Universitat Autònoma de Barcelona

PRÁCTICA 1-APC
REGRESIÓN

Grupo A403-1130

Adrià Orozco Lorente 1490952
Arnau Cruz Gargallo 1494996

ÍNDICE

INTRODUCCIÓN

En esta práctica realizaremos una introducción a la aplicación de modelos de regresión en ejemplos reales de manera que se estudiarán todo tipo de datos con tal de realizar un estudio de los mismos con el objetivo de evaluar errores y validar resultados.

Un modelo de regresión es un modelo matemático que busca determinar la relación entre una variable independiente, con respecto a otras variables. Este se utiliza con el fin de determinar si existe o no una relación entre una variable independiente y las demás variables así mismo buscando determinar cuál es el impacto sobre estas.

Para realizar estos modelos, es necesario el uso de técnicas de regresión, estas son un tipo de técnicas estadísticas usadas para el modelado predictivo y la minería de datos.

Para aplicar estas técnicas se usará un notebook con lenguaje python que usa librerías como pueden ser numpy, scikit-learn, matplotlib o spicy, las cuales serán útiles para evaluar el comportamiento de las variables del modelo y extraer conclusiones sobre este.

OBJETIVOS

Como objetivos principales de esta práctica encontramos conocer la aplicación de modelos de regresión, sobre todo haciendo énfasis en analizar los atributos para seleccionar los más representativos y normalizarlos, evaluar correctamente el error en el modelo, visualizar los datos y el modelo resultante y saber aplicar el proceso de descenso de gradiente.

Además, también ser capaz de aplicar técnicas de regresión en casos reales, validar los resultados con datos reales, y fomentar la capacidad de presentar resultados técnicos de aprendizaje computacional de forma adecuada delante de otras personas.

PRESENTACIÓN DE LA BASE DE DATOS

El caso kaggle que nos ha tocado estudiar y analizar es el siguiente dataset de abalone: https://www.kaggle.com/rodolfomendes/abalone-dataset

Para entender un poco mejor la base de datos explicaremos que es un abalone. El término “abalone” o también conocido como haliótidos es una familia de moluscos gasterópodos conocidos como orejas de mar.

La base de datos contiene diferentes muestras con los siguientes atributos físicos para cada muestra: Sex, Length, Diameter, Height, Whole weight, Shucked weight, Viscera weight, Shell weight, Rings.
Mediante estas muestras deberíamos ser capaces de predecir la edad del abalone.

APARTADO C

En este apartado se procede a cargar todos los datos relacionados con la base de datos de nuestro grupo de prácticas, en este caso, nuestra base de datos como se ha comentado anteriormente contiene muestras de moluscos gasterópodos. Mediante la carga de todos estos valores se procede a manipular-los de forma que podamos entender todo lo que significa y a partir de aquí se fijara un atributo objetivo a predecir mediante los otros atributos de la base de datos.

Para realizar esto primeramente se crea un nuevo notebook al que llamaremos Abalone_Notebook.ipynb, localizado en la carpeta Code de nuestro proyecto en Github.

En nuestro caso, ya teníamos instalados el entorno necesario para ejecutar este notebook, y las librerías a utilizar a procesar estos datos, debido a prácticas realizadas en otras asignaturas. Así, de esta manera comenzamos importando las siguientes librerías:

Sklearn: Esta librería sirve para realizar el aprendizaje automático, y nos ayudara a realizar el análisis predictivo. Incluye varios algoritmos de clasificación, regresión y análisis de grupos.

Numpy: Esta librería nos da soporte para crear vectores y matrices de una dimensión considerable y multidimensional, está creada precisamente para procesar grandes cantidades de datos de la forma más óptima posible y contiene una larga colección de funciones matemáticas de alto nivel para operar con ellas.

Pandas: Esta librería es una extensión de la librería de numpy, hecha para realizar la manipulación y el análisis de datos en python, ofreciéndonos así estructuras de datos y operaciones para manipular tablas numéricas y series temporales, de manera que con esta librería podremos importar de forma sencilla nuestra base de datos.

Matplotlib: Esta librería consigue la generación de gráficos a partir de datos contenidos en listas, los cuales habrán sido importados en nuestro caso previamente con uso de las funciones de la librería pandas, de manera que podamos representar los datos deseados de manera gráfica usando varios tipos de técnicas.

Scipy: Esta librería nos será útil para aplicar módulos de optimización, algebra lineal, e integración de datos entre otras, cabe destacar que es parte del conjunto de la biblioteca numpy y extiende bibliotecas de computación científica.

Con estas librerías importadas en nuestro notebook, el primer paso es cargar la base de datos mediante un archivo csv, con la función read_csv de la librería pandas, a la cual le indicaremos por parámetros que nuestros valores están separados por comas en el archivo origen.

Esta carga de la base de datos que se acaba de mencionar será guardada en una variable llamada dataset, la cual ahora contiene una matriz de datos en la cual las filas corresponden

a los valores de la misma base de datos, y las columnas corresponden a cada atributo de la misma.

Para hacer una prueba de que la carga ha funcionado correctamente podemos aplicar la función head() la cual nos devolverá las primeras 5 filas guardadas en nuestro dataset.

Gracias a la aplicación de esta función podemos apreciar los diferentes atributos que contiene nuestra base de datos, los cuales explicamos a continuación así respondiendo a la pregunta también de **Cuál es el tipo de cada atributo**:

“Sex”: Se refiere al género que puede tener el molusco en cuestión, cogiendo los valores de M (Para el sexo masculino), F (Para el sexo femenino) y I (Para los infantiles). Como podemos ver el tipo de este atributo es un carácter de tipo letra con longitud 1.

“Length”: Se refiere a la medición del caparazón del molusco, y esta medido en milímetros, de manera que su tipo de dato para ser representado es un numero flotante o decimal.

“Diameter”: Se refiere al diámetro del caparazón del molusco, o lo que es lo mismo la medida perpendicular a la longitud de su caparazón, este atributo también se expresa en milímetros y su tipo de dato para ser representado es también un numero flotante o decimal.

“Height”: Se refiere a la altura total del molusco teniendo en cuenta su cuerpo a parte del caparazón en milímetros, y su tipo de dato para ser representado es también un numero flotante o decimal.

“Whole Weight”: Se refiere al peso total de todo el molusco, en gramos, y su tipo de dato para ser representado es también un numero flotante o decimal.

“Shucked weight”: Se refiere al peso del molusco sin contar el caparazón, es decir, la carne del mismo, también expresado en gramos y su tipo de dato para ser representado es también un numero flotante o decimal.

“Viscera weight”: Se refiere al peso intestinal del molusco, expresado también en gramos y su tipo de dato para ser representado es también un numero flotante o decimal.

“Shell weight”: Se refiere al peso del caparazón del molusco, expresado también en gramos y su tipo de dato para ser representado es también un numero flotante o decimal.

“Rings”: Anillos que tiene el molusco, representado como un numero entero positivo, cabe destacar que sabiendo el número de anillos, se puede obtener la edad del molusco sumando 1,5 al número de anillos que tiene.

Una vez conocidos los datos con los que trataremos, observaremos como se representan estos en su totalidad y que relación guardan con otros atributos, con el fin de eliminar atributos redundantes y establecer relaciones para predecir una variable objetivo la cual tendremos que especificar también a partir de la representación de estos datos.

Para comenzar a observar cómo se representan los datos, usaremos funciones de la librería pandas primeramente.

El primer paso es saber la dimensión total de los datos obtenidos en nuestra variable dataset el cual podremos saber con la función shape(), así obtenemos que nuestro dataset contiene un total de 4177 filas.

Posteriormente necesitamos saber si existen valores nulos en alguna de las filas de nuestro dataset debido a que estos valores no nos servirán posteriormente para el análisis de datos, así que se hace uso de la función isnull combinada con la función sum() de manera que obtenemos los valores nulos que existen por cada columna de nuestro dataset, dando así como resultado que no contiene valores nulos, de manera que en este apartado no descartaremos datos todavía.

Después de haber realizado estos dos simples pasos, pasamos a utilizar la función describe() la cual nos da datos relevantes sobre cada columna de nuestro dataset, tal como la media de valores, los máximos, los mínimos, los percentiles, etc.
De manera que así podemos observar posibles inconsistencias en los datos para seguir realizando la criba de estos. En este caso vemos que los valores representados con estas funciones parecen correctos a priori, pero nos llama la atención que la altura mínima encontrada es igual a 0, cosa que no tiene sentido, así que se procede a la búsqueda de las filas del dataset que contengan este valor filtrando por el atributo “Height” e igualándolo a 0.
Esta consulta nos devuelve dos filas, las cuales habrá que descartar ya que aportan inconsistencias. Para descartar estas filas hacemos uso de la función drop() a la cual como parámetros le pasaremos en una lista los índices de las filas con altura igual a 0.
Posteriormente se comprueba de nuevo la misma consulta para ver que no encuentra resultados y se comprueba de nuevo la dimensión del dataset para ver que contiene dos filas menos.

Siguiendo estos procedimientos, sabemos que se puede calcular la edad de un molusco sumando 1,5 al número de sus anillos, así que se crea una variable llamada age que sea igual al resultado de esta suma, utilizando la siguiente sentencia:
dataset[‘age’] = dataset.Rings +1.5

Como paso extra, podemos decir que renombramos el nombre de las variables para evitar problemas luego de consultas como pueden ocurrir a partir de espacios entre los nombres, o el uso de mayúsculas. Así que se cambian los espacios por “_”, y se sustituyen las mayúsculas por minúsculas usando la función rename de la librería pandas pasando por parámetro las columnas con su nombre antiguo y especificando el nuevo.

Para comprobar que estos cambios han sido ejecutados correctamente podremos utilizar de nuevo la función head() para ver mismamente los primeros nuevos valores y así comprobar que los datos han sido actualizados y además aparece el nuevo atributo **“age”**.

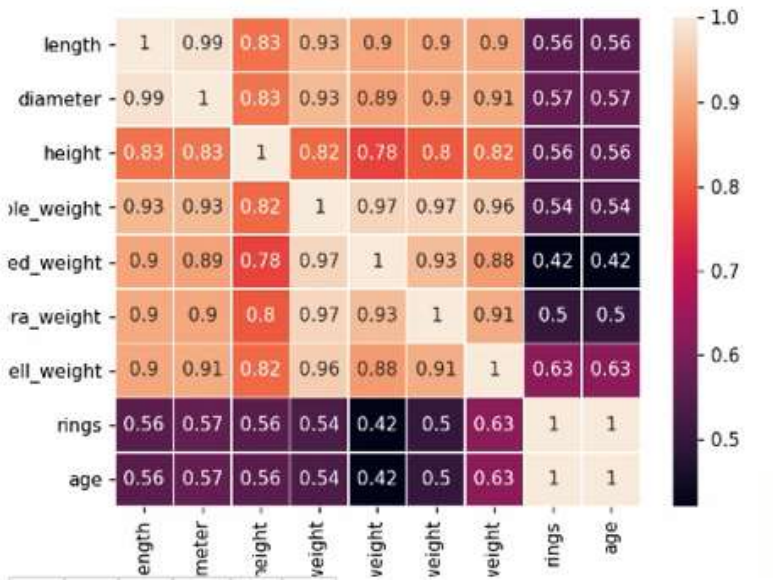
	sex	length	diameter	height	whole_weight	shucked_weight	viscera_weight	shell_weight	rings	age
0	M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	15	16.5
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	7	8.5
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	9	10.5
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	10	11.5
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	7	8.5

En este instante, se procede a utilizar funciones de la librería matplotlib, con objetivo de ver la distribución de los atributos de forma general, y de intentar establecer relaciones entre ellos por tal de obtener las primeras conclusiones acerca de los datos para posteriormente definir una variable objetivo e intentar predecirla a partir de las que más relación tengan.

Primeramente, y previo a la realización de los gráficos estudiamos la correlación entre variables para hacernos una idea de cuales pueden estar relacionados de alguna manera:

	length	diameter	height	whole_weight	shucked_weight	viscera_weight	shell_weight	rings	age
length	1.000000	0.986802	0.828108	0.925217	0.897859	0.902960	0.898419	0.556464	0.556464
diameter	0.986802	1.000000	0.834298	0.925414	0.893108	0.899672	0.906084	0.574418	0.574418
height	0.828108	0.834298	1.000000	0.819886	0.775621	0.798908	0.819596	0.557625	0.557625
whole_weight	0.925217	0.925414	0.819886	1.000000	0.969389	0.966354	0.955924	0.540151	0.540151
shucked_weight	0.897859	0.893108	0.775621	0.969389	1.000000	0.931924	0.883129	0.420597	0.420597
viscera_weight	0.902960	0.899672	0.798908	0.966354	0.931924	1.000000	0.908186	0.503562	0.503562
shell_weight	0.898419	0.906084	0.819596	0.955924	0.883129	0.908186	1.000000	0.627928	0.627928
rings	0.556464	0.574418	0.557625	0.540151	0.420597	0.503562	0.627928	1.000000	1.000000
age	0.556464	0.574418	0.557625	0.540151	0.420597	0.503562	0.627928	1.000000	1.000000

Para observarlo mejor podemos crear un mapa de calor respecto a esta tabla de correlaciones de forma que obtengamos un resultado más vistoso para entender todavía mejor las relaciones entre los atributos de nuestro dataset, facilitando así la toma de decisiones en los gráficos que se pretenden contraer de forma posterior:

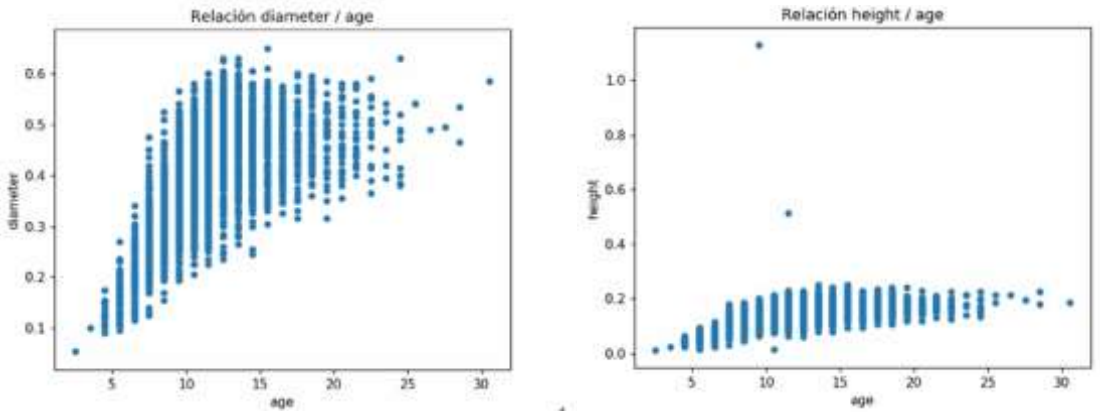


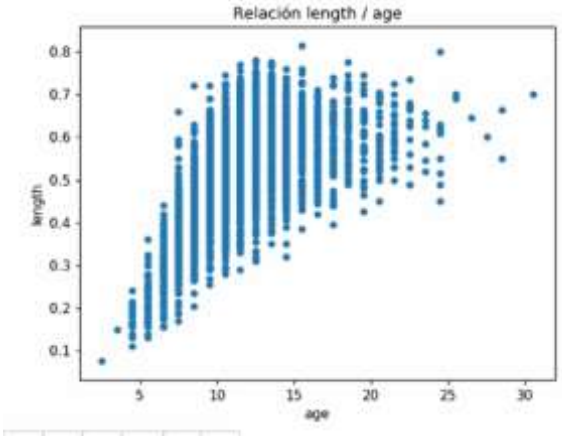
Viendo los siguientes datos podemos considerar estrechas relaciones entre todas las variables excepto la edad, lo cual pueden suponer que estas pueden intentar ser las variables a predecir.

Para encontrar datos más relevantes realizamos gráficos que consisten en nubes de puntos entre dos atributos con los que podremos observar ciertas similitudes en la distribución que estos representan.

Como primeros resultados ante estas graficas podemos decir que los atributos que guardan más relación entre si son aquellos relacionados con el tamaño y la edad los cuales muestran por supuesto variancias según la muestra, pero demuestran una relación del tipo proporcional, de manera que a mayor edad mayores son los tamaños de los moluscos.

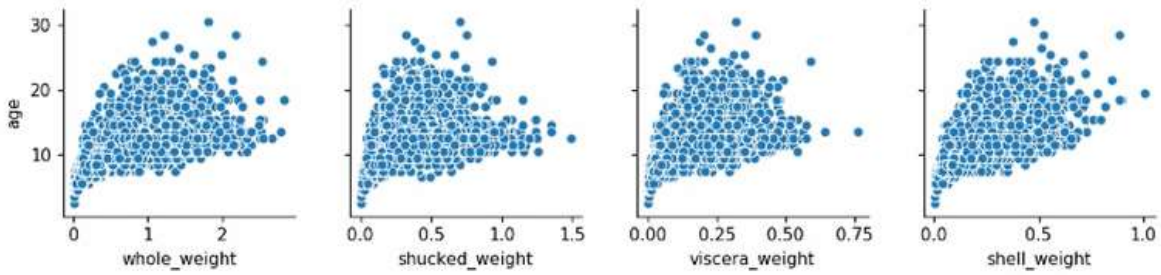
Como podemos ver en alguno de los siguientes gráficos:



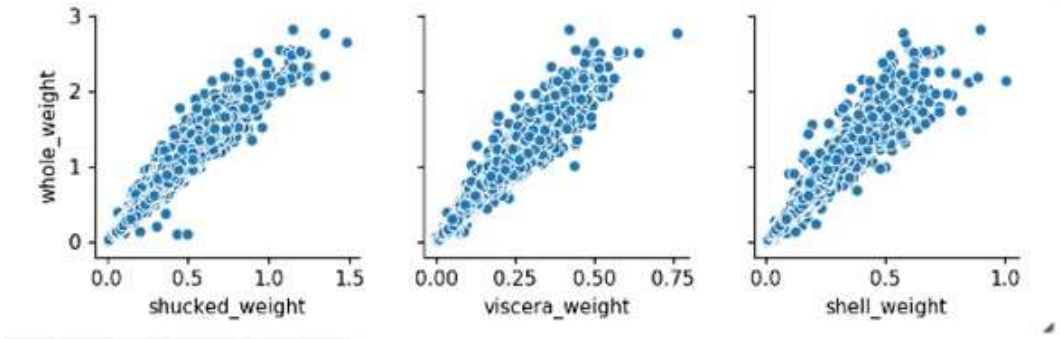


Por supuesto la relación entre los anillos y la edad de los moluscos es totalmente proporcional por la fórmula que hemos explicado anteriormente por lo que no conviene incluir la variable en el estudio en este caso.

Habiendo observado también la tabla de correlaciones, vemos que existe mucha correlación entre las variables que indican los distintos pesos del molusco y la variable que indica el peso total. Como la variable del peso de la carne del mismo es la que más correlación tiene con la edad vamos a observar las relaciones entre ellas y esta última para ver cuál es la mejor opción para realizar comparaciones.

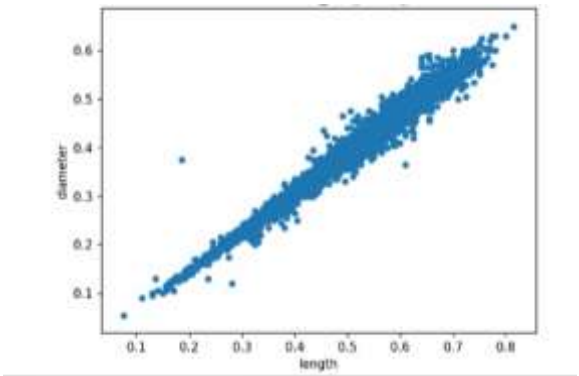


En la imagen anterior comparamos las relaciones entre los pesos y la edad para saber, la estimación real entre los diferentes atributos relacionados con el peso del molusco, viendo así que la correlación que hay entre ellos realmente es muy parecida.



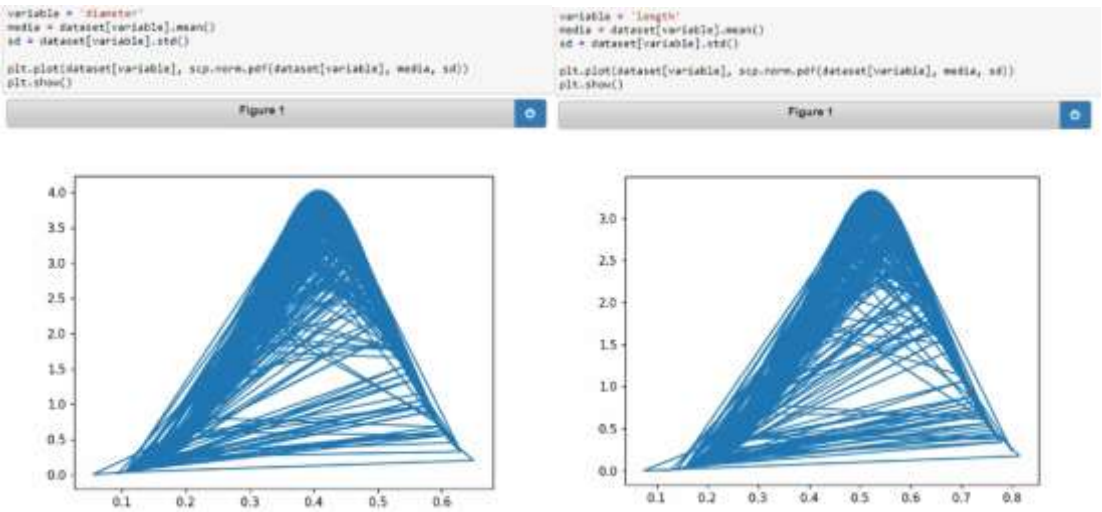
Por lo que comparamos todos los pesos del molusco con el peso total, de manera que vemos que nos salen graficas que son prácticamente iguales, lo que es un indicativo de que “whole_weight” puede ser una de las variables predictoras y podremos descartar los demás pesos.

Otra de las correlaciones destacables puede ser la que hemos comparado anteriormente entre el diámetro y la longitud, los cuales tienen una correlación muy similar como se puede ver en la siguiente imagen:



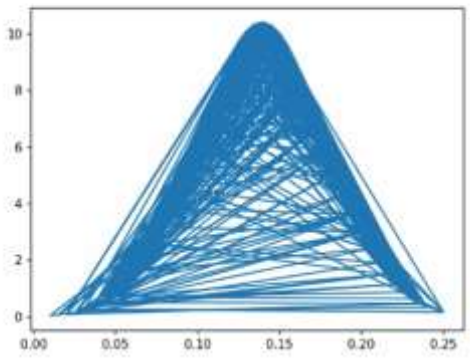
Este grafico nos puede dar otro indicador de que una de estas variables puede ser descartada para el posterior estudio y la otra puede ser una de las predictores.

Otro de los gráficos que podemos hacer para analizar los datos es el de la distribución normal, con los que podemos responder a la pregunta de **qué atributos tienen una distribución Gaussiana** de manera que calculamos la media y la desviación standard de cada uno de ellos y los representamos mediante la función norm de scipy.stats y así podremos ver la distribución que siguen los datos obtenidos.
Como primeros resultados obtenemos que los atributos principales con distribución Gaussiana son la longitud y el diámetro:



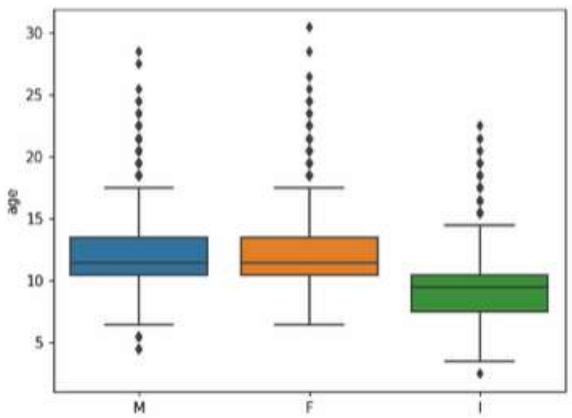
Otro de los gráficos que nos llama la atención es el referente a la altura el cual muestra varios outliers que alteran la distribución de esta, por lo que se procede a borrarlos y de nuevo ejecutar el grafico de la distribución normal de forma que queda de la siguiente manera:

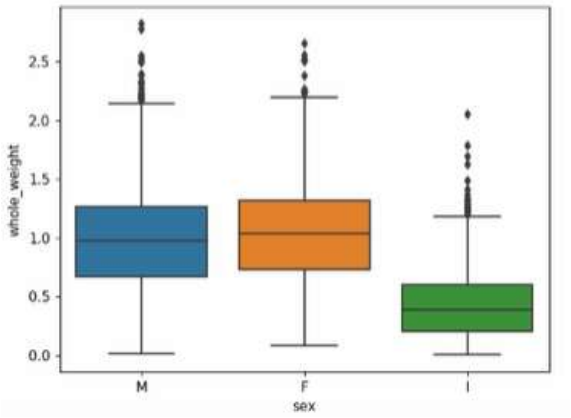
```
variable = 'height'
media = dataset[variable].mean()
sd = dataset[variable].std()

plt.plot(dataset[variable], scp.norm.pdf(dataset[variable], media, sd))
plt.show()
```

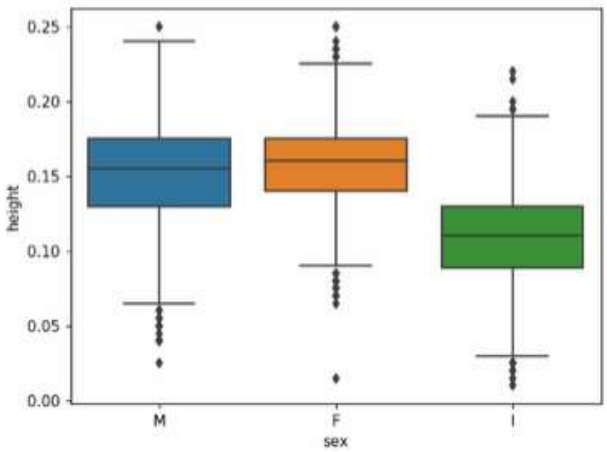


En cuanto a los gráficos relacionados con el peso, sí que es cierto que representan un punto máximo donde se predicen la mayoría de datos pero comienzan desde un valor elevado no igual a 0 por lo que no los consideramos una distribución gaussiana.

Para continuar la búsqueda de posibles relaciones entre atributos, veremos si el sexo del molusco influye en las medidas y pesos de este así como con la edad, de manera que si así fuera, la consideraríamos otra de las posibles variables predictoras. Para observar esto, se realizan varios gráficos comparando todos estos datos en su conjunto.







Como resultados más relevantes del estudio de la variable “sex” encontramos que tiene cierta relación con la edad y las anteriores variables que hemos considerado interesantes, por lo que puede aportar información relevante para la clasificación de datos de tal manera que se consiguen clasificar los moluscos infantiles en una altura peso y edad menor, mientras que los moluscos masculinos y femeninos ocupan una distribución de datos más normalizada aunque presenta cierta tendencia a que aquellos femeninos son algo más altos y pesados a mayor edad, pero no por mucho.
Teniendo en cuenta esto, podemos observar que “sex” puede ser un atributo interesante a tener en cuenta

Después de haber analizado cada uno de los atributos, consideramos que **el atributo objetivo a predecir es** Age, debido a la poca correlación que mantiene con las otras variables, y que las demás están muy correlacionadas entre sí, además consideramos que es interesante poder intuir la edad de un molusco a través de sus dimensiones, peso y sexo, de manera que esta es una de las fundamentales decisiones para tener en cuenta el atributo objetivo.

Con este paso, finalizamos el análisis de los datos para posteriormente entrenar modelos y poder predecir la variable objetivo entre otras operaciones, cosa que se muestra seguidamente en el apartado B.

APARTADO B

Para empezar con este apartado, se procede a continuar con el analsis de datos realizado en al apartado anterior, debido a que nos interesa hacer una regresion lineal para la predicción de la variable objetivo obtenida, de manera que eliminaremos los atributos que a nuestro juicio nos parezcan redundantes para el estudio, o tengan poca relación con la variable objetivo, de esta manera también podemos responder a la pregunta de **cuáles son los atributos más importantes para hacer una buena predicción** los cuales como hemos dicho ya, son aquellos que guardan una mejor correlación con nuestra variable objetivo, y además por otra parte también es interesante usar aquellos que siguen una distribución gaussiana.

La decisión tomada en cuanto a la criba de datos se basa en que usaremos las columnas "diameter, height, whole_weight, sex y age"

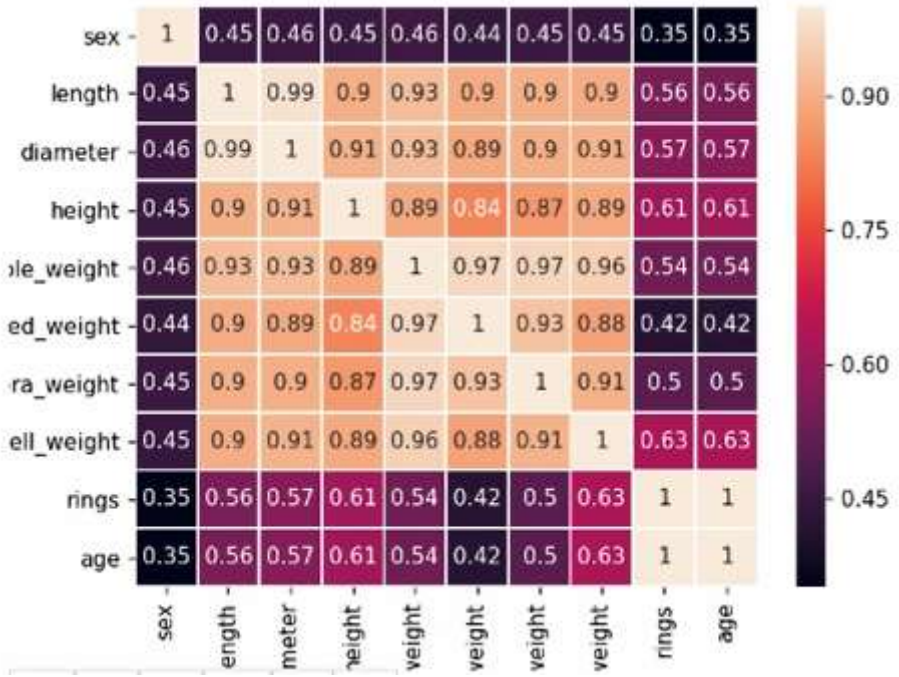
La decisión de eliminar las columnas referentes al peso del molusco solamente quedándonos con la que nos indica el peso total, viene dada de que la correlación entre ellas con el peso total es prácticamente del 100%, cosa que indica que los resultados no contendrán una gran variación entre ellas y por tanto nos podemos quedar con un atributo que sea más general, igual caso que entre longitud y diámetro, si bien el parámetro de longitud nos puede ser beneficioso en el momento del estudio, consideramos que debido a su estrecha relación con el diámetro y las gráficas mostradas, este último se puede adaptar mejor ante el estudio de la variable objetivo Age.

También eliminamos el atributo "rings" ya que no tiene sentido para la realización del siguiente modelo predictivo ya que de antemano sabemos que la edad la hemos obtenido a partir de este dato previamente y no encontraríamos conclusiones relevantes frente a la hipótesis general planteada, que viene a ser si se puede determinar la edad de un molusco de este tipo a partir de sus dimensiones, altura y sexo, como se ha comentado previamente.

Por otra parte como se ha visto que el sexo del molusco tiene una cierta relación entre estas variables al ver los diagramas de tipo caja, se procede a cambiar sus valores que están expresados en caracteres, por valores enteros comprendidos entre el 0 y el 2 para posteriormente poder filtrar mejor los datos.

Previamente al borrado de estos atributos volvemos a mostrar la correlación entre todos los datos tal y como se pide en el apartado, así respondiendo a la pregunta **que correlación hay entre los atributos de nuestra base de datos,** destacar que se había mostrado previamente en el apartado anterior, pero se vuelve a realizar la tabla y su diagrama de calor con el cambio realizado en la variable sexo:

	sex	length	diameter	height	whole_weight	shucked_weight	viscera_weight	shell_weight	rings	age
sex	1.000000	0.448156	0.457655	0.451387	0.460626	0.440336	0.454032	0.445524	0.351541	0.351541
length	0.448156	1.000000	0.986794	0.900868	0.925328	0.898129	0.903033	0.898363	0.556572	0.556572
diameter	0.457655	0.986794	1.000000	0.907187	0.925499	0.893330	0.899716	0.906026	0.574551	0.574551
height	0.451387	0.900868	0.907187	1.000000	0.888850	0.837485	0.866757	0.891857	0.610107	0.610107
whole_weight	0.460626	0.925328	0.925499	0.888850	1.000000	0.969370	0.966290	0.955954	0.540621	0.540621
shucked_weight	0.440336	0.898129	0.893330	0.837485	0.969370	1.000000	0.931831	0.883194	0.421156	0.421156
viscera_weight	0.454032	0.903033	0.899716	0.866757	0.966290	0.931831	1.000000	0.908133	0.503977	0.503977
shell_weight	0.445524	0.898363	0.906026	0.891857	0.955954	0.883194	0.908133	1.000000	0.628169	0.628169
rings	0.351541	0.556572	0.574551	0.610107	0.540621	0.421156	0.503977	0.628169	1.000000	1.000000
age	0.351541	0.556572	0.574551	0.610107	0.540621	0.421156	0.503977	0.628169	1.000000	1.000000



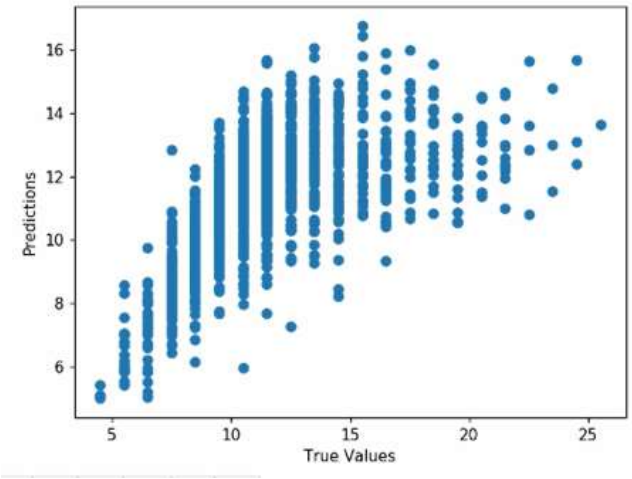
Como podemos observar la correlación es la misma que habíamos comentado previamente, con la diferencia de que age ha sido cambiado con valores arbitrarios establecidos por nosotros, vemos que de esta manera no guardan mucha correlación, pero sabemos que si pueden ser interesantes para el estudio, y además en el filtrado de datos podremos retirar los infantes por ejemplo que sabemos que son menores de 10 años debido a los diagramas anteriormente realizados.

Para realizar el borrado de las columnas comentadas, se hace uso de la función drop con parámetros de los nombres de las columnas deseadas a borrar.

El paso siguiente a la realización de este primer filtrado consiste en realizar modelos de entrenamiento y de test teniendo en cuenta estas variables predictores y la variable objetivo, de tal forma que con estos conjuntos generados, posteriormente podemos observar la

tendencia de la regresión y comprobar resultados entre ellos la media del error cuadrático para saber si nos estamos alejando mucho respecto a los valores reales.
Para realizarlo hacemos uso de la funcion train_test_split a la cual le indicamos los atributos de nuestro dataset que queremos utilizar como variables predictoras y nuestra variable objetivo, además añadiendo así un tamaño que en nuestro caso hemos decidido que sea de un 30 por ciento respecto a los valores del conjunto de test para las posteriores comprobaciones. Posteriormente a esto se generará la regresión lineal y se entrenara al modelo para poder predecir resultados.
Previamente a establecer todos los datos del dataset para entrenar los modelos, realizamos estos mismos pero solo teniendo en cuenta uno de los atributos en cada caso de tal forma que podamos observar la media del error cuadrático por cada uno de los atributos y así poder responder a la pregunta de **con que atributo se consigue un MSE menor.**
Los resultados obtenidos son los siguientes:
MSE: Height 6,97
MSE: Diameter 7,4
MSE: Whole_Weight 7,8
MSE: Sex 9,3
Por lo que podemos decir que con el atributo referente a la altura obtenemos un MSE menor.

En este instante se procede a la creación de un nuevo modelo de regresión lineal el cual tendrá en cuenta todos los atributos predictores mencionados anteriormente, de forma que intentaremos predecir la variable age a través de todos los factores tenidos en cuenta, siguiendo el mismo procedimiento.
El resultado que obtenemos lo comparamos respecto a predecir la variable age mediante solo el uso de uno de los atributos de nuestra base de datos de forma que el nuevo resultado de la media del error cuadrático es 6,9, ligeramente más óptimo que en nuestro mejor atributo “height”.



Para comprobar el resultado de forma gráfica representamos en una nube de puntos los resultados encontrados con los valores reales. Como podemos observar la tendencia con la que se encuentran los valores precedidos es la correcta, pero el hecho de tener una gran cantidad de valores tan dispares entre si hace que durante la predicción se produzcan alteraciones a mayor escala, por lo tanto es necesario normalizar los datos para obtener unos mejores resultado.

Podemos decir **que la normalización influye en la regresión** de manera que reduce el rango máximo de los valores de los atributos con objetivo de dejar de tener tanta influencia en el error cuadrado, por tanto reducimos el tamaño de los datos mediante la normalización o estandarización por tal de obtener valores más cercanos y poder reducir este error medio.
Para ello estandarizamos todos los datos de nuestro conjunto, ya que no tendría sentido normalizar uno y dejar los demás en sus valores absolutos y volvemos a entrenar el modelo con los nuevos datos y a comprobar el MSE.
Ahora con los datos comprendidos en un rango menor debido a la estandarización que tiene en cuenta la media de los datos y la desviación estándar, el MSE baja a 0,6, de manera que a simple vista podemos observar que los resultados son más óptimos que si no se hubiera realizado esa normalización. También considerar que esta reducción es proporcional al MSE que se había encontrado sin haber realizado este paso, ya que afecta a todos los datos de nuestro conjunto.

Para saber **cómo mejora la regresión cuando se filtran los atributos de las muestras que no contienen información** en nuestro caso, cargamos de nuevo la base de datos en un nuevo dataset al que llamamos dataset2, al que le añadiremos la misma configuración que teníamos en el anterior dataset, pero con la diferencia que observaremos los valores más alejados de la distribución de las variables, con lo que decidimos eliminar las muestras de los atributos que tengan más de 20 años de edad, con lo que así conseguimos bajar 2 puntos el MSE, por otra parte se eliminan las muestras también que pesen más de 1.6g como peso total consiguiendo así reducir el error 0,5 puntos más.
De manera que ahora el MSE pasa a ser de 4,1 mientras antes era de 6,7.
Incluso normalizando datos el resultado mejora pasando de obtener un MSE de 0,6 a uno de 0,58.

En cuanto **a la aplicación de un PCA** cuyo significado viene a ser el de encontrar y eliminar variables que entre ellas sean prácticamente iguales por lo tanto cuya correlación sea muy cercana a 1, lo hemos realizado anteriormente para el filtrado de atributos, como se ha comentado al principio del apartado, de manera que eliminamos los atributos de lenght,

shucked_weight, shell_weight, y viscera_weight, cuya explicación se encuentra al inicio del apartado, por lo tanto el **espacio se reduce** de 9 atributos a 5 atributos contando entre ellos la variable objetivo.