Applied Machine Learning

Feature Engineering: Feature Transformations

Learning goals

- Normalization
- Box-Cox Transformation

FEATURE TRANSFORMATIONS

- **Normalization:** The feature is transformed to have a mean of 0 and standard deviation of 1

$$z^{(i)} = \frac{x^{(i)} - \mathrm{mean}(x)}{\mathrm{sd}(x)}$$

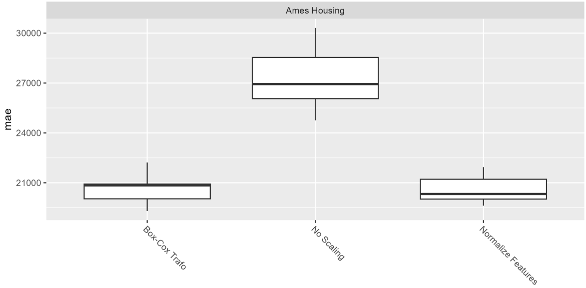
  Or use **robust** versions (median, IQR)

- **Box-Cox Transformation:** Stabilizes variance, makes the data more normal distribution-like

$$z^{(i)} = \begin{cases} \frac{\left(x^{(i)}\right)^{\lambda} - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \log\left(x^{(i)}\right) & \text{if } \lambda = 0 \end{cases}$$

FEATURE TRANSFORMATIONS

To illustrate the effect of transforming the features we evaluate a k-NN learner without scaling, with normalization, and with a Box-Cox transformation:



OTHER COMMON TRANSFORMATIONS

- Polynomials: $x_j \longrightarrow x_j, x_j^2, x_j^3, \ldots$
- Interactions: $x_j, x_k \longrightarrow x_j, x_k, x_j \times x_k$
- Basis expansions: BSplines, TPB, ...
- Fourier expansions

These transformations are used to improve simple models, e.g. linear regression, and most likely will **not** improve complex machine learning models.

FEATURE TRANSFORMATIONS - OTHER DATA TYPES

Feature transformations allow handling a variety of data types:

- **Dates:**
  - Time since X
  - Birthday $\rightarrow$ age
  - Extract month, day of the week, ...

- **Other:**
  - Use outputs of neural networks (images, text)
  - Bag-of-words (text)
  - Statistics (time-series)

FEATURE TRANSFORMATIONS - SUMMARY

- Transformations of the target variable can make modelling highly skewed data easier
- Scaling single features can lead to better results
- For some learners, e.g. tree-based methods scaling has no effect!
- Feature transformations can further improve models - use domain knowledge!