Life Sciences and
Facility Management

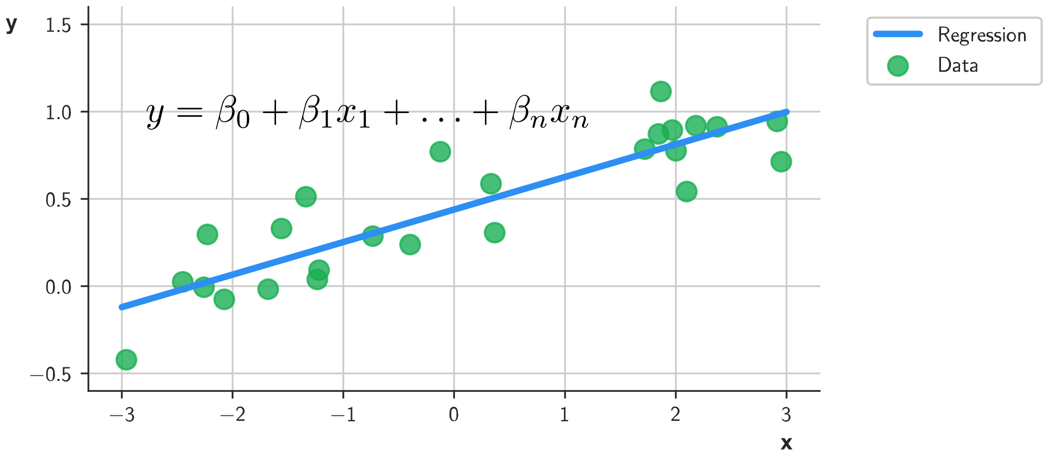
Institute of
Computational Life Sciences

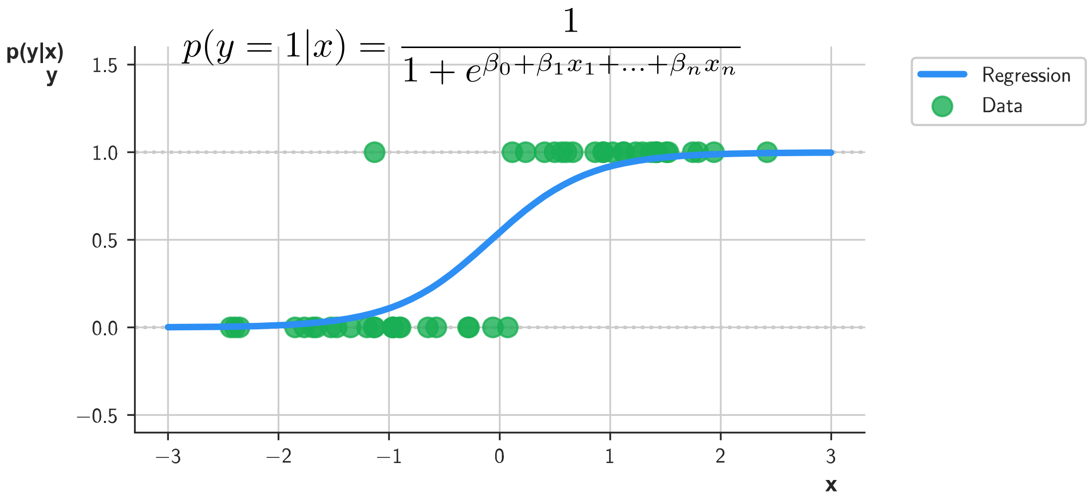
Roundup regression

Machine Learning

Norman Juchler

Regression: Summary of key points





Linear (or non-linear) regression:

- Used for regression tasks where the output is a continuous variable.
- Models the relationship between features and output with a linear or non-linear equation.
- Objective is to minimize the difference between the predicted and actual values (e.g., by minimizing mean squared error).

Logistic regression:

- Used for binary classification tasks where the output is categorical.
- Models the probability of an instance belonging to a particular class, using the sigmoid function to convert linear outputs into probabilities.
- Predictions are made by setting a threshold (usually 0.5) on the output probability.

Interpretation of model parameters

- **Linear regression**:
  - $y = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n$
  - Each coefficient β_i represents the change in the target variable for a one-unit increase in the corresponding feature
- **Logistic regression**:
  - $p(y=1|x) = \frac{1}{1+e^{\beta_0+\beta_1 x_1+\ldots+\beta_n x_n}}$
  - For each one-unit increase in a feature, the log-odds of the positive class increase by the value of the feature's weight β_i.

What does it mean if $\beta_i = 0$?

- No relationship: The feature does not contribute to explaining the variation in the target variable or the log odds.
- Feature exclusion: The model has effectively "excluded" the feature as uninformative*
- (We can even apply a t-test to examine if $\beta_i = 0$)

*This applies to regularized regression, see below.

Logistic regression: Interpretation

- The logistic regression model consists of two main components:
  - The linear model, which is a linear combination of the input features
  - The sigmoid, which maps the linear model's output to a probability value between 0 and 1

$$p(y=1|x) = \sigma(x) = \frac{1}{1+e^{-z(x)}}, \qquad \text{with } z = \beta_0 + \beta_1 \cdot x_1 + ... + \beta_n \cdot x_n$$

- We can rewrite this expression as (with $p = p(y=1|x)$)

$$\log \frac{p}{1-p} = z(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n$$

- Interpretation:

  In logistic regression, we model the so-called log-odds
  (i.e., the logarithm of the odds ratio) using a linear regression!

$$\begin{aligned} p &= \frac{1}{1+e^{-z(x)}} \\ \frac{1}{p} &= 1 + e^{-z(x)} \\ \frac{1}{p} - 1 &= e^{-z(x)} \\ \log \frac{p}{1-p} &= \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n \end{aligned}$$

Logistic regression: Learning check

Which of the following statements are true?

- ☐ a. The logistic function is the inverse of the logit function
- ☐ b. The logistic function maps the linear model's output to a probability value between 0 and 1
- ☐ c. The logarithm of the odds (*log-odds*) is modeled as a linear combination of the input features
- ☐ d. The ratio $\frac{p}{(1-p)}$ is called the odds: Probability of the event occurring divided by the probability of the event not occurring

Regularized regression

- Idea: Prevent overfitting by penalizing large coefficients, encouraging simpler models that generalize better.

- How? Modify the loss function!

Regularized loss = Original loss + Penalty term on coefficients

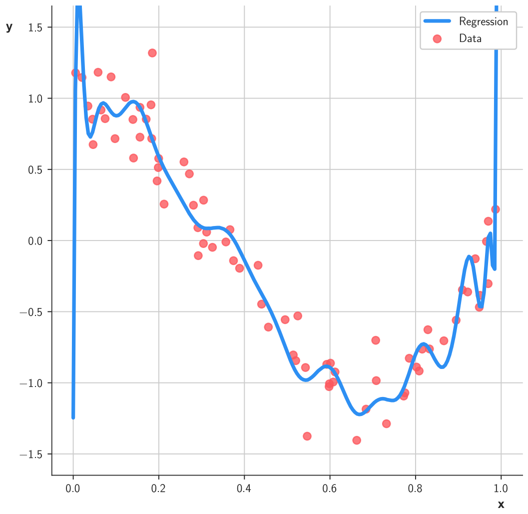
- Example: (Ridge regression)

$$\mathcal{L}(\beta|X, y) = MSE(\beta|X, y) + \lambda \sum_{j=1}^{p} \beta_j^2$$

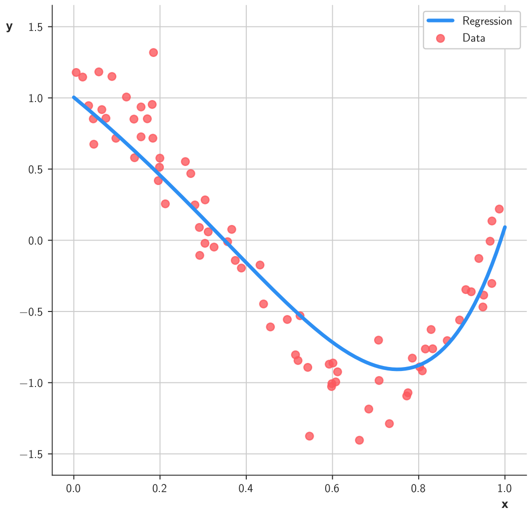
Recap:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i|\beta))^2$$

- This introduces a **hyperparameter λ** (or α in sklearn):
  - Controls the strength of regularization
  - Larger α increases penalty, leading to smaller coefficients



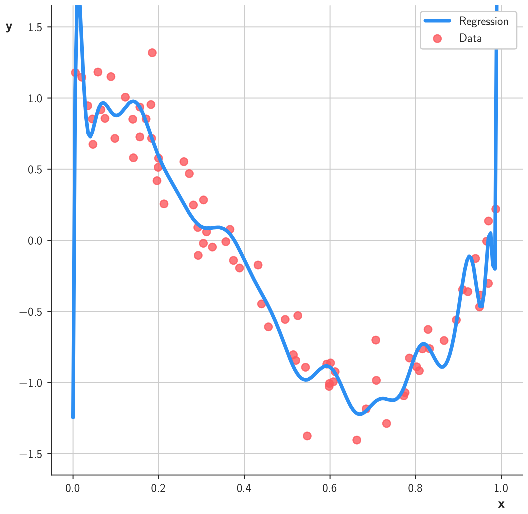
Polynomial regression without regularization showing overfitting



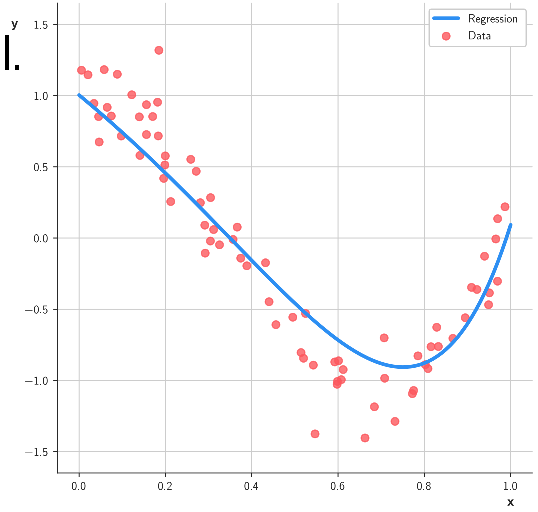
Polynomial regression with regularization (ridge), which in this case prevents overfitting.

Regularized regression

- Types of regularization:
  - L1 regularization (Lasso):
    - Adds absolute values of coefficients to the loss function.
    - Encourages sparsity; some coefficients become zero, effectively excluding unimportant features.
  - L2 regularization (Ridge):
    - Adds squared values of coefficients to the loss function.
    - Shrinks coefficients towards zero but usually keeps all features in the model.
  - Elastic net (ElasticNet):
    - Combines L1 and L2 regularization
    - Balances sparsity and small coefficients.
- Summary:
  - Reduce overfitting by constraining model complexity
  - Improves interpretability + robustness (especially in high dimensions)
  - Feature selection comes along for free (e.g., in Lasso)



Polynomial regression without regularization showing overfitting



Polynomial regression with regu-larization (ridge), which in this case prevents overfitting.