Situación problema: Reporte final

Carlos Téllez Bermúdez A01637089

Héctor Daniel Franco González A01637419

Paulina Correa Delgado A01637235

Sebastián Denhi Vega Saint Martin A01637397

Matemáticas y ciencia de datos

Grupo 603

Tecnológico de Monterrey campus Guadalajara

Jueves 1 de diciembre del 2022

Introducción

La finalidad de este proyecto fue recolectar los datos de nuestras comidas durante varias semanas para almacenarlas dentro de una base de datos y después crear un modelo que (según los datos nutrimentales) pueda predecir las calorías de un alimento y con esto mejorar el control que tenemos a la hora de organizar nuestras comidas. Consideramos que es un proyecto importante en nuestra sociedad ya que en México esta es una problemática que afecta a todos por igual y cada vez es peor.

Según la organización Mayo Clinic, la obesidad es una enfermedad compleja que no solo consiste en tener una cantidad excesiva de grasa corporal sino que también presenta un grave problema médico que aumenta el riesgo de problemas de la salud (Mayo Clinic, 2011) que según el Centro Nacional para la Prevención de Enfermedades Crónicas y Promoción de la Salud incluye Presión arterial alta (hipertensión), Colesterol LDL alto, colesterol HDL bajo o niveles altos de triglicéridos (dislipidemia) y Diabetes entre muchos otros (CDC, 2022). Aunque muchos pensarían que la única razón por la cual una persona puede padecer de obesidad es por una dieta poco balanceada o falta de actividad física; sin embargo existen factores hereditarios, fisiológicos y del entorno que pueden llegar a ser causantes importantes. (Mayo Clinic, 2011)

En México, según la encuesta nacional de nutrición: el 42.6% de los hombres y el 35.5% de las mujeres padecen de obesidad (Gobierno de México,2015). Esto nos habla de la precaria situación en la que se encuentra México y por esto la importancia de este proyecto. El gobierno mexicano ya ha implementado varias iniciativas en contra de esta situación como la regulación de alimentos altamente calóricos en escuelas, la regulación de publicidad en estos alimentos y los famosos sellos nutrimentales que se encuentran en todos los productos alimenticios en México (Gobierno de México, s.f). Según los datos, si no se crea un cambio considerable se pronostica que en México en 2030 1 de cada 3 adultos tendrá obesidad en México (El Sol de México, 2022).

Para almacenar los datos utilizamos excel y para editarlos y limpiarlos utilizamos python en google collaboratory. Para recolectar los datos de informaciones nutrimentales hicimos una investigación de cada comida en las páginas de Myfitnesspal(MyFitnessPal | MyFitnessPal., s.f) y MyNetDiary(MyNetDiary, s.f) donde encontramos información sobre las calorías, carbohidratos, proteínas, lípidos, sólidos por comida. Después de las 17 semanas juntamos nuestras bases de datos haciendo uso de la librería de Pandas(Rodríguez,2021).

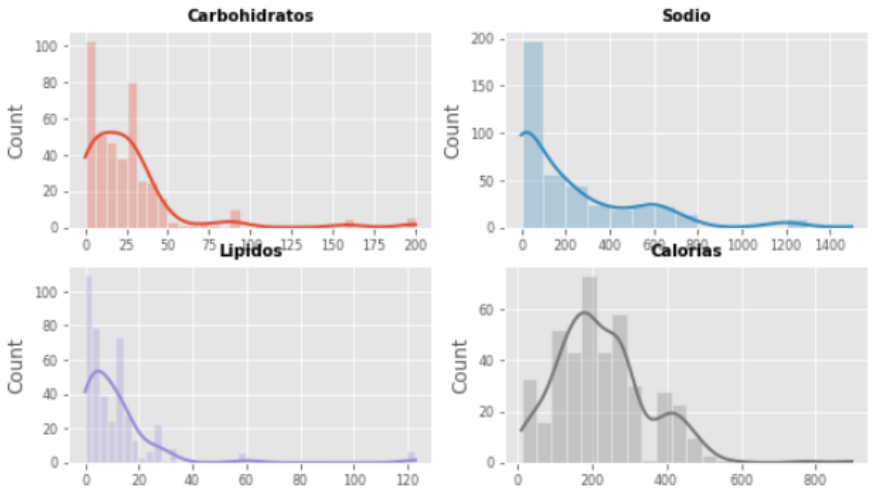
Para empezar a analizar los datos, primero separamos la base datos en 3 bases de datos diferentes, una para cada tipo de comida para hacer una gráfica de dispersión para ver de qué manera se dividen las clases. Después, creamos un modelo de entrenamiento con variables x, y para cada tipo de información nutrimental (Sodio, Proteína, etc.). Luego de esto, generamos un modelo de regresión logística para cada uno de nuestros modelos de entrenamiento para aproximar una predicción. Finalmente, a partir del modelo de entrenamiento y de la predicción generada por el modelo de regresión logística, generamos una matriz de confusión para cada variable, para observar y analizar la información obtenida.

Descripción de la solución

El primer paso de esta evidencia se realizó de manera individual, ya que consistió en reunir datos sobre la alimentación de cada integrante. Los datos recaudados incluían proteínas, horario de comidas, lípidos, carbohidratos, calorías, y sodio. Posteriormente, para este trabajo lo primero que hicimos fue reunir todas las bases de datos del equipo en una sola. De esta manera, podríamos utilizarlas como una misma en Python para obtener la regresión. Después de tener unificadas las bases de datos con la información nutrimental de cada integrante, el equipo realizó un análisis de la distribución de la columna calorías para determinar valores atípicos, eliminarlos y mejorar los resultados del análisis después. Luego, para terminar la limpieza de la base de datos; se hizo una categorización y una homogeneización con los datos de la columna “tipo de comida” para que tuviera sólo las variables de “comida”, “desayuno” y “cena”, ya que algunas bases de datos tienen variables como “11:22” o “cena”. Esto con el fin de mejorar la exactitud de los datos.

```
#fw significa first word y lo utilicé para categorizar si la comida era desayuno, comida o cena por medio de una separación de
#alimentos por hora en la que se comieron.
#Adicionalmente se agrega el tipo de comida en una lista que será una columna en una nueva base de datos creada por listas.
tip = []
for x in fw:
    if x in desa:
        tip.append("desayuno")
    elif x in comi:
        tip.append("comida")
    elif x in cena:
        tip.append("cena")
    else:
        tip.append("Modifícame")
```

Para empezar con el análisis de datos, lo primero que se hizo fue separar la base de datos en tres bases diferentes. De aquí, se realizó una gráfica de dispersión para cada base, las cuales representaban un tipo de comida cada una.

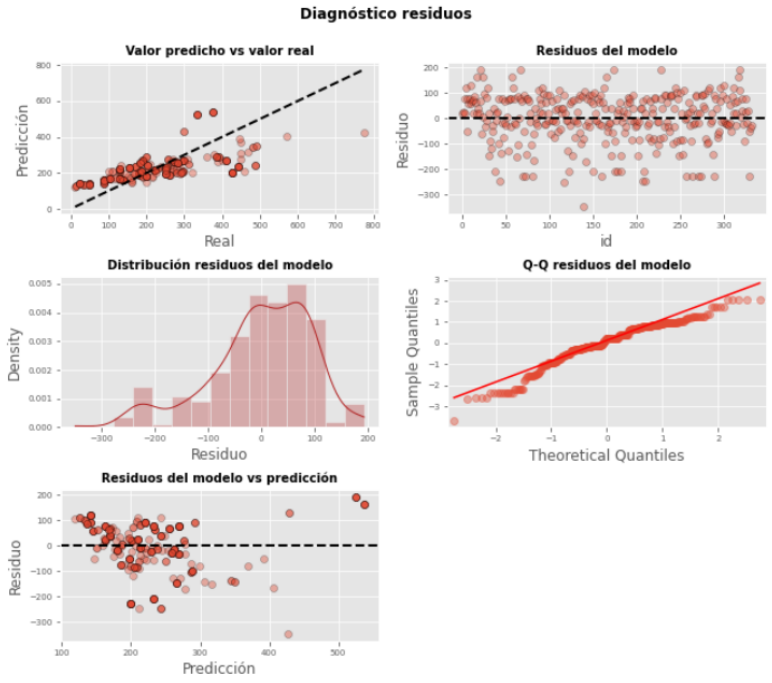


Después, el equipo creó un modelo de entrenamiento con variables x, y para cada tipo de dato nutricional (Sodio, Proteína, Calorías, Carbohidratos). Luego de esto, generamos un modelo de regresión logística para cada uno de nuestros modelos de entrenamiento para aproximar una predicción. Finalmente, a partir del modelo de entrenamiento y de la predicción generada por el modelo de regresión logística, generamos una matriz de confusión para cada variable, para observar y analizar la información obtenida.

```
X = datos[['Carbohidratos', 'Sodio', 'Lipidos', "Proteinas"]]
y = datos['Calorias']

X_train, X_test, y_train, y_test = train_test_split(
                                       X,
                                       y.values.reshape(-1,1),
                                       train_size   = 0.8,
                                       random_state = 1234,
                                       shuffle      = True
                                   )
```

Interpretación de los resultados.



(fig 1)

(fig 2)

(fig 3)

OLS Regression Results

Dep. Variable:	y	R-squared:	0.385
Model:	OLS	Adj. R-squared:	0.378
Method:	Least Squares	F-statistic:	51.20
Date:	Mon, 28 Nov 2022	Prob (F-statistic):	1.88e-33
Time:	12:30:58	Log-Likelihood:	-1982.7
No. Observations:	332	AIC:	3975.
Df Residuals:	327	BIC:	3994.
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	117.3156	10.087	11.630	0.000	97.471	137.160
Carbohidratos	1.6584	0.256	6.467	0.000	1.154	2.163
Sodio	0.0574	0.019	3.031	0.003	0.020	0.095
Lipidos	0.3206	0.496	0.647	0.518	-0.654	1.295
Proteinas	3.6469	0.457	7.988	0.000	2.749	4.545

Omnibus:	32.152	Durbin-Watson:	1.851
Prob(Omnibus):	0.000	Jarque-Bera (JB):	38.696
Skew:	0.782	Prob(JB):	3.96e-09
Kurtosis:	3.595	Cond. No.	673.

OLS Regression Results

Dep. Variable:	y	R-squared:	0.384
Model:	OLS	Adj. R-squared:	0.379
Method:	Least Squares	F-statistic:	68.25
Date:	Mon, 28 Nov 2022	Prob (F-statistic):	2.57e-34
Time:	12:31:00	Log-Likelihood:	-1982.9
No. Observations:	332	AIC:	3974.
Df Residuals:	328	BIC:	3989.
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	116.3585	9.970	11.671	0.000	96.746	135.971
Carbohidratos	1.7865	0.163	10.970	0.000	1.466	2.107
Sodio	0.0580	0.019	3.075	0.002	0.021	0.095
Proteinas	3.7393	0.433	8.631	0.000	2.887	4.592

Omnibus:	32.650	Durbin-Watson:	1.859
Prob(Omnibus):	0.000	Jarque-Bera (JB):	39.363
Skew:	0.795	Prob(JB):	2.83e-09
Kurtosis:	3.566	Cond. No.	666.

En las gráficas de la figura 1, se puede observar el comportamiento de los datos respecto al modelo de predicción por el entrenamiento, después de haber eliminado la variable que no cumple con la hipótesis nula, la cual fue la variable lípidos. El modelo final obtuvo una R^2 de 0.384, lo que significa que nuestro modelo es capaz de explicar el 38.4% de variabilidad de nuestros datos con la ecuación: *Calorías = 1.7865Carbohidratos + 0.0580Sodio + 3.7393Proteinas + 116.3585.* Consecuentemente, realizamos un test de normalidad con la predicción que pudo realizar python, y obtuvimos las gráficas de la figura 1.

En la primera gráfica, vemos la manera en la que python predice el comportamiento de los datos, comparados con el valor real. En la segunda gráfica, se muestra más detalladamente el comportamiento de los residuos. Mientras más datos estén en la línea 0, significa que la predicción fue más precisa. En la tercera gráfica, tenemos un histograma que muestra la distribución de los residuos, que se puede interpretar como normal debido a que forman la famosa campana, esto a que los datos están distribuidos conforme a su probabilidad de ocurrencia. Finalmente, utilizamos el test de Shapiro para encontrar el error del modelo, el cual arrojó 86.37881626515929 como resultado.

Después de hacer un análisis de los resultados que obtuvimos después de crear nuestros modelos. Obtuvimos que la primera vez que intentamos hacer el modelo de regresión con todos los datos nutrimentales, el p valor de lípidos era mayor a 0.5 por lo cual, lípidos, hacía que nuestro modelo no fuera significativo, por esto hicimos un segundo modelo después de eliminar lípidos de nuestras variables y después de hacer el modelo de regresión otra vez obtuvimos los coeficientes de todas las variables nutrimentales. Además obtuvimos que nuestra R2 fue de 0.384 como mencionamos anteriormente.

También como dijimos antes, según el test de Shapiro tenemos un error de 86.37881626515929, la causa que creemos probable es que no realizamos mediciones rigurosas, cuidando que las calorías fueran exactas, sino que si comíamos una comida, no registramos exactamente los ingredientes que los demás registraron. Adicionalmente, la base de datos analizada fue un conjunto de 4 bases de datos, cada una llenada por una persona diferente. Esto pudo haber ocasionado la variación de nuestros registros, volviendo los datos poco precisos

Conclusiones

En conclusión y como ha sido mencionado antes, aunque removimos la variable de lípidos para que nuestro modelo fuera más acertado el modelo final no es confiable. Sin embargo, considerando los coeficientes de ecuación obtenida (*Calorías = 1.7865Carbohidratos + 0.0580Sodio + 3.7393Proteínas + 116.3585*) así como el p valor de cada variable, las proteínas son el valor más importante para descifrar el número de calorías en un alimento seguidas de los carbohidratos y el sodio.

Bibliografía

CDC. (2022). *Efectos del Sobrepeso y la Obesidad en la Salud.* https://www.cdc.gov/healthyweight/spanish/effects.html

Di no a la obesidad, pero sí al ejercicio. (2021). *Gobierno de México*. https://www.gob.mx/profeco/documentos/di-no-a-la-obesidad-pero-si-al-ejercicio?state=published

El Sol de México. (2022, 29 abril). *Proyecciones y posibles repercusiones de la alta prevalencia de obesidad, a menos que actuemos.* https://www.elsoldemexico.com.mx/analisis/proyecciones-y-posibles-repercusiones-de-la-alta-prevalencia-de-obesidad-a-menos-que-actuemos-8209691.html

ESTADÍSTICAS A PROPÓSITO DEL DÍA MUNDIAL CONTRA LA OBESIDAD (12 DE NOVIEMBRE). (2020). *INEGI*. Recuperado de https://www.inegi.org.mx/contenidos/saladeprensa/aproposito/2020/EAP_Obesidad20.pdf

Gastosmedicos.Mx, E. (2021, July 26). *Causas y consecuencias de la Obesidad en México:* Gastos Médicos. https://gastosmedicos.mx/guias/obesidad-en-mexico-causas-y-consecuencias/

Gobierno de México. (2015). *10 Datos sobre la obesidad en México*. gob.mx. https://www.gob.mx/epn/es/articulos/10-datos-sobre-la-obesidad-en-mexico

Gobierno de México. (s. f.). *México y las políticas públicas ante la obesidad.* https://www.insp.mx/avisos/5091-dia-mundial-obesidad-politicas.html

Mayo Clinic. (2021). *Obesidad- Síntomas y causas.* https://www.mayoclinic.org/es-es/diseases-conditions/obesity/symptoms-causes/syc-20375742

MyFitnessPal | *MyFitnessPal*. (s. f.). https://www.myfitnesspal.com/es

MyNetDiary. (s. f.). MyNetDiary - Free Calorie Counter and Diet Assistant. https://www.mynetdiary.com/

Rodríguez, D. (2021, 3 julio). Pandas: *Cambiar los tipos de datos en los DataFrames. A*nalytics Lane. https://www.analyticslane.com/2021/07/15/pandas-cambiar-los-tipos-de-datos-en-los-dataframes/