Facultad de Estudios Superiores

Acatlán

Estudio Alimenticio y de Salud

De la Cruz Cruz Santiago
Méndez Cárdenas María Fernanda
Meza Molina Marlenne Ivonne
Moranchel González José Manuel
Ramírez Jiménez Mariana

.

Análisis Multivariado

Mayo 2018

ANÁLISIS MULTIVARIADO

El análisis multivariante ha evolucionado al convertirse en el soporte matemático (estadístico) de las técnicas de aprendizaje máquina (machine learning) en contraste con su otra función de componente aislado y extensión a varias dimensiones de la estadística descriptiva e inferencial.

Tabla de contenidos

1. Marco Teórico

1.1 Introducción

“Aquellos que piensan que no tienen tiempo para una alimentación saludable tarde o temprano encontrarán tiempo para la enfermedad".–Edward Stanley.

Es una realidad que en México la educación de la salud no existe, son pocas las personas que toman los cuidados necesarios para llevar una vida sana y plena en base a su alimentación, y eso solo contando al 56.4 % que goza de una economía “estable". Es por eso que el presente proyecto pretende analizar los hábitos alimenticios que llevan personas cercanas a nosotros y a nuestros conocidos para que en base a esa información podamos crear perfiles alimenticios y llegar a conclusiones sobre sus hábitos alimenticios con la finalidad de predecir su estado de salud futuro y con ello evitar posibles enfermedades.

En la actualidad estamos expuestos a una cantidad desmesurada de alimentos chatarra o de poco valor nutricional, aunado a esto, las generaciones futuras tienen cada vez menos la costumbre de pensar en su régimen alimenticio debido en gran medida al bombardeo de publicidad por parte de las compañías de comida rápida que se muestran como una solución factible para un alimento rápido y de buen sabor. No por nada desde el año 2000, la diabetes mellitus en México es la primera causa de muerte entre las mujeres y la segunda entre los hombres. En 2010, esta enfermedad causó cerca de 83 000 muertes en el país. Y a pesar de los escasos intentos del gobierno que ha gastado en publicidad para concientizar al pueblo mexicano desde el año 2005 las muertes por diabetes solo han ido en aumento de forma exponencial. José Narro Robles, titular de la Secretaría de Salud, subraya que las defunciones por diabetes pasaron de 98 mil 500 en 2015 a 105 mil en 2016, lo que representa un aumento de siete mil muertes.

Mientras que el Sistema Nacional de Vigilancia Epidémica destaca en 2017 los casos de diabetes en el país pasaron de 307 mil 247 a 335 mil 134, lo cual solo ha ido en aumento comparándolo con los primeros meses del 2018.

1.2 Objetivos

- Por medio del análisis que realizamos buscamos encontrar la relación entre los hábitos alimenticios de la población mexicana y su salud, específicamente hablando, de la obesidad.
- La principal meta que nos propusimos con este proyecto es la de predecir la salud futura de una persona en base a su alimentación presente.

1.3 Justificación

Actualmente los regímenes alimenticios de las personas carecen de valor nutricional por lo que son propensas a ver en deterioro su buen estado de salud, y debido a la poca importancia que le toman debido que en el presenten pueden llegar a gozar de buena salud, no toman en cuenta lo importante que es cuidarse para mantener ese estado y poder vivir así por siempre.

1.4 Resumen Ejecutivo

1.4.1 Descripción del Proyecto

El Proyecto consiste en recopilar datos en base a una encuesta que tiene preguntas del estilo de un estudio epidemiologico para poder darnos una idea de la alimentación que llevan los mexicanos asi como sus gustos y la frecuencia con la que consumen ciertos alimentos.

Los datos serán usados para predecir aspectos nutricionales y serán usados para entrenar un modelo que nos permitirá predecir los por venires de la salud de las personas.

1.4.2 Diagrama del Proyecto

1.4.3 Bitácora de Actividades

ANÁLISIS DE VIABILIDAD

Una vez que optamos por el tema de la alimentación fue momento de ponerse a pensar que tan factible seria el recolectar los datos que necesitabamos ¿De dondé los obtendriamos? ¿Qué métodos utilizariamos para recolectarlos? ¿Cómo los iriamos registrando? Y una vez obtenidos los datos ¿Cómo los utilizaríamos y que herramientas usaríamos para su explotación?

PLANIFICACIÓN DETALLADA

Elaboramos un plan que consistía en dividirnos el numero de personas que había que entrevistar, cada quien se encargaría de recolectar 100 registros y así obtener 500 personas. Una vez obtenidos los datos los meteríamos a los modelos de Machine Learning trabajados en clase, cada quien probaría con 2 modelos.

EJECUCIÓN

Creamos las encuestas en Google para ahorrarnos tiempo y poder compartirlo a nuestros contactos, también aplicamos encuestas personales a familiares, amigos y personas que nos encontramos en el metro y en el centro.

SEGUIMINETO Y CONTROL

Una vez que íbamos obteniendo los datos los registrábamos en un excel y a la par limpiábamos la base quitando outlayers, datos ilógicos, cambiando formatos, imputando missings y modificando campos como comida favorita y restaurante favorito para que pudieran ser categorías menores y poder jugar con los datos.

CIERRE

Una vez que obtuvimos la información empezamos un Análisis descriptivo de los datos para darnos una idea de como manejarlos y después empezamos a aplicar los modelos predictivos.

1.5 Hipótesis

Tras un análisis descriptivo aplicado a los datos de las personas encuestada se notó los hábitos alimenticios tienden a variar bastante entre edades de 16 y 24 con edades mas avanzadas Por lo que se plantea que el grupo joven es el que consume mas variado debido a que las enfermedades a esa edad son poco comunes de desarrollarse. Por lo que suponemos que el comportamiento de la población de México en general es similar y por tanto el plan de ataque ira enfocado a ese sector principalmente.

2. Etapas de una predicción

2.1 Selección y limpieza de datos

Puntos principales

1. Los métodos de recolección y análisis de datos deben elegirse en función de las preguntas clave de evaluación y los recursos disponibles de la evaluación.
2. Las evaluaciones de impacto deben aprovechar al máximo los datos existentes y subsanar las carencias con nuevos datos.
3. Los métodos de recolección y análisis de datos deben elegirse de forma que se complementen entre sí los puntos fuertes y débiles. Planificación y Recolección de datos

Antes de tomar decisiones sobre qué datos recopilar y cómo analizarlos, debe decidirse la finalidad de la evaluación (es decir, los usuarios y usos previstos) y las preguntas clave de evaluación Una vez que está clara la finalidad de la evaluación, debe acordarse un pequeño número de preguntas clave de evaluación de alto nivel Aprovechar al máximo los datos existentes La planificación de la recolección de datos debe comenzar por revisar en qué medida pueden utilizarse los datos existentes. En términos de indicadores, la evaluación debe procurar basarse en distintos tipos de indicadores (insumos, productos, resultados, impactos) para reflejar los principales resultados en la teoría del cambio del programa. Las evaluaciones de impacto deben utilizar idealmente los indicadores que se seleccionaron para supervisar el desempeño durante el periodo de aplicación del análisis, es decir, los indicadores clave del desempeño. Es especialmente importante comprobar si existen datos de referencia para los indicadores seleccionados, así como para las características sociales y demográficas y otras características pertinentes de la población objeto de estudio. Cuando el diseño de evaluación implica comparar cambios a lo largo del tiempo en distintos grupos, los datos de referencia pueden utilizarse para determinar la equivalencia de los grupos antes de que empiece el programa Es importante aprovechar al máximo los datos existentes en aras de la eficiencia, los datos deben tener una calidad suficiente para no comprometer la validez de las constataciones de la evaluación

2.2 Análisis de Datos

Las técnicas de análisis de datos tienen en el machine learning un sólido apoyo para la generación de conocimiento en la organización. El aprendizaje automático explota la estadística y muchas otras áreas de las matemáticas. Su ventaja es la velocidad.

Los métodos de aprendizaje automático son muy superiores en el análisis que se practica a datos procedentes de múltiples fuentes. La información transaccional, la que llega de los medios de comunicación social o la que tiene su origen en sistemas como el CRM puede desbordar la capacidad de las técnicas de análisis de datos tradicionales.

Por el contrario, el machine learning de alto rendimiento puede analizar todo un conjunto Big Data, en vez de obligar a los usuarios de negocio a conformarse con una muestra representativa que, al fin y al cabo, se queda en eso, una muestra.

Esta escalabilidad no sólo permite que las soluciones predictivas basadas en sofisticados algoritmos sean más exactas, sino que también impulsa la importancia de la velocidad del software. De esta forma, ya es posible interpretar en tiempo real los miles de millones de filas y columnas que hay que investigar, a la vez que no se detiene el análisis del flujo de datos que va llegando.

3. Herramientas utilizadas

La principal herramienta utilizada para el proyecto es el programa traductor Jupyter Notebook, en el cual vamos a correr nuestro código en lenguaje Python.

3.1 NOTEBOOK DOCUMENT

Los documentos del cuaderno (o çuadernos", en minúsculas) son documentos producidos por la aplicación Jupyter Notebook , que contiene tanto código de computadora (por ejemplo, python) como elementos de texto enriquecido (párrafos, ecuaciones, figuras, enlaces, etc.). Los documentos del cuaderno son documentos legibles por humanos que contienen la descripción del análisis y los resultados (figuras, tablas, etc.) así como documentos ejecutables que se pueden ejecutar para realizar análisis de datos.

3.2 JUPYTER NOTEBOOK APP

La aplicación Jupyter Notebook es una aplicación cliente-servidor que permite editar y ejecutar documentos portátiles a través de un navegador web. La aplicación Jupyter Notebook se puede ejecutar en un escritorio local que no requiere acceso a Internet (como se describe en este documento) o puede instalarse en un servidor remoto y acceder a través de Internet. Además de mostrar / editar / ejecutar documentos de cuaderno, la aplicación Jupyter Notebook tiene un "Tablero de instrumentos"(Panel de instrumentos portátil), un "panel de control"que muestra los archivos locales y permite abrir documentos de cuaderno o cerrar sus núcleos.

3.3 KERNEL

Un núcleo de portátil es un "motor computacional"que ejecuta el código contenido en un documento de Notebook . El núcleo IPython (IPython es un shell interactivo que añade funcionalidades extra

al modo interactivo incluido con Python, como resaltado de líneas y errores mediante colores, una sintaxis adicional para el shell, autocompletado mediante tabulador de variables, módulos y atributos; entre otras funcionalidades. Es un componente del paquete SciPy), al que se hace referencia en esta guía, ejecuta el código python.

Cuando abre un documento de Notebook, el kernel asociado se inicia automáticamente. Cuando se ejecuta el bloc de notas (celda por celda o con el menú Celda ->Ejecutar todo), el kernel realiza el cálculo y produce los resultados. Dependiendo del tipo de cálculos, el kernel puede consumir CPU y RAM significativas. Tenga en cuenta que la memoria RAM no se libera hasta que el kernel se apaga.

3.4 NOTEBOOK DASHBOARD

El tablero de instrumentos del portátil es el componente que se muestra primero cuando inicia la aplicación Jupyter Notebook. El tablero de instrumentos portátil se utiliza principalmente para abrir documentos portátiles, y para gestionar las ejecutan núcleos (visualizar y apagado). El panel de instrumentos del Notebook tiene otras características similares a un administrador de archivos, a saber, navegación de carpetas y cambio de nombre / eliminación de archivos.

4. Librerias

4.1 Librería Numpy

El desarrollo y la principal finalidad del módulo Numpy es la creación y modificación de arrays multidimensionales. Para este fin utilizaremos las clase ndarray del ingles N-dimensional array o usando su alias simplemente array (no confundir con la clase array.array que ofrece menos funcionalidad). En Python cada clase puede tener atributos que se pueden llamar con el método visto anteriormente o simplemente escribiendo a continuación de la clase un punto y el atributo. En la mayoría de los IDEs al cargar la clase y escribir el punto aparecen todos los atributos disponibles en orden alfabético por lo que en caso de dudar siempre podemos utilizar este método para escribir el comando. En el caso de ndarray los principales atributos son los siguientes:

1. ndarray.ndim . Proporciona el número de dimensiones de nuestro array. El array identidad es un array cuadrado con una diagonal principal unitaria.
2. ndarray.shape . Devuelve la dimensión del array, es decir, una tupla de enteros indicando el tamaño del array en cada dimensión. Para una matriz de n filas y m columnas obtendremos (n,m).
3. ndarray.size .Es el número total de elementos del array.
4. ndarray.dtype . Es un objeto que describe el tipo de elementos del array.
5. ndarray.itemsize .Devuelve el tamaño del array en bytes.
6. ndarray.data . El buffer contiene los elementos actuales del array.

Recuperado de: `https://docs.scipy.org/doc/`

4.2 Statsmodels

Statsmodels es un modulo de python que prove clases y funciones para la estimación de diferentes modelos estadisticos, como también para pruebas estadísticas y exploración de datos estadisticos. Una extensa lista de resultados estadisticos disponibles para cada estimador. Los resultados son

puestos a prueba contra paqueterias existentes para asegurar de que están bien.
Recuperado de : `https://www.statsmodels.org/stable/index.html`

4.3 Math

Este modulo siempre esta disponible, el cual proporciona acceso a las funciones matematicas definidas por C standard. Estas funciones no pueden ser usadas con números complejos, patra lo cual hay que usar cmath si se requieren funciones para números complejos.
Recuperado de : `https://docs.python.org/3/library/math.html`

4.4 Pygal

Esta paqueteria proporciona una forma mas visual de representar los datos a treves de graficas de barras, de caja, de pie, histogramas, pirámides entre otros.
Recuperado de: `http://pygal.org/en/stable/documentation/index.html`

4.5 Matplotlib

Matplotlib es una libreria par ahacer graficos en 2d de colecciones de python. Esta diseñado para crear simples graficos con poco codigos.
Recuperado de: `https://matplotlib.org/contents.html#`

4.6 Pandas

Entre otras funciones, le da a Python la capacidad de trabajar con datos similares a hojas de cálculo (tablas) para cargar, manipular, alinear y fusionar datos rápidamente, proporciona una gran variedad de métodos para modificar y operar en esta tabla; en particular, permite consultas y combinaciones de tablas similares a SQL. Pandas tiene dos tipos de datos principales: las Series y los DataFrame, un DataFrame también se puede considerar como un diccionario o colección de Series.

Pandas es:

1. Manejo sencillo de los datos faltantes (representados como NaN) tanto datos flotantes como no flotantes. Mutabilidad de tamaño: las columnas se pueden insertar y eliminar de DataFrame y objetos dimensionales superiores
2. Alineación de datos automática y explícita : los objetos se pueden alinear explícitamente con un conjunto de etiquetas, o el usuario simplemente puede ignorar las etiquetas y dejar que Series , DataFrame , etc. alineen automáticamente los datos en los cálculos.
3. Grupo potente y flexible por funcionalidad para realizar operaciones de combinación de aplicación dividida en conjuntos de datos, para agregar y transformar datos Facilite la conversión de datos irregulares e indexados de forma diferente en otras estructuras de datos de Python y NumPy en objetos de DataFrame
4. Inteligente basado en etiquetas de rebanado , la indexación de fantasía , y de subconjuntos de grandes conjuntos de datos Conjuntos intuitivos de fusión y unión de datos Reestructuración y rotación flexibles de conjuntos de datos Etiquetado jerárquico de ejes (es posible tener etiquetas múltiples por marca)

5. Herramientas IO robustas para cargar datos desde archivos planos (CSV y delimitados), archivos de Excel, bases de datos y guardar / cargar datos desde el formato HDF5 ultrarrápido Funcionalidad específica de la serie de tiempo : generación de rango de fechas y conversión de frecuencia, estadísticas de ventanas en movimiento, regresiones lineales de ventanas móviles, cambio de fecha y retraso, etc.

Recuperado de: `https://pandas.pydata.org/pandas-docs/stable/`

4.7 Scipy

Es una colección de algoritmos matemáticos, funciones creadas en la extensión de Numpy. Proporciona al usuario comandos y clases para manipular y visualizar datos.
Recuperado de: `https://docs.scipy.org/doc/`

4.8 Sklearn

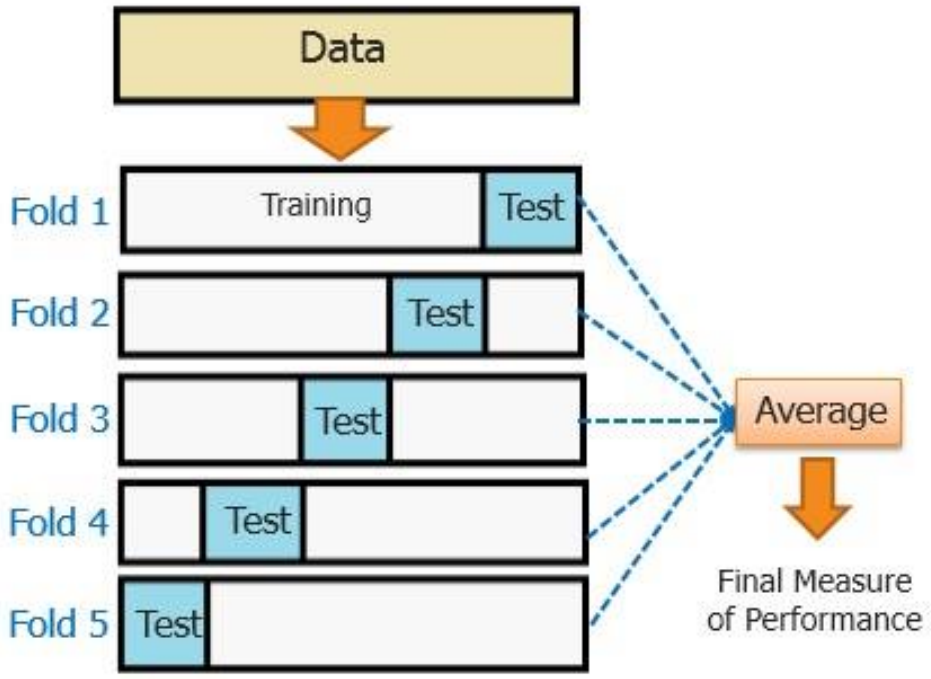
Scikit-learn fue iniciada en 2007 por David Cournapeau como un proyecto Google Summer of Codees, es un módulo de Python que incluye la implementación de una amplia gama de algoritmos de Machine Learning de última generación para problemas supervisados y no supervisados. Este paquete se enfoca en brindar aprendizaje automático a personas que no sean especialistas usando un lenguaje de alto nivel de propósito general. Se enfatiza la facilidad de uso, el rendimiento, la documentación y la consistencia API. Tiene dependencias mínimas y se distribuye bajo la licencia BSD simplificada, lo que fomenta su uso en entornos académicos y comerciales.

La podemos utilizar para clasificaciones, extraccion de características, regresión, agrupación, reducción de dimensión, selección de modelos (a través de la comparación, validación, hiperparametrización), o preprocesamiento.

Los algoritmos que usaremos son: Reducción de dimensiones:

1. PCA (Análisis de Componentes Principales). El cual se utiliza para descomponer un conjunto de datos multivariados en un conjunto de componentes principales, las cuales son ortogonales y explican la cantidad máxima de la varianza. Ayudan en los modelos de Máquina vector soporte y K-Means.
2. Clasificación: en estos encontramos tanto modelos del aprendizaje supervisado como no suoervisado.
3. Regresión: estos métodos destinados a la regresión en los que se espera que el valor objetivo sea una combinación lineal de las variables de entrada, es decir, si es el valor predicho, se espera que:
   Entre estos estudiaremos la regresión logística.
4. Agrupación: La agrupación de datos no etiquetados se puede realizar con el módulo sklearn.cluster. Cada algoritmo de agrupación viene en dos variantes: una clase, que implementa el método fit (ajuste) para aprender los Clústers de los datos de entrenamiento, y una función que, dada la información del entrenamiento, devuelve una matriz de etiquetas enteras correspondientes a los diferentes clústers. Para la clase, las etiquetas sobre los datos de entrenamiento se pueden encontrar en el atributo labels. Los algoritmos de agrupamiento que veremos son: clúster Jerárquico (average, centroid, ward), K-Means y Gaussianos mixtos.
5. Selección de modelo: para la selección y evaluación de los modelos existen varios métodos para obtener el mejor, tales como: 1. Cross Validation: Debemos especificar el porcentaje

con el que va a entrenar y con el que va a validar, para eso existe una función auxiliar: train test plit que nos ayudará a realizar la partición.



2. Hiperparametrización: Se revisan los parámetros con los que cuenta cada modelo y mediante simulaciones se verificará con que valores de los parámetros se ajusta mejor el modelo. En scikit-learn se proporcionan dos enfoques para los candidatos de búsqueda de muestreo: para los valores dados, se consideran GridSearchCV exhaustivamente todas las combinaciones de parámetros, mientras que se RandomizedSearchCV puede muestrear un número dado de candidatos de un espacio de parámetros con una distribución específica.

Recuperado de: `http://scikit-learn.org/stable/documentation.html`

4.9 Seaborn

Seaborn es una biblioteca de visualización de Python basada en matplotlib. Proporciona una interfaz de alto nivel para dibujar gráficos estadísticos atractivos. Su objetivo es visualizar datos complejos de forma sencilla y extraer conclusiones.

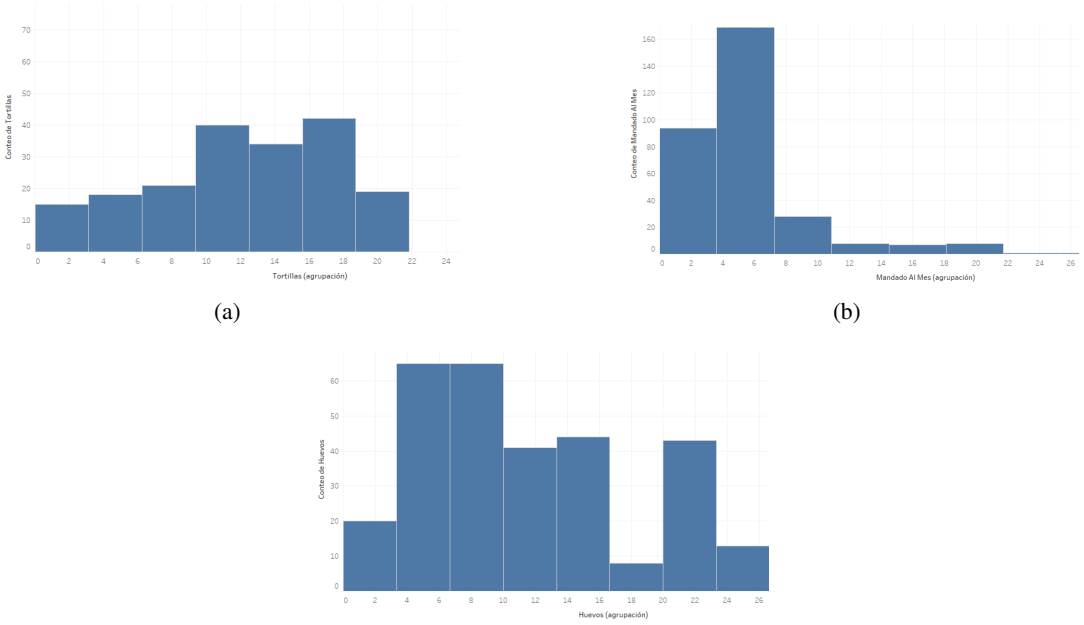
Seaborn pretende hacer que la visualización sea una parte central de la exploración y comprensión de los datos. Las funciones de trazado funcionan en marcos de datos y matrices que contienen un conjunto de datos completo y realizan internamente la agregación necesaria y el ajuste estadístico del modelo para producir gráficos informativos. Como dice Michael Waskom: “Si matplotlib ïntenta facilitar las cosas fáciles y las cosas difíciles posibles", seaborn intenta hacer también un conjunto bien definido de cosas difíciles”. Pero se debe considerar como un complemento de matplotlib, no como un reemplazo para él. El hecho de trabajar con DataFrames no funciona tan bien con Matplotlib, lo que lo hace algo ineficiente al hacer un análisis exploratorio con Pandas. Eso es exactamente lo que aborda Seaborn: las funciones de trazado funcionan en DataFrames y arrays que contienen un conjunto de datos completo.
Recuperado de: `https://seaborn.pydata.org/`

5. Análisis Explorativo

5.1 Análisis descriptivo de los datos

5.1.1 Análisis Univariante

Histograma



(a) (b)

(c)

Figura 5.1: *Ejemplos de histogramas de columnas*

Gráfica

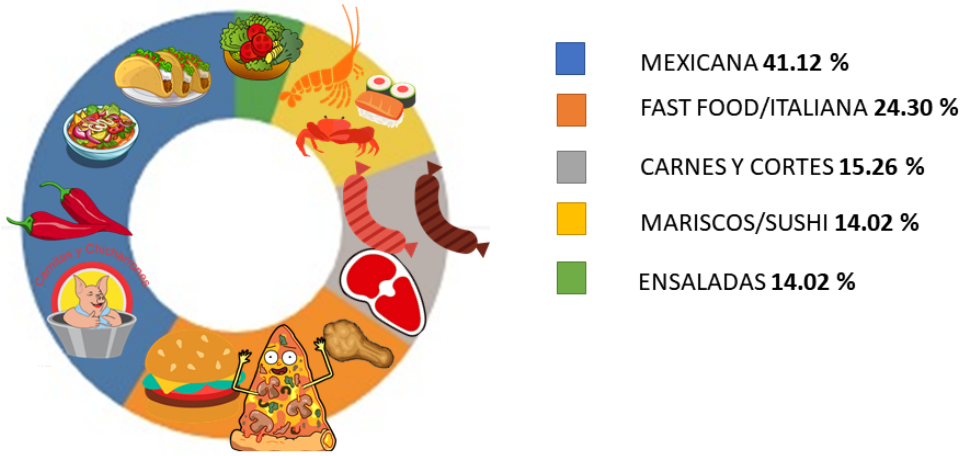


Figura 5.2: Agrupacion por comida

5.1.2 Análisis Bivariante

5.2 No Supervisada

Análisis Clúster

Clustering (análisis de conglomerados) se refiere a la tarea de agrupar un conjunto de objetos de tal forma que se obtengan grupos (clústeres) lo más similares entre sí. El objetivo del análisis clúster es partir un conjunto de datos en grupos.

1. Clustering jerárquico: Los métodos jerárquicos son la columna vertebral del análisis clúster. Parte de su popularidad recae en que no son afectados por el llamado “efecto de orden” aunado a que no se necesita intuir a priori cuántos clústeres están presentes en los datos.
   - Average. La distancia entre clústeres es la distancia promedio entre pares de observaciones. El método tiende a unir clústeres con varianzas pequeñas y sesga a producir clústeres con varianza igual. Otra característica es que es menos influenciado por valores atípicos que la mayoría de los métodos, además computacionalmente es más rápido que la mayoría de los métodos.
   - Centroid. La distancia entre los clústeres es la distancia euclídea cuadrada de cada uno de los centroides de los clústeres. En este método los datos extremos tampoco afectan demasiado, se trabaja directamente con los datos coordenados y si los grupos a fusionar son de tamaños muy distintos, el centroide del nuevo grupo estará muy cerca del centroide del grupo más grande.
   - Ward. El método de la mínima varianza de Ward une clústeres tales que en cada generación es minimizada la suma de cuadrados dentro del clúster sobre todas las particiones obtenidas al fusionar dos clústeres de cada generación previa. En este método se realiza un análisis de varianza (ANOVA) en cada fusión de la jerarquía y tiende a unir clústeres con un número pequeño de observaciones sesgando a producir clústeres esféricos con aproximadamente el mismo tamaño. Es muy sensible a datos atípicos.
     (Muller-Cyran,2016, Introduction to Machine Learning with Python).
2. Clustering de optimización: El clustering de optimización particiona un conjunto de datos al optimizar algún criterio específico. A diferencia de los métodos jerárquicos, el error escala linealmente con el número de observaciones por lo que estos métodos son útiles en grandes

conjuntos de datos. El número de particiones debe especificarse a priori.

- K-means Clustering. Es el más común de los algoritmos de clustering, consiste en los siguientes pasos:
  - Un conjunto de puntos conocido como "semillas" es seleccionado como las medias de los clústeres finales.
  - Cada observación es asignada a la semilla más cercana formando clústeres temporales. Las semillas son reemplazadas por las medias de los clústeres temporales y se repite el proceso hasta que ocurra un cambio no significativo en la posición de los centros.
  - Forma los clústeres finales asignando cada observación al centroide más cercano.

  Los primeros dos pasos son el algoritmo de búsqueda heurística. El tercer paso es la iteración extra que realiza las asignaciones finales a cada clúster.
  (Pedregosa, Fabian,2011, Scikit-learn: Machine Learning in Python).

3. Clustering Difuso: El clustering difuso se basa en criterios de densidad, son una poderosa alternativa a los métodos jerárquicos o de optimización ya que sobrepasan algunas de las debilidades de éstos, por ejemplo, pueden generar clústeres de cualquier forma y tamaño además de no ser afectados por valores extremos.
   - Modelos Gaussianos Mixtos Los modelos gaussianos mixtos son un tipo de modelo de densidad que comprenden cierto número de funciones componente usualmente gaussianas.
     (Muller-Cyran,2016, Introduction to Machine Learning with Python).

5.3 Supervisada

Aprendizaje máquina

Conocido en inglés como Machine Learning, es un método de análisis de datos que automatiza la construcción de modelos analíticos a través del uso de algoritmos que iterativamente aprenden de los datos. El aprendizaje máquina permite a las computadoras encontrar patrones careciendo de programación explicita que direccione la búsqueda.

1. Regresión logística La regresión logística nos permite modelar un evento dicotómico en forma probabilística donde representa la probabilidad de éxito, en otras palabras, la probabilidad de que ocurran casos que representen una característica de interés (tomar un producto, incumplir un préstamo, enfermar, etc.).
2. Árboles de decisión Un árbol empírico representa una segmentación de los datos creada aplicando una serie de reglas simples donde cada regla asigna una observación a un grupo basado en los valores de su vector de variables de entrada. Dichas reglas son aplicadas en cascada, en consecuencia, se formará una jerarquía de segmentos dentro de los segmentos. Tal jerarquía es llamada árbol y cada segmento se denomina nodo. El segmento inicial contendrá la totalidad de los datos y es conocido como nodo raíz. Los sucesores de un nodo formarán una rama, donde los nodos terminales son llamados hojas. El tipo de decisión involucrada dependerá expresamente del contexto.
   (Nisbet,Robert. (2009). Handbook of Statistical Analysis and Data Mining Applications)
3. Redes neuronales artificiales Las redes neuronales imitan a los sistemas naturales que aprenden a responder apropiadamente a los cambios del ambiente. Las entradas que ingresan a través de las dendritas son ponderadas por sinapsis adaptables antes de ser sumadas, si la suma es mayor que cierto umbral adaptable, la neurona envía una señal mediante su axón a otras neuronas. Las redes neuronales artificiales (ANN) fueron desarrolladas originalmente por investigadores que buscaban imitar la neurofisiología del cerebro humano. Al combinar

muchos elementos simples de cómputo (neuronas) en un sistema altamente interconectado intentaron reproducir fenómenos complejos como la inteligencia. En años recientes se han incorporado métodos estadísticos y análisis numérico a las redes. Si bien todavía se debate si las ANN son verdaderamente inteligentes, es un hecho que son un modelo matemático muy útil. Las ANN son una clase de regresión no lineal muy flexible. Al detectar relaciones complejas no lineales en los datos, las redes pueden ayudar a predecir en problemas del mundo real.

4. Análisis Discriminante El análisis discriminante (en lo sucesivo AD) es una técnica estadística que permite asignar o clasificar nuevos individuos dentro de grupos previamente definidos. AD intenta realizar la misma tarea que la regresión lineal múltiple al predecir una salida, sin embargo, la regresión está limitada a los casos donde la variable dependiente es tal que la combinación de sus predictoras, a través de la ecuación de regresión, producirá los valores numéricos estimados de la media mediante combinaciones ponderadas de los valores de las variables independientes. EL AD nos permitirá adentrarnos dentro de variables de interés no continuas como podrían ser: Intención de voto, estatus de empleo, intención de compra de un producto, si un cliente es sujeto de crédito o no, marca de preferencia, y en general, cualquier otra variable categórica que sea de utilidad para el investigador. El AD entra dentro de la minería de datos en la categoría de modelación supervisada, debido a que es necesaria una variable objetivo que nos permita clasificar. Su objetivo fundamental es producir una regla o esquema de clasificación tal que nos permita predecir la población a la que es más probable pertenecer una nueva observación.
5. Máquinas de soporte vectorial. Una máquina de soporte vectorial, es una técnica de modelación supervisada utilizada en problemas tanto de clasificación como de regresión. Consiste en construir un hiperplano (o un conjunto de hiperplanos) en una mayor dimensión, que incluso pudiese ser infinita, para separar patrones linealmente. Dado un vector de pesos w y un sesgo b, la separación entre el hiperplano propuesto y el punto en los datos más cercano es llamada el margen de separación y se denota por la letra . El objetivo de la máquina de soporte vectorial es encontrar el hiperplano particular tal que el margen de separación se maximice. Bajo esta condición, la superficie de decisión es llamada hiperplano óptimo.
6. K-Vecinos más cercanos La idea detrás de este método se basa en la estimación a partir de un número fijo k de observaciones lo más cercanas al punto estudiado. Para problemas de clasificación, la categoría elegida dependerá de la mayoría de voto de los k vecinos involucrados. La cercanía será computada basada en un criterio de distancia, en consecuencia, la estandarización de los datos será mandatoria.
7. Clasificador ingenuo de Bayes Es un clasificador basado en el Teorema de Bayes que nos permite conocer la probabilidad de pertenencia de una observación en el conjunto de entrenamiento dadas sus características asumiendo independencia entre las mismas. El clasificador modela la probabilidad de pertenencia a una clase C basado en las predictoras independientes.
8. Ensambles Basados en el principio: “Dos cabezas piensan mejor que una”, los ensambles de modelos combinan las predicciones de un conjunto de algoritmos con la intención de mejorar la generalización/robustez que tendría un estimador individual. Los ensambles se dividen en dos grupos fundamentalmente: Ensambles por promedio (Averaging) Ensambles por impulso (Boosting) Respectivamente, los primeros basan la estimación en el promedio de las estimaciones individuales, mientras que los segundos son construidos de forma secuencial buscando reducir el sesgo del estimador combinado, en resumen, se construyen varios modelos débiles para producir un ensamble poderoso.

Recuperado de: (Pedregosa, Fabian,2011, Scikit-learn: Machine Learning in Python).

6. Predicciones

metricas (model, Xt,Xv, yt,yv)	
ROC train:0.970	ROC test:0.785
ACC train:0.897	ACC test:0.739

metricas (model, Xt,Xv, yt,yv)	
ROC train:1.000	ROC test:0.984
ACC train:1.000	ACC test:0.935

metricas (model, Xt,Xv, yt,yv)	
ROC train:0.967	ROC test:0.798
ACC train:0.864	ACC test:0.696

MLPClassifier(activation='logistic', alpha=0.0004, batch_size='auto', beta_1=0.9, beta_2=0.999, early_stopping=False, epsilon=1e-08, hidden_layer_sizes=(30,20,25), learning_rate='adaptive' , learning_rate_init=0.001, max_iter=1000, momentum=0.9, nesterovs_momentum=True, power_t=0.5, random_state=None, shuffle=True, solver='adam', tol=0.0001, validation_fraction=0.1, verbose=False, warm start=False)

metricas (model, Xt,Xv, yt,yv)	
ROC train:0.697	ROC test:0.640
ACC train:0.667	ACC test:0.651

KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski', metric_params=None, n_jobs=1, n_neighbors=42, p=2,weights='uniform')

metricas (model, Xt,Xv, yt,yv)	
ROC train:0.764	ROC test:0.640
ACC train:0.676	ACC test:0.674

GaussianNB(priors=None)

metricas (model, Xt,Xv, yt,yv)	
ROC train:0.851	ROC test:0.709
ACC train:0.732	ACC test:0.652

6.1 Estrategias de Uso

Análisis Discriminante Lineal (ADL) es una generalización del discriminante lineal de Fisher, un método utilizado en estadística, reconocimiento de patrones y aprendizaje de máquinas para encontrar una combinación lineal de rasgos que caracterizan o separan dos o más clases de objetos o eventos. La combinación resultante puede ser utilizada como un clasificador lineal, o, más comúnmente, para la reducción de dimensiones antes de la posterior clasificación.

LDA está estrechamente relacionado con el análisis de varianza (ANOVA) y el análisis de regresión, el cual también intenta expresar una variable dependiente como la combinación lineal de otras características o medidas. Sin embargo, ANOVA usa variables independientes categóricas y una variable dependiente continua, mientras que el análisis discriminante tiene variables independientes continuas y una variable dependiente categórica (o sea, la etiqueta de clase). La regresión logística y la regresión probit son más parecidas a ADL que ANOVA, pues también explican una variable categórica por los valores de variables independientes continuas. Estos otros métodos son preferibles en aplicaciones donde no es razonable asumir que las variables independientes están normalmente distribuidas, lo cual es una suposición fundamental del método ADL.

LinearDiscriminantAnalysis(n_components=None, priors=None, shrinkage=None, solver='svd', sotre_covariance=False, tol=0.00001)

metricas (model, Xt,Xv, yt,yv)	
ROC train:0.973	ROC test:0.917
ACC train:0.930	ACC test:0.870

6.2 Resultados

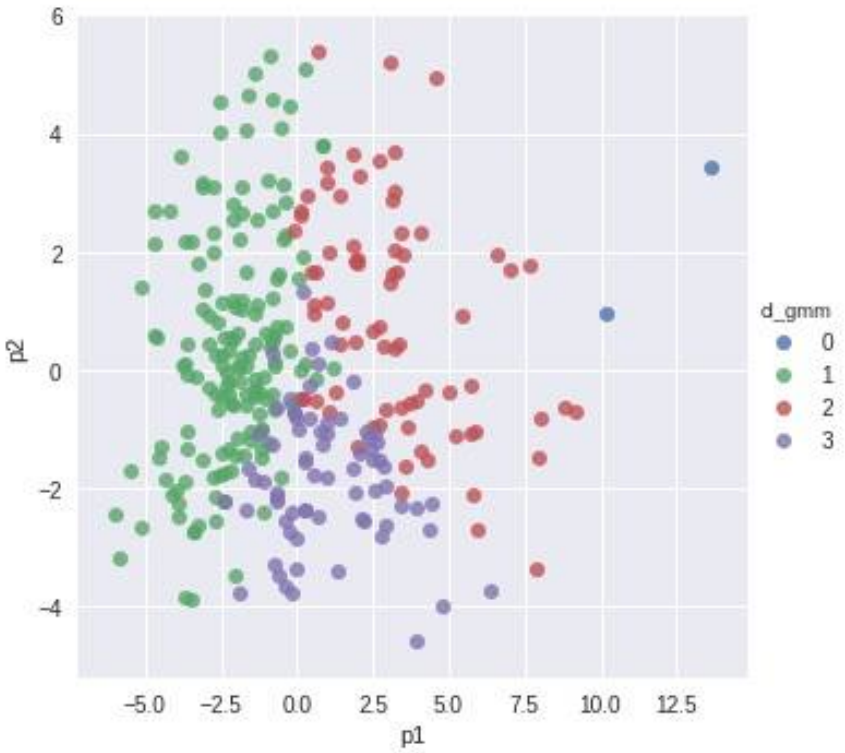


Figura 6.1: Representacion de los clusters

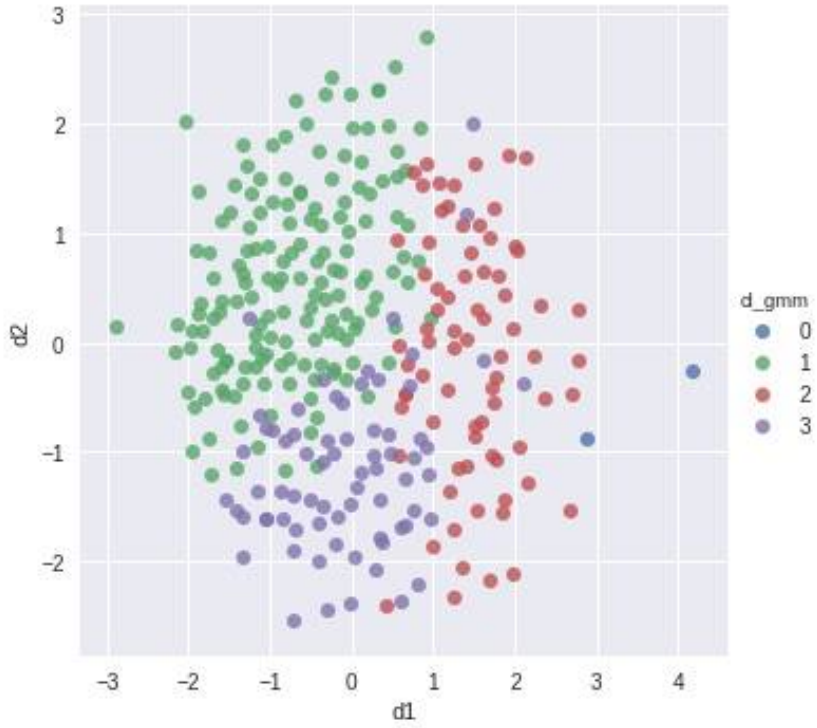


Figura 6.2: Representacion de los clusters

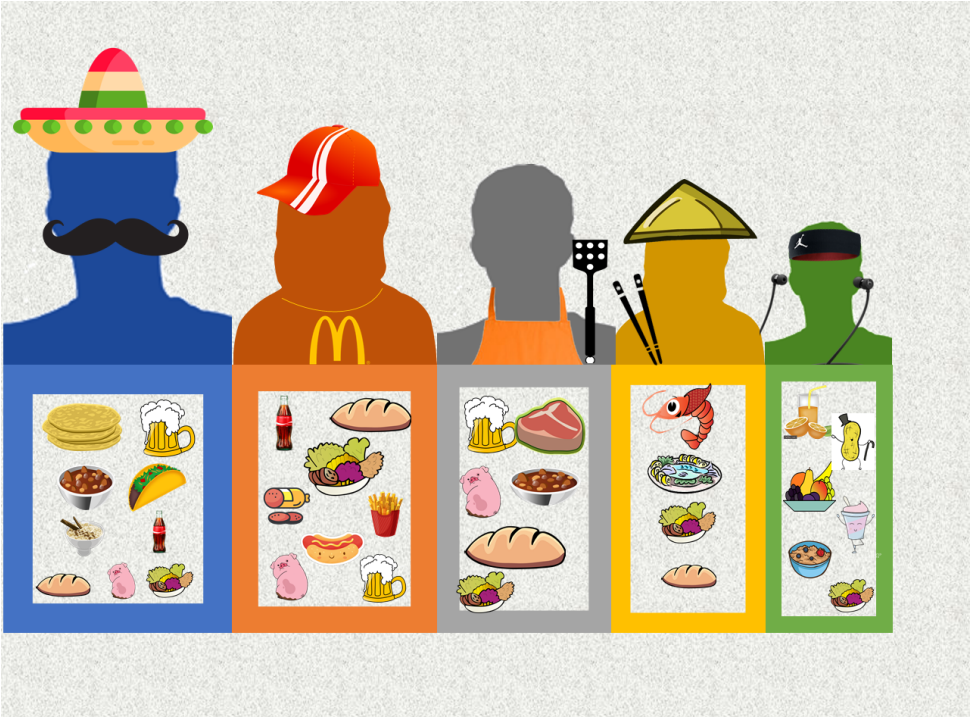


Figura 6.3: Representacion de los clusters

7. Conclusiones

7.1 Conclusión

La mala alimentación como ya todos sabemos es un problema de carácter mundial los hábitos en casa no son buenos, la vida tan acelerada que llevamos no ayuda de mucho y nos lleva a la peor alternativa que es consumir comida chatarra, no sabemos balancear nuestra comida y reemplazar algún alimento con esta comida ni nos nutre y a veces no nos satisface como debería. Se puede consumir claro pero hay que saber en qué momento recordemos que todo exceso es malo y más en la comida que nos puede llevar a padecer enfermedades mortales.

Observando una gran disparidad entre el consumo de los distintos grupos de alimentos lo cual nos condujo a la conclusion para minimizar la merma de los restaurantes se deberian lanzar promociones a nuestro parecer lo mas viable seria lanzar promociones especificas a usuarios que no consumen ciertos tipos de alimentos distinto a lo de su consumo habitual.

Bibliografía

[1] Chen, Daniel Y. (2017). *Pandas for Everyone: Python Data Analysis.* Boston: Addison-Wesley. Recuperado de: `https://www.safaribooksonline.com/library/view/pandas-for-everyone/9780134547046/ch01.xhtml#ch01`

[2] Pedregosa, Fabian; Varoquaux, Gaël; Gramfor, Alexandre; (2011). *Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research.* Recuperado de: `http://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf`

[3] Müller-Cyran, Andreas y Guido, Sarah. (2016). *Introduction to Machine Learning with Python. O'Reilly Media.* Recuperado de: `https://www.safaribooksonline.com/library/view/introduction-to-machine/9781449369880/ch02.html#supervised-learning`

[4] Nisbet,Robert; Elder,John;Elder,John;Miner,Gary. (2009). *Handbook of Statistical Analysis and Data Mining Applications*.

[5] Perez Lizaur, Ana Bertha;Palacios Gonzalez, Berenice; Castro Becerra,Ana Laura. (2014). *Sistema Mexicano de Alimentos Equivalentes* 4ª. Ed. México

[6] Murphy, Kevin P. (2012). *Machine Learning: A Probabilistic Perspective*, Ed. Mit Pr.

[7] Raschka, Sebastian. (2015). *Python Machine Learning*. Ed: Packt.

[8] Shai Ben David, Shai Shalev-Schwartz. (2014). *Undertanding Machine Learning: From Theory to Algorithms*. Ed: Cambridge Univ Pr

[9] `http://scikit-learn.org/stable/documentation.html`

[10] `https://docs.scipy.org/doc/`

[11] `https://www.statsmodels.org/stable/index.html`

[12] `https://docs.python.org/3/library/math.html`

[13] `http://pygal.org/en/stable/documentation/index.html`

[14] https://matplotlib.org/contents.html#

[15] https://pandas.pydata.org/pandas-docs/stable/

[16] https://seaborn.pydata.org/