2 Linear Regression

2.1 A Motivating Example

We look at life satisfaction in relation to gross domestic product (GDP) for a subset of countries (each dot is one country):

▸ Code

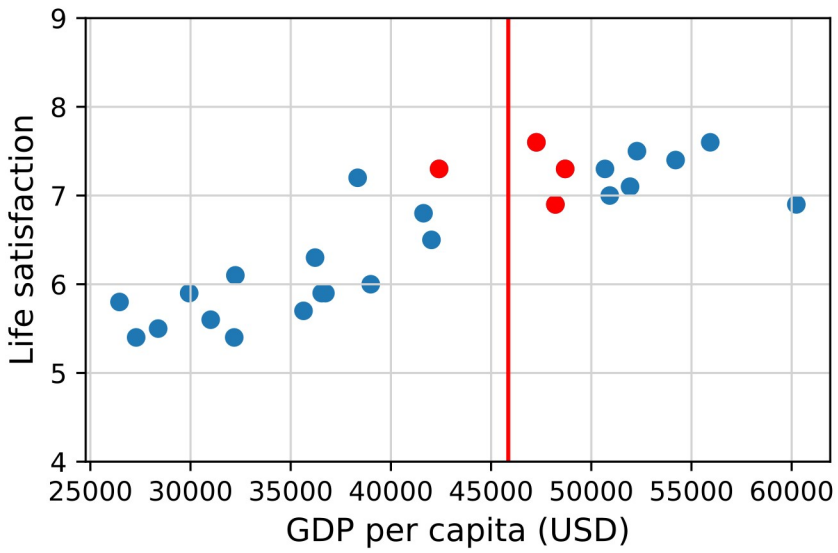


Figure 2.1: Scatterplot illustrating the relationship between money and happiness

Let's assume we do not have the life satisfaction value for certain countries, but we know their GDP. We would like to find a model which systematically leverages the available data to predict new values. This is a *regression problem*, as the target variable (life satisfaction) is a continuous numeric. It is *univariate*, as we only have one independent variable (GDP).

Example: What is the estimated life statisfaction for a country (such as Canada) with a GDP of USD 45856.62 (red line in Figure 2.1)?

2.2 Nearest Neighbor Regression

An obvious possibility would be to calculate the average from the k "closest" available values (Nearest Neighbours). The intuition here is that examples (countries) with similar values for their independent variables (GDP) should typically have similar values for their dependent variables (life happiness). For example, $k = 4$ data points are selected with a GDP closest to the GDP of Canada (red points in Figure 2.1). The predicted life satisfaction is then calculated as the average of these 4 observed output values.

Exemplary implementation using `pandas`:

▸ Code

```
Predicted life statisfaction for GDP USD 45856.62: 7.275
```

There is no theoretical specification for the optimal number k of neighbors to consider. Hence, this number

must be determined empirically. With a larger number, the flexibility of the model to map “local” patterns in the data decreases; on the other hand, the variance decreases and the prediction becomes more stable.

The nearest neighbor method is an example for *instance-based learning*. It makes predictions based on specific examples (instances) from the training data. To make predictions for a new instance, the algorithm calculates the similarity between the new instance and the instances in the training dataset. Commonly used similarity measures include euclidean distance and cosine similarity. Unlike model-based machine learning paradigms that involve generalizing from the entire training dataset, instance-based learning relies on memorizing and storing individual training instances and uses them directly during the prediction phase. Instance-based learning is also called lazy-learning, because it defers the generalization step until prediction time. Instead of creating a general model during training, it waits until a new, unseen instance needs to be classified or predicted. At that point, it looks for similar instances in the training data and uses their labels or values to make predictions. Furthermore, it does not impose any particular assumptions about the underlying data distribution, making it a flexible approach suitable for various types of data. However, it also has some disadvantages:

- Inference on a new data point generally requires the algorithm to consider explicitly all training data (to select the k closest points and compute the output)
- It is sensitive to outliers and noise
- Training samples further away than the k neighbours are ignored for inference, i.e. a significant part of the information contained in the training data is lost

2.3 Linear Regression

2.3.1 Univariate Linear Regression

In contrast to instance-based learning, in model-based learning the parameters of a general mathematical model (the *hypothesis*) are adapted in order to capture the general pattern of the data. The parametrised model can then make predictions for new input data.

One of the simplest cases of model-based learning is **univariate linear regression** (also called simple linear regression). Here, the output depends only on one input variable x, and the hypothesis is a linear combination of two parameters θ_0 and θ_1 and the input value x:

$$h_{\theta_0,\theta_1}(x) = \theta_0 + \theta_1 x$$

θ_0 represents the intercept with the y-axis and θ_1 represents the slope of a straight line.

Given fixed parameter values for θ_0 and θ_1 and a specific value for the input $x^{(m)}$, the model can predict corresponding output values:

$$\hat{y}^{(m)} = h_{\theta_0,\theta_1}(x^{(m)}) = \theta_0 + \theta_1 x^{(m)}$$

For instance, we can fit diverse univariate regression lines to the GDP-Happiness data from above. The actual values for θ_0 and θ_1 determine how well the line fits the data, as illustrated in Figure 2.2 with three different combinations of parameter values (blue, green and orange line).

▸ Code

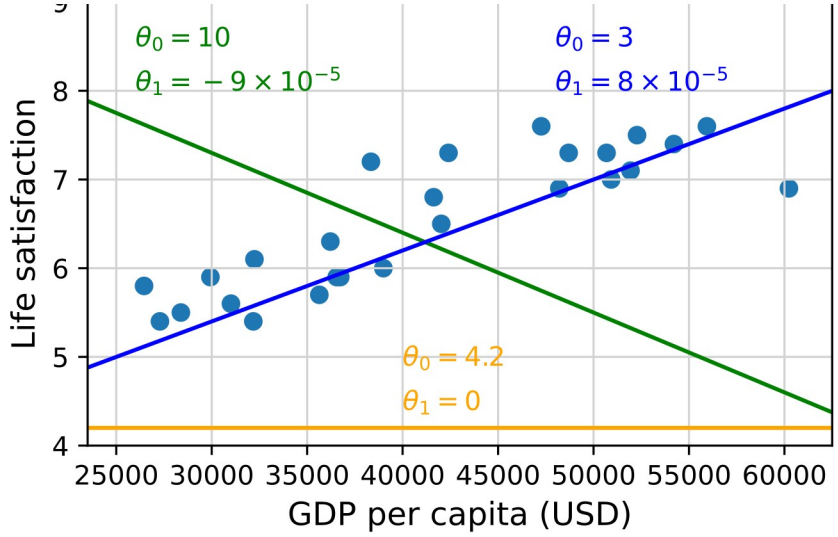


Figure 2.2: Three different parameter combinations for a linear model attempting to fit the GDP-life statisfaction data.

2.3.2 Training a Univariate Regression Model

The training of a regression model aims at finding values for the model parameters θ_0 and θ_1 that fit best the available training data. The training data consists of M samples of (input, expected output) values $(x^{(m)}, y^{(m)})$.

A crucial ingredient for the training is a **loss function**, which measures the agreement of the model predictions with the expected output values on the training data. For linear regression the *residual sum of squares* RSS (also called *sum of residuals* or *sum of squared errors*) is used:

$$\mathscr{L}_{RSS}(\theta_0, \theta_1; \{x^{(m)}, y^{(m)}\}) = \sum_{m=1}^{M} (y^{(m)} - \hat{y}^{(m)})^2 = \sum_{m=1}^{M} \epsilon_m^2$$

with $\epsilon_m = y^{(m)} - \hat{y}^{(m)}$ the *residual* of the m'th sample $(x^{(m)}, y^{(m)})$ (recall that $\hat{y}^{(m)}$ is the predicted output of our model).

The learning (training) consists of minimizing a **cost function** J, which is a function of the model parameters and depends implicitly on the training data. In the simplest case, the cost function is equal to the loss function and a normalising factor for mathematical convenience:

$$J(\theta_0, \theta_1) = \frac{1}{2M} \sum_{m=1}^{M} (y^{(m)} - \hat{y}^{(m)})^2 \tag{2.1}$$

This is also called the least squares approach. In general, however, the cost function can contain additional terms to increase the robustness of the model structure, such as regularisation terms.

Minimising the cost function results in specific values for the model parameters $\hat{\theta}_0$ and $\hat{\theta}_1$, which can then be used for inference on new data samples:

$$\hat{y}^{(m)} = h_{\hat{\theta}_0, \hat{\theta}_1}(x^{(m)}) = \hat{\theta}_0 + \hat{\theta}_1 x^{(m)}$$

2.3.2.1 Closed Form Solution for Univariate Linear Regression

In linear regression, closed-form expressions exist for the optimal model parameters. And for the univariate

case (simple linear regression) the algebraic expressions are particularly accessible, and can be derived as follows:

1. Set the partial derivatives of the cost function with respect to the model parameters to zero:

$$\frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = 0 \;\text{ and }\; \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = 0$$

2. Solve the equations for θ_0 and θ_1. Simple calculus (see derivation in Section 2.7) results in

$$\hat{\theta}_0 = \mu_y - \theta_1 \mu_x \;\text{ and }\; \hat{\theta}_1 = \frac{\sum_{m=1}^{M} (x^{(m)} - \mu_x)(y^{(m)} - \mu_y)}{\sum_{m=1}^{M} (x^{(m)} - \mu_x)^2} = \frac{\tilde{s}_{xy}}{\tilde{s}_x^2} \tag{2.2}$$

where $\mu_x = \frac{1}{M} \sum_{m=1}^{M} x^{(m)}$ and $\mu_y = \frac{1}{M} \sum_{m=1}^{M} y^{(m)}$ are the sample means. $\tilde{s}_{xy}$ is the covariance and $\tilde{s}_x^2$ is the variance of x.

2.3.2.2 The Normal Equation

Alternatively to the approach resulting in Equation 2.2, the model can be expressed in matrix form:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{e},$$

where the column vector $\boldsymbol{y}$ contains all the expected output values from the training data, $\boldsymbol{\theta}$ contains the model parameters, and $\boldsymbol{e}$ contains the residuals. The design matrix $\boldsymbol{X}$ contains the unit vector as the first column, since θ_0 is always an additive constant. The second column represents the input values from the training set.

$$\begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(M)} \end{pmatrix} = \begin{pmatrix} 1 & x^{(1)} \\ 1 & x^{(2)} \\ 1 & \vdots \\ 1 & x^{(M)} \end{pmatrix} \begin{pmatrix} \theta_0 \\ \theta_1 \end{pmatrix} + \begin{pmatrix} e^{(1)} \\ e^{(2)} \\ \vdots \\ e^{(M)} \end{pmatrix}$$

This yields a simple expression for the vector of the residuals: $\boldsymbol{e} = \boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}$

The sum of squared residuals, i.e. the loss, can be expressed as the scalar product of the residual vector with itself (for simplicity, we skip the normalization factor $\frac{1}{2M}$ in the derivation below):

$$\begin{aligned} J =& e^T e = (y - \boldsymbol{X}\boldsymbol{\theta})^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}) \\ =& \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta} \\ =& \boldsymbol{y}^T\boldsymbol{y} - 2\boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta} \end{aligned}$$

The gradient of the squared residuals (derivation in Section 2.8) yields:

$$\frac{\partial}{\partial \boldsymbol{\theta}} J = -2\boldsymbol{X}^T\boldsymbol{y} + 2\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta}$$

Equating the gradient with zero produces the so-called normal equations:

$$\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta} = \boldsymbol{X}^T\boldsymbol{y}$$

which leads to a closed form-expression for the optimal parameters:

$$\boldsymbol{\theta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} \tag{2.3}$$

$$\boldsymbol{\theta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} \tag{2.3}$$

2.3.3 Multivariate Linear Regression

Multivariate linear regression, also called multiple linear regression, is the natural extension of the simple linear regression framework to more than one predictor variables (features). For example, to predict the life satisfaction in a country, in addition to the GDP (x_1), also homicide rate (x_2), household net wealth (x_3), number of years in education (x_4) etc. can be employed. In this case, our input is a vector of N features $(x_1, x_2, \ldots x_N)$ (note that we denote the features as subscripts $_1$, $_2$ etc.), and we want to predict an output that is as close as possible to the expected output y.
One possibility would be to run four separate simple linear regressions, each of which uses a different predictor for life satisfaction. However, in this approach it is unclear how to combine the three separate predictions into a single value. Furthermore, each of the four regressions ignores the other features, which leads to incorrect estimates when features are correlated.
The $N > 1$ features (independent variables) can be combined into a single linear combination. The model then takes the following form for the m'th sample (we denote the i'th feature of the m'th sample as $x_i^{(m)}$):

$$\hat{y}^{(m)} = h_\theta(x^{(m)}) = \theta_0 x_0^{(m)} + \theta_1 x_1^{(m)} + \theta_2 x_2^{(m)} + \ldots + \theta_N x_N^{(m)} = \boldsymbol{\theta}^T \boldsymbol{X}_{m,:} \tag{2.4}$$

Here, we introduce a new artificial feature variable $x_0^{(m)} := 1$ for all $m = 1, \ldots, M$ to simplify the notation (i.e., writing the sum as a vector product).

Using the residuals $\epsilon^{(m)} = y^{(m)} - \hat{y}^{(m)}$, we can express the equation above in matrix form for all samples together:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

The rules of matrix multiplication assure that this corresponds to a system of M equations of the form of Equation 2.4. The dimensions are $\boldsymbol{X} : M \times (N+1), \boldsymbol{\theta} : (N+1) \times 1, \boldsymbol{y} : M \times 1$

For example, consider a subset of the life satisfaction data with $M = 3, N = 4$:

	GDP per capita (USD)	Homicide rate	Household net wealth	Years in education	Happiness
x_0	x_1	x_2	x_3	x_4	y
1	29932.5	4.8	70160	18	5.9
1	31007.8	1	104458	16.4	5.6
1	32181.2	1	232666	16.9	5.4

This can be expressed using the corresponding matrices:

$$\boldsymbol{X} = \begin{pmatrix} 1 & 29932.5 & 4.8 & 70160 & 18 \\ 1 & 31007.8 & 1 & 104458 & 16.4 \\ 1 & 32181.2 & 1 & 232666 & 16.9 \end{pmatrix} \quad \boldsymbol{\theta} = \begin{pmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{pmatrix} \quad \boldsymbol{y} = \begin{pmatrix} 5.9 \\ 5.6 \\ 5.4 \end{pmatrix}$$

Note that the normal equations from Section Section 2.3.2.2 can also be applied to multivariate linear regression. Hence, we can plug X and y from above into Equation 2.3 to solve for the optimal model parameters $\hat{\boldsymbol{\theta}}$.

2.4 Residual Plots

A **residual plot** is a scatter plot which has for each sample i the residual value (ϵ_i) on the vertical axis. There are different options for the values on the horizontal axis, e.g. an input variable $x_n^{(m)}$ or the predicted value $y^{(m)}$.

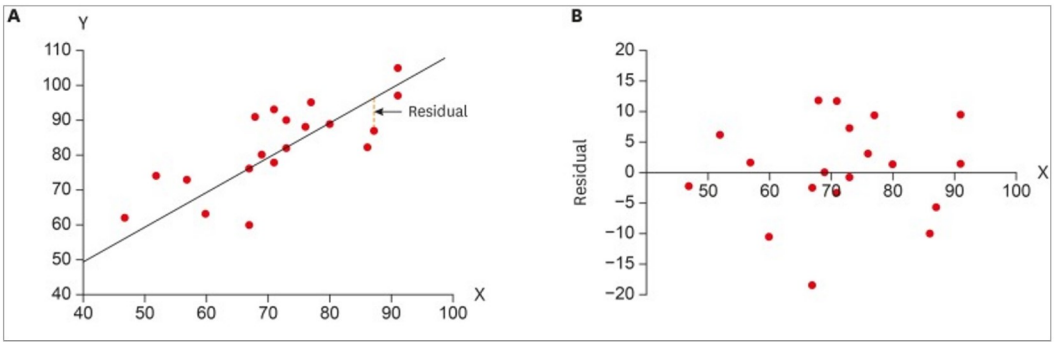


Figure 2.3: Residuals Plot: Figure A shows the original data with input X and output Y; Figure B shows the residual plot, with values X on the horizontal axis.)

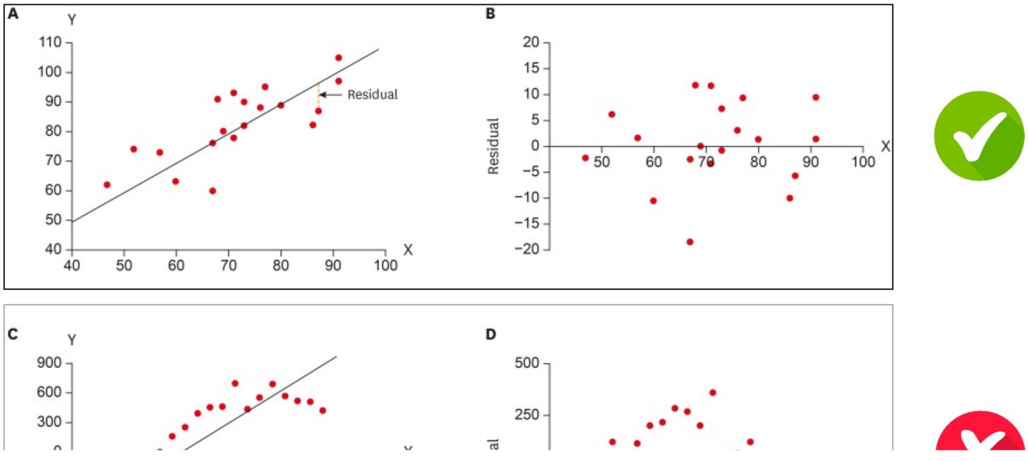
2.5 Basic Assumptions in Linear Regression

Applying linear regression models is only possible if several basic assumptions are valid for the data. The fundamental assumptions are: Lineraity, Independence, Normality, and Homoscedasticity. If these assumptions do not hold for the data, then linear regression cannot yield reasonable results.

In the following, we explain what these assumptions are, and how they can be verified visually by means of the residual plots.

1. Linearity: The relationship between X and Y is linear.

The fundamental assumtion to apply linear regression is that there exists a linear relationship between the parameters x and the output y. Figure Figure 2.4 shows such an example. In this case, the dots are randomly scattered along the regression line. Also, most observations should lie near the regression line, while observations far away from the line are less frequent.



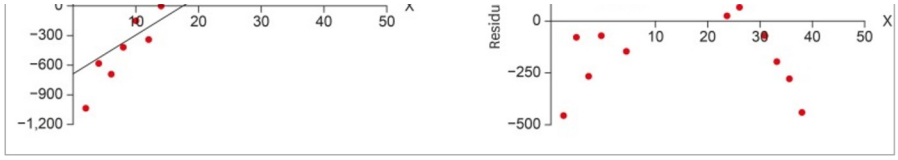


Figure 2.4: Top figure shows a linear relation, while bottom figure does not (left: data samples, right: residual plot)

2. Independence: The residuals are independent of each other.

Independence assumes that the residual (the difference between the prediction and the actual outcome for one observation) does not affect or correlate with the residuals of other observations. This might be violated, for instance, if we collect data from an individual at different points in time. Figure Figure 2.5 shows such an example, where we see a repeating pattern (autocorrelation) of the residuals over time.

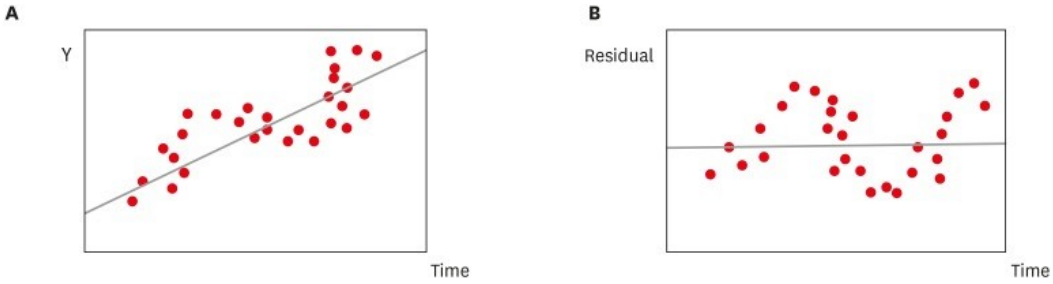


Figure 2.5: Scatter plot over time, with autocorrelation pattern of the data (left: data samples, right: residual plot)

3. Normality: The expected output values are normally distributed.

As mentioned in item “Linearity”, we assume that the there is a linear relation between input and output values. Usually, there will be some noise within samples for the same input: For instance, assume that we measure the temperature (x) and the amount of ice-cream sold on each day (y). Then even for the same temperature on different days, you would expect different amounts of ice-cream sold. However, we assume that the output values follow a normal distribution: Let μ_x be the mean of all outputs for a give input x (e.g. for a specific temperature, the mean of all amounts of ice-cream sold on days with that temperature). Then we assume that for each input x, output values close to mean μ_x are more likely than output values further away from the mean, and that these values follow a normal distribution.

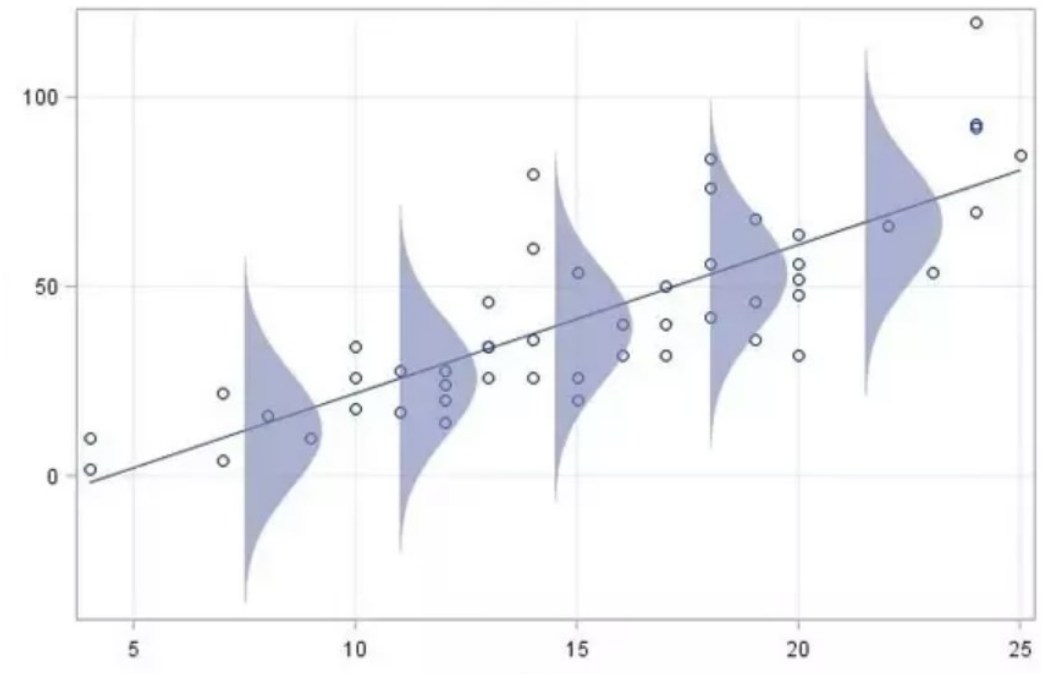


Figure 2.6: Normal distribution of outputs for each input value

4. Homoscedasticity (equality of variance): The variance of the residual is the same for any value of X.

In addition to normality, we also assume that the degree of variability of residuals (i.e. distance between straight line and true output values) is equal across different input values. Bottom example of Figure Figure 2.7 shows a setting where homoscedasticity is violated, since variance increases for larger input values.

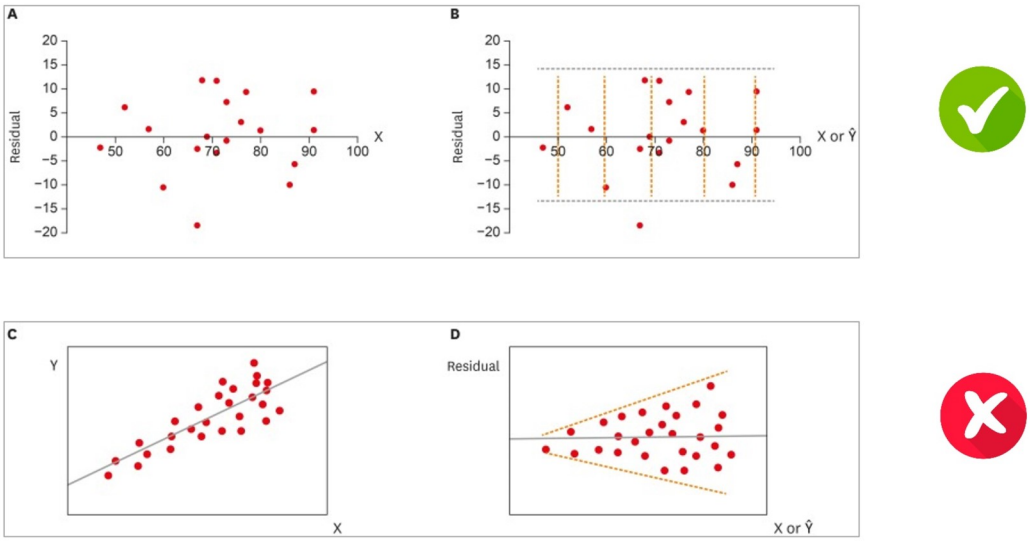


Figure 2.7: Top figure shows a setting with homoscedasdicity, while bottom figure does not (left: data samples, right: residual plot)

2.6 Evaluating Regression Models

Once a regression model is trained, we want to evaluate its quality. For this, we assess its generalisability on an independent test set (see Section 1.6.2). This test set consists of pairs of inputs $x^{(i)}$ and expected output values $y^{(i)}$. The corresponding predictions of the model based on the inputs are denoted with $\hat{y}^{(i)}$. There are several metrics that can be used to quantify how well the expected and predicted output values correspond:

- Mean Absolute Error: $MAE = \frac{\sum_{i=1}^{I} |y^{(i)} - \hat{y}^{(i)}|}{I}$
- Mean Squared Error: $MSE = \frac{\sum_{i=1}^{I} (y^{(i)} - \hat{y}^{(i)})^2}{I}$
- Root Mean Squared Deviation: $RMSD = \sqrt{MSE} = \sqrt{\frac{\sum_{i=1}^{I} (y^{(i)} - \hat{y}^{(i)})^2}{I}}$

with I number of samples in the independent test set.

Note that these errors are similar to the loss and cost function that we used when *training* the regression model. However, during training we used the loss on the training data to *find* the proper values for parameters θ_i, whereas here we assume fixed parameters θ_i and we only *evaluate* how well the model performs on the test data.

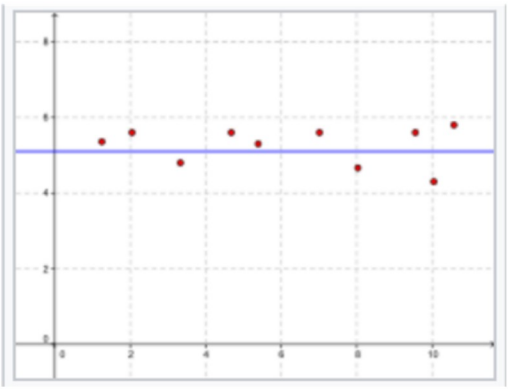
2.6.1 Coefficient of Determination

One way to measure the quality of a regression model is the **coefficient of determination** R. It measures the fraction of the variance of the data which can be explained by the model:

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

where $SS_{\text{res}} = \sum_i (y^{(i)} - \hat{y}^{(i)})^2 = \sum_i {e^{(i)}}^2$ is the sum of squares of residuals (also called residual sum of squares), which quantifies the variance that can not be explained by the model, and $SS_{\text{tot}} = \sum_i (y^{(i)} - \mu_y)^2$ is the total sum of squared distance between each expected output $y^{(i)}$ and the mean value over all y. Note that SS_{tot} is proportional to the variance of the data.

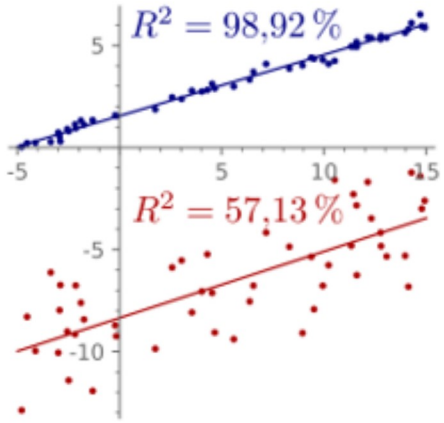
A trivial baseline model, which always predicts μ_y, will have $R^2 = 0$:



Baseline model which always predicts the mean (blue line) of the training data (red dots)

Models that have worse predictions than this baseline (e.g., $\hat{y}$ is set to a constant value that is not equal μ_y) will have a negative R^2.

In the best case, the modeled values exactly match the observed values, which results in $SS_{\text{res}} = 0$ and $R^2 = 1$. The smaller the agreement, the smaller R^2 becomes. The following figure shows two separate datasets (red and blue), and a regression line for each dataset.



High R^2 value for small SS_{res} (blue) and smaller R^2 in a situation with larger SS_{res} (red)

2.7 Derivation of θ_0 and θ_1 for Univariate Linear Regression

$$J = \frac{1}{2M} \sum_{m=1}^{M} (y^{(m)} - \theta_0 - \theta_1 x^{(m)})^2$$

Starting with the partial derivative with respect to θ_0:

$$\frac{\partial}{\partial \theta_0} J = 2 \cdot (-1) \frac{1}{2M} \sum_{m=1}^{M} (y^{(m)} - \theta_0 - \theta_1 x^{(m)})$$

and setting it to zero, since we are looking for an extremum:

$$\begin{aligned}
0 &= \sum_{m=1}^{M} y^{(m)} - \sum_{m=1}^{M} \theta_0 - \theta_1 \sum_{m=1}^{M} x^{(m)} \\
&\text{with } \mu_y = \frac{1}{M} \sum_{m=1}^{M} y^{(m)} \text{ and } \mu_x = \frac{1}{M} \sum_{m=1}^{M} x^{(m)} \\
0 &= M\mu_y - M\theta_0 - \theta_1 M \mu_x \\
0 &= \mu_y - \theta_0 - \theta_1 \mu_x
\end{aligned}$$

it turns out, the expression for θ_0 depends on θ_1:

$$\theta_0 = \mu_y - \theta_1 \mu_x$$

We can then insert this back into the formula for J:

$$\begin{aligned}
J &= \frac{1}{2M} \sum_{m=1}^{M} (y^{(m)} - \mu_y + \theta_1 \mu_x - \theta_1 x^{(m)})^2 \\
&= \frac{1}{2M} \sum_{m=1}^{M} [(y^{(m)} - \mu_y) - \theta_1 (x^{(m)} - \mu_x)]^2 \\
&= \frac{1}{2M} \sum_{m=1}^{M} [(y^{(m)} - \mu_y)^2 - 2(y^{(m)} - \mu_y)\theta_1 (x^{(m)} - \mu_x) + \theta_1^2 (x^{(m)} - \mu_x)^2],
\end{aligned}$$

thus allowing us to calculate the partial derivative with respect to θ_1:

$$\begin{aligned}
\frac{\partial}{\partial \theta_1} J &= \frac{1}{2M} \sum_{m=1}^{M} [-2(y^{(m)} - \mu_y)(x^{(m)} - \mu_x) + 2\theta_1 (x^{(m)} - \mu_x)^2] \\
\frac{\partial}{\partial \theta_1} J &= \frac{1}{M} \sum_{m=1}^{M} [-(y^{(m)} - \mu_y)(x^{(m)} - \mu_x) + \theta_1 (x^{(m)} - \mu_x)^2],
\end{aligned}$$

Setting to zero

$$0 = \sum_{m=1}^{M} \theta_1 (x^{(m)} - \mu_x)^2 - \sum_{m=1}^{M} (y^{(m)} - \mu_y)(x^{(m)} - \mu_x)$$

and rearranging the resulting expression produces

$$\theta_1 = \frac{\sum_{m=1}^{M} (y^{(m)} - \mu_y)(x^{(m)} - \mu_x)}{\sum_{m=1}^{M} (x^{(m)} - \mu_x)^2} = \frac{\tilde{s}_{xy}}{\tilde{s}_x^2},$$

where $\tilde{s}_{xy}$ is the covariance of the input and the target values. $\tilde{s}_x^2$ is the variance of the dependent variable.

2.8 Derivation of the Gradient of the Squared Residuals in Matrix Form

$$J = \boldsymbol{y}^T\boldsymbol{y} - 2\boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta}$$

Calculating the partial derivatives separately for the additive terms:

$$\frac{\partial}{\partial\boldsymbol{\theta}}\boldsymbol{y}^T\boldsymbol{y} = 0$$

$$\frac{\partial}{\partial\boldsymbol{\theta}}\left(-2\boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{y}\right) = -2\boldsymbol{X}^T\boldsymbol{y}$$

The third terms makes use of matrix calculus (see here for an overview of the most important rules):

$$\frac{\partial}{\partial\boldsymbol{b}}\boldsymbol{b}^T\boldsymbol{A}\boldsymbol{b} = 2\boldsymbol{A}\boldsymbol{b}$$

for a symmetric matrix $\boldsymbol{A} : D \times D$ and vector $\boldsymbol{b} : D \times 1$, under the condition that $\boldsymbol{A}$ is not a function of $\boldsymbol{b}$. Setting $\boldsymbol{A} = \boldsymbol{X}^T\boldsymbol{X}$ and $\boldsymbol{b} = \boldsymbol{\theta}$:

$$\frac{\partial}{\partial\boldsymbol{\theta}}\boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta} = 2\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta}$$

Bringing all terms together:

$$\frac{\partial}{\partial\boldsymbol{\theta}}J = -2\boldsymbol{X}^T\boldsymbol{y} + 2\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta}$$