Simple Linear Regression

Tuesday, 5 March 2024 7:07 pm

→ Supervised Learning

→ 1 input column, 1 output column

→ Separate input & output column into different variables (X, y)

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=2)

from sklearn.linear_model import LinearRegression
lr = LinearRegression()

lr.fit(X_train, y_train)

lr.predict(X_test.iloc[0].values.reshape(1,1))
```

→ Input should be dataframe

m = lr.coef_

b = lr.intercept_

→ offset

→ weightage of input variable on output variable [how much the output column depends on the input column].

Dependence

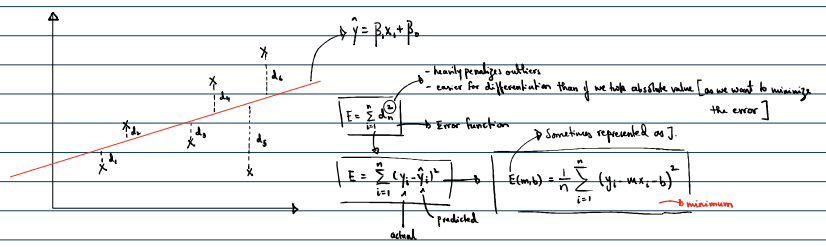
→ line of best fit: a line which passes as close as possible to the data points

→ How to find m & b for line of best fit?

① closed form solution → Ordinary Least squares (OLS)

② Non-closed form solution → Gradient descent

① $b = \bar{y} - m\bar{x}$ $\quad m = \dfrac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}$



$\frac{\partial E}{\partial b} = \frac{\partial}{\partial b}\sum_{i=1}^{n}(y_i - mx_i - b)^2 = 0$

$= \sum \frac{\partial}{\partial b}(y_i - mx_i - b)^2 = 0$

$= \sum -2(y_i - mx_i - b) = 0$

$= \sum (y_i - mx_i - b) = 0$

$\frac{\sum y_i}{n} - \frac{\sum m x_i}{n} - \frac{\sum b}{n} = 0$

$\bar{y} - m\bar{x} - \frac{mb}{n} = 0$

$b = \bar{y} - m\bar{x}$ ①

$E = \sum (y_i - mx_i - \bar{y} + m\bar{x})^2$

$\frac{\partial E}{\partial m} = \sum \frac{\partial}{\partial m}(y_i - mx_i - \bar{y} + m\bar{x})^2 = 0$

$= \sum 2(y_i - mx_i - \bar{y} + m\bar{x})(-x_i + \bar{x}) = 0$

$= \sum -2(y_i - mx_i - \bar{y} + m\bar{x})(x_i - \bar{x}) = 0$

$= \sum (y_i - mx_i - \bar{y} + m\bar{x})(x_i - \bar{x}) = 0$

$= \sum [(y - \bar{y}) - m(x_i - \bar{x})](x_i - \bar{x}) = 0$

$= \sum [(y - \bar{y})(x_i - \bar{x}) - m(x_i - \bar{x})^2] = 0$

$= \sum (y_i - \bar{y})(x_i - \bar{x}) = m\sum (x_i - \bar{x})^2$

$$m = \frac{\sum_{i=1}^{n}(y_i-\bar{y})(x_i-\bar{x})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}$$ ②

→ Regression Metrics

- MAE ①
- MSE ②
- RMSE ③
- R^2 ④
- Adjusted R^2 ⑤

① $MAE = \dfrac{\sum_{i=1}^{n}|y_i - \hat{y}|}{n}$

Pro	Con
→ same units	→ Modulus function graph isn't differentiable at 0
→ Robust outliers	

solution

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

Pro	Con
→ loss function is differentiable	→ not same units (not easy interpret)
	→ Not robust to outliers

→ points far away from the main distribution

solution

③ $RMSE = \sqrt{\dfrac{\sum_{i=1}^{n}(y_i - \hat{y})^2}{n}}$

Pro	Con
→ loss function is differentiable	→ Not robust to outliers
→ same units	→ scale dependent, alone seeing it doesn't tell us anything

solution

⑥ MAPE (Mean absolute percentage error)

$y \to \hat{y} \to y - \hat{y} \to \text{Avg}(y - \hat{y}/y)$

→ Division by zero

→ Outliers

④ R^2 - How much better regression line is compared to the mean line

→ Coefficient of determination / goodness of fit

$$R^2 = 1 - \frac{SS_R}{SS_m} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

⑤ $R^2_{adj} = 1 - \left[\dfrac{(1-R^2)(n-1)}{(n-1-k)}\right]$, where n = # of rows, k = # of input cols

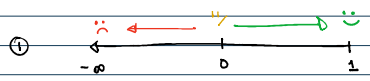
* Columns → irrelevant → R^2_{adj} ↓
* Columns → relevant → R^2_{adj} ↑

* when we have lots of cols
* there may exist cols which aren't as effective in predicting the output.

```
mean_absolute_error(
```

$$R^2 = 1 - \frac{SS_R}{SS_m} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

solution

① 

② ___% of variance in the output column is being explained by the input columns.

Sklearn.metrics — mean_absolute_error, mean_squared_error, r^2_score (y_test, y_pred)

* penalizes original R2 as you increase the # of features

Problem: if we add irrelevant columns, R^2 ↑ or — instead of ↓.

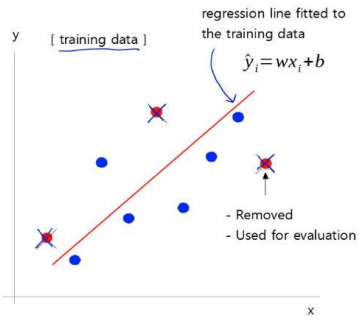
When SST is high (Data is very dispersed) there is a high chance the SSR will be high also but overall R2 will not get penalized very much.

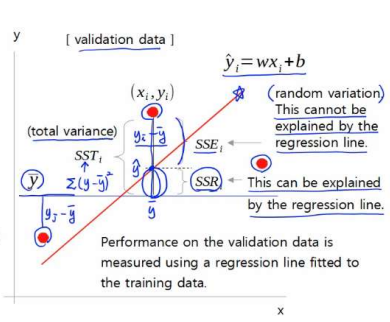
→ How much % the features are we able to exploit.

→ " " " " variance in the features are we able to utilize.

■ Performance evaluation metrics: MSE and R-squared

- The performance of the linear regression model can be measured by MSE or R^2.
- Split the data set into training and validation data, and apply the regression line fitted to the training data to the validation data to evaluate the performance of the model.
- R-squared is a measure that determines the proportion of the variance of the dependent variable that can be explained by the independent variables. R^2 = SSR / SST *





① mean squared error

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

② R^2

(SS: Sum of Squares)

$$SST = \sum_{i}^{n}(y_i - \bar{y})^2 \quad \text{(Total) variance of } y$$

$$SSE = \sum_{i}^{n}(y_i - \hat{y}_i)^2 \quad \text{(Error, Residual)}$$

$$SSR = \sum_{i}^{n}(\hat{y}_i - \bar{y})^2 \quad \text{(Regression)}$$

$$\left(R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}\right)$$

$(SST = SSE + SSR)$