UNIVERSITAT POLITÈCNICA DE VALÈNCIA

etsinf

Escola Tècnica Superior d'Enginyeria Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica

Universitat Politècnica de València

Minería de Texto Web

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Borja García Alonso

Tutores: Jorge Serrano Cobos

Ángeles Calduch Losa

2018-2019

Agradecimientos

Después de un periodo largo e intenso, mi etapa académica llega a su fin y es conveniente realizar un apartado para todas aquellas personas que me han ofrecido su mano en este camino, tanto personal como académicamente. Realizar este trabajo es como presentar a todo el mundo a lo que realmente me quiero dedicar en mi vida profesional, por esto mismo me gustaría agradecer y dedicar este proyecto a todas las personas que me han apoyado tanto en este proceso como en todas las etapas de mi vida.

En primer lugar, deseo expresar todo mi agradecimiento a mis tutores de Trabajo Final de Grado, Jorge Serrano Cobos y Ángeles Calduch Losa, por la dedicación y atención con la que han dirigido este trabajo, por las ideas y el respeto a mis sugerencias que han facilitado la creación de este proyecto. He tenido la suerte de encontraros como tutores, gracias.

Asimismo, agradezco a mis compañeros del departamento de Business Intelligence de Nunsys por todo el apoyo y los conocimientos que he aprendido y espero aprender de ellos. Gracias a ellos he conocido el mundo al que me quiero dedicar y, por suerte, he encontrado un grupo donde la amistad, las ganas de enseñar y aprender son las virtudes que se pueden ver cada día en la oficina.

Mi más sincero agradecimiento a Carlos Expósito, compañero y amigo de Nunsys, que sin él no podría haber realizado este proyecto. Todo la ayuda, horas y conocimientos que me ha regalado para este trabajo es un pequeño reflejo de su profesionalidad y compañerismo. Espero aprender mucho más de ti y que tus cualidades te brinden un buen camino, tanto profesional como personalmente.

Gracias a mis amigos, por todo el apoyo y amistad recibido. Todos los momentos difíciles de esta vida se hacen más fáciles si ellos te acompañan.

Gracias a Marta Robles Ruíz, amiga y compañera, por toda la fuerza que me ha dado durante todo este Trabajo Final de Grado. Has sido mi pilar en estos meses de trabajo y esfuerzo. Necesaria para superar cualquier obstáculo, tanto personal como académicamente. Espero verte crecer y apoyarte durante todo ese camino, con las mismas ganas que siempre entregas tú.

Por último, me gustaría agradecer a mi familia que siempre hayan estado ahí para mí, en lo bueno y en lo malo. La persona que soy hoy y todos los consejos que aplico en mi día a día son gracias a ellos. Estoy orgulloso de ser vuestro hijo, tío y hermano. Por el amor y ganas que me habéis dado siempre, por todo y más, este trabajo también es vuestro.

Gracias a todos.

Resumen

En este Trabajo Final de Grado se explica un proyecto de análisis de datos desde el principio. Para ello se analizan diferentes *keywords*, las cuales se basarán en 30 Denominaciones de Origen diferentes sobre vinos españoles, gracias a un proceso completo de *Business Intelligence* y algunas de sus ramificaciones, como el *Machine Learning*. Este proceso ayudará a extraer, explicar y presentar toda la información necesaria de la historia pasada, presente y futura de los vinos.

Palabras clave: *keywords*, vinos, *Business Intelligence, Machine Learning*, *Data Mining*, Denominaciones de Origen, análisis de datos

Abstract

In this Final Degree Project we will explain a data analisis project from the beginning. For this, we will analyze different keywords, which will be based on 30 differents Denominations of Origin of Spanish wines, using to a complete Business Intelligence process and some of its ramifications, such as Machine Learning. This process will help us extract, explain and present all the necessary information from the past, present and future history of the wines.

Keywords : keywords, wines, Business Intelligence, Machine Learning, Data Mining, Denominations of Origin, data analysis

Resum

En aquest Treball Final de Grau s'explica un projecte d'anàlisi de dades des de principi. Per això, s'analitzan diferents *keywords*, les quals es basaran en 30 Denominacions d'Origen diferents sobre vins espanyols, gràcies a un procés complet de *Business Intelligence* i algunes de les seues ramificacions, com el *Machine Learning*. Aquest procés ajudarà a extraure, explicar i presentar tota la informació necessària de la història passada, present i futura dels vins.

Paraules claus : *keywords*, vins, *Business Intelligence, Machine Learning, Data Mining*, Denominacions d'Origen, análisis de dades

Tabla de contenidos

1. Introducción

Este Trabajo Final de Grado (TFG) trata de analizar y proporcionar valor suficiente a un grupo de palabras claves o *keywords*, las cuales se extraen de Google.

Por otro lado, la realización de este trabajo también está orientada hacia la ayuda al proyecto "Diseño de Un Método y Desarrollo de Una Herramienta de *Online Information Intelligence* Orientada a la Recomendación Geolocalizada del Mercado de Vino (CSO2016-78775-R)", financiado por la Agencia Estatal de Investigación y del cual los tutores de este Trabajo Final de Grado forman parte.

Para analizar y obtener información importante de las *keywords*, se construye un sistema de *Business Intelligence*, en el cual el trabajo se centrará en su rama de minería de datos y *Machine Learning*.

Aunque el título del proyecto nombra la minería de textos, ya que es un trabajo en el que se intenta sustraer información de los datos obtenidos en una web, se proporciona mucha importancia a la parte de *Business Intelligence* (BI). Sin un sistema de BI, no se puede aplicar el *Data Mining* (DM)*, Machine Learning* (ML) o ramificaciones similares y, por tanto, no se pueden encontrar patrones en los datos que se analizarán en el presente trabajo.

En un principio, para crear un sistema de *Business Intelligence*, se hace hincapié en todos los puntos relacionados con la parte de análisis y diseño de los datos, desde su proceso ETL, de sus siglas *Extraction, Transformation* y *Load* en inglés, hasta la representación de los mismos de una forma visual. Posteriormente, el proyecto se adentra en la parte de la minería de datos y aprendizaje automático, la cual se basa en encontrar patrones en los volúmenes de datos que se habrán analizado con anterioridad.

Los patrones son un conjunto de sucesos u objetos organizados siguiendo una regla (Tresquatreicinc, 2017). Estos patrones se podrán encontrar con facilidad en la parte visual creada en la representación de los datos del sistema BI. Además, pueden ser de dos tipos:

- Patrones de repetición: Elementos presentados de forma periódica.
- Patrones de recurrencia: Elementos ordenados de forma irregular y hay que introducir la regla con la que suceden. Es decir, para predecir cuál va a ser el siguiente elemento se debe observar el comportamiento de los anteriores.

Por otra parte, después de encontrar los patrones en los datos, se aplica *Machine Learning*. En este punto del trabajo, con la información obtenida con anterioridad, se aplica la parte predictiva para proporcionarle más valor a los datos.

Los datos que se utilizan se centran en la tendencia del número de búsquedas sobre la Denominación de Origen (D.O.) de los vinos existentes en España. Estos datos son obtenidos desde la web de *Google Trends*, la cual nos proporciona una cifra sobre la tendencia de búsqueda de las *keywords* que se introducen en la herramienta (Google Trends, 2006).

Por último, el proyecto no solo se basa en explicar el propósito y resultados de realizar un trabajo de BI con predicción, también explica las herramientas utilizadas en todo el proceso, su importancia e implementación.

2. Entorno

El entorno en el cual se ha desarrollado este Trabajo Final de Grado ha sido heterogéneo. La idea principal ha sido académica pero, sin embargo, la estancia en el mundo laboral ha servido para conseguir perfeccionar la idea y conocer el esqueleto de un proceso de *Business Intelligence* y sus posteriores ramificaciones.

La parte de *Data Mining* y *Machine Learning* ha sido un proceso de formación personal tanto con el fin de adquirir los conocimientos suficientes para los objetivos de este trabajo, como para dirigirla a un mundo laboral en un futuro.

3. Motivación

Los conocimientos adquiridos en el Grado de Ingeniería Informática y la información que podía obtener a través de diferentes medios de información, generaban que cada vez diera más valor a los datos y su aplicación en la informática.

Como se puede observar, muchos gigantes tecnológicos proporcionan mucha importancia al tratamiento y análisis de los datos. Google, Facebook, Amazon o Apple son claros ejemplos de esto, ya que realizan inversiones extraordinarias para analizar los datos que obtienen a partir de sus usuarios.

La elección de este Trabajo Final de Grado ha sido tomada para la formación personal en esta bifurcación de la informática. Gracias a mi estancia en el mundo laboral dentro de este sector, he conseguido aumentar los conocimientos y observar la importancia e influencia que posee la creación de un sistema para el análisis de datos en todos los ámbitos que conocemos.

En definitiva, como se puede observar en el día a día, los datos son el nuevo diamante del siglo XXI y el objetivo que espero alcanzar es construir mis conocimientos de la informática alrededor de ellos. Por tanto, este trabajo es considerada una oportunidad para abrir camino en esta rama de la informática.

4. Objetivos

El objetivo principal en el que se centra este Trabajo Final de Grado es la construcción de un sistema de *Business Intelligence* para analizar datos y, posteriormente, encontrar patrones y predecir qué podría pasar en el futuro a partir de esta información. La información que se analiza, como se ha comentado ya con anterioridad, se basa en la tendencia de búsquedas sobre diferentes vinos con Denominación de Origen existentes en España.

Estos datos se tratan dentro de un sistema de *Business Intelligence*, ya que este proceso permite proporcionar el valor suficiente a los datos para adquirir información que ayude a analizar y encontrar los patrones que se intentan obtener en este proyecto.

Así pues, el trabajo no se centra simplemente en la obtención, análisis y predicción de los datos, también se basa en enfocar la complejidad y el desarrollo de un proyecto BI.

Como se explica en la introducción, la parte considerada como minería de datos no podría ser tratada correctamente sin antes la creación de un sistema BI. Se puede aplicar el mismo sentido con el *Machine Learning*, por consiguiente, se puede considerar el *Business Intelligence* como el concepto que engloba diferentes formas de tratamiento de los datos, y tanto el *Data Mining* como el *Machine Learning* que se aplican serán ramificaciones del mismo.

Respecto al título del TFG, se hace referencia a la minería de textos, rama que busca extraer información importante de webs, artículos u otros tipos de documentos. En este trabajo no se realiza ningún algoritmo para extraer información ya que la página que se utiliza para obtener datos sobre las *keywords*, *Google Trends*, es una herramienta de Google que proporciona información sobre la tendencia de búsquedas de las palabras claves que se necesitan (Google Trends, 2006).

En otras palabras, Google realiza el proceso de minería de textos para adquirir información relevante de las búsquedas que se realizan en su navegador y, gracias a la herramienta de *Google Trends*, se puede descargar toda la información necesaria para este trabajo.

En resumen, este TFG trata de analizar la información y proporcionar un valor más importante a los datos, después de ser extraída de un proceso de minería de textos realizado por Google.

Por último, los conocimientos adquiridos en el mundo laboral durante los últimos meses convierten a este proyecto como un modelo a seguir, ya que podría ser presentado como un servicio para un usuario final interesado en el análisis de los datos relacionados con los vinos.

5. Impacto Esperado

El impacto esperado con este Trabajo Final de Grado es la formación personal en el ámbito del análisis de datos y su relación con el sector informático.

Este proyecto pretende enseñar cómo se pueden adquirir datos independientes desde una web y, si se estructuran de la forma correcta y se realiza un análisis completo, cómo se puede encontrar información valiosa para cualquier persona o cliente que necesite realizar una investigación sobre un tema en cuestión.

Desde otro punto de vista más personal, con este trabajo se pretende aumentar los conocimientos en el mundo del análisis de datos y demostrar la capacidad para conducir un proyecto desde cero ya que, como se ha mencionado en el apartado de objetivos, podría ser un trabajo guiado hacia la venta a un cliente.

6. Contexto Tecnológico

6.1. Business Intelligence

Para comenzar este apartado, se debe tener en cuenta que *Business Intelligence* engloba todas las formas de análisis de datos inteligentes y, el *Data Mining* o el *Machine Learning* son dos ejemplos de ramificaciones y formas distintas de obtener información después de realizar un sistema de BI (Conexiónesan, 2018).

Una vez que se han aclarado estos conceptos, se pretende explicar la evolución histórica y el contexto actual del *Business Intelligence* y las dos formas de análisis de datos que se aplican en este Trabajo Final de Grado, el *Data Mining* y el *Machine Learning*. La minería de textos, como se ha explicado en apartados anteriores, es un proceso aplicado por Google del cual la realización de este trabajo se ha beneficiado para obtener datos que se analizan con más profundidad.

Los orígenes del *Business Intelligence* se sitúan en octubre de 1958, donde se cita por primera vez en el artículo *A Business Intelligence System*. El autor de este artículo define el *Business Intelligence* como "the ability to apprehend the interrelationships of presented facts in such a way as to guide action towards a desired goal" (Luhn, 1958).

Por tanto, Hans Peter Luhn, autor del artículo mencionado e investigador de IBM, reconocida multinacional de tecnología y consultoría, fue la primera persona en acuñar el término de BI (Breve Historia del Business Intelligence: Origen y Evolución, 2017) pero, aunque realizó el primer paso para abrir las puertas a un campo muy importante en la actualidad, no fue el único en hacer referencia a este ámbito.

Varios acontecimientos que marcarían el futuro del *Business Intelligence* se producirían en los años siguientes. Cuatro años después de publicarse el artículo de H.P.Luhn, el canadiense Keneth Iverson asentó las bases del primer lenguaje de programación multidimensional en su libro *A Programing Language* (APL) (Iverson, 1962). Este lenguaje originó en 1962 la tecnología OLAP, la cual se explica en apartados posteriores, moldeando así también lo que iba a ser el *Business Intelligence* en un futuro.

En la década de los 70, el científico informático Edgar Frank Codd, definió el modelo relacional y publicó las reglas para bases de datos relacionales a través de su artículo *A Relational Model of Data for Large Shared Data Banks.*

Para el descontento de E.F.Codd, IBM no explotó sus conceptos hasta que otras compañías empezaron a poner en práctica las ideas que publicó en su artículo.

A consecuencia de esto, Lawrence Joseph Ellison, a partir de las ideas que adquirió del artículo *A Relational Model of Data for Large Shared Data Banks,* desarrolló el *Relational Software System* o, como se conoce en la actualidad, *Oracle Corporation*, creando así uno de los sistemas gestores de bases de datos relacionales más importantes en todo el mundo.

Posteriormente, durante los años 80, surgió el lenguaje de consulta estructurada o conocido más popularmente por sus siglas en inglés como SQL (*Structured Query Language*), el cual se convertiría en el lenguaje de programación estándar para los

gestores de bases de datos relacionales. La principal característica de SQL es la recuperación y modificación, de forma sencilla, de información dentro de una base de datos (Prado, 2017).

En la misma década, se realiza un gran avance dentro del *Business Intelligence* gracias al *reporting* y el *Data Warehouse*, conceptos que se tratan en puntos posteriores de este mismo trabajo. Hasta entonces, aunque ya existían potentes sistemas de bases de datos, no había ninguna herramienta que facilitara la explotación de la información que contenían. Finalmente, en 1985 Microsoft lanza Excel 1.0, herramienta que se convierte en una de las más utilizadas en el mercado.

Por último, el *Business Intelligence* alcanza su popularidad en la década de los noventa a manos de Howard Dresner, después de que en 1989 definiera la inteligencia de negocio o BI como *"concepts and methods to improve business decision making by using fact-based support system"* (Cebotarean, 2011). Gracias a toda la repercusión que tuvo el BI en esta década, se generaron durante los años 90 y se consolidaron en la década del 2000 múltiples aplicaciones y herramientas que ofrecían acceso a las bases de datos y la información estructurada que ofrecían las empresas.

En este mismo milenio, ya no solo se realizan análisis sobre datos estructurados, sino que también se empieza a tener en cuenta otros tipos de información no estructurada.

6.2. Minería de datos (DM) y Machine Learning (ML)

Una vez descrita la historia del *Business Intelligence*, se procede a explicar el *Data Mining* y el *Machine Learning*.

Como se ha comentado en apartados anteriores, el *Data Mining* o minería de datos y el *Machine Learning* o aprendizaje automático forman parte del BI, ya que este engloba todos los conceptos que se refieren al análisis de datos.

También se puede decir que la minería de datos y el ML son muy similares, simplemente se diferencian en sus objetivos. Para aclarar esto, primero se van a definir los dos conceptos y después se explica de forma más concreta el por qué la minería de datos y el *Machine Learning* se asemejan o incluso forman parte uno del otro (Chambi, 2016).

La minería de datos o la exploración de datos (DM o KDD) es un campo de la estadística y la ciencia de la computación que intenta descubrir patrones dentro de un conjunto grande de datos. Con el *Data Mining*, después de realizar la extracción y transformación de los datos, se pueden encontrar patrones importantes que pueden proporcionar información con mucho valor. Un ejemplo de minería de datos podría ser qué palabras son más buscadas en los meses de verano.

Con esta definición del *Data Mining*, se puede deducir que la minería de textos es una forma específica de DM que se relaciona con el texto (Chang, 2018). Por ejemplo, en las búsquedas que se realizan durante los meses de agosto, en cuántas ocasiones se introduce la palabra coche en una frase. Otro ejemplo podría ser la de suponer de qué trata un artículo a partir de las palabras que más se nombran del mismo.

Por otra parte, el *Machine Learning* se centra más en un objetivo predictivo. Es decir, después de disponer los datos analizados y una vez se han encontrado patrones en la muestra, se puede aplicar el aprendizaje automático para predecir qué puede pasar en

un futuro. En resumen, un algoritmo de ML utilizará los patrones mostrados con anterioridad y seguirá abasteciéndose de los nuevos datos que se introduzcan para decir qué puede suceder en un futuro.

Por tanto, tanto el *Machine Learning* como el *Data Mining* encuentran patrones, pero el concepto de ML también engloba la predicción de datos a partir de la información obtenida (Mayorga Muñoz, 2019). En los dos conceptos se pueden utilizar técnicas muy parecidas, pero, como se ha dicho, los objetivos son diferentes.

6.3. Herramientas para el análisis de datos

6.3.1. Sistemas de gestión de base de datos

En todo proyecto dirigido hacia el tratamiento y el análisis del dato, se necesita una base de datos (BBDD) para almacenar toda la información que se utiliza para analizar y extraer conocimiento que posteriormente se representan en otras herramientas.

Para seleccionar una base de datos se deben tener en cuenta criterios importantes como la cantidad a almacenar, la velocidad de sus procedimientos o la seguridad con la que puede ser tratada la información. Los motores de bases de datos más utilizados son los siguientes:

- *Oracle*: Base de datos más utilizada en el mundo. Puede ser usada en casi cualquier sistema operativo y es considerada una de las BBDD más completas y robustas del sector. Algunas de las ventajas más importantes que pueden ser interesante de este motor son la gran cantidad de herramientas que hay para su monitorización y administración, y la abundancia de perfiles en esta tecnología. *Oracle* también destaca por:
    - Soporte de transacciones
    - Estabilidad
    - Escalabilidad
    - Multiplataforma
- *SQL Server*: Gestor de base de datos desarrollado por Microsoft y que se basa en el lenguaje estructurado de SQL. Aunque únicamente se pueda utilizar en sistemas *Windows*, no ha sido un impedimento para ser uno de los motores de bases de datos que compiten directamente contra *Oracle*. Algunas de sus características más importantes son:
    - Variedad de aplicaciones de procesamiento de transacciones
    - Utiliza como lenguaje de programación Transact-SQL (T-SQL)
    - Visualización de datos e informes en plataformas móviles
    - Escalabilidad y seguridad.

Por último, destaca su integración con *Microsoft Azure*, ya que esta mejora le ha proporcionado un gran salto en flexibilidad y rendimiento, permitiendo la compatibilidad con la nube.

- MongoDB: Probablemente es la base de datos NoSQL más popular en la actualidad. Las bases de datos NoSQL se identifican por no utilizar SQL como lenguaje principal para las consultas. MongoDB utiliza estructuras BSON, formato de intercambio que utiliza para transferir y almacenar información.

  Esta BBDD tiene la posibilidad de trabajar con datos estructurados y no estructurados. Otras características de MongoDB son:

  - Indexación y replicación
  - Almacenamiento en ficheros
  - Escalabilidad horizontal
  - *Open Source*

(Marín, 2019), (21 base de datos más utilizadas por los desarrolladores, 2019).

6.3.2. Extracción, Transformación y Carga (ETL)

Una herramienta ETL permite realizar las funciones de extracción necesarias de las fuentes de datos, la transformación y limpieza de los mismos y la carga en un *Data Warehouse,* concepto que se menciona en el punto 6.3.3*. Data Warehouse*.

Antes de elegir una herramienta ETL, se debe considerar qué tipo de base de datos se va a utilizar para crear el *Data Warehouse*. Los tipos de bases de datos que se pueden distinguir son las siguientes:

- ROLAP: Sistema OLAP gestionado por un motor de base de datos relacional.
- MOLAP: Sistema OLAP gestionado por un motor de base de datos multidimensional.
- HOLAP: Sistema OLAP gestionado por un motor de base de datos híbrido.

OLAP (*On-Line Analytical Processing*), es una solución utilizada en *Business Intelligence* que intenta agilizar el tratamiento de grandes cantidades de datos. La característica más destacable de OLAP, es la rapidez que dispone para extraer información de sentencias SQL de tipo SELECT, función que nos permite visualizar los datos dentro de una tabla de BBDD (¿Qué es OLAP?, 2011).

Los principales ejemplos de herramientas ETL son los siguientes:

- Informática *PowerCenter*: Posiblemente es la herramienta ETL más importante del mercado. Incluye la solución *PowerCenter*, una de las más populares del sector.

- *Oracle Data Integrator*: A diferencia de otras herramientas, envía los datos al destino de base de datos y utiliza su propio motor para realizar las

transformaciones. Con este método, evita utilizar los recursos de *hardware* como realizan la mayoría de las herramientas.

- *Microsoft SQL Server Integration Services (*SSIS*)*: Herramienta ETL muy intuitiva y fácil de utilizar. Como en la mayoría de las herramientas creadas por Microsoft, solo se puede utilizar en sistemas *Windows*.

- *Kettle*: Herramienta ETL que utiliza la suite de Pentaho. Algunas de las características más importante de Kettle es su interfaz gráfica *Spoon*, además de ser una herramienta de código abierto.

(Principales categorías de herramientas ETL, 2018), (Carisio, 2018).

6.3.3. *Data Warehouse*

Un almacén de datos (*Data Warehouse*) es un gran almacén de datos e información que recoge todos aquellos datos que son realmente necesarios para la realización de análisis, informes y cuadros de mando. Esta herramienta se ha hecho parte fundamental en la toma de decisiones de las empresas (Concepto de Data Warehouse, 2018).

La información que se guarda en un *Data Warehouse* (DW) proviene después de realizar el proceso de *Data Warehousing*. Este proceso realiza la extracción de los datos de aplicaciones externas e internas, después de depurarlos y estructurarlos, en la forma adecuada para el análisis.

Antes de realizar el proceso y almacenar toda la información, es importante decidir la estructura que tendrá el almacén de datos. En este análisis se deben tomar decisiones respecto a las tablas que componen el *Data Warehouse* y de los atributos de la misma. Una buena organización de la estructura facilitará el procedimiento de análisis de los datos posterior.

En definitiva, aunque la finalidad principal de un almacén de datos es guardar la información que se ha transformado para su empleo empresarial, gracias a esta herramienta se facilitan las tomas de decisiones de una compañía y la calidad de las mismas.

6.3.4. Reporting y cuadros de mando

Las herramientas de *reporting* sirven para realizar informes estáticos donde el usuario final puede analizar la información de los datos que se han extraído del *Data Warehouse*, después de realizar la limpieza y tratamiento correspondiente en el proceso ETL.

Los informes se pueden entregar en diferentes formatos como PDF, XML, CSV, etc. El formato dependerá de la suite que utilicemos y las características que posea.

Por otra parte, un cuadro de mando (*dashboard*), es una herramienta que permite monitorizar y aclarar los objetivos de una empresa en sus diferentes áreas de trabajo. Cada año las empresas aumentan su inversión en este tipo de *software*, dada la importancia que han encontrado en medir y analizar cada acción que se toma.

La era de la información y la digitalización está cambiando el rumbo de las empresas, y las compañías que poseen herramientas de cuadros de mando y *reporting* adquieren ventajas competitivas respecto a la competencia que no han invertido en estos proyectos. Estos tipos de *software* permiten analizar los datos almacenados disponibles en las bases de datos de la empresa de una manera gráfica e intuitiva.

Algunas de las herramientas de *reporting* y *dashboard* más importantes son las siguientes:

- IBM Cognos: Suite que permite trabajar con estructuras de datos relacionales y estructuras dimensionales. IBM Cognos dispone de herramientas necesarias para mejorar tanto el rendimiento financiero como la gestión de estrategias. La suite de IBM es pionera en el análisis predictivo y se ha convertido en una de las suites más utilizadas en el mundo de la analítica y estrategia empresarial (IBM Cognos, 2014).

Ilustración 1: Logo de IBM Cognos Analytics (IBM Cognos, 2014)

- *Tableau*: Una de las mejores herramientas para la inteligencia de negocio y la visualización de los datos. En 2018, Gartner Inc. la consideró por quinto año consecutivo la herramienta más óptima para analítica e inteligencia artificial (Las 10 herramientas de Business Intelligence que deberías conocer, 2018).

Ilustración 2: Logo de Tableau (Logotipo de Tableau, 2017)

- *QlikView*: Herramienta de *Business Intelligence* que permite a las organizaciones medir, monitorear y realizar un seguimiento de procesos clave dentro de la empresa. *QlikView* propone interfaces altamente interactivas y de fácil uso, ofreciéndole al usuario un acceso instantáneo a información de alto nivel (QlikView, 1993).

Ilustración 3: Logo de QlikView (Logodix)

- *Power BI*: Esta herramienta creada por Microsoft, se ha convertido en una de las soluciones más utilizadas en las empresas. Permite realizar análisis empresariales gracias a la visualización de datos y la facilidad de compartir la información con toda la organización. Como muchas de sus competidoras, permite conectarse a cientos de orígenes de datos y visualizarlos con paneles e informes dinámicos (Power BI, 2014).

Ilustración 4: Logo de Microsoft Power BI (Power BI, 2014)

6.3.5. Herramientas para la predicción de datos

En este punto se explican los dos lenguajes de programación más utilizados en el análisis de datos mediante técnicas de *Data Mining* o *Machine Learning*, *Python* y R. Estos lenguajes, como se puede observar en el mundo de la analítica de datos, son los más buscados por las empresas cuando intentan extraer información de los datos de una forma predictiva o encontrar patrones para su toma de decisiones. La descripción de los dos lenguajes es la siguiente:

- *Python*: Lenguaje de programación interpretado y multiparadigma. Este lenguaje es compatible con la programación orientada a objetos, programación imperativa y la programación funcional, aunque esta última la soporta en menor medida.

  Este lenguaje posee una gran comunidad que mejora en gran medida todas sus características, mediante la creación de nuevos paquetes o librerías libres, funciones, etc.

  Hay dos motivos principales que han propiciado el creciente uso de *Python* en el campo del análisis de datos. Estos motivos son los siguientes:

  1. Gran cantidad de librerías creadas para la finalidad del análisis de datos. Ejemplos de estas librerías pueden ser:

     a. Pandas: Implementa funciones para realizar cálculos estadísticos y matemáticos.
     b. MatPloit: Permite la visualización y la representación de gráficos de los datos analizados.
     c. Mlpy: Dispone de algoritmos utilizados para el aprendizaje automático.

  2. Su integración con aplicaciones con sistemas de gestión de bases de datos como SQL, *Oracle*, MongoDB o Pentaho, entre otras.

  Estos motivos se suman a que *Python* dispone una fácil compresión y una rápida curva de aprendizaje, lo que ayuda a convertirse en un lenguaje de gran calidad para el análisis de datos.

Ilustración 5: Logo de Python (Correa, 2012)

- R: Lenguaje de programación de código abierto con un enfoque al análisis estadístico. R nació a partir del *software* libre S, su antecesor, con el objetivo de mejorar las visualizaciones y el análisis de datos que disponía este último lenguaje.

  R es un es un lenguaje multiparadigma, multiplataforma y orientado a objetos, además de disponer también una comunidad de desarrolladores que también mejoran y enriquecen el lenguaje, al igual que *Python*.

  Al ser un lenguaje orientado más al análisis estadístico, dispone de características que ayudan al manejo de elementos de este ámbito, como pueden ser matrices y vectores, lo que permite ser un lenguaje muy preciso y exacto para el análisis de datos.

  Por otra parte, también dispone de muchos paquetes que ayudan a que disponga capacidades avanzadas para la visualización de los datos y los resultados de análisis. Respecto a la parte del aprendizaje automático, también tiene implementados una gran cantidad de algoritmos como consecuencia a sus orígenes en el ámbito académico.

  Al contrario de *Python*, R dispone de una curva de aprendizaje más lenta y complicada, ya que es un lenguaje que fue creado exclusivamente para el análisis estadístico de los datos conllevando a que sea más apropiado para profesionales en este mismo sector (Rochina, 2016).

Ilustración 6: Logo de R (R logo, 2016)

6.4. Tecnología utilizada

Este apartado se basa en explicar las herramientas utilizadas en el presente Trabajo Final de Grado. Como se ha comentado en puntos anteriores, para un proyecto de análisis son fundamentales cuatro herramientas para las diferentes fases del trabajo: motor de base de datos, herramienta para realizar un proceso ETL, *Data Warehouse* y *software* para *reporting* o cuadros de mando. Por otra parte, también se menciona el lenguaje de programación utilizado para las predicciones de datos.

En primer lugar, como gestor de base de datos se utiliza *SQL Server*. Como se ha explicado en el apartado de herramientas, *SQL Server* está desarrollado por *Microsoft* y utiliza el lenguaje SQL como lenguaje estructurado. Para utilizar este lenguaje y manipular la base de datos se utiliza *SQL Server Management Studio* (SSMS), aplicación lanzada por Microsoft en 2005 para facilitar el tratamiento de todos los componentes de *Microsoft SQL Server*.

Con el gestor de base de datos y SSMS, se crea el servidor y las bases de datos que se necesitan para este proyecto. Una de estas bases de datos será lo que llamamos *Data Warehouse,* que contendrá todos los datos estructurados y limpios para su posterior análisis.

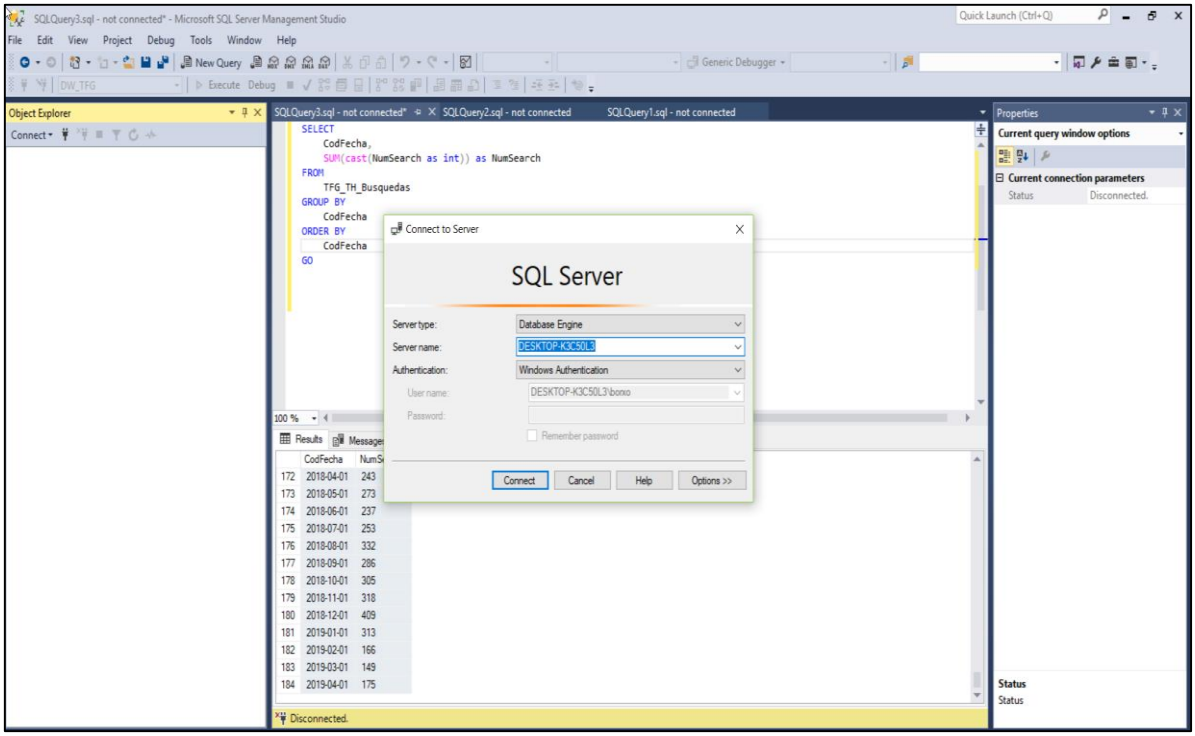


Ilustración 7: Introducción de credencias en SQL Server Management Studio (Elaboración propia, 2019)

Una vez explicado el gestor de base de datos y su herramienta para emplearlo, el siguiente paso es nombrar el *software* utilizado para el proceso de extracción, transformación y carga de los datos. A partir de este punto, los datos están preparados para ser analizados en el *Data Warehouse*.

Para este proceso, en el proyecto se emplea *Microsoft SQL Server Integration Services* (SSIS). Con este componente, como se detalla más adelante, se extraen los datos obtenidos desde la web *Google Trends*, se realiza la transformación respectiva y se cargan en el *Data Warehouse* que se utiliza para analizar.

Para llevar a cabo todas estas funciones, antes de todo se establecen las conexiones correspondientes con el servidor y las bases de datos desde *Microsoft SQL Server Integration Services*. Este es uno de los puntos que se explican en el desarrollo del proyecto.

Se debe hacer énfasis que *Microsoft SQL Server Integration Services* es un componente. Esto significa que la herramienta instalada y que permite la implementación del mismo es *SQL Server Data Tools* (SSTD) ya que, con este *software* y su componente de ayuda gráfica, *Visual Studio*, es posible la creación de proyectos SQL y paquetes de SSIS, los cuales se mencionarán en apartados posteriores.

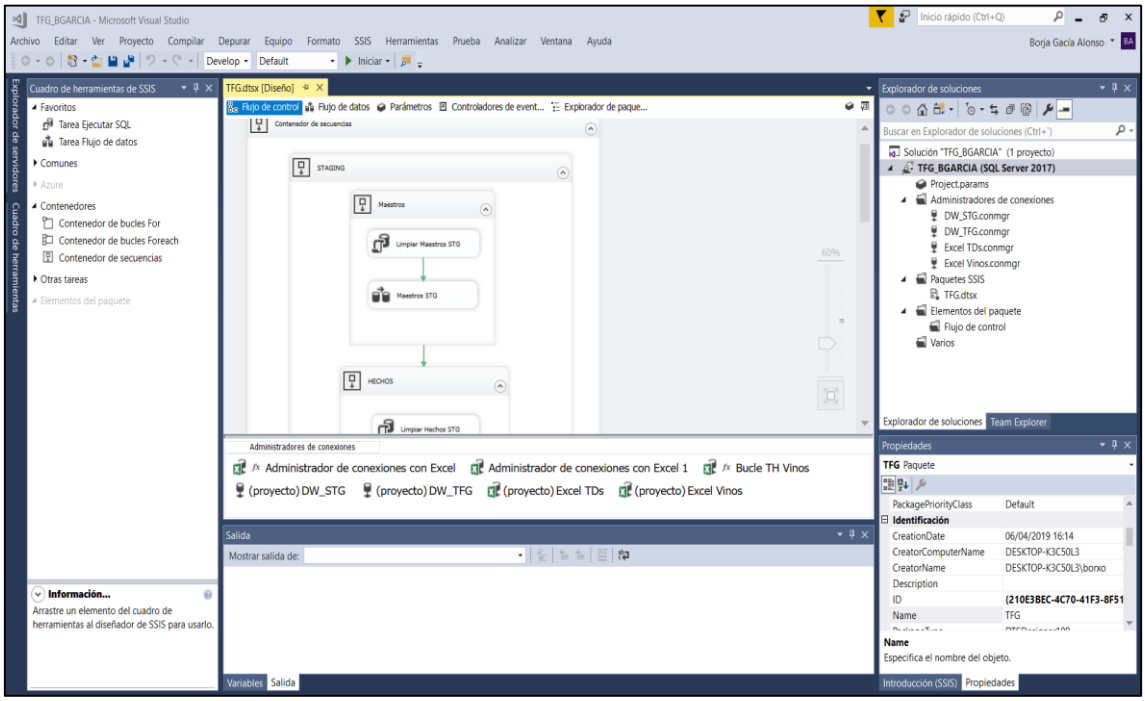


Ilustración 8: Implementación de un paquete de Microsoft SQL Server Integration Services (Elaboración propia, 2019)

Por otra parte, *Power BI* es el *software* en este Trabajo Final de Grado para representar la información de forma interactiva. Esta herramienta de Microsoft permite visualizar los datos con gráficos intuitivos, los cuales pueden ayudar a detectar patrones y si es necesario, en la toma de decisiones.

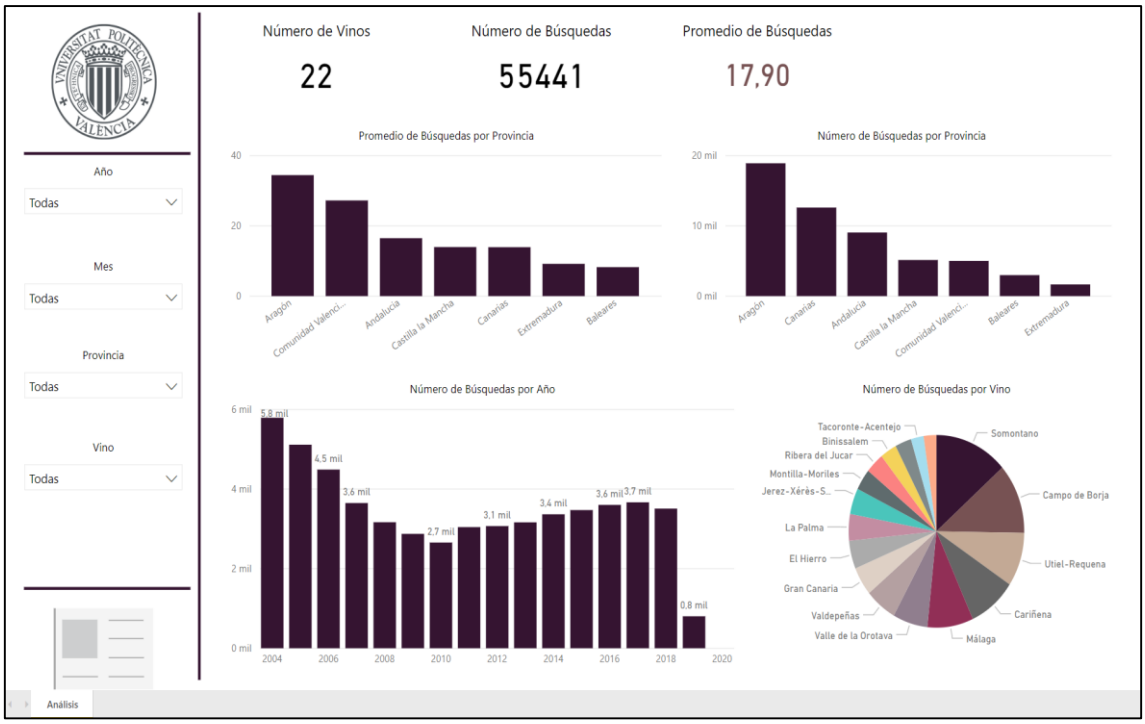


Ilustración 9: Cuadro de mando en Power BI (Elaboración propia, 2019)

Finalmente, es importante destacar la herramienta y el lenguaje utilizado para la parte de predicción que dispone este proyecto.

En el punto 6.3.5 “Herramientas para la predicción de datos”, se expone que hay dos lenguajes de programación que destacan para realizar *Data Mining* o *Machine Learning*, estos lenguajes son R y *Python*.

En este proyecto se utiliza *Python*, un lenguaje más centrado para programadores, en lugar de utilizar una programación guiada hacia estadísticos como puede ser R. También, en nuestra opinión, *Python* es un lenguaje que ha tenido una gran mejora en los últimos años, con su gran cantidad de librerías para el análisis de datos.

Para utilizar *Python* es necesario descargarse un *software* llamado Anaconda. Esta herramienta permite utilizar tanto *Python* como R para la ciencia de datos y el aprendizaje automático, *Machine Learning*.

Anaconda proporciona la ventaja para usar *Python* de forma intuitiva, además de descargar todos los paquetes necesarios para el análisis que se necesiten de una forma sencilla, con el sistema de gestión de paquetes que dispone, llamado Conda, se puede instalar, correr y actualizar *software* para el análisis de datos y *Machine Learning,* como *TensorFlow,* con solo apretar un botón.

Ilustración 10: Logo de Anaconda (Anaconda, 2012)

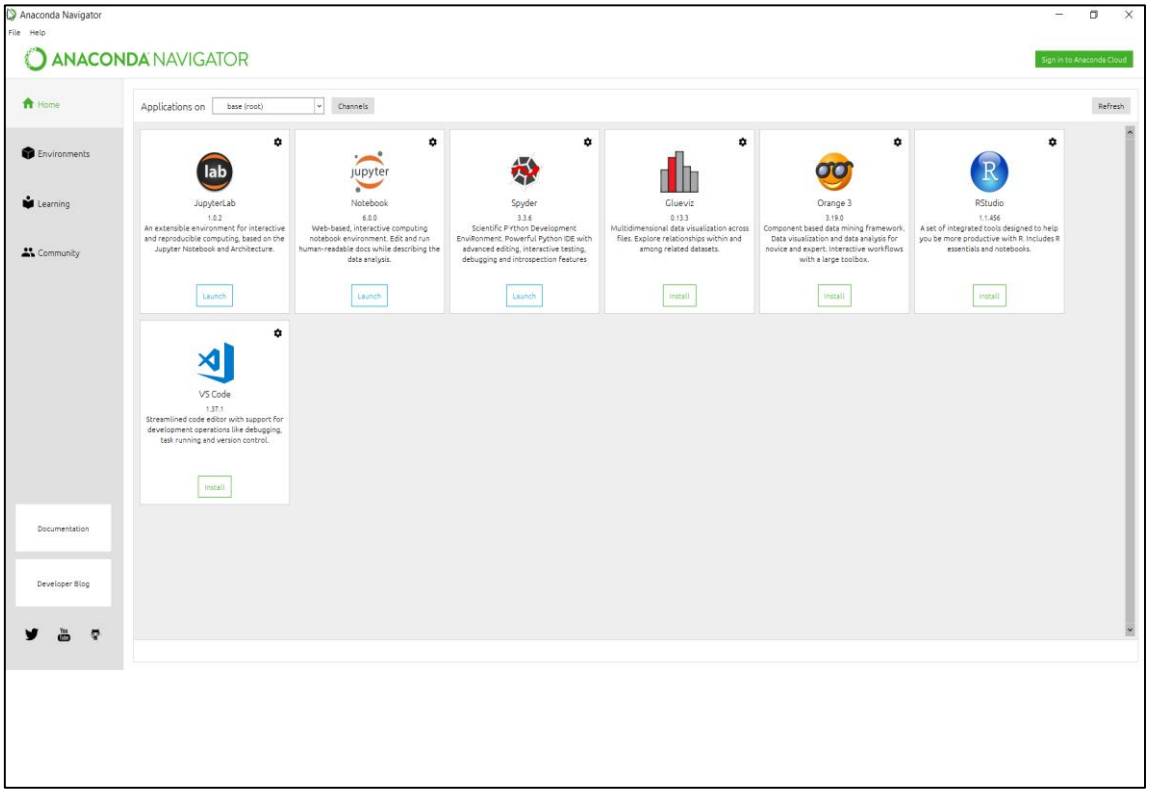


Ilustración 11: Navegador de Anaconda (Elaboración propia, 2019)

6.5. Solución propuesta

En esta sección se expone en qué consiste este Trabajo Final de Grado. Además, se explica brevemente las fases de desarrollo que tiene el proyecto.

Con este trabajo se obtienen patrones sobre la búsqueda de las Denominaciones de Origen analizadas. Estos patrones se pueden asociar al momento actual que pasan los vinos, con grandes descensos en las ventas si se compara con décadas anteriores.

El número de búsquedas que se estudian son obtenidas desde la web de *Google Trends*, donde se analizan las *keywords* necesarias para el trabajo. Esta web ofrece la tendencia de búsquedas en un valor entre 0 y 100, donde 0 la tendencia mínima de búsqueda y 100 es el máximo.

La información se obtiene desde el 1 de enero del año 2004, hasta el 1 de abril del 2019, donde se ha elegido este mes cerrado para disponer de datos de meses completos.

Las *keywords* que se eligen son diferentes Denominaciones de Orígen que existen en España, ya que el trabajo está dirigido a estos tipos de vinos. En resumen, el nivel de detalle que va a disponer este Trabajo Final de Carrera será las diferentes Denominación de Origen que hay por Comunidades Autónomas (C.A) españolas. El total de D.O. que se han escogido para la realización del trabajo son 30.

Las Denominaciones de Origen elegidas son las siguientes:

1. Málaga	11. Valle de la Orotava	21. Utiel-Requena
2. Montilla-Moriles	12. La Palma	22. Ribera del Guadiana
3. Jerez-Xérès-Sherry	13. Gran Canaria	23. Tierra del Vino de Zamora
4. Campo de Borja	14. El Hierro	24. Cigales
5. Calatayud	15. Ribera del Jucar	25. Ribera del Duerno
6. Cariñena	16. Valdepeñas	26. Rueda
7. Somontano	17. Almansa	27. Montsant
8. Binissalem	18. Jumilla	28. Cataluña
9. Pla i Llevant	19. La Mancha	29. Penedés
10. Tacoronte-Acentejo	20. Uclés	30. Pla de Bages

Tabla 1: Denominaciones de Origen seleccionadas para el análisis

Después de la obtención de estas palabras claves, se realizan todas las fases que contiene un proyecto de *Business Intelligence* o de análisis. Un resumen de las fases que se explican durante el desarrollo de este proyecto son las siguientes:

1. Instalación de las herramientas
2. Preparación del entorno
3. Orígenes y descarga de los datos
4. Proceso ETL (*Extract, Transform and Load*)
    a. Extracción: Se extraen los datos de su origen y se cargan en una base de datos.
    b. Transformación: Se estructuran y se limpian los datos de la forma que se necesiten para su posterior análisis.
    c. Carga: Se cargan los datos ya transformados en otra de las bases de datos, la cual dispondrá de los datos que se utilizarán para su análisis. Esta base de datos es a la que llamamos *Data Warehouse*.
5. Carga y visualización de los datos en *Power BI*
6. Predicciones con *Python*

También es necesario hacer hincapié en las validaciones de los datos. Cualquier proyecto de *Business Intelligence* o de análisis de datos debe tener una validación exhaustiva.

Muchas empresas realizan su validación en alguna de las fases de arriba mencionadas, como puede ser en la parte de carga del proceso ETL o en la parte de visualización de los datos. En la realización de este proyecto no se hace la validación en un punto en concreto, sino que están presente en todos los pasos en los que sea necesario. De esta forma no se pierde ninguna información que pueda ayudar para el análisis y tampoco se dispondrá de datos anómalos al final del trabajo. Por tanto, si las validaciones son hechas desde un principio, se evita generar inconvenientes en las capas superiores del proyecto.

En resumen, en este Trabajo Final de Grado se analiza la situación pasada y presente de las búsquedas de vinos con un sistema completo de análisis de datos, desde su obtención hasta su visualización de forma intuitiva. Además, se realiza una parte de predicción que completará el proyecto respondiendo a la pregunta: "¿qué situación van a tener las búsquedas sobre los vinos en los próximos meses?". Todos los datos analizados, las *keywords*, han sido obtenidas de *Google Trends*, herramienta que permite la extracción de información sobre la tendencia de búsquedas en Google.

7. Diseño detallado

Cuando se habla de diseño se hace referencia a la estructura de la base de datos, sus objetos y la organización de todos los archivos y herramientas necesarios para su realización.

Con este apartado se pretende iniciar la comprensión de todo lo que se va a tratar en los apartados posteriores, ya que en los siguientes puntos se explica la implantación y el desarrollo del trabajo realizado.

7.1. Instancias y bases de datos

Una instancia de *SQL Server* es una instalación del motor de base de datos de *SQL Server*, que se resume en un servicio de *Windows* que ejecuta un proceso sqlservr.exe. Este proceso contiene una configuración determinada y sus propias bases de datos, tanto las del sistema como las del usuario.

El presente proyecto dispondrá de una instancia de bases de datos y dos bases de datos. La instancia se crea cuando se realiza la instalación del gestor de base de datos, *SQL Server*, y tendrá como nombre DESKTOP-K3C50L3. Las dos bases de datos se

crean una vez que se ha conectado a la instancia, las cuales se llamarán DW_STG (*Staging*) y DW_TFG.

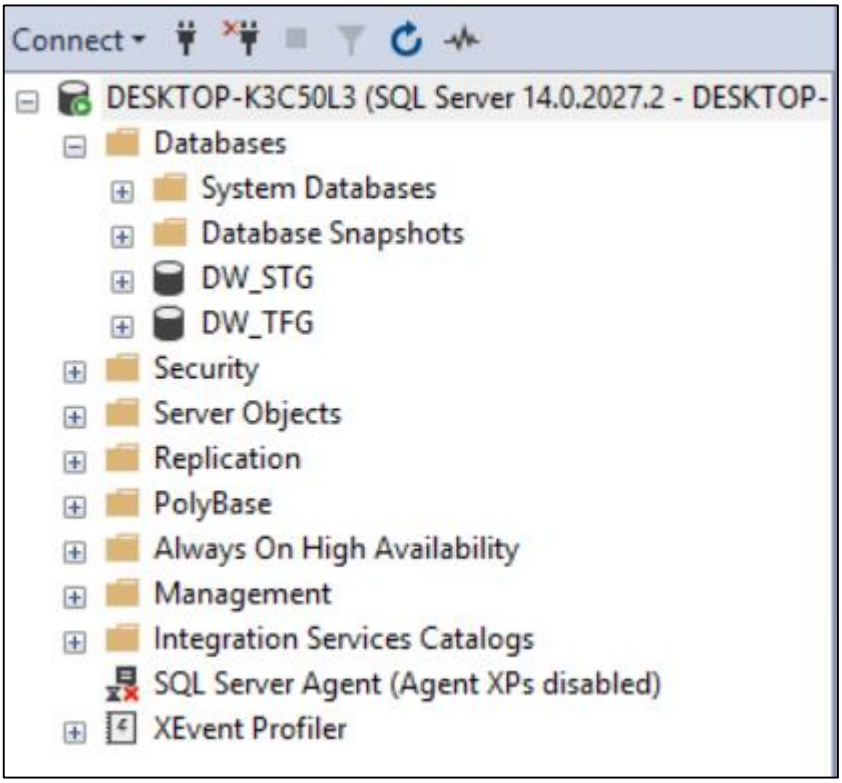


Ilustración 12: Estructura de la instancia DESKTOP-K3C50L3 en SQL Server Management Studio (Elaboración propia, 2019)

Las dos bases de datos poseen dos funciones diferentes:

1. DW_STG: Almacena todos los datos obtenidos de *Google Trends* sin ninguna transformación en sus datos.

2. DW_TFG: Esta base de datos actúa como *Data Warehouse*, parte fundamental en un proyecto de análisis y explicada en el apartado 7: “Herramienta para el análisis de datos”. En ella estarán todos los datos transformados y limpios para su posterior análisis.

7.2. Objetos de base de datos

Estas dos bases de datos disponen de dos objetos diferentes: tablas y vistas. Las tablas guardan todos los datos que se han cargado después de la extracción. Estos objetos ordenan la información en filas y columnas. Por ejemplo, una tabla que contiene los datos de los diferentes departamentos de una empresa, contiene una fila por cada departamento. Cada una de estas filas están organizadas con diferentes columnas con sus respectivos detalles, como puede ser el director del departamento, horas facturadas, etc.

Las tablas pueden ser de dos tipos: Tablas de Hechos (TH) y Tablas de Dimensiones (TD):

- Tablas de Hechos (TH): Las Tablas de Hechos contienen los indicadores de negocio, como pueden ser las ventas, búsquedas, envíos, reclamaciones, compras, etc. Es decir, son todas las medidas numéricas que se pueden incluir en un sistema de *Business Intelligence.*

- Tabla de Dimensiones (TD): Las Tablas de Dimensiones contienen los detalles de la Tabla de Hechos. Por ejemplo, una Tabla de Dimensiones de clientes almacena los diferentes aspectos de los mismos: nombre, dirección postal, número de teléfono, etc. Además de esta información, contiene también una clave por cada cliente, con la cual se puede realizar la unión con la Tabla de Hechos para obtener la información correspondiente.

Una de las características que diferencia una Tabla de Hechos con una Tabla de Dimensiones es su nivel de detalle. Estas tablas guardan la máxima información posible en forma de claves, las cuales ayudarán a enlazar con las Tablas de Dimensiones para obtener información más detallada.

Por esto mismo, las Tablas de Hechos incluyen las claves subrogadas de las dimensiones que sean necesarias para un proyecto y sus respectivas medidas. Si una TH no contiene las claves de una TD, no podrán relacionarse entre ellas. En el caso de crear una unión forzada entre las dos tablas, se puede correr el riesgo de que la información obtenida no sea fiable.

Respecto a las Tablas de Dimensiones, contienen un número menor de filas que las Tablas de Hechos, ya que su información no puede duplicar dentro de la misma tabla. Por ejemplo, un código de cliente puede aparecer varias veces en una TH con referencia a las compras, ya que se puede realizar más de una compra. Sin embargo, ese mismo cliente solo puede presentarse una vez en la TD, ya que su nombre y su dirección no puede cambiar.

Por otra parte, las vistas son consultas que se visualizan como una tabla a partir de una o un conjunto de las mismas. La única diferencia que existe entre una tabla y una vista es que, a diferencia de la tabla, la vista no almacena los datos, solo contiene la estructura.

La estructura de una vista no tiene que ser completamente idéntica a la tabla, ya que pueden cargarse todos los campos de sus tablas orígenes o solo parte de ellos.

Se dispone de diferentes tablas y vistas en cada una de las diferentes bases de datos. Los nombres que disponen estos objetos tienen relación con su origen, destino o uso. La nomenclatura que se usa es diferente para cada BBDD:

1. DW_STG:
    a. Tablas: Uso_Origen_TipoDeTabla_DatosQueContiene
2. DW_TFG:
    a. Tablas: Uso_TipoDeTabla_DatosQueContiene

b. Vistas:
UsoDelObjeto_TipoDeObjeto_Uso_Origen_TipoDeTabla_DatosQueContiene

A continuación, se detalla el significado de cada una las siglas que se utilizan para este proyecto:

1. TFG: Uso de la tabla para el Trabajo Final de Grado.
2. TD: Tabla de Dimensiones.
3. TH: Tabla de Hechos.
4. XLS: Tabla que carga todos sus datos desde un Excel.
5. PBI: El uso posterior del objeto es para representar datos en Power BI.
6. VW: El objeto es una vista.

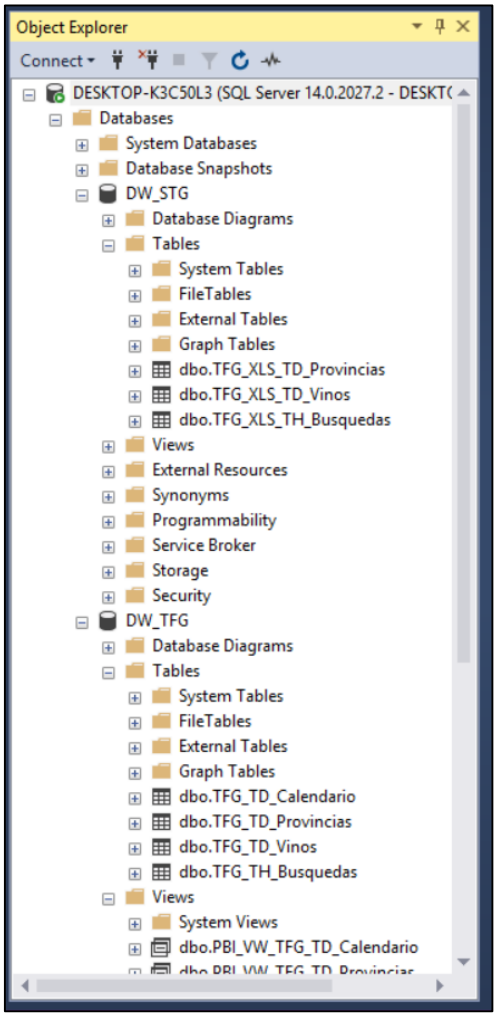


Ilustración 13: Tablas y vistas de DW_STG (Elaboración propia, 2019)

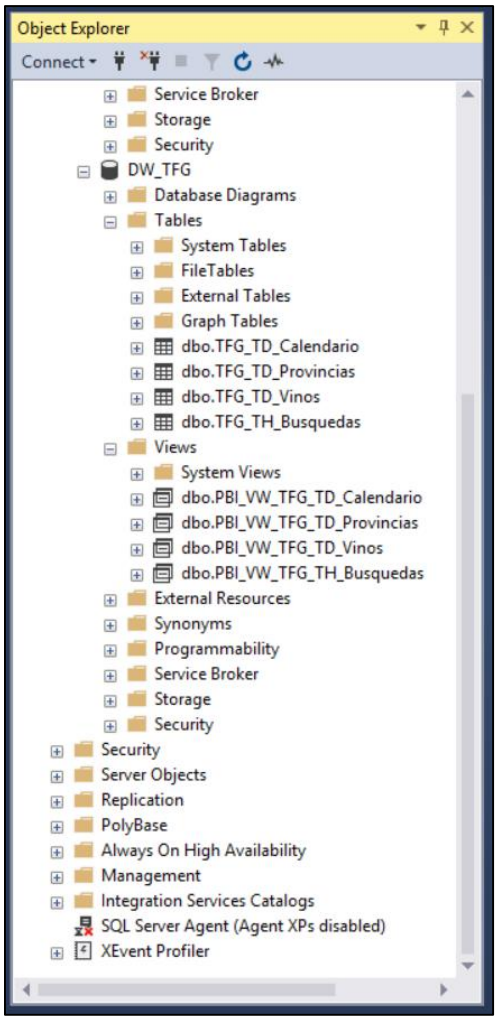


Ilustración 14: Tablas y vistas de DW_TFG (Elaboración propia, 2019)

En definitiva, en la parte de análisis se dispone de un modelo en estrella que contiene una Tabla de Hechos central, TFG_TH_Busquedas, y tres dimensiones de detalle correspondientes a la fecha, las Comunidades Autónomas y los vinos, TFG_TD_Calendario, TFG_TD_Provincias y TFG_TD_Vinos respectivamente.

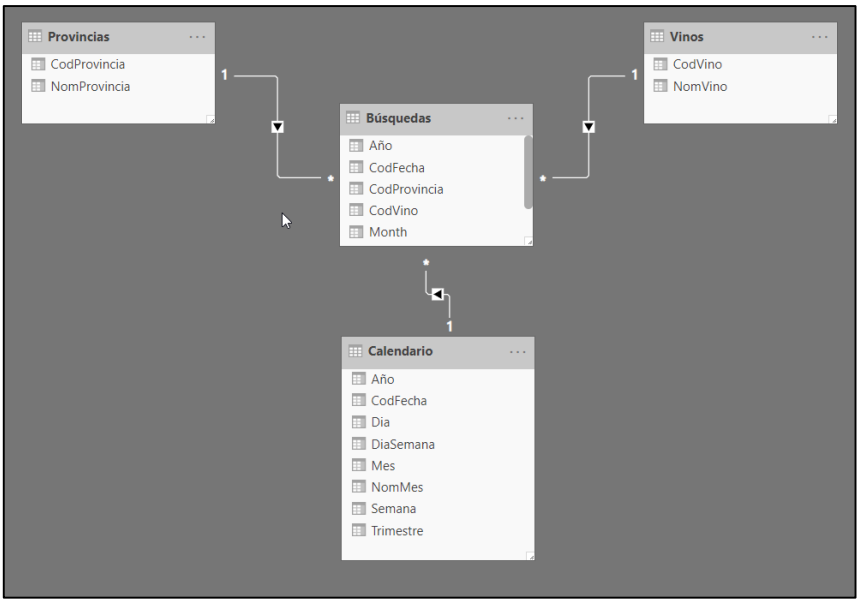


Ilustración 15: Modelo en estrella de las tablas utilizadas y sus relaciones (Elaboración propia, 2019)

7.3. *Microsoft SQL Server Integration Services* (SSIS)

Antes de comenzar a explicar la estructura que se dispondrá en *Microsoft SQL Server Integration Services*, se deben aclarar algunos conceptos que lo componen:

- Solución: Una solución o archivo .sln es un contenedor que agrupa y administra los proyectos que se utilizan. Una sola solución permite manejar uno o más proyectos relacionados.
- Proyecto de *Integration Services*: Contenedor en el que se desarrollan los paquetes del SSIS. Estos proyectos almacenan y agrupan los archivos relacionados con los paquetes. Por ejemplo, puede incluir todos los archivos necesarios para realizar un proceso ETL.
- Paquete de *Integration Services*: Un paquete es una colección organizada de conexiones, elementos de flujo de control y de datos, controladores de eventos, parámetros y configuraciones que se pueden manipular con la ayuda de herramientas gráficas o programación.

Por tanto, para la realización del proyecto se crea un archivo por cada uno de estos conceptos, TFG_BGARCIA.sln, TFG_BGARCIA.dtproj y TFG.dtsx respectivamente.

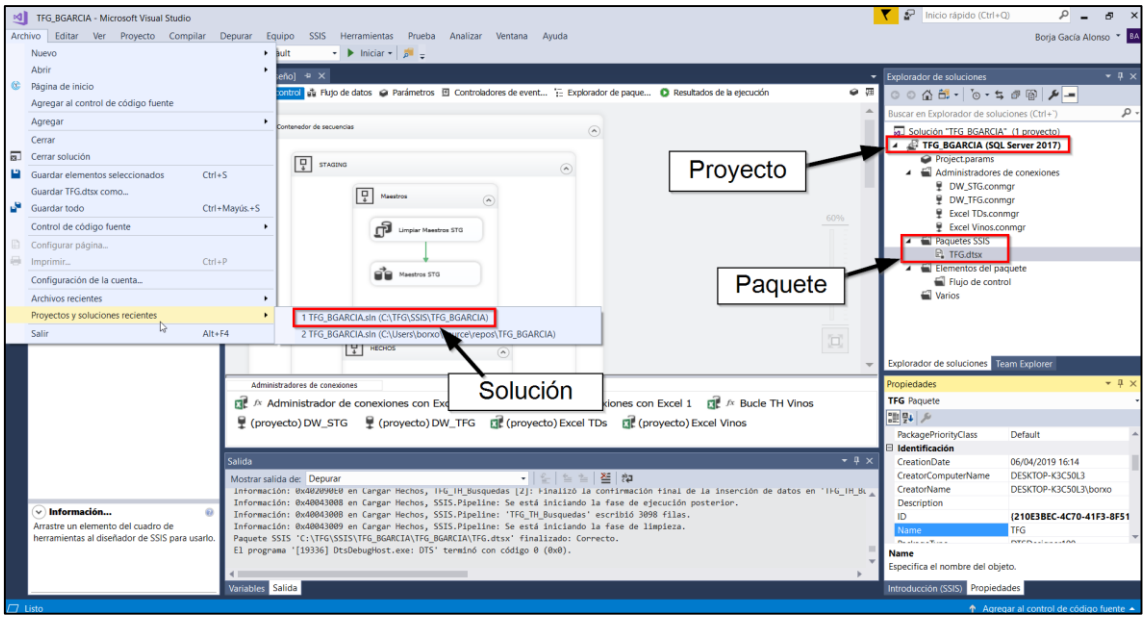


Ilustración 16: Estructura del paquete SQL Server Integration Services (Elaboració propia, 2019)

Dentro del paquete, se almacena todo el proceso de Extracción, Transformación y Carga y las conexiones necesarias para la realización del Trabajo Final de Grado.

Algunas de las conexiones existente son a las distintas bases de datos, DW_STG y DW_TFG, o a los diferentes orígenes. Por otra parte, los contenedores de secuencia que contiene el paquete se diferencian fundamentalmente en dos partes, la primera que hace referencia a la base de datos que contiene toda la información sin transformar, DW_TFG, y la segunda que hace referencia al *Data Warehouse*, el cual contendrá ya los datos transformados y listos para analizar, DW_TFG.

7.4. Power BI

Como se comenta a lo largo de todo el trabajo, *Microsoft Power BI* será la herramienta utilizada para representar gráficamente los datos correspondientes. Además, *Power BI* permitirá interactuar con la información de forma sencilla y cómoda.

Para cargar los datos en *Power BI* se usan las vistas de *SQL Server*. En el siguiente punto se hará referencia al por qué se hace uso de estas vistas para enviar los datos a *Power BI* y no las tablas.

Respecto a la estructura gráfica, se hace uso de cuatros secciones claramente definidas dentro de un cuadro de mando *Power BI*:

- Sección 1: En esta sección se sitúa el logo de la empresa. Como es un Trabajo Final de Grado, se ha introducido el logo de la Universidad Politécnica de Valencia (UPV).

- Sección 2: Sección donde se incluyen los filtros que ayudan al usuario final a interactuar con el cuadro de mando.

- Sección 3: Representa la última fecha de actualización del gráfico.

- Sección 4: Esta sección corresponde a todos los gráficos que incluye el cuadro de mando. Estos gráficos son los que proporcionan la información necesaria para analizar la situación y muestran detalles importantes para la toma de decisiones.

Ilustración 17: Estructura del cuadro de mano en Power BI (Elaboración propia. 2019)

Power BI Desktop, herramienta previamente utilizada para crear la parte gráfica del trabajo antes de subir el cuadro de mando al portal de PBI, dispone de todos los campos que se han cargado en la vista de SQL.

Estos campos se pueden observar en la parte derecha del panel. Además de disponer de los campos correspondientes a la vista, también es posible realizar nuevos campos mediante funciones DAX, lenguaje que se utiliza en *Power BI* para cálculos avanzados.

Respecto a los cálculos avanzados, en el trabajo se intentan llevar a cabo dentro de las consultas SQL, es decir, se guardan en el *Data Warehouse*. El objetivo de disponer los cálculos en la parte de programación SQL, cargados en las tablas o vistas, es que se pueden reutilizar para otras herramientas. Por ejemplo, si un cliente dispone de todos sus cálculos en las vistas de SQL para posteriormente visualizar los datos en *Power BI*, en un futuro puede decidir cambiar de herramienta gráfica sin ningún inconveniente, ya que los tendrá almacenados en su base de datos y no perderá ningún cálculo.

Para diferenciar los campos que se han creado en DAX con los que se cargan directamente de la vista y conservar un orden dentro de PBI, se añade una almohadilla delante de las medidas realizadas en DAX. Por ejemplo, #_MediaBusquedasVinos.

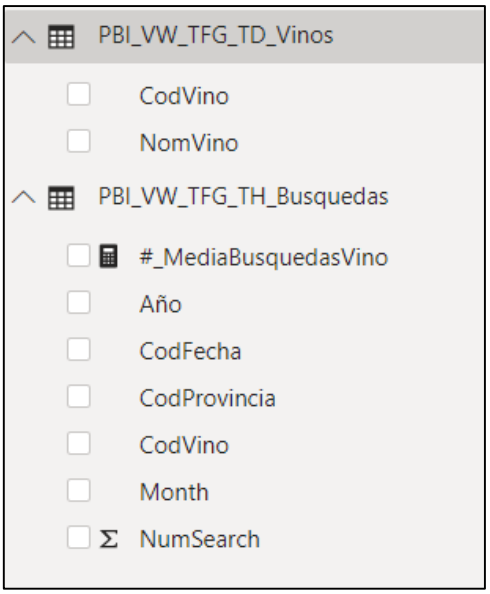


Ilustración 18: Campos o columnas en Power BI (Elaboración propia, 2019)

Por último, en *Power BI* se aplica lenguaje de negocio como se podrá ver al finalizar el proyecto. Esto significa que, por ejemplo, aunque el campo con el que se diseñe esté nombrado como NomVino y sea bastante intuitivo, no es la forma idónea de entrega al usuario final. Por esto mismo, sería conveniente realizar un cambio en el nombre a una forma correcta, como puede ser Nombre Vino, Denominación de Origen o Vino.

8. Desarrollo del proyecto

Este apartado se centra en cómo se ha implantado todo lo necesario para el Trabajo Final de Grado y en el desarrollo que se ha realizado.

Se diferencian varios puntos para describir esta parte del proyecto. Estos puntos son: Instalación de herramientas, preparación del entorno, descarga de datos, proceso ETL (*Extraction-Transformation-Load*), *Data Warehouse*, representación de datos en *Power BI y* Predicciones *con Python*.

8.1. Instalación de herramientas

8.1.1. *SQL Server*

Para instalar *SQL Server* hay que descargarse el Centro de instalación de *SQL Server*. Para realizar este proyecto se ha descargado *SQL Server Developer*, una edición gratuita con todas las características que se pueden usar como bases de datos de desarrollo y pruebas en un entorno que no sea de producción.

Una vez que se ha instalado el *Centro de instalación de SQL Server*, solo hay que seleccionar el asistente para iniciarlo. Con este método de ayuda se crea el motor de base de datos, donde se eligen las propiedades que se necesiten para nuestro proyecto y la creación de la instancia (DESKTOP-K3C50L3).

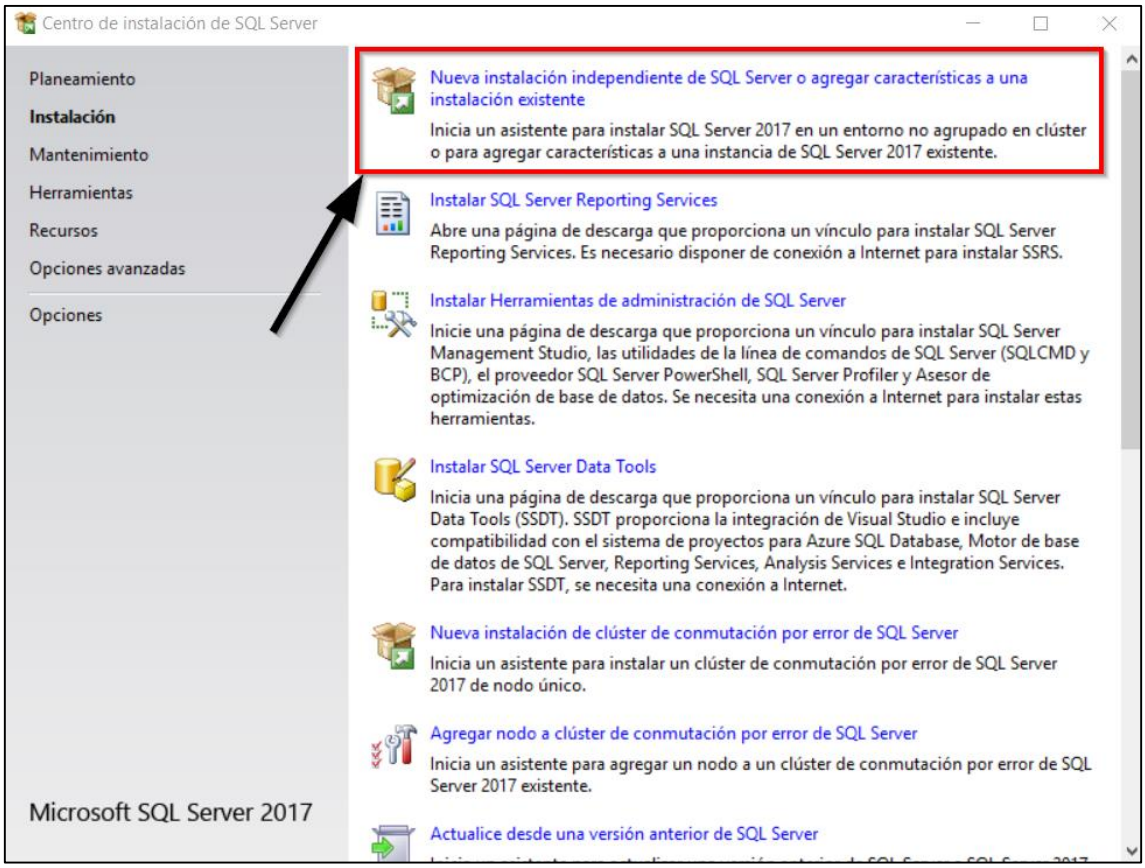


Ilustración 19: Asistente de instalación de SQL Server (Elaboración propia, 2019)

8.1.2. *Microsoft SQL Server Management Studio*

Para gestionar el motor de base de datos, se ha instalado *Microsoft SQL Server Management Studio*. Para instalar esta herramienta se debe hacer desde el mismo Centro de instalación de *SQL Server* o desde la web oficial de Microsoft. En este caso se ha optado por la primera, debido a su facilidad.

Una vez instalada, hay que conectarse introduciendo la instancia y sus credenciales correspondientes. En este caso son las credenciales de *Windows*, ya que al crear el motor de base de datos se ha proporcionado esta propiedad a la instancia.

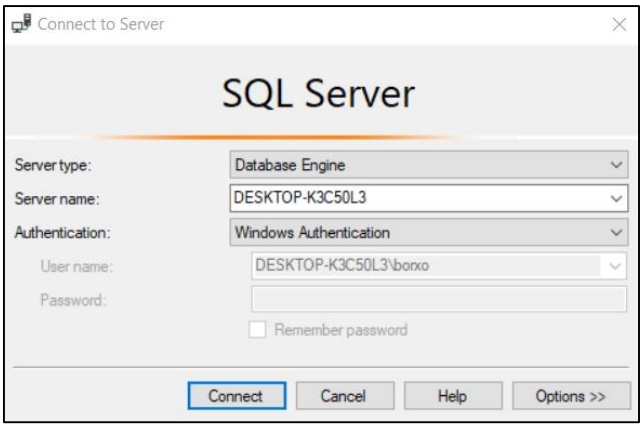


Ilustración 20: Introducción de las credenciales en SQL Server Management Studio (Elaboración propia, 2019)

Una vez conectado, ya se pueden visualizar todos los elementos que se pueden gestionar de la instancia que se ha creado: bases de datos, seguridad, etc.

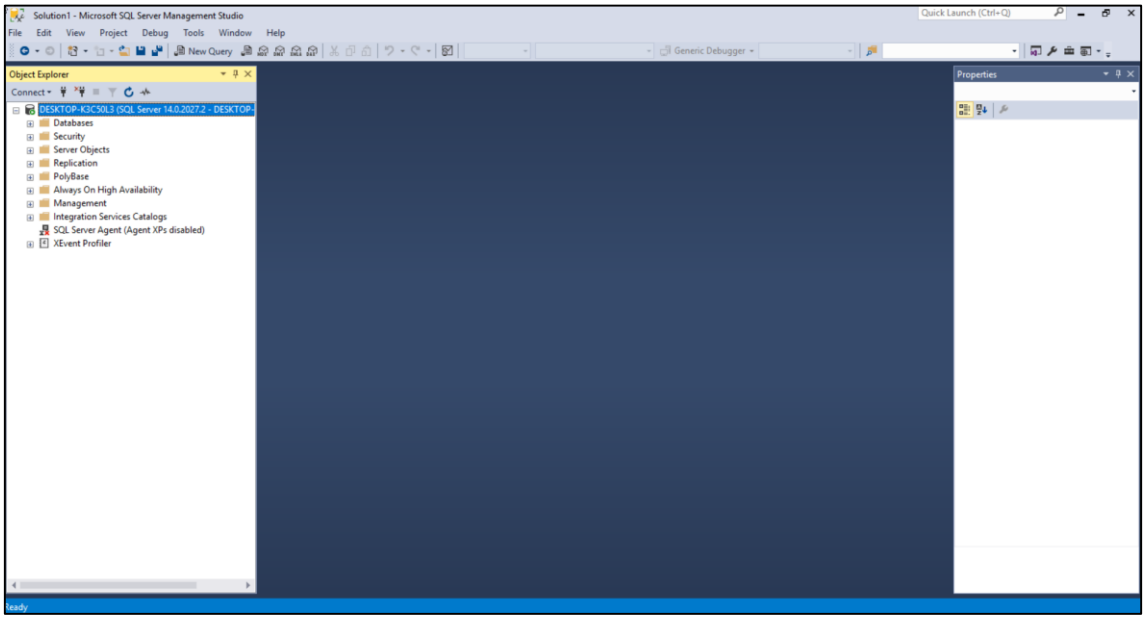


Ilustración 21: Visualización de la interfaz de SQL Server Management Studio (Elaboración propia, 2019)

8.1.3. SQL Server Data Tools

SQL Server Data Tools (SSDT) es la herramienta que se ha descargado para gestionar los paquetes de SSIS. Cuando se descarga SSDT, se debe seleccionar en la instalación los paquetes de *Business Intelligence* para que sea compatible con los proyectos de *SQL Server Integration Services* (SSIS).

La descarga de SSDT incluye *Visual Studio*, que ayudará a gestionar este proyecto con una interfaz visual. *SQL Server Data Tools* incluye compatibilidad con el motor de base

de datos como con otras herramientas de Microsoft, y su instalación también se realiza desde el Centro de instalación de *SQL Server*.

8.1.4. *Power BI*

Para obtener *Power BI* hay que descargar *Power BI Desktop*, herramienta que permite diseñar el cuadro de mando para el análisis, tanto la versión web como la versión móvil.

Esta herramienta no es donde se presenta el trabajo, ya que la ventaja que ofrece PBI es que dispone de un portal donde el usuario final puede observar e interactuar con el cuadro de mando sin tener que manipular las capas correspondientes al diseño.

Para subir el cuadro de mando al portal y poderlo compartir con otros usuarios se necesita una licencia de *Power BI Pro*. La licencia que se ha utilizado en este proyecto es gracias a Nunsys, especialmente al departamento de B*usiness Intelligence*, empresa de la cual se han adquirido los conocimientos para poder gestionar estas herramientas.

Power BI Desktop es una herramienta gratuita que se puede descargar desde la web oficial de Microsoft:

Ilustración 22: Página de descarga de Power BI (Power BI, 2014)

8.2. Preparación del entorno

Como se ha dicho en apartados anteriores, antes de comenzar a desarrollar se han creado las dos bases de datos correspondiente a la parte de *SQL Server Management Studio* y la solución, proyecto y paquete haciendo referencia a *Microsoft SQL Server Integration Services*.

Respecto a las bases de datos que se necesita para almacenar la información, desde *SQL Server Management Studio* se selecciona la opción *New Database* en el explorador

de objetos y, a continuación, se proporciona un nombre para identificarlas. De esta forma, se crea tanto la base de datos que almacenará toda la información sin transformar, DW_STG, como el *Data Warehouse*, DW_TFG.

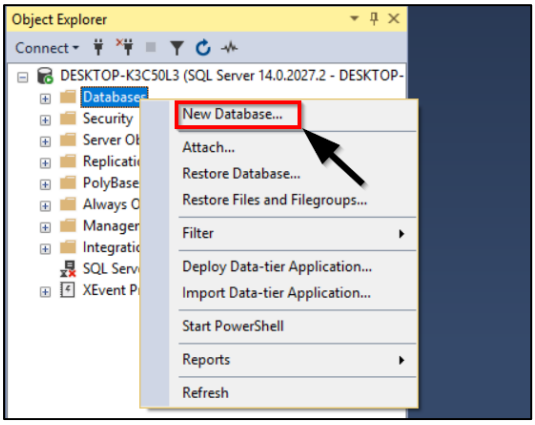


Ilustración 23: Creación de una nueva base de datos (Elaboración propia, 2019)

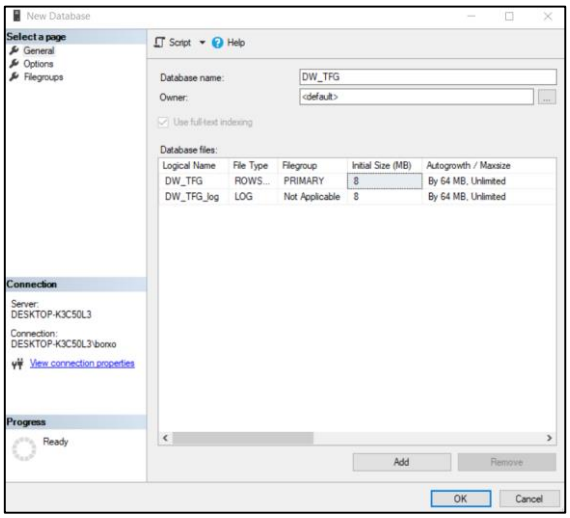


Ilustración 24: Propiedades de una nueva base de datos (Elaboración propia, 2019)

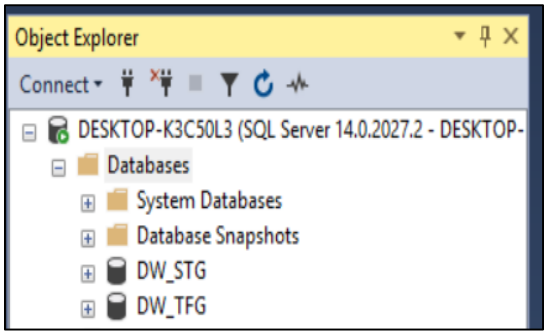


Ilustración 25: Bases de datos utilizadas en este proyecto (Elaboración propia, 2019)

Para trabajar con *SQL Server Integration Services*, en *Visual Studio* se crea un proyecto con el nombre TFG_BGARCIA, el cual incluye dentro del mismo un archivo

TFG_BGARCIA.sln con su respectivo proyecto y un paquete vacío, que se renombra como TFG.dtproj.

El proyecto tiene que ser del tipo *Intregation Services Project*, ya que es el formato que permite crear un proceso ETL para realizar toda la preparación que necesitan los datos utilizados para su análisis.

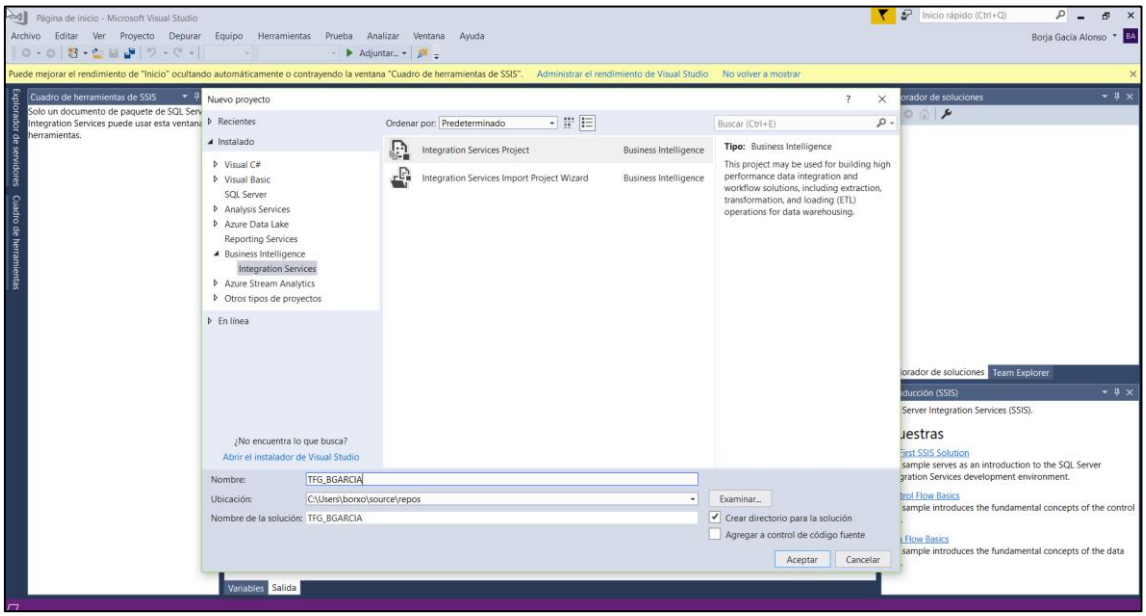


Ilustración 26: Creación de un proyecto de SQL Server Integration Services (Elaboración propia, 2019)

Posteriormente, se establece conexión con las diferentes fuentes de origen que se van a utilizar, tanto de extracción como de cargas. Las conexiones que se necesitan en este proyecto son cuatro:

1. DW_STG: Conexión a la base de datos para almacenar la información sin transformar.
2. DW_TFG: Conexión al *Data Warehouse.*
3. Excel TDs: Conexión a hoja de Excel con diferentes pestañas que contienen las Tablas de Dimensiones que se cargarán en la base de datos.
4. Excel Vinos: Conexión a una hoja de Excel que contiene las pestañas con el número de búsquedas de las diferentes Denominaciones de Origen. Esta hoja de Excel es el que se utiliza para construir la TH.

Para establecer la conexión con las hojas de Excel que se emplean en este TFG, deben haberse creado con anterioridad. Por esto mismo se han diseñado los orígenes y descargado toda la información en paralelo a este mismo punto.

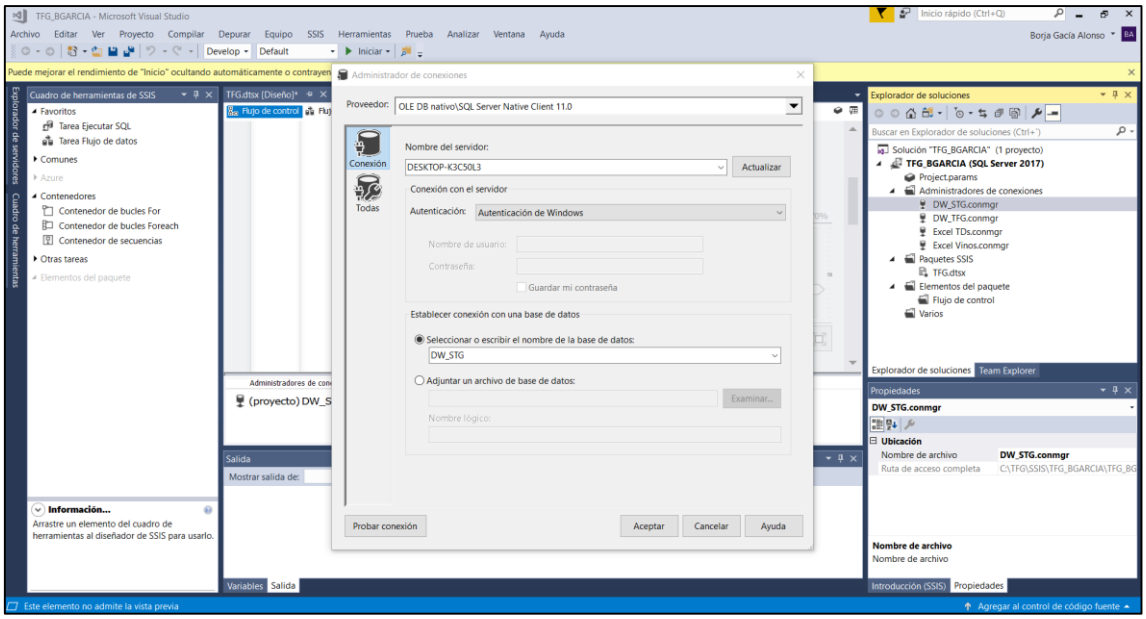


Ilustración 27: Creación de la conexión a la base de datos DW_STG en el paquete de SQL Server Integration Services (Elaboración propia, 2019)

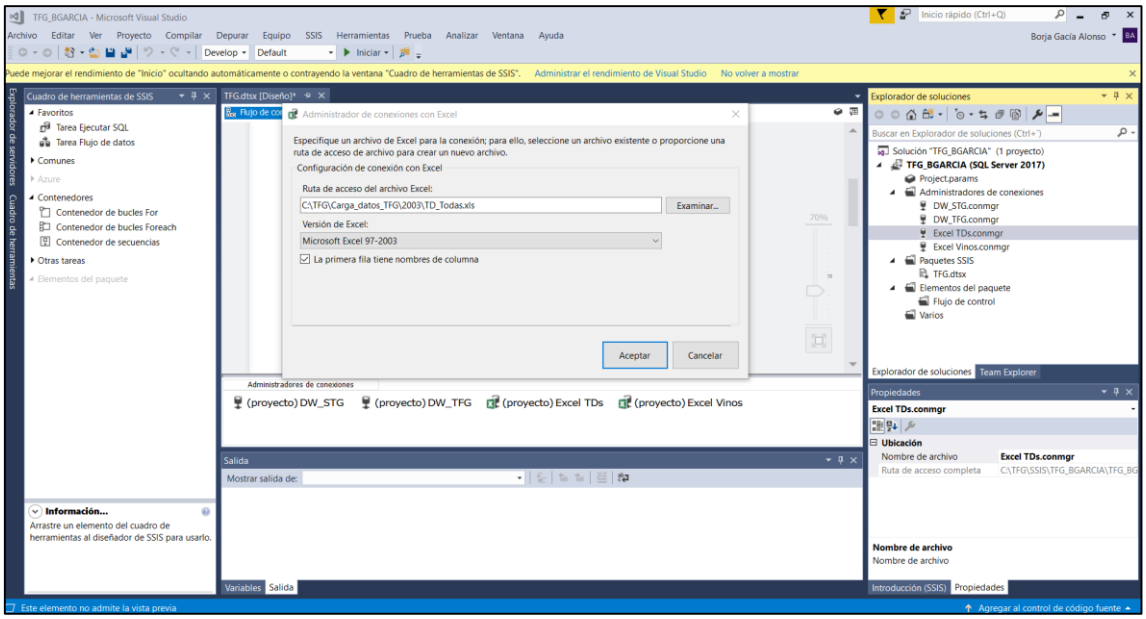


Ilustración 28: Creación de la conexión al excel que contiene las Tablas de Dimensiones en SQL Server Integration Services (Elaboración propia, 2019)

8.3. Orígenes y descarga de datos

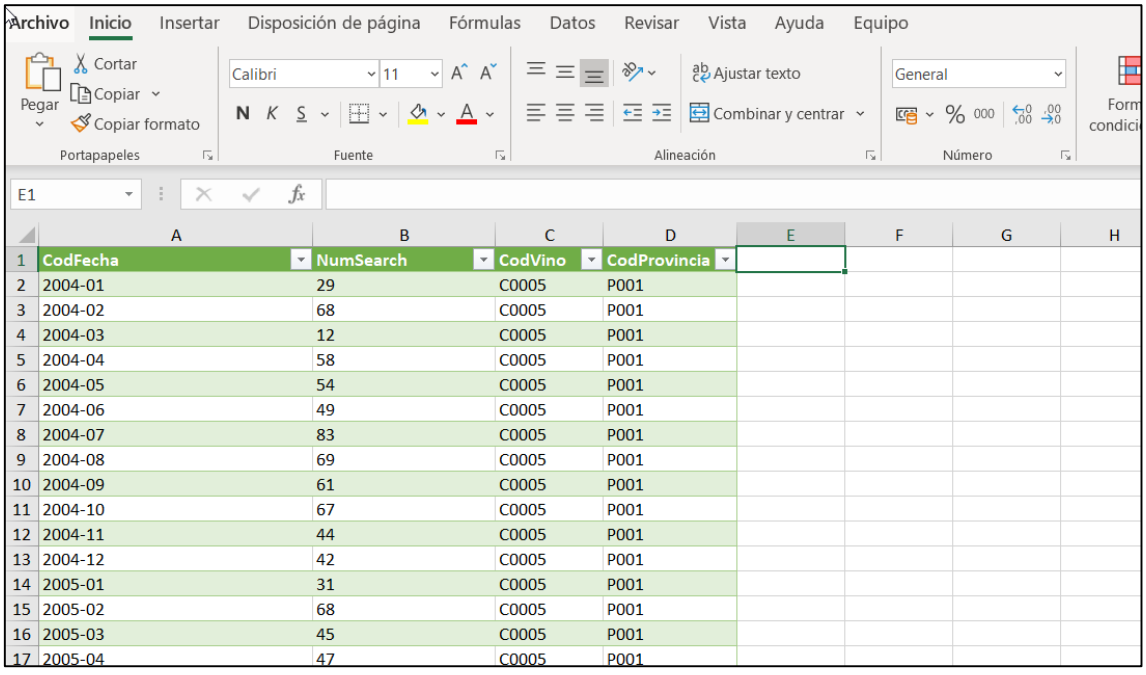
Los orígenes de datos que se han utilizado, exceptuando las bases de datos que ya se han comentado, DW_STG y DW_TFG, son dos hojas de Excels que contienen información correspondiente a la Tabla de Hechos y las Tablas de Dimensiones que se utilizan en este proyecto.

Respecto a la Tabla de Hechos, los datos que se han introducido para su posterior análisis son extraídos desde la web *Google Trends*. Esta página proporciona la tendencia de búsquedas de la *keyword* que se requiera.

Las *keywords* buscadas son 30 Denominaciones de Origen diferentes, las cuales se han descargado sus datos de enero del 2004 hasta abril de 2019. Los datos han sido descargados en una hoja de Excel, dispuestos en dos columnas diferentes: fecha y número de búsquedas. El total de búsquedas que se analizarán es de 97.889.

Una vez se dispone de una hoja de Excel para cada tipo de vino, se crea una nueva que incluye todas las D.O. en diferentes pestañas. Esta decisión se ha tomado dado que de esta forma tan solo es necesario crear una conexión en nuestro proyecto TFG_BGARCIA, en lugar de 30 si se hubiera optado por disponer una hoja de Excel por cada Denominación de Origen.

Por otra parte, para cada Denominación de Origen y Comunidad Autónoma, se ha añadido una nueva clave en dos nuevas columnas, CodVino y CodProvincia respectivamente. Como se ha explicado en apartados anteriores, es fundamental que la TH contenga toda la información posible en claves, ya que se utilizarán las mismas para unirlas y extraer más información de las Tablas de Dimensiones, además de tener identificados todos los datos. Esta tarea se realiza para todas las D.O.



	A	B	C	D	E	F	G	H
1	CodFecha	NumSearch	CodVino	CodProvincia				
2	2004-01	29	C0005	P001				
3	2004-02	68	C0005	P001				
4	2004-03	12	C0005	P001				
5	2004-04	58	C0005	P001				
6	2004-05	54	C0005	P001				
7	2004-06	49	C0005	P001				
8	2004-07	83	C0005	P001				
9	2004-08	69	C0005	P001				
10	2004-09	61	C0005	P001				
11	2004-10	67	C0005	P001				
12	2004-11	44	C0005	P001				
13	2004-12	42	C0005	P001				
14	2005-01	31	C0005	P001				
15	2005-02	68	C0005	P001				
16	2005-03	45	C0005	P001				
17	2005-04	47	C0005	P001				

Ilustración 29: Pestaña del excel de carga de la TH con todas las columnas necesarias para extraer la información (Elaboración propia, 2019)

Respecto a las Tablas de Dimensiones, son un mismo Excel con dos pestañas: Vinos y Comunidades Autónomas. En las TD se almacena el código de la D.O. y el código de la Comunidad Autónoma respectivamente, además del nombre correspondiente.

Es importante destacar que en estas tablas no pueden existir códigos duplicados. De todas formas, cuando se realiza el proceso ETL se asegura con el código SQL que no se permite que ningún código se repita.

8.4. Proceso ETL (*Extract-Transformation-Load*)

En este punto se habla de la implantación del proceso ETL, una de las etapas más importantes del proyecto. En este proceso, como ya se ha hablado en puntos anteriores, se realiza la extracción de los datos de su origen, la transformación correspondiente y su respectiva carga a nuestro *Data Warehouse*.

Para realizar el proceso de Extracción, Transformación y Carga de los datos se divide en dos etapas: *Staging* (STG) y DW. Esta parte del proyecto se realiza desde el proyecto y paquete de *SQL Server Integration Services* que se ha creado en *Visual Studio*, herramienta que ayuda de forma gráfica a interactuar con el SSTD.

Para realizar estas dos etapas se necesitan varios objetos que van incluidos en un paquete de *Business Intelligence*:

1. Contenedor de secuencias: Agrupa el flujo de control en subsistemas más sencillos.
2. Tarea Ejecutar SQL: Ejecuta instrucciones SQL o procedimientos almacenados en una base de datos relacional.
3. Tarea Flujo de datos: Mueve datos entre orígenes y destinos durante las transformaciones y limpiezas. También se pueden ejecutar instrucciones SQL.
4. Origen de Excel: Conecta y extrae datos de una hoja excel.
5. Origen de OLE DB: Extrae datos de una base de datos relacional compatible con OLE DB. Se obtienen los datos desde una vista o tabla de base de datos, o se utiliza una consulta SQL para extraer los mismos de uno de los objetos anteriormente mencionados.
6. Destino de OLE DB: Carga los datos extraídos de un origen en una base relacional compatible con OLE DB.

Para diferenciar STG y DW se ha creado un contenedor de secuencias para cada una de las partes. A su vez, dentro de los mismos contenedores de secuencias, se añaden dos más, que harán referencia a las Tablas de Dimensiones y las Tablas de Hechos, llamados Maestros y Hechos respectivamente.

A continuación, en el interior de los contenedores Maestros y Hechos se ha añadido una Tarea Ejecutar SQL y una Tarea Flujo de datos.

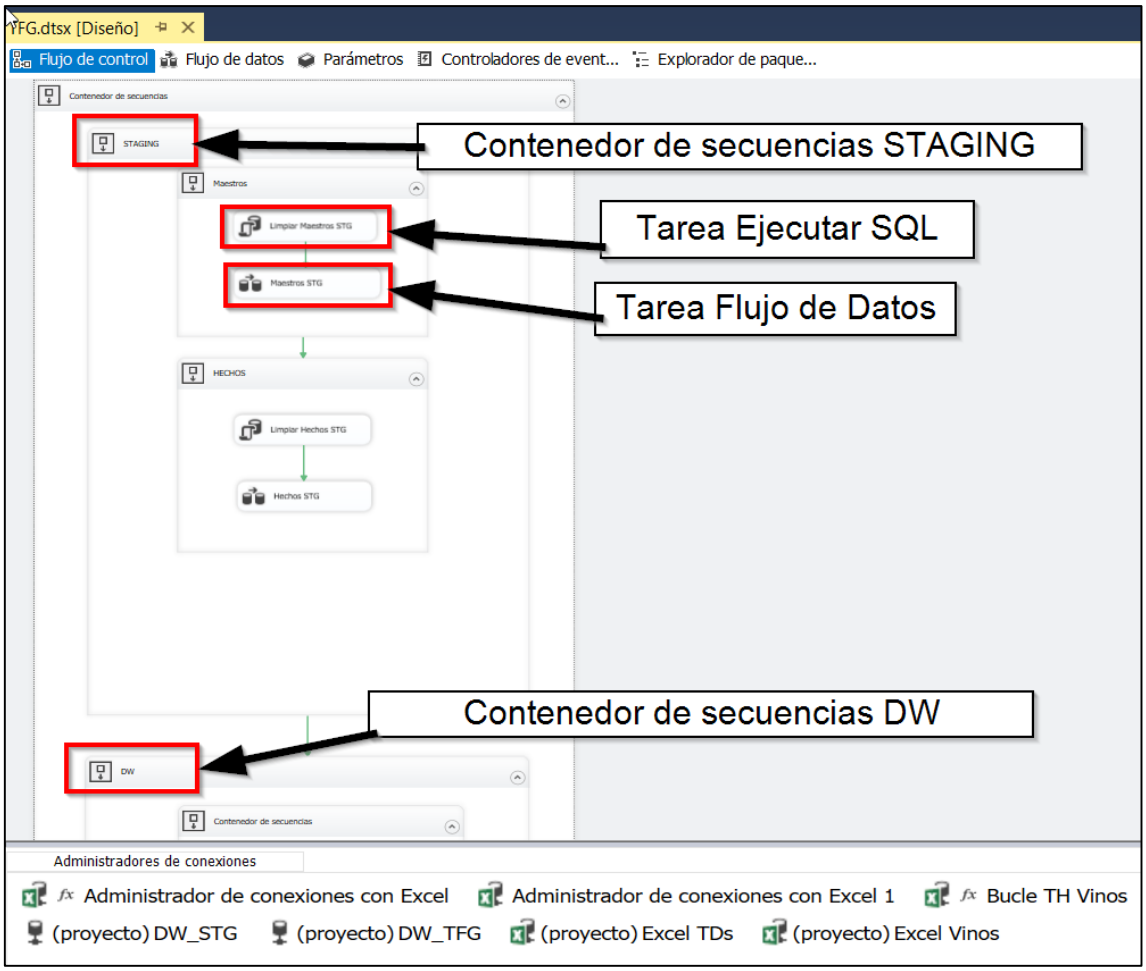


Ilustración 30: Organización del paquete SQL Server Integration Services. Explicación de los contenedores para el proceso ETL (Elaboración propia, 2019)

La primera de ellas, Tarea Ejecutar SQL, se utiliza para limpiar las tablas que se utilizan en este proyecto. Cuando se habla de limpiar tablas hace referencia a borrar todos o parte de los datos que existían dentro de la misma, para que no dupliquen con los nuevos que se van a añadir. Para ello se utiliza la función TRUNCATE de SQL, que borrará todas las filas de la tabla correspondiente.

Por ejemplo, se dispone de una hoja de Excel como origen, la cual se actualiza cada 2 horas y añade 20 filas nuevas en cada una de estas actualizaciones. Al realizar de nuevo la carga de toda esta información de la hoja de Excel a nuestra tabla de BBDD, se cargarán todas las filas existentes más las 20 nuevas. Si no se realiza la función TRUNCATE a la tabla correspondiente, todas las filas duplicarán en un futuro.

Algunos programadores de SQL pueden pensar que, si se realiza la función DELETE la cual se puede añadir un filtro para solo borrar parte de los datos, será una consulta más eficiente. Por ejemplo, si se escribe la función DELETE con un filtro de fecha, se puede eliminar y cargar tan solo los últimos dos meses, que será un número de filas inferior que si se borran y cargan todas las del año.

La lógica utilizada para aplicar un TRUNCATE en lugar de un DELETE es la siguiente: Cuando se realiza un DELETE con un filtro de fecha, la consulta irá fila por fila comprobando cuáles cumplen la condición del filtro, sin embargo, el TRUNCATE

simplemente borra todas las filas instantáneamente. Esto mismo hace que la función DELETE sea menos eficiente, ya que en este proyecto no se trata con un almacén de datos con grandes cantidades de filas.

El inconveniente proviene cuando se trabaja con almacenes de datos mucho más voluminosos. Si se realiza un TRUNCATE a una tabla de millones de filas, se deberán cargar de nuevo desde el origen para no perder esta información, por tanto, se podría ralentizar la consulta de carga. Por el contrario, con un DELETE se puede borrar la cantidad mínima de filas imprescindibles para realizar posteriormente la carga de esa misma porción.

En estas situaciones, antes de implementar cualquiera de las dos soluciones, se debe estudiar y validar qué opción es la más recomendable. En el caso de este Trabajo Final de Grado, como no se tratan volúmenes grandes de información, el proceso de limpieza se realiza mediante la instrucción TRUNCATE.

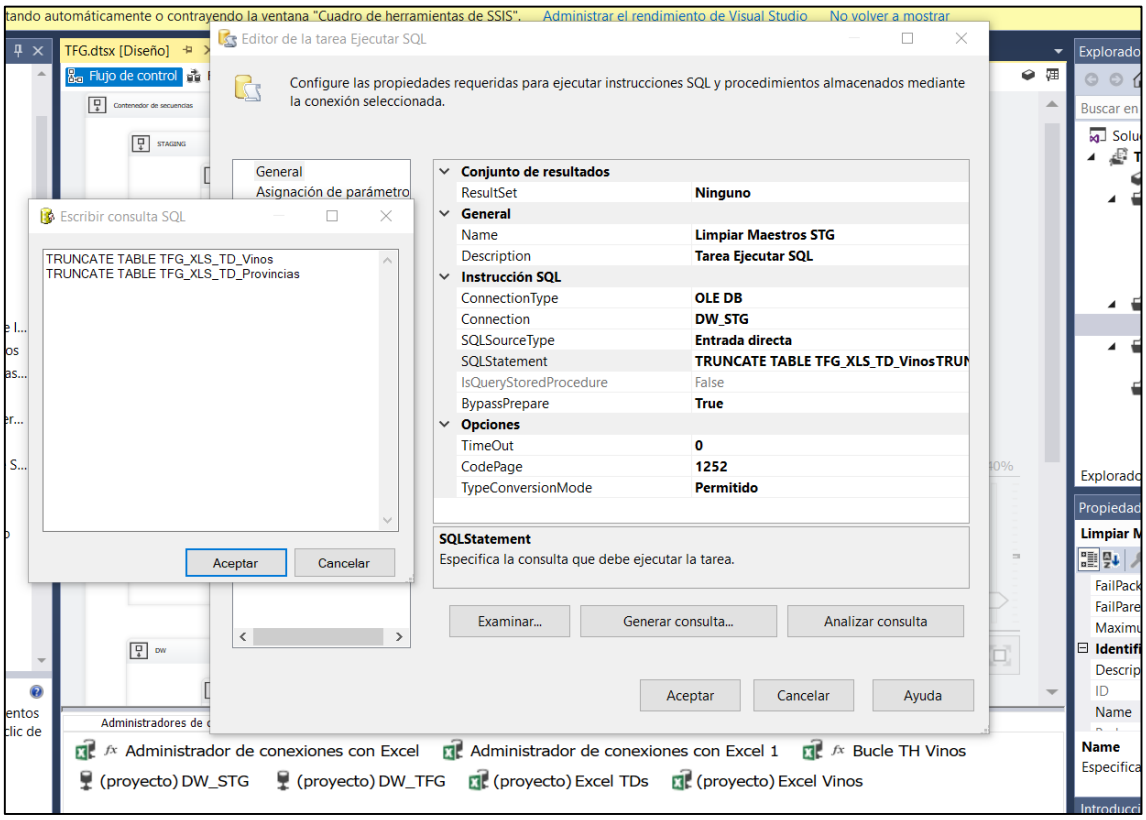


Ilustración 31: Ejemplo de la tarea Ejecutar SQL con la función TRUNCATE (Elaboración propia, 2019)

Antes de mencionar qué tablas se han truncado en este proceso ETL, es importante ver cómo se cargan los datos y cómo se pueden crear tablas desde el paquete de *SQL Server Integration Services.*

Dentro de la Tarea Flujo de Datos, concretamente para la parte de STG, se utilizan las tareas Origen de Excel y Destino de OLE DB, tanto para las Tablas de Dimensiones como las Tablas de Hechos.

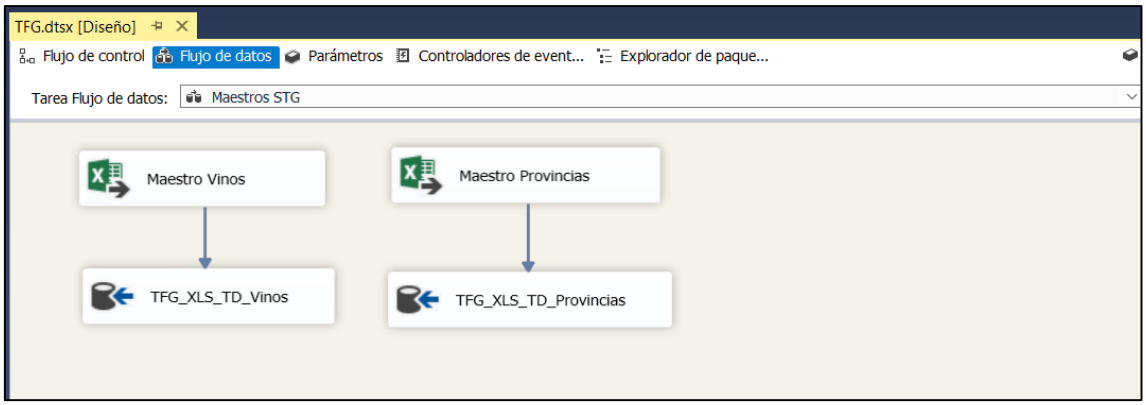


Ilustración 32: Extracción de los datos desde Excel hasta DW_STG (Elaboración propia, 2019)

Para cargar los datos desde una hoja de Excel, en las características de la tarea Origen de Excel, tan solo se introduce la conexión y pestaña correspondiente para extraer los datos que se necesitan.

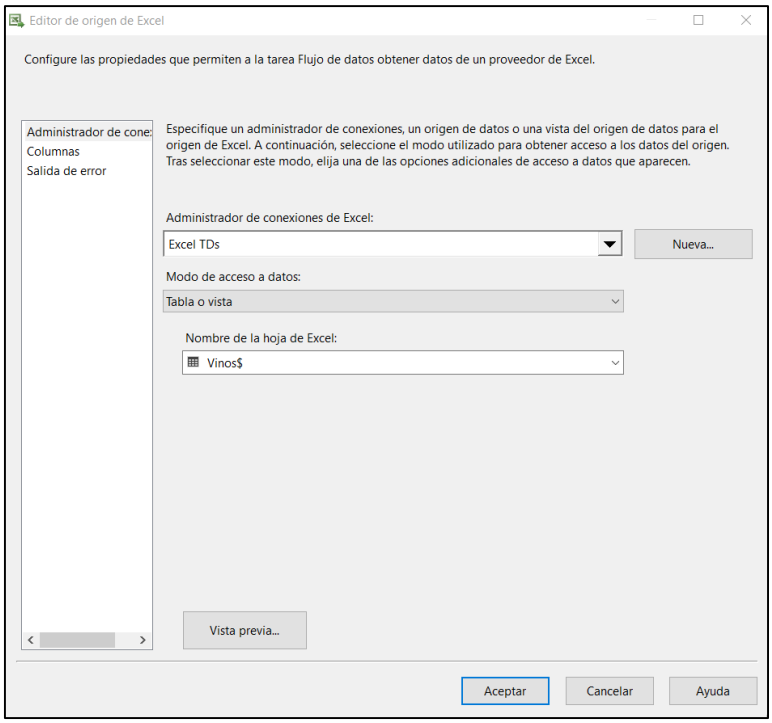


Ilustración 33: Carga de la información desde una hoja de Excel (Elaboración propia, 2019)

A continuación, en la tarea Destino de OLE DB, se introduce la conexión a la base de datos DW_STG. Dentro de la misma tarea, se selecciona la opción Nueva. Esta opción creará la tabla en DW_STG y las columnas de la misma. Los datos contienen el mismo formato que proviene de la hoja de Excel, por defecto serán cadenas de carácteres (nvarchar).

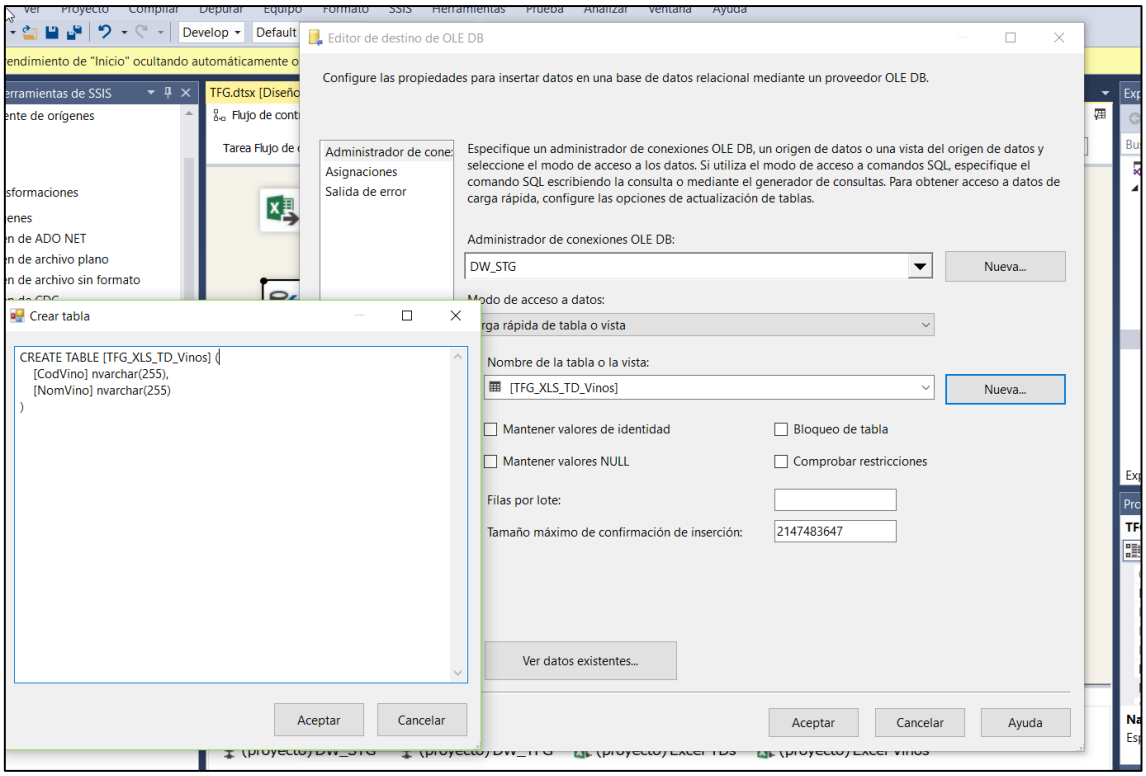


Ilustración 34: Creación y carga de una nueva tabla en la base de datos en TFG_STG (Elaboración propia, 2019)

Las tablas que se limpian y cargan en DW_STG (contenedor de secuencias *Staging*) son las siguientes:

- Tablas de Dimensiones:
  - TFG_XLS_TD_Provincias
  - TFG_XLS_TD_Vinos
- Tablas de Hechos:
  - TFG_XLS_TH_Busquedas

Por otra parte, en el contenedor de DW se aplican los mismos pasos, aunque en esta ocasión se utiliza la Tarea Origen de OLE DB en lugar de Origen de Excel, ya que se extraen los datos desde DW_STG. En esta base de datos ahora se dispone de toda la información que se ha obtenido anteriormente desde *Google Trends,* en archivos de Excel.

En la parte de DW se llevan a cabo los cambios necesarios para su posterior análisis. En la tarea Origen de OLE DB se introduce la conexión creada para DW_STG y se crea una consulta que cargue los datos en el formato que facilite el desarrollo del proyecto, además de añadir nuevas columnas en el caso que se requieran. Los datos extraídos desde Excel se cargan como cadena de caracteres en DW_STG, por consiguiente, se aplica el formato correcto para cada columna en la transformación y carga a DW_TFG. El formato que se aplica en las columnas es el siguiente:

- CodFecha: Se convierte la columna a tipo Date.
- NumSearch: Se convierte la columna a tipo int.
- CodVino: Se convierte la columna a un nvarchar de longitud 50.

- CodProvincia: Se convierte la columna a un nvarchar de longitud 50.
- NomVino: Se convierte la columna a un nvarchar de longitud 250.
- NomProvincia: Se convierte la columna a un nvarchar de longitud 250.

En la tarea Destino OLE DB se realizarán los mismos pasos que se han trazado en el contenedor de secuencia Staging, pero con conexión al *Data Warehouse*.

En resumen, en esta parte se crean y truncan las tablas que se utilizan en el análisis posterior.

- Tablas de Dimensiones:
    - TFG_TD_Provincias
    - TFG_TD_Vinos
- Tablas de Hechos:
    - TFG_TD_Busquedas

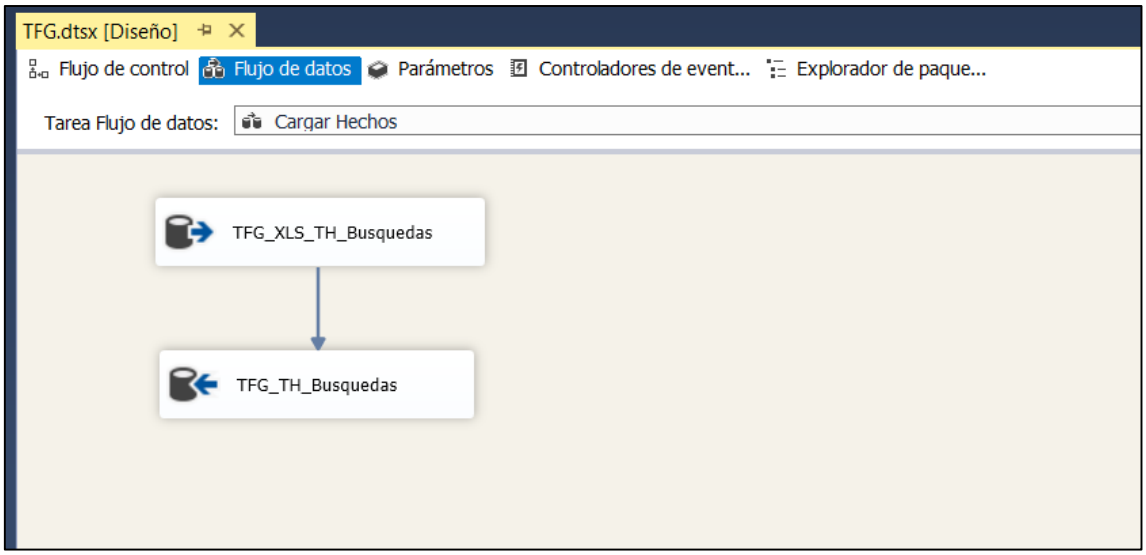


Ilustración 35: Carga de datos desde DW_STG a DW_TFG (Elaboración propia, 2019)

```
SELECT
    cast(CodFecha as date) as CodFecha,
    cast(NumSearch as int) as NumSearch,
    cast(CodVino as nvarchar(50)) as CodVino,
    cast(CodProvincia as nvarchar(50)) as CodProvincia
FROM
    TFG_XLS_TH_Busquedas
```

Ilustración 36: Consulta de carga para TFG_TH_Busquedas (Elaboración propia, 2019)

Todo este proceso es con el fin de disponer de la información necesaria y más integra posible. Las tablas contenidas en el *Data Warehouse* deben estar validadas y confirmar la calidad de todos sus datos.

8.5. Carga y visualización de datos en Power BI

Una vez disponible toda la información en nuestro *Data Warehouse*, se realiza la carga de la misma en *Power BI* para representar gráficamente los datos. Desde la herramienta *Power BI Desktop*, se establece conexión a la instancia DESKTOP-K3C50L3 y la base de datos DW_TFG. Una vez conectada, se extrae la información a través de las tablas.

En un principio, esta es la forma más habitual de trabajar. Sin embargo, en este proyecto se efectúa una pequeña modificación: utilizar vistas de SQL en lugar de tablas.

Se utilizan las vistas SQL para cargar los datos en *Power BI* porque se puede modificar la estructura sin afectar a las tablas, tanto las TH como las TD, las cuales contienen todos los datos correctos y limpios. Gracias a las validaciones aplicadas, se asegura la calidad de los datos de las tablas.

Por esto mismo, es conveniente que cualquier modificación aplicada se realice a nivel de vista y no de tabla, ya que si se aplican cambios a nivel de tabla puede verse afectada la integridad del *Data Warehouse*.

Por otra parte, si se necesitan crear nuevas medidas como sumas, medias o divisiones para la representación gráfica, se pueden realizar a nivel de vista. De esta forma no se modifica la carga de ninguna tabla dentro de *SQL Server Integration Services*.

Por tanto, para cargar los datos se ha creado una vista por cada tabla:

- PBI_VW_TFG_TD_Provincias
- PBI_VW_TFG_TD_Vinos
- PBI_VW_TFG_TH_Busquedas
- PBI_VW_TFG_Calendario

```
CREATE VIEW [dbo].[PBI_VW_TFG_TH_Busquedas] AS

SELECT
    CodFecha,
    YEAR(CodFecha) as Año,
    MONTH(CodFecha) as Month,
    cast(NumSearch as int) as NumSearch,
    CodVino,
    CodProvincia,
    NumVinos
FROM
    TFG_TH_Busquedas
```

Ilustración 37: Consulta de carga de la vista PBI_VW_TFG_TH_Busquedas (Elaboración propia, 2019)

A continuación, se cargan los datos en *Power BI.* Para ello, se siguen los pasos del asistente de conexión que ofrece *Power BI Desktop*, donde se introduce la instancia, base de datos y contraseña correspondiente. En la imagen siguiente se puede ver un ejemplo del código que se genera para conectar con las diferentes vistas de DW_TFG:

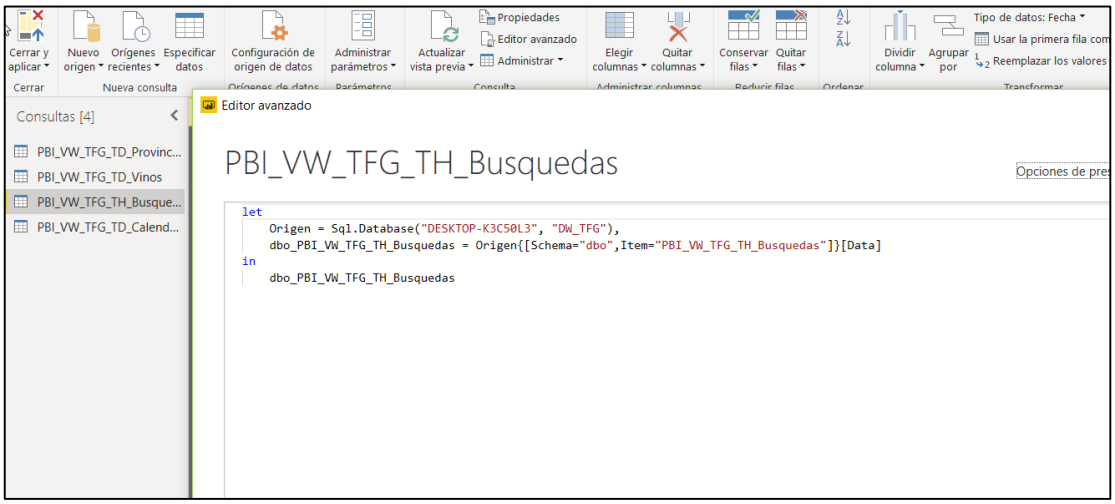


Ilustración 38: Conexión a la base de datos DW_TFG desde Power BI (Elaboración propia, 2019)

Después de cargar los datos, se cerciora de que las tablas se han entrelazado correctamente. *Power BI* identifica automáticamente los campos que se pueden relacionar, para realizar esta función automáticamente deben contener el mismo nombre, como puede ser el campo CodProvincia de la PBI_VW_TFG_TD_Provincias y PBI_VW_TFG_TH_Busquedas.

En ocasiones *Power BI* no detecta las relaciones automáticamente, entonces se debe crear manualmente la relación, ya que si los campos no están bien enlazados pueden dar lugar a duplicados a la hora de representar los gráficos. En este proyecto, se puede observar que PBI ha detectado el diagrama en estrella:

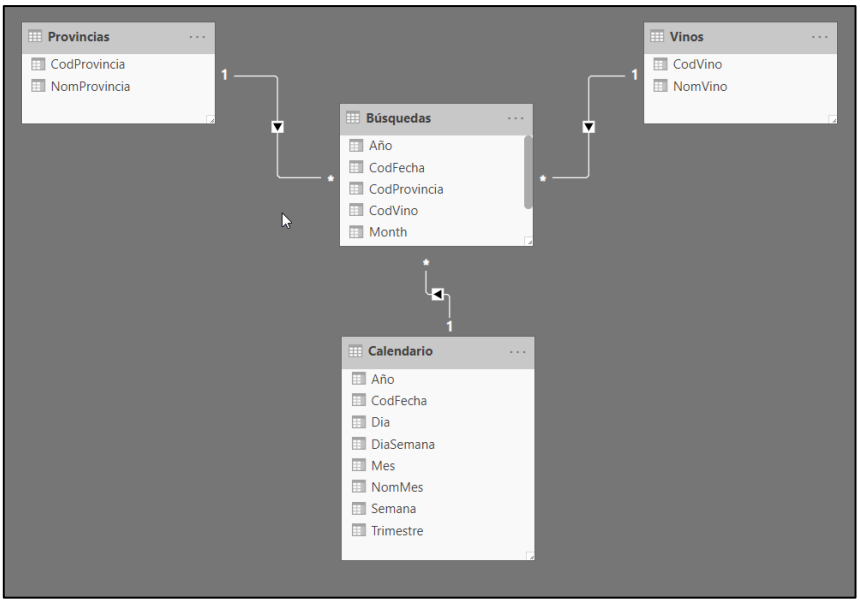


Ilustración 39: Diagrama en estrella detectado por Power BI (Elaboración propia, 2019)

Una vez terminada esta parte, en el menú inicial de *Power BI Desktop* se dispone de todas las tablas con sus correspondientes datos, los cuales se pueden utilizar para crear gráficos interactivos:

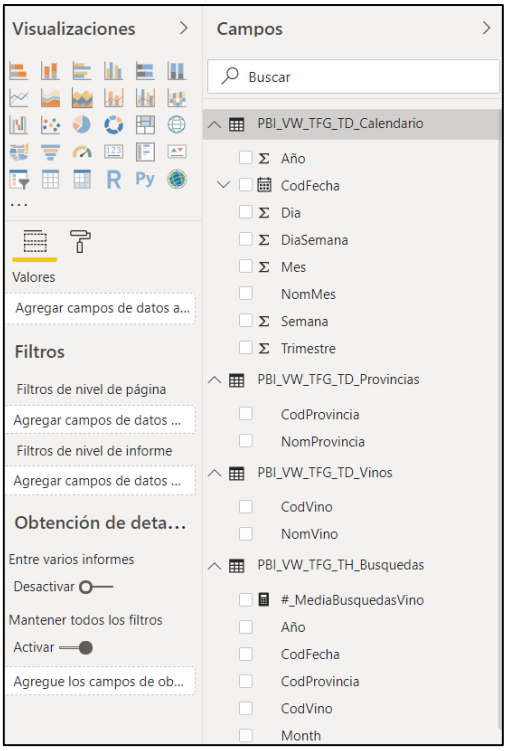


Ilustración 40: Columnas cargadas en Power BI desde las diferentes vistas (Elaboración propia, 2019)

Como también se puede observar en la imagen anterior, *Power BI* ofrece un cuadro de herramientas con una gran variedad de visualizaciones. Con estos objetos y los campos extraídos de las vistas se llevan a cabo los gráficos del cuadro de mando, los cuales permitirán al usuario final interactuar con la información.

En este trabajo, las visualizaciones que más se utilizan son los gráficos de barras, gráficos de tarta e indicadores, los cuales nos permiten analizar los datos de forma sencilla e intuitiva. Por otra parte, también será un recurrente los filtros desplegables, que permitirán filtrar el cuadro de mando al gusto del usuario.

En *Power BI* se pueden añadir varias pestañas o ventanas para cada cuadro de mando. En este trabajo se establecen dos pestañas diferentes: Análisis y Predicción. La pestaña de Análisis contiene toda la información relativa a la carga de todos los datos realizada en el proceso ETL. Por otra parte, la ventana de Predicción se explicará en el apartado Predicciones con Python.

La pestaña de Análisis, su diseño se compone por tres indicadores, tres gráficos de barras y un gráfico de tarta. Los filtros que se utilizan son los respectivos al Año, Mes, Provincia y Denominación de Origen.

Cada gráfico representa la siguiente información:

- Indicadores:
  - Número de D.O.: Proporciona información del número de Denominaciones de Origen que se visualizan en el cuadro de mando.
  - Número de Búsquedas: Proporciona como información la suma total de Número de Búsquedas de las Denominaciones de Origen que se analizan.
  - Promedio de Búsquedas: Proporciona el valor medio de búsquedas de las Denominaciones de Origen. La fórmula que se ha utilizado para su cálculo es: Total de búsquedas / Número de D.O.
- Gráficos de Barras:
  - Promedio de Búsquedas por Comunidad Autónoma: Representa gráficamente el número de búsquedas distribuido en diferentes barras por cada Comunidad Autónoma.
  - Número de Búsquedas por Comunidad Autónoma: Representa gráficamente el número totales de búsqueda distribuido en diferentes barras por cada Comunidad Autónoma.
  - Número de Búsquedas por Año: Representa gráficamente el número de búsquedas distribuido en diferentes barras por cada año.
- Gráficos de Tarta:
  - Número de búsquedas por Vino: Representa gráficamente en un gráfico de tarta las diferentes Denominaciones de Orígenes. El tamaño de la sección correspondiente a cada D.O. se rige por el número de búsquedas.

Después de expresar los datos gráficamente en *Power BI Desktop*, se debe subir el cuadro de mando al portal de *Power BI*, para que los usuarios finales puedan interactuar con él.

Por último, antes de subir el archivo al portal, *Power BI Desktop* nos ofrece la creación del cuadro de mando en versión móvil. Después de terminar gráficamente la composición de nuestras pestañas, se debe componer la parte de visualización móvil

con los filtros y gráficos necesarios para la misma. De este modo, el usuario final podrá acceder desde cualquier parte con la aplicación móvil con total comodidad.

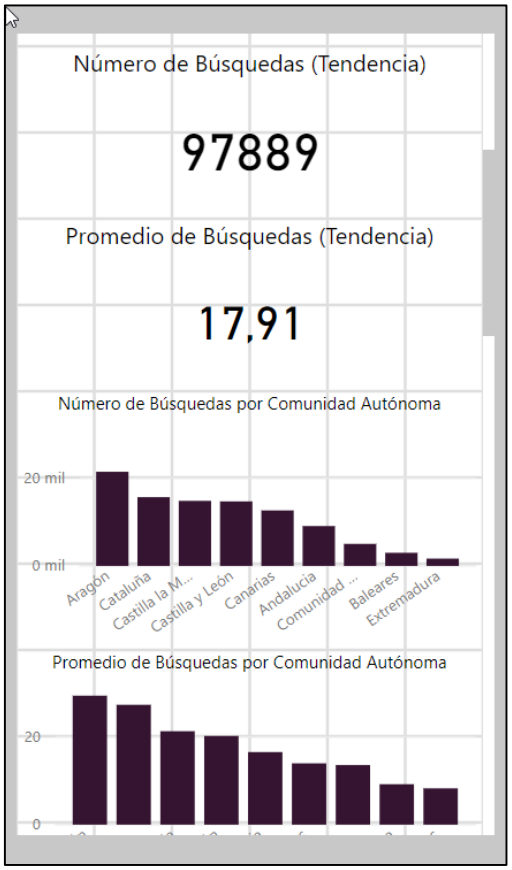


Ilustración 41: Versión móvil del cuadro de mando de Power BI (Elaboración propia, 2019)

8.6. Predicciones con Python

En este apartado se menciona el desarrollo realizado para detectar patrones y predecir datos con Python. Para llevar a cabo este punto, se ha realizado un estudio sobre series temporales y cómo predecir las búsquedas a partir de una red neuronal.

Antes de presentar el desarrollo, se explicarán algunos conceptos fundamentales para la comprensión de este apartado.

Las series temporales son conjuntos de muestras en intervalos de tiempo regulares. Analizar el comportamiento de una serie temporal nos puede ayudar a detectar patrones para realizar posteriormente pronósticos de cómo será su comportamiento futuro (Prónostico de series temporales en Python, 2019).

Las series de tiempo tienen dos características fundamentales:

- Son dependientes del tiempo. Todos los valores analizados son dependientes de una fecha o periodo de tiempo.
- Suelen tener algún tipo de estacionalidad, o de tendencias a crecer o decrecer. Cuando se analiza una serie temporales se pueden observar patrones que demuestran estas fluctuaciones en el tiempo.

Por tanto, si se observan los datos de la tabla TFG_TH_Busquedas con sus respectivas columnas (CodFecha, NumSearch, CodVino, CodProvincia), se puede deducir que se dispone para este TFG de una serie temporal. Esta tabla muestra el número de búsquedas en un periodo de tiempo desde enero del año 2004, hasta abril del año 2019, agrupando los meses por año.

Por otra parte, las redes neuronales, como su propio nombre indica proviene de la idea de imitar el funcionamiento de su homólogo biológico: un conjunto de neuronas conectadas entre si y que trabajan en grupo. Con la experiencia que crean estas neuronas, intentar reforzar y crear conexiones para aprender nuevas funcionalidades que se quedan fijas en el tejido (Julián, 2016).

Las redes neuronales artificiales se organizan en diferentes unidades de procesamiento interconectadas que simulan versiones abstractas de neuronas.

Estas capas están divididas en tres partes:

- Capa de entrada: Contiene las unidades de procesamiento que representan los datos de entrada.
- Capas ocultas: Donde el algoritmo realiza las funciones para aprender y predecir resultados.
- Capa de salida: Unidad o unidades que representan el campo o campo destinos.

Los datos de entrada se dirigen desde cada neurona hasta cada neurona de la capa siguiente. El resultado de todo este proceso se almacena en la capa de salida.

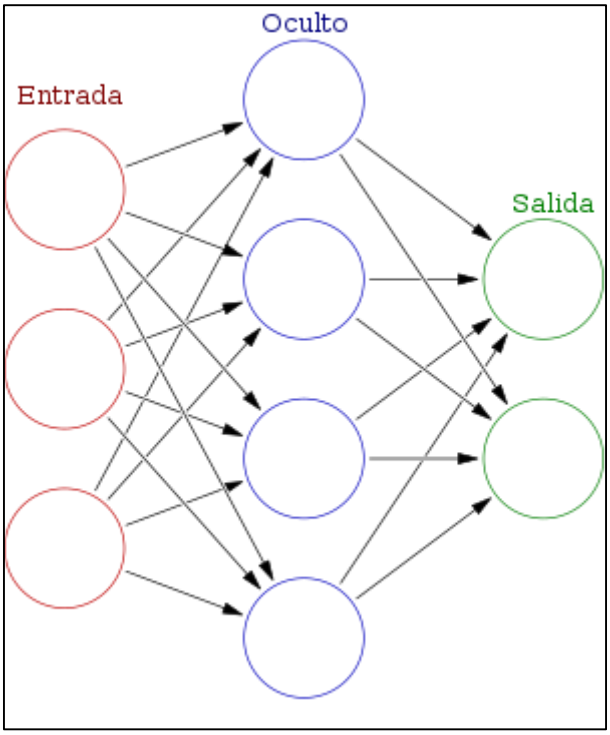


Ilustración 42: Capas de una red neuronal artificial (Red Neuronal Artificial, 2016)

La red neuronal aprende examinando los registros individuales, creando una predicción por cada uno de ellos y realizando ajustes según la ponderación. La ponderación viene

dada por el algoritmo, donde hay diferentes tipos que se pueden adaptar más o menos dependiendo del problema a solucionar.

Estas redes aprenden mediante el proceso de entrenamiento y test. Es decir, la red neuronal se alimenta de un conjunto de datos llamado entrenamiento, con los cuales realizará cálculos y emitirá resultados diferentes por cada etapa que realice. Las etapas son el número de veces que se analizarán los datos de entrada. Estos resultados se compararán con otro conjunto de datos llamado test, los cuales ya disponen de los resultados conocidos. La red neuronal intentará aprender automáticamente hasta parecerse en la mayor medida de lo posible a los datos conocidos.

Una vez que se han explicado los dos conceptos que se van a utilizar para la predicción de datos en *Python*, se pueden diferenciar cinco etapas para su programación:

- Preparación de los datos: Esta etapa ya se ha realizado en el desarrollo de nuestro *Data Warehouse*.
- Diseño del modelo
- Entrenamiento del modelo
- Evaluación del modelo
- Predicción

Como ya se ha explicado durante el punto 8 “Desarrollo del proyecto” todo lo correspondiente a la preparación de los datos, este apartado se basará en las cuatro etapas siguientes.

Para el desarrollo del modelo, entrenamiento, evaluación y predicción se ha realizado una investigación para comprender y aprender a programar predicciones mediante *Python*.

El modelo que se ha creado para este proyecto tomará los siete meses previos para predecir el siguiente. La función utilizada es *series_to_supervised()*, que permitirá ordenar las siete columnas de entrada y la columna de salida necesarias para la red neuronal. Estos valores se utilizarán para crear la misma y predecir los datos que se desean.

Asimismo, en varios artículos se ha podido leer que normalizar los datos entre los valores -1 y 1 produce un efecto positivo en los algoritmos de series temporales. Para ello, se utiliza la función MinMaxScaler que transformará los datos utilizados dentro de este rango.

```
# convert series to supervised learning
def series_to_supervised(data, n_in=1, n_out=1, dropnan=True):
    n_vars = 1 if type(data) is list else data.shape[1]
    df = pd.DataFrame(data)
    cols, names = list(), list()
    # input sequence (t-n, ... t-1)
    for i in range(n_in, 0, -1):
        cols.append(df.shift(i))
        names += [('var%d(t-%d)' % (j+1, i)) for j in range(n_vars)]
    # forecast sequence (t, t+1, ... t+n)
    for i in range(0, n_out):
        cols.append(df.shift(-i))
        if i == 0:
            names += [('var%d(t)' % (j+1)) for j in range(n_vars)]
        else:
            names += [('var%d(t+%d)' % (j+1, i)) for j in range(n_vars)]
    #
    agg = pd.concat(cols, axis=1)
    agg.columns = names
    # Borrar filas sin valores
    if dropnan:
        agg.dropna(inplace=True)
    return agg


# load dataset
values = df.values
# ensure all data is float
values = values.astype('float32')
# normalize features
scaler = MinMaxScaler(feature_range=(-1, 1))
values=values.reshape(-1, 1) # esto lo hacemos porque tenemos 1 sola dimension
scaled = scaler.fit_transform(values)
# frame as supervised learning
reframed = series_to_supervised(scaled, PASOS, 1)
reframed.head()
```

Ilustración 43: Función series_to_supervised() (Elaboracion propia, 2019)

Una vez se dispone de los datos obtenidos con la función *series_to_supervised(),* se guardan dentro de una variable para subdividirlos posteriormente en una muestra para el entrenamiento y otra para el test. Como se ha explicado en este mismo apartado, la red neuronal aprenderá mediante este proceso.

La red neuronal que se ha programado incluye siete entradas (las columnas obtenidas por la función *series_to_supervised()*), otras siete neuronas en el nivel de capa oculta y una sola salida.

Por otra parte, cuando se crea el modelo de red neuronal, se debe elegir la función de activación, el optimizador y la métrica de pérdida. Estos conceptos tienen el siguiente significado:

- Función de activación: Esta función devuelve una salida a partir de una entrada. Como se disponen valores entre -1 y 1, la tangente hiperbólica es nuestra elección.
- Optimizador: El optimizador elegido permitirá al algoritmo calcular las ponderaciones de las neuronas, basándose en los datos de entrada y la métrica de pérdida.

- Métrica de pérdida: Valor que permite al algoritmo evaluar sus interacciones para guiar en el ajuste de los parámetros.

Como los valores manipulados en este trabajo pueden variar mucho, tanto con incrementos como decrementos, el optimizador Adam y la métrica de pérdida *Mean Absolute Error* serán los utilizados, ya que trabajan bien con valores dispersos.

Para diferenciar si el modelo dispone de unos buenos resultados, el *acuracy* utilizado será el *Mean Squared Error*. Este valor proporcionará información suficiente para saber si el modelo está prediciendo correctamente los datos o, por el contrario, no se asemeja el entrenamiento con el test. Este campo debe disminuir en las diferentes interacciones del algoritmo.

Cuando el algoritmo disponga de un *Mean Squared Error* bastante óptimo, se aplicará la función *predict()* de forma que se consiga predecir los próximos doce meses a partir de sus siete anteriores. Estos resultados se almacenarán en una hoja de Excel y cargados en nueva Tabla de Hechos, TFG_TH_Predicciones.

```
def crear_modeloFF():
    model = Sequential()
    model.add(Dense(PASOS, input_shape=(1,PASOS),activation='tanh'))
    model.add(Flatten())
    model.add(Dense(1, activation='tanh'))
    model.compile(loss='mean_absolute_error',optimizer='Adam',metrics=["mape"])
    model.summary()
    return model
```

Ilustración 44: Modelo de Red Neuronal (Elaboración propia, 2019)

```
def agregarNuevoValor(x_test,nuevoValor):
    for i in range(x_test.shape[2]-1):
        x_test[0][0][i] = x_test[0][0][i+1]
    x_test[0][0][x_test.shape[2]-1]=nuevoValor
    return x_test

results=[]
for i in range(12):
    parcial=model.predict(x_test)
    results.append(parcial[0])
    print(x_test)
    x_test=agregarNuevoValor(x_test,parcial[0])
```

Ilustración 45: Predicción de los próximos 12 meses (Elaboración propia, 2019)

En un futuro, se podrán realizar diferentes predicciones modificando la fuente de activación, el optimizador, la métrica de pérdida y el uso de las capas. Por esta razón, se deja abierto a un posterior estudio de estas características del modelo, ya que serán fundamentales para la formación personal y laboral.

Por último, se debe mencionar que en este punto se ha realizado la búsqueda de patrones y sus posteriores predicciones. Así pues, se ha aplicado tanto *Machine Learning* para predecir los resultados, como *Data Mining* para encontrar patrones. Se

debe recordar que el aprendizaje automático puede incluir la minería de datos en su realización.

9. Análisis de los resultados

En este apartado se comentarán los datos analizados y se compararán con los datos que se han podido obtener de diferentes fuentes. Este análisis estará dividido en dos puntos: Análisis de los datos obtenidos en *Google Trends* y Análisis de las predicciones.

Se debe recordar que este análisis se realiza con 30 Denominaciones de Origen. Como es una muestra relativamente pequeña, algunos datos pueden diferir con las fuentes de las cuales se ha extraído información relevante a la historia del consumo vitivinícola.

9.1. Análisis de los datos obtenidos en *Google Trends*

En diferentes webs y artículos, se ha podido comprobar que el consumo de vino en España ha sufrido una caída desde los años 80 de más del 50%. En la década actual ha conseguido un ligero aumento, sin embargo, no ha vuelto a alcanzar el consumo de las décadas anteriores.

Concretamente, el Observatorio Español del Mercado del Vino (OEMV), analizó el consumo del vino y confirmó la reducción de este de hasta un 67,3 % en tan solo tres décadas (1987-2017).

Por otra parte, en enero del 2019, el OEMV realizó un balance de la evolución de los últimos 10 años del vino en España. En esta demostración aseguran que a partir del 2008 se detuvo el descenso del consumo de vino en el país en cuestión.

Relacionando los datos del Observatorio Español del Mercado del Vino y otros artículos que se han investigado, se puede deducir que no solo no ha descendido el consumo del vino desde el año 2008, sino que ha sufrido un ligero ascenso en esta última década.

Esta información que se explica en los diferentes artículos se puede comparar con los datos representados en la primera pestaña de *Power BI*, Análisis, concretamente con el gráfico Número de Búsquedas por Año/Mes.

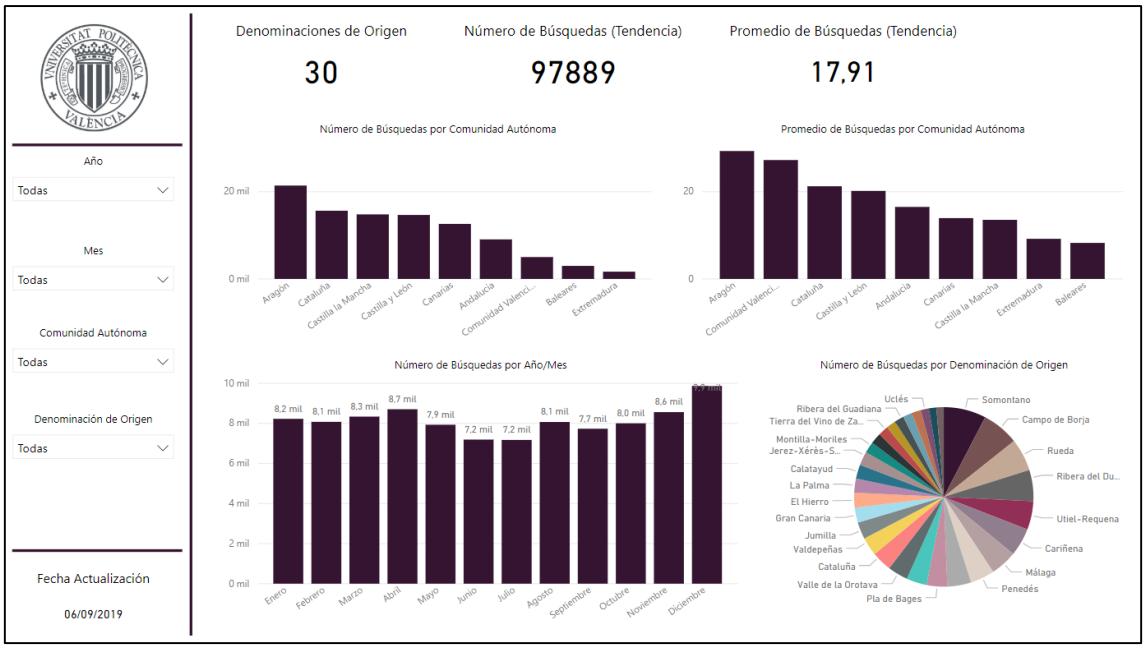


Ilustración 46: Pestaña análisis del cuadro de mando de Power BI (Elaboración propia, 2019)

En este gráfico se puede observar un descenso de búsquedas hasta el año 2010, asociado al menor consumo de vinos en España. En los años 2008, 2009 y 2010 se puede ver que el descenso de búsquedas es mucho menor si se comparar con los años anteriores.

Si se asocian estos datos con los analizados por el OEMV, se puede deducir que la información representada en el cuadro de mando es fiable. Con este gráfico se puede demostrar que a partir del 2008 hay un cambio en el sector vitivinícola, donde el descenso de las décadas anteriores se estabiliza y comienza a sufrir un ligero aumento en la década actual.

En los diferentes artículos se ha podido leer que el consumo del vino en España ha ascendido en los últimos años, aunque de forma muy ligera, sin alcanzar las décadas anteriores. En PBI, se puede afirmar que en los últimos años ha aumentado la tendencia de búsquedas de los vinos, exceptuando 2018, que sufre un ligero retroceso. Este último descenso se puede justificar que para realizar este análisis no se dispone de todos los datos del sector vitivinícola español, solo una pequeña muestra con la tendencia de búsquedas de 30 Denominaciones de Origen. Es decir, como se ha comentado al inicio de este apartado, algunos datos pueden diferir al no poder analizarse un conjunto tan grande como el de la OEMV u otros estudios estadísticos.

También se puede afianzar, aunque solo con la década anterior, que aunque los últimos 10 años contengan ligeros ascensos, continua muy lejos de la tendencia de búsqueda de los años anteriores, concretamente en 2004 y 2005.

En el gráfico Número de Búsquedas por Año/Mes se puede reducir su nivel de información y observar los meses con más tendencia de búsquedas.

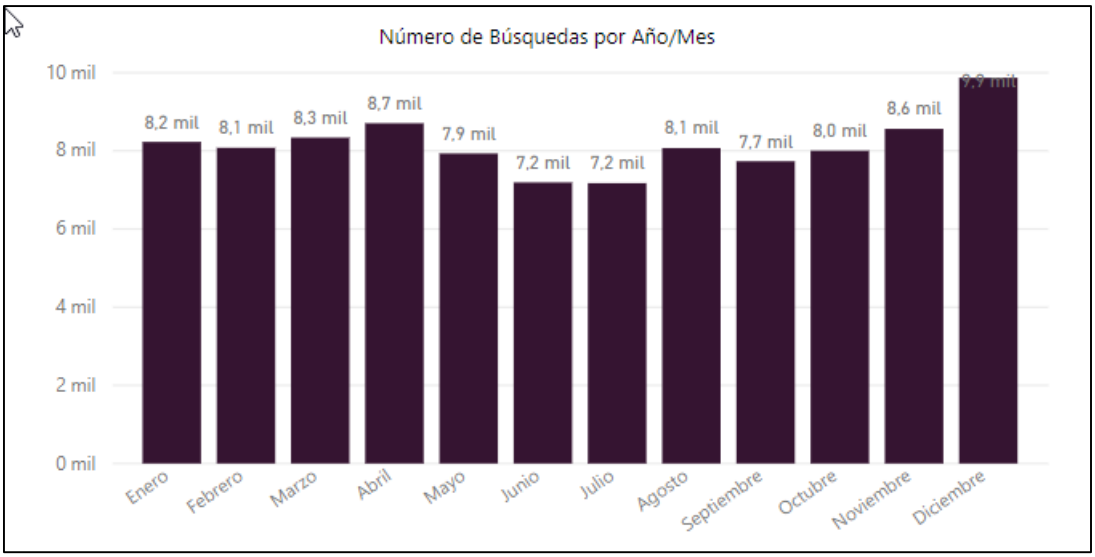


Ilustración 47: Gráfico de barras con el Número de Búsquedas por Año/Mes (Elaboración propia, 2019)

Si se organiza por meses, abril, noviembre y diciembre son los meses donde se realizan más búsquedas de vinos.

La tendencia de búsquedas del primer mes, abril, se puede relacionar con la época de comuniones. Abril y mayo son los meses con más homenajes de este tipo y donde, por motivo de celebración, se consume vino. Es un dato coherente que durante este mes aumente la tendencia de búsqueda para realizar los preparativos del mes en cuestión y su predecesor.

Haciendo referencia a noviembre y diciembre, también es lógico que el número de búsquedas aumenten en estos meses. En diciembre se celebra la Navidad, festividad donde en muchos hogares aumenta el consumo de vino para celebrar las reuniones familiares y de amigos.

Otros datos que se pueden analizar en esta pestaña son las Denominaciones de Origen con más tendencia de búsquedas o las Comunidades Autónomas que las reciben a causa de sus D.O. correspondientes.

Por ejemplo, las tres comunidades que disponen de más número de tendencia de búsquedas son Aragón, Cataluña y Castilla la Mancha. Sin embargo, si después se calcula el promedio de búsquedas (Tendencia de búsqueda/Número de D.O) se puede ver que este orden cambia y son Aragón, Comunidad Valenciana y Cataluña las que encabezan este gráfico.

Por último, de las Denominaciones de Origen que encabezan las búsquedas son Somontano, Campo de Borja, Rueda, Ribera del Duero, Utiel-Requena, Cariñena, Málaga y Penedés.

Estas D.O. se analizan con Python para predecir sus próximos 12 meses representando sus resultados en *Power BI*.

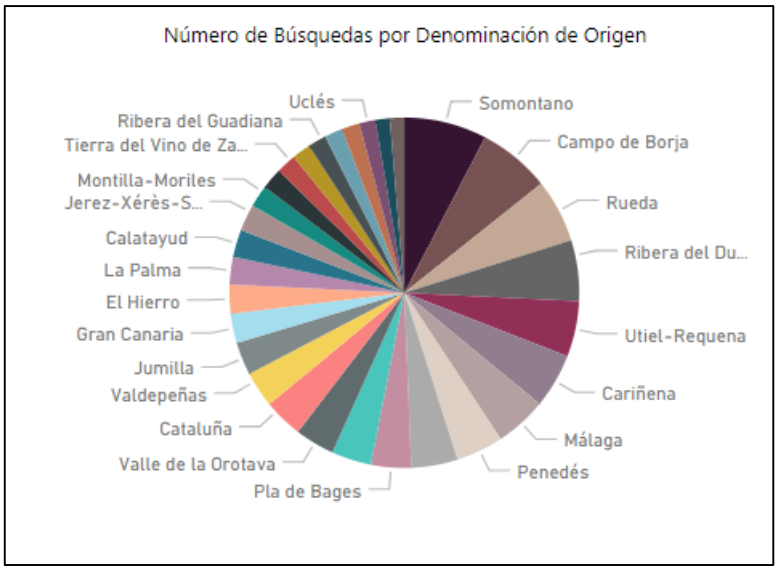


Ilustración 48: Gráfico de barras con el Número de Búsquedas por D.O. (Elaboración propia, 2019)

Si se relacionan las Denominaciones de Origen y las Comunidades Autónomas, se puede demostrar por qué Aragón, Comunidad Valenciana y Cataluña son las que encabezan la lista.

Si se filtra el gráfico por Aragón, se ve que se analizan cuatro Denominaciones de Origen con esta comunidad (Calatayud, Campo de Borja, Somontano y Cariñena), de los cuáles tres son de las más buscadas. Por otra parte, la Comunidad Valenciana y Cataluña también poseen D.O. con tendencias de búsquedas altas, como son Utiel-Requena o Penedés.

Por último, se puede ver que Comunidades Autónomas, como Castilla y León y Castilla la Mancha, salen desfavorecidas en el promedio de búsquedas respecto a la Comunidad Valenciana, aunque las dos contienen número mayor de búsquedas que la última. Esto es a causa de que, en este proyecto, el promedio de búsquedas se rige por el número de Denominaciones de Origen analizadas por comunidad. Como las dos C.A mencionadas poseen un número mayor de D.O. que la Comunidad Valenciana, su promedio desciende desfavorablemente, aunque sus búsquedas sean mayores.

9.2. Análisis de las Predicciones

Las predicciones realizadas con *Python* se han calculado a partir de las ocho Denominaciones de Origen con más tendencia de búsqueda de la parte de análisis. Estas Denominaciones de Origen son las siguientes:

- Somontano
- Campo de Borja
- Rueda
- Ribera del Duero
- Cariñena
- Utiel-Requena
- Málaga

- Penedés

La predicción que se ha realizado en este apartado se centra en conseguir los valores para los próximos 12 meses después de abril del 2019. Para ello, los valores que influyen en esta predicción son los 12 meses anteriores que contienen un resultado de tendencia de búsquedas.

Cuando se predice un mes, por ejemplo, noviembre del 2019, se utilizan los meses anteriores que ya han sido calculados por el algoritmo.

Si se observa la pestaña de predicción, ahora se puede visualizar el año 2019 completo y parte del 2020. En 2019 hay un pequeño descenso si se compara con 2018, aunque sigue con la misma estabilidad de los últimos años. Esta pequeña bajada de tendencia de búsquedas seguramente sea calculada por el algoritmo a causa de que 2018 también tuvo una pequeña disminución respecto al año 2017.

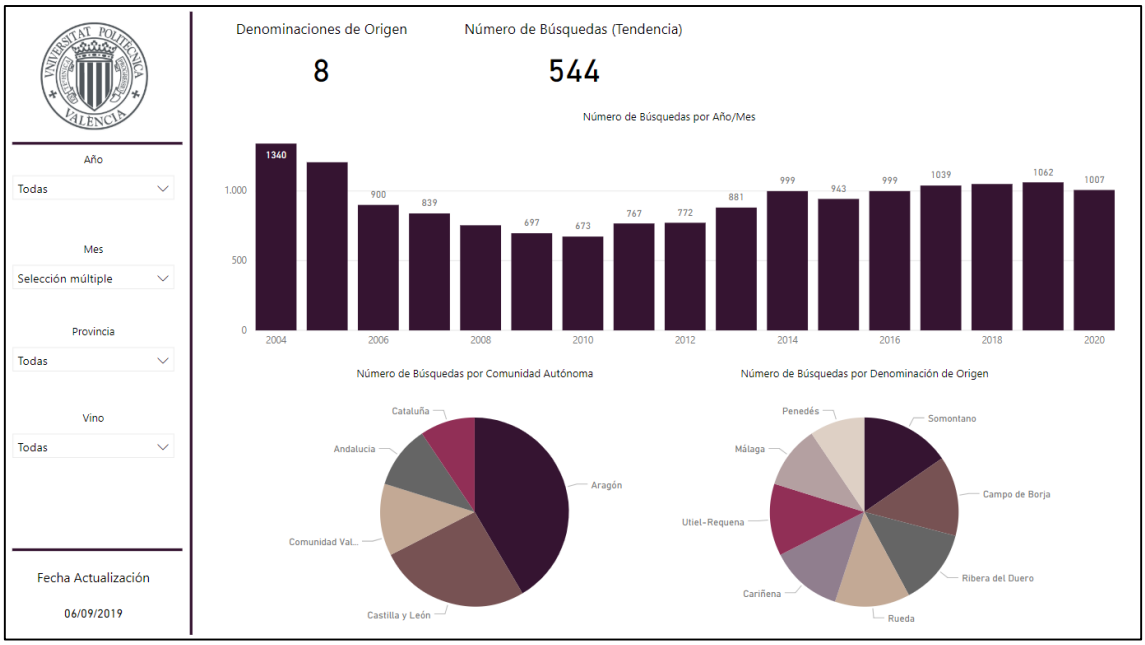


Ilustración 49: Pestaña de Predicción del cuadro de mando de Power BI (Elaboración propia, 2019)

En cuanto a 2020, en sus primeros 4 meses, tiene una tendencia de búsquedas bastante alta, 1000 búsquedas. Si estos valores continúan siendo semejantes, podría verse un pequeño ascenso durante este año.

Sin embargo, si la comparación se realiza con enero, febrero, marzo y abril del año 2020 y del año 2019, la tendencia de búsqueda resulta ser menor. Por tanto, la red neuronal pronostica otro pequeño descenso para el año 2020.

Se debe recordar que la muestra analizada no dispone de toda la información vitivinícola de España y, como consecuencia, podría haber creado una ligera disminución en el año 2018, la cual también puede estar propiciando que el algoritmo creado genere pequeños descensos de búsquedas para cada año.

A pesar de esto, con este algoritmo y su representación gráfica en *Power BI*, el algoritmo proporciona tendencias de búsquedas dentro de unos valores coherentes, los cuales se mantienen dentro de una estabilidad desde el año 2016. En versiones futuras se podría realizar una comparación de los resultados del algoritmo con todo el año 2019, para confirmar que el modelo es óptimo.

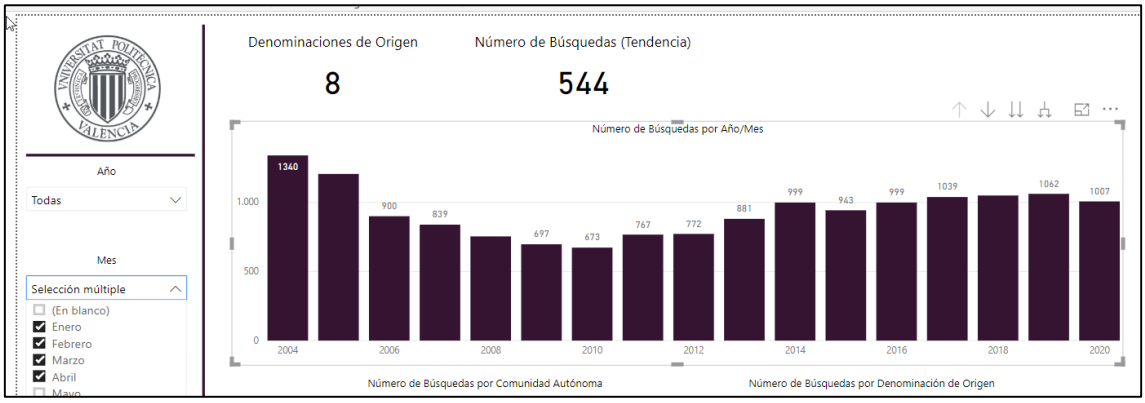


Ilustración 50: Gráfico de barras con el Número de Búsquedas por Año/Mes (Elaboración propia, 2019)

Por último y relacionándolo con los artículos buscados, según nuestros datos, el consumo de vino en España continuará con la misma tendencia que los últimos cuatro años.

10. Conclusión

Con el análisis realizado se puede concluir que la tendencia de búsquedas de las Denominaciones de Origen está estrechamente ligada con el consumo de vino español.

Como se ha podido observar durante los dos análisis realizados, los años en los que el consumo de vinos ha sido menor en España coincide con los años con menos tendencia de búsqueda en Google. De la misma manera, los años donde se han generado incrementos de consumo, las búsquedas también han aumentado, exceptuando casos eventuales, los cuales han podido generarse porque la muestra estudiada no contiene toda la información vitivinícola del país.

Dejando a un lado el análisis sobre las Denominaciones de Origen, con este Trabajo Final de Grado se puede demostrar que todo el proceso analítico y de predicción llevado a cabo podría servir para diferentes ámbitos del mundo empresarial. Por ejemplo, la realización de predicciones climatológicas para las estaciones de esquís, para saber cuándo sería conveniente abrir o cerrar respecto a años anteriores.

Otro ejemplo podría ser la predicción de ventas de una empresa, donde el sistema y el algoritmo creado en este trabajo podría ayudar a disponer del stock necesario y no quedarse sin reservas o, por el contrario, llegar a tener una rotura de stock.

Respecto a mi opinión sobre el proyecto realizado, *Google Trends* es una herramienta fiable para realizar un análisis de datos y predicción gracias a la información que nos proporciona a partir de la minería de texto. Este mismo análisis, transportado a otros negocios, podría interesar a empresas que quisieran conocer el estado del mercado de los productos que comercializan.

Asimismo, este Trabajo Final de Grado podría ampliarse con varias características, como puede ser con el aumento de *keywords* analizadas, creando un modelo con cientos o miles de palabras claves que pueda compararse con otras tipologías de expresiones de búsqueda y aumentar las conclusiones ya obtenidas. También se podría ampliar el detalle, añadiendo provincias o bodegas para un análisis más exhaustivo.

En definitiva, todo el proceso de análisis realizado, desde la extracción, pasando por la transformación y llegando hasta la visualización y predicción de los datos, se podría dar forma como servicio y comercializarlo en las empresas o compañías interesadas. Invirtiendo en este servicio, la empresa dispondría de una importante ayuda para sus análisis y tomas de decisiones, conllevando así a un mayor beneficio económico.

11. Bibliografía

¿Qué es OLAP? (2011). Obtenido de Business Intelligence fácil: https://www.businessintelligence.info/definiciones/que-es-olap.html

21 base de datos más utilizadas por los desarrolladores. (2019). Obtenido de Diarlu: https://www.diarlu.com/gestores-bases-datos/

Anaconda. (2012). Obtenido de Anaconda: https://www.anaconda.com/distribution/

Breve Historia del Business Intelligence: Origen y Evolución. (2017). Obtenido de Time Manager: https://www.timemanagerweb.com/2017/01/31/breve-historia-del-business-intelligence/

Carisio, E. (2018). *Herramientas ETL: comparativa y principales categorías*. Obtenido de Mediacloud: https://blog.mdcloud.es/herramientas-etl-comparativa-y-principales-categorias/

Cebotarean, E. (2011). Business intelligence. *Journal of Knowlede Management, Economics and Information Technology*, pág. 2.

Chambi, J. (2016). *Machine Learning y Data Mining*. Obtenido de Perú Analítica.

Chang, J. (2018). *¿Qué es la minería de texto, cómo funciona y por qué es útil?* Obtenido de Universo abierto: https://universoabierto.org/2018/02/22/que-es-la-mineria-de-textos-como-funciona-y-por-que-es-util/

Concepto de Data Warehouse. (2018). Obtenido de NeoAttack: https://neoattack.com/neowiki/data-warehouse/

Conexiónesan. (2018). Obtenido de ¿Cuáles son las diferencias entre la minería de datos y el Business Intelligence: https://www.esan.edu.pe/apuntes-empresariales/2018/10/cuales-son-las-diferencias-entre-la-mineria-de-datos-y-el-business-intelligence/

Correa, D. (2012). *Claves para ser más productivo con Python*. Obtenido de Maestros del Web: http://www.maestrosdelweb.com/claves-para-ser-mas-productivo-con-python/

El OEMV hace balance de los últimos 10 años de evolución del vino en el mundo y en España. (2019). Obtenido de Vinetur: https://www.vinetur.com/2019012349147/el-oemv-hace-balance-de-los-ultimos-10-anos-de-evolucion-del-vino-en-el-mundo-y-en-espana.html

Fernández, C. G. (2017). *ANÁLISIS ESTADÍSTICO Y FINANCIERO DEL SECTOR*.

Google Trends. (2006). Obtenido de Google: https://trends.google.com/trends/?geo=US

Herramientas ETL: comparativa y principales categorías. (s.f.).

IBM Cognos. (2014). Obtenido de IBM: https://www.ibm.com/es-es/products/cognos-analytics

Iverson, K. E. (1962). A programming Language. En K. E. Iverson, *A programming Language* (pág. 256).

Julián, G. (2016). *Las redes neuronales: qué son y por qué están volviendo*. Obtenido de Xataka: https://www.xataka.com/robotica-e-ia/las-redes-neuronales-que-son-y-por-que-estan-volviendo

Las 10 herramientas de Business Intelligence que deberías conocer. (2018). Obtenido de Ticbeat: https://www.ticbeat.com/tecnologias/las-10-herramientas-de-business-intelligence-que-deberias-conocer/

Logotipo de Tableau. (2017). Obtenido de A Medium Corporation: https://medium.com/@tylerkeamogarrett/tableau-logo-f94ce90e65ad

Luhn, H. (1958). A Business Intelligence System. *IBM Journal*.

Marín, R. (2019). *Los gestores de bases de datos (SGBD) más usados*. Obtenido de Revista Digital INESEM : https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/

Mayorga Muñoz, L. (2019). *Data Mining vs Machine Learning: ¿Cuás es la diferencia?* Obtenido de El Periódico de Aragón: https://www.elperiodicodearagon.com/noticias/mas-voces/data-mining-vs-machine-learning-cual-es-diferencia_1364595.html

Power BI. (2014). Obtenido de Microsoft: https://powerbi.microsoft.com/es-es/

Prado, E. P. (2017). *¿Qué es y por qué aprender SQL?* Obtenido de DevCode.

Principales categorías de herramientas ETL. (2018). Obtenido de Bigeek: https://blog.bi-geek.com/4-tipos-herramientas-etl/

Prónostico de series temporales en Python. (2019). Obtenido de Aprende Machine Learning: https://www.aprendemachinelearning.com/pronostico-de-series-temporales-con-redes-neuronales-en-python/

QlikView. (1993). Obtenido de Qlik: https://www.qlik.com/es-es

R logo. (2016). Obtenido de Wikimedia: https://commons.wikimedia.org/wiki/File:R_logo.svg

Radiografía del sector del vino en España. (2018). Obtenido de Solunion: https://www.solunion.es/blog/radiografia-del-sector-del-vino-en-espana/

Red Neuronal Artificial. (2016). Obtenido de Wikipedia: https://es.wikipedia.org/wiki/Red_neuronal_artificial

Rochina, P. (2016). *Python vs R para el análisis de datos*. Obtenido de Revista Digital: https://revistadigital.inesem.es/informatica-y-tics/python-r-analisis-datos/

Tresquatreicinc. (2017). *De patrones y series*. Obtenido de Diez dedos en mis manos: http://diezdedosenmismanosmatematicas.blogspot.com/2017/11/de-patrones-y-series.html#