Explainable Artificial Intelligence **Paderborn University**
Summer Semester 2025 Dr. Stefan Heindorf

Exercise Sheet 2

Due date May 12, 2025

1 Task: Standard Error Calculation & Feature Importance

You are given a small dataset from a university study in which four students reported the number of hours they studied (X_1) and slept (X_2) before an exam, along with their resulting grade (y) on the German scale (1.0 is best, 5.0 is fail):

Hours Studied (X_1)	Hours Slept (X_2)	Grade (y)
1	8	4.0
5	4	1.3
4	6	1.7
2	7	3.0

a. Train a linear regression model (including an intercept) to predict the exam grade y from the features X_1 (Hours Studied) and X_2 (Hours Slept). Compute all regression coefficients by hand, clearly showing each step of your calculations. Report the final coefficients for each variable and for the intercept.

b. Compute the standard error for each regression coefficient. Present your computation process step by step, with all intermediate results clearly shown and explained. Finally, calculate and report the t-statistic for each coefficient.

c. Write a brief interpretation of your findings: What does each coefficient represent in this context? How do the number of hours studied or slept affect the predicted grade? Which feature seems most important, and why?

2 Task: Analyze Linear Models with Python

a. Download the bike rental dataset from this link: `https://github.com/christophM/interpretable-ml-book/blob/master/data/bike.csv`. Load the data as a `pandas` dataframe.[1]

b. Preprocess the data (e.g., convert columns types, remove unnecessary columns, encode categorical features using one-hot encoding, etc.) in a way that the final dataset contains the following features:

- `workdayY`: whether it is a workday (`1` if `workday="Y"`, `0` otherwise)
- `windspeed`: wind speed (km/h)

[1] https://pandas.pydata.org/

- `weatherMISTY`: whether the weather is misty (`1` if `weather=MISTY`, `0` otherwise)
- `weatherBAD`: whether the weather is bad (`1` if `weather=BAD`, `0` otherwise)
- `temp`: temperature (in Celsius)
- `seasonSUMMER`: whether it is summer (`1` if `season=SUMMER`, `0` otherwise)
- `seasonSPRING`: whether it is spring (`1` if `season=SPRING`, `0` otherwise)
- `seasonFALL`: whether it is fall (`1` if `season=FALL`, `0` otherwise)
- `hum`: relative humidity (in %)
- `holidayY`: whether it is a holiday (`1` if `holiday="Y"`, `0` otherwise)
- `cnt_2d_bfr`: number of rented bikes two days before
- `cnt`: number of rented bikes on the current day including both casual and registered users (target variable)

c. Split the dataset into a training set (80%) and a test set (20%).

d. Use the `statsmodels` library[2] to train a linear model, compute coefficients, and their standard errors. Report the results in a table.

e. How well does the linear model predict the number of rented bikes? What is its mean squared error on the train and the test set?

f. Visualize feature weights with Python.[3] The result should look reletively similar to the feature weights plot on the slides[4].

g. Visualize effect plot with Python.[5] The result should look reletively similar to the effect plot on the slides[6].

h. For each plot, write a paragraph that describes what can be seen in the plot and what this means for the bike rental company. What features are most important? What features are least important? Are there any unexpected findings/outliers in the plots that you did not expect? How can they be explained?

[2] `https://www.statsmodels.org/stable/index.html`

[3] For example, you can use `matplotlib`'s errorbar for this. `https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.errorbar.html`

[4] `https://christophm.github.io/interpretable-ml-book/limo_files/figure-html/fig-linear-weights-plot-1.png`

[5] For example, you can use `matplotlib`'s boxplot for this. `https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.boxplot.html`

[6] `https://christophm.github.io/interpretable-ml-book/limo_files/figure-html/fig-linear-effects-1.png`

3 Task: Analyze Decision Trees with Python

a. Preprocess the data similarly as for Task 2.

b. Use `scikit-learn` to train a decision tree on the bike rental dataset (with a maximal depth of `2`).

c. How well does the model predict the number of rented bikes? What is its mean squared error on the train and the test set?

d. Visualize the tree with `scikit-learn`.[7] Write a short paragraph that describes what is shown in the plot and what this means for the decision tree and the bike rental company.

e. Visualize the tree with `dtreeviz`.[8] Write a short paragraph that describes what is shown in the plot and what this means for the decision tree and the bike rental company.

f. Create a table of feature importances.[9] Write a short paragraph that describes what is shown in the plot and what this means for the decision tree and the bike rental company.

g. Pick a feature and calculate its importance by hand. Compare your results with the results from `scikit-learn`.

4 Task: Encoding Categorical Features

In the lecture slides, you learned about various encoding methods for categorical features. In this task, we will explore how different encoding methods affect the linear regression model.

a. Use the same dataset as in Task 2, but suppose that we want to predict the number of rented bikes (`cnt`) only based on the `weather` feature. Therefore, keep only the `cnt` and `weather` columns in the dataset.

b. Create three different versions of the dataset: one with treatment coding, one with effect coding, and one with dummy coding for the `weather` column.

c. Train a linear regression model on each of the three datasets. Report the coefficients and their standard errors in a table. Compare the results and discuss the differences. Furthermore, check if the interpretation of the bias/intercept aligns with the encoding method used[10].

[7] For example with `plot_tree` from `scikit-learn` library. `https://scikit-learn.org/stable/modules/generated/sklearn.tree.plot_tree.html`

[8] More information can be found here: `https://github.com/parrt/dtreeviz/`

[9] For example with `scikit-learn`'s `feature_importances_`, see `https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html#sklearn.tree.DecisionTreeRegressor.feature_importances_`

[10] See the page 30 of the lecture slides for more information.