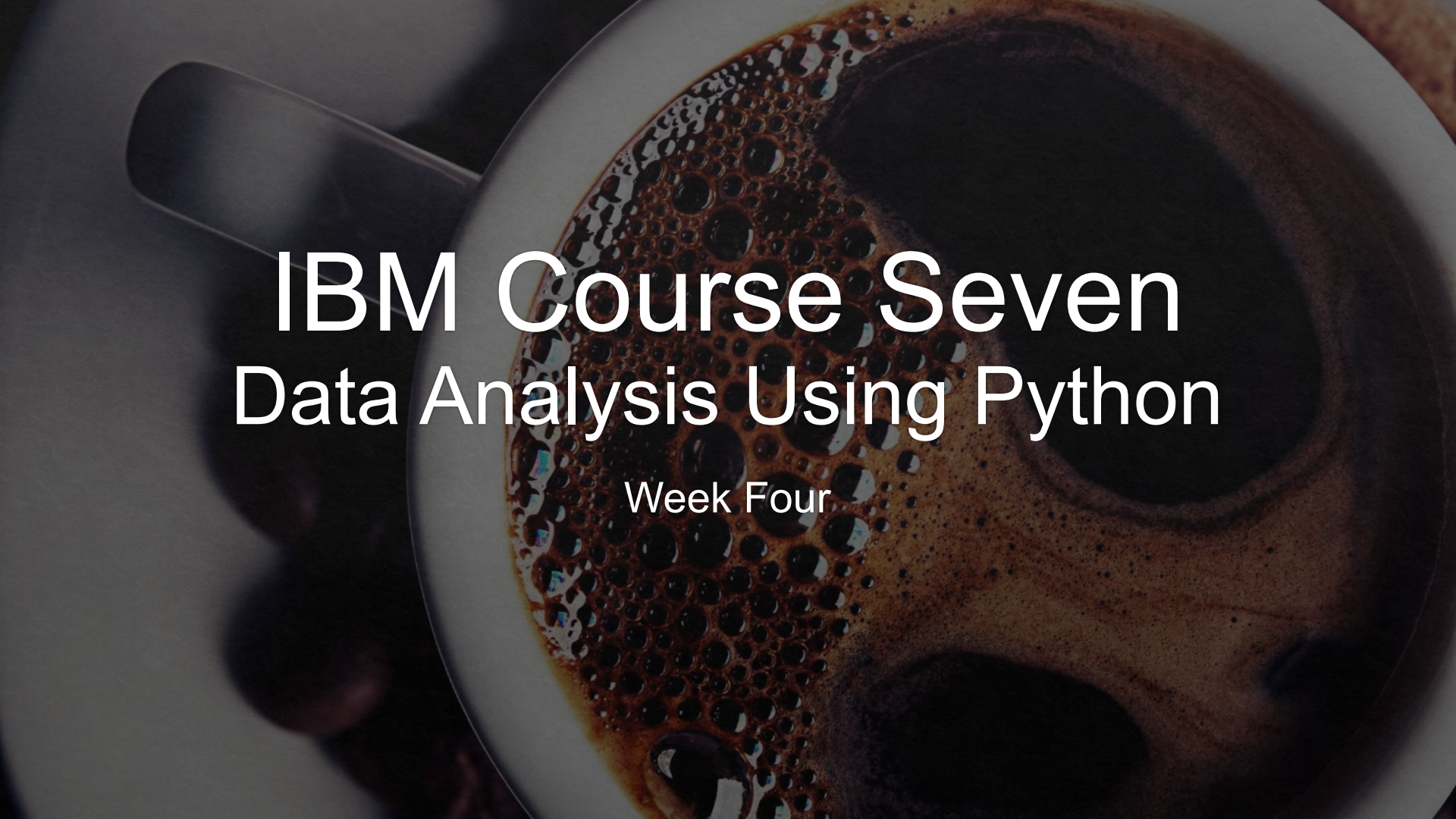


IBM Course Seven
Data Analysis Using Python

Week Four

Objectives

Describe using R^2 and MSE using in sample evaluation

Demonstrate the use of model evaluation using visualisation

Demonstrate using prediction and decision making in model creation

Demonstrate how to process linear regression in Python

Demonstrate how to progress polynomial regression and pipelines in Python

What is a model?

A model is simply a system for mapping inputs to outputs.

For example, if we want to predict house prices, we could make a model that takes in the square footage of a house and outputs a price.

A model represents a theory about a problem: there is some connection between the square footage and the price and we make a model to learn that relationship.

Models are useful because we can use them to predict the values of outputs for new data points given the inputs.

What is model development or model fitting?

Model fitting is a measure of how well a machine learning model generalizes to similar data to that on which it was trained.

A model that is well-fitted produces more accurate outcomes.

Simple Linear Regression

A statistical method that allows us to summarize and study relationships between two continuous (quantitative) variables:

One variable, denoted x, is regarded as the predictor, explanatory, or independent variable.

The other variable, denoted y, is regarded as the response, outcome, or dependent variable.

What should you know before you jump to Linear Regression?

Every Mathematical/Statistical equation is trying to establish a relationship between two variables as mentioned before. This relationship could be :

Deterministic: where the relationship between two variables is explained perfectly - eg. All scientific formulae (Hooke’s Law, Ohm’s Law, Circumference)

Statistical: Where the relationship between two variables exists, but is not explained perfectly -eg. Height and weight — as height increases, you'd expect weight to increase, but not perfectly.

Linear Regression Model tries and explains the statistical relationship!

$\hat{Y} = b_0 + b_1 x_1$

Where:

$\hat{y}$ = is the predicted response (or fitted value) for experimental unit

Y1 = denotes the observed response for experimental unit i

X1 = denotes the predictor value for experimental unit i

B0 = Intercept of the line

B1 = slope of the line

Lets try Something!

https://drive.google.com/file/d/1PBb4u8Ujfk2jwp-Emh0CJhpSjm6fTGf9/view?usp=sharing

Use the data in the above file to create a scatter plot and find the best fitting line that runs through it!

Once you have the line, cross check with your peers if they have the same line as you, discuss if it is any different!

You have 30 mins for the exercise

What did you learn from the activity?

Homework:

Find out the following:

- Mean Squared Error
- R2
- Pearson Correlation Coefficient - R

And discuss how it affects the lines that you were trying to plot in the previous activity

Multiple Linear Regression (MLR)

$$\hat{y} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \ldots\ldots\ldots\ldots + b_nx_n$$

How Python runs it?

Import Libraries:

From sklearn in linear-model import LinearRegression

Build a function first:

Lm = LinearRegression()

Lm.intercept_

Lm.coefficient_

X = df[[‘Depenedent Variable’]]

Y = df[[‘Target Variable’]]

lm.fit(X,Y)

yhat = lm.predict(x,y)

MLR

Z = df[[‘dependent var 1’ , ‘dependent var 2’ , ‘dependent var 3’]]

lm.fit(z, df[‘target variable’]

Yhat = lm.predict(z)

Polynomial Linear Regression

The goal of polynomial regression is to model a non-linear relationship between the independent and dependent variables

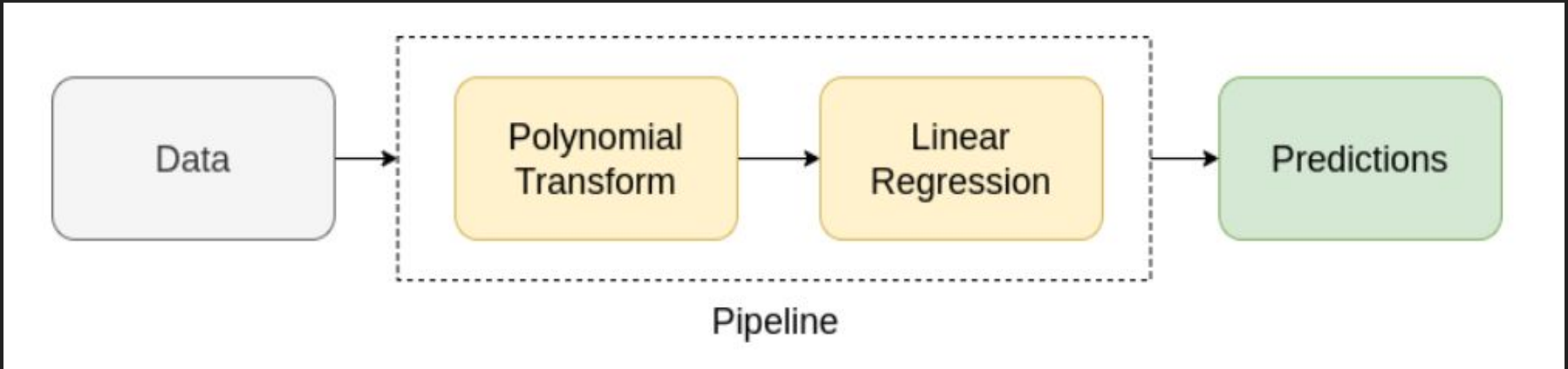
General equation:
$\hat{y} = b0 + b1x + b2x^2 + \cdots + bnx^n$ - Equation of the order “n”

```
F = np.polyfit(x,y,n)

P = poly1d(f)

print(p)
```

Polynomial Regression and Pipelines



To Do!

With the Lab, learn how to construct a pipeline!

In your breakout rooms, find examples over the internet that will help you with practicing Pipeline construction!