Linear Regression

$$y = \beta_0 + \underbrace{\beta_1 X}_{\text{relation (Slope)}}$$

$y \rightarrow$ Prediction

$\beta_0 \rightarrow$ Intercept

$X \rightarrow$ Predictor

$\beta_0 \Rightarrow$ where line intercept y axis when $X = 0$

$\beta_1 \Rightarrow \beta_1$ is calculated by minimizing SSE

Sum of Square error

$$y = \underbrace{[f(x)]}_{\text{our mode}} + e \times \text{irreductible error}$$

$f(x)$ represent some function of x which define y.

$$\beta_1 = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sum (X_i - \bar{X})^2}$$

$\bar{x}$ and $\bar{y}$ are means of y and x

Residual sum of square

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$\hat{y}_i \Rightarrow$ predicted value

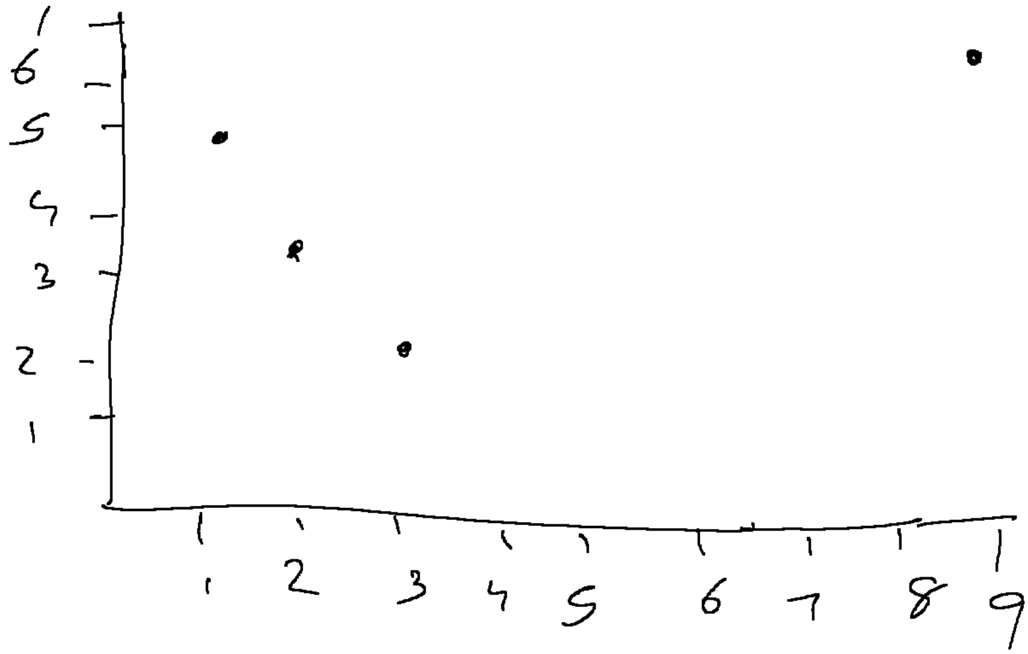
$y_i \Rightarrow$ observed value

Residual Standard error measures average error

$$RSE = \sqrt{\frac{RSS}{n-2}}$$

observed values (x_i, y_i)

y_i	x_i
5	1
4	2
7	9
3	3



Objective of project -
build own linear regression model no scikit-learn which outputs regression output of data analysis feature of excel along with a scatterplot

$$SST = \sum (y - \bar{y})^2$$

$$SSR = \sum (\hat{y} - \bar{y})^2$$

$$SSE = \sum (y - \hat{y})^2$$

$$R^2 = SSR \;/\; SST$$

$$S = \sqrt{\left(\frac{SSE}{n-2}\right)}$$

with best fit trendline and equation of line.

Multi linear regression.

$$y = X\beta + \epsilon$$

as

some relationship with y

$$y = f(X) + \epsilon$$

but here for multi linear we have matrix of X (input vector)

where β is coefficient.

Best β minimize how far $X\beta$ (prediction are from actual y)

$S(\beta)$ (Sum of Square error)

$$S(\beta) = e_1^2 + e_2^2 + \dots e_n^2$$

$$e = y - X\beta$$

Transpose of

$$S(\beta) = \begin{bmatrix} e_1 \\ e_2 \\ e_n \end{bmatrix} \begin{bmatrix} e_1 & e_2 & e_n \end{bmatrix}$$

$$= e_1^2 + e_2^2 + \dots + e_n^2$$

$$e = (y - X\beta)$$

$$S(\beta) = \overbrace{(y - X\beta)(y - X\beta)^T}^{\text{matrix}}$$

$$S(\beta) = |y - X\beta|^2$$

Our goal is to minimize Sum of error (SE) for best fit.

$$SE = S(\beta) = (y - X\beta)(y - X\beta)^T$$

$$\frac{d(S)}{d\beta} = (y - X\beta)(y - X\beta)^T$$

$$= -2X^T(y - X\beta) \rightarrow 0$$

$$\Rightarrow -2X^Ty + 2X^TX\beta = 0$$

$$\cancel{2}X^TX\beta = \cancel{2}X^Ty$$

$(X^TX) \Rightarrow$ capture relationship among predictors.

$X^T y \Rightarrow$ captures how predictor relate to target.

$$X^T X \beta = X^T y$$

$$\hat{\beta} = \frac{X^T y}{X^T X}$$

matrix of coefficient of predictor $= X^T y \, (X^T X)^{-1}$

Intercept $\qquad \hat{\beta} = 0$

Python

beta = $[\beta_1, \beta_2, \beta_3, \beta_4, \beta_5]$

y_pred = X @ beta

Goodness of fit.

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

$$R^2_{adj} = 1 - (1 - R^2)\frac{n-1}{n-p-1}$$

Mean Square error

$$MSE = \frac{SSE}{n-p-1}$$

$S = \sqrt{MSE}$ (standard error of regression)

$$\hat{\beta} = (X^T X)^{-1} X^T (X\beta + \epsilon)$$

$$\hat{\beta} = \beta + (X^T X)^{-1} X^T \varepsilon$$

$$\mathrm{Var}(\hat{\beta}) = \mathrm{Var}[(X^T X)^{-1} X^T \varepsilon]$$

$\Rightarrow$ Simplify

$$\mathrm{Var}(\hat{\beta}) = \sigma^2 (X^T X)^{-1}$$

$$\mathrm{Var}(\hat{\beta}_j) = s^2 (X^T X)^{-1}]_j$$

$$SE(\hat{\beta}_j) = s\sqrt{[(X^T X)^{-1}]_j}$$

---

future ideas

Hypothesis testing

$H_0 : \beta_j = 0$

$$t_j = \frac{\hat{\beta}_j}{SE(\hat{\beta}_j)}$$

95% confidence interval

$$\hat{\beta}_j \pm t \times SE(\hat{\beta}_j)$$

$$F = \frac{(SSR/p)}{SSE/(n-p-1)}$$