



# **TECNOLOGÍAS COMPUTACIONALES EMERGENTES.**

**Análisis de datos, Investigación e Innovación Tecnológica**

**Guillermo Rodríguez López**  
**Luis Jara Obregón**  
**Diego Cordero Guzmán**



# Tecnologías Computacionales Emergentes

Análisis de datos, Investigación e Innovación Tecnológica



## **Tecnologías Computacionales Emergentes. Análisis de datos, Investigación e Innovación Tecnológica**

### **© Autores:**

Guillermo Rodríguez López

Docente de la Universidad Católica de Cuenca.

Luis Jara Obregón

Docente de la Universidad Católica de Cuenca.

Diego Cordero Guzmán

Docente de la Universidad Católica de Cuenca.

© Universidad Católica de Cuenca

© Editorial Universitaria Católica de Cuenca

**Primera edición:** 03 de mayo de 2023

**ISBN:** 978-9942-27-192-1

**e-ISBN:** 978-9942-27-193-8

**Editor:** Dr. Ebingen Villavicencio Caparó

**Edición y corrección:** Dra. Nube Rodas Ochoa

**Diseño y maquetación:** Od. Juan Pablo Cárdenas López

**Diseño de portada:** Od. Juan Pablo Cárdenas López

Impreso por Editorial Universitaria Católica (EDUNICA)

**Dirección:** Tomás Ordóñez 6-41 y Presidente Córdova

**Teléfono:** 2830135

**E-mail:** edunica@ucacue.edu.ec



Esta obra cumplió con el proceso de revisión por pares académicos bajo la modalidad de doble par ciego.

Queda prohibida la reproducción total o parcial de la obra sin permiso por escrito de la Universidad Católica de Cuenca, quien se reserva los derechos para la primera edición.

# Prólogo

En este libro presentamos técnicas de las ciencias computacionales para el análisis de datos usando herramientas informáticas de avanzada, que las denominamos “**Tecnologías Computacionales Emergentes**” (TCE). A lo largo del libro, guiamos a los lectores a conocer el campo de la Informática, la Innovación y la Investigación como elementos que se complementan. Este libro está organizado por capítulos, donde se analizan tópicos específicos de nueva generación.

Capítulo I. Dedicamos a la ciencia de datos y la investigación, como eje de estudio para el desarrollo de nuestra región. Mostramos la teoría de grafos usando algoritmos computacionales simples y complejos para el análisis de mapas geoespaciales. Demostramos que los grafos son eficientes para estudiar las redes urbanas de varias ciudades de Latinoamérica. Revisamos problemáticas del crecimiento poblacional, incremento del tráfico vehicular, movilidad de la población, aglomeraciones comerciales, análisis de las redes sociales, patrones y agrupamientos. Para los casos prácticos, utilizamos aplicaciones como: *Python*, *Networkx*, *OSMnx*, *MatplotLib*, *NumPy* y *Pandas*. Estas herramientas sirven para la extracción, preparación y análisis de datos. Presentamos casos de estudio enfocados en desastres naturales o sociales que afectan la resiliencia de las infraestructuras urbanas. Es decir, en los ejemplos prácticos, se aplican estas técnicas para medir la capacidad que tienen las infraestructuras urbanas y sociales para resistir, amortiguar, adaptarse y recuperarse de eventos adversos. Por medio de estos análisis, pretendemos motivar a los planificadores urbanos para que diseñen mejores ciudades para el futuro.

Capítulo II. Presentamos conceptos modernos de planificación y gestión empresarial con soporte en las TCE. Alineamos los conceptos de gestión con la tecnología informática para brindar apoyo a decisiones gerenciales. A partir de esta relación, planteamos maximizar la productividad de los recursos empresariales valiosos en la actualidad, por ejemplo, datos, procesos, involucrados y resultados, con un monitoreo constante de las TCE en tableros de control. Demostramos cómo dinamizar los procesos y actividades de gestión usando un *Balanced Scorecard*. Estos, elementos de gestión ahora enlazados a herramientas computacionales para medir el rendimiento de las organizaciones.

Capítulo III. Enfocamos el contenido en la Inteligencia de Negocios para alinearlo a los procesos empresariales revisados en el Capítulo II. Aquí se muestran los procedimientos técnicos, desde la extracción de datos y depuración, hasta la elaboración de indicadores de desempeño y la presentación de la información en los tableros de control. Usamos herramientas para servicios de integración, servicios de análisis y servicios de reportes. Los ejemplos prácticos son realizados en SQL Server R2 y Bu-

Business Intelligence de Microsoft. Aquí se demuestra la creación y uso de un datamart, datawarehouse y cubos OLAP.

Capítulo IV. Está centrado en el manejo de los tableros de control o dashboard. Presentamos casos de estudio, para diseñar presentaciones con granularidad fina. Estos tableros se crean para dinamizar los informes gerenciales, que ayudan a la gerencia a tomar decisiones, con una información relevante y confiable de fácil manejo.

Capítulo V. En este capítulo describimos las técnicas de minería de datos. Aquí demostramos y usamos el manejo de diversos algoritmos para la búsqueda de patrones, clasificaciones, segmentaciones y comportamientos de los datos masivos. Presentamos el proceso KDD para la extracción, depuración, transformación, interpretación y presentación de los resultados. También, usamos un caso de estudio como ejemplo de análisis.

Capítulo VI. En este último capítulo, mostramos el uso de Tableau y Power BI. como herramientas modernas para la aplicación de la Inteligencia de Negocios (Business Intelligence BI). Usando datos históricos y actuales de empresas dedicadas a la actividad comercial.

Los objetivos de cada uno de los capítulos de este libro están orientados al uso óptimo de las TCE en problemas cotidianos. Por otro lado, también se presentan algunas alternativas para aprovechar las ciencias computacionales de última generación. Los ejemplos expuestos están dirigidos a los sectores comerciales, industriales, urbanísticos de movilidad y transporte, y están orientados a una mejor gestión, toma de decisiones y generación de nuevas políticas públicas.

Este libro está sustentado en experiencias de los autores y en una amplia revisión literaria de trabajos académicos, libros y artículos científicos. Otros de los propósitos es reducir el tiempo y la complejidad de crear nuevas aplicaciones informáticas. En los casos prácticos, usamos el concepto de la transferencia del aprendizaje (transfer learning). Es decir, construimos nuevas aplicaciones, tomando como base otras aplicaciones existentes para acelerar el desarrollo. Finalmente, queremos transferir el conocimiento adquirido para mejorar el desarrollo y la calidad de vida de la sociedad. Pero, sobre todo, queremos motivar a los lectores, especialmente a los estudiantes, a que aprecien la importancia de las ciencias computacionales.

## CONTENIDO

<b>PRÓLOGO</b> .....	3
<b>CAPÍTULO I</b>	
1.1 Tecnologías Computacionales Emergentes (TCE).....	13
1.2 Ciencias Computacionales para el Análisis de Datos.....	14
1.2.1 Grafos y Redes Urbanas.....	14
1.2.1 El grado de un grafo.....	15
1.2.2 Conectividad de grafos o redes.....	18
1.2.3 Malla y Orgánico.....	20
1.2.4 Centralidad de Intermediación (Betweenness Centrality).....	21
1.2.5 Centralidad Cercana (Closeness Centrality CC).....	23
1.2.6 Elevaciones en mapas georreferenciales.....	27
1.2.7 Mediciones de tiempos de viaje dentro de una red urbana.....	28
<b>CAPÍTULO II</b>	
2.1 Combinación de Modelos Gerenciales y TCE.....	35
2.1.1 Planificación empresarial estratégica basada en las Tecnologías de la Información.....	35
2.1.2 Calidad Total con intermediación de las TIC.....	36
2.1.3 Kaizen o automatización.....	37
2.1.4 Justo a tiempo en la era digitalizada usando TIC.....	38
2.1.5 Reingeniería: El Modelado y análisis de procesos de negocios.....	38
2.1.6 Análisis de Benchmarking con aplicaciones TCE.....	39
2.1.7 Empoderamiento (Empowerment).....	39
2.1.8 La externalización (Outsourcing) de las TCE como estrategia.....	39
2.2 Fusión de las empresas con la Tecnología.....	40
2.2.1 Cuadro de mando integral.....	40
2.2.2 Aportes o beneficios del BSC.....	41
2.2.3 Estructura del BSC.....	42
2.2.4 Mapa y matriz estratégicos del BSC.....	42
2.2.5 Elementos principales del BSC.....	45
2.2.6 Los indicadores claves de desempeño ( <i>key performance indicators KPI</i> ).....	47
2.2.7 Elementos claves del KPI.....	48
2.2.8 Técnicas para identificar los KPI's.....	48
2.2.9 Técnicas para medir los KPIS.....	49
<b>CAPÍTULO III</b>	
3.1 Inteligencia de Negocios (Business Intelligence BI).....	53
3.1.1 El BI como parte de las Tecnologías Informáticas TI.....	53
3.1.2 La inteligencia de negocios en el nuevo milenio.....	53
3.1.3 Herramientas de inteligencia de negocios (BI).....	54
3.2 Almacén de datos (Datawarehouse Dw).....	56
3.2.1 Modelo Estrella.....	56
3.2.2 Modelo Copo de nieve.....	57

3.2.3	La Dimensión tiempo (Tabla) en el DW.....	57
3.2.4	La granularidad.....	57
3.2.5	La tienda de datos (Datamart o DM).....	59
3.2.6	Datamart OLAP.....	59
3.2.7	Datamart OLTP.....	59
3.3	Procesos de extracción, transformación y carga (del inglés, extraction, transformation and loading ETL).....	59
3.3.1	Beneficios de los procesos ETL.....	61
3.4	Herramientas de extracción transformación y carga (ETL).....	61
3.4.1	Tarea ejecutar SQL.....	62
3.4.2	Conectar con un origen de datos.....	62
3.4.2	Tarea flujo de datos.....	63
3.4.3	Varios flujos.....	64
3.4.4	Entradas del registro.....	64
3.4.5	Tarea flujo de datos en Integration Services de SQL Server.....	65
3.4.6	Tareas de preparación de datos.....	65
3.4.7	Tareas de flujo de trabajo.....	65
3.4.8	Tareas de SQL Server.....	65
3.4.9	Tareas de scripting.....	66
3.4.10	Tareas de Analysis Services.....	66
3.4.11	Tareas de mantenimiento.....	66
3.4.12	Tareas personalizadas.....	66
3.5	Arquitecturas ROLAP, MOLAP Y HOLAP.....	66
3.6	La generación de cubos, dimensiones, medidas y jerarquías.....	69
3.6.1	Creación del proyecto Analysis Services, caso de estudio.....	69
3.6.2	Creación del origen de datos.....	70
3.6.3	Creación de la vista del origen de datos.....	70
3.6.4	Creación de las dimensiones del cubo.....	71
3.6.5	Creación y diseño del cubo.....	71
3.6.6	Implementación del cubo.....	72
3.6.7	Creación de la tabla tiempos con sus respectivas jerarquías.....	73
3.7	Indicadores claves de desempeño (KPI's).....	75
3.7.1	Lenguaje MDX para consultas multidimensionales.....	77
3.7.2	Sintaxis básica del lenguaje MDX.....	78
3.7.3	Usar expresiones de dimensiones.....	80
3.7.4	Expresiones de dimensiones.....	80
3.7.5	Expresiones de jerarquías.....	81
3.7.6	Bases de datos y lenguaje NoSQL (No only SQL).....	81

## CAPÍTULO IV

4.1	Tableros de control (Dashboards).....	87
4.1.1	Creación del proyecto del servicio de reportes.....	87
4.1.2	Definir el origen de datos.....	88
4.1.3	Definir un conjunto de datos para el informe.....	89
4.1.4	Agregar la Tabla Pívor al informe del Reporting Services SSRS.....	90
4.1.5	Diseño del informe georreferencial con mapas, gráficos agrupados (clústeres)	90
4.1.6	Informes para investigación científica con JSon, Csv y Phytion o Spark.....	91

4.1.7 Extracción de datos de Twitter.....	91
4.2 Herramientas avanzadas de exploración OLAP.....	94
4.2.1 Drill Down/up (Desglosar) y Drill Through (perforar a través de).....	94
4.2.2 Configuración del Drilldown, sub-informes o anidamiento.....	95
4.2.3 Implementación y vista previa del Drilldown.....	95
4.2.2 Drill-Through (Perforar a través) informes detallados.....	97
4.2.3 Especificando la consulta.....	97
4.2.4 Definiendo filtros.....	97
4.3 Caso de estudio para aplicar técnicas de BI.....	98
4.3.1 Abstracción del problema.....	99
4.3.2 Caracterizar el problema de la empresa.....	99
4.3.3 Establecer el objetivo que va a solucionar o apoyar el proyecto BI.....	100
4.3.4 Normalizar los requerimientos del negocio, tanto estratégicos como tecnológicos.....	100
4.3.5 Mapear el BSC.....	100
4.3.6 Análisis de la data y meta-data.....	101
4.3.7 Construcción del Datamart ajustado al caso de estudio.....	102
4.3.8 Crear el ETL para extraer y transformar la información.....	102
4.3.9 Construcción del cubo OLAP con Análisis Services (SSAS).....	104
4.3.10 Creación y configuración de los KPIS.....	106
4.4. Creación de los tableros de control.....	107
4.4.1 Uso de tablas dinámicas y técnicas de Drill Down, Drill Through.....	107
4.4.2 Generación de gráficos dinámicos, mapas, líneas, pie y barras.....	110
4.3.13 Clasificación ABC con los productos más vendidos.....	112
4.3.14 Evaluación de los resultados y mejora continua.....	113

## CAPÍTULO V

5. Minería de datos.....	117
5.1 Definición de minería de datos (Datamining).....	117
5.2 Minería de texto (Textmining).....	118
5.2.1 Caso práctico.....	118
5.3 Tipos de algoritmos de minería de datos.....	122
5.3.1 Árboles de decisión de Microsoft.....	122
5.3.2 Cómo funciona el algoritmo de árboles de decisión.....	122
5.3.3 Naive Bayes.....	123
5.3.4 Algoritmo de clústeres de Microsoft.....	124
5.3.5 Algoritmo de red neuronal de Microsoft.....	126
5.3.6 Algoritmos de regresión lineal de Microsoft.....	126

## CAPÍTULO VI

6.1 Visión general.....	131
6.2 Hadoop.....	131
6.3 HDFS.....	131
6.4 Spark.....	132

ÍNDICE DE FIGURAS

Figura 1. Representación general de un grafo..... 14

Figura 2. Redes de (a) Guayaquil, (b) Quito y (c) Panamá, obtenidas usando código osmnx..... 16

Figura 3. Imágenes de las redes de Guayaquil (a), Quito (b) y Panamá (c), usando osmnx..... 17

Figura 4. Betweenness Centrality BC, (a) ciudad de Cuenca, Ecuador y (b) Panamá, Panamá..... 22

Figura 5. Closeness Centrality CC. (a) Cuenca-Ecuador, (b) Barcelona-España. 23

Figura 6. Centralidad Cercana (CC), de (a) Manta-Ecuador y (b) Caracas-Venezuela..... 24

Figura 7. Ciudad de Guadalajara-México. (a) Los nodos BC y (b) las intersecciones de CC..... 25

Figura 8. Muestra las mediciones de BC y CC de Barcelona-España..... 26

Figura 9. (a) Red Urbana de elevaciones de Guayaquil, (b) red urbana de elevaciones de Lima..... 28

Figura 10. Cálculo de Rutas óptimas, desde x a y distancia 4.31 km., línea de color rojo..... 32

Figura 11. Método para aplicar kaizen en la práctica..... 37

Figura 12. Implementación del mapa y matriz estratégica del BSC..... 43

Figura 13. Ejemplo de operacionalización de la estrategia..... 44

Figura 14. Interpretación de las perspectivas del BSC..... 44

Figura 15. Método para diseñar las perspectivas del BSC..... 45

Figura 16. Interfaz de SQL Server para el diseño de un kpventas en Analysis Services..... 47

Figura 17. Comparación de los modelos (a) Estrella y (b) Copo de nieve de un DW..... 57

Figura 18. Dimensión tiempo dentro de un cubo OLAP de SQL Server..... 58

Figura 19. Insertando una tarea ejecutar SQL desde Business Intelligence..... 62

Figura 20. Administrador de conexiones en Busiess Intelligence de SQL Server 63

Figura 21. Tarea flujo de datos para mezclar tablas y cargar al destino datawarehouse..... 64

Figura 22. Tarea flujo de datos ejecutada, para transformar una columna derivada 65

Figura 23. Representación de las tres arquitecturas OLAP..... 67

Figura 24. Ventana del nuevo proyecto de Analysis Services, para creación de los cubos OLAP..... 69

Figura 25. Ventana del diseñador del origen de datos, conexión con el DW..... 70

Figura 26. Ventana del diseñador del origen de datos, conexión con el DW..... 70

Figura 27. Ventana de dimensiones y definición de atributos y jerarquías del cubo 71

Figura 28. Ventana proceso de creación del cubo OLAP con sus requerimientos 72

Figura 29. Implementación del cubo OLAP satisfactoriamente..... 72

Figura 30. Cubo OLAP Procesado y listo para ser examinado (SSAS)..... 73

Figura 31. Proceso de para la creación de la dimensión tiempo..... 74

Figura 32. Proceso de para la creación de la dimensión tiempo..... 74

Figura 33. Proceso de cubo OLAP terminado y listo para ser examinado SSAS 75

Figura 34. Sintaxis SQL básica con MDX en SQL Server..... 78

Figura 35. Ejemplo de consulta SQL usando lenguaje MDX.....	78
Figura 36. Clasificación de los sistemas de gestión de bases de datos nosql.....	83
Figura 37. Ventana para iniciar un proyecto de SSRS con Tableros de control.	88
Figura 38. Configuración del origen de datos para el reporte.....	89
Figura 39. Definición del conjunto de datos para crear la consulta del reporte....	89
Figura 40. Creación de la tabla pivot para mostrar el tablero de control final...	90
Figura 41. Formato final del informe (a) mapa (b) dashboard.....	91
Figura 42. Datos de Twitter. Catástrofes naturales (a) lineal x, y (b) pastel.....	93
Figura 43. Navegación por un cubo de datos mediante Drill Down y Drill Up.....	94
Figura 44. Configuración de grupos y anidamiento para crear un Drilldown en SQL Server.....	95
Figura 45. Resultado de un reporte aplicando Drill Down y Drill Up.....	96
Figura 46. Despliegue de un Drill Down.....	96
Figura 47. Formularios vinculados por medio de Drillthrough.....	98
Figura 48. Mapa estratégico del BSC.....	100
Figura 49. Fuente de datos de la base relacional MS Access de la empresa.....	101
Figura 50. Tabla de análisis de la data de origen vs destino.....	101
Figura 51. Modelo estrella para el datamart de la solución.....	102
Figura 52. Diseño de Tarea Ejecutar SQL en Integration Services.....	103
Figura 53. Diseño de Tarea lujo de datos en Integration Services.....	103
Figura 54. Diseño del Flujo de control completo para el ETL de la solución en el SSIS.....	104
Figura 55. Ventana para configurar la dimensión tiempo en sus distintas jerarquías	104
Figura 56. Ventana de SSAS luego de crear el cubo con los elementos de diseño	105
Figura 57. Ventana de SSAS para implementar el cubo OLAP.....	105
Figura 58. Ventana para crear KPIS en SSAS.....	106
Figura 59. SSAS de kpiclientes y kpventas en un gauge de estado y tendencia	107
Figura 60. SSRS muestra la creación de un reporte con Drill Down.....	108
Figura 61. SSRS Asignando origen de datos para el reporte.....	108
Figura 62. SSRS Configuración y definición de grupos para armar el Drill Down	109
Figura 63. SSRS Presentación del informe para su respectivo análisis con Drill Down	109
Figura 64. Distribución de ventas por provincias utilizando un mapa ESRI.....	110
Figura 65. SSRS Configuración y definición de un gráfico dinámico dentro del informe.....	111
Figura 66. Power BI Análisis de una base de datos del módulo de ventas.....	112
Figura 67. Reporte de los productos más vendidos usando el criterio ABC.....	113
Figura 68. Despliegue del proceso para el descubrimiento del conocimiento.....	118
Figura 69. Muestra el análisis de Text Mining realizado desde Python.....	121
Figura 70. Histograma de un árbol de decisión.....	122
Figura 71. Histograma de un árbol de decisión nuevo nodo de análisis.....	123
Figura 72. Histograma de un árbol de decisión nuevo nodo de análisis.....	124
Figura 73. Representación de los clusters A y B.....	125
Figura 74. Gráfico de dispersión para todos los grupos de datos.....	125
Figura 75. Línea de mejor representación del método de regresión lineal.....	126
Figura 76. Gráfico de análisis de la tendencia de las palabras repetidas de un texto usando Spark.....	134
Figura 77. Gráfico analítico del conteo de palabras repetidas de un texto con Spark y Python.....	135

## Índice de Tablas

Tabla 1. Estadística de conectividad para Guayaquil y Quito.....	19
Tabla 2. Muestra las mediciones de tiempos realizadas en Waze.....	29
Tabla 3. Criterios de calidad según Schnaars.....	36
Tabla 4. Ilustración del diseño de un KPI completo dentro del BSC.....	46
Tabla 5. Aportes de un proceso de inteligencia de negocios BI.....	54
Tabla 6. Diferencias entre ROLAP y OLAP.....	68
Tabla 7. Términos y definiciones de la sintaxis del lenguaje MDX.....	79
Tabla 8. Conversiones y usos de signos y símbolos en MDX.....	80

# CAPÍTULO I

## Ciencias Computacionales



## 1.1 Tecnologías Computacionales Emergentes (TCE)

Las Tecnologías Emergentes han evolucionado desde las últimas dos décadas y en el presente, aún son un tema de discusión en diferentes proyectos relacionados al campo tecnológico e informático. Por tanto, resulta oportuno profundizar en el uso de las TCE y aprovecharlas en el ámbito académico o investigativo, como herramientas y técnicas en desarrollo y perfeccionamiento. La literatura muestra a autores que han aportado a este tema desde décadas atrás para referirse a las nuevas tecnologías de la información y comunicación (TIC) o a la realidad aumentada (Haller, 2006). Sin embargo, más recientemente, se está abriendo el campo de ciencias modernas para referirse a la biotecnología, la nanotecnología y el desarrollo de nuevas tecnologías. Estas tecnologías permiten fusionar eventos en tiempo real con procesos digitalizados a través de las computadoras (Einsiedel, 2009). En los últimos tres años se han realizado investigaciones sobre tecnologías emergentes de aprendizaje basadas en dispositivos móviles y las redes sociales web 2.0 con aprendizaje profundo y algoritmos de entrenamiento (Wu *et al.*, 2016) y (Murray & Kidd, 2017). Otros estudios, contemplan la nano electrónica y hardware de seguridad aplicados en el campo de la salud y la medicina. Por ejemplo, el monitoreo del efecto de distintos anestésicos (Tehranipoor *et al.*, 2017) y (Kimuyu, 2017).

También encontramos los estudios que tratan sobre las redes convolucionales para mejorar las imágenes de baja resolución y reconstruirlas con mínimos márgenes de error (Akhtar *et al.*, 2014)). Más adelante, hallamos conferencias que abordan el uso de red profunda para reconocimientos de imágenes a través de análisis de multi-vistas y vistas cruzadas, este tema se amplía en Kan *et al.* (2016).

Otro campo explotado dentro de las tecnologías emergentes es la minería de datos, en Ranu *et al.* (2013), proporcionan un mecanismo para modelar la creciente heterogeneidad en los datos generados por los sistemas actuales, utilizando subgrafos discriminativos minados. Estas técnicas permiten identificar la influencia de los nodos de un subgrafo en toda la red y descubrir relaciones complejas, comportamientos e insidencias, colectivamente. Otros estudios demuestran que los grafos de red mutados, la minería de datos y los aprendizajes automáticos se pueden combinar para entender mejor los problemas de cáncer y el efecto de sus fármacos (Zhang *et al.*, 2017) En cuanto a la teoría de grafos se cuenta con un campo extenso por estudiar. Los grafos se utilizan para evaluar la resistencia, robustez y vulnerabilidades de las redes urbanas (Yaoli *et al.*, 2013) y (Wang, 2015) y (Masucci & Molinero, 2016) . Las investigaciones sobre estos temas sirven para presentar mediciones de los índices del tráfico vehicular, los embotellamientos y los tiempos estimados que tardan los traslados a través de las rutas recomendadas por las herramientas informáticas.

En el campo empresarial e industrial las tecnologías emergentes también están siendo consideradas en múltiples investigaciones. En este tópico se encuentran tecnologías como, *Cloud Computing*, Inteligencia de datos, Computación verde, Informática móvil, auto computación, Computación social como las empresas 2.0 (Lewis, 2010).

Igualmente, están los sistemas empresariales orientados a los negocios inteligentes y los sistemas de planificación de recursos empresariales ERP de tercera generación (Xu L. D., 2011). Estas tecnologías han evolucionado con la aparición del internet de las cosas *IoT*, permiten numerosas oportunidades de negocios en campos tan diversos como salud electrónica, ciudades inteligentes, hogares inteligentes, vehículos autónomos, entre muchos otros (Elkhodr *et al.*, 2016) y (Holstein *et al.*, 2018). A continuación, se discuten algunas definiciones importantes en el campo empresarial para obtener resultados competitivos, que son claves para la aplicación de las tecnologías emergentes.

## 1.2 Ciencias Computacionales para el Análisis de Datos

Las ciencias computacionales son una combinación entre las distintas ciencias y la tecnología computacional. Las ciencias computacionales son una nueva área, creada para desarrollar estudios o investigaciones en los campos de la ciencia y la sociedad. Entre la variedad de aplicaciones está la inteligencia artificial, el aprendizaje automático y los algoritmos complejos (Dasgupta, 2016), la teoría de grafos, para análisis de redes de calles o redes sociales con modelamientos matemáticos (Gross *et al.*, 2018). Otro campo de aplicación consiste en que se pueden combinar las teorías matemáticas, la medicina y las tecnologías informáticas para modelar y simular el comportamiento de células y microorganismos, la movilidad, velocidad, motricidad, alteraciones o modificaciones, para predecir futuras transformaciones en la medicina para nuevos diagnósticos (Kent & Williams, 1996).

### 1.2.1 Grafos y Redes Urbanas

En tecnologías computacionales, los grafos son ampliamente utilizados en varios campos, entre ellos, el análisis de redes, por ejemplo. Los grafos son estructuras complejas y están formados por vértices ( $V$ ), también llamados nodos, y por bordes también conocidos como aristas (*del inglés Edge, E*).

Los grafos pueden ser utilizados para el análisis de redes sociales para evaluar el comportamiento de los miembros de una red, su grado de conectividad, longitud, centralidad, dispersión, resistencia, entre otros.

Un grafo se representa como  $G = (V, E)$  y está formado por un par de valores, relacionados entre sí formando una red simple o compleja (Ver Figura 1).

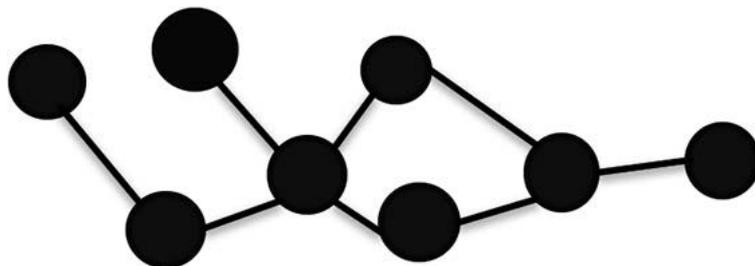


Figura 1. Representación general de un grafo

### 1.2.1 El grado de un grafo

Se conoce como el número de incidencias o aristas que conecta con ese vértice, por ejemplo, en el grafico el vértice a tiene grado 4, el vértice c tiene grado 2 y el vértice h tiene grado 1. La teoría de grafos tiene más de un siglo de invención, sin embargo, es en la actualidad que se aborda con más fuerza dada su estrecha relación con las TC, ya que sin ella es muy difícil y demanda mucho más tiempo el poder realizar análisis complejos. Existe una serie de características y teoremas para estudiarlos y analizarlos, tales como, grafos conexos, grafos no conexos, grafos dirigidos, bipartitos, duales, entre otros. Esto amplían con más detalle Seymour *et al.* (1989). En este libro se abordará el punto de vista aplicativo, en estudios concretos de investigación, tomando como ejemplos varias ciudades de Latinoamérica y Europa. Por consiguiente, para aplicar estas estructuras a estudios de investigación es preciso hacer referencia a algunas aplicaciones computacionales como *Networkx*, *OSMnx*, *ArcGIS*, entre otras.

A continuación, mostramos el proceso completo para analizar las redes viales para algunas ciudades de distintos países aplicando las propiedades de la teoría de grafos. Luego desarrollamos varias mediciones, topológicas y morfológicas de la red, tomando como parámetros principales el número de cruces de calles (nodos) y el número de tramos de calles (enlaces). Medimos los nodos sin salida que tiene una red y determinamos el promedio con el total de nodos de la red. Después, análogamente, evaluamos los cruces de calles que tienen cuatro enlaces para ubicar dónde es mejor planificada la red. Para la extracción de las redes y los cálculos utilizamos *Python*, *Networkx* y *OSMnx*.

El primer proceso es la carga de las redes urbanas. Para esto existen varios métodos tales como: nombre del lugar y país, *bounding box*, coordenadas del polígono, o coordenada de centro y radio. Esta vez utilizamos el método *bounding box*. Los parámetros son las coordenadas de latitud y longitud (norte, sur, este y oeste). La primera medición de las redes urbanas consiste en calcular cuántos callejones sin salida tiene cada red. Los resultados son presentados en el mapa de la Figura 2 (Boeing, 2017), y el código del algoritmo a continuación.

```

%%time
import osmnx as ox
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
print('South Quito Network')
north, south, east, west = -0.238316, -0.355507, -78.499167, -78.618005
Gno = ox.graph_from_bbox(north, south, east, west, network_type='drive')
culdesacs = [key for key, value in Gno.graph['streets_per_node'].items() if value==1]
nc = ['r' if node in culdesacs else 'none' for node in Gno.nodes()]

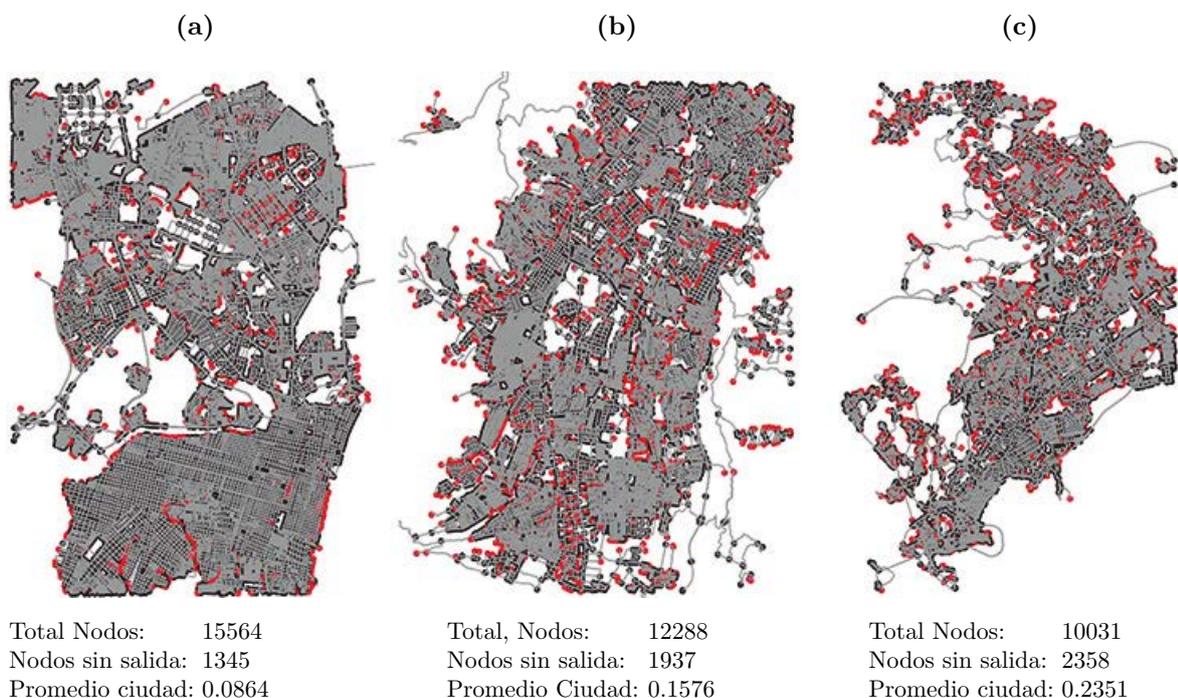
```

```

fig, ax = ox.plot_graph(Gno, node_color=nc)
n=len(Gno)
nss=len(culdesacs)
pnss=nss/n
print(' Total Nodos: ', n, ' Nodos sin salida: ', nss)
print('Promedio de nodos sin salida de la ciudad: ', round(pnss,4))
#para almacenar la red en formato de gráfico con extensión .tiff
fig.savefig("Quito_CuldeSacs.tiff", dpi=400, format="tiff")
print('listo')

```

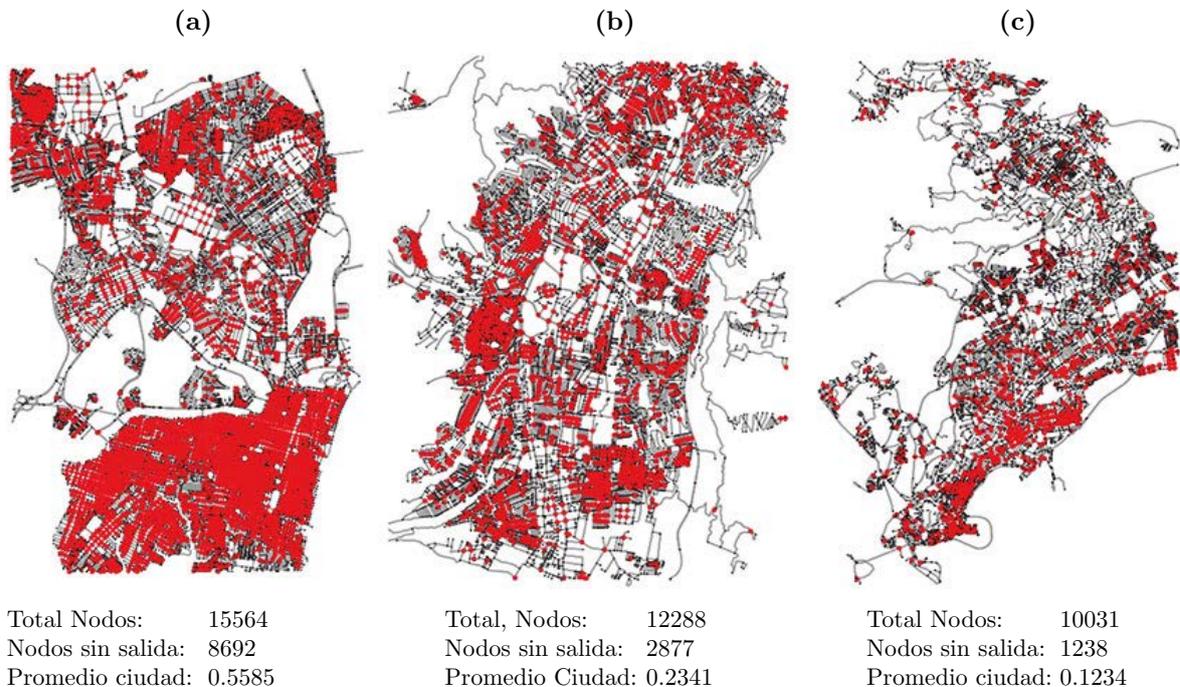
Mostramos cómo, al aplicar un análisis de grafos de red con métricas de malla y orgánico, se puede determinar la morfología de las ciudades de cualquier lugar del mundo. En la Figura 2, se calculan y presentan los nodos sin salida; es decir, los nodos de la red que están conectados por un solo borde. Para el caso de las ciudades, se conoce como un callejón sin salida o de retorno (por donde ingresa debe salir). En este caso, si una red tiene un número elevado de este tipo de nodos afecta a la conectividad de la ciudad y la hace más vulnerable frente a eventos climáticos o sociales adversos. En las redes urbanas evaluadas, la ciudad de Panamá tiene mayor promedio de callejones sin salida con el 0.2351, en una superficie menor. Luego, le sigue Quito sur, con 0.1576 y Guayaquil, con 0.0864. Esta última ciudad muestra mejor planificación urbana. Estos resultados hacen prever que el mayor promedio de callejones sin salida en una ciudad complica la circulación vehicular, puesto que, si se bloquea algún acceso que conecta estos nodos quedan aislados y posiblemente atrapados luego de un evento adverso.



**Figura 2.** Redes de (a) Guayaquil, (b) Quito y (c) Panamá, obtenidas usando código *OSMnx*.

En la Figura 2, se muestra las calles sin salida con los vértices pintados de rojo, en contraste con el análisis anterior. Este análisis permite ver qué sectores de las ciudades tienen una estructura autoorganizada, aspecto más frecuente en los nuevos asentamientos periféricos. Estas medidas generalmente son utilizadas para determinar si una ciudad o red urbana es planificada o autoorganizada (sin planificación o desorganizada), y resultan ser mediciones importantes pues permiten examinar el comportamiento de las ciudades y saber cuándo tienen un crecimiento acelerado. Este estudio se puede complementar con el siguiente código, que sirve para identificar el número de intersecciones con cuatro enlaces de tipo malla. Esto es típico de ciudades planificadas con calles en forma de cuadros, el resultado se muestra en la Figura 3. El código es el mismo que se utilizó en la extracción de los callejones sin salida, únicamente se debe cambiar la siguiente línea resaltada, cambiando el 1 por el 4. Sin embargo, los resultados obtenidos son relevantes para conocer mejor la morfología de una ciudad.

```
culdesacs = [key for key, value in Gno.graph['streets_per_node'].items() if value==4]
```



**Figura 3.** Imágenes de las redes de Guayaquil (a), Quito (b) y Panamá (c), usando *OSMnx*.

ver que las ciudades tienen su estructura tipo malla en sectores planificados y centros coloniales o históricos. Del mismo modo, mostramos una clasificación de las ciudades, según la planificación basados en el criterio de estructuras de calles de cuatro enlaces. En nuestro análisis tenemos primero, Guayaquil, con 0.5585, seguido de Quito, con 0.2341 y Panamá, con 0.1234. Las redes urbanas que poseen este índice más alto predicen una mejor planificación urbana. Estos nodos se agrupan habitualmente en los centros históricos bien planificados urbanísticamente. Según estos datos, podemos decir que la ciudad de Panamá es de tipo orgánico (autoorganizada), puesto que los cruces de calles con cuatro enlaces tienen un promedio bajo, por lo que da la

aparición de tener una planificación desordenada. Según los análisis y pruebas realizadas en las investigaciones, las ciudades autoorganizadas tienen ventajas siempre y cuando tengan vías eficientes de conectividad (amplias avenidas). La conectividad ayuda a compensar la intermediación de sus nodos (cruces de calles). Sin embargo, si estas redes urbanas no tienen vías de acceso amplias (avenidas de cuatro carriles en adelante), podrían sufrir mayor congestión del tráfico vehicular. Por lo tanto, para afirmar esta primera opinión es necesario realizar otras mediciones que la confirmen.

Cuando se analizan redes de calles urbanas utilizando las propiedades de redes y grafos se pueden hacer muchos estudios de las características de ciudades grandes o pequeñas. Para empezar con este largo estudio, primero, se deben extraer las redes urbanas de las ciudades usando las coordenadas georreferenciales, distancia y nombres de los lugares, tal como se aprecia en el siguiente código con *OSMnx* y *Python*.

```
G2 = ox.graph_from_bbox(north, south, east, west, network_type='drive')
```

```
G2 = ox.graph_from_point((lat,lon), distance, network_type='all')
```

```
G2 = ox.graph_from_place(lugar, network_type='drive', which_result = 2)
```

De este modo, con estos dos tipos de valores obtenidos, tanto el número de grado  $k=4$  (cruces de calles) y el número de grado  $k=1$  (calles sin salida) nos indican que los sectores con mayor número de vértices de grado  $k=4$  son zonas planificadas y los sectores con mayor vértice de grado  $k=1$  son zonas más autoorganizadas. Generalmente, este último se encuentra más en áreas urbanas modernas o de asentamientos urbanos recientes.

### 1.2.2 Conectividad de grafos o redes

La conectividad es la capacidad que tiene un grafo o una red para enlazarse entre cada nodo a través de sus bordes o aristas (*edges*). Se calcula utilizando la siguiente relación matemática  $C = \frac{E}{V}$ , donde  $E$  es el número de aristas (*edges*) y  $V$  es el número de vértices del grafo. Para describir y explicar mejor este concepto se muestra el siguiente ejemplo realizado con un análisis a las redes de calles de Guayaquil y Quito, las ciudades más pobladas del Ecuador.

Código basado en Python con librerías de *Networkx* y *OSMnx*.

```

import osmnx as ox

G = ox.graph_from_place('Guayaquil, Ecuador', network_type='drive')

ox.plot_graph(ox.project_graph(G))

# v son los vertices

v=len(G)

# e son los bordes

e=G.number_of_edges()

p=e/v

print ("Vertices es: ",v)

print ("Edges es: ",e)

print ("La conectividad de Guayaquil es: ", p)

```

A continuación, se visualizan los datos obtenidos con el código anterior, para contrastar las tres ciudades analizadas, ver Tabla 1.

<b>Ciudades</b>	<b>Vértices (Nodes)</b>	<b>Bordes (Edges)</b>	<b>Conectividad</b>
Guayaquil	36717	103531	2.82
Quito	24436	62855	2.57
Ciudad Panamá	16871	39461	2.34

**Tabla 1.** Estadística de conectividad para Guayaquil y Quito

Estos datos indican que la ciudad de Guayaquil es más extensa que Quito por tener mayor número de nodos de conexión (vértices) y enlaces (*edges*). También indica que el grado de conectividad de Guayaquil es mayor que el de Quito. Por último, encontramos que la ciudad de Panamá tiene menor conectividad y menor número de nodos. Esto significa que Guayaquil tiene mejor conectividad de su red y responde mejor a la circulación del transporte que Quito y Panamá. A partir de este breve análisis, podemos exponer que Guayaquil es una ciudad más fuerte y resistente ante posibles fenómenos sociales (huelgas, disturbios políticos, eventos culturales, deportivos) o desastres naturales (inundaciones, deslaves, terremotos, entre otros). En el caso que estos fenómenos provocaran el bloqueo de calles e intersecciones de la ciudad, podemos decir que Guayaquil tendría mejor capacidad para reaccionar y dispersar el tráfico por otras vías, que Quito y Panamá. Es decir, habrá menor grado de congestión del transporte vehicular (Wang, 2015).

Siguiendo este mismo mecanismo, se puede ir evaluando otras propiedades de las redes de calles, como la malla, orgánico, centralidad de interconexión (del inglés *Be-*

*tweenness Centrality*) y cercanía. La centralidad cercana (del inglés *Closeness Centrality CC*), en cambio, consiste en mediciones del número de saltos desde un vértice  $i$  al resto de vértices analizados (Wang, 2015). Esta medida ayuda a determinar el grado de aglomeración que puede tener una ciudad.

### 1.2.3 Malla y Orgánico

Con las fórmulas para malla  $M$  y orgánico  $O$  y  $F$  número de lados, se calculan medidas para comprobar si una ciudad es planificada u organizada con calles tipo cuadradas o autoorganizadas. La malla  $M$  de un gráfico se utiliza para medir su estructura. Es importante tener en cuenta que un gráfico simple es un gráfico donde ninguno de sus bordes se cruza con otro. Además, la región (o cara) de un gráfico se define como un área delimitada por bordes y no puede continuar dividiéndose en sub áreas. Partiendo del número de caras  $F$  de un gráfico  $G$  y tomando la fórmula de Euler, la malla estaría representada por la siguiente ecuación:

$$F = e - v + 1 \quad (1)$$

$$M = \frac{F}{2v-5} \quad (2)$$

Donde  $F$  es el número de caras de un grafo  $G$  y  $(v, e) \in (V, E)$ , entonces,  $M$  es la malla de  $G$  en función de  $F$ ,  $M = 0$  (cero), si la gráfica simple es un árbol tipo (autoorganizado). Cuando  $M = 1$ , el gráfico simple es un gráfico completo (planificado).

*Indicador Orgánico:* esta medida se utiliza para determinar si una red ha sido planificada o no, gracias al cálculo del número de vértices de grado 1 y grado 3. En nuestro contexto, el primero representa los callejones sin salida y el segundo, el número de intersecciones sin terminar. El indicador orgánico es representado por la ecuación:

$$O = \frac{V(1)+V(3)}{\sum_{j \neq 2} V(j)} \quad (3)$$

Donde,  $V(1)$  representa la suma de todos los vértices de grado 1,  $V(3)$  representa la suma de todos los vértices de grado 3 y  $V(j)$ , representa la suma de todos los vértices excepto los vértices de grado 2).

$$rv=(n2+n3)/(n1+n2+n3+n4)$$

$$v=len(G)$$

$$e=G.number\_of\_edges()$$

$$CI = e - v + 1$$

$$M = CI / (2 * v - 5)$$

```

rv = round(rv,2)
M = round(M,2)
print ('La Malla de Cuenca es : ', M)
print ('El Orgánico de Cuenca es : ', rv)

```

**Resultado del código:**

```

La Malla de Cuenca es: 0.622
El Orgánico de Cuenca es: 0.844

```

Según (Wang, 2015), estos datos significan que al analizar la red urbana de Cuenca, esta tiene un bajo grado de estructura urbanística de tipo Malla, pero tiene un porcentaje muy significativo de autoorganizada o jerárquica, típico de las ciudades con crecimiento acelerado.

#### 1.2.4 Centralidad de Intermediación (Betweenness Centrality)

La centralidad de intersección es una medida de centralidad en un grafo o red, basado en las trayectorias más cortas. La centralidad de intersección es una medida que calcula la importancia relativa de un vértice o borde en un grafo en accesibilidad. La centralidad de intersección de un vértice en un grafo  $G = (V, E)$  se define como (Wang, 2015):

$$B_i = \sum_{j \neq g \in G} \frac{B_{jg}(i)}{B_{jg}} \quad (4)$$

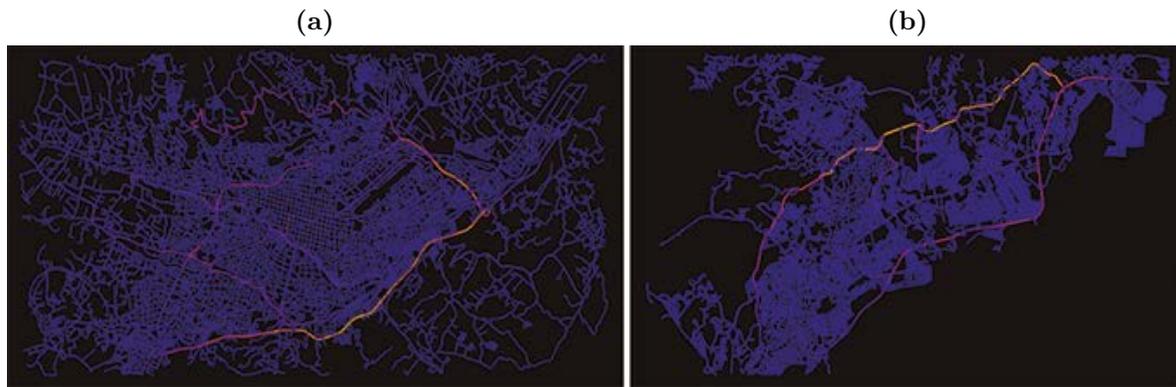
Donde,  $B_i$  es la centralidad de intermediación de un vértice o borde en accesibilidad.  $\beta_{jg}(i)$  es el número de las rutas más cortas entre los vértices o los bordes  $j$  y  $g$ , que pasan por el vértice o el borde  $i$ .  $\beta_{jg}$  es el número de caminos más cortos entre ellos. A continuación, se aplica la medida de centralidad de interconexión con *Networkx*, *OSMNX* y *Python*.

Aquí podemos calcular la  $BC$  de una red a partir de sus bordes o calles y visualizar las vías de mayor intermediación, es decir las calles propensas a mayor flujo y volumen de tráfico vehicular. Ver Figura 4.

```

edge_Betweenness = nx.betweenness_centrality(nx.line_graph(G2))

```



**Figura 4.** Betweenness Centrality BC, (a) ciudad de Cuenca, Ecuador y (b) Panamá, Panamá

En la Figura 4, visualizamos la red urbana de Cuenca (a) y Panamá (b) con las calles de mayor intermediación o de alto tráfico vehicular. Los bordes de color naranja más claros son los cruces de calle de mayor centralidad intermedia. El código para calcular la *BC* de las calles de redes urbanas se detalla a continuación:

```
import matplotlib.cm as cm
import matplotlib.colors as colors
import networkx as nx
import osmnx as ox
import pandas as pd
%matplotlib inline
ox.config(log_console=True, use_cache=True)
edge_Betweenness = nx.betweenness centrality(nx.line_graph(G2))
print('Muestra la Centralidad de Intermediacion de las calles de Panama')
ev = [edge_Betweenness[edge + (0,)] for edge in G2.edges()]
# color scale converted to list of colors for graph edges
norm = colors.Normalize(vmin=min(ev)*0.8, vmax=max(ev))
cmap = cm.ScalarMappable(norm=norm, cmap=cm.plasma)
ec = [cmap.to_rgba(cl) for cl in ev]
# color the edges in the original graph with closeness centralities in the line graph
fig, ax = ox.plot_graph(G2, bgcolor='k', axis_off=True, node_size=0, node_zorder=2,
    edge_color=ec, edge_linewidth=1.5, edge_alpha=1, fig_height=14, fig_width=11)
#La siguiente linea es para guardar el mapa en extensión tiff o png
fig.savefig("BC_Calles_PANAMA.tiff", dpi=400, format="tiff")
```

En la Figura 4 (b) mostramos el resultado de este código. Se aprecia el mapa de la ciudad de Panamá, formado por un anillo o vía principal; en el sector norte tiene mayor  $BC$ . Por lo tanto, el Corredor Norte del anillo vial tiene mayor carga de tráfico vehicular, que es el lado en el cual está creciendo la ciudad.

### 1.2.5 Centralidad Cercana (Closeness Centrality CC)

La medición de centralidad cercana  $C_i$  se efectúa a partir de la ecuación

$$C_i = \frac{V-1}{\sum_{i \neq v \in G} d_{iv}} \quad (5)$$

Para aplicar la medición de la  $CC$ , cargamos las redes urbanas de varias ciudades de Latinoamérica, entre ellas, ciudades de Ecuador. El código utilizado para el desarrollo de este análisis se muestra a continuación (generado en *Python* con librerías de *OpenStreetMap OSM* y *OSMnx*). El resultado de esta medición se puede observar en la Figura 5.

```
import osmnx as ox, networkx as nx, matplotlib.cm as cm, pandas as pd, numpy as np

ccen = nx.closeness centrality(G2)

df = pd.DataFrame(data=pd.Series(node_centrality).sort_values(), columns=['cc'])

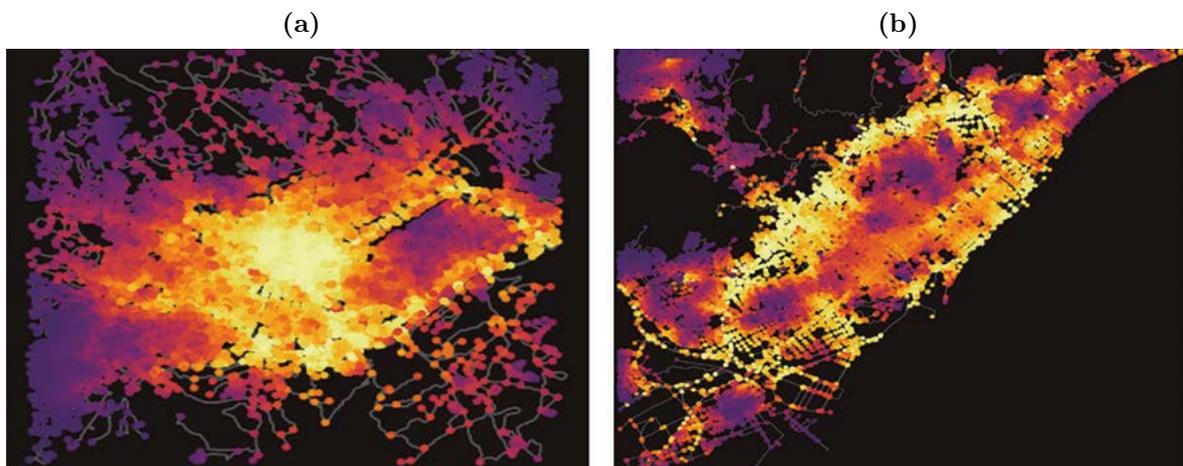
df['colors'] = ox.get_colors(n=len(df), cmap='inferno', start=0.2)

df = df.reindex(G.nodes())

nc = df['colors'].tolist()

fig, ax = ox.plot_graph(G, bgcolor='k', node_size=30, node_color=nc, node_edgecolor='none',
                        node_zorder=2, edge_color='#555555', edge_linewidth=1.5, edge_alpha=1)
```

#### Resultado del Código:



**Figura 5.** Closeness Centrality CC. (a) Cuenca-Ecuador, (b) Barcelona-España.

En la Figura 5, analizamos las intersecciones de calles, donde los nodos de color amarillo más claros son los cruces de calle de menor  $CC$ . Aquí mostramos que la centralidad de cercanía es posible calcular tanto para los nodos de la red (Figura 5), como también, para los enlaces o calles (Figura 6). Los colores amarillos con mayor intensidad representan los lugares que tienen el menor promedio de centralidad de cercanía (Boeing, 2017). La centralidad cercana es una buena aproximación para detectar los lugares de una red urbana proclives a congestión de tráfico vehicular y a alta concentración de personas o transeúntes, centros comerciales, centros médicos, entre otros. Esta información es importante para tomar decisiones de urbanización y políticas públicas sobre planificación urbana y organización del tránsito vehicular. Este es el código para calcular  $CC$ , considerando las calles o enlaces de la red urbana.

```
import osmnx as ox, networkx as nx, matplotlib.cm as cm, pandas as pd, numpy as np

edge_centrality = nx.closeness centrality(nx.line_graph(G))

ev = [edge_centrality[edge + (0,)] for edge in G.edges()]

norm = colors.Normalize(vmin=min(ev)*0.8, vmax=max(ev))

cmap = cm.ScalarMappable(norm=norm, cmap=cm.inferno)

ec = [cmap.to_rgba(cl) for cl in ev]

# Poner color en el borde del grafico original con closeness centralities

fig, ax = ox.plot_graph(G, bgcolor='k', axis_off=True, node_size=0,
node_color='w', node_edgecolor='gray', node_zorder=2, edge_color=ec,
edge_linewidth=1.5, edge_alpha=1)
```



**Figura 6.** Centralidad Cercana ( $CC$ ), de (a) Manta-Ecuador y (b) Caracas-Venezuela

En la Figura 6, presentamos el caso de Manta, cuyos bordes de color más claros son las calles de mayor  $CC$ . También se puede realizar una serie de medidas de centralidad como indica el siguiente código:

```

s=sum(pd.Series(node_centrality).sort_values())
Mx=max(pd.Series(node_centrality).sort_values())
m1=min(pd.Series(node_centrality).sort_values())
c=s/10730

print ("Promedio de centralidad cercana de Cuenca es: ", c)
print ("Máximo de centralidad cercana de Cuenca es: ", Mx)
print ("mínimo de centralidad cercana de Cuenca es: ", m1)

```

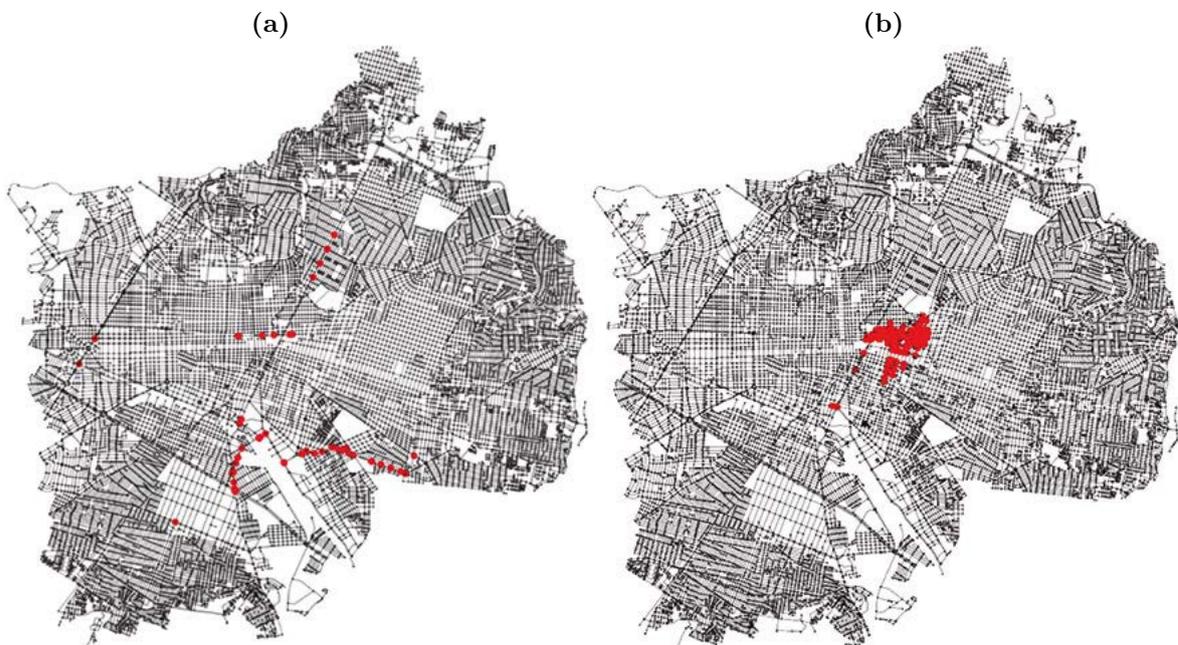
**Resultado del código:**

Promedio de centralidad cercana de Cuenca es: **0.0222269575644**

Máximo de centralidad cercana de Cuenca es: **0.0277354110549**

Mínimo de centralidad cercana de Cuenca es: **0.0143541950132**

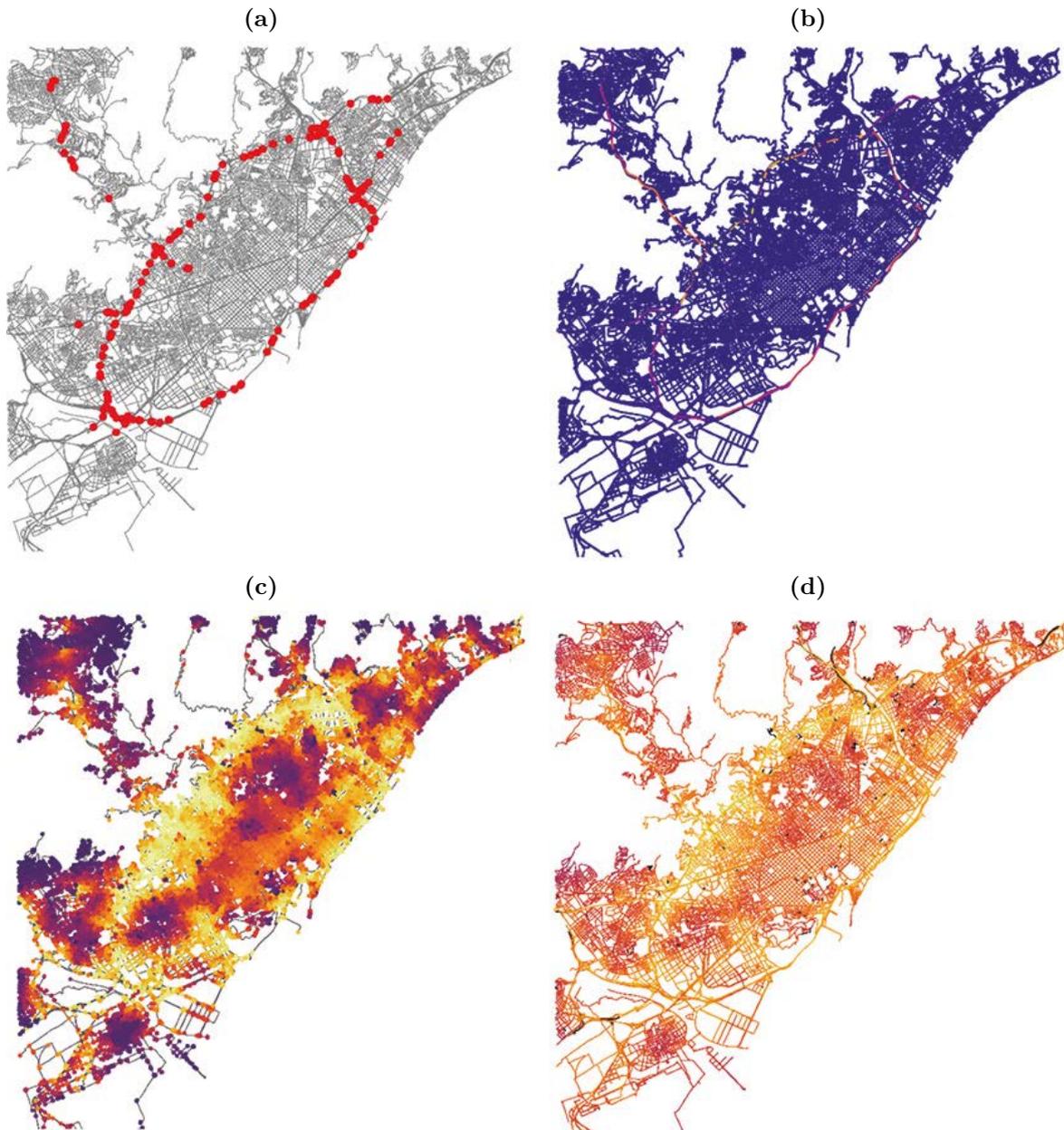
Otro ejemplo de centralidad es buscar los nodos (cruces de calle) de intermediación más altos, vulnerables a la congestión del tráfico vehicular. También, tenemos los nodos de menor  $CC$  que generalmente se aglomeran en el centro de la ciudad. Estos dos tipos de nodos son los que utilizan los viajeros con mayor frecuencia para dirigirse desde un nodo origen a otro de destino, en cualquier parte de la red. Ver Figura 7.



**Figura 7.** Ciudad de Guadalajara-México. (a) Los nodos BC y (b) Las intersecciones de CC.

En la Figura 7, describimos cómo se presenta el mapa completo con las dos mediciones  $BC$  y  $CC$  de centralidad. Primero, calculamos los nodos (intersecciones) y segundo, calculamos los bordes (calles o enlaces) de mayor  $CC$ . En otro caso, aplicamos el

estudio a la ciudad de Barcelona, en España (Ver Figura 8), y se demuestra que las medidas de centralidad son aplicables en cualquier ciudad del mundo. Los resultados de estos cálculos revelan que las grandes ciudades son aptas para aglomerar los sitios de mayor movilidad humana alrededor de sus principales anillos viales y en los centros de comercio y finanzas.



**Figura 8.** Muestra las mediciones de BC y CC de Barcelona-España.

Esta aglomeración, directa o indirectamente causa el aumento del flujo de transporte vehicular y genera embotellamiento y contaminación ambiental. Las mediciones de *BC* se localizan en las principales avenidas que conectan al resto de nodos de la red (Figura 8a y 8b). En cambio, en la *CC*, la tendencia es que los puntos de mayor aglomeración se ubiquen en el centro de las redes urbanas, tal como se muestra en la Figura 8c y 8d.

### 1.2.6 Elevaciones en mapas georreferenciales

Estas métricas permiten medir el grado de pendiente de los distintos nodos, ubicados en una red urbana, localizando el tipo de relieve de una ciudad. Generalmente podemos observar qué lugares son más altos o bajos, por ejemplo, en el caso de inundaciones. El código es detallado a continuación:

```
import networkx as nx

import numpy as np

import osmnx as ox

key='AIzaSyCaCxPkJlw8MSVGoVMzJeY3UQGkBoNVI5M'

ox.config(log_console=True, use_cache=True)

ox.__version__

place='Guayaquil'

G = ox.add_node_elevations(G, api_key=key)

G = ox.add_edge_grades(G)

edge_grades = [data['grade_abs'] for u, v, k,
data in ox.get_undirected(G).edges(keys=True, data=True)]

avg_grade = np.mean(edge_grades)

print('Average street grade in {} is {:.1f}%'.format(place, avg_grade*100))

med_grade = np.median(edge_grades)

print('Median street grade in {} is {:.1f}%'.format(place, med_grade*100))

G_proj = ox.project_graph(G)

print('Guayaquil')

print('Elevaciones: Nodos color amarillo intenso, Sectores altos de Guayaquil')

nc = ox.get_node_colors_by_attr(G_proj, 'elevation', cmap='plasma', num_bins=20)

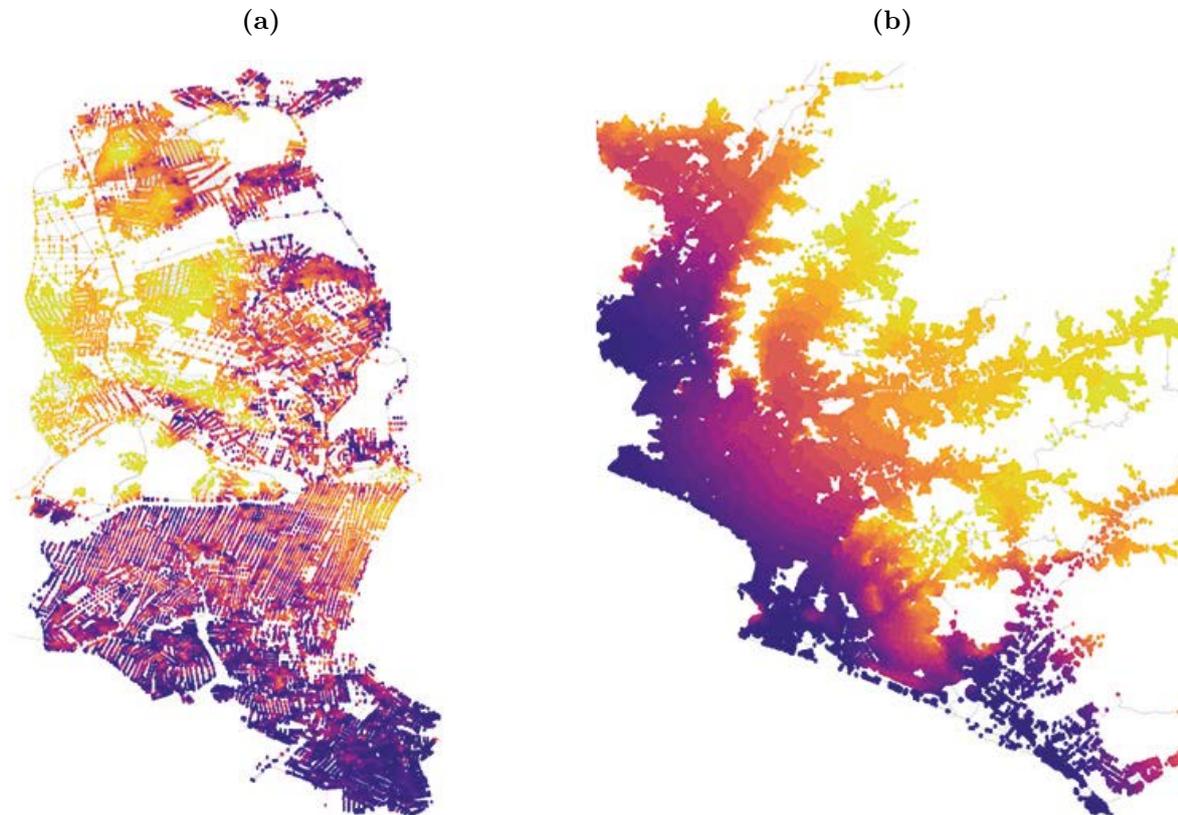
fig, ax = ox.plot_graph(G_proj, fig_height=16, fig_width=14,

node_color=nc, node_size=18, node_zorder=2, edge_color='#dddddd')

fig.savefig("CCT_Elev_Guayaquil.tiff", dpi=300, format="tiff")
```

Aquí observamos que, si la red tiene un grupo de nodos altos, esa zona es segura para evitar las inundaciones, pero vulnerable a deslaves. En cambio, si el grupo de nodos es bajo, esa zona es vulnerable a inundaciones o tsunamis (si está cerca al océano). Para esto, mostramos la red urbana de Guayaquil-Ecuador y Lima-Perú. En Guayaquil podemos identificar las zonas altas de la ciudad, ubicadas en el lado noroeste

marcado con nodos de color amarillo. En cambio, las zonas bajas se agrupan en el lado sureste de la ciudad, marcados con nodos de color púrpura oscuros. Para el caso de Lima, los nodos altos están al este (región montañosa) de la ciudad y los nodos bajos están al lado oeste junto al océano. Ver Figura 9.



**Figura 9.** (a) Red urbana de elevaciones de Guayaquil, (b) red urbana de elevaciones de Lima.

### 1.2.7 Mediciones de tiempos de viaje dentro de una red urbana

En las TCE encontramos aplicaciones que facilitan scripts para utilizar los dispositivos móviles de los usuarios como fuentes de información de tráfico. Por medio de estas aplicaciones, podemos proporcionar interfaces simples para alertar posibles situaciones caóticas de tráfico. Un caso específico es *WazeRouteCalculator*, un script de la *App* Waze propiedad de Google, sirve para calcular los tiempos y rutas más cortas de traslado. Para los cálculos utiliza el mapa de la red urbana y datos ponderados por los usuarios conductores. *Waze* evalúa y devuelve la información actualizada de las condiciones de la ruta actual (Barata et al., 2014). Además, permite comprender la dinámica de la movilidad de una ciudad y los patrones de comportamiento urbano. También, caracteriza las propiedades del tráfico vehicular e identifica sus condiciones aproximadas al tiempo real (Silva et al., 2013). A continuación, mostramos algunas pruebas realizadas para medir las distancias y tiempos de rutas entre un origen y un destino OD. Estos cálculos fueron realizados en distintas horas y días en varias ciudades del Ecuador.

Origen	Destino	Tiempo h,m viernes 15h00	Tiempo h,m martes 10h00	Distancia, Km
La Troncal	Guayaquil	1,26	1,25	80
El Triunfo	Guayaquil	1,07	1,06	66
Cuenca	Guayaquil	3,22	3,15	198.7
Cuenca	Loja	2,50	2,51	213.8
Cuenca*	Riobamba*	5,38*	4,13	345.7/253.3
Guayaquil	Machala	3,43	3,36	186.2
Guayaquil	Quito	7,37	6,54	432.4
Quito	Ibarra	2,21	1,35	112.6
Quito	Ambato	2,28	2,11	156.9
Quito	Santo Domingo	2,54	2,38	152

**Tabla 2.** Muestra las mediciones de tiempos realizadas en *Waze*.

En la tabla 2, observamos que trasladarse de una ciudad a otra en un vehículo toma distintos tiempos de duración, según la hora o los días de la semana. Esto significa que podemos planificar mejor nuestros viajes, para cambiar de día y horario, y de este modo prever el tiempo de viaje si se toma uno de estos escenarios.

Para otro caso de estudio, realizamos pruebas utilizando *Python*, donde calculamos tiempo y distancia entre dos coordenadas georreferenciales de la red urbana de Cuenca, Ecuador; luego, guardamos los resultados en un archivo *csv*.

```

import WazeRouteCalculator

import time

region = 'EU'

from_address = '-2.907060, -79.007466'

to_address = '-2.889703, -78.984568'

route = WazeRouteCalculator.WazeRouteCalculator(from_address, to_address, region,
log_lvl=None)

route_time, route_distance = route.calc_route_info()

h=int(route_time/60)

b = abs(route_time/60) - abs(int(route_time/60))

m=int(b*60)

print('Ruta: Estadio A. Serrano-Aeropuerto Cuenca-Ecuador ' ,from_address,'
Destino: ', to_address )

print('El tiempo estimado es: ', h, 'horas ', m, ' minutos')
```

```
print('La distancia estimada es: ', route_distance, 'Km')
archivo=open("Lista_Waze3.CSV","a")
archivo.write('R:Lima-Aeropuerto-PUCP')
archivo.write(",")
archivo.write(time.strftime("%H:%M:%S"))
archivo.write(",")
archivo.write(str(h))
archivo.write(",")
archivo.write(str(m))
archivo.write(",")
archivo.write(str(route_distance))
archivo.write(",")
archivo.write("\n")
print('Hora calculada>> ', time.strftime("%H:%M:%S"))
```

#### **Resultados luego de ejecutar el código**

Ruta: estadio A. Serrano-aeropuerto Cuenca-Ecuador

Origen: -2.907060, -79.007466

Destino: -2.889703, -78.984568

El tiempo estimado es: 0 horas 13 minutos

La distancia estimada es: 4.161 Km

Hora calculada>> 13:59:26

Código para extraer, analizar y visualizar la ruta más corta de la red evaluada:

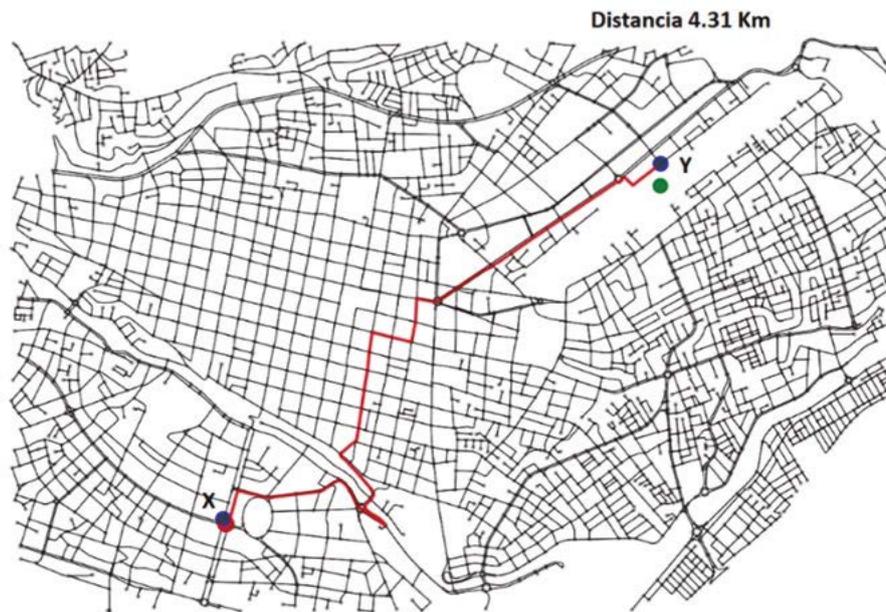
```
import networkx as nx
import numpy as np
import osmnx as ox
from sklearn.neighbors import KDTree
import folium
import matplotlib.pyplot as plt
#Extrae la red (Sector estadio-aeropuerto Cuenca)
import osmnx as ox
```

```

north, south, east, west = -2.881953, -2.911597, -78.972910, -79.018589
G2 = ox.graph_from_bbox(north, south, east, west, network_type='drive')
print('Ruta: Estadio-Aeropuerto *Sin Interruccion')
ini1=ox.geocode(from_address)
des1=ox.geocode(to_address)
nodes, _ = ox.graph_to_gdfs(G2)
nodes.head()
tree = KDTree(nodes[['y', 'x']], metric='euclidean')
ini1_idx = tree.query([ini1], k=1, return_distance=False)[0]
des1_idx = tree.query([des1], k=1, return_distance=False)[0]
closest_node_to_ini1 = nodes.iloc[ini1_idx].index.values[0]
closest_node_to_des1 = nodes.iloc[des1_idx].index.values[0]
route = nx.shortest_path(G2, closest_node_to_ini1, closest_node_to_des1)
fig, ax = ox.plot_graph_route(G2, route, fig_height=16,
                             fig_width=14,
                             show=False, close=False,
                             edge_color='black',
                             orig_dest_node_color='green',
                             route_color='red')
ax.scatter(ini1[1], ini1[0], c='red', s=250)
ax.scatter(des1[1], des1[0], c='green', s=200)
ax.scatter(G2.node[closest_node_to_ini1]['x'],
           G2.node[closest_node_to_ini1]['y'],
           c='blue', s=200)
ax.scatter(G2.node[closest_node_to_des1]['x'],
           G2.node[closest_node_to_des1]['y'], c='blue', s=200)
length=nx.shortest_path_length(G=G2,source=closest_node_to_ini1, target = clo-
sest_node_to_des1, weight='length')
print('Distancia >> ', round(length/1000,2), 'Km.')
fig.savefig("EstaioAeropuestoCuenca.tiff", dpi=400, format="tiff")

```

El resultado de este script se visualiza en la Figura 10, que demuestra una fracción de la red de Cuenca-Ecuador. Calculamos y mapeamos la ruta corta (más óptima) para encontrar la distancia y tiempo estimado entre  $X$  y  $Y$ . Esto es aplicable a cualquier red urbana grande, mediana o pequeña. En este ejemplo, se calcula la ruta para viajar desde el redondel de la Av. Fray Vicente Solano (sector el estadio) hasta la estación del aeropuerto Mariscal Lamar de la Av. España. El resultado es: El tiempo estimado **14 minutos**, distancia **4.31 km** y hora de medición **17:08**.



**Figura 10.** Cálculo de rutas óptimas, desde  $x$  a  $y$  y distancia 4.31 km., línea de color rojo.

Código para extraer, analizar y visualizar la ruta más corta de la red evaluada:

## CAPÍTULO II

### Tecnologías computacionales emergentes y gerencia



## 2.1 Combinación de Modelos Gerenciales y TCE

El objetivo de este apartado consiste en analizar cómo evolucionan los modelos de gestión dentro de las empresas y organizaciones, a la par con el desarrollo de la Tecnología Computacional, la inteligencia artificial y el aprendizaje automático, para la optimización de negocios exitosos en el entorno local y global.

Spender (2015) señala que, únicamente las empresas modernas que no se autogestionan tienen una gestión relevante. Por lo tanto, para ser sostenibles, las empresas modernas deben estar en permanente cambio, transformación tecnológica, creatividad e innovación. Es decir, deben generar investigación, ciencia y tecnología para ser competitivas y ofrecer mejor garantía de permanencia en el mercado satisfaciendo al cliente. Esta nueva filosofía de empresa y organización es respaldada por Uddina & Hossain (2015), quienes plantean desarrollar una transición de gestión del enfoque tradicional al enfoque científico, y luego, de la gestión científica a la fase moderna a través de la revolución tecnológica y la globalización (Cloud computing) (Dornelas *et al.*, 2017).

En este contexto, también abordamos el rol de los gerentes profesionales contemporáneos, quienes se encuentran (o deben encontrarse) a la vanguardia de las actividades empresariales. Estos gerentes están utilizando técnicas y modelos modernos para garantizar un alto desempeño y el monitoreo de sus metas e indicadores de gestión. Los gerentes, en la nueva era, no deberían enfocarse únicamente en el superávit de la empresa para satisfacer a los accionistas, sino también, deben enfocarse en la gestión social y el bienestar de sus empleados para lograr un desarrollo local y regional (Kim & Lim, 2017). Es precisamente allí donde el análisis de datos computacionales ayuda a detectar el comportamiento de todos los involucrados de una empresa. Sin embargo, el éxito no se conseguirá solo con el conocimiento de estos enfoques o en sus instrumentos; la clave está en el arte de enlazar, los datos, la información y el conocimiento. Para esto, es necesario combinar diferentes técnicas y métodos de inteligencia computacional para lograr resultados diferenciadores que exige la sociedad actual. Por ende, es importante medir el desempeño de sus involucrados dentro de las organizaciones por medio de indicadores. A continuación, describimos algunos modelos de gestión vanguardistas, enmarcados en la sostenibilidad empresarial y social.

### 2.1.1 Planificación empresarial estratégica basada en las Tecnologías de la Información

Las tecnologías de la información en la actualidad son un instrumento clave para las diversas actividades de la humanidad, en campos como la investigación, la universidad, la salud, las empresas y la sociedad en general (Boston College ITS, 2015). Por lo tanto, la planificación estratégica es un plan a futuro de la organización, que emplea tecnologías de la información como base para la gestión de sus procesos y el análisis de sus datos, y cuya información (generada) proporciona conocimiento para tomar decisiones (Rezaei *et al.*, 2016). Generalmente, todas las empresas por medio

de sus directivos y gerentes diseñan planes estratégicos cada periodo, sin embargo, en un estudio realizado en EE. UU. se revela que solo el 10% de los planes se ejecutan satisfactoriamente (Kaplan & Norton, The execution premium, 2008). Esto implica, que existe una gran brecha a superar en el deseo ideal de los gerentes con respecto a los objetivos y metas planteadas, los cuales deberán ser monitoreados de manera automatizada, lo mismo que las actividades, para verificar su cumplimiento satisfactorio, desde la alta gerencia y en los niveles operativos (Molina-Valencia *et al.*, 2014)).

### 2.1.2 Calidad Total con intermediación de las TIC

La calidad total es una alusión a la mejora continua que tiene el objeto de optimizar cada proceso de manera progresiva. Este es un concepto que explica cómo ofrecer el mayor grado de satisfacción a un cliente por medio de un bien o un servicio. En la actualidad, existe una variedad de estudios que relacionan la tecnología informática y la calidad total para mejorar el desempeño de las empresas (khanam *et al.*, 2013). Sin embargo, a decir de algunos autores, hace falta investigar cómo se relacionan las TIC con la Gestión de la Calidad Total GCT (Martínez-Lorente *et al.*, 2004).

Según (Koenes & Soriano, 1996), la calidad total o simplemente la calidad puede ser definida de varias formas, debido a que, para medir la calidad de algún servicio o producto ofertado, son varios los indicadores, métricas o dimensiones, cualitativas o cuantitativas, que califican la condición de calidad. Por tal motivo, el término calidad total, frecuentemente, es concebido como una utopía que todos aspiran alcanzar o tener.

Según el esquema de Schnaars, la calidad se puede definir a partir de cuatro puntos de vista y ocho dimensiones. Ver Tabla 3.

Cuatro puntos de vista	Ocho dimensiones
Algunas personas consideran que un producto/servicio de calidad es el que funciona eficazmente.	Fiabilidad Durabilidad Conformidad con las especificaciones
Algunas personas consideran que un producto/servicio de calidad es el que tiene un diseño excelente.	Funcionalidad (performance) Características externas Capacidad para prestar un servicio Aspectos estéticos
Algunas personas consideran que un producto/servicio de calidad es el que logra un nivel superior de satisfacción de las necesidades y deseos de los consumidores, usuarios o clientes.	Calidad percibida
Algunas personas consideran que un producto/servicio de calidad es el que alcanza el más alto nivel posible de excelencia alcanzable.	

**Tabla 3.** Criterios de calidad según *Schnaars*.

**Fuente:** (Koenes & Soriano, 1996)

Para determinar estas métricas es preciso usar las aplicaciones informáticas, pues permiten que la información oportuna del rendimiento sea visualizada en menor tiempo. Las Tecnologías Informáticas detectan mejor qué procesos funcionan bien y cuáles fallan o deben ser ajustados, usando los datos e información disponibles. Además, constituye un factor fundamental para mejorar el desempeño del gobierno en la planificación estratégica, operativa, seguimiento y evaluación organizacional.

### 2.1.3 Kaizen o automatización

Según Thomas Friedman (2016), “la globalización y la digitalización son dos mega tendencias que aplanaron el mundo” (p. 57), desarrollando negocios de tecnología digital, información y comunicación. Esto ha impulsado negocios tecnológicos en países desarrollados como EE. UU. Japón, Alemania, Corea del Sur, entre otros (LEE et al., 2010) Por otra parte, la convergencia de la tecnología y la fertilización de las ideas cruzadas ayuda a desarrollar destrezas en un campo y aplicarlas en otro. Esto, en la actualidad, ha significado la cura de viejos males con ideas y conocimientos frescos. Kaizen es lo opuesto a la complacencia. Se trata de una filosofía de gestión creada por el japonés Massaki Imai. Consiste en un sistema enfocado en la mejora continua de toda la empresa y sus componentes, de manera armónica y proactiva (Molina-Valencia *et al.*, 2014).

Kaizen y la tecnología de la información son dos perspectivas para mejorar el rendimiento de las empresas e industrias. Existen estudios para integrar kaizen y automatización en proyectos de reingenierías de procesos (Lyu-Jr., 1996). Desde este propósito, se sugiere utilizar técnicas de simulación animada, técnicas de caja blanca en vez de negra y procesos proactivos en lugar de reactivos, para simplificar los procesos de fabricación en un 50% (Katayama, 2017). Sin embargo, si no se tienen las herramientas apropiadas para medir de forma confiable el desempeño de sus colaboradores cabe preguntarse ¿cómo lograr esto?

A continuación, se listan varios pasos para el desarrollo del método Kaizen. Ver Figura 11.

- Mostrar resultados significativos a partir de cambios pequeños acumulados en el tiempo.
- Kaizen actúa cuando existe la participación de todos, enfocados en la calidad.
- Si el cambio no es grande, el impacto no será significativo (Universitat Barcelona, 2014).



**Figura 11.** Método para aplicar kaizen en la práctica.

Los beneficios de emplear el método Kaizen anclado a la tecnología son varios, citamos algunos:

1. Mejora de los tiempos de entrega
2. Incremento de los niveles de satisfacción
3. Mayor grado de compromiso en el equipo
4. Mejores tasas de retención del talento
5. Impulso a la competitividad
6. Mejoramiento en la relación con los proveedores
7. Optimización de la resolución de problemas y búsqueda de soluciones
8. Fortalecimiento de vínculos para resolver cualquier desafío

#### **2.1.4 Justo a tiempo en la era digitalizada usando TIC**

El campo de mayor desarrollo y mejora es la optimización del tiempo en las empresas e industrias, gracias a la automatización de los procesos. Esto permite mejorar los tiempos de respuesta y entrega de productos y servicios, en áreas tales como contabilidad, producción, compras, ventas, entregas, cadenas de suministro, entre otras (Xu & Chen, 2016). A pesar de esto, existen inconvenientes con la entrega oportuna de la información generada por la cantidad de dispositivos actuales. Por lo tanto, esta falencia es remediada con el Internet de las Cosas IoT, que permite mejorar la fabricación, detección de fallos y la asistencia técnica en tiempo real.

El método justo a tiempo (*traducción del inglés Just in Time*) es un sistema de organización de la producción de fábricas de origen japonés. Esta metodología tiene como objetivo la creación de lugares de trabajo más organizados, ordenados, limpios y seguros, es decir, un lugar de trabajo en el que cualquiera estaría orgulloso de trabajar. Este método es muy eficiente para proyectos de software o de sistemas de información en general, puesto que, por un lado, ayuda a optimizar el tiempo invertido en el desarrollo de cualquier aplicación, reutilizar recursos de código y módulos de desarrollo para mejorar constantemente el producto; y, por otro lado, este método permite aprovechar mejor el desempeño del equipo de trabajo, reduciendo así el costo final del producto terminado.

#### **2.1.5 Reingeniería: El Modelado y análisis de procesos de negocios**

“La reingeniería de procesos de negocios (*BPR, Business Process Reengineering*) es una herramienta para ayudar a las organizaciones a mejorar la calidad, los servicios al cliente, reducir los costos operativos y convertirse en líderes en su dominio” (Bhaskar & Singh, 2014). Sin embargo, en la actualidad es prioritario contemplar la implementación de servicios de TIC en todas las dependencias de la nueva organización sometida a este proceso. Attaran (2004) revisa una serie de compañías que han implementado TI en su reingeniería.

Para Gunasekaran *et al.* (2014), es importante vincular los procedimientos de producción y los servicios de la organización con las metas y los objetivos empresariales. Para implementar los modelos es preciso determinar, el análisis, las herramientas y técnicas que ayuden a innovar el negocio. El estudio de Mlay *et al.* (2013) mostró que solo el 30.4% de los proyectos de reingeniería ha entregado sistemas de información utilizables; esto significa que los sistemas de información actuales no son reutilizables en su mayoría, por lo que es preciso desarrollar aplicaciones con nuevos paradigmas tales como aplicaciones de análisis e inteligencia de negocios que manejen grandes volúmenes de datos con variedad y velocidad propias de las TCE.

### **2.1.6 Análisis de Benchmarking con aplicaciones TCE**

En la actualidad, este instrumento empresarial está vinculado estrechamente con las tecnologías computacionales que permiten la comparación del rendimiento de los procesos utilizando el análisis de sus datos. Esta técnica permite obtener información proactiva para perfeccionar las operaciones y actividades internas de las empresas en sus distintos departamentos y niveles (Le *et al.*, 2015).

### **2.1.7 Empoderamiento (Empowerment)**

El empoderamiento es un factor clave de las organizaciones que debe ser estudiado de manera permanente por investigadores y empresarios, puesto que la sociedad y las tecnologías están en constante evolución (Elnaga & Imran, 2014). Cuestión similar describe Ibrahim (2015), para quien el empoderamiento está enfocado en los empleados mediante el suministro de productos y servicios, y la generación de valor para los clientes con el uso de tecnologías inteligentes. Este empoderamiento tiene éxito si se ejecuta desde una mentalidad adecuada, por parte de gerentes y empleados, que motive y proporcione capacitaciones y actualización de conocimientos de distintas maneras, especialmente tecnológicos. Además, el empoderamiento es un factor que impulsa a las empresas a generar calidad de producción y competitividad en el mercado a través de los servicios proporcionados por las TCE y la Web. Por ejemplo, comunicación permanente a través de las redes sociales, geolocalización, aprendizaje en línea, entre otros, mejoran la relación dentro de las organizaciones (Felhi & Akai-chi, 2012) y (Anshari & Almunawar, 2012).

### **2.1.8 La externalización (Outsourcing) de las TCE como estrategia**

Las empresas modernas usan estrategias para mejorar su competitividad y optimizar sus recursos, específicamente en la cadena de suministros (Gunasekaran *et al.*, 2014). contratando servicios de TI y TCE, desde los más pequeños hasta una gama completa de sistemas integrales (Varajãoa *et al.*, 2017). Sin embargo, es recomendable

evaluar cuidadosamente qué servicios son los idóneos para el beneficio económico y de rendimiento de la empresa. Resulta importante revisar las distintas teorías relacionadas con la externalización, aspecto que amplía (Pankowska, 2019). En este trabajo, se revisa teorías de costos, agencia, recursos, actividades, alianzas y partes interesadas. La mayoría de aportes en este campo está relacionada con las tareas colaborativas que buscan nuevos modelos de negocios y la aplicación de la economía de blockchain (uso de moneda electrónica).

## **2.2 Fusión de las empresas con la Tecnología**

La idea de fusionar y alinear las empresas con la tecnología para formar organizaciones robustas y eficientes mediante procesos de acoplamiento simultáneo constituye un reto para los gerentes modernos (Smaczny, 2001), tal es el caso de las empresas europeas y de África Subsahariana, que han crecido en tecnología e innovación en las últimas décadas (An IDC White Paper, 2016). Tomando en cuenta lo anterior, este libro propone aplicar las mismas estrategias en América Latina, a través de un esfuerzo entre académicos, empresarios e industriales. Los resultados de varias investigaciones revelan que las empresas exitosas han logrado fusionar las tecnologías en la nube, redes sociales, móviles, inteligencia de negocios empresariales y el big data, con logros significativos expresados en nuevas propuestas para hacer negocios. En la actualidad, gracias al impulso de la investigación, se han mejorado los índices de eficiencia en la alineación de los procesos empresariales a las bondades de las tecnologías de la información y comunicación.

Otra corriente importante de investigación propone, combinar las empresas y la tecnología enfocadas en la gestión de tecnologías verdes, para garantizar un marco de administración sostenible con medidas ambientales (Erek, 2019). Además, actualmente, el 82% de empresarios y gerentes ve a las TCE como conductoras de ventajas competitivas y facilitadoras de la eficiencia empresarial, y solo el 18% lo mira como un gasto necesario. Sin embargo, los resultados de toda fusión generan una gran cantidad de datos e información difíciles de entender; por ese motivo, se debe seguir evaluando las ventajas y desventajas generadas por estos procesos a través de las TCE.

### **2.2.1 Cuadro de mando integral**

El objetivo de este módulo es lograr que el lector comprenda el valor que tiene el cuadro de mando integral (del inglés Balanced Scorecard BSC) como técnica de transición entre los procesos del negocio y las Tecnologías Emergentes, para superar los límites de rendimiento de una empresa.

En An IDC White Paper (2016) se propone priorizar las siguientes acciones para avanzar con la fusión y la alineación de los negocios de forma sostenible.

1. Evaluar las necesidades y los objetivos del negocio en relación con el alcance, la velocidad y la escala de transformación digital.
2. Identificar las características organizativas (cultura, colaboración, liderazgo, resistencia al cambio, etc.) que pueden influir en la selección de estructuras organizativas de transformación digital.
3. Seleccionar e implementar un enfoque organizacional que se ajuste mejor a las necesidades inmediatas y a la retroalimentación, basándose en los resultados.
4. Adoptar un enfoque de trampolín para el cambio organizacional con el propósito de aumentar la amplitud y alcance de la empresa digital y su próxima transformación.

En este apartado ampliamos el análisis del Balanced Scorecard como un instrumento de fácil aplicación para la fusión y alineamiento entre empresa y tecnología.

El *Balanced Scorecard* (BSC / Cuadro de mando integral) es una herramienta que permite enlazar estrategias y objetivos claves con desempeño y resultados a través de cuatro áreas críticas, en cualquier empresa: desempeño financiero, conocimiento del cliente, procesos internos de negocio y aprendizaje & crecimiento.

Para (Kaplan & Norton, 2005), el cuadro de mando integral complementa indicadores de medición de los resultados de las actuaciones (acciones) con los indicadores financieros y no financieros; esto, como factores clave que influirán en los resultados del futuro, derivados de la visión de la empresa, puesto que, generalmente, resulta preocupante convertir la visión y la estrategia en objetivos e indicadores estratégicos. A partir de esto, se derivan las cuatro perspectivas básicas de un BSC que son: financiera, clientes, procesos operativos internos y aprendizaje & crecimiento.

En las nuevas empresas (*Start ups*), el *BSC* sigue siendo un instrumento clave para lograr la misión, visión y objetivos institucionales, en aquellas empresas que están centradas en la estrategia e inteligencia de negocios, las cuales luchan por identificar los factores de éxito (Azeroual & Theel, 2018) (Gold, 2019). El *BSC* está siendo utilizado inclusive en empresas sociales sin fines de lucro, para mejorar la calidad del servicio y la sostenibilidad (Sudaryo & Purnamasari, 2018).

### **2.2.2 Aportes o beneficios del BSC.**

Uno de los aportes del *BSC* a la cadena de valor de las empresas es desarrollar empleados de élite, capacitados en todas las áreas, con habilidades para entender y empoderarse rápidamente de los objetivos y la visión de la empresa; eliminando esas prerrogativas que definían restricciones a nivel estratégico y ejecutivo. Lo que se espera con esto es lograr que todos los empleados aporten valor agregado, gracias a lo que saben y a la información que proporcionan. Por eso, hoy se debe invertir en el conocimiento de los empleados debido a que es un factor clave de éxito. Sin embargo,

no siempre funciona todo a la perfección, sobre todo, si no se dispone de un modelo bien determinado anclado a las TCE, las cuales miden el desempeño de cada involucrado (empleados, clientes, proveedores, mandos medios, ejecutivos, accionistas), cuyo valor debe articularse a todas las áreas de la organización.

Además, el *BSC* ayuda a ligar los indicadores históricos con los indicadores futuros de cada una de las perceptivas, lo cual proporciona información predictiva con mayor facilidad. Del mismo modo, se disminuye la incertidumbre y a partir de ello, se facilita la toma de decisiones y el fortalecimiento de las unidades del negocio que presenten más vulnerabilidades. Se recomienda fortalecimiento del personal, actualización y modernización de la tecnología, optimizar los recursos físicos y corregir los procesos de producción que proporcionen mayor satisfacción a sus clientes y directivos.

Otra ayuda que brinda el *BSC* en las organizaciones consiste en equilibrar los indicadores estratégicos en cada nivel de la empresa, asegurando la integridad homogénea en el desempeño final y por consiguiente en los resultados. El *BSC* se adecúa satisfactoriamente con las recomendaciones y las buenas prácticas planteadas en COBIT 5.0., que garantiza la seguridad, fiabilidad y satisfacción de clientes y usuarios de los recursos computacionales.

### **2.2.3 Estructura del *BSC*.**

El cuadro de mando integral proporciona una estructura para transformar una estrategia en términos, hechos o acciones operativas que se alcen con los perfiles tecnológicos. Este aspecto es ampliado por Kaplan (2010). Según Kaplan y Norton (1993), el *BSC* inicia cuando los directivos de la empresa deciden trabajar como un equipo y se enfocan en convertir las estrategias claves del negocio en objetivos estratégicos específicos. El *BSC* arranca operacionalizando los ejes estratégicos sobre los cuales se guiará la empresa para generar competitividad. Luego, define los objetivos específicos sobre los que se establecen los indicadores de medición. Y, finalmente, genera las iniciativas o acciones inducidas a ejecutar cada estrategia.

### **2.2.4 Mapa y matriz estratégicos del *BSC*.**

En esta etapa, describimos de forma práctica, didáctica e ilustrativa, cómo se conforman los componentes estratégicos dentro del *BSC*, operacionalizando la matriz alineada al mapa estratégico. Para ello, construimos el mapa estratégico con las cuatro perspectivas principales (aunque a veces pueden ser más). Dentro de ellas, formulamos los objetivos estratégicos, conectados jerárquicamente de abajo hacia arriba, puesto que los objetivos inferiores sirven para dar cumplimiento a los de las perspectivas superiores. Esto se puede apreciar mejor en la Figura 12.

Una vez finalizada la elaboración del mapa estratégico, sigue el diseño de la matriz

estratégica. Para ello, se toma cada objetivo del mapa y se operacionaliza tal como se describe en la Figura 12. Para elaborar estos instrumentos se debe tener claros ciertos conceptos tales como estrategias, objetivos estratégicos, indicadores claves de desempeño (del inglés, Key performance indicators KPI), metas, iniciativas, involucrados o responsables (opcional).

En el siguiente ejemplo se muestra cómo se procede con la operacionalización de la estrategia desde la definición de la visión institucional. La visión es la luz al final de túnel o el estado ideal a donde pretende llegar la empresa a mediano o largo plazo.

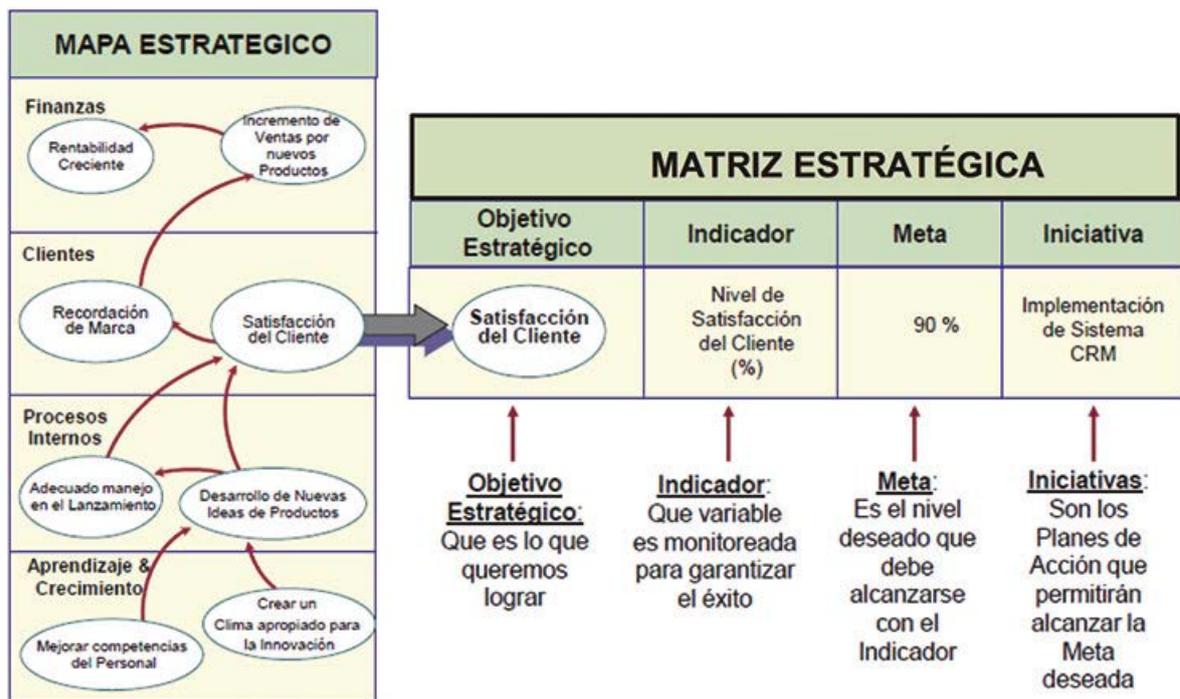


Figura 12. Implementación del mapa y matriz estratégica del BSC.

Después, se definen las perspectivas adecuadas para el negocio que, en la mayoría de casos, son finanzas, clientes, procesos internos y aprendizaje. Precisamente, en las dos perspectivas más bajas intervienen las TCE fortaleciendo las dos perspectivas superiores, en orden jerárquico descendente, donde cada una debe poseer un indicador relacionado con las otras perspectivas. Generalmente, estos indicadores son calculados y se visualizan de manera elocuente usando las tecnologías emergentes, ya sea en tableros de control (*del inglés, dashboards*), diagramas o gráficos estadísticos, como apreciamos en la Figura 13.

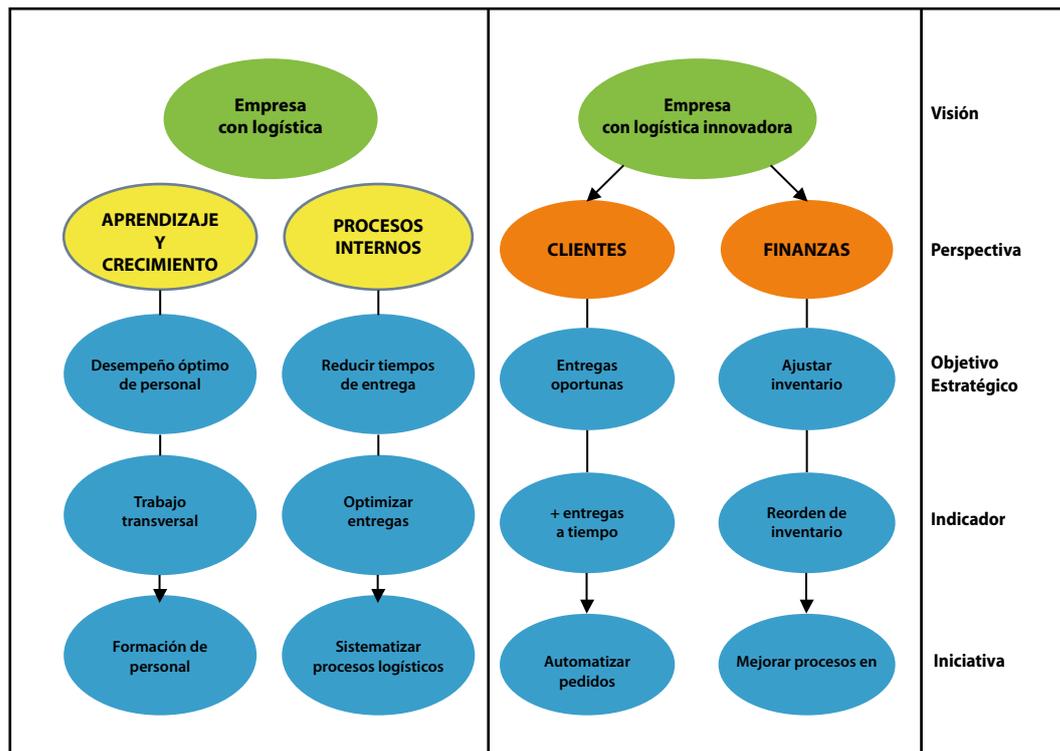


Figura 13. Ejemplo de Operacionalización de la estrategia.

Los objetivos estratégicos se deben formular y acoplar recíprocamente, para seguir la cadena de valor de la organización, de tal forma que, el cumplimiento o los resultados de los objetivos y las perspectivas inferiores den soporte a los objetivos o perspectivas inmediatas en el nivel superior hasta alcanzar la visión, tal como se aprecia en la Figura 14.

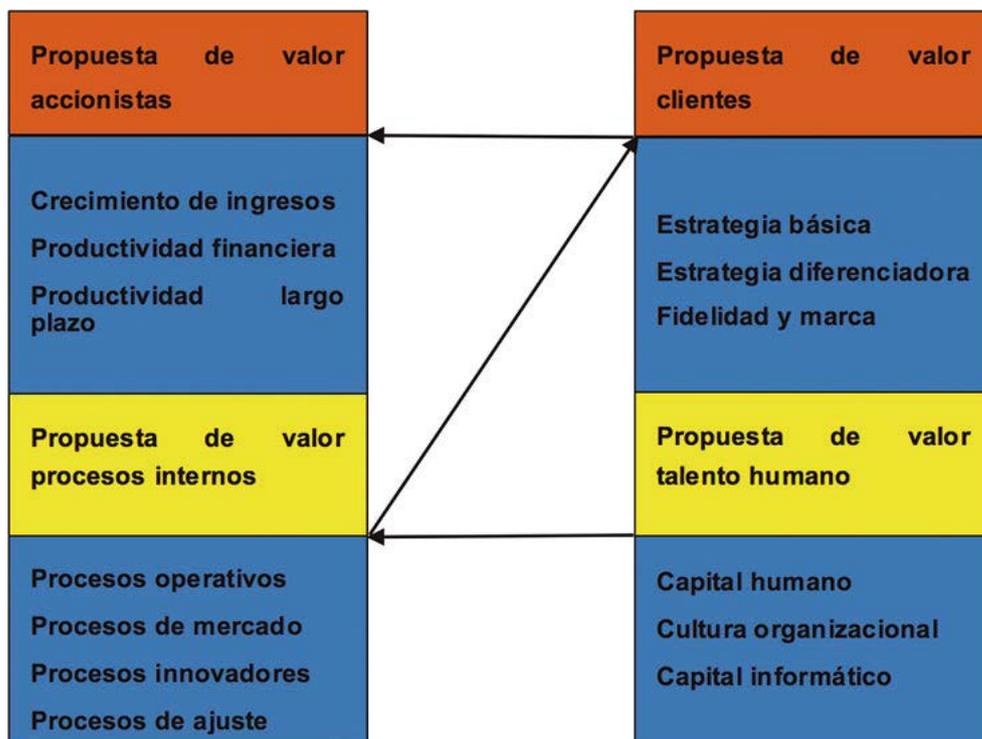


Figura 14. Interpretación de las perspectivas del BSC.

En resumen, las perspectivas de proceso y de aprendizaje sirven para cimentar las bases necesarias del andamiaje estratégico redistribuido en el *BSC* y para alcanzar las metas y cambios de mayor relevancia. Los resultados esperados están orientados a mejorar la fidelidad y satisfacción de clientes, el bienestar social y el mayor rendimiento económico. Es por este motivo, primero se deben afianzar las bases mediante un diseño y un trabajo intenso que facilite una proyección de optimismo y logre generar una cadena de valor sostenible para la organización. Ver la Figura 15.



Figura 15. Método para diseñar las perspectivas del BSC.

Visto así, el modelo para diseñar el BSC es de fácil implementación y monitoreo a través de TCE. Por ejemplo, herramientas de inteligencia de negocios, minería o analítica de datos, big data, entre otras, según los datos masivos a procesar.

### 2.2.5 Elementos principales del BSC.

Para las mediciones del rendimiento de los procesos y estrategias, enmarcadas dentro de las perspectivas del BSC, se cuenta con los siguientes elementos principales:

- a) Indicador: es un instrumento para monitorear, predecir y administrar el desempeño necesario para evaluar y alcanzar una meta determinada a futuro.
- b) Meta: es un valor numérico que se desea alcanzar en un tiempo determinado aplicado a un medidor.
- c) Umbral: es un rango de desempeño preestablecido y expresado en porcentajes, que define el comportamiento, el estado y las tendencias de los semáforos del indicador para los distintos escenarios (Ramírez-Granados *et al.*, 2014).

Es recomendable establecer los valores umbral de los indicadores en la configuración de los diseñadores de los KPIS. El umbral también es un valor que se usa para determinar cuándo cambiar el indicador que representa el estado del KPI. Una banda es el intervalo de valores entre dos umbrales, la colección de umbrales y bandas de un *KPI*, se denomina patrón de resultados.

#### ¿Por qué son importantes los valores de umbral?

Asignar los valores iniciales detenidamente es una operación muy significativa, puesto que influyen en dos elementos importantes del cuadro de mando (procesos y

estrategias), facilitan la corrección de errores y permiten realizar ajustes desde las interfaces del usuario, según los cambios o comportamientos del negocio. También, contribuyen a transformar la presentación de procesos complejos en indicadores simples de entender y a canalizar un interfuncionamiento satisfactorio (O'Connor, 2019).

- **Selección de indicadores:** un indicador *KPI* bien identificado muestra la proximidad de las medidas del rendimiento real a los valores del objetivo.
- **Valor de los resultados:** para definir los valores de los resultados que arrojen los *KPIs*, se debe considerar que estos tengan significado para hacer un análisis estratégico y tomar buenas decisiones.

### Caso de estudio

La comprensión del *BSC* es clave para motivar su implementación, razón por la cual, en este libro describimos un ejemplo completo que consiste en definir, identificar, ordenar y elaborar los principales elementos que tiene un *BSC*. Este caso de estudio plantea hacer un análisis de negocios inteligente, que permita mantener o mejorar el rendimiento continuo de la organización y dar cumplimiento a los objetivos institucionales planteados en el plan estratégico, tal como recomienda Verbert *et al.* (2013). Ver Tabla 4.

Indicador, Meta, Umbral-Banda			
<b>Indicador</b>	Tasa de conversión (compras de clientes)		
<b>Meta</b>	50% (número de compras/número de visitas)		
<b>Umbral-Banda</b>	>=50%	<50% y >=25%	<25%
<b>Semáforo</b>			
<b>Color</b>	verde	amarillo	rojo
<b>Significado</b>	aceptable	alerta	inaceptable
<b>Riesgo</b>	<b>Bajo</b>	<b>Mediano</b>	<b>Alto</b>
<b>Toma de decisión</b>	mantener, permanecer	mejorar, monitorear, evaluar, ajustar	innovar, cambiar, crear, decisiones categóricas de mejoramiento o cambios de estrategias y objetivos

**Tabla 4.** Ilustración del diseño de un KPI completo dentro del BSC

**Fuente.** El autor

### 2.2.6 Los indicadores claves de desempeño (*key performance indicators KPI*).

Para Jituri *et al.* (2018) es valioso saber identificar los *KPI* adecuados que satisfagan el desempeño de una empresa. Los *KPI* definen valores que nos explican en qué rango óptimo de rendimiento nos deberíamos situar para alcanzar los objetivos. Los *KPI* son mediciones cuantificables, acordadas de antemano, que reflejan los factores críticos de éxito para caracterizar y evaluar una organización (Ushveridze, 2017), además, en la actualidad se usan para manejar grandes cantidades de datos y empresas nuevas (Azeroual & Theel, 2018). Para tener una mejor lectura del rendimiento de la empresa, los *KPI* se deben clasificar en grupos diferentes dependiendo de la organización. Este es un concepto que está apegado a la realidad y es por esa razón que lo describimos en este libro. Por ejemplo, un *KPI* puede ser el porcentaje de ventas provenientes de los clientes, la tasa de graduados de los estudiantes universitarios, la cantidad de reservaciones por mes, la cantidad de clientes dados de baja en un año, el producto más vendido, por citar algunos.

Los *KPI's* son el eslabón entre la empresa y las aplicaciones de inteligencia de negocios, porque son la base para armar los tableros de control. Se diseñan en la etapa de *Analysis Services* pues están vinculados estrechamente con los datos y manipulados por gestores como *SQL Server*, *Oracle*, o cualquier otra herramienta de *BI*. Ver Figura 16. Un *KPI* es una fuente confiable y fidedigna para tomar una decisión basada en información, en tanto corrige o mejora el rendimiento de una entidad con el propósito de cumplir los objetivos institucionales y la visión. Por consiguiente, los *KPI'S* deberían ser regulados y evaluados según la evolución o cambios que sufra la organización.

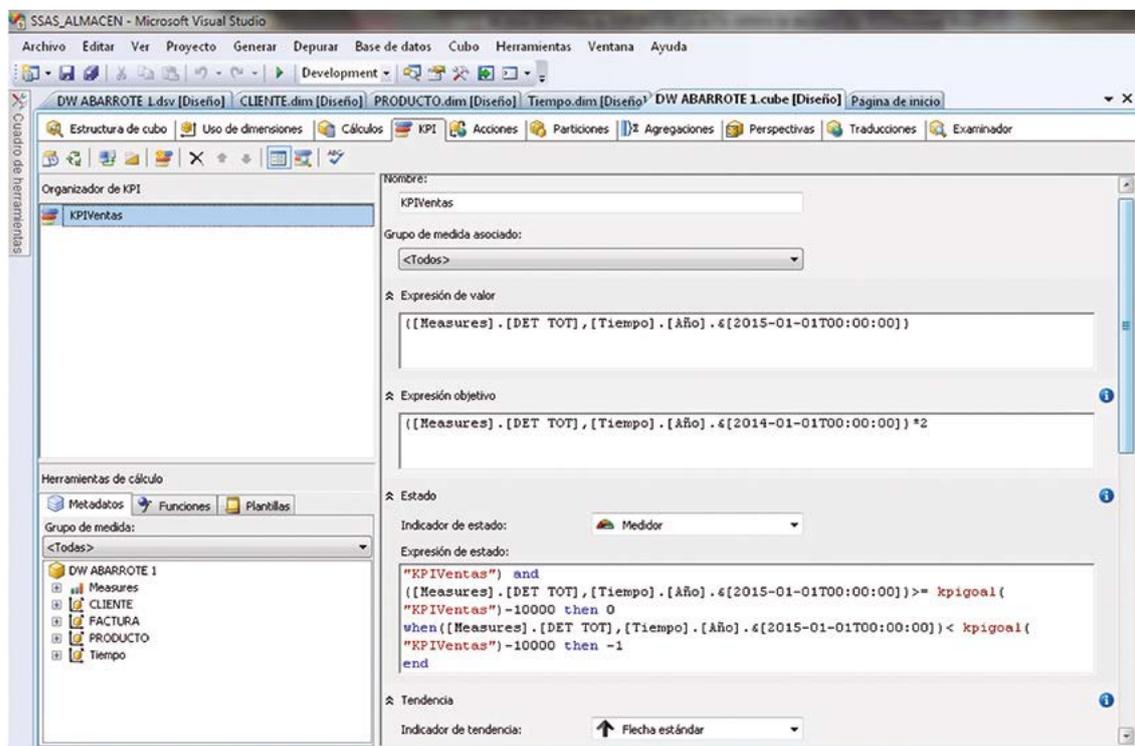


Figura 16. Interfaz de *SQL Server* para el diseño de un *KPIVentas* en *Analysis Services*.

### 2.2.7 Elementos claves del KPI

Dentro de la herramienta de Business Intelligence de *SQL Server de Microsoft, Power BI, Tableau, Oracle, SAP BI*, entre otras, intervienen en la etapa de *Analysis Services* (fundamento de los ejemplos de este libro). A continuación, se detallan los elementos principales para crear un *KPI*.

- 1. Nombre:** para crear un KPI, lo primero es asignar un nombre que servirá más adelante como referencia en los distintos valores de configuración del KPI.
- 2. Grupo de medida asociado:** los KPIS se pueden asociar con todos los grupos de medida o con un solo grupo.
- 3. Expresión de valor:** Es una medida física como Sales (ventas), una medida calculada como Profit (ganancia), o un cálculo que se define dentro del KPI mediante una expresión de Expresiones Multidimensionales (MDX).
- 4. Objetivo:** una expresión objetivo es un valor, o una expresión MDX que se resuelve en un valor, que define el objetivo de la medida definida por la expresión de valor. Por ejemplo, una expresión objetivo podría ser la cantidad en la que los responsables de una compañía desean incrementar las ventas o el beneficio [1].
- 5. El estado:** una expresión de estado es un MDX que *Analysis Services* utiliza para evaluar el estado actual de la expresión de valor en comparación con la expresión objetivo. Este es un valor normalizado en el rango de -1 a + 1, donde -1 es muy malo y + 1 es muy bueno. La expresión de estado presenta un gráfico para ayudar a determinar fácilmente el estado de la expresión de valor en comparación con la expresión objetivo.
- 6. La tendencia:** una expresión de tendencia es una expresión MDX que *Analysis Services* utiliza para evaluar la tendencia actual de la expresión de valor en comparación con la expresión objetivo. La tendencia ayuda al usuario corporativo a comprobar ágilmente si la expresión de valor está mejorando o decayendo en relación con la expresión objetivo. Puede asociar uno de los diversos gráficos con la expresión de tendencia para ayudar a los usuarios corporativos a comprender dicho comportamiento.

### 2.2.8 Técnicas para identificar los KPI's.

Los *Key Performance Indicators (KPI's)* se traducen literalmente al español como Indicadores Claves de Desempeño o Indicadores Claves de Rendimiento. Son medidas que permiten visualizar con claridad el desempeño de nuestras acciones o actividades para lograr los objetivos o las metas planteadas. Un *KPI*, es fundamental para monitorear el cumplimiento de cada actividad, que en conjunto llevará al éxito de una organización (Somalo, 2017). Un KPI bien ejecutado y monitoreado genera valor agregado para una empresa, por lo que, resulta relevante utilizarlos y aplicarlos dentro de un tablero de control (*dashboard*).

---

<sup>1</sup> Algoritmos de minería de datos (*Analysis Services: Minería de datos*) Microsoft (2018)

Lo primero que debemos saber para crear los *KPIS*, es conocer el plan estratégico de la corporación para la cual vamos a desarrollar los indicadores. Dicho de otro modo, se debe conocer bien el negocio, su modelo, los objetivos, las estrategias, los alcances, la actividad del negocio, la ubicación, visión, entre otras características y propiedades claves para mejorar el desempeño de la empresa. Luego, se recomienda algunos *KPIS* basados en: 1) El servicio a tiempo, 2) Servicios de calidad al cliente, 3) Rendimiento y desarrollo del capital humano y 4) Crecimiento financiero o social. Para identificar los *KPIS*, dentro de una organización, se deben enfocar las cuatro perspectivas del cuadro de mando integral (BAhare-Kolahi & Valmohammadi, 2015). A continuación, mostramos algunos *KPIS* que hemos utilizado en proyectos tecnológicos para el análisis de datos empresariales:

- Tráfico de clientes: visitas de los clientes
- Tasa de Conversión: tráfico de clientes para ventas realizadas
- Ticket medio: monto que cada cliente compra en cada visita
- Los productos que gustan más a los clientes o productos más vendidos
- La fidelidad del cliente: clientes que repiten las compras o frecuentan más
- Nivel de enganche: tiempo de estancia del cliente dentro del negocio
- Porcentaje de captación: número de personas que pasan por delante del negocio y son captadas
- Tasa de abandono: número de personas que se marchan sin comprar nada
- Gestión de Staff: horarios, apertura y cierre del negocio, calidad de atención del personal a los clientes

### 2.2.9 Técnicas para medir los *KPIS*

Una vez que ya se han definido los principales *KPIS* del negocio, es posible medirlos usando las técnicas más apropiadas para cada uno. Se propone relacionar los *KPIS* vs la forma o técnica de medición para aprovechar los recursos computacionales de la empresa (Leverger *et al.*, 2018).

1. Comparación del indicador en diferentes periodos de tiempo; ejemplo: qué día tiene las mayores visitas o compras en el local, conteo de visitas y sumatoria de las facturas realizadas.
2. Nivel de las tasas de desempeño, conversión de compras dividido por el número de visitas.
3. Tendencias de los tickets medios de compra o el promedio de compras, diarias, mensuales, por temporadas, etc.
4. Mapas de áreas más visitadas, más compradas, se pueden usar mapas de calor para poder graficarlos o segmentar el local y clasificar con la ubicación de los productos vendidos.
5. Relación ventas repetidas vs ventas nuevas (fidelidad), por horarios, días de la semana o temporadas.

- 6.** Medición de la tasa de abandonos del total que entran vs los que compran.
- 7.** Nivel o tasa de rebotes de una sucursal a otra o de una tienda a otra.

## **CAPÍTULO III**

### **Inteligencia de negocios empresariales**



### 3.1 Inteligencia de Negocios (*Business Intelligence BI*)

*El objetivo de este apartado es conocer y entender la estructura y los beneficios de las técnicas y herramientas de BI como aporte de TCE, para auto superar el desempeño en la gestión de las organizaciones modernas.*

#### 3.1.1 El *BI* como parte de las Tecnologías Informáticas TI

Las tecnologías informáticas son un conjunto de herramientas e instrumentos poderosos, de uso simple, para abstraer procesos complejos y solventar la toma de decisiones. Las TI son lo que hace posible encontrar nuevas formas de operar y ejecutar las estrategias de la empresa con un alto grado de efectividad (Olszak & Ziemia, 2006) (Borissova et al., 2020). Las TI ayudan a obtener grandes resultados garantizando mayor rapidez al retorno de la inversión.

Por otro lado, en la actualidad, el *BI* que es parte de las TI se enfoca principalmente en medir, analizar, planear, ejecutar y supervisar los procesos y actividades enfocados en el negocio, como si tuvieran inteligencia, autonomía y un comportamiento aleatorio. Según lo definido por empresas como “*Tableau*”, la inteligencia de negocios combina, análisis de negocios, minería, visualizaciones e infraestructura de datos robusta, para contar con una visión integral de los datos de la empresa, adaptándose rápidamente a los cambios del mercado.

#### 3.1.2 La inteligencia de negocios en el nuevo milenio

En el contexto de la Inteligencia de Negocios, esta es aceptada como el conjunto de herramientas, metodologías y técnicas, y prácticas enfocadas en la creación y administración de la información que permite tomar mejores decisiones a los usuarios de una organización (Cevallos-Cuenca & Vera-Flores, 2020), con el objetivo de superar permanentemente sus propias barreras.

Una definición alternativa y extensa refiere un conjunto de metodologías, aplicaciones y tecnologías que permiten reunir, depurar y transformar datos de los sistemas transaccionales e información desestructurada (interna y externa a la compañía) en información estructurada (Ghiglione, 2021). El *BI* tiene por objeto, la explotación directa de la información (*reporting*, análisis OLTP / OLAP, alertas...), para su análisis y conversión en conocimiento, dando, de este modo, soporte a la toma de decisiones correctas de negocios sostenibles. Ver Tabla 5.

<b>Observar</b>	¿Qué está ocurriendo?
<b>Comprender</b>	¿Por qué ocurre?
<b>Predecir</b>	¿Qué ocurriría?
<b>Colaborar</b>	¿Qué debería hacer el equipo?
<b>Decidir</b>	¿Qué camino se debe seguir?
<b>*Mejorar</b>	¿Qué errores corregir?

**Tabla 5.** Aportes de un proceso de inteligencia de negocios BI.  
**Fuente.** Sennexus Business Intelligence [2],\*nuevo aporte

Adicional a estas definiciones, la inteligencia de negocios está orientada a la gerencia estratégica. El *BI* se alinea con el negocio por medio de instrumentos de planificación estratégica como un *balanced scorecard BSC*, donde el *BSC* contempla indicadores de desempeño para brindar distintos servicios gerenciales a la organización.

Se debe aclarar que, la transformación e integración de la información, el análisis, minería de datos y los tableros de control son levantados sobre un motor dorsal de un *datawarehouse o datamart*. Estos son procesos que se deben desarrollar como parte del diseño e implementación del *BI*. Según (Morales, & Catucuamba-Jima, 2019), las etapas de implementación del *BI* se describen a continuación:

- Análisis de requerimientos de la organización
- Adquisición de los datos
- Diseño y modelado de la arquitectura de la solución
- Construcción del *Datawarehouse*
- Extracción, transformación y carga de datos de diversas fuentes de información
- Modelado de la solución del *BI*
- Diseño e implementación de tableros de control

### 3.1.3 Herramientas de inteligencia de negocios (*BI*)

Las herramientas para desarrollar *BI* son diversas, sin embargo aquí analizaremos un listado de las más utilizadas y mejor clasificadas según expertos, las cuales se utilizan para desarrollar modelos de Inteligencia de Negocios (En página de Alonso de Molina, 2015).

*Tableau*, herramienta de nueva generación para soluciones ágiles para *BI*

*Power BI*: herramienta de alta flexibilidad para adaptarse al negocio con conexión a diversas fuentes de datos.

*SQL Server*: adecuada para realizar un análisis panorámico de la empresa y tomar

<sup>2</sup> Sennexus, [http://www.sinnexus.com/business\\_intelligence/sistemas\\_informacion.aspx](http://www.sinnexus.com/business_intelligence/sistemas_informacion.aspx)

las mejores decisiones.

*IBM Cognos Analytics:* Este brinda un sistema efectivo para llevar a cabo evaluaciones correctas de la información y una posterior toma de decisiones

*Microsoft Dynamics CRM:* efectiva para la administración de clientes.

*Oracle Business Intelligence:* una de las más completas en el mercado ya que cuenta con paneles interactivos, análisis predictivos en tiempo real, entre otros.

*Ultimus:* un entorno integrado que permite compartir información entre aplicaciones.

*Office SharePoint Server:* facilita el acceso a la información en cualquier momento y lugar.

*QlikView:* mantiene las bases de datos al alcance de una manera sin precedentes.

*Microsoft Performance Point Server:* permite supervisar, alinear y hacer un plan de negocio.

*JetReports:* especial para crear informes ERP.

*Eclipse BIRT Project:* genera informes para aplicaciones web de código abierto.

*JasperReports:* permite crear informes de rápida impresión.

*LogiReport:* aplicación gratuita basada en web de LogiXML

*OpenI:* aplicación web orientada al reporting OLAP.

*SPSS:* programa estadístico especialmente empleado en ciencias sociales e investigaciones de mercado.

*Pentaho:* incluye herramientas para generar informes, minería de datos, ETL, entre otros.

*RapidMiner:* permite analizar datos a través de un entorno gráfico.

*Crystal Reports:* genera informes desde bases de datos múltiples.

*ApeSoft:* ofrece una interface sencilla similar a Microsoft Excel.

*SAS Institute:* facilita la gestión de riesgo financiero, el desarrollo de modelos de minería de datos, etc.

*NiMbox:* organiza los datos de la empresa en interactivas aplicaciones.

### 3.2 Almacén de datos (*Datawarehouse Dw*)

Un *Datawarehouse* no se compra, se construye. El proceso de construcción y desarrollo de un DW es evolutivo (Silva et al., 2019). Al iniciar el ensamblado del *BI* se debe construir el motor dorsal para administrar la información que se utiliza en los servicios del *BI*. Este motor es el *Datawarehouse*, conocido también como almacén o bodega de datos. Para esto se presentan algunas definiciones del *Datawarehouse*, según algunos autores.

Para (Clark & Clark, 2004), el *DW* o bodega de datos, es el lugar de almacenamiento central para todas o casi todas las partes de datos de los diversos sistemas comerciales de una empresa reúnen.

Otras definiciones aluden al proceso mediante el cual una organización o empresa particular almacena todos aquellos datos e información necesarios para su propio desempeño (Rosenkranz et al., 2010). Se reconoce que este tipo de material se organiza de manera tal para facilitar el análisis y la realización de reportes en las situaciones en que sea necesario hacerlo. Contar con un fácil acceso a los datos de importancia tendrá directa relación con la efectividad de las diversas tareas de la empresa.

También se dice que el conjunto de *Datamarts* (se explica más adelante) da lugar a un *DW*. Algunos autores definen al *DW* como una estructura de almacenamiento de datos de distintas fuentes, que es una de sus fortalezas principales. Por lo tanto, podemos ver que el *DW* es el motor de los distintos servicios del *BI* y que su diseño debe ser cuidadosamente analizado, contemplando dos principios, el primero, los fines o propósitos que va a tener el *BI*, y el segundo, los datos que se tienen en las fuentes de extracción.

Un modelo multidimensional se puede representar como un esquema en estrella, copo de nieve (*snowflake*) o constelación de hechos. Sin embargo, las herramientas nuevas de *BI* no se alinean demasiado a los *DW*; son herramientas abiertas a cualquier estructura de datos, por lo que el futuro del *DW* no es tan prometedor.

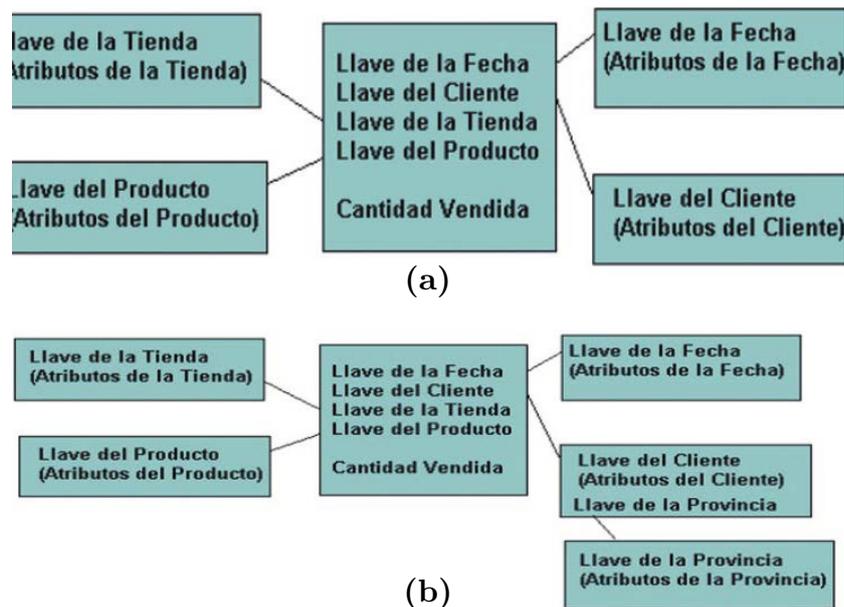
#### 3.2.1 Modelo Estrella

La topología más común para construir un *DW* es este modelo, que está formado por una tabla central llamada *fact-table* o tabla de hechos y un conjunto de tablas satélites conocidas como dimensiones. La tabla de hechos está formada en este orden, primero, las claves foráneas de las tablas dimensionales y, segundo, las medidas o *measures* a utilizarse en el análisis. En cambio, en las tablas dimensionales están los atributos de la información y el agrupamiento que se puede dar a los reportes, *KPI'S* o tableros de control. Este modelo es el más utilizado generalmente por su facilidad de implementación.

### 3.2.2 Modelo Copo de nieve

La topología copo de nieve es una extensión del modelo estrella. La diferencia radica en que las dimensiones pueden estar relacionadas con otras tablas, que pueden ser otras dimensiones de agrupamiento con un nivel de normalización superior. También se dice que esta es una estructura jerárquica.

En cualquiera de los casos, para construir un *DW* con estas topologías, se desarrolla un proceso de desnormalización para llenar la tabla de hechos, aunque este no es el único caso, porque también se puede dar al llenar alguna tabla de dimensión. Para entender mejor estos dos modelos podemos revisar la Figura 17.



**Figura 17.** Comparación de los modelos (a) Estrella y (b) Copo de nieve de un *DW*  
**Fuente.** (Trujillo, 2006)

### 3.2.3 La Dimensión tiempo (Tabla) en el *DW*

La dimensión tiempo es una de las más importantes de un *DW*, ya que esta dimensión debe existir en casi todas las estructuras de este tipo. Sin embargo, no es tomada con la suficiente atención por muchos autores. La dimensión tiempo permite generar un análisis de la información, jerárquicamente, por los distintos rangos de tiempos especificados, que también se los denomina nivel de granularidad. La tabla tiempo generalmente se debe construir en una plataforma de un servidor robusto, como es el caso de *Analysis Services de SQL Server*.

### 3.2.4 La granularidad

Cuando un objeto dispone de una propiedad que se puede descomponer jerárquicamente en más de tres niveles, se dice que tiene una granularidad fina. Mientras el

objeto se acerca al nivel superior se dice que el grano es grueso. Por ejemplo, la propiedad fecha se podría descomponer en años, semestres, trimestres, meses, semanas o días. Este comportamiento se puede representar en un esquema de tipo estrella o copo de nieve, mejor todavía, pero este último es más costoso. Cada tabla de hechos y dimensiones tiene su propio grano o granularidad.

La tabla de hechos (fact-table) posee un cierto nivel de detalle dependiendo de la cantidad de asociaciones o relaciones existentes con las tablas de dimensiones. El grano del modelo dimensional es el nivel de detalle más fino que está implícito al unir las tablas de hechos y dimensiones. Por ejemplo, en un modelo dimensional que consta de las dimensiones: fecha, cliente, localidad, producto y ventas, la granularidad se puede analizar en la dimensión localidad, descomponiendo, país, ciudad, cantón, sector, calle (Hartzler *et al.*, 2015).

De forma similar, se puede encontrar un nivel de granularidad más preciso en la dimensión tiempo, en la cual se identifica los distintos rangos de fechas como son los años, semestres, trimestres, meses, quincenas, semanas o días. Si queremos determinar un grano más fino todavía, se puede trabajar a nivel de minutos o segundos.

Por lo tanto, la granularidad permite realizar análisis de la información agrupada según el nivel de detalle. Sin embargo, mientras más sea esta granularidad, el proceso general del BI puede perder optimización en el desempeño del tiempo debido al tamaño que alcanzaría esta tabla tanto en el número de columnas y más en el número de registros, pero ganaría en la flexibilidad de los informes e indicadores. Ver Figura 18.

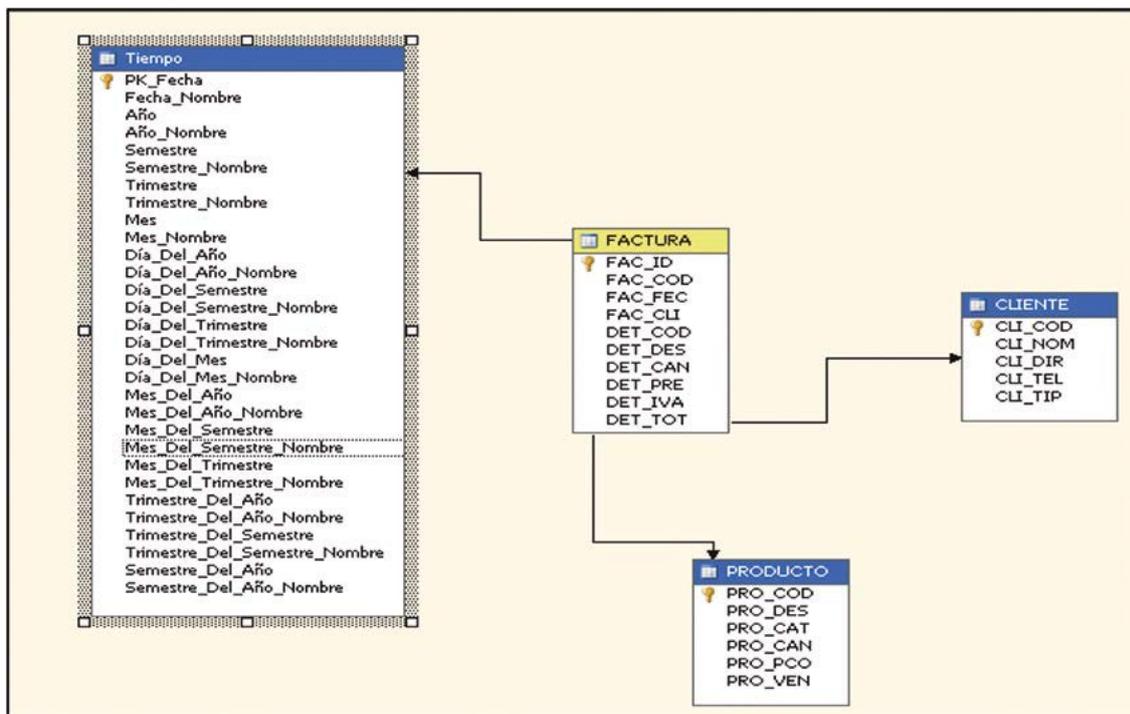


Figura 18. Dimensión tiempo dentro de un cubo OLAP de SQL Server

### 3.2.5 La tienda de datos (*Datamart o DM*)

Para definir un *DM* se puede decir que el conjunto de *DM* es un Datawarehouse, dicho en otras palabras, cuando tenemos un *DW* específico para un área determinada eso se puede interpretar como un Datamart. Por lo tanto, el *DM* hereda la mayoría de propiedades de un *DW*, como los modelos estrella y copo de nieve para su estructura, y las tablas de hechos y las dimensiones, con sus respectivas medidas. De esta forma se pueden plantear dos tipos de *Datamarts*:

### 3.2.6 *Datamart OLAP*

Se basan en estructuras dimensionales conocidas como los cubos *OLAP*, que se construyen agregando según los requisitos de cada área o departamento, de las dimensiones y los indicadores necesarios de cada cubo relacional. El modo de creación, explotación y mantenimiento de los cubos *OLAP* es muy heterogéneo y depende de la herramienta final que se utilice.

### 3.2.7 *Datamart OLTP*

Pueden basarse en un simple extracto del *Datawarehouse*, no obstante, lo común es introducir mejoras en su rendimiento (las agregaciones y los filtrados suelen ser las operaciones más usuales), aprovechando las características particulares de cada área de la empresa. Las estructuras más comunes, en este sentido, son las tablas report, que vienen a ser fact-tables reducidas (que agregan las dimensiones oportunas), y las vistas materializadas, que se construyen con la misma estructura que las anteriores, pero con el objetivo de explotar la reescritura de consultas (*queries*).

Los *DM* que están dotados con estas estructuras óptimas de análisis presentan las siguientes ventajas:

- poco volumen de datos
- mayor rapidez de consulta
- consultas *SQL* y/o *MDX* sencillas
- validación directa de la información
- facilidad para visualizar el historial de los datos

## 3.3 Procesos de extracción, transformación y carga (*del inglés, extraction, transformation and loading ETL*)

*El objetivo esencial de este componente es desarrollar procesos de extracción, transformación y carga para poblar un Datawarehouse o un Datamart, de forma integral y óptima para el análisis de la data.*

**ETL** - este término viene de las siglas en inglés, *Extract-Transform-Load* que significan “Extraer, Transformar y Cargar” los datos de una o varias estructuras de una empresa. *ETL* es el proceso que organiza el flujo de los datos entre diferentes sistemas en una organización y aporta los métodos y herramientas necesarias para mover datos desde múltiples fuentes a un almacén de datos, reformatearlos, limpiarlos y cargarlos en otra base de datos, *datamart* o bodega de datos. *ETL* forma parte de la Inteligencia Empresarial (*Business Intelligence*), también llamado “Gestión de los Datos” (*Data Management*). Los *ETL* se pueden realizar inclusive para estructuras *NoSQL* (*No only SQL*), con ciertas diferencias propias de cada estructura.

La idea es que una aplicación *ETL* lea los datos primarios de una base de datos; luego, realice transformación, validación, el proceso cualitativo y filtración; y al final, escriba datos en un nuevo almacén de datos. En este momento, los datos estarán disponibles para la siguiente etapa de análisis (Vila Guerrero, 2019).

Lista de las más populares herramientas y aplicaciones *ETL* del mercado, según (Vila Guerrero, 2019).

- IBM Websphere DataStage (anteriormente Ascential DataStage y Ardent DataStage)
- Pentaho Data Integration (Kettle ETL) - Una herramienta Open Source Business Intelligence
- SAS ETL Studio
- Oracle Warehouse Builder
- Informática PowerCenter
- Cognos Decisionstream
- Ab Initio
- Business Objects Data Integrator (BODI)
- Microsoft SQL Server Integration Services (SSIS)\*

Actualmente, la limpieza de datos, que podría entenderse como una acción integrada en la fase de transformación de datos, se tiende a considerar como una fase separada del proceso *ETL*. Esta visión corresponde a una concepción más moderna y práctica del proceso. Para ahorrar tiempo y ganar en efectividad es conveniente unificar criterios, por ejemplo, introduciendo “av” en vez de “avenida” en todos los registros de una base de datos de direcciones postales, antes de empezar el proceso *ETL* propiamente dicho. Por tal motivo, tan importante es tener la información consolidada, como que todos los datos sean correctos y con una visión única para los usuarios. Solo así se pueden lograr unos circuitos de trabajo y análisis de dichos datos realmente óptimos y efectivos.

Los procesos *ETL* pueden incluir: Sistemas *legacy*, es decir, legados, heredados o antiguos; sistemas nuevos, basados en *Windows*, *Linux* y también en las redes sociales modernas: *Facebook*, *Twitter*, *LinkedIn*, etc.

Los sistemas *legacy* o heredados se caracterizan, generalmente, por ser cerrados, no permitir cambios y tener un difícil acceso (normalmente se necesita algún tipo de driver especial). Son sistemas que procesan hacia adentro y, por lo tanto, no permiten la agregación de una computadora que trabaje en paralelo.

Por el contrario, los sistemas nuevos o modernos (basados en Windows o Linux) son abiertos, amplios e interconectados. Un ejemplo sería un clúster de servidores Linux, el cual permite la interconexión de los distintos nodos entre sí.

### 3.3.1 Beneficios de los procesos *ETL*

A cualquier empresa u organización le beneficia poner en marcha un proceso *ETL* para mover y transformar los datos que maneja en tiempo real (Bustamante Martínez *et al.*, 2013):

- Sirve para integrar grandes o medianas cantidades de datos de distintas fuentes en un solo destino llamado *Datawarehouse* o *Datamart*.
- Puede crear una *Master Data Management*, es decir, un repositorio central estandarizado de todos los datos de la organización. Por ejemplo, si se desea implementar *DW* para el área de ventas donde se tenga tablas de clientes, productos facturas, se debe extraer desde el origen y llenar en el destino (*DW*), siguiendo los procesos de transformado, purificado y mezclado de la información para que sea integral (Ver Figura 21 y 22).
- Posibilita a los directivos tomar decisiones estratégicas basadas en el análisis de los datos cargados en las bases nuevas y actualizadas del *datamart* o *datawarehouse*.
- Permite tener una visión global de todos los datos consolidados en un *datawarehouse*.

Los procesos ETL son esenciales para las actividades gerenciales que desean manejar *BI*, debido a su capacidad para integrar grandes bases de datos.

### 3.4 Herramientas de extracción transformación y carga (*ETL*)

Las herramientas que tiene SQL Server para desarrollar los procesos ETL se clasifican en un conjunto de tareas donde las principales son; la tarea de ejecutar SQL, tarea de ejecutar paquete, tarea de servicios web y la tarea de flujo de datos. Estas tareas permiten realizar conexiones a las fuentes de datos, desarrollar sentencias SQL, procesos de limpieza, procesos de transformación mezclas de datos, procesos en paralelo dentro de contenedores, limpieza y carga, entre otros. A continuación, se presenta los más importantes y los utilizados con mayor frecuencia en la preparación de la estructura para levantar un *BI*.

### 3.4.1 Tarea ejecutar *SQL*

La tarea “ejecutar *SQL*” resuelve instrucciones *SQL* o procedimientos almacenados de un paquete. La tarea puede contener una sola instrucción *SQL* o múltiples instrucciones que se ejecutarán de forma secuencial. Puede usar la tarea “ejecutar *SQL*” para los siguientes fines [3]:

- Truncar una tabla o vista en preparación para insertar datos.
- Crear, modificar y quitar objetos de base de datos, como tablas y vistas.
- Volver a crear tablas de hechos y tablas de dimensiones antes de cargar datos en ellas.
- Ejecutar procedimientos almacenados. Si la instrucción *SQL* invoca un procedimiento almacenado que devuelve resultados de una tabla temporal. Para esto usamos la opción `WITH RESULT SETS` que define los metadatos del conjunto de resultados.
- Guardar en una variable el conjunto de filas devuelto por una consulta.

La tarea “ejecutar *SQL*” puede usarse en combinación con los contenedores de bucles *Foreach* y *For*, para ejecutar varias instrucciones *SQL*. Estos contenedores implementan flujos de control que se repiten en un paquete y pueden ejecutar repetidamente la tarea “ejecutar *SQL*”. Por ejemplo, un paquete puede usar el contenedor de bucles *Foreach* para enumerar los archivos de una carpeta y ejecutar la tarea de *SQL* repetidamente almacenada en cada archivo. Ver Figura 19.

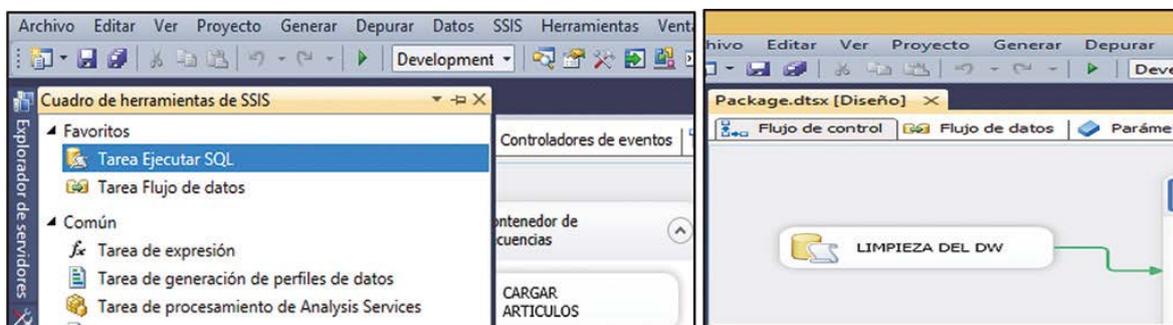
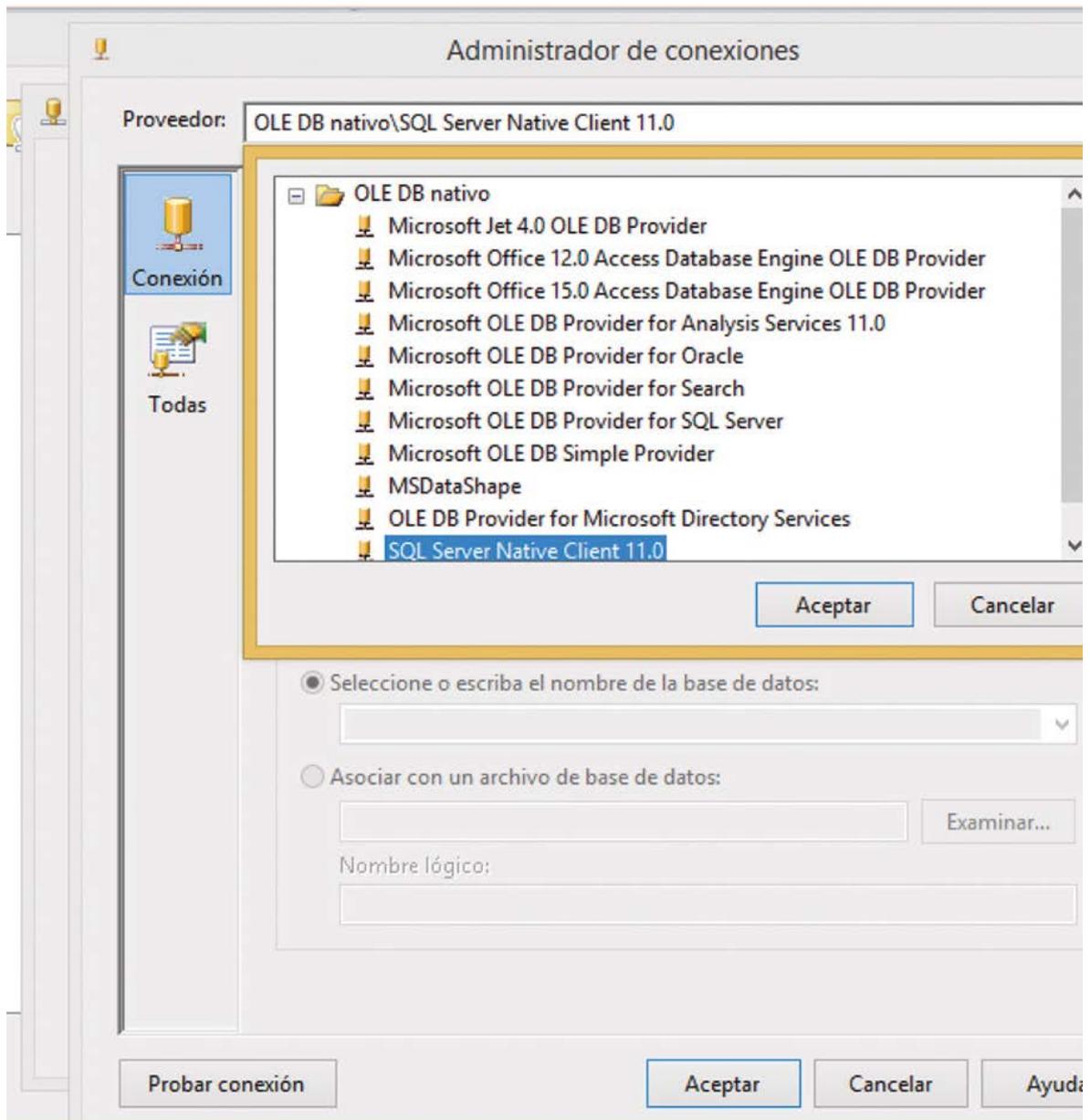


Figura 19. Insertando una tarea ejecutar *SQL* desde *Business Intelligence*

### 3.4.2 Conectar con un origen de datos

La tarea “ejecutar *SQL*” puede usar distintos tipos de administradores de conexión para conectar con el origen de datos en el que se ejecuta la instrucción *SQL* o el procedimiento almacenado. La tarea puede usar los tipos de conexión mostrados en la Figura 20.

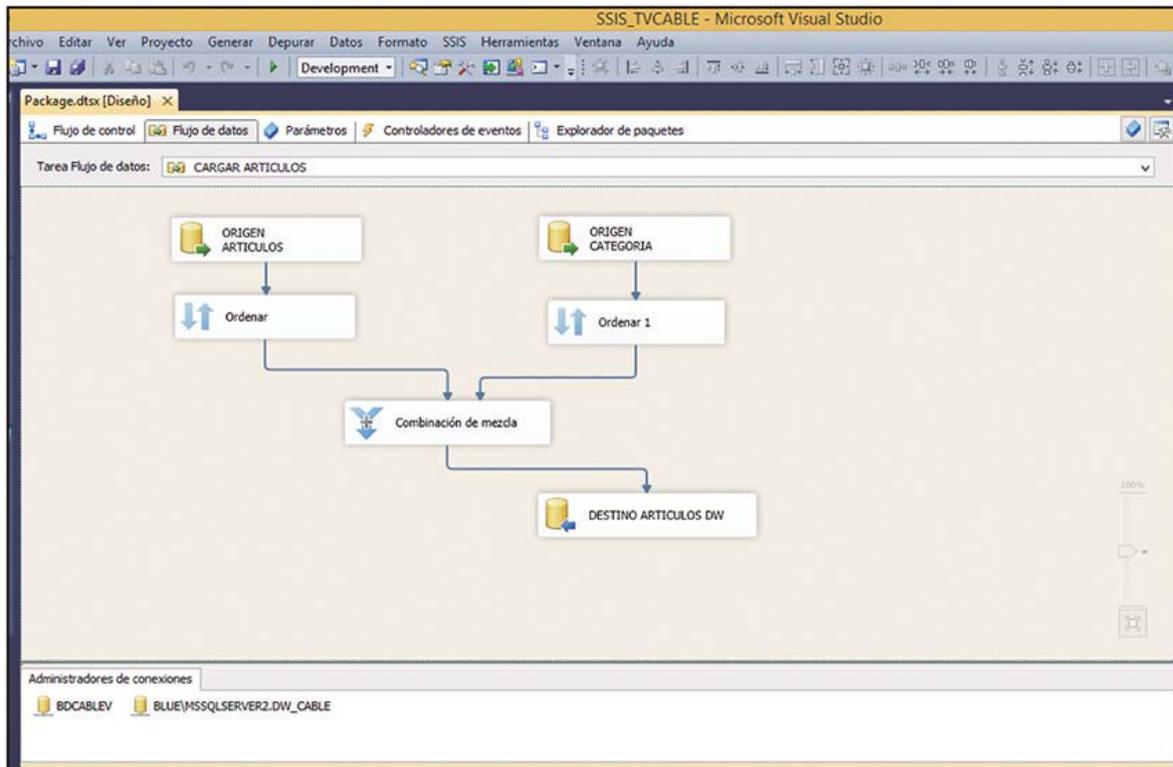
<sup>3</sup> Microsoft, Extract, transform, and load (ETL), En <https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl>, 2022.



**Figura 20.** Administrador de conexiones en *Business Intelligence de SQL Server*.

### 3.4.2 Tarea flujo de datos

Esta tarea encapsula el motor del flujo de datos para mover entre orígenes y destinos, permitiendo al usuario transformar, limpiar y modificar datos durante su ejecución. Adicionar una tarea flujo de datos a un flujo de control de paquetes admite que el paquete extraiga, transforme y cargue datos, se amplía en la página web de Microsoft. Este flujo de datos se forma de un componente, pero, normalmente es un conjunto de componentes de flujo de datos conectados entre uno o más orígenes que extrae datos, transforma, enruta y reduce datos para enviar a un destino que carga estos datos. Ver Figura 21.



**Figura 21.** Tarea flujo de datos para mezclar tablas y cargar al destino *datawarehouse*

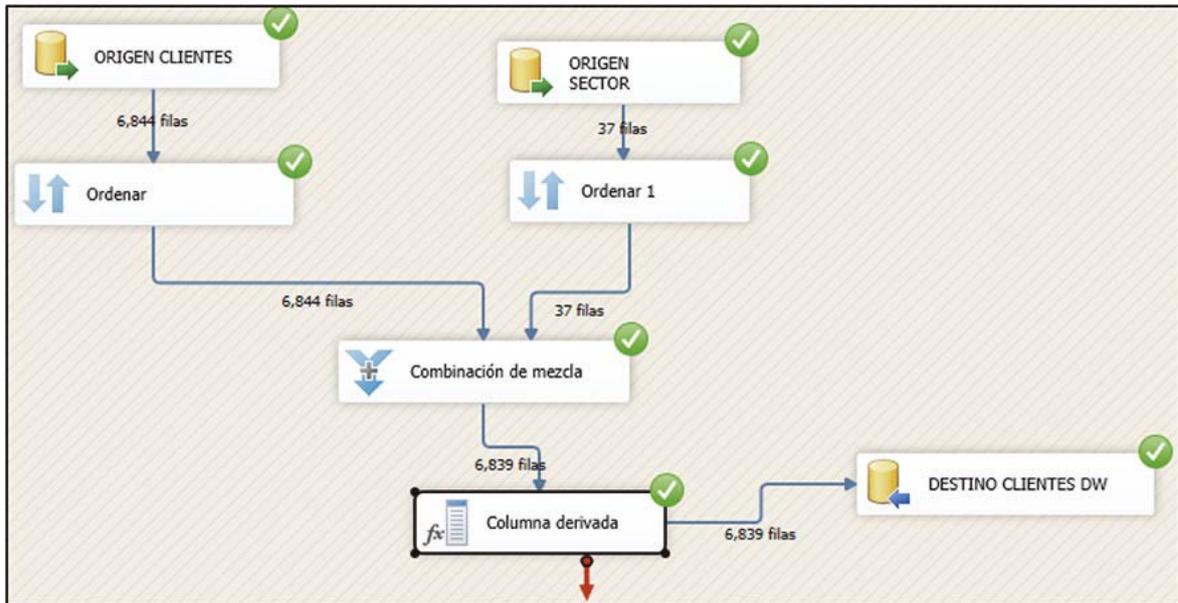
En el tiempo de ejecución, la tarea flujo de datos genera un procedimiento de ejecución y el motor de flujo de datos ejecuta. Se puede crear una tarea flujo de datos que no tenga flujo de datos, pero solo se ejecuta, si incluye por lo menos un flujo de datos.

### 3.4.3 Varios flujos

El manual de usuario de Microsoft describe que, una tarea de flujo de datos puede contener varios flujos de datos. Si una tarea copia varios conjuntos de datos, y si el orden en que los datos se copian no es significativo, puede ser más conveniente incluir varios flujos de datos. Por ejemplo, se puede crear cinco flujos de datos, y cada uno de ellos copiaría datos de un archivo plano en una tabla de dimensiones diferente, en un esquema de estrella de almacenamiento de datos.

### 3.4.4 Entradas del registro

*Integration Services* (SSIS) suministra un conjunto de eventos de registro que son utilizables por todas las tareas. *Integration Services* proporciona entradas del registro personalizadas para numerosas tareas. Por ejemplo, convierte una columna derivada uniendo dos columnas del origen de datos. Además, es factible configurar mensajes personalizados para cada registro. Ver Figura 22.



**Figura 22.** Tarea flujo de datos ejecutada, para transformar una columna derivada.

La tarea Flujo de datos incluye las siguientes entradas de registro personalizadas.

### 3.4.5 Tarea flujo de datos en *Integration Services de SQL Server*

Ejecuta flujos de datos para extraer, aplicar transformaciones de nivel de columna y cargar datos.

### 3.4.6 Tareas de preparación de datos

Estas tareas llevan a cabo los procesos siguientes: ejecutar métodos Web, copiar archivos o directorios, aplicar operaciones a documentos XML, descargar archivos o datos y generar perfiles de los datos para la limpieza.

### 3.4.7 Tareas de flujo de trabajo

Tareas que se comunican con otros procesos para enviar y recibir mensajes entre paquetes, ejecutar paquetes, ejecutar programas o archivos por lotes, enviar mensajes de correo electrónico, entre otros.

### 3.4.8 Tareas de *SQL Server*

Estas sirven para acceso, inserción, eliminación, modificación y copia de objetos y datos.

### 3.4.9 Tareas de *scripting*

Son tareas que aumentan la funcionalidad de los paquetes mediante *scripts*.

### 3.4.10 Tareas de *Analysis Services*

Estas funciones sirven para creación, modificación, eliminación y procesamiento de objetos de *Analysis Services*. Aquí se generan los procesos de cubos de datos, generación de KPI'S, entre otros que se utilizan en el análisis.

### 3.4.11 Tareas de mantenimiento

Realizan funciones administrativas como reducir bases de datos de *SQL Server*, volver a generar y reorganizar índices, crear copias de seguridad y ejecutar trabajos del Agente *SQL Server*.

### 3.4.12 Tareas personalizadas

En este apartado, se desarrolla la programación de tareas personalizadas mediante un lenguaje compatible con extensiones de archivo *.com* como *Visual Basic*, o un lenguaje de programación *.net*, como *C#*. Si quiere tener acceso a una tarea personalizada en el Diseñador *SSIS*, puede crear y registrar una interfaz de usuario.

## 3.5 Arquitecturas *ROLAP*, *MOLAP* y *HOLAP*.

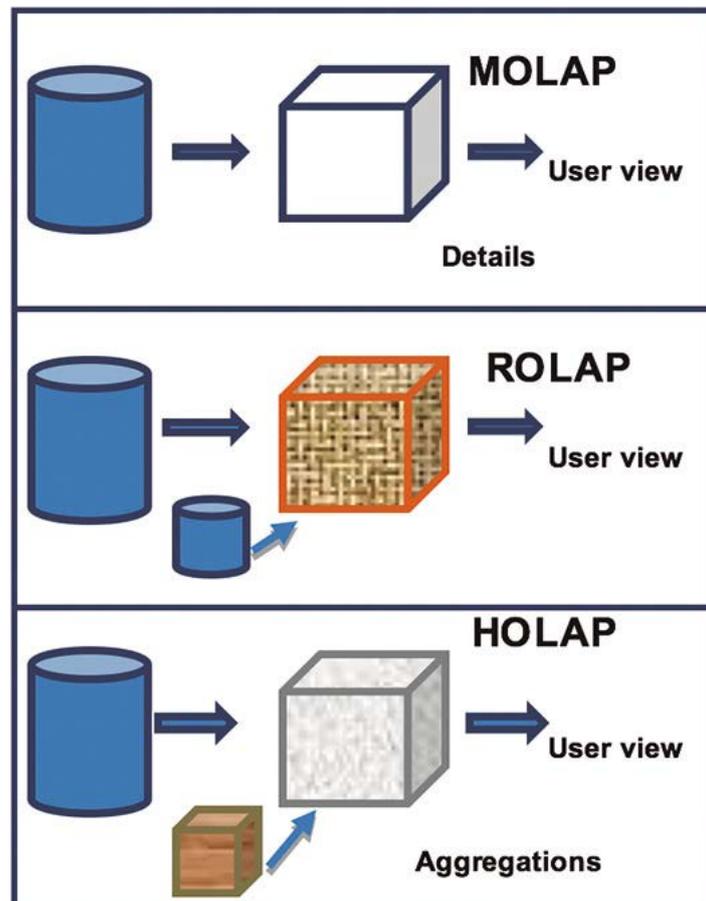
*ROLAP* significa *Relational OnLine Analytical Processing*, en español, Procesamiento Analítico OnLine Relacional.

*MOLAP* es el acrónimo inglés de *Multi-dimensional Online Analytical Processing*, español, Procesamiento Analítico Multidimensional en Línea.

*HOLAP* es acrónimo inglés de *Hybrid Online Analytical Process*, en español, Procesamiento Analítico en Línea Híbrido.

El objetivo de conocer estas arquitecturas es para discutir las funciones, aplicaciones, ventajas y desventajas de cada una, aprovechar el máximo de sus bondades, y aplicar la mejor solución según el tipo de negocio.

Estas arquitecturas indican las distintas formas de diseñar, construir y almacenar los datos dentro de una de estas estructuras analíticas multidimensionales, con base en los recursos y tecnologías que se dispongan, ver Figura 23.



**Figura 23.** Representación de las tres arquitecturas OLAP.  
**Fuente.** (Tamayo & Moreno, 2006)

*ROLAP* es un modelo *OLAP* Relacional, cuya arquitectura está formada por una tabla de hechos central que está relacionada con tablas dimensionales, estableciendo una estrella, copo de nieve o constelación; se levanta sobre una base de datos relacional. Dicha arquitectura puede acomodar mayores cantidades de datos que las otras arquitecturas multidimensionales afines. Su desventaja es la lentitud de las transacciones en comparación con las otras. El modelo *ROLAP* está basado en un *datawarehouse* para el almacenamiento de datos y son poblados por medio de un proceso *ETL*.

Los cubos se crean dinámicamente en esta arquitectura al momento de generar las consultas de implementación. Con esto se logra mayor flexibilidad, posibilitando la obtención de consultas *ad-hoc* (consultas adecuadas para estos fines) (Zhang *et al.*, 2019).

La arquitectura *MOLAP* usa bases de datos multidimensionales para proporcionar y gestar el análisis. Su principal premisa es que el *OLAP* está mejor implantado almacenando los datos multidimensionalmente. Por el contrario, la arquitectura *ROLAP* cree que las capacidades *OLAP* están perfectamente implantadas sobre bases de datos relacionales. Un sistema *MOLAP* usa una base de datos propietaria multidimensional, en la que la información se almacena multidimensionalmente, es decir, para ser visualizada en varias dimensiones de análisis.

El sistema *MOLAP* utiliza una arquitectura de dos niveles: la base de datos multidimensionales y el motor analítico. La base de datos multidimensional es la encargada del manejo, acceso y obtención del dato. El nivel de aplicación es el responsable de la ejecución de los requerimientos *OLAP*. El nivel de presentación se integra con el de aplicación y proporciona una interfaz a través del cual, los usuarios finales visualizan los análisis *OLAP*. Una arquitectura cliente/servidor permite a varios usuarios acceder a la misma base de datos multidimensional.

La información procedente de los sistemas operacionales se carga en el sistema *MOLAP*, mediante una serie de rutinas por lotes. Una vez cargado el dato elemental en la Base de Datos multidimensional (*MDDDB*), se realiza una serie de cálculos por lotes, que permite calcular los datos agregados, a través de las dimensiones de negocio, rellenando la estructura *MDDDB*.

Tras rellenar esta estructura, se generan unos índices y algoritmos de tablas hash para mejorar los tiempos de accesos a las consultas. Una vez que el proceso de compilación se ha acabado, la *MDDDB* está lista para su uso. Los usuarios solicitan informes a través de la interface, y la lógica de aplicación de la *MDDDB* obtiene el dato, como se muestra en la Tabla 6.

*MOLAP* requiere unos cálculos intensivos de compilación. Lee datos pre compilados y tiene capacidades limitadas de crear agregaciones dinámicamente, o de hallar ratios que no se hayan precalculado y almacenado previamente.

<b>Elementos</b>	<b>Multidimensional</b>	<b>Relacional</b>
Datos	detalle y pre calculados (agregados)	detalle y agregados
Estructura	matrices comprimidas	tablas relacionales
Administración	especialista en BDMD	administrador de BD
Acceso	lenguaje especializado	SQL

**Tabla 6.** Diferencias entre *ROLAP* y *OLAP*  
Fuente. (Tamayo & Moreno, 2006)

Finalmente, tenemos el modelo *HOLAP* que es una solución híbrida que combina las dos técnicas anteriores, con el propósito de brindar un mejor performance del manejo de los datos y de su análisis. Esta solución maneja los datos de detalle de mayor volumen en una datawarehouse, y los datos agregados los mantiene en una estructura *MOLAP* multidimensional (Hee-Lee & Chan-Park, 2005).

### 3.6 La generación de cubos, dimensiones, medidas y jerarquías

*El objetivo es desarrollar estructuras multidimensionales (el cubo OLAP) capaces de analizar la información de forma dinámica e intuitiva con las técnicas y herramientas que existen para el enriquecimiento de los datos.*

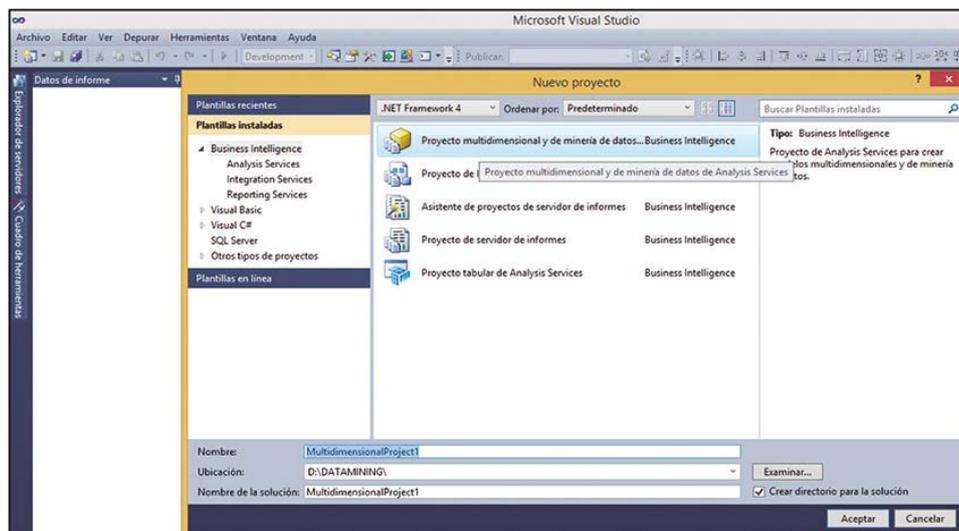
#### 3.6.1 Creación del proyecto Analysis Services, caso de estudio

Un cubo *OLAP* es una estructura de datos que supera las limitaciones de las bases de datos relacionales y proporciona un análisis rápido de datos. Los cubos pueden mostrar y sumar grandes cantidades de datos, a la vez que proporcionan a los usuarios acceso mediante la búsqueda de los puntos de datos. De este modo, los datos se pueden resumir o reorganizar según sea necesario, para procesar la variedad más amplia de preguntas pertinentes al área de interés de un usuario (Microsoft, 2016d).

El modelo que usamos aquí para el ejemplo es el *ROLAP*, con el propósito de explicar la creación y el uso del DW o *DM*, con los procesos *ETL* de transformación y carga de datos limpios.

Para la explicación, usamos una base de datos transaccional BDBase1 de *MS Access* como datos de origen y se carga por medio de un proceso *ETL* al *Datawarehouse* de *SQL Server*, en un proyecto llamado SSIS de *Integration Services*. Posteriormente se diseña el cubo en un proyecto *SSAS* del *Analysis Services*, como muestran las siguientes figuras.

Los cubos *OLAP* son analizados en un estudio de caso como metodología aplicada, para ilustrar su funcionalidad e implementación en el mundo real. Para esto, se utiliza procesos de diseño, paso a paso, donde, cada etapa de creación se construye mediante la herramienta *Analysis Services de Business Intelligence de SQL Server 2012*. Ver Figura 24.



**Figura 24.** Ventana del nuevo proyecto de *Analysis Services*, para creación de los cubos *OLAP*.

### 3.6.2 Creación del origen de datos

Una vez creado el proyecto dentro de Analysis Services, las etapas que se deben desarrollar sistemáticamente son creación de una conexión con los datos de origen *Datawarehouse*, tipo estrella. Ver Figura 25.

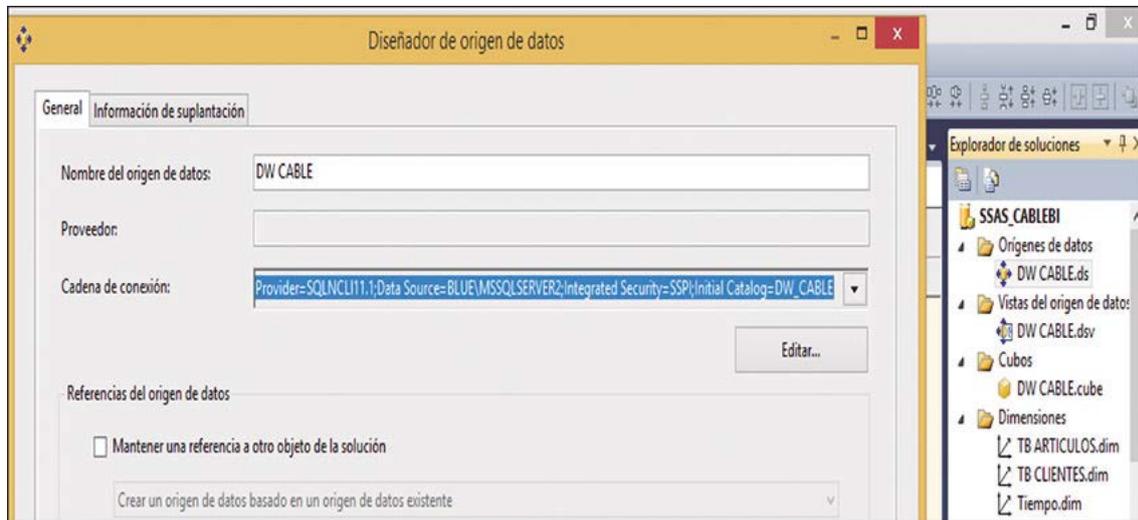


Figura 25. Ventana del diseñador del origen de datos, conexión con el DW.

### 3.6.3 Creación de la vista del origen de datos

En la segunda etapa, definimos la vista del origen de datos. En este paso es necesario seleccionar las dimensiones y la tabla de hechos con los cuales se debe generar el cubo correcto para el análisis de la data con los indicadores claves de desempeño. Ver Figura 26.

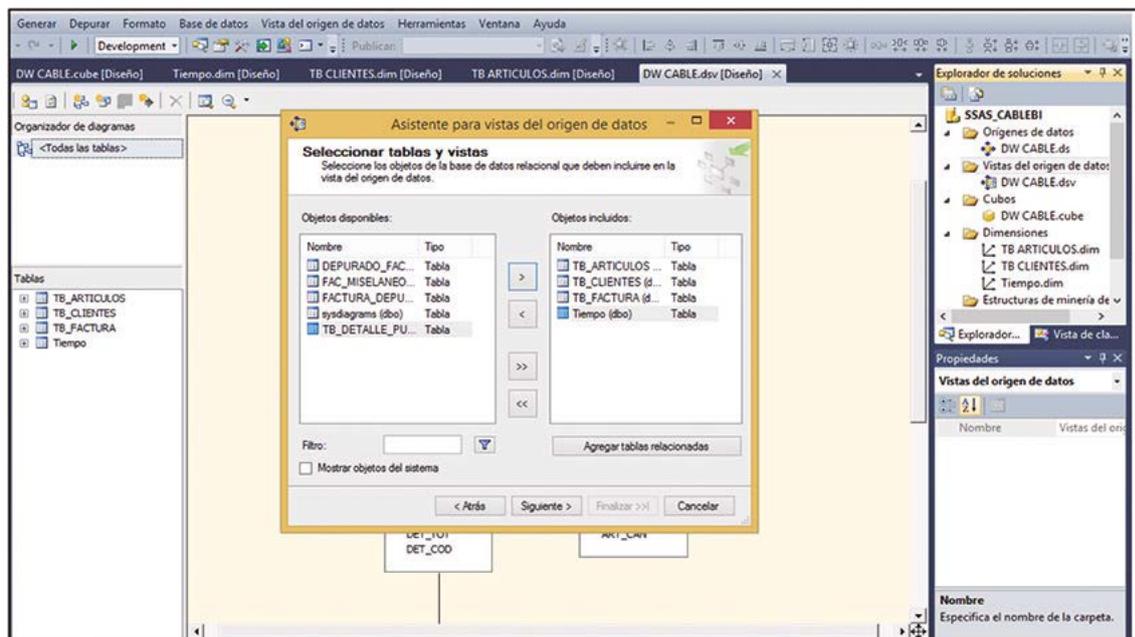


Figura 26. Ventana del diseñador del origen de datos, conexión con el DW.

### 3.6.4 Creación de las dimensiones del cubo

La tercera etapa en la creación de los cubos es una de las más cruciales para tener éxito en el desarrollo de los mismos, aquí definimos las dimensiones con sus respectivos atributos. Aquí establecemos solo los necesarios, ya que algunos de estos pueden ser utilizados como jerarquías para agrupamientos. A partir de esta fase se pueden hacer *drill down* o *drill through* dentro del análisis del cubo, específicamente cuando se diseñen los tableros de control. Ver Figura 27.

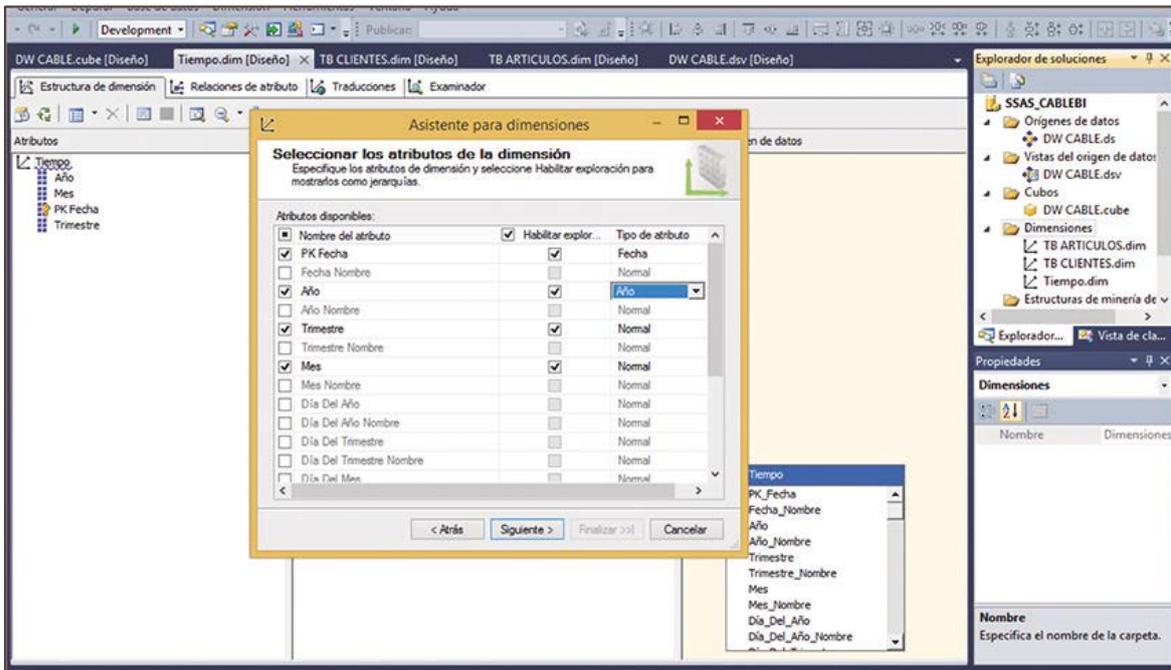


Figura 27. Ventana de dimensiones y definición de atributos y jerarquías del cubo.

### 3.6.5 Creación y diseño del cubo

Para este proceso se desarrolló siguiendo las normas sugeridas por la documentación de *Microsoft* (Microsoft, 2022), en esta etapa se crear el cubo, la cual debe contemplar los siguientes ítems:

- Seleccionar el método de creación, aplicando tablas existentes, crear un cubo vacío o generar en el origen de datos con plantillas del modelo de *Adventure Works enterprise* o estándar *editions*.
- Definir tablas de grupo o medidas, concretamente la tabla de hechos, también se puede buscar sugerencias del propio diseñador.
- Escoger las medidas (*measures*).
- Elegir las dimensiones existentes y finalizar la creación del cubo. Ver Figura 28.

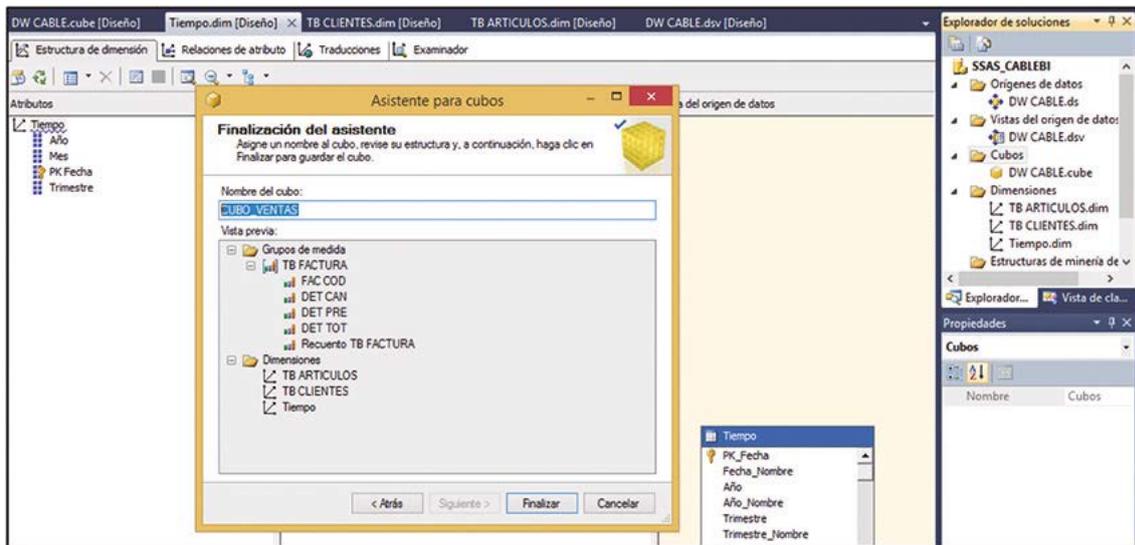


Figura 28. Ventana proceso de creación del cubo OLAP con sus requerimientos.

### 3.6.6 Implementación del cubo

Luego de finalizar con todos los pasos de la construcción del cubo dentro del *Analysis Services*, nos toca implementar el cubo, para lo cual, la herramienta procesa y ejecuta las instrucciones configuradas para generar el cubo correctamente. Ver Figura 29.

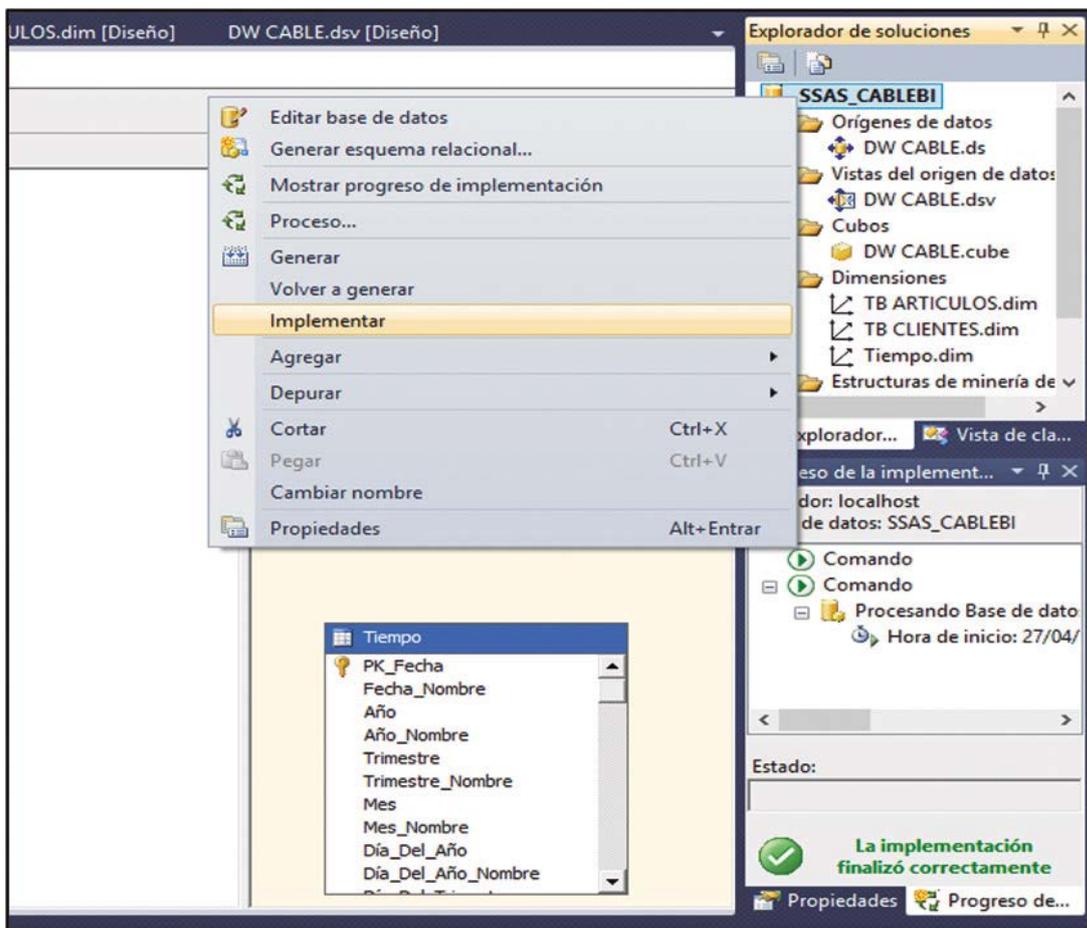


Figura 29. Satisfactoria implementación del cubo OLAP.

Si en este paso no se dieron errores, entonces se puede decir que el cubo fue creado satisfactoriamente y está listo para el análisis estratégico de la data y para presentar los resultados requeridos en los *dashboards* respectivos. Ver Figura 30.

Dimensión	Jerarquía	Operador	Expresión de filtro	Pará...
Tiempo	Jerarquía	Igual	{ 2014-01-01 0...	<input type="checkbox"/>
<Seleccionar dimensi...				

Año	Trimestre	Mes	TOTAL VENTAS
2014-01-01 ...	2014-01-01 ...	2014-01-01 ...	37884.5
2014-01-01 ...	2014-01-01 ...	2014-02-01 ...	37356.3
2014-01-01 ...	2014-01-01 ...	2014-03-01 ...	37530.94
2014-01-01 ...	2014-04-01 ...	2014-04-01 ...	37924.8
2014-01-01 ...	2014-04-01 ...	2014-05-01 ...	37808.15
2014-01-01 ...	2014-04-01 ...	2014-06-01 ...	38900.66
2014-01-01 ...	2014-07-01 ...	2014-07-01 ...	39678.81
2014-01-01 ...	2014-07-01 ...	2014-08-01 ...	38545.59
2014-01-01 ...	2014-07-01 ...	2014-09-01 ...	37878.37
2014-01-01 ...	2014-10-01 ...	2014-10-01 ...	38114.3
2014-01-01 ...	2014-10-01 ...	2014-11-01 ...	38016.08
2014-01-01 ...	2014-10-01 ...	2014-12-01 ...	37307.98

Figura 30. Cubo OLAP Procesado y listo para ser examinado (SSAS).

### 3.6.7 Creación de la tabla tiempos con sus respectivas jerarquías

Uno de los apartados importantes que se tiene en esta etapa de analysis services es la generación de la tabla de tiempos con sus respectivas jerarquías.

En *Microsoft SQL Server Analysis Services*, una dimensión de tiempo es un tipo de dimensión cuyos atributos representan periodos de tiempo, como horas, semestres, trimestres, meses y días. Los periodos en una dimensión de tiempo proporcionan niveles de granularidad basados en tiempo para la elaboración de análisis e informes. Los atributos se organizan en jerarquías y la granularidad de la dimensión de tiempo se determina en gran parte según los requisitos empresariales y de informes de los datos históricos. Por ejemplo, la mayoría de datos financieros y de ventas en las aplicaciones de *Business Intelligence* utilizan una granularidad mensual o trimestral.

Normalmente, los cubos de *Analysis Services* incorporan algún formato de dimensión de tiempo. Un cubo puede incluir más de una dimensión de tiempo o varias jerarquías de esa misma dimensión, según la granularidad de los datos y los requisitos de informes. Sin embargo, no todos los cubos necesitan una dimensión de tiempo. Algunas aplicaciones *OLAP*, como la estimación de costos basada en la actividad, no requieren una dimensión de tiempo, porque la estimación de costos de una dimensión basada en actividades no se basa en el tiempo, sino en la actividad. Ver Figura 31.

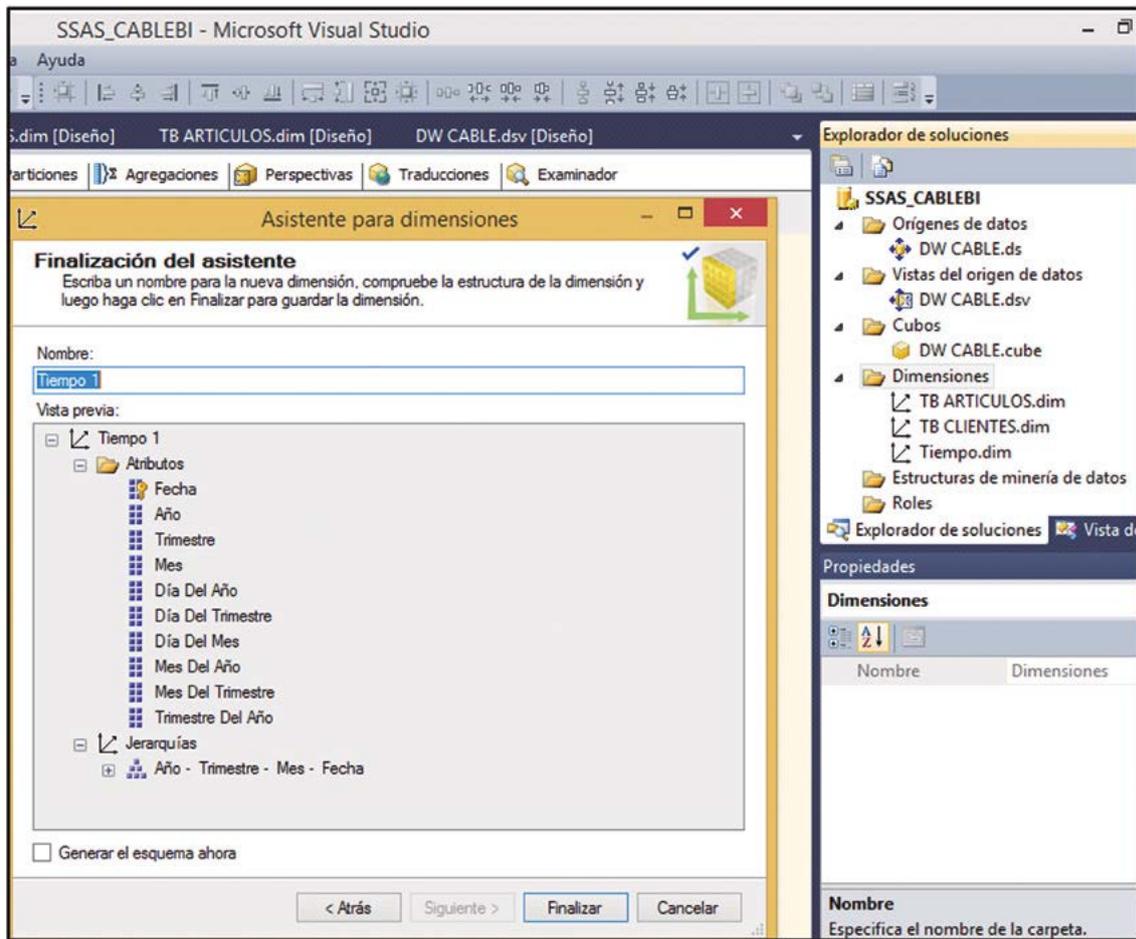


Figura 31. Proceso para la creación de la dimensión tiempo.

Después de crear la dimensión tiempo se deben generar las jerarquías, de acuerdo a la necesidad del negocio. Estas jerarquías permiten mejorar la navegación intuitiva por parte del usuario y es precisamente una de las herramientas que hace inteligentes a las aplicaciones de BI. Ver Figura 32.

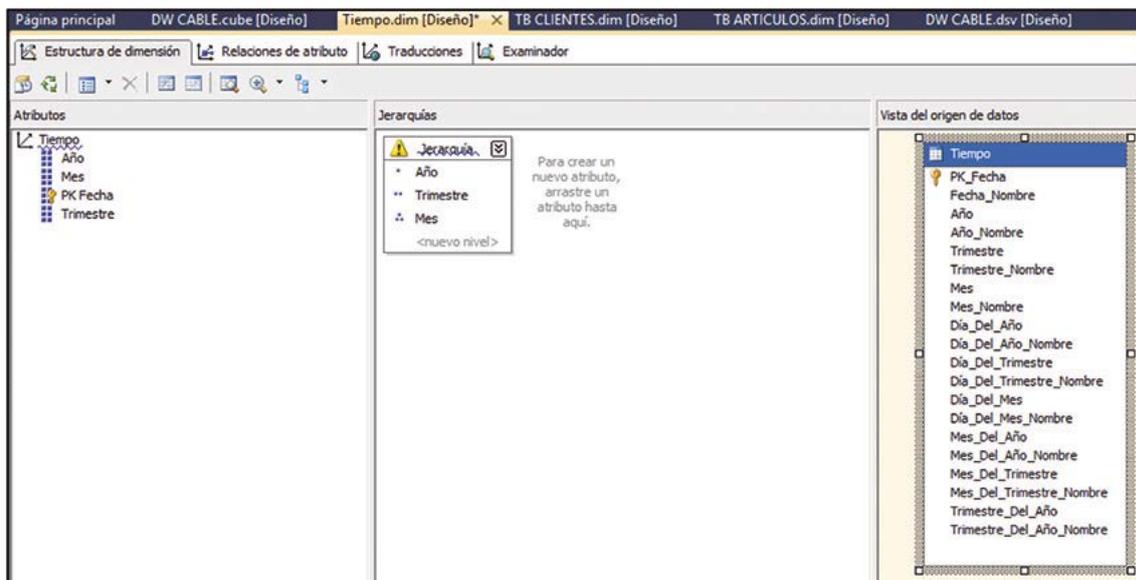


Figura 32. Proceso para la creación de la dimensión tiempo

Las jerarquías se crean arrastrando los atributos de la dimensión tiempo hasta el área de jerarquías, luego, de igual forma, se crean los nuevos niveles jerárquicos según la necesidad y el número de niveles que traiga la dimensión.

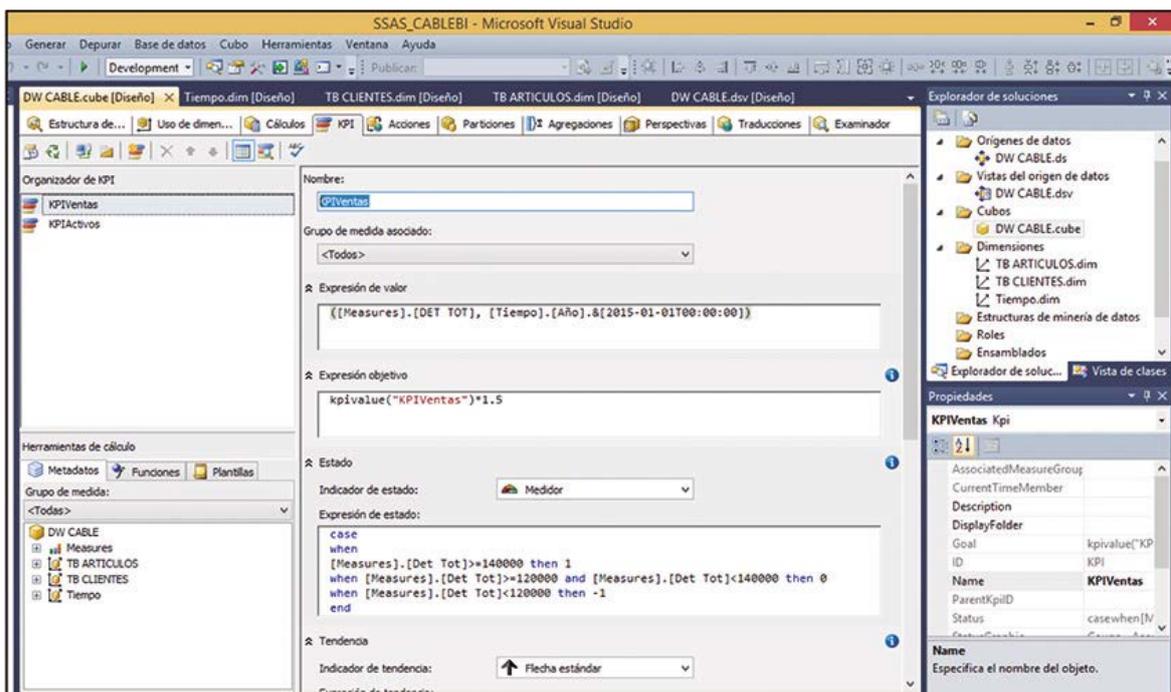
### 3.7 Indicadores claves de desempeño (KPI's)

*El objetivo de este módulo es diseñar e implementar los KPIS en la herramienta de Analysis Services, alineándolo con el estudio del BSC.*

Los *Key Performance Indicators (KPI)* son objetos que se encuentran dentro de la herramienta de SSAS, con los cuales se desarrolló una serie de indicadores de rendimiento claves para evaluar los objetivos del *BSC*, además, son útiles para monitorear el comportamiento y el cumplimiento de cada objetivo estratégico de la empresa. A través de estos indicadores se pueden tomar decisiones efectivas reduciendo el índice de incertidumbre sobre los efectos o resultados de cada decisión.

Los indicadores permiten visualizar el estado actual de cada objetivo estratégico y la tendencia con respecto a valores históricos comparados en un tiempo específico. En tal sentido, los *KPI'S* deben ser creados técnicamente con fundamento teórico y estratégico.

Los *KPI'S* se crean luego de que el cubo o los cubos *OLAP*, en el proyecto del *SSAS*, hayan sido creados satisfactoriamente. Para crear los *KPI'S*, se realiza desde la vista *KPI - Nuevo KPI del SSAS*. Ver Figura 33.



**Figura 33.** Proceso de cubo *OLAP* terminado y listo para ser examinado SSAS.

Los componentes principales en el diseño son:

- Nombre del *KPI*: la descripción que se ponga aquí es la que se usará en el resto de parámetros y código *DMX* que servirá para configurar el *KPI*.
- Grupo de medida de asociados: se escoge el componente en el cual se encuentra la medida o grupos de medidas, dimensión o la tabla de hechos.
- Expresión Valor (*KPIValue*): se pone una constante o una expresión *DMX* que será la referencia para la medición del *KPI*.
- Expresión Objetivo (*KPIGoal*): igual que con el *KPIValue* se puede poner una constante, una variable o una expresión *DMX*.
- Estado: en este espacio, seleccionamos tanto el indicador de estado que puede ser un medidor, semáforo, etc., junto con la expresión de estado, generalmente *DMX*, para controlar el *KPI*, el cual indica tres estados distintos e independientes posibles que pueden ser insuficiente, aceptable y excelente. Va a depender del criterio organizacional.

```

case
when
[Measures].[Det Tot]>=kpigoal("KPIVentas") then 1
when [Measures].[Det Tot]>=kpigoal("KPIVentas")-20000 and [Measures].[Det Tot]<kpigoal("KPIVentas") then 0
when [Measures].[Det Tot]<kpigoal("KPIVentas")-20000 then -1
end
    
```

Código *DMX* expresado en el campo Estado del *KPI*.

- Tendencia: este es otro de los elementos importantes que se debe definir en el *KPI* para configurarlo correctamente. Aquí definimos primero el indicador de tendencia y luego la expresión de tendencia que generalmente debe ser un código *DMX*, con las condiciones de acuerdo a lo que se quiera medir. A continuación, se sitúa un ejemplo.

Código *DMX* válido para la definición de la tendencia del *KPI*.

```

case
When
((KpiValue("KPIVentas") -
(KpiValue("KPIVentas"),
ParallelPeriod
([Tiempo].[Año].[Año],1,
[Tiempo].[Año].CurrentMember))) /
(KpiValue("KPIVentas"),
ParallelPeriod
    
```

```

    ((Tiempo).[Año].[Año],1,
     [Tiempo].[Año].CurrentMember))) >=1
Then 1
When(( KpiValue("KPIVentas") -
      (KpiValue ( "KPIVentas" ),
       ParallelPeriod
        ([Tiempo].[Año].[Año],1,
         [Tiempo].[Año].CurrentMember))) /
      (KpiValue("KPIVentas"),
       ParallelPeriod
        ([Tiempo].[Año].[Año],1,
         [Tiempo].[Año].CurrentMember))) <1
Then -1
Else 0
End

```

### 3.7.1 Lenguaje *MDX* para consultas multidimensionales

Es un lenguaje de consulta usado para sacar datos de una base de datos multidimensional. Más concretamente, *MDX* se usa para consultar datos de una base de datos OLAP con Analysis Services y otras herramientas que soporten este lenguaje (Golfarelli et al., 2012). *MDX* ayuda a crear Reportes con más facilidad que con *SQL*, ya que con *SQL* tenemos que ir especificando todas las relaciones de las tablas para poder sacar un dato en concreto. En cambio, en *MDX* ya está todo relacionado *a priori*.

En *MDX* encontramos muchas mejoras en comparación con el lenguaje de consulta *SQL*. En el lenguaje *MDX* no es necesario especificar los campos de las tablas relacionados, porque cuando se genera el cubo define cuáles serán los metadatos y por tanto las jerarquías y sus relaciones.

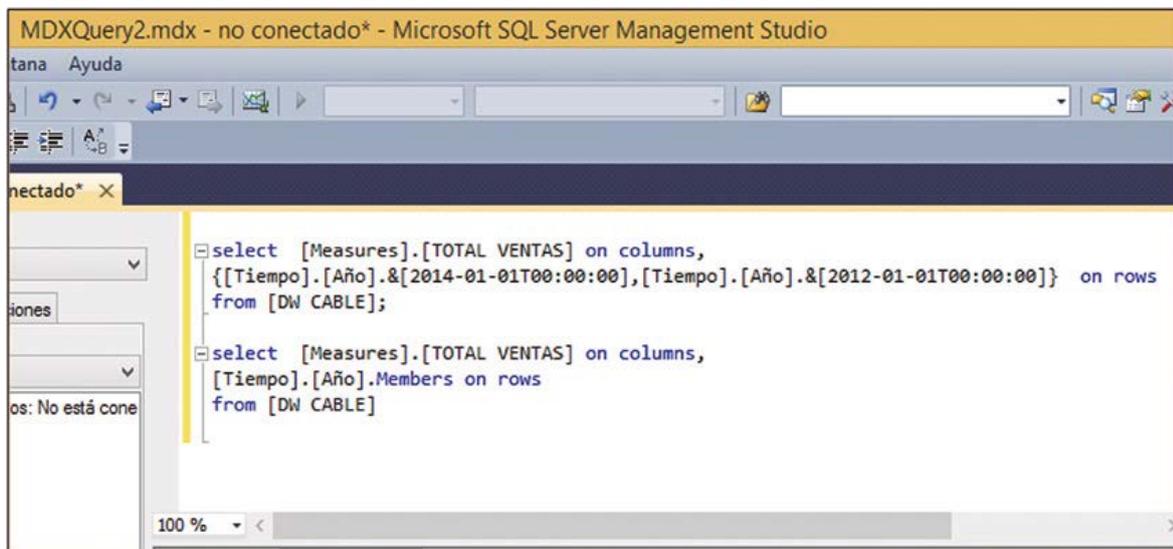
*MDX* es posicional, ya que las agregaciones ya están calculadas y solo se tiene que especificar en qué posición se encuentra el dato que se quiere sacar, además, permite ver los resultados anteriores y siguientes al especificado porque cuenta con las propiedades *prevmember* y *nextmember*.

*MDX* es un lenguaje referenciado, *SQL* no es referenciado y requiere hacer muchos cambios. Al contrario, en *MDX* tiene referenciados los datos y no precisa usar las funciones de agregado ya que cada cálculo está guardado en una celda y solo se tiene que especificar la referencia a ella. En caso que se desee sacar el total de un grupo de registros, no es necesario porque en *MDX* este valor ya se encuentra en una celda, solo se debe referenciar esta celda para mostrar.

La consulta de *MDX* (Expresiones multidimensionales) básica es la instrucción *SELECT*: Esta consulta es utilizada con más frecuencia en *MDX*. Es importante comprender cómo una instrucción *MDX SELECT* debe especificar un conjunto de resultados. Se debe conocer la sintaxis de la instrucción *SELECT* y cómo crear una consulta simple. A partir de aquí, tendrá un conocimiento sólido de cómo utilizar *MDX* para realizar consultas de datos multidimensionales. Ver Figura 34, 35.

```
[ WITH <SELECT WITH clause> [ , <SELECT WITH clause> ... ] ]
SELECT [ * | ( <SELECT query axis clause>
    [ , <SELECT query axis clause> ... ] ) ]
FROM <SELECT subcube clause>
[ <SELECT slicer axis clause> ]
[ <SELECT cell property list clause> ]
```

**Figura 34.** Sintaxis SQL básica con MDX en SQL Server.  
**Fuente.** (Microsoft, 2016)



**Figura 35.** Ejemplo de consulta SQL usando lenguaje MDX.

### 3.7.2 Sintaxis básica del lenguaje MDX

Las expresiones multidimensionales (MDX) tienen varios elementos utilizados por la mayor parte de las instrucciones, o bien que influyen en las mismas [4]. Ver tabla 7.

<sup>4</sup> Microsoft, <https://docs.microsoft.com/es-es/sql/mdx/mdx-syntax-elements-mdx>

<b>Término</b>	<b>Definición</b>
Identificadores	Son los nombres de objetos, como cubos, dimensiones, miembros y medidas.
Tipos de datos	Definen los tipos de datos que contienen las celdas, las propiedades de miembro y las propiedades de celda. <i>MDX</i> solo admite el tipo de datos OLE VARIANT. Para obtener más información acerca de la coerción, la conversión y la manipulación del tipo de datos VARIANT, vea el tema sobre VARIANT y VARIANTARG en la documentación de Platform SDK.
Expresiones (MDX)	Las expresiones son unidades de sintaxis que <i>Microsoft SQL Server Analysis Services</i> puede resolver en valores (escalares) u objetos. Las expresiones incluyen funciones que devuelven un solo valor, una expresión de conjunto, etc.
Operadores	Son elementos de sintaxis que funcionan con una o más expresiones <i>MDX</i> simples para crear expresiones <i>MDX</i> más complejas.
Funciones	Las funciones son elementos de sintaxis que toman cero, uno o más valores de entrada y devuelven un valor escalar o un objeto. Algunos ejemplos son la suma función para agregar varios valores, el miembro función para devolver un conjunto de miembros de una dimensión o un nivel, y así sucesivamente.
Comentarios	Son fragmentos de texto insertados en instrucciones o scripts <i>MDX</i> para explicar el objetivo de la instrucción. <i>Analysis Services</i> ejecuta los comentarios.
Palabras clave reservadas	Son palabras que se han reservado para ser usadas con <i>MDX</i> y que no deberían utilizarse para nombres de objeto que se empleen en instrucciones <i>MDX</i> .
Miembros, tuplas y conjuntos	Son conceptos básicos de los datos multidimensionales, cuya comprensión es imprescindible antes de crear una consulta <i>MDX</i> .

**Tabla 7.** Términos y definiciones de la sintaxis del lenguaje *MDX*.

**Fuente:** Microsoft: <https://docs.microsoft.com/es-es/sql/mdx/mdx-syntax-elements-mdx>

Convención	Uso
<i>cursiva</i>	Indica argumentos proporcionados por el usuario de la sintaxis de <i>MDX</i> .
(barra vertical)	Separa los elementos de sintaxis con corchetes o llaves. Solo se puede seleccionar uno de los elementos.
[ ] (corchetes)	Indica los elementos de sintaxis opcionales. No escriba los corchetes.
[,] ...n	Indica que el elemento anterior se puede repetir muchas veces. En algunos casos, los elementos se separan mediante comas.
<etiqueta >:: =	Indica el nombre de un bloque de sintaxis. Esta convención se utiliza para agrupar y asignar etiquetas a porciones de sintaxis largas o una unidad de sintaxis que se puede utilizar en más de un lugar dentro de una instrucción. Cada ubicación en la que se puede utilizar el bloque de sintaxis se indica con la etiqueta incluida entre corchetes angulares: <etiqueta >.

**Tabla 8.** Conversiones y usos de signos y símbolos en *MDX*.

**Fuente:** Microsoft: <https://docs.microsoft.com/es-es/sql/mdx/mdx-syntax-elements-mdx>

### 3.7.3 Usar expresiones de dimensiones

Las expresiones de dimensiones y jerarquía se usan normalmente al pasar parámetros a funciones *MDX* (Expresiones multidimensionales) para devolver miembros, conjuntos o tuplas de una jerarquía.

### 3.7.4 Expresiones de dimensiones

Una expresión de dimensión contiene un identificador de dimensión o una función de dimensión. Las expresiones de dimensiones raramente se utilizan solas. Lo normal es que, en lugar de ello, especifique una jerarquía en una dimensión. La única excepción se da cuando se trabaja con la dimensión Measures, que no tiene ninguna jerarquía. A continuación, se muestra un miembro calculado que utiliza la expresión *[Measures]* junto con las funciones *Members* y *Count ()* para devolver el número de miembros de la dimensión Measures: [5]

```
WITH MEMBER [Measures].[MeasureCount] AS
COUNT([Measures].MEMBERS)
SELECT [Measures].[MeasureCount] ON 0
FROM [Adventure Works]
```

<sup>5</sup> Microsoft: <https://docs.microsoft.com/es-es/sql/mdx/using-dimension-expressions>

Un identificador de dimensión aparece como `Dimension_Name` en la notación de BNF, que se utiliza para describir las instrucciones *MDX*.

### 3.7.5 Expresiones de jerarquías

Del mismo modo, las expresiones de jerarquía contienen un identificador de jerarquía o una función de jerarquía. En el ejemplo siguiente mostramos el uso de la expresión de jerarquía `[Date].[Calendar]`, junto con las funciones `.Levels` y `.Count`, para devolver el número de niveles de la jerarquía `Calendar` de la dimensión `Date`:

```
WITH MEMBER [Measures].[CalendarLevel-
Count] AS
[Date].[Calendar].Levels.Count
SELECT [Measures].[CalendarLevelCount] ON 0
FROM [Adventure Works]
```

Estas expresiones son utilizadas con mayor frecuencia en situaciones de jerarquía, junto con la función `Members`, para devolver los miembros de una jerarquía. A continuación, demostramos esto para devolver los miembros de `[Date].[Calendar]` en el espacio de filas:

```
SELECT [Measures].[Internet Sales Amount] ON 0,
[Date].[Calendar].MEMBERS ON 1
FROM [Adventure Works]
```

### 3.7.6 Bases de datos y lenguaje *NoSQL* (*No only SQL*)

Las estructuras de almacenamiento *NoSQL* aparecen en el momento que las redes sociales invadieron el *Internet* con aplicaciones y plataformas web y móviles de consumo masivo. Por otro lado, surgió la computación en la nube, *Big Data*, inteligencia artificial, *Internet de las Cosas* (*IoT del inglés Internet of Things*) y las grandes cantidades de datos producto de sensores inteligentes. Las bases de datos *NoSQL* rompen las características de las bases de datos *SQL* de tipo relacional, persistente, recurrente y *ACID* (*atomicity, consistency, isolation and durability*). Esta disrupción permite al *NoSQL* trabajar con grandes cantidades de datos y la posibilidad de escalar a miles de nodos distribuidos en el *Internet* o en la nube.

Por lo tanto, esta escalada del volumen de información y variedad provocó una nueva necesidad de gestión, para consultar, extraer, ordenar, transformar y analizar estos datos con rapidez y versatilidad. Las bases de datos *NoSQL* se consumen en plataformas de Inteligencia Artificial, Inteligencia de Negocios y analítica de datos en general, usando aprendizaje automático (*machine learning*) o aprendizaje profundo

(*Deep learning*) (Sarasa, 2019). Estos nuevos requerimientos, rompen con los esquemas tradicionales de entidad-relación y las estructuras tipo tabla con columnas y filas de tipo relacional.

Para cambiar al nuevo modelo *NoSQL* se adoptaron algunos formatos principales de clasificación:

1. **clave-valor** como estándar de almacenamiento de datos no estructurados usados en Redis, *Dynamo*, *Cassandra*, *Voldemort*, *HBase*. o *MongoD* para guardar en archivos BSon, *JSon* usados en *Google* o *Amazon* (Martín *et al.*, 2013). El formato tiene la siguiente forma tipo diccionario emparejados {clave: valor}; por ejemplo:  

```
{nombre: "Juan", edad:20, dirección: "Av. 10 de agosto",
amigos: [
  {nombre: "Pedro", edad:22},{nombre: "Diego", edad: 25},{nombre:
"Luis", edad: 21}
]}
```
2. **documentales** mucho más escalables que la clave valor, basadas en archivos JSon, XML. Estas bases de datos gestionan datos semiestructurados, por ejemplo, *MogoDB*, *10gen* o *CouchDB*.
3. **bases de datos orientadas a grafos** como *Neo4j*, *InfoGrid* y tienen gran versatilidad, al trabajar con datos complejos con altos niveles de interrelación (WHITE PAPER, 2015), trabajan bien con archivos de *Hadoop Distributed File System (HDFS)*.
4. **orientadas a columnas o tabulares**, gestionadas por *HBase* de *Apache* para *Facebook*, *BigTable* de *Google*, *LevelDB* o *Hypertable*. Este tipo de bases de datos trabajan con sistema de ficheros en ambiente distribuido, escalable y portátil como *Hadoop* o los *RDD (Resilient Distributed Dataset)* de *Apache Spark* (WHITE PAPER, 2015).

En términos generales, estas bases de datos garantizan alta disponibilidad a fallos, baja latencia, escalabilidad y flexibilidad, con capacidad para trabajar en ambiente distribuido basados en clústeres. Sin embargo, existen algunas dificultades actualmente en estas estructuras que es preciso puntualizar y que citamos a continuación:

1. Cuando el volumen de los datos crece muy rápidamente en momentos puntuales pueden llegar a superar el Terabyte de información, poniendo en conflictos al hardware.
2. Cuando la escalabilidad de la solución relacional no es viable tanto a nivel de costes como a nivel técnico.
3. Cuando tenemos elevados picos de uso del sistema por parte de los usuarios en múltiples ocasiones.
4. Cuando el esquema de la base de datos no es homogéneo, es decir, cuando en cada inserción de datos, la información que se almacena puede tener campos distintos.

### 3.7.7 Tolerancia a Particiones (*Partition Tolerance*)

Entendida como la habilidad de un sistema de tener diferentes regiones o divisiones lógicas en la red, y de ser capaz de seguir funcionando aunque una de estas partes quede inaccesible durante un tiempo (Benymol & Abraham, 2020). A partir de este concepto, se genera una clasificación de los sistemas de gestión de bases de datos *NoSQL*, que difícilmente cumplen con los tres elementos del CAP, pero sí con dos por lo menos, como muestra el siguiente cuadro. Ver Figura 36.

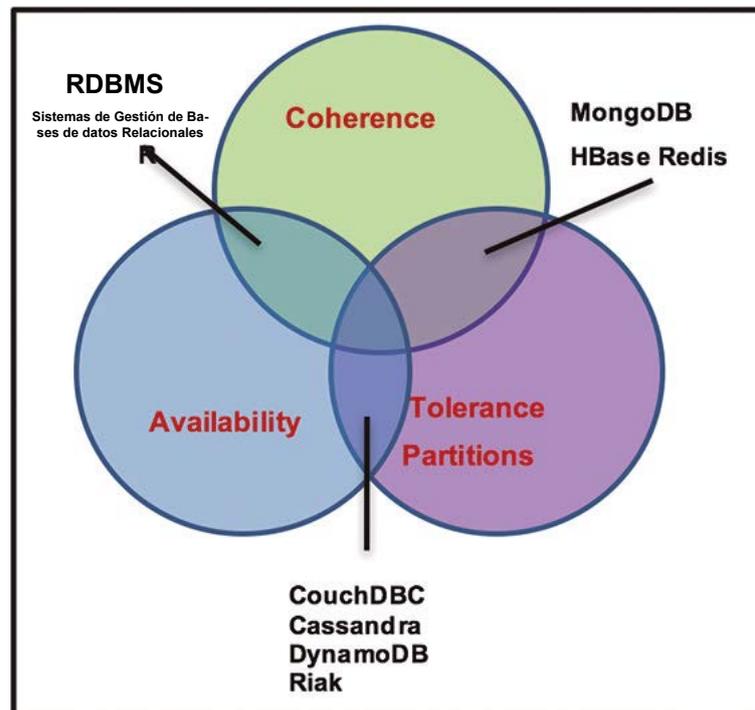


Figura 36. Clasificación de los sistemas de gestión de bases de datos *NoSQL*.

1. **El Teorema CAP:** este teorema de coherencia, disponibilidad y tolerancia a particiones (*del inglés CAP*) se ha adoptado por varias organizaciones de internet y por la comunidad *NoSQL*.
2. **Coherencia:** en sistema distribuido, habitualmente se dice que se encuentra en un estado consistente, si y solo si, después de una operación de escritura, todas las operaciones de lectura posteriores son capaces de ver las actualizaciones desde la parte del sistema que están leyendo (Gómez, 2014).
3. **Disponibilidad (*Availability*):** la alta disponibilidad se produce cuando el sistema ha sido diseñado e implementado de modo que se pueda continuar operando (lecturas, escrituras), incluso después de que un nodo quede indisponible, o de que algunas partes de hardware tengan que ser retiradas, debido a errores o actualizaciones (Benymol & Abraham, 2020).



# CAPÍTULO IV

## Reportes gerenciales



## 4.1 Tableros de control (*Dashboards*)

*El objetivo de estudiar los tableros de control (del inglés dashboard), es describir las técnicas para diseñar e implementar reportes gerenciales de alto nivel estratégico, que brinden soporte de confianza para tomar decisiones correctas.*

En este nivel decimos que la parte más compleja ya ha pasado. El desarrollo de los tableros de control es el último escalón en el proceso de BI. Sin embargo, del diseño del *dashboard* depende el éxito de todos los procesos anteriores. Dicho de otro modo, si la presentación y diseño de los dashboards cumplen con las expectativas y requerimientos del cliente, el proyecto BI ha tenido éxito. Caso contrario, todo lo realizado deberá someterse a un nuevo estudio o rediseño para satisfacer los requerimientos y expectativas del cliente.

Por tal motivo, el proceso de la construcción de aplicaciones BI es recomendable que se realice aplicando la metodología ágil o *SCRUM*. Especialmente, para desarrollar por medio de *sprints* e incrementos o etapas. Comenzando desde la más crítica e importante hasta terminar con la menos valiosa de forma colaborativa con los expertos de la gestión empresarial.

Si hasta aquí todo está bien, entonces se pueden diseñar los dashboards con las distintas herramientas compatibles de *BI*, desde aplicaciones libres o con licencia. En esta última, contamos con diseñadores como el mismo *MS Excel* y el *Reporting Services* de *SSRS*, *Power BI*, *Tableau*, entre otras. Como ejemplo se presentarán algunos modelos realizados desde Reporting Services.

### 4.1.1 Creación del proyecto del servicio de reportes

Para crear un informe en *SQL Server*, primero, debe diseñar un proyecto de servidor de informes donde guardará el archivo de definición de informe (.rdl) y cualquier otro archivo de recursos que este requiera. Luego, creará el archivo de definición de informe real y definirá un origen de datos, un conjunto de datos y establecerá el diseño del informe. Cuando se lo ejecuta, los datos reales se recuperan y combinan con el diseño y luego se representan en pantalla, desde donde se pueden exportar, imprimir o guardar (Microsoft, 2022).

Lo que se debe tomar muy en cuenta es que cada elemento u objeto que se incluya en el *dashboard* debe cumplir con los requerimientos y expectativas del cliente, de tal forma que, este sea amigable y flexible en su manejo, pero muy ilustrativo en la información y en el conocimiento de los resultados visualizados. Esto, para garantizar a los mandos estratégicos para que las decisiones tomadas sean efectivas y generen un alto rendimiento productivo.

Para iniciar, se debe comenzar creando el proyecto desde Microsoft Visual Studio 2012, de donde se selecciona la opción del proyecto de Servidor de Informes. Ver Figura 37.

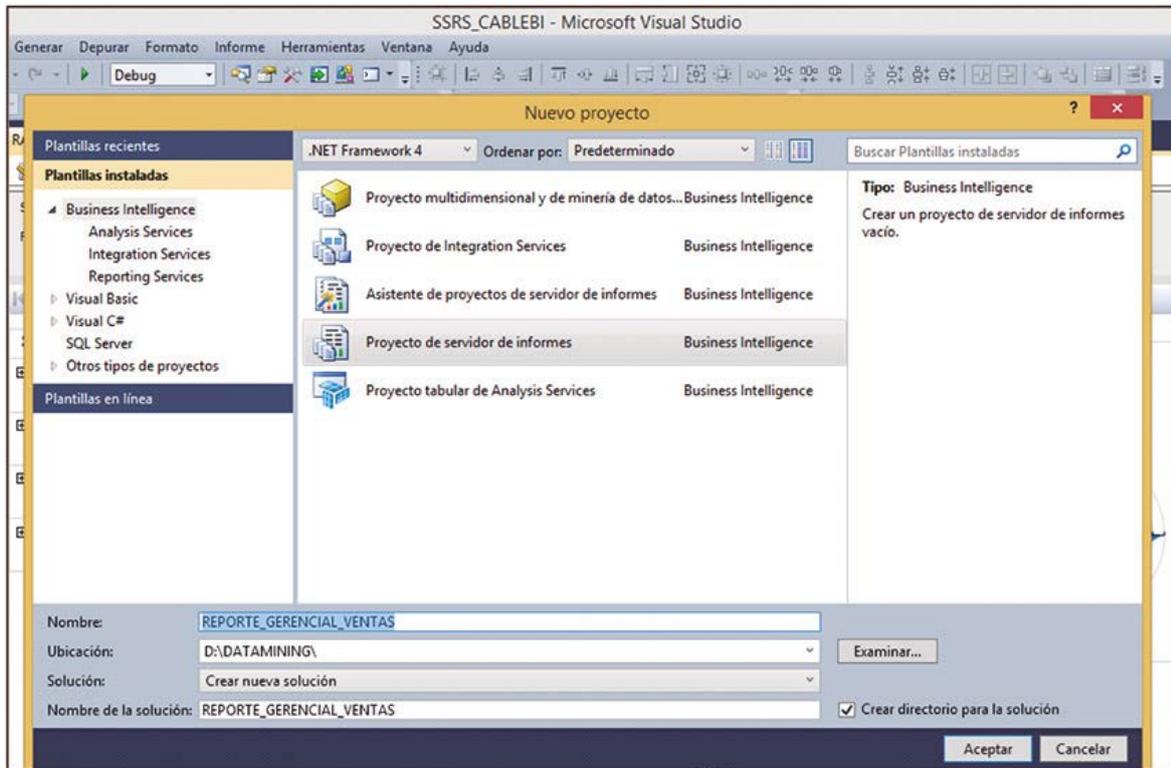


Figura 37. Ventana para iniciar un proyecto de SSRS con Tableros de Control

Los objetos y configuraciones realizadas en el proceso de desarrollo del tablero de control se detallan a continuación.

#### 4.1.2 Definir el origen de datos

Después de añadir un informe al proyecto en desarrollo, necesitamos definir un origen de datos, que es la información de conexión que el informe utilizará. Esto ayuda a cargar los datos procedentes de una base de datos relacional, no relacional, multi-dimensional u otro tipo de almacenamiento. Luego, se debe agregar nuevo informe, esto se hace desde el explorador de soluciones en el cual se debe seleccionar un reporte como nuevo elemento y definir su nombre.

Propiedades del origen de datos: en esta fase se debe definir el nombre del origen de datos, el tipo del origen de datos (*SQL Server Analysis Services*) y la cadena de conexión. Para el ejemplo se usa el origen de datos del cubo OLAP SSAS\_CABLEBI (DW). Ver Figura 38.

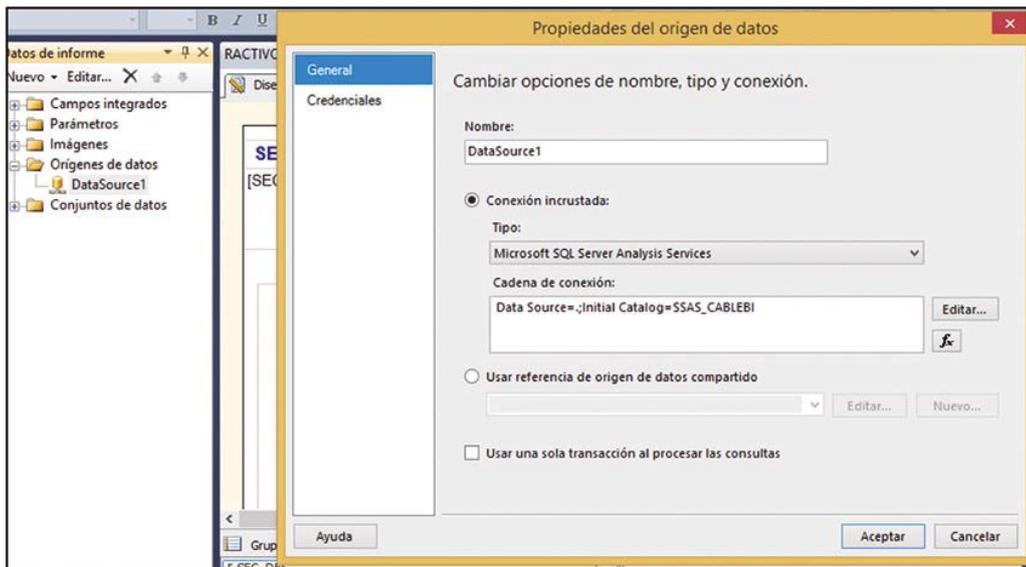


Figura 38. Configuración del origen de datos para el reporte.

#### 4.1.3 Definir un conjunto de datos para el informe

Después de definir el origen de datos, necesitamos definir un conjunto de datos. En *Reporting Services*, los datos que se utilizan para los informes derivan de un conjunto de datos, el cual contiene un puntero a un origen de datos y la consulta que empleará para el informe, así como campos y variables calculados. Podemos usar el Creador de consultas dentro del Diseñador de informes para construir la consulta. En este libro, se creará una consulta que recupere información sobre pedidos de ventas de la base de datos.

En el cuadro, se define la vista de datos con la consulta *SQL* extendida con lenguaje *DMX*, con los filtros y en el orden respectivo para elegir la data necesaria que debe llenar el tablero de control. Ver Figura 39.

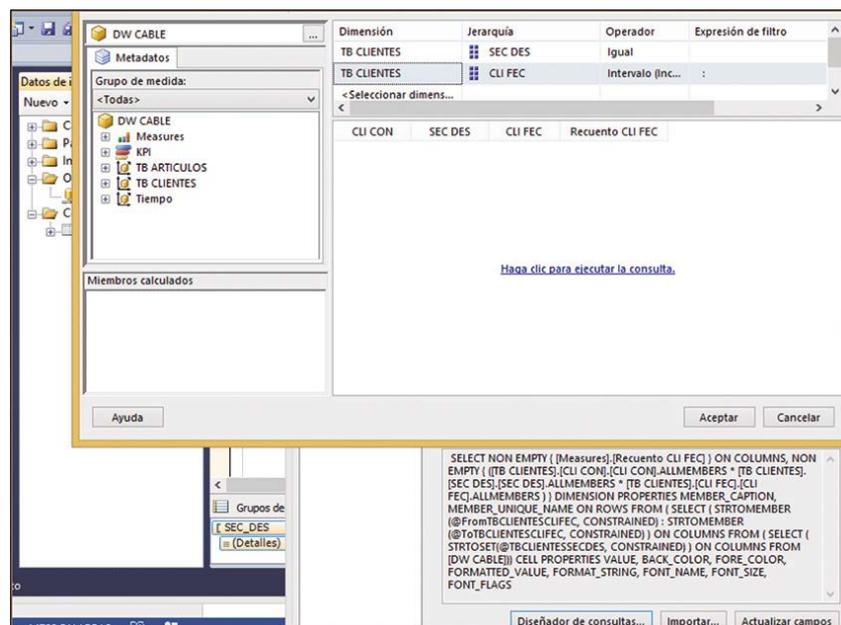


Figura 39. Definición del conjunto de datos para crear la consulta del reporte.

#### 4.1.4 Agregar la Tabla Pívor al informe del Reporting Services SSRS

Una vez que se ha completado con la configuración inicial de los datos para el informe, se procede a construir la tabla pívot de datos y los gráficos estadísticos dinámicos. Para esto es recomendable poner más de un gráfico para que la interpretación del conocimiento sea fluido, entendible y explicable, para el personal estratégico.

Después de definir un conjunto de datos, vamos a comenzar a diseñar el *dashboard*. El diseño del informe se crea arrastrando y colocando en la superficie de diseño regiones de datos, cuadros de texto, imágenes y otros elementos que se desean incluir en el informe.

Los elementos que contienen filas de datos repetidas procedentes de conjuntos de datos subyacentes se denominan regiones de datos. Un informe básico solo contendrá una región de datos, pero puede agregar más si, por ejemplo, desea agregar un gráfico al informe de tabla. Después de agregar una región de datos, puede agregar campos a la misma. Ver Figura 40.

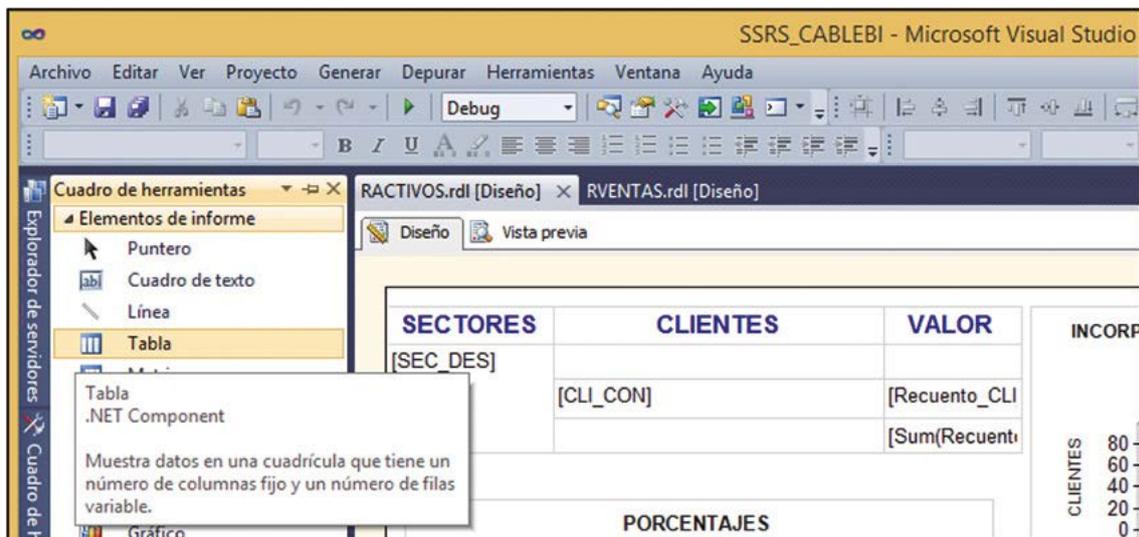


Figura 40. Creación de la tabla pívot para mostrar el tablero de control final.

#### 4.1.5 Diseño del informe georreferencial con mapas, gráficos agrupados (*clústeres*)

Para mejorar la presentación de los *dashboards*, los mapas geográficos tipo ESRI son una excelente alternativa. Para esto, primero descargamos los datos geoespaciales de la web, luego agregamos al proyecto de diseño y configuramos el mapa ESRI por regiones. Finalmente, cada región se enlaza con los datos y algunos campos del informe, según la granularidad y los totales calculados (*Sales Orders*). En esta etapa podemos dar formato a los campos de moneda, de fecha y cambiar los estilos, los encabezados de columna, la metodología y los procesos de implementación; estos procesos se amplían con mayor detalle en (Rodríguez-López, 2017). El resultado puede verse en la Figura 41a y 41b.

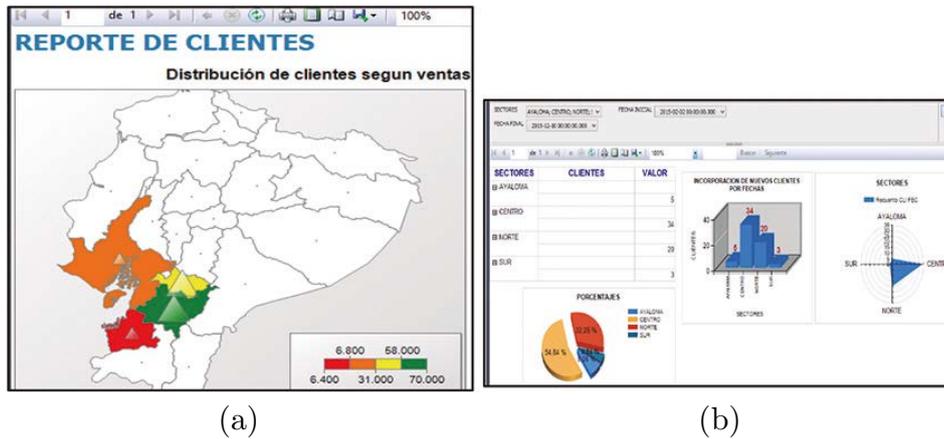


Figura 41. Formato final del informe (a) Mapa (b) dashboard.

El último paso es realizar la publicación del reporte en un servidor de informes, en modo conectado, usando los servicios de *sharepoint*, *SQL Server Data Tools*, herramientas de reportes disponibles en la nube o simplemente en modo local sin conexión [6].

#### 4.1.6 Informes para investigación científica con *JSON*, *Csv* y *Python* o *Spark*.

Las nuevas herramientas de TCE, en la actualidad, también permiten extraer datos de los distintos sitios web, de las redes sociales y otras fuentes. Luego de extraer estos datos, hay una serie de herramientas para analizarlos usando archivos JSON con *dataframes*, *graphframes* y *datasets* [7], donde se realiza una serie de transformaciones y consultas para su visualización. Este proceso se describe a continuación, utilizando lenguaje *Python* de Anaconda, para extraer datos de *twitter* por medio de un filtro denominado *keywords* (palabras claves).

El formato estándar de Java Scrip Objet Notation JSON consiste en archivos de texto para intercambio rápido de información formados por líneas, adoptado actualmente por el estándar XML. Por otro lado, los archivos de valores separados por comas (del inglés *comma-separated values* CSV) son más conocidos en herramientas de cálculo en ambiente propietario y libre; estos tienen una estructura por columnas en forma de tabla. Sin embargo, en el campo de la investigación o de la ciencia, estos archivos son utilizados en un alto porcentaje por su estandarización y facilidad de uso.

#### 4.1.7 Extracción de datos de *Twitter*

Para trabajar con estas herramientas y una red social, lo primero es conectarse y generar el token o timbrado.

<sup>6</sup> Microsoft, servidor de informes: <https://docs.microsoft.com/es-es/sql/reporting-services/report-server-sharepoint/reporting-services-report-server-sharepoint-mode>

<sup>7</sup> Databricks Community: <https://docs.databricks.com/spark/latest/dataframes-datasets/introduction-to-dataframes-python.html>

```

from tweepy import Stream
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener
import sqlite3
from sqlite3 import Error
import json
import csv

# URL para crear con: Apps: https://apps.twitter.com/
app_id = "pz1T1zCqhUBtJ7CEtMdS1qHb8"
app_secret = "yMpg1YcrmF641YmYCEbk1GygZBAgTCVHzcOQxPuXDs2uUa4u9k"
token_access = "892408714011058176-WzKZ4uTCYG5rujrEZXDuvUI7XC5RCx9"
token_secret = "LbpxV8LkmLX4nqaeBS5TXmQcvqOSbYpT5i25Os61LpOD7"
#Archivo CSV en sqlite
PATH_DATABASE = "D:\base_clima\desastre_tweet.db"
Luego se procede a crear la base de datos en SQLite
def createTable(db_file):
    try:
        conn = sqlite3.connect(db_file)
        c = conn.cursor()
        c.execute('CREATE TABLE tweet (tweet_text text, username text)')
        conn.commit()
    except Error as e:
        print(e)
    finally:
        conn.close()
createTable(PATH_DATABASE)

```

Luego, se crea un Scrip, para capturar los datos del Twitter.

```

Import re
class TwitterStream(StreamListener):
    def __init__(self, connection):
        self.conn = connection
    def on_data(self, data):
        # Procesar los datos para guardarlos
        json_python = json.loads(data)
        tweet_text = json_python["text"]
        tweet_new = re.sub(r'^\w', ' ', tweet_text)
        user_name = json_python["user"]["screen_name"]
        query_text = "INSERT INTO clima2 (tweet_text, username)
            VALUES ({}{})" format("'+tweet_new+'", "'"+user_name+'")"
        c = self.conn.cursor()
        c.execute(query_text)
        self.conn.commit()
        return(True)
    def on_error(self, status):
        print (status)

```

Finalmente, se envían los parámetros de llamado. Es mejor utilizar por separado o de uno en uno los criterios de filtrado, para extraer los datos que se están investigando de la métrica de tendencias. Por ejemplo, primero solo *keyword* = “fire in California”, luego *keyword* = “hurricane Maria” y *keyword* = “Terremoto en Mexico”. También es recomendable no usar caracteres extraños en los *Keyword*, como tildes, dos puntos, rayas oblicuas, entre otros símbolos más, que es preciso evitar. Los comentarios van precedidos del símbolo #.

```

auth = OAuthHandler(app_id, app_secret)
auth.set_access_token(token_access, token_secret)
conn = sqlite3.connect(PATH_DATABASE)
twitterStream = Stream(auth, TwitterStream(conn))
twitterStream.filter(track=["fire in California", "hurricane Maria",
                           "Terremoto en Mexico", "California fires"])

```

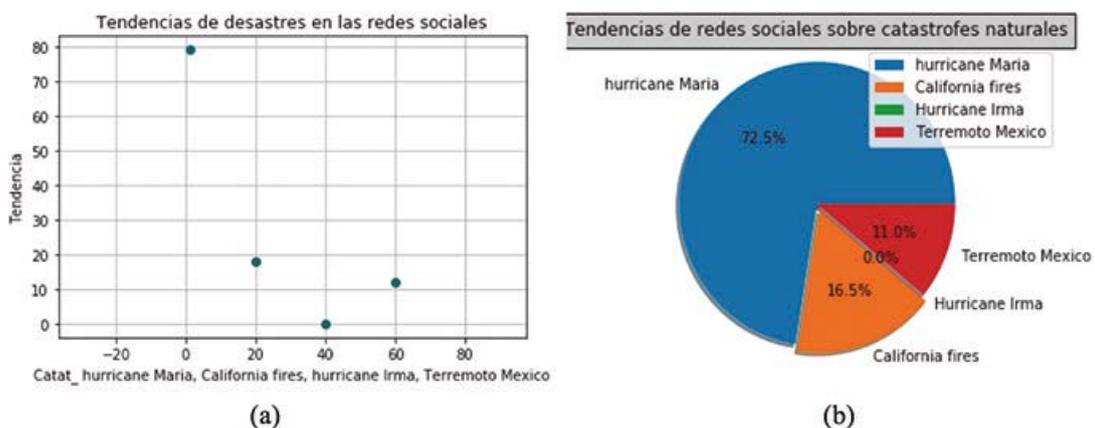
Para llamar el archivo.csv de datos y hacer las consultas, se puede utilizar el siguiente código.

```

x=[]
y=[]
with open('D:\base_clima\clima3.csv','r') as f:
    l=f.read().splitlines()
    for l2 in l:
        df=l2.split(',')
        x.append([df[0],df[1]])
    for l3 in sorted(x):
        y.append(l3[1])
        res = [k for k in y if 'Hurricane Maria' in k] #filtro
        z.append(len(res)) # almacenar los totales el vector -> z [ ]
print ('La frecuencia es: ', z)

```

**Resultado:** La frecuencia es: [79, 18, 0, 12]



**Figura 42.** Datos de Twitter. Catástrofes naturales (a) lineal x, y (b) pastel.

Este código servirá para hacer este tipo de gráficos en *Python*.

```

from pylab import *
import matplotlib.pyplot as plt
import numpy as np
Series = ["hurricane Maria", "California fires", "Hurricane Irma", "Terremoto Mexico"]
expl = (0, 0.05, 0, 0)
pie(z, explode=explode, labels=impr, autopct='%1.1f%%', shadow=True)
plt.title("Tendencias de redes sociales sobre catástrofes naturales",
bbox={"facecolor": "0.8", "pad": 5})
plt.legend()
plt.hold(True)
plt.grid()
plt.axis('equal')
plt.show()

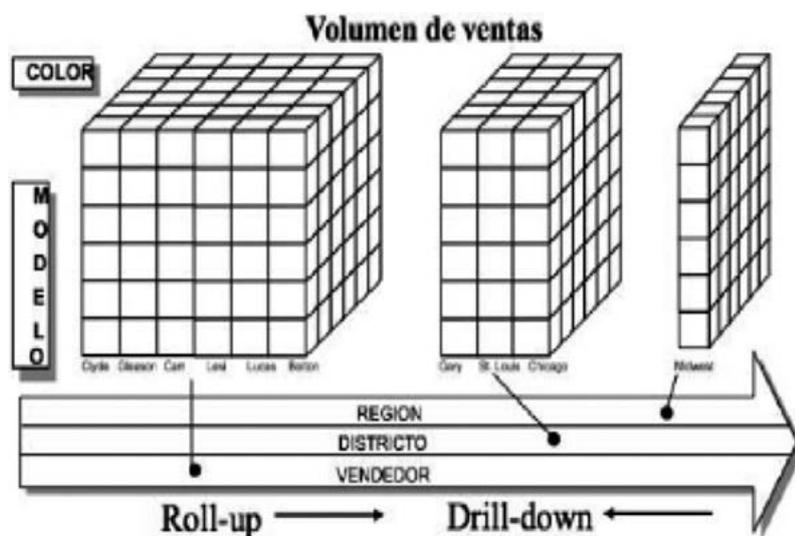
```

## 4.2 Herramientas avanzadas de exploración OLAP

*El objetivo de este apartado es describir y explicar paso a paso el uso de técnicas avanzadas para construir tableros de control intuitivos, amigables y flexibles con un mejor nivel de exploración para el usuario.*

### 4.2.1 Drill Down/up (Desglosar) y Drill Through (perforar a través de)

Estas técnicas le admiten ver a los reportes en otros niveles jerárquicos, ir del resumen al detalle, agrupar, rotar el cubo, entre otras operaciones. Ver Figura 43.



**Figura 43.** Navegación por un cubo de datos mediante Drill Down y Drill up.

Fuente. (López, Vela, & Mondejar, 2010)

#### 4.2.2 Configuración del *Drilldown*, sub-informes o anidamiento

Por medio de esta técnica podemos visualizar los datos de diversas formas para mostrar la relación desde lo general a lo detallado. Se pueden poner todos los datos en el informe, luego configurar para que el usuario al hacer clic vea los detalles; este es un desglose de acción. Se consiguen representar los datos en una región especial como una tabla o gráfico, que está anidada dentro de otra región de datos, como una matriz. Podemos visualizar los datos en un informe principal o también, poner los datos de detalle en otro repositorio de otro informe independiente para que se muestren cuando un usuario haga clic en un enlace, cuya configuración se ilustra a continuación. Ver Figura 44.

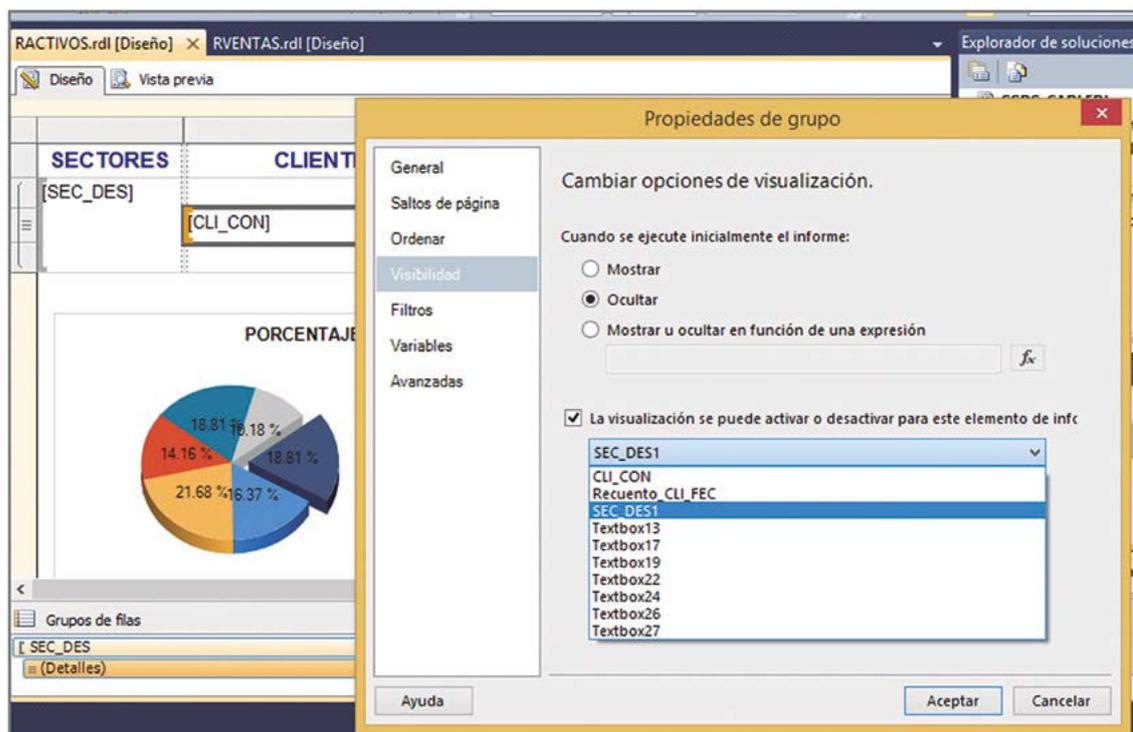


Figura 44. Configuración de grupos y anidamiento para crear un Drilldown en SQL Server.

#### 4.2.3 Implementación y vista previa del *Drilldown*

*Drill Down* y *Drill up* es una técnica para el aprovechamiento de datos, que ayuda a diseñar tableros de control e informes estratégicos, instrumentos que permiten ir de lo general al detalle a manera de un desglose. Por ejemplo, echar un vistazo al comportamiento o movimiento de la data por continente, país, estado, hasta el nivel más inferior que exista en el agrupamiento (Negash y & Gray, 2008) y el *Drill Up* se encarga de reagrupar desde el detalle hasta el nivel más general, ver Figura 22 y 23. Coincidiendo con la mayoría de autores y por experiencias obtenidas de varios estudios realizados, se concluye que estas herramientas de explotación son las que mayormente aportan para desarrollar informes gerenciales que permiten tomar decisiones altamente efectivas. Ver Figura 45.

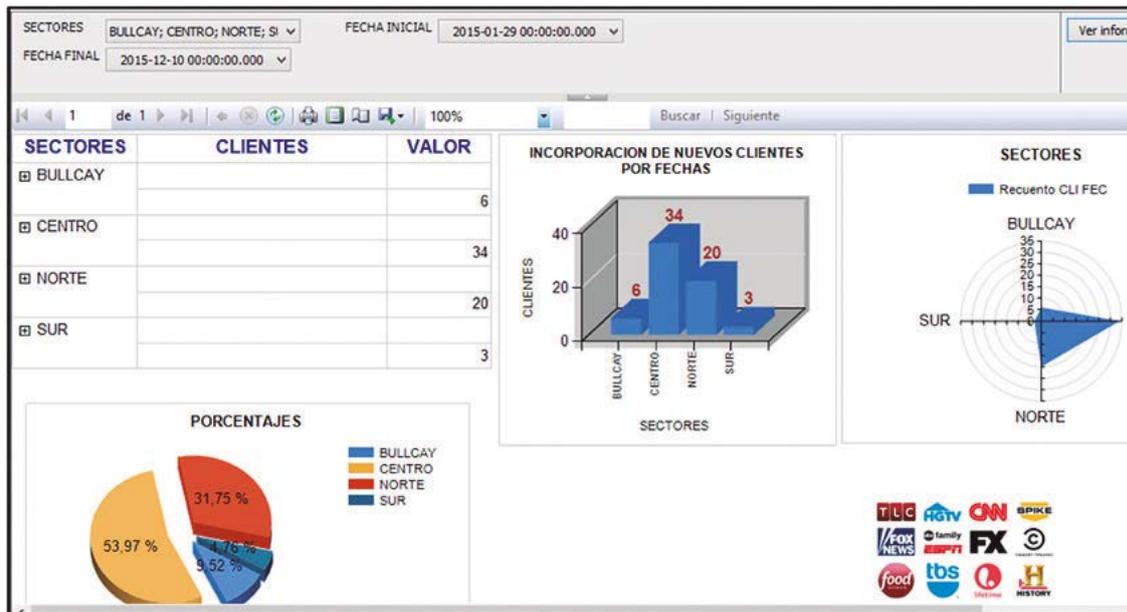


Figura 45. Resultado de un reporte aplicando Drill Down y Drill Up

Estos procesos altamente técnicos son los que debemos realizar porque en la actualidad los orígenes de datos se llenan muy fácilmente, y la organización y la relación de los datos no son las correctas como para poder hacer un análisis intuitivo. Para que estos procesos de explotación se den a mayor detalle depende mucho del nivel fino de granularidad, como ya se explicó en el apartado anterior. Ver Figura 46.

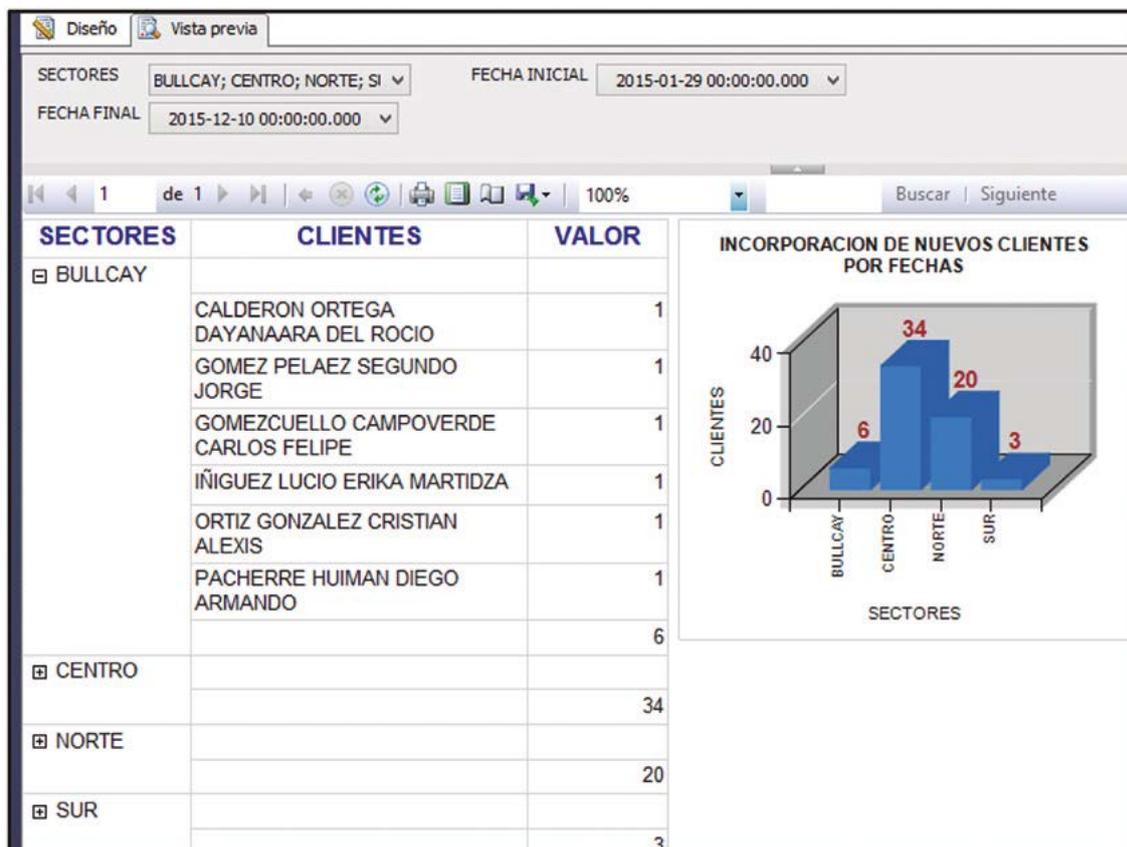


Figura 46. Despliegue de un Drill Down.

### 4.2.2 *Drill-Through* (Perforar a través) informes detallados

Según (Negash y & Gray, 2008), se puede definir un *Drill-Through*, como una herramienta para generar reportes y poderlos vincular por medio de links relacionados según el contenido de la información que los agrupa para observarlos simultáneamente, de uno en uno. Esta técnica nos ayuda a consultar distintas tablas de información de forma rápida.

Un informe detallado es un informe que un usuario abre haciendo clic en un enlace dentro de otro reporte. Los informes detallados comúnmente contienen detalles sobre un elemento que está incluido en un informe de síntesis original. Por ejemplo, en esta ilustración, se hace constar el resumen del informe de ventas, listas de órdenes y totales. Cuando un usuario hace clic en un número de orden en la lista de resumen, otro informe se abre con un nuevo detalle sobre esta orden.

### 4.2.3 Especificando la consulta

Las consultas realizadas a la base de datos para obtener información más detallada que el contenido en el modelo multidimensional se definen a partir de tablas o vistas existentes en el DBMS. Las vistas se manipulan en aquellos casos en que la consulta involucre campos de más de una tabla.

La configuración de los elementos de la consulta consiste en:

- emparejar la consulta con un nombre apropiado;
- asociar el nombre de la tabla o vista con los datos recuperados;
- especificar los campos de la consulta que se desean representar como columnas en el panel de *Drill Through*;
- definir los campos que se desean utilizar como filtros en la consulta, permitiendo al usuario restringir el resultado de la misma a los valores expresados en estos.

### 4.2.4 Definiendo filtros

Los filtros de la consulta son condiciones que se imponen sobre determinados campos para limitar el conjunto de resultados de la consulta. En general, los campos definidos como filtros se corresponden con dimensiones del modelo multidimensional para poder sincronizar las consultas realizadas en el cubo con el resultado detallado de la consulta a la base de datos. En este caso, se asocia el filtro a un nivel fijo de la dimensión del modelo multidimensional. A medida que se navega en el cubo, la consulta detallada se actualiza aplicando restricciones sobre los mismos valores filtrados (Nicolás Palombo, 2008). Sin embargo pueden definirse filtros que no estén relacionados con dimensiones, lo que permite restringir el resultado de la consulta

a la base de datos desde el propio panel de la consulta (Ernesto Agustin Forbes, 2008). Esto puede ser de utilidad para agregar algún dato no incluido en el modelo multidimensional que sea relevante en aquellos casos que se analice en detalle la información. Ver Figura 47.

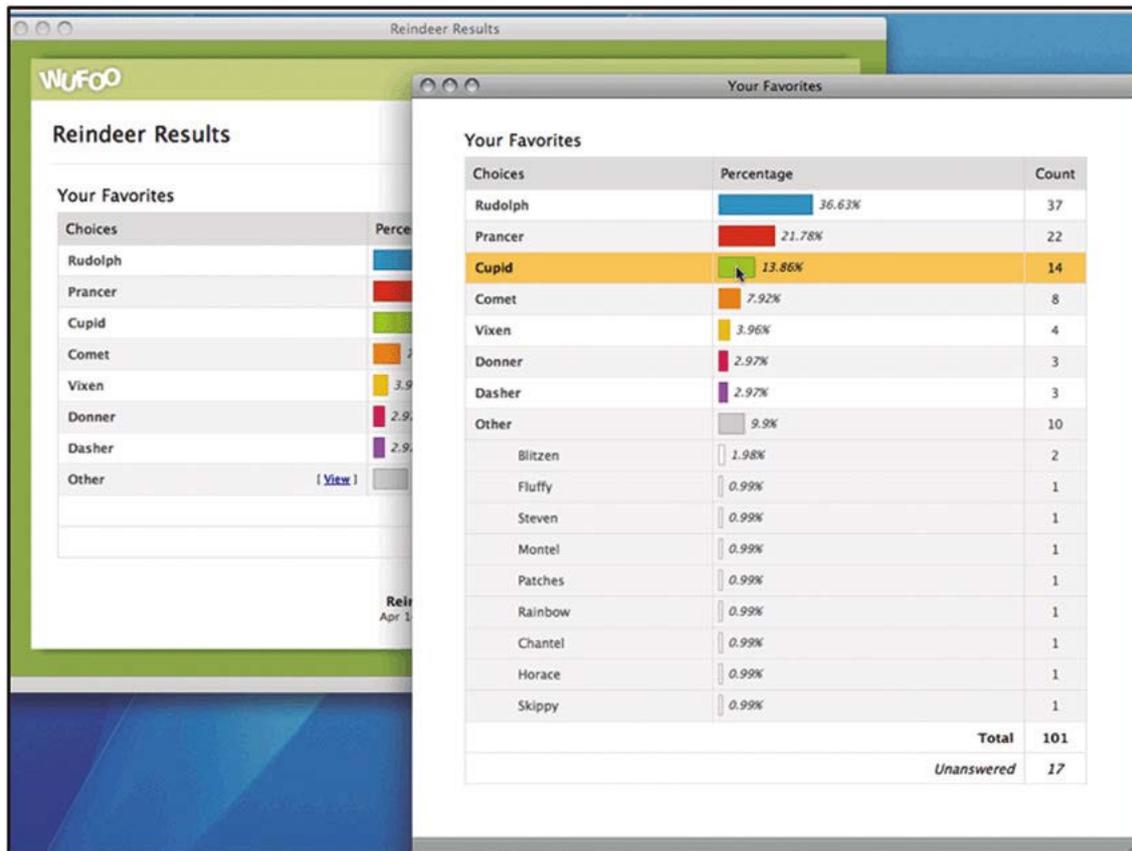


Figura 47. Formularios vinculados por medio de *Drillthrough*

### 4.3 Caso de estudio para aplicar técnicas de BI

A continuación, se expone un caso de estudio, en el cual se describen los principales elementos de desarrollo de un proyecto BI, como son los requerimientos, planificación, análisis, diseño e implementación de un proyecto junto con las herramientas apropiadas para cada etapa.

Etapas: servicios de integración de datos (ETL o SSIS)

Etapas: servicios de análisis de la información (SSAS)

Etapas: servicios de reportes gerenciales (*Dashboards*)

**Caso de estudio:** una empresa proveedora de servicios por cable quiere desarrollar un proyecto *BI* para gestionar el área de comercialización. Dentro de ella se pretende monitorear el flujo de ventas por periodos de tiempo que ayuden a tomar decisiones, determinar el índice de clientes nuevos y clientes dados de baja. También se requiere un tablero de control con tablas dinámicas y gráficos dinámicos, para presentar los movimientos de cobranzas en función de las dimensiones de tiempo y de los sectores

dentro de la ciudad donde ellos operan. Otro requerimiento es visualizar el comportamiento de los productos más comercializados y de mayor monto de ventas para aplicar la clasificación ABC de inventarios. El objetivo es tener una visión clara y global del comportamiento de los clientes sobre los pagos del servicio, para determinar qué sectores son los más problemáticos y que épocas del año son bajas sobre los requerimientos planteados; lo mismo que presentar el comportamiento de indicadores de compras, ventas y cobranzas y, por consiguiente, tener una buena percepción de los márgenes de ganancia. Para este estudio, la base de datos de origen del sistema transaccional de la empresa está en *MS Access*, nombre: *DBCablev.mdb*, para el destino *SQL Server R2 Data Tools*.

#### 4.3.1 Abstracción del problema:

A continuación, describimos los pasos para implementar el *BI*.

1. Caracterizar el problema de la empresa.
2. Establecer el objetivo que va a solucionar o apoyar el proyecto *BI*.
3. Normalizar los requerimientos del negocio, tanto estratégicos como tecnológicos.
4. Mapear el *BSC*.
5. Analizar la data y Metadata.
6. Construir el *Datamart* o *DW* ajustado al caso de estudio.
7. Diseñar el proceso *ETL* para extraer y transformar la información desde la fuente de origen y llenar el *Datamart* mediante *Integración Services* (SSIS).
8. Construir el cubo *OLAP* en la herramienta *Análisis Services* (SSAS).
9. Definir el *KPIS*.
10. Crear los tableros de control, tablas dinámicas, *Drill Dow* y *Drill Through*.
11. Generar los gráficos dinámicos.
12. Ejecutar la evaluación final de los resultados obtenidos versus los objetivos y requerimientos planteados.
13. Implementar procesos de mejora continua.

#### 4.3.2 Caracterizar el problema de la empresa

El problema propuesto consiste en que la empresa de servicio de televisión por cable cuenta con un sistema transaccional cuya base de datos está en *MS Access*. Pero este sistema, no cuenta con reportes de análisis gerencial en el cual puedan monitorear la información de manera flexible, dinámica, amigable y rápida, en los distintos escenarios. La empresa no dispone de herramientas de *BI* para crear tableros de control, que ayuden a la gerencia a tomar buenas decisiones. La empresa tampoco dispone de una plataforma de análisis de datos que dé soporte específico al área de ventas y cobranzas, centrado en el servicio al cliente.

### 4.3.3 Establecer el objetivo que va a solucionar o apoyar el proyecto *BI*.

El objetivo principal es de carácter financiero y de optimización de recursos. Además, pretende tener una visión clara y global del comportamiento de los clientes sobre los pagos del servicio para aumentar su satisfacción. La empresa también quiere segmentar los sectores más problemáticos e identificar que épocas del año son las más bajas en cuanto a cobros para mejorar las utilidades.

### 4.3.4 Normalizar los requerimientos del negocio, tanto estratégicos como tecnológicos

1. Determinar los objetivos institucionales de la empresa.
2. Alinear los objetivos de la organización con la TCE por medio del *BSC*.
3. Adquirir datos y preprocesarlos.
4. Determinar los principales indicadores del negocio. Esto se amplía en (Rodríguez *et al.*, 2017).
5. Analizar el grado de incidencia de nuevos clientes y clientes dados de baja.
6. Implementar algoritmos de predicción y analíticos con los datos obtenidos, para analizar los requerimientos planteados, con granularidad de las dimensiones tiempo y ubicación.
7. Desarrollar los tableros de control gráficos, dinámicos e intuitivos.
8. Ajustar la aplicación *BI*.

### 4.3.5 Mapear el BSC

En esta etapa es preciso desarrollar las perspectivas de del BSC para identificar los requerimientos, el objetivo principal y la visión de la empresa, con el proyecto *BI*, que se propondrá como solución más adelante. Ver Figura 48.

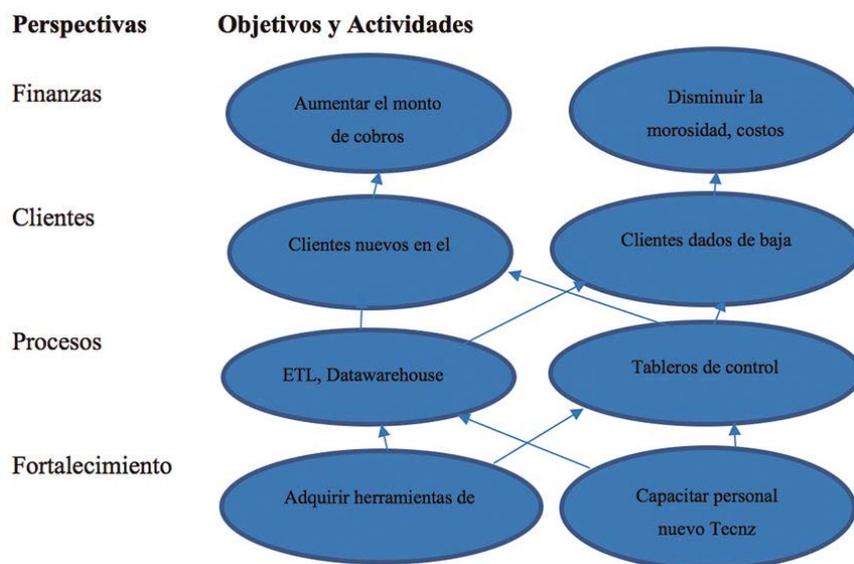


Figura 48. Mapa estratégico del BSC.

### 4.3.6 Análisis de la data y meta-data

En esta etapa primero tomamos la base de datos origen para analizar su estructura y la meta-data (tablas, campos, relaciones, tipos de datos, tamaños). Ver Figura 49.

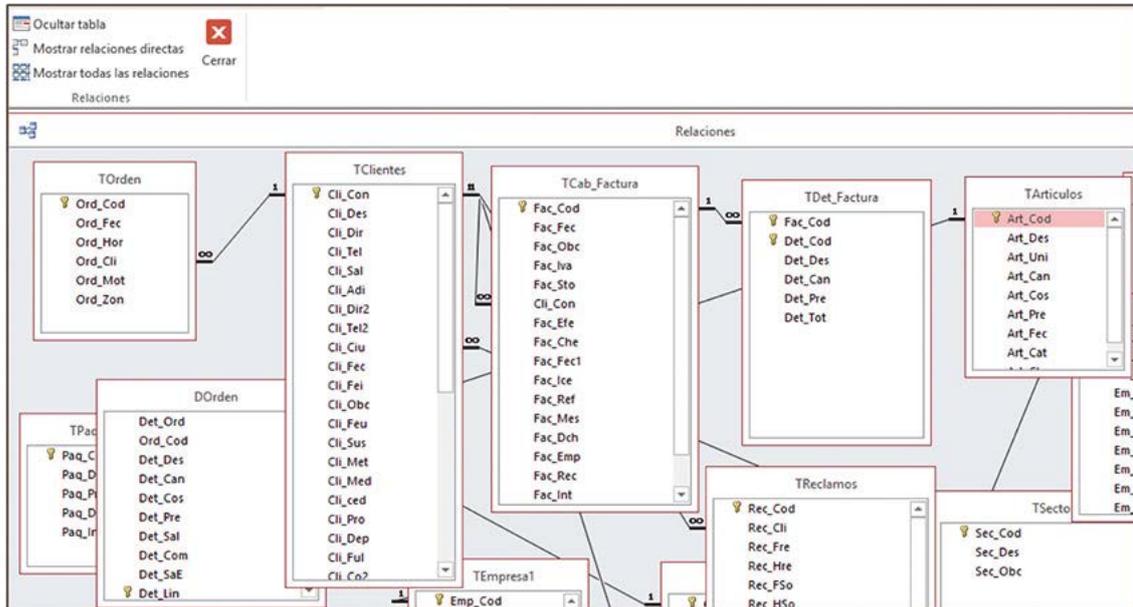


Figura 49. Fuente de datos de la base relacional MS Access de la empresa.

Durante el desarrollo de la solución, tomamos las tablas TCientes, TCab\_Factura, TDet\_Factura y TArtículos y TSector para importar la información al *datamart* por medio del proceso *ETL*, tal como se indicará más adelante. Para esto es necesario desarrollar un análisis de la data. Ver Figura 50.

	A	B	C	D	E	F	G	H	I	J
1		ANÁLISIS DE LA DATA								
2		DATOS DE ORIGEN (ACCESS)				DATOS DE DESTINO		(DataMart SQL SERVER)		
3		TCientes	Cli_Con							
4			Cli_Des	TCientes	Cli_Con	Integer				
5			Cli_Dir		Cli_Des	Vchar(120)				
6			Cli_Tel		Cli_Dir	Vchar(120)				
7			Cli_Sus		Cli_Tel	Vchar(30)				
8			Cli_Fec		Cli_Sus	bit				
9					Cli_Fec	Date				
10		TSector	Sec_Cod		Sec_Des	Vchar(80)				
11			Sec_Des							
12				Tcab_Facturas	Fac_Cod	Integer				
13		Tcab_Facturas	Fac_Cod		Fac_Cli	Integer				
14			Fac_Cli		Fac_Fec	Date				
15			Fac_Fec		Det_Cod	Integer				
16					Det_Can	real				
17		TDet_Factura	Fac_Cod		Det_Des	Vchar(120)				
18			Det_Cod		Det_Pre	real				
19			Det_Can		Det_Tot	real				
20			Det_Des		Fac_Id	Autonumerico				
21			Det_Pre							
22			Det_Tot	TArticulos	Art_Cod	Integer				
23					Art_Des	Vchar(120)				
24		TArticulos	Art_Cod		Art_Pre	real				
25			Art_Des		Art_Cat	Vchar(80)				
26			Art_Pre		Art_Can	Integer				
27			Art_Cat		Art_Cos	real				
28			Art_Can							
29			Art_Cos	Tiempo	fecha					
30					Año					
31					Trimestre					
32					Mes					
33										

Figura 50. Tabla de análisis de la data de origen vs destino.

### 4.3.7 Construcción del *Datamart* ajustado al caso de estudio

Luego de establecer el análisis de la data, ya se puede diseñar el datamart en *SQL Server*. Luego, advertimos la necesidad de establecer un usuario y un password para esta estructura, requisito usado en la etapa de *Analysis Services*. El modelo seleccionado para esta *datamart* es el tipo estrella. Ver figura 51.

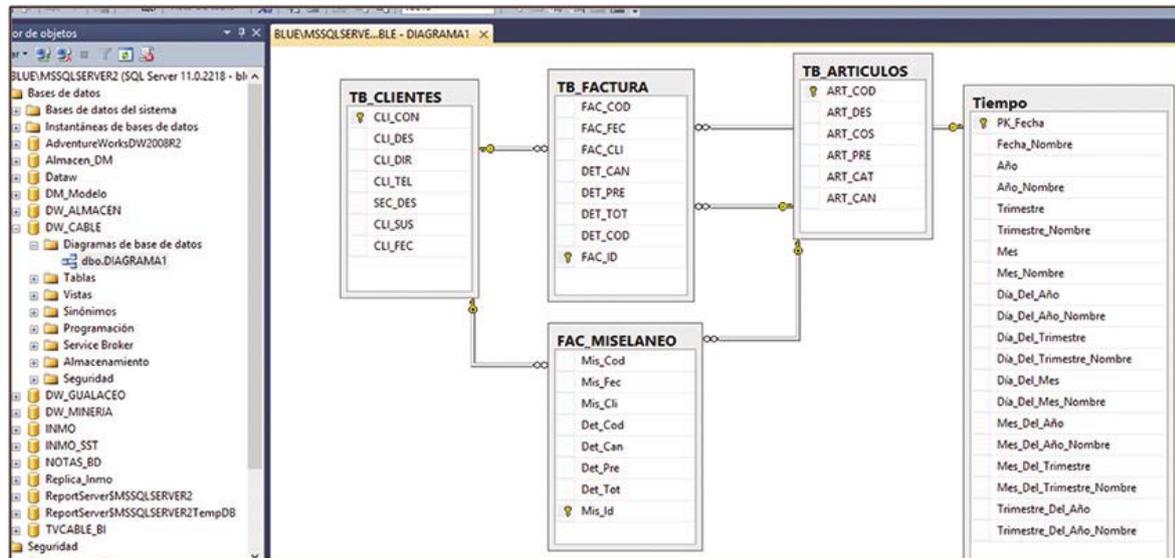


Figura 51. Modelo Estrella para el *datamart* de la solución.

Es importante que el datamart quede relacionado para usar el modelo ROLAP en la construcción de los procesos del *BI*. En este punto destacamos la importancia de crear la tabla de tiempos con jerarquías, desde el *Analysis services* en el origen de datos, que es el *datamart*. Para esto es importante conocer las fechas de la primera y última transacción realizadas en la base de datos de la empresa, aunque para la última fecha se puede definir una a futuro lejano o cercano; es opcional; pero debe ser menor a la fecha de la última transacción de la empresa.

### 4.3.8 Crear el ETL para extraer y transformar la información

En este paso diseñamos un *ETL* para llenar el *Datamart* de tipo estrella, construido anteriormente. El *ETL* se comienza creando un proyecto en *Integration Services*, donde se da un nombre específico. Luego viene el proceso propiamente dicho.

- Comenzamos ligando el *datawarehouse* o *datamart* en la tarea ejecutar *SQL* del SSIS (Ver Figura 19).
- Luego, se define un código *SQL* para limpiar las tablas del *datawarehouse* para cargar en cada análisis que se realice, con datos nuevos o actualizados. Ver Figura 52.

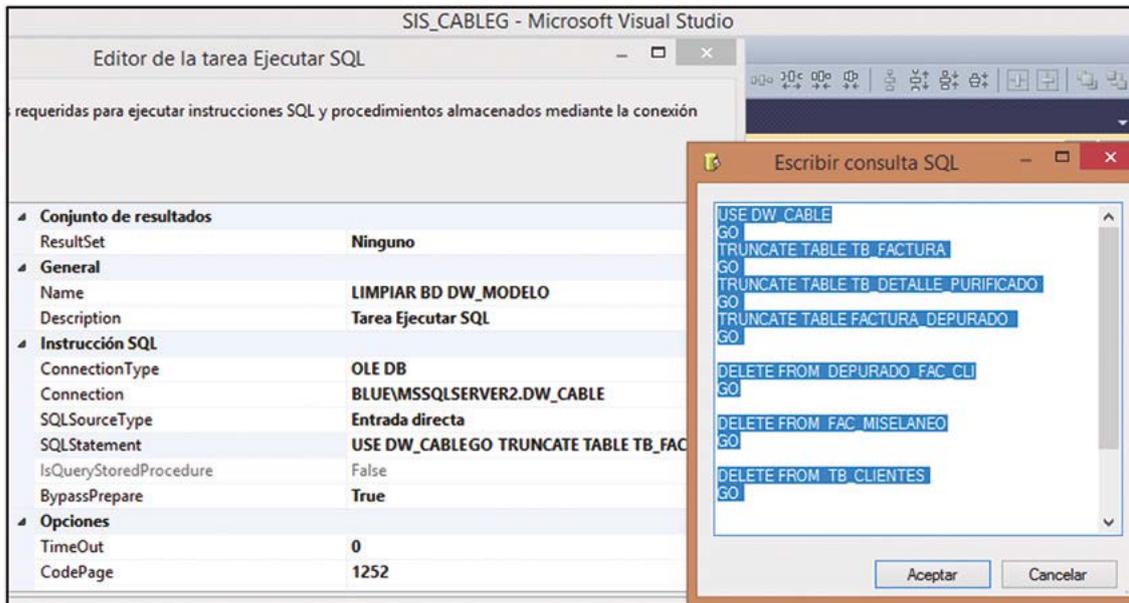


Figura 52. Diseño de Tarea Ejecutar SQL en *Integration Services*.

- El siguiente paso, diseñamos los flujos de datos para cada tabla del datamart. Aquí se puede apreciar cómo se exportan los datos desde el origen de la base de datos Access, las tablas TClientes y TSector, hacia el *datamart*, llenando las tablas DW\_Cable y TClientes. El propósito es combinar el campo *sector* de la tabla TSector con los campos de TClientes determinados en el análisis de la data del paso 5. Aquí se hace el proceso inverso a la normalización que es la desnormalización de las dos tablas en cuestión. Ver Figura 53.

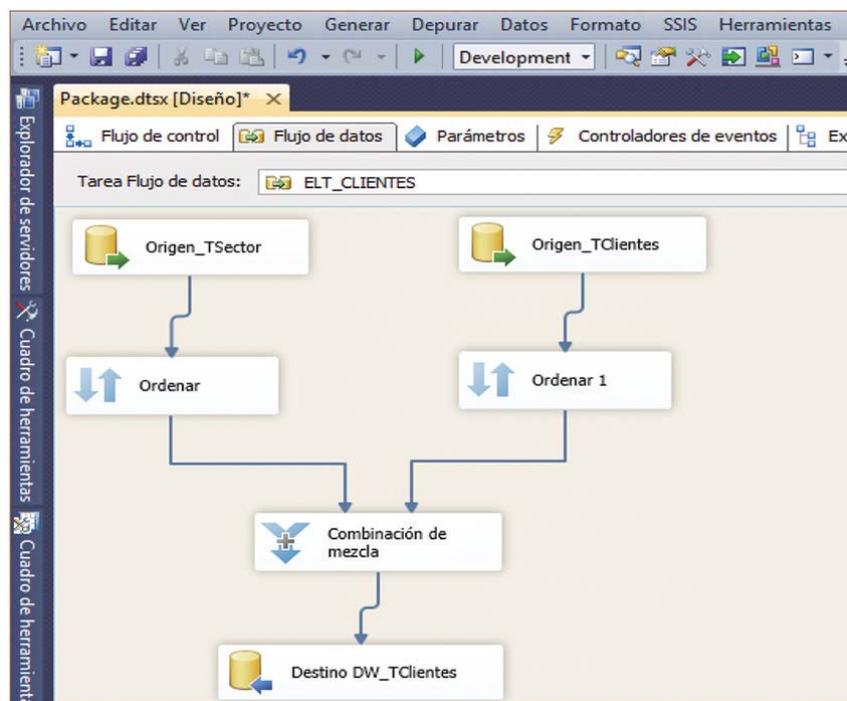


Figura 53. Diseño de Tarea Flujo de datos en *Integration Services*.

Finalmente, el flujo de control completo, luego de diseñar todas las tareas de los flujos de datos y la tarea de ejecutar SQL, queda así. Ver Figura 54.

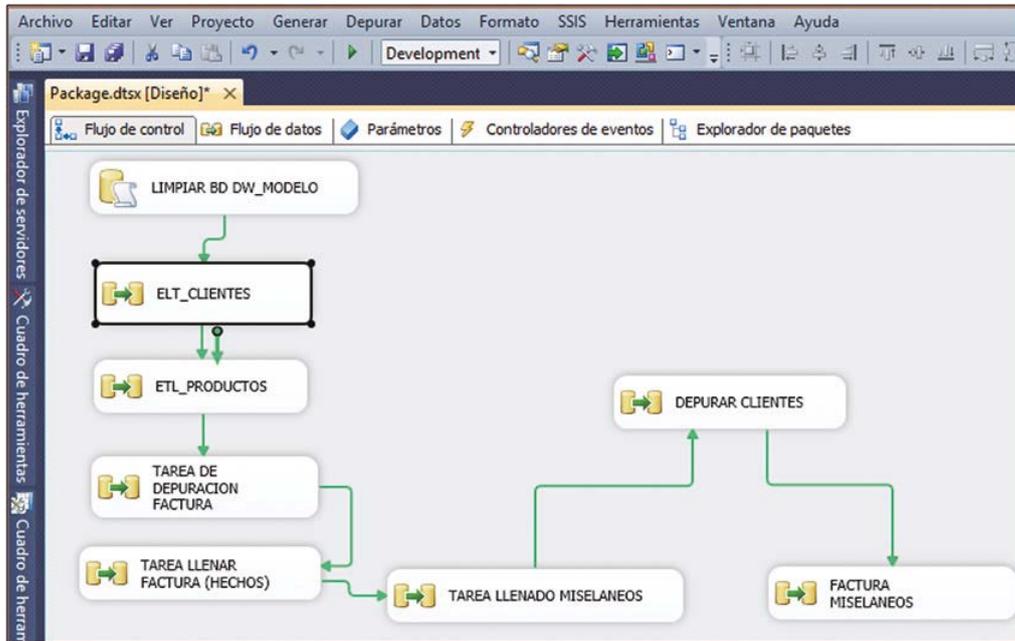


Figura 54. Diseño del Flujo de control completo para el ETL de la solución en el SSIS.

#### 4.3.9 Construcción del cubo OLAP con *Análisis Services* (SSAS)

Luego que ya se tiene definido el ETL, ejecutado y llenado el datamart, se construye el cubo OLAP junto con los KPI's. En esta etapa, se debe identificar correctamente las medidas (*measures*) y las dimensiones del cubo. Los pasos para crear un cubo OLAP en SSAS se detallan a continuación. Ver Figura 23.

- Definir el origen de datos, para este caso, DW\_Cable del *datamart* de *SQL Server*
- Crear una nueva vista de datos como se aprecia en la Figura 22
- Crear las dimensiones, en esta etapa es importante la dimensión tiempo la misma que se debe formatear o configurar para cada bloque de tiempo con: año, semestre, cuatrimestre, trimestre, mes, quincena o diario (fecha). Pueden ser definidos todos o algunos de ellos, siempre y cuando, en el momento de la creación de la tabla *tiempo* lo haya creado con estas jerarquías. Ver Figura 55.

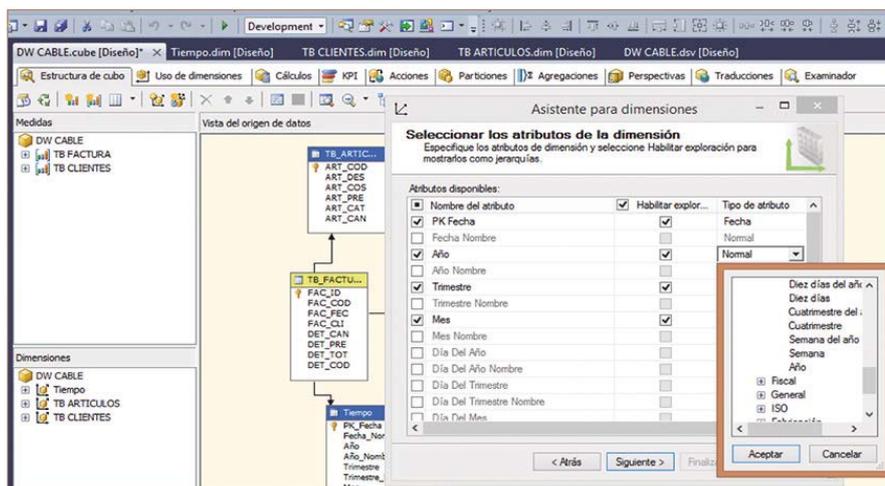


Figura 55. Ventana para configurar la dimensión tiempo en sus distintas jerarquías.

- Finalmente, se crea el nuevo cubo OLAP con la tabla de hechos y con las dimensiones establecidas anteriormente. Dentro de la tabla de hechos se encuentran los *measures*. Todos estos elementos se describen muy bien en la gráfica que se encuentra a continuación. Ver Figura 56.

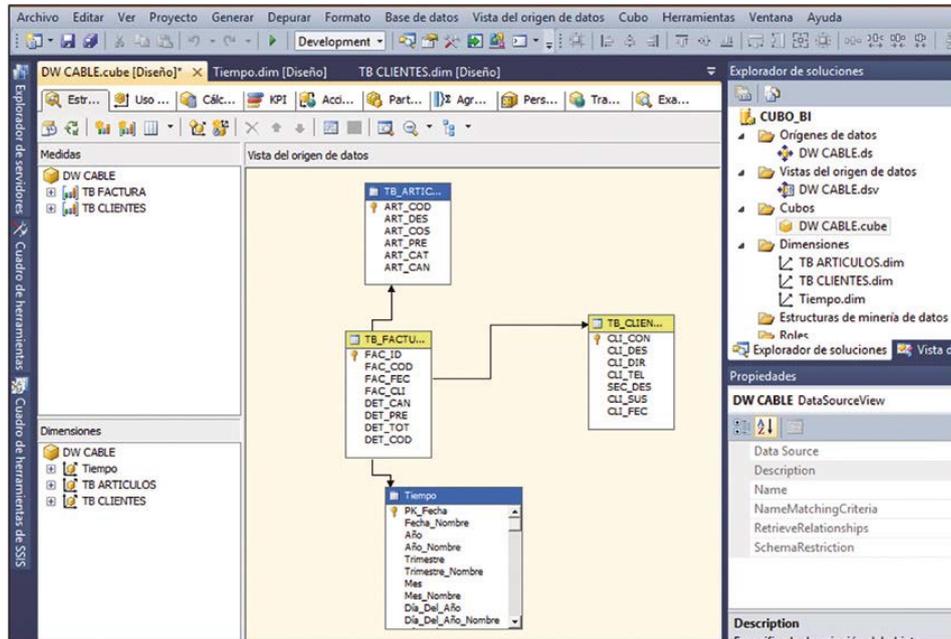


Figura 56. Ventana de SSAS luego de crear el cubo con los elementos de diseño.

Una vez que ha terminado de diseñar el cubo OLAP, se lo implementa. Si todo está bien, quedará creado y compilado, listo para el análisis desde cualquier herramienta de reportes con tableros de control tales como *Reporting Services*, *Crystal Report* o Excel, entre otros. Ver Figura 57.

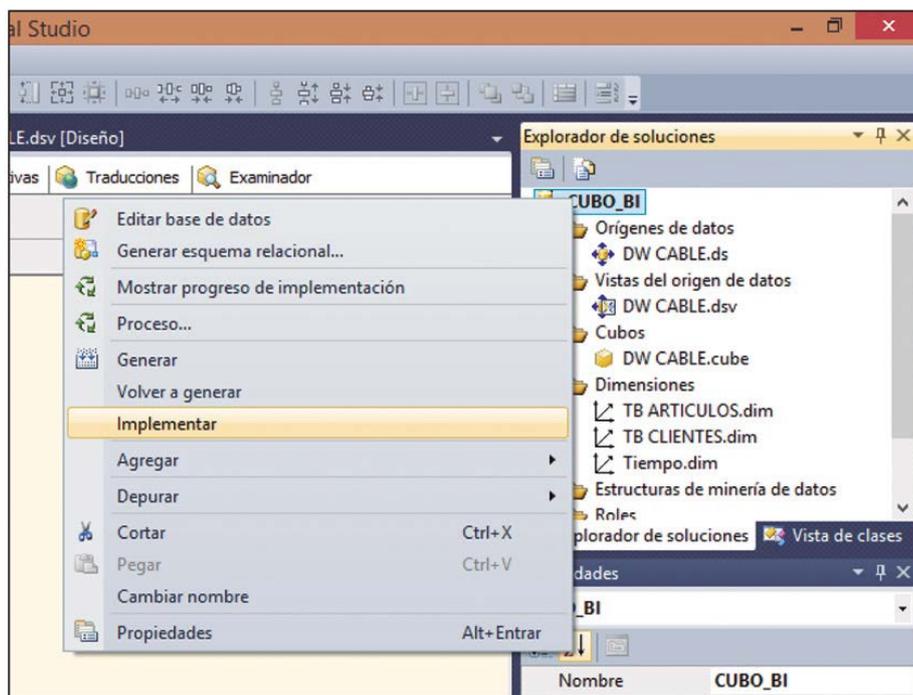


Figura 57. Ventana para configurar la dimensión tiempo en sus distintas jerarquías.

### 4.3.10 Creación y configuración de los *KPIS*

El siguiente paso es la creación de los *KPIS* desde la pestaña del mismo nombre. Ver Figura 58. En esta ventana se deben definir los siguientes requerimientos, que son obligatorios para diseñar un *KPI* consistente y productivo.

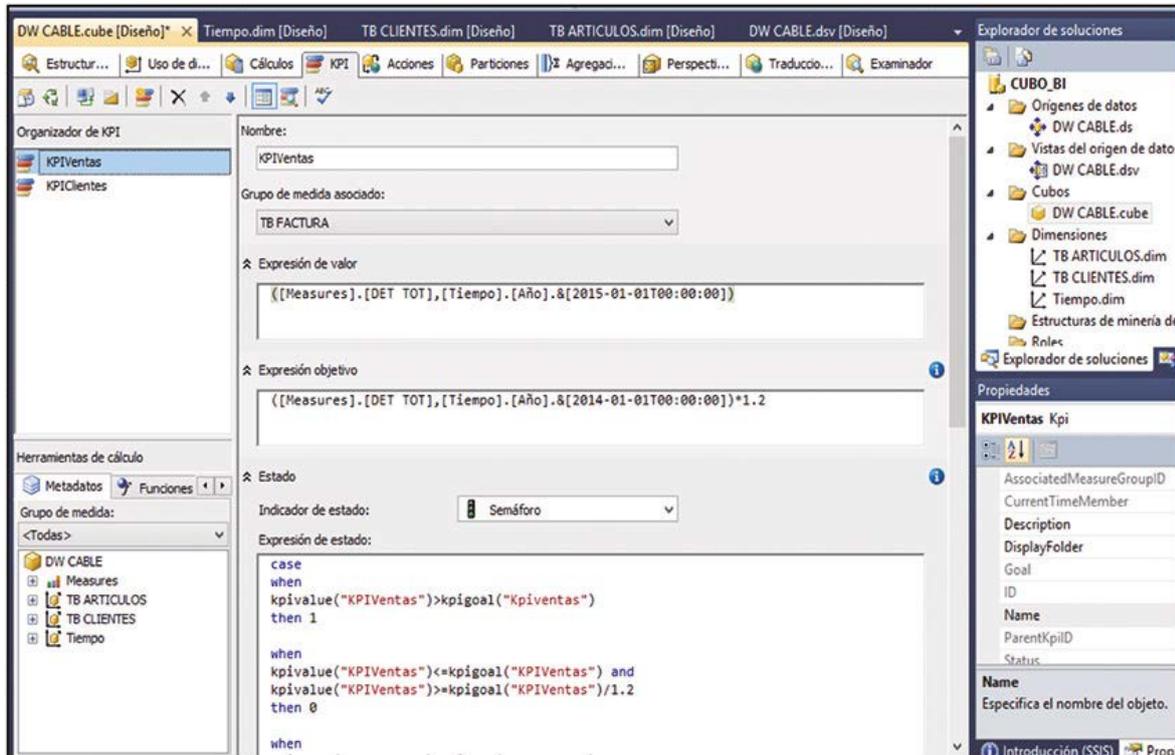


Figura 58. Ventana para crear *KPIS* en SSAS.

Nombre del *KPI*, ejemplo KPIVentas

Grupo de medida asociado:

TB\_FACTURA, esta estructura es la tabla de hechos.

KPIValue, con este parámetro se define el valor total de la medida del *KPI*.

KPIGoal, En esta entrada se define el valor total del objetivo con el cual se va a contrastar el KPIValue.

*Indicador de estado*: es un conjunto de opciones, en el cual se puede seleccionar un medidor o gauge, donde se va a observar el rendimiento del *KPI* en tres fases que son: cumplido, igualado o no cumplido (verde, amarillo y rojo).

### Tendencia

*Indicador de tendencia*, es el gráfico de la tendencia el cual tiene tres momentos: (1) Hacia arriba mejorado, (2) horizontal igual o mantener y (3) hacia abajo desmejorado, con respecto a otro periodo de tiempo equivalente. Ver Figura 59.

Mostrar estructura	Valor	Objetivo	Estado	Tendencia	Peso
KPIClientes	1100	1320			
KPIVentas	4445727	4568849			

Figura 59. SSAS de KPIClientes y KPIVentas en un *gauge* de estado y tendencia.

Para desarrollar los *KPIValue*, *KPIGoal*, *Status* y *Trend* o tendencias, se debe configurar y realizar los ajustes, por medio del código DMX. Para programar las condiciones necesarias de los *KPIS* se debe revisar el apartado alusivo a este tema.

#### 4.4. Creación de los tableros de control

##### 4.4.1 Uso de tablas dinámicas y técnicas de *Drill Down*, *Drill Through*

Estas dos técnicas fueron usadas para dar solución a este problema, y explorar datos de las estructuras dimensionales, en forma vertical u horizontal. Esto significa que el usuario tendría mayor flexibilidad para agrupar los datos, mejorar la intuición de navegabilidad y exploración de informes. Estas herramientas fueron construidas en la plataforma de Servicios de Reportes de *SQL Server (SSRS)*. Para esto fue necesario realizar los procesos de integración (*Integration Services ETL*) y de análisis (*Analysis Services*). Para tener mayor eficiencia se usaron como origen de datos el cubo OLAP. Con estos procesos previos realizados correctamente, desarrollamos los tableros de control o reportes gerenciales, los mismos que detallamos a continuación.

En primer lugar, abrimos un proyecto de *Reporting Services* en *SQL Server*, luego, asignamos un nombre y creamos uno a uno los reportes dinámicos desde el explorador de soluciones. Estos reportes deben tener una visión gerencial; en ellos se puede agregar tablas pívot, gráficos dinámicos o los *KPIS* para la toma de decisiones, Ver Figura 60.

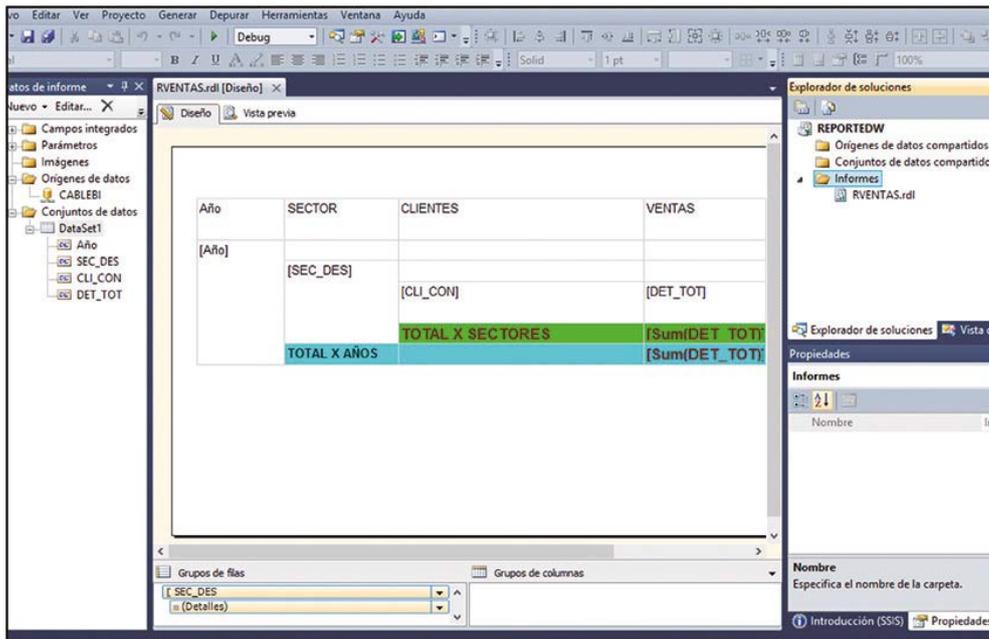


Figura 60. SSRS muestra la creación de un reporte con *Drill Down*.

En estos reportes organizamos la información de manera intuitiva para el análisis de las ventas del negocio en los distintos periodos de tiempo. Se utiliza navegación *drill down* para satisfacer los requerimientos gerenciales establecidos en el caso de estudio propuesto.

El siguiente paso es definir el origen de datos *CABLEBI*. Para el estudio de este caso utilizamos como origen al proceso desarrollado en Analysis Services *CUBOBI*. Este es un cubo desarrollado en la etapa anterior. Esta configuración se describe en la Figura 61.

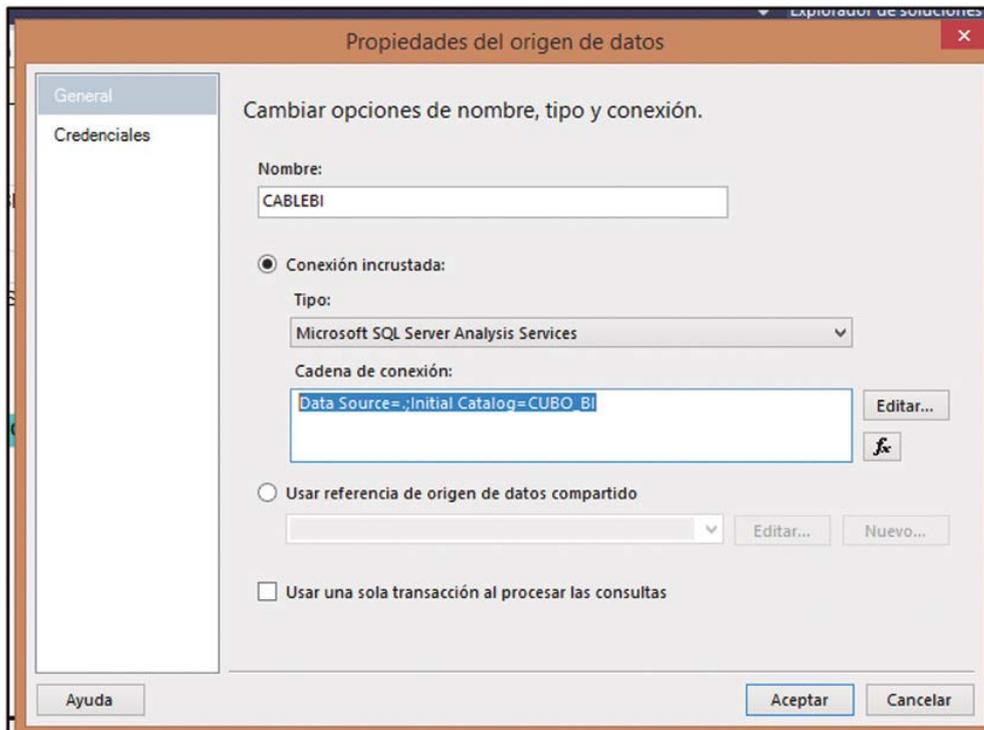


Figura 61. SSRS Asignando origen de datos para el reporte.

El siguiente paso fue definir el conjunto de datos seleccionando desde el cubo, clasificado por campos, año, Sec\_Des (Sector), Cli\_Con (muestra el nombre de los clientes) y Det\_Tot (Total ventas). Luego de esto, creamos la tabla *Pivot* para conseguir una navegación *Drill Down*. Aquí se muestra como los datos se agrupan por sectores, por años y se totalizan por cada grupo. Revisar Figura 62.

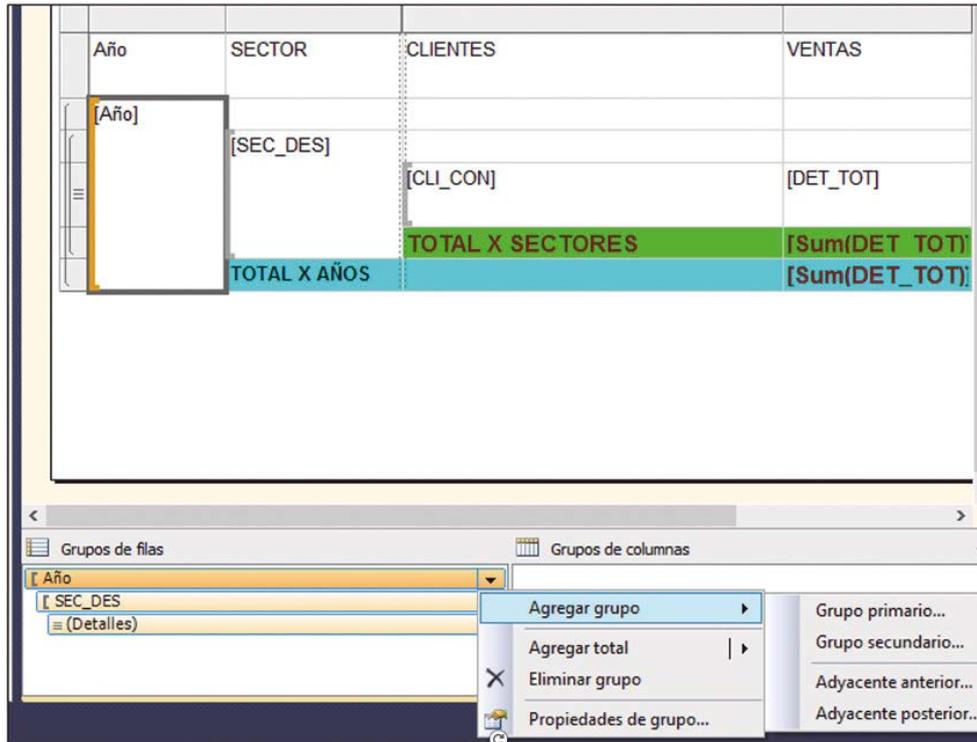


Figura 62. SSRS Configuración y definición de grupos para armar el *Drill Down*.

Finalmente, con todos los pasos anteriores ejecutados correctamente y completados con éxito, representamos el resultado terminado, según los requerimientos iniciales. Esto se presenta en la Figura 63.

Año	SECTOR	CLIENTES	VENTAS
2016-01-01 00:00:00.000	BULLCAY	GOMEZCUELLO CAMPOVERDE CARLOS FELIPE	14,752
		GUNCAY TORRES VICTOR	177,024
		IDROVO ROJAS SEBASTIAN	14,752
		MONTENEGRO AMBROSI RICARDO ROMEO	14,752
		<b>TOTAL X SECTORES</b>	<b>221,28</b>
	CANCAY		
		<b>TOTAL X SECTORES</b>	<b>29,50</b>
	<b>TOTAL X AÑOS</b>		<b>250,78</b>

Figura 63. SSRS presentación del informe para su respectivo análisis con *Drill Down*.

#### 4.4.2 Generación de gráficos dinámicos, mapas, líneas, pie y barras.

Las técnicas de *BI* también nos permiten ampliar las presentaciones en mapas georreferenciales y representar en estos mapas los indicadores. Los reportes de este tipo visualizan claramente la información con las opciones de ampliar o reducir los grupos, por medio de una correspondencia biyectiva, según los totales de los campos cuantitativos de las tablas (*datasets*). Además, esto permite filtrar por campos especificados, por años específicos, luego por uno o más sectores, mientras que la tabla pívot muestra y oculta los datos detallados de cada columna, según la necesidad del usuario. Este ejemplo de reporte dinámico muestra la versatilidad y navegabilidad del informe y un poder de análisis superior para la toma de decisiones. En la figura 64, se muestra que la distribución de las ventas es mayor en la provincia del Azuay y Cañar; luego le sigue Guayas y, al último, El Oro, respectivamente.

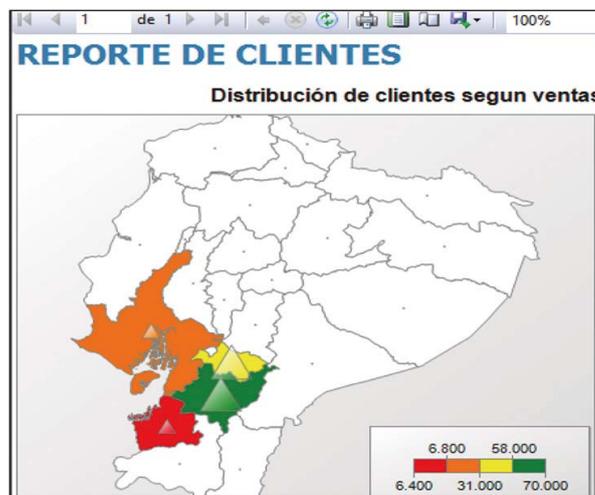
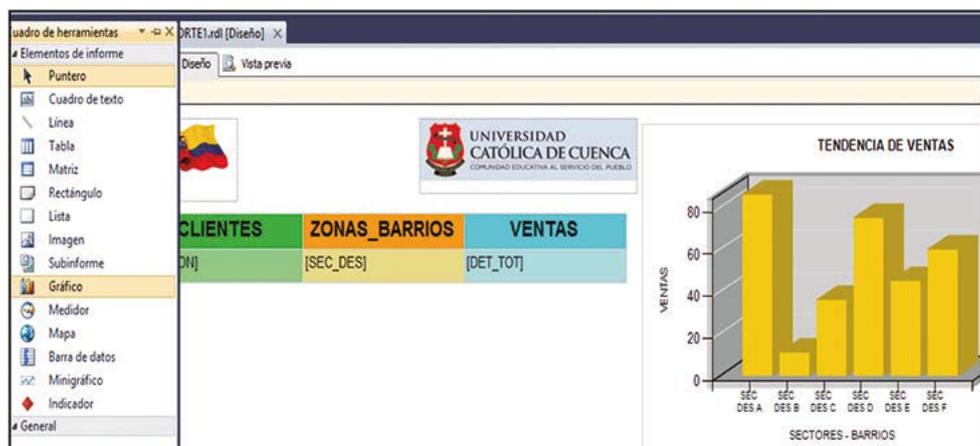
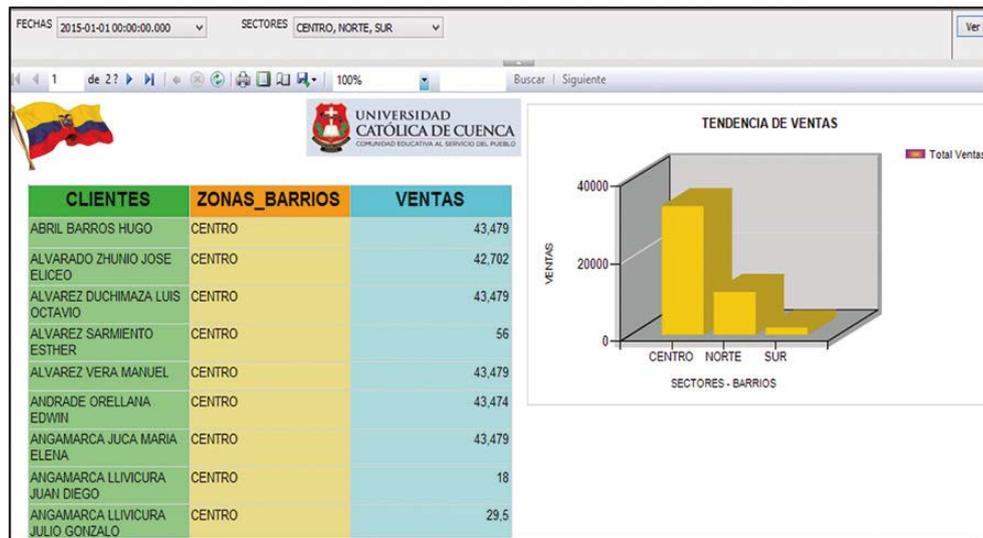


Figura 64. Distribución de ventas por provincias utilizando un mapa ESRI.

Ahora bien, para agregar un gráfico dinámico a un reporte gerencial en SSRS, se lo debe establecer seleccionado un tipo de gráfico en la sección del reporte y luego, asignando los parámetros necesarios para que la gráfica visualice las tendencias de los datos obtenidos, tal como mostramos en la Figura 65a.



(a) Análisis de tendencia diseño

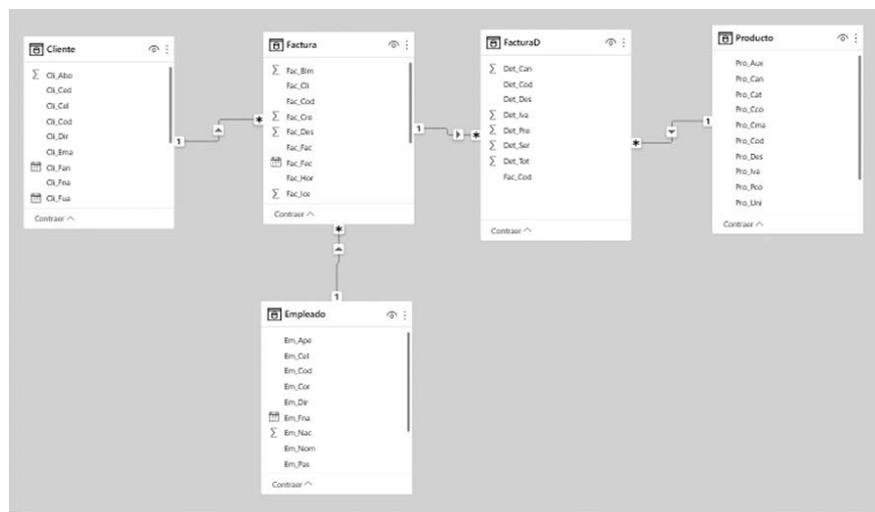


(a) Análisis de tendencia ejecución.

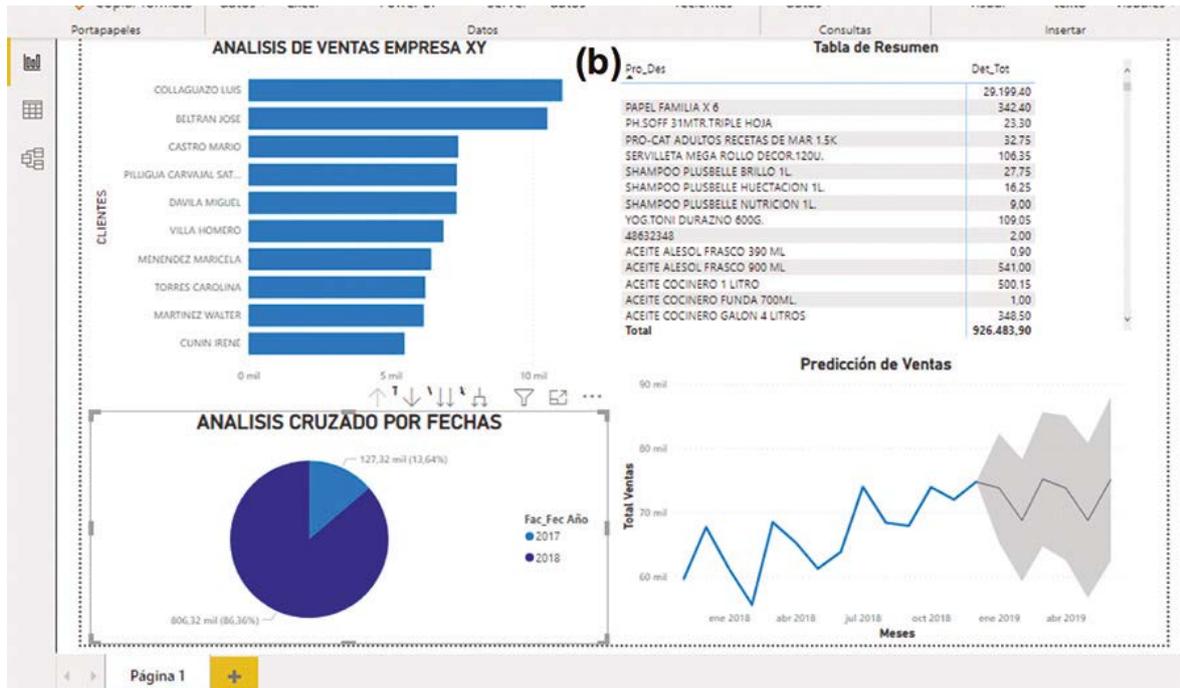
**Figura 65.** SSRS Configuración y definición de un gráfico dinámico dentro del informe.

A continuación, visualizamos los resultados del reporte en el cual analizamos el comportamiento de la información dinámicamente. Estos tableros de control ayudan a la generación de nuevo conocimiento para dar soporte a la toma de decisiones. En el ejemplo, se toman el año, los sectores centro, norte y sur de una ciudad específica y se visualizan en el gráfico los datos agregados y totalizados, según el nivel seleccionado (grado de granula). Ver la Figura 65b.

También se muestra cómo es posible obtener otros resultados interesantes desde otras herramientas de Inteligencia de Negocios como *Power BI*. En este proceso se migra la información de *Microsoft Access*. Seleccionamos las tablas necesarias, de preferencia, las tablas transaccionales relacionadas con el módulo de ventas, puesto que aquí encontramos la tabla factura, con las fechas de cada venta, y la tabla factura detalle, con los productos vendidos. Las dos tablas se pueden usar como tabla de Hechos. Luego, las tablas de clientes, empleados y productos serán las dimensiones. Ver Figura 66a.



(a) Modelo del datamart.



(b) Modelo del tablero de control.

Figura 66. Power BI. Análisis de una base de datos del módulo de ventas.

En la Figura 66b, se muestra la facilidad con la que en la actualidad se pueden diseñar proyectos de análisis de BI. Una vez que se extraen los datos a la herramienta de *Power BI*, en cuestión de minutos, podemos levantar un tablero de control (*dashboard*) de alto valor analítico, que muestre información de los clientes que más compran de dentro de la empresa, la tabla de productos más vendidos, la clasificación de las ventas por años, predecir las ventas para n meses (o puntos). La presentación es flexible para agregar nuevos datos y formatear el diseño de las etiquetas.

#### 4.3.13 Clasificación ABC con los productos más vendidos

Otra técnica de gran utilidad para el análisis de datos y toma de decisiones es la categorización del inventario, con la cual filtramos y visualizamos los productos más vendidos y menos vendidos, según el requerimiento. Esta información permite identificar la importancia de las mercancías más valiosas para la empresa, lo que facilita tomar una serie de decisiones relacionadas con la cadena de suministro (Toro López, 2016). El cálculo se desarrolla utilizando el historial de las facturas diarias vendidas. Para el ejemplo, tomamos el 10% de productos clase A, 30% de productos clase B, y 60% de productos C, que refieren su orden de importancia.

Para el caso de estudio, evaluamos los materiales más usados por la empresa como parte de la solución del estudio de caso planteado. Esto se aprecia con mejor claridad en la Figura 67.

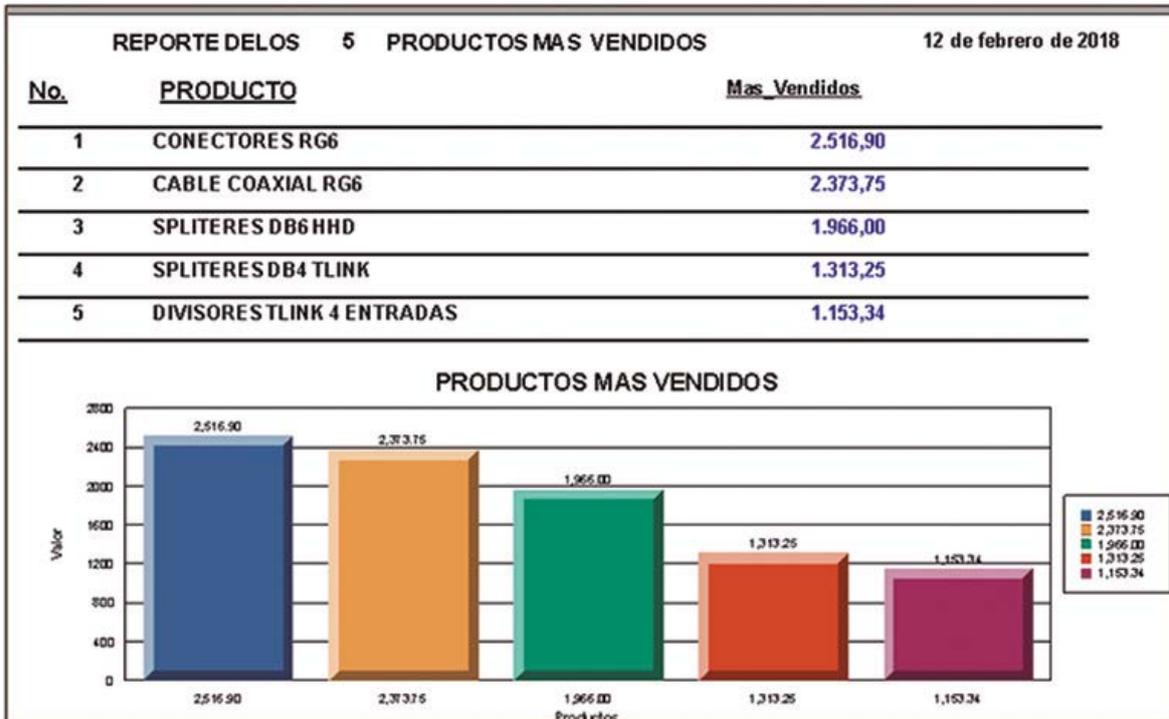


Figura 67. Reporte de los productos más vendidos usando el criterio ABC.

#### 4.3.14 Evaluación de los resultados y mejora continua

La generación de todo este proceso debe tener una etapa en la que se realice una evaluación de los resultados obtenidos para realizar ajustes y correctivos oportunos. Este proceso se efectúa con el objetivo de evaluar los resultados para descubrir si la aplicación muestra la información requerida. Además, para comprobar si genera algún nuevo conocimiento que beneficie al negocio. Por lo tanto, si la evaluación es favorable, cumplimos con la visión y los objetivos planteados en el *BSC*, evidenciando la generación de valor agregado impulsado por las herramientas tecnológicas revisadas. En caso contrario, ajustaremos los procesos expuestos en el *BSC* y procesos del *BI* como: *ETL*, cubos *OLAP*, tableros de control y reportes dinámicos en cada área. Esto implica que deberemos hacer un análisis del comportamiento de los datos mostrados en los reportes, informes y *KPIS* desarrollados con el fin de detectar errores y así poder afinar nuevamente la herramienta del proyecto. Es decir, lo que pretendemos como meta final es satisfacer las expectativas propuestas por el cliente o directivos del negocio. Esto es necesario para cumplir con los principios propuestos en este libro. Articulamos las técnicas empresariales de mejora continua con las TCE, para optimizar los distintos escenarios del negocio.



# CAPÍTULO V

## Analítica de datos



## 5. Minería de datos

*El objetivo de este capítulo es ampliar las técnicas de extracción del conocimiento (del inglés, Knowledge Discovery in Databases KDD) las cuales son utilizadas ampliamente en las áreas de investigación, negocios, redes sociales, entre otras aplicaciones, con fines descriptivos y predictivos.*

### 5.1 Definición de minería de datos (*Datamining*)

La minería de datos (*del inglés Datamining*) es una de las técnicas más aprovechadas por las ciencias computacionales, en la actualidad, inclusive, tiene un gran potencial a futuro en distintas áreas productivas para descubrir el conocimiento de datos, aplicando el modelo KDD (Quiroz-Gil & Valencia, 2012). Estudios de publicaciones realizadas sobre este tema exponen que, la minería de datos, mediante el conocimiento adquirido de los clientes, debe ser capaz de interpretar sus objetivos, expectativas y deseos (Vieira-Braga *et al.*, 2009); que está orientada al cliente, y que para alcanzar todos estos conocimientos no le basta desarrollar un CRM. Necesita capacidad analítica para identificar patrones y predicciones a partir de los datos estratégicos, extraídos de distintas fuentes de datos, similar a lo que se hacen en un *BI*. Esta aplicación debe realizar procesos de extracción, transformación y carga a fuentes de destino de las cuales se desarrolla el trabajo de minería.

En la actualidad, los datos de aplicaciones financieras, marketing, comercio electrónico y redes sociales orientadas a los negocios son analizados por medio de la minería de datos. Así también, otras aplicaciones web medioambientales y sociales están siendo sometidas a procesos de minería por medio de algoritmos descriptivos y predictivos, para analizar patrones de comportamiento o de clasificación.

Al igual que en la inteligencia de negocios, la minería de datos es una herramienta de análisis para aprovechar mejor la información histórica. Estas tecnologías tienen una alta variedad de aplicaciones en el mundo real, especialmente, en el campo profesional, tal es el caso de la medicina, en ambientes de trabajo, logística de aprovisionamiento de materiales y productos, fraudes, relación con clientes, rutas de viajes (minería de grafos), entre otras.

La minería de datos pertenece a un área más amplia conocida como el descubrimiento del conocimiento sobre las bases de datos (*del inglés KDD knowledge discovery in databases*), basadas en cinco etapas [8].

1. selección de datos,
2. pre procesamiento,
3. transformación,
4. minería (algoritmos),
5. interpretación y evaluación.

---

<sup>8</sup> Microsoft, 2018

La minería de datos es un proceso compuesto por varios pasos o etapas que deben generar un nuevo conocimiento o, en su defecto, guiar hasta la consecución del mismo. La minería de datos identifica comportamientos, patrones, agrupamientos y clasificaciones ocultos, fruto de las actividades cotidianas de las organizaciones. Estos resultados, difíciles de identificar a simple vista por el humano, son revelados por las herramientas de datamining. Los resultados son presentados en informes gráficos de relativa facilidad de interpretación. Con el conocimiento revelado por el datamining, cada organismo tiene información de valor para tomar buenas decisiones que les permita ser más competitivos e innovadores.

También es el factor que impulsa a realizar cambios a los modelos de negocio con efectividad y mayor rapidez, en muchos casos, excluyen a las organizaciones que no utilizan estas técnicas y herramientas [9]. Para todos estos casos, la minería de datos genera el nuevo conocimiento a partir de analizar los grandes volúmenes de datos, con algoritmos muy complejos como árboles de decisión, redes bayesianas, redes neuronales, clusterización, entre otros. Ver Figura 68.

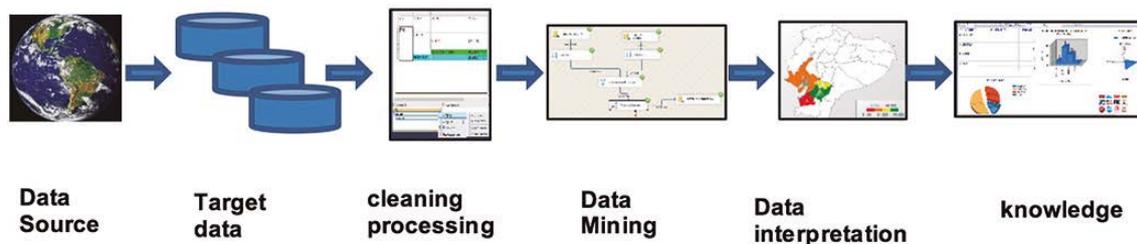


Figura 68. Despliegue del proceso para el descubrimiento del conocimiento.

## 5.2 Minería de texto (*Textmining*)

Es una técnica de las tecnologías emergentes para extraer grandes cantidades de datos no estructurados desde la web, redes sociales, entre otros medios. Tiene por objetivo transformarlos en datos comprensibles, y luego, poder establecer patrones, agruparlos, clasificarlos y generar conocimiento (Botta-Ferret & Cabrera-Gato, 2007). La minería de texto también permite encontrar relaciones usando datos estructurados y no estructurados para generar conocimiento (Contreras-Barrera, 2014). Otros autores hablan de la minería de texto en la obtención de información valiosa, útil e interesante de textos no estructurados, que se encuentran en versión digital para ser consumidos posteriormente (Dang, 2014) (Kalra, 2013).

### 5.2.1 Caso práctico

Para el siguiente caso trabajamos con Python, para extraer datos desde la red social Twitter, utilizando las llaves de acceso proporcionadas por esta red. La extracción de datos la realizamos por un lapso de 4 horas continuas, durante 5 días de la sema-

<sup>9</sup> Microsoft, 2018

na, considerando que este tiempo es el que dura la popularidad del tema evaluado y comentado en la red social.

*El propósito de revisar este caso práctico es presentar algunos patrones de los datos extraídos desde las redes sociales, acerca de un determinado tema el que se lo clasificará por la polaridad sentimental (Rodríguez, et al., 2019).*

Se toman datos de eventos ocurridos de tipo climático, social, político y deportivo para analizarlos de forma masiva. El número de tweets extraídos fueron 5737, es decir, comentarios generados. Luego, se procedió a organizar y limpiar la información obtenida. Después, a clasificar, organizar interpretar esta información y presentar los resultados como aporte para un nuevo conocimiento. El código de extracción es detallado a continuación.

```
#Conexión con Twitter

#Importar librerías y credenciales para el token de acceso
from tweepy import Stream
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener
import sqlite3
from sqlite3 import Error
import json
import csv

# URL para crear Apps: https://apps.twitter.com/
app_id = "9mwmv7Yt5R783Jry1bYEiGe2n"
app_secret = "NQazRU4BpffmwKmWD2pxhebAjitfEFOPHErOKTrWjBrn4jpXSE"
token_access = "442102127-wvZfxbOkw3V8eQfbJgI7iSkyokI96tw8U2u5ROj0"
token_secret = "L0PJfiYiT0cToCgZKDv6fMJD64iPbr0V9D6S0eHiVXRuW"
PATH_DATABASE = "E:\base_clima\desastre_tweet.db"

print('listo conexión')

#Construir la estrucutra de la base de datos para almacenar los datos no estructura-
dos

def createTable(db_file):
    """ create a database connection to a SQLite database """
```

```

try:

    conn = sqlite3.connect(db_file)

    c = conn.cursor()

    c.execute("""CREATE TABLE evento3
        (tweet_text text, username text)""")

    conn.commit()

except Error as e:

    print(e)

finally:

    conn.close()

createTable(PATH_DATABASE)

print('listo conexión con base de datos')

#preparar las palabras clave

import re

class TwitterStream(StreamListener):

    def __init__(self, connection):

        self.conn = connection

    def on_data(self, data):

        # Procesar los datos para guardarlos

        json_python = json.loads(data)

        tweet_text = json_python["text"]

        tweet_new = re.sub(r'[\w]', ' ', tweet_text)

        user_name = json_python["user"]["screen_name"]

        query_text = "INSERT INTO evento3
VALUES ({}{})".format("'" + tweet_new + "'", "'" + user_name + "'")

        c = self.conn.cursor()

        c.execute(query_text)

        self.conn.commit()

        return(True)

    def on_error(self, status):

```

```

    print (status)

#ejecutar primero

print('listo')

#Extraccion de datos y medicion del tiempo empleado

%%time

auth = OAuthHandler(app_id, app_secret)

auth.set_access_token(token_access, token_secret)

conn = sqlite3.connect(PATH_DATABASE)

twitterStream = Stream(auth, TwitterStream(conn))

print('Ejecutando...')

try:

    twitterStream.filter(track=["Hoy no circula", "congestion de vehiculos guayaquil",
"wikileaks", "trafico autos Ecuador"])

except:

    print('Detenido')

```

*Primer resultado:* Copa América, luego de los comentarios de un jugador emblemático de una selección participante en este evento: “Lamentablemente la corrupción, los árbitros y todo eso no permiten que la gente disfrute del show”, además: “está todo armado para Brasil”. Este tema en la red social generó un 78% de sentimiento positivo y el 22% de sentimiento negativo. Ver Figura 69.



**Figura 69.** Muestra el análisis de *Text Mining* realizado desde *Python*.

## 5.3 Tipos de algoritmos de minería de datos

### 5.3.1 Árboles de decisión de Microsoft

Un algoritmo importante dentro de la minería de datos son los árboles de decisión. Este es un algoritmo de clasificación y regresión proporcionado por *Microsoft SQL Server Analysis Services*, para el modelado de predicción de atributos discretos y continuos (Otukeya & Blaschke, 2010) y (MoonChae et al., 2001).

Para los atributos discretos, el algoritmo hace pronósticos basándose en las relaciones entre las columnas de entrada de un conjunto de datos. Utiliza los valores, conocidos como estados, de estas columnas, para predecir los estados de una columna que se designa como elemento de predicción. Específicamente, el algoritmo identifica las columnas de entrada que se correlacionan con la columna de predicción. Por ejemplo, en un escenario para predecir qué clientes van a comprar una bicicleta, se tiene que, nueve de diez clientes jóvenes compran una bicicleta, pero solo lo hacen dos de diez clientes de edad mayor. El algoritmo infiere que la edad es un buen elemento de predicción en la compra de bicicletas. El árbol de decisión realiza pronósticos basándose en la tendencia hacia un resultado concreto. Para los atributos continuos, el algoritmo usa la regresión lineal con el objeto de determinar dónde se divide un árbol de decisión.

Si se define más de una columna como elemento de predicción, o si los datos de entrada contienen una tabla anidada que se haya establecido como elemento de predicción, el algoritmo genera un árbol de decisión independiente para cada columna de predicción.

### 5.3.2 Cómo funciona el algoritmo de árboles de decisión

El algoritmo de árboles de decisión de *Microsoft* genera un modelo de minería de datos mediante la creación de una serie de particiones en el árbol. Estas particiones se representan como nodos. El algoritmo adiciona un nodo al modelo cada vez que una columna de entrada tiene una correlación significativa con la columna de predicción. La forma en que el algoritmo determina una división varía en función de si predice una columna continua o una columna discreta.

El histograma muestra que la edad de una persona ayuda a distinguir si esa persona comprará una bicicleta. Ver Figura 70.

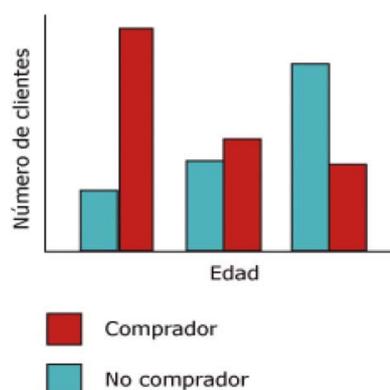
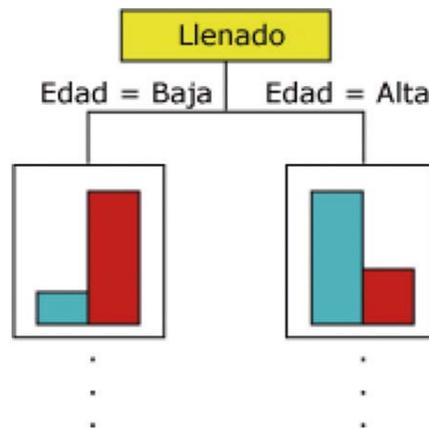


Figura 70. Histograma de un árbol de decisión.

La correlación que aparece en el diagrama hará que el algoritmo de árboles de decisión de Microsoft cree un nuevo nodo en el modelo. Ver Figura 71.



**Figura 71.** Histograma de un árbol de decisión nuevo nodo de análisis.  
**Fuente:** (Microsoft, 2015)

### 5.3.3 *Naive Bayes*

El Microsoft algoritmo *Bayes Naive* es un algoritmo de clasificación basado en teoremas de Bayes y se puede manejar para el modelado de predicción y exploración. La palabra Naive (ingenuo en inglés) del término *Bayes Naive* procede del hecho que el algoritmo usa técnicas Bayesianas, pero no tiene en cuenta las dependencias que puedan existir.

Desde el punto de vista computacional, el algoritmo es menos complejo que otros algoritmos de Microsoft. Por tanto, resulta útil para generar rápidamente modelos de minería de datos que detectan las relaciones entre las columnas de entrada y las columnas de predicción. Se puede utilizar este algoritmo para realizar la indagación inicial de los datos y, más adelante, aplicar los resultados para crear modelos de minería adicionales con otros algoritmos más complejos y precisos desde el punto de vista computacional.

**Caso de estudio:** Como parte de su estrategia promocional, el departamento de comercialización de la empresa *Adventure Works Cycles* ha decidido atraer a posibles clientes, realizando un envío por correo de folletos. Para minimizar costos, desean enviar los folletos solo a los clientes de los que esperan recibir respuesta. La empresa almacena información en una base de datos sobre datos demográficos y respuestas a envíos de correo anteriores. Desean utilizar estos datos para ver el modo en que los datos demográficos como la edad o la ciudad pueden ayudarles a predecir la respuesta a una promoción, comparando los clientes potenciales con los que tienen características similares y con los que han adquirido productos de la empresa en el pasado. En concreto, lo que desean es ver las diferencias entre los clientes que adquirieron una bicicleta y los que no lo hicieron [10]. Mediante el algoritmo *Bayes Naive* de

<sup>10</sup> Microsoft (2018) Algoritmos de minería de datos (*Analysis Services: Minería de datos*)

*Microsoft* (Gavilán, 2019), el departamento de comercialización puede predecir rápidamente un resultado de un perfil de cliente concreto y, por tanto, puede determinar qué clientes responderán a los folletos con más probabilidad. Con el Visor *Bayes Naive* de *Microsoft de SQL Server Data Tools (SSDT)*, también se puede investigar visualmente qué columnas de entrada específicas contribuyen a conseguir respuestas positivas a los folletos.

**Cómo funciona el algoritmo:** El algoritmo *Bayes Naive* de *Microsoft* calcula la probabilidad de cada estado de cada columna de entrada, dado cada posible estado de la columna de predicción. Para entender cómo funciona, utilice el Visor *Bayes Naive* de *Microsoft de SQL Server Data Tools (SSDT)*, para consultar una representación visual del modo en que el algoritmo distribuye los estados. Ver Figura 72.

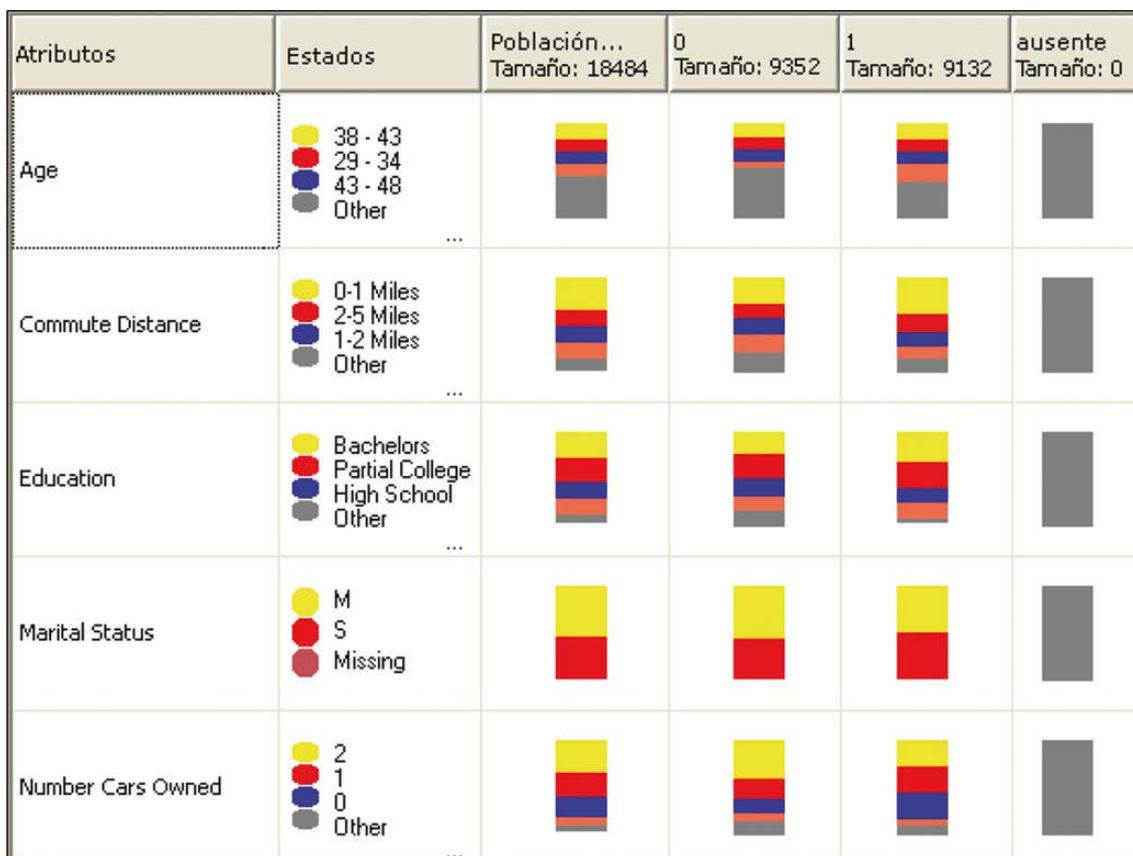


Figura 72. Histograma de un árbol de decisión nuevo nodo de análisis.

Fuente: (Microsoft, 2015)

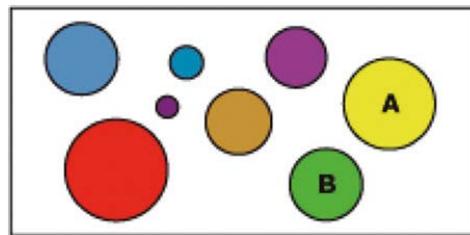
En este caso, el Visor *Bayes Naive* de *Microsoft* muestra cada columna de entrada del conjunto de datos e indica cómo se distribuyen los mejores resultados, dado cada estado de la columna de predicción.

### 5.3.4 Algoritmo de clústeres de *Microsoft*

Este es el algoritmo de segmentación o clústeres. Recorre en iteración un conjunto

de datos para agruparlos en clústeres que contengan características similares de los casos. Estas agrupaciones son útiles para la exploración de datos, la identificación de anomalías y la creación de predicciones (Gallego et al., 2015).

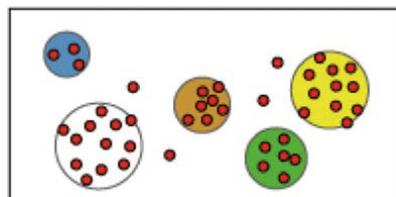
Los modelos de agrupación en clústeres igualan las relaciones en un conjunto de datos que no se podrían derivar lógicamente a través de la observación casual. Por ejemplo, fácilmente puede admitir que las personas que viajan para sus trabajos en bicicleta no viven regularmente a grandes distancias del trabajo. Sin embargo, el algoritmo puede hallar otras características que no son evidentes acerca de los trabajadores que se desplazan en bicicleta. En el siguiente diagrama, el clúster **A** representa los datos sobre las personas que suelen conducir hasta el trabajo, en tanto que, el clúster **B** representa los datos sobre las personas que van hasta allí en bicicleta. Ver Figura 73.



**Figura 73.** Representación de los clústeres A y B.  
**Fuente:** (Microsoft, 2015)

**Caso de estudio:** Considere un grupo de personas que comparten información demográfica similar y que obtienen productos equivalentes de la empresa *Adventure Works*. Este grupo de personas representa un clúster de datos. En una base de datos pueden existir varias agrupaciones como estas. Mediante la observación de las columnas que forman un clúster, puede ver con mayor claridad la forma en que los registros de un conjunto de datos se relacionan entre sí.

**Cómo funciona el algoritmo de clústeres:** Para Microsoft Developer Network (2015), el algoritmo de clústeres de *Microsoft* identifica primero las relaciones de un conjunto de datos y genera una serie de clústeres basándose en ellas. Un gráfico de dispersión es una forma útil de representar visualmente el modo en que el algoritmo agrupa los datos, tal como se muestra en el siguiente diagrama. El gráfico de dispersión representa todos los casos del conjunto de datos; cada caso es un punto del gráfico. Los clústeres agrupan los puntos del gráfico e ilustran las relaciones que identifica el algoritmo. Ver Figura 74.



**Figura 74.** Gráfico de dispersión para todos los grupos de datos.  
**Fuente:** (Microsoft, 2015)

### 5.3.5 Algoritmo de red neuronal de *Microsoft*

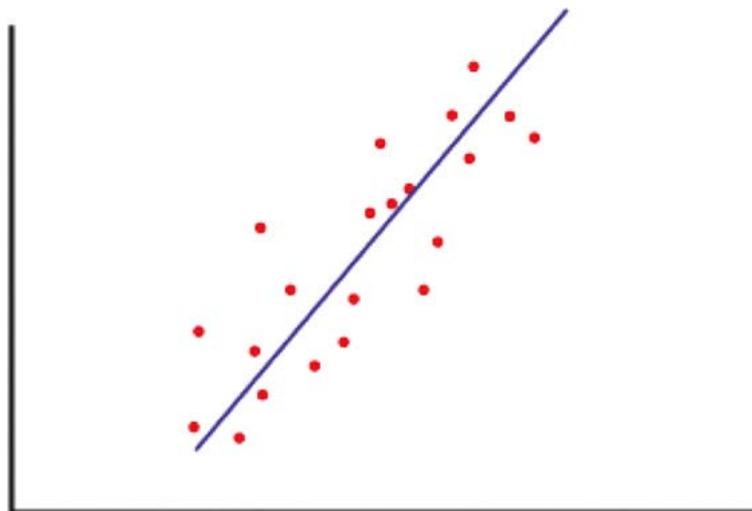
El algoritmo de red neuronal de *Microsoft* es una ejecución de la arquitectura de red neuronal populares y adaptable para el aprendizaje automático. El algoritmo marcha mediante la prueba de cada estado posible del atributo de entrada y calcula las probabilidades para cada combinación basándose en los datos de entrenamiento. Puede usar estas probabilidades para tareas de clasificación o regresión, para pronosticar un resultado en función de algunos atributos de entrada. Una red neuronal también puede utilizarse para el análisis de asociación [11].

Cuando se crea un *DataminiG*, con el algoritmo de red neuronal de *Microsoft*, puede incluir varias salidas, y el algoritmo creará varias redes. Este contiene el número de redes incluidas en un modelo de *DataminiG* único. Esto depende del número de Estados (o valores de atributo) en las columnas de entrada. También depende del número de columnas de predicción que utiliza el patrón de minería de datos y el número de estados en dichas columnas.

### 5.3.6 Algoritmos de regresión lineal de *Microsoft*.

El algoritmo de regresión lineal de *Microsoft* es una variación del algoritmo de árboles de decisión de *Microsoft* que ayuda a calcular una relación lineal entre una variable independiente y otra dependiente y, luego, a utilizar esa relación para la predicción.

La correlación toma la forma de una ecuación para la línea que mejor represente una serie de datos. La línea del siguiente diagrama muestra la mejor representación lineal de los datos. Ver Figura 75.



**Figura 75.** Línea de mejor representación del método de regresión lineal.

**Fuente:** (Microsoft, 2015)

<sup>11</sup> Microsoft (2018) Algoritmos de minería de datos (*Analysis Services: Minería de datos*)

En el diagrama se puede observar que tiene un error asociado con su distancia, en relación a la línea de regresión. Aquí se puede ajustar en los puntos a y b, en la ecuación de regresión, para normalizar el ángulo y la ubicación de la línea de regresión. Puede obtener la ecuación de regresión ajustando los puntos a y b hasta que la suma de los errores que están asociados a todos los puntos alcance su valor mínimo. También existen, otros modelos de regresión que manejan más variables y también, métodos no lineales de regresión. Sin embargo, la regresión lineal es un método útil y conocido para modelar una respuesta a un cambio de algún factor subyacente.

Ejemplo: Puede utilizar la regresión lineal para determinar una relación entre dos columnas continuas. Puede utilizar la regresión lineal para calcular una línea de tendencias en los datos de fabricación o ventas. También podría utilizar la regresión lineal como precursor para el desarrollo de modelos de minería de datos más complejos, con el fin de evaluar las relaciones entre las columnas de datos. Aunque hay muchas maneras de calcular la regresión lineal que no requieren herramientas de minería de datos.

La ventaja de utilizar el algoritmo de regresión lineal de *Microsoft* para esta tarea es que se calculan y se prueban automáticamente todas las posibles relaciones entre las variables. No tiene que seleccionar un método de cálculo, como, por ejemplo, para resolver los mínimos cuadrados. Sin embargo, la regresión lineal podría simplificar en exceso las relaciones en escenarios en los que varios factores afectarían al resultado. El funcionamiento del algoritmo de regresión lineal es una variación del algoritmo de árboles de decisión. Al escoger el algoritmo de regresión lineal de *Microsoft*, se invoca un caso especial del algoritmo de árboles de decisión, con parámetros que limitan el comportamiento del algoritmo y requieren ciertos tipos de datos de entrada. Además, en un modelo de regresión lineal, el total de datos se utiliza para calcular las relaciones en el primer paso, mientras que, en un modelo de árboles de decisión estándar, los datos se dividen repetidamente en árboles o subconjuntos más pequeños. Este algoritmo es utilizado en varios casos como análisis de datos en computadoras cuánticas con regresión lineal, con optimización de mínimos cuadrados (Schuld et al., 2016), o con datos de un entorno distribuido en una red, evaluando cada nodo y sus nodos vecinos (Yuan et al., 2019).



# CAPÍTULO VI

Big data o grandes datos



## 6.1 Visión general

El término *Big Data* se origina a partir de la expansión exponencial de los datos en la Web alrededor del mundo, donde, gestionar tanta información con las herramientas convencionales es una tarea cada vez más compleja, puesto que, los dispositivos computacionales no pueden crecer a la misma velocidad, ni en su capacidad de almacenamiento, como tampoco a la velocidad de procesamiento. Por esta razón, el *Big Data* se crea para combatir estos dos factores importantes dentro de las ciencias computacionales, y de este modo, dar salida a diversas aplicaciones en el campo de los negocios, investigación, producción y la comunicación de la sociedad en general. La mayoría de autores que hablan acerca de la evolución del *Big Data* coinciden en que el tamaño de los datos ha superado las capacidades de las máquinas individuales. De allí que, entremos en una nueva necesidad de técnicas para escalar la gestión de la información en varios nodos, de manera distribuida, denominada clústeres (Zaharia et al., 2016). Para trabajar en *Big Data* existen actualmente dos modelos de desarrollo, el uno es *Hadoop*, con *MapReduce HDFS* y el otro es *Spark*, con RDD compatible con herramientas de desarrollo como *Java*, *Python*, *Scala* y *R*.

## 6.2 Hadoop

Es una herramienta de cálculo de grandes cantidades de datos que pertenece a *Google* y que nació a partir de un proyecto de búsqueda de datos en la Web junto con *Nutch*, como rastreador Web, y *Hadoop*, de código abierto para el procesamiento de datos con cómputo distribuido en varios clúster, nodos o núcleos. Las características de *Hadoop* son poder de cómputo, tolerancia a fallos, flexibilidad, bajo costo y escalabilidad. Todo esto, solo agregando más nodos (SAS Institute Inc., 2021), dando origen a las Vs principales de *Big Data*, como son, volumen, velocidad y variabilidad. Entre las aplicaciones relevantes de *Hadoop* están un sistema de recomendación basado en la Web como Facebook de personas que quizás conozcas. Otro es, LinkedIn, sobre empleos que quizás te interesen, o *Netflix*, *eBay* para analizar artículos que quizás desees adquirir.

## 6.3 HDFS

Es un potente sistema de archivos o estructura de datos que maneja grandes cantidades de datos en forma distribuida (del inglés, *Hadoop distributed file system*). Son archivos que permiten procesar, leer, escribir y crecer por encima de la capacidad de un solo ordenador y dividir el archivo para distribuir el contenido en varios discos duros u ordenadores (denominados *clusters*) (Joyanes-Aguilar, 2016). Se usa en clústeres distribuidos (ordenadores) a gran escala de diversas procedencias, garantizando un alto rendimiento, disponibilidad y flexibilidad a la tolerancia a fallos (Kala-Karun & Chitharanjan, 2013). Sin embargo, una debilidad de los DDFS se da al aplicar en el aprendizaje profundo (*Deep Learning*), donde es poco eficiente por el problema

de archivos pequeños, puesto que estos archivos cuyo tamaño es menor que el de un HDFS restringe la expansión de un clúster. Por otro lado, aumenta el tiempo de acceso por tener que saltar con mayor frecuencia de un nodo a otro, afectando el rendimiento. Este tema se amplía en Dong, *et al.* (2010).

## 6.4 Spark

Spark es una herramienta o lenguaje de programación para implementar *Big Data* con gran eficiencia, basado en RDD, Conjunto de Datos Distribuidos Resilientes (*Resilient Distributed Dataset*), similar a su par HDFS. Spark consta de un programa controlador que primero ejecuta la función principal (*main*) del usuario, y luego, ejecuta las operaciones en varios *clusters* en paralelo, logrando velocidad y gestión de grandes volúmenes de datos (Spark, 2021). *Spark* trabaja con varios intérpretes potentes como *Scala*, *Java*, *R* y *Python*, en el caso de este último, usan distintas bibliotecas como *Numpy*, *Pandas*, *C*, entre otras.

Para desarrollar *Big Data* con *Python* y *Spark*, se puede proceder de dos maneras. La primera es usando el *bin/spark-submit*, ubicado en el directorio *Spark*; con esto se consigue cargar las bibliotecas de java, *Scala* y *Spark* y enviar aplicaciones a un clúster. Igualmente, se puede utilizar *bin/spark-submit* para iniciar un Shell interactivo. Sin embargo, para usar HDFS, debe compilar un *PySpark* que esté vinculado a la versión del HDFS. Luego de esto, necesita importar algunas clases como:

```
From pyspark import SparkContext, SparkConf
```

Estas clases sirven para configurar el entorno o contexto de trabajo de Python y Spark. Para iniciar una aplicación en *Spark* debe crear un objeto *SparkConf* que defina la información sobre la aplicación y componentes a usar. Luego, ya se puede crear el objeto *SparkContext*, que indica a *Spark* como acceder a un clúster, tal como se indica a continuación.

```
Conf = SparkConf().setAppName(appName).setMaster(master)
sc = SparkContext(conf=Conf)
```

Donde, *appName* es un nombre de la aplicación y se muestra en el clúster y masters, es una *url* de clusters *Spark* para ejecutar de modo distribuido, o una cadena local para ejecutar de modo local.

La otra forma es utilizando *Google Colab*. Para esto se procede de la siguiente manera:

1. Instalar: `!pip install pyspark`
2. Importar clases: `import pyspark`
3. Crear el contexto de trabajo: `sc = pyspark.SparkContext('local [*]')`

Luego de configurar estos pasos y comandos básicos, ya podemos trabajar con diversos ejemplos, como se muestra a continuación:

Cargar archivo de datos:

```
Datos = [ ]

with open('Análisis.txt') as f:

    for línea in f:

        datos.extend(línea.split())

print(datos)

Crear y cargar un RDD

rdd1=sc.parallelize(datos)
```

Una vez creado el RDD, ya podemos trabajar con todas las transformaciones, como:

```
filter

#filtrar ciertas palabras de un texto

x=rdd1.filter(lambda línea: "redes" in línea)

x.count()

#resultado
```

También, podemos dividir todas las palabras de un texto y contar, generando un diccionario de clave valor con la palabra y el número de repeticiones de la misma, respetivamente.

```
cp=rdd1.flatMap(lambda línea: línea.split(" ")).countByValue()

cp
```

Para poder realizar otras transformaciones o acciones con el RDD, es preciso convertir en ítems de tuplas por cada clave, valor.

```
s=cp.items()

s
```

Una vez convertido en tuplas, se puede realizar la búsqueda de las palabras con mayor repetición y ordenar de forma descendente.

```

buscar=sc.parallelize(s).sortBy(lambda x: x[1],ascending=False).take(25)
buscar
#método ascendente
ord=sc.parallelize(bus).sortBy(lambda x: x[1],ascending=True).take(25)
ord

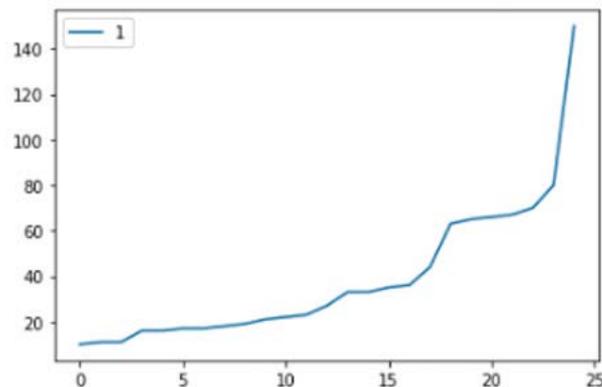
```

Graficar datos de un *RDD* con *BIG DATA* y *Spark*. Ver Figura 76. Para graficar, es preciso convertir a *dataset* de tipo *Numpy* o *Pandas*.

```

import matplotlib.pyplot as plt
import pandas as pd
df=pd.DataFrame(ord)
df.plot()
plt.savefig("TendenciaTexto.jpg")
#Resultado

```



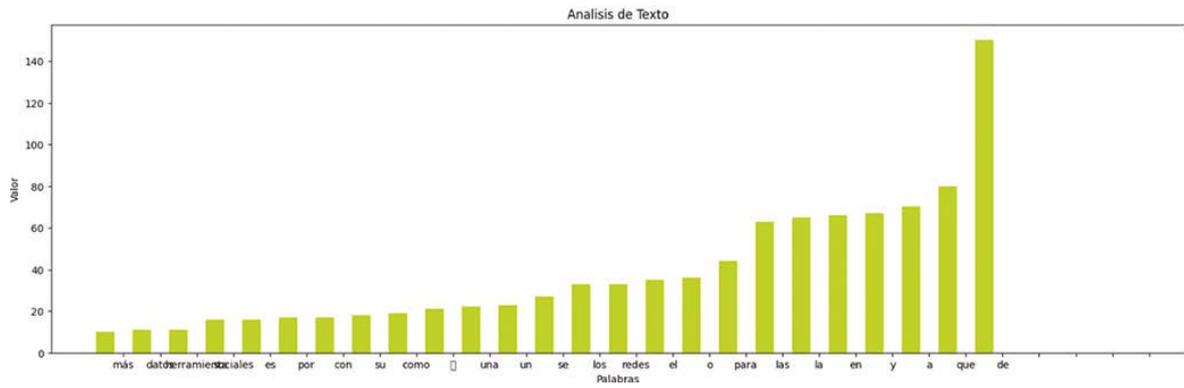
**Figura 76.** Gráfico de análisis de la tendencia de las palabras repetidas de un texto usando *Spark*.

Otro gráfico más interesante, aplicando técnicas de *Big Data*, podría ser como el de la Figura. 77. Para esto, primero preparamos el *dataset*.

```

import numpy as np
df.columns = ['Palabras', 'Valor']
X = np.arange(30)
#Luego preparamos el grafico
plt.figure(figsize=(15,20),dpi=100)
plt.bar(df['Palabras'], df['Valor'], color = "#CCFF00", width = 0.50)
plt.title('Analisis de Texto')
ax = plt.subplot()
plt.xticks(X+0.50, df['Palabras'])
ax.set_xticklabels(df['Palabras'])
ax.set_xlabel('Palabras')
ax.set_ylabel('Valor')

```



**Figura 77.** Gráfico analítico del conteo de palabras repetidas de un texto con *Spark* y *Python*.

### Reconocer:

Que parte del contenido de este libro tiene resultados del proyecto de investigación “Ciudades Resilientes y Sostenibles en el Ecuador”.

## REFERENCIAS BIBLIOGRÁFICAS:

- Bustamante Martínez, A., Galvis Lista, E. A., & Gómez Flórez, L. C. (2013). Técnicas de modelado de procesos de ETL: una revisión de alternativas y su aplicación en un proyecto de desarrollo de una solución de BI. *Scientia et Technica* Año XVIII, Vol. 18, No 1, Abril de 2013. Universidad Tecnológica de Pereira. ISSN 0122-1701.
- Hartzler, A., Chaudhuri, S., Fey, B., Flum, D., & Lavallee, D. (2015). Integrating Patient-Reported Outcomes into Spine Surgical Care through Visual Dashboards: Lessons Learned from Human-Centered Design. *EGEMS (Wash DC)*. 2015 Mar 13;3(2):1133. doi: 10.13063/2327-9214.1133.
- Morales, M. R., & Catucuamba-Jima, B. E. (2019). Inteligencia de Negocios y Analítica de Datos, una Propuesta de Mejora. Repositorio digital Universidad Central del Ecuador: <http://www.dspace.uce.edu.ec/handle/25000/19365>
- Negash y, S., & Gray, P. (2008). *Business Intelligence. Handbook on Decision Support Systems 2* pp 175–193.
- Silva, G. E., Zapata, V. M., Morales, K. P., & Toaquiza, L. M. (2019). Análisis de metodologías para desarrollar Data Warehouse aplicado a la toma de decisiones. *Ciencia Digital*, <https://doi.org/10.33262/cienciadigital.v3i3.4..922>, 3(3.4).
- Akhtar, N., Shafait, F., & Ajmal, M. (2014). Sparse Spatio-spectral Representation for Hyperspectral Image Super-resolution. *Computer Vision, ECCV 2014, 13th European Conference on Computer Vision, Proceedings, Part VII, Zürich, CH, Sep 6-12*.
- An IDC White Paper. (2016). *The Fusion of Business and Technology in the Age of Digital Transformation*. IDC CEMA Research Presents, Oracle.
- Anshari, M., & Almunawar, M. N. (2012). Framework of Social Customer Relationship Management in E-Health Services. *Journal of e-Health Management*, Vol. 2012 (2012), Article ID 766268, 15 pages, DOI: 10.5171/2012.766268.
- Attaran, M. (2004). Exploring the relationship between information technology and business process reengineering. *Information & Management*, Volume 41, Issue 5, Pages 585-596.
- Azeroual, O., & Theel, H. (2018). The Effects of Using Business Intelligence Systems on an Excellence Management and Decision-Making Process by Start-up Companies: A Case Study. DOI: 10.18775/ijmsba.1849-5664-5419.2014.43.1004.
- Azeroual, O., & Theel, H. (2018). The Effects of Using Business Intelligence Systems on an Excellence Management and Decision-Making Process by Start-Up Companies: A Case Study. *International Journal of Management Science and Business Administration*, Volume 4, Issue 3, Pages 30-40.
- BAhare-Kolahi, J., & Valmohammadi, C. (2015). Key performance indicators measurement in service business: a fuzzy VIKOR approach. *Total Quality Management & Business Excellence*.
- Barata, J., Ferro, R., & Ferreira, J. (2014). *My Traffic Manager*. My Traffic Manager, Volume 17.
- Benymol, J., & Abraham, S. (2020). Performance analysis of NoSQL and relational databases with MongoDB and MySQL. *Materials Today: Proceedings*, Vol. 24.

- <https://doi.org/https://doi.org/10.1016/j.matpr.2020.03.634>
- Bhaskar, H. L., & Singh, R. (2014). Business process reengineering: a recent review. *GJBM*, Vol. 8 No. 2.
- Birant, D., & Kut, A. (2007). ST-DBSCAN: An algorithm for clustering spatial-temporal data. *Data & Knowledge Engineering*, Volume 60, Issue 1.
- Boeing, G. (2017). OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, <https://doi.org/10.1016/j.compenvurbsys.2017.05.004>, 65.
- Boeing, G. (2020). Urban Street Network Analysis in a Computational Notebook. [arXiv:2001.06505v1](https://arxiv.org/abs/2001.06505v1) [physics.soc-ph] 17 Jan 2020.
- Borissova , D., Cvetkova, P., Garvanov, I., & Garvanova, M. (2020). A Framework of Business Intelligence System for Decision Making in Efficiency Management. In: Saeed, K., Dvorský, J. (eds) *Computer Information Systems and Industrial Management*. CISIM 2020.
- Boston College ITS. (2015). Information Technology Strategic Plan. Boston College ITS, version 1.1, spring 2015.
- Botta-Ferret, E., & Cabrera-Gato, J. E. (2007). Minería de textos: una herramienta útil para mejorar la gestión. *Acimed*; 16(4) .
- Cevallos-Cuenca, J., & Vera-Flores, H. (2020). Inteligencia de negocios de nueva generación en manufactura. *Polo del Conocimiento*, 5(2), 294-315.
- Clark, A., & Clark, E. (2004). *Diccionario Inglés a Español de Computación e Internet: English-to-Spanish Computer and Internet Dictionary*.
- Contreras-Barrera, M. (2014). Minería de texto: Una visión actual. *Biblioteca Universitaria*, vol. 17, núm. 2.
- Cordero-Guzman,, D. M., & Guillermo-Rodríguez , M. G. (2017). LA INTELIGENCIA DE NEGOCIOS: UNA ESTRATEGIA PARA LA GESTIÓN DE LAS EMPRESAS PRODUCTIVAS. // BUSINESS INTELLIGENCE: A STRATEGY FOR THE MANAGEMENT OF PRODUCTIVE ENTERPRISES. *Ciencia UNEMI*, Vol. 10, Núm. 23.
- Dang, S. (2014). Text Mining : Techniques and its Application. *IJETI International Journal of Engineering & Technology Innovations*, Vol. 1 Issue 4,.
- Dasgupta, S. (2016). *Computer Science: A Very Short Introduction*. Oxford University Press.
- Dong, B., Qiu, J., Zheng, Q., Zhong, X., Li, J., & Li, Y. (2010). A Novel Approach to Improving the Efficiency of Storing and Accessing Small Files on Hadoop: A Case Study by PowerPoint Files. 2010 IEEE International Conference on Services Computing (págs. 65-72). Miami, FL, USA: IEEE.
- Dornelas, J. S., Rodrigues-de-Souza, K. R., & Amorim, A. N. (2017). CLOUD COMPUTING: SEARCHING ITS USE IN PUBLIC MANAGEMENT ENVIRONMENTS. *JISTEM*, Vol. 14, No. 2, May/Aug., 2017 pp. 281-306.
- Einsiedel, E. F. (2009). *Emerging Technologies: From Hindsight to Foresight*. UBC Press.
- Elkhodr, M., Shahrestani , S., & Cheung , H. (2016). EMERGING WIRELESS TECHNOLOGIES IN THE INTERNET OF THINGS: A COMPARATIVE STUDY. *International Journal of Wireless & Mobile Networks (IJWMN)* Vol.

- 8, No. 5, DOI: 10.5121/ijwmn.2016.8505 .
- Elnaga, A. A., & Imran, A. (2014). The Impact of Employee Empowerment on Job Satisfaction Theoretical Study. *American Journal of Research Communication*, Vol 2(1).
- Erek, K. (2019). FROM GREEN IT TO SUSTAINABLE INFORMATION SYSTEMS MANAGEMENT: MANAGING AND MEASURING SUSTAINABILITY IN IT ORGANIZATIONS. <https://www.researchgate.net/publication>.
- Felhi, F., & Akaichi, J. (2012). Adaptation of Web services to the context based on workflow: Approach for self-adaptation of service-oriented architectures to the context. arXiv:1211.4867, *International Journal of Web & Semantic Technology (IJWesT)* Vol.3.
- Gallego, J., Navarro, L., & Castillo, L. (2015). Aplicación de técnicas de minería de datos en atención primaria en salud (aps) para el análisis de riesgos en mujeres gestantes de la población manizaleña atendida por assbasalud. *Biosalud* vol.14 no.2 Manizales jJuly/Dec. 2015.
- Gavilán, I. G. (2019). La carrera digital: Cómo transformar tu compañía para triunfar en el mundo digital. Exlibric.
- Ghiglione, F. A. (2021). EL CUADRO DE MANDO INTEGRAL COMO HERRAMIENTA DE EFICIENCIA EN LA GESTIÓN EMPRESARIAL. *Cienc. adm.* no.18 La Plata jul. 2021.
- Gold, R. (2019). Building the IT Organization Balanced Scorecard. Information Systems Audit and Control Association. All rights reserved. [www.isaca.org](http://www.isaca.org).
- Golfarelli, M., Mandreoli, F., Penzo, W., Rizzi, S., & Turrinchia, E. (2012). OLAP query reformulation in peer-to-peer data warehousing. *Information Systems*, Volume 37, Issue 5, July 2012, Pages 393-411. <https://doi.org/https://doi.org/10.1016/j.is.2011.06.003>
- Gross, J. L., Yellen, J., & Anderson, M. (2018). *Graph Theory and Its Applications*. CRC Press.
- Gunasekaran, A., Irani, Z., King-Lun, C., Filippi, L., & Papadopoulos, T. (2014). Performance measures and metrics in outsourcing decisions: A review for research and applications. *Int. J. Production Economics*.
- Haller, M. (2006). *Emerging Technologies of Augmented Reality: Interfaces and Design: Interfaces and Design*. Idea Group Pub.
- Hee-Lee, J., & Chan-Park, S. (2005). Intelligent profitable customers segmentation system based on business intelligence tools. *Expert Systems with Applications*, Volume 29, Issue 1.
- Holstein, T., Dodig-Crnkovic, G., & Pelliccione, P. (2018). Ethical and Social Aspects of Self-Driving Cars. arXiv:1802.04103v1 [cs.CY, Gothenburg, Sweden.
- Ibrahim, M. (2015). PARADIGM OF EMPLOYEES EMPOWERMENT: PROPOSAL FOR BUSINESS ACTORS IN ACEH PROVINCE. *European Journal of Research and Reflection in Management Sciences*, Vol. 3 No. 3.
- Jituri, S., Fleck, B., & Ahmad, R. (2018). A Methodology to Satisfy Key Performance Indicators for Successful ERP Implementation in Small and Medium Enterprises. *International Journal of Innovation, Management and Technology*, Vol. 9, No. 2.

- Joyanes-Aguilar, L. (2016). *Big Data, Análisis de grandes volúmenes de datos en organizaciones*. México: Alfaomega Grupo Editor.
- Kala-Karun, A., & Chitharanjan, K. (2013). Locality Sensitive Hashing based incremental clustering for creating affinity groups in Hadoop — HDFS - An infrastructure extension. 2013 International Conference on Circuits, Power and Computing Technologies (ICCPCT), 2013, pp. 1243-1249, doi: 10.1109/ICCPCT.2013.6528999. (págs. 1243-1249). Nagercoil, India: IEEE Xplore.
- Kalra, P. (2013). TEXT MINING: CONCEPTS, PROCESS AND APPLICATIONS. *Journal of Global Research in Computer Science*, Volume 4, No. 3.
- Kan, M., Shan, S., & Chen, X. (2016). Multi-view Deep Network for Cross-view Classification. 2016 IEEE Conference on Computer Vision and Pattern Recognition, DOI 10.1109/CVPR.2016.524.
- Kaplan, R. S. (2010). *Conceptual Foundations of the Balanced Scorecard*. Harvard Business School, Harvard University.
- Kaplan, R. S., & Norton, D. P. (2008). *The execution premium*. Harvard Business School Publishing Corporation.
- Kaplan, R., & Norton, D. (2005). El Balanced Scorecard, mediciones que impulsan el desempeño. *Harvard Business Review*, ISSN 0717-9952, Vol. 83, N<sup>o</sup>. 7, 2005 .
- Katayama, H. (2017). Legend and Future Horizon of Lean Concept and Technology. *Procedia Manufacturing*, Volume 11.
- Kent, A., & Williams, J. G. (1996). *Encyclopedia of Computer Science and Technology: Volume 35 - Supplement 20: Acquiring Task-Based Knowledge and Specifications to Seek Time Evaluation*. CRC Press.
- khanam, S., Siddiqui, J., & Talib, F. (2013). Role of Information Technology in Total Quality Management: A Literature Review. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, Volume 2, Issue 8.
- Kim, D., & Lim, U. (2017). Social Enterprise as a Catalyst for Sustainable Local and Regional Development. *Sustainability* 2017, 9, 1427; doi:10.3390/su9081427.
- Kimuyu, P. (2017). *The Practice of Anaesthesia and Technology Advancement*. GRIN Verlag.
- Koenes, A., & Soriano, C. (1996). *The Successful Small Business*, *Gestion de la Calidad Total*. Díaz-de-Santos, MAPCAL, S.A.
- Le, H. -A., Hamani, N., Kermad, L., Dafaoui, E., & ElMhamedi, A. (2015). Development of a Framework of a TMS Benchmark Analysis Applied on an European Airline Company. *IFAC-PapersOnLine*, Volume 48, Issue 3, 2015, Pages 424-429.
- LEE, S. M., OLSON, D. L., & TRIMI, S. (2010). The Impact of Convergence on Organizational Innovation. *Organizational Dynamics*, Vol. 39, No. 3.
- Leverger, C., Lemaire, V., Malinowski, S., Guyet, T., & Roz'e, L. (2018). Day-ahead time series forecasting: application to capacity planning. arXiv:1811.02215v1 [cs.AI] 6 Nov 2018.
- Lewis, G. A. (2010). *Emerging Technologies for Software-Reliant Systems of Systems*. TECHNICAL NOTE CMU/SEI-2010-TN-019.
- Lyu-Jr., J. (1996). Applying Kaizen and automation to process reengineering. *Journal of Manufacturing Systems*, Volume 15, Issue 2, 1996, Pages 125-132.

- Martín, A., Chávez, S., Rodríguez, N., Valenzuela, V., & Murazzo, M. (2013). Bases de Datos NoSql en Cloud Computing. SEDICI.
- Martínez-Lorente, A. R., Sánchez-Rodríguez, C., & Dewhurst, F. W. (2004). The effect of information technologies on TQM: An initial analysis. *International Journal of Production Economics* 89(1):77-93, DOI: 10.1016/j.ijpe.2003.06.001.
- Masucci, A. P., & Molinero, C. (2016). Robustness and closeness centrality for self-organized and planned cities. *Eur. Phys. J. B* (2016) 89: 53.
- Microsoft. (2022). Información general de los cubos OLAP de Service Manager para análisis avanzado. Microsoft: <https://docs.microsoft.com/es-es/system-center/scsm/olap-cubes-overview?view=sc-sm-2022>
- Mlay , S. V., Watundu, S., & Zlotnikov, I. (2013). A quantitative analysis of business process reengineering and organizational resistance: the case of Uganda. *Computer Science & Information Systems, The African Journal of Information Systems*, 5(1).
- Molina-Valencia, R. G., González-Millán, J. J., & Forero, A. F. (2014). Diseño de la planeación estratégica y el cuadro de mando integral balanced scorecard en la empresa social del Estado: Centro de Salud Firavitoba. XXII Congreso Internacional del CLAD sobre la Reforma del Estado y de la Administración Pública, Madrid, España.
- MoonChae, Y., Hee-Ho, S., Won-Cho, K., Lee, D. H., & Ji, S. H. (2001). Data mining approach to policy analysis in a health insurance domain. *International Journal of Medical Informatics*, Volume 62, Issues 2–3.
- Murray, J., & Kidd, W. (2017). *Using Emerging Technologies to Develop Professional Learning*. Routledge.
- O'Connor, R. (2019). *Computer Standards & Interfaces*. School of Computing, Dublin City University, Glasnevin, Dublin 9, Ireland.
- Olszak, C., & Ziemba, E. (2006). Business Intelligence Systems in the Holistic Infrastructure Development Supporting Decision-Making in Organisations. *Interdisciplinary Journal of Information, Knowledge, and Management* Volume 1, 2006.
- Otukeya, J. R., & Blaschke, T. (2010). Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms. *International Journal of Applied Earth Observation and Geoinformation*.
- Pankowska, M. (2019). Information Technology Outsourcing Chain: Literature Review and Implications for Development of Distributed Coordination. *Sustainability* 2019, 11, 1460; doi:10.3390/su11051460.
- Quiroz-Gil, N. L., & Valencia, C. A. (2012). Aplicación del proceso de kdd en el contexto de bibliomining: El caso Elogim. *Rev. Interam. Bibliot. Medellín (Colombia)* Vol. 35 n° 1 .
- Ramírez-Granados, M., Hernández, J., & Lyons, A. (2014). A Discrete-event Simulation Model for Supporting the First-tier Supplier Decision-Making in a UK's Automotive Industry. *Journal of applied research and technology*, 12(5), 860-870.
- Ranu, S., Hoang, M., & Singh, A. K. (2013). Mining discriminative subgraphs from global-state networks. In *Proceedings of the 19th ACM International Conference on Knowledge Discovery and Data Mining*.

- Rezaei, E., Rostami, Y., Ghafouri, M., Fashkhorani, A., & Ahmadi, M. (2016). The infrastructure attitude to strategic planning of information technology in organizations. *Marketing and Branding Research* 3(2016) 97-108.
- Rodríguez, G., Loja, F. A., Torres, H. H., & Torres, J. L. (2019). Análisis de sentimientos en redes sociales de eventos políticos, deportivos y climáticos usando técnicas computacionales. *Conti* 2019.
- Rodríguez-López, M. S.-V.-O. (2017). Análisis e inteligencia de negocios con evaluación de indicadores claves de desempeño. *I+D Tecnológico*, 13(2), 66-74.
- Rosenkranz, C., Räckers, M., Behrmann, W., & Holten, R. (2010). SUPPORTING FINANCIAL DATA WAREHOUSE DEVELOPMENT: A COMMUNICATION THEORY-BASED APPROACH. *ICIS 2010 PROCEEDINGS*.
- Sarasa, A. (2019). *Introducción a las bases de datos NSQL clave-valor usando Redis*. Barcelona: Editorial UOC.
- SAS Institute Inc. (2021). *sas.com*. [https://www.sas.com/es\\_mx/insights/big-data/hadoop.html](https://www.sas.com/es_mx/insights/big-data/hadoop.html)
- Schuld, M., Sinayskiy, I., & Petrucci, F. (2016). Prediction by linear regression on a quantum computer. *Phys. Rev. A* 94, 022342 (2016), 10.1103/PhysRevA.94.022342.
- Seymour Lipschutz. (McGraw-Hill, 1989). *Estructura de datos*. McGraw-Hill.
- Silva, T. H., Vaz-de-Melo, P., Carneiro-Viana, A., Marques-Almeida, J., Salles, J., & Loureiro, A. (2013). Traffic Condition Is More Than Colored Lines on a Map: Characterization of Waze Alerts. *Conference: Proc. of the International Conference on Social Informatics (SocInfo'13)*.
- Smaczny, T. (2001). Is an alignment between business and information technology the appropriate paradigm to manage IT in today's organisations? *Management Decision*, Vol. 39 Issue: 10, pp.797-802, <https://doi.org/10.1108/EUM00000000006521> .
- Somalo, N. (2017). *Marketing digital que funciona*. Andalucía: Almuzara.
- Spark. (2021). *spark.apache.org*. Guía de programación de RDD: <https://spark.apache.org/docs/latest/rdd-programming-guide.html>
- Spender, J. C. (2015). *Modern Management: Origins and Development*. Elsevier, Editors, DOI: 10.1016/B978-0-08-097086-8.73132-1.
- Sudaryo, Y., & Purnamasari, D. (2018). Information Technology (IT) Development Strategy of STIE INABA in the Perspective of IT Balanced Scorecard. *American Journal of Humanities and Social Sciences Research (AJHSSR)*, Volume-02, Issue-12, pp-72-80.
- Tehranipoor, M., Forte, D., Garrett, S. R., & Bhunia, S. (2017). *Security Opportunities in Nano Devices and Emerging Technologies*. CRC Press.
- Toro López, F. J. (2016). *Costos ABC y presupuestos*. Ecoe Ediciones.
- Trujillo, A. (2006). Modelo multidimensional. *Ingeniería Industrial*, 27(1).
- Uddina, N., & Hossain, F. (2015). Evolution of modern management through Taylorism: An adjustment of Scientific Management comprising behavioral science. *Procedia Computer Science*, doi: 10.1016/j.procs.2015.08.537.
- Ushveridze, A. (2017). Some thoughts on how business analytics can benefit from using principles of classical physics. *Business Dynamics in KPI Space* .

- Varajãoa, J., Cruz-Cunha, M., & Fraga, M. (2017). IT/IS Outsourcing in Large Companies – Motivations and Risks. *Procedia Computer Science* 121 (2017) 1047–1061.
- Verbert, K., Govaerts, S., Duval, E., Santo, J. L., Van-Assche, F., Parra, G., & Klerkx, J. (2013). Learning Dashboards: an Overview and Future Research Opportunities. *Future Research Opportunities*.
- Vieira-Braga, L. P., Ortiz-Valencia, L. I., & Ramirez-Carvajal, S. S. (2009). *Introducción a la Minería de Datos*. Editora E-papers.
- Vila Guerrero, S. (2019). Diseño e implementación de procesos ETL con el fin de mejorar la toma de decisiones en una compañía. *RiuNet*, <http://hdl.handle.net/10251/124633>.
- Wang, J. (2015). Resilience of Self-Organised and Top-Down Planned Cities—A Case Study on London and Beijing Street Networks. *PLoS ONE* 10(12): e0141736. <https://doi.org/10.1371/journal.pone.0141736>.
- Wu, T. T., Gennari, R., Huang, Y. M., Xie, H., & Cao, Y. (2016). *Emerging Technologies for Education: First International Symposium, SETE 2016, Held in Conjunction with ICWL 2016*. Springer International Publishing.
- Xu, L. D. (2011). Enterprise Systems: State-of-the-Art and Future Trends. *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, VOL. 7, NO. 4.
- Xu, Y., & Chen, M. (2016). Improving Just-in-Time Manufacturing Operations by Using Internet of Things Based Solutions. *Procedia CIRP*, Vol. 36, <https://doi.org/10.1016/j.procir.2016.10.030>.
- Yaoli, W., Song, G., & Yu, L. (2013). Exploration into urban street closeness centrality and its application methods A case study of Qingdao. *GEOGRAPHICAL RESEARCH*, 2013, 32(3): 452-464.
- Yuan, D., Proutiere, A., & Shi, G. (2019). Distributed Online Linear Regression. *arXiv:1902.04774v1 [cs.LG]* 13 Feb 2019.
- Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., . . . Stoica, I. (2016). Apache Spark: un motor unificado para el procesamiento de big data. *Comunicaciones de la ACM*.
- Zhang, W., Chien, J., Yong, J., & Kuang, R. (2017). Network-based machine learning and graph theory algorithms for precision oncology. doi: 10.1038/s41698-017-0029-7.
- Zhang, Y., Zhang, Y., Wang, S., & Lu, J. (2019). Fusion OLAP: Fusing the Pros of MOLAP and ROLAP Together for In-Memory OLAP. *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 9, pp. 1722-1735, 1 Sept. 2019, doi: 10.1109/TKDE.2018.2867522.

Otros títulos de la colección  
Ciencias, Ingenierías y Medio Ambiente



***Análisis de Funciones Especiales***

Carlos Fernando Méndez Martínez

***Topografía aplicada a las Ciencias Agrícolas***

Carlos Eloy Balmaseda Espinosa

***Bioestadística***

Froilán Segundo Méndez Vélez, Milton Bolívar Romo Toledo y Gabriela Alejandra Ortega Castro

***Desafiando a la Ciudad Letrada. Formas antagonistas de urbanismo en América Latina***

Antonio di Campli

***Análisis de ecuaciones y sus lugares geométricos***

Carlos Fernando Méndez Martínez

Otros títulos de la colección  
Salud y Bienestar



***Correlación entre la Medicina de Laboratorio y las Ciencias Básicas y Clínicas***

Julio César Sempértegui Vega, Sandra Patricia Ochoa Zamora, Poleth Estefania Sempértegui Alvarado y Mateo Esteban Zea Cabrera

***Patología Estructural Básica***

Yolanda Vanegas Cobeña, Nancy Vanegas Cobeña y Leonardo Morales Vanegas

***Esquizofrenia. El enigma continua***

Douglas Calvo de la Paz

***Enfermería: investigación y el cuidado directo***

Edison Gustavo Moyano Brito, Nube Johanna Pacurucu Ávila, Isabel Cristina Mesa Cano, Lizette Espinosa Martín y Zoila Katherine Salazar Torres

***Canino retenido: historia, diagnóstico y tratamiento actual***

Diego Palacios Vivar, Yonatan Torres Cruz y Vinicio Barzallo Sardi

***Programa de intervención en educación emocional dirigido a niños y niñas de educación general básica***

Galo Bravo Corral

***Epidemiología en salud bucal: caso Cuenca***

Ebingen Villavicencio Caparó, Napoleón Reinoso Vintimilla y Liliana Encalada Verdugo

***Enfermería en la salud de Ecuador y Cañar***

María Fernanda Peralta Cárdenas, María Erlinda Aguaiza Pichazaca, José Ivo Contreras Briceño y Luis Hermmann Rodríguez Rangel

***Salud pública y enfermería en el contexto ecuatoriano***

Erica Paola Rojas Verdugo, Zandra Maribel Regalado Vazquez, Johanna Rosalí Reyes Reinoso y Angela María Quintero de Contreras

***Manejo de hiponatremia***

Carlos Enrique Flores Montesinosca

***Disturbio hidroelectrolítico***

Andrés Bueno Castro

***Reumatología***

Álvaro González Ortega

***Endodoncia***

Fernanda Katherine Sacoto Figueroa  
María Emilia Guerrero Coello  
María Elizabeth Moscoso Abad  
Sara Ivanna Cedillo Orellana

***Tecnología Farmacéutica: Trabajos Prácticos***

Hermel Salinas Medina, Sebastián Peña Peña y Diego Andrade Campoverde

***Familias Saludables, Niños Saludables. Una guía para comprender la relación entre la parentalidad y la alimentación***

María Alejandra Aguirre Quezada, Xavier Rodrigo Yambay Bautista, Nancy Beatriz Cordero Zumba, Sandra Patricia Ochoa Zamora, María Daniela Encalada Torres, María Isabel Herrera Jaramillo y Diana Mariela Méndez Pedroza

Otros títulos de la colección  
Ciencias Sociales y Humanidades



***Violencia Intrafamiliar. Beneficios de un Proyecto Social***

Vanessa Quito Calle, Mónica Tamayo Piedra y Olga Neira Cárdenas

***Hitos de la Constitución ecuatoriana***

Colectivo de autores

***El Perfeccionamiento de los Contratos***

Fernando Moreno Morejón

***Tópicos Actuales de Derecho Tributario Ecuatoriano***

Diego Adrián Ormaza Ávila, Ana Fabiola Zamora Vázquez, Teodoro Javier Cárdenas Parra, Amanda Fabiola Palacios Palacios y Evelin Daniela Vaca Asitimbay

***La tentativa inidónea o delito imposible. Una visión integral***

Silvio José Castellanos Herrera

***Evolución socioeconómica de Latinoamérica: caso ecuatoriano***

Froilán Méndez Vélez y Patricio Montero Vélez

***Matemática financiera: una experiencia en el aula***

William Sarmiento Espinoza y Kléber Luna Altamirano

***Teoría del derecho. Una introducción***

Juan Antonio García Amado

***Responsabilidad jurídica médica***

Agustín Borja Pozo y Juan Antonio García Amado



*Tecnologías Computacionales Emergentes. Análisis de  
datos, Investigación e Innovación Tecnológica*  
se imprimió en la ciudad de Cuenca, Ecuador, en mayo de  
2023, en la Editorial Universitaria Católica (EDUNICA), con  
un tiraje de 40 ejemplares.





ISBN: 978-9942-27-192-1



ISBN: 978-9942-27-193-8

