¿Son ecológicos los turistas europeos?

Estudio de la consideración medioambiental en sus decisiones vacacionales

TRABAJO FIN DE MÁSTER UNIVERSITARIOS EN GESTIÓN DE EMPRESAS, PRODUCTOS Y SERVICIOS

ALUMNO: IGNACIO VICENTE COSTA NAVARRO

TUTOR: ÁNGEL PEIRÓ SIGNES

CURSO ACADÉMICO: 2019 - 2020

Abstract

There are many decisions that a tourist must make when preparing their vacations (choice of hotel, means of transport, restaurants to eat and dine, activities to do …) and there are many effects that these decisions cause about the place you visit. Every day, more people are looking for rural and / or natural destinations in which to enjoy their vacations in a calm and relaxed way, leaving behind the psychological stress of overcrowded cities to give way to sustainable activities that are committed to preservation and appreciation of the environment that welcomes them. This paper aims to analyze whether this supposed tourist fashion is related to a series of decisions, prior to the trip and those of European citizens, which are sustainable and respectful of the environment. For this, the results of the "Flash Eurobarometer 432 (Preferences of Europeans towards Tourism, January 2016)" questionnaire are used, which is carried out in 33 different European countries. In addition, it is intended to extract behavior patterns to know if age, sex, place of residence or working status have a noticeable impact on the preferences of these tourists. Finally, it is intended to create a Machine Learning model that allows predicting the ecological level of tourists. This study allows future analysis to know the level of perception that tourist destinations have about sustainable tourism and, in this way, to be able to carry out actions that promote it in order to improve the experience of its visitors.

Resumen

Son muchas las decisiones que un turista debe tomar a la hora de preparar sus vacaciones (elección del hotel, del medio de transporte, de los restaurantes donde comer y cenar, de las actividades que realizar, etc.) y son muchos los efectos que éstas causan sobre el lugar que se visita. Cada día, son más las personas que buscan destinos rurales y/o naturales en los que disfrutar de sus vacaciones de forma tranquila y relajada, dejando atrás el estrés psicológico de las ciudades superpobladas para dar paso a actividades sostenibles que apuestan por la preservación y la apreciación del medio que las acoge. En este trabajo se pretende analizar si dicha supuesta moda turística, tiene relación con una serie de decisiones, previas al viaje y propias de ciudadanos europeos, que sean sostenibles y respetuosas con el medio ambiente. Para ello, se utilizan los resultados del cuestionario "Flash Eurobarometer 432 (Preferences of Europeans towards Tourism, January 2016)", el cual se lleva a cabo en 33 países europeos. Además, se pretenden extraer patrones de comportamiento para conocer si la edad, el sexo, el lugar de residencia o la condición laboral tienen un impacto notorio en las preferencias de estos turistas. Por último, se pretende crear un modelo de Machine Learning que permita predecir el nivel ecológico de los turistas. Dicho estudio permite futuros análisis para conocer el nivel de percepción que los destinos turísticos tienen acerca del turismo sostenible y, de esta forma, poder realizar acciones que lo promuevan con el fin de mejorar la experiencia de sus visitantes.

Índice

Índice de figuras

Índice de tablas

1. Introducción

La investigación que se plantea en este proyecto se centra principalmente en conocer el nivel de consideración medioambiental que tienen los europeos en el momento de preparar sus vacaciones y realizar actividades relacionadas con el turismo.

Esta investigación se plantea debido al reciente auge durante los últimos años de la conservación y protección del medio ambiente, y en la actualidad con motivo de la continua preocupación por los altos niveles de contaminación. Si a esto le sumamos un estilo de vida mucho más saludable, en el que la práctica de deporte y el consumo de alimentos naturales tienen un peso considerable, una masificación urbana en las ciudades y una dependencia total de la tecnología, se tiene como resultado una tendencia en el sector turístico que lleva a las personas a visitar destinos rurales, en los que disfrutar de unas vacaciones de forma tranquila y relajada, desconectando del trabajo y tareas diarias.

Este tipo de turismo, conocido como "turismo sostenible" o "ecoturismo", se define como un enfoque cuyo objetivo es proteger el medio ambiente y la cultura de las comunidades que albergan a los turistas, así como satisfacer las necesidades de los turistas y mantener el crecimiento de la industria del turismo (Yilmaz et al., 2019). Se entiende que todas aquellas actividades, tanto las de desplazamiento al destino como las propias realizadas en el mismo, que cumplan con dicho desarrollo sostenible, entrarían dentro de la definición.

Si bien nos paramos a pensar con detenimiento, son múltiples las medidas sostenibles que han sido establecidas, y que a nivel personal se llevan a cabo a diario, con el fin de respetar el medio que nos rodea. Cabe la posibilidad pues, de que en el sector turístico exista también un cambio de comportamiento, en el que los turistas tomen como habitual el desarrollo de actividades sostenibles durante sus períodos vacacionales.

Sin embargo, existe la posibilidad también de que esta tendencia turística sea simplemente una moda, lo cual lleve a los turistas a desarrollar actividades sostenibles por el mero hecho de que otras personas ya lo hagan; o incluso a querer aparentar que están realizando este tipo de actividades cuando en realidad no lo están haciendo.

Por tanto, se puede decir que el objetivo concreto de la investigación, es averiguar si realmente los turistas europeos son ecológicos durante sus estancias vacacionales. Muchos de ellos posiblemente lo sean en su día a día, pero, es importante conocer si también lo son como turistas en su desplazamiento, en su estancia y en sus actividades en el destino vacacional. De esta manera, podremos saber su nivel de consideración medioambiental con respecto al turismo, y seremos capaces también de ver si estamos ante el principio de un cambio en el comportamiento de los turistas.

Este conocimiento servirá como punto de partida para, posteriormente en futuros trabajos, poder establecer e implantar medidas turísticas sostenibles, tanto a nivel público como privado, que se adecuen a aquello que realmente buscan los turistas europeos con el fin de satisfacer sus necesidades durante sus períodos vacacionales. Del mismo modo, también se tendrá información para desarrollar estrategias en los destinos vacacionales, que sirvan para mejorar el nivel de concienciación de los turistas acerca del turismo sostenible.

2. Estudio del estado del arte

Para conocer el estado del arte de la situación del turismo ecológico y sostenible a nivel mundial y europeo, se ha realizado una búsqueda de literatura sobre el tema. Cabe destacar que, aunque a nivel europeo no se ha podido extraer demasiada información, existe mucha documentación de destinos vacacionales visitados por turistas estadounidenses. Si bien no pueden extrapolarse los comportamientos a los turistas europeos, sirve como punto de partida para conocer las tendencias en diferentes regiones.

Los criterios de inclusión para la búsqueda bibliográfica han sido:

- Estar publicado en SCOPUS.
- En inglés y castellano.
- En los últimos 10 años.
- Áreas científicas de medio ambiente, naturaleza, empresa y sociedad.

En la tabla siguiente se resume las búsquedas realizadas en SCOPUS y los resultados encontrados.

Tabla 1. Resumen búsquedas SCOPUS

Id	Estrategia de búsqueda	Items
W1	TITLE-ABS-KEY (tourism AND destination AND "sustainable practices") AND (LIMIT-TO (PUBYEAR , 2019) OR LIMIT-TO (PUBYEAR , 2018) OR LIMIT-TO (PUBYEAR , 2017) OR LIMIT-TO (PUBYEAR , 2016) OR LIMIT-TO (PUBYEAR , 2015) OR LIMIT-TO (PUBYEAR , 2014) OR LIMIT-TO (PUBYEAR , 2012) OR LIMIT-TO (PUBYEAR , 2011) OR LIMIT-TO (PUBYEAR , 2010) OR LIMIT-TO (PUBYEAR , 2009)) AND (LIMIT-TO (SUBJAREA , "BUSI") OR LIMIT-TO (SUBJAREA , "ENVI") OR LIMIT-TO (SUBJAREA , "SOCI") OR LIMIT-TO (SUBJAREA , "ENER"))	15
W2	TITLE-ABS-KEY (hotel AND "environmentally-friendly") AND (LIMIT-TO (PUBYEAR , 2019) OR LIMIT-TO (PUBYEAR , 2018) OR LIMIT-TO (PUBYEAR , 2017) OR LIMIT-TO (PUBYEAR , 2016) OR LIMIT-TO (PUBYEAR , 2015) OR LIMIT-TO (PUBYEAR , 2014) OR LIMIT-TO (PUBYEAR , 2013) OR LIMIT-TO (PUBYEAR , 2012) OR LIMIT-TO (PUBYEAR , 2011) OR LIMIT-TO (PUBYEAR , 2010) OR LIMIT-TO (PUBYEAR, 2009)) AND (LIMIT-TO (SUBJAREA , "BUSI") OR LIMIT-TO (SUBJAREA , "ENVI") OR LIMIT-TO (SUBJAREA , "SOCI") OR LIMIT-TO (SUBJAREA , "ENER"))	45
W3	TITLE-ABS-KEY (tourism AND transport AND "low impact") AND (LIMIT-TO (PUBYEAR , 2019) OR LIMIT-TO (PUBYEAR , 2018) OR LIMIT-TO (PUBYEAR , 2017) OR LIMIT-TO (PUBYEAR , 2016) OR LIMIT-TO (PUBYEAR , 2015) OR LIMIT-TO (PUBYEAR , 2014) OR LIMIT-TO (PUBYEAR , 2013) OR LIMIT-TO (PUBYEAR , 2012) OR LIMIT-TO (PUBYEAR , 2011) OR LIMIT-TO (PUBYEAR , 2010) OR LIMIT-TO (PUBYEAR, 2009)) AND (LIMIT-TO (SUBJAREA , "BUSI") OR LIMIT-TO (SUBJAREA , "ENVI") OR LIMIT-TO (SUBJAREA , "SOCI") OR LIMIT-TO (SUBJAREA , "ENER"))	3

Id	Estrategia de búsqueda	Items
W4	TITLE-ABS-KEY (tourism AND destination AND eco-label) AND (LIMIT-TO (PUBYEAR , 2019) OR LIMIT-TO (PUBYEAR , 2018) OR LIMIT-TO (PUBYEAR , 2017) OR LIMIT-TO (PUBYEAR , 2016) OR LIMIT-TO (PUBYEAR , 2015) OR LIMIT-TO (PUBYEAR , 2014) OR LIMIT-TO (PUBYEAR , 2013) OR LIMIT-TO (PUBYEAR , 2012) OR LIMIT-TO (PUBYEAR , 2011) OR LIMIT-TO (PUBYEAR , 2010) OR LIMIT-TO (PUBYEAR, 2009)) AND (LIMIT-TO (SUBJAREA , "BUSI") OR LIMIT-TO (SUBJAREA , "ENVI") OR LIMIT-TO (SUBJAREA , "SOCI") OR LIMIT-TO (SUBJAREA , "ENER"))	18

Un total de 81 documentos (artículos, extractos de revistas científicas e investigaciones) han sido obtenidos en las 4 búsquedas. Tras la lectura de los resúmenes y de algunas partes de los documentos, se descartan algunos de ellos por desencaje en el trabajo de investigación o por similitud entre los mismos. De esta manera, se reduce la cantidad de documentos a 17.

3. Descripción del contexto

Para empezar a realizar este trabajo de investigación, primero se deben entender los impactos y aportaciones del turismo a la sociedad, a la economía y al medio ambiente. Centrando el foco en los turistas ecológicos, se debe conocer también que factores definen al turismo sostenible y de qué manera la industria de servicios gestiona dicha sostenibilidad, con el fin de satisfacer las necesidades de los turistas, además de comprender que consecuencias tienen las certificaciones medio ambientales en negocios y servicios públicos. Por último, se identifican, según otros autores, cuáles son los factores de decisión en la elección de vacaciones sostenibles, cuáles son las actitudes de los turistas sostenibles y qué variables afectan a dicho turismo.

3.1. Turismo y medio ambiente

- **Turismo internacional**

El turismo puede considerarse la mayor migración en la historia de la humanidad, ya que cada año se lleva a cabo por más de un 10% de la población mundial (Noor et al., 2016.). La contribución de los viajes y la industria del turismo al PIB mundial (9,6 por ciento) y al empleo (más de 272 millones de puestos) testifica también su gran impacto en aspectos socioeconómicos en todo el mundo. El World Travel & Tourism Council (WTTC) predice que, para los próximos 10 años, viajar y el turismo contribuirán en 11 trillones de dólares al PIB mundial y darán trabajo a 350 millones de personas. (Khatter et al., 2019).

Como puede comprobarse, el turismo tiene un grandísimo impacto en las actividades económicas. Es la industria que más empleo genera y tiene una cartera de servicios muy amplia: viajes, alojamiento, mantenimiento de la cultura y tradiciones y preservación de los ecosistemas. Además del impacto económico, también participa del desarrollo urbano e impacta sobre el cambio climático y la degradación ambiental. Según Sucheran y Moodley (2019) “el turismo ha sido acusado de ser un despojador de ambientes prístinos, un destructor de estilos de vida valorados y un explotador de naciones pobres y no existe el turismo de impacto cero".

Los viajes turísticos al destino incluyen actividades previas, durante y posteriores al viaje, las cuales mayoritariamente tienen un impacto medioambiental. El fenómeno del turismo es sensible al factor natural, cultural y ambiental, de hecho, solo un pequeño grupo de actividades turísticas se considera beneficioso y responsable de cara a la sostenibilidad (Noor et al., 2016). En la actualidad, destinos turísticos populares están siendo afectados por superpoblación, contaminación de agua y aire, basura, playas sucias, aglomeración y escasez de recursos. De acuerdo a evaluaciones recientes, muchos destinos alrededor del Mar Mediterráneo están, o se encuentran cerca de estar, superpoblados y presentan riesgos medioambientales para los humanos.

- **Impacto medio ambiental**

El consumo de energía en el turismo es mayor que en cualquier otro tipo de industria, y fluctúa dependiendo de la región y del perfil de sus visitantes (Navratil et al., 2019). Un hotel libera de media entre 160 y 200 kg de CO_2 por metro cuadrado de habitación cada año, dependiendo del combustible utilizado para generar electricidad, calor y frío. Además, según Bastič y Gojčič (2012), el mismo hotel también produce un exceso de 1 kg de residuos por huésped/día (que se traduce a varias toneladas cada mes), de los cuales un 50 - 60% podría ser reciclado o reutilizado.

Con respecto al desplazamiento, Chenoweth (2009) proporciona la siguiente información basada en los diferentes medios de transporte:

- Los viajes aéreos corresponden aproximadamente al 46% de los viajes internacionales, y según predicciones, la tasa anual de pasajeros viajando por aire incrementará de media un 4,6% entre 2005 y 2025. Dichos viajes aéreos tienen una implicación muy significativa en cuanto al cambio climático provocado por el turismo, aunque algunas estimaciones sugieren que actualmente sólo representa el 1% del total de emisiones de dióxido de carbono.
- El transporte por carretera representa el 43% de los viajes internacionales, con una mayoría de uso del automóvil. Dicho transporte también impacta de forma significativa en el cambio climático, aunque la media de emisiones de dióxido de carbono de los nuevos automóviles está disminuyendo con el tiempo, ya que los coches se vuelven cada vez más eficientes. Incluso con esta reducción gradual de emisiones, una persona viajando en coche hasta el destino vacacional sería responsable de muchas más emisiones de dióxido de carbono que otra que hubiese volado al mismo lugar. En cambio, para una familia de 4 miembros, el uso del coche sería más beneficioso para el medio ambiente; sin embargo, los turistas tienden a viajar utilizando el transporte aéreo.
- El transporte ferroviario corresponde al 4% de los viajes internacionales. Sus emisiones de dióxido de carbono dependen del tipo de fuente de alimentación y de la región. Aquellos trenes alimentados con energía eléctrica propia de una planta nuclear o hidráulica supondrán un menor impacto que los trenes alimentados con energía eléctrica propia de una planta térmica que utilice carbón.
- El transporte marítimo representa el 7% de los viajes internacionales. El viaje en crucero también es particularmente intensivo en el uso del carbón como fuente de alimentación.
- La opción que supone una menor emisión de dióxido de carbono en viajes es el uso de la bicicleta, ya que este medio de transporte no necesita de combustibles fósiles.

Sin embargo, en el alojamiento las cosas no son tan sencillas debido a que, por lo general, una persona duerme, come, se ducha y utiliza una serie de electrodomésticos tanto si se encuentra en casa como si se hospeda en un hotel, por lo que realmente se debe considerar sólo el cambio de emisiones para calcular el impacto de su viaje. El resultado de este impacto viene dado por el mantenimiento de su casa durante su ausencia (normalmente con una reducción significativa en consumo de energía) junto con el uso de instalaciones y/o electrodomésticos que no suele utilizar. El impacto medioambiental más grande resultante del alojamiento probablemente venga dado por el mantenimiento de ciertos espacios que necesitan ser calentados o

refrigerados estén o no completamente ocupados; al igual que en el transporte, las emisiones de dióxido de carbono por persona se reducirán cuando las tasas de ocupación sean altas.

Tal y como indica Chenoweth (2009), "estar ausente de casa supondrá una reducción en el uso de energía, aunque no llegará a ser cero ya que muchos electrodomésticos se dejan en modo standby (el cual sigue consumiendo energía), y los sistemas de calefacción pueden dejarse encendidos para evitar que las tuberías se congelen. Debido a este uso residual, una vivienda temporalmente desocupada puede consumir aproximadamente un 25% de su consumo normal de energía, siendo mayor esta cantidad en invierno y menor en verano".

Una vez en el destino, una de las actividades turísticas con una gran probabilidad de generar impactos medioambientales negativos es la elección de medio de transporte para desplazarse durante el período vacacional, el cual contribuye al incremento de la contaminación del aire, a la congestión del tráfico, al ruido y al riesgo de accidentes. Algunas de las actividades lúdicas pueden tener también un impacto negativo perturbando el hábitat local o sobre utilizando los espacios naturales.

- **Conciencia medio ambiental**

A medida que se han ido haciendo realidad los impactos perjudiciales del turismo, y ha ido aumentando la aparición del ambientalismo y la "conciencia verde", el rol del turismo con respecto al medio ambiente ha vuelto a ser evaluado (Sucheran y Moodley, 2019). Dichas preocupaciones por proteger el medio ambiente han traído cambios en las demandas y los comportamientos de los consumidores. De acuerdo con Han et al. (2011), un gran número de clientes muestran un aumento en su conciencia medioambiental y una clara preferencia por compañías verdes y sus productos, demostrando su disponibilidad para comprar y pagar más por productos y servicios respetuosos con el medio ambiente. Es por ello que, con el fin de satisfacer estas emergentes necesidades verdes, los responsables de negocios de distintos campos están realizando nombrosos esfuerzos por cambiar su cultura empresarial para ser más responsables con el medio ambiente y para que sus productos y servicios también lo sean.

De acuerdo con la Organización Mundial del Turismo (UNWTO) y con la Adventure Travel Trade Associaton (ATTA), sabemos que las tendencias mundiales en turismo han registrado un crecimiento muy rápido en los segmentos de aventura y turismo rural, con un especial énfasis en la búsqueda de paz, tranquilidad y harmonía con la naturaleza. El turismo actual busca convertir una ubicación en global al hacer que las tendencias turísticas sean más integrales y, al mismo tiempo, especializarse en segmentos de mercado y en productos que valoran las características locales en un contexto internacional. Es por eso que cada nación, país y región debe trabajar hacia un posicionamiento que sea distintivo, que agregue ventaja y que consolide su prestigio en relación con otros competidores. De hecho, cada destino debe mirar continuamente externa e internamente para asegurar que mantengan una percepción realista de su posicionamiento y el camino que deben seguir (Guerreiro, 2017). La industria del turismo debería, en la práctica, evitar daños al desarrollar y gestionar actividades turísticas. Los turistas están exigiendo más sobre vacaciones ecológicas y culturalmente sensibles (Noor et al., 2016).

3.2. Sostenibilidad

El daño causado por la raza humana al medio ambiente mediante el turismo está cada vez más reconocido, y es por ello que nuevos conceptos como turismo sostenible se vuelven más comunes. Noor et al. (2016) define el turismo sostenible como las actividades vacacionales realizadas en el destino con la responsabilidad de proteger el medio ambiente, mejorar la vertiente sociocultural e incrementar el beneficio económico de las personas residentes. Según Yilmaz et al. (2019), el turismo sostenible se define como un enfoque que tiene como objetivo proteger el medio ambiente y la cultura de las comunidades que albergan a los turistas, así como satisfacer las necesidades de los turistas y mantener el crecimiento de la industria del turismo.

Guerreiro (2017) considera los siguientes factores completamente necesarios para medir el nivel de sostenibilidad en el turismo:

- La calificación del destino, a través de la continua innovación en productos y servicios con el fin de consolidar una oferta única. Esto requiere una inversión en formación de múltiples recursos humanos, así como en la evolución de la calidad de la oferta del alojamiento y en la capacidad de ofrecer productos de animación turística y mejoras en las instalaciones disponibles.
- La sostenibilidad propia del destino, promoviendo actividades turísticas que minimicen la huella ecológica, que regulen el uso de recursos naturales compartidos y que fomenten programas para disminuir la estacionalidad en términos de la oferta turística y la empleabilidad del sector.
- La efectividad de la promoción, con el aumento del prestigio del destino y la atracción de flujos turísticos, diversificando los canales comerciales de marketing y comunicación.
- La eficiencia de la accesibilidad, tanto interna como externa.

Además, Dávid (2011) destaca las siguientes características para que el turismo se pueda considerar como sostenible:

- Integración del turismo en la planificación.
- Apoyo de la economía local.
- Intervención de comunidades locales.
- Comunicación entre stakeholders y la comunidad.
- Intervención de la población local en la planificación.
- Formación y desarrollo de habilidades dentro de los recursos humanos locales.
- Marketing responsable.
- Desarrollo de la política turística como parte orgánica de la política general de la sociedad local.

En general, es evidente a partir de la literatura que los turistas se están volviendo más conscientes del medio ambiente y algunos autores esperan que los turistas se vuelvan más exigentes en su elección de productos y destinos turísticos sostenibles. Con respecto al transporte, Chenoweth (2019) destaca que los viajeros preocupados por el medio ambiente tienen opciones abiertas que les permiten viajar con carbono neutral o casi carbono neutral, y como un beneficio adicional, tales opciones de viaje tienden a ser mucho más bajas en costo y

proporcionan significativamente una gran interacción con la gente local en los países que se visitan.

3.3. Gestión medioambiental

El término "gestión medioambiental" está presente en la legislación ambiental, y se considera una estrategia de gestión que en última instancia tiene como objetivo dar forma o cambiar el comportamiento de las personas en su entorno para regular los efectos de las actividades, productos y servicios de las personas en el medio ambiente. El crecimiento de la gestión medioambiental en la industria de servicios, particularmente en el sector turístico, es relativamente reciente, y la implementación de dicha gestión en este sector ha mejorado en los últimos años.

La gestión medioambiental se ha convertido en una práctica necesaria para que las empresas turísticas mantengan su posición competitiva (Guerreiro, 2017). Las consecuencias del cambio climático y la degradación ambiental han creado un imperativo para que las industrias adopten e implementen políticas y prácticas ambientalmente sostenibles (ESPP). Los ESPP son los principios, pautas y prácticas básicos formulados para ayudar a una organización a alcanzar sus objetivos de gestión medioambiental. De una encuesta reciente de más de 1,000 CEOs de 100 países y 25 negocios diferentes, el 89 por ciento indicó que el compromiso con la sostenibilidad es crítico para el éxito de su negocio. Una gran mayoría de las pequeñas y medianas empresas (PYMES) no están convencidas de la necesidad de abordar los problemas de sostenibilidad ambiental. Además, las organizaciones más pequeñas no tienen tanta presión de los clientes, los stakeholders y otros impulsores, lo que resulta en un menor esfuerzo para abordar la sostenibilidad ambiental. Incluso si una pequeña empresa se embarca en la implementación de ESPP, a menudo faltan recursos humanos con la experiencia sobre cómo implementar la gestión ambiental (Khatter et al., 2019).

En el sector turístico, la industria hotelera es responsable de aproximadamente el 21% de todas las emisiones de CO2. A medida que las personas están cada vez más preocupadas por el calentamiento global, se vuelve más común que los viajeros tomen una decisión ecológica para seleccionar un hotel. Por lo tanto, un número creciente de hoteles ha empezado a implementar prácticas ecológicas y estrategias medio ambientales, y ha convertido los procedimientos de compra para que sean más respetuosos con el medio ambiente.

La práctica más relevante adoptada por los hoteles con respecto a sus proveedores es la colaboración con los proveedores de servicios en la adquisición de alimentos, bebidas, materiales culinarios, mantelería y otros aspectos logísticos. Otros esfuerzos dirigidos a generar prácticas más sostenibles incluyen reducir el impacto de los proveedores en el transporte, minimizar las huellas de carbono y utilizar el transporte ecológico, como los automóviles eléctricos, así como promover estaciones de carga para este tipo de vehículos en los hoteles. Algunas prácticas con respecto a sus clientes son la gestión del agua, las comunicaciones visibles sobre prácticas ecológicas a los huéspedes, establecer un programa de reciclaje de materiales en todas las secciones del hotel, etc. (Alonso-Almeida et al., 2017).

Un "hotel verde" se define como un hotel ecológico que realiza/sigue varias prácticas/programas respetuosos con el medio ambiente, como el ahorro de agua/energía, el uso de políticas de compra ecológicas y la reducción de las emisiones/desechos para proteger el medio ambiente natural y reducir costes operacionales. La gestión ecológica puede crear una tremenda ventaja competitiva para un hotel al permitir la diferenciación de marca, cultivar la lealtad del cliente y mejorar la reputación de un hotel (Han et al., 2011). Sin embargo, los hoteleros europeos no han percibido su compromiso medioambiental como un importante factor de marketing, ya que creen que los huéspedes tienen un interés limitado en cuestiones medio ambientales y que el comportamiento ecológico en los hoteles implica costos de inversión significativos (Bastič y Gojčič, 2012).

Por otra parte, la tecnología online complementa las formas tradicionales de consumo y se usa cada vez más para fines de marketing ecológico. Dicha tecnología proporciona a los usuarios un espacio para interactuar y conectarse con otras personas, por lo tanto, es importante que los gerentes de los hoteles hagan un esfuerzo para comunicar sus prácticas ecológicas y escuchar adecuadamente las opiniones de los clientes sobre estas prácticas. Las redes sociales pueden ser una herramienta efectiva para que los hoteles comuniquen sus prácticas ecológicas a los clientes y los motiven a ser ecológicos.

3.4. Certificaciones turísticas

De acuerdo con Martínez et al. (2019), estudios previos han demostrado que los consumidores apoyan a empresas ecológicas que consideran prioritario desarrollar prácticas medio ambientales. Con el objetivo de reducir la degradación ambiental e involucrar a los consumidores conscientes de los problemas ecológicos, numerosas empresas están implementando programas de certificación ambiental. Las certificaciones medio ambientales intentan promover el compromiso medio ambiental de las empresas y diferenciar a las empresas respetuosas con el medio ambiente de aquellas que se venden como ecológicas, pero realmente no lo son.

El "green washing" se define como el acto de engañar a los consumidores con respecto a las prácticas medio ambientales de una empresa o los beneficios medio ambientales de un producto o servicio (Martínez et al., 2019). Es por ello que las certificaciones medio ambientales brindan credibilidad sobre el desempeño medio ambiental y las iniciativas ecológicas adoptadas por las empresas, lo que reduce las afirmaciones de "green washing". Una de las certificaciones más importantes es la ISO 14001, que es un estándar internacional para la operación sostenible (Martínez et al., 2019).

La certificación más conocida es la etiqueta ecológica, que se otorga para fomentar el turismo sostenible en el mundo. Es una forma significativa de proporcionar transparencia para mostrar consistencia en las prácticas medio ambientales y generar confianza en el consumidor. Las etiquetas ecológicas son herramientas utilizadas por países u organizaciones para crear conciencia sobre la mayor calidad ecológica de ciertos productos y servicios en comparación con los productos y servicios no etiquetados. Al mismo tiempo, las etiquetas ecológicas son una herramienta que ayuda a los clientes a reconocer fácilmente productos o servicios que no dañan el medio ambiente. Hoy en día, hay casi 60 sistemas de etiquetado ecológico en la industria del

turismo basados en características como regiones geográficas, subsectores, restricciones, temas de turismo, sistema de gestión, etc. (Yilmaz et al., 2019).

A pesar de los intentos de fomentar el turismo sostenible utilizando diversas certificaciones medio ambientales, el problema es que los turistas tienen dificultades para reconocer dichas etiquetas ecológicas, certificaciones y programas. La investigación realizada por Park & Boo (2010) descubrió que muchos esquemas de certificación ambiental no son bien conocidos en la industria del turismo. Por ejemplo, el programa de certificación turística más popular, Green Globe, fue reconocido por menos del 5% del total de encuestados, y el 15% de los encuestados no conocía la certificación medio ambiental en general (Mazhenova et al., 2016).

3.5. Factores de decisión en la elección de vacaciones sostenibles

En la actualidad, los turistas disponen de muchas opciones para preparar sus vacaciones. Gracias a Internet, son autosuficientes y pueden elegir libremente el transporte, el alojamiento e incluso las actividades a realizar en el destino. Por otro lado, una forma más restrictiva para los turistas es dejar que los agentes de viajes les preparen los paquetes turísticos. De acuerdo con Noor et al. (2016), los agentes de viaje juegan un papel importante en el cambio de comportamientos y actitudes hacia formas más responsables de turismo, y deberían ser capaces de preparar paquetes turísticos sostenibles y seguir siendo competitivos.

Independientemente de la forma de preparar las vacaciones, como Noor et al. (2016) explica, hay una lista de factores que motivan a los turistas sostenibles a elegir transporte, alojamiento y actividades sostenibles:

Medio ambientales

1. Ahorro de costes al reducir el consumo de papel, energía, agua y otros suministros.
2. Administración de productos y servicios de diseño de paquetes vacacionales con menor impacto medio ambiental y social.
3. Promover vacaciones más verdes y justas.

Socioculturales

1. Buenas prácticas laborales y respeto de los derechos humanos. Aumentar la moral del personal, permitir una mayor retención de personal de alta calidad, mejorar las condiciones de trabajo, etc.
2. Sensibilización sobre temas sostenibles con el apoyo del agente de viajes.
3. Proteger la privacidad, salud y seguridad del cliente.
4. Formación al cliente en aspectos relacionados con la sostenibilidad.
5. Involucrar al cliente en el turismo sostenible.

Desarrollo económico

1. Cooperación con el destino de manera sostenible mediante el establecimiento de vínculos y el desarrollo de asociaciones con las partes interesadas.
2. Impulsar el desempeño de sostenibilidad de mejores prácticas con proveedores.

Según Dávid (2011), el paisaje intacto, la diversidad de la vida silvestre y el medio ambiente limpio y no contaminado se han convertido en los factores más importantes de atracción para los turistas que visitan un destino. Aun así, Jung et al. (2014) también creen que una gran experiencia gastronómica, que abarca la comida y el vino, puede desempeñar un papel clave en el atractivo de un destino turístico.

3.6. Actitudes y comportamientos de turistas sostenibles

- **Información y datos**

Las actitudes respetuosas con el medio ambiente de las personas, sin duda, juegan un papel importante al influir en sus comportamientos ecológicos de compra. Las personas que creen que sus actividades ecológicas específicas pueden causar cambios positivos tienen más probabilidades de mostrar comportamientos ecológicos de consumo.

Una premisa fundamental del comportamiento del consumidor es que los consumidores desarrollan ciertas actitudes hacia los productos que revelan sus preferencias y, en consecuencia, sus comportamientos reales (González-Rodríguez et al., 2019). De acuerdo con Noor et al. (2016), los turistas sostenibles presentan las siguientes actitudes cuando viajan:

Medio ambientales

1. Preservan el medio ambiente natural protegiendo la vida silvestre y los habitantes. No compran productos hechos de plantas o animales en peligro de extinción.
2. Vuelan en aviones que consumen menos combustible.

Socioculturales

1. Se abren a otras culturas y tradiciones a través del respeto. Se ganan la bienvenida de la gente local siendo tolerantes y respetando la diversidad.
2. Respetan los derechos humanos y no generan ningún conflicto.
3. Respetan los recursos culturales realizando actividades respetuosas con el patrimonio artístico, arqueológico y cultural.

Desarrollo económico

1. Contribuyen al desarrollo económico mediante la compra de productos locales.

Cada vez más clientes con conciencia medio ambiental buscan hoteles que sigan prácticas para proteger el medio ambiente. Una de las limitaciones que encuentran los turistas al elegir hoteles ecológicos es la falta o la dificultad de acceder a información al respecto.

Según Mensah (2004), el 90% de los huéspedes preferiría quedarse en un hotel que implementa la gestión ecológica. Baker et al. (2019) afirman que el 71% de los turistas estadounidenses planean más vacaciones ecológicas que el año anterior, y la mitad tiende a gastar más dinero en alojamiento ecológico. Un estudio reciente de la compañía líder mundial de reserva de habitaciones de hotel (Booking.com) encontró que el 87 por ciento de 4.768 encuestados quería viajar de manera sostenible y el 68 por ciento de estos querían hospedarse en hoteles ecológicos (Khatter et al., 2019).

- **Estrategia sostenible**

La forma en que las personas se relacionan con el medio ambiente y el bienestar de los demás determina su disposición a participar en actividades ambiental y socialmente responsables. Los turistas pertenecen a tres grupos principales: verdes (inclinados a actuar en nombre de otros), grises (no interesados en el bienestar de los demás) y marrones (ambiguos a tales problemas). Para un futuro sostenible, los turistas deseados son los verdes, aunque solo representan una pequeña fracción de la población.

Según Budeanu (2007), para recibir una respuesta positiva, las ofertas de productos y servicios de turismo sostenible deben estar orientadas a grupos que estén dispuestos a escuchar (los turistas verdes y marrones). Un buen entendimiento de cómo las actitudes, las personalidades y los estilos de vida influyen en las decisiones turísticas se traducirá en el diseño de políticas y propuestas más efectivas y exitosas de productos turísticos sostenibles.

- **Turistas sostenibles**

Los turistas sostenibles se preocupan en gran medida por la vida silvestre, el transporte, la conservación, el uso de los recursos, la contaminación y las prácticas de las empresas relacionadas con el turismo, y buscan comprar productos y servicios ecológicos a aquellas empresas que respeten el medio ambiente. Estas personas están dispuestas a cambiar su comportamiento de compra a uno más ecológico (por ejemplo, evitar comprar productos desechables, reciclar, reducir el consumo de agua y energía, etc.), sacrificando la calidad y a veces pagando más por los productos (Mazhenova et al., 2016).

Dada la misma calidad y función cumplidas, es probable que los clientes prefieran aquellas alternativas respetuosas con el medio ambiente. Sin embargo, dichas alternativas pueden ser de difícil acceso, menos cómodas, menos atractivas o requerir tiempo adicional para los turistas (Budeanu, 2007).

Sin embargo, otros investigadores también han observado que, aunque los consumidores expresan su preocupación por el medio ambiente, estas preocupaciones no siempre se traducen en la compra o el consumo de productos y servicios ecológicos. La razón sugerida para esto es que los clientes desconfían mucho de las compañías que se anuncian a sí mismas como industria sostenible.

3.7. Variables que afectan al turismo sostenible

Al intentar encontrar patrones de comportamiento, es importante saber qué variables necesitan ser medidas. Para estudiar más fácilmente cómo reacciona la gente al turismo sostenible, es interesante ordenarlos en grupos. Como en muchos otros casos, las conclusiones se extraen ordenando a las personas por género, por edad, por educación y por ingresos.

Han et al. (2011) encuentra interesantes los siguientes puntos:

- Las mujeres y los hombres juegan diferentes roles y muestran comportamientos diferentes en la sociedad porque están socializados de manera diferente. La literatura

sugiere que los hombres y las mujeres difieren en sus patrones y comportamientos de consumo. Las mujeres están más preocupadas por el bienestar de otras personas, perciben las relaciones interpersonales como más importantes y revelan una mayor preferencia por la información y la comunicación. Además, las mujeres tienden a ser más conscientes del medio ambiente.

- Algunos resultados iniciales muestran que la edad está significativamente relacionada con comportamientos de compra respetuosos con el medio ambiente. Más específicamente, los resultados indican que los clientes que con frecuencia toman decisiones de compra ecológicas tienen más probabilidades de ser jóvenes.
- Los que tienen un alto nivel educativo y tienen mayores ingresos tienden a ser más conscientes con el medio ambiente y a participar más activamente en la compra de productos ecológicos.

Por tanto, se podría decir que los clientes con conciencia medio ambiental son más propensos a ser mujeres, jóvenes, con alto nivel de educación y ganar más dinero que el promedio.

4. Objetivos de la investigación

4.1. Preguntas de investigación

En esta investigación se plantea analizar cómo de importante es el medio ambiente para los turistas europeos en el momento de tomar decisiones en la preparación de sus vacaciones. Las preguntas de investigación que ayudan a realizar dicho trabajo de investigación son las siguientes:

- ¿En qué medida está impactando el turismo europeo en el medio ambiente?
- ¿Qué nivel de consideración medio ambiental tienen los turistas europeos?
- ¿Qué aspectos ecológicos son más importantes para los turistas ecológicos europeos en sus vacaciones?
- ¿Qué variables describen a un turista ecológico europeo?
- ¿Qué modelo de predicción es el óptimo para clasificar el nivel ecológico de los turistas europeos?

4.2. Importancia de la investigación

Se considera que esta investigación es importante en varios aspectos, tanto para proveedores de servicios turísticos públicos y/o privados, como para los mismos destinos vacacionales.

- **Contribución para proveedores de servicios turísticos**

El principal beneficio para los proveedores de servicios turísticos tras realizar esta investigación será la capacidad de conocer el comportamiento actual de sus posibles clientes. De este modo, podrán mejorar su experiencia de compra y así responder mejor a sus expectativas, adecuando las ofertas a sus intereses. Una estrategia de negocio centrada en el cliente permite diseñar productos y servicios teniendo en cuenta al cliente en todas las fases y decisiones de conceptualización y planificación. Gracias a esto se consigue una personalización máxima, que se traduce en un mayor éxito en la venta de dichos productos y servicios, incrementando considerablemente los beneficios económicos.

Además de incrementar las ventas, también les permitirá crear y/o mejorar la estrategia de sostenibilidad en base a los niveles generales europeos de concienciación medio ambiental, con el fin de aportar valor social y ambiental.

- **Contribución para destinos vacacionales**

En cuanto a la contribución para destinos vacacionales, basándose en los niveles de concienciación medio ambiental podrán, por una parte, enfocar de manera más eficiente las campañas de marketing para atraer turistas, y por otra parte, diseñar campañas de formación que muestren a los turistas la importancia de reciclar, respetar el medio ambiente y las poblaciones locales y contribuir con el desarrollo sostenible del destino vacacional, en el momento de la reserva de sus vacaciones o al llegar al destino.

5. Datos y variables

5.1. Selección de datos y muestras

Existe mucha documentación referente al turismo sostenible y a medidas implantadas a nivel público y privado en destinos vacacionales concretos, que dan una visión específica del nivel ecológico de dicha región. Si nos centramos en obtener información propia de comportamientos y actitudes de turistas, aunque se dispone de menos información, la mayoría de ella hace referencia a la sociedad estadounidense o asiática. Dicha escasez de documentación y resultados de la región europea es uno de los motores de este trabajo de investigación.

Los datos que alimentan la investigación corresponden a las respuestas del cuestionario *Flash Eurobarometer 432 (Preferences of Europeans towards Tourism, January 2016)*. La encuesta, realizada por vía telefónica a unos 30.105 encuestados de diferentes grupos sociales y demográficos, analiza los patrones de viaje de los ciudadanos europeos en los 28 Estados miembros de la Unión Europea y en Turquía, la Antigua República Yugoslava de Macedonia, Islandia, Montenegro y Moldavia. Fue diseñado para explorar una variedad de aspectos relacionados con las vacaciones en 2015 y 2016, en particular:

- Las razones de los encuestados para irse de vacaciones en 2015.
- Fuentes de información y herramientas utilizadas para reservar vacaciones.
- Perfiles de viaje de los encuestados, destinos preferidos y tipos de vacaciones.
- Satisfacción con varios aspectos de las vacaciones en 2015.
- Planes para vacaciones en 2016, incluido el impacto potencial de la situación económica actual en estos planes.

Aparte del informe, que sólo utiliza estadísticas descriptivas para dar una visión general de las tendencias en turismo europeo, no existe evidencia de otros estudios con el mismo alcance de esta investigación (preguntas de investigación, variables y metodologías).

El cuestionario y el conjunto de datos están disponibles en la siguiente dirección web: https://search.gesis.org/research_data/ZA6654.

5.2. Preprocesamiento de datos

El cuestionario genera un conjunto de datos con 625 variables y 30.105 registros, del cual se define la variable correspondiente a la pregunta Q8C como nuestra variable objetivo, a partir de la que girará toda la investigación.

Debido al gran tamaño del conjunto de datos, y tras un análisis rápido del mismo, se decide realizar un preprocesamiento con el fin de crear un conjunto de datos reducido que incluya las variables adecuadas y los valores preparados para trabajos posteriores.

Todos los trabajos incluidos en esta investigación (preprocesamiento, análisis y creación de modelo de predicción) se han realizado programando *scripts* en lenguaje de programación Python.

Python es un veterano lenguaje de programación presente en multitud de aplicaciones y sistemas operativos. Podemos encontrarlo corriendo en servidores, en aplicaciones iOS, Android, Linux, Windows o Mac. Es un lenguaje de programación versátil multiplataforma y multiparadigma que se destaca por su código legible y limpio. Una de las razones de su éxito es que cuenta con una licencia de código abierto que permite su utilización en cualquier escenario. Python es ideal para trabajar con grandes volúmenes de datos ya que, el ser multiplataforma, favorece su extracción y procesamiento (OpenWebinars, 2019).

En la preparación de datos se han realizado los siguientes cambios:

- Eliminación de variables que no aportan valor a la investigación.
- Modificación de valores (conversión de tipos de datos, relleno de valores nulos, creación de categorías, etc.).
- Eliminación de registros irrelevantes para la investigación.
- Creación de nuevas variables.
- Recodificación de variables.

La Tabla 2 proporciona una visión general de las 41 variables que conforman el conjunto de datos reducido tras realizarse las anteriores modificaciones.

Tabla 2. Definición de variables

Tipo de variable	Código original	Nuevo nombre	Descripción	Medida
Dependiente	-	eco_tourist	Nivel ecológico de los turistas europeos	Categórica
Independiente	q1	traveltimes	Número de viajes/vacaciones realizados en 2015	Numérica
	q1r	traveltimes_cat	Cantidad de viajes/vacaciones realizaos en 2015 dividido en grupos	Categórica
	q2a1	long_vac	Número de viajes/vacaciones de larga duración (más de 13 días)	Numérica
	q2a1r	long_vac_cat	Número de viajes/vacaciones de larga duración dividido en grupos	Categórica
	q2a2	med_vac	Número de viajes/vacaciones de media duración (de 4 a 13 días)	Numérica
	q2a2r	med_vac_cat	Número de viajes/vacaciones de media duración dividido en grupos	Categórica
	q2a3	short_vac	Número de viajes/vacaciones de corta duración (menos de 4 días)	Numérica
	q2a3r	short_vac_cat	Número de viajes/vacaciones de corta duración dividido en grupos	Categórica
	q2b_1	Long_accom	Tipo de alojamiento para los viajes/vacaciones de larga duración	Categórica
	q2b_2	med_accom	Tipo de alojamiento para los viajes/vacaciones de media duración	Categórica
	q2b_3	short_accom	Tipo de alojamiento para los viajes/vacaciones de corta duración	Categórica
	q4a	vis_cntry	País visitado en las vacaciones principales	Categórica
	q5a	reason_vac_prin	Razón principal para ir de vacaciones	Categórica

Tipo de variable	Código original	Nuevo nombre	Descripción	Medida
Independiente	q5b.1	reason_vac_1	Segunda razón para ir de vacaciones	Categórica
	q5b.2	reason_vac_2	Tercera razón para ir de vacaciones	Categórica
	q5b.3	reason_vac_3	Cuarta razón para ir de vacaciones	Categórica
	q7a	same_dest	Motivo principal para volver al mismo lugar de vacaciones	Categórica
	q7b.1	same_dest_1	Segundo motivo para volver al mismo lugar de vacaciones	Categórica
	q7b.2	same_dest_2	Tercer motivo para volver al mismo lugar de vacaciones	Categórica
	q7b.3	same_dest_3	Cuarto motivo para volver al mismo lugar de vacaciones	Categórica
	q8a_1	accom_qual	Nivel de satisfacción con la calidad del alojamiento	Categórica
	q8a_2	accom_sec	Nivel de satisfacción con la seguridad del alojamiento	Categórica
	q8a_3	envi_char	Nivel de satisfacción con las características del entorno	Categórica
	q8a_4	price_lvl	Nivel de satisfacción con el nivel general de precios	Categórica
	q8a_5	tour_welc	Nivel de satisfacción con la bienvenida dada a los turistas	Categórica
	q8a_6	avail_serv	Nivel de satisfacción con las actividades/servicios disponibles	Categórica
	q8a_7	spec_inst	Nivel de satisfacción con las instalaciones accesibles para personas con necesidades especiales	Categórica
	q8b.1	vac_company_1	Compañía en las vacaciones principales	Categórica
	q8b.2	vac_company_2	Segunda compañía en vacaciones secundarias	Categórica
	q8b.3	vac_company_3	Tercera compañía en vacaciones secundarias	Categórica
	q8b.4	vac_company_4	Cuarta compañía en vacaciones secundarias	Categórica
	q8c.1	eco_aspect_1	Aspecto ecológico principal para elegir destino a visitar en vacaciones	Categórica
	q8c.2	eco_aspect_2	Segundo aspecto ecológico para elegir destino a visitar en vacaciones	Categórica
	q8c.3	eco_aspect_3	Tercer aspecto ecológico para elegir destino a visitar en vacaciones	Categórica
Control	d1	age	Edad de los turistas	Numérica
	d1r2	age_cat	Edad de los turistas dividida en grupos	Categórica
	d2	sex	Género de los turistas	Categórica
	d5r	job	Oficio de los turistas	Categórica
	d13	living_place	Lugar de residencia de los turistas	Categórica

El conjunto de datos original contiene un número muy elevado de variables, la mayoría de las cuales son variables dummy en las que los datos se encuentran codificados de manera binaria en múltiples columnas. Con la eliminación de algunas variables con información irrelevante (identificación del cuestionario, códigos internos, etc.) y la conversión de variables dummy a variables en formato "alto", en el que cada columna contiene la información de una variable, se consigue reducir el número de variables a 40. Dichas variables también se recodifican para facilitar su entendimiento y su uso en trabajos de análisis posteriores.

Además, y con el fin de abordar el propósito de la investigación, se crea la variable categórica "eco_tourist" que puede tomar los siguientes valores:

- No ecológico – Si el turista no ha seleccionado ningún aspecto ecológico en las variables eco_aspect_1, eco_aspect_2 y eco_aspect_3.
- Poco ecológico – Si el turista ha seleccionado 1 aspecto ecológico en las variables eco_aspect_1, eco_aspect_2 y eco_aspect_3.
- Ecológico – Si el turista ha seleccionado 2 aspectos ecológicos en las variables eco_aspect_1, eco_aspect_2 y eco_aspect_3.
- Muy ecológico – Si el turista ha seleccionado 3 aspectos ecológicos en las variables eco_aspect_1, eco_aspect_2 y eco_aspect_3.

Por otra parte, todos aquellos registros que contengan valores "Don't know" ó 0 en la variable "traveltimes" son excluidos del conjunto de datos con el fin de reducirlo y centrarse en aquellos casos que ofrezcan información importante. De esta forma, todos aquellos turistas que no hayan viajado, no hayan querido responder a dicha pregunta o no tuviesen información clara para responderla, no son considerados en este estudio. Se reduce el número de registros de 30.105 a 19.478.

El código para realizar este trabajo de preprocesamiento se puede encontrar en el ***Anexo 3 – TFM_1.py***.

5.3. Análisis descriptivo

5.3.1. Análisis general

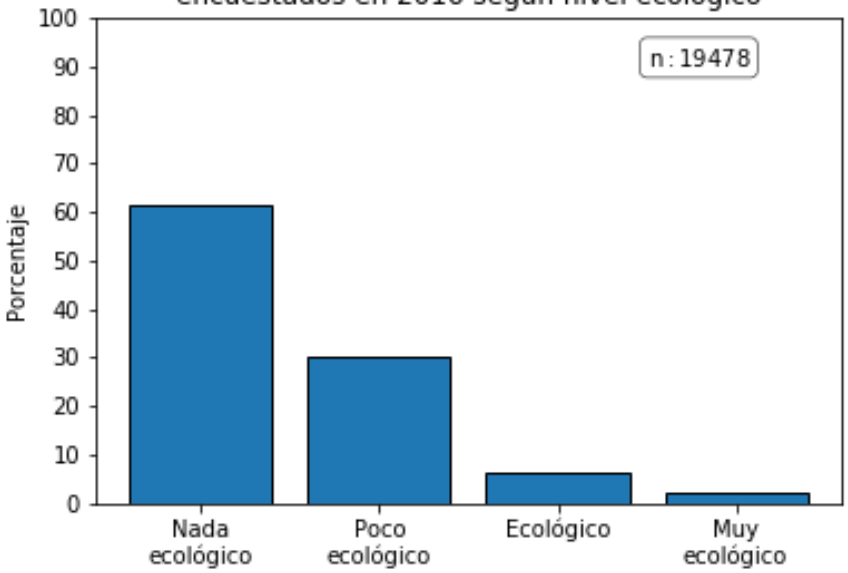
Con el fin de entender los datos y ver las distribuciones que sigue cada variable se realizan una serie de análisis descriptivos. Para estudiar las variables categóricas, se generan las correspondientes tablas de frecuencias y porcentajes y se presentan los datos en gráficos de barras. Para estudiar las variables numéricas, se generan las tablas con sus estadísticos y se presentan las distribuciones mediante histogramas.

Para la variable objetivo "eco_tourist", variable categórica con 4 grupos, se genera la siguiente tabla y el siguiente gráfico.

Tabla 3. Porcentaje del tipo de turista según el nivel ecológico de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Nada ecológico	11944	61,32	61,32
Poco ecológico	5900	30,29	91,61
Ecológico	1243	6,38	97,99
Muy ecológico	392	2,01	100
N	19478	100	

Figura 1. Clasificación de los turistas encuestados en 2016 según nivel ecológico



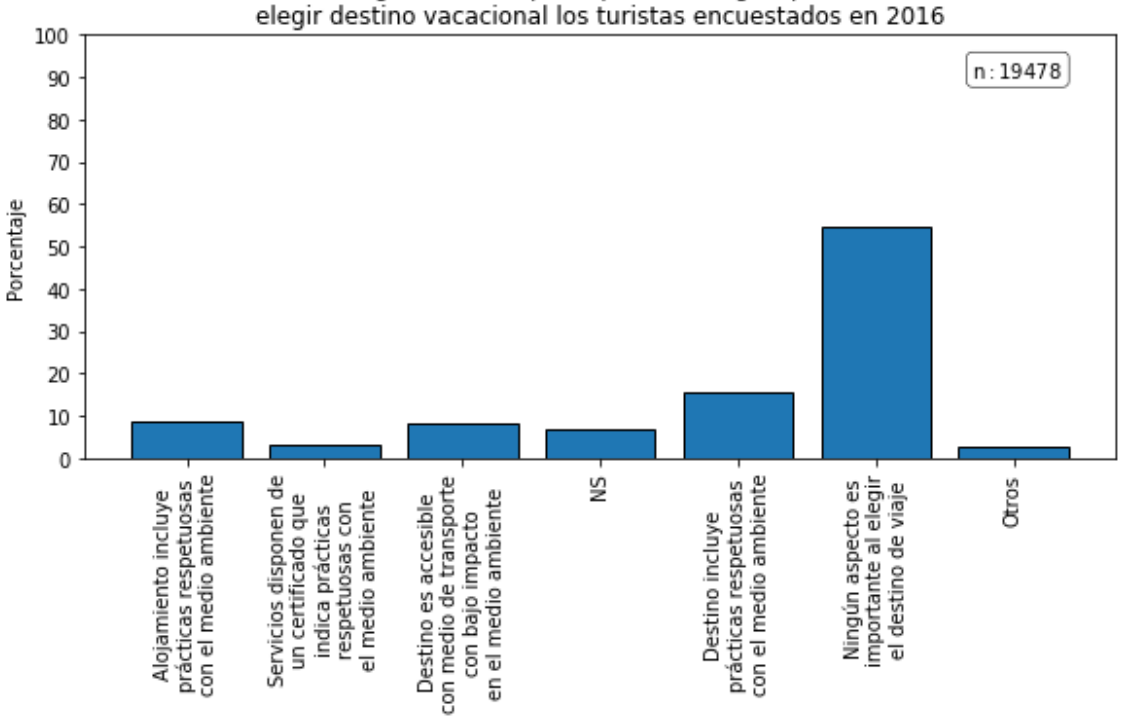
De los 19.478 turistas incluidos en el estudio, más de la mitad son catalogados como turistas nada ecológicos, ya que no consideran como esencial ningún aspecto ecológico a la hora de elegir su destino vacacional. A primera vista, se puede concluir que, al menos los turistas europeos encuestados en 2016 son mayoritariamente no ecológicos, pero para continuar el trabajo de investigación se seguirá estudiando los patrones de ese 40% ecológico.

La Tabla 4 muestra como ese elevado porcentaje de turistas que consideran que ningún aspecto es importante, marca esa clasificación de turistas nada ecológicos. De dicha tabla también se extrae que el principal aspecto ecológico para elegir destino vacacional es que el mismo destino incluya prácticas respetuosas con el medio ambiente. La decisión de los turistas ecológicos para elegir destino, depende principalmente de la región a visitar como entidad pública, y no tanto de otro tipo de servicios privados.

Tabla 4. Porcentaje del principal aspecto ecológico para elegir destino vacacional los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Alojamiento incluye prácticas respetuosas con el medio ambiente	1675	8,60	8,60
Servicios disponen de certificado ecológico	639	3,28	11,88
Destino es accesible con transporte de bajo impacto medio ambiental	1623	8,33	20,21
NS (No sabe)	1319	6,77	26,98
Destino incluye prácticas respetuosas con el medio ambiente	3070	15,76	42,74
Ningún aspecto es importante	10625	54,55	97,29
Otros	528	2,71	100
N	19478	100	

Figura 2. Principal aspecto ecológico para elegir destino vacacional los turistas encuestados en 2016

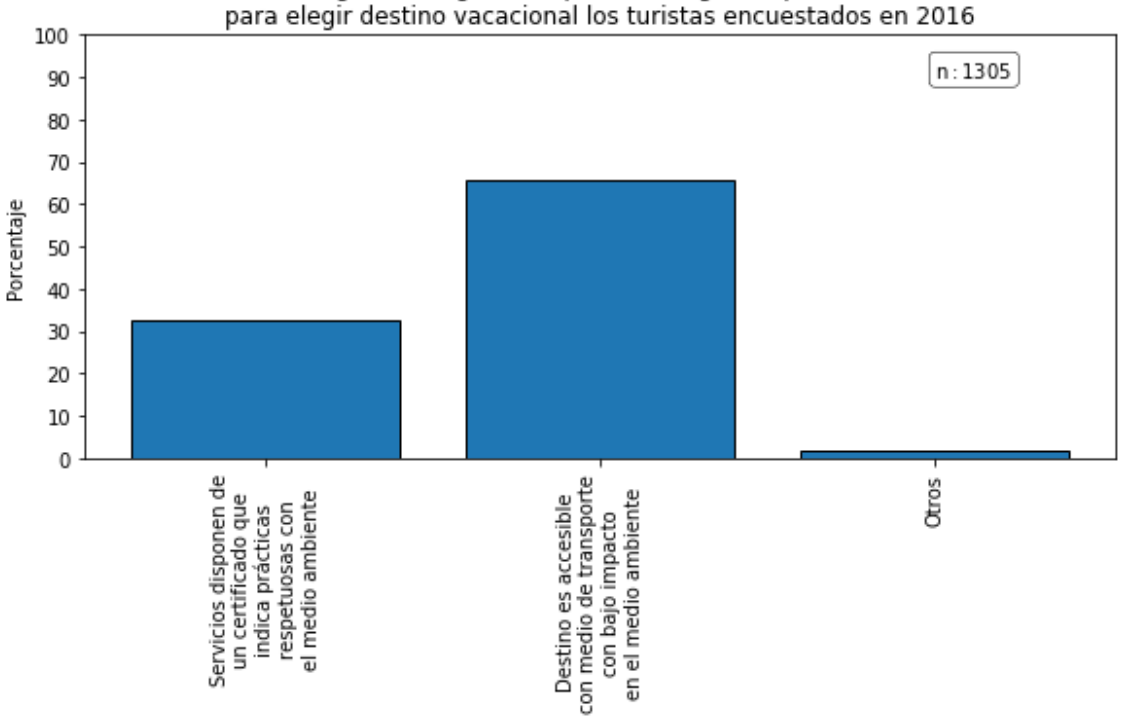


Si nos centramos en aquellos turistas que no sólo respondieron el cuestionario con un aspecto ecológico, si no también con un segundo, nos quedamos con un número reducido de 1.305. Para este caso, tal y como muestra la Tabla 5, la gran mayoría de turistas consideran importante que el destino sea accesible con transporte de bajo impacto medio ambiental (en la Tabla 4 se encontraba en la tercera posición de importancia, prácticamente a la par con el segundo). De nuevo, se encuentra un aspecto que vuelve a depender del destino y de las facilidades que ofrezca para llegar al mismo con transportes ecológicos.

Tabla 5. Porcentaje del segundo aspecto ecológico importante para elegir destino vacacional los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Servicios disponen de certificado ecológico	427	32,72	32,72
Destino es accesible con transporte de bajo impacto medio ambiental	856	65,59	98,31
Otros	22	1,69	100
N	1305	100	

Figura 3. Segundo aspecto ecológico importante para elegir destino vacacional los turistas encuestados en 2016

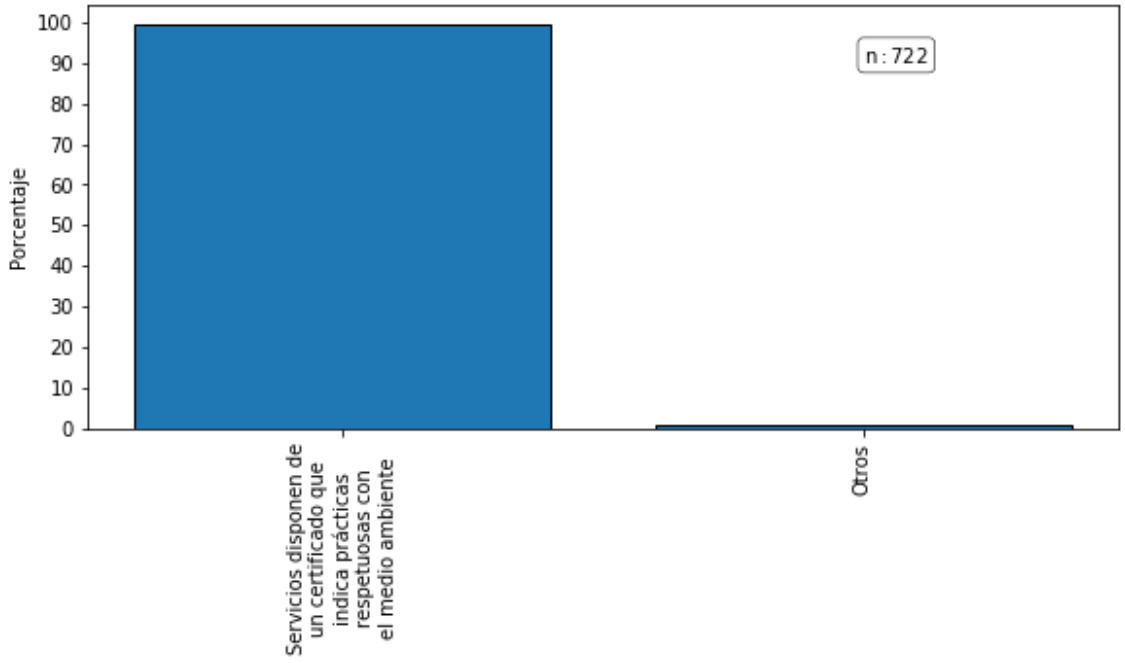


Finalmente, los turistas que responden el cuestionario con 3 aspectos ecológicos como importantes para elegir destino vacacional (N = 722), consideran, prácticamente por unanimidad tal y como muestra la Tabla 6, que es necesario que los servicios dispongan de certificado ecológico. Este aspecto ya se muestra en segundo lugar de importancia en la Tabla 5.

Tabla 6. Porcentaje del tercer aspecto ecológico importante para elegir destino vacacional los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Servicios disponen de certificado ecológico	717	99,31	99,31
Otros	5	0,69	100
N	722	100	

Figura 4. Tercer aspecto ecológico importante
para elegir destino vacacional los turistas encuestados en 2016



A modo general, y con el fin de entender otra serie de distribuciones, se puede consultar el ***Anexo 1 – Tablas y gráficos análisis*** del que se extrae la siguiente información.

La edad de los turistas sigue una distribución normal con una media de 52 años, ya que se encuestó a más personas de edades elevadas que a personas jóvenes. En cuanto al género, se encuestaron ligeramente a más mujeres que hombres, de los cuales prácticamente la mitad son empleados o autónomos y el resto desempleados o jubilados. Con respecto a la nacionalidad y lugar de residencia, se encuestaron en prácticamente la misma proporción a personas de 33 países europeos que residen en ciudades, zonas rurales y pueblos.

Haciendo referencia a los viajes/vacaciones realizadas, se tiene una media de 4 una vez quitados los outliers de la investigación. Destacar que más de la mitad de turistas encuestados no realiza viajes/vacaciones de larga duración (más de 13 días), que para viajes de media duración (entre 4 y 13 días) lo normal es hacerlo entre 1 y 2 veces, y que para viajes de corta duración (menos de 4 días) es más fácil ver números de veces más elevados ya que se empiezan a igualar las frecuencias.

Para cualquiera de los tres tipos de vacaciones según la duración (largas, medias y cortas) los tipos de alojamiento más utilizados son el alojamiento comercial (hoteles, hostales, etc.), el alojamiento con amigos o familia y el alquiler privado respectivamente; y los países más visitados son el mismo país de residencia, seguido de España, Italia y Francia. La principal razón por la que viajan los turistas encuestados es la búsqueda de sol y playa y la visita a familiares, y el principal motivo para repetir el destino en próximas vacaciones son los aspectos naturales (entorno natural, condiciones meteorológicas, etc.). Por último, destacar que la gran mayoría de turistas viajan en familia o en pareja y que todos los aspectos evaluados (accesibilidad instalaciones, servicios disponibles, bienvenida turistas, nivel precios, etc.) presentan un nivel elevado en cuanto a satisfacción.

El código para realizar este trabajo de análisis general se puede encontrar en el ***Anexo 4 – TFM_Analysis.py***.

5.3.2. Correlación de variables

Una vez conocida la distribución de los datos, el siguiente punto es conocer qué relación tiene cada variable independiente con la variable objetivo y si existe diferencias significativas entre los grupos, de las que se puedan obtener conclusiones.

Partiendo del conjunto de datos reducido y siendo la variable objetivo de tipo categórica, se utilizará el estadístico de Chi-cuadrado para compararla con las variables independientes categóricas y evaluar si la hipótesis nula es cierta. Para esta investigación, la hipótesis nula consistirá en afirmar que todos los grupos de las distintas variables independientes presentarán el mismo nivel ecológico.

Para obtener resultados más realistas, se reducen los grupos de ciertas variables, incluyendo la variable objetivo. Haciendo esto, se pretende encontrar diferencias entre grupos que sean realmente significativas, pues es más sencillo encontrarlas cuando se disponen de muchas clases para ordenar los datos.

La variable objetivo "eco_tourist" se reduce a 2 grupos (No ecológico ⊂ Nada ecológico; Ecológico ⊂ Poco ecológico, Ecológico, Muy ecológico), y de forma general, se eliminan todos aquellos registros que tienen como respuesta NS (No sabe) en alguna de las variables a estudiar.

La primera variable independiente que presenta diferencias significativas entre sus grupos es la edad de los turistas, cuyos resultados pueden observarse en la Tabla 7. En esta variable también se reducen los grupos de la siguiente forma:

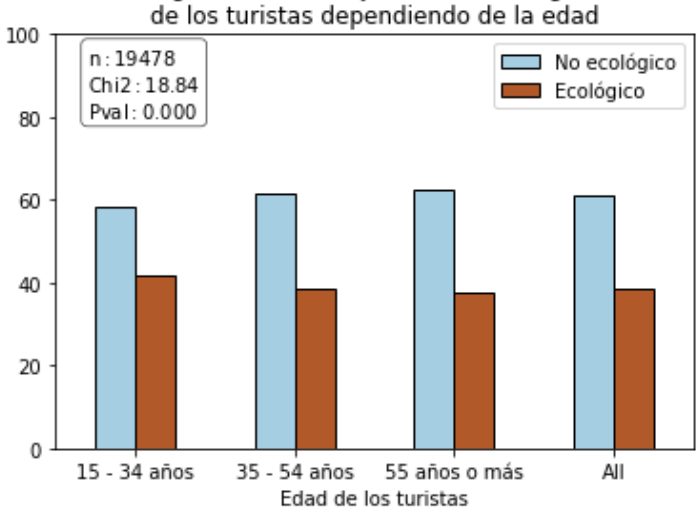
- 15 – 34 años ⊂ 15 – 24 años, 25 – 34 años
- 35 – 55 años ⊂ 35 – 44 años, 45 – 54 años
- 55 años o más ⊂ 55 – 64 años, Más de 65 años

Tabla 7. Nivel ecológico de los turistas dependiendo de la edad

		15 – 34 años	35 – 55 años	55 años o más	All
No ecológico	**%**	**58,20**	**61,40**	**62,40**	**61,30**
	N	*2133*	*4169*	*5642*	*11944*
Ecológico	**%**	**41,80**	**38,60**	**37,60**	**38,70**
	N	*1346*	*2629*	*3559*	*7534*
TOTAL	**%**	**100**	**100**	**100**	**100**
	N	*3479*	*6798*	*9201*	*19478*

Chi^2 = 18,842; p-value = 8,101e-05. Fuente: Elaboración propia

Figura 5. Porcentaje de nivel ecológico de los turistas dependiendo de la edad



Con respecto a la edad, con un *p-value* tan bajo, se puede rechazar la hipótesis nula que afirma que los diferentes grupos de edad son igual de ecológicos y se concluye que, de los turistas encuestados en 2016, las personas jóvenes (entre 15 y 34 años) presentan comportamientos ligeramente más ecológicos que el resto. Destacar también, que este grupo representa el 17,86% del total, lo cual es una cantidad a tener en cuenta.

Otra variable que presenta diferencias significativas entre sus grupos es la nacionalidad, cuyos resultados pueden observarse en la Tabla 8. En este caso no tiene sentido reducir los grupos ya que el objetivo es encontrar que turistas son más y menos ecológicos sus nacionalidades.

Tabla 8. Nivel ecológico de los turistas dependiendo de su nacionalidad

	No ecológico		Ecológico		TOTAL	
	%	*N*	**%**	*N*	**%**	*N*
Austria	**60,60**	*488*	**39,40**	*308*	**100**	*796*
Bélgica	**43,00**	*370*	**57,00**	*234*	**100**	*604*
Bulgaria	**69,40**	*341*	**30,60**	*215*	**100**	*556*
Croacia	**58,40**	*197*	**41,60**	*125*	**100**	*322*
Chipre	**59,20**	*227*	**40,80**	*143*	**100**	*370*
República Checa	**66,00**	*428*	**34,00**	*270*	**100**	*698*
Dinamarca	**73,60**	*488*	**26,40**	*308*	**100**	*796*
Estonia	**88,20**	*208*	**11,80**	*131*	**100**	*339*
Finlandia	**69,00**	*465*	**31,00**	*293*	**100**	*758*
Francia	**58,70**	*685*	**41,30**	*432*	**100**	*1117*
Alemania	**59,30**	*675*	**40,70**	*426*	**100**	*1101*
Grecia	**57,30**	*361*	**42,70**	*227*	**100**	*588*
Hungría	**65,80**	*342*	**34,20**	*216*	**100**	*558*
Islandia	**74,90**	*272*	**25,10**	*171*	**100**	*443*
Irlanda	**51,50**	*495*	**48,50**	*313*	**100**	*808*
Italia	**61,90**	*591*	**38,10**	*373*	**100**	*964*
Letonia	**62,10**	*167*	**37,90**	*105*	**100**	*272*
Lituania	**55,70**	*172*	**44,30**	*108*	**100**	*280*
Luxemburgo	**43,70**	*234*	**56,30**	*148*	**100**	*382*
Macedonia	**71,10**	*144*	**28,90**	*91*	**100**	*235*

Malta	**59,40**	*174*	**40,60**	*109*	**100**	*283*
Moldavia	**50,70**	*91*	**49,30**	*57*	**100**	*148*
Montenegro	**60,10**	*177*	**39,90**	*111*	**100**	*288*
Polonia	**62,90**	*567*	**37,10**	*357*	**100**	*924*
Portugal	**65,50**	*283*	**34,50**	*178*	**100**	*461*
Rumanía	**52,00**	*251*	**48,00**	*159*	**100**	*410*
Eslovaquia	**58,20**	*380*	**41,80**	*240*	**100**	*620*
Eslovenia	**75,80**	*218*	**24,20**	*137*	**100**	*355*
España	**63,60**	*597*	**36,40**	*377*	**100**	*974*
Suecia	**73,80**	*489*	**26,20**	*309*	**100**	*798*
Holanda	**75,00**	*459*	**25,00**	*289*	**100**	*748*
Turquía	**17,90**	*267*	**82,10**	*168*	**100**	*435*
Reino Unido	**51,00**	*642*	**49,00**	*405*	**100**	*1047*
All	**61,30**	*11945*	**38,70**	*7533*	**100**	*19478*

Chi^2 = 993,318; p-value = 4,379e-188. Fuente: Elaboración propia

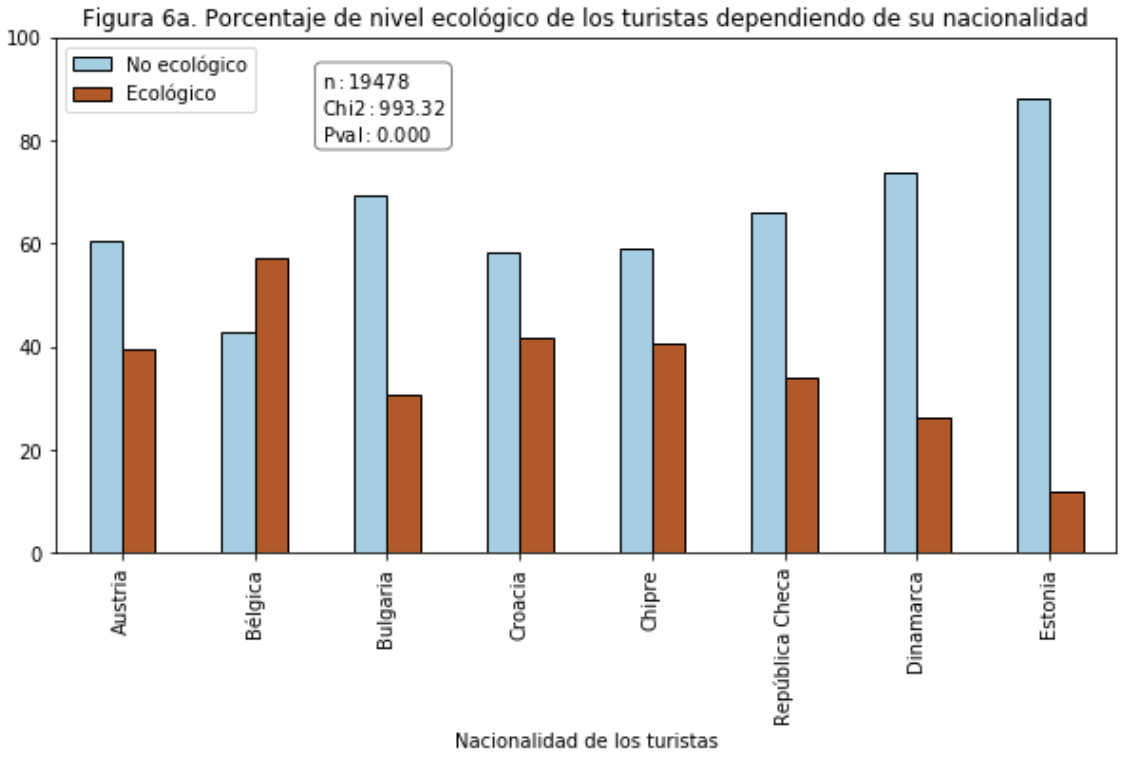


Figura 6b. Porcentaje de nivel ecológico de los turistas dependiendo de su nacionalidad

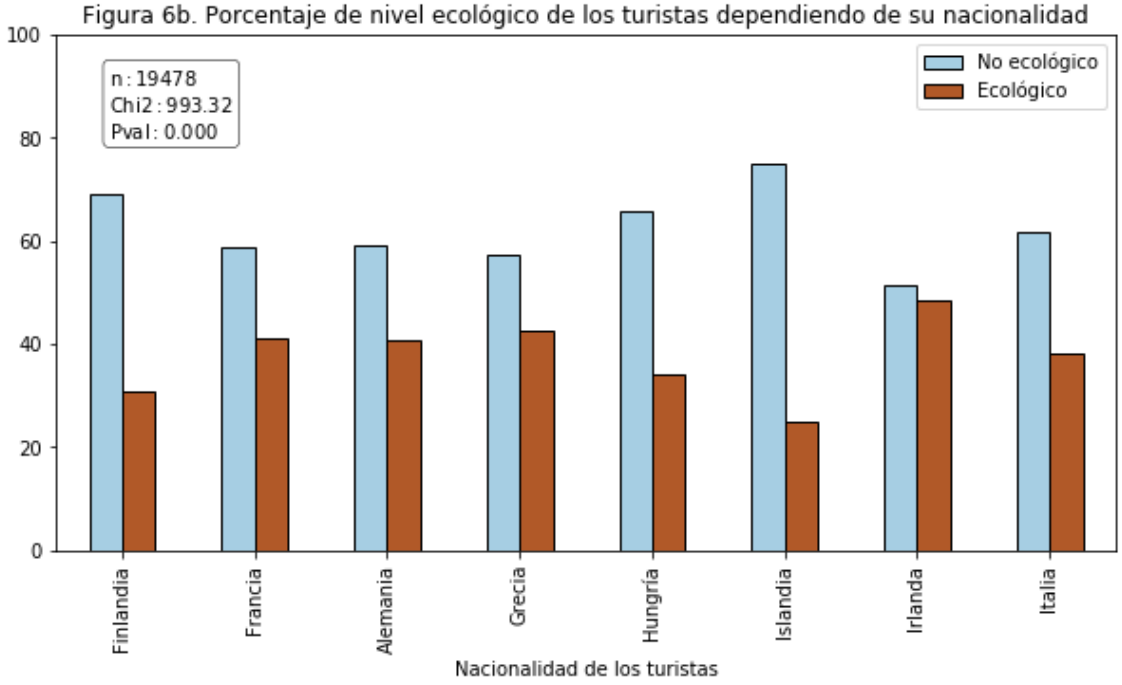


Figura 6c. Porcentaje de nivel ecológico de los turistas dependiendo de su nacionalidad

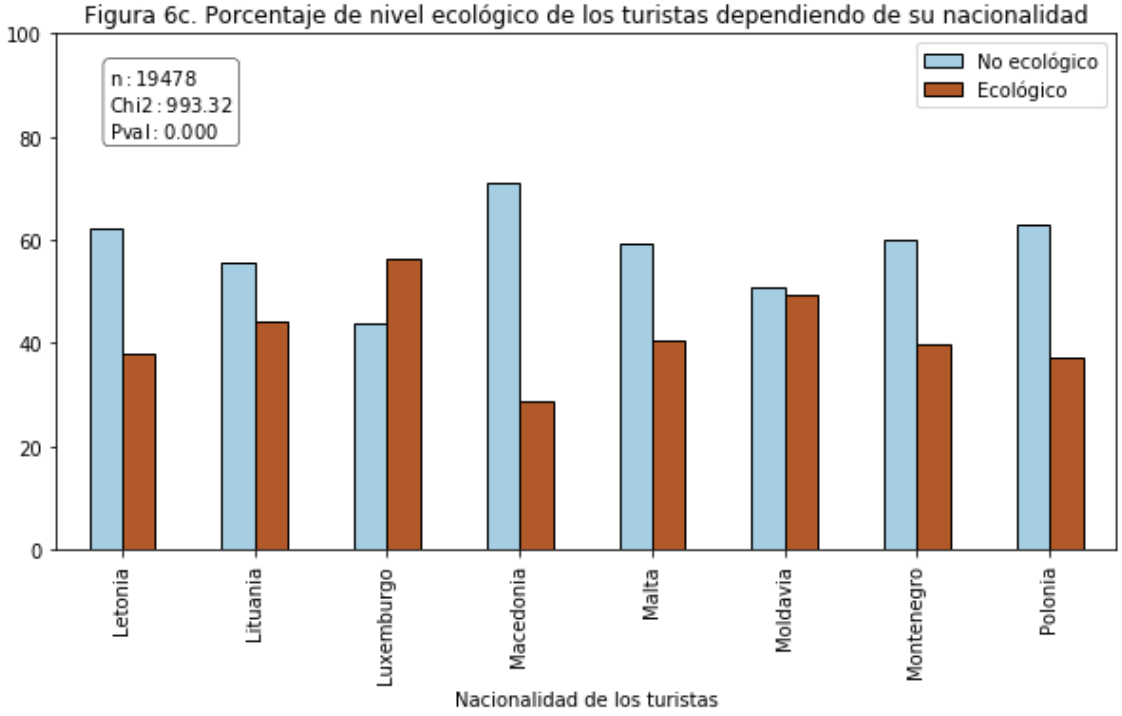
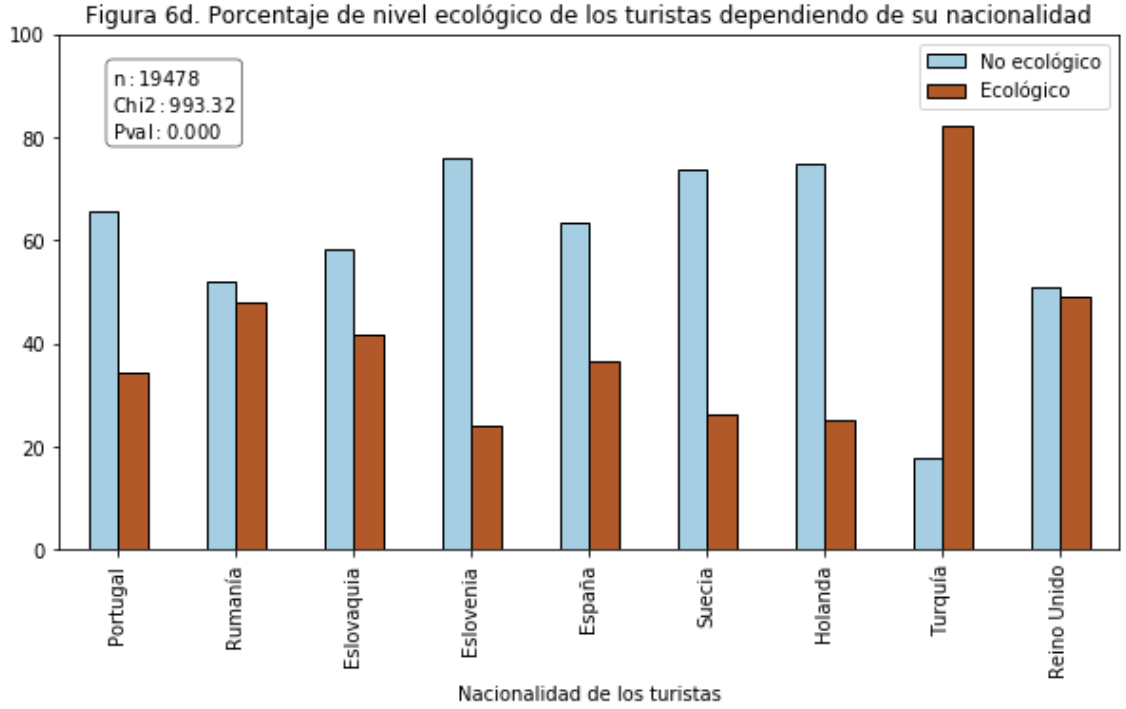


Figura 6d. Porcentaje de nivel ecológico de los turistas dependiendo de su nacionalidad



En este caso, el *p-value* aún es menor que con la edad de los turistas, por tanto, también se puede rechazar la hipótesis nula. Como se puede comprobar en los resultados se encuentran diferencias muy significativas entre los turistas de distintas nacionalidades. Destacar positivamente el alto nivel ecológico de los turistas turcos, belgas y luxemburgueses, y negativamente a los turistas estonios, eslovenos, holandeses, islandeses, suecos, daneses y macedonios (entre otros tantos).

Para estudiar el nivel ecológico de los turistas en base a las veces que viajan en distintas duraciones, se reducen los grupos de las variables *long_vac_cat, med_vac_cat* y *short_vac_cat* de la siguiente manera:

- Ninguna ⊂ Ninguna
- Pocas ⊂ 1 vez, 2 veces, 3 veces, 4 o 5 veces
- Bastantes ⊂ De 6 a 10 veces
- Muchas ⊂ Más de 10 veces

Los resultados significativos correspondientes a vacaciones de larga y media duración se presentan en la Tabla 9 y en la Tabla 10 respectivamente.

Tabla 9. Nivel ecológico de los turistas dependiendo de la cantidad de veces que realizaron viajes de larga duración

		Ninguna	Pocas	Bastantes	Muchas	All
No ecológico	**%**	**63,70**	**58,40**	**45,90**	**42,40**	**61,30**
	N	*7090*	*4653*	*112*	*88*	*11943*
Ecológico	**%**	**36,30**	**41,60**	**54,10**	**57,60**	**38,70**
	N	*4473*	*2935*	*71*	*56*	*7535*
TOTAL	**%**	**100**	**100**	**100**	**100**	**100**
	N	*11563*	*7588*	*183*	*144*	*19478*

Chi^2 = 94,291; p-value = 2,622e-20. Fuente: Elaboración propia

Figura 7. Porcentaje de nivel ecológico de los turistas dependiendo de la cantidad de viajes de larga duración

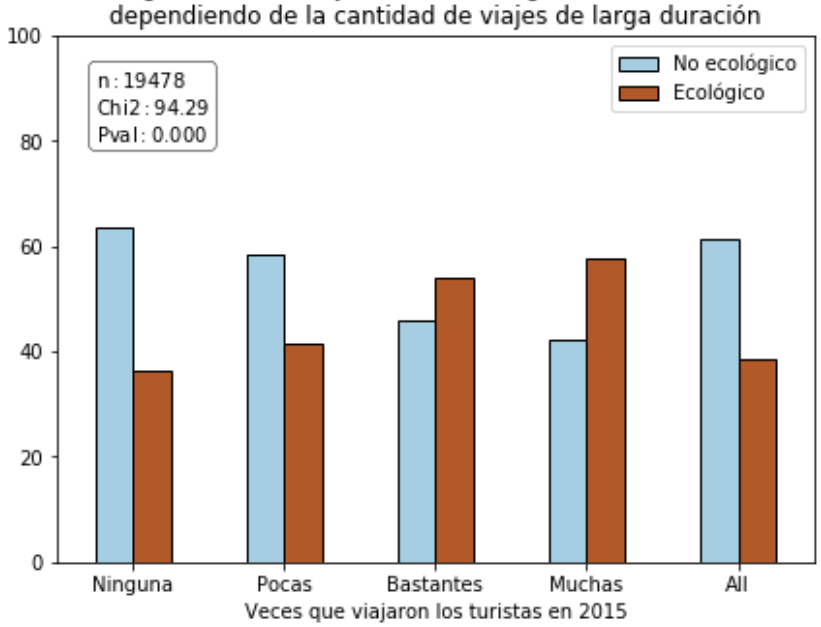
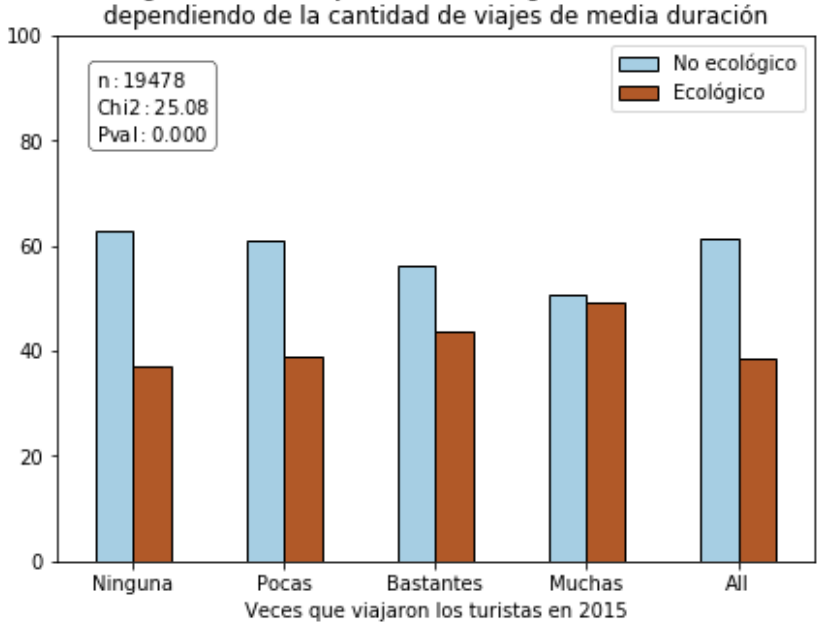


Tabla 10. Nivel ecológico de los turistas dependiendo de la cantidad de veces que realizaron viajes de media duración

		Ninguna	Pocas	Bastantes	Muchas	All
No ecológico	**%**	**63,00**	**61,10**	**56,30**	**50,70**	**61,30**
	N	*3306*	*8069*	*430*	*139*	*11944*
Ecológico	**%**	**37,00**	**38,90**	**43,70**	**49,30**	**38,70**
	N	*2086*	*5089*	*271*	*88*	*7534*
TOTAL	**%**	**100**	**100**	**100**	**100**	**100**
	N	*5392*	*13158*	*701*	*227*	*19478*

Chi^2 = 25,083; p-value = 1,484e-05. Fuente: Elaboración propia

Figura 8. Porcentaje de nivel ecológico de los turistas dependiendo de la cantidad de viajes de media duración



Gráficamente se puede observar como aquellos turistas que realizan más de 6 viajes de larga y media duración son más ecológicos que el resto, aunque estos perfiles no abundan ya que sólo representan el 1,68% y el 4,76% respectivamente del total para cada caso. Comentar también que en el caso de vacaciones de corta duración no existen diferencias significativas entre los grupos, y los turistas presentan mayoritariamente comportamientos no ecológicos independientemente de la cantidad de veces que realice este tipo de vacaciones.

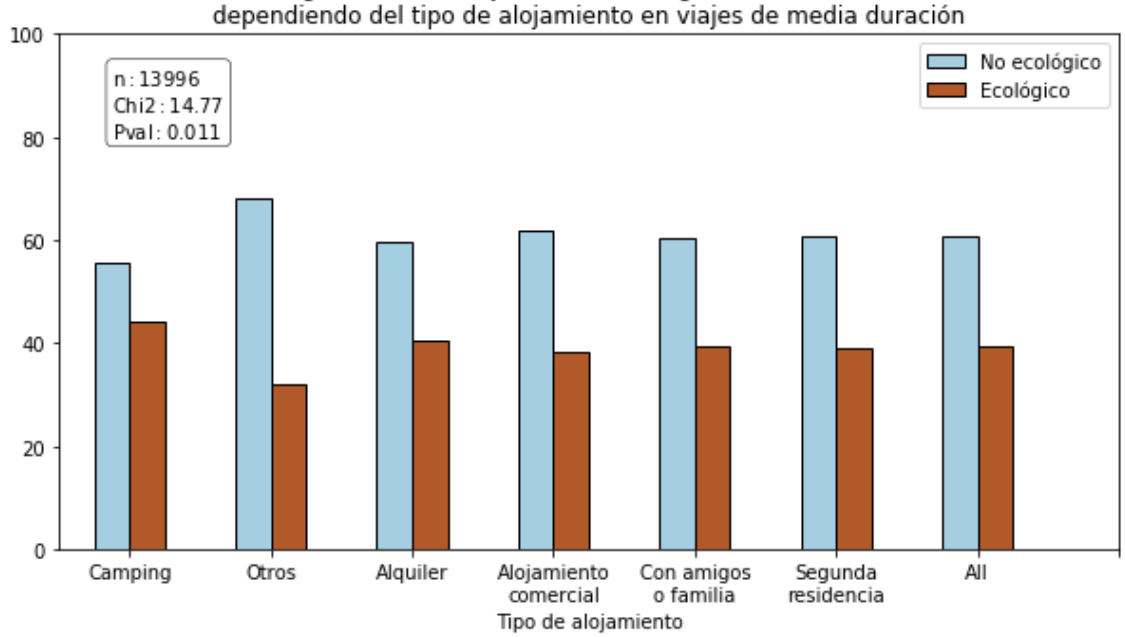
El tipo de alojamiento también presenta diferencias significativas para vacaciones de media y corta duración. Los resultados pueden observarse en la Tabla 11 y en la Tabla 12 respectivamente.

Tabla 11. Nivel ecológico de los turistas dependiendo del tipo de alojamiento en sus viajes de media duración

		Camping	Otros	Alquiler	Alojamiento comercial	Con amigos o familia	Segunda residencia	All
No ecológico	**%**	**55,80**	**68,00**	**59,50**	**61,70**	**60,40**	**60,80**	**60,70**
	N	*470*	*76*	*1359*	*4347*	*1659*	*588*	*8499*
Ecológico	**%**	**44,20**	**32,00**	**40,50**	**38,30**	**39,60**	**39,20**	**39,30**
	N	*303*	*49*	*879*	*2811*	*1073*	*382*	*5497*
TOTAL	**%**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	N	*773*	*125*	*2238*	*7158*	*2732*	*970*	*13996*

Chi^2 = 14,774 p-value = 0,011. Fuente: Elaboración propia

Figura 9. Porcentaje de nivel ecológico de los turistas dependiendo del tipo de alojamiento en viajes de media duración



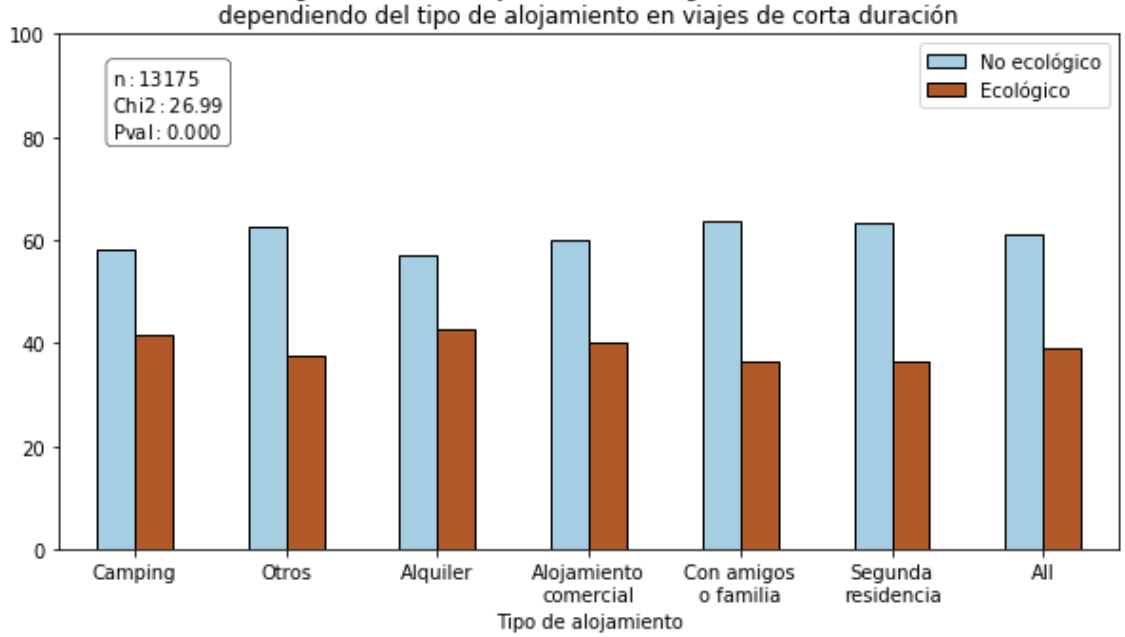
Para vacaciones de media duración, aquellos turistas que se alojan en camping son ligeramente más ecológicos que el resto. Este grupo representa un 5,52% del total.

Tabla 12. Nivel ecológico de los turistas dependiendo del tipo de alojamiento en sus viajes de corta duración

		Camping	Otros	Alquiler	Alojamiento comercial	Con amigos o familia	Segunda residencia	All
No ecológico	**%**	**58,30**	**62,50**	**57,10**	**60,00**	**63,60**	**63,40**	**61,00**
	N	*346*	*88*	*774*	*3681*	*2562*	*592*	*8043*
Ecológico	**%**	**41,70**	**37,50**	**42,90**	**40,00**	**36,40**	**36,60**	**39,00**
	N	*220*	*56*	*494*	*2349*	*1635*	*378*	*5132*
TOTAL	**%**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	N	*566*	*144*	*1268*	*6030*	*4197*	*970*	*13175*

Chi^2 = 26,989; p-value = 5,732e-05. Fuente: Elaboración propia

Figura 10. Porcentaje de nivel ecológico de los turistas dependiendo del tipo de alojamiento en viajes de corta duración



Para vacaciones de corta duración, aquellos turistas que se alojan en campings o alquileres privados son ligeramente más ecológicos que el resto, y representan el 4,29% y el 9,62% respectivamente. Por otra parte, aquellos que se alojan con amigos o familiares o en segundas residencias presentan los comportamientos menos ecológicos, y representan el 31,86% y el 7,36% respectivamente.

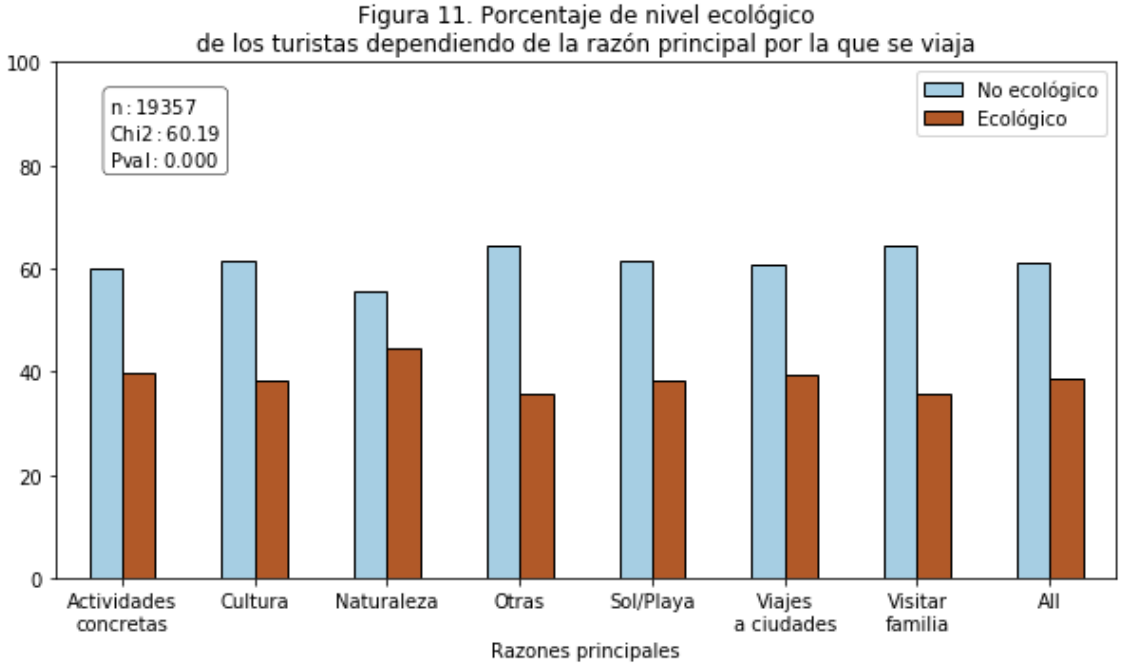
La razón principal por la que deciden ir de vacaciones es también una variable con diferencias significativas, cuyos resultados se pueden observar en la Tabla 13. Para facilitar su estudio se reducen los grupos de la siguiente forma:

- Actividades concretas ⊂ Eventos concretos, Actividades deportivas, Terapia salud
- Cultura ⊂ Cultura
- Naturaleza ⊂ Naturaleza
- Otras ⊂ Otras
- Sol/Playa ⊂ Sol/Playa
- Viajes a ciudades ⊂ Viajes a ciudades
- Visitar familia ⊂ Visitar familia

Tabla 13. Nivel ecológico de los turistas dependiendo de la razón principal por la que viajan

		Actividades concretas	Cultura	Naturaleza	Otras	Sol/Playa	Viajes a ciudades	Visitar familia	All
No ecológico	%	**60,10**	**61,60**	**55,50**	**64,40**	**61,50**	**60,60**	**64,30**	**61,30**
	N	*1802*	*1119*	*1547*	*752*	*2783*	*1056*	*2801*	*11860*
Ecológico	%	**39,90**	**38,40**	**44,50**	**35,60**	**38,50**	**39,40**	**35,70**	**38,70**
	N	*1139*	*707*	*978*	*475*	*1759*	*668*	*1771*	*7497*
TOTAL	%	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	N	*2941*	*1826*	*2525*	*1227*	*4542*	*1724*	*4572*	*19357*

Chi^2 = 60,186 p-value = 4,126e-11. Fuente: Elaboración propia



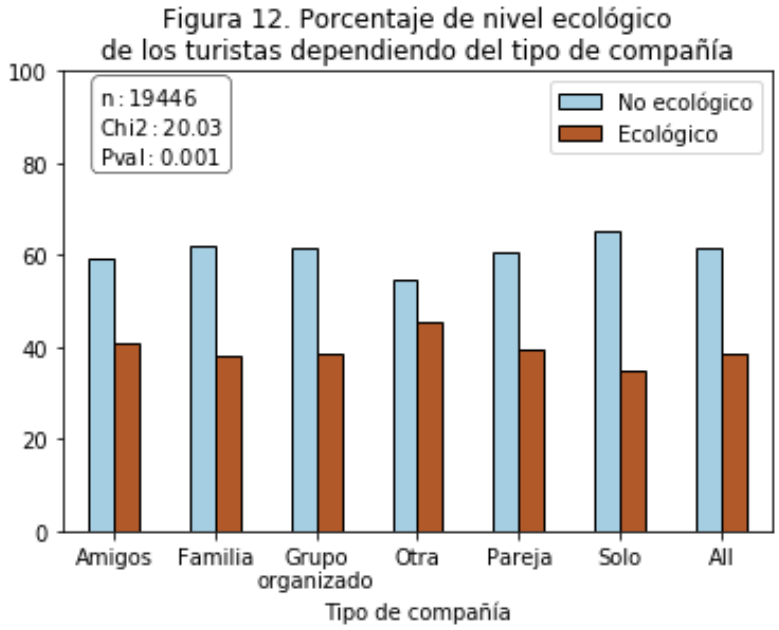
En este caso, y como era de esperar, aquellos turistas cuya razón principal para ir de vacaciones fue la naturaleza (montaña, entorno natural, etc.), presentan comportamientos más ecológicos que el resto, aunque este grupo sólo representa el 2,71% del total.

Por último, también se estudia el nivel ecológico de los turistas dependiendo del tipo de compañía que disfrutaron en sus vacaciones principales. Los resultados se pueden observar en la Tabla 14.

Tabla 14. Nivel ecológico de los turistas dependiendo del tipo de compañía que disfrutaron en sus vacaciones

		Amigos	Familia	Grupo organizado	Otra	Pareja	Solo	All
No ecológico	%	**59,40**	**61,80**	**61,50**	**54,50**	**60,40**	**65,30**	**61,30**
	N	*1305*	*4727*	*419*	*69*	*4335*	*1064*	*11919*
Ecológico	%	**40,60**	**38,20**	**38,50**	**45,50**	**39,60**	**34,70**	**38,70**
	N	*824*	*2986*	*264*	*43*	*2738*	*672*	*7527*
TOTAL	%	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	N	*2129*	*7713*	*683*	*112*	*7073*	*1736*	*19446*

Chi^2 = 20,031 p-value = 0,001. Fuente: Elaboración propia



En este caso el *p-value* también nos indica que existen diferencias significativas. De los resultados y del gráfico se puede extraer que aquellos turistas que viajan solos son ligeramente menos ecológicos que el resto. Este grupo representa un 8,93% del total.

El resto de tablas de contingencia y gráficos correspondientes a aquellas variables que no presentan diferencias significativas en sus grupos con respecto al nivel ecológico, se pueden consultar en el ***Anexo 2 – Tablas y gráficos correlación***. El código para realizar este trabajo de análisis de correlaciones se puede encontrar en el ***Anexo 5 – TFM_Correlation.py***.

6. Metodología

Una vez conocidas y entendidas qué variables tienen más efecto sobre el nivel ecológico de los turistas europeos, el siguiente paso es realizar un modelo que permita ser proactivo y adelantarse a los hechos. Un modelo de predicción es un algoritmo numérico que utiliza una serie de variables independientes o predictoras para que, a partir de los datos, obtener un valor en la salida o variable objetivo. Estos modelos de predicción, de forma sencilla, pueden dividirse en dos grupos:

- Modelos de clasificación – a la salida del modelo se obtiene la probabilidad de pertenecer a una clase o grupo.
- Modelos de regresión – a la salida del modelo se obtiene un valor.

Para la investigación, se pretende utilizar un modelo de clasificación que permita predecir, a partir de datos distintos a los aspectos ecológicos, si un turista será o no ecológico en sus próximas vacaciones. Para ello, se utilizarán técnicas de *Machine Learning* programadas en lenguaje Python, siguiendo el estándar marcado durante toda la investigación, con el fin de adaptarse a las nuevas demandas tecnológicas en el mundo del dato.

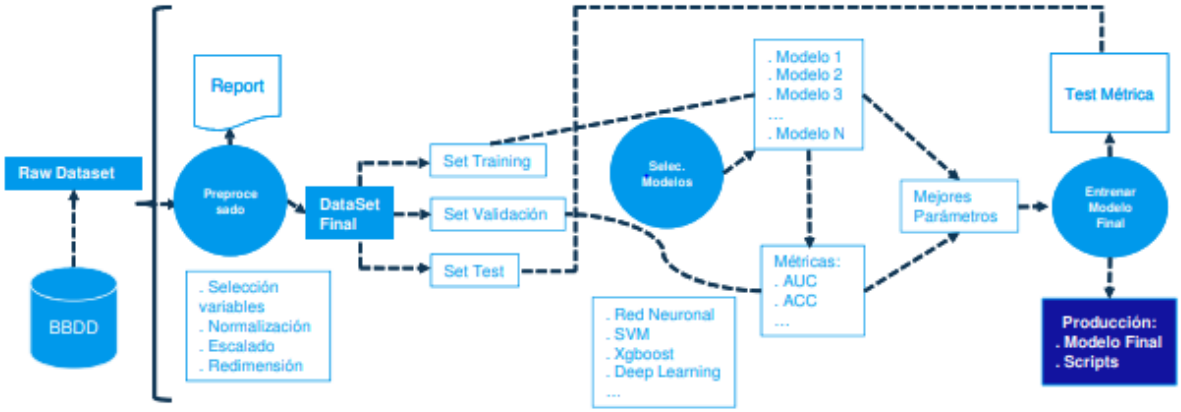
El *Machine Learning* o Aprendizaje Automático, tal y como indica su propio nombre, es una rama de la Inteligencia Artificial que desarrolla técnicas para construir sistemas, que permiten a los ordenadores aprender, de forma automática, a partir de los datos. Aunque existe cierta incertidumbre acerca de la posibilidad de que el aprendizaje automático elimine por completo la necesidad de conocimiento experto humano, es importante destacar que la intuición humana nunca podrá ser reemplazada en su totalidad, ya que el tratamiento y preparación de los datos depende de ello. El objetivo fundamental de la Inteligencia Artificial no es otro que ser una herramienta que complemente todas las acciones humanas.

Dentro del *Machine Learning* existen dos tipos de aprendizaje: el aprendizaje supervisado y el aprendizaje no-supervisado. El primero, el cual abarca la mayoría de los problemas, es aquel tipo de aprendizaje en el que se utiliza un conjunto de datos etiquetados, es decir, en el que existe una variable objetivo definida para predecir. En el aprendizaje no-supervisado se tiene un conjunto de datos formado simplemente por entradas, sin información que permita clasificar en categorías de manera directa los ejemplos de los que se dispone.

Para cualquier problema de predicción de aprendizaje supervisado, antes de empezar a entrenar el algoritmo, se debe trabajar en los datos tal y como se ha hecho en la parte de preprocesamiento y análisis, ya que entender los datos te permitirá enfocar mejor la elección de los algoritmos a utilizar. El siguiente paso es la modelización, que consiste en la prueba de varios modelos para finalmente utilizar el mejor de ellos. Para hacer esto, el conjunto de datos debe dividirse en 3 grupos: conjunto de entrenamiento, conjunto de validación y conjunto de test (normalmente se distribuye en un 70%, un 15% y un 15% del conjunto de datos total). Durante la modelización, se utiliza el mismo conjunto de entrenamiento con diferentes modelos y se utiliza el conjunto de validación para comprobar el rendimiento de cada uno de ellos en base a una serie de métricas (ROC, AUC, etc.). Una vez se conoce qué modelo funciona mejor para ese caso de uso en concreto, se debe entrenar en la última etapa utilizando el conjunto de test. Para ello se predice la variable objetivo y se comparan los resultados obtenidos con los

valores etiquetados para cada registro. En la Figura 13 se puede observar esta explicación de manera esquematizada.

Figura 13. Esquema proceso de creación modelo de predicción



Debe prestarse especial atención a lo que se conoce como sobreajuste, ya que no se quiere que el modelo aprenda perfectamente todos los casos, pues de esta forma no es capaz de generalizar y cuando le aparecen datos nuevos no sabe interpretarlos correctamente. También destacar la importancia de el reentrenamiento de modelos, ya que el modelo debe seguir aprendiendo de datos nuevos. Los datos cambian de forma continua y las tendencias lo hacen de la misma manera, por lo que con un modelo anticuado que se entrenó una sola vez se tiene un modelo que predice valores erróneos, propios de la época en la que se entrenó.

Dentro del aprendizaje supervisado, se pueden clasificar los modelos en tres grupos:

- Modelos lineales
  - Regresión lineal – utilizada para problemas de regresión continua.
  - Regresión logística – utilizada para problemas de clasificación.
- Modelos de árboles
  - Árboles de decisión – con arquitecturas internas muy sencillas que dan solución a problemas de poca complejidad.
  - Random Forest – combinación de varios árboles de decisión en paralelo.
  - XGBoost – combinación de varios árboles de decisión en secuencia.
- Modelos complejos
  - SVM – utilizado tanto para clasificación como para regresión, con un coste computacional elevado.
  - Redes Neuronales – utilizadas tanto para clasificación como para regresión. Ya están consideradas como modelos de Deep Learning.

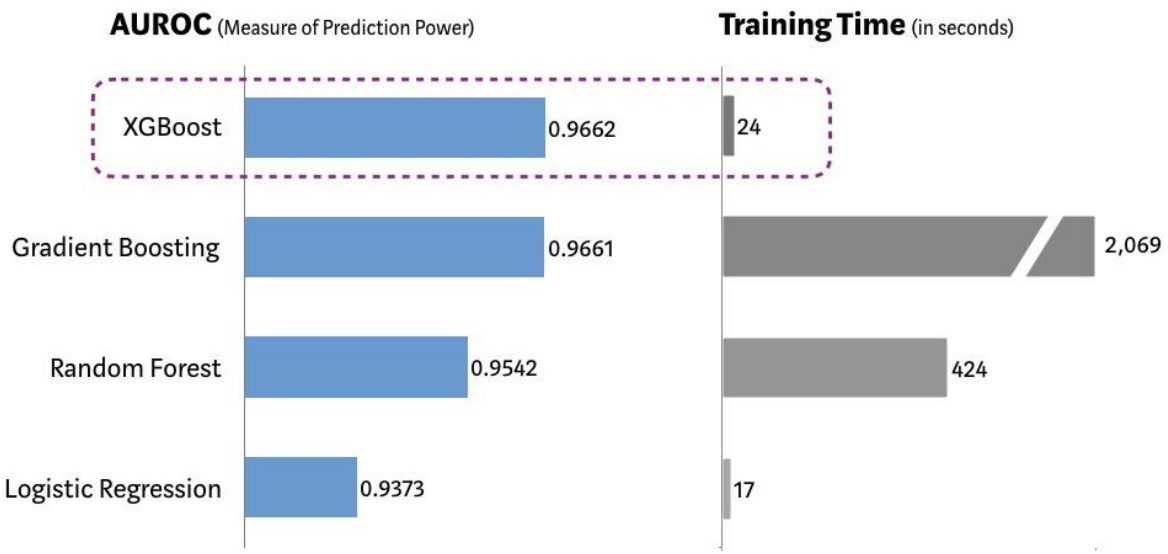
6.1. Modelo XGBoost

Dependiendo del caso de uso y de los datos, se debe utilizar un algoritmo u otro. El mismo problema, las limitaciones de hardware y la exigencia en los resultados de predicción serán los que marcarán las decisiones. Normalmente los modelos basados en árboles o los complejos son

los que mejores resultados dan al usuario, aunque requieren de unos costes computacionales elevados. En algunos casos los modelos lineales simples ya darán resultados válidos, y será preferible utilizarlos ya que, aunque con otros modelos superiores se puedan obtener resultados mejores, el tiempo de entrenamiento será mucho menor.

En 2019, Vishal Morde (actual Jefe de Ciencia de Datos en Uber) realizó un estudio en el que, para un problema de clasificación, se comparaba el rendimiento de varios modelos basados en algoritmos distintos. Para que el estudio no estuviese sesgado y se utilizasen datos que fuesen a funcionar mejor en un modelo que en otro, se creó un conjunto de datos aleatorios con 20 variables y 1 millón de muestras etiquetadas también de forma aleatoria. Los resultados, tal y como se observa en la Figura 14, demostraron que, de manera generalizada, el modelo de predicción que mejor funciona en problemas de clasificación actualmente es el basado en el algoritmo XGBoost. Es por ello que su uso está tan extendido en las competiciones de Data Science recientes.

Figura 14. Comparativa entre modelos de clasificación para un caso generalizado



Basándonos en los resultados de Vishal Morde, y en base a experiencia propia en otros proyectos de Machine Learning, se selecciona el algoritmo XGBoost de clasificación para la creación del modelo de predicción del nivel ecológico de los turistas europeos.

XGBoost es un algoritmo de Machine Learning basado en un árbol de decisiones que utiliza una estructura de *gradient boosting*. En los problemas de predicción que involucran datos no estructurados (imágenes, texto, etc.), las redes neuronales artificiales tienden a superar a todos los demás algoritmos. Sin embargo, cuando se trata de datos estructurados, como se ha podido comprobar, los algoritmos basados en el árbol de decisión se consideran los mejores en su clase en este momento.

Los árboles de decisión, en su forma más simple, son fáciles de visualizar e interpretar. Entre los distintos algoritmos basados en árboles de decisión se encuentran los siguientes:

- Decision tree – es un modelo en forma de árbol donde cada nodo representa las variables, cada rama una decisión y cada nodo hoja una salida.
- Bagging – es un modelo que recoge un conjunto de varios Decision Tree, todos con el mismo peso en la votación de la predicción final.
- Random Forest – es un modelo compuesto de igual manera que el algoritmo Bagging pero con la diferencia de que para cada Decision Tree se selecciona una serie de variables elegidas al azar.
- Boosting – es un modelo que recoge un conjunto de varios Decision Tree, aunque los distribuye de forma secuencial. Cada uno de ellos aprende de los errores del anterior.
- Gradient Boosting – es un caso especial del modelo Boosting en el que los errores son minimizados utilizando un algoritmo *gradient descent*.
- XGBoost – es un modelo Gradient Boosting mejorado. Es una combinación perfecta de técnicas de optimización de software y hardware para producir resultados superiores utilizando menos recursos informáticos en el menor tiempo posible.

El algoritmo XGBoost fue desarrollado como un proyecto de investigación en la Universidad de Washington. Tianqi Chen y Carlos Guestrin presentaron su artículo en la Conferencia SIGKDD en 2016 e impresionaron al mundo del Machine Learning. Desde su introducción, este algoritmo no solo ha sido acreditado por ganar numerosas competiciones de Kaggle, sino también por ser la fuerza impulsora bajo el capó para varias aplicaciones industriales de vanguardia. Como resultado, existe una fuerte comunidad de científicos de datos que contribuyen a los proyectos de código abierto de XGBoost con alrededor de 350 colaboradores y unos 3.600 *commit* en GitHub. El algoritmo se diferencia del resto de las siguientes maneras:

- Una amplia gama de aplicaciones: se puede utilizar para resolver problemas de regresión, clasificación, clasificación y predicciones definidas por el usuario.
- Portabilidad: se ejecuta sin problemas en Windows, Linux y OS X.
- Lenguajes: admite todos los principales lenguajes de programación, incluidos C ++, Python, R, Java, Scala y Julia.
- Integración en la nube: admite clústeres AWS, Azure y Yarn y funciona bien con Flink, Spark y otros ecosistemas.

Las técnicas de optimización de software y hardware con las que consigue resultados tan buenos son las siguientes:

1. Paralelización: XGBoost aborda el proceso de construcción secuencial de árboles mediante la implementación en paralelo. Esto es posible debido a la naturaleza intercambiable de los bucles utilizados para construir aprendices base; el bucle externo que enumera los nodos de hoja de un árbol y el segundo bucle interno que calcula las características. Este anidamiento de bucles limita la paralelización porque sin completar el bucle interno (más exigente computacionalmente de los dos), el bucle externo no puede iniciarse. Por lo tanto, para mejorar el tiempo de ejecución, el orden de los bucles se intercambia mediante la inicialización a través de un análisis global de todas las instancias y la clasificación mediante subprocesos paralelos.

2. Poda de árboles: XGBoost utiliza el parámetro "max_depth" como se especifica en lugar del criterio primero, y comienza a podar los árboles hacia atrás. Este enfoque de "profundidad primero" mejora significativamente el rendimiento computacional.

3. Optimización de hardware: este algoritmo ha sido diseñado para hacer un uso eficiente de los recursos de hardware. Esto se logra mediante el reconocimiento de caché mediante la asignación de búferes internos en cada subproceso para almacenar estadísticas de gradiente. Otras mejoras, como la informática "fuera del núcleo", optimizan el espacio disponible en el disco al tiempo que manejan grandes marcos de datos que no caben en la memoria.

4. Regularización: penaliza los modelos más complejos mediante la regularización LASSO (L1) y Ridge (L2) para evitar el sobreajuste.

5. Croquis cuantitativo ponderado: XGBoost emplea el algoritmo de croquis cuantitativo ponderado distribuido para encontrar efectivamente los puntos de división óptimos entre los conjuntos de datos ponderados.

6. Validación cruzada: el algoritmo viene con un método de validación cruzada incorporado en cada iteración, eliminando la necesidad de programar explícitamente esta búsqueda y especificar el número exacto de iteraciones de refuerzo requeridas en una sola ejecución (Towards Data Science, 2019).

7. Resultados

En este apartado se presentan los resultados de dos modelos de Machine Learning creados con algoritmos XGBoost. El primer modelo predice el nivel ecológico de los turistas y los clasifica en dos grupos: turistas ecológicos y turistas no ecológicos. El segundo modelo clasifica a los turistas en los cuatro grupos vistos anteriormente.

Para cualquiera de los dos casos, se utiliza el conjunto de datos reducido, aunque se prescinde de aquellas variables que, como en la parte de análisis se ha podido ver, no presentan diferencias significativas en relación a la variable objetivo, además de las variables correspondientes a los aspectos ecológicos, ya que si se incluyesen se estarían manipulando los resultados porque contienen información directamente relacionada con la variable objetivo que se quiere predecir.

Una vez se tiene el conjunto de datos organizado, el siguiente paso es la conversión de todas las variables categóricas de las que se disponga en variables numéricas *dummies*. Esta fase es estrictamente necesaria para trabajar con algoritmos de Machine Learning, ya que son algoritmos numéricos e internamente no reconocen las clases. Esto supone un aumento considerable de las variables, ya que la gran mayoría de ellas en el conjunto de datos son variables categóricas.

Cuando se tiene todo el conjunto de datos formado por variables numéricas, lo siguiente es separar las características del objetivo. Todas aquellas variables que ayudan al modelo a clasificar y predecir la variable objetivo, son consideradas como características. Además, ambas cosas deben dividirse, tal y como se ha visto en la metodología, en conjunto de entrenamiento y conjunto de test y validación. En este caso de uso concreto, se ha destinado el 80% de las muestras del conjunto de datos para el conjunto de entrenamiento, y el 20% restante equivale al conjunto de test. Esta división se ha realizado de manera totalmente aleatoria, con el fin, de nuevo, de no intervenir en los resultados del modelo. Con esto, se tiene un conjunto de entrenamiento (comúnmente llamado *X_train*) que contiene todas las variables que utilizará el modelo para entrenarse y un conjunto de clasificación de entrenamiento (comúnmente llamado *y_train*) que incluye las clases correspondientes para cada muestra. Como se ha explicado en la metodología, el modelo se entrena con *X_train* y compara los resultados de clasificación obtenidos con los reales de *y_train*, de esta manera puede comparar y calcular los errores para mejorar en las siguientes iteraciones. Por último, también se dispone de un conjunto de test (comúnmente llamado *X_test*) que contiene todas las variables que utilizará el modelo ya entrenado para predecir las clases, y un conjunto de clasificación de test (comúnmente llamado *y_test*) que utiliza el modelo entrenado para comparar con las predicciones y evaluar su rendimiento final ante la entrada de nuevos datos nunca vistos.

Disponiendo de los diferentes conjuntos de datos, el siguiente paso es realizar un escalado de *X_train* y *X_test* con media 0 y desviación típica 1. En el caso de las variables numéricas, se pueden tener valores muy diferentes en cuanto a magnitud, dependiendo de la unidad de medida que se utilice para cada una de ellas. Para anular dichas diferencias se debe realizar el escalado, con el que se igualan los pesos de los valores sean excesivamente altos o bajos.

7.1. Modelo 2 clases

El modelo de 2 clases es más sencillo en términos computacionales, así que se utiliza para comparar el rendimiento de 5 modelos distintos y comprobar si realmente el modelo basado en el algoritmo XGBoost es el que mejor funciona para nuestro caso de uso concreto.

Se comparan los siguientes modelos de clasificación de Machine Learning:

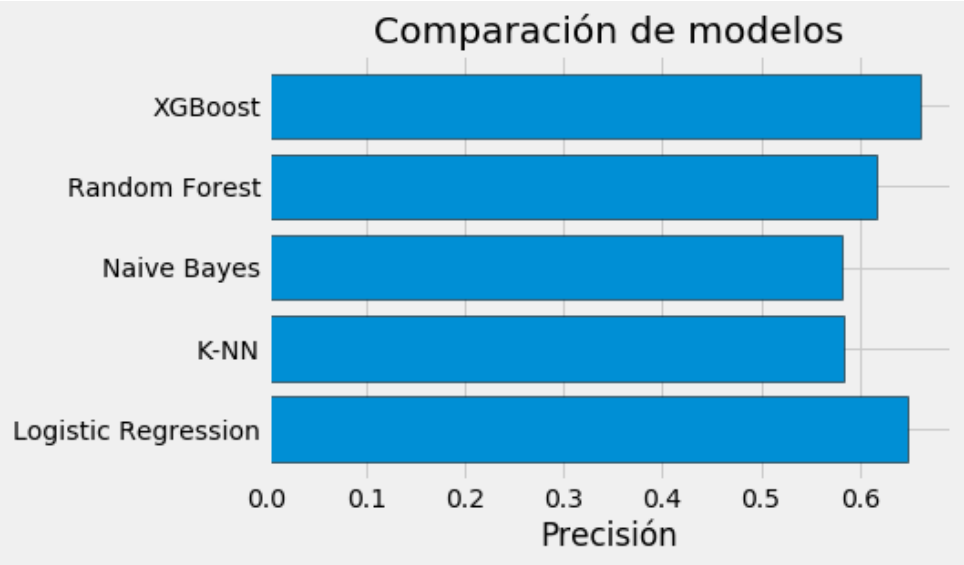
1. Logistic Regression
2. K-Nearest Neighbors
3. Naive Bayes
4. Random Forest
5. XGBoost

Para medir el rendimiento de cada modelo se utiliza la precisión, la cual indica el porcentaje de aciertos conseguidos por parte del modelo para las predicciones realizadas. A continuación, en la Tabla 15 se pueden observar los resultados de rendimiento para cada modelo.

Tabla 15. Precisión por modelo de clasificación de Machine Learning

Modelo	Precisión (%)
Logistic Regression	64,78
K-Nearest Neighbors	58,41
Naive Bayes	58,12
Random Forest	61,76
XGBoost	66,04

Figura 15. Comparación de modelos de clasificación para este caso de uso



Para este caso de uso, el modelo basado en el algoritmo XGBoost es el que mejor funciona ya que, con los parámetros establecidos por defecto, ofrece la mayor precisión a la hora de clasificar a los turistas en sus predicciones. El hecho de que la Regresión Logística le siga bastante de cerca, se debe a que no se tiene un número excesivamente elevado de muestras. En el supuesto de que se pudiera disponer de conjuntos de datos de años anteriores con los que poder ampliar el número de registros, se vería como la precisión del XGBoost y Random Forest mejorarían, ya que tendrían más datos de los que aprender, pero la precisión de la Regresión Logística apenas cambiaría, debido a que es un algoritmo sencillo que no está pensado para trabajar con problemas complejos o con grandes cantidades de datos.

En base a los resultados obtenidos en la comparación, se elige el modelo XGBoost como modelo para resolver el problema de predicción del nivel ecológico de los turistas europeos. Para mejorar el resultado de precisión, bien se puede alimentar y entrenar el modelo con más datos, o ajustar los parámetros internos del mismo. Como ya se ha comentado, la primera opción no puede llevarse a cabo ya que los cuestionarios de años anteriores son diferentes y no incluyen preguntas acerca de los aspectos ecológicos de los turistas. En cambio, la segunda opción siempre puede implementarse.

Cuando se quieren modificar los parámetros de un modelo, no se sabe con exactitud si los valores que se definan afectarán positiva o negativamente al rendimiento del mismo. La solución a este problema pasa por la creación de una rejilla que recoja los valores que se desean probar para cada parámetro, los cuales serán utilizados, combinándolos de todas las formas posibles, en múltiples iteraciones. Una de todas las iteraciones, recogerá los valores óptimos para cada parámetro, que definirá la configuración que mejor resultado de precisión proporcione.

En la Tabla 16 se pueden ver aquellos parámetros del modelo XGBoost que se han llevado a evaluación con diferentes valores.

Tabla 16. Configuración de valores de diferentes parámetros del modelo XGBoost

Parámetro	Significado	Valores a evaluar
loss	Función de pérdida (error) a ser optimizado	ls, lad, huber
n_estimators	Número de árboles utilizados en el proceso	100, 500, 900, 1100, 1500
max_depth	Máxima profundidad de cada árbol	1, 2, 4, 6, 8
min_samples_split	Mínimo número de muestras para dividir un nodo	2, 4, 6, 10
max_features	Máximo número de variables a considerar para realizar divisiones	auto, sqrt, log2, None

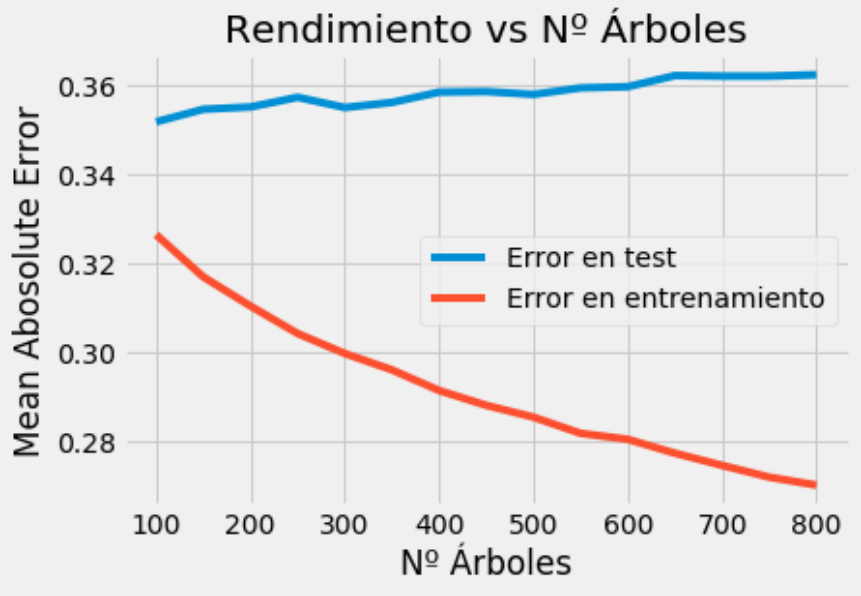
Teniendo en cuenta la cantidad de valores que puede tomar cada parámetro en la Tabla 16, se obtiene un total de 1.200 combinaciones posibles que debe evaluar el modelo. En nuestro caso, la iteración número 14 es la que mejor resultado de precisión da. El modelo debe configurarse con los siguientes parámetros:

```
XGBClassifier(base_score=0.5, booster=None, colsample_bylevel=1,
colsample_bynode=1, colsample_bytree=1, gamma=0, gpu_id=-1,
importance_type='gain', interaction_constraints=None,
```

```
learning_rate=0.300000012, loss='ls', max_delta_step=0,max_depth=2,
max_features='sqrt', min_child_weight=1,min_samples_leaf=2,
min_samples_split=6, missing=nan, monotone_constraints=None,
n_estimators=100, n_jobs=0, num_parallel_tree=1,
objective='binary:logistic', random_state=42, reg_alpha=0,
reg_lambda=1, scale_pos_weight=1, subsample=1, tree_method=None,
validate_parameters=False, ...)
```

Si realmente se quiere obtener el mejor resultado, es muy importante respetar los valores anteriores. A continuación, se demuestra cómo afecta el cambio de valores de un único parámetro al rendimiento del modelo. Para ello, se selecciona el parámetro *n_estimators* y se vuelve a poner a prueba el modelo con los siguientes nuevos valores: 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750 y 800; manteniendo los valores óptimos para el resto de parámetros.

Figura 16. Rendimiento del modelo variando el número de árboles



La Figura 16 muestra el error que presenta el modelo de predicción cuando se evalúa el conjunto de entrenamiento y cuando se evalúa el conjunto de test. Siempre habrá una diferencia entre el error de entrenamiento y el error de test (el error de entrenamiento siempre es menor), pero si hay una diferencia significativa, como ocurre en este caso cuando se aumenta el número de árboles, el modelo presenta un sobreajuste porque no sabe generalizar de forma correcta, y por tanto su rendimiento se desploma en cuanto debe tratar con datos nunca antes vistos. Es por ello que mantener los valores óptimos es la mejor solución.

El siguiente paso, una vez se tiene el modelo ya entrenado, es evaluar su funcionamiento utilizando el conjunto de test (*X_test*) para obtener las predicciones de clasificación, las cuales se comparan con las clases reales del conjunto de clasificación de test (*y_test*).

En esta ocasión se obtiene una precisión en las predicciones de un 70%. Se ha conseguido mejorar el modelo gracias a la búsqueda de la mejor configuración para los parámetros.

Se obtiene la siguiente matriz de confusión:

		Valores reales	
		No ecológico (%)	Ecológico (%)
Predicciones	No ecológico	2133 (73,4)	394 (39,9)
	Ecológico	775 (26,6)	594 (60,1)

Como puede observarse en dicha matriz, con este modelo se tienen 775 falsos positivos y 394 falsos negativos. El modelo es ligeramente mejor discriminando a los turistas no ecológicos, ya que el porcentaje de aciertos frente a fallos es superior para este grupo. El modelo tiende a clasificar en mayor medida como turistas ecológicos a aquellos que no lo son.

Para evaluar la validez del modelo, se compara dicho resultado de precisión con el porcentaje de acierto si la clasificación de clases se realizara de forma aleatoria. Si cada grupo estuviese compuesto del mismo número de respuestas, se podría clasificar de forma aleatoria a todos los turistas, sin ninguna información previa adicional, con una probabilidad de acierto del 25%. En este caso, las distribuciones entre grupos son claramente muy diferentes, por lo que el criterio de probabilidad proporcional puede utilizarse para evaluar la precisión predictiva del modelo (Morrison, 1969; Perreault et al., 1979; Huberty, 1984; Hair et al., 1998). El criterio de probabilidad proporcional se define como $\sum_i =_{1,K} (p_i)^2$ donde p_i representa la probabilidad esperada de que una observación elegida al azar sea clasificada en el grupo i. Las probabilidades esperadas (p_i) se calculan tomando la relación del número de muestras para cada grupo con respecto al número total de muestras. En el estudio se tiene un total de 19.478 muestras, y las probabilidades esperadas para cada clase son 61,32% y 38,78% respectivamente. Aplicando estos valores a la fórmula anterior, se obtiene un criterio de probabilidad proporcional del 52,64%. Dicho valor representa el acierto de una clasificación basada en el azar. Hair et al. (1998) recomiendan que la precisión predictiva del modelo sea al menos un 25% mayor que el criterio de probabilidad proporcional para que un modelo sea considerado como válido (1,25 x 52,64% = 65,80%). El 70% de precisión conseguido en el modelo de 2 clases supera el mínimo recomendado por Hair et al. (1998).

Teniendo en cuenta las limitaciones del estudio y los pocos datos de los que se dispone, se puede afirmar que se tienen unos resultados buenos, aunque mejorables siempre y cuando se disponga de más datos de próximos años, que le permitan al modelo reentrenarse conociendo más casos. Aun así, un acierto del 70% en la clasificación de los turistas en ecológicos o no ecológicos puede ser de gran ayuda para muchas instituciones.

Para finalizar la presentación de resultados del modelo de 2 clases, se muestra en la Tabla 17 las 10 características con más peso a la hora de predecir la variable objetivo.

Tabla 17. Importancia de características en el modelo de predicción de 2 clases

Características	Importancia
Nacionalidad turca	0,038
Instalaciones adaptadas – NS	0,036
Nivel general de precios – NS	0,021
Nacionalidad estonia	0,016
Nacionalidad belga	0,015
Nacionalidad luxemburguesa	0,014
Nacionalidad holandesa	0,012
Repetir destino – No repite	0,012
Servicios disponibles – NS	0,010
Repetir destino – NS	0,010

Aunque esta serie de características las ordena el modelo según un nivel de importancia interno propio, es interesante destacar que, así como en el análisis se encontraron diferencias significativas para los distintos niveles ecológicos en las personas turcas, belgas y luxemburguesas, de forma positiva, y en las personas estonias y holandesas (entre otras) de forma negativa, el modelo también tiene muy en cuenta estas consideraciones para decidir si un turista es ecológico o no lo es.

7.2. Modelo 4 clases

Para este caso se tienen los 4 grupos de nivel ecológico: Nada ecológico, Poco ecológico, Ecológico y Muy ecológico.

Basándonos en los buenos resultados ofrecidos por el algoritmo XGBoost para clasificación de 2 grupos, se decide utilizar también para una clasificación multiclase. Entrenar un modelo XGBoost multiclase conlleva un elevado coste computacional, y el hardware del que se dispone no permite realizar una comparación exhaustiva de modelos. Por la misma razón, tampoco se puede realizar la búsqueda de los mejores parámetros de configuración del modelo, así que se decide evaluar su funcionamiento utilizando los parámetros establecidos por defecto. En caso de disponer de hardware más potente, se recomienda realizar los mismos pasos que se han llevado a cabo para conseguir el mejor modelo clasificador de 2 clases.

Antes de probar el poder predictivo para las 4 clases, se debe definir el modelo indicándole que la clasificación en este caso no es binaria. Esto se consigue mediante la siguiente línea de código: `gb = XGBClassifier(objective='multi:softmax',num_class = 4)`. El entrenamiento y la validación del conjunto de test se hace de la misma manera que en el anterior modelo.

En esta ocasión se obtiene una precisión en las predicciones de un 61,27%.

Se obtiene la siguiente matriz de confusión:

		Valores reales			
		Nada ecológico (%)	Poco ecológico (%)	Ecológico (%)	Muy ecológico (%)
Predicciones	Nada ecológico	2182 (66,2)	231 (40,5)	11 (40,7)	1 (20,0)
	Poco ecológico	871 (26,4)	267 (46,8)	6 (22,2)	3 (60,0)
	Ecológico	181 (5,5)	57 (10,0)	6 (22,2)	1 (20,0)
	Muy ecológico	60 (1,8)	15 (2,6)	4 (14,8)	0 (0,0)

Como puede observarse en dicha matriz, el modelo también tiende a clasificar como turistas ecológicos a aquellos que no lo son.

Para evaluar la validez del modelo, se compara dicho resultado de precisión con el porcentaje de acierto si la clasificación de clases se realizara de forma aleatoria, de la misma manera que se ha hecho para el modelo de 2 clases. En el estudio se tiene un total de 19.478 muestras, y las probabilidades esperadas para cada clase son 61,32%, 30,29%, 6,38% y 2,01% respectivamente. Aplicando estos valores a la fórmula anterior, se obtiene un criterio de probabilidad proporcional del 47,20%. Con un modelo XGBoost configurado con los parámetros por defecto, el 61,27% de precisión conseguido ya supera el mínimo recomendado por Hair et al. (1998), que en este caso corresponde a un 59,00% (1,25 x 47,20% = 59,00%).

Se puede considerar que los resultados son buenos, pero, de igual forma que en para el modelo de 2 clases, también son mejorables. En este caso, no solo los pocos datos de los que se dispone, sino también las prestaciones del hardware utilizado marcan las limitaciones del estudio. Aun así, un acierto del 61,27% en la clasificación de los turistas en 4 niveles ecológicos, de nuevo puede ser de gran ayuda para muchas instituciones.

Para finalizar la presentación de resultados del modelo de 4 clases, se muestra en la Tabla 18 las 10 características con más peso a la hora de predecir la variable objetivo.

Tabla 18. Importancia de características en el modelo de predicción de 4 clases

Características	Importancia
Nacionalidad turca	0,022
Instalaciones adaptadas – NS	0,018
Nivel general de precios – NS	0,012
Nacionalidad belga	0,011
Nacionalidad estonia	0,011
Nacionalidad británica	0,009
Servicios disponibles – NS	0,008
Repetir destino – NS	0,008
Repetir destino – No repite	0,007
Repetir destino – Instalaciones	0,007

Destacar que, para este modelo de 4 clases, las características que más peso tienen son muy similares, y por tanto la mayoría de ellas también tienen relación con los análisis que se han realizado. El código para el entrenamiento y evaluación de ambos modelos se puede encontrar en el ***Anexo 6 – TFM_XGBoost.py.***

8. Discusión

Este estudio contribuye de forma activa a la literatura académica del turismo, de la demografía y de la innovación con las siguientes ideas.

Primero, se reafirma la idea de que la sociedad europea y la sociedad americana son muy diferentes. Mientras los estudios acerca de la ecología y el respeto medio ambiental en los Estados Unidos son bastante amplios, en Europa se están centrando en algunos aspectos concretos como las certificaciones medio ambientales y la gestión, sobre todo, de alojamientos como los hoteles, sin estudiar en detalle los comportamientos de los principales actores del turismo, que son las personas que deciden viajar. Este estudio ayudará a entender, desde el punto de vista ecológico, las decisiones y las actitudes de la sociedad europea en el momento de preparación de sus vacaciones y durante el disfrute de las mismas.

Segundo, el estudio, basado en los datos de los que se dispone, demuestra que los turistas europeos, de forma general, no consideran los aspectos ecológicos como aspectos importantes a la hora de preparar sus vacaciones. Aunque hay zonas geográficas con una alta concienciación medio ambiental, es cierto que la gran mayoría tiene otras preocupaciones durante las vacaciones. Ligado con el primer punto de este apartado, los estudios realizados en la sociedad americana muestran una conciencia medio ambiental más elevada que en el caso de Europa.

Tercero, esta investigación, basada en los datos de los que se dispone, demuestra por otra parte que los turistas europeos jóvenes son más conscientes de los problemas medioambientales que los turistas de mediana y elevada edad. En este caso, y de acuerdo con Han et al. (2011), se reafirma la idea de que las nuevas generaciones comparten una mayor implicación y preocupación por los problemas de contaminación y sostenibilidad. Debido a eso, no sólo en su día a día si no también durante sus vacaciones, presentan comportamientos sostenibles y respetuosos con el medio ambiente.

Cuarto, se deduce de la investigación que, en términos de actitudes ecológicas y sostenibles de cara a unas posibles vacaciones, no existen diferencias de género entre hombres y mujeres. Así como Han et al. (2011) llega a la conclusión de que las mujeres presentan comportamientos más ecológicos, para nuestro caso de estudio, tanto hombres como mujeres europeas pertenecen mayoritariamente al grupo de turistas no ecológicos. Esta diferencia, de nuevo puede ir ligada a las diferencias demográficas entre sociedades.

Por último, se demuestra que las técnicas de aprendizaje automático ofrecen resultados muy válidos con costes bajos de tiempo y computación. El tratamiento computacional de la información basada en técnicas estadísticas nos permite trabajar con grandes cantidades de datos. Esta investigación sirve como punto de partida a nuevos proyectos de análisis y predicción basados en macro cuestionarios, cuya tecnología a utilizar, sin ninguna duda, debe ser aquella que permita trabajar con Machine Learning.

9. Implicaciones

Los resultados de los análisis de la investigación y ambos modelos de predicción de nivel ecológico tienen importantes implicaciones administrativas para el sector público y privado del turismo.

En cuanto al sector privado, los proveedores de servicios turísticos (hoteles, restaurantes, aerolíneas, etc.) tendrán la capacidad de conocer, gracias a los hallazgos propios de los análisis, el comportamiento actual de sus posibles clientes. Con esto, podrán definir su perfil de cliente sobre el que dibujar su modelo de negocio, que les permita ofrecer experiencias personalizadas a cada cliente. Además, conociendo la tipología de clientela también podrán definir un plan de acción para adaptarse de la mejor manera posible a las demandas ecológicas y sostenibles por parte de los gobiernos. Una reforma integral de los servicios para adaptarse a dichas demandas, de un proveedor cuyos clientes no son ecológicos, supondría el cierre definitivo del negocio. Por otra parte, disponer de un modelo de predicción que permita clasificar a futuros clientes, les da la posibilidad de adaptar las campañas de marketing en base al nivel ecológico de cada persona. Esto supondrá una notable reducción en los costes de marketing (no se enviarán campañas masivas), además de aumentar los beneficios, ya que cada turista recibirá el contenido que les interesa, lo cual aumenta la propensión a compra.

En cuanto al sector público, los mismos destinos vacacionales y gobiernos, basándose en los resultados que les proporcionan los modelos de predicción, del mismo modo que los proveedores de servicios turísticos, también pueden enfocar sus campañas de marketing para atraer turistas. En este caso, conociendo el nivel ecológico se podrían realizar clústeres por países y preparar dichas campañas para cada país. Gracias a los resultados de los análisis, en el ámbito local se podrían llevar a cabo dos acciones. Una de ellas, y para aquellas personas del país que son ecológicas, sería la promoción de actividades para la conservación del medio ambiente (limpieza de playas, reforestación de bosques, etc.). Conocer que zonas del país son más ecológicas que otras, daría como resultado un mayor éxito en este tipo de actividades. Otra de las acciones a realizar sería la formación en valores ecológicos a aquellas personas que son consideradas como no ecológicas. Del mismo modo, conocer que zonas del país son menos ecológicas, le permite al gobierno aunar los esfuerzos para conseguir una mayor tasa de conversión ecológica en las personas, ya que se podría destinar una mayor cantidad de recursos a dichas zonas. Esta formación contribuiría a mostrar a los turistas la importancia de reciclar, respetar el medio ambiente y las poblaciones locales y contribuir con el desarrollo sostenible del destino vacacional, en el momento de la reserva de sus vacaciones o al llegar al destino.

10. Conclusiones

Este estudio ha sido planteado para investigar y conocer la situación actual del turismo ecológico europeo, centrándose en el grupo de los turistas y no en el de los proveedores turísticos. Como se ha podido comprobar, el turismo impacta considerablemente de forma positiva y negativa tanto a la economía, a la sociedad y a la naturaleza. En este último caso, el impacto negativo suele ser mayor debido a la contaminación del transporte y a los malos hábitos que presentan los turistas en los destinos vacacionales que visitan. Durante los últimos años, la preocupación por el medio ambiente y por su conservación ha crecido, al igual que el interés por el turismo rural en busca de espacios naturales. En la investigación se plantea la hipótesis de que este incremento de actitudes ecológicas se haya transmitido también al ámbito del turismo europeo.

Según los resultados obtenidos, se muestra como la mayoría de turistas europeos, en base a la consideración de aspectos ecológicos a la hora de tomar decisiones con respecto a sus vacaciones, son considerados como turistas no ecológicos. Dicha conclusión parece indicar que los turistas europeos, de forma generalizada, no consideran la ecología y la sostenibilidad como aspectos primordiales en el desarrollo de sus vacaciones, y que por tanto, los impactos de dicho turismo en la naturaleza seguirán siendo mayoritariamente negativos.

Si centramos la investigación en aquellos turistas que sí son ecológicos, se extrae la conclusión de que los aspectos ecológicos que más importancia tienen a la hora de preparar sus vacaciones son los siguientes:

- Que el destino incluya prácticas sostenibles con el medio ambiente.
- Que el destino sea accesible con transporte ecológico.
- Que los servicios dispongan de certificado ecológico.

Todos los 3 aspectos influyen positivamente en la reducción del impacto negativo medio ambiental.

Por otra parte, también se extrae la conclusión de que el perfil de turista ecológico europeo actualmente es el siguiente:

- Persona joven (de 15 a 34 años).
- De nacionalidad turca, belga o luxemburguesa.
- Que se aloje en campings o alquileres privados durante sus vacaciones.
- Que su razón principal para irse de vacaciones sea la naturaleza.
- Que no viaje solo.

Por último, añadir que esta investigación debe ser dinámica y necesitaría de una actualización continua de datos y de un reentrenamiento del modelo para asegurar que los resultados que se extraen de la misma siguen siendo válidos en el momento en el que se quieran extraer nuevas conclusiones.

11. Limitaciones del estudio y futuras líneas de investigación

Los resultados de esta investigación son aplicables al contexto del turismo europeo. El estudio encuentra una limitación en los datos, ya que sólo se dispone de información válida del año 2016, lo cual reduce la posibilidad de encontrar más variedad de información. Para próximos estudios sería interesante poder añadir más datos a la investigación e incluso utilizar nuevas técnicas de análisis y modelización ya que, con el tiempo, las utilizadas en este estudio quedarán obsoletas. Por otra parte, podrían realizarse próximos estudios centrados por país, basados en cuestionarios con un enfoque más directo a la ecología y la sostenibilidad turística con el fin de conseguir comparaciones más profundas y ser capaces de diferenciar zonas geográficas con más detalle. Por último, otro estudio interesante a realizar sería el de centrarse en aquellos turistas realmente ecológicos y prepara otro cuestionario que permita extraer conclusiones y patrones de comportamiento entre los mismos, para identificar acciones de transición para aquellos turistas no ecológicos. Además, también sería posible preparar modelos de predicción para cualquier de las otras variables que fuese de interés, como por ejemplo, conocer si el turista repetirá mismo destino vacacional o no.

12. Referencias

Noor, A., Wibisono, N., & Athar, H. S. (2016). Sustainable holiday indicators. Paper presented at the Heritage, Culture and Society: Research Agenda and Best Practices in the Hospitality and Tourism Industry - Proceedings of the 3rd International Hospitality and Tourism Conference, IHTC 2016 and 2nd International Seminar on Tourism, ISOT 2016, 371-376.

Guerreiro, M. (2017). Azores: More than a tourist destination. Worldwide Hospitality and Tourism Themes, 9(6), 653-658. doi:10.1108/WHATT-09-2017-0059

Alonso-Almeida, M. -. -., Fernández Robin, C., Celemín Pedroche, M. S., & Astorga, P. S. (2017). Revisiting green practices in the hotel industry: A comparison between mature and emerging destinations. Journal of Cleaner Production, 140, 1415-1428. doi:10.1016/j.jclepro.2016.10.010

Dávid, L. (2011). Tourism ecology: Towards the responsible, sustainable tourism future. Worldwide Hospitality and Tourism Themes, 3(3), 210-216. doi:10.1108/17554211111142176

Han, H., Hsu, L. T. J., Lee, J. -., & Sheu, C. (2011). Are lodging customers ready to go green? an examination of attitudes, demographics, and eco-friendly intentions. International Journal of Hospitality Management, 30(2), 345-355. doi:10.1016/j.ijhm.2010.07.008

Martínez, P., Herrero, Á., & Gómez-López, R. (2019). Corporate images and customer behavioral intentions in an environmentally certified context: Promoting environmental sustainability in the hospitality industry. Corporate Social Responsibility and Environmental Management, 26(6), 1382-1391. doi:10.1002/csr.1754

Yilmaz, Y., Üngüren, E., & Kaçmaz, Y. Y. (2019). Determination of managers' attitudes towards eco-labeling applied in the context of sustainable tourism and evaluation of the effects of eco-labeling on accommodation enterprises. Sustainability (Switzerland), 11(18) doi:10.3390/su11185069

González-Rodríguez, M. R., Díaz-Fernández, M. C., & Font, X. (2019). Factors influencing willingness of customers of environmentally friendly hotels to pay a price premium. International Journal of Contemporary Hospitality Management, doi:10.1108/IJCHM-02-2019-0147

Sucheran, R., & Moodley, V. (2019). Guest dynamics and perceptions towards environmentally-friendly practices in hotels in KwaZulu- natal, south africa. African Journal of Hospitality, Tourism and Leisure, 8(3)

Gil-Soto, E., Armas-Cruz, Y., Morini-Marrero, S., & Ramos-Henríquez, J. M. (2019). Hotel guests' perceptions of environmental friendly practices in social media. International Journal of Hospitality Management, 78, 59-67. doi:10.1016/j.ijhm.2018.11.016

Khatter, A., McGrath, M., Pyke, J., White, L., & Lockstone-Binney, L. (2019). Analysis of hotels' environmentally sustainable policies and practices: Sustainability and corporate social responsibility in hospitality and tourism. International Journal of Contemporary Hospitality Management, 31(6), 2394-2410. doi:10.1108/IJCHM-08-2018-0670

Mazhenova, S., Choi, J. -., & Chung, J. (2016). International tourists' awareness and attitude about environmental responsibility and sustainable practices. Global Business and Finance Review, 21(2), 132-146. doi:10.17549/gbfr.2016.21.2.132

Bastič, M., & Gojčič, S. (2012). Measurement scale for eco-component of hotel service quality. International Journal of Hospitality Management, 31(3), 1012-1020. doi:10.1016/j.ijhm.2011.12.007

Navratil, J., Picha, K., Buchecker, M., Martinat, S., Svec, R., Brezinova, M., & Knotek, J. (2019). Visitors' preferences of renewable energy options in "green" hotels. Renewable Energy, 138, 1065-1077. doi:10.1016/j.renene.2019.02.043

Chenoweth, J. (2009). Is tourism with a low impact on climate possible? Worldwide Hospitality and Tourism Themes, 1(3), 274-287. doi:10.1108/17554210910980611

Budeanu, A. (2007). Sustainable tourist behaviour – a discussion of opportunities for change. International Journal of Consumer Studies, 31(5), 499-508. doi:10.1111/j.1470-6431.2007.00606.x

Jung, T. H., Ineson, E. M., & Miller, A. (2014). The slow food movement and sustainable tourism development: A case study of mold, wales. International Journal of Culture, Tourism, and Hospitality Research, 8(4), 432-445. doi:10.1108/IJCTHR-01-2014-0001

Morrison, D.G., 1969. On the interpretation of discriminant analysis. Journal of Marketing Research 6 2 , 156–163.

Perreault, W.D., Behrman, D.N., Armstrong, G.M., 1979. Alternative approaches for interpretation of multiple discriminant analysis in marketing research. Journal of Business Research 7, 151–173.

Huberty, C.J., 1984. Issues in the use and interpretation of discriminant analysis. Psychological Bulletin 95, 156–171.

Hair, J.F. Jr., Anderson, R.E., Tatham, R.L., Black, W.C., 1998. Multivariate Data Analysis. 5th edn. Prentice-Hall, New Jersey.

OpenWebinars (2019). Definición de Python. Buscado el 13 Abril de 2020, en https://openwebinars.net/blog/que-es-python/

Towards Data Science (2019). Algoritmo XGBoost. Buscado el 20 Abril de 2020, en https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d (20/04/2020)

13. Anexos

Anexo 1 – Tablas y gráficos análisis

Figura 1. Comprobación de outliers para la variable "age"

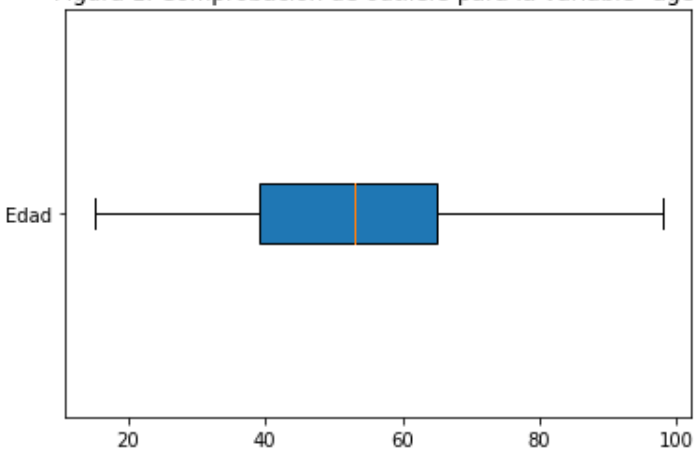


Tabla 1. Estadísticos de la edad de los turistas encuestados en 2016

N	Válidos	19478
	Perdidos	0
Media		51,65
Mediana		53,00
Desviación típica		16,81
Mínimo		15
Máximo		98
Percentiles	25	39,25
	50	53,00
	75	65,00

Figura 2. Distribución de edades de los turistas encuestados en 2016

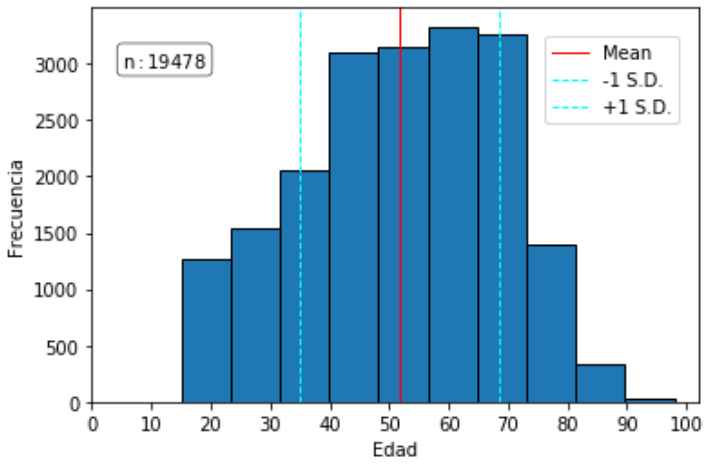


Tabla 2. Porcentaje de edades de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
15 – 24 años	1430	7,34	7,34
25 – 34 años	2049	10,52	17,86
35 – 44 años	3000	15,40	33,26
45 – 54 años	3798	19,50	52,76
55 – 64 años	4182	21,47	74,23
Más de 65 años	5019	25,77	100
N	19478	100	

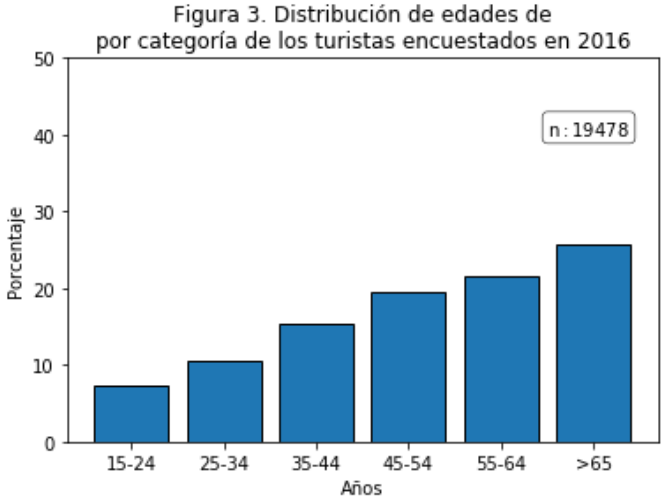


Tabla 3. Porcentaje de género de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Femenino	11066	56,81	56,81
Masculino	8412	43,19	100
N	19478	100	

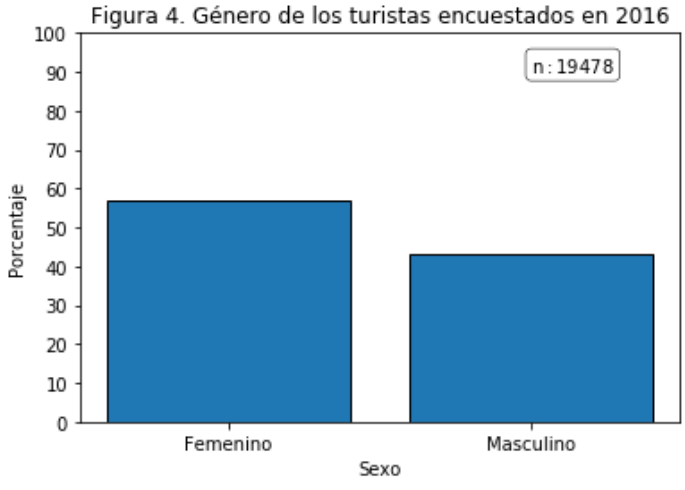


Tabla 4. Porcentaje de oficios de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Empleados	7564	38,83	38,83
Trabajadores manuales	1182	6,07	44,90
No trabajan	8609	44,20	89,10
Rehúsan	79	0,41	89,51
Autónomos	2044	10,49	100
N	19478	100	

Figura 5. Situación laboral de los turistas encuestados en 2016

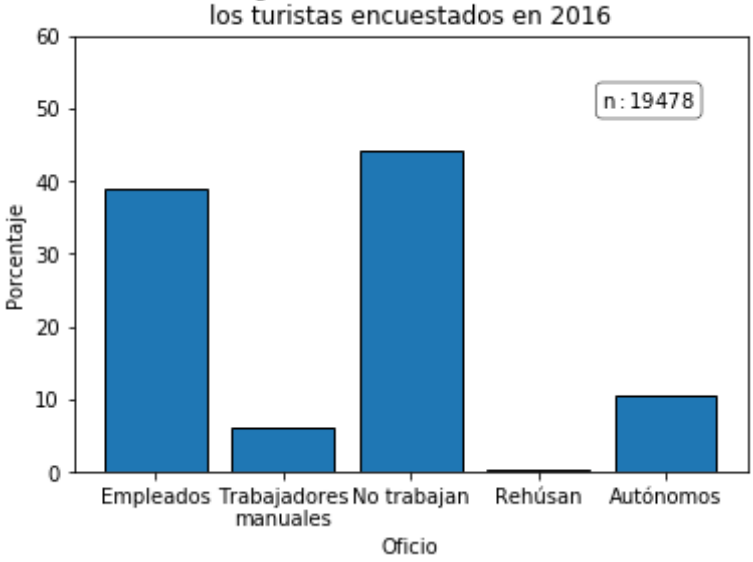


Tabla 5. Porcentaje de lugar de residencia de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
NS (No sabe)	151	0,78	0,78
Ciudad	6361	32,66	33,44
Zona Rural	5576	28,63	62,07
Pueblo	7390	37,93	100
N	19478	100	

Figura 6. Lugar de residencia
de los turistas encuestados en 2016

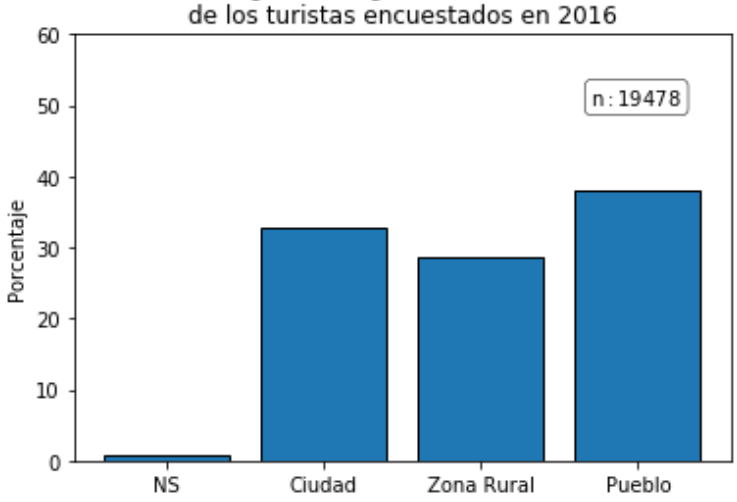


Tabla 6. Porcentaje de nacionalidad de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Austria	797	4,09	4,09
Bélgica	604	3,10	7,19
Bulgaria	555	2,85	10,04
Croacia	321	1,65	11,69
Chipre	370	1,90	13,59
República Checa	697	3,58	17,17
Dinamarca	797	4,09	21,26
Estonia	339	1,74	23,00
Finlandia	758	3,89	26,89
Francia	1116	5,73	32,62
Alemania	1101	5,65	38,27
Grecia	588	3,02	41,29
Hungría	557	2,86	44,15
Islandia	442	2,27	46,42
Irlanda	808	4,15	50,57
Italia	964	4,95	55,52
Letonia	273	1,40	56,92
Lituania	280	1,44	58,36
Luxemburgo	382	1,96	60,32
Macedonia	236	1,21	61,53
Malta	282	1,45	62,98
Moldavia	148	0,76	63,74
Montenegro	288	1,48	65,22
Polonia	923	4,74	69,96
Portugal	462	2,37	72,33
Rumanía	409	2,10	74,43
Eslovaquia	619	3,18	77,61
Eslovenia	354	1,82	79,43
España	974	5,00	84,43
Suecia	799	4,10	88,53
Holanda	748	3,84	92,37
Turquía	434	2,23	94,60
Reino Unido	1052	5,40	100
N	19478	100	

Figura 7. Nacionalidad de los turistas encuestados en 2016

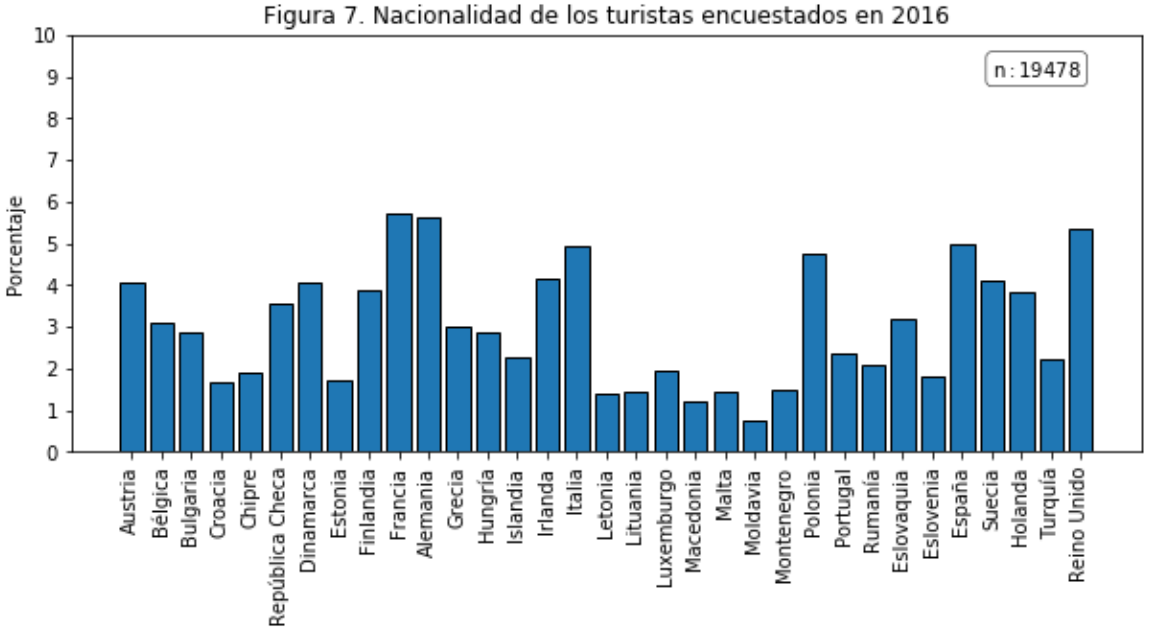


Figura 8. Comprobación de outliers para la variable "traveltimes"

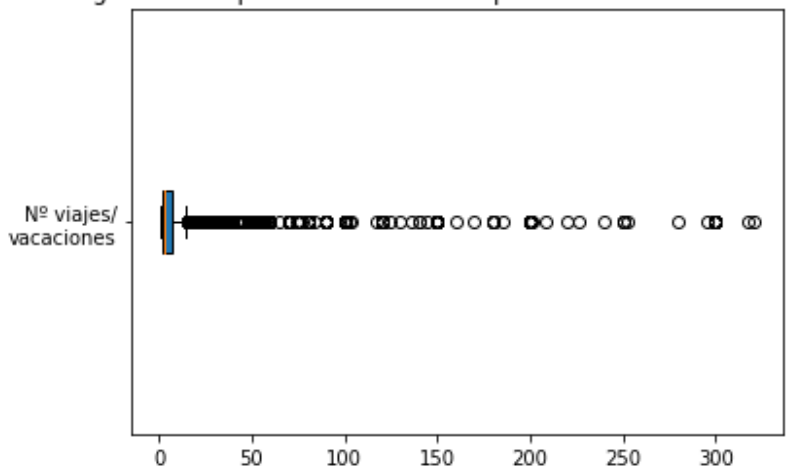


Tabla 7. Estadísticos del número de viajes de los turistas encuestados en 2016

N	Válidos	17078
	Perdidos	2400
Media		3,84
Mediana		3,00
Desviación típica		2,89
Mínimo		2
Máximo		14
Percentiles	25	2,00
	50	3,00
	75	5,00

Figura 9. Nº veces que viajaron el año anterior los turistas encuestados en 2016

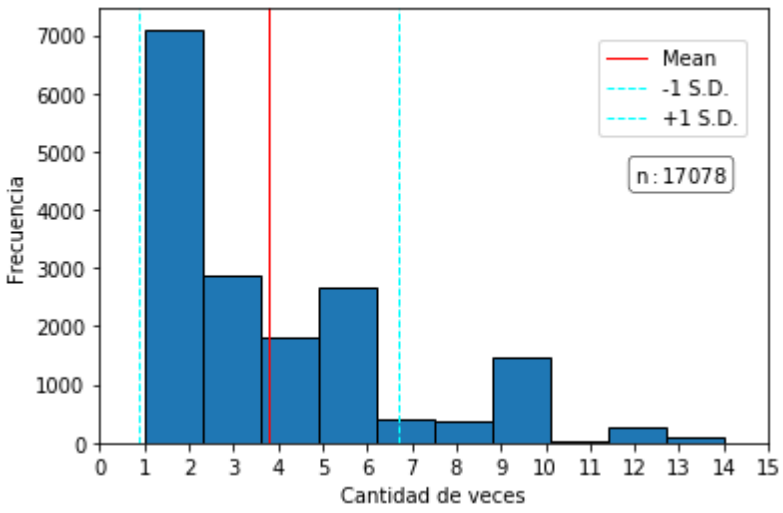


Tabla 8. Porcentaje de número de viajes de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
1 vez	3802	22,26	22,26
2 veces	3296	19,30	41,56
3 veces	2866	16,78	58,34
4 o 5 veces	3610	21,14	79,48
De 6 a 10 veces	3123	18,29	97,77
Más de 10 veces	381	2,23	100
N	17078	100	

Figura 10. Nº veces por categoría que viajaron el año anterior los turistas encuestados en 2016

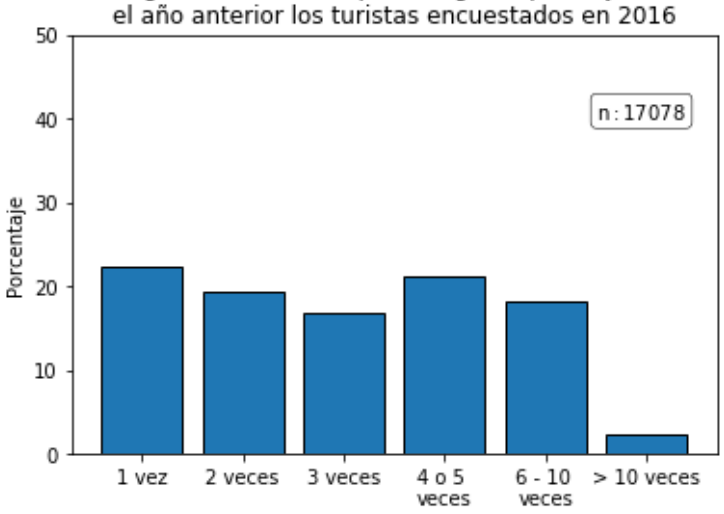


Tabla 9. Porcentaje de número de viajes de larga duración de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Ninguno	11563	59,36	59,36
1 vez	5182	26,60	85,96
2 veces	1580	8,11	94,07
3 veces	500	2,57	96,64
4 o 5 veces	326	1,67	98,31
De 6 a 10 veces	183	0,94	99,25
Más de 10 veces	144	0,75	100
N	19478	100	

Figura 11. Nº veces por categoría que viajaron (> 13 noches) el año anterior los turistas encuestados en 2016

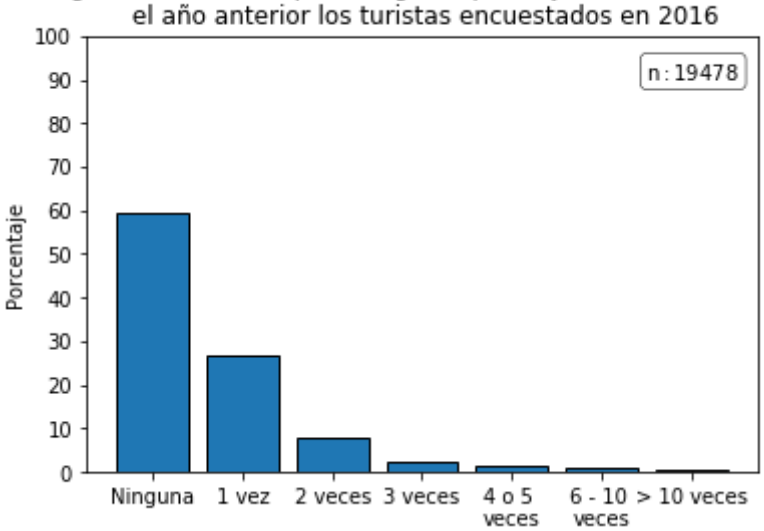


Tabla 10. Porcentaje de número de viajes de media duración de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Ninguno	5392	27,68	27,68
1 vez	6288	32,28	59,96
2 veces	3698	18,99	78,95
3 veces	1745	8,96	87,91
4 o 5 veces	1427	7,33	95,24
De 6 a 10 veces	701	3,60	98,84
Más de 10 veces	227	1,16	100
N	19478	100	

Figura 12. Nº veces por categoría que viajaron (4-13 noches) el año anterior los turistas encuestados en 2016

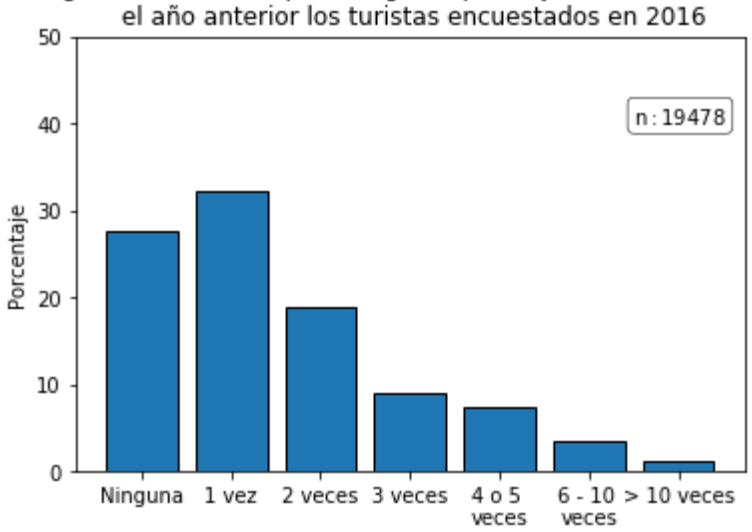


Tabla 11. Porcentaje de número de viajes de corta duración de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Ninguno	6231	32,00	32,00
1 vez	3972	20,39	52,39
2 veces	3110	15,97	68,36
3 veces	1847	9,48	77,84
4 o 5 veces	1910	9,81	87,65
De 6 a 10 veces	1473	7,56	95,21
Más de 10 veces	935	4,79	100
N	19478	100	

Figura 13. Nº veces por categoría que viajaron (< 3 noches) el año anterior los turistas encuestados en 2016

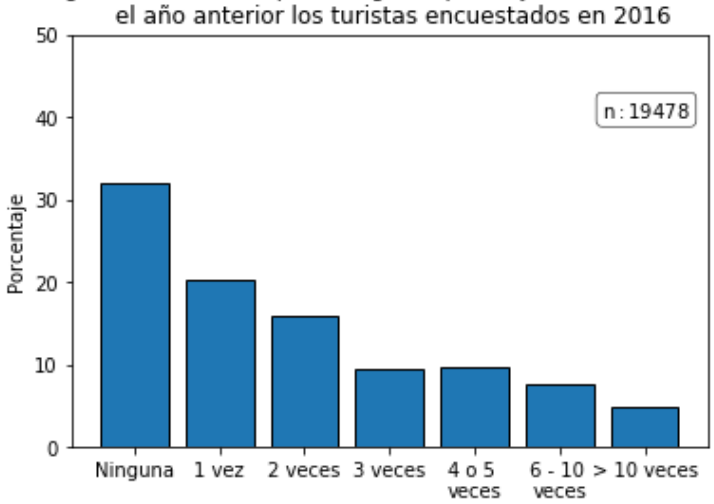


Tabla 12. Porcentaje de tipos de alojamiento en vacaciones de larga duración

	Frecuencia	Porcentaje	Porcentaje Acumulado
Camping	615	7,77	7,77
NS (No sabe)	56	0,71	8,48
Otros	119	1,50	9,98
Alquiler	1319	16,66	26,64
Alojamiento comercial	3137	39,63	66,27
Con amigos o familia	1575	19,90	86,17
Segunda residencia	1094	13,83	100
N	7915	100	

Figura 14. Tipo de alojamiento para viajes de larga duración (> 13 noches)

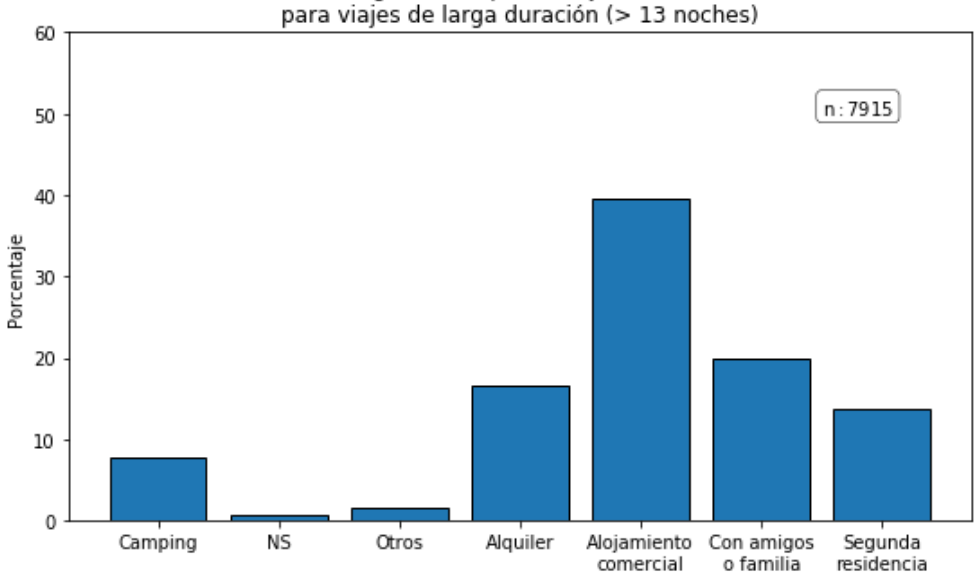


Tabla 13. Porcentaje de tipos de alojamiento en vacaciones de media duración

	Frecuencia	Porcentaje	Porcentaje Acumulado
Camping	774	5,49	5,49
NS (No sabe)	90	0,64	6,13
Otros	125	0,89	7,02
Alquiler	2238	15,89	22,91
Alojamiento comercial	7158	50,82	73,73
Con amigos o familia	2732	19,40	93,13
Segunda residencia	969	6,87	100
N	14086	100	

Figura 15. Tipo de alojamiento para viajes de media duración (4-13 noches)

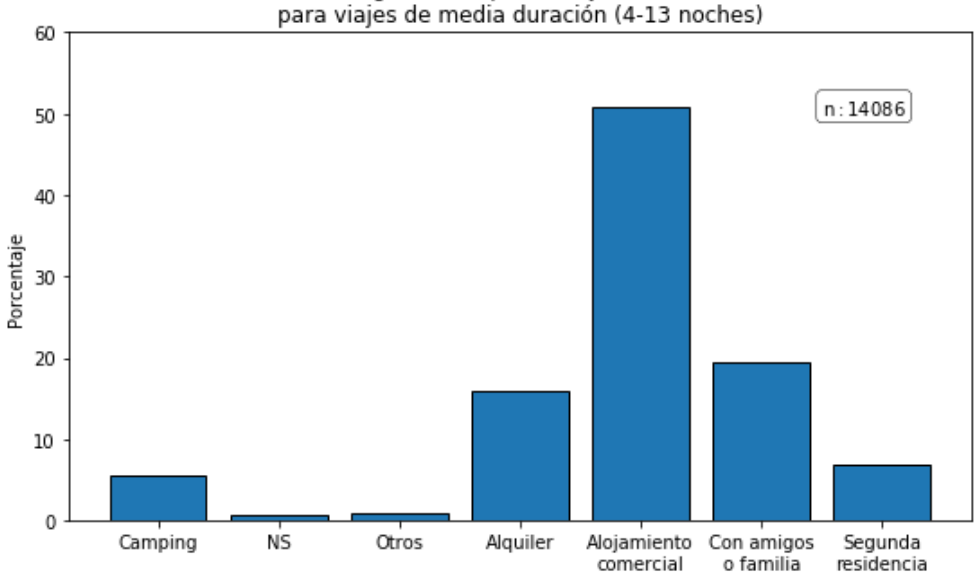


Tabla 14. Porcentaje de tipos de alojamiento en vacaciones de corta duración

	Frecuencia	Porcentaje	Porcentaje Acumulado
Camping	566	4,27	4,27
NS (No sabe)	72	0,54	4,81
Otros	144	1,09	5,90
Alquiler	1268	9,57	15,47
Alojamiento comercial	6030	45,52	60,99
Con amigos o familia	4197	31,68	92,67
Segunda residencia	970	7,33	100
N	13247	100	

Figura 16. Tipo de alojamiento
para viajes de corta duración (< 3 noches)

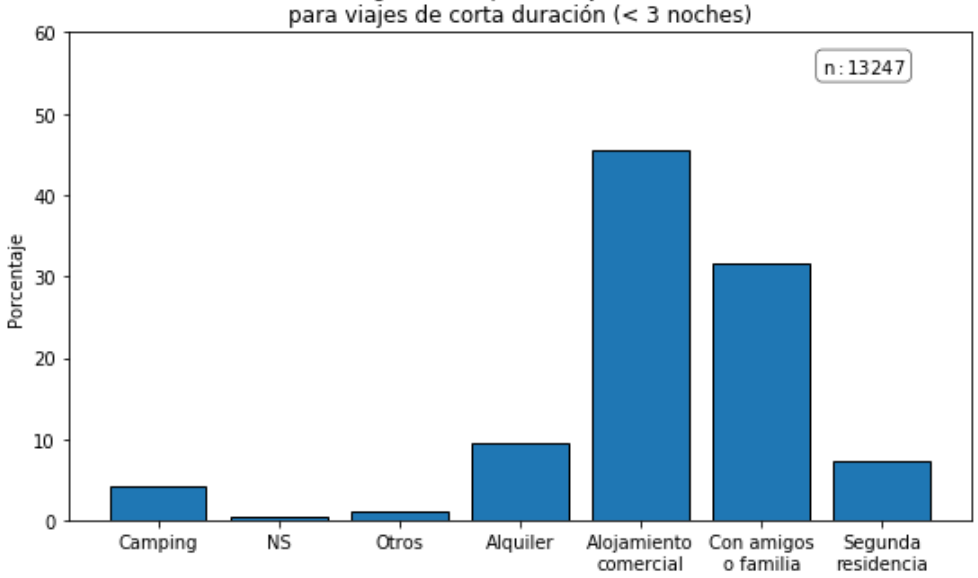


Tabla 15. Porcentaje de países visitados por los turistas encuestados en 2016 en sus vacaciones principales

	Frecuencia	Porcentaje	Porcentaje Acumulado
Asia/Oceania	464	2,38	2,38
Austria	275	1,41	3,79
Bélgica	99	0,51	4,30
Bulgaria	117	0,60	4,90
País de residencia	8938	45,89	50,79
Croacia	629	3,23	54,02
Chipre	66	0,34	54,36
República Checa	127	0,65	55,01
Dinamarca	109	0,56	55,57
NS (No sabe)	123	0,63	56,20
Estonia	74	0,38	56,58
Finlandia	68	0,35	56,93
Francia	873	4,48	61,41
Alemania	563	2,89	64,30
Grecia	781	4,01	68,31
Hungría	138	0,71	69,02
Irlanda	55	0,28	69,30
Italia	1036	5,32	74,62
Letonia	45	0,23	74,85
Lituania	25	0,13	74,98
Luxemburgo	16	0,08	75,06
Malta	49	0,25	75,31
Holanda	148	0,76	76,07
Norte de África/Oriente Medio	308	1,58	77,65
Otros	1138	5,84	83,49
Polonia	113	0,58	84,07
Portugal	280	1,44	85,51
Rumanía	84	0,43	85,94
Eslovaquia	80	0,41	86,35

Eslovenia	53	0,27	86,62
España	1243	6,38	93,00
Suecia	187	0,96	93,96
Sudamérica/América Central	253	1,30	95,26
EE.UU./Canadá	487	2,50	97,76
Reino Unido	436	2,24	100
N	19478	100	

Figura 16. País visitado en las vacaciones principales de los turistas encuestados en 2016

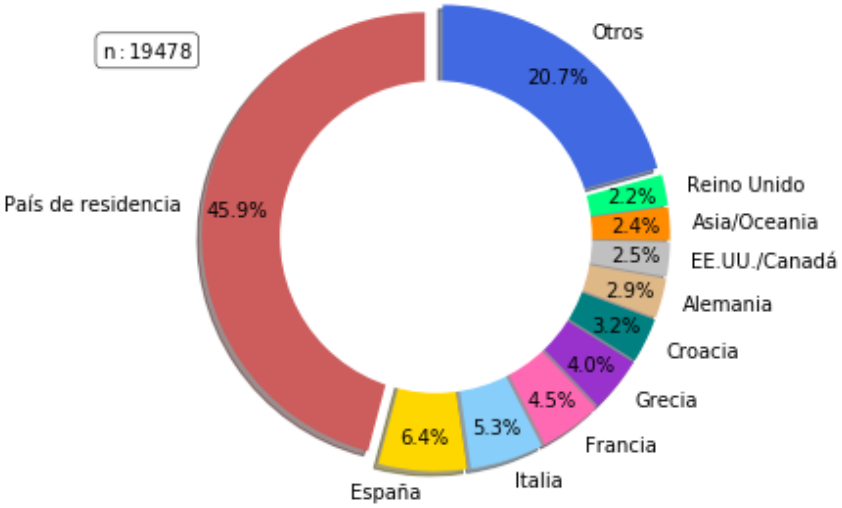


Tabla 16. Porcentaje de la razón principal de las vacaciones de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Viajes a ciudades	1724	8,85	8,85
Cultura	1825	9,37	18,22
NS (No sabe)	121	0,62	18,84
Naturaleza	2524	12,96	31,80
Otras	1227	6,30	38,10
Eventos concretos	662	3,40	41,50
Actividades deportivas	740	3,80	45,30
Sol/Playa	4542	23,32	68,62
Visita a familia	4571	23,47	92,09
Terapia de salud	1541	7,91	100
N	19478	100	

Figura 17. Razón principal por la que viajan los turistas encuestados en 2016

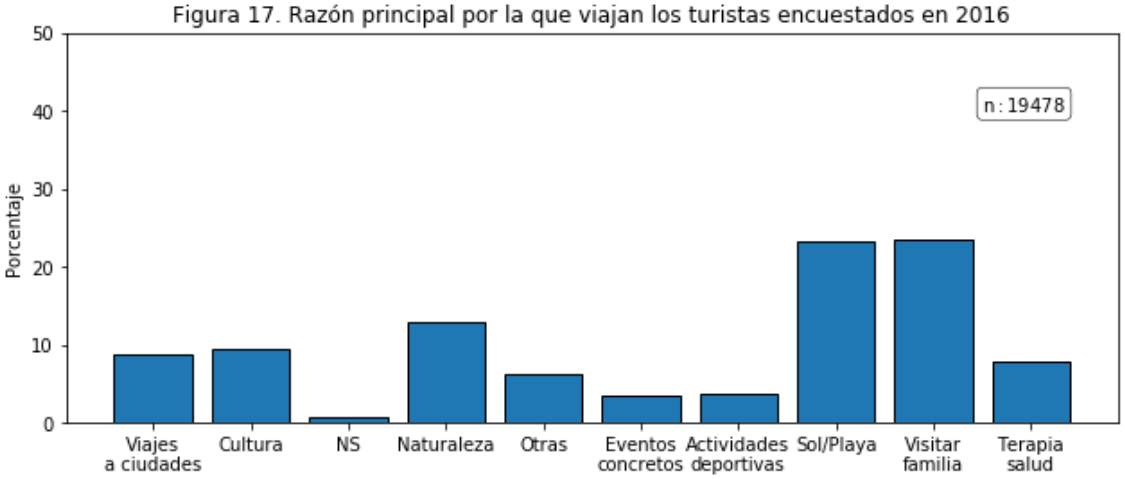


Tabla 17. Porcentaje de la segunda razón de las vacaciones de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Viajes a ciudades	2683	13,86	13,86
Cultura	1541	7,96	21,82
NS (No sabe)	4464	23,06	44,88
Naturaleza	2228	11,51	56,39
Otras	1692	8,74	65,13
Eventos concretos	426	2,20	67,33
Actividades deportivas	3442	17,78	85,11
Visita a familia	1525	7,88	92,99
Terapia de salud	1359	7,01	100
N	19357	100	

Figura 18. Segunda razón por la que viajan los turistas encuestados en 2016

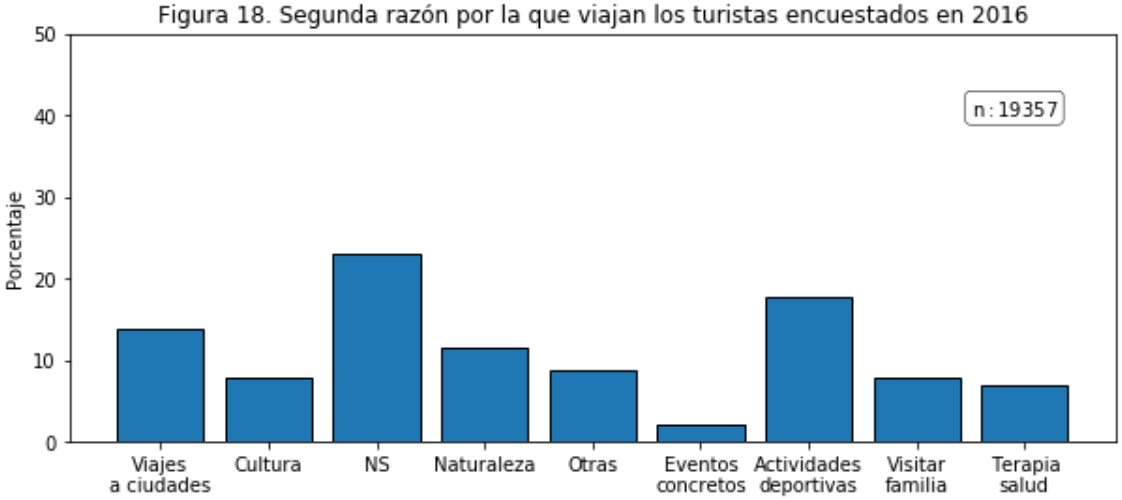


Tabla 18. Porcentaje de la tercera razón de las vacaciones de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Viajes a ciudades	461	10,05	10,05
Cultura	1246	27,18	37,23
Naturaleza	1218	26,56	63,79
Otras	187	4,08	67,87
Eventos concretos	302	6,59	74,46
Actividades deportivas	352	7,68	82,14
Visita a familia	819	17,86	100
N	4585	100	

Figura 19. Tercera razón por la que viajan los turistas encuestados en 2016

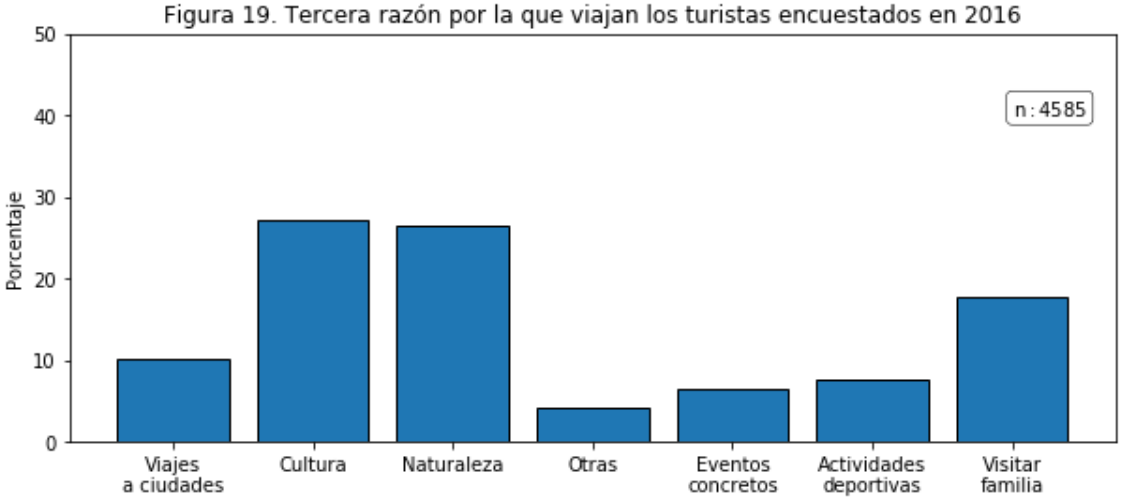


Tabla 19. Porcentaje de la cuarta razón de las vacaciones de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Cultura	456	32,52	32,52
Naturaleza	222	15,83	48,35
Otras	73	5,21	53,56
Eventos concretos	218	15,55	69,11
Actividades deportivas	33	2,35	71,46
Visita a familia	400	28,54	100
N	1402	100	

Figura 20. Cuarta razón por la que viajan los turistas encuestados en 2016

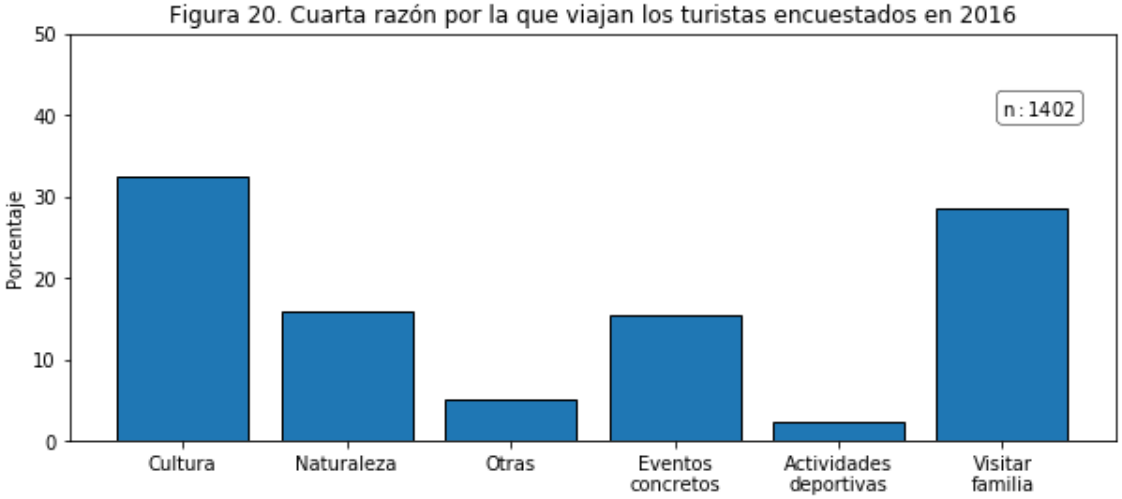


Tabla 20. Porcentaje del motivo principal de los turistas encuestados en 2016 para repetir destino en sus vacaciones

	Frecuencia	Porcentaje	Porcentaje Acumulado
Accesibilidad instalaciones	358	1,84	1,84
Cultura	2980	15,30	17,14
NS (No sabe)	838	4,30	21,44
Bienvenida a turistas	1391	7,14	28,58
No repite destino	1099	5,64	34,22
Otros	1584	8,13	42,35
Actividades/servicios	1439	7,39	49,74
Precios	1660	8,52	58,26
Aspectos naturales	5152	26,45	84,71
Calidad alojamiento	2978	15,29	100
N	19478	100	

Figura 21. Motivo principal por el que repiten destino los turistas encuestados en 2016

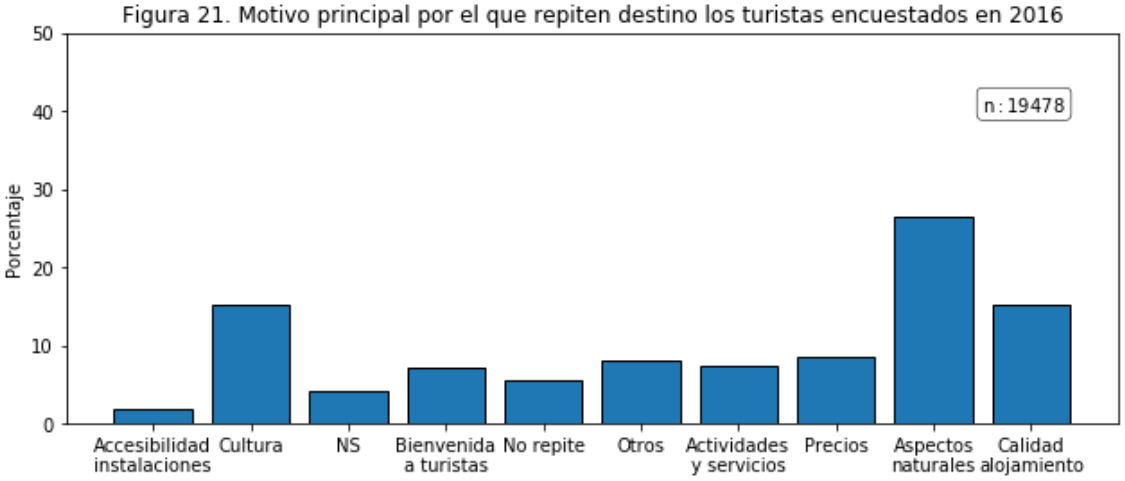


Tabla 21. Porcentaje del segundo motivo de los turistas encuestados en 2016 para repetir destino en sus vacaciones

	Frecuencia	Porcentaje	Porcentaje Acumulado
Accesibilidad instalaciones	172	0,98	0,98
Cultura	1470	8,38	9,36
NS (No sabe)	2945	16,79	26,15
Bienvenida a turistas	1307	7,45	33,60
No repite destino	118	0,67	34,27
Otros	1484	8,46	42,73
Actividades/servicios	1123	6,40	49,13
Precios	1861	10,61	59,74
Aspectos naturales	3443	19,63	79,37
Calidad alojamiento	3619	20,63	100
N	17542	100	

Figura 22. Segundo motivo por el que repiten destino los turistas encuestados en 2016

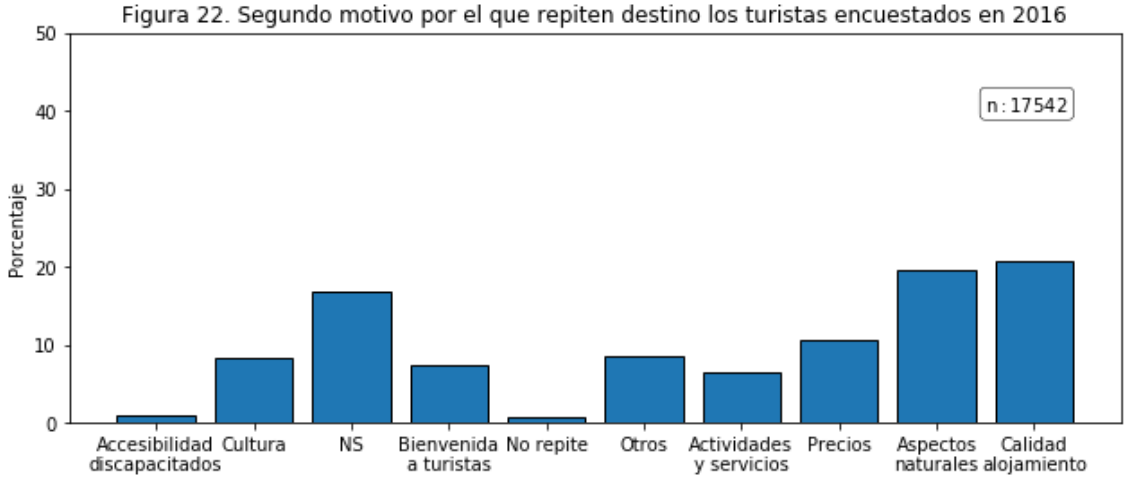


Tabla 22. Porcentaje del tercer motivo de los turistas encuestados en 2016 para repetir destino en sus vacaciones

	Frecuencia	Porcentaje	Porcentaje Acumulado
Accesibilidad instalaciones	203	4,23	4,23
Cultura	1150	23,97	28,20
Bienvenida a turistas	1119	23,32	51,52
Otros	155	3,23	54,75
Actividades/servicios	993	20,70	75,45
Precios	1177	24,55	100
N	4797	100	

Figura 23. Tercer motivo por el que repiten destino los turistas encuestados en 2016

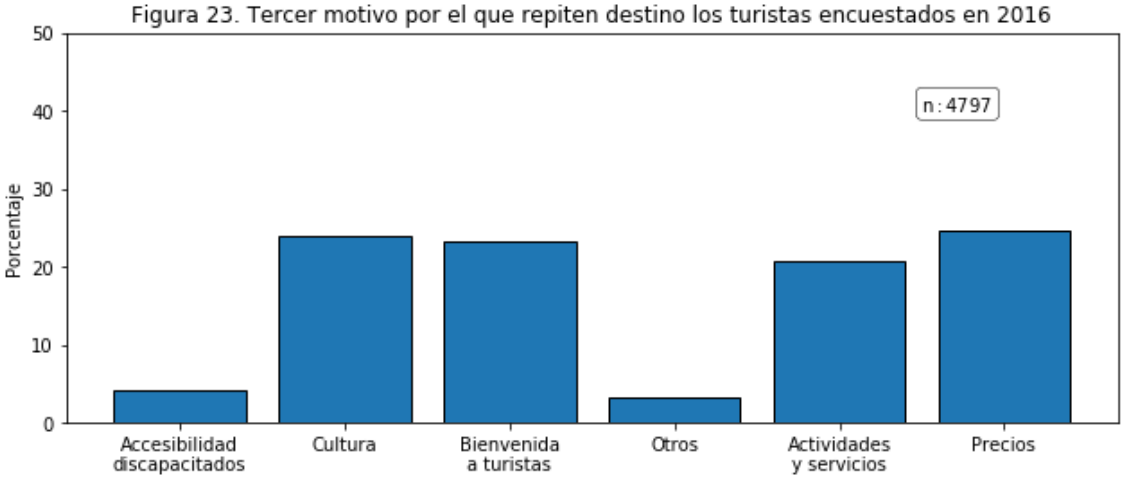


Tabla 23. Porcentaje del cuarto motivo de los turistas encuestados en 2016 para repetir destino en sus vacaciones

	Frecuencia	Porcentaje	Porcentaje Acumulado
Accesibilidad instalaciones	225	11,49	11,49
Cultura	891	45,51	57,00
Bienvenida a turistas	290	14,81	71,81
Otros	58	2,96	74,77
Actividades/servicios	494	25,23	100
N	1958	100	

Figura 24. Cuarto motivo por el que repiten destino los turistas encuestados en 2016

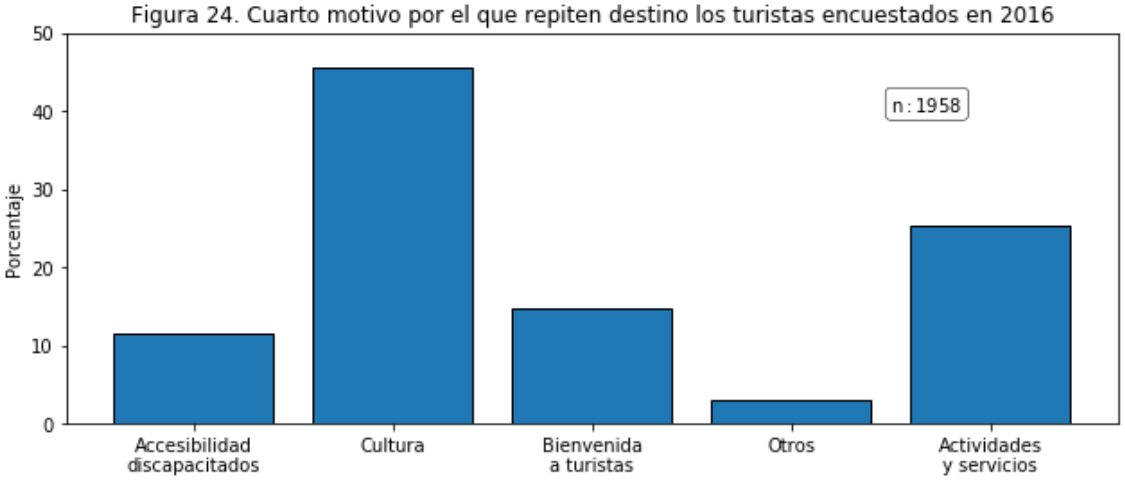


Tabla 24. Porcentaje del nivel de satisfacción de los turistas encuestados en 2016 con la calidad del alojamiento

	Frecuencia	Porcentaje	Porcentaje Acumulado
Insatisfecho	131	0,67	0,67
Poco satisfecho	493	2,53	3,20
Bastante satisfecho	6186	31,76	34,96
Muy satisfecho	12047	61,85	96,81
NS (No sabe)	621	3,19	100
N	19478	100	

Figura 25. Nivel de satisfacción de los turistas encuestados en 2016 con la calidad del alojamiento

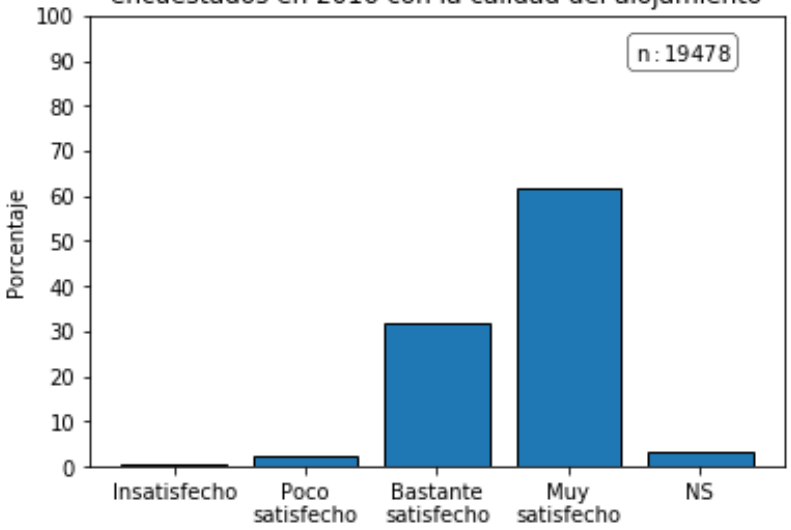


Tabla 25. Porcentaje del nivel de satisfacción de los turistas encuestados en 2016 con la seguridad del alojamiento

	Frecuencia	Porcentaje	Porcentaje Acumulado
Insatisfecho	95	0,49	0,49
Poco satisfecho	294	1,51	2,00
Bastante satisfecho	5119	26,28	28,28
Muy satisfecho	13226	67,90	96,18
NS (No sabe)	744	3,82	100
N	19478	100	

Figura 26. Nivel de satisfacción de los turistas encuestados en 2016 con la seguridad del alojamiento

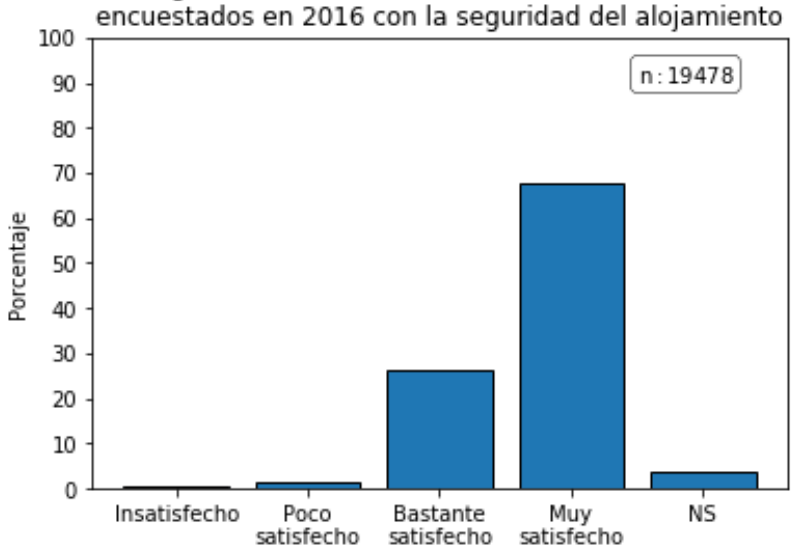


Tabla 26. Porcentaje del nivel de satisfacción de los turistas encuestados en 2016 con las características del entorno

	Frecuencia	Porcentaje	Porcentaje Acumulado
Insatisfecho	109	0,56	0,56
Poco satisfecho	429	2,20	2,76
Bastante satisfecho	4903	25,17	27,93
Muy satisfecho	13545	69,54	97,47
NS (No sabe)	493	2,53	100
N	19478	100	

Figura 27. Nivel de satisfacción de los turistas encuestados en 2016 con las características del entorno

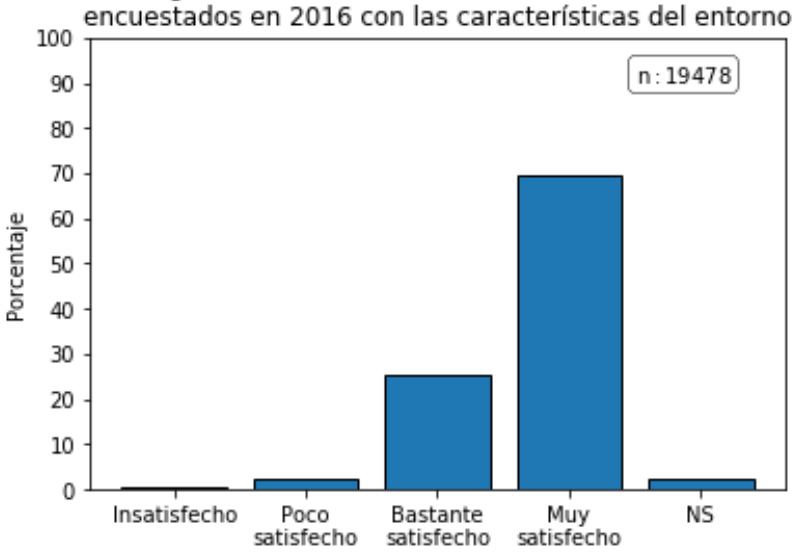


Tabla 27. Porcentaje del nivel de satisfacción de los turistas encuestados en 2016 con el nivel general de precios

	Frecuencia	Porcentaje	Porcentaje Acumulado
Insatisfecho	284	1,46	1,46
Poco satisfecho	1650	8,47	9,93
Bastante satisfecho	9513	48,84	58,77
Muy satisfecho	6884	35,34	94,11
NS (No sabe)	1147	5,89	100
N	19478	100	

Figura 28. Nivel de satisfacción de los turistas encuestados en 2016 con el nivel general de precios

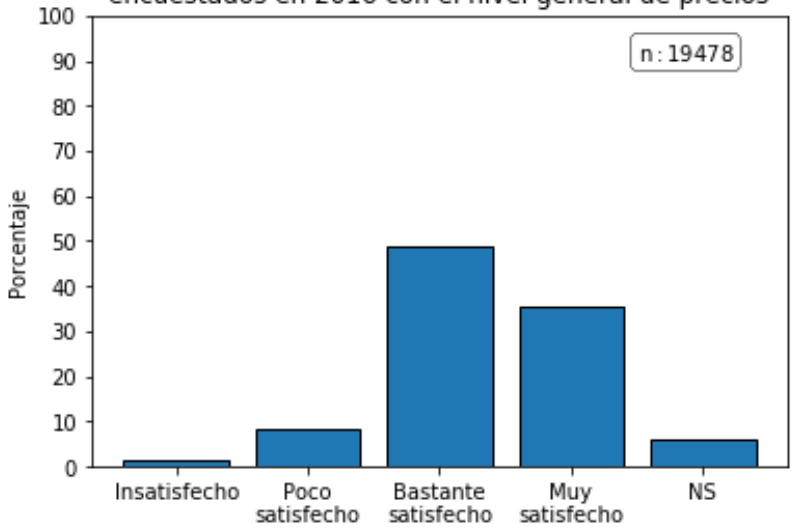


Tabla 28. Porcentaje del nivel de satisfacción de los turistas encuestados en 2016 con la bienvenida recibida

	Frecuencia	Porcentaje	Porcentaje Acumulado
Insatisfecho	164	0,84	0,84
Poco satisfecho	666	3,42	4,26
Bastante satisfecho	6437	33,05	37,31
Muy satisfecho	9965	51,16	88,47
NS (No sabe)	2246	11,53	100
N	19478	100	

Figura 29. Nivel de satisfacción de los turistas encuestados en 2016 con la bienvenida recibida

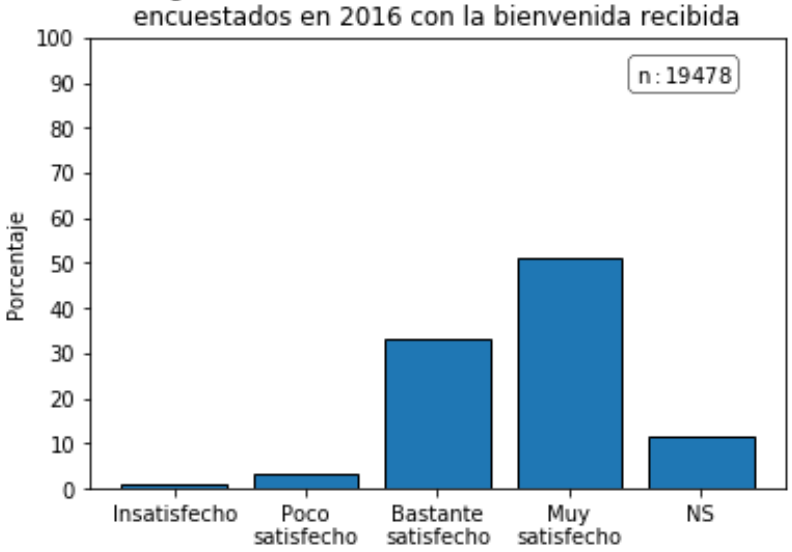


Tabla 29. Porcentaje del nivel de satisfacción de los turistas encuestados en 2016 con los servicios disponibles

	Frecuencia	Porcentaje	Porcentaje Acumulado
Insatisfecho	168	0,86	0,86
Poco satisfecho	808	4,15	5,01
Bastante satisfecho	7411	38,05	43,06
Muy satisfecho	9571	49,14	92,20
NS (No sabe)	1515	7,78	100
N	19478	100	

Figura 30. Nivel de satisfacción de los turistas encuestados en 2016 con los servicios disponibles

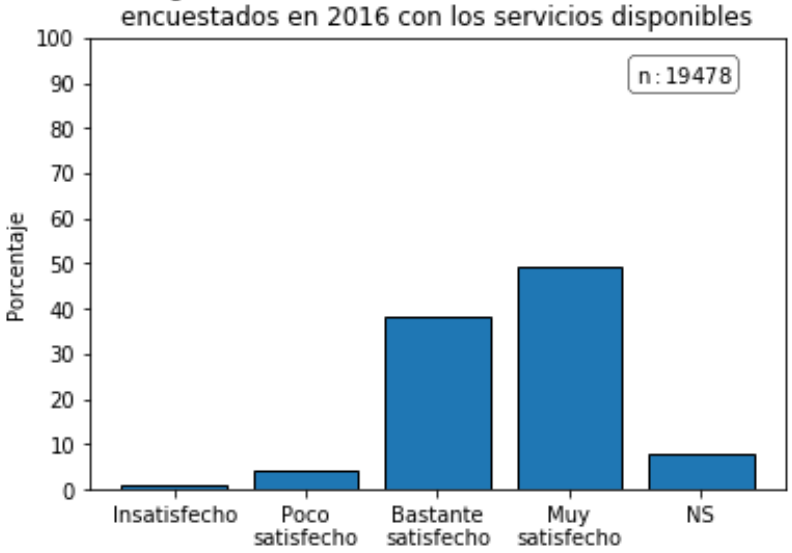


Tabla 30. Porcentaje del nivel de satisfacción de los turistas encuestados en 2016 con la accesibilidad de las instalaciones

	Frecuencia	Porcentaje	Porcentaje Acumulado
Insatisfecho	596	3,06	3,06
Poco satisfecho	1430	7,34	10,40
Bastante satisfecho	4727	24,27	34,67
Muy satisfecho	4203	21,58	56,25
NS (No sabe)	8522	43,75	100
N	19478	100	

Figura 31. Nivel de satisfacción de los turistas encuestados en 2016 con la accesibilidad de las instalaciones

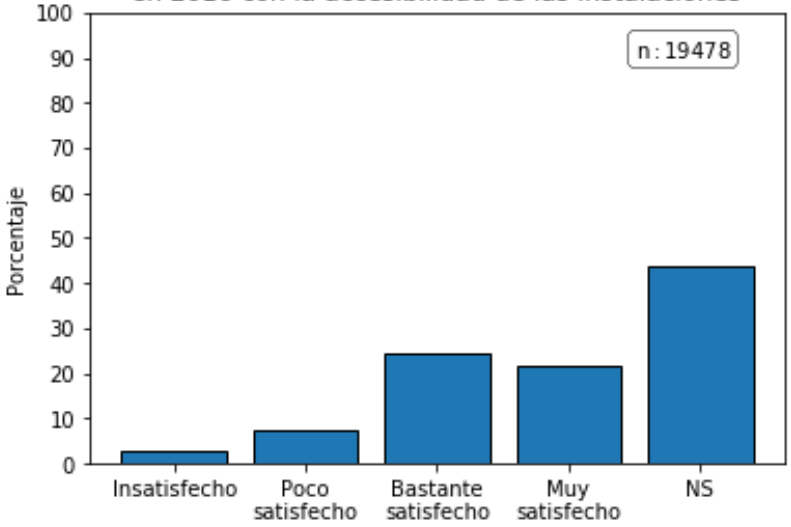


Figura 32. Puntuación media de los aspectos evaluados por los turistas encuestados en 2016

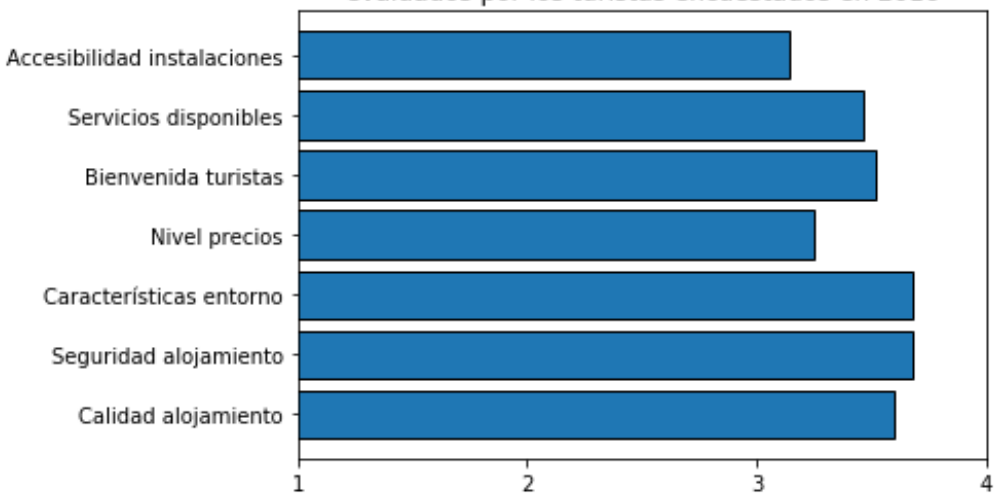
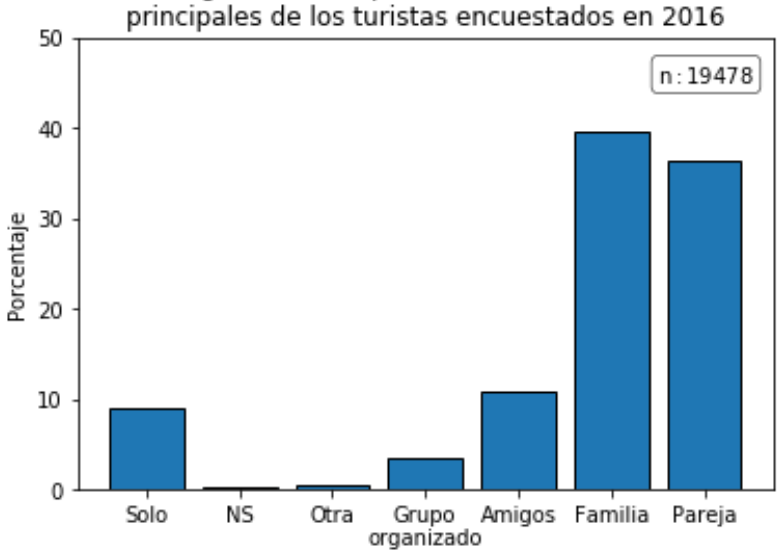


Tabla 31. Porcentaje del tipo de compañía en las vacaciones principales de los turistas encuestados en 2016

	Frecuencia	Porcentaje	Porcentaje Acumulado
Solo	1735	8,91	8,91
NS (No sabe)	31	0,16	9,07
Otra	113	0,58	9,65
Grupo organizado	684	3,51	13,16
Amigos	2129	10,93	24,09
Familia	7713	39,60	63,69
Pareja	7072	36,31	100
N	19478	100	

Figura 33. Compañía en las vacaciones principales de los turistas encuestados en 2016



Anexo 2 – Tablas y gráficos correlación

Tabla 37. Nivel ecológico de los turistas dependiendo del género

		Femenino	Masculino	All
No ecológico	**%**	**61,00**	**61,80**	**61,30**
	N	*6786*	*5158*	*11944*
Ecológico	**%**	**39,00**	**38,20**	**38,70**
	N	*4280*	*3254*	*7534*
TOTAL	**%**	**100**	**100**	**100**
	N	*11066*	*8412*	*19478*

Chi^2 = 1,427; p-value = 0,232. Fuente: Elaboración propia

Figura 39. Porcentaje de nivel ecológico de los turistas dependiendo del género

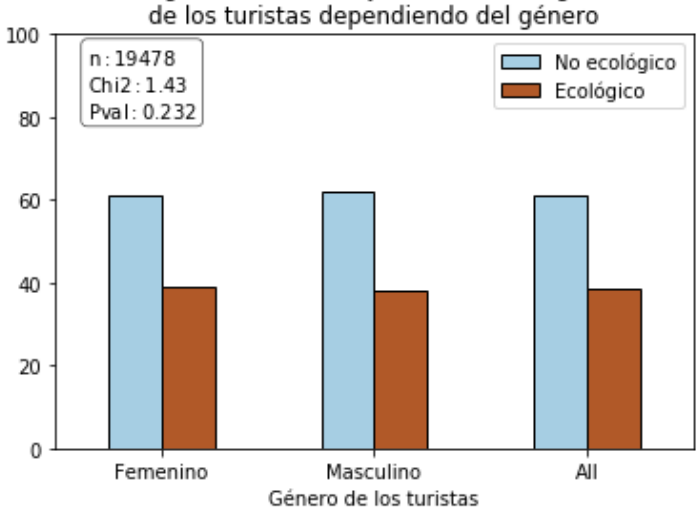


Tabla 38. Nivel ecológico de los turistas dependiendo del tipo de oficio

		No trabajan	Empleados	Autónomos	All
No ecológico	%	**61,20**	**61,50**	**60,90**	**61,30**
	N	*5328*	*5363*	*1253*	*11944*
Ecológico	%	**38,80**	**38,50**	**39,10**	**38,70**
	N	*3360*	*3383*	*791*	*7534*
TOTAL	%	**100**	**100**	**100**	**100**
	N	*8688*	*8746*	*2044*	*19478*

Chi^2 = 0,401; p-value = 0,818. Fuente: Elaboración propia

Figura 40. Porcentaje de nivel ecológico de los turistas dependiendo del oficio

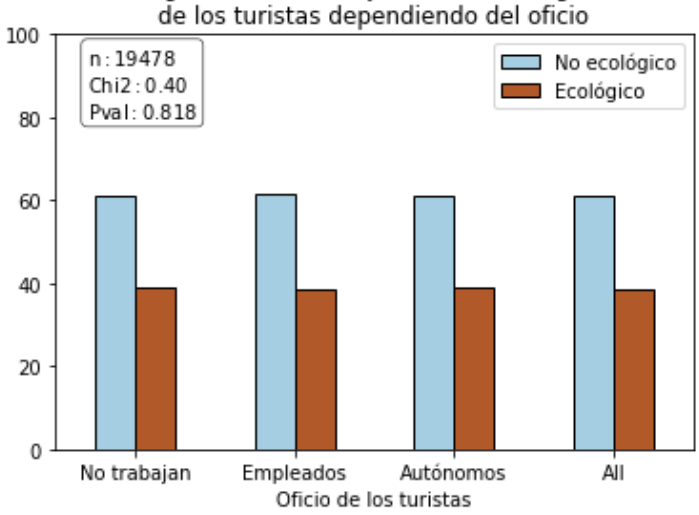


Tabla 39. Nivel ecológico de los turistas dependiendo de su lugar de residencia

		Ciudad	Zona Rural	Pueblo	All
No ecológico	%	**60,60**	**61,80**	**61,60**	**61,30**
	N	*3901*	*3419*	*4532*	*11852*
Ecológico	%	**39,40**	**38,20**	**38,40**	**38,70**
	N	*2460*	*2157*	*2858*	*7475*
TOTAL	%	**100**	**100**	**100**	**100**
	N	*6361*	*5576*	*7390*	*19327*

Chi^2 = 2,219; p-value = 0,330. Fuente: Elaboración propia

Figura 41. Porcentaje de nivel ecológico de los turistas dependiendo del lugar de residencia

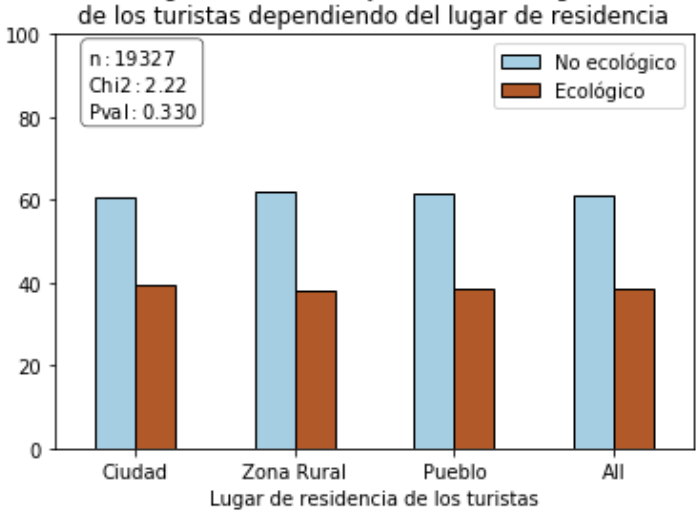


Tabla 41. Nivel ecológico de los turistas dependiendo de la cantidad de veces que realizaron viajes

		Pocas	Bastantes	Muchas	All
No ecológico	**%**	**60,90**	**62,20**	**65,40**	**61,30**
	N	*8318*	*1914*	*233*	*10465*
Ecológico	**%**	**39,10**	**37,80**	**34,60**	**38,70**
	N	*5256*	*1209*	*148*	*6613*
TOTAL	**%**	**100**	**100**	**100**	**100**
	N	*13574*	*3123*	*381*	*17078*

Chi^2 = 4,559; p-value = 0,102. Fuente: Elaboración propia

Figura 43. Porcentaje de nivel ecológico de los turistas dependiendo de la cantidad de viajes

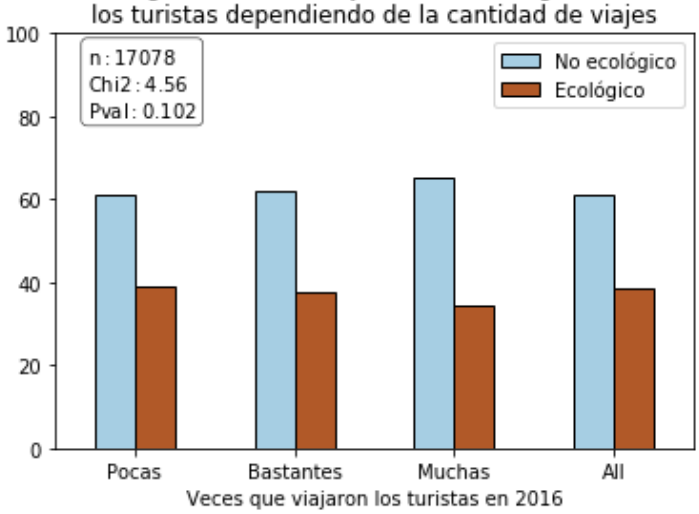


Tabla 44. Nivel ecológico de los turistas dependiendo de la cantidad de veces que realizaron viajes de corta duración

		Ninguna	Pocas	Bastantes	Muchas	All
No ecológico	**%**	**61,90**	**60,90**	**61,00**	**62,80**	**61,30**
	N	*3821*	*6647*	*903*	*573*	*11944*
Ecológico	**%**	**38,10**	**39,10**	**39,00**	**37,20**	**38,70**
	N	*2410*	*4192*	*570*	*362*	*7534*
TOTAL	**%**	**100**	**100**	**100**	**100**	**100**
	N	*6231*	*10839*	*1473*	*935*	*19478*

Chi^2 = 2,844; p-value = 0,416. Fuente: Elaboración propia

Figura 46. Porcentaje de nivel ecológico de los turistas dependiendo de la cantidad de viajes de corta duración

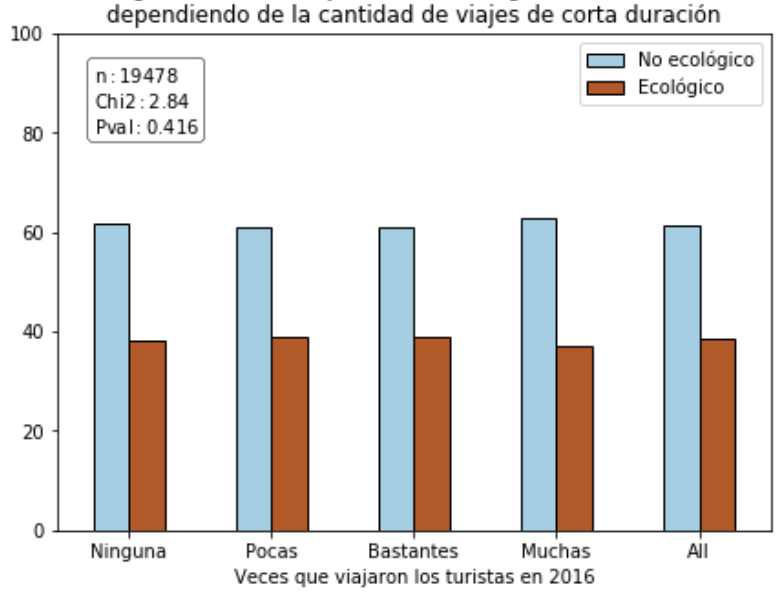
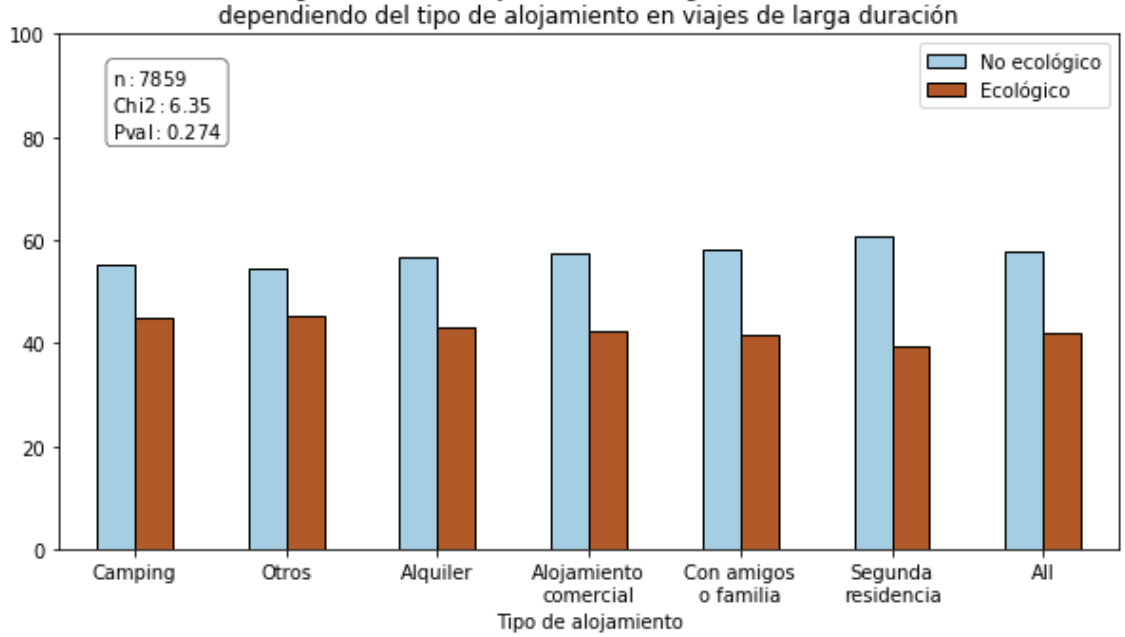


Tabla 45. Nivel ecológico de los turistas dependiendo del tipo de alojamiento en sus viajes de larga duración

		Camping	Otros	Alquiler	Alojamiento comercial	Con amigos o familia	Segunda residencia	All
No ecológico	**%**	**55,10**	**54,60**	**56,90**	**57,60**	**58,20**	**60,60**	**57,80**
	N	*355*	*69*	*762*	*1813*	*910*	*632*	*4541*
Ecológico	**%**	**44,90**	**45,40**	**43,10**	**42,40**	**41,80**	**39,40**	**42,20**
	N	*260*	*50*	*557*	*1324*	*665*	*462*	*3318*
TOTAL	**%**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	N	*615*	*119*	*1319*	*3137*	*1575*	*1094*	*7859*

Chi^2 = 6,346; p-value = 0,274. Fuente: Elaboración propia

Figura 47. Porcentaje de nivel ecológico de los turistas dependiendo del tipo de alojamiento en viajes de larga duración



Anexo 3 – TFM_1.py

```
# -*- coding: utf-8 -*-
"""
Created on Wed Mar 18 12:45:36 2020

@author: ncosn
"""

#load basiclibraries
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from pandas.api.types import CategoricalDtype

# Get working directory
os.getcwd()

# Change working directory
os.chdir("C:/Users/ncosn/Downloads")
os.getcwd()

# Reads data from SPSS file and stores it in a dataframe called df
df = pd.read_spss("ZA6654_v1-0-0.sav")
print(df.head())
print(df.shape)

copy = df

##### DATA CLEANING AND PREPARATION #####

# 1. Elimination of some unuseful variables
df = df.iloc[:, 9:]
df = df.drop(df.columns[1], axis = 1)
df = df.drop(df.columns[15:34], axis = 1)
df = df.drop(df.columns[178:188], axis = 1)
df = df.drop(df.columns[221:400], axis = 1)
```

```
df = df.drop(df.columns[225:302], axis = 1)
df = df.drop(df.columns[225:228], axis = 1)
df = df.drop(df.columns[227:267], axis = 1)
df = df.drop(df.columns[228:], axis = 1)
df = df.drop(df.columns[12:15], axis = 1)
df = df.drop(df.columns[219], axis = 1)
df = df.drop(df.columns[221], axis = 1)

# 2. Values modification for each variable/question
q2016 = df
j = q2016.shape[1]

    # q1 -> traveltimes
q2016 = q2016.rename(columns = {"q1":"traveltimes"})
q2016["traveltimes"].replace({"No Travel" : 0}, inplace = True)
print(q2016.traveltimes.isnull().sum()) # Checking for missing values
(NaN = 234, who don't know their traveltimes)
q2016 = q2016[q2016["traveltimes"].notnull()] # Eliminate those who do
not know (234 cases)
q2016 = q2016.reset_index(drop = True)
q2016["traveltimes"] = q2016["traveltimes"].astype(float)

    # q1r -> traveltimes_cat
q2016 = q2016.rename(columns = {"q1r":"traveltimes_cat"})
        # Categories ordering
my_categories=["None", "Once", "Twice", "3 times", "4 or 5 times", "6
to 10 times", "More than 10 times"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["traveltimes_cat"] = q2016.traveltimes_cat.astype(my_cat_type)

    # q2a1 -> long_vac
i = q2016.shape[0]
q2016 = q2016.rename(columns = {"q2a1":"long_vac"})
for row in range(i):
    if(q2016.ix[row,1] == 0): # Filling with 0 for those who did not
travel
        q2016.ix[row,3] = 0

q2016 = q2016[q2016["long_vac"] != "DK"] # Eliminate those who do not
know (193 cases)
q2016 = q2016.reset_index(drop = True)
q2016["long_vac"] = q2016["long_vac"].astype(str).astype(float)

    # q2a1r -> long_vac_cat
q2016 = q2016.rename(columns = {"q2a1r":"long_vac_cat"})
q2016["long_vac_cat"].fillna("None", inplace = True)
        # Categories ordering
my_categories=["None", "Once", "Twice", "3 times", "4 or 5 times", "6
to 10 times", "More than 10 times"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["long_vac_cat"] = q2016.long_vac_cat.astype(my_cat_type)

    # q2a2 -> med_vac
q2016 = q2016.rename(columns = {"q2a2":"med_vac"})
q2016["med_vac"].fillna(0, inplace = True)

q2016 = q2016[q2016["med_vac"] != "DK"] # Eliminate those who do not
know (114 cases)
q2016 = q2016.reset_index(drop = True)
q2016["med_vac"] = q2016["med_vac"].astype(str).astype(float)
```

```
    # q2a2r -> med_vac_cat
q2016 = q2016.rename(columns = {"q2a2r":"med_vac_cat"})
q2016["med_vac_cat"].fillna("None", inplace = True)
        # Categories ordering
my_categories=["None", "Once", "Twice", "3 times", "4 or 5 times", "6
to 10 times", "More than 10 times"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["med_vac_cat"] = q2016.med_vac_cat.astype(my_cat_type)

    # q2a3 -> short_vac
q2016 = q2016.rename(columns = {"q2a3":"short_vac"})
q2016["short_vac"].fillna(0, inplace = True)

q2016 = q2016[q2016["short_vac"] != "DK"] # Eliminate those who do not
know (118 cases)
q2016 = q2016.reset_index(drop = True)
q2016["short_vac"] = q2016["short_vac"].astype(str).astype(float)

    # q2a3r -> short_vac_cat
q2016 = q2016.rename(columns = {"q2a3r":"short_vac_cat"})
q2016["short_vac_cat"].fillna("None", inplace = True)
        # Categories ordering
my_categories=["None", "Once", "Twice", "3 times", "4 or 5 times", "6
to 10 times", "More than 10 times"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["short_vac_cat"] = q2016.short_vac_cat.astype(my_cat_type)

    # q2b_1 -> long_accom
q2016 = q2016.rename(columns = {"q2b_1" : "long_accom"})
my_categories = ["A camp site", "Don't know", "Other type of
accommodation",
                 "Paid but private accommodation",
                 "Paid commercial accommodation",
                 "Staying with friends or relatives",
                 "Your own property or second home", "No travel", "No
long staying"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["long_accom"] = q2016.long_accom.astype(my_cat_type)

i = q2016.shape[0]
for row in range(i):
    if(q2016.ix[row,1] == 0): # Filling with 0 for those who did not
travel
        q2016.ix[row,9] = "No travel"

q2016["long_accom"].fillna("No long staying", inplace = True) #
Filling with "No long staying" for null values

    # q2b_2 -> med_accom
q2016 = q2016.rename(columns = {"q2b_2" : "med_accom"})
my_categories = ["A camp site", "Don't know", "Other type of
accommodation",
                 "Paid but private accommodation",
                 "Paid commercial accommodation",
                 "Staying with friends or relatives",
                 "Your own property or second home", "No travel", "No
medium staying"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["med_accom"] = q2016.med_accom.astype(my_cat_type)
```

```
i = q2016.shape[0]
for row in range(i):
    if(q2016.ix[row,1] == 0): # Filling with 0 for those who did not
travel
        q2016.ix[row,10] = "No travel"

q2016["med_accom"].fillna("No medium staying", inplace = True) #
Filling with "No medium staying" for null values


    # q2b_3 -> short_accom
q2016 = q2016.rename(columns = {"q2b_3" : "short_accom"})
my_categories = ["A camp site", "Don't know", "Other type of
accommodation",
                 "Paid but private accommodation",
                 "Paid commercial accommodation",
                 "Staying with friends or relatives",
                 "Your own property or second home", "No travel", "No
short staying"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["short_accom"] = q2016.short_accom.astype(my_cat_type)

i = q2016.shape[0]
for row in range(i):
    if(q2016.ix[row,1] == 0): # Filling with 0 for those who did not
travel
        q2016.ix[row,11] = "No travel"

q2016["short_accom"].fillna("No short staying", inplace = True) #
Filling with "No short staying" for null values

    # traveltimes reconsideration (if no long_vac, no med_vac and no
short_vac, traveltimes = 0)
i = q2016.shape[0]
for row in range(i):
    if(q2016.ix[row,3] == 0 and q2016.ix[row,5] == 0 and
q2016.ix[row,7] == 0):
        q2016.ix[row,1] = 0
        q2016.ix[row,2] = "None"

        # Creation of a dataset for those who do not travel
notravel = q2016[q2016["traveltimes"] == 0]

        # Removing from the questionary those who do not travel
q2016 = q2016[q2016["traveltimes"] != 0]
q2016 = q2016.reset_index(drop = True)

    # q4a -> vis_cntry
q2016 = q2016.rename(columns = {"q4a" : "vis_cntry"})
q2016["vis_cntry"].replace({"In [OUR COUNTRY]" : "Country of
residence"}, inplace = True)

    # q5a -> reason_vac_prin
q2016 = q2016.rename(columns = {"q5a" : "reason_vac_prin"})

    # q5b.1 -> reason_vac_1
q2016 = q2016.rename(columns = {"q5b.1" : "reason_vac_1"})
my_categories = ["City trips", "Culture (e.g. religious, gastronomy,
arts)",
```

```
                  "Don't know", "Nature (mountain, lake, landscape,
etc.)",
                  "Other (DO NOT READ OUT)",
                  "Specific events (sporting
events/festivals/clubbing)",
                  "Sport-related activities", "Sun/beach",
                  "Visiting family/friends/relatives",
                  "Wellness/Spa/health treatment", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["reason_vac_1"] = q2016.reason_vac_1.astype(my_cat_type)

    # q5b.2 -> reason_vac_2
q2016 = q2016.rename(columns = {"q5b.2" : "reason_vac_2"})
my_categories = ["City trips", "Culture (e.g. religious, gastronomy,
arts)",
                  "Don't know", "Nature (mountain, lake, landscape,
etc.)",
                  "Other (DO NOT READ OUT)",
                  "Specific events (sporting
events/festivals/clubbing)",
                  "Sport-related activities", "Sun/beach",
                  "Visiting family/friends/relatives",
                  "Wellness/Spa/health treatment", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["reason_vac_2"] = q2016.reason_vac_2.astype(my_cat_type)

    # q5b.3 -> reason_vac_3
q2016 = q2016.rename(columns = {"q5b.3" : "reason_vac_3"})
my_categories = ["City trips", "Culture (e.g. religious, gastronomy,
arts)",
                  "Don't know", "Nature (mountain, lake, landscape,
etc.)",
                  "Other (DO NOT READ OUT)",
                  "Specific events (sporting
events/festivals/clubbing)",
                  "Sport-related activities", "Sun/beach",
                  "Visiting family/friends/relatives",
                  "Wellness/Spa/health treatment", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["reason_vac_3"] = q2016.reason_vac_3.astype(my_cat_type)

        # Filling of the 3 reasons for going holiday
q2016["q5b.4"].replace({"Sport-related activities (e.g. scuba-diving,
cycli" : "Sport-related activities"},
     inplace = True)
q2016["q5b.5"].replace({"Nature (mountain, lake, landscape etc.)" :
"Nature (mountain, lake, landscape, etc.)"},
     inplace = True)
q2016["q5b.7"].replace({"Visiting family/ friends / relatives" :
"Visiting family/friends/relatives"},
     inplace = True)
q2016["q5b.8"].replace({"Specific events (sporting
events/festivals/clu..." : "Specific events (sporting
events/festivals/clubbing)"},
     inplace = True)

i = q2016.shape[0]
for row in range(i):
```

```
    x = 0
    while x < 3:
        for col in range(155, 165):
            if(q2016.ix[row, col] != "Not mentioned"):
                q2016.ix[row, 155+x] = q2016.ix[row, col]
                if(col > 155):
                    q2016.ix[row, col] = "Not mentioned"
                x+=1
        x = 3

q2016["reason_vac_1"].fillna("Not mentioned", inplace = True)

        # Elimination of the remaining variables related to holiday
reasons
q2016 = q2016.drop(q2016.columns[158:175], axis = 1)

    # q7a -> same_dest
q2016 = q2016.rename(columns = {"q7a" : "same_dest"})

    # q7b.1 -> same_dest_1
q2016 = q2016.rename(columns = {"q7b.1" : "same_dest_1"})
my_categories = ["Accessible facilities for people with special
needs",
                 "Cultural and historical attractions", "Don't know",
                 "How tourists are welcomed",
                 "I don't go back to the same place (DO NOT READ
OUT)",
                 "Other (DO NOT READ OUT)",
                 "The activities/services available",
                 "The general level of prices", "The natural
features",
                 "The quality of the accommodation", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["same_dest_1"] = q2016.same_dest_1.astype(my_cat_type)

    # q7b.2 -> same_dest_2
q2016 = q2016.rename(columns = {"q7b.2" : "same_dest_2"})
my_categories = ["Accessible facilities for people with special
needs",
                 "Cultural and historical attractions", "Don't know",
                 "How tourists are welcomed",
                 "I don't go back to the same place (DO NOT READ
OUT)",
                 "Other (DO NOT READ OUT)",
                 "The activities/services available",
                 "The general level of prices", "The natural
features",
                 "The quality of the accommodation", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["same_dest_2"] = q2016.same_dest_2.astype(my_cat_type)

    # q7b.3 -> same_dest_3
q2016 = q2016.rename(columns = {"q7b.3" : "same_dest_3"})
my_categories = ["Accessible facilities for people with special
needs",
                 "Cultural and historical attractions", "Don't know",
                 "How tourists are welcomed",
                 "I don't go back to the same place (DO NOT READ
OUT)",
```

```
                  "Other (DO NOT READ OUT)",
                  "The activities/services available",
                  "The general level of prices", "The natural
features",
                  "The quality of the accommodation", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["same_dest_3"] = q2016.same_dest_3.astype(my_cat_type)

        # Filling of the 3 reasons to repeat the same destination
i = q2016.shape[0]
for row in range(i):
    x = 0
    while x < 3:
        for col in range(159, 169):
            if(q2016.ix[row, col] != "Not mentioned"):
                q2016.ix[row, 159+x] = q2016.ix[row, col]
                if(col > 159):
                    q2016.ix[row, col] = "Not mentioned"
                x+=1
        x = 3

q2016["same_dest_1"].fillna("Not mentioned", inplace = True)

        # Elimination of the remaining variables related to repetition
reasons
q2016 = q2016.drop(q2016.columns[162:179], axis = 1)

    # q8a_1 -> accom_qual
q2016 = q2016.rename(columns = {"q8a_1" : "accom_qual"})

    # q8a_2 -> accom_sec
q2016 = q2016.rename(columns = {"q8a_2" : "accom_sec"})

    # q8a_3 -> envi_char
q2016 = q2016.rename(columns = {"q8a_3" : "envi_char"})

    # q8a_4 -> price_lvl
q2016 = q2016.rename(columns = {"q8a_4" : "price_lvl"})

    # q8a_5 -> tour_welc
q2016 = q2016.rename(columns = {"q8a_5" : "tour_welc"})

    # q8a_6 -> avail_serv
q2016 = q2016.rename(columns = {"q8a_6" : "avail_serv"})

    # q8a_7 -> spec_inst
q2016 = q2016.rename(columns = {"q8a_7" : "spec_inst"})

    # q8b.1 -> vac_company_1
q2016 = q2016.rename(columns = {"q8b.1" : "vac_company_1"})
my_categories = ["Alone", "With my partner/spouse",
                  "With my family (adults only)", "Don't know",
                  "With my family (including children under 18 years
old)",
                  "With friend(s)", "With an organised group",
"Other", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["vac_company_1"] = q2016.vac_company_1.astype(my_cat_type)
```

```
    # q8b.2 -> vac_company_2
q2016 = q2016.rename(columns = {"q8b.2" : "vac_company_2"})
my_categories = ["Alone", "With my partner/spouse",
                 "With my family (adults only)", "Don't know",
                 "With my family (including children under 18 years
old)",
                 "With friend(s)", "With an organised group",
"Other", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["vac_company_2"] = q2016.vac_company_2.astype(my_cat_type)

    # q8b.3 -> vac_company_3
q2016 = q2016.rename(columns = {"q8b.3" : "vac_company_3"})
my_categories = ["Alone", "With my partner/spouse",
                 "With my family (adults only)", "Don't know",
                 "With my family (including children under 18 years
old)",
                 "With friend(s)", "With an organised group",
"Other", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["vac_company_3"] = q2016.vac_company_3.astype(my_cat_type)

    # q8b.4 -> vac_company_4
q2016 = q2016.rename(columns = {"q8b.4" : "vac_company_4"})
my_categories = ["Alone", "With my partner/spouse",
                 "With my family (adults only)", "Don't know",
                 "With my family (including children under 18 years
old)",
                 "With friend(s)", "With an organised group",
"Other", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["vac_company_4"] = q2016.vac_company_4.astype(my_cat_type)

    # q8b.5 -> vac_company_5
q2016 = q2016.rename(columns = {"q8b.5" : "vac_company_5"})
my_categories = ["Alone", "With my partner/spouse",
                 "With my family (adults only)", "Don't know",
                 "With my family (including children under 18 years
old)",
                 "With friend(s)", "With an organised group",
"Other", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["vac_company_5"] = q2016.vac_company_5.astype(my_cat_type)

    # q8b.6 -> vac_company_6
q2016 = q2016.rename(columns = {"q8b.6" : "vac_company_6"})
my_categories = ["Alone", "With my partner/spouse",
                 "With my family (adults only)", "Don't know",
                 "With my family (including children under 18 years
old)",
                 "With friend(s)", "With an organised group",
"Other", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["vac_company_6"] = q2016.vac_company_6.astype(my_cat_type)

    # q8b.7 -> vac_company_7
```

```
q2016 = q2016.rename(columns = {"q8b.7" : "vac_company_7"})
my_categories = ["Alone", "With my partner/spouse",
                 "With my family (adults only)", "Don't know",
                 "With my family (including children under 18 years
old)",
                 "With friend(s)", "With an organised group",
"Other", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["vac_company_7"] = q2016.vac_company_7.astype(my_cat_type)

    # q8b.8 -> vac_company_8
q2016 = q2016.rename(columns = {"q8b.8" : "vac_company_8"})
my_categories = ["Alone", "With my partner/spouse",
                 "With my family (adults only)", "Don't know",
                 "With my family (including children under 18 years
old)",
                 "With friend(s)", "With an organised group",
"Other", "Not mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["vac_company_8"] = q2016.vac_company_8.astype(my_cat_type)

        # Filling of the travel company
i = q2016.shape[0]
for row in range(i):
    x = 0
    while x < 8:
        for col in range(169, 177):
            if(q2016.ix[row, col] != "Not mentioned"):
                q2016.ix[row, 169+x] = q2016.ix[row, col]
                if(col > 169):
                    q2016.ix[row, col] = "Not mentioned"
                x+=1
        x = 8

        # Elimination of the remaining variables related to travel
company
q2016 = q2016.drop(q2016.columns[173:177], axis = 1)

    # q8c.1 -> eco_aspect_1
q2016 = q2016.rename(columns = {"q8c.1" : "eco_aspect_1"})
my_categories = ["Local destination eco-friendly practices",
"Accommodation eco-friendly practices",
                 "Destination accesible with transport with low
impact",
                 "Certified place", "Other (DO NOT READ OUT)",
                 "None of these aspects", "Don't know", "Not
mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["eco_aspect_1"] = q2016.eco_aspect_1.astype(my_cat_type)

    # q8c.2 -> eco_aspect_2
q2016 = q2016.rename(columns = {"q8c.2" : "eco_aspect_2"})
my_categories = ["Local destination eco-friendly practices",
"Accommodation eco-friendly practices",
                 "Destination accesible with transport with low
impact",
                 "Certified place", "Other (DO NOT READ OUT)",
```

```
                  "None of these aspects", "Don't know", "Not
mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["eco_aspect_2"] = q2016.eco_aspect_2.astype(my_cat_type)

    # q8c.3 -> eco_aspect_3
q2016 = q2016.rename(columns = {"q8c.3" : "eco_aspect_3"})
my_categories = ["Local destination eco-friendly practices",
"Accommodation eco-friendly practices",
                  "Destination accesible with transport with low
impact",
                  "Certified place", "Other (DO NOT READ OUT)",
                  "None of these aspects", "Don't know", "Not
mentioned"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["eco_aspect_3"] = q2016.eco_aspect_3.astype(my_cat_type)

        # Filling of the 3 ecological aspects to choose the holidays
q2016["eco_aspect_1"].fillna("Local destination eco-friendly
practices", inplace = True)
q2016["eco_aspect_2"].fillna("Accommodation eco-friendly practices",
inplace = True)
q2016["eco_aspect_3"].fillna("Destination accesible with transport
with low impact", inplace = True)
q2016["q8c.4"].replace({"certified place" : "Certified place"},
inplace = True)

i = q2016.shape[0]
for row in range(i):
    x = 0
    while x < 3:
        for col in range(173, 180):
            if(q2016.ix[row, col] != "Not mentioned"):
                q2016.ix[row, 173+x] = q2016.ix[row, col]
                if(col > 173):
                    q2016.ix[row, col] = "Not mentioned"
                x+=1
        x = 3

        # Elimination of the remaining variables related to
eco_aspects
q2016 = q2016.drop(q2016.columns[177:180], axis = 1)

    # q8c.4 -> eco_tourist (target variable)
q2016 = q2016.rename(columns = {"q8c.4" : "eco_tourist"})
my_categories = ["No ecological", "Not very ecological", "Ecological",
"Very ecological"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["eco_tourist"] = q2016.eco_tourist.astype(my_cat_type)

        # Creation of the new target variable
i = q2016.shape[0]
for row in range(i):
    x = 0
    for col in range(173, 176):
        if(q2016.ix[row, col] == "Not mentioned" or q2016.ix[row, col]
== "None of these aspects"
           or q2016.ix[row, col] == "Don't know"):
            x+=1
```

```
    if(x == 3):
        q2016.ix[row, 176] = "No ecological"
    if(x == 2):
        q2016.ix[row, 176] = "Not very ecological"
    if(x == 1):
        q2016.ix[row, 176] = "Ecological"
    if(x == 0):
        q2016.ix[row, 176] = "Very ecological"

    # d1 -> age
q2016 = q2016.rename(columns = {"d1":"age"})

q2016["age"] = q2016["age"].astype(str)
q2016["age"].replace({"15 years" : "15"}, inplace = True)
q2016["age"].replace({"98 years" : "98"}, inplace = True)
q2016["age"] = q2016["age"].astype(float)

    # d1r2 -> age_cat
q2016 = q2016.rename(columns = {"d1r2":"age_cat"})

    # d2 -> sex
q2016 = q2016.rename(columns = {"d2":"sex"})

    # d5r -> job
q2016 = q2016.rename(columns = {"d5r":"job"})

    # d13 -> living_place
q2016 = q2016.rename(columns = {"d13":"living_place"})
```

Anexo 4 – TFM_Analysis.py

```
# -*- coding: utf-8 -*-
"""
Created on Wed Mar 25 10:49:20 2020

@author: ncosn
"""

#load basiclibraries
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from pandas.api.types import CategoricalDtype

# Get working directory
os.getcwd()

# Change working directory
os.chdir("C:/Users/ncosn/Downloads")
os.getcwd()

# Reads data from SPSS file and stores it in a dataframe called q2016
q2016 = pd.read_spss("quest_2016_r.sav")
i = 1
##### DESCRIPTIVE ANALYSIS #####

# Demographic variables
    # 1. Age
data=q2016.age.describe()
```

```
mean = round(data[1], 1) # Store the mean
std = round(data[2], 1) # Store the standard deviation
n = round(data[0], 1)
Q1 = data[4]
Q3 = data[6]
print(data)

# We check if there are outliers
plt.boxplot(q2016.age.dropna(), patch_artist = True,
            vert = False, labels = ["Edad"])
plt.title('Figura '+str(i)+'. Comprobación de outliers para la
variable "age"')
plt.show()
i+=1

max_value = Q3 + 1.5 * (Q3 - Q1)
min_value = Q1 - 1.5 * (Q3 - Q1)

print("max_value = ", max_value)
print("min_value = ", min_value)

print("Range of acceptable values is between ", min_value, "and ",
max_value)

# We create the histogram
plt.hist(q2016.age, edgecolor ='black')
plt.xticks(np.arange (0, 110, step=10))
plt.title('Figura '+str(i)+'. Distribución de edades''\n' 'de los
turistas encuestados en 2016')
plt.ylabel('Frecuencia')
plt.xlabel('Edad')
        # Add reference lines and store their names in label for later
legend
plt.axvline(x = mean, linewidth = 1, linestyle = "solid", color =
"Red", label = "Mean")
plt.axvline(x = mean-std, linewidth = 1, linestyle = "--", color =
"Cyan", label = "-1 S.D.")
plt.axvline(x = mean+std, linewidth = 1, linestyle = "--", color =
"Cyan", label = "+1 S.D.")
        # Add the legends (mean, std deviation and total of counts)
plt.legend(loc = "upper left", bbox_to_anchor = (0.73, 0.95)) # Mean
and Std Deviation

props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.85,
textstr, bbox = props)

plt.show()
i+=1

    # 2. Age_cat
print(q2016.age_cat.describe())
# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["age_cat"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
```

```
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=('15-24','25-34', '35-44', '45-54', '55-64', '>65')
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.xlabel("Años")
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Distribución de edades de''\n''por
categoría de los turistas encuestados en 2016')

props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(4.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 3. sex
print(q2016.sex.describe())
# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["sex"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=('Femenino','Masculino')
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
plt.xlabel("Sexo")
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Género de los turistas encuestados en
2016')

props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(1, 90, textstr, bbox = props)

plt.show()
i+=1

    # 4. job
print(q2016.job.describe())
# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["job"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=('Empleados','Trabajadores''\n''manuales','No
trabajan','Rehúsan','Autónomos')
plt.yticks(np.arange(0, 70, step = 10))
plt.xticks(mytable.index,objects)
plt.xlabel("Oficio")
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Situación laboral de''\n''los turistas
encuestados en 2016')
```

```
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 50, textstr, bbox = props)

plt.show()
i+=1

    # 5. living_place
print(q2016.living_place.describe())
# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["living_place"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=('NS','Ciudad','Zona Rural','Pueblo')
plt.yticks(np.arange(0, 70, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Lugar de residencia''\n''de los
turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(2.7, 50, textstr, bbox = props)

plt.show()
i+=1

    # 6. isocntry
print(q2016.isocntry.describe())
# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["isocntry"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize = (10,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=('Austria', 'Bélgica', 'Bulgaria', 'Croacia', 'Chipre',
'República Checa',
         'Dinamarca', 'Estonia', 'Finlandia', 'Francia', 'Alemania',
'Grecia', 'Hungría',
         'Islandia', 'Irlanda', 'Italia', 'Letonia', 'Lituania',
'Luxemburgo', 'Macedonia',
         'Malta', 'Moldavia', 'Montenegro', 'Polonia', 'Portugal',
'Rumanía', 'Eslovaquia',
         'Eslovenia', 'España', 'Suecia', 'Holanda', 'Turquía', 'Reino
Unido')
plt.yticks(np.arange(0, 11, step = 1))
plt.xticks(mytable.index,objects, rotation = 90)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Nacionalidad de los turistas
encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(29, 9, textstr, bbox = props)
```

```
plt.show()
i+=1

    # 7. traveltimes
data=q2016.traveltimes.describe()
mean = round(data[1], 1) # Store the mean
std = round(data[2], 1) # Store the standard deviation
n = round(data[0], 1)
Q1 = data[4]
Q3 = data[6]
print(data)

# We check if there are outliers
plt.boxplot(q2016.traveltimes.dropna(), patch_artist = True,
            vert = False, labels = ["N° viajes/""\n""vacaciones"])
plt.title('Figura '+str(i)+'. Comprobación de outliers para la
variable "traveltimes"')
plt.show()
i+=1

max_value = Q3 + 1.5 * (Q3 - Q1)
min_value = Q1 - 1.5 * (Q3 - Q1)

print("max_value = ", max_value)
print("min_value = ", min_value)

print("Range of acceptable values is between ", min_value, "and ",
max_value)

# We remove the outliers
df = q2016[q2016["traveltimes"]<=max_value]
df = df[df["traveltimes"]>=min_value]

data=df.traveltimes.describe()
mean = round(data[1], 1) # Store the mean
std = round(data[2], 1) # Store the standard deviation
n = round(data[0], 1)
Q1 = data[4]
Q3 = data[6]
print(data)

# We create the histogram
plt.hist(df.traveltimes, edgecolor ='black')
plt.xticks(np.arange (0, 16, step=1))
plt.title('Figura '+str(i)+'. N° veces que viajaron''\n''el año
anterior los turistas encuestados en 2016')
plt.ylabel('Frecuencia')
plt.xlabel('Cantidad de veces')
        # Add reference lines and store their names in label for later
legend
plt.axvline(x = mean, linewidth = 1, linestyle = "solid", color =
"Red", label = "Mean")
plt.axvline(x = mean-std, linewidth = 1, linestyle = "--", color =
"Cyan", label = "-1 S.D.")
plt.axvline(x = mean+std, linewidth = 1, linestyle = "--", color =
"Cyan", label = "+1 S.D.")
        # Add the legends (mean, std deviation and total of counts)
plt.legend(loc = "upper left", bbox_to_anchor = (0.73, 0.95)) # Mean
and Std Deviation
```

```
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.8, ymin + (ymax - ymin)*0.6, textstr,
bbox = props)

plt.show()
i+=1

    # 8. traveltimes_cat
# Categories ordering
my_categories=["None", "Once", "Twice", "3 times", "4 or 5 times", "6
to 10 times", "More than 10 times"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
df["traveltimes_cat"] = df.traveltimes_cat.astype(my_cat_type)

print(df.traveltimes_cat.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["traveltimes_cat"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("1 vez", "2 veces", "3 veces", "4 o 5 ""\n""veces", "6 - 10
""\n""veces", "> 10 veces")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. N° veces por categoría que
viajaron''\n''el año anterior los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(4.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 9. long_vac_cat
# Categories ordering
my_categories=["None", "Once", "Twice", "3 times", "4 or 5 times", "6
to 10 times", "More than 10 times"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["long_vac_cat"] = q2016.long_vac_cat.astype(my_cat_type)

print(q2016.long_vac_cat.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["long_vac_cat"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Ninguna", "1 vez", "2 veces", "3 veces", "4 o 5
""\n""veces", "6 - 10 ""\n""veces", "> 10 veces")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
```

```
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. N° veces por categoría que viajaron (>
13 noches)''\n'' el año anterior los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5.5, 90, textstr, bbox = props)

plt.show()
i+=1

    # 10. med_vac_cat
# Categories ordering
my_categories=["None", "Once", "Twice", "3 times", "4 or 5 times", "6
to 10 times", "More than 10 times"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["med_vac_cat"] = q2016.med_vac_cat.astype(my_cat_type)

print(q2016.med_vac_cat.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["med_vac_cat"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Ninguna", "1 vez", "2 veces", "3 veces", "4 o 5
""\n""veces", "6 - 10 ""\n""veces", "> 10 veces")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. N° veces por categoría que viajaron (4-
13 noches)''\n'' el año anterior los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 11. short_vac_cat
# Categories ordering
my_categories=["None", "Once", "Twice", "3 times", "4 or 5 times", "6
to 10 times", "More than 10 times"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["short_vac_cat"] = q2016.short_vac_cat.astype(my_cat_type)

print(q2016.short_vac_cat.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["short_vac_cat"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Ninguna", "1 vez", "2 veces", "3 veces", "4 o 5
""\n""veces", "6 - 10 ""\n""veces", "> 10 veces")
plt.yticks(np.arange(0, 60, step = 10))
```

```
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. N° veces por categoría que viajaron (<
3 noches)''\n'' el año anterior los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 12. long_accom
# Categories ordering
my_categories = ["A camp site", "Don't know", "Other type of
accommodation",
                 "Paid but private accommodation",
                 "Paid commercial accommodation",
                 "Staying with friends or relatives",
                 "Your own property or second home", "No travel", "No
long staying"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["long_accom"] = q2016.long_accom.astype(my_cat_type)

df = q2016[q2016["long_accom"]!="No long staying"]
print(df.long_accom.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["long_accom"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(9,5))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Camping", "NS", "Otros", "Alquiler",
"Alojamiento""\n""comercial", "Con amigos""\n""o familia",
"Segunda""\n""residencia")
plt.yticks(np.arange(0, 70, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Tipo de alojamiento''\n''para viajes de
larga duración (> 13 noches)')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5.5, 50, textstr, bbox = props)

plt.show()
i+=1

    # 13. med_accom
# Categories ordering
my_categories = ["A camp site", "Don't know", "Other type of
accommodation",
                 "Paid but private accommodation",
                 "Paid commercial accommodation",
                 "Staying with friends or relatives",
                 "Your own property or second home", "No travel", "No
medium staying"]
```

```
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["med_accom"] = q2016.med_accom.astype(my_cat_type)

df = q2016[q2016["med_accom"]!="No medium staying"]
print(df.med_accom.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["med_accom"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(9,5))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Camping", "NS", "Otros", "Alquiler",
"Alojamiento""\n""comercial", "Con amigos""\n""o familia",
"Segunda""\n""residencia")
plt.yticks(np.arange(0, 70, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Tipo de alojamiento''\n''para viajes de
media duración (4-13 noches)')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5.5, 50, textstr, bbox = props)

plt.show()
i+=1

    # 14. short_accom
# Categories ordering
my_categories = ["A camp site", "Don't know", "Other type of
accommodation",
                 "Paid but private accommodation",
                 "Paid commercial accommodation",
                 "Staying with friends or relatives",
                 "Your own property or second home", "No travel", "No
medium staying"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["short_accom"] = q2016.short_accom.astype(my_cat_type)

df = q2016[q2016["short_accom"]!="No short staying"]
print(df.short_accom.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["short_accom"], columns="count")
n=mytable.sum()
#mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(9,5))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Camping", "NS", "Otros", "Alquiler",
"Alojamiento""\n""comercial", "Con amigos""\n""o familia",
"Segunda""\n""residencia")
plt.yticks(np.arange(0, 70, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
```

```
plt.title ('Figura '+str(i)+'. Tipo de alojamiento''\n''para viajes de
corta duración (< 3 noches)')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5.5, 55, textstr, bbox = props)

plt.show()
i+=1

    # 15. vis_cntry
print(q2016.vis_cntry.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["vis_cntry"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)

# We select the top 10 countries
mytable = mytable.sort_values("count", ascending = False)
mytable = mytable.reset_index()
df = mytable[:11].copy()
df = df[df["vis_cntry"]!="Other"]
df = df.reset_index(drop = True)

# We create the others
new_row = pd.DataFrame(data = {'vis_cntry' : ['Others'], 'count' :
[mytable['count'][11:].sum()+mytable['count'][2]]})

# We combine the top 10 with others
df = pd.concat([df, new_row])
# We create a barchart with the percentages
labels = ['País de residencia', 'España', 'Italia', 'Francia',
          'Grecia', 'Croacia', 'Alemania', 'EE.UU./Canadá',
          'Asia/Oceania', 'Reino Unido', 'Otros']
sizes = np.array(df["count"])
colors = ['indianred', 'gold', 'lightskyblue', 'hotpink',
'darkorchid', 'teal', 'burlywood', 'silver', 'darkorange',
'springgreen', 'royalblue']
explode = (0.05,0.05,0.05,0.05,0.05,0.05,0.05,0.05,0.05,0.05,0.05)
plt.pie(sizes, explode=explode, labels=labels, colors=colors,
autopct='%1.1f%%', shadow=True, startangle=90, pctdistance = 0.85)
#draw circle
centre_circle = plt.Circle((0,0),0.70,fc='white')
fig = plt.gcf()
fig.gca().add_artist(centre_circle)
# Set aspect ratio to be equal so that pie is drawn as a circle.
plt.axis('equal')
plt.tight_layout()
plt.title('Figura '+str(i)+'. País visitado en las
vacaciones''\n''principales de los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(-1.5, 0.8, textstr, bbox = props)
plt.show()

i+=1

    # 16. reason_vac_prin
```

```
print(q2016.reason_vac_prin.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["reason_vac_prin"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(10.5,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Viajes""\n""a ciudades", "Cultura", "NS", "Naturaleza",
"Otras", "Eventos""\n""concretos",
         "Actividades""\n""deportivas", "Sol/Playa",
"Visitar""\n""familia", "Terapia""\n""salud")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Razón principal por la que viajan los
turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(8.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 17. reason_vac_1
df = q2016[q2016["reason_vac_1"]!="Not mentioned"]
print(df.reason_vac_1.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["reason_vac_1"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(10,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Viajes""\n""a ciudades", "Cultura", "NS", "Naturaleza",
"Otras", "Eventos""\n""concretos",
         "Actividades""\n""deportivas","Visitar""\n""familia",
"Terapia""\n""salud")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Segunda razón por la que viajan los
turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(7.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 18. reason_vac_2
df = q2016[q2016["reason_vac_2"]!="Not mentioned"]
print(df.reason_vac_2.describe())

# We obtain the percentages for each class
```

```
mytable=pd.crosstab( index=df["reason_vac_2"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(10,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Viajes""\n""a ciudades", "Cultura", "Naturaleza", "Otras",
"Eventos""\n""concretos",
         "Actividades""\n""deportivas","Visitar""\n""familia")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Tercera razón por la que viajan los
turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(6, 40, textstr, bbox = props)

plt.show()
i+=1

    # 19. reason_vac_3
df = q2016[q2016["reason_vac_3"]!="Not mentioned"]
print(df.reason_vac_3.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["reason_vac_3"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(10,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Cultura", "Naturaleza", "Otras", "Eventos""\n""concretos",
         "Actividades""\n""deportivas","Visitar""\n""familia")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Cuarta razón por la que viajan los
turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 20. same_dest
print(q2016.same_dest.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["same_dest"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(10.5,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
```

```
objects=("Accesibilidad""\n""instalaciones", "Cultura", "NS",
"Bienvenida""\n""a turistas", "No repite",
         "Otros", "Actividades""\n""y servicios", "Precios",
"Aspectos""\n""naturales", "Calidad""\n""alojamiento")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Motivo principal por el que repiten
destino los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(8.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 21. same_dest_1
df = q2016[q2016["same_dest_1"]!="Not mentioned"]
print(df.same_dest_1.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["same_dest_1"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(10.5,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Accesibilidad""\n""discapacitados", "Cultura", "NS",
"Bienvenida""\n""a turistas", "No repite",
         "Otros", "Actividades""\n""y servicios", "Precios",
"Aspectos""\n""naturales", "Calidad""\n""alojamiento")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Segundo motivo por el que repiten
destino los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(8.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 22. same_dest_2
df = q2016[q2016["same_dest_2"]!="Not mentioned"]
print(df.same_dest_2.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["same_dest_2"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(10.5,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Accesibilidad""\n""discapacitados", "Cultura",
"Bienvenida""\n""a turistas",
         "Otros", "Actividades""\n""y servicios", "Precios")
```

```
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Tercer motivo por el que repiten
destino los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(4.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 23. same_dest_3
df = q2016[q2016["same_dest_3"]!="Not mentioned"]
print(df.same_dest_3.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["same_dest_3"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(10.5,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Accesibilidad""\n""discapacitados", "Cultura",
"Bienvenida""\n""a turistas",
         "Otros", "Actividades""\n""y servicios")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Cuarto motivo por el que repiten
destino los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 40, textstr, bbox = props)

plt.show()
i+=1

    # 24. accom_qual
# Categories ordering
my_categories = ["Not at all satisfied", "Not very satisfied", "Fairly
satisfied",
                 "Very satisfied", "Don't know"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["accom_qual"] = q2016.accom_qual.astype(my_cat_type)
print(q2016.accom_qual.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["accom_qual"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Insatisfecho", "Poco""\n""satisfecho",
"Bastante""\n""satisfecho", "Muy""\n""satisfecho", "NS")
plt.yticks(np.arange(0, 110, step = 10))
```

```
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Nivel de satisfacción de los
turistas''\n''encuestados en 2016 con la calidad del alojamiento')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 90, textstr, bbox = props)

plt.show()
i+=1

    # 25. accom_sec
# Categories ordering
my_categories = ["Not at all satisfied", "Not very satisfied", "Fairly
satisfied",
                 "Very satisfied", "Don't know"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["accom_sec"] = q2016.accom_sec.astype(my_cat_type)
print(q2016.accom_sec.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["accom_sec"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Insatisfecho", "Poco""\n""satisfecho",
"Bastante""\n""satisfecho", "Muy""\n""satisfecho", "NS")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Nivel de satisfacción de los
turistas''\n''encuestados en 2016 con la seguridad del alojamiento')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 90, textstr, bbox = props)

plt.show()
i+=1

    # 26. envi_char
# Categories ordering
my_categories = ["Not at all satisfied", "Not very satisfied", "Fairly
satisfied",
                 "Very satisfied", "Don't know"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["envi_char"] = q2016.envi_char.astype(my_cat_type)
print(q2016.envi_char.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["envi_char"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
```

```
objects=("Insatisfecho", "Poco""\n""satisfecho",
"Bastante""\n""satisfecho", "Muy""\n""satisfecho", "NS")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Nivel de satisfacción de los
turistas''\n''encuestados en 2016 con las características del
entorno')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 90, textstr, bbox = props)

plt.show()
i+=1

    # 27. price_lvl
# Categories ordering
my_categories = ["Not at all satisfied", "Not very satisfied", "Fairly
satisfied",
                 "Very satisfied", "Don't know"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["price_lvl"] = q2016.price_lvl.astype(my_cat_type)
print(q2016.price_lvl.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["price_lvl"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Insatisfecho", "Poco""\n""satisfecho",
"Bastante""\n""satisfecho", "Muy""\n""satisfecho", "NS")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Nivel de satisfacción de los
turistas''\n''encuestados en 2016 con el nivel general de precios')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 90, textstr, bbox = props)

plt.show()
i+=1

    # 28. tour_welc
# Categories ordering
my_categories = ["Not at all satisfied", "Not very satisfied", "Fairly
satisfied",
                 "Very satisfied", "Don't know"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["tour_welc"] = q2016.tour_welc.astype(my_cat_type)
print(q2016.tour_welc.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["tour_welc"], columns="count")
n=mytable.sum()
```

```
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Insatisfecho", "Poco""\n""satisfecho",
"Bastante""\n""satisfecho", "Muy""\n""satisfecho", "NS")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Nivel de satisfacción de los
turistas''\n''encuestados en 2016 con la bienvenida recibida')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 90, textstr, bbox = props)

plt.show()
i+=1

    # 29. avail_serv
# Categories ordering
my_categories = ["Not at all satisfied", "Not very satisfied", "Fairly
satisfied",
                 "Very satisfied", "Don't know"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["avail_serv"] = q2016.avail_serv.astype(my_cat_type)
print(q2016.avail_serv.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["avail_serv"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Insatisfecho", "Poco""\n""satisfecho",
"Bastante""\n""satisfecho", "Muy""\n""satisfecho", "NS")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Nivel de satisfacción de los
turistas''\n''encuestados en 2016 con los servicios disponibles')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 90, textstr, bbox = props)

plt.show()
i+=1

    # 30. spec_inst
# Categories ordering
my_categories = ["Not at all satisfied", "Not very satisfied", "Fairly
satisfied",
                 "Very satisfied", "Don't know"]
my_cat_type = CategoricalDtype(categories=my_categories,
ordered=False)
q2016["spec_inst"] = q2016.spec_inst.astype(my_cat_type)
print(q2016.spec_inst.describe())
```

```
# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["spec_inst"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Insatisfecho", "Poco""\n""satisfecho",
"Bastante""\n""satisfecho", "Muy""\n""satisfecho", "NS")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Nivel de satisfacción de los turistas
encuestados''\n''en 2016 con la accesibilidad de las instalaciones')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(3.5, 90, textstr, bbox = props)

plt.show()
i+=1

        # Global evaluation
df = q2016.iloc[:,21:28]
df = df.astype(str)
df["accom_qual"].replace({"Not at all satisfied" : 1, "Not very
satisfied" : 2,
  "Fairly satisfied" : 3, "Very satisfied" : 4, "Don't know" :
np.NaN}, inplace = True)
df["accom_sec"].replace({"Not at all satisfied" : 1, "Not very
satisfied" : 2,
  "Fairly satisfied" : 3, "Very satisfied" : 4, "Don't know" :
np.NaN}, inplace = True)
df["envi_char"].replace({"Not at all satisfied" : 1, "Not very
satisfied" : 2,
  "Fairly satisfied" : 3, "Very satisfied" : 4, "Don't know" :
np.NaN}, inplace = True)
df["price_lvl"].replace({"Not at all satisfied" : 1, "Not very
satisfied" : 2,
  "Fairly satisfied" : 3, "Very satisfied" : 4, "Don't know" :
np.NaN}, inplace = True)
df["tour_welc"].replace({"Not at all satisfied" : 1, "Not very
satisfied" : 2,
  "Fairly satisfied" : 3, "Very satisfied" : 4, "Don't know" :
np.NaN}, inplace = True)
df["avail_serv"].replace({"Not at all satisfied" : 1, "Not very
satisfied" : 2,
  "Fairly satisfied" : 3, "Very satisfied" : 4, "Don't know" :
np.NaN}, inplace = True)
df["spec_inst"].replace({"Not at all satisfied" : 1, "Not very
satisfied" : 2,
  "Fairly satisfied" : 3, "Very satisfied" : 4, "Don't know" :
np.NaN}, inplace = True)
df = df.astype(float)

df["accom_qual"].fillna(df["accom_qual"].mean(), inplace = True)
df["accom_sec"].fillna(df["accom_sec"].mean(), inplace = True)
df["envi_char"].fillna(df["envi_char"].mean(), inplace = True)
df["price_lvl"].fillna(df["price_lvl"].mean(), inplace = True)
df["tour_welc"].fillna(df["tour_welc"].mean(), inplace = True)
df["avail_serv"].fillna(df["avail_serv"].mean(), inplace = True)
```

```
df["spec_inst"].fillna(df["spec_inst"].mean(), inplace = True)

df = pd.DataFrame({"aspects" : ["Calidad alojamiento", "Seguridad
alojamiento",
                                "Características entorno", "Nivel
precios",
                                "Bienvenida turistas", "Servicios
disponibles",
                                "Accesibilidad instalaciones"],
"values" : [df.accom_qual.mean(),
                                df.accom_sec.mean(),
df.envi_char.mean(), df.price_lvl.mean(),
                                df.tour_welc.mean(),
df.avail_serv.mean(), df.spec_inst.mean()]})

plt.barh(df["aspects"], df["values"], edgecolor = "Black")
plt.xticks(np.arange(1, 5, step = 1))
plt.xlim(1, 4)
plt.title ('Figura '+str(i)+'. Puntuación media de los
aspectos''\n''evaluados por los turistas encuestados en 2016')
plt.show()
i+=1

    # 31. vac_company_1
q2016["vac_company_1"].replace({"With my family (adults only)" : "With
my family",
     "With my family (including children under 18 years old)" : "With
my family"}, inplace = True)
print(q2016.vac_company_1.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["vac_company_1"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Solo", "NS", "Otra", "Grupo""\n""organizado", "Amigos",
"Familia", "Pareja")
plt.yticks(np.arange(0, 60, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Compañía en las
vacaciones''\n''principales de los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5.5, 45, textstr, bbox = props)

plt.show()
i+=1

    # 32. eco_aspect_1
print(q2016.eco_aspect_1.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["eco_aspect_1"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
```

```
plt.figure(figsize=(9.5,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Alojamiento incluye""\n""prácticas respetuosas""\n""con el
medio ambiente",
         "Servicios disponen de""\n""un certificado que""\n""indica
prácticas""\n""respetuosas con""\n""el medio ambiente",
         "Destino es accesible""\n""con medio de transporte""\n""con
bajo impacto""\n""en el medio ambiente",
         "NS",
         "Destino incluye""\n""prácticas respetuosas""\n""con el medio
ambiente",
         "Ningún aspecto es""\n""importante al elegir""\n""el destino
de viaje",
         "Otros")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects, rotation = 90)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Principal aspecto ecológico
para''\n''elegir destino vacacional los turistas encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(5.7, 90, textstr, bbox = props)

plt.show()
i+=1

    # 33. eco_aspect_2
df = q2016[q2016["eco_aspect_2"]!="Not mentioned"]
print(df.eco_aspect_2.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=df["eco_aspect_2"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(9.5,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Servicios disponen de""\n""un certificado que""\n""indica
prácticas""\n""respetuosas con""\n""el medio ambiente",
         "Destino es accesible""\n""con medio de transporte""\n""con
bajo impacto""\n""en el medio ambiente",
         "Otros")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects, rotation = 90)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Segundo aspecto ecológico
importante''\n''para elegir destino vacacional los turistas
encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(2, 90, textstr, bbox = props)

plt.show()
i+=1

    # 34. eco_aspect_3
df = q2016[q2016["eco_aspect_3"]!="Not mentioned"]
print(df.eco_aspect_3.describe())
```

```
# We obtain the percentages for each class
mytable=pd.crosstab( index=df["eco_aspect_3"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.figure(figsize=(9.5,4))
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Servicios disponen de""\n""un certificado que""\n""indica
prácticas""\n""respetuosas con""\n""el medio ambiente",
         "Otros")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects, rotation = 90)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Tercer aspecto ecológico
importante''\n''para elegir destino vacacional los turistas
encuestados en 2016')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(1, 90, textstr, bbox = props)

plt.show()
i+=1

    # 35. eco_tourist
# Categories ordering
my_categories=["No ecological", "Not very ecological", "Ecological",
"Very ecological"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["eco_tourist"] = q2016.eco_tourist.astype(my_cat_type)
print(q2016.eco_tourist.describe())

# We obtain the percentages for each class
mytable=pd.crosstab( index=q2016["eco_tourist"], columns="count")
n=mytable.sum()
mytable=(mytable/n)*100
print(mytable)
# We create a barchart with the percentages
plt.bar(mytable.index,mytable['count'],edgecolor='black')
objects=("Nada""\n""ecológico", "Poco""\n""ecológico", "Ecológico",
"Muy""\n""ecológico")
plt.yticks(np.arange(0, 110, step = 10))
plt.xticks(mytable.index,objects)
plt.ylabel('Porcentaje')
plt.title ('Figura '+str(i)+'. Clasificación de los
turistas''\n''encuestados en 2016 según nivel ecológico')
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5) #
Total of counts
textstr = "$\mathrm{n}:%.0f$"%(n)
plt.text(2.5, 90, textstr, bbox = props)

plt.show()
i+=1
```

Anexo 5 – TFM_Correlation.py

```
# -*- coding: utf-8 -*-
"""
```

```
Created on Fri Apr  3 09:33:26 2020

@author: ncosn
"""
#load basiclibraries
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from pandas.api.types import CategoricalDtype
import scipy.stats as stats

# Get working directory
os.getcwd()

# Change working directory
os.chdir("C:/Users/ncosn/Downloads")
os.getcwd()

# Reads data from SPSS file and stores it in a dataframe called q2016
q2016 = pd.read_spss("quest_2016_r.sav")
res = q2016.eco_tourist.describe()
n  = res[0]
i = 38

# Transform the dataset (groups reduction) to obtain realistic results
    # eco_tourist
q2016["eco_tourist"].replace({"Not very ecological" : "Ecológico"},
inplace = True)
q2016["eco_tourist"].replace({"Ecological" : "Ecológico"}, inplace =
True)
q2016["eco_tourist"].replace({"Very ecological" : "Ecológico"},
inplace = True)
q2016["eco_tourist"].replace({"No ecological" : "No ecológico"},
inplace = True)

my_categories=["No ecológico","Ecológico"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["eco_tourist"] = q2016.eco_tourist.astype(my_cat_type)

    # age_cat
q2016["age_cat"].replace({"15 - 24 years" : "15 - 34 años"}, inplace =
True)
q2016["age_cat"].replace({"25 - 34 years" : "15 - 34 años"}, inplace =
True)
q2016["age_cat"].replace({"35 - 44 years" : "35 - 54 años"}, inplace =
True)
q2016["age_cat"].replace({"45 - 54 years" : "35 - 54 años"}, inplace =
True)
q2016["age_cat"].replace({"55 - 64 years" : "55 años o más"}, inplace
= True)
q2016["age_cat"].replace({"65 years and older" : "55 años o más"},
inplace = True)

my_categories=["15 - 34 años", "35 - 54 años", "55 años o más"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["age_cat"] = q2016.age_cat.astype(my_cat_type)

    # job
q2016["job"].replace({"Not working" : "No trabajan"}, inplace = True)
q2016["job"].replace({"Refusal" : "No trabajan"}, inplace = True)
```

```
q2016["job"].replace({"Employees" : "Empleados"}, inplace = True)
q2016["job"].replace({"Manual workers" : "Empleados"}, inplace = True)
q2016["job"].replace({"Self-employed" : "Autónomos"}, inplace = True)

my_categories=["No trabajan", "Empleados", "Autónomos"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["job"] = q2016.job.astype(my_cat_type)

    # traveltimes_cat
q2016["traveltimes_cat"].replace({"None" : "Ninguna"}, inplace = True)
q2016["traveltimes_cat"].replace({"Once" : "Pocas"}, inplace = True)
q2016["traveltimes_cat"].replace({"Twice" : "Pocas"}, inplace = True)
q2016["traveltimes_cat"].replace({"3 times" : "Pocas"}, inplace =
True)
q2016["traveltimes_cat"].replace({"4 or 5 times" : "Pocas"}, inplace =
True)
q2016["traveltimes_cat"].replace({"6 to 10 times" : "Bastantes"},
inplace = True)
q2016["traveltimes_cat"].replace({"More than 10 times" : "Muchas"},
inplace = True)

my_categories=["Ninguna", "Pocas", "Bastantes", "Muchas"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["traveltimes_cat"] = q2016.traveltimes_cat.astype(my_cat_type)

    # long_vac_cat
q2016["long_vac_cat"].replace({"None" : "Ninguna"}, inplace = True)
q2016["long_vac_cat"].replace({"Once" : "Pocas"}, inplace = True)
q2016["long_vac_cat"].replace({"Twice" : "Pocas"}, inplace = True)
q2016["long_vac_cat"].replace({"3 times" : "Pocas"}, inplace = True)
q2016["long_vac_cat"].replace({"4 or 5 times" : "Pocas"}, inplace =
True)
q2016["long_vac_cat"].replace({"6 to 10 times" : "Bastantes"}, inplace
= True)
q2016["long_vac_cat"].replace({"More than 10 times" : "Muchas"},
inplace = True)

my_categories=["Ninguna", "Pocas", "Bastantes", "Muchas"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["long_vac_cat"] = q2016.long_vac_cat.astype(my_cat_type)

    # med_vac_cat
q2016["med_vac_cat"].replace({"None" : "Ninguna"}, inplace = True)
q2016["med_vac_cat"].replace({"Once" : "Pocas"}, inplace = True)
q2016["med_vac_cat"].replace({"Twice" : "Pocas"}, inplace = True)
q2016["med_vac_cat"].replace({"3 times" : "Pocas"}, inplace = True)
q2016["med_vac_cat"].replace({"4 or 5 times" : "Pocas"}, inplace =
True)
q2016["med_vac_cat"].replace({"6 to 10 times" : "Bastantes"}, inplace
= True)
q2016["med_vac_cat"].replace({"More than 10 times" : "Muchas"},
inplace = True)

my_categories=["Ninguna", "Pocas", "Bastantes", "Muchas"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["med_vac_cat"] = q2016.med_vac_cat.astype(my_cat_type)

    # short_vac_cat
q2016["short_vac_cat"].replace({"None" : "Ninguna"}, inplace = True)
q2016["short_vac_cat"].replace({"Once" : "Pocas"}, inplace = True)
q2016["short_vac_cat"].replace({"Twice" : "Pocas"}, inplace = True)
```

```
q2016["short_vac_cat"].replace({"3 times" : "Pocas"}, inplace = True)
q2016["short_vac_cat"].replace({"4 or 5 times" : "Pocas"}, inplace =
True)
q2016["short_vac_cat"].replace({"6 to 10 times" : "Bastantes"},
inplace = True)
q2016["short_vac_cat"].replace({"More than 10 times" : "Muchas"},
inplace = True)

my_categories=["Ninguna", "Pocas", "Bastantes", "Muchas"]
my_cat_type = CategoricalDtype(categories=my_categories, ordered=True)
q2016["short_vac_cat"] = q2016.short_vac_cat.astype(my_cat_type)

    # reason_vac_prin
q2016["reason_vac_prin"].replace({"Nature (mountain, lake, landscape,
etc.)" : "Naturaleza"}, inplace = True)
q2016["reason_vac_prin"].replace({"Sun/beach" : "Sol/Playa"}, inplace
= True)
q2016["reason_vac_prin"].replace({"Culture (e.g. religious,
gastronomy, arts)" : "Cultura"}, inplace = True)
q2016["reason_vac_prin"].replace({"City trips" : "Viajes a ciudades"},
inplace = True)
q2016["reason_vac_prin"].replace({"Visiting family/friends/relatives"
: "Visitar familia"}, inplace = True)
q2016["reason_vac_prin"].replace({"Wellness/Spa/health treatment" :
"Actividades concretas"}, inplace = True)
q2016["reason_vac_prin"].replace({"Specific events (sporting
events/festivals/clubbing)" : "Actividades concretas"}, inplace =
True)
q2016["reason_vac_prin"].replace({"Sport-related activities" :
"Actividades concretas"}, inplace = True)
q2016["reason_vac_prin"].replace({"Other (DO NOT READ OUT)" :
"Otras"}, inplace = True)

    # vac_company_1
q2016["vac_company_1"].replace({"With my family (adults only)" :
"Familia"}, inplace = True)
q2016["vac_company_1"].replace({"With my partner/spouse" : "Pareja"},
inplace = True)
q2016["vac_company_1"].replace({"Alone" : "Solo"}, inplace = True)
q2016["vac_company_1"].replace({"With friend(s)" : "Amigos"}, inplace
= True)
q2016["vac_company_1"].replace({"With my family (including children
under 18 years old)" : "Familia"}, inplace = True)
q2016["vac_company_1"].replace({"With an organised group" : "Grupo
organizado"}, inplace = True)
q2016["vac_company_1"].replace({"Other" : "Otra"}, inplace = True)

##### VARIABLE RELATION (ECO-TOURIST BASED ON...) #####
    # 1. age_cat
    # Descriptive comparison
my_ct = pd.crosstab(q2016.eco_tourist, q2016.age_cat, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(q2016.eco_tourist, q2016.age_cat)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
```

```
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired")
plt.ylim(0, 100)
plt.xticks(rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Edad de los turistas")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico\nde los
turistas dependiendo de la edad')

plt.show()
i+=1

    # 2. sex
    # Descriptive comparison
my_ct = pd.crosstab(q2016.eco_tourist, q2016.sex, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(q2016.eco_tourist, q2016.sex)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired")
plt.ylim(0, 100)
plt.xticks(np.arange(0, 3, step = 1), ["Femenino", "Masculino",
"All"],rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Género de los turistas")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico\nde los
turistas dependiendo del género')
```

```
plt.show()
i+=1

    # 3. job
    # Descriptive comparison
my_ct = pd.crosstab(q2016.eco_tourist, q2016.job, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(q2016.eco_tourist, q2016.job)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired")
plt.ylim(0, 100)
plt.xticks(rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Oficio de los turistas")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico\nde los
turistas dependiendo del oficio')

plt.show()
i+=1

    # 4. living_place
df = q2016[q2016["living_place"]!="DK"]
res = df.living_place.describe()
n = res[0]

    # Descriptive comparison
my_ct = pd.crosstab(df.eco_tourist, df.living_place, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(df.eco_tourist, df.living_place)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
```

```
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired")
plt.ylim(0, 100)
plt.xticks(np.arange(0, 4, step = 1),['Ciudad','Zona Rural','Pueblo',
'All'], rotation = 0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Lugar de residencia de los turistas")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico\nde los
turistas dependiendo del lugar de residencia')

plt.show()
i+=1

    # 5. isocntry
res = q2016.isocntry.describe()
n = res[0]
    # Descriptive comparison
my_ct = pd.crosstab(q2016.eco_tourist, q2016.isocntry, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(q2016.eco_tourist, q2016.isocntry)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

df = my_ct.iloc[:, 0:8]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
df.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (10,5))
plt.ylim(0, 100)
plt.xticks(np.arange(0, 8, step = 1),['Austria', 'Bélgica',
'Bulgaria', 'Croacia', 'Chipre', 'República Checa',
         'Dinamarca', 'Estonia'])
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.25, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
```

```
plt.xlabel("Nacionalidad de los turistas")
plt.title ('Figura '+str(i)+'a. Porcentaje de nivel ecológico de los
turistas dependiendo de su nacionalidad')

plt.show()

df = my_ct.iloc[:, 8:16]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
df.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (10,5))
plt.ylim(0, 100)
plt.xticks(np.arange(0, 8, step = 1),['Finlandia', 'Francia',
'Alemania', 'Grecia', 'Hungría',
         'Islandia', 'Irlanda', 'Italia'])
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Nacionalidad de los turistas")
plt.title ('Figura '+str(i)+'b. Porcentaje de nivel ecológico de los
turistas dependiendo de su nacionalidad')

plt.show()

df = my_ct.iloc[:, 16:24]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
df.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (10,5))
plt.ylim(0, 100)
plt.xticks(np.arange(0, 8, step = 1),['Letonia', 'Lituania',
'Luxemburgo', 'Macedonia',
         'Malta', 'Moldavia', 'Montenegro', 'Polonia'])
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Nacionalidad de los turistas")
plt.title ('Figura '+str(i)+'c. Porcentaje de nivel ecológico de los
turistas dependiendo de su nacionalidad')

plt.show()

df = my_ct.iloc[:, 24:33]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
df.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (10,5))
plt.ylim(0, 100)
```

```
plt.xticks(np.arange(0, 9, step = 1),['Portugal', 'Rumanía',
'Eslovaquia',
         'Eslovenia', 'España', 'Suecia', 'Holanda', 'Turquía', 'Reino
Unido'])
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Nacionalidad de los turistas")
plt.title ('Figura '+str(i)+'d. Porcentaje de nivel ecológico de los
turistas dependiendo de su nacionalidad')

plt.show()
i+=1

    # 7. traveltimes_cat
data=q2016.traveltimes.describe()
mean = round(data[1], 1) # Store the mean
std = round(data[2], 1) # Store the standard deviation
n = round(data[0], 1)
Q1 = data[4]
Q3 = data[6]

max_value = Q3 + 1.5 * (Q3 - Q1)
min_value = Q1 - 1.5 * (Q3 - Q1)

# We remove the outliers
df = q2016[q2016["traveltimes"]<=max_value]
df = df[df["traveltimes"]>=min_value]

data=df.traveltimes.describe()
mean = round(data[1], 1) # Store the mean
std = round(data[2], 1) # Store the standard deviation
n = round(data[0], 1)
Q1 = data[4]
Q3 = data[6]

    # Descriptive comparison
my_ct = pd.crosstab(df.eco_tourist, df.traveltimes_cat, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(df.eco_tourist, df.traveltimes_cat)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired")
```

```
plt.ylim(0, 100)
plt.xticks(rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Veces que viajaron los turistas en 2015")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico de\nlos
turistas dependiendo de la cantidad de viajes')

plt.show()
i+=1

    # 8. long_vac_cat
res = q2016.long_vac_cat.describe()
n = res[0]
    # Descriptive comparison
my_ct = pd.crosstab(q2016.eco_tourist, q2016.long_vac_cat, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(q2016.eco_tourist, q2016.long_vac_cat)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (7,5))
plt.ylim(0, 100)
plt.xticks(rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Veces que viajaron los turistas en 2015")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico de los
turistas\ndependiendo de la cantidad de viajes de larga duración')

plt.show()
i+=1

    # 9. med_vac_cat
    # Descriptive comparison
```

```
my_ct = pd.crosstab(q2016.eco_tourist, q2016.med_vac_cat, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(q2016.eco_tourist, q2016.med_vac_cat)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (7,5))
plt.ylim(0, 100)
plt.xticks(rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Veces que viajaron los turistas en 2015")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico de los
turistas\ndependiendo de la cantidad de viajes de media duración')

plt.show()
i+=1

    # 10. short_vac_cat
    # Descriptive comparison
my_ct = pd.crosstab(q2016.eco_tourist, q2016.short_vac_cat, normalize
= "columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(q2016.eco_tourist, q2016.short_vac_cat)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (7,5))
plt.ylim(0, 100)
plt.xticks(rotation=0)
```

```
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Veces que viajaron los turistas en 2015")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico de los
turistas\ndependiendo de la cantidad de viajes de corta duración')

plt.show()
i+=1

    # 11. long_accom
df = q2016[q2016["long_accom"]!="No long staying"]
df = df[df["long_accom"]!="Don't know"]
res = df.long_accom.describe()
n = res[0]

    # Descriptive comparison
my_ct = pd.crosstab(df.eco_tourist, df.long_accom, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(df.eco_tourist, df.long_accom)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (10,5))
plt.ylim(0, 100)
plt.xticks(np.arange(0, 7, step = 1), ["Camping", "Otros", "Alquiler",
"Alojamiento""\n""comercial", "Con amigos""\n""o familia",
"Segunda""\n""residencia", "All"], rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Tipo de alojamiento")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico de los
turistas\ndependiendo del tipo de alojamiento en viajes de larga
duración')

plt.show()
i+=1
```

```
    # 12. med_accom
df = q2016[q2016["med_accom"]!="No medium staying"]
df = df[df["med_accom"]!="Don't know"]
res = df.long_accom.describe()
n = res[0]

    # Descriptive comparison
my_ct = pd.crosstab(df.eco_tourist, df.med_accom, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(df.eco_tourist, df.med_accom)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (10,5))
plt.ylim(0, 100)
plt.xticks(np.arange(0, 8, step = 1), ["Camping", "Otros", "Alquiler",
"Alojamiento""\n""comercial", "Con amigos""\n""o familia",
"Segunda""\n""residencia", "All"], rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Tipo de alojamiento")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico de los
turistas\ndependiendo del tipo de alojamiento en viajes de media
duración')

plt.show()
i+=1

    # 13. short_accom
df = q2016[q2016["short_accom"]!="No short staying"]
df = df[df["short_accom"]!="Don't know"]
res = df.short_accom.describe()
n = res[0]

    # Descriptive comparison
my_ct = pd.crosstab(df.eco_tourist, df.short_accom, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
```

```
ct = pd.crosstab(df.eco_tourist, df.short_accom)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (10,5))
plt.ylim(0, 100)
plt.xticks(np.arange(0, 7, step = 1), ["Camping", "Otros", "Alquiler",
"Alojamiento""\n""comercial", "Con amigos""\n""o familia",
"Segunda""\n""residencia", "All"], rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Tipo de alojamiento")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico de los
turistas\ndependiendo del tipo de alojamiento en viajes de corta
duración')

plt.show()
i+=1

    # 14. reason_vac_prin
df = q2016[q2016["reason_vac_prin"]!="Don't know"]
res = df.reason_vac_prin.describe()
n = res[0]

    # Descriptive comparison
my_ct = pd.crosstab(df.eco_tourist, df.reason_vac_prin, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(df.eco_tourist, df.reason_vac_prin)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired",
figsize = (10,5))
plt.ylim(0, 100)
```

```
plt.xticks(np.arange(0, 8, step = 1), ["Actividades\nconcretas",
"Cultura", "Naturaleza", "Otras",
          "Sol/Playa", "Viajes\na ciudades", "Visitar\nfamilia",
"All"], rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Razones principales")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico\nde los
turistas dependiendo de la razón principal por la que se viaja')

plt.show()
i+=1

    # 15. vac_company_1
df = q2016[q2016["vac_company_1"]!="Don't know"]
res = df.vac_company_1.describe()
n = res[0]

    # Descriptive comparison
my_ct = pd.crosstab(df.eco_tourist, df.vac_company_1, normalize =
"columns", margins = True)*100
my_ct = round(my_ct, 1)
print(my_ct)

    # Statistical comparison
ct = pd.crosstab(df.eco_tourist, df.vac_company_1)

        # Perform the Chi2 test (p value < 0.05) of the single
crosstab
res = stats.chi2_contingency(ct)
print(res)

Chi2 = res[0]
P_val = res[1]

    # Graphical comparison.  We need to transpose the crosstab and
then plot
my_ct.T.plot(kind = "bar", edgecolor = "Black", colormap = "Paired")
plt.ylim(0, 100)
plt.xticks(np.arange(0, 7, step = 1), ["Amigos", "Familia",
"Grupo\norganizado", "Otra",
          "Pareja", "Solo", "All"], rotation=0)
props = dict(boxstyle = "round", facecolor = "white", lw = 0.5)
textstr =
"$\mathrm{n}:%.0f$\n$\mathrm{Chi2}:%.2f$\n$\mathrm{Pval}:%.3f$"%(n,
Chi2, P_val)
xmin, xmax, ymin, ymax = plt.axis()
plt.text(xmin + (xmax - xmin)*0.05, ymin + (ymax - ymin)*0.8, textstr,
bbox = props)
plt.legend(["No ecológico", "Ecológico"])
plt.xlabel("Tipo de compañía")
plt.title ('Figura '+str(i)+'. Porcentaje de nivel ecológico\nde los
turistas dependiendo del tipo de compañía')

plt.show()
```

```
i+=1
```

Anexo 6 – TFM_XGBoost.py

```
#!/usr/bin/env python
# coding: utf-8

# In[59]:


#load basiclibraries
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import multiprocessing
import timeit
from sklearn.preprocessing import scale
from sklearn.cluster import KMeans
from sklearn.metrics import confusion_matrix


# In[60]:


# Get working directory
os.getcwd()

# Change working directory
os.chdir("C:/Users/ncosn/Downloads")
os.getcwd()


# In[61]:


# Reads data from SPSS file and stores it in a dataframe called pf
pf = pd.read_spss("quest_2016_r.sav")
print(pf)
pf.shape


# In[62]:


pf.head()


# In[63]:


import seaborn as sns
plt.figure(figsize=(20,8))
sns.countplot(pf['eco_tourist'])


# In[64]:


# We rename some columns and values
```

```
pf = pf.drop(pf.columns[2], axis = 1)
pf = pf.drop(pf.columns[3], axis = 1)
pf = pf.drop(pf.columns[4], axis = 1)
pf = pf.drop(pf.columns[5], axis = 1)
pf = pf.drop(pf.columns[33], axis = 1)
pf = pf.drop(pf.columns[10], axis = 1)
pf = pf.drop(pf.columns[10], axis = 1)
pf = pf.drop(pf.columns[10], axis = 1)
pf = pf.drop(pf.columns[11], axis = 1)
pf = pf.drop(pf.columns[11], axis = 1)
pf = pf.drop(pf.columns[11], axis = 1)
pf = pf.drop(pf.columns[19], axis = 1)
pf = pf.drop(pf.columns[19], axis = 1)
pf = pf.drop(pf.columns[19], axis = 1)

pf["eco_tourist"].replace({"Not very ecological" : 1}, inplace = True)
pf["eco_tourist"].replace({"Ecological" : 1}, inplace = True)
pf["eco_tourist"].replace({"Very ecological" : 1}, inplace = True)
pf["eco_tourist"].replace({"No ecological" : 0}, inplace = True)

print(pf.dtypes) # We see the data types
print(pf.isnull().sum()) # We check if there are missing values


# In[65]:


pf.head(40)


# In[66]:


pfc = pf.drop(['eco_aspect_1','eco_aspect_2','eco_aspect_3'], axis =1)
print(pfc.dtypes)


# In[67]:


# Save table to varible X
X = pfc

categorical_vars = set(X.columns[X.dtypes == object])
numerical_vars = set(X.columns) - categorical_vars
categorical_vars = list(categorical_vars)
numerical_vars = list(numerical_vars)

X[categorical_vars] = X[categorical_vars].astype(str)

print(categorical_vars)
print(numerical_vars)


# In[68]:


from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder

ohe = OneHotEncoder(sparse = False)
```

```
ohe_fit = ohe.fit(X[categorical_vars])
X_ohe = pd.DataFrame(ohe.fit_transform(X[categorical_vars]))
X_ohe.columns = pd.DataFrame(ohe_fit.get_feature_names())

X[categorical_vars].head()

X_ohe.head()


# In[69]:


X = pd.concat((X_ohe, X[numerical_vars].reset_index()), axis=1)

X.dtypes


# In[70]:


X = X.drop(columns ='index')


# In[71]:


# # # Split Into Training and Testing Sets

# Separate out the features and targets
features = X.drop(columns='eco_tourist')
targets = pd.DataFrame(X['eco_tourist'])

# Split into 80% training and 20% testing set
X_train, X_test, y_train, y_test = train_test_split(features, targets,
test_size = 0.2, random_state = 1)

print(X_train.shape)
print(X_test.shape)
print(y_train.shape)
print(y_test.shape)


# In[72]:


from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
print(X_train)


# In[73]:


# # # Models to Evaluate

# We will compare five different machine learning Classification
models:

# 1 - Logistic Regression
```

```
# 2 - K-Nearest Neighbors Classification
# 3 - Naive Bayes
# 4 - Random Forest Classification
# 5 - XGBoost


# In[83]:


def cross_val(X_train, y_train, model):
    # Applying k-Fold Cross Validation
    from sklearn.model_selection import cross_val_score
    accuracies = cross_val_score(estimator = model, X = X_train, y =
y_train, cv = 5)
    return accuracies.mean()


# In[84]:


# Takes in a model, trains the model, and evaluates the model on the
test set
def fit_and_evaluate(model):

    # Train the model
    model.fit(X_train,np.asarray(y_train["eco_tourist"]))

    # Make predictions and evalute
    model_pred = model.predict(X_test)
    model_cross =
cross_val(X_train,np.asarray(y_train["eco_tourist"]), model)

    # Return the performance metric
    return model_cross


# In[85]:


# # Logistic Regression
from sklearn.linear_model import LogisticRegression
logr = LogisticRegression()
logr_cross = fit_and_evaluate(logr)

print('Logistic Regression Performance on the test set: Cross
Validation Score = %0.4f' % logr_cross)


# In[86]:


# # K-NN
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors = 5, metric = 'minkowski', p =
2)
knn_cross = fit_and_evaluate(knn)

print('KNN Performance on the test set: Cross Validation Score =
%0.4f' % knn_cross)
```

```
# In[87]:


from sklearn.naive_bayes import GaussianNB
naive = GaussianNB()
naive_cross = fit_and_evaluate(naive)

print('Naive Bayes Performance on the test set: Cross Validation Score
= %0.4f' % naive_cross)


# In[88]:


# # Random Forest Classification
from sklearn.ensemble import RandomForestClassifier
random = RandomForestClassifier(n_estimators = 10, criterion =
'entropy')
random_cross = fit_and_evaluate(random)

print('Random Forest Performance on the test set: Cross Validation
Score = %0.4f' % random_cross)


# In[89]:


# XGBClassifier
from xgboost import XGBClassifier
gb = XGBClassifier()
gb_cross = fit_and_evaluate(gb)
y_predxgb = gb.predict(X_test)
print('Gradient Boosting Classification Performance on the test set:
Cross Validation Score = %0.4f' % gb_cross)


# In[90]:


gb_cross = 0.6604
print('Gradient Boosting Classification Performance on the test set:
Cross Validation Score = %0.4f' % gb_cross)


# In[104]:


# Now, to better understand the results, I will show in a graph the
model that has the better Cross Validation Score

# Dataframe to hold the results
model_comparison = pd.DataFrame({'model': ['Logistic Regression', 'K-
NN',
                                           'Naive Bayes', 'Random
Forest',
                                            'Gradiente Boosting'],
                                 'score': [logr_cross, knn_cross,
naive_cross,
                                           random_cross, gb_cross]})
```

```
plt.barh(model_comparison['model'], model_comparison['score'],
edgecolor = "Black")
plt.xlabel('K-Fold Cross Validation')
plt.title ('Comparación de modelos')
plt.show()


# In[20]:


# Hyperparameter Tuning with Random Search and Cross Validation

# Here we will implement random search with cross validation to select
the optimal hyperparameters for the gradient boosting regressor.
# We first define a grid then peform an iterative process of: randomly
sample a set of hyperparameters from the grid, evaluate the
hyperparameters using 4-fold cross-validation,
# and then select the hyperparameters with the best performance.

# Loss function to be optimized
loss = ['ls', 'lad', 'huber']

# Number of trees used in the boosting process
n_estimators = [100, 500, 900, 1100, 1500]

# Maximum depth of each tree
max_depth = [2, 3, 5, 10, 15]

# Minimum number of samples per leaf
min_samples_leaf = [1, 2, 4, 6, 8]

# Minimum number of samples to split a node
min_samples_split = [2, 4, 6, 10]

# Maximum number of features to consider for making splits
max_features = ['auto', 'sqrt', 'log2', None]

# Define the grid of hyperparameters to search
hyperparameter_grid = {'loss': loss,
                       'n_estimators': n_estimators,
                       'max_depth': max_depth,
                       'min_samples_leaf': min_samples_leaf,
                       'min_samples_split': min_samples_split,
                       'max_features': max_features}


# In[21]:


# After training, we can compare all the different hyperparameter
combinations and find the best performing one.

# Create the model to use for hyperparameter tuning
model = XGBClassifier(random_state = 42)

# Set up the random search with 4-fold cross validation
from sklearn.model_selection import RandomizedSearchCV, GridSearchCV
random_cv = RandomizedSearchCV(estimator=model,

param_distributions=hyperparameter_grid,
                               cv=4, n_iter=25,
```

```
                                    scoring = 'neg_mean_absolute_error',
                                    n_jobs = -1, verbose = 3,
                                    return_train_score = True,
                                    random_state=42)


# In[22]:


# Fit on the training data
random_cv.fit(X_train, y_train)


# In[23]:


# Get all of the cv results and sort by the test performance
random_results =
pd.DataFrame(random_cv.cv_results_).sort_values('mean_test_score',
ascending = False)

random_results.head(10)


# In[24]:


random_cv.best_estimator_


# In[25]:


# I will focus on a single one, the number of trees in the forest
(n_estimators).
# By varying only one hyperparameter, we can directly observe how it
affects performance.
# In the case of the number of trees, we would expect to see a
significant affect on the amount of under vs overfitting.

# Here we will use grid search with a grid that only has the
n_estimators hyperparameter.
# We will evaluate a range of trees then plot the training and testing
performance to get an idea of what increasing the number of trees does
for our model.
# We will fix the other hyperparameters at the best values returned
from random search to isolate the number of trees effect.


# In[26]:


trees_grid = {'n_estimators': [100, 150, 200, 250, 300, 350, 400, 450,
500, 550, 600, 650, 700, 750, 800]}

model = XGBClassifier(loss = 'ls', max_depth = 2,
                                    min_samples_leaf = 2,
                                    min_samples_split = 6,
                                    max_features = 'sqrt',
                                    random_state = 42)
```

```
# Grid Search Object using the trees range and the random forest model
grid_search = GridSearchCV(estimator = model, param_grid=trees_grid,
cv = 4,
                           scoring = 'neg_mean_absolute_error',
verbose = 2,
                           n_jobs = -1, return_train_score = True)


# In[27]:


# Fit the grid search
grid_search.fit(X_train, y_train)


# In[30]:


# Get the results into a dataframe
results = pd.DataFrame(grid_search.cv_results_)

# Plot the training and testing error vs number of trees
figsize=(10, 8)
plt.style.use('fivethirtyeight')
plt.plot(results['param_n_estimators'], -1 *
results['mean_test_score'], label = 'Error en test')
plt.plot(results['param_n_estimators'], -1 *
results['mean_train_score'], label = 'Error en entrenamiento')
plt.xlabel('N° Árboles'); plt.ylabel('Mean Abosolute Error');
plt.legend();
plt.title('Rendimiento vs N° Árboles');


# In[31]:


results.sort_values('mean_test_score', ascending = False).head(5)


# In[32]:


# # # Evaluate Final Model on the Test Set

# We will use the best model from hyperparameter tuning to make
predictions on the testing set.

# For comparison, we can also look at the performance of the default
model. The code below creates the final model, trains it (with
timing), and evaluates on the test set.

# Default model
default_model = XGBClassifier(random_state = 42)

# Select the best model
final_model = grid_search.best_estimator_

final_model


# In[36]:
```

```
def_cross = fit_and_evaluate(default_model)
y_pred_def = default_model.predict(X_test)
print('Gradient Boosting Classification Performance on the test set:
Cross Validation Score = %0.4f' % def_cross)


# In[37]:


best_cross = fit_and_evaluate(final_model)
y_pred_best = final_model.predict(X_test)
print('Gradient Boosting Classification Performance on the test set:
Cross Validation Score = %0.4f' % best_cross)


# In[56]:


cmxgb = confusion_matrix(y_test, y_predxgb)
print(cmxgb)


# In[57]:


pd.options.display.float_format = '{:.3f}'.format
# Extract the feature importances into a dataframe
feature_results = pd.DataFrame({'feature': list(features.columns),
                                'importance':
gb.feature_importances_})

# Show the top 10 most important
feature_results = feature_results.sort_values('importance', ascending
= False).reset_index(drop=True)

feature_results.head(10)


# In[58]:


# XGBClassifier for multi-class
from xgboost import XGBClassifier
gb = XGBClassifier(objective='multi:softmax',num_class = 4)
gb_cross = fit_and_evaluate(gb)
y_predxgb = gb.predict(X_test)
print('Gradient Boosting Classification Performance on the test set:
Cross Validation Score = %0.4f' % gb_cross)
```