augusto"  "Erick"   "Jonhatan" "Paty"
## [21,] "Adriana" "Angel"  "Ari"     "Augusto"  "Erick"   "Mary"     "Paty"
## [22,] "Adriana" "Angel"  "Ari"     "Augusto"  "Erick"   "Olga"     "Paty"
## [23,] "Adriana" "Angel"  "Ari"     "Augusto"  "Jonhatan" "Mary"     "Paty"
## [24,] "Adriana" "Angel"  "Ari"     "Augusto"  "Jonhatan" "Olga"     "Paty"
## [25,] "Adriana" "Angel"  "Ari"     "Augusto"  "Mary"     "Olga"     "Paty"
## [26,] "Adriana" "Angel"  "Ari"     "Daniel"   "Erick"   "Jonhatan" "Paty"
## [27,] "Adriana" "Angel"  "Ari"     "Daniel"   "Erick"   "Mary"     "Paty"
## [28,] "Adriana" "Angel"  "Ari"     "Daniel"   "Erick"   "Olga"     "Paty"
## [29,] "Adriana" "Angel"  "Ari"     "Daniel"   "Jonhatan" "Mary"     "Paty"
## [30,] "Adriana" "Angel"  "Ari"     "Daniel"   "Jonhatan" "Olga"     "Paty"
## [31,] "Adriana" "Angel"  "Ari"     "Daniel"   "Mary"     "Olga"     "Paty"

```

```

## [32,] "Adriana" "Angel"    "Erick"      "Jonhatan"  "Mary"       "Paty"
## [33,] "Adriana" "Angel"    "Erick"      "Jonhatan"  "Olga"       "Paty"
## [34,] "Adriana" "Angel"    "Erick"      "Mary"        "Olga"       "Paty"
## [35,] "Adriana" "Angel"    "Jonhatan"   "Mary"        "Olga"       "Paty"
## [36,] "Adriana" "Ari"     "Augusto"    "Daniel"     "Erick"      "Paty"
## [37,] "Adriana" "Ari"     "Augusto"    "Daniel"     "Jonhatan"   "Paty"
## [38,] "Adriana" "Ari"     "Augusto"    "Daniel"     "Mary"        "Paty"
## [39,] "Adriana" "Ari"     "Augusto"    "Daniel"     "Olga"       "Paty"
## [40,] "Adriana" "Ari"     "Augusto"    "Erick"      "Jonhatan"   "Paty"
## [41,] "Adriana" "Ari"     "Augusto"    "Erick"      "Mary"        "Paty"
## [42,] "Adriana" "Ari"     "Augusto"    "Erick"      "Olga"       "Paty"
## [43,] "Adriana" "Ari"     "Augusto"    "Jonhatan"   "Mary"        "Paty"
## [44,] "Adriana" "Ari"     "Augusto"    "Jonhatan"   "Olga"       "Paty"
## [45,] "Adriana" "Ari"     "Augusto"    "Mary"        "Olga"       "Paty"
## [46,] "Adriana" "Ari"     "Daniel"     "Erick"      "Jonhatan"   "Paty"
## [47,] "Adriana" "Ari"     "Daniel"     "Erick"      "Mary"        "Paty"
## [48,] "Adriana" "Ari"     "Daniel"     "Erick"      "Olga"       "Paty"
## [49,] "Adriana" "Ari"     "Daniel"     "Jonhatan"   "Mary"        "Paty"
## [50,] "Adriana" "Ari"     "Daniel"     "Jonhatan"   "Olga"       "Paty"
## [51,] "Adriana" "Ari"     "Daniel"     "Mary"        "Olga"       "Paty"
## [52,] "Adriana" "Ari"     "Erick"      "Jonhatan"   "Mary"        "Paty"
## [53,] "Adriana" "Ari"     "Erick"      "Jonhatan"   "Olga"       "Paty"
## [54,] "Adriana" "Ari"     "Erick"      "Mary"        "Olga"       "Paty"
## [55,] "Adriana" "Ari"     "Jonhatan"   "Mary"        "Olga"       "Paty"
## [56,] "Adriana" "Augusto" "Daniel"     "Erick"      "Jonhatan"   "Paty"
## [57,] "Adriana" "Augusto" "Daniel"     "Erick"      "Mary"        "Paty"
## [58,] "Adriana" "Augusto" "Daniel"     "Erick"      "Olga"       "Paty"
## [59,] "Adriana" "Augusto" "Daniel"     "Jonhatan"   "Mary"        "Paty"
## [60,] "Adriana" "Augusto" "Daniel"     "Jonhatan"   "Olga"       "Paty"
## [61,] "Adriana" "Augusto" "Daniel"     "Mary"        "Olga"       "Paty"
## [62,] "Adriana" "Augusto" "Erick"      "Jonhatan"   "Mary"        "Paty"
## [63,] "Adriana" "Augusto" "Erick"      "Jonhatan"   "Olga"       "Paty"
## [64,] "Adriana" "Augusto" "Erick"      "Mary"        "Olga"       "Paty"
## [65,] "Adriana" "Augusto" "Jonhatan"   "Mary"        "Olga"       "Paty"
## [66,] "Adriana" "Daniel"   "Erick"      "Jonhatan"   "Mary"        "Paty"
## [67,] "Adriana" "Daniel"   "Erick"      "Jonhatan"   "Olga"       "Paty"
## [68,] "Adriana" "Daniel"   "Erick"      "Mary"        "Olga"       "Paty"
## [69,] "Adriana" "Daniel"   "Jonhatan"   "Mary"        "Olga"       "Paty"
## [70,] "Adriana" "Erick"    "Jonhatan"   "Mary"        "Olga"       "Paty"
## [71,] "Angel"     "Ari"     "Augusto"    "Daniel"     "Erick"      "Paty"
## [72,] "Angel"     "Ari"     "Augusto"    "Daniel"     "Jonhatan"   "Paty"
## [73,] "Angel"     "Ari"     "Augusto"    "Daniel"     "Mary"        "Paty"
## [74,] "Angel"     "Ari"     "Augusto"    "Daniel"     "Olga"       "Paty"
## [75,] "Angel"     "Ari"     "Augusto"    "Erick"      "Jonhatan"   "Paty"
## [76,] "Angel"     "Ari"     "Augusto"    "Erick"      "Mary"        "Paty"
## [77,] "Angel"     "Ari"     "Augusto"    "Erick"      "Olga"       "Paty"
## [78,] "Angel"     "Ari"     "Augusto"    "Jonhatan"   "Mary"        "Paty"
## [79,] "Angel"     "Ari"     "Augusto"    "Jonhatan"   "Olga"       "Paty"
## [80,] "Angel"     "Ari"     "Augusto"    "Mary"        "Olga"       "Paty"
## [81,] "Angel"     "Ari"     "Daniel"     "Erick"      "Jonhatan"   "Paty"

```

```

## [82,] "Angel"   "Ari"      "Daniel"    "Erick"     "Mary"      "Paty"
## [83,] "Angel"   "Ari"      "Daniel"    "Erick"     "Olga"      "Paty"
## [84,] "Angel"   "Ari"      "Daniel"    "Jonhatan"  "Mary"      "Paty"
## [85,] "Angel"   "Ari"      "Daniel"    "Jonhatan"  "Olga"      "Paty"
## [86,] "Angel"   "Ari"      "Daniel"    "Mary"       "Olga"      "Paty"
## [87,] "Angel"   "Ari"      "Erick"     "Jonhatan"  "Mary"      "Paty"
## [88,] "Angel"   "Ari"      "Erick"     "Jonhatan"  "Olga"      "Paty"
## [89,] "Angel"   "Ari"      "Erick"     "Mary"       "Olga"      "Paty"
## [90,] "Angel"   "Ari"      "Jonhatan"  "Mary"       "Olga"      "Paty"
## [91,] "Angel"   "Augusto"  "Daniel"    "Erick"     "Jonhatan"  "Paty"
## [92,] "Angel"   "Augusto"  "Daniel"    "Erick"     "Mary"       "Paty"
## [93,] "Angel"   "Augusto"  "Daniel"    "Erick"     "Olga"      "Paty"
## [94,] "Angel"   "Augusto"  "Daniel"    "Jonhatan"  "Mary"      "Paty"
## [95,] "Angel"   "Augusto"  "Daniel"    "Jonhatan"  "Olga"      "Paty"
## [96,] "Angel"   "Augusto"  "Daniel"    "Mary"       "Olga"      "Paty"
## [97,] "Angel"   "Augusto"  "Erick"     "Jonhatan"  "Mary"      "Paty"
## [98,] "Angel"   "Augusto"  "Erick"     "Jonhatan"  "Olga"      "Paty"
## [99,] "Angel"   "Augusto"  "Erick"     "Mary"       "Olga"      "Paty"
## [100,] "Angel"  "Augusto"  "Jonhatan"  "Mary"       "Olga"      "Paty"
## [101,] "Angel"  "Daniel"   "Erick"     "Jonhatan"  "Mary"      "Paty"
## [102,] "Angel"  "Daniel"   "Erick"     "Jonhatan"  "Olga"      "Paty"
## [103,] "Angel"  "Daniel"   "Erick"     "Mary"       "Olga"      "Paty"
## [104,] "Angel"  "Daniel"   "Jonhatan"  "Mary"       "Olga"      "Paty"
## [105,] "Angel"  "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [106,] "Ari"    "Augusto"  "Daniel"    "Erick"     "Jonhatan"  "Paty"
## [107,] "Ari"    "Augusto"  "Daniel"    "Erick"     "Mary"       "Paty"
## [108,] "Ari"    "Augusto"  "Daniel"    "Erick"     "Olga"      "Paty"
## [109,] "Ari"    "Augusto"  "Daniel"    "Jonhatan"  "Mary"      "Paty"
## [110,] "Ari"    "Augusto"  "Daniel"    "Jonhatan"  "Olga"      "Paty"
## [111,] "Ari"    "Augusto"  "Daniel"    "Mary"       "Olga"      "Paty"
## [112,] "Ari"    "Augusto"  "Erick"     "Jonhatan"  "Mary"      "Paty"
## [113,] "Ari"    "Augusto"  "Erick"     "Jonhatan"  "Olga"      "Paty"
## [114,] "Ari"    "Augusto"  "Erick"     "Mary"       "Olga"      "Paty"
## [115,] "Ari"    "Augusto"  "Jonhatan"  "Mary"       "Olga"      "Paty"
## [116,] "Ari"    "Daniel"   "Erick"     "Jonhatan"  "Mary"      "Paty"
## [117,] "Ari"    "Daniel"   "Erick"     "Jonhatan"  "Olga"      "Paty"
## [118,] "Ari"    "Daniel"   "Erick"     "Mary"       "Olga"      "Paty"
## [119,] "Ari"    "Daniel"   "Jonhatan"  "Mary"       "Olga"      "Paty"
## [120,] "Ari"    "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [121,] "Augusto" "Daniel"   "Erick"     "Jonhatan"  "Mary"      "Paty"
## [122,] "Augusto" "Daniel"   "Erick"     "Jonhatan"  "Olga"      "Paty"
## [123,] "Augusto" "Daniel"   "Erick"     "Mary"       "Olga"      "Paty"
## [124,] "Augusto" "Daniel"   "Jonhatan"  "Mary"       "Olga"      "Paty"
## [125,] "Augusto" "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [126,] "Daniel"  "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"

```

10.9.4 ¿en cuántas combinaciones de grupos seis está 'Olga' en la cuarta o en la quinta o en la sexta columna?

```
combinaciones[which(combinaciones[,4] == 'Olga' | combinaciones[,5] == 'Olga' | combinaciones[,6] == 'Olga'),]
```

```
##      [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## [1,] "Adriana" "Angel" "Ari"   "Augusto" "Daniel" "Olga"
## [2,] "Adriana" "Angel" "Ari"   "Augusto" "Erick"  "Olga"
## [3,] "Adriana" "Angel" "Ari"   "Augusto" "Jonhatan" "Olga"
## [4,] "Adriana" "Angel" "Ari"   "Augusto" "Mary"    "Olga"
## [5,] "Adriana" "Angel" "Ari"   "Augusto" "Olga"   "Paty"
## [6,] "Adriana" "Angel" "Ari"   "Daniel"  "Erick"  "Olga"
## [7,] "Adriana" "Angel" "Ari"   "Daniel"  "Jonhatan" "Olga"
## [8,] "Adriana" "Angel" "Ari"   "Daniel"  "Mary"    "Olga"
## [9,] "Adriana" "Angel" "Ari"   "Daniel"  "Olga"   "Paty"
## [10,] "Adriana" "Angel" "Ari"   "Erick"  "Jonhatan" "Olga"
## [11,] "Adriana" "Angel" "Ari"   "Erick"  "Mary"    "Olga"
## [12,] "Adriana" "Angel" "Ari"   "Erick"  "Olga"   "Paty"
## [13,] "Adriana" "Angel" "Ari"   "Jonhatan" "Mary"    "Olga"
## [14,] "Adriana" "Angel" "Ari"   "Jonhatan" "Olga"   "Paty"
## [15,] "Adriana" "Angel" "Ari"   "Mary"    "Olga"   "Paty"
## [16,] "Adriana" "Angel" "Augusto" "Daniel"  "Erick"  "Olga"
## [17,] "Adriana" "Angel" "Augusto" "Daniel"  "Jonhatan" "Olga"
## [18,] "Adriana" "Angel" "Augusto" "Daniel"  "Mary"    "Olga"
## [19,] "Adriana" "Angel" "Augusto" "Daniel"  "Olga"   "Paty"
## [20,] "Adriana" "Angel" "Augusto" "Erick"  "Jonhatan" "Olga"
## [21,] "Adriana" "Angel" "Augusto" "Erick"  "Mary"    "Olga"
## [22,] "Adriana" "Angel" "Augusto" "Erick"  "Olga"   "Paty"
## [23,] "Adriana" "Angel" "Augusto" "Jonhatan" "Mary"    "Olga"
## [24,] "Adriana" "Angel" "Augusto" "Jonhatan" "Olga"   "Paty"
## [25,] "Adriana" "Angel" "Augusto" "Mary"    "Olga"   "Paty"
## [26,] "Adriana" "Angel" "Daniel"  "Erick"  "Jonhatan" "Olga"
## [27,] "Adriana" "Angel" "Daniel"  "Erick"  "Mary"    "Olga"
## [28,] "Adriana" "Angel" "Daniel"  "Erick"  "Olga"   "Paty"
## [29,] "Adriana" "Angel" "Daniel"  "Jonhatan" "Mary"    "Olga"
## [30,] "Adriana" "Angel" "Daniel"  "Jonhatan" "Olga"   "Paty"
## [31,] "Adriana" "Angel" "Daniel"  "Mary"    "Olga"   "Paty"
## [32,] "Adriana" "Angel" "Erick"  "Jonhatan" "Mary"    "Olga"
## [33,] "Adriana" "Angel" "Erick"  "Jonhatan" "Olga"   "Paty"
## [34,] "Adriana" "Angel" "Erick"  "Mary"    "Olga"   "Paty"
## [35,] "Adriana" "Angel" "Jonhatan" "Mary"    "Olga"   "Paty"
## [36,] "Adriana" "Ari"   "Augusto" "Daniel"  "Erick"  "Olga"
## [37,] "Adriana" "Ari"   "Augusto" "Daniel"  "Jonhatan" "Olga"
## [38,] "Adriana" "Ari"   "Augusto" "Daniel"  "Mary"    "Olga"
## [39,] "Adriana" "Ari"   "Augusto" "Daniel"  "Olga"   "Paty"
## [40,] "Adriana" "Ari"   "Augusto" "Erick"  "Jonhatan" "Olga"
## [41,] "Adriana" "Ari"   "Augusto" "Erick"  "Mary"    "Olga"
## [42,] "Adriana" "Ari"   "Augusto" "Erick"  "Olga"   "Paty"
## [43,] "Adriana" "Ari"   "Augusto" "Jonhatan" "Mary"    "Olga"
```

```

## [44,] "Adriana" "Ari"      "Augusto"   "Jonhatan" "Olga"      "Paty"
## [45,] "Adriana" "Ari"      "Augusto"   "Mary"       "Olga"      "Paty"
## [46,] "Adriana" "Ari"      "Daniel"    "Erick"     "Jonhatan" "Olga"
## [47,] "Adriana" "Ari"      "Daniel"    "Erick"     "Mary"       "Olga"
## [48,] "Adriana" "Ari"      "Daniel"    "Erick"     "Olga"      "Paty"
## [49,] "Adriana" "Ari"      "Daniel"    "Jonhatan" "Mary"       "Olga"
## [50,] "Adriana" "Ari"      "Daniel"    "Jonhatan" "Olga"      "Paty"
## [51,] "Adriana" "Ari"      "Daniel"    "Mary"       "Olga"      "Paty"
## [52,] "Adriana" "Ari"      "Erick"     "Jonhatan" "Mary"       "Olga"
## [53,] "Adriana" "Ari"      "Erick"     "Jonhatan" "Olga"      "Paty"
## [54,] "Adriana" "Ari"      "Erick"     "Mary"       "Olga"      "Paty"
## [55,] "Adriana" "Ari"      "Jonhatan" "Mary"       "Olga"      "Paty"
## [56,] "Adriana" "Augusto" "Daniel"   "Erick"     "Jonhatan" "Olga"
## [57,] "Adriana" "Augusto" "Daniel"   "Erick"     "Mary"       "Olga"
## [58,] "Adriana" "Augusto" "Daniel"   "Erick"     "Olga"      "Paty"
## [59,] "Adriana" "Augusto" "Daniel"   "Jonhatan" "Mary"       "Olga"
## [60,] "Adriana" "Augusto" "Daniel"   "Jonhatan" "Olga"      "Paty"
## [61,] "Adriana" "Augusto" "Daniel"   "Mary"       "Olga"      "Paty"
## [62,] "Adriana" "Augusto" "Erick"    "Jonhatan" "Mary"       "Olga"
## [63,] "Adriana" "Augusto" "Erick"    "Jonhatan" "Olga"      "Paty"
## [64,] "Adriana" "Augusto" "Erick"    "Mary"       "Olga"      "Paty"
## [65,] "Adriana" "Augusto" "Jonhatan" "Mary"       "Olga"      "Paty"
## [66,] "Adriana" "Daniel"  "Erick"    "Jonhatan" "Mary"       "Olga"
## [67,] "Adriana" "Daniel"  "Erick"    "Jonhatan" "Olga"      "Paty"
## [68,] "Adriana" "Daniel"  "Erick"    "Mary"       "Olga"      "Paty"
## [69,] "Adriana" "Daniel"  "Jonhatan" "Mary"       "Olga"      "Paty"
## [70,] "Adriana" "Erick"   "Jonhatan" "Mary"       "Olga"      "Paty"
## [71,] "Angel"    "Ari"     "Augusto"  "Daniel"   "Erick"     "Olga"
## [72,] "Angel"    "Ari"     "Augusto"  "Daniel"   "Jonhatan" "Olga"
## [73,] "Angel"    "Ari"     "Augusto"  "Daniel"   "Mary"       "Olga"
## [74,] "Angel"    "Ari"     "Augusto"  "Daniel"   "Olga"      "Paty"
## [75,] "Angel"    "Ari"     "Augusto"  "Erick"    "Jonhatan" "Olga"
## [76,] "Angel"    "Ari"     "Augusto"  "Erick"    "Mary"       "Olga"
## [77,] "Angel"    "Ari"     "Augusto"  "Erick"    "Olga"      "Paty"
## [78,] "Angel"    "Ari"     "Augusto"  "Jonhatan" "Mary"       "Olga"
## [79,] "Angel"    "Ari"     "Augusto"  "Jonhatan" "Olga"      "Paty"
## [80,] "Angel"    "Ari"     "Augusto"  "Mary"      "Olga"      "Paty"
## [81,] "Angel"    "Ari"     "Daniel"   "Erick"    "Jonhatan" "Olga"
## [82,] "Angel"    "Ari"     "Daniel"   "Erick"    "Mary"       "Olga"
## [83,] "Angel"    "Ari"     "Daniel"   "Erick"    "Olga"      "Paty"
## [84,] "Angel"    "Ari"     "Daniel"   "Jonhatan" "Mary"       "Olga"
## [85,] "Angel"    "Ari"     "Daniel"   "Jonhatan" "Olga"      "Paty"
## [86,] "Angel"    "Ari"     "Daniel"   "Mary"      "Olga"      "Paty"
## [87,] "Angel"    "Ari"     "Erick"    "Jonhatan" "Mary"       "Olga"
## [88,] "Angel"    "Ari"     "Erick"    "Jonhatan" "Olga"      "Paty"
## [89,] "Angel"    "Ari"     "Erick"    "Mary"      "Olga"      "Paty"
## [90,] "Angel"    "Ari"     "Jonhatan" "Mary"      "Olga"      "Paty"
## [91,] "Angel"    "Augusto" "Daniel"   "Erick"    "Jonhatan" "Olga"
## [92,] "Angel"    "Augusto" "Daniel"   "Erick"    "Mary"       "Olga"
## [93,] "Angel"    "Augusto" "Daniel"   "Erick"    "Olga"      "Paty"

```

```

## [94,] "Angel"   "Augusto" "Daniel"    "Jonhatan" "Mary"      "Olga"
## [95,] "Angel"   "Augusto" "Daniel"    "Jonhatan" "Olga"      "Paty"
## [96,] "Angel"   "Augusto" "Daniel"    "Mary"       "Olga"      "Paty"
## [97,] "Angel"   "Augusto" "Erick"     "Jonhatan" "Mary"      "Olga"
## [98,] "Angel"   "Augusto" "Erick"     "Jonhatan" "Olga"      "Paty"
## [99,] "Angel"   "Augusto" "Erick"     "Mary"       "Olga"      "Paty"
## [100,] "Angel"  "Augusto" "Jonhatan"  "Mary"       "Olga"      "Paty"
## [101,] "Angel"  "Daniel"   "Erick"     "Jonhatan" "Mary"      "Olga"
## [102,] "Angel"  "Daniel"   "Erick"     "Jonhatan" "Olga"      "Paty"
## [103,] "Angel"  "Daniel"   "Erick"     "Mary"       "Olga"      "Paty"
## [104,] "Angel"  "Daniel"   "Jonhatan"  "Mary"       "Olga"      "Paty"
## [105,] "Angel"  "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [106,] "Ari"    "Augusto"  "Daniel"    "Erick"     "Jonhatan"  "Olga"
## [107,] "Ari"    "Augusto"  "Daniel"    "Erick"     "Mary"       "Olga"
## [108,] "Ari"    "Augusto"  "Daniel"    "Erick"     "Olga"      "Paty"
## [109,] "Ari"    "Augusto"  "Daniel"    "Jonhatan"  "Mary"      "Olga"
## [110,] "Ari"    "Augusto"  "Daniel"    "Jonhatan"  "Olga"      "Paty"
## [111,] "Ari"    "Augusto"  "Daniel"    "Mary"       "Olga"      "Paty"
## [112,] "Ari"    "Augusto"  "Erick"     "Jonhatan"  "Mary"      "Olga"
## [113,] "Ari"    "Augusto"  "Erick"     "Jonhatan"  "Olga"      "Paty"
## [114,] "Ari"    "Augusto"  "Erick"     "Mary"       "Olga"      "Paty"
## [115,] "Ari"    "Augusto"  "Jonhatan"  "Mary"       "Olga"      "Paty"
## [116,] "Ari"    "Daniel"   "Erick"     "Jonhatan"  "Mary"      "Olga"
## [117,] "Ari"    "Daniel"   "Erick"     "Jonhatan"  "Olga"      "Paty"
## [118,] "Ari"    "Daniel"   "Erick"     "Mary"       "Olga"      "Paty"
## [119,] "Ari"    "Daniel"   "Jonhatan"  "Mary"       "Olga"      "Paty"
## [120,] "Ari"    "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [121,] "Augusto" "Daniel"   "Erick"     "Jonhatan"  "Mary"      "Olga"
## [122,] "Augusto" "Daniel"   "Erick"     "Jonhatan"  "Olga"      "Paty"
## [123,] "Augusto" "Daniel"   "Erick"     "Mary"       "Olga"      "Paty"
## [124,] "Augusto" "Daniel"   "Jonhatan"  "Mary"       "Olga"      "Paty"
## [125,] "Augusto" "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [126,] "Daniel"  "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"

```

10.9.5 ¿en cuántas combinaciones de grupos seis está 'Olga' en cualquier columna de la 1 a la 6?

```

combinaciones[which(combinaciones[,1] == 'Olga' | combinaciones[,2] == 'Olga' | combinaciones[,3] == 'Olga' | combinaciones[,4] == 'Olga' | combinaciones[,5] == 'Olga' | combinaciones[,6] == 'Olga'),]

```

```

##      [,1]   [,2]   [,3]   [,4]   [,5]   [,6]
## [1,] "Adriana" "Angel" "Ari"  "Augusto" "Daniel" "Olga"
## [2,] "Adriana" "Angel" "Ari"  "Augusto" "Erick"  "Olga"
## [3,] "Adriana" "Angel" "Ari"  "Augusto" "Jonhatan" "Olga"
## [4,] "Adriana" "Angel" "Ari"  "Augusto" "Mary"   "Olga"
## [5,] "Adriana" "Angel" "Ari"  "Augusto" "Olga"   "Paty"
## [6,] "Adriana" "Angel" "Ari"  "Daniel"  "Erick"  "Olga"
## [7,] "Adriana" "Angel" "Ari"  "Daniel"  "Jonhatan" "Olga"
## [8,] "Adriana" "Angel" "Ari"  "Daniel"  "Mary"   "Olga"

```

```

## [9,] "Adriana" "Angel"   "Ari"      "Daniel"    "Olga"     "Paty"
## [10,] "Adriana" "Angel"   "Ari"      "Erick"     "Jonhatan" "Olga"
## [11,] "Adriana" "Angel"   "Ari"      "Erick"     "Mary"      "Olga"
## [12,] "Adriana" "Angel"   "Ari"      "Erick"     "Olga"     "Paty"
## [13,] "Adriana" "Angel"   "Ari"      "Jonhatan"  "Mary"      "Olga"
## [14,] "Adriana" "Angel"   "Ari"      "Jonhatan"  "Olga"     "Paty"
## [15,] "Adriana" "Angel"   "Ari"      "Mary"      "Olga"     "Paty"
## [16,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Erick"    "Olga"
## [17,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Jonhatan" "Olga"
## [18,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Mary"      "Olga"
## [19,] "Adriana" "Angel"   "Augusto"  "Daniel"    "Olga"     "Paty"
## [20,] "Adriana" "Angel"   "Augusto"  "Erick"    "Jonhatan" "Olga"
## [21,] "Adriana" "Angel"   "Augusto"  "Erick"    "Mary"      "Olga"
## [22,] "Adriana" "Angel"   "Augusto"  "Erick"    "Olga"     "Paty"
## [23,] "Adriana" "Angel"   "Augusto"  "Jonhatan" "Mary"      "Olga"
## [24,] "Adriana" "Angel"   "Augusto"  "Jonhatan" "Olga"     "Paty"
## [25,] "Adriana" "Angel"   "Augusto"  "Mary"      "Olga"     "Paty"
## [26,] "Adriana" "Angel"   "Daniel"   "Erick"    "Jonhatan" "Olga"
## [27,] "Adriana" "Angel"   "Daniel"   "Erick"    "Mary"      "Olga"
## [28,] "Adriana" "Angel"   "Daniel"   "Erick"    "Olga"     "Paty"
## [29,] "Adriana" "Angel"   "Daniel"   "Jonhatan" "Mary"      "Olga"
## [30,] "Adriana" "Angel"   "Daniel"   "Jonhatan" "Olga"     "Paty"
## [31,] "Adriana" "Angel"   "Daniel"   "Mary"      "Olga"     "Paty"
## [32,] "Adriana" "Angel"   "Erick"    "Jonhatan" "Mary"      "Olga"
## [33,] "Adriana" "Angel"   "Erick"    "Jonhatan" "Olga"     "Paty"
## [34,] "Adriana" "Angel"   "Erick"    "Mary"      "Olga"     "Paty"
## [35,] "Adriana" "Angel"   "Jonhatan" "Mary"      "Olga"     "Paty"
## [36,] "Adriana" "Ari"     "Augusto"  "Daniel"   "Erick"    "Olga"
## [37,] "Adriana" "Ari"     "Augusto"  "Daniel"   "Jonhatan" "Olga"
## [38,] "Adriana" "Ari"     "Augusto"  "Daniel"   "Mary"      "Olga"
## [39,] "Adriana" "Ari"     "Augusto"  "Daniel"   "Olga"     "Paty"
## [40,] "Adriana" "Ari"     "Augusto"  "Erick"    "Jonhatan" "Olga"
## [41,] "Adriana" "Ari"     "Augusto"  "Erick"    "Mary"      "Olga"
## [42,] "Adriana" "Ari"     "Augusto"  "Erick"    "Olga"     "Paty"
## [43,] "Adriana" "Ari"     "Augusto"  "Jonhatan" "Mary"      "Olga"
## [44,] "Adriana" "Ari"     "Augusto"  "Jonhatan" "Olga"     "Paty"
## [45,] "Adriana" "Ari"     "Augusto"  "Mary"      "Olga"     "Paty"
## [46,] "Adriana" "Ari"     "Daniel"   "Erick"    "Jonhatan" "Olga"
## [47,] "Adriana" "Ari"     "Daniel"   "Erick"    "Mary"      "Olga"
## [48,] "Adriana" "Ari"     "Daniel"   "Erick"    "Olga"     "Paty"
## [49,] "Adriana" "Ari"     "Daniel"   "Jonhatan" "Mary"      "Olga"
## [50,] "Adriana" "Ari"     "Daniel"   "Jonhatan" "Olga"     "Paty"
## [51,] "Adriana" "Ari"     "Daniel"   "Mary"      "Olga"     "Paty"
## [52,] "Adriana" "Ari"     "Erick"    "Jonhatan" "Mary"      "Olga"
## [53,] "Adriana" "Ari"     "Erick"    "Jonhatan" "Olga"     "Paty"
## [54,] "Adriana" "Ari"     "Erick"    "Mary"      "Olga"     "Paty"
## [55,] "Adriana" "Ari"     "Jonhatan" "Mary"      "Olga"     "Paty"
## [56,] "Adriana" "Augusto" "Daniel"   "Erick"    "Jonhatan" "Olga"
## [57,] "Adriana" "Augusto" "Daniel"   "Erick"    "Mary"      "Olga"
## [58,] "Adriana" "Augusto" "Daniel"   "Erick"    "Olga"     "Paty"

```

```

## [59,] "Adriana" "Augusto" "Daniel"    "Jonhatan" "Mary"      "Olga"
## [60,] "Adriana" "Augusto" "Daniel"    "Jonhatan" "Olga"      "Paty"
## [61,] "Adriana" "Augusto" "Daniel"    "Mary"       "Olga"      "Paty"
## [62,] "Adriana" "Augusto" "Erick"     "Jonhatan" "Mary"      "Olga"
## [63,] "Adriana" "Augusto" "Erick"     "Jonhatan" "Olga"      "Paty"
## [64,] "Adriana" "Augusto" "Erick"     "Mary"       "Olga"      "Paty"
## [65,] "Adriana" "Augusto" "Jonhatan"   "Mary"       "Olga"      "Paty"
## [66,] "Adriana" "Daniel"  "Erick"     "Jonhatan" "Mary"      "Olga"
## [67,] "Adriana" "Daniel"  "Erick"     "Jonhatan" "Olga"      "Paty"
## [68,] "Adriana" "Daniel"  "Erick"     "Mary"       "Olga"      "Paty"
## [69,] "Adriana" "Daniel"  "Jonhatan"   "Mary"       "Olga"      "Paty"
## [70,] "Adriana" "Erick"   "Jonhatan"   "Mary"       "Olga"      "Paty"
## [71,] "Angel"    "Ari"     "Augusto"    "Daniel"    "Erick"     "Olga"
## [72,] "Angel"    "Ari"     "Augusto"    "Daniel"    "Jonhatan"  "Olga"
## [73,] "Angel"    "Ari"     "Augusto"    "Daniel"    "Mary"       "Olga"
## [74,] "Angel"    "Ari"     "Augusto"    "Daniel"    "Olga"      "Paty"
## [75,] "Angel"    "Ari"     "Augusto"    "Erick"    "Jonhatan"  "Olga"
## [76,] "Angel"    "Ari"     "Augusto"    "Erick"    "Mary"       "Olga"
## [77,] "Angel"    "Ari"     "Augusto"    "Erick"    "Olga"      "Paty"
## [78,] "Angel"    "Ari"     "Augusto"    "Jonhatan" "Mary"      "Olga"
## [79,] "Angel"    "Ari"     "Augusto"    "Jonhatan" "Olga"      "Paty"
## [80,] "Angel"    "Ari"     "Augusto"    "Mary"     "Olga"      "Paty"
## [81,] "Angel"    "Ari"     "Daniel"    "Erick"    "Jonhatan"  "Olga"
## [82,] "Angel"    "Ari"     "Daniel"    "Erick"    "Mary"       "Olga"
## [83,] "Angel"    "Ari"     "Daniel"    "Erick"    "Olga"      "Paty"
## [84,] "Angel"    "Ari"     "Daniel"    "Jonhatan" "Mary"      "Olga"
## [85,] "Angel"    "Ari"     "Daniel"    "Jonhatan" "Olga"      "Paty"
## [86,] "Angel"    "Ari"     "Daniel"    "Mary"     "Olga"      "Paty"
## [87,] "Angel"    "Ari"     "Erick"     "Jonhatan" "Mary"      "Olga"
## [88,] "Angel"    "Ari"     "Erick"     "Jonhatan" "Olga"      "Paty"
## [89,] "Angel"    "Ari"     "Erick"     "Mary"     "Olga"      "Paty"
## [90,] "Angel"    "Ari"     "Jonhatan"  "Mary"     "Olga"      "Paty"
## [91,] "Angel"    "Augusto"  "Daniel"    "Erick"    "Jonhatan"  "Olga"
## [92,] "Angel"    "Augusto"  "Daniel"    "Erick"    "Mary"       "Olga"
## [93,] "Angel"    "Augusto"  "Daniel"    "Erick"    "Olga"      "Paty"
## [94,] "Angel"    "Augusto"  "Daniel"    "Jonhatan" "Mary"      "Olga"
## [95,] "Angel"    "Augusto"  "Daniel"    "Jonhatan" "Olga"      "Paty"
## [96,] "Angel"    "Augusto"  "Daniel"    "Mary"     "Olga"      "Paty"
## [97,] "Angel"    "Augusto"  "Erick"     "Jonhatan" "Mary"      "Olga"
## [98,] "Angel"    "Augusto"  "Erick"     "Jonhatan" "Olga"      "Paty"
## [99,] "Angel"    "Augusto"  "Erick"     "Mary"     "Olga"      "Paty"
## [100,] "Angel"   "Augusto"  "Jonhatan" "Mary"     "Olga"      "Paty"
## [101,] "Angel"   "Daniel"   "Erick"     "Jonhatan" "Mary"      "Olga"
## [102,] "Angel"   "Daniel"   "Erick"     "Jonhatan" "Olga"      "Paty"
## [103,] "Angel"   "Daniel"   "Erick"     "Mary"     "Olga"      "Paty"
## [104,] "Angel"   "Daniel"   "Jonhatan"  "Mary"     "Olga"      "Paty"
## [105,] "Angel"   "Erick"    "Jonhatan"  "Mary"     "Olga"      "Paty"
## [106,] "Ari"     "Augusto"  "Daniel"    "Erick"    "Jonhatan"  "Olga"
## [107,] "Ari"     "Augusto"  "Daniel"    "Erick"    "Mary"       "Olga"
## [108,] "Ari"     "Augusto"  "Daniel"    "Erick"    "Olga"      "Paty"

```

```

## [109,] "Ari"      "Augusto" "Daniel"    "Jonhatan" "Mary"      "Olga"
## [110,] "Ari"      "Augusto" "Daniel"    "Jonhatan" "Olga"      "Paty"
## [111,] "Ari"      "Augusto" "Daniel"    "Mary"       "Olga"      "Paty"
## [112,] "Ari"      "Augusto" "Erick"     "Jonhatan" "Mary"       "Olga"
## [113,] "Ari"      "Augusto" "Erick"     "Jonhatan" "Olga"      "Paty"
## [114,] "Ari"      "Augusto" "Erick"     "Mary"       "Olga"      "Paty"
## [115,] "Ari"      "Augusto" "Jonhatan"  "Mary"       "Olga"      "Paty"
## [116,] "Ari"      "Daniel"   "Erick"     "Jonhatan" "Mary"       "Olga"
## [117,] "Ari"      "Daniel"   "Erick"     "Jonhatan" "Olga"      "Paty"
## [118,] "Ari"      "Daniel"   "Erick"     "Mary"       "Olga"      "Paty"
## [119,] "Ari"      "Daniel"   "Jonhatan"  "Mary"       "Olga"      "Paty"
## [120,] "Ari"      "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [121,] "Augusto"  "Daniel"   "Erick"     "Jonhatan" "Mary"       "Olga"
## [122,] "Augusto"  "Daniel"   "Erick"     "Jonhatan" "Olga"      "Paty"
## [123,] "Augusto"  "Daniel"   "Erick"     "Mary"       "Olga"      "Paty"
## [124,] "Augusto"  "Daniel"   "Jonhatan"  "Mary"       "Olga"      "Paty"
## [125,] "Augusto"  "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"
## [126,] "Daniel"   "Erick"    "Jonhatan"  "Mary"       "Olga"      "Paty"

```

10.9.6 ¿en cuántas combinaciones de grupos seis están 'Adriana', 'Mary', 'Olga' y 'Paty' en cualquier columna de la 1 a la 6.

Deben de estar ellas, en cuáles y cuántas aparecen?

```

combinaciones[which((combinaciones[,1] == 'Adriana' | combinaciones[,2] ==
= 'Adriana' | combinaciones[,3] == 'Adriana' | combinaciones[,4] == 'Adri
ana' | combinaciones[,5] == 'Adriana' | combinaciones[,6] == 'Adriana') &
(combinaciones[,1] == 'Mary' | combinaciones[,2] == 'Mary' | combinacione
s[,3] == 'Mary' | combinaciones[,4] == 'Mary' | combinaciones[,5] == 'Mar
y' | combinaciones[,6] == 'Mary') & (combinaciones[,1] == 'Olga' | combin
aciones[,2] == 'Olga' | combinaciones[,3] == 'Olga' | combinaciones[,4] ==
= 'Olga' | combinaciones[,5] == 'Olga' | combinaciones[,6] == 'Olga') & (
combinaciones[,1] == 'Paty' | combinaciones[,2] == 'Paty' | combinaciones
[,3] == 'Paty' | combinaciones[,4] == 'Paty' | combinaciones[,5] == 'Paty
' | combinaciones[,6] == 'Paty')),]

##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] "Adriana" "Angel"   "Ari"     "Mary"     "Olga"    "Paty"
## [2,] "Adriana" "Angel"   "Augusto"  "Mary"     "Olga"    "Paty"
## [3,] "Adriana" "Angel"   "Daniel"   "Mary"     "Olga"    "Paty"
## [4,] "Adriana" "Angel"   "Erick"   "Mary"     "Olga"    "Paty"
## [5,] "Adriana" "Angel"   "Jonhatan" "Mary"     "Olga"    "Paty"
## [6,] "Adriana" "Ari"     "Augusto"  "Mary"     "Olga"    "Paty"
## [7,] "Adriana" "Ari"     "Daniel"   "Mary"     "Olga"    "Paty"
## [8,] "Adriana" "Ari"     "Erick"   "Mary"     "Olga"    "Paty"
## [9,] "Adriana" "Ari"     "Jonhatan" "Mary"     "Olga"    "Paty"
## [10,] "Adriana" "Augusto" "Daniel"  "Mary"     "Olga"    "Paty"
## [11,] "Adriana" "Augusto" "Erick"   "Mary"     "Olga"    "Paty"
## [12,] "Adriana" "Augusto" "Jonhatan" "Mary"     "Olga"    "Paty"
## [13,] "Adriana" "Daniel"  "Erick"   "Mary"     "Olga"    "Paty"

```

```

## [14,] "Adriana" "Daniel" "Jonhatan" "Mary" "Olga" "Paty"
## [15,] "Adriana" "Erick" "Jonhatan" "Mary" "Olga" "Paty"

```

10.10 Analizando datos con PERMUTACIONES de grupos de 6

Preguntas:

- 10.10.1 ¿en cuántas permutaciones de grupos seis está ‘Adriana’ en la primera columna

Se utiliza head() y tail() para ver solos los primeros seis y los últimos seis registros, respectivamente. Esto es por ser son bastantes registros y optimizar memoria.

```

head(permutaciones[which(permutaciones[,1] == 'Adriana'),])

```

```

##      [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## [1,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Erick"
## [2,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Jonhatan"
## [3,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Mary"
## [4,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Olga"
## [5,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Paty"
## [6,] "Adriana" "Angel" "Ari" "Augusto" "Erick" "Daniel"

```

```

tail(permutaciones[which(permutaciones[,1] == 'Adriana'),])

```

```

##      [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## [15115,] "Adriana" "Paty" "Olga" "Mary" "Erick" "Jonhatan"
## [15116,] "Adriana" "Paty" "Olga" "Mary" "Jonhatan" "Angel"
## [15117,] "Adriana" "Paty" "Olga" "Mary" "Jonhatan" "Ari"
## [15118,] "Adriana" "Paty" "Olga" "Mary" "Jonhatan" "Augusto"
## [15119,] "Adriana" "Paty" "Olga" "Mary" "Jonhatan" "Daniel"
## [15120,] "Adriana" "Paty" "Olga" "Mary" "Jonhatan" "Erick"

```

- 10.10.2 ¿en cuántas permutaciones de grupos seis está ‘Adriana’ en la primera o en la segunda o en la tercera columna?

```

head(permutaciones[which(permutaciones[,1] == 'Adriana' | permutaciones[,2] == 'Adriana' | permutaciones[,3] == 'Adriana'),])

```

```

##      [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## [1,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Erick"
## [2,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Jonhatan"
## [3,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Mary"
## [4,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Olga"
## [5,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Paty"
## [6,] "Adriana" "Angel" "Ari" "Augusto" "Erick" "Daniel"

```

```

tail(permutaciones[which(permutaciones[,1] == 'Adriana' | permutaciones[,2] == 'Adriana' | permutaciones[,3] == 'Adriana'),])

```

```

##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]
## [45355,] "Paty" "Olga" "Adriana" "Mary" "Erick" "Jonhatan"
## [45356,] "Paty" "Olga" "Adriana" "Mary" "Jonhatan" "Angel"
## [45357,] "Paty" "Olga" "Adriana" "Mary" "Jonhatan" "Ari"
## [45358,] "Paty" "Olga" "Adriana" "Mary" "Jonhatan" "Augusto"
## [45359,] "Paty" "Olga" "Adriana" "Mary" "Jonhatan" "Daniel"
## [45360,] "Paty" "Olga" "Adriana" "Mary" "Jonhatan" "Erick"

```

10.10.3 ¿en cuántas permutaciones de grupos seis está 'Paty' en la cuarta o en la quinta o en la sexta columna?

```

head(permutaciones[which(permutaciones[,4] == 'Paty' | permutaciones[,5]
== 'Paty' | permutaciones[,6] == 'Paty'),])

##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]
## [1,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Paty"
## [2,] "Adriana" "Angel" "Ari" "Augusto" "Erick" "Paty"
## [3,] "Adriana" "Angel" "Ari" "Augusto" "Jonhatan" "Paty"
## [4,] "Adriana" "Angel" "Ari" "Augusto" "Mary" "Paty"
## [5,] "Adriana" "Angel" "Ari" "Augusto" "Olga" "Paty"
## [6,] "Adriana" "Angel" "Ari" "Augusto" "Paty" "Daniel"

tail(permutaciones[which(permutaciones[,4] == 'Paty' | permutaciones[,5]
== 'Paty' | permutaciones[,6] == 'Paty'),])

##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]
## [45355,] "Olga" "Mary" "Jonhatan" "Paty" "Daniel" "Erick"
## [45356,] "Olga" "Mary" "Jonhatan" "Paty" "Erick" "Adriana"
## [45357,] "Olga" "Mary" "Jonhatan" "Paty" "Erick" "Angel"
## [45358,] "Olga" "Mary" "Jonhatan" "Paty" "Erick" "Ari"
## [45359,] "Olga" "Mary" "Jonhatan" "Paty" "Erick" "Augusto"
## [45360,] "Olga" "Mary" "Jonhatan" "Paty" "Erick" "Daniel"

```

10.10.4 ¿en cuántas permutaciones de grupos seis está 'Olga' en la cuarta o en la quinta o en la sexta columna?

```

head(permutaciones[which(permutaciones[,4] == 'Olga' | permutaciones[,5]
== 'Olga' | permutaciones[,6] == 'Olga'),])

##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]
## [1,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Olga"
## [2,] "Adriana" "Angel" "Ari" "Augusto" "Erick" "Olga"
## [3,] "Adriana" "Angel" "Ari" "Augusto" "Jonhatan" "Olga"
## [4,] "Adriana" "Angel" "Ari" "Augusto" "Mary" "Olga"
## [5,] "Adriana" "Angel" "Ari" "Augusto" "Olga" "Daniel"
## [6,] "Adriana" "Angel" "Ari" "Augusto" "Olga" "Erick"

tail(permutaciones[which(permutaciones[,4] == 'Olga' | permutaciones[,5]
== 'Olga' | permutaciones[,6] == 'Olga'),])

```

```

##      [,1]  [,2]  [,3]      [,4]  [,5]  [,6]
## [45355,] "Paty" "Mary" "Jonhatan" "Olga" "Daniel" "Erick"
## [45356,] "Paty" "Mary" "Jonhatan" "Olga" "Erick" "Adriana"
## [45357,] "Paty" "Mary" "Jonhatan" "Olga" "Erick" "Angel"
## [45358,] "Paty" "Mary" "Jonhatan" "Olga" "Erick" "Ari"
## [45359,] "Paty" "Mary" "Jonhatan" "Olga" "Erick" "Augusto"
## [45360,] "Paty" "Mary" "Jonhatan" "Olga" "Erick" "Daniel"

```

10.10.5 ¿en cuántas permutaciones de grupos seis está 'Olga' en cualquier columna de la 1 a la 6?

```

head(permutaciones[which(permutaciones[,1] == 'Olga' | permutaciones[,2]
== 'Olga' | permutaciones[,3] == 'Olga' | permutaciones[,4] == 'Olga' | p
ermutaciones[,5] == 'Olga' | permutaciones[,6] == 'Olga')),)

##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]
## [1,] "Adriana" "Angel" "Ari" "Augusto" "Daniel" "Olga"
## [2,] "Adriana" "Angel" "Ari" "Augusto" "Erick" "Olga"
## [3,] "Adriana" "Angel" "Ari" "Augusto" "Jonhatan" "Olga"
## [4,] "Adriana" "Angel" "Ari" "Augusto" "Mary" "Olga"
## [5,] "Adriana" "Angel" "Ari" "Augusto" "Olga" "Daniel"
## [6,] "Adriana" "Angel" "Ari" "Augusto" "Olga" "Erick"

tail(permutaciones[which(permutaciones[,1] == 'Olga' | permutaciones[,2]
== 'Olga' | permutaciones[,3] == 'Olga' | permutaciones[,4] == 'Olga' | p
ermutaciones[,5] == 'Olga' | permutaciones[,6] == 'Olga')),)

##      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]
## [90715,] "Paty" "Olga" "Mary" "Jonhatan" "Daniel" "Erick"
## [90716,] "Paty" "Olga" "Mary" "Jonhatan" "Erick" "Adriana"
## [90717,] "Paty" "Olga" "Mary" "Jonhatan" "Erick" "Angel"
## [90718,] "Paty" "Olga" "Mary" "Jonhatan" "Erick" "Ari"
## [90719,] "Paty" "Olga" "Mary" "Jonhatan" "Erick" "Augusto"
## [90720,] "Paty" "Olga" "Mary" "Jonhatan" "Erick" "Daniel"

```

10.10.6 ¿en cuántas permutaciones de grupos seis están 'Adriana', 'Mary', 'Olga' y 'Paty' en cualquier columna de la 1 a la 6.

Deben de estar ellas, en cuáles y cuántas aparecen?

```

head(permutaciones[which((permutaciones[,1] == 'Adriana' | permutaciones[
,2] == 'Adriana' | permutaciones[,3] == 'Adriana' | permutaciones[,4] ==
'Adriana' | permutaciones[,5] == 'Adriana' | permutaciones[,6] == 'Adrian
a') & (permutaciones[,1] == 'Mary' | permutaciones[,2] == 'Mary' | combin
aciones[,3] == 'Mary' | permutaciones[,4] == 'Mary' | permutaciones[,5] ==
'Mary' | permutaciones[,6] == 'Mary') & (permutaciones[,1] == 'Olga' |
permutaciones[,2] == 'Olga' | permutaciones[,3] == 'Olga' | permutaciones
[,4] == 'Olga' | permutaciones[,5] == 'Olga' | permutaciones[,6] == 'Olga
') & (permutaciones[,1] == 'Paty' | permutaciones[,2] == 'Paty' | permu
ta

```

```

ciones[,3] == 'Paty' | permutaciones[,4] == 'Paty' | permutaciones[,5] ==
'Paty' | permutaciones[,6] == 'Paty'))),])

##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] "Adriana" "Angel" "Ari" "Mary" "Olga" "Paty"
## [2,] "Adriana" "Angel" "Ari" "Mary" "Paty" "Olga"
## [3,] "Adriana" "Angel" "Ari" "Olga" "Mary" "Paty"
## [4,] "Adriana" "Angel" "Ari" "Olga" "Paty" "Mary"
## [5,] "Adriana" "Angel" "Ari" "Paty" "Mary" "Olga"
## [6,] "Adriana" "Angel" "Ari" "Paty" "Olga" "Mary"

tail(permutaciones[which((permutaciones[,1] == 'Adriana' | permutaciones[,2] ==
'Adriana' | permutaciones[,3] == 'Adriana' | permutaciones[,4] ==
'Adriana' | permutaciones[,5] == 'Adriana' | permutaciones[,6] == 'Adrian
a') & (permutaciones[,1] == 'Mary' | permutaciones[,2] == 'Mary' | combin
aciones[,3] == 'Mary' | permutaciones[,4] == 'Mary' | permutaciones[,5] =
= 'Mary' | permutaciones[,6] == 'Mary') & (permutaciones[,1] == 'Olga' |
permutaciones[,2] == 'Olga' | permutaciones[,3] == 'Olga' | permutaciones[,4] ==
'Olga' | permutaciones[,5] == 'Olga' | permutaciones[,6] == 'Olga
') & (permutaciones[,1] == 'Paty' | permutaciones[,2] == 'Paty' | perm
utaciones[,3] == 'Paty' | permutaciones[,4] == 'Paty' | permutaciones[,5] ==
'Paty' | permutaciones[,6] == 'Paty'))),])

##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [8995,] "Paty" "Olga" "Jonhatan" "Mary" "Adriana" "Erick"
## [8996,] "Paty" "Olga" "Jonhatan" "Mary" "Angel" "Adriana"
## [8997,] "Paty" "Olga" "Jonhatan" "Mary" "Ari" "Adriana"
## [8998,] "Paty" "Olga" "Jonhatan" "Mary" "Augusto" "Adriana"
## [8999,] "Paty" "Olga" "Jonhatan" "Mary" "Daniel" "Adriana"
## [9000,] "Paty" "Olga" "Jonhatan" "Mary" "Erick" "Adriana"

```

10.10.6.1 ¿En cuantas ocasiones de las PERMUTACIONES en grupos de SEIS se encuentra juntas “Adriana”, “Mary”, “Olga” y “Paty”.

Ya sean en las columnas 1-4, o en las columnas 2-5, o en las columnas 3-6. en ese orden?.

```

permutaciones[which((permutaciones[,1] == 'Adriana' & permutaciones[,2] =
= 'Mary' & permutaciones[,3] == 'Olga' & permutaciones[,4] == 'Paty') | (
permutaciones[,2] == 'Adriana' & permutaciones[,3] == 'Mary' & permutacio
nes[,4] == 'Olga' & permutaciones[,5] == 'Paty') | (permutaciones[,3] ==
'Adriana' & permutaciones[,4] == 'Mary' & permutaciones[,5] == 'Olga' & p
ermutaciones[,6] == 'Paty'))),]

```

```

##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] "Adriana" "Mary" "Olga" "Paty" "Angel" "Ari"
## [2,] "Adriana" "Mary" "Olga" "Paty" "Angel" "Augusto"
## [3,] "Adriana" "Mary" "Olga" "Paty" "Angel" "Daniel"
## [4,] "Adriana" "Mary" "Olga" "Paty" "Angel" "Erick"
## [5,] "Adriana" "Mary" "Olga" "Paty" "Angel" "Jonhatan"

```

```

## [6,] "Adriana" "Mary"   "Olga"    "Paty"   "Ari"     "Angel"
## [7,] "Adriana" "Mary"   "Olga"    "Paty"   "Ari"     "Augusto"
## [8,] "Adriana" "Mary"   "Olga"    "Paty"   "Ari"     "Daniel"
## [9,] "Adriana" "Mary"   "Olga"    "Paty"   "Ari"     "Erick"
## [10,] "Adriana" "Mary"   "Olga"    "Paty"   "Ari"     "Jonhatan"
## [11,] "Adriana" "Mary"   "Olga"    "Paty"   "Augusto" "Angel"
## [12,] "Adriana" "Mary"   "Olga"    "Paty"   "Augusto" "Ari"
## [13,] "Adriana" "Mary"   "Olga"    "Paty"   "Augusto" "Daniel"
## [14,] "Adriana" "Mary"   "Olga"    "Paty"   "Augusto" "Erick"
## [15,] "Adriana" "Mary"   "Olga"    "Paty"   "Augusto" "Jonhatan"
## [16,] "Adriana" "Mary"   "Olga"    "Paty"   "Daniel"  "Angel"
## [17,] "Adriana" "Mary"   "Olga"    "Paty"   "Daniel"  "Ari"
## [18,] "Adriana" "Mary"   "Olga"    "Paty"   "Daniel"  "Augusto"
## [19,] "Adriana" "Mary"   "Olga"    "Paty"   "Daniel"  "Erick"
## [20,] "Adriana" "Mary"   "Olga"    "Paty"   "Daniel"  "Jonhatan"
## [21,] "Adriana" "Mary"   "Olga"    "Paty"   "Erick"   "Angel"
## [22,] "Adriana" "Mary"   "Olga"    "Paty"   "Erick"   "Ari"
## [23,] "Adriana" "Mary"   "Olga"    "Paty"   "Erick"   "Augusto"
## [24,] "Adriana" "Mary"   "Olga"    "Paty"   "Erick"   "Daniel"
## [25,] "Adriana" "Mary"   "Olga"    "Paty"   "Erick"   "Jonhatan"
## [26,] "Adriana" "Mary"   "Olga"    "Paty"   "Jonhatan" "Angel"
## [27,] "Adriana" "Mary"   "Olga"    "Paty"   "Jonhatan" "Ari"
## [28,] "Adriana" "Mary"   "Olga"    "Paty"   "Jonhatan" "Augusto"
## [29,] "Adriana" "Mary"   "Olga"    "Paty"   "Jonhatan" "Daniel"
## [30,] "Adriana" "Mary"   "Olga"    "Paty"   "Jonhatan" "Erick"
## [31,] "Angel"    "Adriana" "Mary"   "Olga"    "Paty"   "Ari"
## [32,] "Angel"    "Adriana" "Mary"   "Olga"    "Paty"   "Augusto"
## [33,] "Angel"    "Adriana" "Mary"   "Olga"    "Paty"   "Daniel"
## [34,] "Angel"    "Adriana" "Mary"   "Olga"    "Paty"   "Erick"
## [35,] "Angel"    "Adriana" "Mary"   "Olga"    "Paty"   "Jonhatan"
## [36,] "Angel"    "Ari"     "Adriana" "Mary"   "Olga"    "Paty"
## [37,] "Angel"    "Augusto" "Adriana" "Mary"   "Olga"    "Paty"
## [38,] "Angel"    "Daniel"  "Adriana" "Mary"   "Olga"    "Paty"
## [39,] "Angel"    "Erick"   "Adriana" "Mary"   "Olga"    "Paty"
## [40,] "Angel"    "Jonhatan" "Adriana" "Mary"   "Olga"    "Paty"
## [41,] "Ari"      "Adriana" "Mary"   "Olga"    "Paty"   "Angel"
## [42,] "Ari"      "Adriana" "Mary"   "Olga"    "Paty"   "Augusto"
## [43,] "Ari"      "Adriana" "Mary"   "Olga"    "Paty"   "Daniel"
## [44,] "Ari"      "Adriana" "Mary"   "Olga"    "Paty"   "Erick"
## [45,] "Ari"      "Adriana" "Mary"   "Olga"    "Paty"   "Jonhatan"
## [46,] "Ari"      "Angel"   "Adriana" "Mary"   "Olga"    "Paty"
## [47,] "Ari"      "Augusto" "Adriana" "Mary"   "Olga"    "Paty"
## [48,] "Ari"      "Daniel"  "Adriana" "Mary"   "Olga"    "Paty"
## [49,] "Ari"      "Erick"   "Adriana" "Mary"   "Olga"    "Paty"
## [50,] "Ari"      "Jonhatan" "Adriana" "Mary"   "Olga"    "Paty"
## [51,] "Augusto"  "Adriana" "Mary"   "Olga"    "Paty"   "Angel"
## [52,] "Augusto"  "Adriana" "Mary"   "Olga"    "Paty"   "Ari"
## [53,] "Augusto"  "Adriana" "Mary"   "Olga"    "Paty"   "Daniel"
## [54,] "Augusto"  "Adriana" "Mary"   "Olga"    "Paty"   "Erick"
## [55,] "Augusto"  "Adriana" "Mary"   "Olga"    "Paty"   "Jonhatan"

```

```

## [56,] "Augusto" "Angel"      "Adriana" "Mary"   "Olga"    "Paty"
## [57,] "Augusto" "Ari"        "Adriana" "Mary"   "Olga"    "Paty"
## [58,] "Augusto" "Daniel"     "Adriana" "Mary"   "Olga"    "Paty"
## [59,] "Augusto" "Erick"      "Adriana" "Mary"   "Olga"    "Paty"
## [60,] "Augusto" "Jonhatan"   "Adriana" "Mary"   "Olga"    "Paty"
## [61,] "Daniel"   "Adriana"   "Mary"    "Olga"   "Paty"    "Angel"
## [62,] "Daniel"   "Adriana"   "Mary"    "Olga"   "Paty"    "Ari"
## [63,] "Daniel"   "Adriana"   "Mary"    "Olga"   "Paty"    "Augusto"
## [64,] "Daniel"   "Adriana"   "Mary"    "Olga"   "Paty"    "Erick"
## [65,] "Daniel"   "Adriana"   "Mary"    "Olga"   "Paty"    "Jonhatan"
## [66,] "Daniel"   "Angel"     "Adriana" "Mary"   "Olga"    "Paty"
## [67,] "Daniel"   "Ari"       "Adriana" "Mary"   "Olga"    "Paty"
## [68,] "Daniel"   "Augusto"   "Adriana" "Mary"   "Olga"    "Paty"
## [69,] "Daniel"   "Erick"     "Adriana" "Mary"   "Olga"    "Paty"
## [70,] "Daniel"   "Jonhatan"  "Adriana" "Mary"   "Olga"    "Paty"
## [71,] "Erick"    "Adriana"   "Mary"    "Olga"   "Paty"    "Angel"
## [72,] "Erick"    "Adriana"   "Mary"    "Olga"   "Paty"    "Ari"
## [73,] "Erick"    "Adriana"   "Mary"    "Olga"   "Paty"    "Augusto"
## [74,] "Erick"    "Adriana"   "Mary"    "Olga"   "Paty"    "Daniel"
## [75,] "Erick"    "Adriana"   "Mary"    "Olga"   "Paty"    "Jonhatan"
## [76,] "Erick"    "Angel"     "Adriana" "Mary"   "Olga"    "Paty"
## [77,] "Erick"    "Ari"       "Adriana" "Mary"   "Olga"    "Paty"
## [78,] "Erick"    "Augusto"   "Adriana" "Mary"   "Olga"    "Paty"
## [79,] "Erick"    "Daniel"    "Adriana" "Mary"   "Olga"    "Paty"
## [80,] "Erick"    "Jonhatan"  "Adriana" "Mary"   "Olga"    "Paty"
## [81,] "Jonhatan" "Adriana"   "Mary"    "Olga"   "Paty"    "Angel"
## [82,] "Jonhatan" "Adriana"   "Mary"    "Olga"   "Paty"    "Ari"
## [83,] "Jonhatan" "Adriana"   "Mary"    "Olga"   "Paty"    "Augusto"
## [84,] "Jonhatan" "Adriana"   "Mary"    "Olga"   "Paty"    "Daniel"
## [85,] "Jonhatan" "Adriana"   "Mary"    "Olga"   "Paty"    "Erick"
## [86,] "Jonhatan" "Angel"     "Adriana" "Mary"   "Olga"    "Paty"
## [87,] "Jonhatan" "Ari"       "Adriana" "Mary"   "Olga"    "Paty"
## [88,] "Jonhatan" "Augusto"   "Adriana" "Mary"   "Olga"    "Paty"
## [89,] "Jonhatan" "Daniel"    "Adriana" "Mary"   "Olga"    "Paty"
## [90,] "Jonhatan" "Erick"     "Adriana" "Mary"   "Olga"    "Paty"

```

11 Crear combinaciones en R

Esta práctica tiene la finalidad de generar combinaciones con un conjunto de nombres simulando nombres de alumnos. Se identifica en enlace en donde se encuentra el archivo markdown que genera las combinaciones.

El aspecto de combinaciones se encuentra inmerso en el ejercicio anterior, pero se presenta otro ejemplo de combinaciones y permutaciones con números.

11.1 Combinaciones y permutaciones con números

Se carga el paquete previamente instalado en R Studio que permite ejecutar combinaciones y permutaciones

```
library(gtools)
```

11.2 Combinaciones con números

Las combinaciones NO repiten elementos en un mismo grupo

NO HAY GRUPOS CON EL MISMO ELEMENTO MAS DE UNA VEZ

```
N <- 100 # Número de elementos
n <- 2 # grupos de 2 en 2

alumnos <- c(1:N) # Son Los alumnos con id un número consecutivo

combinaciones <- combinations(N, n, alumnos)
head(combinaciones) # Las primeras combinaciones

##      [,1] [,2]
## [1,]    1    2
## [2,]    1    3
## [3,]    1    4
## [4,]    1    5
## [5,]    1    6
## [6,]    1    7

tail(combinaciones) # Las últimas combinaciones

##      [,1] [,2]
## [4945,]  97   98
## [4946,]  97   99
## [4947,]  97  100
## [4948,]  98   99
## [4949,]  98  100
## [4950,]  99  100

nrow(combinaciones) # ¿cuántas combinaciones?
```

```

## [1] 4950

# De acuerdo a La Fórmula. Debe salir el mismo número que nrow(combinaciones)
factorial(N) / (factorial(n) * (factorial(N-n)))
## [1] 4950

```

11.3 Permutaciones con números

Las permutaciones SI repiten elementos en un mismo grupo, pero en diferente orden o posición en el grupo

Bo hay grupos con el mismo elemento más de una vez

El número de permutaciones es MAYOR al de las combinaciones con el mismo número de elementos en los grupos.

```

permutaciones <- permutations(N, n, alumnos)
head(permutaciones) # Las primeras permutaciones

##      [,1] [,2]
## [1,]    1    2
## [2,]    1    3
## [3,]    1    4
## [4,]    1    5
## [5,]    1    6
## [6,]    1    7

tail(permutaciones) # Las últimas permutaciones

##      [,1] [,2]
## [9895,] 100   94
## [9896,] 100   95
## [9897,] 100   96
## [9898,] 100   97
## [9899,] 100   98
## [9900,] 100   99

nrow(permutaciones) # ?cuántas permutaciones?

## [1] 9900

# De acuerdo a La Fórmula
factorial(N) / factorial(N-n) # debe salir lo mismo que nrow(permutaciones)

## [1] 9900

```

11.4 con grupos de 3. Combinaciones de 50 alumnos

```
N <- 50
n <- 3

alumnos <- c(1:N)

combinaciones <- combinations(N, n, alumnos)
head(combinaciones) # Las primeras combinaciones

##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    1    2    4
## [3,]    1    2    5
## [4,]    1    2    6
## [5,]    1    2    7
## [6,]    1    2    8

tail(combinaciones) # Las últimas combinaciones

##      [,1] [,2] [,3]
## [19595,]  46   48   50
## [19596,]  46   49   50
## [19597,]  47   48   49
## [19598,]  47   48   50
## [19599,]  47   49   50
## [19600,]  48   49   50

nrow(combinaciones) # ¿cuántas combinaciones?
## [1] 19600

# De acuerdo a La Fórmula. Debe salir el mismo número que nrow(combinaciones)
factorial(N) / (factorial(n) * (factorial(N-n)))

## [1] 19600
```

11.5 con grupos de 3. Permutaciones de 50 alumnos

```
N <- 50
n <- 3

alumnos <- c(1:N)

permutaciones <- permutations(N, n, alumnos)
head(permutaciones) # Las primeras permutaciones

##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    1    2    4
## [3,]    1    2    5
## [4,]    1    2    6
```

```

## [5,]    1    2    7
## [6,]    1    2    8

tail(permuciones) # Las últimas permutaciones

## [,1] [,2] [,3]
## [117595,] 50   49   43
## [117596,] 50   49   44
## [117597,] 50   49   45
## [117598,] 50   49   46
## [117599,] 50   49   47
## [117600,] 50   49   48

nrow(permuciones) # ¿cuántas permutaciones?

## [1] 117600

# De acuerdo a La Fórmula
factorial(N) / factorial(N-n) # debe salir lo mismo que nrow(permuciones)

## [1] 117600

```

El archivo markdown en donde se encutra este ejemplo está en el enlace siguiente:

<http://rpubs.com/rpizarro/519015>

12 Desarrollar diagramas de árbol en R

13 Elaborar práctica mediante teorema del binomio en R

14 Elaborar prácticas en R con los temas de Teoría de probabilidad.

Esta práctica tiene la finalidad de mostrar aspectos elementales de probabilidad en R.

14.1 Conceptos

REQUERIMIENTOS BÁSICOS PARA LA ASIGNACIÓN DE PROBABILIDADES

1. La probabilidad asignada a cada resultado experimental debe estar entre 0 y 1, inclusive. Si denota con E_i el i -ésimo resultado experimental y con $P(E_i)$ su probabilidad, entonces exprese este requerimiento como

$$0 \leq P(E_i) \leq 1 \text{ para toda } i$$

2. La suma de las probabilidades de los resultados experimentales debe ser igual a 1.0. Para resultados experimentales n escriba este requerimiento como

$$P(E_1) + P(E_2) + \cdots + P(E_n) = 1$$

El método de frecuencia relativa para la asignación de probabilidades es el más conveniente cuando existen datos para estimar la proporción de veces que se presentarán los resultados si el experimento se repite muchas veces.

La frecuencia relativa es la proporción de la frecuencia absoluta sobre el total de los datos.

Complemento. Dado un evento A, el complemento de A se define como el evento que consta de todos los puntos muestrales que no están en A. El complemento de A se denota A^c .

UNIÓN DE DOS EVENTOS La unión de A y B es el evento que contiene todos los puntos muestrales que pertenecen a A o a B o a ambos. La unión se denota $A \cup B$.

INTERSECCIÓN DE DOS EVENTOS Dados dos eventos A y B, la intersección de A y B es el evento que contiene los puntos muestrales que pertenecen tanto a A como a B.

La probabilidad condicional y se expresa $P(A | B)$. La notación $|$ indica que se está considerando la probabilidad del evento A dada la condición de que el evento B ha ocurrido. Por tanto, la notación $P(A | B)$ se lee “la probabilidad de A dado B”.

14.2 Una introducción a la teoría de probabilidad

14.2.1 La población

```
# Probabilidades  
poblacion <- c(1,2,3,4,5,6,7,8,9)  
poblacion  
  
## [1] 1 2 3 4 5 6 7 8 9
```

14.2.2 Determinar subconjuntos de una población

- A Elementos MENORES O IGUALES A 3
- B Elementos MAYORES O IGUALES A 2 Y MENORES O IGUALES A 6

```
A <- poblacion <=3  
B <- poblacion >= 2 & poblacion <=6  
  
A  
  
## [1] TRUE TRUE TRUE FALSE FALSE FALSE FALSE FALSE FALSE  
  
B  
  
## [1] FALSE TRUE TRUE TRUE TRUE TRUE FALSE FALSE FALSE
```

14.2.2.1 Los elementos de los subconjuntos

```
A <- which(poblacion <=3)  
B <- which(poblacion >= 2 & poblacion <=6)  
  
A  
  
## [1] 1 2 3  
  
B  
  
## [1] 2 3 4 5 6
```

14.2.3 Operaciones con conjuntos y complementos

```
AUB <- union(A, B)  
AIB <- intersect(A, B)  
  
AUB  
  
## [1] 1 2 3 4 5 6  
  
  
AIB  
  
## [1] 2 3
```

```

ComplA <- setdiff(poblacion, A) # Los que no están en A, es el complemento
ComplB <- setdiff(poblacion, B) # Los que no están en B, es el complemento

ComplA
## [1] 4 5 6 7 8 9

ComplB
## [1] 1 7 8 9

```

14.2.4 Probabilidades por la frecuencia

```

PA <- length(A) / length(poblacion) # Frecuencia / n
PB <- length(B) / length(poblacion) # Frecuencia / n

PA
## [1] 0.3333333

PB
## [1] 0.5555556

PAUB <- length(AUB) / length(poblacion)
PAIB <- length(AIB) / length(poblacion)

PAUB
## [1] 0.6666667

PAIB
## [1] 0.2222222

PComplA <- length(ComplA) / length(poblacion)
PComplB <- length(ComplB) / length(poblacion)
PComplA
## [1] 0.6666667

PComplB
## [1] 0.4444444

```

14.2.4.1 Probabilidad condicional

```

PACondDadaB <- PAIB / PB    #  $P(A|B)$ 
PBCondDadaA <- PAIB / PA    #  $P(B|A)$ 

PACondDadaB
## [1] 0.4

PBCondDadaA

```

```
## [1] 0.6666667
```

El archivo markdown en el servicio rpubs, que muestra un cálculo de probabilidades en R es el siguiente: <http://rpubs.com/rpizarro/519044>

Otro buen enlace que se recomienda observar sobre probabilidades es el siguiente:

<http://rpubs.com/rpizarro/519050>

14.3 Simulando. Teorema de Bayes

En el estudio de la probabilidad condicional vio que revisar las probabilidades cuando se obtiene más información es parte importante del análisis de probabilidades. Por lo general, se suele iniciar el análisis con una estimación de probabilidad inicial o probabilidad previa de los eventos que interesan. Después, de fuentes como una muestra, una información especial o una prueba del producto, se obtiene más información sobre estos eventos. Dada esta nueva información, se modifican o revisan los valores de probabilidad mediante el cálculo de probabilidades revisadas a las que se les conoce como probabilidades posteriores.

El teorema de Bayes es un medio para calcular estas probabilidades

14.3.1 Un conjunto de personas de una empresa X

```
personas <- c("Ingeniero", "Economista", "Otros")  
personas
```

```
## [1] "Ingeniero" "Economista" "Otros"
```

- Dentro de una empresa X, el 20 % son Ingenieros, el 20% son Economistas y el resto tienen otra profesión

```
son <- c(0.20, 0.20, 0.60)  
son
```

```
## [1] 0.2 0.2 0.6
```

- Dentro de la empresa, el 75% de los Ingenieros son Directivos
- Dentro de la empresa, el 50% de los Economistas son Directivos
- Dentro de la empresa, el 20% de otras profesiones son Directivos

```
sondirectivos <- c(0.75, 0.50, 0.20)
sondirectivos
## [1] 0.75 0.50 0.20
```

- ¿Cuál es la probabilidad de que una persona sea INGENIERO y ocupe un puesto DIRECTIVO?.
- Aplicando LEY MULTIPLICACIÓN DE PROBABILIDADES PARA EVENTOS INDEPENDIENTES

```
prob.son.directivos <- (son * sondirectivos)
prob.son.directivos
## [1] 0.15 0.10 0.12
```

- Determinando la probabilidad Total de que sean directivos cualquier persona

```
suma <- sum(son * sondirectivos)
suma
## [1] 0.37
```

- Aplicando Teorema de Bayes conforme a Fórmula

```
probabilidadBayes <- prob.son.directivos / suma
probabilidadBayes
## [1] 0.4054054 0.2702703 0.3243243
personas
## [1] "Ingeniero" "Economista" "Otros"
```

- La probabilidad de que un Directivo sea Ingeniero es del:

```
paste(round(probabilidadBayes[1] * 100,2), "%")
## [1] "40.54 %"
```

15 Elaborar prácticas para determinar valores de variables aleatorias discretas y continuas

15.1 Conceptos

Una variable aleatoria es una descripción numérica del resultado de un experimento.

A una variable aleatoria que asuma ya sea un número finito de valores o una sucesión infinita de valores tales como 0, 1, 2, . . . , se le llama variable aleatoria discreta.

A una variable que puede tomar cualquier valor numérico dentro de un intervalo o colección de intervalos se le llama variable aleatoria continua. Los resultados experimentales basados en escalas de medición tales como tiempo, peso, distancia y temperatura pueden ser descritos por variables aleatorias continuas.

La distribución de probabilidad de una variable aleatoria describe cómo se distribuyen las probabilidades entre los valores de la variable aleatoria. En el caso de una variable aleatoria discreta x , la distribución de probabilidad está definida por una función de probabilidad, denotada por $f(x)$. La función de probabilidad da la probabilidad de cada valor de la variable aleatoria.

CONDICIONES REQUERIDAS PARA UNA FUNCIÓN DE PROBABILIDAD DISCRETA

$$f(x) \geq 0$$

$$\sum f(x) = 1$$

FUNCIÓN DE PROBABILIDAD UNIFORME DISCRETA

$$f(x) = 1/n$$

donde

n = número de valores que puede tomar la variable aleatoria.

15.2 Distribucion de probabilidad uniforme. Caso licitaciones

Objetivo: Determinar probabilidades bajo la distribución uniforme

Al estudiar licitaciones de embarque, una empresa dedicada a la fabricación de circuitos impresos, encuentra que los contratos nacionales tienen licitaciones bajas distribuidas uniformemente entre 20 y 25 unidades (en miles de dólares).

15.3 Probabilidad de que la baja licitación de embarque del próximo contrato nacional:

- a) Sea inferior a 22 000 dólares.
- b) Rebase los 24 000 dólares.
- c) $P(X < x) = 1/5$.
- d) $P(X > x) = 2/5$.

15.4 a) Sea inferior a 22 000 dólares.

- Utilizando el dunif y sumando la 20, 21, y 22
- Primero se observa como trabajan las funciones dunif(), punif(), qunif() y runif() respectivamente.
- `dunif(x, min = 0, max = 1, log = FALSE)`
- `punif(q, min = 0, max = 1, lower.tail = TRUE, log.p = FALSE)`
- `qunif(p, min = 0, max = 1, lower.tail = TRUE, log.p = FALSE)`
- `runif(n, min = 0, max = 1)`. Esta para generar números aleatorios conforme a distribución de probabilidad uniforme

```
dunif(x=20:25, min = 20, max = 25) # 0.2 a partir de 20 a 25
## [1] 0.2 0.2 0.2 0.2 0.2 0.2
punif(q=20:25, min = 20, max = 25) # sumando Las probabilidades para dar
1
## [1] 0.0 0.2 0.4 0.6 0.8 1.0
qunif(p = 0.20, min=20, max=25) # Da La cantidad conforme a La probabilidad proporcionada
## [1] 21
```

Determinando los que están por debajo de 22

```
punif(q=22, min = 20, max = 25, lower.tail = TRUE)
## [1] 0.4
# Lower.tail = TRUE: significa Los que sean menores o iguales al valor de q, de lo contrario Los que sean mayores, cola hacia la izquierda o derecha
# Lower.tail Logical; if TRUE (default), probabilities are P[X ??? x], otherwise, P[X > x].
punif(q=20, min = 20, max = 25, lower.tail = TRUE)
```

```
## [1] 0
```

Sea inferior a 22 000 dólares.

- Utilizando *dunif()*,
- sumando distribuciones 21, 22 porque son inferiores o iguales a 22

```
suma <- dunif(x=21, 20, 25) + dunif(x=22, 20, 25) # Sin contar la x=20  
suma
```

```
## [1] 0.4
```

“Por lo tanto, la probabilidad de que la baja licitación de embarque del próximo contrato nacional sea inferior a 22 000 dólares es: 0.4.”

- mismo inciso a)
- Utilizando *punif()*,
- *punif(q, min = 0, max = 1, lower.tail = TRUE, log.p = FALSE)*
- Ya genera automáticamente la suma

```
suma <- punif(22, min=20, max=25, lower.tail=T)  
suma
```

```
## [1] 0.4
```

15.5 b) Rebase los 24 000 dólares.

- Utilizando *dunif()*
- determinar los que sean menores o iguales a 24 y resta a 1 para determinar el complemento, o sea que rebase los 24

```
suma <- dunif(x=21, 20, 25) + dunif(x=22, 20, 25) + dunif(x=23, 20, 25) +  
dunif(x=24, 20, 25) # soin contar x=20
```

```
suma
```

```
## [1] 0.8
```

```
rebase24 <- 1 - suma # p = 1 - p , conforme a complemento f?rmula  
rebase24
```

```
## [1] 0.2
```

Por lo tanto, la probabilidad de que la baja licitación de embarque del próximo contrato nacional rebase los 24 000 dólares es: 0.2.

Mismo inciso b) Rebase los 24 000 dólares.

- Utilizando *punif()*
- cambianos el lower.tail = FALSE , para que saque los que rebasen 24000, o cola hacia la derecha

```
rebasa24 <-punif(24, min=20, max=25, lower.tail=F)
rebase24
## [1] 0.2
```

15.6 c) $P(X < x) = 1/5.$

- para el inciso c) y d) utilizaremos la función *qunif()*
- Necesitamos obtener el valor de x (Dólares) para satisfacer: $P(X < x) = 1/5,$ empleamos para tal propósito, la función de cuantiles con el área de cola hacia la izquierda:

```
qunif(1/5, min=20, max=25, lower.tail=F)
## [1] 24
```

Por lo tanto, la cantidad de dólares que sea inferior para obtener una probabilidad de 1/5 son: 21 000 dólares.

15.7 d) $P(X > x) = 2/5.$

Necesitamos obtener el valor de x (Dólares) para satisfacer: $P(X > x) = 2/5,$ empleamos para tal propósito, la función de cuartiles con el área de cola hacia la derecha:

```
qunif(2/5, min=20, max=25, lower.tail=F)
## [1] 23
```

Por lo tanto, la cantidad de dólares que se debe rebasar para obtener una probabilidad de 2/5 son: 23 000 dólares.

15.8 Generando números aleatorios conforme distribución de probabilidad uniforme con *runif()*

```
# Generar números entre 22 y 24 conforme a prob. uniforme
runif(n=1, min=22, max=24)
## [1] 22.89263
runif(n=1, min=22, max=24)
```

```
## [1] 23.81796
runif(n=1, min=22, max=24)
## [1] 22.6214
# Generar 5 n?mero de aleatorios, al mismo tiempo, conforme a prob. uniforme
runif(n=5, min=22, max=24)
## [1] 22.92038 22.06305 22.85248 22.65752 23.18259
```

El ejercicio de probabilidad uniforme en markdown en el servicio rpubs, se encuentra en el siguiente enlace:

<http://rpubs.com/rpizarro/519233>

16 Prácticas de distribuciones de probabilidad en R

16.1 Distribución binomial

La distribución de probabilidad binomial es una distribución de probabilidad que tiene muchas aplicaciones. Está relacionada con un experimento de pasos múltiples al que se le llama experimento binomial.

PROPIEDADES DE UN EXPERIMENTO BINOMIAL

1. El experimento consiste en una serie de n ensayos idénticos.
2. En cada ensayo hay dos resultados posibles. A uno de estos resultados se le llama *éxito* y al otro se le llama *fracaso*.
3. La probabilidad de éxito, que se denota p , no cambia de un ensayo a otro. Por ende, la probabilidad de fracaso, que se denota $1 - p$, tampoco cambia de un ensayo a otro.
4. Los ensayos son independientes.

FUNCIÓN DE PROBABILIDAD BINOMIAL

$$f(x) = \binom{n}{x} p^x (1-p)^{n-x}$$

donde

$f(x)$ = probabilidad de x éxitos en n ensayos

n = número de ensayos

$$\binom{n}{x} = \frac{n!}{x!(n-x)!}$$

p = probabilidad de un éxito en cualquiera de los ensayos

$1 - p$ = probabilidad de un fracaso en cualquiera de los ensayos

En un experimento binomial lo que interesa es el número de éxitos en n ensayos. Si x denota el número de éxitos en n ensayos, es claro que x tomará los valores 0, 1, 2, 3, ..., n . Dado que el número de estos valores es finito, x es una variable aleatoria discreta. La distribución de probabilidad correspondiente a esta variable aleatoria se le llama distribución de probabilidad binomial.

16.2 Probabilidad binomial. Caso lectores de novela

Objetivo. Determinar la probabilidad binomial de lectores de novela

Descripción:

La última novela de un autor ha tenido un gran éxito, hasta el punto de que el 80% de los lectores ya la han leído. Un grupo de 4 amigos son aficionados a la lectura:

- 1 ¿Cuál es la probabilidad de que en el grupo, hayan leído la novela 2 personas?
- 2 ¿y Cómo máximo 2?. Acumulado hasta 2, incluye: 0, 1 y 2

16.3 Valores de los parámetros

```
n <- 4
p <- 0.80 # EXITO
q <- (1 - p) # 0.2. # FRACASO
x <- 2 # Variable discreta a determinar su probabilidad
```

16.3.1 1 ¿Cuál es la probabilidad de que en el grupo, hayan leído la novela 2 personas?. $x = 2$

```
dbinom(x,n, p)
## [1] 0.1536
```

16.3.2 2 ¿y cómo máximo 2?. Acumulado hasta 2, incluye: 0, 1 y 2

```
# sumar La probabilidad de 0 mas La prob. de 1 mas La prob de
# x menor o igual a 2

x <- 2; n <- 4; p <- 0.80 # todo en una misma linea
dbinom(0,n, p) + dbinom(1,n, p) + dbinom(2,n, p)

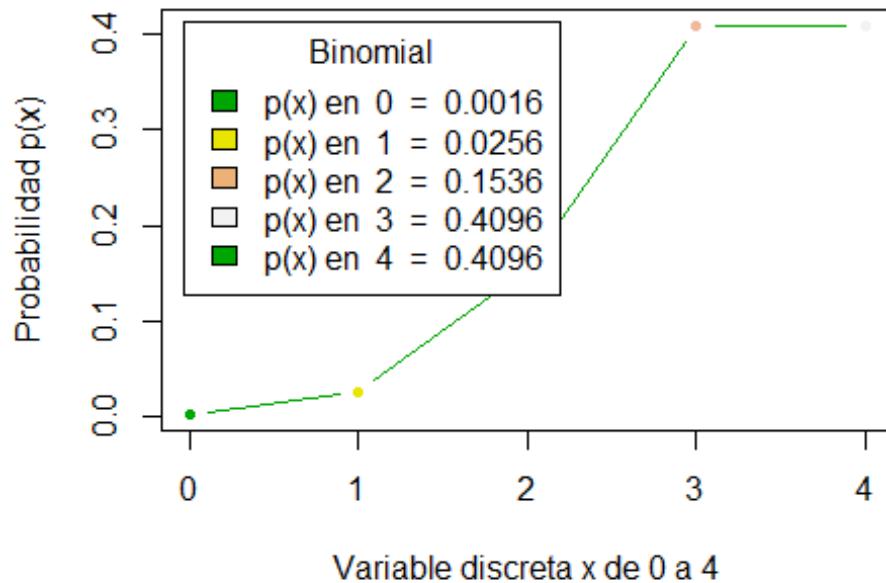
## [1] 0.1808

# Es Lo mismo que si utilizamos pbinom
pbinom(x, n , p) # Esta es probabilidad acumulada hasta el valor de x
## [1] 0.1808
```

16.3.3 Plot la probabilidades desde 0 hasta 4

```
plot (x=c(0:4), y=dbinom(0:4, n, p), type="b", col=terrain.colors(5), pch=20,
      xlab = 'Variable discreta x de 0 a 4', ylab = 'Probabilidad p(x)')

legend("topleft", inset=.03, title="Binomial",
       as.character(paste("p(x) en ", 0:4, " = ", round(dbinom(0:4, n, p),
4))), fill=terrain.colors(4), horiz=FALSE)
```



El ejercicio de probabilidad binomial. Caso lectores de novela se encuentra en
<http://rpubs.com/rpizarro/519239>

16.4 Distribución de probabilidad de Poisson

La distribución de probabilidad de Poisson suele emplearse para modelar las llegadas aleatorias a una línea de espera (fila).

PROPIEDADES DE UN EXPERIMENTO DE POISSON

1. La probabilidad de ocurrencia es la misma para cualesquiera dos intervalos de la misma magnitud.
2. La ocurrencia o no-ocurrencia en cualquier intervalo es independiente de la ocurrencia o no-ocurrencia en cualquier otro intervalo.

a **función de probabilidad de Poisson** se define mediante la ecuación

FUNCIÓN DE PROBABILIDAD DE POISSON

$$f(x) = \frac{\mu^x e^{-\mu}}{x!}$$

en donde

$f(x)$ = probabilidad de x ocurrencias en un intervalo
 μ = valor esperado o número medio de ocurrencias
en un intervalo
 $e = 2.71828$

16.5 Probabilidad de Poisson. Caso fallas de componentes electrónicos

Poisson. fallos de componentes electrónicos

Rubén Pizarro

8 de octubre de 2017

Objetivo: Poisson

Una empresa electrónica observa que el número de componentes que fallan antes de cumplir 100 horas de funcionamiento es una variable aleatoria de Poisson. Si el número promedio de estos fallos es 8,

- ¿cuál es la probabilidad de que falle 1 componente en 25 horas?
- ¿y de que fallen no más de 2 componentes en 50 horas?
- ¿cuál es la probabilidad de que fallen por lo menos 10 en 125 horas?

16.5.1 ¿cuál es la probabilidad de que falle un componente en 25 horas?

```
print ("8 fallos promedio en 100 hrs, entonces")
```

```
## [1] "8 fallos promedio en 100 hrs, entonces"
```

```
print ("Cuántos en 25 hrs, entonces")
```

```

## [1] "Cuántos en 25 hrs, entonces"
print ("2")
## [1] "2"

m <- 2
x <- 1

# Se busca  $p(x=1)$ , con valor de  $m = 2$ . Conforme Poisson

# Aplicando Poisson
# 1. ¿cuál es la probabilidad de que falle 1 componente en 25 horas?
prob <- dpois(x,m)
prob

## [1] 0.2706706

print (paste("Conclusión: Hay una probabilidad de ", round(prob, 4)* 100
, "de que fallen ", x, " componente ", " en 25 hrs."))
## [1] "Conclusión: Hay una probabilidad de 27.07 de que fallen 1 componente en 25 hrs."

```

16.5.2 ¿y de que fallen no más de 2 componentes en 50 horas?

```

print ("8 fallos promedio en 100 hrs, entonces")
## [1] "8 fallos promedio en 100 hrs, entonces"

print ("Cuántos en 50 hrs, entonces")
## [1] "Cuántos en 50 hrs, entonces"

print ("4")
## [1] "4"

m <- 4
x <- 2

# Se busca  $p(x \leq 2)$ , con valor de  $m = 4$ . Conforme Poisson

# Aplicando Poisson
# 2. ¿y de que fallen no más de 2 componentes en 50 horas?

dpois(0,4)
## [1] 0.01831564

dpois(1,4)
## [1] 0.07326256

```

```

dpois(2,4)
## [1] 0.1465251

sumatoria <- dpois(0,4) + dpois(1,4) + dpois(2,4)
sumatoria

## [1] 0.2381033

# La sumatoria es ...
prob <- ppois(0:2,m) # La sumatoria (acumulado) de las probabilidades de
# 0,1 y 2

prob # el último valor de prob, es lo mismo que sumatoria

## [1] 0.01831564 0.09157819 0.23810331

print (paste("Conclusión: Hay una probabilidad de ", round(sumatoria, 4)*
100 , "de que fallen no m?s de ", x, " componentes ", " en 50 hrs."))
## [1] "Conclusión: Hay una probabilidad de 23.81 de que fallen no m?s de
# 2 componentes en 50 hrs."

```

16.5.3 ¿cuál es la probabilidad de que fallen por lo menos 10 en 125 horas?

```

print ("8 fallos promedio en 100 hrs, entonces")
## [1] "8 fallos promedio en 100 hrs, entonces"

print ("Cuántos en 125 hrs, entonces")
## [1] "Cuántos en 125 hrs, entonces"

print (paste("8 * 125 / 100 = ", 8 * 125 / 100))
## [1] "8 * 125 / 100 = 10"

print ("por lo menos 10, entonces p(x >= 10), ")
## [1] "por lo menos 10, entonces p(x >= 10), "

# Se busca p(x < 10)), con valor de m = 10. Conforme Poisson
# Luego encontrar 1 - p(x < 10) para determinar del 10 en adelante
# es decir, por lo menos 10

m <- 10

sumatoria <- 0
for (i in 0:10) {

  prob <- dpois(i, m)
  print(prob)

  sumatoria <- sumatoria + prob
}

```

```

}

## [1] 4.539993e-05
## [1] 0.0004539993
## [1] 0.002269996
## [1] 0.007566655
## [1] 0.01891664
## [1] 0.03783327
## [1] 0.06305546
## [1] 0.09007923
## [1] 0.112599
## [1] 0.12511
## [1] 0.12511

sumatoria

## [1] 0.5830398

# La sumatoria es ...
prob <- ppois(0:10,m) # La sumatoria (acumulado) de las probabilidades de
# 0,1,2,3,4,5..., 9, 10.  $p(x \leq 10)$ , con valor  $m = 10$ 

round(prob,4) # el último valor de prob, es lo mismo que sumatoria

## [1] 0.0000 0.0005 0.0028 0.0103 0.0293 0.0671 0.1301 0.2202 0.3328 0.
4579
## [11] 0.5830

porlomenos10 <- 1 - sumatoria
porlomenos10

## [1] 0.4169602

```

16.5.4 Graficando las probabilidad $p(x \leq 10)$

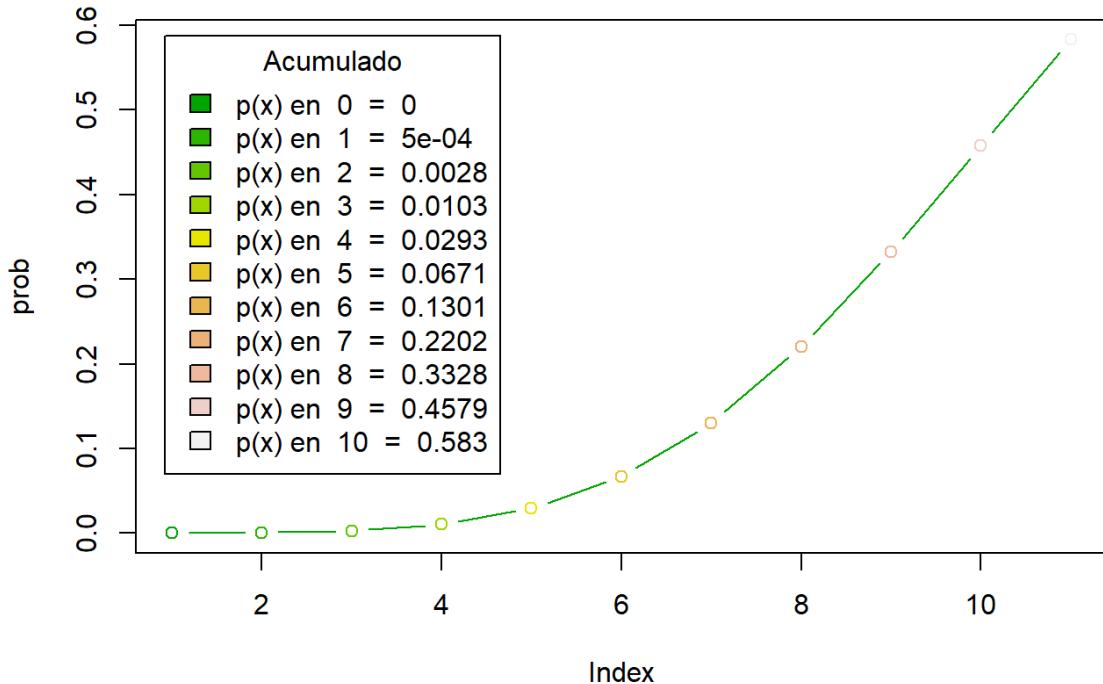
```

plot(prob, main=paste("La probabilidad de que fallen hasta 10 piezas en 1
25 horas es",round(sumatoria,4)), type="b", col=terrain.colors(11))

legend("topleft", inset=.03, title="Acumulado",
       as.character(paste("p(x) en ", 0:10, " = ", round(prob,4))), fill=t
errain.colors(11), horiz=FALSE)

```

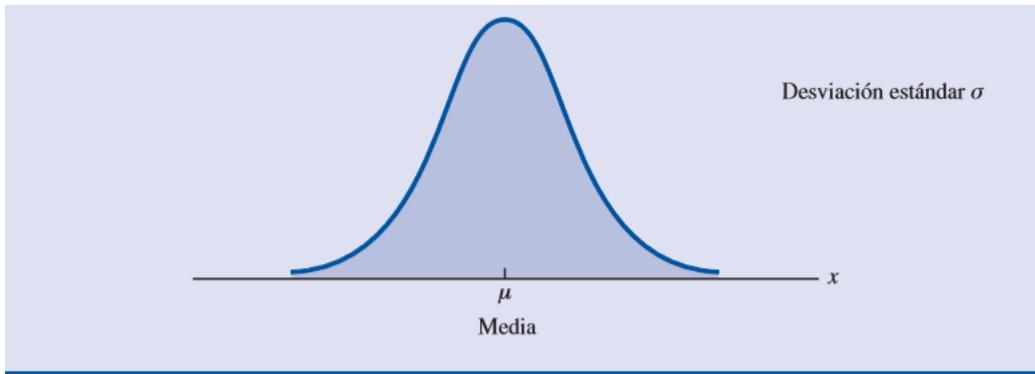
La probabilidad de que fallen hasta 10 piezas en 125 horas es 0.583



El caso de fallas de componentes electrónico usando la probabilidad de Poisson se encuentra en el siguiente enlace: <http://rpubs.com/rpizarro/519242>

16.6 Distribución de probabilidad normal

La distribución de probabilidad más usada para describir variables aleatorias continuas es la distribución de probabilidad normal. La distribución normal tiene gran cantidad de aplicaciones prácticas, en las cuales la variable aleatoria puede ser el peso o la estatura de las personas, puntuaciones de exámenes, resultados de mediciones científicas, precipitación pluvial u otras cantidades similares. La distribución normal también tiene una importante aplicación en inferencia estadística.



FUNCIÓN DE DENSIDAD DE PROBABILIDAD NORMAL

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

donde

$$\begin{aligned}\mu &= \text{media} \\ \sigma &= \text{desviación estándar} \\ \pi &= 3.14159 \\ e &= 2.71828\end{aligned}$$

16.7 Probabilidad Normal Bolsa de New York

El volumen de negociaciones en la Bolsa de Nueva York es más intenso en la primera media hora (en la mañana temprano) y la última media hora (al final de la tarde) de un día de trabajo. A continuación se presentan los volúmenes (en millones de acciones) de 13 días de enero y febrero.

214, 163, 265, 194, 180, 202, 198, 212, 201, 174, 171, 211, 211

La distribución de probabilidad de los volúmenes de negociaciones es aproximadamente normal.

- a) Calcule la media y la desviación estándar a usar como estimaciones de la media y de la desviación estándar de la población.
- b. ¿Cuál es la probabilidad de que, en un día elegido al azar, el volumen de negociaciones en la mañana temprano sea superior a 180 millones de acciones?

- c. ¿Cuál es la probabilidad de que, en un día elegido al azar, el volumen de negociaciones en la mañana temprano sea superior a 230 millones de acciones?
- d. ¿Cuántas acciones deberán ser negociadas para que el volumen de negociaciones en la mañana temprano de un día determinado pertenezca al 5% de los días de mayor movimiento?

16.7.1 Generando la muestra

```
'Primero: Generamos los valores de las acciones de 13 dias '
## [1] "Primero: Generamos los valores de las acciones de 13 dias "
muestra <- c(214, 163, 265, 194, 180, 202, 198, 212, 201, 174, 171, 211, 2
11)

muestra
## [1] 214 163 265 194 180 202 198 212 201 174 171 211 211
```

16.7.2 a) Calcule la media y la desviación estándar

```
media <- mean(muestra)
desv <- sd(muestra)

media
## [1] 199.6923

desv
## [1] 26.03966
```

16.7.3 b. ¿Cuál es la probabilidad de que, en un día elegido al azar, el volumen de negociaciones en la mañana temprano sea superior a 180 millones de acciones?

```
superior180 <- pnorm(180, mean = media, sd = desv, lower.tail = F)
superior180
## [1] 0.7752482
```

16.7.4 c) ¿Cuál es la probabilidad de que, en un día elegido al azar, el volumen de negociaciones en la mañana temprano sea superior a 230 millones de acciones?

```
superior230 <- pnorm(230, mean = media, sd = desv, lower.tail = F)
superior230
```

```
## [1] 0.1222313
```

16.7.5 d. ¿Cuántas acciones deberán ser negociadas para que el volumen de negociaciones en la mañana temprano de un día determinado pertenezca al 5% de los días de mayor movimiento?

```
# 5% = 5/100  
acciones5porc <- qnorm(5/100, mean = media, sd = desv)  
acciones5porc  
## [1] 156.8609
```

El caso de probabilidad Normal, Bolsa de New York, se encuentra en el siguiente enlace: <http://rpubs.com/rpizarro/519245>

16.8 Probabilidad normal. Caso alumnos de sistemas

Objetivo. Realizar análisis de probabilidad para con los promedios de alumnos de la sistemas computacionales de una Institución educativa superior

Descripción: De un conjunto de datos de los alumnos inscritos en carrera de sistemas computacionales de la institución, con no de control “modificado”. Se tienen datos (variables) tales como semestre que cursa, créditos aprobados, carga de créditos inscritos y el promedio que ser? la variables de interés.

- Se determinará la densidad de los datos conforme la media y la desviación estándar del promedio
- Se generará gráfica de gauss conforme probabilidad normal de acuerdo a su densidad y a la variable promedio
- Se obtendrán algunas probabilidades conforme la Distribución Normal.

16.8.1 Probabilidades conforme a Distribución Normal

- ¿Qué probabilidad existe de que el promedio de los alumnos de sistemas al final del semestre sea de 85 ?
- ¿Cual es la probabilidad de que el promedio de los alumnos de sistemas esté entre 86 y 90?
- ¿Será posible afirmar que el 20% o más, los alumnos de sistemas obtendrán un promedio superior a 90 ?

- ¿Cuál será el promedio del 10% de los alumnos de sistemas ?

16.8.2 Los datos, la media y la desviación estándard.

Se presentan los primeros y los últimos datos

```
datosalumnos <- read.csv("https://raw.githubusercontent.com/rpizarro/probabilidad-y-estadistica/master/practicas%20R/datos/Promedio%20alumnos%20de%2010y%2020semestre%20ISC%20del%20ITD/alumnos%20inscritos%20de%20e%20ISC%20SIN%20nocontrols.csv")
```

datosalumnos

##	No..Control	Semestre	Cr..Apr.	Carga	Promedio
## 1	10040001	8	154	23	83.24
## 2	10040002	8	193	32	88.19
## 3	10040003	1	NA	27	0.00
## 4	10040004	2	22	28	82.40
## 5	10040005	5	77	28	81.59
## 6	10040006	1	NA	27	0.00
## 7	10040007	2	17	28	78.00
## 8	10040008	1	NA	27	0.00
## 9	10040009	3	50	27	90.00
## 10	10040010	1	NA	27	0.00
## 11	10040011	7	160	34	85.80
## 12	10040012	9	227	8	87.61
## 13	10040013	1	NA	27	0.00
## 14	10040014	5	110	27	82.42
## 15	10040015	7	169	35	88.49
## 16	10040016	7	132	27	85.38
## 17	10040017	6	108	22	81.33
## 18	10040018	1	NA	27	0.00
## 19	10040019	1	NA	27	0.00
## 20	10040020	14	204	22	78.84
## 21	10040021	3	55	31	95.67
## 22	10040022	1	NA	27	0.00
## 23	10040023	3	55	31	93.17
## 24	10040024	11	187	21	84.88
## 25	10040025	2	22	28	78.40
## 26	10040026	1	NA	27	0.00
## 27	10040027	7	153	24	84.82
## 28	10040028	7	170	34	87.54
## 29	10040029	10	150	17	81.64
## 30	10040030	2	22	24	84.60
## 31	10040031	8	209	31	91.09
## 32	10040032	1	NA	27	0.00
## 33	10040033	11	202	23	83.68
## 34	10040034	9	230	10	88.58
## 35	10040035	6	98	34	84.09
## 36	10040036	2	22	24	83.40

```

## 437 10040437 1 NA 27 0.00
## 438 10040438 5 77 33 87.00
## 439 10040439 7 101 27 81.00
## 440 10040440 8 179 28 83.46
## 441 10040441 5 105 27 83.61
## 442 10040442 10 181 29 84.50
## 443 10040443 8 209 27 90.29
## 444 10040444 10 213 27 80.11
## 445 10040445 7 165 28 84.94
## 446 10040446 1 NA 27 0.00
## 447 10040447 7 146 23 86.13
## 448 10040448 7 164 30 91.17
## 449 10040449 6 101 27 84.00
## 450 10040450 1 NA 27 0.00
## 451 10040451 2 22 28 78.20
## 452 10040452 14 244 6 79.88
## 453 10040453 1 NA 27 0.00
## 454 10040454 3 55 27 83.58
## 455 10040455 5 81 27 82.72
## 456 10040456 10 236 4 86.43
## 457 10040457 11 137 22 76.70
## 458 10040458 1 NA 27 0.00
## 459 10040459 1 NA 27 0.00
## 460 10040460 4 72 28 79.44
## 461 10040461 2 22 28 80.40
## 462 10040462 11 230 5 84.26
## 463 10040463 4 54 19 77.83
## 464 10040464 5 110 27 86.29
## 465 10040465 6 100 27 79.32
## 466 10040466 6 95 27 80.62
## 467 10040467 3 55 27 87.92
## 468 10040468 7 175 34 90.03
## 469 10040469 11 202 29 82.43
## 470 10040470 2 22 23 83.20
## 471 10040471 3 50 27 86.00
## 472 10040472 4 73 28 82.44

promedios <- datosalumnos$Promedio[which(datosalumnos$Promedio > 0)] # Los que tienen promedio
media <- mean(promedios)

media

## [1] 84.21332

desv <- sd(promedios)

desv

## [1] 4.48788

```

16.8.3 Densidad de los datos conforme la media y la desviación estándar del promedio

Se presentan los primeros y los últimos datos

```
# La densidad Normal de cada promedio se genera conforme a La f?rmula ]
densidad <- dnorm(promedios, mean = media, sd = desv)
# densidad

tabla <- data.frame(promedios, densidad)
tabla

##      promedios     densidad
## 1      83.24 0.086827088
## 2      88.19 0.060030635
## 3      82.40 0.081925411
## 4      81.59 0.074933169
## 5      78.00 0.034092025
## 6      90.00 0.038712280
## 7      85.80 0.083507635
## 8      87.61 0.066754388
## 9      82.42 0.082072245
## 10     88.49 0.056451863
## 11     85.38 0.085939705
## 12     81.33 0.072316637
## 13     78.84 0.043409513
## 14     95.67 0.003417908
## 15     93.17 0.012132992
## 16     84.88 0.087917821
## 17     78.40 0.038416517
## 18     84.82 0.088084731
## 19     87.54 0.067538886
## 20     81.64 0.075418072
## 21     84.60 0.088563912
## 22     91.09 0.027481058
## 23     83.68 0.088267813
## 24     88.58 0.055372159
## 25     84.09 0.088859714
## 26     83.40 0.087445446
## 27     79.43 0.050372214
## 28     89.38 0.045821114
## 29     86.24 0.080275897
## 30     91.83 0.021057306
## 31     80.60 0.064284662
## 32     72.50 0.002948850
## 33     76.00 0.016656223
## 34     79.36 0.049535704
## 35     77.82 0.032224627
## 36     81.00 0.068793500
## 37     82.12 0.079730669
```

```

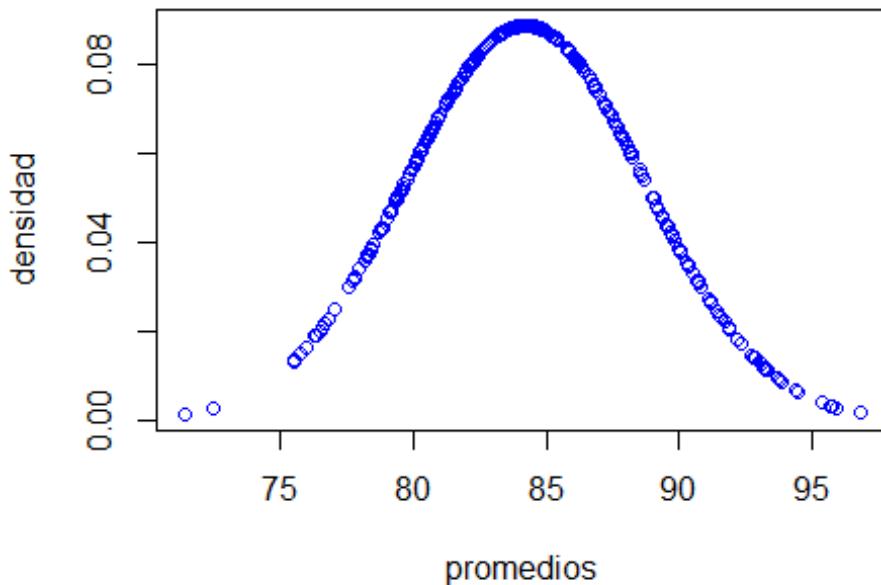
## 338    94.45 0.006593223
## 339    78.28 0.037095453
## 340    92.17 0.018463597
## 341    93.69 0.009563683
## 342    80.96 0.068353169
## 343    85.80 0.083507635
## 344    77.61 0.030113589
## 345    82.94 0.085386406
## 346    85.11 0.087136524
## 347    86.83 0.074997849
## 348    82.82 0.084710795
## 349    82.69 0.083917181
## 350    91.83 0.021057306
## 351    89.23 0.047592022
## 352    86.07 0.081602356
## 353    87.15 0.071761179
## 354    90.35 0.034902488
## 355    81.32 0.072213006
## 356    81.71 0.076086345
## 357    93.75 0.009296636
## 358    81.00 0.068793500
## 359    86.00 0.082120638
## 360    79.54 0.051689947
## 361    88.49 0.056451863
## 362    80.72 0.065660114
## 363    87.00 0.073306989
## 364    81.00 0.068793500
## 365    83.46 0.087649737
## 366    83.61 0.088093641
## 367    84.50 0.088712081
## 368    90.29 0.035543235
## 369    80.11 0.058525021
## 370    84.94 0.087735545
## 371    86.13 0.081145003
## 372    91.17 0.026736348
## 373    84.00 0.088792906
## 374    78.20 0.036225688
## 375    79.88 0.055772644
## 376    83.58 0.088012546
## 377    82.72 0.084105924
## 378    86.43 0.078685161
## 379    76.70 0.021890931
## 380    79.44 0.050491860
## 381    80.40 0.061957457
## 382    84.26 0.088888456
## 383    77.83 0.032326999
## 384    86.29 0.079868068
## 385    79.32 0.049058592
## 386    80.62 0.064515090
## 387    87.92 0.063203173

```

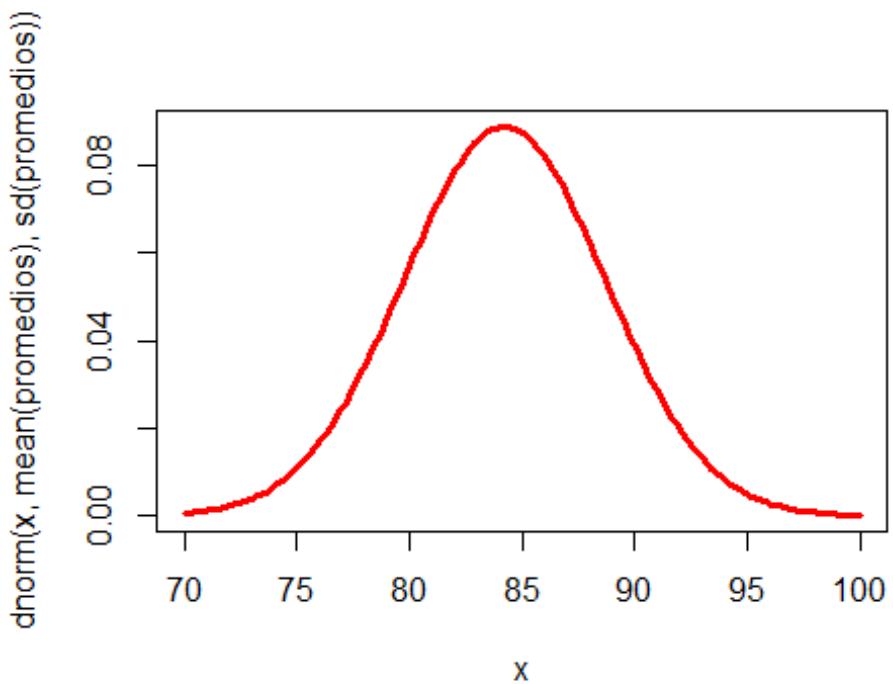
```
## 388      90.03 0.038379187
## 389      82.43 0.082145149
## 390      83.20 0.086655971
## 391      86.00 0.082120638
## 392      82.44 0.082217710
```

16.8.4 Gráfica de gauss conforme probabilidad normal de acuerdo a su densidad y a la variable promedio

```
plot(tabla, col="blue") ## promedio y densidad se ve salir campana de gauss
```



```
# Mejorando La campana de gauss
curve(dnorm(x, mean(promedios), sd(promedios)),
      xlim = c(70, 100), col = "red", lwd = 3)
```

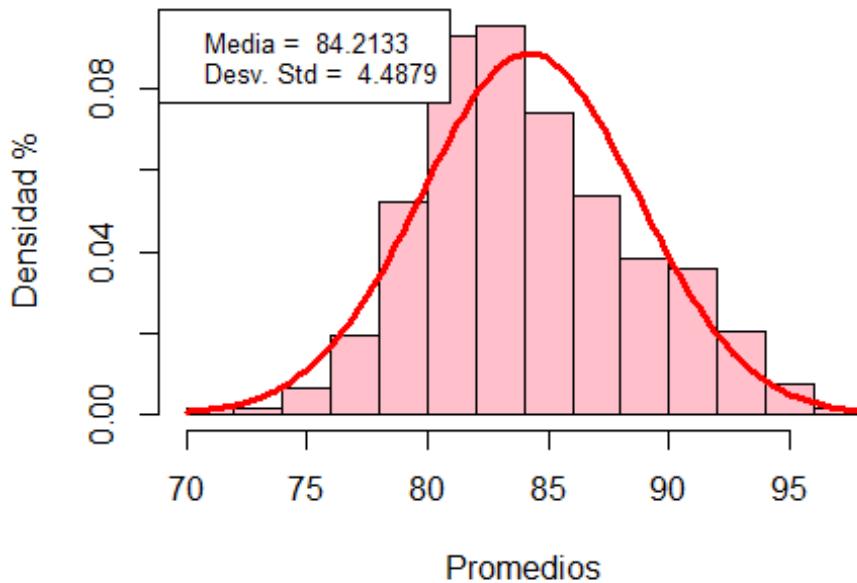


```
# Aun mejor, combinando histograma con gráfica de Gauss de la distribución Normal
hist(promedios, freq = F,
      ylab = "Densidad %",
      xlab = "Promedios", main = "Promedio de Alumnos de Sistemas", col =
      "Pink")
# Pink. Color de La prevenci?n del C?ncer

curve(dnorm(x, mean(promedios), sd(promedios)),
       col = "red", lwd = 3, add = TRUE)

legend("topleft", legend=c(paste("Media = ", round(media,4)), paste("Desv.
. Std = ", round(desv,4))), cex=0.8)
```

Promedio de Alumnos de Sistemas



16.8.5 ¿Qué probabilidad existe de que el promedio de los alumnos de sistemas al final del semestre sea de 85 ?

```
prob <- pnorm(85, mean = media, sd = desv)
prob <- round(prob, 4)
probporc <- prob * 100

paste("¿Qué probabilidad existe de que el promedio de los alumnos de sistemas al final del semestre sea de 85 ? es: ", probporc, " % ")

## [1] "¿Qué probabilidad existe de que el promedio de los alumnos de sistemas al final del semestre sea de 85 ? es: 56.96 % "
```

16.8.6 * ¿Cual es la probabilidad de que el promedio de los alumnos de sistemas esté entre 86 y 90?

```
prob <- pnorm(90, mean = media, sd = desv) - pnorm(86, mean = media, sd = desv)
prob <- round(prob, 4)

probporc <- prob * 100
paste("¿Cual es la probabilidad de que el promedio de los alumnos de sistemas esté entre 86 y 90? es: ", probporc, " % ")

## [1] "¿Cual es la probabilidad de que el promedio de los alumnos de sistemas esté entre 86 y 90? es: 24.66 % "
```

16.8.7 ¿Será posible afirmar que el 20% o más, los alumnos de sistemas obtendrán un promedio superior a 90 ?

Primero sacar probabilidad superior a 90 y luego comparar con el 20% para aceptar o rechazar la hipótesis

```
prob <- pnorm(90, mean = media, sd = desv, lower.tail = F)
prob <- round(prob, 4)

probporc <- prob * 100

if (prob >= 20) {
  paste("¿SI es posible afirmar que el 20% de los alumnos de sistemas obtendrán un promedio superior a 90, dado que la probabilidad SUPERIOR a 90 es: ", probporc, " % ")
} else {
  paste("¿NO es posible afirmar que el 20% de los alumnos de sistemas obtendrán un promedio superior a 90, dado que la probabilidad SUPERIOR a 90 es ", probporc, " % ")
}

## [1] "¿NO es posible afirmar que el 20% de los alumnos de sistemas obtendrán un promedio superior a 90, dado que la probabilidad SUPERIOR a 90 es 9.86 % "
```

16.8.8 ¿Cuál será el promedio del 10% de los alumnos de sistemas ?

qnorm()

```
promedio <- qnorm(0.10, mean = media, sd = desv)
paste("¿Cuál será el promedio del 10% de los alumnos de sistemas ?: ", promedio)

## [1] "¿Cuál será el promedio del 10% de los alumnos de sistemas ?: 78.4618670761607"
```

El ejercicio de probabilidad normal para alumnos de sistemas se encuentra en el siguiente enlace: <http://rpubs.com/rpizarro/519248>

17 Prácticas de regresión lineal

Un ejercicio recomendado de correlación y regresión lineal en R, se encuentra en el siguiente enlace: https://rpubs.com/Joaquin_AR/223351

18 Prácticas en R para muestreo, Estimaciones y pruebas de hipótesis

Una práctica sugerida para prueba de hipótesis se encuentra en el enlace:
<https://rpubs.com/acatania/399401>

Una práctica sugerida para la prueba T-student, de comparación de medias independientes y comparación de medias dependientes, se encuentra en el siguiente enlace: https://rpubs.com/Joaquin_AR/218467

Conclusiones

Como resultado se presentaron un conjunto de 18 prácticas que sugieren el uso del lenguaje de programación R y R Studio para fortalecer el aprendizaje de la estadística descriptiva y algunos aspectos de probabilidad y regresión.

El uso del lenguaje de programación “R” en la formación de la estadística descriptiva, favorece el pensamiento computacional siendo éste parte fundamental del perfil profesional del Ingeniero en sistemas Computacionales. (Pizarro, Calzada, & Valdez, 2018).

La justificación de argumentar esta aserción es que el uso del lenguaje de programación “R” y el servicio RPubs en alumnos, han sido factores para detonar la motivación, el interés, el gusto y la formación del pensamiento computacional en alumnos de la carrera de ISC del ITD, con ello contribuir en la formación profesional, innovar en procesos de enseñanza-aprendizaje y consolidar el perfil profesional del próximo profesionista en ésta área. (Pizarro, Calzada, & Valdez, 2018)

Con esta experiencia se aportan evidencias de que el uso del lenguaje de programación “R” como estrategia favorece el pensamiento computacional además, que la incorporación de tecnología en la nube ayuda a mejorar los procesos de enseñanza-aprendizaje en los temas de estadística descriptiva se cumple dado que por medio de “R” el alumno desarrolla destrezas de programación, análisis de datos, visualización de datos, generación de un portafolio de evidencias, toma de decisiones entre otros, que sin duda alguna son parte de los objetivos de aprendizaje del estudio de la estadística así como del Ingeniero en Sistemas Computacionales. (Pizarro, Calzada, & Valdez, 2018).

Para el maestro representa una alternativa como estrategia didáctica para la enseñanza de la estadística descriptiva y una oportunidad de innovar en su parte instruccional.