Overview

In this assignment, you'll practice some data manipulation in pandas, and perform a simple regression task using the Abalone data set from the UCI repository

Abalone are a type of mollusk (you may have eaten one before). Each row of this data set is an individual abalone, with a variety of measurements. You can learn about the age of an abalone by counting rings in its shell (kind of like counting tree rings).

https://archive.ics.uci.edu/ml/datasets/Abalone

This link describes the data. Take a look at the attribute information so you know what each column is referring to.

Please put all your code into a single script `hw2.py`, which will generate the image `regression.png`. Submit both files to canvas.

Please submit the following files to Canvas:

1. `hw2.py` - a script with all your code, which will generate the figure
2. `regression.png`

Additionally, **please include comments in your code** to explain what you're doing (doesn't have to be detailed, but should be clear).

1 Read the data into a DataFrame

use pandas to read the data at this link into a DataFrame:
https://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.data

Hints:

- decide if `pd.read_table` or `pd.read_csv` is more appropriate.
- you can pass the url provided as a string (like in the Lecture 5 exercise)
- You'll want to pass in at least 2 keyword arguments:
  - `names` - a list of column names (`'sex', 'length', 'diam', 'height', 'wt_whole', 'wt_shucked', 'wt_viscera', 'wt_shell', 'rings'`)
  - `index_col=False` - indicates that pandas should just create an index for each entry (we don't have ids for the data)

2 Set up the data for a regression problem

The problem we're going to try to solve is to predict the number of rings in an abalone shell (the `'rings'` column) from the other features. This is a proxy for the age of the animal.

This means we want a response vector y that contains the data in the 'rings' column, and a design matrix X that contains all the data we'll use to predict the response (the other columns).

Use the `pasty` library's `dmatrices` function to form your data and response matrices (see lecture 6 for an example).

Hints:

- Since ‘sex’ is categorical you’ll want to use `‘C(sex)’` in your model specification.

you’ll see that `X` has a column called ‘Intercept’. We will not need this column, so remove it from the dataframe:

```
X.drop(X[["Intercept"]], axis=1, inplace=True)
```

3 Split the data into train and test sets

Now, we’re going to start using Scikit learn.

Split your data into train and test sets (See lecture 6 for an example)

Set your test size to be 30% of the data

4 Fit a Linear Regression model to the data

1. Use Scikit learn’s linear regression class to fit the model:

   ```
   from sklearn.linear_model import LinearRegression
   ```

2. Use your training data to fit the model
3. Predict the number of rings in your test data using the `predict` method.
4. create a scatter plot of your prediction vs. the true number of rings. Save this figure as `regression.png` and sumbit it with your homework

5 (Bonus) Try another regression classifier

Pick another regression classifier (e.g., try ridge, lasso, decision trees, nearest neighbors, ...) and repeat parts 3 and 4. If you do this, name your image after the classifier you used e.g., `lasso.png`