**1. Simple Linear Regression**

**Five Key Assumptions of Linear Regression Algorithm**

1. Linear relationship (most important) 🡪plot graph to see them

* sns.lmplot(x='GDP\_bigmac', y='dollar\_price', data=df, ci=None)
* sns.jointplot(x='GDP\_bigmac', y='dollar\_price', data=df,

kind='reg', ci=None,color='orange')

1. Normal Distribution of Residuals

* If there is non-normal distribution in residuals. You can conclude that there are some unusual data points that we have to observe closely to make a good model.

1. No or less Multicollinearity

* Multicollinearity = independent variables of a given dataset are highly correlated to each other
* Use Correlation matrix to test Multicollinearity

1. No or less Autocorrelation

* In simple terms, when the value of f(x+1) is not independent of the value of f(x). This situation usually occurs in the case of stock prices, where the price of a stock is dependent on its previous one.
* Usually, most of the residual autocorrelations should fall within the 95% confidence intervals around zero. Which are located at about +/- 2-over the square root of N, where N is the dataset’s size.

1. No or less Heteroscedasticity

* Get rid of by cut out outlier

https://dataaspirant.com/assumptions-of-linear-regression-algorithm/#t-1609311604908

step 0: import library

* from sklearn.linear\_model import LinearRegression

step 1: create object/instance (create an instance from LinearRegression class)

* model = LinearRegression()

step 2: fitting model

* model.fit(df[['GDP\_bigmac']], df['dollar\_price'])

step 3: model scoring

* model.score(df[['GDP\_bigmac']], df['dollar\_price']) 🡪 R-square
* print(model.intercept\_)
* print(model.coef\_)
* y=α+βx

step 4: predict

- model.predict([[9000]]) # 2D array

train-test-split

* X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3)

Where

X = independent input dataframe, y = output series

* Use X\_train, y\_train instead in step 2

https://www.youtube.com/watch?v=PC7Zk\_u6g4w&list=PLoTScYm9O0GH\_3VrwwnQafwWQ6ibKnEtU&index=3