TRAIN-TEST SPLIT

In the next few slides, we will break down the train-test split process line-by-line.

```
1  from sklearn.model_selection import train_test_split
2  from sklearn import linear_model
3
4  features = ['TV', 'Radio', 'Newspaper']
5  X = advertising_df[features]
6  y = advertising_df['Sales']
7
8  # Split the data
9  X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30, random_state=42)
10
11 # Fit the model
12 linear_reg = linear_model.LinearRegression()
13 linear_reg_model = linear_reg.fit(X_train, y_train)
14
15 # Perform prediction with the model
16 y_pred = linear_reg_model.predict(X_test)
17
18 # Test the error
19 RMSE = mean_squared_error(y_test, y_pred, squared=False)
20
21 print(RMSE)
```

1. Separate dataset into features & outcomes
2. Split dataset into Train Set & Test Set
3. Train the model (aka “fitting” the model)
4. Generate predictions for model evaluation

TRAIN-TEST SPLIT

1 Separate dataset into features & outcomes

```
1  from sklearn.model_selection import train_test_split
2  from sklearn import linear_model
3
4  features = ['TV', 'Radio', 'Newspaper']
5  y = advertising_df['Sales']
6  X = advertising_df[features]
7
```

Slice the original Dataframe. X is a Dataframe that contains all the **features**. Y is a series with all **outcome**.

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9
...	...	...	...	...
195	38.2	3.7	13.8	7.6
196	94.2	4.9	8.1	14.0
197	177.0	9.3	6.4	14.8
198	283.6	42.0	66.2	25.5
199	232.1	8.6	8.7	18.4

Features and outcomes are fed into `train_test_split` function

2 Split dataset into Train Set & Test Set

```
8  # Split the data
9  X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30, random_state=42)
```

1 2 3 4

Data is split into:

- **70% data for `X_train` and `y_train`** (for training the model)
- **30% data for `X_test` and `y_test`** (for testing the model)

	TV	Radio	Newspaper	Sales
0	1	37.8	69.2	3
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9
...	...	...	...	...
195	2	3.7	13.8	4
196	94.2	4.9	8.1	14.0
197	177.0	6.4	14.8	
198	283.6	42.0	66.2	25.5
199	232.1	8.6	8.7	18.4

X_train y_train X_test y_test

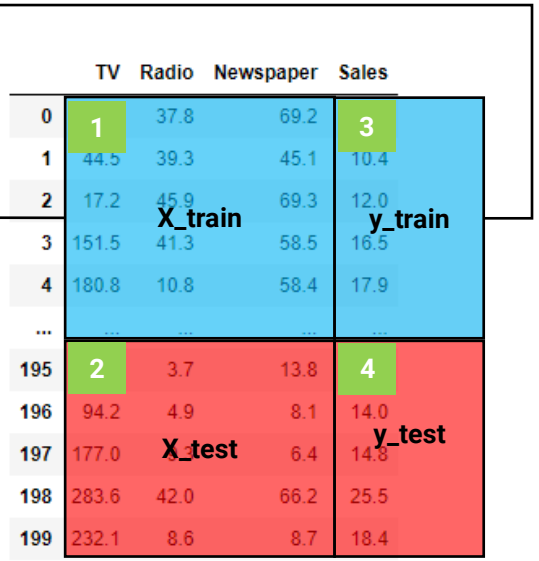
TRAIN-TEST SPLIT

3 Train the model (aka “fitting” the model)

```
11 # Fit the model
12 linear_reg = linear_model.LinearRegression()
13 linear_reg_model = linear_reg.fit(X_train, y_train)
```

Create the model object (eg Linear Regression)

Model is being passed **70% of features and outcomes** so that it can learn from the data



4 Generate predictions for model evaluation

Now we evaluate the model using **30% of the features and get it to predict the outcome of these 30%** of data (y_pred is the predicted outcome)

```
15 # Perform prediction with the model
16 y_pred = linear_reg_model.predict(X_test)
17
18 # Test the error
19 RMSE = mean_squared_error(y_test, y_pred, squared=False)
```

Now calculate the error of the **prediction** (y_pred) versus the **actual outcome** (y_test)