

Practical 1

Jumping Rivers

Object Oriented Programming

For the practical we are going to create a `Model` class which fits a linear regression model on initialisation and then add some methods for common things that we might wish to do with our regression models.

The code below will define a `Model` class which fits the model given an input, `X`, and response `y`.

```
class Model:
    def __init__(self,X,y):
        self.X = X
        self.y = y
        if len(X.shape) == 1:
            X = X.values.reshape(-1,1)
        X = np.c_[np.ones(X.shape[0]),X]
        xTx = np.dot(X.T,X)
        inverse_xTx = np.linalg.inv(xTx)
        xTy = np.dot(X.T,y)
        coef = np.dot(inverse_xTx,xTy)
        self.intercept = coef[0]
        self.gradients = coef[1:]
```

Our model class has 4 instance attributes

- `.X` the input predictor variables
- `.y` the output response variable
- `.intercept` the intercept of the fitted model
- `.gradients` the estimated gradients of the fitted model

1. First we will add a `predict` method to our model class. The following definition will perform a model prediction given an input `X`, intercept, `intercept` and gradients, `gradients`.

```
def predict(X, intercept,gradients):
    if len(X).shape == 1:
        X = X.values.reshape(-1,1)
    return intercept + np.dot(X,gradients)
```

Modify the function to use the instance attributes of the model and add it as a method to the `Model` class.

2. Add magic methods to the class that allows `print(model)` and `model` to show the intercept and gradients of the fitted model

3. Update your initialisation method to automatically calculate the fitted values and residuals of the model and store them as instance attributes. Fitted values are calculated by predicting from the model using the original **X** inputs. Residuals are defined as the fitted values - observed response.
4. Create a new class which inherits from `Model`. Add a `plot_one` method which could be used to show a scatter plot of the response against a chosen input, together with a line to show the fitted model.
5. Add a class method to your new subclass that allows you to take an existing instance of the original `Model` class and create an instance of the new subclass.