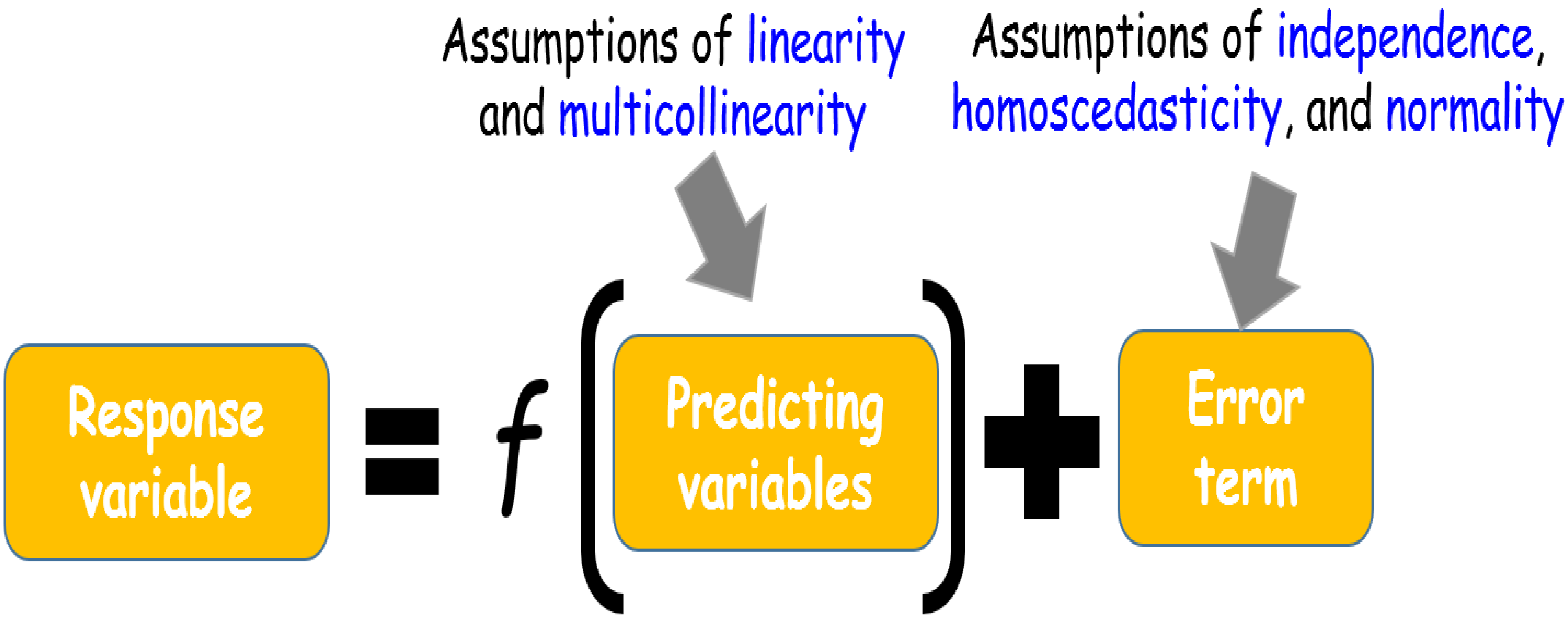
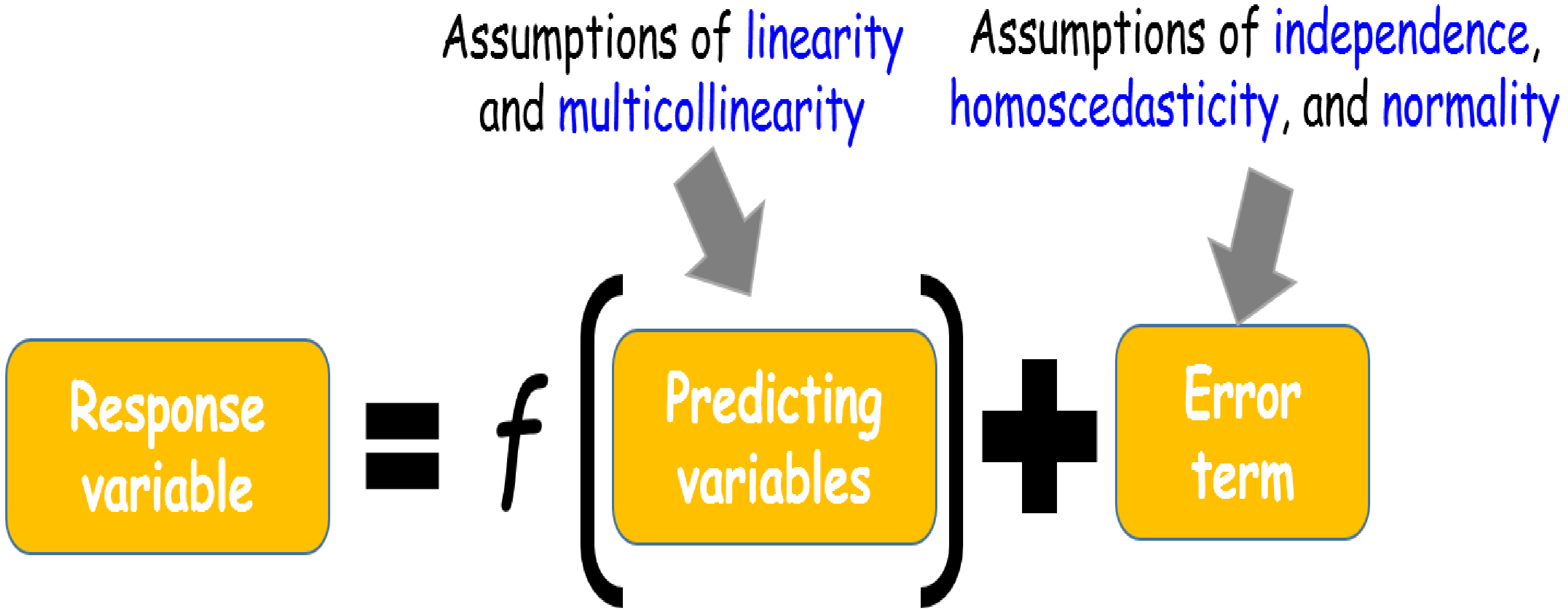
Linear regression assumptions



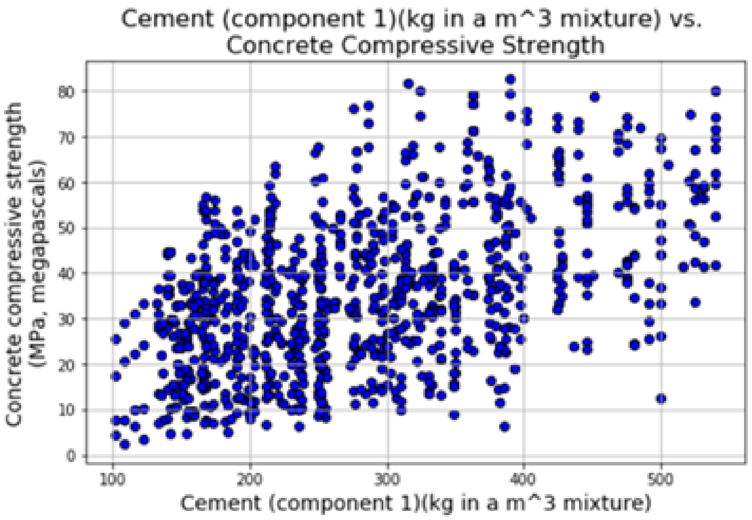
- **Linearity:** The expected value of the dependent variable is a linear function of each independent variable, holding the others fixed (note this does not restrict you to use a nonlinear transformation of the independent variables i.e. you can still model $f(x) = ax^2 + bx + c$, using both x^2 and x as predicting variables.
- **Independence:** The errors (residuals of the fitted model) are independent of each other.
- **Homoscedasticity (constant variance):** The variance of the errors is constant with respect to the predicting variables or the response.

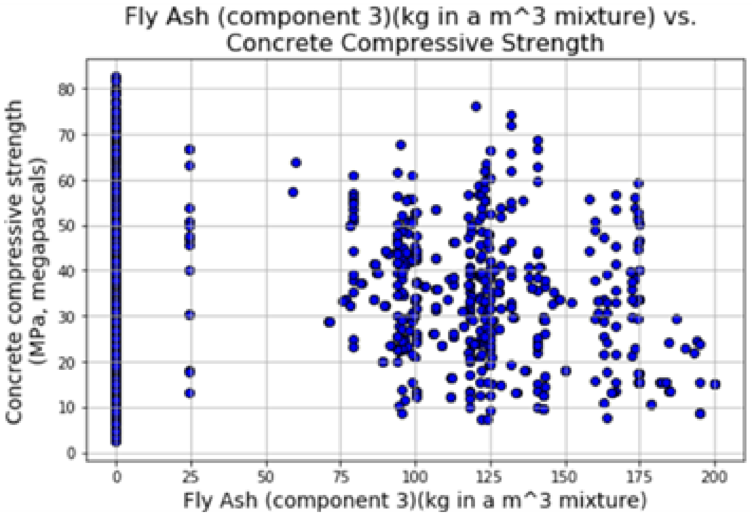
Linear regression assumptions

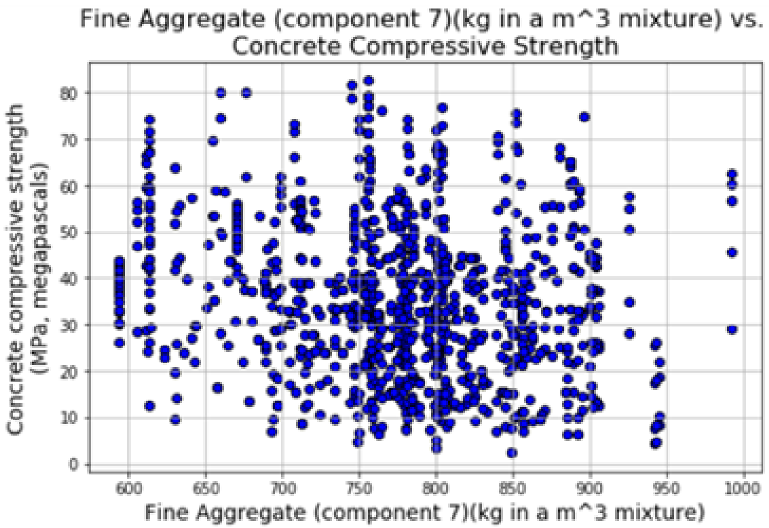


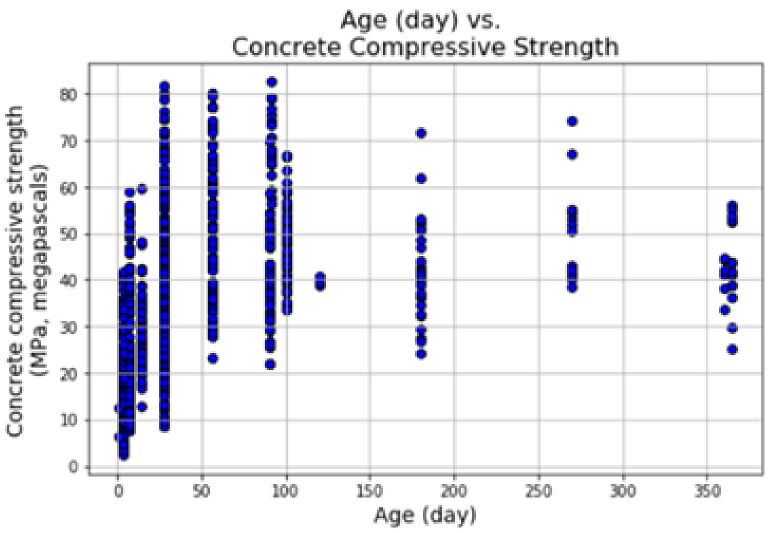
- **Normality:** errors are generated from a Normal distribution. Note, this is not a necessary condition to perform linear regression unlike the top three above. However, without this assumption being satisfied, you cannot calculate the so-called ‘confidence’ or ‘prediction’ intervals easily.
- For multiple linear regression, judging **multicollinearity** is also critical: minimal or no linear dependence between the predicting variables.
- **Outliers** can also be an issue impacting the model quality by having a disproportionate influence on the estimated model parameters.

Scatterplot of variables to check for linearity

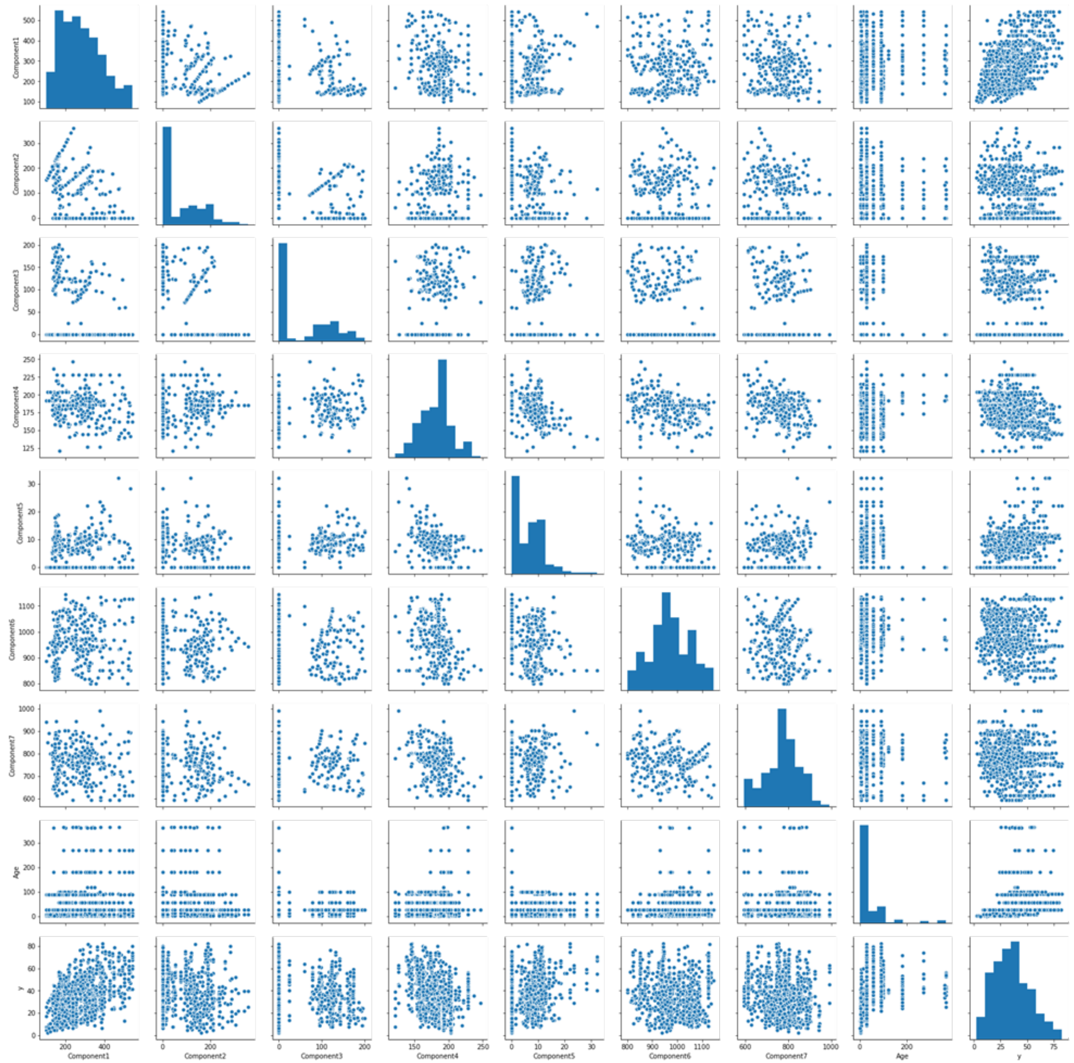




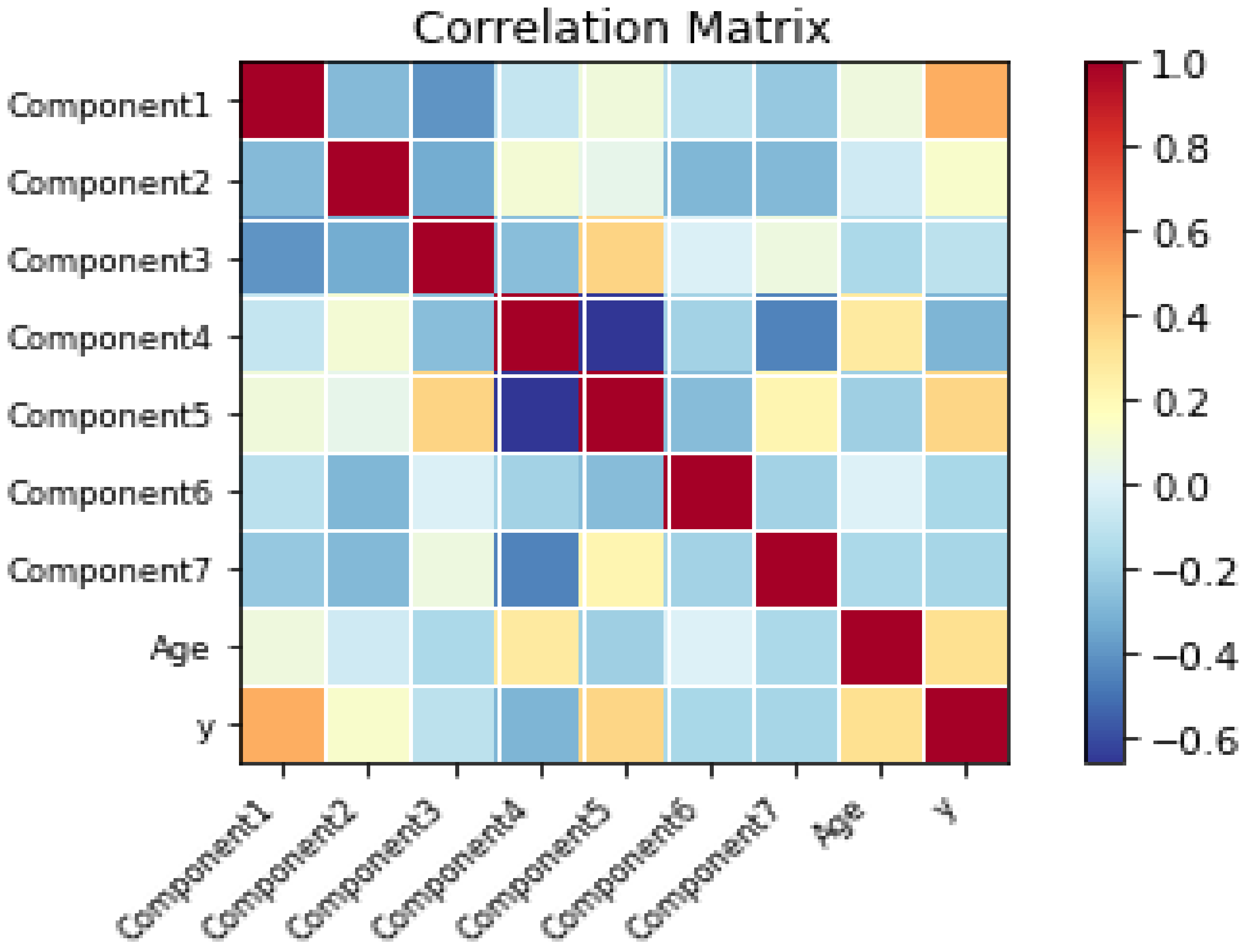




Pairwise scatter plots for checking multicollinearity



Correlation heatmap for checking multicollinearity



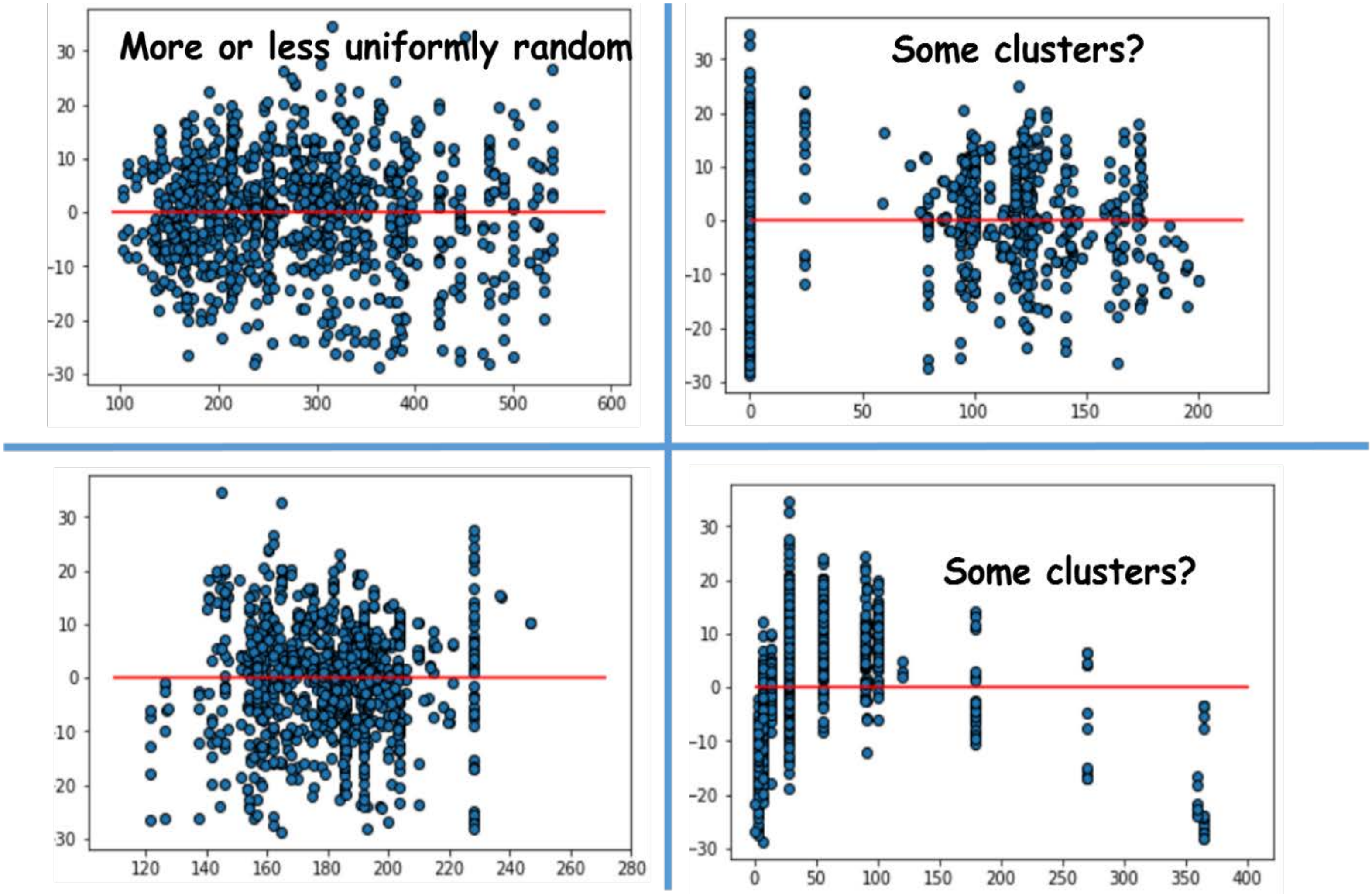
Model fitting using statsmodel.ols() function

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.615
Model:                            OLS   Adj. R-squared:                  0.612
Method:                 Least Squares   F-statistic:                     204.3
Date:                Sun, 02 Jun 2019   Prob (F-statistic):         6.76e-206
Time:                        18:25:55   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept    -23.1638     26.588     -0.871      0.384     -75.338      29.010
Component1     0.1198      0.008     14.110      0.000       0.103       0.136
Component2     0.1038      0.010     10.245      0.000       0.084       0.124
Component3     0.0879      0.013      6.988      0.000       0.063       0.113
Component4    -0.1503      0.040     -3.741      0.000      -0.229      -0.071
Component5     0.2907      0.093      3.110      0.002       0.107       0.474
Component6     0.0180      0.009      1.919      0.055      -0.000       0.036
Component7     0.0202      0.011      1.883      0.060      -0.001       0.041
Age            0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
Omnibus:                        5.379   Durbin-Watson:                   1.281
Prob(Omnibus):                  0.068   Jarque-Bera (JB):                5.305
Skew:                          -0.174   Prob(JB):                       0.0705
Kurtosis:                       3.045   Cond. No.                     1.06e+05
==============================================================================
```

Model fitting using statsmodel.ols() function

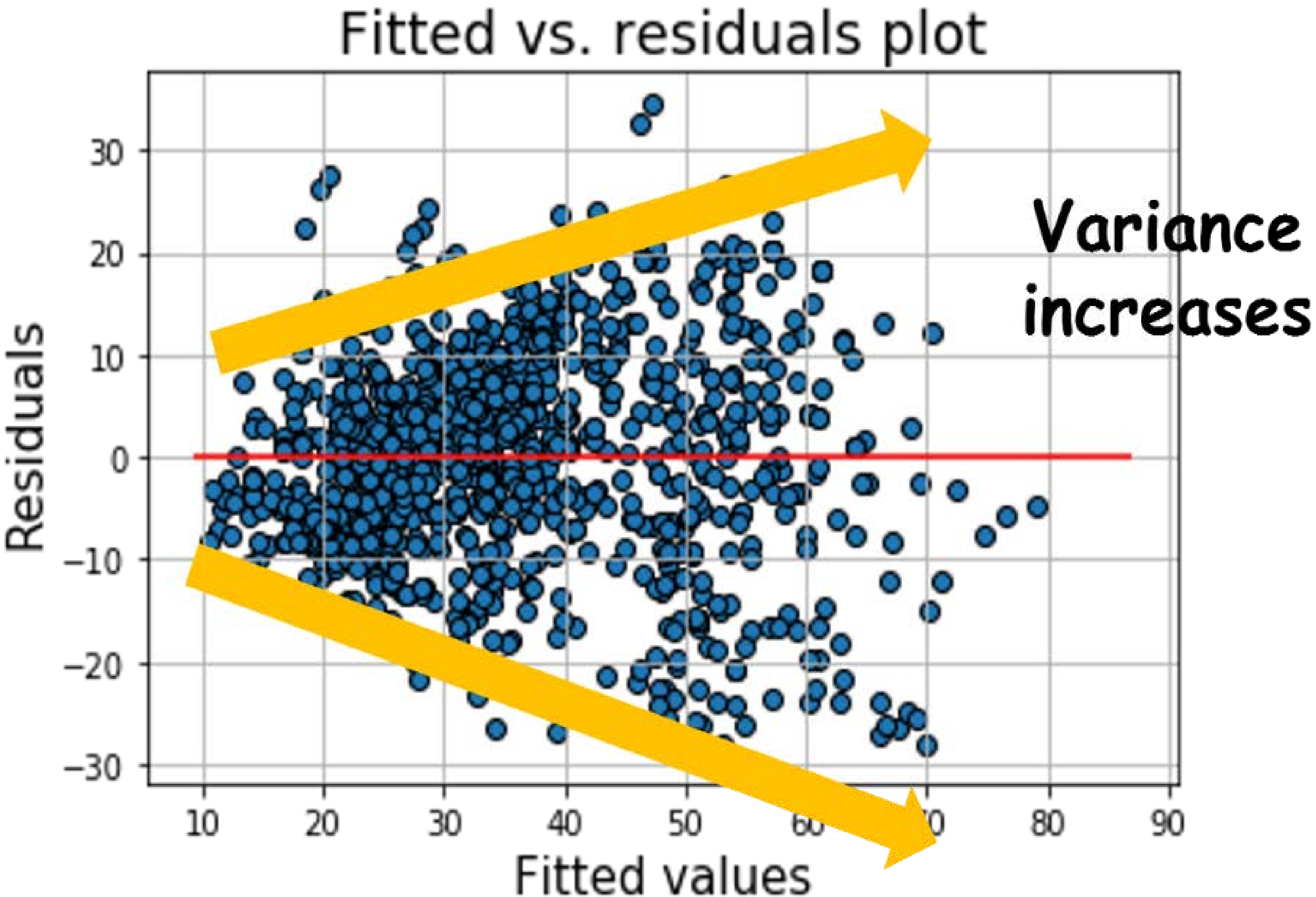
- **R-squared** is also called the coefficient of determination. It’s a statistical measure of how well the regression line fits the data.
- **Adjusted R-squared** actually adjusts the statistics based on the number of independent variables present.
- The ratio of deviation of the estimated value of a parameter from its hypothesized value to its standard error is called **t-statistic.**
- **F-statistic** is calculated as the ratio of mean squared error of the model and mean squared error of residuals.
- AIC stands for **Akaike Information Criterion**, which estimates the relative quality of statistical models for a given dataset.
- BIC stands for **Bayesian Information Criterion**, which is used as a criterion for model selection among a finite set of models. BIC is like AIC, however it adds a higher penalty for models with more parameters.

Residuals vs. predicting variables plots



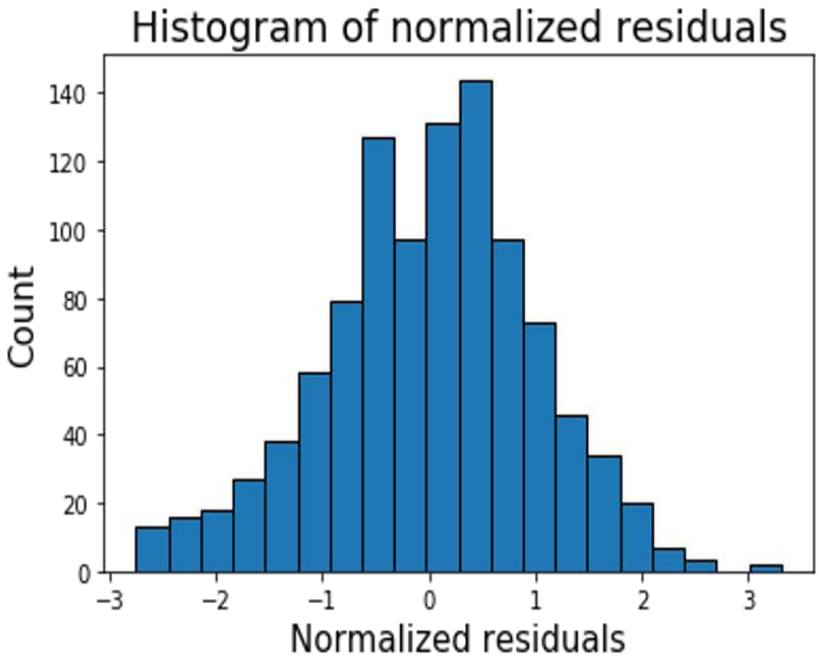
If the residuals are distributed uniformly randomly around the zero x-axes and do not form specific clusters, then the assumption holds true. In this particular problem, we observe some clusters.

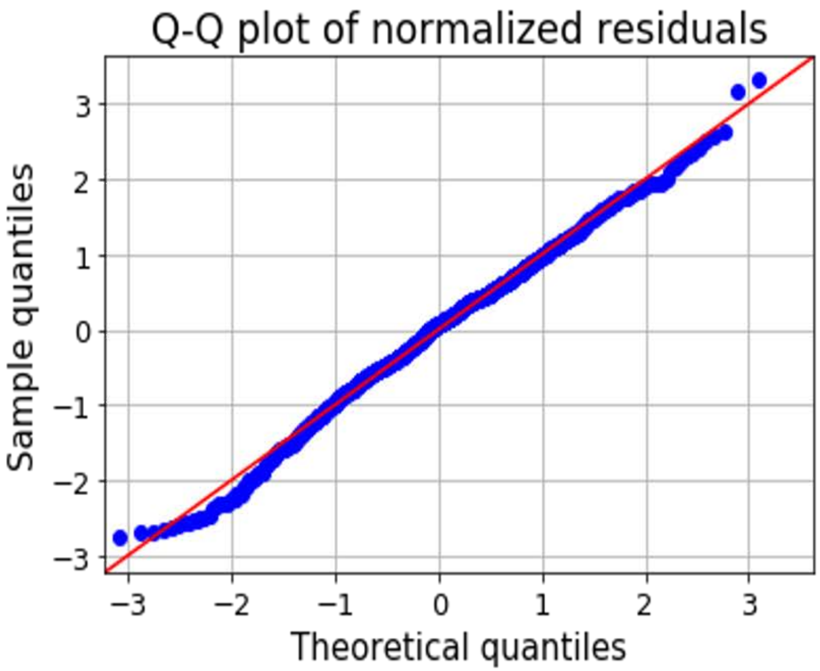
Fitted vs. residuals plot to check homoscedasticity



Histogram and Q-Q plot of normalized residuals

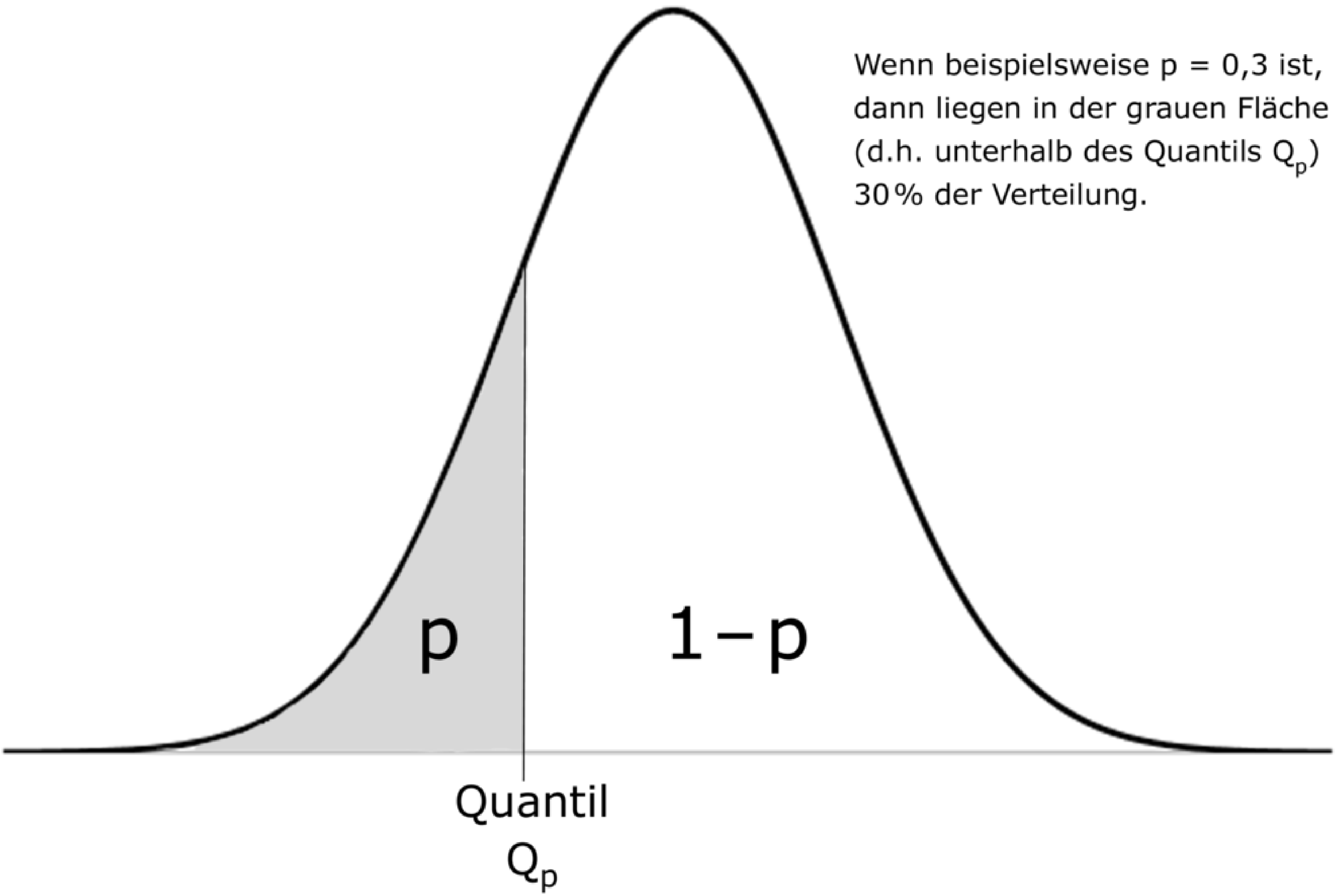
Checking for Normality





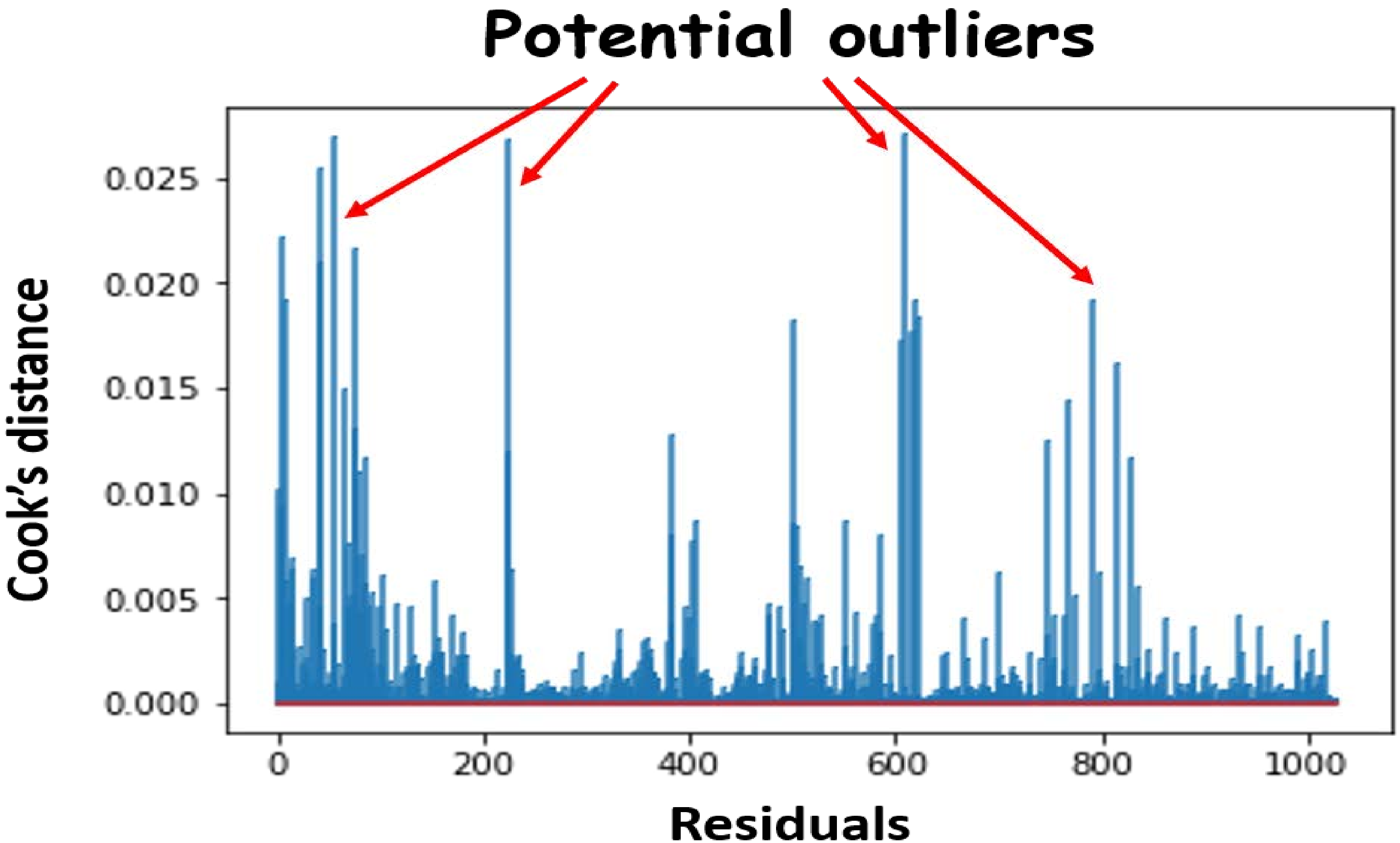
Q-Q plot should be linear for normal distribution.

Quantile



Wenn beispielsweise p = 0,3 ist, dann liegen in der grauen Fläche (d.h. unterhalb des Quantils Q_p) 30 % der Verteilung.
p
1 − p
Quantil
Q_p

Outlier detection using Cook's distance plot



Outlier detection using Cook's distance plot

Definition

Each element in the Cook's distance D is the normalized change in the fitted response values due to the deletion of an observation. The Cook's distance of observation i is

$$D_i = \frac{\sum_{j=1}^{n} \left(\hat{y}_j - \hat{y}_{j(i)}\right)^2}{p\, MSE},$$

where

- $\hat{y}_j$ is the jth fitted response value.
- $\hat{y}_{j(i)}$ is the jth fitted response value, where the fit does not include observation i.
- MSE is the mean squared error.
- p is the number of coefficients in the regression model.

Cook's distance is algebraically equivalent to the following expression:

$$D_i = \frac{r_i^2}{p\, MSE}\left(\frac{h_{ii}}{(1 - h_{ii})^2}\right),$$

where r_i is the ith residual, and h_{ii} is the ith leverage value.

$$D_i > 4/n$$

Variance influence factors

```
from statsmodels.stats.outliers_influence import variance_inflation_factor as vif
```

```
for i in range(len(df1.columns[:-1])):
    v=vif(np.matrix(df1[:-1]),i)
    print("Variance inflation factor for {}: {}".format(df.columns[i],round(v,2)))
```

```
Variance inflation factor for Cement (component 1)(kg in a m^3 mixture): 26.23
Variance inflation factor for Blast Furnace Slag (component 2)(kg in a m^3 mixture): 4.44
Variance inflation factor for Fly Ash (component 3)(kg in a m^3 mixture): 4.56
Variance inflation factor for Water  (component 4)(kg in a m^3 mixture): 92.59
Variance inflation factor for Superplasticizer (component 5)(kg in a m^3 mixture): 5.52
Variance inflation factor for Coarse Aggregate  (component 6)(kg in a m^3 mixture): 85.97
Variance inflation factor for Fine Aggregate (component 7)(kg in a m^3 mixture): 73.46
Variance inflation factor for Age (day): 2.43
```

There are few features with VIF > 10, thereby indicating significant multicollinearity

We can compute the variance influence/inflation factors for each predicting variable. It is the ratio of variance in a model with multiple terms, divided by the variance of a model with one term alone. Again, we take advantage of the special outlier influence class in statsmodels.