Linear Regression Introduction

Regression: predicted output is a continuous numerical value

Linear Regression Formula (a Type of Regression): *y* = 0 + 1 1+ 2 2 +...+ +

- *Y* is the response.
- β0 is the intercept.
- β1 is the coefficient for *x*1 (the first feature).
- βn is the coefficient for *x*n (the nth feature).
- ϵ is the error term (random irreducible error)

Note: The equation is called **linear** because the highest degree of independent variables (i.e. *x*i) is 1

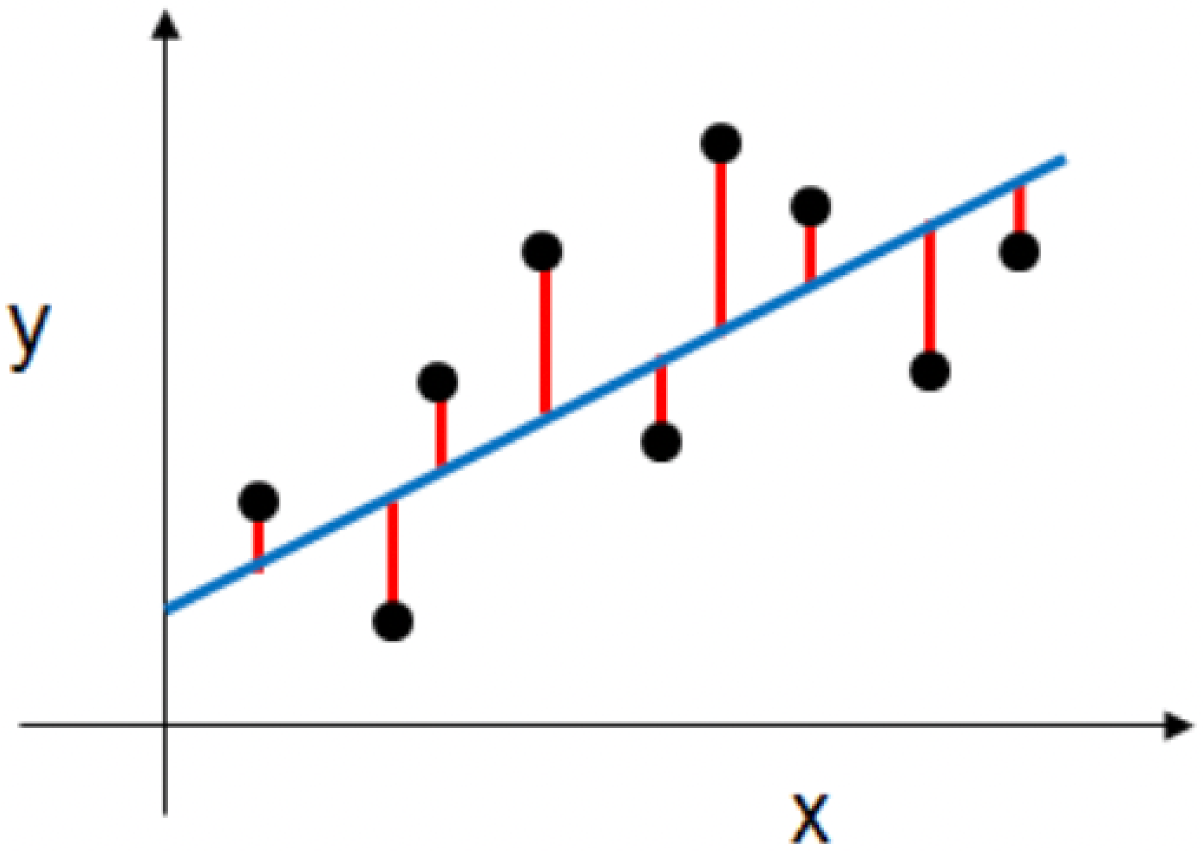
Linear Regression Introduction: Part 2

The β values in the previous equation are called the **model coefficients**:

- These values are estimated (or "learned") during the model fitting process using the **least squares criterion**.
- Specifically, we are trying to find the line (mathematically) that minimizes the **sum of squared residuals** (or "sum of squared errors").
- Once we've learned these coefficients, we can use the model to predict the response.

In the diagram to the right:

- The black dots are the **observed values** of x and y.
- The blue line is our **least squares line**.
- The red lines are the **residuals**, which are the vertical distances between the observed values and the least squares line.



Model Prediction

$$SS_{residuals} = \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

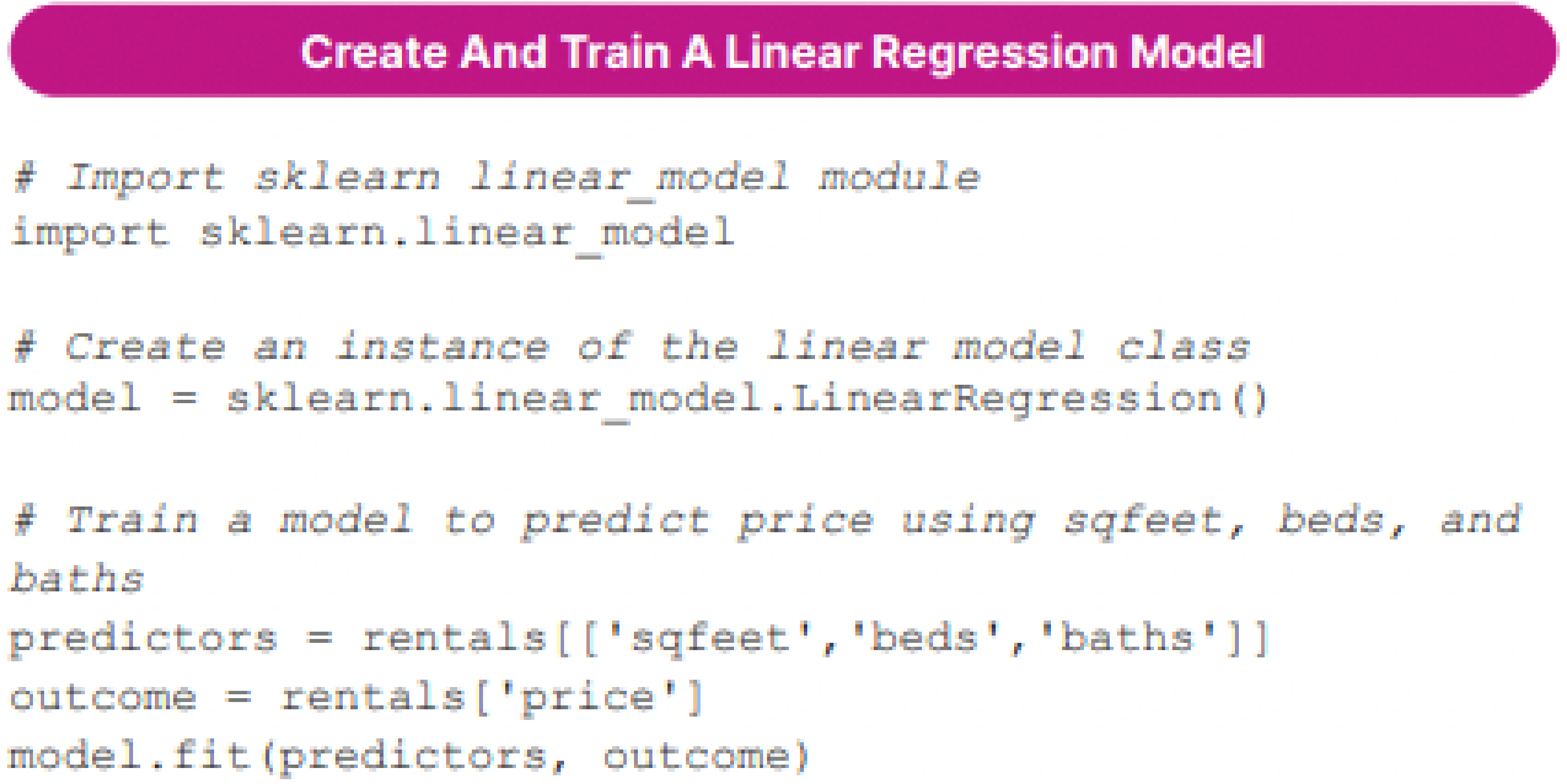
Observed Result

Linear Regression Assumptions

Let's start with some basics. Linear Regression assumes:

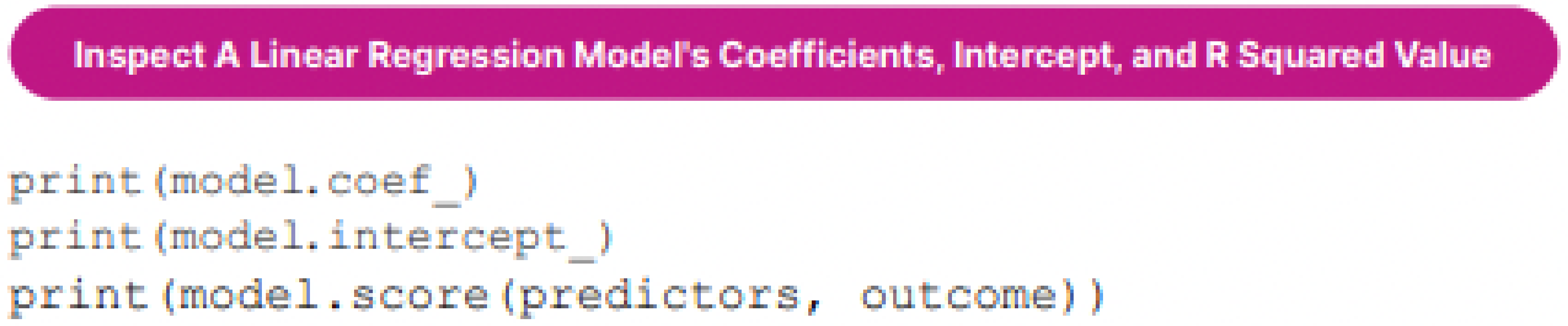
- Data is **normally distributed** (but doesn't have to be - good topic to research)
    - residuals should be normally distributed, however
    - test with **histogram**/ **Q-Q plot**/ or **Kolmogorov-Smirnov Test**
- X's significantly explain y (**low p-values**)
- X's are independent of each other (**little to no multicollinearity**)
    - test with tried and true **correlation matrix** or a **variance inflation factor (VIF)** from statsmodel
- Resulting values pass a **linear assumption**
    - use **scatter plots/pairplot** to check for **linear relationships** between **target/features**
- There should be **little to no autocorrelation** in the data (i.e. **residuals** should be independent from each other)
    - Use **Durbin-Watson test** or **scatter plots** to check
- residuals must be **equal** across the regression line (i.e. **homoscedasticity** assumption)
    - check with **lmplot/Levene's test/NCV test**, etc.

Linear Regression Syntax Basics



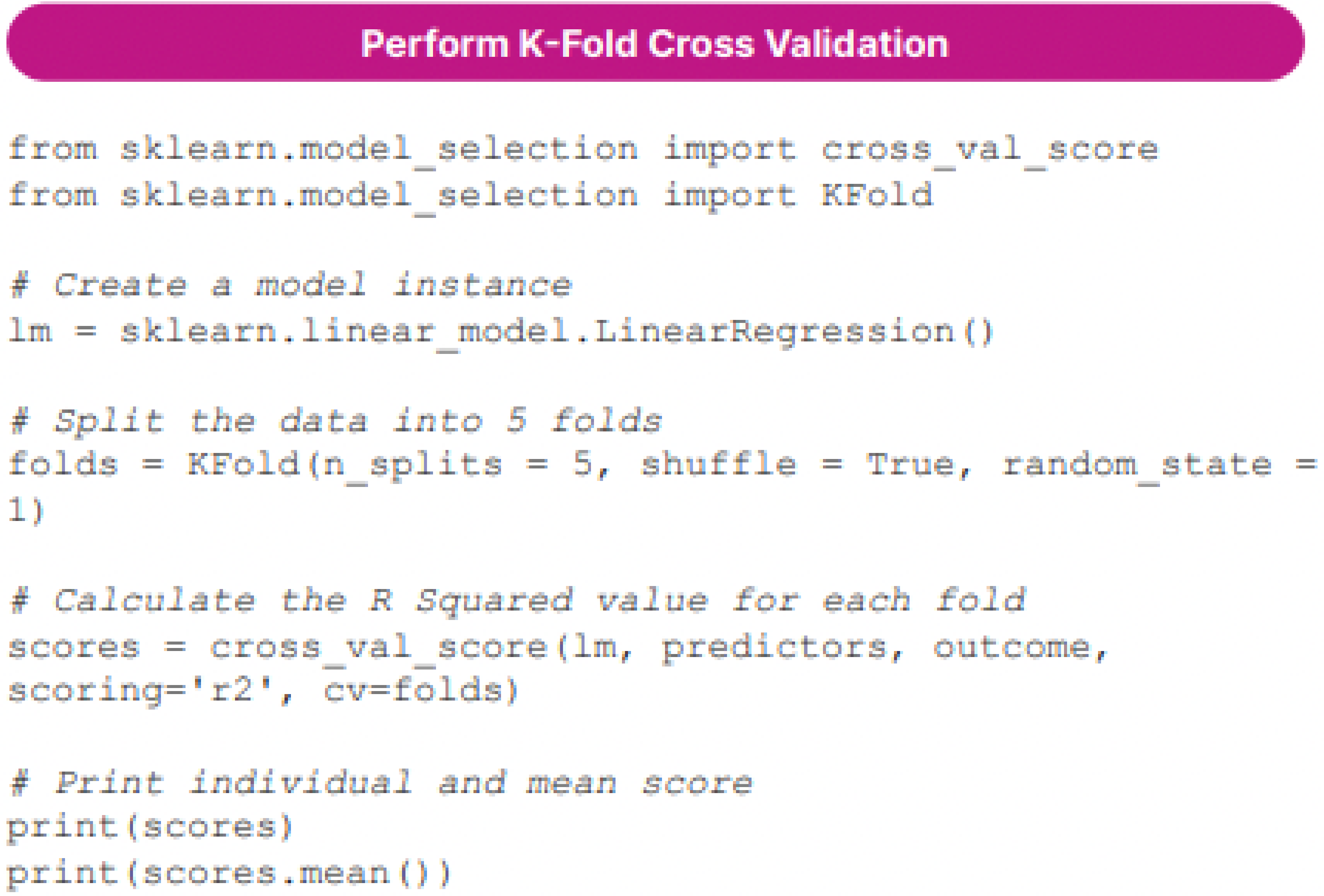
Create And Train A Linear Regression Model

```
# Import sklearn linear_model module
import sklearn.linear_model

# Create an instance of the linear model class
model = sklearn.linear_model.LinearRegression()

# Train a model to predict price using sqfeet, beds, and
baths
predictors = rentals[['sqfeet','beds','baths']]
outcome = rentals['price']
model.fit(predictors, outcome)
```



Inspect A Linear Regression Model's Coefficients, Intercept, and R Squared Value

```
print(model.coef_)
print(model.intercept_)
print(model.score(predictors, outcome))
```



Perform K-Fold Cross Validation

```
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold

# Create a model instance
lm = sklearn.linear_model.LinearRegression()

# Split the data into 5 folds
folds = KFold(n_splits = 5, shuffle = True, random_state =
1)

# Calculate the R Squared value for each fold
scores = cross_val_score(lm, predictors, outcome,
scoring='r2', cv=folds)

# Print individual and mean score
print(scores)
print(scores.mean())
```