



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

Programación de un Web Dashboard para la presentación de estadísticas en una empresa.

*Proyecto Profesional Presentado en cumplimiento de los requisitos para obtener la
Licenciatura en Actuaría*

Autor:
Alejandro Rangel Narváez

Tutor:
M. en C. Antonio Soriano Flores

Coordinación de la Carrera de Actuaría
Departamento de Matemáticas





Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Hoja de Datos del Jurado

1. Datos del alumno

Rangel

Narváez

Alejandro

55 62 41 13 46

Universidad Nacional Autónoma de México

Facultad de Ciencias

Actuaría

093197830

2. Datos del tutor

M en C

Antonio

Soriano

Flores

3. Datos del sinodal 1

Dr

Miguel Ehécatl

Morales

Trujillo

4. Datos del sinodal 2

Act

Harim

García

Lamont

5. Datos del sinodal 3

L en CC

Marisol

Flores

Cruz

6. Datos del sinodal 4

M en F

María del Rosario

Espinosa

Tuñón

7. Datos del trabajo escrito

Programación de un Web Dashboard para la presentación de estadísticas en una empresa

72 p

2016

Declaración de Autoría

Yo, Alejandro Rangel Narvárez, declaro que este proyecto profesional titulado: "Programación de un Web Dashboard para la presentación de estadísticas en una empresa." y el trabajo presentado son totalmente de mi autoría. . Y confirmo que:

- Este trabajo es una obra original
- Este trabajo fue realizado en su totalidad para obtener un grado de licenciatura en esta Universidad.
- Ninguna parte de este trabajo ha sido previamente presentado para obtener un título o cualquier otra titulación en esta Universidad.
- Todas las figuras, tablas y datos son totalmente de mi autoría.
- He reconocido todas las principales fuentes de ayuda y nombrado todas las bibliografías usadas las cuales se encuentran en línea, están perfectamente señaladas, con su respectiva liga y son de libre acceso y consumo.

Firma: Alejandro Rangel Narvárez.

El M. en C. Antonio Soriano Flores certifica que el presente trabajo ha sido revisado y que el mismo cumple con los requisitos que debe tener un trabajo de este tipo

Firma: M. en C. Antonio Soriano Flores.

“La estadística es la gramática de la ciencia.”

Karl Pearson



Agradecimientos

A Dios:

Por haberme dado las armas necesarias para seguir adelante, la capacidad de poder estudiar y la sabiduría para entender las cosas más difíciles.

A mi Madre:

Por la oportunidad de existir, por su sacrificio en algún tiempo incomprendido, por su ejemplo de superación incansable, por su comprensión y confianza, por su amor incondicional, porque sin su apoyo no hubiera sido posible la culminación de mi carrera profesional.

A mis Hermanas:

Por su apoyo, aliento y estímulo mismos que posibilitaron la conquista de esta meta.

A mi Esposa:

Por todo el amor y el apoyo brindado, porque eres de esa clase de personas que todo lo comprenden y dan lo mejor de sí mismas sin esperar nada a cambio y porque sabes escuchar y brindar ayuda cuando es necesario.

A mi Tutor, Sinodales y Maestros:

Porque gracias a su apoyo y consejo he llegado a realizar la más grande de mis metas.

Índice general

Hoja de Datos del Jurado	III
Declaración de Autoría	V
1. Introducción y Objetivos	1
1.1. Introducción	1
1.2. Objetivos	1
1.3. Metodología	1
2. Bases de Datos	3
2.1. Las Bases de Datos	3
2.2. Modelo de BD Relacional	3
3. Web Dashboard	5
3.1. Web Dashboard	5
3.2. Herramientas para el desarrollo de un Web Dashboard.	5
3.2.1. Suites de Inteligencia de Negocios	5
3.2.2. Lenguajes de Programación.	6
4. Aplicación Web	7
4.1. Caso a Implementar	7
4.2. Metodología.	7
4.2.1. Variables y métricas	7
4.2.2. Datos	8
4.2.3. Interfaz de Usuario.	8
4.3. Recursos.	8
4.4. Implementación.	9
4.5. Resultados	30
5. Conclusiones	35
A. Análisis de Componentes Principales	37
A.1. Cálculo de las Componentes	38
A.2. Cálculo de la m-ésima componente	38
A.3. Explicación de la varianza	39
B. Análisis por K Medias	41
C. Conectar R a una BD	43
D. Script Completo R	55
Bibliografía	59

Capítulo 1

Introducción y Objetivos

1.1. Introducción

Uno de los recursos más valiosos para una empresa son sus datos históricos de operación ya que con éstos, se pueden elaborar estadísticas para conocer y medir el comportamiento del negocio, y así tomar las decisiones adecuadas que permitirán mejorar el rumbo de la compañía

En la actualidad la mayor parte de las empresas, administran y almacenan dichos datos históricos en bases de datos digitales, que son 'formatos' muy adecuados para almacenar registros individuales.

A pesar de los beneficios de esta herramienta, para el administrativo promedio de una organización, extraer información y elaborar estadísticas de estas bases de datos presentan dos principales inconvenientes:

- Operar estas bases de datos requieren conocimientos previos que el administrativo carece: Programación, configuraciones de redes, seguridad informática, firewalls¹, VPN², ODBC³ entre otros.
- Organizar grandes volúmenes de datos de millones de registros y plasmarla en reportes con Excel, PowerPoint o PDF, podrían hacer muy lenta la entrega de la información.

Para evitar esto y aumentar la rapidez y la potencia para organizar estos datos y presentar las estadísticas, es conveniente hacer uso de las nuevas tecnologías de la información y del desarrollo web para programar un Dashboard.

1.2. Objetivos

El objetivo es mostrar como el actuario, tomará los datos de una empresa almacenados en una base de datos digital y con las herramientas de las nuevas tecnologías de la información y del desarrollo web, programará un Dashboard para presentar las estadísticas.

1.3. Metodología

Este trabajo se desarrolla en dos partes:

- Parte teórica: En donde se explican las definiciones y se exponen las bases para programar un Dashboard (capítulos 2 y 3).
- Aplicación web: En esta parte se programará un Dashboard que es la aplicación que contiene las estadísticas que se pretenden mostrar (capítulo 4).

¹Es una parte de un sistema o una red que está diseñada para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas

²Virtual Private Network traducido Red Privada Virtual permite que la computadora en la red envíe y reciba datos sobre redes compartidas o públicas como si fuera una red privada

³Open Data Base Connectivity es un estándar de las bases de datos y el objetivo es hacer posible el acceder a cualquier dato desde cualquier aplicación sin importar qué sistema de gestión de bases de datos se utilice

Capítulo 2

Bases de Datos

2.1. Las Bases de Datos

Una Base de Datos (BD) es un almacén de datos, diseñado para guardar datos eficientemente de manera que podamos recuperarlos, modificarlos y darles los usos que necesitemos.

La administración de las BD se realiza con un sistema llamado DBMS (Database Management System 'Sistema de Administración de Bases de Datos'). El DBMS es un conjunto de aplicaciones de software que permite: un fácil acceso a los datos, el acceso a la información por parte de múltiples usuarios y la manipulación de los datos (insertar, eliminar, editar).

Una BD puede ser local, es decir que puede utilizarla sólo un usuario en un equipo, o puede ser distribuida, es decir que la información se almacena en equipos remotos y se puede acceder a ella a través de una red.

2.2. Modelo de BD Relacional

En general, un modelo de BD, es una descripción de algo conocido como contenedor de datos, así como los métodos para almacenar y recuperar información de esos contenedores.

Existen muchos modelos de BD, pero para propósito de este trabajo, sólo se describirá al Modelo Relacional que en la actualidad es el más utilizado en el mundo empresarial y de negocios. Las características de este modelo son:

- La información se guarda en objetos llamados **Tablas**.
- Las tablas se representan gráficamente como una estructura rectangular formada por **filas** y **columnas**.
- Cada columna representa un **campo** y cada fila representa un **registro** de la BD.
- Cada tabla debe poseer una **clave primaria**, esto es, un identificador único de cada registro compuesto por una o más columnas.
- Para establecer una relación entre dos tablas, es necesario incluir, en forma de columna, en una de ellas la clave primaria de la otra, a esta columna se le llama clave **externa** o **foránea**.
- La información puede ser recuperada, almacenada, actualizada y manipulada mediante **consultas** (queries en inglés), que ofrecen una amplia flexibilidad para administrar la información.
- El lenguaje más habitual para construir las consultas de las BD relacionales es **SQL** (Structured Query Language), un estándar implementado por los principales sistemas de gestión de BD relacionales.
- Entre los DBMS actuales más populares encontramos: Oracle¹, Microsoft SQL Server², MySQL³, PostgreSQL⁴ y DB2⁵.

¹<https://www.oracle.com/database/index.html>

²<https://www.microsoft.com/en-us/server-cloud/products/sql-server/features.aspx>

³<http://www.mysql.com/>

⁴<http://www.postgresql.org/>

⁵<http://www-01.ibm.com/software/data/db2/>

Este trabajo utilizará el DBMS Sql Server Microsoft ya que presenta una gran:

- Robustez: Los datos no se corrompen si falla el equipo o si la BD crece demasiado.
- Escalabilidad: Las BD de Microsoft SQL Server pueden crecer indefinidamente en volumen mientras se haga crecer en concordancia el hardware y los sistemas que la soportan.
- Potencia: Se pueden programar copias de seguridad sin necesidad de desconectar a los usuarios, ejecuta procesos de forma mucho más rápida y contiene registros de transacciones que permiten retroceder o analizar cambios en los datos.

Capítulo 3

Web Dashboard

3.1. Web Dashboard

Un Web Dashboard es una interfaz web con gráficos, tablas, indicadores visuales y mecanismos de alerta que son consolidados en una plataforma de información con el fin de tener una visión clara del negocio. Se utilizan para el análisis de las operaciones y la toma de decisiones. Los aspectos principales que hay que considerar al hacer un Web Dashboard son:

- Establecer el área del negocio para el cual está destinado: Ventas, Operaciones, Atención a clientes, Administración, etc.
- Los datos sobre los que vamos a trabajar.
- La estructura de la BD del Dashboard.
- La lógica de negocio del área que solicita la aplicación.
- La forma en cómo nos interesa ver las métricas para que se muestren de la forma más clara posible.

El primer paso consiste en establecer contacto con los responsables del área para conocer sus necesidades de información.

Para el segundo paso se identifican todas las BD necesarias, las tablas y los campos con los que se trabajarán.

En el tercer y cuarto paso se diseña y construye la BD que alimenta al Dashboard y que refleja la lógica de negocio.

En la quinta parte se diseña la interfaz de usuario con los elementos de análisis: tablas, gráficos, indicadores y alertas.

3.2. Herramientas para el desarrollo de un Web Dashboard.

Actualmente hay dos tipos de herramientas para el desarrollo de Web Dashboards: las Suites de Inteligencia de Negocios y los Lenguajes de programación.

3.2.1. Suites de Inteligencia de Negocios

Este tipo de software, está construido con los lenguajes de programación que describiremos más adelante. Generalmente son programas que se instalan en la computadora y se ejecutan de manera fácil y práctica, siendo así, tienen la considerable ventaja de que, para usarlos, no es necesario saber programar y la considerable desventaja de que no son gratuitos, pudiendo llegar a ser bastante costosos.

Estas suites, como programa final, tienen un carácter funcional predefinido y por tanto más limitado, lo que podría ocasionar que no se adapten perfectamente al modelo del área de negocio que se pretende medir.

Específicamente se trata de herramientas que asisten el análisis y la presentación de los datos e incluyen la funcionalidad ETL (Extracción, Transformación y Carga por sus siglas en inglés).

Los procesos ETL son un término estándar que se utiliza para referirse al movimiento y transformación de datos. Se trata del proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatearlos y cargarlos en otra BD con el objeto de analizarlos.

Las fases de un proceso ETL son:

- Extracción de los datos desde uno o varias fuentes.
- Transformación de dichos datos reformateándolos y limpiándolos cuando sea necesario.
- Carga de los datos en otra BD con el objeto de analizarlos o apoyar un proceso de negocio.

Entre las Suites más reconocidas para hacer Dashboards con procesos ETL están: Pentaho¹, JasperSoft², Tableau³, YellowFin⁴, IBM Cognos⁵, Crystal Reports⁶, BIRT Analytics⁷, entre muchos otros.

3.2.2. Lenguajes de Programación.

Sobre los lenguajes de programación, podemos decir que prácticamente se pueden construir un Dashboard con cualquiera de ellos y mencionaremos a los más reconocidos para esto y son:

- Java⁸: El lenguaje Java, integrado en un entorno de desarrollo llamado Eclipse IDE⁹ proporcionan una de las mejores y más potentes herramienta para la creación de un Web Dashboard. Esta tecnología es la que utilizará este trabajo.
- .NET¹⁰ Como tal no es un lenguaje de programación, sino un 'marco' que contiene varios lenguajes como son C, C++, C#, Visual Basic, Visual C, entre otros. Todos estos lenguajes están integrados en un entorno de desarrollo llamado Visual Studio¹¹ que junto con otras bibliotecas ayudan en la creación de Dashboards. Una desventaja de .NET y sus bibliotecas es que no es gratuito.
- Action Script¹²: Este es un lenguaje desarrollado por la empresa Adobe¹³ y está integrado en un entorno de desarrollo llamado Adobe Flash Builder¹⁴. Ayuda a elaborar Dashboards muy estéticos se interactivos en formato Flash. Este lenguaje al igual que .NET no es gratuito.
- PHP¹⁵: (Hypertext Pre-Processor) es un lenguaje de programación gratuito para el desarrollo web de contenido dinámico que también ayuda a crear Dashboards basados en este lenguaje.
- JavaScript¹⁶, JQuery¹⁷(basado en JScript) y HTML5¹⁸ son otros lenguajes de programación que ayudan a crear dashboards dinámicos y potentes.

Cuando un Dashboard se construye a partir de un lenguaje, no existe un proceso ETL predefinido y toda la implementación queda a cargo del programador. Para la construcción del Dashboard este trabajo empleará el lenguaje de programación Java.

¹<http://www.pentaho.com>

²<http://www.jaspersoft.com/>

³<http://www.tableau.com/>

⁴<https://www.yellowfinbi.com/>

⁵<http://www-01.ibm.com/software/analytics/cognos/>

⁶<http://www.crystalreports.com/mx/>

⁷<http://birtanalytics.actuate.com/>

⁸<http://www.java.com/>

⁹<http://www.eclipse.org/>

¹⁰<https://www.microsoft.com/net>

¹¹<https://www.visualstudio.com/>

¹²<https://www.adobe.com/devnet/actionscript/learning.html>

¹³<http://www.adobe.com/>

¹⁴<https://www.adobe.com/mx/products/flash-builder.html>

¹⁵<http://php.net/>

¹⁶<https://developer.mozilla.org/es/docs/Web/JavaScript>

¹⁷<http://jquery.com/>

¹⁸<https://www.w3.org/html/>

Capítulo 4

Aplicación Web

4.1. Caso a Implementar

Se implementará una aplicación tomando una muestra de datos de Ekofon Inc. Ekofon Inc es una compañía de telecomunicaciones dedicada a la comercialización de larga distancia en México cuyo principal producto son las llamadas a destinos internacionales. La muestra proviene de un cliente mayorista que compra los minutos a Ekofon y los revende bajo su propia marca.

4.2. Metodología.

Se desarrollará un Dashboard que medirá las ventas y rentabilidad para establecer si las estrategias del área comercial están incrementando las ventas y rindiendo los márgenes esperados.

- Como primer paso se definirán las variables y métricas que se pretenden mostrar en la aplicación.
- En el segundo paso se identificarán los datos necesarios y se diseñará la BD del Dashboard.
- En el tercer paso se desarrollará la interfaz de usuario.

4.2.1. Variables y métricas

Las variables que se pretenden medir son:

- **Llamadas:** Una llamada telefónica es la operación por medio de la cual dos personas se interconectan a través de un teléfono y es el principal producto que ofrece Ekofon. Los clientes establecen y generan las llamadas así como su origen, destino, duración y otras variables y toda esta información se recibe en una BD y se almacena para su posterior uso. La unidad en la que se miden las llamadas telefónicas son los minutos.
- **Minutos:** Es la unidad fundamental en la cual se dividen las llamadas y también es la unidad básica de tarificación y tienen una medida de 60 segundos naturales. La tarifa total que pagará un cliente por cada llamada se calcula multiplicando la cantidad de minutos que duró la llamada por la tarifa unitaria. Existen tablas de tarifas que utiliza la plataforma telefónica de Ekofon para determinar que tarifa unitaria tiene cada minuto dependiendo del destino final de la llamada.
- **Ingreso:** El ingreso es la remuneración que recibe la empresa por los servicios de telefonía que ofrece y se calcula multiplicando el número de minutos totales por la tarifa individual de cada minuto.
- **Costo:** A Ekofon por el tamaño de la inversión y administración de infraestructura, le es imposible contar con terminales propias en cada uno de los puntos del planeta y por ello, cuando un cliente genera una llamada, la empresa paga un costo de transferencia a un terminador para que pueda completarse dicha llamada. Este costo también se fracciona en minutos, es decir, se paga dicho costo por cada minuto generado y el costo total es el total de los minutos por su costo individual. Al igual que las tarifas existen tablas para determinar el costo de cada minuto dependiendo del destino final.
- **Margen:** Es la diferencia entre los ingresos sin IVA y los costos y nos sirve para determinar si se gana o pierde dinero. Si el margen es positivo se considera ganancia y si es negativo pérdida.
- **Margen Porcentual:** Es el cociente entre el margen y el ingreso sin IVA y nos expresa la rentabilidad obtenida por unidad, es decir expresa el rendimiento obtenido por cada peso ingresado.

Se medirá el comportamiento de las variables anteriores agrupándolas por día, semana y mes y las métricas serán:

- Series temporales: Dado que las variables se agruparán en días, semanas y meses es posible ordenarlos cronológicamente en una secuencia de datos y graficarlos para ver si las variables crecen, decrecen o se mantienen constantes a lo largo el tiempo.
- Análisis Descriptivo: Se utilizarán algunas medidas descriptivas como son medias, medianas, máximo, mínimos, rangos, así como sus frecuencias absolutas y relativas
- Análisis de Componentes Principales: Se usará esta metodología para ver si existe información redundante en las variables y así reducir las a un sistema de menor dimensión que nos permita detectar días, semanas o meses atípicos. La explicación matemática detallada de esta metodología se encuentra en el apéndice **A**.
- Análisis de K Medias: Aplicadas las componentes principales, se empleará en el nuevo sistema, un análisis de K medias que particionará las observaciones en K grupos que presenten características similares. La explicación detallada de esta metodología está en el apéndice **B**.

4.2.2. Datos

Definidas las variables y métricas, se identificará y extraerá de la BD central de Ekofon los registros necesarios y con ellos se diseñarán las tablas así como la estructura general de la BD que tendrá la aplicación.

Como segundo paso, se programará un paquete de integración de datos¹ que cada 24 horas extraerá la información de la BD de Ekofon y la insertará en la BD del Dashboard para tener actualizaciones automáticas de la información.

Por último, se diseñarán y programarán las rutinas que calcularán las métricas mostradas en el Dashboard y cada 24 horas se ejecutarán mediante una tarea programada, para que recalculen las métricas con los nuevos registros ingresados a la BD.

4.2.3. Interfaz de Usuario.

Definidos los puntos anteriores, se programará la interfaz de usuario que contendrá las gráficas y tablas presentadas en el Dashboard. Se podría decir que la interfaz de usuario en sí mismo es el Dashboard.

4.3. Recursos.

La aplicación está programada con el lenguaje Java y para su implementación se han ocupado los siguientes recursos:

- JDK SE versión 8². También se puede trabajar con la versión 7.
- Eclipse Mars 2 versión 4.5.2³. Existen varias versiones de Eclipse y las más recomendables para este tipo de aplicaciones además de la versión Mars son Indigo, Kepler o Juno.
- Bibliotecas Java ADF⁴. Que servirán para implementar la aplicación. Existen muchas bibliotecas basadas en Java para este tipo de aplicaciones y se puede usar cualquiera dependiendo de las necesidades y gustos. Entre las más reconocidas se encuentra Mojarrá Java Server Faces⁵, Apache Trinidad⁶, Rich Faces⁷, Prime Faces⁸ y Ice Faces⁹

¹Un paquete de integración de datos permite mover datos de origen a destino sin modificar los datos del origen y permitiendo hacer iteraciones y cambios de información antes de llegar al destino

²[https://msdn.microsoft.com/es-es/library/ms141134\(v=sql.120\).aspx](https://msdn.microsoft.com/es-es/library/ms141134(v=sql.120).aspx)

³<http://www.oracle.com/technetwork/java/javase/overview/index.html>

⁴<https://eclipse.org/mars/>

⁵<http://www.oracle.com/technetwork/developer-tools/adf/overview/index.html>

⁶<https://myfaces.apache.org/trinidad/>

⁷<http://richfaces.jboss.org/>

⁸<http://www.primefaces.org/>

⁹<http://www.icesoft.org/java/home.jsf>

- Base de Datos SQL Server de Microsoft 2014. Se puede utilizar cualquier otra BD relacional como MySQL u Oracle o con la que se esté más familiarizado.
- R statistics versión 3.2.2¹⁰. Que servirá para el cálculo de los análisis descriptivos, de componentes principales y k medias.
- Glassfish Server Open Source version 4.1¹¹. El servidor de aplicaciones en el cual se publicará la aplicación para que pueda ser accesible por medio de un navegador a través de internet.

4.4. Implementación.

Se definieron las variables a medir: llamadas, minutos, ingreso, costo, margen y margen porcentual (sec 4.2.1). En general, las variables se definen dependiendo el área a la que esté destinada la aplicación.

Se definieron las métricas que se aplicarán a las variables (sec 4.2.1): Series temporales, análisis descriptivo, componentes principales y K medias. Las métricas dependerán totalmente de las necesidades de los usuarios.

Se descargó, instaló y configuró todas las herramientas necesarias explicadas en la sección 4.3.

Descargadas y configuradas las herramientas se diseñó la estructura de la BD del Dashboard y se migró la información de la BD de Ekofon a esta nueva BD.

La BD del dashboard constituye una tabla central llamada 'tblMuestra' en donde día a día se van concentrando cada uno de los registros individuales (fig 4.1).

Puesto que los registros provienen de una BD normalizada se puede constatar mediante consultas la inexistencia de duplicados, además de que todos los datos en cada columna son del mismo tipo. Nótese en el recuadro rojo inferior derecho que al momento de redactar estas líneas había poco más de 3.1 millones de registros en la muestra.

	Fecha	Origen	Terminacion	ZonaTerminacion	Minutos	Ingreso	Costo
1	2015-01-08 23:27:43.000	5513147577	0018174543832	United States - Tennessee - Memphis	37	12.3321	2.569
2	2015-01-08 19:57:23.000	8332742186	015563818514	Mexico - Ciudad Cuauhtemoc	0	0	0.189
3	2015-01-08 21:25:56.000	2222841784	015526300483#	Mexico - Ciudad Cuauhtemoc	17	22.6661	0.714
4	2015-01-08 21:44:20.000	8183483041	0155274#	Mexico - Ciudad Cuauhtemoc	0	0	0
5	2015-01-08 21:45:38.000	4579480252	0018478759328#	United States - Georgia	0	0	0.189
6	2015-01-08 21:43:43.000	2222841784	0445546762228#	Mexico Cel - Ciudad Camargo	7	14	2.60925
7	2015-01-08 21:41:42.000	5558390958	0018168789036	United States - Michigan - Detroit	9	2.9997	0.616
8	2015-01-08 23:13:32.000	8183739060	0018325068366	United States - Texas	0	0	0
9	2015-01-08 23:12:49.000	4579480260	0012098255180#	United States - California	1	1.25	0.231
10	2015-01-08 23:11:02.000	5558390958	0018168789036	United States - Michigan - Detroit	4	1.3332	0.259
11	2015-01-08 23:14:12.000	5552741410	0015596461556#	United States - California	1	0.3333	0.007
12	2015-01-08 23:16:58.000	8115050203	0448187786432	Mexico Cel - Minatitlan	0	0	0
13	2015-01-08 23:18:27.000	8115050203	0448181760166	Mexico Cel - Minatitlan	1	1	0.37275
14	2015-01-08 23:18:32.000	4454574772	0017739614269	United States - Georgia - Atlanta	1	1.25	0.1295
15	2015-01-08 19:19:41.000	3477882291	592	Guinea Bissau	0	0	0.09450001

Query executed successfully. | 69.24.90.63 (12.0 SP1) | sa (59) | Tesis | 00:04:59 | 3178214 rows

FIGURA 4.1: tblMuestra

¹⁰<https://www.r-project.org/>

¹¹<https://glassfish.java.net/>

A partir de la tabla muestra se creó la tabla llamada 'tblDiarios' que da forma a las variables diarias que se desean medir (fig 4.2). Cada fila representa un día.

Dia	LlamadasTotales	Llamadas	Amin	Minutos	Ingreso	Costo	Margen	MargenPorcentual	
1	2014-12-28	5	43	40	11.4931036686075	3.36699998378754	8.12610368481998	0.707041711197279	
2	2014-12-29	297	196	2214	1885	665.074656646827	283.696001317818	381.378655329009	0.573437360027886
3	2014-12-30	305	191	2300	1943	746.277843056054	261.857748345472	484.420094710582	0.649114936505219
4	2014-12-31	452	302	3232	2717	1090.71887885702	396.565754074603	694.153124782418	0.636417997559399
5	2015-01-01	387	235	3151	2680	1050.70387705647	412.088258663658	638.615618392809	0.607797907990863
6	2015-01-02	348	253	2838	2483	886.908711184715	304.472003156319	582.436708028396	0.65670423650523
7	2015-01-03	290	200	2240	1915	802.095777721241	322.8347505091442	479.261026819799	0.597610970798752
8	2015-01-04	306	207	2527	2211	843.106117793198	345.777247868013	497.328669925186	0.589878695497521
9	2015-01-05	411	287	3427	2973	1008.42025968535	393.281003686241	615.139255987111	0.610002873384404
10	2015-01-06	425	303	3270	2843	907.756980431491	371.768253564369	535.988726867122	0.590453985396341
11	2015-01-07	428	283	3565	3092	1127.96086134582	488.416254443116	639.544606902701	0.566991842376191
12	2015-01-08	364	214	3002	2578	880.180517250094	347.72150357347	532.459013676624	0.604942967086071
13	2015-01-09	531	327	4130	3540	1279.61715865752	549.449254905805	730.167903751713	0.570614342588023
14	2015-01-10	461	273	4005	3466	1112.3115498958	484.925005637575	627.386544258221	0.564038505504142
15	2015-01-11	472	286	3763	3240	1131.22060304058	541.791250514798	589.429352525782	0.521056061869338
16	2015-01-12	515	354	4680	4113	1356.41897999814	558.804755150341	797.614124847801	0.588029359226328
17	2015-01-13	509	368	5610	5074	1572.43051349089	678.340250409441	894.09026308145	0.568603989435765
18	2015-01-14	632	438	5964	5320	1614.4166386333	923.084758256562	691.331880376739	0.428223956463922

FIGURA 4.2: tblDiarios

Las variables agrupadas por semanas y meses están dadas por las vistas¹² vwSemanales (fig 4.3) y vwMensuales (fig 4.4) construidas a partir de la tabla 'tblDiarios'. Cada fila representa una semana y un mes respectivamente.

IDSEMANA	SEMANA	Llamadas	Minutos	Ingreso	Costo	Margen	MargenPorcentual	
1	201452	2014 52	5	40	11.4931036686075	3.36699998378754	8.12610368481998	0.707041711197279
2	201501	2015 01	1584	15834	6084.88586231552	2327.29176432732	3757.5940979882	0.617529101286757
3	201502	2015 02	1973	21732	7447.46793030665	3177.35252633737	4270.11540396927	0.573364725290392
4	201503	2015 03	2700	32412	10064.5510274275	4784.95853109914	5279.59249632832	0.524573076527767
5	201504	2015 04	3145	36138	10882.3097441217	5350.3152899018	5531.9944542199	0.508347454198142
6	201505	2015 05	3380	42098	12578.9390567048	6225.58304452267	6353.35601218211	0.505078845166649
7	201506	2015 06	3871	44464	13360.0276759472	6604.13255316461	6755.89512278263	0.505679727251205
8	201507	2015 07	5082	57999	18137.2079965883	8566.65605217218	9570.55194441615	0.527675039411601
9	201508	2015 08	6742	72849	23325.170861904	12151.7707882242	11173.4000736798	0.479027576682357
10	201509	2015 09	7751	90737	29193.7592251547	15477.4742825173	13716.2849426374	0.469836201526894
11	201510	2015 10	9278	101611	31760.2813382601	16004.175274506	15756.1060637541	0.496094662888651
12	201511	2015 11	10590	122091	36544.594332952	18185.177541706	18359.4167915892	0.50238326447425
13	201512	2015 12	12512	141284	41885.2395406571	19912.1265607069	21973.1129799502	0.524602777038469
14	201513	2015 13	13471	151971	45026.2737486383	20651.6555394733	24374.618209165	0.541342113834196
15	201514	2015 14	14371	160742	46872.4215252132	20816.3935451193	26056.02798000939	0.555892508478105

FIGURA 4.3: Vista Semanal

IDMES	MES	Llamadas	Minutos	Ingreso	Costo	Margen	MargenPorcentual	
1	201412	2014 Dec	694	6585	2513.56448222851	945.48650372168	1568.07797850683	0.623846330417822
2	201504	2015 Apr	78578	839638	263746.555313468	108825.181535583	154921.373777885	0.587382787745837
3	201508	2015 Aug	157648	1537398	532440.108347556	160551.958747606	371888.149599949	0.698460059205749
4	201512	2015 Dec	156657	1488296	547636.183214059	142427.194922617	405208.988291443	0.739923695167993
5	201502	2015 Feb	22748	256801	81202.47197033	41344.4764216007	39857.9955487294	0.490847071297199
6	201501	2015 Jan	11609	135627	42734.6313777669	20008.7931437539	22725.838234013	0.531789733556385
7	201507	2015 Jul	162108	1561835	526791.143181355	165497.859477354	361293.283704001	0.685837805900949
8	201506	2015 Jun	135063	1346117	430803.457342628	154510.589604801	276292.867737827	0.641343199616164
9	201503	2015 Mar	51356	581611	173923.015416439	83423.9999420526	90499.0154743866	0.520339503415904
10	201505	2015 May	105147	1098189	342159.914314516	144781.885473976	197378.028840504	0.576859008268014
11	201511	2015 Nov	139836	1399676	500342.955801636	138023.411026427	362319.544775209	0.724142391881406
12	201510	2015 Oct	144956	1440126	509726.706473555	144556.553097384	365170.153376171	0.716403807645335
13	201509	2015 Sep	144963	1410717	486047.858438384	139566.842561164	346481.01587722	0.712853703317249
14	201604	2016 Apr	80218	785510	270219.394860648	79802.9352301152	190416.459630533	0.704673547687909
15	201602	2016 Feb	103127	1036486	347283.21221923	100055.179602631	247228.032616599	0.711891689312443

FIGURA 4.4: Vista Mensual

¹²Creación de vistas en Microsoft SQL Server

<https://msdn.microsoft.com/en-us/library/ms187956.aspx>

Posteriormente con el uso de R Statistics (apéndice C) se dará forma a las demás métricas que se insertarán en forma de tablas (fig 4.5) y con éstas se harán las vistas finales (fig 4.6) que serán leídas por la aplicación concluyendo con esto la estructura de datos.

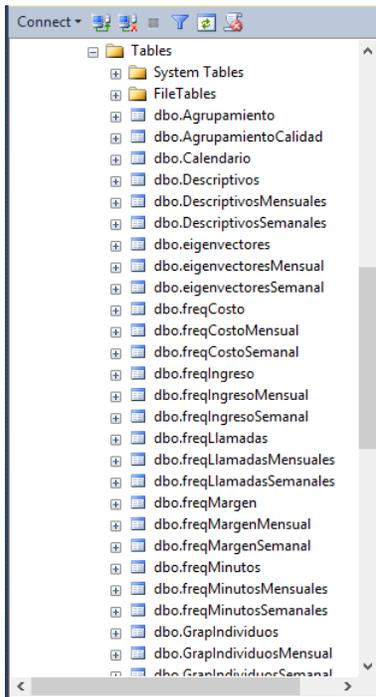


FIGURA 4.5: Tablas

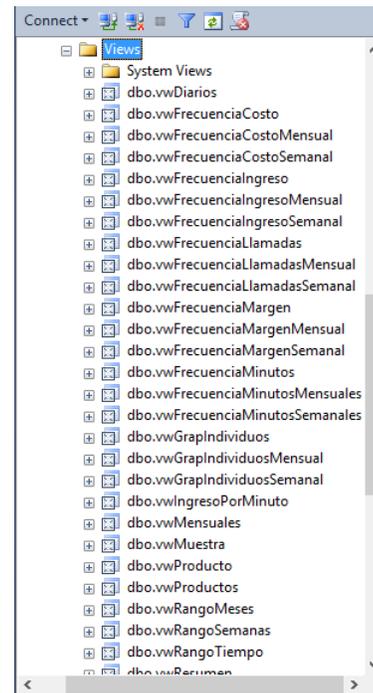


FIGURA 4.6: Vistas

Se construirá ahora la interfaz de usuario que propiamente es el Dashboard. Se ejecuta Eclipse IDE y una vez abierto se hará una nueva aplicación ADF¹³ mediante 'File' ->'New' ->'ADF Application' (fig 4.7)

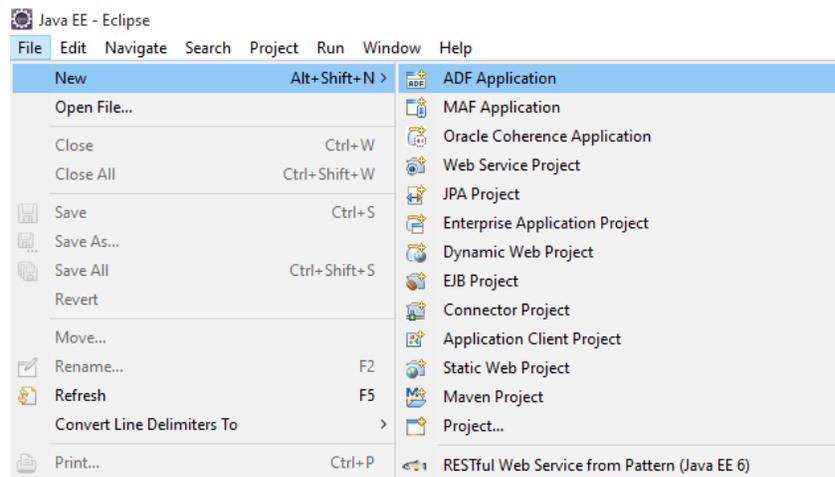


FIGURA 4.7: Nueva aplicación ADF

¹³Oracle Application Development Framework - Oracle ADF
<http://www.oracle.com/technetwork/developer-tools/adf/overview/index.html>

Se abre el wizard para configurar la aplicación (fig 4.8). En name se pondrá 'ProyectoProfesional_AlejandroRangelNarvaez', en 'Runtime Environment' se usará 'Glassfish 4' y damos click en 'New JPA Project'¹⁴

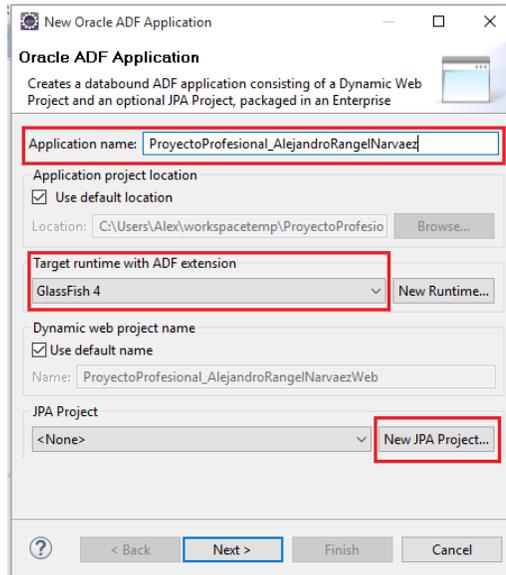


FIGURA 4.8: Nombre, runtime environment, proyecto JPA

Se abre el wizard para configurar el proyecto JPA (fig 4.9) en esta primera parte se deja todo como aparece por defecto y damos click en 'Next'.

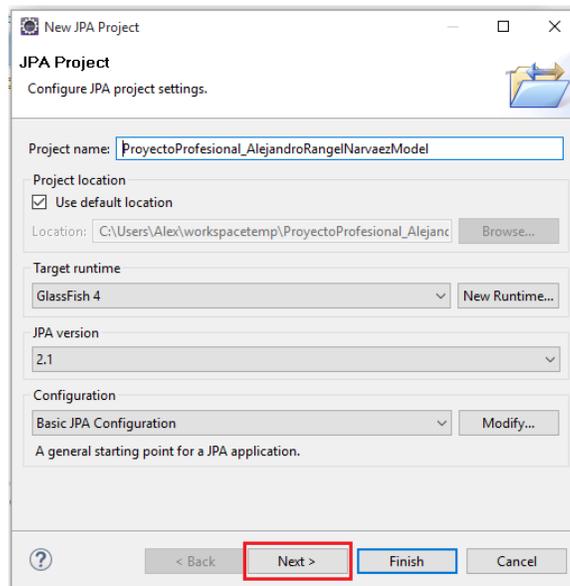


FIGURA 4.9: Nuevo proyecto JPA

¹⁴Introducción a Java Persistence API
<http://docs.oracle.com/javasee/6/tutorial/doc/bnbpz.html>

En 'Platform' se selecciona 'EclipseLink 2.5.x'¹⁵ y click en 'Add connection'¹⁶ (fig 4.10)

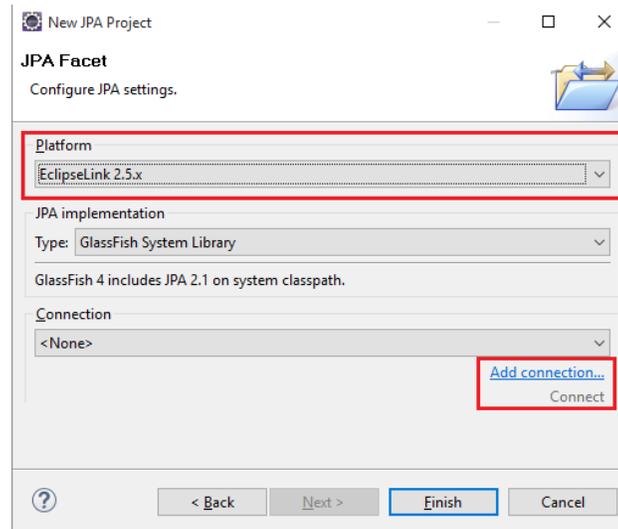


FIGURA 4.10: Plataforma

En 'Connection Profile Types' se usará 'SQL Server' en name pondremos 'Conexion' y click en 'Next' (fig 4.11)

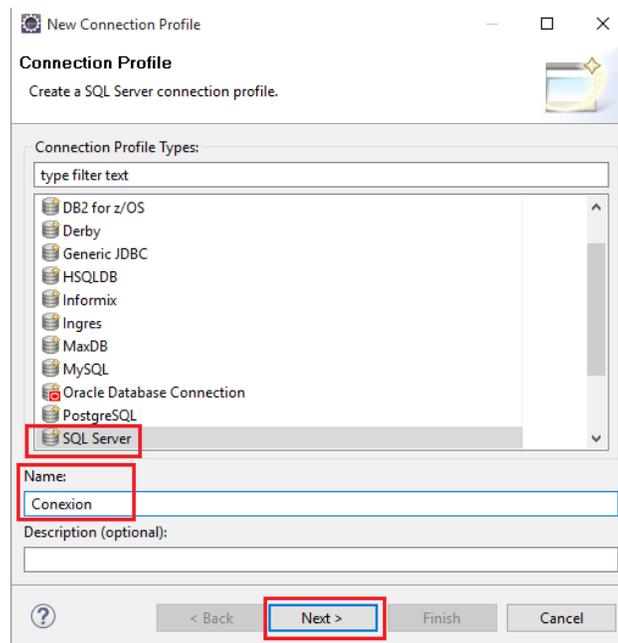


FIGURA 4.11: Conexión

¹⁵Eclipse Link

<http://www.eclipse.org/eclipselink/>

¹⁶Creación de un perfil de conexión de BD en Eclipse

<http://help.eclipse.org/kepler/index.jsp?topic=%2Forg.eclipse.datatools.connectivity.doc.user%2Fdoc%2Fhtml%2Fasc1229700344899.html>

Click en 'New Driver Definition'¹⁷ (fig 4.12)

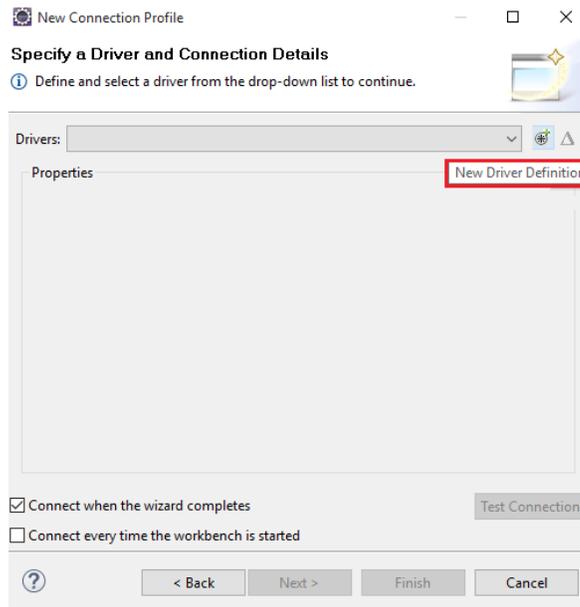


FIGURA 4.12: Driver

Se especificará 'Microsoft SQL Server 2008 JDBC Driver'¹⁸ y click en ok (fig 4.13)

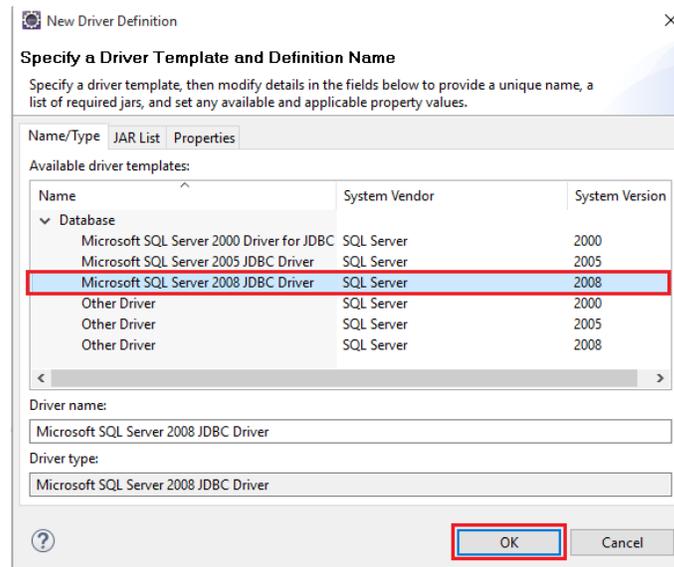


FIGURA 4.13: Especificación del driver

¹⁷Java JDBC API

<http://docs.oracle.com/javase/7/docs/technotes/guides/jdbc/>

¹⁸Driver JDBC para Microsoft SQL Server

[https://msdn.microsoft.com/en-us/library/mt484311\(v=sql.110\).aspx](https://msdn.microsoft.com/en-us/library/mt484311(v=sql.110).aspx)

Se llenan todos los campos necesarios para conectar a la BD¹⁹: Database, Host, Port Number, Username y Password (fig 4.14) y click en 'Test Connection'

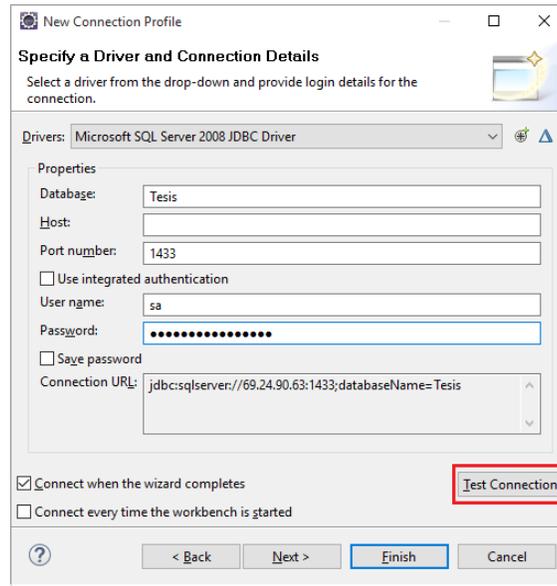


FIGURA 4.14: Detalles de la conexión

Al recibir 'Ping succeeded', click en Ok para regresar a continuar configurando el proyecto (fig 4.15)

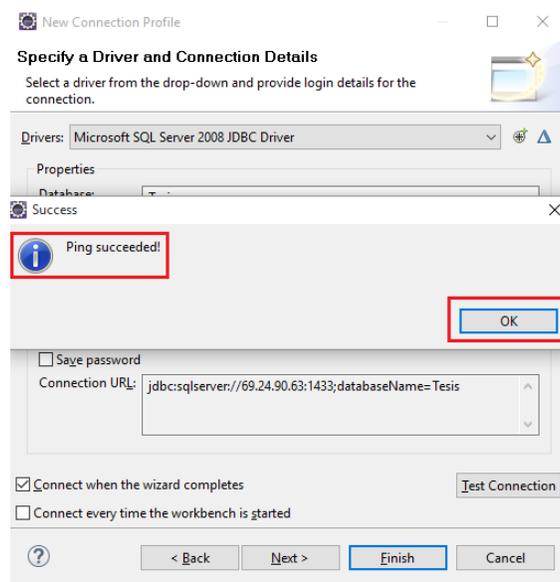


FIGURA 4.15: PING Exitoso

¹⁹Guía paso a paso para el uso de JDBC en Eclipse
<http://www.ccs.neu.edu/home/kathleen/classes/cs3200/JDBCtutorial.pdf>

Regresando al wizard inicial se observará el 'JPA Project' configurado (fig 4.16), click en 'Next'

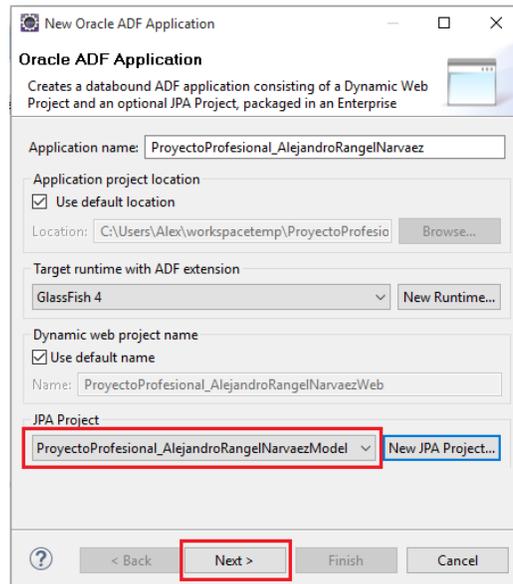


FIGURA 4.16: JPA Configurado

Se seleccionan las bibliotecas ADF²⁰ que ya deberán estar descargadas y configuradas y click en 'Finish' (fig 4.17)

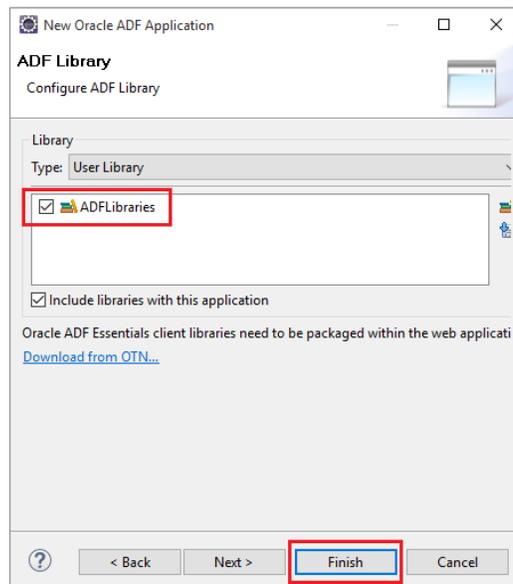


FIGURA 4.17: Bibliotecas ADF

²⁰Bibliotecas ADF

<http://www.oracle.com/technetwork/developer-tools/adf/overview/adfessentials-1719844.html>

Puesto que el proyecto se basa en el 'Modelo - Vista - Controlador'^{21 22}, en 'Project Explorer' se han creado 3 carpetas, el proyecto en sí mismo, otra con sufijo 'Model' que es donde se configura el modelo de datos y otro con sufijo 'Web' que es donde se configura la interfaz de usuario (fig 4.18).

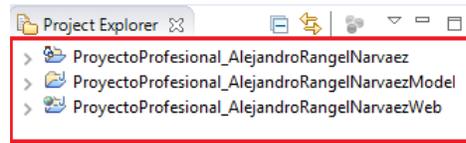


FIGURA 4.18: Proyecto

En el proyecto con sufijo 'model' se creará el paquete 'modelo.datos'²³ (fig 4.19)

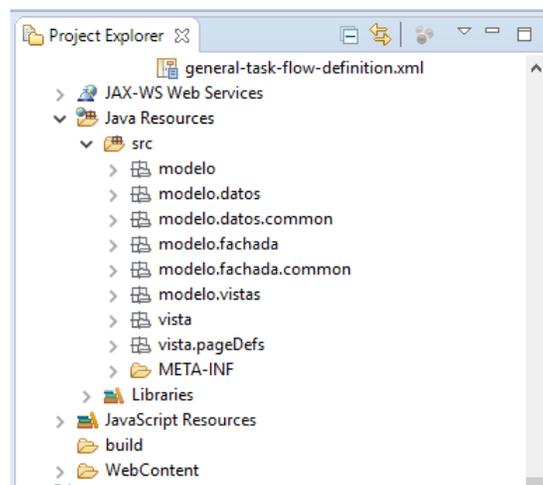


FIGURA 4.19: JPA Entities

²¹El objetivo del Modelo-Vista-Controlador (MVC) para el desarrollo de aplicaciones Java es separar claramente la funcionalidad de la aplicación en un conjunto de componentes que cooperan entre si.

²²Introducción al Modelo ADF

<https://docs.oracle.com/middleware/1212/adf/ADFCG/model1.htm#ADFCG1201>

²³Creación de paquetes en Eclipse

<http://www.vogella.com/tutorials/Eclipse/article.html#create-package>

Dentro del paquete 'modelo.datos' se crearán todos los componentes del negocio²⁴ (fig 4.20).

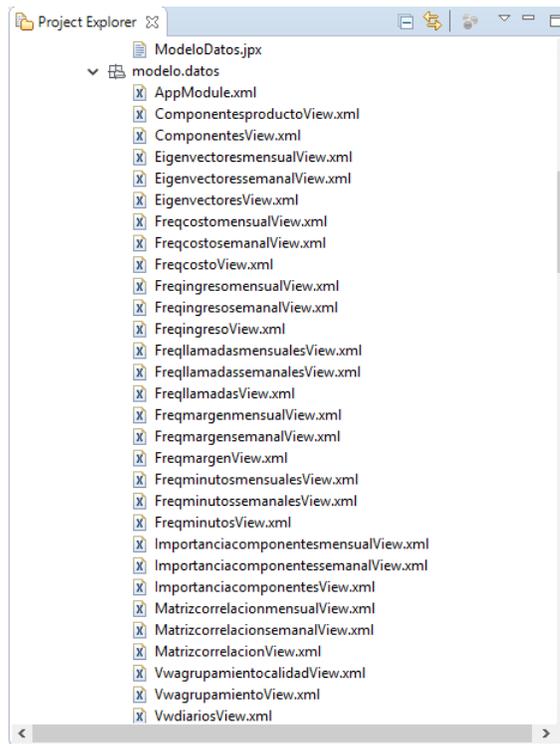


FIGURA 4.20: Modelo Datos

Se implementará ahora la parte web²⁵. Click derecho en la carpeta con sufijo web, 'New' ->'ADF Page'²⁶ (fig 4.21)

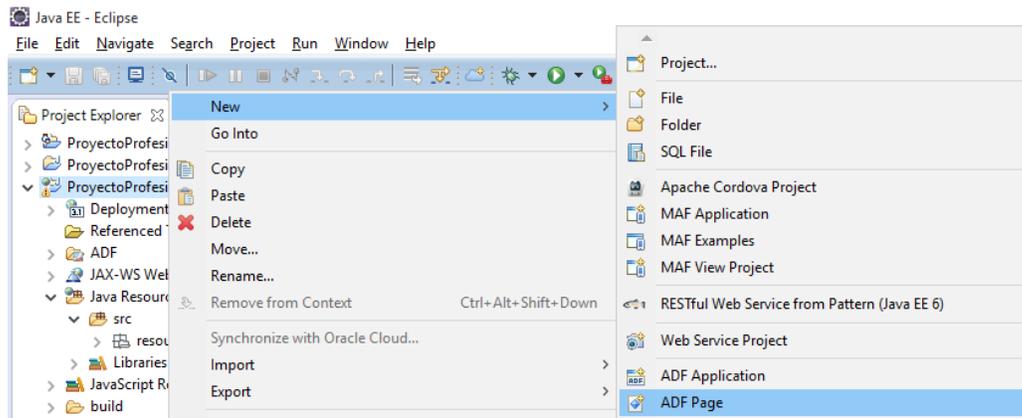


FIGURA 4.21: Nueva Página ADF

²⁴Introducción a ADF Business Components

<https://docs.oracle.com/middleware/1212/adf/ADFFD/bcintro.htm#ADFFD123>

²⁵Introducción a las interfaces web con ADF

http://docs.oracle.com/cd/E15051_01/web.1111/b31974/web_getstarted.htm#ADFFD559

²⁶Introducción a ADF Faces

<http://docs.oracle.com/middleware/12211/adf/concepts/GUID-ED22CB41-8164-47A3-8A80-BCA3C6ECABAB.htm#ADFCG201>

Se selecciona la carpeta Web Content y en 'File Name' se pondrá 'principal.jsf'²⁷ y click en finish (fig 4.22)

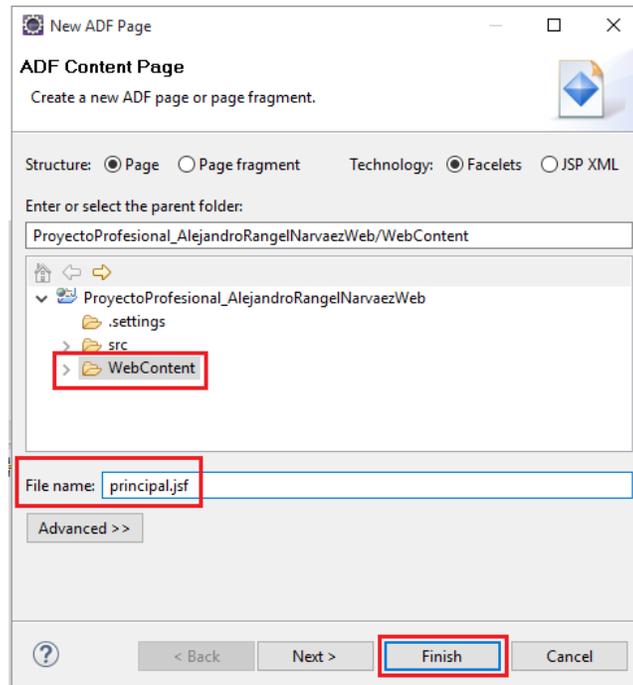


FIGURA 4.22: Principal.jsf

Se hace otra vez click derecho en la carpeta con sufixo 'Web' ->'New' ->'ADF Task Flow'²⁸ (fig 4.23)

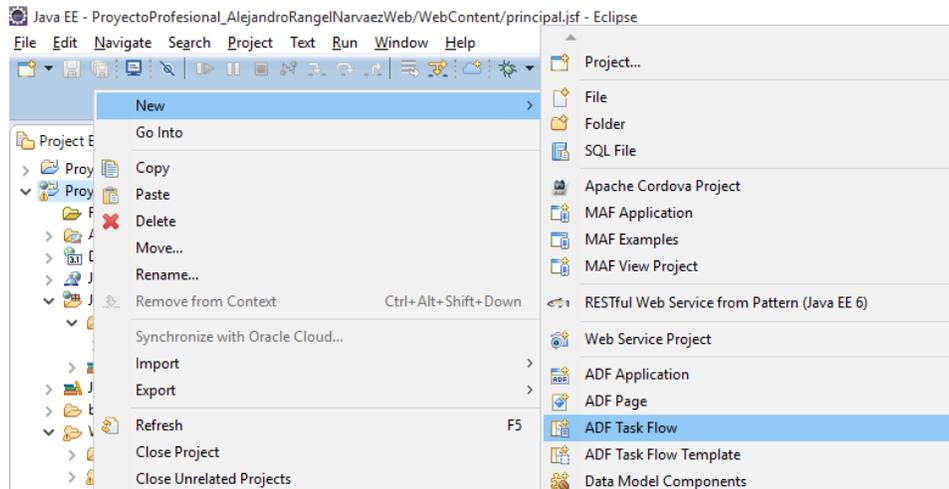


FIGURA 4.23: Task Flow

²⁷ Guía de Introducción a la tecnología Java Server Faces

<https://docs.oracle.com/javasee/7/tutorial/jsf-intro.htm>

²⁸ Introducción a ADF Task Flow

https://docs.oracle.com/cd/E23943_01/web.1111/b31974/taskflows.htm#ADFFD1631

En file name se usará 'general-task-flow-definition', se dejará en la carpeta 'WEB-INF' y click en 'Finish' (fig 4.24)

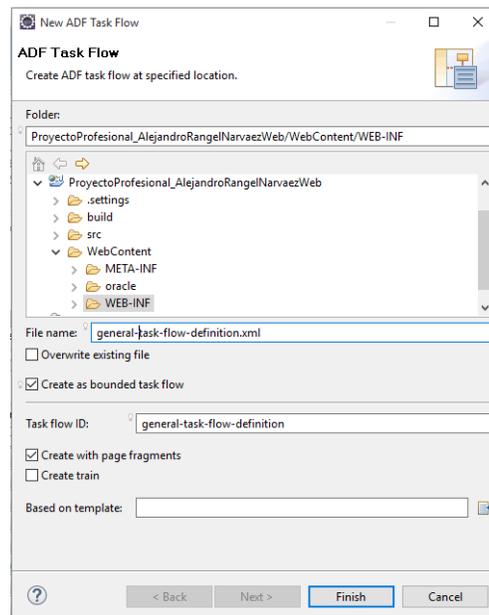


FIGURA 4.24: Configuración General taskflow

Se abre el task flow que se acaba de crear y de 'Palette'²⁹ se añadirá con 'drag and drop' un 'Wildcard Control Flow Rule' (fig 4.25)

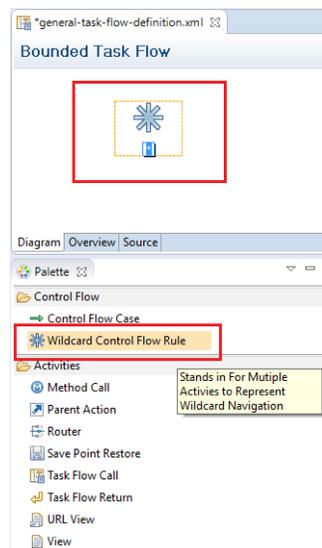


FIGURA 4.25: General taskflow

²⁹Tutorial editor visual Eclipse

<https://eclipse.org/articles/Article-VE-Custom-Widget/customwidget.html>

Ahora se añadirán 3 'Views' con 'Drag and Drop' desde la paleta que serán la vista diaria, semanal y mensual llamando a cada una vwDiarios, vwSemanales, vwMensuales (fig 4.26)

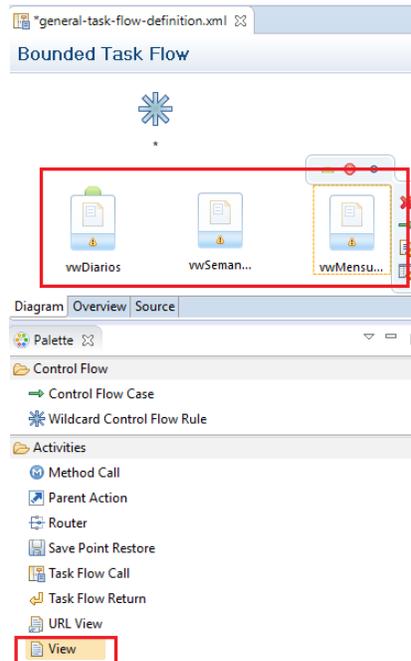


FIGURA 4.26: Vistas

Se añaden 3 'Control Flow'³⁰ que llamaremos 'irDiarios', 'irSemanales' e 'irMensuales' como se muestra en la fig 4.27.

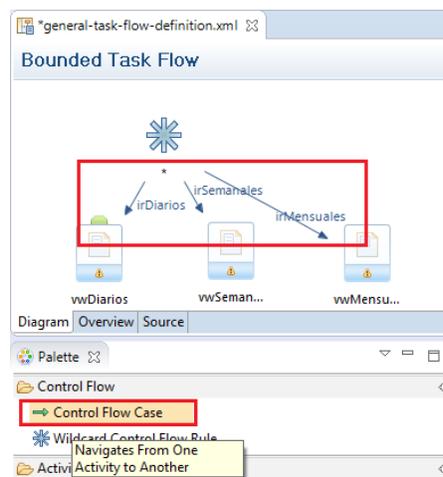


FIGURA 4.27: Control Flow

³⁰Introducción a ADF Controller Task Flow
<https://docs.oracle.com/middleware/1212/adf/ADFCG/controller.htm#ADFCG183>

Doble click en vwDiarios para crear un facelet³¹ que se llamará 'vwDiarios.jsff' el cual será el contenedor de la vista diaria (fig 4.28)

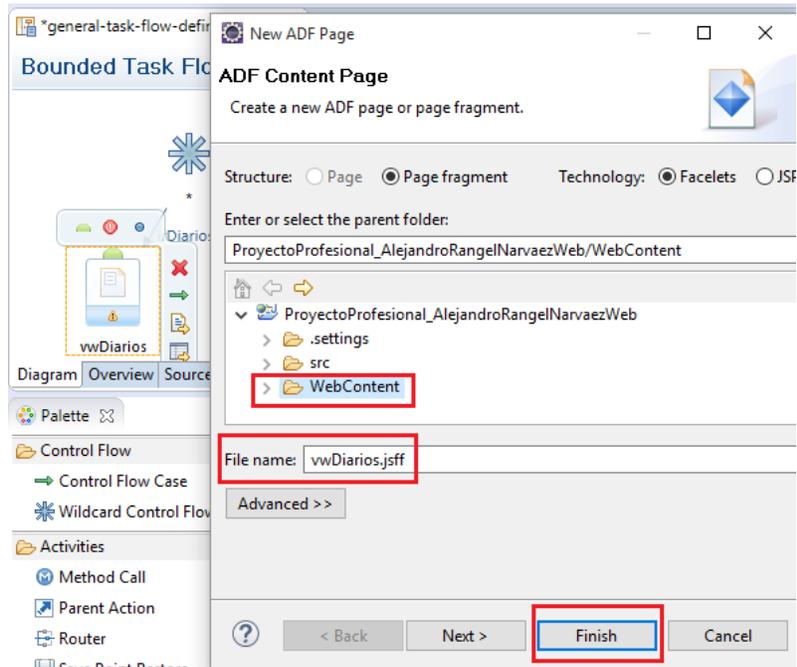


FIGURA 4.28: vwDiarios.jsff

Se hace lo mismo con las otras dos 'View' para crear los contenedores de la vista semanal y mensual llamadas 'vwSemanales.jsff' y 'vwMensuales.jsff' (fig 4.29)

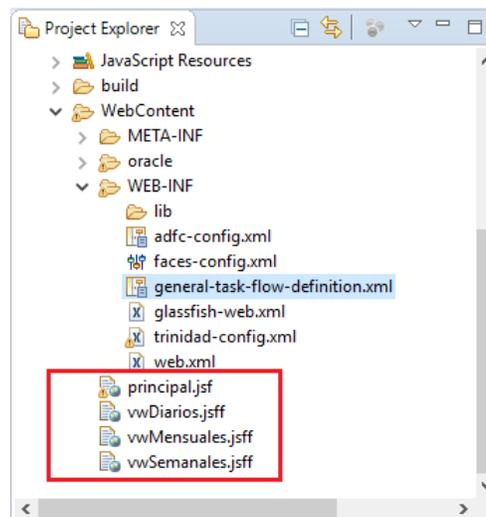
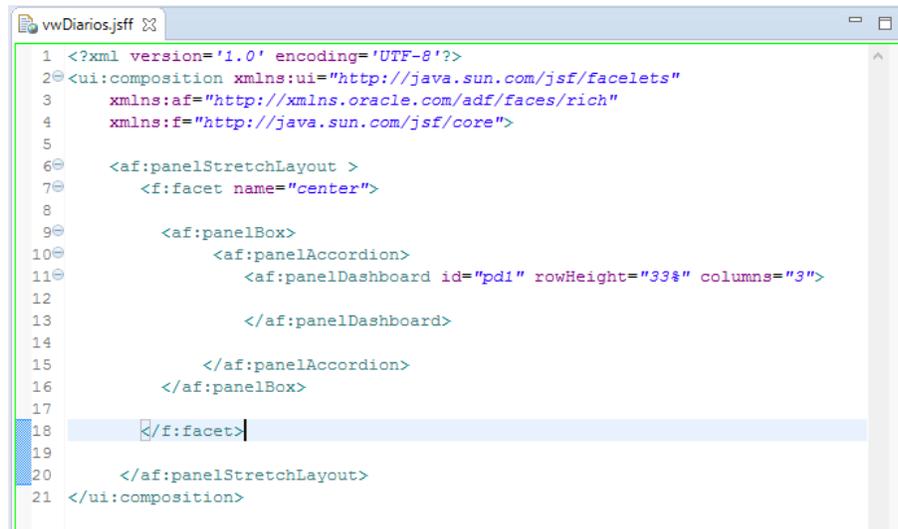


FIGURA 4.29: Vista Diaria, vista semanal, vista mensual

³¹Introducción a Java Facelets
<http://docs.oracle.com/javasee/6/tutorial/doc/giepx.html>

Se abre el archivo llamada vwDiarios.jsff para ir agregando los elementos web³². Se insertarán en este orden un 'Panel Stretch Layout' con un 'Facet' llamado 'Center', un 'Panel Box', un 'Panel Accordion' y un 'Panel Dashboard' de 3 columnas con cada Fila con 33 % de altura (fig 4.30)



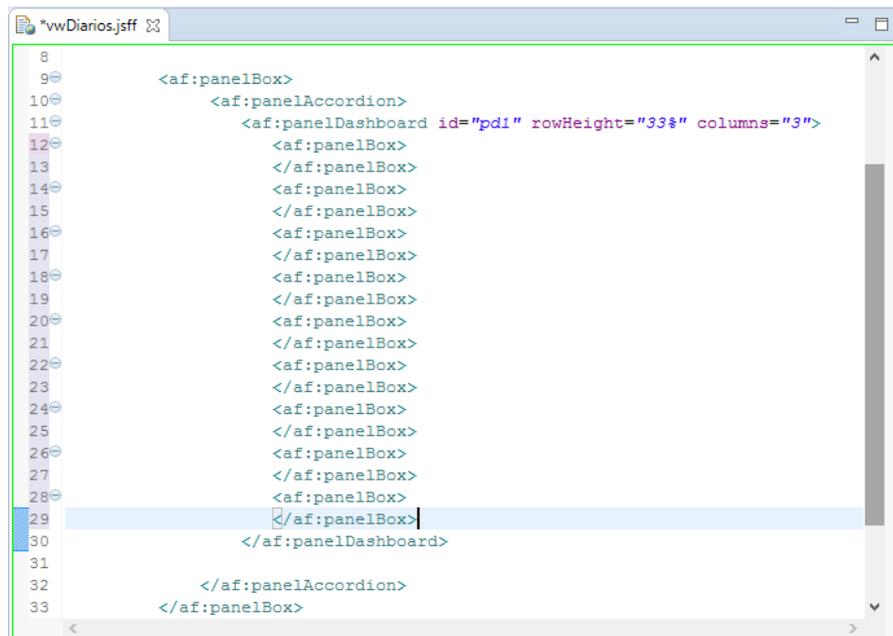
```

1 <?xml version='1.0' encoding='UTF-8'?>
2 <ui:composition xmlns:ui="http://java.sun.com/jsf/facelets"
3   xmlns:af="http://xmlns.oracle.com/adf/faces/rich"
4   xmlns:f="http://java.sun.com/jsf/core">
5
6   <af:panelStretchLayout >
7     <f:facet name="center">
8
9       <af:panelBox>
10        <af:panelAccordion>
11          <af:panelDashboard id="pd1" rowHeight="33%" columns="3">
12
13          </af:panelDashboard>
14
15          </af:panelAccordion>
16        </af:panelBox>
17
18      </f:facet>
19
20    </af:panelStretchLayout>
21  </ui:composition>

```

FIGURA 4.30: Layout

Se insertan ahora 9 'Panel Boxes' que contendrá cada uno un gráfico o una tabla que se mostrará en la aplicación (fig 4.31)



```

8
9   <af:panelBox>
10     <af:panelAccordion>
11       <af:panelDashboard id="pd1" rowHeight="33%" columns="3">
12         <af:panelBox>
13         </af:panelBox>
14         <af:panelBox>
15         </af:panelBox>
16         <af:panelBox>
17         </af:panelBox>
18         <af:panelBox>
19         </af:panelBox>
20         <af:panelBox>
21         </af:panelBox>
22         <af:panelBox>
23         </af:panelBox>
24         <af:panelBox>
25         </af:panelBox>
26         <af:panelBox>
27         </af:panelBox>
28         <af:panelBox>
29         </af:panelBox>
30       </af:panelDashboard>
31
32     </af:panelAccordion>
33   </af:panelBox>

```

FIGURA 4.31: Panel Boxes

³²Oracle ADF Faces Rich Client Components

<http://www.oracle.com/technetwork/developer-tools/adf/overview/index-092391.html>

Se insertará ahora la primera tabla³³ haciendo las ligas necesarias al modelo de datos (fig 4.32)

```

52 <f:facet name="toolbar"/>
53 <f:facet name="statusbar"/>
54 <af:table value="#{bindings.Vwresumen2View1.collectionModel}" var="row"
55       rows="#{bindings.Vwresumen2View1.rangeSize}"
56       emptyText="#{bindings.Vwresumen2View1.viewable ? 'No data to display.' : 'Access Denied.'}"
57       rowBandingInterval="0"
58       selectedRowKeys="#{bindings.Vwresumen2View1.collectionModel.selectedRow}"
59       selectionListener="#{bindings.Vwresumen2View1.collectionModel.makeCurrent}"
60       rowSelection="single" fetchSize="#{bindings.Vwresumen2View1.rangeSize}" id="t1"
61       summary="Resumen" width="100%" columnStretching="multiple">
62   <af:column sortProperty="#{bindings.Vwresumen2View1.hints.Var1.name}" sortable="true"
63     headerText="Variable" id="c1" rowHeader="unstyled" width="100%">
64     <af:outputText value="#{row.Var1}" shortDesc="#{bindings.Vwresumen2View1.hints.Var1.tooltip}"
65       id="ot1">
66       <af:convertNumber maxFractionDigits="0"/>
67     </af:outputText>
68   </af:column>
69   <af:column sortProperty="#{bindings.Vwresumen2View1.hints.Total.name}" sortable="true"
70     headerText="#{bindings.Vwresumen2View1.hints.Total.label}" id="c2" width="100%"
71     align="right">
72     <af:outputText value="#{row.Total}" shortDesc="#{bindings.Vwresumen2View1.hints.Total.tooltip}"
73       id="ot2">
74     <af:convertNumber pattern="#{bindings.Vwresumen2View1.hints.Total.format}"
75       maxFractionDigits="2"/>
76     </af:outputText>
77   </af:column>
78   <af:column sortProperty="#{bindings.Vwresumen2View1.hints.Media.name}" sortable="true"

```

FIGURA 4.32: Tabla

Ahora se insertará el primer gráfico de barras³⁴ haciendo los respectivas ligas necesarias con el modelo de datos (fig 4.33).

```

127 <af:gridRow marginTop="5px" height="100%" marginBottom="5px" id="gri">
128 <af:gridCell marginStart="5px" width="50%" id="gci" halign="stretch" valign="stretch"
129   landmark="banner">
130   <dvt:lineChart id="barChart1" var="row" value="#{bindings.VwdiariosView1.collectionModel}"
131     timeAxisType="enabled" seriesEffect="color" zoomAndScroll="live" dataCursor="on">
132     <dvt:chartLegend id="c11" position="top"/>
133     <f:facet name="dataStamp">
134       <dvt:chartDataItem id="d11" value="#{row.Llamadas}" group="#{row.Dia}"
135         series="#{bindings.VwdiariosView1.hints.Llamadas.label}"/>
136     </f:facet>
137     <dvt:chartXAxis id="cxa1" title="Dia"/>
138     <dvt:chartYAxis id="cya1">
139       <dvt:chartTickLabel id="ct11" scaling="none">
140         <af:convertNumber maxFractionDigits="0"/>
141       </dvt:chartTickLabel>
142     </dvt:chartYAxis>
143     <dvt:chartValueFormat id="cvf1" scaling="none" type="y">
144       <af:convertNumber maxFractionDigits="0"/>
145     </dvt:chartValueFormat>
146     <f:facet name="seriesStamp">
147       <af:group id="g6">
148         <dvt:chartSeriesStyle id="css3" series="#{bindings.VwdiariosView1.hints.Llamadas.label}"
149           color="#3A7C93"/>
150       </af:group>
151     </f:facet>
152   </dvt:lineChart>
153 </af:gridCell>

```

FIGURA 4.33: Gráfico

³³ Creación de Tablas ADF

https://docs.oracle.com/cd/E21764_01/web.1111/b31974/web_tables_forms.htm#ADFFD726

³⁴ Uso de Gráficos con Oracle ADF

<https://docs.oracle.com/middleware/1213/adf/develop-faces/dvt-charts.htm#ADFUI13784>

Concluidas todas las tablas y gráficos de las 3 vistas seleccionar el 'general-task-flow-definition' y con 'drag and drop' se creará una 'Region'³⁵ en 'principal.jsf' (fig 4.34)

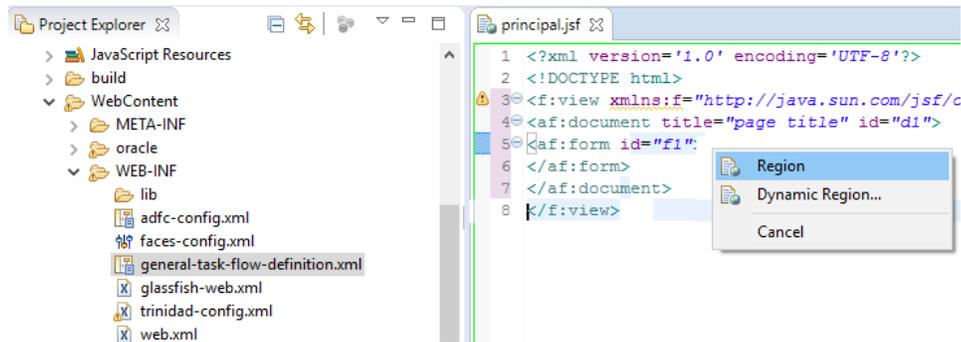


FIGURA 4.34: Region Dinámica

Ahora se programará el acceso con las librerías 'Apache Shiro'³⁶, click derecho en la carpeta con sufijo model y seleccionar 'Properties' (fig 4.35)

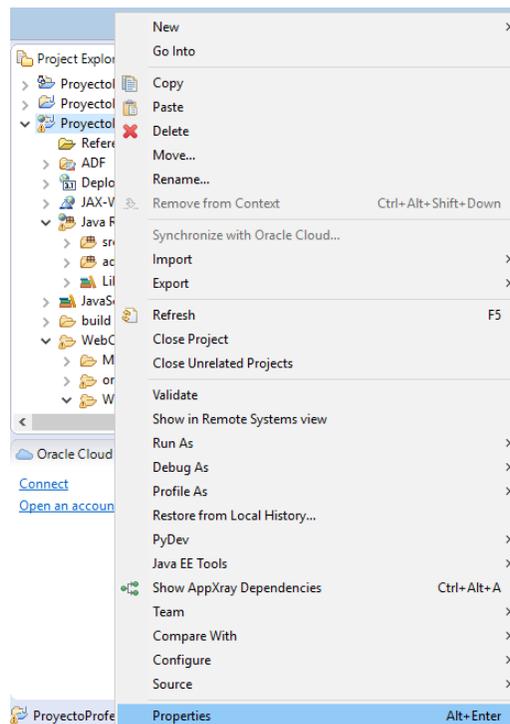


FIGURA 4.35: Propiedades

³⁵ Creación de Regiones ADF

https://docs.oracle.com/cd/E23943_01/web.1111/b31974/taskflows_regions.htm#ADFFD22565

³⁶ Librerías Apache Shiro

<http://shiro.apache.org/>

Seleccionamos 'Java Build Path' ->'Libraries' ->'Add External JARs...' ³⁷ (fig 4.36)

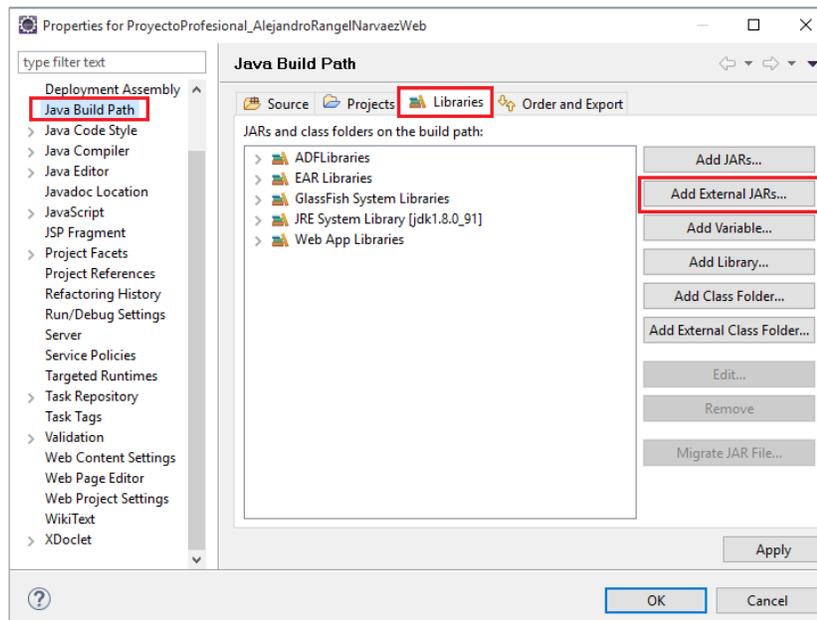


FIGURA 4.36: Añadir Jars externos

Ir donde se tengan descomprimidas las bibliotecas de Apache Shiro, seleccionar los jars repectivos y 'Abrir' (fig 4.37)

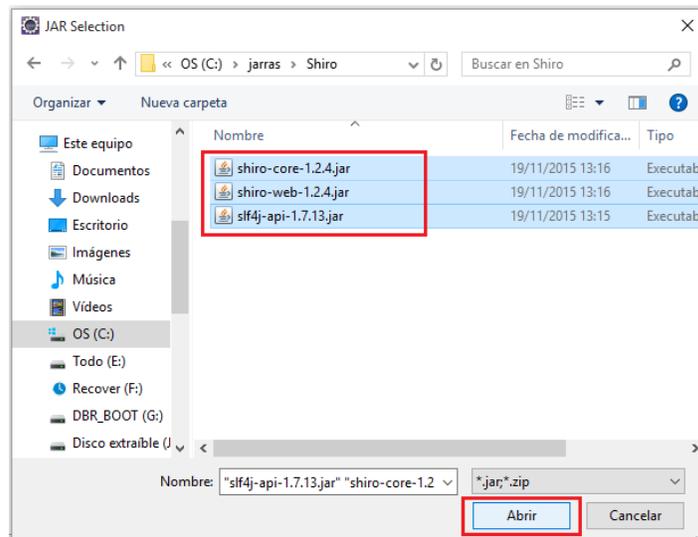


FIGURA 4.37: Apache Shiro

³⁷Tutorial añadir librerías externas a un proyecto Java
https://wiki.eclipse.org/FAQ_How_do_I_add_an_extra_library_to_my_project's_classpath%3F

Se creará una página llamada 'login.jsf' y se añadirán los 'Beans'³⁸ necesarios siguiendo las especificaciones de la biblioteca Apache Shiro³⁹ (fig 4.38)

```

25     </body>
26   </html>
27 </af:panelGroupLayout>
28 </af:panelGroupLayout>
29 </f:facet>
30 <f:facet name="center">
31   <af:decorativeBox id="db1">
32     <f:facet name="center">
33       <af:panelFormLayout id="pf11">
34         <af:inputText label="Usuario" id="username" value="#{LoginBean.userName}"
35           showRequired="true" required="true"/>
36         <af:inputText label="Contraseña" id="password" secret="true" value="#{LoginBean.password}"
37           required="true" showRequired="true"/>
38       <!--
39         <af:selectBooleanCheckbox text="Remember Me" label="Remember Me" id="rememberMe"
40           value="#{LoginBean.remember}"/>-->
41     </f:facet>
42     <af:panelGroupLayout id="pg11">
43       <af:link text="Acceso" id="log" partialSubmit="false"
44         action="#{LoginBean.login}" inlineStyle="color:red"/>
45     </af:panelGroupLayout>
46   </f:facet>
47 </af:panelFormLayout>
48 </f:facet>
49 <f:facet name="top">
50 </af:decorativeBox>
51 </f:facet>

```

FIGURA 4.38: Beans

Concluida la aplicación se publica en Glassfish Server⁴⁰ haciendo click derecho en 'principal.jsf' ->'Run on Server'

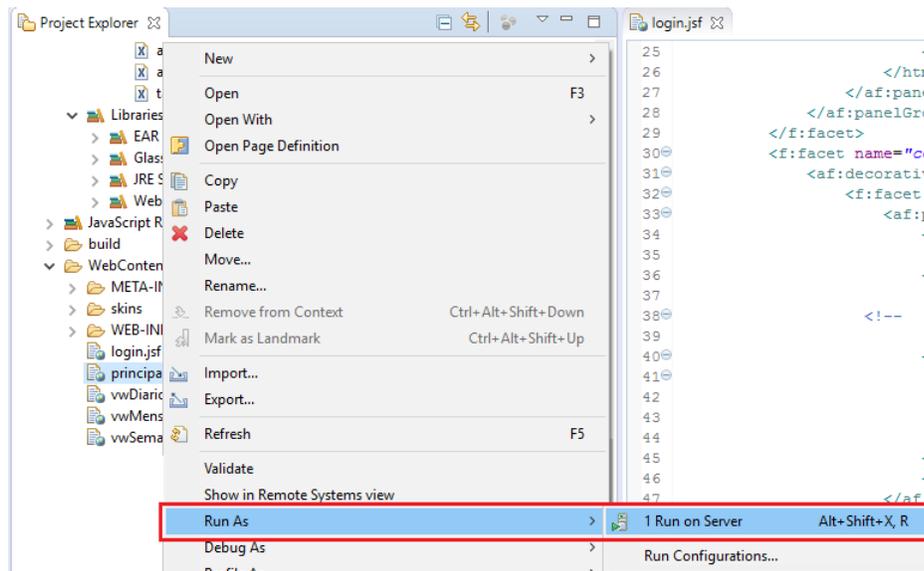


FIGURA 4.39: Ejecutar aplicación

³⁸Documentación Java Beans

<https://docs.oracle.com/javase/tutorial/javabeans/>

³⁹Documentación Apache Shiro

<http://shiro.apache.org/documentation.html>

⁴⁰Desplegando Aplicaciones Oracle ADF en Glassfish

https://blogs.oracle.com/shay/entry/deploying_oracle_adf_applications_to

Seleccionar 'Glassfish 4'⁴¹ que ya deberá estar configurado y ejecutándose y click en 'Finish' (fig 4.40)

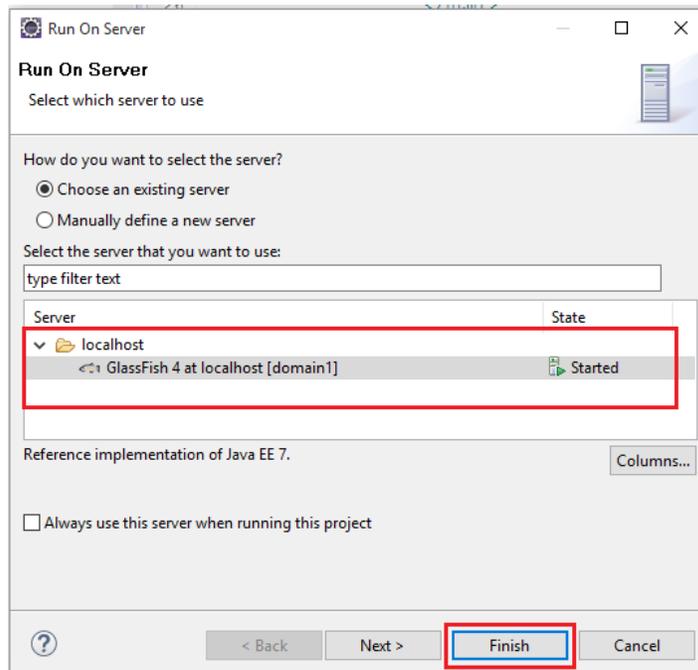


FIGURA 4.40: Seleccionar glassfish

Ir al administrador de Glassfish⁴² y en 'Applications' se tiene lista la aplicación, click en el proyecto (fig 4.41)

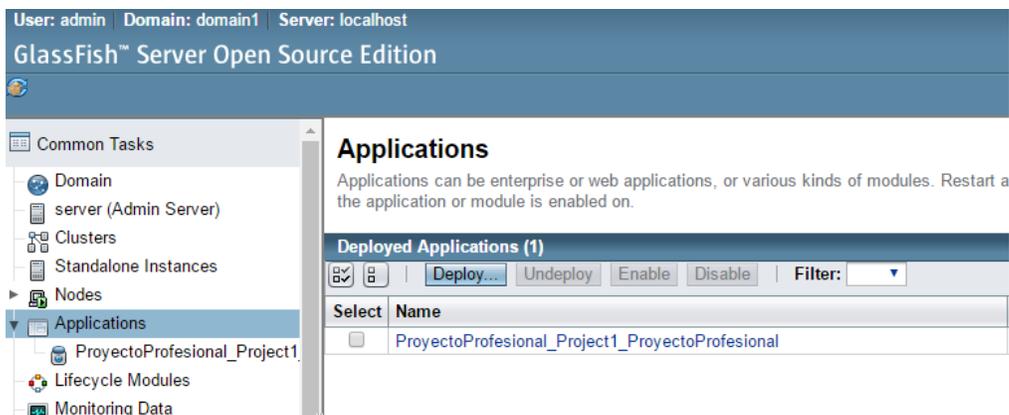


FIGURA 4.41: Aplicación publicada

⁴¹ Configuración de Glassfish Server para aplicaciones ADF

https://docs.oracle.com/middleware/1212/adf/ADFAG/ap_glassfish.htm#ADFAG20931

⁴² Guía de administración de Glassfish

<https://glassfish.java.net/docs/4.0/administration-guide.pdf>

Para probar la aplicación se hará click en 'Launch'⁴³ (fig 4.42)

Modules and Components (8)					
Module Name	Engines	Component Name	Type	Action	
ProyectoProfesional_VistaControlador_webapp.war	[web]	-----	-----	Launch	
ProyectoProfesional_VistaControlador_webapp.war		default	Servlet		
ProyectoProfesional_VistaControlador_webapp.war		BIGRAPHSERVLET	Servlet		
ProyectoProfesional_VistaControlador_webapp.war		Faces Servlet	Servlet		
ProyectoProfesional_VistaControlador_webapp.war		jsp	Servlet		
ProyectoProfesional_VistaControlador_webapp.war		resources	Servlet		
ProyectoProfesional_VistaControlador_webapp.war		BIGAUGESERVLET	Servlet		
ProyectoProfesional_VistaControlador_webapp.war		MapProxyServlet	Servlet		

FIGURA 4.42: Lanzar aplicación

Se muestran dos ligas y seleccionamos la que deseemos, en este caso se utilizará la primera liga (fig 4.43).

← → ↻ localhost:4848/common/applications/webApplicationLinks.jsf?appID=ProyectoProfesional_Project1_ProyectoProfesional&contextRc

Web Application Links

If the server or listener is not running, the link may not work. In this event, check the status of the server instance. After launching the web application, use the browser's Back button to return to this screen.

Application Name: ProyectoProfesional_Project1_ProyectoProfesional

Links:

- [server] <http://CyberFish:8080/ProyectoProfesional-AlejandroRangelNarvaez>
- [server] <https://CyberFish:8181/ProyectoProfesional-AlejandroRangelNarvaez>

FIGURA 4.43: Links

Se abre la aplicación mostrándose el login donde ingresaremos usuario y password que dará acceso al contenido y con esto se ha finalizado el desarrollo del dashboard (fig 4.44)

← → ↻ localhost:8080/ProyectoProfesional-AlejandroRangelNarvaez/

* Usuario

* Contraseña

Acceso

FIGURA 4.44: login

⁴³ Guía de Desarrollo de aplicaciones en Glassfish
<https://glassfish.java.net/docs/4.0/application-development-guide.pdf>

4.5. Resultados

Los resultados fueron los siguientes: El dashboard puede ser consultado en la siguiente liga

- <http://69.24.90.63:33331/ProyectoProfesional-AlejandroRangelNarvaez/faces/login.jsf>

- User: Demo.

- Pass: Demo.

Al validar usuario y contraseña lo primero que se muestra es la Vista Diaria. Del del lado superior izquierdo, en rojo, está el periodo de tiempo y los días transcurridos, del lado superior derecho están los controles para cambiar a las Vistas semanal y mensual, y sobre éstos se encuentra el botón para salir de la aplicación. En la parte de abajo tenemos la sección para descargar los datos que podremos acceder haciendo click en la barra gris (fig 4.45).



FIGURA 4.45: Vista Diaria

Revisando cada sección de la aplicación, en el primer cuadro del extremo superior izquierdo se presenta el resumen con las variables y las medidas definidas: Total, Media, Mediana, Mínimo, Máximo y Rango, como lo muestra la fig 4.46.

Variable	Total	Media	Mediana	Mínimo	Máximo	Rango
Llamadas	1,864,941	3,492.35	3,763	5	8,007	8,002
Minutos	18,588,992	34,829.75	37,506	40	79,310	79,270
Ingreso	6,305,11...	11,802.51	12,176.85	11,49	24,759.6	24,748.1
Costo	1,988,78...	3,727.79	3,996.02	3,37	8,256.32	8,252.95
Margen	4,316,32...	8,074.72	8,272	8,13	18,321.8	18,313.68
MargenP...	0,68	0,66	0,7	0,43	0,77	0,34

FIGURA 4.46: Resumen

A la derecha, en el siguiente cuadro, se tiene el gráfico de la serie temporal de las llamadas que muestra tendencias y comportamiento, junto con su histograma de frecuencias absoluto y relativo. Se puede ajustar el tamaño de los gráficos al tamaño del navegador haciendo click en la doble flecha de la esquina superior derecha (fig 4.47).

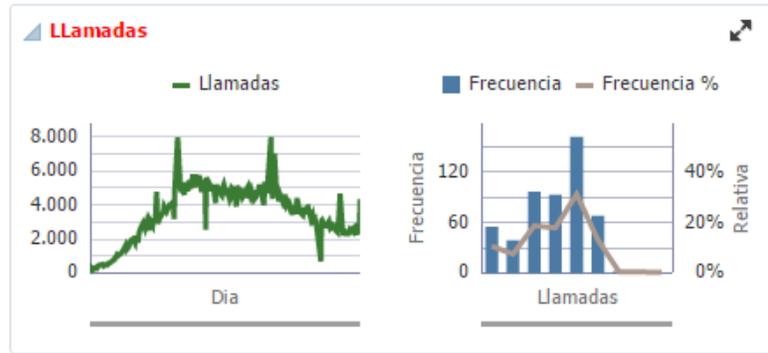


FIGURA 4.47: LLamadas

Continuando a la derecha se tiene la serie temporal de los minutos con sus histogramas de frecuencias absoluta y relativa (fig 4.48) .

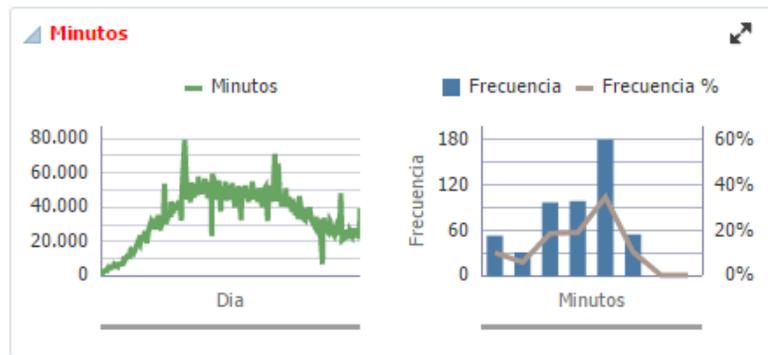


FIGURA 4.48: Minutos

El la siguiente fila en el primer panel está el ingreso (fig 4.49) en el mismo formato que las llamadas y minutos. Se puede hacer zoom a los gráficos posicionando el mouse sobre el gráfico y con la ayuda del scroll (rueda) para ver más a detalle.

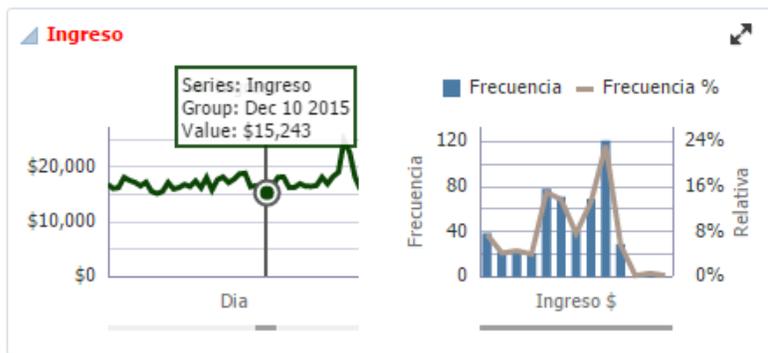


FIGURA 4.49: Ingreso

En seguida y a la derecha está el costo en la fig 4.50

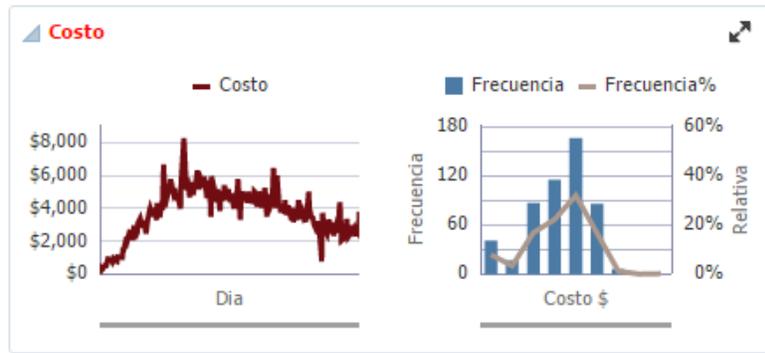


FIGURA 4.50: Costo

Finalmente se muestra el margen y margen porcentual. Las barras en verde son el margen en unidades monetarias y la línea roja representa el margen porcentual a lo largo del tiempo. Del mismo modo que todas las variables, se presenta el histograma de frecuencia absoluta y relativo (fig 4.51).

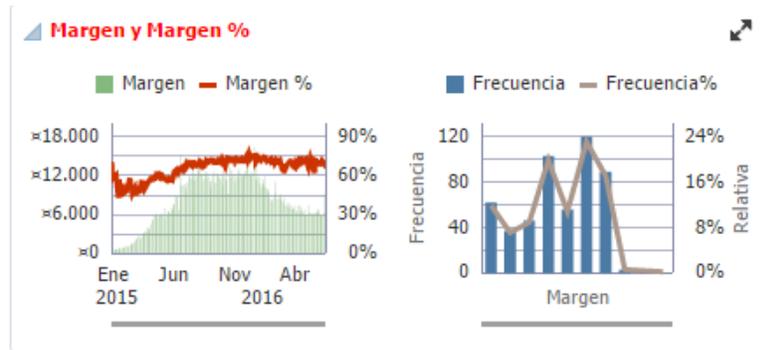


FIGURA 4.51: Margen

En la última fila de paneles están los resultados de las componentes principales y el cluster por K medias. El primer panel de la fig 4.52 muestra la matriz de correlaciones y se puede observar que existe una alta correlación en las variables por lo que se concluye que existe información redundante y las variables son candidatas a un análisis de Componentes Principales.

Matriz de Correlaciones						
	Llamadas	Minutos	Ingreso	Costo	Margen	
Llamadas	1	0,989	0,992	0,935	0,977	
Minutos	0,989	1	0,989	0,953	0,966	
Ingreso	0,992	0,989	1	0,926	0,991	
Costo	0,935	0,953	0,926	1	0,867	
Margen	0,977	0,966	0,991	0,867	1	

FIGURA 4.52: Matriz de correlaciones

En el siguiente panel, en la fig 4.53 están los autovalores y autovectores. La tabla izquierda muestra los autovalores e importancia de las componentes y del lado derecho los autovectores. En la proporción de la varianza y la varianza acumulada se tiene que las dos primeras componentes describen el 99% de la varianza por lo que serán suficientes las dos primeras componentes.

	Cp1	Cp2	Cp3	Cp4	Cp5		Cp1	Cp2	Cp3	Cp4	Cp5
Autovalores	4.835	0.145	0.011	0.009	0	Llamadas	-0.453	-0.091	0.755	-0.465	-0
Desviación ...	2.199	0.381	0.105	0.095	0	Minutos	-0.453	0.062	0.26	0.85	0
Proporción ...	0.967	0.029	0.002	0.002	0	Ingreso	-0.453	-0.2	-0.366	-0.115	-0.779
Proporción ...	0.967	0.996	0.998	1	1	Costo	-0.433	0.803	-0.287	-0.202	0.21
						Margen	-0.444	-0.55	-0.381	-0.08	0.59

FIGURA 4.53: Autovalores, Autovectores, Proporción de la Varianza

Por último en la fig 4.54 se muestra el gráfico de individuos, puesto que dos componentes son suficientes, las observaciones se pueden graficar en el plano \mathbb{R}^2 . Los puntos que se encuentran fuera de la nube central son los días atípicos que resaltan en los históricos. Posteriormente a los puntos se les aplicó una agrupación definiendo 2 medias que particionó la muestra en dos grupos.

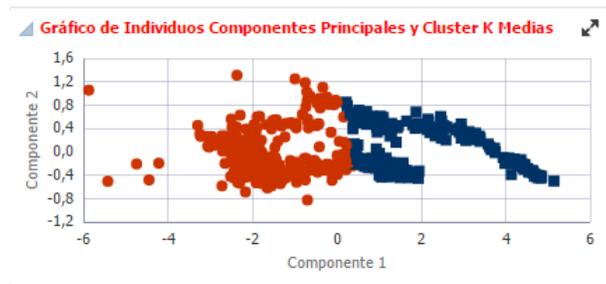


FIGURA 4.54: Gráfico Componentes Principales

Se tiene en esta parte la sección de los datos donde las variables se muestran en tablas y se pueden descargar en formato excel (fig 4.55).

Da	Llamadas	Minutos	Ingreso	Costo	Margen	Margen%
12/28/2014	5	40	\$11	\$3	\$8	70.2%
12/29/2014	196	1.885	\$665	\$284	\$381	57.24%
12/30/2014	191	1.943	\$746	\$262	\$484	64.91%
12/31/2014	302	2.717	\$1.091	\$397	\$694	63.64%
1/1/2015	235	2.680	\$1.051	\$412	\$639	60.78%
1/2/2015	253	2.483	\$887	\$304	\$582	65.87%
1/3/2015	280	1.915	\$802	\$323	\$479	58.25%
1/4/2015	207	2.211	\$843	\$346	\$497	58.95%
1/5/2015	287	2.973	\$1.008	\$393	\$615	61%
1/6/2015	303	2.843	\$908	\$372	\$536	59.05%
1/7/2015	283	3.092	\$1.120	\$408	\$640	56.7%
1/8/2015	214	2.578	\$880	\$348	\$532	60.49%
1/9/2015	327	3.540	\$1.280	\$549	\$730	57.06%
1/10/2015	273	3.466	\$1.112	\$485	\$627	56.4%
1/11/2015	286	3.240	\$1.131	\$542	\$589	52.11%
1/12/2015	354	4.113	\$1.356	\$559	\$798	58.8%
1/13/2015	368	5.074	\$1.572	\$678	\$894	56.86%
1/14/2015	438	5.320	\$1.614	\$923	\$691	42.82%
1/15/2015	421	4.435	\$1.363	\$643	\$720	32.85%
1/16/2015	351	3.935	\$1.318	\$565	\$753	57.12%
1/17/2015	351	4.370	\$1.344	\$635	\$708	52.72%
1/18/2015	417	5.165	\$1.496	\$782	\$715	47.77%
1/19/2015	402	4.664	\$1.443	\$754	\$689	47.77%
1/20/2015	446	5.245	\$1.559	\$712	\$846	54.3%
1/21/2015	504	5.397	\$1.756	\$883	\$873	49.7%
1/22/2015	455	4.927	\$1.531	\$770	\$761	49.71%
1/23/2015	475	5.357	\$1.647	\$765	\$882	65.84%
1/24/2015	396	5.167	\$1.518	\$707	\$812	53.45%
1/25/2015	467	5.381	\$1.528	\$759	\$768	50.3%
1/26/2015	491	6.746	\$2.023	\$948	\$1.075	53.13%
1/27/2015	453	5.646	\$1.687	\$907	\$780	46.25%
1/28/2015	460	5.424	\$1.688	\$853	\$835	49.47%
1/29/2015	503	5.681	\$1.727	\$843	\$883	51.16%
1/30/2015	588	6.345	\$1.789	\$888	\$930	51.71%
1/31/2015	481	6.414	\$1.813	\$891	\$843	51.41%
2/1/2015	484	6.042	\$1.821	\$915	\$907	49.79%

FIGURA 4.55: Datos

Haciendo click en la liga de 'Semanales' aparece la vista semanal en donde se encuentra el mismo análisis anterior pero ahora con las variables agrupadas por semanas (fig 4.56).



FIGURA 4.56: Vista Semanal

Finalmente la liga de 'Mensuales' llevará a la vista mensual en donde las variables son agrupadas por meses (fig 4.57).



FIGURA 4.57: Vista Mensual

Capítulo 5

Conclusiones

Sobre el uso y aplicación de los Dashboards podemos concluir que:

- Los Dashboards son una alternativa muy reciente en la forma en que se presentan las estadísticas para los administrativos de las compañías y son muy útiles para aumentar la rapidez y la entrega de la información.
- Ayudan a la compañía que busca la mejor manera de administrar la variada y numerosa información acerca de sus operaciones y funcionan perfectamente para distribuir la información entre sus colaboradores
- Son una herramienta que permite a los usuarios sumergirse en la información de la organización de manera ágil, rápida y entendible para tener mecanismos de análisis automatizados, puesto que cuando el colaborador ingrese a su horario laboral la información estará publicada y actualizada ya que la actualización se realiza durante la madrugada.
- Este tipo de herramientas son cruciales para un área como la que maneja el Actuario ya que ayuda a comunicar los resultados de las estrategias de la organización mediante el despliegue continuo de la información que puede ser consultada a cualquier hora y en cualquier lugar.

En cuanto a la participación de los actuarios en el análisis de información podemos mencionar:

- Cada día es más la cantidad de información que hay que controlar. Así el Actuario con estas herramientas, permite tener una alternativa de suministro de información, de manera que las personas encargadas de los distintos procesos operativos en las diferentes áreas, tengan un elemento de apoyo en sus respectivas labores.
- El Actuario debe tener un rol importante en una organización para la creación de las condiciones que permitan compartir la información con todos los colaboradores. El promover nuevas ideas, nuevas herramientas, nuevas metodologías a partir del cambiante mundo tecnológico le permite estar a la vanguardia de las nuevas expectativas que esperan los directivos de cualquier compañía.
- Los administrativos dentro de la organización son el centro del proceso y la necesidad de información que poseen es esencial y permanente. Por eso se debe de implementar estas herramientas que permitan agilizar procesos de suministro de información y para poder tomar decisiones lo más pronto posible y generar seguridad, credibilidad y confianza.

De manera personal concluyo:

- Que la realización de este trabajo me ha permitido demostrar de manera clara y sencilla la integración de los conocimientos escolares con la experiencia laboral y pienso que un profesionalista como el actuario, quien históricamente siempre se ha desempeñado en la realización y manejo de datos, se ha enriquecido con soluciones como lo son los Dashboards, que cubren el espectro de necesidades de una organización para la presentación de las estadísticas.

Apéndice A

Análisis de Componentes Principales

El análisis de componentes principales, es una técnica estadística que fue propuesta en el siglo XIX por Karl Pearson como parte del análisis de factores. Sus cálculos tan complejos retrasaron su desarrollo hasta la aparición de los computadores en la segunda mitad del siglo XX, por lo que resulta relativamente nuevo el florecimiento de los métodos basados en componentes principales y por este renacimiento tan reciente resulta poco utilizado por una gran cantidad de investigadores no especializados en estadística.

Lo que busca esta técnica es la reducción de la dimensión del número de variables. Es decir, cuando se cuenta con una fuente de datos con muchas variables, nuestro objetivo será reducirlas a un número menor posible de variables perdiendo la menor cantidad de información.

Estas nuevas variables son las que llamaremos Componentes Principales y serán una combinación lineal de las variables originales y además serán independientes entre sí. Este método tiene sentido si existen altas correlaciones entre las variables que queremos reducir, ya que esto nos indica que existe información redundante y por tanto, unas pocas componentes explicarán gran parte de la variabilidad total.

Se elegirán a las componentes de tal forma que la primera recoja la mayor proporción posible de la variabilidad original, la segunda recogerá la máxima variabilidad posible no recogida por la primera, y así sucesivamente. Del total de componentes se utilizarán aquéllos que contengan el porcentaje de variabilidad que se considere suficiente.

Los principios matemáticos son los siguientes, supongamos que se cuenta con n observaciones para cada una de las cuales se han medido m variables aleatorias de tipo numérico x_1, x_2, \dots, x_m las cuales están correlacionadas entre sí. Estas observaciones se pueden arreglar en una matriz X de dimensiones $n \times m$ del modo siguiente:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix}$$

Sea Σ la matriz de varianzas y covarianzas de X . Se tratará de calcular un nuevo conjunto de variables y_1, y_2, \dots, y_m no correlacionados entre sí cuyas varianzas vayan decreciendo progresivamente donde cada $y_{j=1,2,\dots,m}$ es una combinación lineal de las x_1, x_2, \dots, x_m , ie, $y_j = a_{j1}x_1 + a_{j2}x_2 + \cdots + a_{jm}x_m = a_j^1 x$ siendo $a_j^1 = (a_{j1}, a_{j2}, \dots, a_{jm})$ un vector de constantes y

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}$$

Se impondrá que $a_j^1 = (a_{j1}, a_{j2}, \dots, a_{jm})$ sea normalizado, ie, $a_j^1 a_j^1 = \sum_{k=1}^m a_{kj}^2 = 1$. El primer componente principal se calcula eligiendo a_1 de modo que y_1 tenga la mayor varianza posible sujeta a la restricción $a_1^1 a_1^1 = 1$, el segundo componente principal se calcula eligiendo a_2 de modo que la variable obtenida y_2 contenga la mayor cantidad de varianza posible no recogida por la primera y además que no este correlacionada con la variable y_1 , así se van eligiendo todas las siguientes y_3, y_4, \dots, y_m no correlacionadas entre sí, de manera que las variables aleatorias obtenidas van teniendo cada vez menor varianza.

A.1. Cálculo de las Componentes

Se desea elegir a_1 de modo que se maximice la varianza de y_1 sujeta a la restricción de que $a_1^T a_1 = 1$. Así se obtiene que $Var(y_1) = Var(a_1^T x) = a_1^T \Sigma a_1$. El método habitual para maximizar una función de varias variables sujeta a restricciones es el método de los multiplicadores de Lagrange¹. Así el problema consiste en maximizar la función $a_1^T \Sigma a_1$ sujeta a la restricción $a_1^T a_1 = 1$. Se observa precisamente que la incógnita es a_1 , es el vector desconocido que da la combinación óptima.

Se construye la función L

$$L(a_1) = a_1^T \Sigma a_1 - \lambda(a_1^T - 1)$$

Se busca el máximo derivando e igualando a cero

$$\begin{aligned} \frac{\partial L}{\partial a_1} &= 2\Sigma a_1 - 2\lambda I a_1 = 0 \rightarrow \\ (\Sigma - \lambda I) a_1 &= 0 \end{aligned}$$

Lo anterior es en realidad un sistema lineal de ecuaciones y por el teorema de Roché - Fronebius² para que el sistema tenga una solución distinta de cero la matriz $(\Sigma - \lambda I) = 0$ tiene que ser singular, es decir el determinante $|\Sigma - \lambda I| = 0$ y de este modo λ es un autovalor de Σ que es de orden m y si además es definida positiva tendrá m autovalores distintos $\lambda_1, \lambda_2, \dots, \lambda_m$ tales que $\lambda_1 > \lambda_2 > \dots > \lambda_m > 0$. Desarrollando la expresión anterior se tiene que

$$\begin{aligned} (\Sigma - \lambda I) a_1 &= 0 \\ \Sigma a_1 - \lambda I a_1 &= 0 \\ \Sigma a_1 &= \lambda I a_1 \end{aligned}$$

Lo que implica que

$$\begin{aligned} Var(y_1) = Var(a_1^T x) &= a_1^T \Sigma a_1 = a_1^T \lambda I a_1 = \\ \lambda a_1^T a_1 &= \lambda * 1 = \lambda \end{aligned}$$

Y luego para maximizar la varianza de y_1 se tiene que tomar el mayor autovalor λ_1 y el correspondiente autovector a_1 , así a_1 es un vector que da la combinación de las variables originales que tienen mayor varianza.

A.2. Cálculo de la m-ésima componente

El razonamiento anterior se puede extender de modo que al m-ésimo componente le corresponde el m-ésimo autovalor. Entonces todos los componentes y (en total m) se pueden expresar como el producto de una matriz formada por los autovectores, multiplicada por el vector x que contiene las variables originales x_1, x_2, \dots, x_m

$$y = Ax$$

donde

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}$$

¹El método de los multiplicadores de Lagrange, llamados así en honor a Joseph Louis Lagrange, es un procedimiento para encontrar los máximos y mínimos de funciones de múltiples variables sujetas a restricciones. Este método reduce el problema restringido con n variables a uno sin restricciones de $n + k$ variables, donde k es igual al número de restricciones, y cuyas ecuaciones pueden ser resueltas más fácilmente. Estas nuevas variables escalares desconocidas, una para cada restricción, son llamadas multiplicadores de Lagrange. El método dice que los puntos donde la función tiene un extremo condicionado con k restricciones, están entre los puntos estacionarios de una nueva función sin restricciones construida como una combinación lineal de la función y las funciones implicadas en las restricciones, cuyos coeficientes son los multiplicadores. La demostración usa derivadas parciales y la regla de la cadena para funciones de varias variables

²En álgebra lineal, el teorema de Rouché-Frobenius permite calcular el número de soluciones de un sistema de ecuaciones lineales en función del rango de la matriz de coeficientes y del rango de la matriz ampliada asociadas al sistema. El teorema establece que para que un sistema de ecuaciones lineales tenga solución es condición necesaria y suficiente que la matriz formada por los coeficientes junto con la ampliada por los términos independientes posean el mismo rango. Del mismo modo enuncia que el sistema de ecuaciones tendrá solución si su rango coincide con el número de incógnitas y no tendrá solución si posee un valor menor a tal número. La forma en que lo aplicamos es que establece que una matriz cuadrada es no singular si su rango es igual al número de variables y es singular si ese rango es menor.

Puesto que $Var(y_1) = \lambda_1, Var(y_2) = \lambda_2, \dots, Var(y_m) = \lambda_m$, la matriz de covarianzas de y será

$$\Delta = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_m \end{bmatrix}$$

ya que y_1, y_2, \dots, y_m se ha construido con variables no correlacionadas.

Se tiene que

$$\Delta = Var(Y) = A' Var(X) A = A' \Sigma A$$

o bien

$$\Sigma = A \Delta A'$$

ya que A es una matriz ortogonal (porque $a_i' a_i = 1$ para todas las columnas) por lo que $AA' = I$

A.3. Explicación de la varianza

Cada autovalor corresponde a la varianza del componente y_i que definido por medio del autovector a_i , es decir, $Var(y_i) = \lambda_i$. Sumando todos los autovalores se tendrá la varianza total de las componentes es decir

$$\sum_{i=1}^m Var(y_i) = \sum_{i=1}^m \lambda_i = traza(\Delta)$$

Por las propiedades de la traza tenemos que

$$traza(\Delta) = traza(A' \Sigma A) = traza(AA' \Sigma) = traza(\Sigma AA') = traza(\Sigma)$$

Con lo cual se concluye que

$$traza(\Delta) = traza(\Sigma) = \sum_{i=1}^m Var(x_i)$$

Es decir, la suma de las varianzas de las variables originales y la suma de las varianzas de las componentes son iguales. Este resultado permite concluir que el porcentaje de varianza total que recoge una componente principal está dado por:

$$\frac{\lambda_i}{\sum_{i=1}^m \lambda_i} = \frac{\lambda_i}{\sum_{i=1}^m Var(y_i)} = \frac{\lambda_i}{\sum_{i=1}^m Var(x_i)}$$

Apéndice B

Análisis por K Medias

En general el objetivo de los Análisis por Cluster es agrupar elementos en grupos homogéneos en función de 'alguna similitud' entre ellos.

K -Medias es un método de cluster en donde se particionan n observaciones en k grupos en el que cada observación pertenece al grupo más cercano a la media. Supongamos que se tiene una matriz de datos con n observaciones y m variables.

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix}$$

El objetivo es particionar la muestra de tamaño n en k grupos, eligiendo estos k grupos en forma arbitraria o mediante algún criterio. De esta forma cada observación x_{ij} se reescribe como x_{ijk} entendiendo que x_{ij} es el dato x_{jk} que está asignado al grupo k, $k = 1, 2, \dots, K$. Ahora se define el valor de \bar{x}_{ij} como el promedio de la j-ésima variable que pertenece al grupo k, esto es

$$\bar{x}_{ij} = \frac{1}{n_k} \sum_{i=1}^{n_k} x_{ijk} \quad (1)$$

Donde n_k es el tamaño del grupo k. Se calcula ahora la suma de las distancias cuadráticas de todas las observaciones del grupo k correspondiente a la j-ésima columna respecto a la media dentro del grupo.

$$\sum_{i=1}^{n_k} (x_{ijk} - \bar{x}_{jk})^2$$

Luego se suman estas distancias cuadráticas para cada columna y para cada grupo, esto es

$$\sum_{k=1}^K \sum_{j=1}^m \sum_{i=1}^{n_k} (x_{ijk} - \bar{x}_{jk})^2 \quad (2)$$

Esta suma se llamará suma de cuadrados dentro del grupo (SCDG) y sirve para medir la homogeneidad entre la partición de los K grupos elegidos. Así se utilizará aquella partición en que se obtenga el valor de

$$\min SCDG = \min \sum_{k=1}^K \sum_{j=1}^m \sum_{i=1}^{n_k} (x_{ijk} - \bar{x}_{jk})^2 \quad (3)$$

Se tiene que $\sum_{i=1}^{n_k} (x_{ijk} - \bar{x}_{jk})^2$ esencialmente es la varianza de la j-ésima columna dentro del grupo k, si se designa tal varianza como s_{jk}^2 se tiene que

$$s_{jk}^2 = \frac{1}{n_k} \sum_{i=1}^{n_k} (x_{ijk} - \bar{x}_{jk})^2 \quad (4)$$

Así sustituyendo 4 en 2 se tiene que

$$\sum_{k=1}^K \sum_{j=1}^m n_k s_{jk}^2$$

De tal forma que minimizar 2 es equivalente a minimizar:

$$\min SCDG = \min \sum_{k=1}^K \sum_{j=1}^m n_k s_{jk}^2$$

Se observa que la construcción de todos los valores de SCDG se realiza para cada una de las particiones posibles de K grupos, lo cual si teóricamente es posible significa un esfuerzo de cómputo formidable. En efecto, si se toma $n = 130$ variables, $K = 3$ y se considera que el primer grupo tenga 30 variables ($n_1 = 30$) que el segundo grupo tenga 30 variables ($n_2 = 30$) y que el tercer grupo tenga las 70 variables restantes ($n_3 = 70$) para esta particular condición hay

$$\frac{130!}{30!30!70!} = 7673048982467520126736536463822125374298479568669964800$$

particiones posibles. Se tiene que buscar un camino más corto y eficiente. Supongamos que se tiene particionada la muestra en K grupos, para un grupo $k \in K$ se calcula el vector de medias

$$\bar{x}_k = \begin{bmatrix} \bar{x}_{1k} \\ \vdots \\ \bar{x}_{pk} \end{bmatrix}$$

Luego se mide las distancias cuadráticas entre los puntos de cada grupo con su media, esto es para un particular $k \in K$

$$\min \sum_{i=1}^{n_k} (x_{ijk} - \bar{x}_{jk})^t (x_{ijk} - \bar{x}_{jk}) = \min \sum_{k=1}^K \sum_{i=1}^{n_k} d^2(i, k) \quad (5)$$

Si se entiende que $d^2(i, k)$ es la distancia cuadrática entre el elemento i del grupo k y la media de dicho grupo. Es fácil probar que los criterios establecidos en 3 y 5 son equivalentes. Además por ser $d^2(i, k)$ un escalar trivialmente se tiene que

$$\min \text{traza} [d^2(i, k)] = \min \text{traza} \sum_{k=1}^K \sum_{i=1}^{n_k} (x_{ijk} - \bar{x}_{jk})(x_{ijk} - \bar{x}_{jk})^t$$

y llamando W a la matriz de la suma de cuadrados dentro de los grupos, esto es

$$W = \sum_{i=1}^{n_k} (x_{ijk} - \bar{x}_{jk})(x_{ijk} - \bar{x}_{jk})^t$$

Se tiene que

$$\min \text{traza}(W) = \min SCDG \quad (6)$$

El criterio 6 se conoce como el criterio de la traza y se aplica del siguiente modo:

1. Partir de una asignación inicial, que eventualmente puede ser arbitraria.
2. Comprobar si moviendo algún elemento se reduce la traza(W).
3. Si efectivamente se reduce la traza(W), volver a calcular todas las medias de los dos grupos afectados por el cambio. Luego volver al paso (2). Si no es posible reducir la traza(W) terminar el proceso.

Apéndice C

Conectar R a una BD

En este apéndice se muestra paso a paso como conectar R statistics a una BD para extraer información que sirva para los cálculos. Se usa el protocolo ODBC¹ sobre el sistema operativo Windows, la configuración de ODBC en Linux es completamente diferente. Supondremos que se tienen los drivers de ODBC² instalados y configurados, que se tiene acceso a una BD remota, que se tienen conocimientos de R statistics, R studio y que ambos se encuentran instalados y configurados. Se hará el ejemplo con un DBMS Microsoft SQL Sever versión 2014.

- 1) En Windows ir a 'Inicio'-'>'Panel de Control' ->'Herramientas Administrativas'.



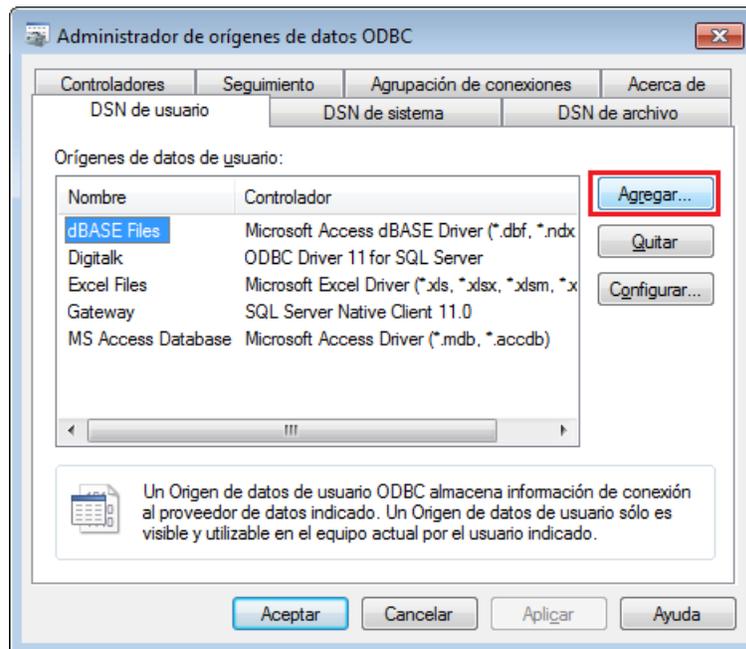
- 2) En Herramientas Administrativas doble click en 'Origenes de Datos ODBC (64 Bits)'



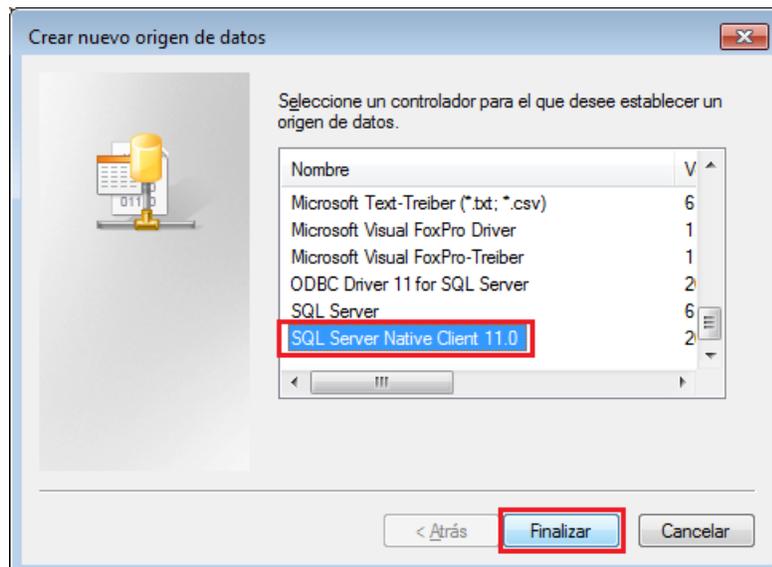
¹ODBC: resumen de conectividad abierta de bases de datos Microsoft Corporation
<https://support.microsoft.com/es-es/kb/110093>

²Driver ODBC para Microsoft SQL Server
<https://www.microsoft.com/en-US/download/details.aspx?id=36434>
 MySQL Server
<https://dev.mysql.com/downloads/connectorodbc/>
 Oracle
<http://www.oracle.com/technetwork/database/windows/index-098976.html>

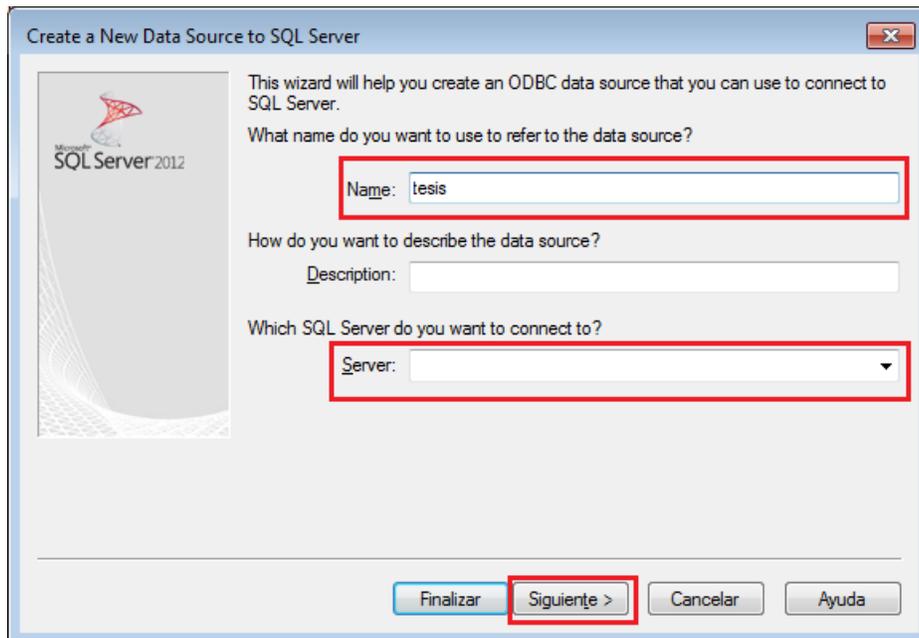
3) En el administrador de orígenes de datos ODBC click en 'Agregar'



4) Seleccionar el driver que en este caso será 'SQL Server Native Client 11.0' y click en 'Finalizar'



5) Se abre el Wizard para crear la conexión a la base de datos, en 'Name' se utilizará el nombre deseado y le llamaremos 'tesis', este nombre es el que posteriormente se ocupará para la conexión con R, se puede poner una breve descripción o dejar en blanco y se pondrá la IP del servidor de BD y click en 'Siguiente'.



Microsoft SQL Server 2012

This wizard will help you create an ODBC data source that you can use to connect to SQL Server.

What name do you want to use to refer to the data source?

Name: tesis

How do you want to describe the data source?

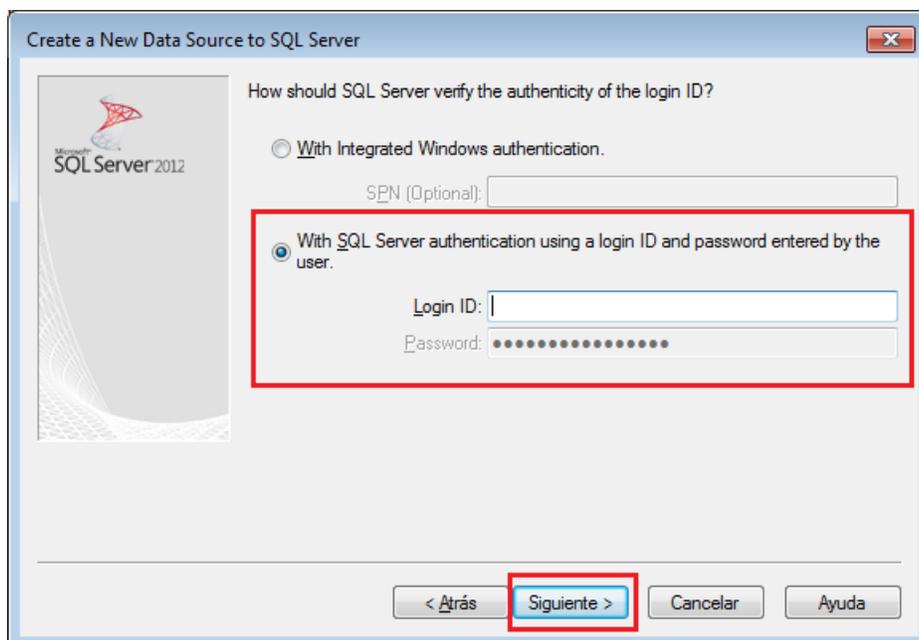
Description:

Which SQL Server do you want to connect to?

Server:

Finalizar Siguiente > Cancelar Ayuda

6) En la siguiente pestaña seleccionar 'With SQL authentication...' e introducir el nombre de usuario y password proporcionado por el administrador, click en 'Siguiente'



Microsoft SQL Server 2012

How should SQL Server verify the authenticity of the login ID?

With Integrated Windows authentication.

SPN (Optional):

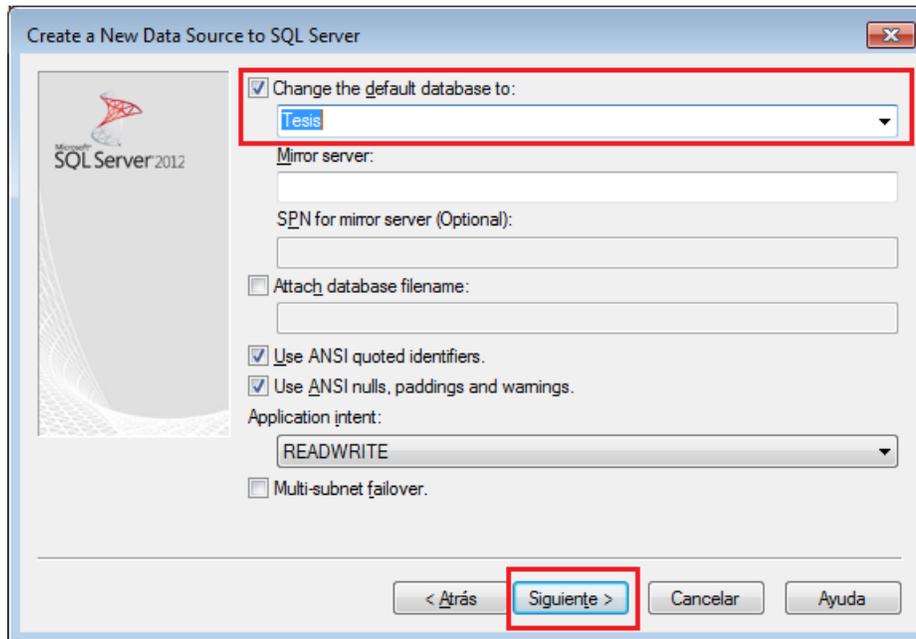
With SQL Server authentication using a login ID and password entered by the user.

Login ID:

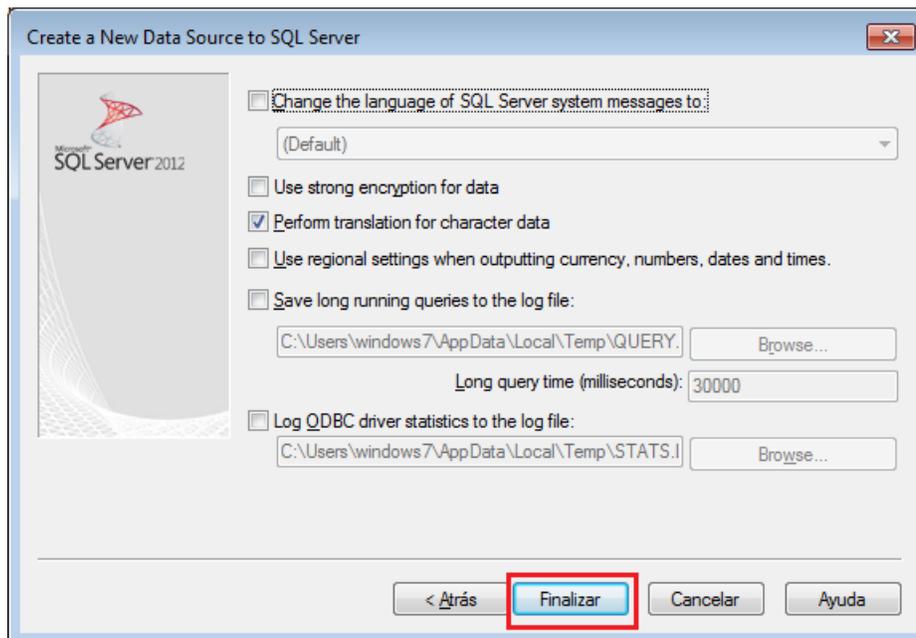
Password: ●●●●●●●●●●●●

< Atrás Siguiente > Cancelar Ayuda

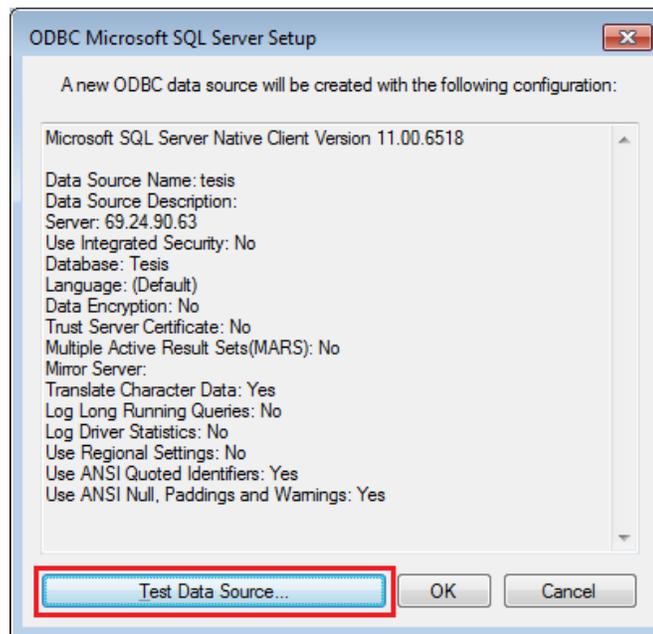
7) Seleccionar la BD deseada, click en siguiente.



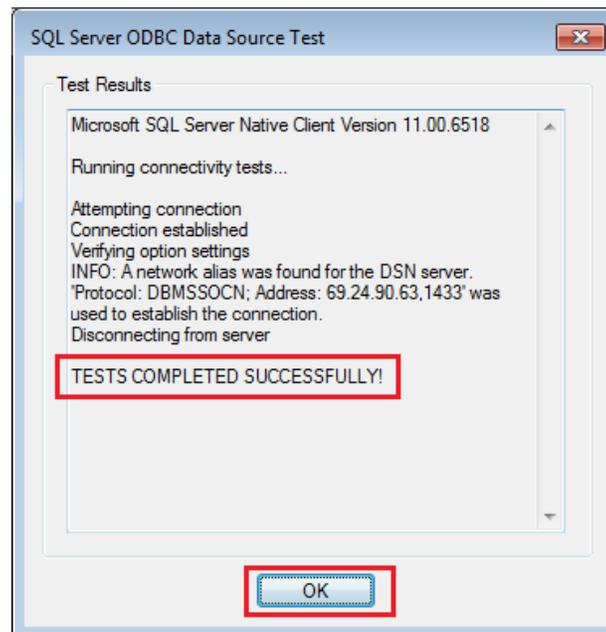
8) En esta vista dejar todo como aparece de manera predeterminada y click en 'Finalizar'.



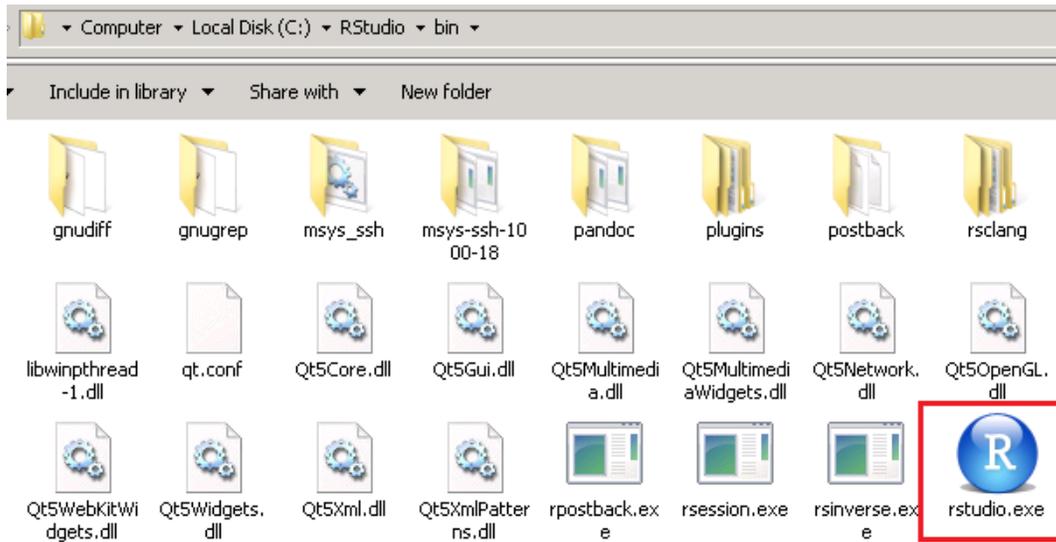
9) Click en 'Test Data Source...'



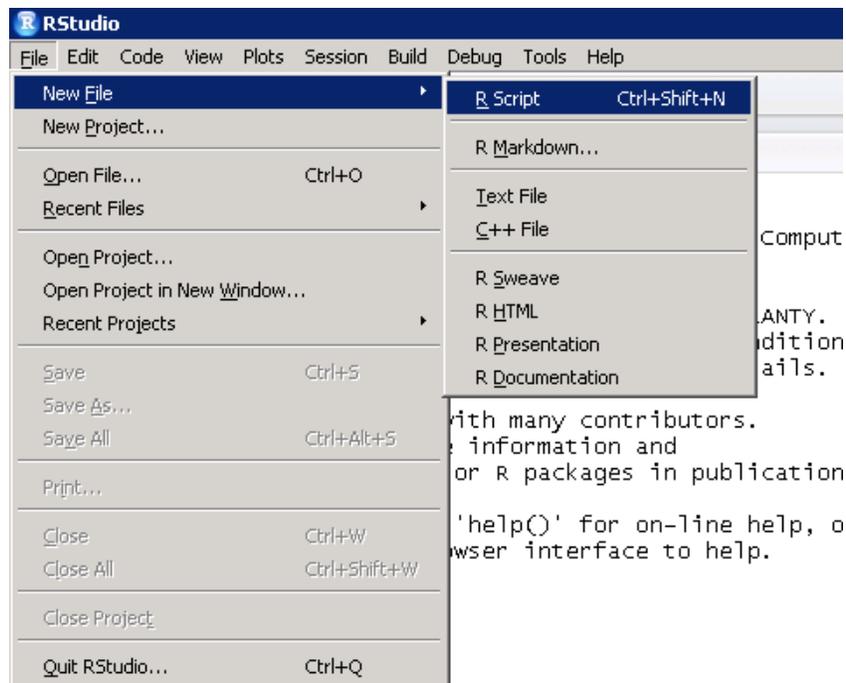
10) Deberá aparecer 'Test Completed Successfully' si se obtiene un mensaje de error comenzar de nuevo. Click en 'OK'.



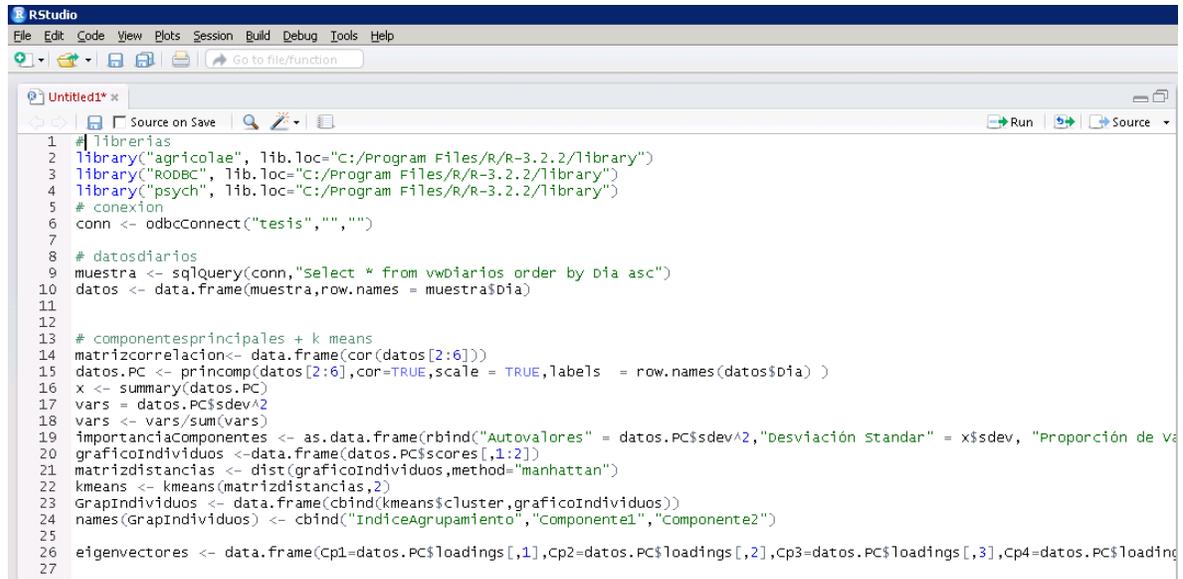
11) Ir donde se encuentre instalado R studio y doble click en rstudio.exe



12) En R studio, 'File' ->'New' ->'R Script'



13) Se define el script R que se desea ejecutar, la conexión a la BD para extraer los datos y procesarlos se hará con el protocolo ODBC con ayuda de la biblioteca RODBC. El comando 'odbc-Connect' invoca el nombre de la conexión ODBC configurada previamente y se pondrá el nombre de usuario y contraseña de la BD. Se ejecutarán los comandos de componentes principales y k medias tomando la muestra de la BD con el comando 'sqlQuery' que hace una consulta y genera el data frame usado en los cálculos.



```

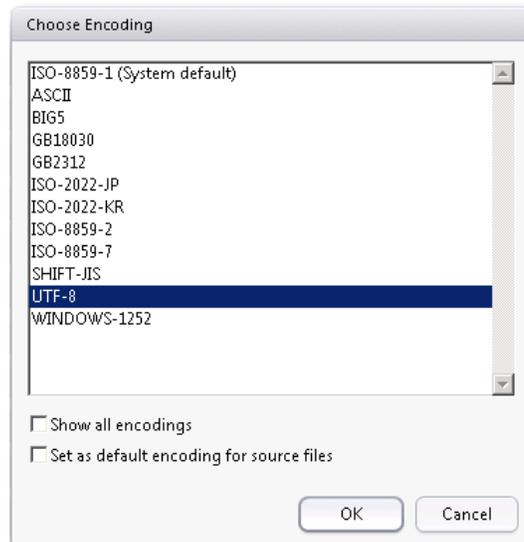
1 # librerias
2 library("agricolae", lib.loc="C:/Program Files/R/R-3.2.2/library")
3 library("RODBC", lib.loc="C:/Program Files/R/R-3.2.2/library")
4 library("psych", lib.loc="C:/Program Files/R/R-3.2.2/library")
5 # conexión
6 conn <- odbcConnect("tesis","","")
7
8 # datosdiarios
9 muestra <- sqlQuery(conn,"select * from vwDiarios order by Dia asc")
10 datos <- data.frame(muestra,row.names = muestra$Dia)
11
12
13 # componentesprincipales + k means
14 matrizcorrelacion<- data.frame(cor(datos[2:6]))
15 datos.PC <- princomp(datos[2:6],cor=TRUE,scale = TRUE,labels = row.names(datos$Dia) )
16 x <- summary(datos.PC)
17 vars = datos.PC$sdev^2
18 vars <- vars/sum(vars)
19 importanciaComponentes <- as.data.frame(rbind("Autovalores" = datos.PC$sdev^2,"Desviación Standar" = x$sdev, "Proporción de Varianza" = vars))
20 graficoIndividuos <-data.frame(datos.PC$scores[,1:2])
21 matrizdistancias <- dist(graficoIndividuos,method="manhattan")
22 kmeans <- kmeans(matrizdistancias,2)
23 GrapIndividuos <- data.frame(cbind(kmeans$cluster,graficoIndividuos))
24 names(GrapIndividuos) <- cbind("IndiceAgrupamiento","Componente1","Componente2")
25
26
27 eigenvectores <- data.frame(Cp1=datos.PC$loadings[,1],Cp2=datos.PC$loadings[,2],Cp3=datos.PC$loadings[,3],Cp4=datos.PC$loadings[,4])

```

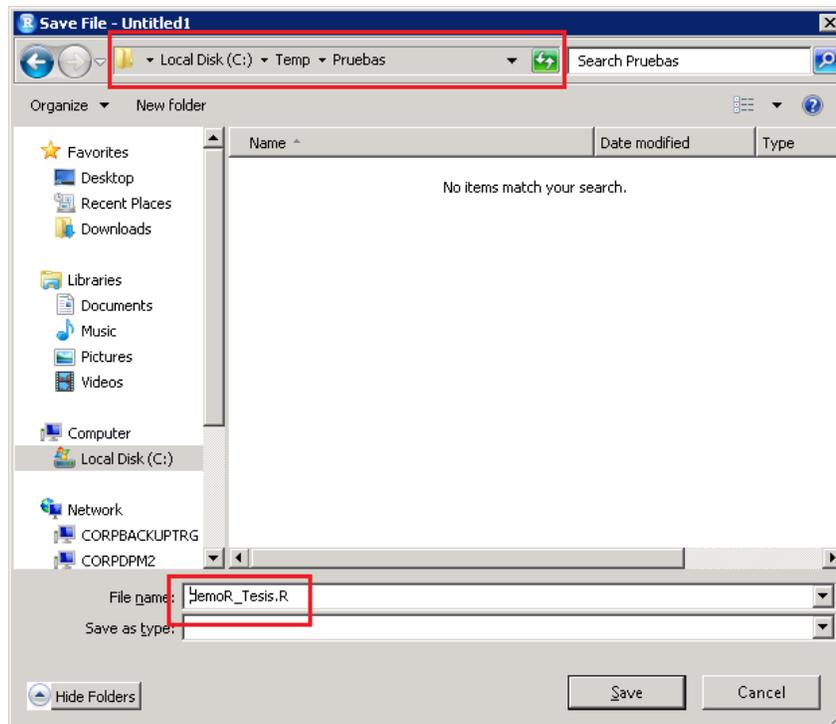
14) Una vez terminado el script R en menu seleccionar 'Save as...'



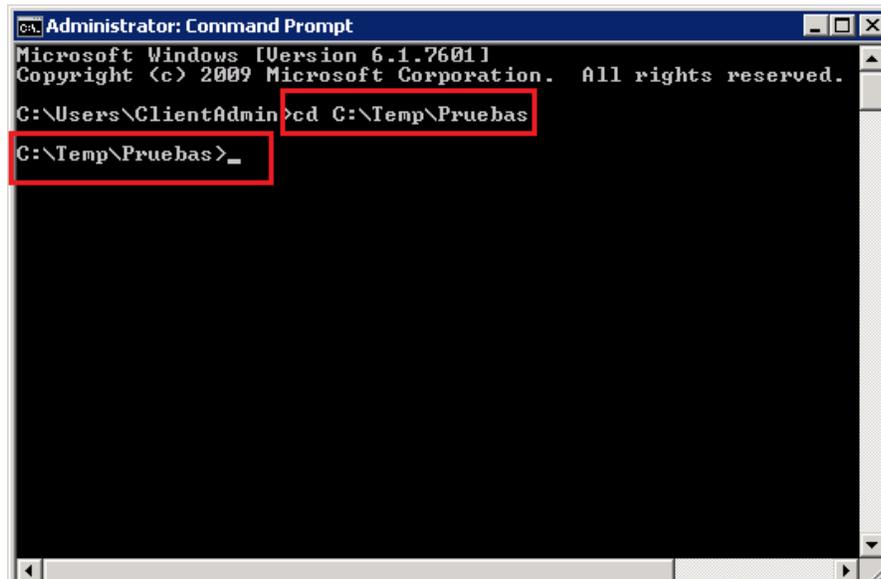
15) En 'Choose Encoding' seleccionar 'UTF8'.



16) Finalmente guardar el archivo con extensión .R, en este caso se usará 'demoR_Tesis.R' y se guardará en el directorio 'C:\Temp\Pruebas'

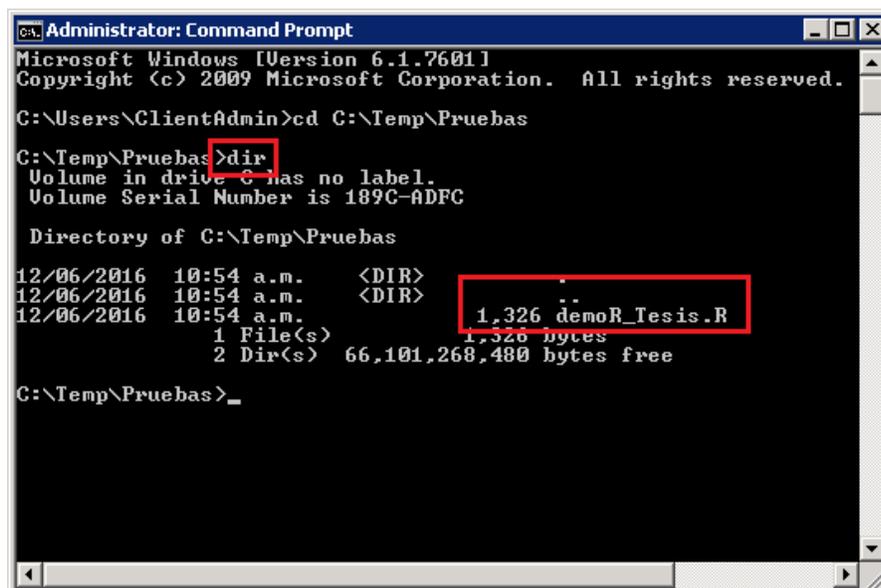


17) Guardado el archivo, en Windows ir a 'Inicio', buscar 'Símbolo del sistema' y ejecutar como administrador. Con el comando 'cd C:\Temp\Pruebas' cambiaremos al directorio donde se ha guardado el archivo R.



```
Administrator: Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\ClientAdmin>cd C:\Temp\Pruebas
C:\Temp\Pruebas>
```

18) Con el comando 'dir' se comprobará que allí se encuentra el archivo R creado previamente.



```
Administrator: Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\ClientAdmin>cd C:\Temp\Pruebas
C:\Temp\Pruebas>dir
Volume in drive C has no label.
Volume Serial Number is 189C-ADFC

Directory of C:\Temp\Pruebas

12/06/2016 10:54 a.m. <DIR>
12/06/2016 10:54 a.m. <DIR>
12/06/2016 10:54 a.m. 1,326_demoR_Tesis.R
                    1 File(s) 1,326 bytes
                    2 Dir(s) 66,101,268,480 bytes free

C:\Temp\Pruebas>
```

19) Para ejecutar el archivo utilizar el comando 'C:\Program Files\R\R-3.2.2\bin\R.exe CMD BATCH demoR_Tesis.R'.

```

Administrator: Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\ClientAdmin>cd C:\Temp\Pruebas

C:\Temp\Pruebas>dir
Volume in drive C has no label.
Volume Serial Number is 189C-ADFC

Directory of C:\Temp\Pruebas

12/06/2016  10:54 a.m.    <DIR>          .
12/06/2016  10:54 a.m.    <DIR>          ..
12/06/2016  10:54 a.m.                1,326 demoR_Tesis.R
                1 File(s)          1,326 bytes
                2 Dir(s)    66,101,268,480 bytes free

C:\Temp\Pruebas>C:\Program Files\R\R-3.2.2\bin\R.exe CMD BATCH demoR_Tesis.R
  
```

20) Concluido el proceso, con el comando 'dir' se comprueba que se han creado dos archivos uno llamado .RData que se puede abrir con R studio y demoR_Tesis.Rout que se puede abrir con cualquier block de notas.

```

Administrator: Command Prompt

C:\Temp\Pruebas>dir
Volume in drive C has no label.
Volume Serial Number is 189C-ADFC

Directory of C:\Temp\Pruebas

12/06/2016  10:54 a.m.    <DIR>          .
12/06/2016  10:54 a.m.    <DIR>          ..
12/06/2016  10:54 a.m.                1,326 demoR_Tesis.R
                1 File(s)          1,326 bytes
                2 Dir(s)    66,100,957,184 bytes free

C:\Temp\Pruebas>C:\PROGRA~1\R\R-3.2.2\bin\R.exe CMD BATCH demoR_Tesis.R

C:\Temp\Pruebas>dir
Volume in drive C has no label.
Volume Serial Number is 189C-ADFC

Directory of C:\Temp\Pruebas

12/06/2016  11:10 a.m.    <DIR>          .
12/06/2016  11:10 a.m.    <DIR>          ..
12/06/2016  10:54 a.m.                1,326 demoR_Tesis.R
12/06/2016  11:10 a.m.                1,471 demoR_Tesis.Rout
                2 File(s)          2,797 bytes
                2 Dir(s)    66,100,776,960 bytes free

C:\Temp\Pruebas>_
  
```

21) Se abre el archivo demoR_Tesis.Rout con un block de notas y se comprueba que allí están los resultados. Para insertar los datos en una BD y que se puedan reutilizar en algun otra aplicación o reporte, se aplica el comado 'sqlSave' que es parte de la librería RODBC.

```
Rutina.Rout
1
2 R version 3.2.2 (2015-08-14) -- "Fire Safety"
3 Copyright (C) 2015 The R Foundation for Statistical Computing
4 Platform: x86_64-w64-mingw32/x64 (64-bit)
5
6 R es un software libre y viene sin GARANTIA ALGUNA.
7 Usted puede redistribuirlo bajo ciertas circunstancias.
8 Escriba 'license()' o 'licence()' para detalles de distribucion.
9
10 R es un proyecto colaborativo con muchos contribuyentes.
11 Escriba 'contributors()' para obtener más información y
12 'citation()' para saber cómo citar R o paquetes de R en publicaciones.
13
14 Escriba 'demo()' para demostraciones, 'help()' para el sistema on-line de ayuda,
15 o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.
16 Escriba 'q()' para salir de R.
17
18 [Previously saved workspace restored]
19
20 > # librerias
21 > library("agricolae", lib.loc="C:/Program Files/R/R-3.2.2/library")
22 > library("RODBC", lib.loc="C:/Program Files/R/R-3.2.2/library")
23 > library("psych", lib.loc="C:/Program Files/R/R-3.2.2/library")
24 > # conexion
25 > conn <- odbcConnect("tesis", "sa", "n3wt3mp0r41@2021")
26 > # datosdiarios
```


Apéndice D

Script Completo R

Se muestra en este apéndice el script completo utilizado para calcular los resultados del Dashboard. Este script está basado en el ODBC explicado en el apéndice D. Se ejecuta automáticamente incorporado a una 'Tarea programada de windows'.

```
# Carga de librerías
library('agricolae', lib.loc='C:/Program Files/R/R-3.2.2/library')
library('RODBC', lib.loc='C:/Program Files/R/R-3.2.2/library')
library('psych', lib.loc='C:/Program Files/R/R-3.2.2/library')

# Conexión
conn <- odbcConnect('tesis','usuario','Password')

# Datos muestrales y descriptivos diarios
muestra <- sqlQuery(conn, 'Select * from vwDiarios order by Dia asc')
datos <- data.frame(muestra,row.names = muestra$Dia)
Descriptivos <- describe(datos[2:7])
sqlDrop(conn, 'Descriptivos')
sqlSave(conn, Descriptivos)

# Datos muestrales y descriptivos semanales
muestrasemanal <- sqlQuery(conn,'Select * from vwSemanales order by idsemana asc')
datossemanales <- data.frame(muestrasemanal, row.names=muestrasemana$SEMANA)
DescriptivosSemanales <- describe(datossemanales[3:8])
sqlDrop(conn, 'DescriptivosSemanales')
sqlSave(conn, DescriptivosSemanales)

# Datos muestrales y descriptivos mensuales
muestramensual <- sqlQuery(conn, 'Select * from vwMensuales order by idmes asc')
datosmensuales <- data.frame(muestramensual, row.names = muestramensual$MES)
DescriptivosMensuales <- describe(datosmensuales[3:8])
sqlDrop(conn, 'DescriptivosMensuales')
sqlSave(conn, DescriptivosMensuales)

#frecuencia llamadas semanales
temporalLlamadasSemanales <- hist(datossemanales$Llamadas, plot=FALSE)
freqLlamadasSemanales <- table.freq(temporalLlamadasSemanales)
sqlDrop(conn, 'freqLlamadasSemanales')
sqlSave(conn, freqLlamadasSemanales)

#frecuencia llamadas mensuales
temporalLlamadasMensuales <- hist(datosmensuales$Llamadas, plot=FALSE)
freqLlamadasMensuales <- table.freq(temporalLlamadasMensuales)
sqlDrop(conn, 'freqLlamadasMensuales')
sqlSave(conn, freqLlamadasMensuales)

# frecuencia minutos diarios
temporalMinutos <- hist(datos$Minutos, plot=FALSE)
freqMinutos <- table.freq(temporalMinutos)
```

```
sqlDrop(conn, 'freqMinutos')
sqlSave(conn, freqMinutos)

# frecuencia minutos semanales
temporalMinutosSemanales <- hist(datossemanales$Minutos, plot=FALSE)
freqMinutosSemanales <- table.freq(temporalMinutosSemanales)
sqlDrop(conn, 'freqMinutosSemanales')
sqlSave(conn, freqMinutosSemanales)

# frecuencia minutos mensuales
temporalMinutosMensuales <- hist(datosmensuales$Minutos, plot=FALSE)
freqMinutosMensuales <- table.freq(temporalMinutosMensuales)
sqlDrop(conn, 'freqMinutosMensuales')
sqlSave(conn, freqMinutosMensuales)

# frecuencia ingreso diario
temporalIngreso <- hist(datos$Ingreso, plot=FALSE)
freqIngreso <- table.freq(temporalIngreso)
sqlDrop(conn, 'freqIngreso')
sqlSave(conn, freqIngreso)

# frecuencia ingreso semanal
temporalIngresoSemanal <- hist(datossemanales$Ingreso, plot=FALSE)
freqIngresoSemanal <- table.freq(temporalIngresoSemanal)
sqlDrop(conn, 'freqIngresoSemanal')
sqlSave(conn, freqIngresoSemanal)

# frecuencia ingreso mensual
temporalIngresoMensual <- hist(datosmensuales$Ingreso, plot=FALSE)
freqIngresoMensual <- table.freq(temporalIngresoMensual)
sqlDrop(conn, 'freqIngresoMensual')
sqlSave(conn, freqIngresoMensual)

# frecuencia costo diario
temporalCosto <- hist(datos$Costo, plot =FALSE)
freqCosto <- table.freq(temporalCosto)
sqlDrop(conn, 'freqCosto')
sqlSave(conn, freqCosto)

# frecuencia costo semanal
temporalCostoSemanal <- hist(datossemanales$Costo, plot =FALSE)
freqCostoSemanal <- table.freq(temporalCostoSemanal)
sqlDrop(conn, 'freqCostoSemanal')
sqlSave(conn, freqCostoSemanal)

# frecuencia costo mensual
temporalCostoMensual <- hist(datosmensuales$Costo, plot =FALSE)
freqCostoMensual <- table.freq(temporalCostoMensual)
sqlDrop(conn, 'freqCostoMensual')
sqlSave(conn, freqCostoMensual)

# frecuencia margen diario
temporalMargen <- hist(datos$Margen, plot=FALSE)
freqMargen <- table.freq(temporalMargen)
sqlDrop(conn, 'freqMargen')
sqlSave(conn, freqMargen)

# frecuencia margen semanal
temporalMargenSemanal <- hist(datossemanales$Margen, plot=FALSE)
```

```

freqMargenSemanal <- table.freq(temporalMargenSemanal)
sqlDrop(conn, 'freqMargenSemanal')
sqlSave(conn, freqMargenSemanal)

# frecuencia margen mensual
temporalMargenMensual <- hist(datosmensuales$Margen, plot=FALSE)
freqMargenMensual <- table.freq(temporalMargenMensual)
sqlDrop(conn, 'freqMargenMensual')
sqlSave(conn, freqMargenMensual)

# componentesprincipales + k means diario
matrizcorrelacion<- data.frame(cor(datos[2:6]))
datos.PC <- princomp(datos[2:6], cor=TRUE,scale = TRUE,labels = row.names(datos$Dia) )
x <- summary(datos.PC)
vars = datos.PC$sdev2
vars <- vars/sum(vars) importanciaComponentes <- as.data.frame(rbind('Autovalores' = datos.PC$sdev2, 'Desviación
Standar'
= x$sdev, 'Proporción de Varianza' = vars, 'Proporción Acumulada' = cumsum(vars)))
graficoIndividuos <- data.frame(datos.PC$scores[,1:2])
matrizdistancias <- dist(graficoIndividuos, method='manhattan')
kmeans <- kmeans(matrizdistancias,4)
GrapIndividuos <- data.frame(cbind(kmeans$cluster,graficoIndividuos))
names(GrapIndividuos) <- cbind('IndiceAgrupamiento','Componente1','Componente2')
eigenvectores <- data.frame(Cp1 = datos.PC$loadings[,1], Cp2 = datos.PC$loadings[,2], Cp3=datos.PC$loadings[,3],
Cp4=datos.PC$loadings[,4], Cp5=datos.PC$loadings[,5])
sqlDrop(conn, 'GrapIndividuos')
sqlSave(conn, GrapIndividuos)
sqlDrop(conn, 'importanciaComponentes')
sqlSave(conn, importanciaComponentes)
sqlDrop(conn, 'eigenvectores')
sqlSave(conn, eigenvectores)
sqlDrop(conn, 'matrizcorrelacion')
sqlSave(conn, matrizcorrelacion)

# componentesprincipales + k means semanal
matrizcorrelacionSemanal<- data.frame(cor(datossemanales[3:7]))
datossemanales.PC <- princomp(datossemanales[3:7], cor = TRUE, scale = TRUE, labels = row.names
(datossemanales$SEMANA))
xsemanal <- summary(datossemanales.PC)
varssemanal = datossemanales.PC$sdev2
varssemanal <- varssemanal/sum(varssemanal)
importanciaComponentesSemanal <- as.data.frame(rbind('Autovalores' = datossemanales.PC$sdev2, 'Desviación
Standar' = xsemanal$sdev, 'Proporción de Varianza' = varssemanal, 'Proporción Acumulada' = cumsum(varssemanal)))
graficoIndividuosSemanal <-data.frame(datossemanales.PC$scores[,1:2])
matrizdistanciassemanal <- dist(graficoIndividuosSemanal, method='manhattan')
kmeanssemanal <- kmeans(matrizdistanciassemanal,4)
GrapIndividuosSemanal <- data.frame (cbind (datossemanales$SEMANA, kmeanssemanal$cluster, datossemana-
les.PC $scores[,1:2]))
names(GrapIndividuosSemanal) <- cbind('Semana', 'IndiceAgrupamiento','Componente1','Componente2') eigenvec-
toresSemanal <- data.frame (Cp1=datossemanales.PC$loadings[,1], Cp2=datossemanales.PC$loadings[,2], Cp3 =
datossemanales.PC$loadings[,3], Cp4=datossemanales.PC$loadings[,4], Cp5=datossemanales.PC$loadings[,5])
sqlDrop(conn, 'GrapIndividuosSemanal')
sqlSave(conn, GrapIndividuosSemanal)
sqlDrop(conn, 'importanciaComponentesSemanal')
sqlSave(conn, importanciaComponentesSemanal)
sqlDrop(conn, 'eigenvectoresSemanal')
sqlSave(conn, eigenvectoresSemanal)
sqlDrop(conn, 'matrizcorrelacionSemanal')
sqlSave(conn, matrizcorrelacionSemanal)

```

```

# componentesprincipales + k means mensual
matrizcorrelacionMensual<- data.frame(cor(datosmensuales[3:7]))
datosmensuales.PC <- princomp (datosmensuales[3:7], cor=TRUE, scale = TRUE, labels = row.names(datosmensuales$MES))
xmensual <- summary(datosmensuales.PC)
varsmensual = datosmensuales.PC$sdev^2
varsmensual <- varsmensual/sum(varsmensual)
importanciaComponentesMensual <- as.data.frame(rbind('Autovalores' = datosmensuales.PC$sdev^2, 'Desviación
Standar' = xmensual$sdev, 'Proporción de Varianza' = varsmensual, 'Proporción Acumulada' = cumsum( varsmen-
sual))) graficoIndividuosMensual <- data.frame(datosmensuales.PC$scores[, 1:2])
matrizdistanciasMensual <- dist(graficoIndividuosMensual, method='manhattan')
kmeansmensual <- kmeans(matrizdistanciasMensual,4)
GrapIndividuosMensual <- data.frame( cbind ( datosmensuales $MES, kmeansmensual$cluster, datosmensuales.PC
$scores[, 1:2])
names(GrapIndividuosMensual) <- cbind('Mes','IndiceAgrupamiento','Componente1','Componente2')
eigenvectoresMensual <- data.frame(Cp1 = datosmensuales.PC$loadings[, 1], Cp2 = datosmensuales.PC$loadings[,2],
Cp3 = datosmensuales.PC$loadings[,3], Cp4 = datosmensuales.PC$loadings[,4], Cp5 = datosmensuales.PC$loadings[,5])
sqlDrop(conn,'GrapIndividuosMensual')
sqlSave(conn,GrapIndividuosMensual)
sqlDrop(conn,'importanciaComponentesMensual')
sqlSave(conn,importanciaComponentesMensual)
sqlDrop(conn,'eigenvectoresMensual')
sqlSave(conn,eigenvectoresMensual)
sqlDrop(conn,'matrizcorrelacionMensual')
sqlSave(conn,matrizcorrelacionMensual)

# Este script manda un correo por medio de la base de datos avisando que se ha ejecutado el script, la base de
datos deberá estar configurada base de datos deberá estar configuradacon un servidor de correos.
sqlQuery(conn,'DECLARE tableHTML NVARCHAR(MAX); DECLARE VARCHAR(max); SET tableHTML = 'La
rutina se ha corrido satisfactoriamente en' CONVERT ( VARCHAR(12), GETDATE(),107 ); DECLARE Mergedtable
NVARCHAR(MAX) ; SET Mergedtable = tableHTML SET s = 'Ejecucion de Rutina' CONVERT(VARCHAR(12),GETDATE(),107);
EXEC msdb.dbo.sp_send_dbmail recipients = 'alejandro.rangelcierto.com.mx', profile_name = 'Reporting Services',
subject =s, body = Mergedtable, body_format = 'HTML;')

# cerrar conexión del ODBC
odbcClose(conn)

```

Bibliografía

- [1] Neeraj Sharma et All. *Database Fundamentals*. IBM Corporation, 2010. URL: http://69.24.90.63:2323/Alejandro_Rangel_Narvaez/Libros/Database_fundamentals.pdf.
- [2] Carvajal Carlos Andrés. *Tableros de Control para un proceso del negocio en una compañía de seguros*. Medellín: Universidad Nacional de Colombia, 2009. URL: http://69.24.90.63:2323/Alejandro_Rangel_Narvaez/Libros/Tableros_de_control.pdf.
- [3] Few Stephen. *Information Dashboard Design*. Edit. O'Reilly, 2006. URL: http://69.24.90.63:2323/Alejandro_Rangel_Narvaez/Libros/Information_Dashboard_Design.pdf.
- [4] Yoskovitz Benjamin Croll Alistai. *Lean Analytics use data to build a better sartup faster*. Edit. O'Reilly, 2013. URL: http://69.24.90.63:2323/Alejandro_Rangel_Narvaez/Libros/Lean_Analytics.pdf.
- [5] Oracle Corporation. *Oracle Database Documentation Library*. 2016. URL: <http://docs.oracle.com/en/database/database.html>.
- [6] Oracle Corporation. *MySQL Documentation*. 2016. URL: <http://dev.mysql.com/doc/>.
- [7] Microsoft Corporation. *Microsoft SQL Server Library*. 2016. URL: [https://msdn.microsoft.com/library/mt590198\(v=sql.1\).aspx](https://msdn.microsoft.com/library/mt590198(v=sql.1).aspx).
- [8] García Molina Héctor et all. *Database System the complete book*. 2a. New Jersey: Department of Computer Science Stanford University, Pearson Prentice Hall, 2009. URL: http://69.24.90.63:2323/Alejandro_Rangel_Narvaez/Libros/Database_The_Complete_Book.pdf.
- [9] Microsoft Corporation. *SSIS Tutorial: Creating a Simple ETL Packate*. 2016. URL: [https://msdn.microsoft.com/en-us/library/ms169917\(v=sql.120\).aspx](https://msdn.microsoft.com/en-us/library/ms169917(v=sql.120).aspx).
- [10] R Development Core Team. *The R Manuals*. 2016. URL: <https://cran.r-project.org/manuals.html>.
- [11] R Studio Support. *Documentation for RStudio products and services*. 2016. URL: <https://support.rstudio.com/hc/en-us/categories/200035113-Documentation>.
- [12] Oracle Corporation. *Java Plattform Standard Edition 8 Documentation*. 2016. URL: <https://docs.oracle.com/javase/8/docs/>.
- [13] The Eclipse Foundation. *Eclipse Documentation*. 2016. URL: <http://www.eclipse.org/documentation/>.
- [14] Oracle Corporation. *Oracle ADF Documentation Library*. 2016. URL: <http://docs.oracle.com/middleware/1221/adf/index.html>.
- [15] Oracle Corporation. *Glassfish Server Documentation*. 2015. URL: <https://glassfish.java.net/documentation.html>.
- [16] Rencher C. Alvin. *Methods of Multivariate Analysis*. 2a. New Jersey: Bringham Young University, Edit Wiley Interscience, 2002. URL: http://69.24.90.63:2323/Alejandro_Rangel_Narvaez/Libros/Wiley_series_in_probability_and_mathematical_statistics.pdf.
- [17] Jensen Steuard. *An Introduccion to Lagrange Multipliers*. 2015. URL: http://69.24.90.63:2323/Alejandro_Rangel_Narvaez/Libros/An_Introduction_to_Lagrange_Multipliers.pdf.
- [18] Brian Ripley. *Package RODBC Reference Manual*. 2016. URL: <https://cran.r-project.org/web/packages/RODBC/RODBC.pdf>.