



---

**Universidad de Valladolid**  
**FACULTAD DE CIENCIAS**

**TRABAJO FIN DE GRADO**

Grado en Estadística

**Aplicaciones Shiny como herramienta para la  
Estadística en Enseñanzas Medias**

Autora:  
Elia Muñoz Alonso

Tutora:  
Teresa González Arteaga



## INDICE

<i>INDICE</i> .....	3
<i>RESUMEN</i> .....	4
<i>ABSTRACT</i> .....	4
1. <i>INTRODUCCIÓN</i> .....	5
2. <i>CONTENIDOS DE ESTADISTICA EN LAS ENSEÑANZAS MEDIAS</i> .....	7
2.1 <i>Contenidos de Estadística en las Matemáticas de la Educación Secundaria Obligatoria</i> .....	8
2.2 <i>Contenidos de Estadística en las Matemáticas de Bachillerato</i> .....	12
3. <i>HERRAMIENTAS UTILIZADAS: R, RSTUDIO Y SHINY</i> .....	16
3.1 <i>R</i> .....	16
3.2 <i>RStudio</i> .....	17
3.3 <i>Shiny</i> .....	18
4. <i>APLICACIONES SHINY PARA EL APOYO A LA ENSEÑANZA DE LA ESTADÍSTICA</i> .....	23
4.1 <i>Aplicación Shiny Descripción Univariante</i> .....	23
4.2 <i>Aplicación Shiny Descripción Bivariante</i> .....	30
4.3 <i>Aplicación Shiny Probabilidad</i> .....	35
5. <i>EJEMPLOS PRÁCTICOS</i> .....	45
6. <i>CONCLUSIONES</i> .....	47
<i>Apéndice A: Código Aplicación Shiny Descripción Univariante</i> .....	48
<i>Apéndice B: Código Aplicación Shiny Descripción Bivariante</i> .....	59
<i>Apéndice C: Código Aplicación Shiny Probabilidad</i> .....	78
<i>BIBLIOGRAFIA Y/O REFERENCIAS</i> .....	86
<i>FIGURAS</i> .....	89
<i>TABLAS</i> .....	90

## RESUMEN

La Estadística posee un papel primordial en el desarrollo de la sociedad actual. Por ello, la enseñanza de la Estadística ha venido adquiriendo un mayor relieve en los currículos de Matemáticas del que se presentaba tradicionalmente en las Enseñanzas Medias. Utilizar una herramienta informática como apoyo para el estudio de esta parte de las materias de Matemáticas ayuda a mejorar el aprendizaje, motivar a los alumnos, aumentar su interés e incitarlos a que se sientan atraídos por esta ciencia.

En el presente trabajo se ofrecen una serie de aplicaciones web interactivas, desarrolladas con el paquete estadístico R y específicamente Shiny. Estas aplicaciones recogen los contenidos de Estadística incluidos en las materias de Matemáticas en las Enseñanzas Medias. Además, se describe su uso de manera detallada para facilitar el manejo de estas aplicaciones a toda la comunidad educativa.

### *Palabras clave:*

R, Shiny, Matemáticas, Estadística, Enseñanzas Medias, Aplicaciones Web

---

## ABSTRACT

Statistics plays a fundamental role in the development of today's society. For this reason, the teaching of Statistics has acquired a greater prominence in the Mathematics curricula than was traditionally the case. Using a computer tool for the study of this part of the Mathematics subjects helps to improve learning, to motivate students and to increase their interest and it encourages them to feel attracted by this science.

The present dissertation offers several interactive web applications developed with the statistical package R, specifically with Shiny. They cover the contents of Statistics included in the subjects of Mathematics in Secondary Education. In addition, a detailed description of how those applications are used is presented. This part facilitates the handling of these applications to the entire educational community.

### *Keywords:*

R, Shiny, Mathematics, Statistics, Middle School, Web Application

## 1. INTRODUCCIÓN

La mayoría de las veces y a lo largo del tiempo como dijera Huntsberger: "La palabra estadística a menudo nos trae a la mente imágenes de números apilados en grandes arreglos y tablas, de volúmenes de cifras relativas a nacimientos, muertes, impuestos, poblaciones, ingresos, deudas, créditos y así sucesivamente". Para la mayoría de la gente Huntsberger tiene razón pues al momento de escuchar esta palabra estas son las imágenes que llegan a su cabeza.

La Estadística es mucho más que solo números apilados y gráficos bonitos. La Estadística es una ciencia muy útil e influyente en la mayoría de los campos del conocimiento. Hoy en día su aplicación se ha convertido en imprescindible en múltiples ámbitos de la vida científica y cotidiana. Un claro ejemplo de este protagonismo es la recurrente aparición en campos tan diversos como la Medicina, la Arquitectura, la investigación de mercados, la Meteorología, la Biología, la Política...

La ausencia de la Estadística conllevaría a un caos generalizado, dejando a las empresas y a la sociedad sin información vital a la hora de tomar decisiones.

La posesión de conocimientos de Estadística es un buen vehículo para alcanzar las capacidades de comunicación, tratamiento de la información, resolución de problemas, competencia digital y trabajo cooperativo y en grupo a las que se da gran importancia en los currículos actualmente vigentes en Enseñanzas Medias. Las aplicaciones estadísticas proporcionan una buena oportunidad de mostrar a los estudiantes la utilidad de las Matemáticas para entender la realidad que nos rodea.

Con la entrada en vigor de la LOMCE ha aumentado la carga de Estadística en las materias de Matemáticas de las Enseñanzas Medias. Además, los profesores de estas materias han mostrado mayor interés en el uso de herramientas informáticas que ayuden a solucionar problemas semejantes a los de la vida real. Esto sustenta la oportunidad para desarrollar los recursos docentes que se ofrecen en este trabajo fin de grado.

El objetivo del presente trabajo fin de grado es desarrollar y presentar aplicaciones web interactivas. Estas aplicaciones recogen parte de los contenidos de Estadística estudiados en la Educación Secundaria Obligatoria y el Bachillerato y ponen al alcance de sus estudiantes herramientas sencillas para entrenar su capacidad estadística.

Estas aplicaciones web interactivas se han creado con el paquete Shiny de R. Este paquete combina el poder computacional de R con la interactividad de la web moderna.

R es un software estadístico libre que se ha popularizado en el último tiempo. Esto ha contribuido en el desarrollo de interfaces de usuario como RStudio o complementos que ha incorporado en su repositorio como el paquete utilizado en este TFG, Shiny.

Este TFG aporta las aplicaciones web *Descripción Univariante*, *Descripción Bivariante* y *Probabilidad* desarrolladas con Shiny como recurso docente para la enseñanza de Estadística en dichos niveles de la educación. Además, se explican y se muestran estas aplicaciones para facilitar su comprensión.

A parte de todo lo anterior, el trabajo incluye tres videos grabados para mostrar los ejemplos prácticos posteriores sobre el uso de las aplicaciones creadas. Para la realización de estos videos se utilizan los siguientes ficheros de datos *EjemploSencillo1.txt*, *EjemploSencillo2.txt*, *FutbolSpain.txt* y *CrohnD.csv*.

A lo largo de este trabajo encontraremos las siguientes secciones. La Sección 2 hace referencia a los contenidos de Estadística recogidos en los currículos de Educación Secundaria Obligatoria y de Bachillerato. En la Sección 3 se realiza una breve introducción a las herramientas R y RStudio explicando sus características más globales. Además se realiza una descripción general sobre el paquete Shiny. En la Sección 4 se explican las tres aplicaciones desarrolladas con Shiny. Después, en la Sección 5 se presentan una serie de ejemplos prácticos de uso de las aplicaciones desarrolladas. En la Sección 6 contiene las conclusiones del trabajo. Por último, en los apéndices se incluye el código programado para las tres aplicaciones desarrolladas.

## 2. CONTENIDOS DE ESTADÍSTICA EN LAS ENSEÑANZAS MEDIAS

Las asignaturas de *Matemáticas* estudiadas tanto en la Educación Secundaria Obligatoria, ESO, y el Bachillerato recogen la parte de Estadística impartida en estas etapas educativas.

El currículo de estas asignaturas es regulado por las siguientes normativas, tanto estatales como autonómicas:

- **LEY ORGANICA 8/2013, de 9 de diciembre**, para la Mejora de la Calidad Educativa publicada en el Boletín Oficial del Estado (BOE) número 295 el 10 de diciembre de 2013.
- **REAL DECRETO 1105/2014, de 26 de diciembre**, por el que se establece el currículo básico de la Educación Secundaria Obligatoria y del Bachillerato publicado en el BOE número 3 el 3 de enero de 2015.
- **CORRECCIÓN de errores del Real Decreto 1105/2014, de 26 de diciembre**, por el que se establece el currículo básico de la Educación Secundaria Obligatoria y del Bachillerato publicado en el BOE número 104 el 1 de mayo de 2015.
- **ORDEN ECD/65/2015, de 21 de enero**, por la que se describen las relaciones entre las competencias, los contenidos y los criterios de evaluación de la Educación Primaria, la Educación Secundaria Obligatoria y de Bachillerato publicado en el BOE número 25 el 29 de enero de 2015.
- **REAL DECRETO 310/2016, de 29 de julio**, por el que se regulan las evaluaciones finales de Educación Secundaria Obligatoria y de Bachillerato publicado en el BOE número 183 el 30 de julio de 2016.
- **ORDEN EDU/362/2015, de 4 de mayo**, por la que desarrolla el currículo correspondiente a la Educación Secundaria Obligatoria y regula su implantación, evaluación y desarrollo en la Comunidad de Castilla y León publicada en el Boletín Oficial de Castilla y León (BOCYL) numero 86 el 8 de mayo de 2015.

Los contenidos de Estadística estudiados en Educación Secundaria Obligatoria (ESO) se encuentran dentro de las asignaturas llamadas *Matemáticas* en los dos primeros cursos mientras que en los dos últimos se da la opción de elegir entre *Matemáticas orientadas a las enseñanzas académicas* y *Matemáticas orientadas a las enseñanzas aplicadas*.

Los contenidos de Estadística incluidos en Bachillerato se encuentran recogidos dentro de las asignaturas de *Matemáticas aplicadas a las Ciencias Sociales I* y *II* para los dos cursos de la modalidad de Ciencias Sociales y *Matemáticas I* y *II* para los cursos de la modalidad de Ciencias.

## 2.1 Contenidos de Estadística en las Matemáticas de la Educación Secundaria Obligatoria

El currículo de Matemáticas en ESO divide estas asignaturas en cinco bloques:

- Bloque I: Procesos, métodos y actitudes en matemáticas
- Bloque II: Números y Álgebra
- Bloque III: Geometría
- Bloque IV: Funciones
- Bloque V: Estadística y probabilidad

Los contenidos de *Estadística* se encuentran dentro del último bloque. Por lo tanto, en el presente TFG, consideraremos los contenidos del bloque *Estadística y Probabilidad*.

En el primer curso, los contenidos recogidos en el bloque V son:

- Población e individuo. Muestra.
- Variables estadísticas. Variables cualitativas y cuantitativas discretas.
- Frecuencias absolutas y relativas.
- Organización en tablas de datos recogidos en una experiencia.
- Diagramas de barras y de sectores. Polígonos de frecuencias.
- Medidas de tendencia central.
- Fenómenos deterministas y aleatorios.
- Formulación de conjeturas sobre el comportamiento de fenómenos aleatorios sencillos y diseño de experiencias para su comprobación.
- Frecuencia relativa de un suceso y su aproximación a la probabilidad mediante simulación o experimentación.
- Sucesos elementales equiprobables.
- Espacio muestral en experimentos sencillos.

En el bloque V de la asignatura *Matemáticas* del segundo curso, sus contenidos son:

- Población e individuo. Muestra.
- Variables estadísticas. Variables cualitativas y cuantitativas discretas y continuas.
- Frecuencias absolutas y relativas.
- Organización en tablas de datos recogidos en una experiencia.
- Diagramas de sectores, de barras, histogramas y polígonos de frecuencias. Otros gráficos estadísticos provenientes de los medios de comunicación.
- Medidas de tendencia central.
- Medidas de dispersión.
- Iniciación a la hoja de cálculo.
- Fenómenos deterministas y aleatorios.



- Formulación de conjeturas sobre el comportamiento de fenómenos aleatorios sencillos y diseño de experiencias para su comprobación.
- Frecuencia relativa de un suceso y su aproximación a la probabilidad mediante simulación o experimentación.
- Sucesos elementales equiprobables y no equiprobables.
- Espacio muestral en experimentos sencillos. Tablas y diagramas de árbol sencillos.
- Cálculo de probabilidades mediante la regla de Laplace en experimentos sencillos.

En tercer y cuarto curso se da la opción de elegir entre dos tipos de Matemáticas. Las *Matemáticas orientadas a las enseñanzas académicas* donde se fortalecen tanto los aspectos teóricos como las aplicaciones prácticas en casos reales y las *Matemáticas orientadas a las enseñanzas aplicadas* que hacen mayor hincapié en la aplicación de los contenidos en contextos reales.

En las Matemáticas orientadas a las enseñanzas académicas del tercer curso se recogen los contenidos del bloque V:

- Fases y tareas de un estudio estadístico. Población, muestra.
- Variables estadísticas: cualitativas, cuantitativas discretas y continuas.
- Métodos de selección de una muestra estadística.
- Representatividad de una muestra.
- Frecuencias absolutas, relativas y acumuladas. Agrupación de datos en intervalos.
- Gráficas estadísticas.
- Parámetros de posición central y no central.
- Cálculo, interpretación y propiedades.
- Parámetros de dispersión.
- Diagrama de cajas y bigotes.
- Interpretación conjunta de la media y la desviación típica.
- Utilización de los medios tecnológicos adecuados para el análisis y la producción de información estadística.
- Uso de la calculadora científica, de la hoja de cálculo y de otros programas para hacer representaciones gráficas y calcular parámetros.
- Experiencias aleatorias simples y compuestas en casos sencillos.
- Sucesos y espacio muestral.
- Cálculo de probabilidades mediante la regla de Laplace.
- Diagramas de árbol sencillos y tablas. Regla del producto para contar casos.
- Utilización de la probabilidad para tomar decisiones fundamentales en diferentes contextos.
- Utilización de distintos programas para simular experimentos aleatorios.

El bloque V de las Matemáticas orientadas a las enseñanzas aplicadas del tercer curso recoge los contenidos:

- Fases y tareas de un estudio estadístico. Población, muestra.
- Variables estadísticas: cualitativas, cuantitativas discretas y continuas.
- Métodos de selección de una muestra estadística.
- Representatividad de una muestra.
- Frecuencias absolutas, relativas y acumuladas. Agrupación de datos en intervalos. Gráficas estadísticas.
- Parámetros de posición: central y no central.
- Cálculo, interpretación y propiedades.
- Parámetros de dispersión: rango, recorrido intercuartílico, varianza y desviación típica. Cálculo e interpretación.
- Diagrama de cajas y bigotes.
- Interpretación conjunta de la media y la desviación típica.
- Uso de la calculadora científica, de la hoja de cálculo y de otros programas para la representación gráfica, el cálculo de parámetros y su interpretación.

En las Matemáticas orientadas a las enseñanzas académicas del cuarto curso, los contenidos del bloque V son:

- Introducción a la combinatoria: combinaciones, variaciones y permutaciones.
- Cálculo de probabilidades mediante la regla de Laplace y otras técnicas de recuento.
- Probabilidad simple y compuesta.
- Sucesos dependientes e independientes.
- Experiencias aleatorias compuestas. Utilización de tablas de contingencia y diagramas de árbol para la asignación de probabilidades.
- Probabilidad condicionada.
- Utilización del vocabulario adecuado para describir y cuantificar situaciones relacionadas con el azar y la estadística.
- Identificación de las fases y tareas de un estudio estadístico.
- Gráficas estadísticas: Distintos tipos de gráficas. Análisis crítico de tablas y gráficas estadísticas en los medios de comunicación. Detección de falacias.
- Medidas de centralización y dispersión: interpretación, análisis y utilización.
- Comparación de distribuciones mediante el uso conjunto de medidas de posición y dispersión.
- Introducción a la estadística bidimensional. Dependencia estadística y dependencia funcional.
- Construcción e interpretación de diagramas de dispersión.
- Introducción a la correlación.

- Utilización de medios informáticos para calcular parámetros, representar variables unidimensionales y representar nubes de puntos.

El bloque V de las Matemáticas orientadas a las enseñanzas aplicadas de cuarto curso recoge los contenidos:

- Identificación de las fases y tareas de un estudio estadístico.
- Población y muestra.
- Graficas estadísticas: distintos tipos de graficas.
- Análisis crítico de tablas y graficas estadísticas en los medios de comunicación.
- Interpretación, análisis y utilidad de las medidas de centralización y dispersión.
- Comparación de distribuciones mediante el uso conjunto de medidas de posición y dispersión.
- Introducción a la estadística bidimensional. Dependencia estadística y dependencia funcional.
- Construcción e interpretación de diagramas de dispersión.
- Introducción a la correlación.
- Utilización de medios informáticos para el cálculo de parámetros, la representación de variables unidimensionales y la representación de nubes de puntos.
- Azar y probabilidad. Frecuencia relativa de un suceso aleatorio y probabilidad.
- Cálculo de probabilidades mediante la regla de Laplace.
- Probabilidad simple y compuesta.
- Sucesos dependientes e independientes. Pruebas o experimentos dependientes e independientes. Diagrama de árbol.
- Tablas de contingencia.
- Utilización de la hoja de cálculo para la simulación de experimentos aleatorios.

## 2.2 Contenidos de Estadística en las Matemáticas de Bachillerato

El currículo de Matemáticas en Bachillerato organiza estas asignaturas dependiendo de la modalidad a la que pertenezca dicha asignatura.

Dentro de la modalidad de Bachillerato en Ciencias Sociales se dividen las asignaturas existentes en cuatro bloques:

- Bloque I: Procesos, métodos y actitudes en matemáticas
- Bloque II: Números y Álgebra
- Bloque III: Análisis
- Bloque IV: Estadística y probabilidad

Los contenidos de Estadística se encuentran dentro del último bloque. Por lo tanto, en el presente TFG, consideraremos únicamente los contenidos del bloque *Estadística y Probabilidad*.

Los contenidos del bloque IV de la asignatura *Matemáticas Aplicadas a las Ciencias Sociales I* de primer curso de Bachillerato son:

- Estadística descriptiva bidimensional: Tablas de contingencia.
- Distribución conjunta y distribuciones marginales.
- Distribuciones condicionadas.
- Medias y desviaciones típicas marginales y condicionadas.
- Independencia de variables estadísticas.
- Dependencia de dos variables estadísticas.
- Representación gráfica: diagrama de dispersión (o nube de puntos).
- Dependencia lineal de dos variables estadísticas.
- Covarianza y correlación: Cálculo e interpretación del coeficiente de correlación lineal.
- Regresión lineal.
- Predicciones estadísticas y fiabilidad de las mismas.
- Coeficiente de determinación.
- Experimento aleatorio.
- Espacio muestral. Sucesos.
- Asignación de probabilidades a sucesos mediante la regla de Laplace y a partir de su frecuencia relativa.
- Axiomática de Kolmogorov.
- Aplicación de la combinatoria al cálculo de probabilidades.
- Experimentos simples y compuestos.
- Probabilidad condicionada.
- Dependencia e independencia de sucesos.
- Variables aleatorias discretas.
- Distribución de probabilidad.
- Parámetros: Media, varianza y desviación típica.
- Distribución binomial.
- Caracterización e identificación del modelo.

- Cálculo de probabilidades.
- Manejo de tablas.
- Variables aleatorias continuas.
- Función de densidad y de distribución.
- Interpretación de la media, varianza y desviación típica.
- Distribución normal.
- Tipificación de la distribución normal.
- Asignación de probabilidades en una distribución normal.
- Manejo de la tabla de la función de distribución normal estándar.
- Cálculo de probabilidades mediante la aproximación de la distribución binomial por la normal.
- Corrección por continuidad.

Los contenidos del bloque IV de la asignatura *Matemáticas aplicadas a las Ciencias Sociales II* de segundo curso de Bachillerato son:

- Profundización en la Teoría de la Probabilidad.
- Axiomática de Kolmogorov.
- Asignación de probabilidades a sucesos mediante la regla de Laplace y a partir de su frecuencia relativa.
- Experimentos simples y compuestos.
- Probabilidad condicionada.
- Dependencia e independencia de sucesos.
- Teoremas de la probabilidad total y de Bayes.
- Probabilidades a priori, a posteriori y verosimilitud de un suceso.
- Población y muestra.
- Métodos de selección de una muestra.
- Tamaño y representatividad de una muestra.
- Estadística paramétrica.
- Parámetros de una población y estadísticos obtenidos a partir de una muestra.
- Estimación puntual.
- Media y desviación típica de la media muestral y de la proporción muestral.
- Teorema central del límite.
- Distribución de probabilidad de la media muestral en una población normal.
- Distribución de probabilidad de la media muestral y de la proporción muestral en el caso de muestras grandes.
- Estimación por intervalos de confianza.
- Relación entre nivel de confianza, error máximo admisible y tamaño muestral.
- Intervalo de confianza para la media poblacional de una distribución normal con desviación típica conocida.
- Intervalo de confianza para la media poblacional de una

distribución de modelo desconocido y para la proporción en el caso de muestras grandes.

Las asignaturas de Matemáticas de la rama de Ciencias en el Bachillerato se organizan en cinco bloques:

- Bloque I: Procesos, Métodos y Actitudes Matemáticas
- Bloque II: Números y Álgebra
- Bloque III: Análisis
- Bloque IV: Geometría
- Bloque V: Estadística y Probabilidad

Los contenidos de Estadística se encuentran dentro del último bloque. Por lo tanto, en el presente TFG, consideraremos únicamente los contenidos del bloque *Estadística y Probabilidad*.

En la asignatura *Matemáticas I* de primer curso los contenidos del bloque IV recogidos son:

- Estadística descriptiva bidimensional: Tablas de contingencia.
- Distribución conjunta y distribuciones marginales.
- Medias y desviaciones típicas marginales.
- Distribuciones condicionadas.
- Independencia de variables estadísticas.
- Estudio de la dependencia de dos variables estadísticas.
- Representación gráfica: Nube de puntos.
- Dependencia lineal de dos variables estadísticas.
- Covarianza y correlación: Cálculo e interpretación del coeficiente de correlación lineal.
- Regresión lineal.
- Recta de regresión.
- Estimación.
- Predicciones estadísticas y fiabilidad de las mismas.

En la asignatura *Matemáticas II* del segundo curso los contenidos del bloque IV son:

- Experimento aleatorio.
- Espacio muestral.
- Sucesos.
- Asignación de probabilidades a sucesos mediante la regla de Laplace y a partir de su frecuencia relativa.
- Axiomática de Kolmogorov.
- Aplicación de la combinatoria al cálculo de probabilidades.
- Experimentos simples y compuestos.
- Probabilidad condicionada.
- Dependencia e independencia de sucesos.

- Teoremas de la probabilidad total y de Bayes.
- Probabilidades iniciales y finales y verosimilitud de un suceso.
- Variables aleatorias discretas.
- Distribución de probabilidad.
- Parámetros: Media, varianza y desviación típica.
- Distribución binomial.
- Caracterización e identificación del modelo.
- Tabla de la distribución binomial.
- Cálculo de probabilidades.
- Distribución normal.
- Tipificación de la distribución normal.
- Tabla de la función de distribución normal estándar.
- Asignación de probabilidades en una distribución normal.
- Cálculo de probabilidades mediante la aproximación de la distribución binomial por la normal.

### 3. HERRAMIENTAS UTILIZADAS: R, RSTUDIO Y SHINY

Las aplicaciones web dinámicas aportadas en el presente trabajo fin de grado se desarrollan por medio del programa *R*, con su entorno de desarrollo integrado *RStudio* y su paquete *Shiny*. Las versiones utilizadas para la programación hecha en este TFG son R 3.5.3, RStudio 1.1.463 y Shiny 1.2.0.

#### 3.1 R

R es el lenguaje, entorno, de programación más usado en la comunidad científica para computación estadística y gráficos bajo Licencia Publica General de GNU. R es un producto de código libre en el sentido mas ampliamente posible, también incluido en la filosofía Open Source.

R está basado en el lenguaje S, desarrollado por Bell Laboratories de AT&T R. A partir de S Robert Gentleman y Ross Ihaka de la Universidad de Auckland lo reimplementan en 1993. Actualmente, R es una herramienta estadística muy poderosa, estable, confiable y en constante actualización.

Este software cuenta con ciertas ventajas para su uso generalizado para el análisis de datos:

- Código y descarga libre y por tanto no hay que pagar para adquirirlo
- Técnicas estadísticas básicas, avanzadas y robustas
- Gráficos de alta calidad exportables
- Gran cantidad de información sobre sus funciones y paquetes
- Disponibilidad para todos los sistemas operativos
- Capacidad de trabajar con otros lenguajes
- Creación de aplicaciones web interactivas con el paquete Shiny, herramienta en la que nos centraremos en el presente TFG
- Disponibilidad de miles de paquetes que extienden sus funciones básicas

En las Enseñanzas Medias el presupuesto para herramientas informáticas es reducido. Por lo tanto, el aspecto económico es una ventaja a resaltar para utilizar R en la Estadística incluida en las asignaturas de estas enseñanzas.

La instalación de R es muy simple. En su web <https://www.r-project.org/>, *Figura 1*, basta con seleccionar la opción "Download CRAN". En la página que se abre a continuación hay que escoger el "mirror" de descarga y la versión que corresponde al sistema operativo deseado (Windows, Mac OS o Linux). Están disponibles "mirrors" en distintos países, por lo tanto, es conveniente elegir un "mirror" de una localización cercana.



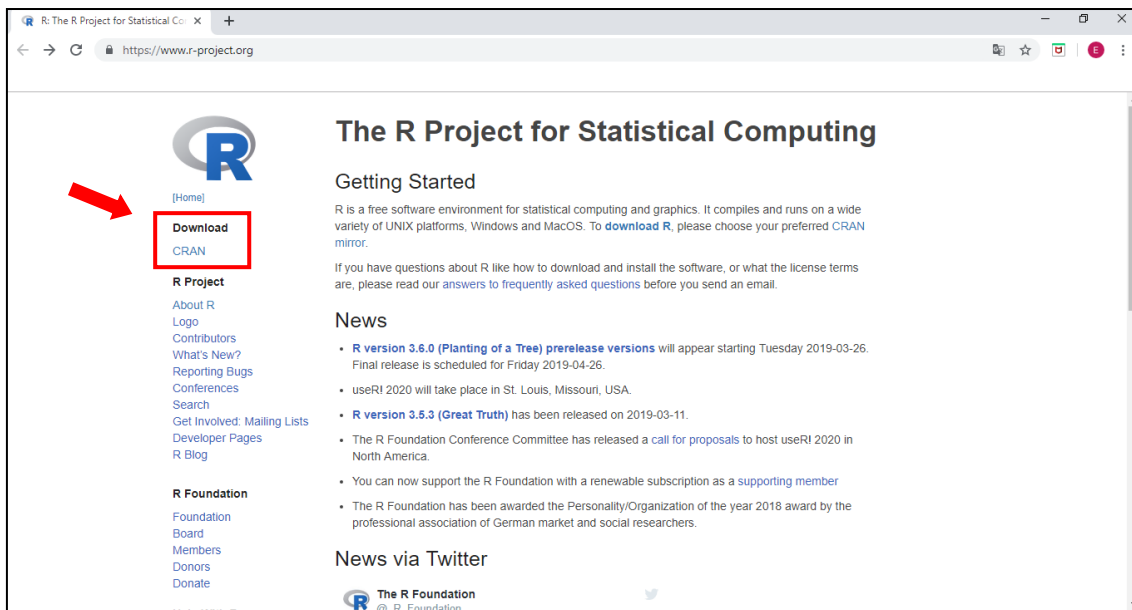


Figura 1: Pagina web principal de descarga R

En el TFG [“R como herramienta de apoyo a la enseñanza de Estadística en Bachillerato”](#) (Revueta, 2017) se explica detalladamente cómo realizar la instalación de R y sus paquetes.

## 3.2 RStudio

RStudio es un entorno de desarrollo integrado (IDE) para R, que al igual que éste es software libre. Está formado por una consola de ejecución, un editor de texto que acepta la ejecución directa y herramientas para la administración del área de trabajo. El objetivo es disponer de una herramienta potente que sea sencilla e intuitiva para crear un entorno “user friendly” para sus usuarios, pero que a su vez soporte los tratamientos requeridos para realizar análisis de calidad y dignos de confianza.

Una vez instalado R (R-project), se puede instalar RStudio. La instalación de RStudio se consigue desde la página oficial del programa <https://www.rstudio.com/> seleccionando “Download now” y a continuación “Download RStudio Desktop”. Elegir la versión del sistema operativo adecuada y seguir las instrucciones de instalación.

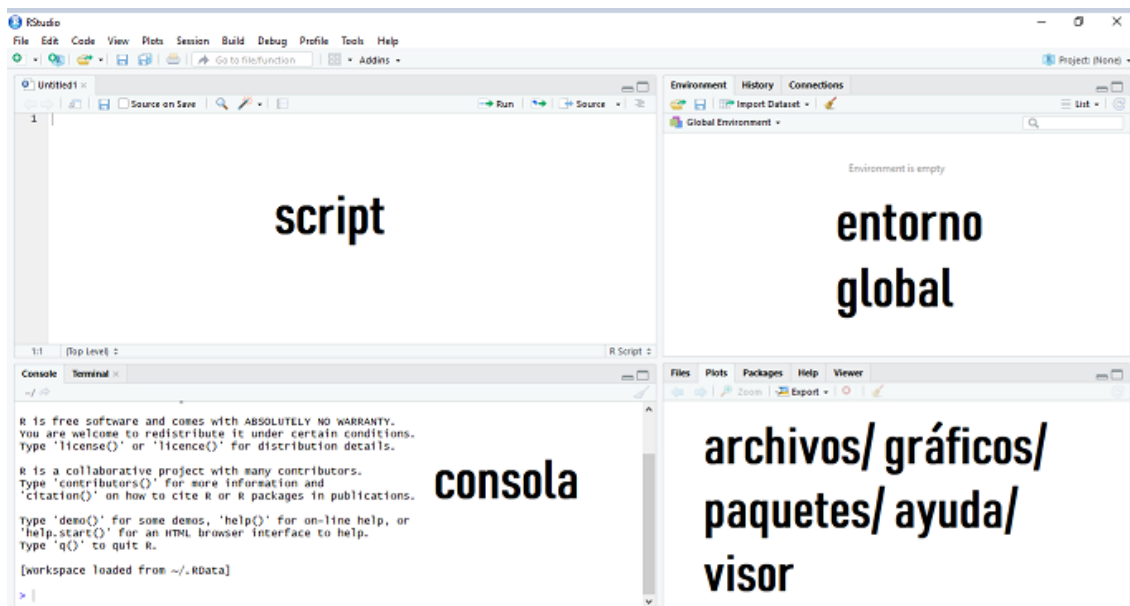


Figura 2: Ventana principal RStudio

Como se muestra en la *Figura 2*, RStudio es un entorno muy organizado ya que podemos observar al tiempo el editor de código fuente (Script), el espacio de trabajo (Consola), los objetos almacenados en memoria (Entorno global) y el directorio de trabajo, los gráficos que se generan, los paquetes para cargarlos e instalarlos directamente, la ayuda y un visor HTML en la ventana inferior izquierda.

Un aspecto a señalar es que RStudio no posee versión en castellano. Esto puede ser visto como un inconveniente para su uso en las Enseñanzas Medias.

### 3.3 Shiny

El software estadístico R cuenta desde 2012 en el repositorio CRAN de la página web del proyecto R con un paquete llamado *Shiny*. *Shiny* permite la construcción de aplicaciones web interactivas (apps) directamente desde R con un marco web elegante y poderoso sin necesidad de contar con conocimientos de HTML, CSS o Java Script.

La instalación de este paquete es similar a la del resto de paquetes de R. Se puede realizar a través de los menús de RStudio o ejecutar la siguiente orden en la consola `install.packages("Shiny", dependencies = T)`. Para más información sobre la instalación de paquetes se puede visitar ["R como herramienta de apoyo a la enseñanza de Estadística en Bachillerato"](#) (Revuelta, 2017).

Las aplicaciones web pueden tener un papel fundamental no solo en el ámbito docente sino también en el profesional o en investigación.

En las subsecciones siguientes se introducen brevemente los elementos fundamentales para entender las aplicaciones Shiny.

### 3.3.1 Estructura y funcionamiento de una aplicación Shiny

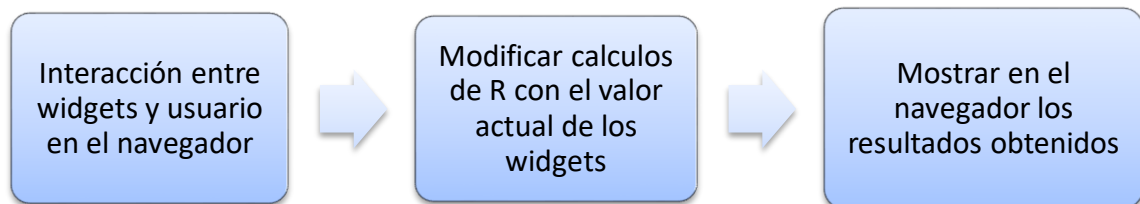
Cada aplicación Shiny se compone de dos partes: una página web que muestra la aplicación al usuario y un ordenador o servidor que la alimenta. El desarrollador de una aplicación Shiny escribe la interfaz de usuario, llamada *UI* en Shiny, y el código que alimenta la aplicación, llamado *SERVER* en Shiny.

El funcionamiento de las aplicaciones web desarrolladas con Shiny es de la misma forma que otras aplicaciones web, con la ventaja de que R y Shiny generan el código HTML que muestra los elementos en la página web y ofrece un modelo de interacción entre los componentes interactivo y dinámico.

Las aplicaciones desarrolladas con esta tecnología se pueden presentar de dos formas. La primera es con un único archivo `app.R` que reúna todo el código en un solo script y la segunda es creando dos archivos, `ui.R` y `server.R`, que separen la parte cliente de la parte servidora. Si están en un único script se encuentran las siguientes partes diferenciadas en el código:

- ***ui***: instrucciones que describen la interfaz de usuario. Es decir, controla el diseño y aspecto de la aplicación mediante una serie de comandos.
- ***server***: instrucciones de los componentes de R que se necesitan para construir los cálculos y gráficos necesarios.
- ***ShinyApp***: crea los objetos de la aplicación Shiny. Solo es necesario cuando se crea un único archivo `app.R`

Para el funcionamiento de estas aplicaciones Shiny podemos distinguir los siguientes pasos:



A continuación se van a describir algunos elementos claves para entender las aplicaciones Shiny: Programación reactiva, Widgets, construcción de interfaz y salida reactiva.

#### ***Programación reactiva***

Shiny está basado en la programación reactiva, es decir, enlaza los valores de entrada y los de salida.

Cuando una entrada cambia, el servidor reconstruye cada salida que depende de ella directa o indirectamente.

En este tipo de programación destaca el uso de valores que cambian con el tiempo y expresiones que registran esos cambios.

## Widgets

Un widget es un elemento web con el que el usuario puede interactuar enviando mensajes a la aplicación Shiny.

Shiny dispone de un conjunto de widgets pre-construidos, se pueden ver en la *Figura 3* y *Tabla 1*. El usuario interactúa con el widget para modificar los valores que toma la aplicación Shiny al evaluar los cálculos, y por lo tanto, reconstruir la solución a mostrar.

En la programación de los widgets se utilizan funciones widget. El primer argumento de cada función widget es el **<nombre>** del widget. Se puede acceder a su valor actual en la función **server** con **input\$<nombre>**.

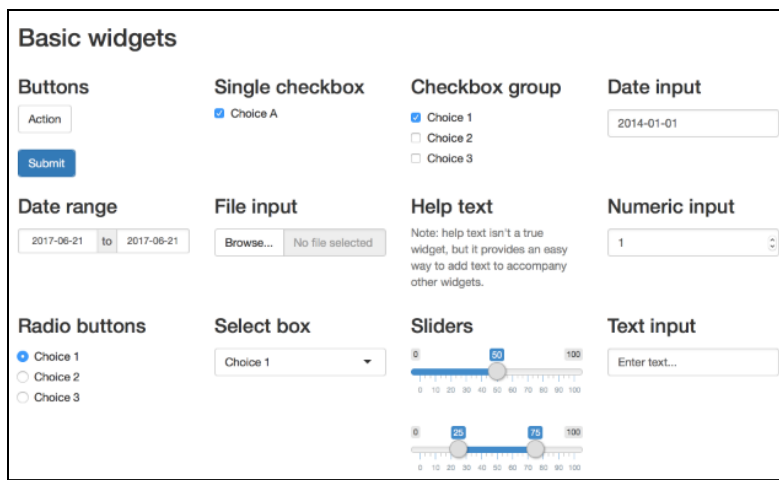


Figura 3: Widgets básicos de Shiny

Widget	Función	Argumentos comunes
<b>Botón de acción</b>	actionButton	inputId, label
<b>Casilla</b>	checkboxInput	inputId, label, value
<b>Grupo de casillas</b>	checkboxGroupInput	inputId, label, choices, selected
<b>Selección de fechas</b>	dateInput	InputId, label, value, min, max, format
<b>Selección rango de fechas</b>	dateRangeInput	inputId, label, start, end, min, max, format
<b>Subir archivo</b>	fileInput	inputId, label multiple
<b>Campo numérico</b>	numericInput	inputId, label, value, min, max, step
<b>Botón de selección</b>	radioButtons	inputId, label, choices, selected
<b>Casilla de selección</b>	selectInput	inputId, label, choices, selected, multiple
<b>Deslizador</b>	sliderInput	inputId, label, min, max, value, step
<b>Botón de envío</b>	submitButton	text
<b>Campo de texto</b>	textInput	inputId, label, value

Tabla 1: Widget básicos de Shiny

## Construcción de interfaz

Para empezar a crear una aplicación Shiny se crea un nuevo directorio en el que se guarda dentro un archivo app.R que contiene ui y server. Se recomienda que cada aplicación viva en su propio directorio único.

Para crear una primera aplicación el mínimo código posible que se necesita debe corresponderse con el script de la *Figura 4*:

The image shows a screenshot of an R script editor window titled 'app.R\*'. The editor contains the following R code:

```
1 library(shiny)
2
3 # Definición de UI
4 ui <- fluidPage(
5
6 )
7
8 # Función server
9 server <- function(input, output) {
10
11 }
12
13 # Correr la aplicación
14 shinyApp(ui, server)
```

The editor interface includes a toolbar with icons for navigation and execution, a 'Run App' button, and a status bar at the bottom showing the time '13:23' and the current context '(Top Level)'. The window title bar also indicates it is an 'R Script'.

Figura 4: Aplicación Shiny mínima

A partir de la aplicación mínima que se muestra en la *Figura 4* se irán añadiendo elementos hasta llegar a la aplicación final que se desea. Algunos de los elementos básicos son:

- Función fluidPage: crea ventana que se ajusta automáticamente a las dimensiones de la pantalla
  - o titlePanel: panel de títulos
  - o sidebarLayout: siempre toma los siguientes dos argumentos
    - sidebarPanel: barra lateral
    - mainPanel: panel principal
  - o Widgets
- Contenido HTML: mediante funciones Shiny paralelas a las etiquetas HTML5 comunes se añaden párrafos, encabezados, formatos de texto e imágenes entre otros.

## Salida reactiva

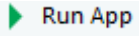
Para obtener una salida reactiva necesitamos un indicador o “reactivo” que haga que la salida depende de otra expresión. Esto es, cuando una entrada del usuario cambia, la aplicación renueva los cálculos realizados con esa entrada y por consiguiente produce una salida nueva.

Para crear estas salidas reactivas se necesitan dos pasos:

- Agregar un objeto R a la interfaz de usuario. Shiny cuenta con funciones que modifican los objetos R en salida para su interfaz.
- Proporcionar el código R para construir el objeto propiamente dicho. Este código para formar el objeto debe especificar el valor de un widget para que sea reactivo.

### 3.3.2 Compartir aplicaciones Shiny

Una vez creada una aplicación Shiny esta se puede compartir con otros usuarios de las dos formas siguientes:

- Como R scripts: solo funciona cuando el usuario cuenta con el software R. Se inicia la aplicación desde la sesión de R del usuario y se ejecuta con el botón  o ejecutando la siguiente orden en consola `library(shiny); runApp('app.R')` si nuestra aplicación se guarda en el fichero app.R.
- Como una página web: los usuarios pueden acceder como un navegador web por la aplicación a través de internet. Esto requiere que la aplicación Shiny específica esté disponible en un servidor público.

En este TFG compartiremos las aplicaciones Shiny creadas de la primera forma que se ha descrito. Además de los scripts que contienen el código de las aplicaciones Shiny, se compartirán varios ficheros de datos que sirven para realizar ejemplos prácticos y se utilizarán en la Sección 5.

## 4. APLICACIONES SHINY PARA EL APOYO A LA ENSEÑANZA DE LA ESTADÍSTICA

Para elaborar estas aplicaciones, en este TFG, se ha tenido en cuenta que los contenidos de Estadística enumerados en la Sección 2.

- Estadístico y gráficos que utilizan una sola variable para los cuales se ha desarrollado la aplicación Shiny Descripción Univariante.
- Estadísticos y gráficos que emplean dos variables para los cuales se ha desarrollado la aplicación Shiny Descripción Bivariante.
- Los contenidos de probabilidad en la aplicación Shiny Probabilidad.

En esta sección se explicara detalladamente cómo operar y la estructura de cada una de las aplicaciones Shiny desarrolladas.

### 4.1 Aplicación Shiny Descripción Univariante

La estadística descriptiva univariante se centra en la clasificación, presentación, descripción y análisis de una única variable. Para ello se utilizan tablas, gráficos y estadísticos resumen.

La aplicación desarrollada con Shiny para la descripción de una variable se llama Descripción Univariante y se ejecuta en RStudio con el fichero “**UnaVariable.R**” escribiendo la orden ***library(shiny); runApp('UnaVariable.R')*** en consola. El código del script *UnaVariable.R* se encuentra en el Apéndice A.

Esta aplicación está dividida en las 5 secciones o ventanas que se muestran en la *Figura 5*.

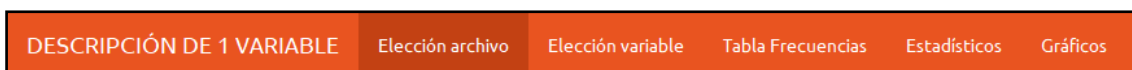


Figura 5: Encabezado Aplicación Shiny Descripción Univariante

Se va a pasar a describir detalladamente la estructura y los contenidos que se recogen en cada una de las ventanas.

## Elección de archivo

DESCRIPCIÓN DE 1 VARIABLE | Elección archivo | Elección variable | Tabla Frecuencias | Estadísticos | Gráficos

Lectura de fichero:  
Buscar fichero | Ningún archivo seleccionado

¿El archivo contiene el nombre de la variable?  
 Sí  
 No

Separador de variables:  
 Espacio  
 Tabulación  
 Coma  
 Punto y Coma

Formato:  
 Comilla Doble  
 Comilla simple  
 Nada

Número de filas a mostrar:  
10

### Fichero de datos

Para variables cualitativas se obtiene:  
Tabla de frecuencias  
Gráficos

Además, para variables numéricas se obtiene:  
Estadísticos de centro o localización  
Estadísticos de dispersión

Las dimensiones del fichero escogido son:  
Número de observaciones:   
Número de variables:

Primeros datos del fichero:

```
5648625324875585854
113787248194378129474
741328947328904732484
634127959932568256578
657891634578165895155
578758785688686515155
55553247853453245235
849235293453245234555
249578493865324959542
578758785688686515155
55553247853453245235
849235293453245234555
249578493865324959542
655786738496734896353
```

Figura 6: Aplicación Shiny Descripción Univariante, Ventana elección archivo

Esta ventana sirve para cargar el fichero de datos donde se encuentre la variable a analizar.

Los datos deben estar en un fichero de datos externo. Al seleccionar la opción “*Buscar fichero*” se abre el explorador de archivos y se busca el fichero deseado.

En la parte izquierda de la pantalla se configura la lectura del archivo de datos. Primero se selecciona el archivo al abrirse el explorador del ordenador. Después se completa la ventana para que la lectura de los datos sea correcta, ver *Figura 7*.

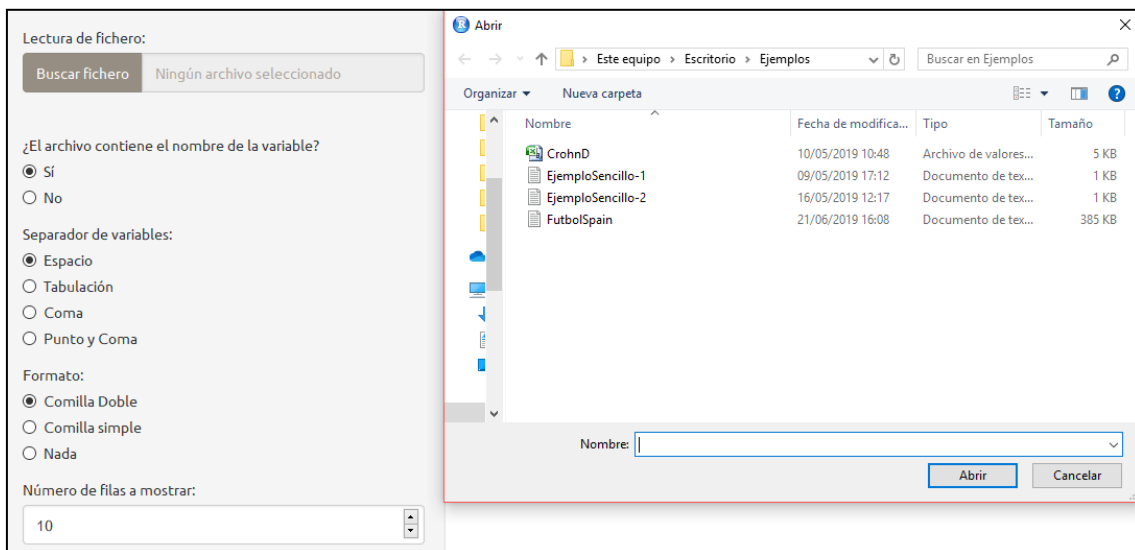
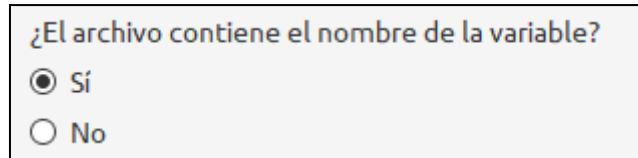


Figura 7: Selección de datos externos

Esta aplicación lee ficheros con extensiones .txt y .csv.

Una vez seleccionado el archivo, la siguiente subsección, *Figura 8*, nos indica si el archivo contiene en la primera fila el nombre de cada variable. Dependiendo si la contiene o no debe marcarse, respectivamente, el botón de **Sí** o el botón de **No**. Por defecto está marcado Sí.





¿El archivo contiene el nombre de la variable?

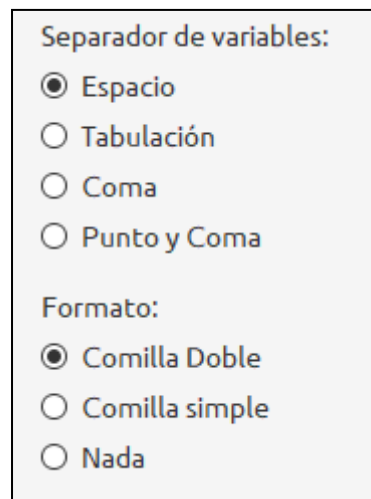
Sí

No

Figura 8: Nombre de la variable

Para que las variables del fichero se lean correctamente, en la subsección *Separador de observaciones de la Figura 9*, se escoge el tipo de separación (espacio, tabulación, coma o punto y coma) que hay entre unas variables y otras dentro de la misma fila. Por defecto, el separador es *Espacio*.

En la subsección *Formato de la Figura 9* se elige si las observaciones de tipo carácter están escritas con “*Comilla Doble*”, ‘*Comilla Simple*’ o *Nada*. Por defecto, *Comilla Doble*.



Separador de variables:

Espacio

Tabulación

Coma

Punto y Coma

Formato:

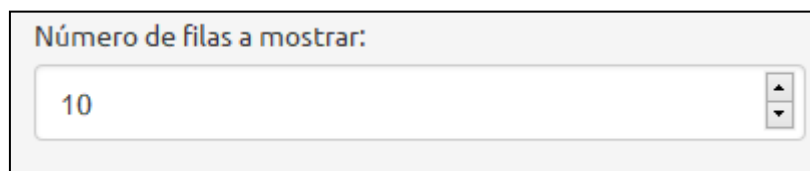
Comilla Doble

Comilla simple

Nada

Figura 9: Separación observaciones

Una vez elegidas las opciones adecuadas en la lectura del fichero, la última subsección que aparece en esta parte de pantalla, *Número de filas a mostrar, Figura 10*, sirve para que el usuario indique el número de filas a mostrar por pantalla del fichero escogido. Por defecto mostrara las 10 primeras filas.



Número de filas a mostrar:

10

Figura 10: Observaciones a mostrar

En el panel principal de la pantalla se encuentra un breve resumen de los estadísticos resumen calculados en esta aplicación Shiny y los datos de las filas desde la primera hasta el número elegido en el selector que muestra la *Figura 10*.

## Elección variable

Figura 11: Aplicación Shiny Descripción Univariante, Ventana elección de variable

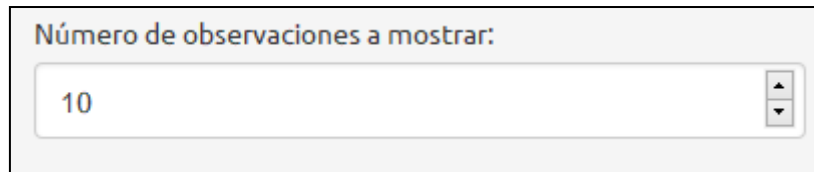
En la pestaña *Elección variable*, Figura 11, el usuario indica la columna, variable, que se quiera analizar del fichero en uso. Para seleccionar la variable se indica el número de columna en la subsección *Variable escogida*, Figura 12, aunque al mostrar los datos en la parte derecha es más fácil ver que el número de columna elegido corresponde a los datos que se quieren analizar.

Figura 12: Selección de variable

En las siguientes dos subsecciones se declara el tipo de variable, *Numérica* o *Cualitativa* y dentro de las Numéricas si es *Continua* o *Discreta*, Figura 13. Por defecto Cualitativa.

Figura 13: Selección tipo de variable

En la última subsección, Figura 14, el usuario especifica el número de observaciones a mostrar en la parte derecha de la pantalla. Por defecto se muestran las 10 primeras observaciones.



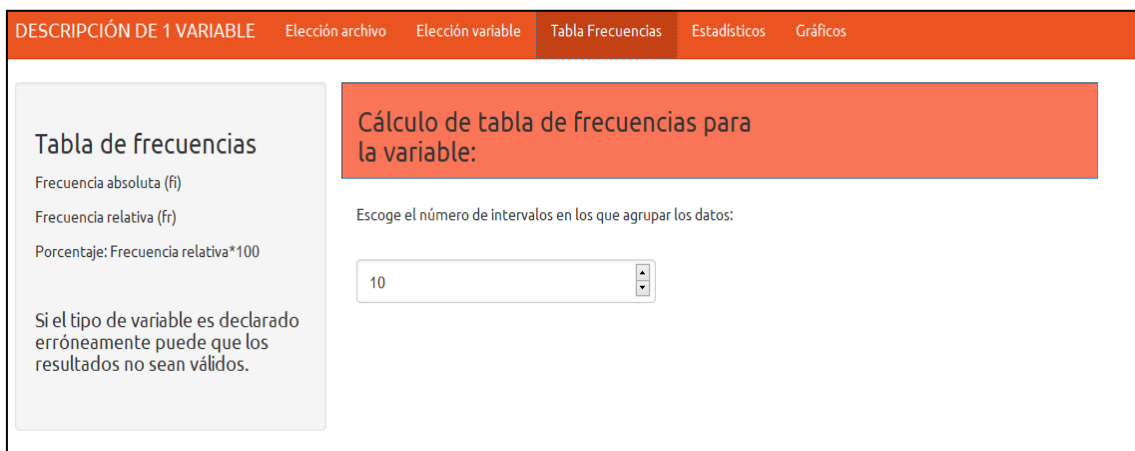
Número de observaciones a mostrar:

10

Figura 14: Observaciones a mostrar

Una vez efectuadas las selecciones oportunas en esta ventana ya estamos preparados para proseguir con los estadísticos resumen y gráficos correspondientes en las siguientes ventanas.

## Tabla de frecuencias



DESCRIPCIÓN DE 1 VARIABLE   Elección archivo   Elección variable   Tabla Frecuencias   Estadísticos   Gráficos

Tabla de frecuencias

Frecuencia absoluta ( $f_i$ )

Frecuencia relativa ( $fr$ )

Porcentaje: Frecuencia relativa\*100

Si el tipo de variable es declarado erróneamente puede que los resultados no sean válidos.

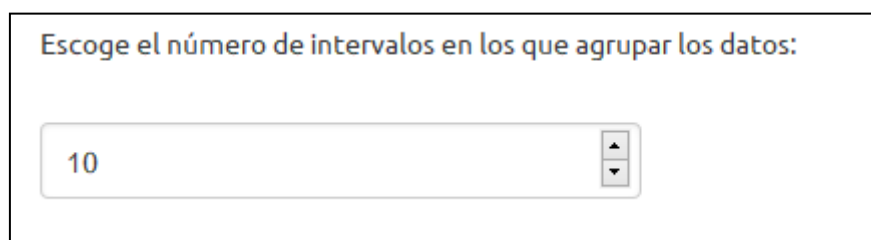
Cálculo de tabla de frecuencias para la variable:

Escoge el número de intervalos en los que agrupar los datos:

10

Figura 15: Aplicación Shiny Descripción Univariante, Ventana Tabla de frecuencias

En la ventana *Tabla de frecuencias* se calcula la tabla de frecuencias de la variable escogida. Si la variable fue declarada como cualitativa o numérica discreta se presenta la tabla con todos los valores mientras que si es numérica continua los datos serán agrupados en intervalos y la tabla será calculada en base a estos intervalos. En este último caso, en la *Figura 16*, se muestra la opción para la elección del número de intervalos en los que se quieren agrupar los datos.



Escoge el número de intervalos en los que agrupar los datos:

10

Figura 16: Número de intervalos

En la parte izquierda de la ventana que aparece en la *Figura 15* se muestran las columnas que aparece en la tabla de frecuencias.

## Tabla de frecuencias

Frecuencia absoluta ( $f_i$ )

Frecuencia relativa ( $f_r$ )

Porcentaje: Frecuencia relativa \* 100

Si el tipo de variable es declarado erróneamente puede que los resultados no sean válidos.

Figura 17: Definiciones tabla de frecuencias

## Estadísticos

Figura 18: Aplicación Shiny Descripción Univariante, Ventana Estadísticos

En la ventana que aparece en la *Figura 18* se exponen una serie de estadísticos de centro y de dispersión.

En la parte izquierda de la ventana se encuentran enumerados los estadísticos y en la derecha el valor calculado de los mismos.

Cuando la variable es cualitativa no se calcularán estos estadísticos. Por ello, aparte de otras razones, es muy relevante declarar adecuadamente el tipo de variable en la segunda ventana, *Figura 11*.

## Gráficos

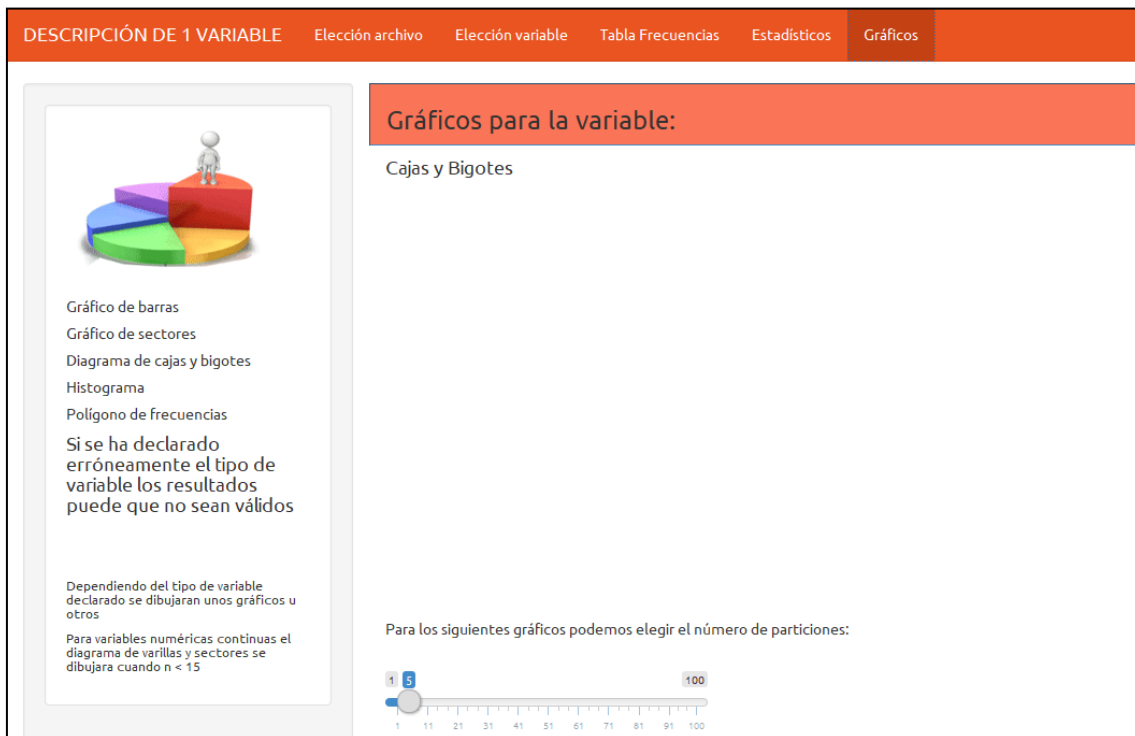


Figura 19: Aplicación Shiny Descripción Univariante, Ventana Gráficos

En la pestaña *Gráficos* se presentan una serie de gráficos. Dependiendo del tipo de variable escogido en la ventana “*Elección variable*” saldrán unos u otros.

El *gráfico de barras* y el *diagrama de sectores* los obtenemos para variables cualitativas, numéricas discretas y para numéricas continuas cuando tiene menos de 15 valores distintos.

El *diagrama de cajas y bigotes*, el *histograma* y el *polígono de frecuencias relativas* lo obtenemos para variables numéricas continuas.

Podemos elegir el número de clases en las que se divide el histograma a partir del cual también obtenemos el polígono de frecuencias, *Figura 20*.

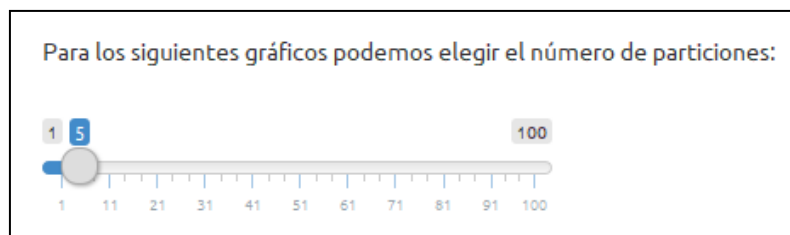


Figura 20: Numero clases del histograma

Cuando se realiza un *histograma* debajo podemos encontrar los valores exactos utilizados en las particiones del rango en intervalos.

El *polígono de frecuencias* se representa en dos formas, superpuesto al histograma anterior y sin el histograma.

## 4.2 Aplicación Shiny Descripción Bivariante

La segunda implementación de una aplicación con Shiny en este TFG se encarga de la estadística descriptiva bivariante. Esto es, el análisis conjunto de dos variables con el objeto de establecer la posible relación que existe entre las mismas.

Se pueden hacer estudios entre dos variables cualitativas, dos variables numéricas o entre una variable cualitativa y una variable numérica. Esta última también puede ser vista como agrupamiento de una variable numérica en varios grupos.

En la aplicación desarrollada para este TFG para la descripción bivariante se recogen las tres posibilidades.

La aplicación desarrollada con Shiny para la descripción de dos variables se llama *Descripción Bivariante* y se ejecuta en RStudio con el fichero “**DosVariables.R**” escribiendo la orden `library(shiny); runApp(“DosVariables.R”)` en consola. El código del script *DosVariables.R* se encuentra en el Apéndice B.

Esta aplicación Shiny se divide en 5 secciones o ventanas como muestra la *Figura 21*.



Figura 21: Encabezado Aplicación Shiny Descripción Bivariante

### Archivo

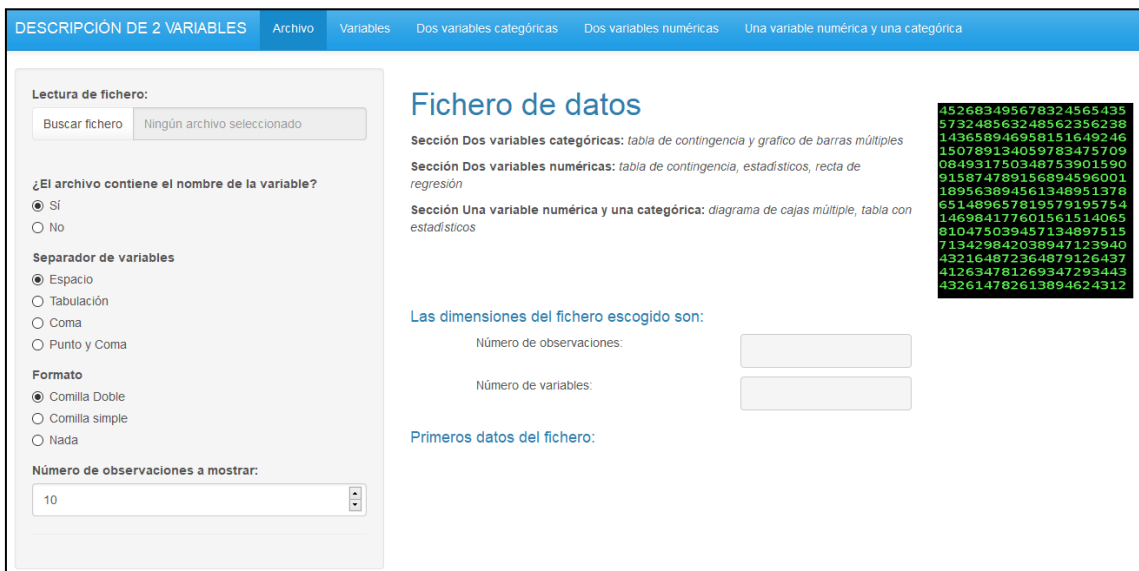


Figura 22: Aplicación Shiny Descripción Bivariante, Ventana Elección de Archivo

Como podemos observar en la *Figura 22* esta ventana es muy similar a la primera ventana, *Elección de archivo*, *Figura 6*, de la aplicación *Descripción Univariante*. Estas se diferencian en el texto inicial del panel principal, por lo tanto obviamos la explicación de nuevo de su funcionamiento y estructura.

## Variables

DESCRIPCIÓN DE 2 VARIABLES Archivo Variables Dos variables categóricas Dos variables numéricas Una variable numérica y una categórica

### Elección de variables

**Primera variable:**

Variable escogida: 1

Selecciona si la variable es numérica o cualitativa:

Cualitativa  
 Numérica

**Segunda variable:**

Variable escogida: 2

Selecciona si la variable es numérica o cualitativa:

Cualitativa  
 Numérica

Número de observaciones a mostrar: 10

Datos variable: Datos variable:

Figura 23: Aplicación Shiny Descripción Bivariante, Ventana Elección Variables

La aplicación Shiny para la descripción bivariante cuenta con un diseño y uso en la ventana “Variables”, Figura 23, muy similar a la ventana *Elección de variable*, Figura 11, de la aplicación Shiny *Descripción Univariable*.

La diferencia fundamental que hay entre estas dos ventanas es el número de variables a elegir y las correspondientes configuraciones de cada una de las variables.

Las variables tienen que tener el mismo número de observaciones para que no haya incoherencias.

A partir de esta ventana, dependiendo de los tipos de las variables que hayamos escogido, los resultados solo se presentarán en una de las tres ventanas restantes.

## Dos variables categóricas

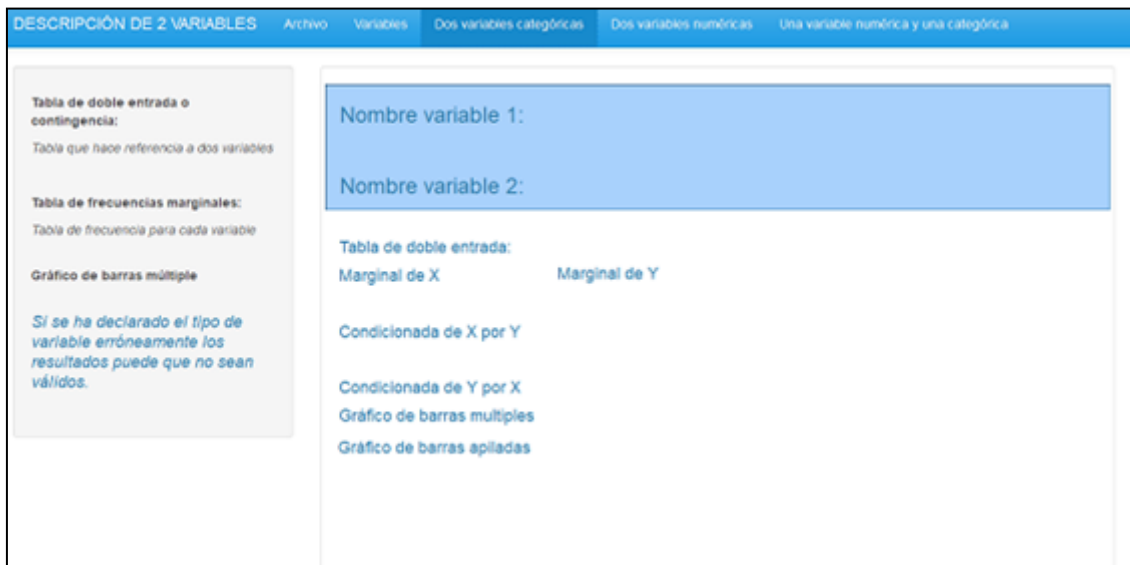


Figura 24: Aplicación Shiny Descripción Bivariante, Ventana Dos variables categóricas

La ventana para dos variables categóricas, *Figura 24*, presenta los resultados en el momento que las variables a analizar hayan sido declaradas categóricas en la ventana correspondiente. En otro caso, esta ventana se presentará vacía.

En el panel de la izquierda se encuentra una breve descripción de los resultados que se expondrán en esta ventana.

En el panel principal, en la parte superior se muestra el nombre de las variables elegidas. Debajo se presenta la tabla de contingencia, la marginal de cada variable y un gráfico de barras múltiple y otro de barras apiladas.



## Dos variables numéricas

DESCRIPCIÓN DE 2 VARIABLES Archivo Variables Dos variables categóricas **Dos variables numéricas** Una variable numérica y una categórica

**Tabla de doble entrada o contingencia:**  
Tabla que hace referencia a dos variables

**Tabla de frecuencias marginales:**  
Tabla de frecuencia para cada variable

**Tabla de distribución condicionada**

**Coefficiente de correlación lineal de Pearson:**  
valor entre -1 y 1 que indica la intensidad y dirección de la relación entre X e Y

**Covarianza:**  
valor que indica el grado de variación conjunta de dos variables

**Recta de regresión:**  
línea recta que mejor se ajusta a la nube de puntos

*Si se ha declarado el tipo de variable erróneamente los resultados puede que no sean válidos.*

Nombre variable 1:  
Nombre variable 2:

Escoger número de intervalos para la o las variables continuas:  
5  
5

Tabla de doble entrada:  
Marginal de X Marginal de Y  
Condicionada de X por Y  
Condicionada de Y por X

Mostrar  
 No mostrar

Media X:  
Media Y:  
Desviación Típica X:  
Desviación Típica Y:  
Covarianza:  
Coeficiente de correlación lineal de Pearson:  
Nube de puntos  
Coeficientes de la ecuación de la recta de regresión:  
Gráfico de la recta de regresión Y sobre X

¿Modificar ejes del gráfico?  
 Sí  
 No

Figura 25: Aplicación Shiny Descripción Bivariante, Ventana Dos variables numéricas

La ventana para dos variables numéricas, *Figura 25*, presenta los resultados en el momento que las variables a analizar hayan sido declaradas numéricas en la ventana correspondiente. En otro caso, esta ventana se presentará vacía.

En el panel de la izquierda se describen los resultados que se encontrarán en este panel principal.

En el panel central, debajo de los nombres de las variables se muestran las opciones para la elección del número de intervalos en los que se quieren agrupar los datos de cada variable escogida si esta es continua, *Figura 26*.

Escoger número de intervalos para la o las variables continuas:

5  
5

Figura 26: Selectores del número de intervalos

Cuando las variables son declaradas numéricas discretas la tabla de doble entrada, la

marginal y las condicionadas de cada variable se realiza con los datos sin agrupar. Si las variables son declaradas numéricas continuas los datos se agrupan en el número de intervalos antes señalado y se crean estas tablas con los datos así agrupados. En el caso de que cada variable sea de un tipo los datos de la variable continua serán agrupados y los de la discreta no.

Después de estas tablas, el usuario tiene la posibilidad de obtener otra tabla adicional conteniendo las columnas  $x_i$ ,  $y_i$ ,  $x_i^2$ ,  $y_i^2$ ,  $x_i*y_i$  que servirán para los cálculos siguientes.

**Tabla de cálculos intermedios:**

Mostrar

No mostrar

Figura 27: Opción mostrar tabla

A continuación, se calculan los estadísticos media, desviación típica, covarianza y el coeficiente de correlación de Pearson. Además, se representa una nube de puntos y la recta de regresión.

En la recta de regresión aparecen los coeficientes de la ecuación de dicha recta.

Al representar la recta sobre la nube de puntos no siempre se muestra el corte de esta con los ejes. El usuario puede tener interés en visualizar estos cortes para representar la ordenada en el origen. Esto es posible ya que se ofrece la opción de modificar los límites de los ejes. Esta opción se muestra en la *Figura 28*.

Coeficientes de la ecuación de la recta de regresión:

Gráfico de la recta de regresión Y sobre X:

**¿Modificar ejes del gráfico?**

Sí

No

<b>Límite inferior del eje X :</b>	<b>Límite superior del eje X :</b>
<input type="text" value="0"/>	<input type="text" value="2"/>
<b>Límite inferior del eje Y :</b>	<b>Límite superior del eje Y :</b>
<input type="text" value="0"/>	<input type="text" value="2"/>

Figura 28: Recta de regresión

## Una variable numérica y una categórica



Figura 29: Aplicación Shiny Descripción Bivariante, Ventana Una variable numérica y una categórica

La ventana para una variable numérica y una categórica, *Figura 29*, presenta los resultados en el momento que las variables a analizar hayan sido declaradas como numérica y cualitativa, o viceversa, en la ventana correspondiente. Esta situación también puede ser considerada como una numérica reunida en varios grupos. En otro caso, esta ventana se presentará vacía.

En el panel izquierdo se ofrece una noción de los resultados que se obtendrán en el panel principal expuesto en la *Figura 29*.

En este caso se calcula una tabla de estadísticos resumen que contiene el mínimo, el máximo, la media, la mediana y el primer y tercer cuartil de la variable numérica en cada valor de la variable cualitativa (o grupo) para después, a partir de ellos, realizarse el gráfico de cajas y bigotes múltiple.

En el gráfico, por defecto, se obtienen las cajas en horizontal pero el usuario tiene la opción de representarlo en vertical, *Figura 30*.



Figura 30: Opción orientación gráfico de cajas

### 4.3 Aplicación Shiny Probabilidad

La tercera implementación de una aplicación con Shiny en este TFG se encarga de la parte de probabilidad. La probabilidad es un cálculo matemático de las posibilidades que existen de que se cumpla un suceso o evento futuro.

Los cálculos realizados en esta aplicación serán relacionados con dos distribuciones de probabilidad, una discreta como es la Binomial y otra continua como lo es la Normal. Además se muestra la aproximación de la binomial por la normal.

La aplicación desarrollada con Shiny para el uso de distribuciones de probabilidad se llama *Probabilidad* y se ejecuta en RStudio con el fichero “**Probabilidad.R**” escribiendo la orden ***library(shiny); runApp(‘Probabilidad.R’)*** en consola. El código del script *Probabilidad.R* se encuentra en el Apéndice C.

La aplicación Shiny *Probabilidad* está dividida en cuatro secciones o ventanas: *Probabilidad*, *Binomial*, *Normal*, *Aproximación Binomial a la Normal*. En la *Figura 31* podemos observar la cabecera de la ventana principal.

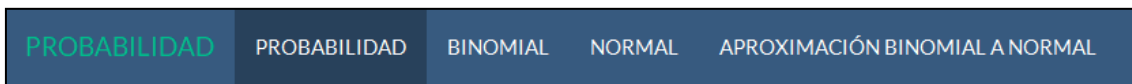


Figura 31: Encabezado Aplicación Shiny Probabilidad

## Probabilidad

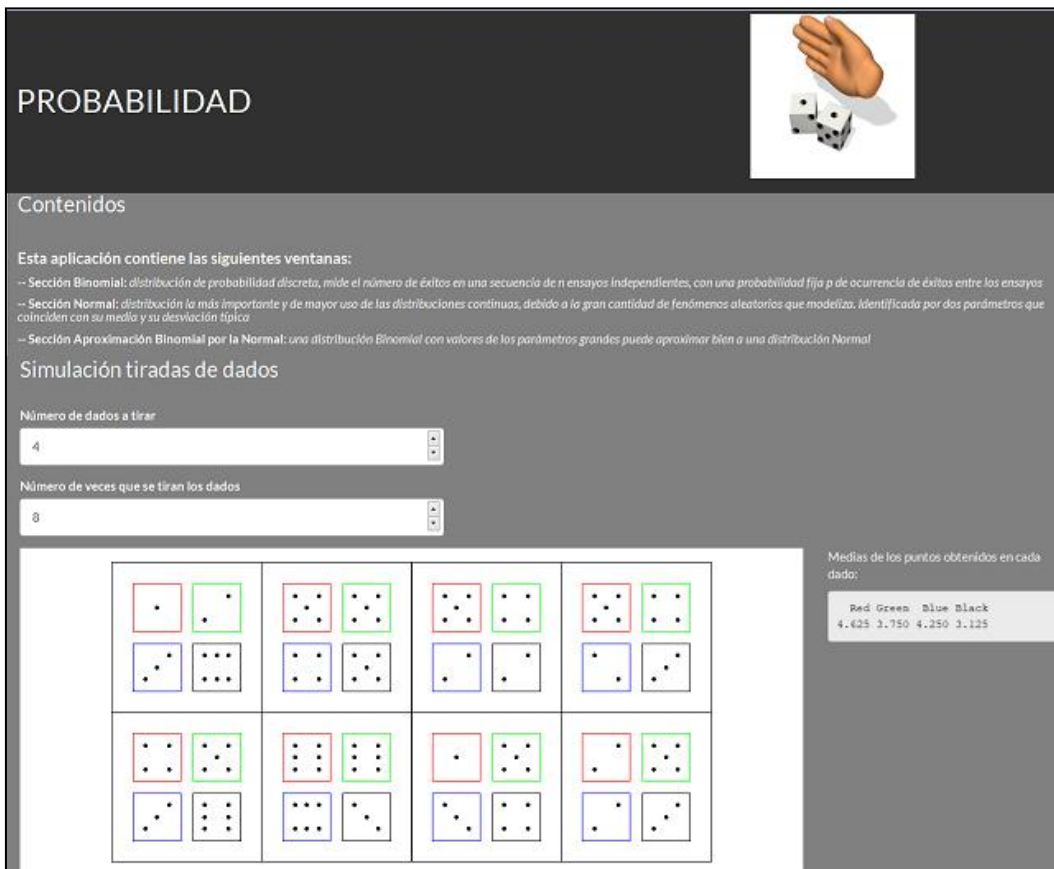


Figura 32: Aplicación Shiny Probabilidad, Ventana Probabilidad

En esta primera ventana de la aplicación encontraremos los contenidos que se

mostrarán a lo largo de la aplicación y una simulación de tiradas de dados con la media de puntos obtenidos para cada dado en la simulación obtenida.


## Binomial

### Distribución binomial

Parámetros de la distribución binomial:

Número de repeticiones,  $n$

Probabilidad de éxito,  $p$



Variable  $X \rightarrow B(n,p)$

Función de probabilidad,  $P(X = x)$ : *probabilidad de que la variable aleatoria  $X$  tome exactamente el valor  $x$*

Función de distribución,  $P(X \leq x)$ : *probabilidad de que una variable aleatoria  $X$  asuma un valor inferior o igual a  $x$*

Figura 33: Aplicación Shiny Probabilidad, Ventana binomial panel izquierdo

La distribución binomial es una distribución de probabilidad discreta que tiene como parámetros el número de repeticiones del experimento,  $n$ , y la probabilidad de éxito,  $p$ . Estos parámetros se introducen en la aplicación en el panel mostrado en la *Figura 33*.

Este panel se va a explicar por partes y, por lo tanto, la misma ventana es dividida en varias *Figuras*.

Los resultados que se muestran inicialmente se calculan con los valores que toman los parámetros por defecto en la aplicación.

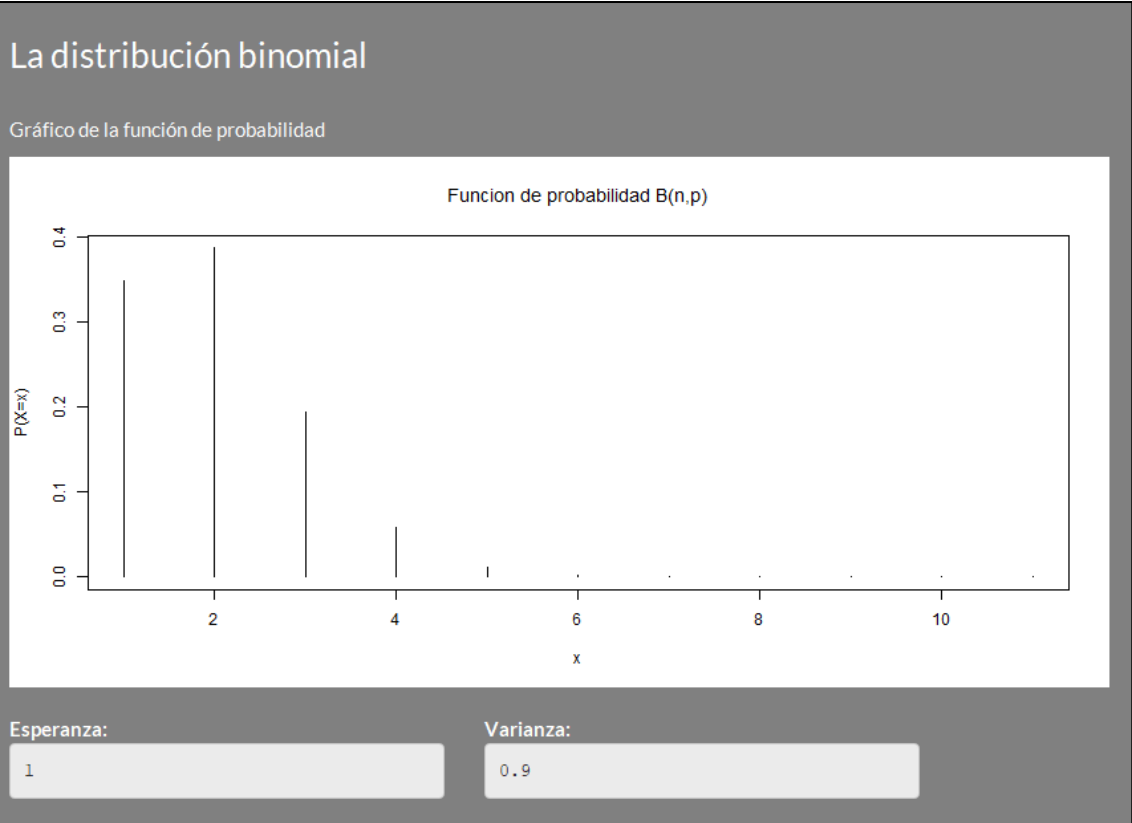


Figura 34: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 1

En la parte superior del panel principal se encuentra el gráfico de la función de probabilidad y el cálculo de la esperanza y la varianza de dicha distribución a partir de los parámetros introducidos, *Figura 34*.

A continuación se calculan detalladamente una serie de probabilidades.

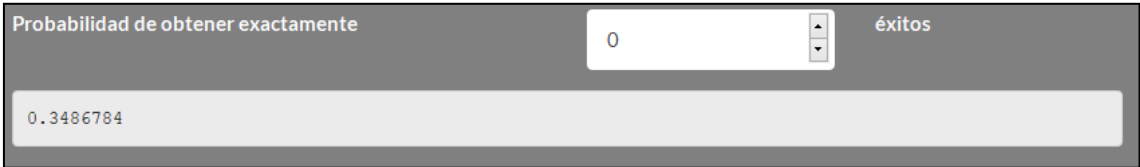


Figura 35: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 2

La *Figura 35* muestra el cálculo de  $P(X = x)$  donde  $x$  es un número que el usuario puede introducir.

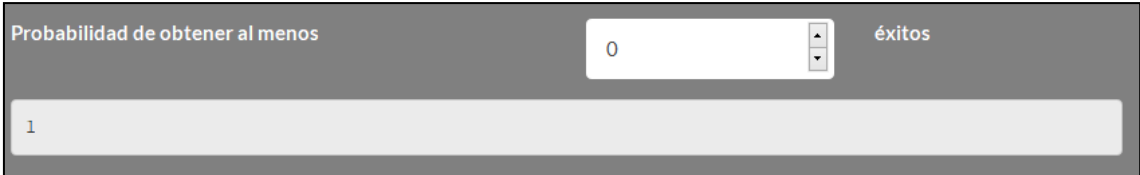


Figura 36: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 3

La probabilidad  $P(X \geq x)$  se calcula en la *Figura 36*,  $x$  número a introducir por el usuario.

Figura 37: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 4

La *Figura 37* corresponde con la probabilidad  $P(X \leq x)$ ,  $x$  es el número que el usuario introduce.

Figura 38: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 5

La probabilidad de que  $x$  se encuentre dentro de un intervalo  $[a, b]$ ,  $P(a \leq X \leq b)$ , se calcula en el trozo de aplicación mostrado en la *Figura 38*,  $a$  es el primer número que el usuario puede introducir y  $b$  el segundo.

Figura 39: Aplicación Shiny Probabilidad, Ventana binomial, panel principal 6

En este caso se calcula el valor de la variable que deja a su derecha la probabilidad introducida. Esto es, dado  $p$  devuelve el valor  $x$  tal que  $P(X \leq x) = p$

Figura 40: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 7

La *Figura 40* muestra cómo obtener una muestra aleatoria de tantos elementos como el usuario indique.

## Normal



Figura 41: Aplicación Shiny Probabilidad, Ventana Normal, panel izquierdo

La distribución Normal o gaussiana es la más importante y de mayor uso en las distribuciones continuas ya que modeliza gran cantidad de fenómenos aleatorios. Esta distribución es identificada por dos parámetros que coinciden con la media,  $\mu$ , y la desviación típica,  $\sigma$ , de la distribución.

En el panel de la aplicación Shiny *Probabilidad* que se observa en la *Figura 41* el usuario introduce estos parámetros para manejar la Normal que interese.

Una vez especificados estos parámetros se pasa a los cálculos del panel principal. Este panel se irá explicando por partes y por lo tanto la misma ventana será dividida en varias *Figuras*.

Los resultados que se muestran inicialmente se calculan con los valores por defecto que toman los parámetros de esta aplicación Shiny.



## La distribución normal

Representación de la función de densidad

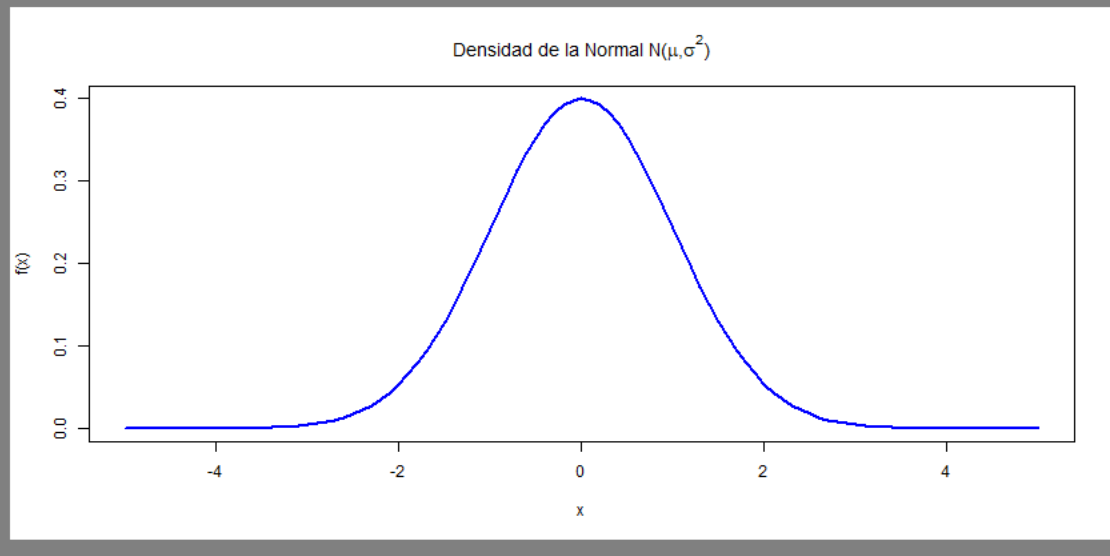


Figura 42: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 1

Después de introducir los parámetros deseados, en la parte superior del panel principal aparece el gráfico de la función de densidad, como podemos observar en la *Figura 42*.

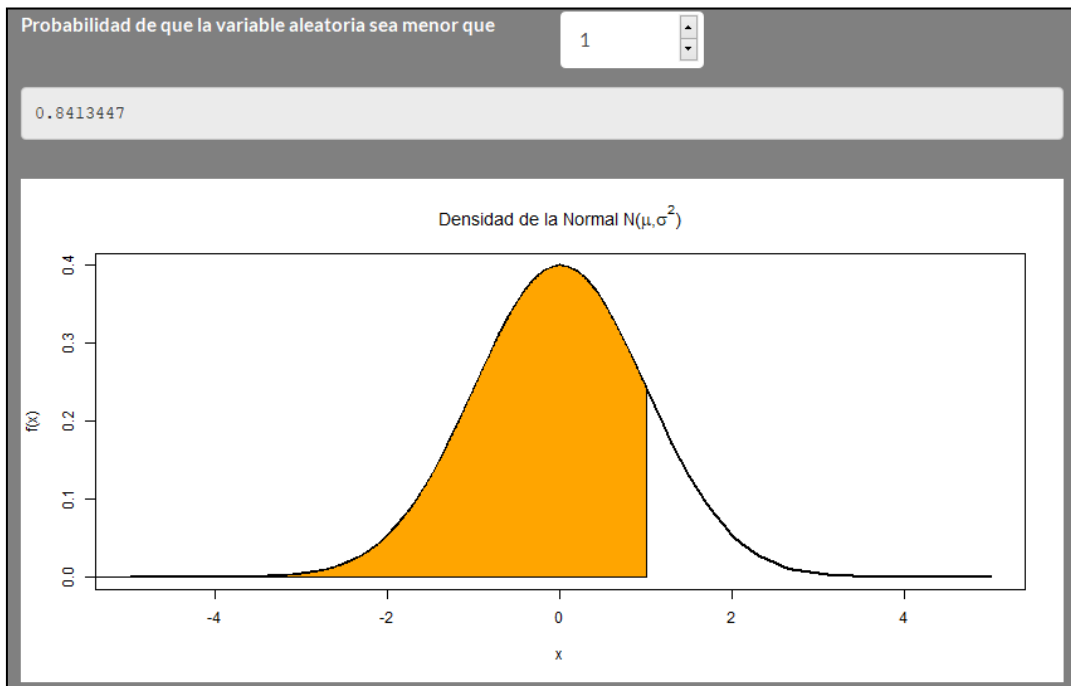


Figura 43: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 2

La probabilidad que se calcula en la *Figura 43* coincide con la función de distribución evaluada en ese punto, es decir,  $F(x) = P(X \leq x)$ . Además, se muestra el gráfico con el área bajo la curva que corresponde a esta probabilidad coloreado.

El usuario fija el valor de  $x$ .

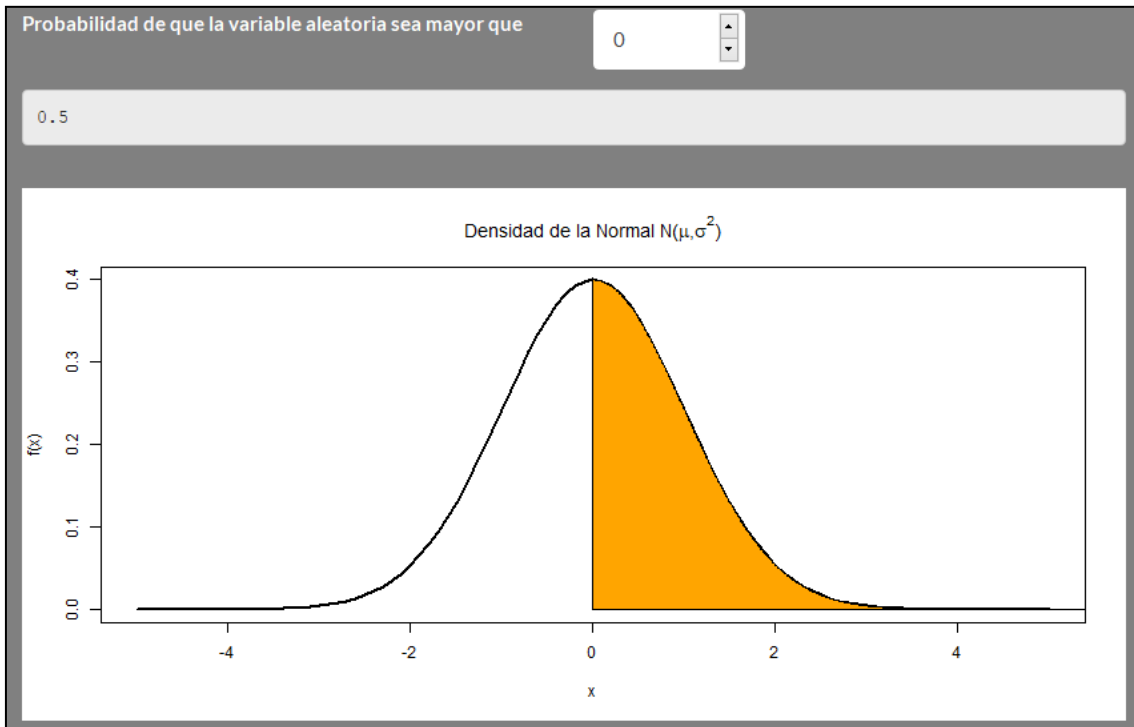


Figura 44: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 3

La probabilidad que se calcula en la *Figura 44* es  $P(X > x)$ . Esta probabilidad equivale a  $P(X > x) = 1 - P(X \leq x) = 1 - F(x)$ .

El valor de  $x$  es el valor que el usuario establece.

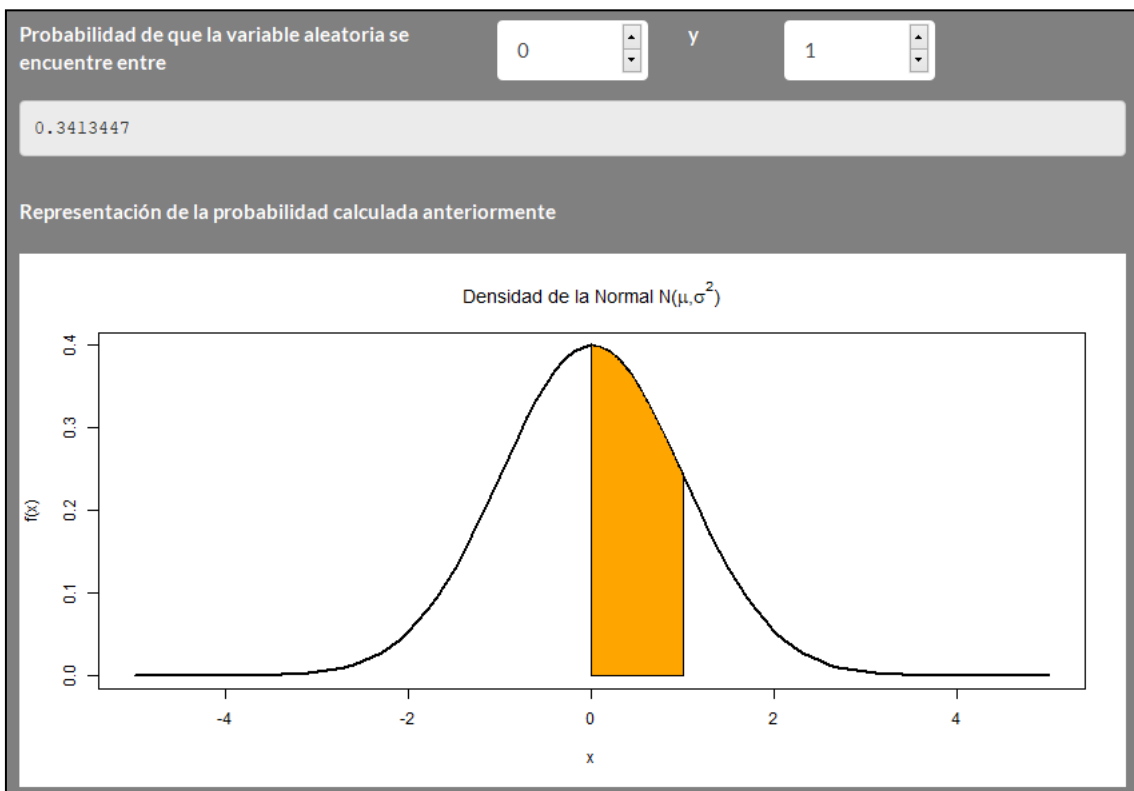
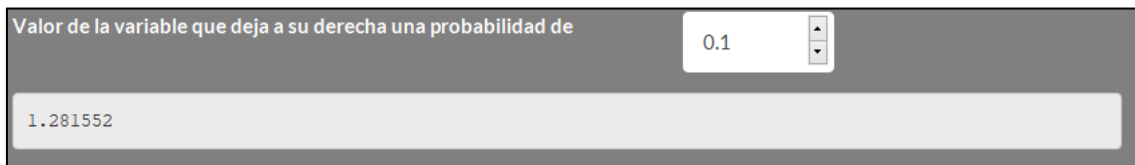


Figura 45: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 4

La *Figura 45* recoge una captura de la ventana de la aplicación Shiny donde se calcula la probabilidad de que  $X$  esté comprendida en un intervalo  $[a, b]$ , es decir  $P(a \leq X \leq b)$ . Esta probabilidad también se puede escribir como una diferencia de valores de la función de distribución,  $F(b) - F(a)$ .

Además de calcular la probabilidad numéricamente, se representa gráficamente dicha probabilidad coloreando el área de campana de Gauss que corresponde.



**Figura 46:** Aplicación Shiny Probabilidad, Ventana Normal, panel principal 5

En la *Figura 46* muestra el cálculo del valor de la variable  $X$  que deja a su derecha la probabilidad indicada por el usuario.



**Figura 47:** Aplicación Shiny Probabilidad, Ventana Normal, panel principal 6

La *Figura 47* muestra cómo obtener una muestra aleatoria de tantos elementos como el usuario indique.

## Aproximación Binomial a Normal

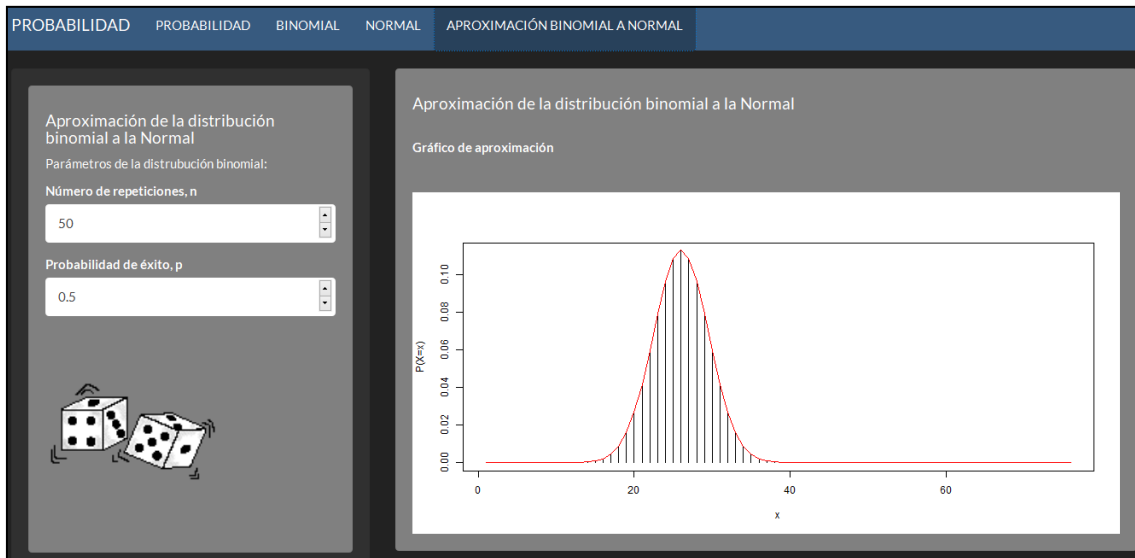


Figura 48: Aplicación Shiny Probabilidad, Ventana Aproximación Binomial a Normal

La aplicación Shiny *Probabilidad* contiene una ventana relativa a la aproximación de la distribución Binomial por una Normal.

En el panel de la izquierda, como se observa en la *Figura 48*, se pide al usuario introducir los parámetros de la distribución Binomial,  $n$  y  $p$ , si la multiplicación de ellos es menor que 3 no aproxima bien y por lo tanto, junto al gráfico, sale un mensaje de aviso de una aproximación de mala calidad.

## 5. EJEMPLOS PRÁCTICOS

Esta sección de ejemplos prácticos tiene por objeto mostrar el uso de las aplicaciones desarrolladas con varios archivos de datos.

Este TFG incluye cuatro archivos de datos, los cuales pueden ser utilizados de ejemplos para probar las aplicaciones.

Estos ficheros son de diferente extensión y por lo tanto de diferente dificultad para realizar los cálculos sin ayuda de una herramienta informática.

Los ficheros más recomendados para realizar los cálculos a mano y después comprobar que se obtienen los mismos resultados que usando las aplicaciones Shiny desarrolladas en el presente TFG.

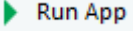
- El archivo ***EjemploSencillo-1.txt*** es un conjunto de datos ficticios que recoge datos de 20 estudiantes sobre cuál es su color preferido, su edad y su altura.
- El archivo, ***EjemploSencillo-2.txt*** es un conjunto de datos ficticios que recoge datos de 30 participantes de una competición de atletismo, que consiguieron medalla, sobre su estado civil, el número de hijos que tiene, su peso, los días que entrena a la semana, los kilómetros que realiza y la medalla conseguida.

Los siguientes dos archivos, *Crohn* y *FutbolSpain*, incluidos son de mayor extensión y por lo tanto la complejidad de calcular los estadísticos y gráficos sin ayuda de una herramienta informática es alta.

Estos dos ficheros contienen datos reales. Esto contribuye a que los estudiantes vean las aplicaciones de la Estadística en la vida cotidiana.

- El archivo ***CrohnD.csv*** es un conjunto de datos obtenido de un estudio sobre los efectos adversos de un medicamento en 117 pacientes afectados por la enfermedad de Crohn (enfermedad inflamatoria crónica de los intestinos) de los que se recoge información de 10 variables diferentes.
- El paquete de R *engsoccerdata* es un repositorio de un conjunto de datos de fútbol con algunas funciones integradas. Este paquete incluye datos de tres ligas inglesas y de algunas europeas. El archivo de datos de la liga española se denomina *spain* en dicho paquete. Este fichero cuenta con los resultados de todos los partidos de la primera división de la liga española desde la temporada 1928-1929 hasta la temporada 2015-2016. El archivo propuesto en este TFG, ***FutbolSpain.txt***, es un subconjunto de este fichero. Se presentan los datos de 6080 partidos de fútbol de la liga española desde la temporada 2000-2001 hasta la temporada 2015-2016.

Para presentar los ejemplos se necesita ejecutar las aplicaciones Shiny desarrolladas. Esto se puede realizar de dos maneras una vez abierto el fichero en RStudio:

- Desde el menú de RStudio con el botón .
- Ejecutando la orden `library(shiny); runApp('UnaVariable.R')`, `library(shiny); runApp('DosVariables.R')` o `library(shiny); runApp('Probabilidad.R')` en consola según corresponda.

En este TFG también se incluyen tres videos realizados con ayuda del software Screencast-O-Matic v2.0 para mostrar el uso de las tres aplicaciones desarrolladas:

- **Ejemplo aplicación Shiny *Descripción Univariante*:**

En el primer vídeo, contenido en el archivo ***Ejemplo práctico de uso de la aplicación Shiny Descripción Univariante.mp4***, se detalla el uso de la aplicación Shiny *Descripción Univariante* con los archivos de datos *EjemploSencillo-1.txt* y *FutbolSpain.txt*.

- **Ejemplo aplicación Shiny *Descripción Bivariante***

El segundo vídeo, ***Ejemplo práctico de uso de la aplicación Shiny Descripción Bivariante.mp4***, muestra como utilizar la aplicación Shiny *Descripción Bivariante* con los archivos de datos *EjemploSencillo-2.txt* y *CronhD.csv*.

- **Ejemplo aplicación Shiny *Probabilidad***

En el tercer vídeo, ***Ejemplo práctico de uso de la aplicación Shiny Probabilidad.mp4***, se incluye un ejemplo del uso de la aplicación Shiny *Probabilidad*. Para esta aplicación, y por lo tanto para este ejemplo no se necesita usar ningún fichero de datos.

## 6. CONCLUSIONES

La finalidad de este trabajo fin de grado ha sido desarrollar una serie de aplicaciones web interactivas que recojan los contenidos de Estadística incluidos en las materias de Matemáticas de las Enseñanzas Medias. El uso de herramientas informáticas, como las aquí planteadas, ayuda a la comunidad docente a resolver ejercicios con conjuntos de datos de dimensiones mayores que los ejercicios planteados de forma usual para resolver manualmente.

Un software que permite desarrollar estas aplicaciones con un marco web elegante y poderoso sin necesidad de contar con conocimientos de HTML, CSS o Java Script es el software estadístico *R*.

*R* es un entorno de programación con un enfoque al análisis estadístico muy poderoso formado por un conjunto de herramientas muy flexibles que pueden ampliarse mediante paquetes, librerías o definiendo nuestras propias funciones. Para trabajar más cómodamente con *R* se utiliza IDE RStudio.

El paquete Shiny, disponible en el repositorio CRAN de *R*, es una herramienta para desarrollar aplicaciones web interactivas con el conocimiento únicamente del lenguaje de programación *R*. Una vez desarrolladas las aplicaciones Shiny, estas permiten al usuario final interactuar con sus datos sin tener que manipular código directamente.

Tras el análisis de los contenidos del bloque *Estadística y Probabilidad* de los currículos de las asignaturas de Matemáticas se ha considerado conveniente para recoger estos contenidos dividirlos en tres bloques. Consecuentemente se ha desarrollado tres aplicaciones: *Aplicación Shiny Descripción Univariante*, *Aplicación Shiny Descripción Bivariante* y *Aplicación Shiny Probabilidad*.

En la *Aplicación Shiny Descripción Univariante* se recogen los contenidos para la descripción de una variable. La *Aplicación Shiny Descripción Bivariante* reúne los contenidos para la descripción conjunta de dos variables. La *Aplicación Shiny Probabilidad* recoge los conceptos relativos al cálculo de probabilidades y la visualización de distribuciones de probabilidad.

Para finalizar, se puede decir que el paquete Shiny de *R* es una buena herramienta para desarrollar aplicaciones web interactivas destacando principalmente las aplicaciones que congregan contenido estadístico, como las tres aplicaciones desarrolladas en este trabajo fin de grado. También cabe resaltar el beneficio que el uso de estas aplicaciones aporta a toda la comunidad educativa al presentar una herramienta amigable con la que trabajar los contenidos de estadística de una forma más fluida con conjuntos de datos pequeños y, especialmente, de grandes dimensiones como se muestra en los videos incluidos como ejemplos prácticos en este TFG.

## Apéndice A: Código Aplicación Shiny Descripción Univariante

```
# install.packages("shiny", dependencies = T)
# install.packages("shinythemes", dependencies = T)
# install.packages("agricolae", dependencies = T)

library(shiny)
library(shinythemes)
library(agricolae)

Mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}

ui <- shinyUI(navbarPage(theme = shinytheme("united"), "DESCRIPCIÓN DE 1
VARIABLE",
  tabPanel("Elección archivo", uiOutput("component1")),
  tabPanel("Elección variable", uiOutput("component2")),
  tabPanel("Tabla Frecuencias", uiOutput("component3")),
  tabPanel("Estadísticos", uiOutput("component4")),
  tabPanel("Gráficos", uiOutput("component5"))

))

server <- function(input, output) {

  output$component1 <- renderUI({
    sidebarLayout(
      sidebarPanel(
        fileInput('file1', 'Lectura de fichero: ',
          accept = c(
            'text/csv',
            'text/comma-separated-values',
            'text/tab-separated-values',
            'text/plain',
            '.csv'
          ),
        ),
        buttonLabel = "Buscar fichero",
        placeholder = "Ningún archivo seleccionado"
      ),
      radioButtons("header",
        label="¿El archivo contiene el nombre de la variable?",
        choices=list("Sí", "No"),
        selected="Sí"),
      radioButtons(inputId = 'sep', label='Separador de variables:',
```



```

      c(Espacio = '\ ',
        "Tabulación" = '\t',
        Coma = ',',
        "Punto y Coma" = ';')
    ),
    '\ '),
  radioButtons('quote', 'Formato:',
    c('Comilla Doble' = '"',
      'Comilla simple' = "'",
      "Nada" = ''),
    ""),
  numericInput(inputId = "obs", label = "Número de filas a mostrar:", value = 10),
  tags$hr()
),
mainPanel(
  fluidRow(column(8,
    h1("Fichero de datos"),
    h4("Para variables cualitativas se obtiene: "),
    p("Tabla de frecuencias", align = "center"),
    p("Gráficos", align = "center"),
    h4("Además, para variables numéricas se obtiene: "),
    p("Estadísticos de centro o localización", align = "center"),
    p("Estadísticos de dispersión", align = "center"),
    br(), br(),
    h4("Las dimensiones del fichero escogido son:"),
    fluidRow(column(2, br()),
      column(6, p("Número de observaciones:")),
      column(4, verbatimTextOutput("dimensionesFilas", placeholder = T))
    ),
    fluidRow(column(2, br()),
      column(6, p("Número de variables:")),
      column(4, verbatimTextOutput("dimensionesColumnas", placeholder
= T))
    ),
    h4("Primeros datos del fichero:"),
    tableOutput('contents')
  ),
  column(4, br(), br(), img(src = "numeros.gif", height = 225, width = 225))
)
)
)
})

```

```

output$component2 <- renderUI({
  sidebarLayout(
    sidebarPanel(
      h4("Elige la variable de estudio del fichero"),

```

```

numericInput(inputId = "var",label = "Variable escogida:",value=1),
radioButtons("tipo",
  label="Selecciona si la variable es numÃ©rica o cualitativa:",
  choices=list("Cualitativa","NumÃ©rica"),
  selected="Cualitativa"),
conditionalPanel(condition = "input.tipo == 'NumÃ©rica'",
  radioButtons("tipoNumerica",
    label="Escoge el tipo de numÃ©rica:",
    choices=list("Continua","Discreta"),
    selected="Continua")
),
numericInput(inputId = "obs1",label = "NÃºmero de observaciones a
mostrar:",value=10)

),
mainPanel(
  fluidRow(column(10,fluidRow(style="border:1px inset;border-
color:#458cc3;background-color:#FA7457",
    column(5,h3("Primeros datos de la variable:")),
    column(5,br()),verbatimTextOutput("nombre")))),
  column(1,br())

),br(),
tableOutput('contentsV')
)
)
})

output$component3 <- renderUI({
  sidebarLayout(
    sidebarPanel(width = 3,
      h3("Tabla de frecuencias"),
      p("Frecuencia absoluta (fi)"),
      p("Frecuencia relativa (fr)"),
      p("Porcentaje: Frecuencia relativa*100"),br(),
      strong(h4(em("Si el tipo de variable es declarado errÃ³neamente puede que los
resultados no sean vÃ¡lidos."))),br()
    ),
    mainPanel(
      fluidRow(column(10,fluidRow(style="border:1px inset;border-
color:#458cc3;background-color:#FA7457",
        column(7,h3("CÃ¡lculo de tabla de frecuencias para la variable:")),
        column(3,br()),verbatimTextOutput("nombre1")))),
        column(1,br()))
      ,br(),
      conditionalPanel(condition = "input.tipo =='NumÃ©rica' && input.tipoNumerica
== 'Continua'",

```

```

        p("Escoge el número de intervalos en los que agrupar los datos:"),
        numericInput(inputId = "particion",label = " ",value=10)
    ),
    tableOutput("view")
)
)
})

```

```

output$component4 <- renderUI({
  sidebarLayout(
    sidebarPanel(width = 3,
      h3("Estadísticos de centro: "),
      strong("Mínimo y Máximo"),br(),
      strong("Cuartiles"),br(),
      strong("Percentil"),br(),
      strong("Media aritmética"),br(),
      strong("Mediana"),br(),
      strong("Moda"),br(),
      h3("Estadísticos de dispersión: "),
      strong("Rango o recorrido"),
      strong("Varianza"),br(),
      strong("Desviación típica"),br(),
      strong("Coeficiente de variación"),br(),
      strong("Rango intercuartilico"),br(),br(),
      strong(h4(em("Si el tipo de variable es declarado erróneamente los resultados
puede que no sean válidos")))),br()
    ),
    mainPanel(
      conditionalPanel(condition = "input.tipo == 'Numérica'",
        fluidRow(column(10,fluidRow(style="border:1px inset;border-
color:#458cc3;background-color:#FA7457",
          column(7,h3("Cálculo de estadísticos para la variable:")),
          column(3,br(),verbatimTextOutput("nombre2")))),
          column(1,br()))
      ,br(),
      fluidRow(
        column(5,
          p("Mínimo y máximo:"),
          verbatimTextOutput("minmax"),
          p("Primer y tercer cuartil:"),
          verbatimTextOutput("cuartiles"),
          fluidRow(column(4,br(),p("Percentil:")),
            column(4,numericInput(inputId = "perc",label = " ",value=20))),
          verbatimTextOutput("percentil"),
          p("Media:"),
          verbatimTextOutput("media"),
          p("Mediana o segundo cuartil:"),

```

```

    verbatimTextOutput("mediana"),
    p("Moda:"),
    verbatimTextOutput("moda")
  ),
  column(5,
    p("Rango o recorrido:"),
    verbatimTextOutput("rango"),
    p("Varianza:"),
    verbatimTextOutput("varianza"),
    p("Desviaci3n t3-pica: "),
    verbatimTextOutput("desviacion"),
    p("Coeficiente de variacion: "),
    verbatimTextOutput("cv"),
    p("Rango intercuartilico: "),
    verbatimTextOutput("rangointercuartilico"))
  )
),
conditionalPanel(condition = "input.tipo == 'Cualitativa'",
  h4("Para obtener resultados en esta secci3n la variable tiene que ser
declarada num3rica"))
))
})

output$component5 <- renderUI({
  sidebarLayout(
    sidebarPanel(width=3,
      wellPanel(style= "background:white",
        img(src = "sectores.gif", height = 150, width = 200), br(),
        h5("Gr3fico de barras"),
        h5("Gr3fico de sectores"),
        h5("Diagrama de cajas y bigotes"),
        h5("Histograma"),
        h5("Pol3gono de frecuencias"),
        h4(strong(em("Si se ha declarado err3neamente el tipo de variable los
resultados puede que no sean v3lidos"))),br(),br(),
        h6("Dependiendo del tipo de variable declarado se dibujaran unos gr3ficos
u otros"),
        h6("Para variables num3ricas continuas el diagrama de varillas y sectores
se dibujara cuando n < 15")
      )
    ),
    mainPanel(
      fluidRow(column(10,fluidRow(style="border:1px inset;border-
color:#458cc3;background-color:#FA7457",
        column(7,h3("Gr3ficos para la variable:"),
        column(3,br(),verbatimTextOutput("nombre3")))),

```

```

        column(1,br())
    ),
    conditionalPanel(condition = "input.tipo == 'Cualitativa'",
        h4("Gráfico de barras"),
        plotOutput(outputId = "barras"),
        h4("Gráfico de sectores"),
        plotOutput(outputId = "sectores")
    ),
    conditionalPanel(condition = "input.tipo == 'Numérica' && input.tipoNumerica
== 'Discreta'",
        h4("Gráfico de barras"),
        plotOutput(outputId = "barrasD"),
        h4("Gráfico de sectores"),
        plotOutput(outputId = "sectoresD")
    ),
    conditionalPanel(condition = "input.tipo == 'Numérica' && input.tipoNumerica
== 'Continua'",
        conditionalPanel(condition = "'length(unique(datasetInput1())[1])' < 16",
            h4("Gráfico de barras"),
            plotOutput(outputId = "barrasC"),
            h4("Gráfico de sectores"),
            plotOutput(outputId = "sectoresC")
        ),
        h4("Cajas y Bigotes"),
        plotOutput(outputId = "cajas"),
        p("Para los siguientes gráficos podemos elegir el número de
particiones: "),
        sliderInput(inputId = "bins",label = " ",min = 1,max=100,value=5),
        h4("Histograma"),
        plotOutput(outputId = "histograma"),
        p("Particiones del histograma:"),
        verbatimTextOutput("clases"),
        h4("Polígono de frecuencias"),
        plotOutput(outputId = "polFrec"),
        plotOutput(outputId = "polFrecSIN")
    )
)
)
}

```

```

##### LECTURA DE DATOS -- COMPONENTE 1
#####

```

```

datasetInput <- reactive({
  req(input$file1)
  inFile <- input$file1

```

```

if(input$header == "SÃ-"){
  a <- TRUE
}else{ a <- FALSE}
db <- read.csv(inFile$datapath, header = a,
              sep = input$sep, quote = input$quote)
db <- data.frame(db)
})

```

```

datasetInput1 <- reactive({
  db <- datasetInput()
  if(input$var <= 0){
    db <- NULL
  }else if(input$var > ncol(db)){
    db <- NULL
  }else{
    nombre <- names(datasetInput())
    db <- as.data.frame(db[,input$var])
    colnames(db) <- nombre[input$var]
    db
  }
})

```

```

output$content <- renderTable({
  if(input$obs <= 0){
    NULL
  }else{
    head(datasetInput(),n=input$obs)
  }
})

```

```

output$dimensionesFilas <- renderPrint({
  cat(nrow(datasetInput()))
})

```

```

output$dimensionesColumnas <- renderPrint({
  cat(ncol(datasetInput()))
})

```

```

output$contentV <- renderTable({
  if(input$obs <=0){
    NULL
  }else{
    head(datasetInput1(),n=input$obs1)
  }
})

```

```

output$nombre <- renderPrint({

```

```

    cat(names(datasetInput1())[1])
  })

  output$nombre1 <- renderPrint({
    cat(names(datasetInput1())[1])
  })
  output$nombre2 <- renderPrint({
    cat(names(datasetInput1())[1])
  })
  output$nombre3 <- renderPrint({
    cat(names(datasetInput1())[1])
  })

##### TABLA DE FRECUENCIAS -- COMPONENTE 2
#####
  output$particionIntervalo <- renderText({
    if(input$tipo=="NumÃ©rica" && input$tipoNumerica=="Continua"){
      "Escoger nÃºmero de intervalos:"
    }
  })

  output$view <- renderTable({
    if(input$tipo == "Cualitativa"){
      fi <- table(datasetInput()[,input$var])
      fr <- round(prop.table(table(datasetInput()[,input$var])),3)
      Porcentaje <- round(fr*100,4)
      tabla <- data.frame(fi,fr,Porcentaje)
      tabla <- tabla[,-c(3,5)]
      colnames(tabla) <- c("Datos","Frecuencia absoluta","Frecuencia
relativa","Porcentaje")
      tabla
    }
    else if(input$tipo == "NumÃ©rica" && input$tipoNumerica == "Discreta"){
      fi <- table(datasetInput()[,input$var])
      fr <- round(prop.table(table(datasetInput()[,input$var])),3)
      Porcentaje <- round(fr*100,4)
      tabla <- data.frame(fi,fr,Porcentaje)
      tabla <- tabla[,-c(3,5)]
      colnames(tabla) <- c("Datos","Frecuencia absoluta","Frecuencia
relativa","Porcentaje")
      tabla
    }
    }else if(input$tipo == "NumÃ©rica" && input$tipoNumerica == "Continua"){
      a <- factor(cut(datasetInput()[,input$var],breaks = input$particion))
      fi <- (table(a))
      fr <- round(prop.table(table(a)),3)
      Porcentaje <- round(fr*100,4)
      tabla <- data.frame(fi,fr,Porcentaje)
    }
  })

```

```

    tabla <- tabla[,-c(3,5)]
    colnames(tabla) <- c("Datos","Frecuencia absoluta","Frecuencia
relativa","Porcentaje")

```

```

    tabla
  }

```

```

})

```

```

##### ESTADISTICOS -- COMPONENTE 3
#####

```

```

output$minmax <- renderPrint({
  summary(datasetInput1()[,1])[c(1,6)]
})

```

```

output$cuartiles <- renderPrint({
  summary(datasetInput1()[,1])[c(2,5)]
})

```

```

output$percentil <- renderPrint({
  quantile( datasetInput1()[,1] , prob = input$perc/100)
})

```

```

output$media <- renderPrint({
  cat(summary(datasetInput1()[,1])[4])
})

```

```

output$mediana <- renderPrint({
  cat(summary(datasetInput1()[,1])[3])
})

```

```

output$moda <- renderPrint({
  cat(Mode(datasetInput1()[,1]))
})

```

```

output$rango <- renderPrint({
  cat(max(datasetInput1()[,1])-min(datasetInput1()[,1]))
})

```

```

output$varianza <- renderPrint({
  cat(var(datasetInput1()[,1]))
})

```

```

output$desviacion <- renderPrint({
  cat(sd(datasetInput1()[,1]))
})

```



```

output$cv <- renderPrint({
  CV <- sd(datasetInput1()[,1])/mean(datasetInput1()[,1])
  CVp <- CV*100
  tabla <- cbind(CV,CVp)
  colnames(tabla) <- c("CV","CV %")
  rownames(tabla) <- c("")
  tabla
})

output$rangointercuartilico <- renderPrint({
  cat(IQR(datasetInput1()[,1]))
})

##### GRAFICOS -- COMPONENTE 4
#####

output$barras <- renderPlot({
  barplot((table(datasetInput1()[,1])), col = "blue",border = NA, main =
names(datasetInput1()[1])
})

output$barrasD <- renderPlot({
  plot((table(datasetInput1()[,1])), type = "h",col = "green", cex = 4, ylab = " ", main =
names(datasetInput1()[1])
})

output$barrasC <- renderPlot({
  plot((table(datasetInput1()[,1])), type = "h",col = "pink", cex = 4, ylab = " ", main =
names(datasetInput1()[1])
})

output$sectores <- renderPlot({
  pie(prop.table(table(datasetInput1()[,1])), main = names(datasetInput1()[1])
})

output$sectoresD <- renderPlot({
  pie(prop.table(table(datasetInput1()[,1])), main = names(datasetInput1()[1])
})

output$sectoresC <- renderPlot({
  pie(prop.table(table(datasetInput1()[,1])), main = names(datasetInput1()[1])
})

output$cajas <- renderPlot({
  boxplot(datasetInput1()[,1],col="red",main = names(datasetInput1()[1])
})

```

```

output$histograma <- renderPlot({
  bins <- seq(min(datasetInput1()[,1]),max(datasetInput1()[,1]),length.out =
input$bins+1)
  hist(datasetInput1()[,1],breaks = bins,col = "darkslategray2",xlab =
"intervalos",ylab="Frecuencias absolutas", main = names(datasetInput1())[1])
})

output$clases <- renderPrint({
  bins <- seq(min(datasetInput1()[,1]),max(datasetInput1()[,1]),length.out =
input$bins+1)
  cat(bins)
})

output$polFrec <- renderPlot({
  bins <- seq(min(datasetInput1()[,1]),max(datasetInput1()[,1]),length.out =
input$bins+1)
  h <- hist(datasetInput1()[,1],breaks = bins,col = "darkslategray2",xlab =
"intervalos",ylab="Frecuencias absolutas", main = names(datasetInput1())[1])
  polygon.freq(h,col="red",frequency=1,lwd=2)
})
output$polFrecSIN <- renderPlot({
  bins <- seq(min(datasetInput1()[,1]),max(datasetInput1()[,1]),length.out =
input$bins+1)
  h <- hist(datasetInput1()[,1],breaks = bins,col = "white",border = "white",xlab =
"intervalos",ylab="Frecuencias absolutas", main = names(datasetInput1())[1])
  polygon.freq(h,col="red",frequency=1,lwd=2)
})
}

```

```
#####
```

```
shinyApp(ui = ui, server = server)
```

## Apéndice B: Código Aplicación Shiny Descripción Bivariante

```
# install.packages("shiny", dependencies = T)
# install.packages("shinythemes", dependencies = T)
# install.packages("shinyWidgets", dependencies = T)

library(shiny)
library(shinythemes)
library(shinyWidgets)
library(ggplot2)
library(scales)

ui <- shinyUI(navbarPage(theme = shinytheme("cerulean"), "DESCRIPCIÓN DE 2
VARIABLES",
  tabPanel("Archivo", uiOutput("component1")),
  tabPanel("Variables", uiOutput("component2")),
  tabPanel("Dos variables categóricas", uiOutput("component3")),
  tabPanel("Dos variables numéricas", uiOutput("component4")),
  tabPanel("Una variable numérica y una categórica",
uiOutput("component5"))
))

server <- function(input, output) {

output$component1 <- renderUI({
  sidebarLayout(
    sidebarPanel(
      fileInput('file1', 'Lectura de fichero:',
        accept = c(
          'text/csv',
          'text/comma-separated-values',
          'text/tab-separated-values',
          'text/plain',
          '.csv',
          '.tsv'
        ),
        buttonLabel = "Buscar fichero",
        placeholder = "Ningún archivo seleccionado"
      ),
      radioButtons("header",
        label="¿El archivo contiene el nombre de la variable?",
        choices=list("Sí", "No"),
        selected="Sí"),
      radioButtons(inputId = 'sep', label='Separador de variables',
        c(Espacio = '\\ ',
          "Tabulación" = '\\t',
          Coma = ','),
```

```

      "Punto y Coma"=';'
    ),
    '\ '),
  radioButtons('quote', 'Formato',
    c('Comilla Doble'='"',
      'Comilla simple'='"',
      'Nada'=''),
    ""),
  numericInput(inputId = "obs",label = "Número de observaciones a
mostrar:",value=10),
  tags$hr()
),
mainPanel(
  fluidRow(
    column(8,
      h1("Fichero de datos"),
      p(strong("Sección Dos variables categóricas: "),em("tabla de contingencia y
grafico de barras múltiples")),
      p(strong("Sección Dos variables numéricas: "),em("tabla de contingencia,
estadísticos, recta de regresión")),
      p(strong("Sección Una variable numérica y una categórica: "),em("
diagrama de cajas múltiple, tabla con estadísticos"))
    ),
    column(3, br(),br(),img(src = "numeros.gif", height = 225, width = 225))
  ),
  h4("Las dimensiones del fichero escogido son:"),
  fluidRow(column(1,br()),
    column(4,p("Número de observaciones:")),
    column(3,verbatimTextOutput("dimensionesFilas",placeholder = T))
  ),
  fluidRow(column(1,br()),
    column(4,p("Número de variables:")),
    column(3,verbatimTextOutput("dimensionesColumnas",placeholder = T))
  ),
  h4("Primeros datos del fichero:"),
  tableOutput('contents')
)
)
})

```

```

output$component2 <- renderUI({
  sidebarLayout(
    sidebarPanel(
      h2("Eleccion de variables"),
      h3("Primera variable: ",style = "font-family: 'Source Sans Pro';"),
      numericInput(inputId = "var1",label = "Variable escogida:",value=1),
      radioButtons("tipo1",

```

```

      label="Selecciona si la variable es numérica o cualitativa:",
      choices=list("Cualitativa","Numérica"),
      selected="Cualitativa"),
conditionalPanel(condition = "input.tipo1 == 'Numérica'",
  radioButtons("tipoNumerica1",
    label="Escoge el tipo de numérica:",
    choices=list("Continua","Discreta"),
    selected="Continua")),

h3("Segunda variable: ",style = "font-family: 'Source Sans Pro'"),
numericInput(inputId = "var2",label = "Variable escogida:",value=2),
radioButtons("tipo2",
  label="Selecciona si la variable es numérica o cualitativa:",
  choices=list("Cualitativa","Numérica"),
  selected="Cualitativa"),
conditionalPanel(condition = "input.tipo2 == 'Numérica'",
  radioButtons("tipoNumerica2",
    label="Escoge el tipo de numérica:",
    choices=list("Continua","Discreta"),
    selected="Continua"))
),
mainPanel(
  fluidRow(column(5,br(),br(),br(),numericInput(inputId = "obs1",label = "Número
de observaciones a mostrar:",value=10)),
    column(5,img(src = "goma.gif", height = 200, width = 300,align="center"))),
  fluidRow(column(4,
    h3("Datos variable:"),
    tableOutput('contentsV1')
  ),
  column(4,
    h3("Datos variable:"),
    tableOutput('contentsV2')
  )
)
)
)
})

output$component3 <- renderUI({
  sidebarLayout(
    sidebarPanel(width = 3,
      p(strong(p("Tabla de doble entrada o contingencia: ")),em("Tabla que hace
referencia a dos variables")),br(),
      p(strong(p("Tabla de frecuencias marginales: ")),em("Tabla de frecuencia para
cada variable")),br(),
      strong("Gráfico de barras múltiple"),br(),br(),
      strong(h4(em("Si se ha declarado el tipo de variable erróneamente los resultados

```

```

puede que no sean válidos.")),br()
),
mainPanel(
  wellPanel(style = "background: #FFFFFF",
    conditionalPanel(condition = "input.tipo1 == 'Cualitativa' && input.tipo2 ==
'Cualitativa'",
      fluidRow(style="border:1px inset;border-color:#458cc3;background-
color:#A8D1FC",column(5, h3("Nombre variable 1: "),br(),
        h3("Nombre variable 2: ")),
        column(5,br(),verbatimTextOutput("nombrev1"),br(),
          verbatimTextOutput("nombrev2"))
      ),br(),
      h4("Tabla de doble entrada:"),
      tableOutput("view1"),
      fluidRow(column(6,h4("Marginal de X"),
        tableOutput("marg11")),
        column(6, h4("Marginal de Y"),
          tableOutput("marg21"))
      ),
      br(),
      h4("Condicionada de X por Y"),
      tableOutput("condX1"),br(),
      h4("Condicionada de Y por X"),
      tableOutput("condY1"),
      h4("Gráfico de barras multiples"),
      plotOutput(outputId = "barras"),br(),
      h4("Gráfico de barras apiladas"),
      plotOutput(outputId = "barrasApiladas"),br()
    ),
    conditionalPanel(condition = "input.tipo1 != 'Cualitativa' || input.tipo2 !=
'Cualitativa'",
      h4("Para obtener resultados en esta sección las dos variables
deben ser declaradas categóricas"))
  )
)
})

output$component4 <- renderUI({
  sidebarLayout(
    sidebarPanel(width = 3,
      p(strong(p("Tabla de doble entrada o contingencia: ")),em("Tabla que hace
referencia a dos variables")),
      p(strong(p("Tabla de frecuencias marginales: ")),em("Tabla de frecuencia para
cada variable")),
      p(strong(p("Tabla de distribución condicionada"))),
      p(strong(p("Coeficiente de correlación lineal de Pearson:")), em(" valor entre -1 y

```

```

1 que indica la intensidad y dirección de la relación entre X e Y)),
    p(strong(p("Covarianza:")),em(" valor que indica el grado de variación conjunta
de dos variables")),
    p(strong(p("Recta de regresión:")),em(" Línea recta que mejor se ajusta a la
nube de puntos")),br(),
    strong(h4(em("Si se ha declarado el tipo de variable erróneamente los resultados
puede que no sean válidos."))),br()
),
mainPanel(
    wellPanel(style = "background: #FFFFFF",
        conditionalPanel(condition = "input.tipo1 == 'Numérica' && input.tipo2 ==
'Numérica'",
            fluidRow(style="border:1px inset;border-color:#458cc3;background-
color:#A8D1FC",column(5, h3("Nombre variable 1: "),br(),
                h3("Nombre variable 2: ")),
                column(5,br(),verbatimTextOutput("nombrev3"),br(),
                    verbatimTextOutput("nombrev4"))
            ),br(),
            textOutput("particionIntervalo1"),
            if(input$tipoNumerica1=="Continua"){
                numericInput(inputId = "particionV1",label = " ",value=5)
            },
            if(input$tipoNumerica2=="Continua"){
                numericInput(inputId = "particionV2",label = " ",value=5)
            },br(),
            h4("Tabla de doble entrada:"),
            tableOutput("view"),
            br(),
            fluidRow(column(6,h4("Marginal de X"),
                tableOutput("marg1")),
                column(6, h4("Marginal de Y"),
                    tableOutput("marg2"))
            ),
            br(),
            h4("Condicionada de X por Y"),
            tableOutput("condX"),br(),
            h4("Condicionada de Y por X"),
            tableOutput("condY"),
            br(),
            h4("Tabla de cálculos intermedios:"),
            radioButtons("resultados",
                label=" ",
                choices=list("Mostrar","No mostrar"),
                selected="No mostrar"),
            tableOutput("tablaIntermedia"),
            p("Media X:"),
            verbatimTextOutput("mediaX"),

```

```

    p("Media Y:"),
    verbatimTextOutput("mediaY"),
    p("Desviaci3n T3-pica X:"),
    verbatimTextOutput("desviacionX"),
    p("Desviaci3n T3-pica Y:"),
    verbatimTextOutput("desviacionY"),
    p("Covarianza:"),
    verbatimTextOutput("cov"),
    p("Coeficiente de correlaci3n lineal de Pearson:"),
    verbatimTextOutput("cor"),
    p("Nube de puntos"),
    plotOutput(outputId = "nube"),
    br(),br(),br(),
    p("Coeficientes de la ecuaci3n de la recta de regresi3n:"),
    verbatimTextOutput("regresionX"),
    p("Gr3fico de la recta de regresi3n Y sobre X:"),
    plotOutput(outputId = "rectaX"),
    radioButtons("modificarRecta",
                label="¿Modificar ejes del gr3fico?",
                choices=list("S3-", "No"),
                selected="No"),
    conditionalPanel(condition = "input.modificarRecta == 'S3-'",
                    fluidRow(column(5,numericInput(inputId = "infX",label = "L3-mite inferior
del eje X :",value=0),
                                numericInput(inputId = "infY",label = "L3-mite inferior del eje Y
:",value=0)),
                                column(5,numericInput(inputId = "supX",label = "L3-mite superior del
eje X :",value=2),
                                numericInput(inputId = "supY",label = "L3-mite superior del eje Y
:",value=2))),
                    plotOutput(outputId = "rectaXModif")
                )
    ),
    conditionalPanel(condition = "input.tipo1 != 'Num3rica' || input.tipo2 !=
'Num3rica'",
                    h4("Para obtener resultados en esta secci3n las dos variables deben
ser declaradas num3ricas"))
    )
  )
}

```

```

output$component5 <- renderUI({
  sidebarLayout(
    sidebarPanel(width = 3,
                h4("Tabla de estad3sticos"),

```



```

    p("Estadísticos básicos con los que se construye el gráfico de cajas"),
    h4("Gráfico de cajas"),
    p("Útil para comparar."),br(),br(),
    strong(h4(em("Si se ha declarado el tipo de variable erróneamente los resultados
puede que no sean válidos."))),br()
  ),
  mainPanel(
    wellPanel(style = "background: #FFFFFF",
      conditionalPanel(condition = "(input.tipo1 == 'Numérica' && input.tipo2 ==
'Cualitativa') || (input.tipo2 == 'Numérica' && input.tipo1 == 'Cualitativa')",
        fluidRow(style="border:1px inset;border-color:#458cc3;background-
color:#A8D1FC",column(5, h3("Nombre variable 1: "),br(),
          h3("Nombre variable 2: ")),
          column(5,br(),verbatimTextOutput("nombrev5"),br(),
            verbatimTextOutput("nombrev6"))
        ),br(),
        h3("Tabla de estadísticos"),
        tableOutput("estadisticos"),
        h3("Gráfico de cajas"),
        radioButtons("orientacion",
          label="Selecciona la orientación de las cajas:",
          choices=list("Horizontal","Vertical"),
          selected="Horizontal"),
        plotOutput(outputId = "cajas")
      ),
      conditionalPanel(condition = "(input.tipo1 == 'Numérica' && input.tipo2 ==
'Numérica')",
        h4("Para obtener resultados en esta sección una variable tiene que
ser declarada numérica y la otra categórica"),
        conditionalPanel(condition = "(input.tipo1 == 'Cualitativa' && input.tipo2 ==
'Cualitativa')",
          h4("Para obtener resultados en esta sección una variable tiene que
ser declarada numérica y la otra categórica"))
        )
      )
    )
  )
}

```

```

##### LECTURA DE DATOS
#####
datasetInput <- reactive({
  req(input$file1)
  inFile <- input$file1
  if(input$header == "SÃ-"){
    a <- TRUE
  }
})

```

```

}else{ a <- FALSE}
db <- read.csv(inFile$datapath, header = a,
              sep = input$sep, quote = input$quote)
db <- data.frame(db)
})

```

```

output$contents <- renderTable({
  if(input$obs <=0){
    NULL
  }else{
    head(datasetInput(),n=input$obs)
  }
})

```

```

# VARIABLE 1
datasetInput1 <- reactive({
  db <- datasetInput()
  if(input$var1 <= 0){
    db <- NULL
  }else if(input$var1 > ncol(db)){
    db <- NULL
  }else{
    nombre <- names(datasetInput())
    db <- as.data.frame(db[,input$var1])
    colnames(db) <- nombre[input$var1]
    db
  }
})

```

```

output$contentsV1 <- renderTable({
  if(input$obs1 <=0){
    NULL
  }else{
    head(datasetInput1(),n=input$obs1)
  }
})

```

```

# VARIABLE 2
datasetInput2 <- reactive({
  db <- datasetInput()
  if(input$var2 <= 0){
    db <- NULL
  }else if(input$var2 > ncol(db)){
    db <- NULL
  }else{
    nombre <- names(datasetInput())
    db <- as.data.frame(db[,input$var2])
  }
})

```

```

    colnames(db) <- nombre[input$var2]
    db
  }
})

output$dimensionesFilas <- renderPrint({
  cat(nrow(datasetInput()))
})

output$dimensionesColumnas <- renderPrint({
  cat(ncol(datasetInput()))
})

output$contentsV2 <- renderTable({
  if(input$obs1 <=0){
    NULL
  }else{
    head(datasetInput2(),n=input$obs1)
  }
})

output$nombrev1 <- renderPrint({
  cat(names(datasetInput1())[1])
})

output$nombrev2 <- renderPrint({
  cat(names(datasetInput2())[1])
})

output$nombrev3 <- renderPrint({
  cat(names(datasetInput1())[1])
})

output$nombrev4 <- renderPrint({
  cat(names(datasetInput2())[1])
})

output$nombrev5 <- renderPrint({
  cat(names(datasetInput1())[1])
})

output$nombrev6 <- renderPrint({
  cat(names(datasetInput2())[1])
})

```

```
##### DOS VARIABLES CATEGORICAS -- COMPONENTE 3
#####
```

```
output$view1 <- renderTable({
  tabla <- ftable(datasetInput1()[,1],datasetInput2()[,1])
  tabla <- addmargins(tabla)
  tabla <- as.data.frame(tabla)
  nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
  nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
  tabla <- cbind(c(levels(nombresV1),"total"),tabla)
  colnames(tabla) <- c(" ",levels(nombresV2),"total")
  tabla
})
```

```
output$marg11 <- renderTable({
  fi <- table(datasetInput1()[,1])
  fr <- round(prop.table(table(datasetInput1()[,1])),3)
  Porcentaje <- round(fr,3)*100
  tabla <- data.frame(fi,fr,Porcentaje)
  tabla <- tabla[,-c(3,5)]
  colnames(tabla) <- c("Datos","fi","fr","Porcentaje")
  tabla
})
```

```
output$marg21 <- renderTable({
  fi <- table(datasetInput2()[,1])
  fr <- round(prop.table(table(datasetInput2()[,1])),3)
  Porcentaje <- round(fr*100,4)
  tabla <- data.frame(fi,fr,Porcentaje)
  tabla <- tabla[,-c(3,5)]
  colnames(tabla) <- c("Datos","fi","fr","Porcentaje")
  tabla
})
```

```
output$condX1 <- renderTable({
  tabla <- ftable(datasetInput1()[,1],datasetInput2()[,1])
  tabla <- prop.table(tabla,2)
  tabla <- data.frame(matrix(tabla, nrow = length(unique(datasetInput1()[,1]))))
  nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
  nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
  tabla <- cbind(c(levels(nombresV1)),tabla)
  colnames(tabla) <- c(" ",levels(nombresV2))
  tabla
})
```

```
output$condY1 <- renderTable({
  tabla <- ftable(datasetInput1()[,1],datasetInput2()[,1])
```

```

tabla <- prop.table(tabla,1)
tabla <- data.frame(matrix(tabla, nrow = length(unique(datasetInput1()[,1]))))
nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
tabla <- cbind(c(levels(nombresV1)),tabla)
colnames(tabla) <- c(" ",levels(nombresV2))
tabla
})

output$barras <- renderPlot({
  a <- data.frame(datasetInput1()[,1],datasetInput2()[,1])
  ggplot(a, aes(x=a[,1], fill=a[,2])) + geom_bar(position="dodge") + labs(x =
names(datasetInput1())[1], fill = names(datasetInput2())[1], y = " ") + theme_bw() +
theme(panel.background = element_blank ())
})

output$barrasApiladas <- renderPlot({
  a <- data.frame(datasetInput1()[,1],datasetInput2()[,1])
  ggplot(a, aes(x=a[,1], fill=a[,2])) + geom_bar(position="fill") + labs(x =
names(datasetInput1())[1], fill = names(datasetInput2())[1], y = "
")+scale_y_continuous(labels=percent)
})

##### DOS VARIABLES NUMERICAS -- COMPONENTE 4
#####

output$particionIntervalo1 <- renderText({
  if(input$tipoNumerica1 == "Continua" || input$tipoNumerica2 == "Continua"){
    "Escoger número de intervalos para la o las variables continuas:"
  }
})

output$view <- renderTable({
  if(input$tipoNumerica1 == "Discreta" && input$tipoNumerica2 == "Discreta"){
    tabla <- ftable(datasetInput1()[,1],datasetInput2()[,1])
    tabla <- addmargins(tabla)
    tabla <- as.data.frame(tabla)
    nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
    nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
    tabla <- cbind(c(levels(nombresV1),"total"),tabla)
    colnames(tabla) <- c(" ",levels(nombresV2),"total")
    tabla
  }else if(input$tipoNumerica1 == "Continua" && input$tipoNumerica2 ==
"Continua"){
    a <- factor(cut(datasetInput1()[,1],breaks = input$particionV1))
    b <- factor(cut(datasetInput2()[,1],breaks = input$particionV2))

```

```

tabla <- ftable(a,b)
tabla <- addmargins(tabla)
tabla <- as.data.frame(tabla)
nombresV1 <- as.factor(unique(sort(a)))
nombresV2 <- as.factor(unique(sort(b)))
tabla <- cbind(c(levels(nombresV1),"total"),tabla)
colnames(tabla) <- c(" ",levels(nombresV2),"total")
tabla
}else if(input$tipoNumerica1 == "Discreta" && input$tipoNumerica2 == "Continua"){
b <- factor(cut(datasetInput2()[,1],breaks = input$particionV2))
tabla <- ftable(datasetInput1()[,1],b)
tabla <- addmargins(tabla)
tabla <- as.data.frame(tabla)
nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
nombresV2 <- as.factor(unique(sort(b)))
tabla <- cbind(c(levels(nombresV1),"total"),tabla)
colnames(tabla) <- c(" ",levels(nombresV2),"total")
tabla
}
}
else if(input$tipoNumerica1 == "Continua" && input$tipoNumerica2 == "Discreta"){
a <- factor(cut(datasetInput1()[,1],breaks = input$particionV1))
tabla <- ftable(a,datasetInput2()[,1])
tabla <- addmargins(tabla)
tabla <- as.data.frame(tabla)
nombresV1 <- as.factor(unique(sort(a)))
nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
tabla <- cbind(c(levels(nombresV1),"total"),tabla)
colnames(tabla) <- c(" ",levels(nombresV2),"total")
tabla
}
}
else{NULL}
})

```

```

output$marg1 <- renderTable({
if(input$tipoNumerica1 == "Discreta"){
fi <- table(datasetInput1()[,1])
fr <- round(prop.table(table(datasetInput1()[,1])),3)
Porcentaje <- round(fr,3)*100
tabla <- data.frame(fi,fr,Porcentaje)
tabla <- tabla[,-c(3,5)]
colnames(tabla) <- c("Datos","fi","fr","Porcentaje")
tabla
}else if(input$tipoNumerica1 == "Continua"){
a <- factor(cut(datasetInput1()[,1],breaks = input$particionV1))
fi <- (table(a))
fr <- round(prop.table(table(a)),3)

```

```

Porcentaje <- round(fr*100,4)
tabla <- data.frame(fi,fr,Porcentaje)
tabla <- tabla[,-c(3,5)]
colnames(tabla) <- c("Datos","fi","fr","Porcentaje")
tabla
}
})

output$marg2 <- renderTable({
if(input$tipoNumerica2 == "Discreta"){
fi <- table(datasetInput2()[,1])
fr <- round(prop.table(table(datasetInput2()[,1])),3)
Porcentaje <- round(fr*100,4)
tabla <- data.frame(fi,fr,Porcentaje)
tabla <- tabla[,-c(3,5)]
colnames(tabla) <- c("Datos","fi","fr","Porcentaje")
tabla
}else if(input$tipoNumerica2 == "Continua"){
a <- factor(cut(datasetInput2()[,1],breaks = input$particionV2))
fi <- (table(a))
fr <- round(prop.table(table(a)),3)
Porcentaje <- round(fr*100,4)
tabla <- data.frame(fi,fr,Porcentaje)
tabla <- tabla[,-c(3,5)]
colnames(tabla) <- c("Datos","fi","fr","Porcentaje")
tabla
}
})

output$condX <- renderTable({
if(input$tipoNumerica1 == "Discreta" && input$tipoNumerica2 == "Discreta"){
tabla <- ftable(datasetInput1()[,1],datasetInput2()[,1])
tabla <- prop.table(tabla,2)
tabla <- data.frame(matrix(tabla, nrow = length(unique(datasetInput1()[,1])))
nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
tabla <- cbind(c(levels(nombresV1)),tabla)
colnames(tabla) <- c(" ",levels(nombresV2))
tabla

}else if(input$tipoNumerica1 == "Continua" && input$tipoNumerica2 ==
"Continua"){
a <- factor(cut(datasetInput1()[,1],breaks = input$particionV1))
b <- factor(cut(datasetInput2()[,1],breaks = input$particionV2))
tabla <- ftable(a,b)
tabla <- prop.table(tabla,2)
tabla <- data.frame(matrix(tabla, nrow = length(unique(a))))

```

```

nombresV1 <- as.factor(unique(sort(a)))
nombresV2 <- as.factor(unique(sort(b)))
tabla <- cbind(c(levels(nombresV1)),tabla)
colnames(tabla) <- c(" ",levels(nombresV2))
tabla
}else if(input$tipoNumerica1 == "Discreta" && input$tipoNumerica2 ==
"Continua"){
  b <- factor(cut(datasetInput2()[,1],breaks = input$particionV2))
  tabla <- ftable(datasetInput1()[,1],b)
  tabla <- prop.table(tabla,2)
  tabla <- data.frame(matrix(tabla, nrow = length(unique(datasetInput1()[,1])))
  nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
  nombresV2 <- as.factor(unique(sort(b)))
  tabla <- cbind(c(levels(nombresV1)),tabla)
  colnames(tabla) <- c(" ",levels(nombresV2))
  tabla
}
else if(input$tipoNumerica1 == "Continua" && input$tipoNumerica2 == "Discreta"){
  a <- factor(cut(datasetInput1()[,1],breaks = input$particionV1))
  tabla <- ftable(a,datasetInput2()[,1])
  tabla <- prop.table(tabla,2)
  tabla <- data.frame(matrix(tabla, nrow = length(unique(a))))
  nombresV1 <- as.factor(unique(sort(a)))
  nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
  tabla <- cbind(c(levels(nombresV1)),tabla)
  colnames(tabla) <- c(" ",levels(nombresV2))
  tabla
}
else{NULL}
})

```

```

output$condY <- renderTable({
  if(input$tipoNumerica1 == "Discreta" && input$tipoNumerica2 == "Discreta"){
    tabla <- ftable(datasetInput1()[,1],datasetInput2()[,1])
    tabla <- prop.table(tabla,1)
    tabla <- data.frame(matrix(tabla, nrow = length(unique(datasetInput1()[,1])))
    nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
    nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
    tabla <- cbind(c(levels(nombresV1)),tabla)
    colnames(tabla) <- c(" ",levels(nombresV2))
    tabla

```

```

  }else if(input$tipoNumerica1 == "Continua" && input$tipoNumerica2 ==
"Continua"){
    a <- factor(cut(datasetInput1()[,1],breaks = input$particionV1))
    b <- factor(cut(datasetInput2()[,1],breaks = input$particionV2))

```



```

tabla <- ftable(a,b)
tabla <- prop.table(tabla,1)
tabla <- data.frame(matrix(tabla, nrow = length(unique(a))))
nombresV1 <- as.factor(unique(sort(a)))
nombresV2 <- as.factor(unique(sort(b)))
tabla <- cbind(c(levels(nombresV1)),tabla)
colnames(tabla) <- c(" ",levels(nombresV2))
tabla
}else if(input$tipoNumerica1 == "Discreta" && input$tipoNumerica2 == "Continua"){
b <- factor(cut(datasetInput2()[,1],breaks = input$particionV2))
tabla <- ftable(datasetInput1()[,1],b)
tabla <- prop.table(tabla,1)
tabla <- data.frame(matrix(tabla, nrow = length(unique(datasetInput1()[,1]))))
nombresV1 <- as.factor(unique(sort(datasetInput1()[,1])))
nombresV2 <- as.factor(unique(sort(b)))
tabla <- cbind(c(levels(nombresV1)),tabla)
colnames(tabla) <- c(" ",levels(nombresV2))
tabla
}
}
else if(input$tipoNumerica1 == "Continua" && input$tipoNumerica2 == "Discreta"){
a <- factor(cut(datasetInput1()[,1],breaks = input$particionV1))
tabla <- ftable(a,datasetInput2()[,1])
tabla <- prop.table(tabla,1)
tabla <- data.frame(matrix(tabla, nrow = length(unique(a))))
nombresV1 <- as.factor(unique(sort(a)))
nombresV2 <- as.factor(unique(sort(datasetInput2()[,1])))
tabla <- cbind(c(levels(nombresV1)),tabla)
colnames(tabla) <- c(" ",levels(nombresV2))
tabla
}
}
else{NULL}
})

```

```

output$tablaIntermedia <- renderTable({
  if(input$resultados == "Mostrar"){
    Xi <- datasetInput1()[,1]
    Yi <- datasetInput2()[,1]
    Xi2 <- Xi^2
    Yi2 <- Yi^2
    XiYi <- Xi*Yi
    tabla <- data.frame(cbind(Xi,Yi,Xi2,Yi2,XiYi))
    names(tabla) <- c("X","Y","Cuadrado de X","Cuadrado de Y","Producto X por Y")
    tabla
  }else{NULL}
})

```

```

output$mediaX <- renderPrint({

```

```

    cat(summary(datasetInput1()[,1])[4])
  })

output$mediaY <- renderPrint({
  cat(summary(datasetInput2()[,1])[4])
})

output$desviacionX <- renderPrint({
  n <- length(datasetInput1()[,1])
  desv <- sd(datasetInput1()[,1])*(n-1)/n
  cat(desv)
})

output$desviacionY <- renderPrint({
  n <- length(datasetInput2()[,1])
  desv <- sd(datasetInput2()[,1])*(n-1)/n
  cat(desv)
})

output$cov <- renderPrint({
  covarianza <- cov(datasetInput1()[,1],datasetInput2()[,1])
  cat(covarianza)
})

output$cor <- renderPrint({
  correlacion <- cor(datasetInput1()[,1],datasetInput2()[,1])
  cat(correlacion)
})

output$nube <- renderPlot({
  plot(datasetInput1()[,1],datasetInput2()[,1],col = 2, pch= 16,ylab= "Variable Y",
xlab="Variable X")
})

output$regresionX <- renderPrint({
  regresion <- lm(datasetInput2()[,1]~datasetInput1()[,1])
  a <- data.frame(summary(regresion)[4])[1]
  rownames(a) <- c("Término independiente", "Pendiente")
  colnames(a) <- c(" ")
  a
})

output$rectaX <- renderPlot({
  plot(datasetInput1()[,1],datasetInput2()[,1],col = 2, pch= 16,ylab= "Variable Y",
xlab="Variable X")
  regresion <- lm(datasetInput2()[,1]~datasetInput1()[,1])
  abline(regresion)
})

```

```

})

output$rectaXModif <- renderPlot({
  plot(datasetInput1()[,1],datasetInput2()[,1],col = 2, pch= 16,ylab= "Variable Y",
xlab="Variable X",xlim=c(input$infX,input$supX),ylim=c(input$infY,input$supY))
  regresion <- lm(datasetInput2()[,1]~datasetInput1()[,1])
  abline(regresion)
})

##### UNA VARIABLE NUMERICA Y UNA CATEGORICA --
COMPONENTE 5 #####

output$cajas <- renderPlot({
  if(input$orientacion == "Horizontal"){
    horizontal = T
  }else if(input$orientacion == "Vertical"){
    horizontal = F
  }
  if(input$tipo1 == "Cualitativa" && input$tipo2 == "Numérica"){
    boxplot(datasetInput2()[,1]~datasetInput1()[,1],horizontal=horizontal,col = "blue")
  }else if(input$tipo2 == "Cualitativa" && input$tipo1 == "Numérica"){
    boxplot(datasetInput1()[,1]~datasetInput2()[,1],horizontal=horizontal,col = "blue")
  }else{NULL}
})

output$estadisticos <- renderTable({
  if(input$tipo1 == "Cualitativa" && input$tipo2 == "Numérica"){
    a <- unique(datasetInput1()[,1])
    col <- matrix(NA,nrow = length(datasetInput2()[,1]),ncol = length(a))
    for(i in 1:length(a)){
      for(j in 1:length(datasetInput2()[,1])){
        if(datasetInput1()[j,1] == a[i]){
          col[j,i] <- datasetInput2()[j,1]
        }
      }
    }
    minimo <- rep(0,length(a))
    maximo <- rep(0,length(a))
    media <- rep(0,length(a))
    mediana <- rep(0,length(a))
    PrimerCuartil <- rep(0,length(a))
    TercerCuartil <- rep(0,length(a))
    for(i in 1:length(a)){
      minimo[i] <- min(na.omit(col[,i]))
      maximo[i] <- max(na.omit(col[,i]))
      media[i] <- round(mean(na.omit(col[,i])),2)
      mediana[i] <- median(na.omit(col[,i]))
    }
  }
})

```

```

    PrimerCuartil[i] <- quantile(na.omit(col[,i]), prob=c(0.25))
    TercerCuartil[i] <- quantile(na.omit(col[,i]), prob=c(0.75))
  }
  nombres <- data.frame(unique(datasetInput1()[,1]))
  tabla <-
cbind(nombres,minimo,maximo,media,mediana,PrimerCuartil,TercerCuartil)
  tabla <- data.frame(tabla)
  colnames(tabla)<- c("Variable
cualitativa","Min","Max","Media","Mediana","Primer Cuartil","Tercer Cuartil")
  tabla
}
else if(input$tipo2 == "Cualitativa" && input$tipo1 == "NumÃ©rica"){
  a <- unique(datasetInput2()[,1])
  col <- matrix(NA,nrow = length(datasetInput1()[,1]),ncol = length(a))
  for(i in 1:length(a)){
    for(j in 1:length(datasetInput1()[,1])){
      if(datasetInput2()[j,1] == a[i]){
        col[j,i] <- datasetInput1()[j,1]
      }
    }
  }
  minimo <- rep(0,length(a))
  maximo <- rep(0,length(a))
  media <- rep(0,length(a))
  mediana <- rep(0,length(a))
  PrimerCuartil <- rep(0,length(a))
  TercerCuartil <- rep(0,length(a))
  for(i in 1:length(a)){
    minimo[i] <- min(na.omit(col[,i]))
    maximo[i] <- max(na.omit(col[,i]))
    media[i] <- round(mean(na.omit(col[,i])),2)
    mediana[i] <- median(na.omit(col[,i]))
    PrimerCuartil[i] <- quantile(na.omit(col[,i]), prob=c(0.25))
    TercerCuartil[i] <- quantile(na.omit(col[,i]), prob=c(0.75))
  }
  nombres <- data.frame(unique(datasetInput2()[,1]))
  tabla <-
cbind(nombres,minimo,maximo,media,mediana,PrimerCuartil,TercerCuartil)
  tabla <- data.frame(tabla)
  colnames(tabla)<- c("Variable
cualitativa","Min","Max","Media","Mediana","Primer Cuartil","Tercer Cuartil")
  tabla
}else{NULL}
})

```

```
#####
```

```
#####
```

```
}  
shinyApp(ui = ui, server = server)
```

## Apéndice C: Código Aplicación Shiny Probabilidad

```
# install.packages("shiny", dependencies = T)
# install.packages("shinythemes", dependencies = T)
# install.packages("TeachingDemos", dependencies = T)

library(shiny)
library(shinythemes)
library(TeachingDemos)

ui <- shinyUI(navbarPage(theme =shinytheme("darkly"),"PROBABILIDAD",
  tabPanel("PROBABILIDAD",uiOutput("component1")),
  tabPanel("BINOMIAL",uiOutput("component2")),
  tabPanel("NORMAL", uiOutput("component3")),
  tabPanel("APROXIMACI3N BINOMIAL A NORMAL", uiOutput("component4"))
))

server <- function(input, output) {

#####          PROBABILIDAD          --          COMPONENTE          1
#####

output$component1 <- renderUI({
  sidebarLayout(
    sidebarPanel(width = 0,
      fluidRow(column(8, br(),br(),br()),h1("PROBABILIDAD")),
      column(4,img(src = "probabilidad.gif", height = 200, width = 200)))
    ,br(),br()
  ),
  mainPanel(width = 12,
    wellPanel(style = "background: grey",
      h3("Contenidos"),br(),
      h4(strong("Esta aplicaci3n contiene las siguientes ventanas:")),
      h5(strong("-- Secci3n Binomial: "),em("distribuci3n de probabilidad discreta, mide
el n3mero de 3xitos en una secuencia de n ensayos independientes, con una probabilidad
fija p de ocurrencia de 3xitos entre los ensayos")),
      h5(strong("-- Secci3n Normal: "),em("distribuci3n la m3is importante y de mayor
uso de las distribuciones continuas, debido a la gran cantidad de fen3menos aleatorios que
modeliza. Identificada por dos par3metros que coinciden con su media y su desviaci3n
t3-pica")),
      h5(strong("-- Secci3n Aproximaci3n Binomial por la Normal: "),em("una
distribuci3n Binomial con valores de los par3metros grandes puede aproximar bien a una
distribuci3n Normal")),
      br(),
      h3("Simulaci3n tiradas de dados"),br(),
      fluidRow(column(5, numericInput(inputId = "ndice", label = "N3mero de dados a
tirar", min = 0, value = 4),
        numericInput(inputId = "rolls", label = "N3mero de veces que se tiran los
dados", min = 0, value = 8) )) ,
      fluidRow(column(9,plotOutput("graficoDados")),
```

```

        column(3,p("Medias de los puntos obtenidos en cada dado:")
              ,verbatimTextOutput("mediaDatos"))
    )
  )
}

##### BINOMIAL -- COMPONENTE 2
#####

output$component2 <- renderUI({
  sidebarLayout(
    sidebarPanel(
      wellPanel(style = "background: grey",
                h4("Distribución binomial"),
                p("Parámetros de la distribución binomial: "),
                numericInput(inputId = "n", label = "Número de repeticiones, n", min = 0, value =
10),
                numericInput(inputId = "p",label = "Probabilidad de éxito, p",min = 0,max = 1,
value = 0.1, step = 0.05),
                br(),
                img(src = "binomial.gif", height = 200, width = 200), br(),br(),
                p("Variable X -> B(n,p)",
                p("Función de probabilidad, P(X = x):",em(" probabilidad de que la variable aleatoria
X tome exactamente el valor x")),
                p("Función de distribución, P(X <= x):",em(" probabilidad de que una variable
aleatoria X asuma un valor inferior o igual a x"))
            )
    ),
    mainPanel(
      wellPanel(style= "background:grey",
                h3("La distribución binomial"),br(),
                p("Gráfico de la función de probabilidad"),
                plotOutput("graficoBinomial"),br(),
                fluidRow(column(5,strong("Esperanza: "),
                            verbatimTextOutput("esperanza")),
                          column(5,strong("Varianza: "),
                            verbatimTextOutput("varianza"))),
                fluidRow(column(6,br()),strong("Probabilidad de obtener exactamente ")),
                          column(3, numericInput(inputId = " exitos",label = " ", value = 0,min = 0)),
                          column(1,br()),strong(" éxitos"))
            ),
                verbatimTextOutput("probabilidad"),br(),
                fluidRow(column(6,br()),strong("Probabilidad de obtener al menos ")),
                          column(3, numericInput(inputId = " exitos1",label = " ", value = 0,min = 0)),

```

```

        column(1,br(),strong(" Éxitos"))
    ),
    verbatimTextOutput("probabilidadAlMenos"),br(),

    fluidRow(column(6,br(),strong("Probabilidad de obtener un maximo de ")),
        column(3, numericInput(inputId = " exitos2",label = " ", value = 0,min = 0)),
        column(1,br(),strong(" Éxitos"))
    ),
    verbatimTextOutput("probabilidadMucho"),br(),

    fluidRow(column(5,br(),strong("Probabilidad de obtener entre ")),
        column(2, numericInput(inputId = " exitosP",label = " ", value = 0,min = 0)),
        column(1,br(),strong(" y")),
        column(2, numericInput(inputId = " exitosS",label = " ", value = 1,min = 0)),
        column(1,br(),strong(" Éxitos"))
    ),
    verbatimTextOutput("probabilidadIntervalo"),br(),

    fluidRow(column(6,br(),strong("Valor de la variable que deja a su derecha la
    probabilidad de ")),
        column(2, numericInput(inputId = "porcentaje",label = " ", value = 0,min = 0,
    max = 1,step = 0.05))
    ),
    verbatimTextOutput("PorcentajeObs"),br(),

    fluidRow(column(6,br(),strong("Generaci3n de una muestra aleatoria de tama±o
    ")),
        column(2, numericInput(inputId = "valores",label = " ", value = 5,min = 0))
    ),
    verbatimTextOutput("muestra"),br()

)
)
)
}}

```

```

##### NORMAL -- COMPONENTE 3
#####

```

```

output$component3 <- renderUI({
  sidebarLayout(
    sidebarPanel(
      wellPanel(style= "background:grey",
        h3("Distribuci3n normal"),
        h4(em("Parámetros de la distribuci3n normal: ")),
        numericInput(inputId = "mu", label = "Media", value = 0),
        numericInput(inputId = "sigma",label = "Desviaci3n Típica", value = 1),
        br(),
        img(src = "normal.gif", height = 200, width = 200),
        br(),br()
      )
    )
  )
})

```



```

),
mainPanel(
  wellPanel(style= "background:grey",
    h3("La distribución normal"),br(),

    strong("Representación de la función de densidad "),br(),br(),
    plotOutput("graficoNormal"),

    fluidRow(column(6,br()),strong("Probabilidad de que la variable aleatoria sea menor
que ")),
      column(2, numericInput(inputId = "menor",label = " ", value = 1, step = 0.1))
    ),
    verbatimTextOutput("Pmenor"),br(),
    plotOutput("graficoNormalMenor"),br(),

    fluidRow(column(6,br()),strong("Probabilidad de que la variable aleatoria sea mayor
que ")),
      column(2, numericInput(inputId = "mayor",label = " ", value =0,step = 0.1))
    ),
    verbatimTextOutput("Pmayor"),br(),
    plotOutput("graficoNormalMayor"),br(),

    fluidRow(column(5,br()),strong("Probabilidad de que la variable aleatoria se
encuentre entre ")),
      column(2, numericInput(inputId = "menorI",label = " ", value = 0, step = 0.1)),
      column(1,br()),strong(" y"),
      column(2, numericInput(inputId = "mayorI",label = " ", value = 1,step = 0.1))
    ),
    verbatimTextOutput("probabilidadIntervaloN"),br(),
    strong("Representación de la probabilidad calculada anteriormente"),br(),br(),
    plotOutput("graficoNormalProbabilidad"),br(),

    fluidRow(column(7,br()),strong("Valor de la variable que deja a su derecha una
probabilidad de ")),
      column(2, numericInput(inputId = "porcentajeN",label = " ", value = 0.1,min = 0,
max = 1, step = 0.05))
    ),
    verbatimTextOutput("PorcentajeObsN"),br(),

    fluidRow(column(8,br()),strong("Generación de una muestra aleatoria de tamaño
")),
      column(2, numericInput(inputId = "valoresN",label = " ", value = 5,min = 0))
    ),
    verbatimTextOutput("muestraN")

  )
)
})

```

```

##### APROXIMACION -- COMPONENTE 4
#####

```

```

output$component4 <- renderUI({
  sidebarLayout(
    sidebarPanel(
      wellPanel(style= "background:grey",
        h4("Aproximaci3n de la distribuci3n binomial a la Normal"),
        p("Par3metros de la distribuci3n binomial: "),
        numericInput(inputId = "n1", label = "N3mero de repeticiones, n", min = 0, value =
50),
        numericInput(inputId = "p1",label = "Probabilidad de 3xito, p",min = 0,max = 1,
value = 0.5,step = 0.05),
        br(),img(src = "dado.gif", height = 200, width = 200),br(),br()
      )
    ),
    mainPanel(
      wellPanel(style= "background:grey",
        h4("Aproximaci3n de la distribuci3n binomial a la Normal"),br(),
        strong("Gr3fico de aproximaci3n"),br(),br(),
        textOutput("NoAprox"),br(),
        plotOutput("graficoBinomialNormal")
      )
    )
  )
})

```

```

##### PROBABILIDAD -- COMPONENTE 1
#####

```

```

output$graficoDados <- renderPlot({
  dice(rolls = input$rolls, ndice = input$ndice ,plot.it = T)
})

```

```

output$mediaDados <- renderPrint({
  tmp <- dice(rolls = input$rolls, ndice = input$ndice)
  colMeans(tmp)
})

```

```

##### BINOMIAL -- COMPONENTE 2
#####

```

```

output$esperanza <- renderPrint({
  cat(input$n*input$p)
})

```

```

output$varianza <- renderPrint({
  cat(input$n*input$p*(1-input$p))
})

```

```

output$probabilidad <- renderPrint({
  cat(dbinom(input$exitos,input$n,input$p))
})

```

```

output$probabilidadAlMenos <- renderPrint({

```

```

cat(1-pbinom(input$ exitos1-1,input$n,input$p))
})

output$probabilidadMucho <- renderPrint({
cat(pbinom(input$ exitos2-1,input$n,input$p,lower.tail = T))
})

output$probabilidadIntervalo <- renderPrint({
if(input$ exitosS >= input$ exitosP){
IC <- pbinom(c(input$ exitosP-1,input$ exitosS),input$n,input$p)
cat(IC[2]-IC[1])
}else{NULL}
})

output$PorcentajeObs <- renderPrint({
cat(qbinom(1-input$porcentaje, input$n,input$p))
})

output$muestra <- renderPrint({
cat(rbinom(input$valores,input$n,input$p))
})

output$graficoBinomial <- renderPlot({
plot(dbinom(0:input$n,input$n,input$p),type =
"h",xlab="x",ylab="P(X=x)",main=expression(paste("Funcion de probabilidad B(", n, ", ", "p,")"))))
})

##### NORMAL -- COMPONENTE 3
#####
output$Pmenor <- renderPrint({
cat(pnorm(input$menor,input$mu,input$sigma))
})

output$graficoNormalMenor <- renderPlot({
curve(dnorm(x,input$mu,input$sigma),input$mu-
(input$mu+10)/2,input$mu+(input$mu+10)/2,lwd=2, ylab =
"f(x)",main=expression(paste("Densidad de la Normal N(", mu, ", ", "sigma^2,")"))
t<-seq(input$mu-(input$mu+10)/2,input$menor,by=0.01)
x <- c(-10000000000,t,input$menor)
y <- c(0,dnorm(t,input$mu,input$sigma),0)
polygon(x,y,col="orange1")
})

output$Pmayor <- renderPrint({
cat(1-pnorm(input$mayor,input$mu,input$sigma))
})

output$probabilidadIntervaloN <- renderPrint({
if(input$mayorI >= input$menorI){
IC <- pnorm(c(input$mayorI,input$menorI),input$mu,input$sigma)
}
}

```

```

    cat(IC[1]-IC[2])
  }else{NULL}

})

output$graficoNormalProbabilidad <- renderPlot({
  if(input$menorI <=input$mayorI){
    curve(dnorm(x,input$mu,input$sigma),input$mu-
(input$mu+10)/2,input$mu+(input$mu+10)/2,lwd=2,          ylab =
"f(x)",main=expression(paste("Densidad de la Normal N(", mu, ", ", sigma^2,")))
    t<-seq(input$menorI,input$mayorI,by=0.01)
    x <- c(input$menorI,t,input$mayorI)
    y <- c(0,dnorm(t,input$mu,input$sigma),0)
    polygon(x,y,col="orange1")
  }else{NULL}
})

output$graficoNormalMayor <- renderPlot({
  curve(dnorm(x,input$mu,input$sigma),input$mu-
(input$mu+10)/2,input$mu+(input$mu+10)/2,lwd=2,          ylab =
"f(x)",main=expression(paste("Densidad de la Normal N(", mu, ", ", sigma^2,")))
  t<-seq(input$mayorI,input$mu+(input$mu+10)/2,by=0.01)
  x <- c(input$mayorI,t,10000000000)
  y <- c(0,dnorm(t,input$mu,input$sigma),0)
  polygon(x,y,col="orange1")
})

output$PorcentajeObsN <- renderPrint({
  cat(qnorm(1-input$porcentajeN, input$mu,input$sigma))
})

output$muestraN <- renderPrint({
  cat(rnorm(input$valoresN,input$mu,input$sigma))
})

output$graficoNormal <- renderPlot({
  x <- seq(1,1)
  a <- input$mu
  curve(dnorm(x,          input$mu,          input$sigma),xlim=c(input$mu-
(input$mu+10)/2,input$mu+(input$mu+10)/2),col="blue",lwd=2,xlab="x",ylab="f(x)",main=ex
pression(paste("Densidad de la Normal N(", mu, ", ", sigma^2,")))
})

##### APROXIMACION BINOMIAL A NORMAL -- COMPONENTE 4
#####
output$NoAprox <- renderText({
  if(input$n1*input$p1 < 3){
    "Esta binomial no aproxima bien a la normal. Modifica el valor de las repeticiones, n, o de la
probabilidad, p"
  }
})

```

```

output$graficoBinomialNormal <- renderPlot({
  plot(dbinom(0:(input$n1+input$n1/2),input$n1,input$p1),type
"b",xlab="x",ylab="P(X=x)"
  lines(dnorm(0:(input$n1+input$n1/2),input$n1*input$p1,sqrt(input$n1*input$p1*(1-
input$p1))),col=2)
})

}
shinyApp(ui = ui, server = server)

```

## BIBLIOGRAFIA Y/O REFERENCIAS

Boletín Oficial de Castilla y León Núm. 86, 8 de mayo de 2015, ref. "CV: BOCYL-D- 08052015-5".

Boletín Oficial del Estado Núm. 295, 10 de diciembre de 2013, ref. "BOE-A-2013- 12886".

Boletín Oficial del Estado Núm. 3, 3 de enero de 2015, ref. "BOE-A-2015- 37".

Boletín Oficial del Estado Núm. 104, 1 de mayo de 2015, ref. "BOE-A-2015- 4782".

Boletín Oficial del Estado Núm. 25, 29 de enero de 2015, ref. "BOE-A-2015- 738".

Boletín Oficial del Estado Núm.1835, 30 de julio de 2016, ref. "BOE-A-2016- 7337".

Learn Shiny [sitio web]. [Consulta 26-04-2019]. Disponible en: <https://Shiny.rstudio.com/>

Shiny-examples [sitio web]. [Consulta 25-04-2019]. Disponible en: <https://github.com/rstudio/Shiny-examples>

RStudio [sitio web]. [Consulta 26-04-2019]. Disponible en: <https://www.rstudio.com/>

*The R Project for Statistical Computing*. [Sitio web]. [Consulta 13-04-2019]. Disponible en: <https://www.r-project.org/>

Colera, J., Oliveira, M.J. and Colera, R. (2016). *Matemáticas Aplicadas a las Ciencias Sociales II*. Editorial Anaya. ISBN: 978-84-1280-8.

Colera, J., Oliveira, M.J., García, R. and Santaella, E. (2008). *Matemáticas I*. Editorial Anaya. ISBN: 978-84-667-7283-9.

Contreras, I., Lobo, B., Pérez, J., Uriondo, J. and Fernández, I. (2011). *Matemáticas, 1º ESO*. Editorial Oxford Educación. ISBN: 978-84-673-6288-6.

Contreras, I., Fernández, I., Pérez, J. and Uriondo, J. (2012). *Matemáticas, 2º ESO*. Editorial Oxford Educación. ISBN: 978-84-673-6288-6.

Contreras, I., Lobo, B., Pérez, S., Pérez, J., Santaella, E. and Fernández, I. (2011). *Matemáticas, 3º ESO*. Editorial Oxford Educación. ISBN: 978-84-673-6289-3.

Contreras, I., Fernández, I., Lobo, B., Pérez, S. and Pérez, J. (2012). *Matemáticas, 4º ESO*. Editorial Oxford Educación. ISBN: 978-84-673-7376-9.

R package engsoccerdata [sitio web]. [Consulta 10-05-2019]. Disponible en: <https://cran.r-project.org/web/packages/engsoccerdata/README.html>

Rdatasets [sitio web]. [Consulta 14-05-2019]. Disponible en: <https://vincentarelbundock.github.io/Rdatasets/datasets.html>

Beeley, C. (2013). *Web application development with R using Shiny*. Editorial Packt Publishing. ISBN: 978-1-78328-447-4

Oviedo, M. (2015). [en línea] *Shiny: crear una aplicación web interactiva desde R*. Disponible en: <https://rpubs.com/moviedo/Shiny>

Introducción a R, R-commander y Shiny [sitio web]. [Consulta 19-04-2019]. Disponible en: <http://knuth.uca.es/moodle/mod/lesson/view.php?id=2904&pageid=3662&startlastseen=no>

Revuelta, A. (2017). *R como herramienta de apoyo a la enseñanza de Estadística en Bachillerato*. Disponible en: <http://uvadoc.uva.es/handle/10324/31138>.

*CrohnD function | R Documentation*. [sitio web]. [Consulta 16-05-2019]. Disponible en: <https://www.rdocumentation.org/packages/robustbase/versions/0.93-4/topics/CrohnD>

Ruiz, M. (2004). *Manual de Estadísticas*. Disponible en: <http://www.eumed.net/cursecon/libreria/drm/drm-estad.pdf>

## Referencias de las imágenes

Dado imagen animada [imagen]. [Consulta 09-05-2019]. Recuperado de: <http://www.gifsanimados.org/img-dado-imagen-animada-0092-120792.htm>

Goma de borrar imagen animada [imagen]. [Consulta 09-05-2019]. Recuperado de: <https://www.pinterest.com.mx/pin/299278337716208592/?autologin=true>

Grafico de sectores imagen animada [imagen]. [Consulta 09-05-2019]. Recuperado de: <https://www.gifmania.com/Gif-Animados-Objetos/Imagenes-Material-de-Oficina/Graficos/>

Lanzamiento dados imagen animada [imagen]. [Consulta 09-05-2019] Recuperado de:  
[http://4.bp.blogspot.com/-x8xgpNOMieQ/Vaj6VivTP4I/AAAAAAAAABY/9qEM\\_leRvw8/s1600/3a346b536b6a6f5de274bbbff7908ec0.jpg](http://4.bp.blogspot.com/-x8xgpNOMieQ/Vaj6VivTP4I/AAAAAAAAABY/9qEM_leRvw8/s1600/3a346b536b6a6f5de274bbbff7908ec0.jpg)

Matriz números imagen animada [imagen]. [Consulta 09-05-2019]. Recuperado de:  
<http://es.phoneky.com/gif-animations/?id=s2s165144>

Pato Donald Probabilidad imagen animada [imagen]. [Consulta 09-05-2019].  
Recuperado de: <https://www.pinterest.es/pin/561120434805976182/>

Probabilidad Normal imagen animada [imagen]. [Consulta 09-05-2019]. Recuperado de:  
[https://www.google.es/search?biw=1517&bih=694&tbm=isch&sa=1&ei=RXveXKDODsrUgwe967-wBQ&q=gif+probabilidad+normal&oq=gif+probabilidad+normal&gs\\_l=img.3...4330.8404..8612...0.0..0.120.2322.4j18.....1....1..gws-wiz-img.....0j0i67j35i39j0i30j0i8i30.WEPpB5pddos#imgrc=u3nFJCbDNGVt-M:](https://www.google.es/search?biw=1517&bih=694&tbm=isch&sa=1&ei=RXveXKDODsrUgwe967-wBQ&q=gif+probabilidad+normal&oq=gif+probabilidad+normal&gs_l=img.3...4330.8404..8612...0.0..0.120.2322.4j18.....1....1..gws-wiz-img.....0j0i67j35i39j0i30j0i8i30.WEPpB5pddos#imgrc=u3nFJCbDNGVt-M:)



## FIGURAS

Figura 1: Pagina web principal de descarga R .....	17
Figura 2: Ventana principal RStudio.....	18
Figura 3: Widgets básicos de Shiny.....	20
Figura 4: Aplicación Shiny mínima .....	21
Figura 5: Encabezado Aplicación Shiny Descripción Univariante .....	23
Figura 6: Aplicación Shiny Descripción Univariante, Ventana elección archivo .....	24
Figura 7: Selección de datos externos.....	24
Figura 8: Nombre de la variable .....	25
Figura 9: Separación observaciones.....	25
Figura 10: Observaciones a mostrar .....	25
Figura 11: Aplicación Shiny Descripción Univariante, Ventana elección de variable .....	26
Figura 12: Selección de variable .....	26
Figura 13: Selección tipo de variable .....	26
Figura 14: Observaciones a mostrar .....	27
Figura 15: Aplicación Shiny Descripción Univariante, Ventana Tabla de frecuencias .....	27
Figura 16: Número de intervalos .....	27
Figura 17: Definiciones tabla de frecuencias.....	28
Figura 18: Aplicación Shiny Descripción Univariante, Ventana Estadísticos .....	28
Figura 19: Aplicación Shiny Descripción Univariante, Ventana Gráficos .....	29
Figura 20: Numero clases del histograma .....	29
Figura 21: Encabezado Aplicación Shiny Descripción Bivariante.....	30
Figura 22: Aplicación Shiny Descripción Bivariante, Ventana Elección de Archivo .....	30
Figura 23: Aplicación Shiny Descripción Bivariante, Ventana Elección Variables .....	31
Figura 24: Aplicación Shiny Descripción Bivariante, Ventana Dos variables categóricas .....	32
Figura 25: Aplicación Shiny Descripción Bivariante, Ventana Dos variables numéricas.....	33
Figura 26: Selectores del número de intervalos .....	33
Figura 27: Opción mostrar tabla .....	34
Figura 28: Recta de regresión.....	34
Figura 29: Aplicación Shiny Descripción Bivariante, Ventana Una variable numérica y una categórica .....	35
Figura 30: Opción orientación grafico de cajas .....	35
Figura 31: Encabezado Aplicación Shiny Probabilidad.....	36
Figura 32: Aplicación Shiny Probabilidad, Ventana Probabilidad.....	36
Figura 33: Aplicación Shiny Probabilidad, Ventana binomial panel izquierdo .....	37
Figura 34: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 1 .....	38
Figura 35: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 2 .....	38
Figura 36: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 3 .....	38
Figura 37: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 4 .....	39
Figura 38: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 5 .....	39
Figura 39: Aplicación Shiny Probabilidad, Ventana binomial, panel principal 6.....	39
Figura 40: Aplicación Shiny Probabilidad, Ventana Binomial, panel principal 7 .....	39
Figura 41: Aplicación Shiny Probabilidad, Ventana Normal, panel izquierdo .....	40
Figura 42: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 1.....	41
Figura 43: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 2 .....	41
Figura 44: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 3 .....	42
Figura 45: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 4 .....	42
Figura 46: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 5 .....	43
Figura 47: Aplicación Shiny Probabilidad, Ventana Normal, panel principal 6 .....	43
Figura 48: Aplicación Shiny Probabilidad, Ventana Aproximación Binomial a Normal.....	44

## TABLAS

Tabla 1: Widget básicos de Shiny.....	20
---------------------------------------	----