Practical task 2

The assignment can be done in R or Python, or other software. Every group will have extra points according to the performance of the Decision Tree that they have built. To improve performance of classification feature engineering methods including correlation analysis, feature transformation, are advisable.

Table 1. Dataset information

Column	Description
work_year	The year the salary was paid.
experience_level	The experience level in the job during the year with the following possible values: EN Entry-level / Junior MI Mid-level / Intermediate SE Senior-level / Expert EX Executive-level / Director
employment_type	The type of employement for the role: PT Part-time FT Full-time CT Contract FL Freelance
job_title	The role worked in during the year.
salary	The total gross salary amount paid.
salary_currency	The currency of the salary paid as an ISO 4217 currency code.
salary*in*usd	The salary in USD (FX rate divided by avg. USD rate for the respective year via fxdata.foorilla.com).
employee_residence	Employee's primary country of residence in during the work year as an ISO 3166 country code.
remote_ratio	The overall amount of work done remotely; possible values are as follows: 0 No remote work (less than 20%) 50 Partially remote 100 Fully remote (more than 80%)
company_location	The country of the employer's main office or contracting branch as an ISO 3166 country code.
company_size	The average number of people that worked for the company during the year: S less than 50 employees (small) M 50 to 250 employees (medium) L more than 250 employees (large)

Source: https://www.kaggle.com/datasets/ruchi798/data-science-job-salaries

Homework

Pre-processing

1. Transform the categorical features into binary variables (dummy variables). Dummy variables are accepted by almost any classifier. **(1 point)**

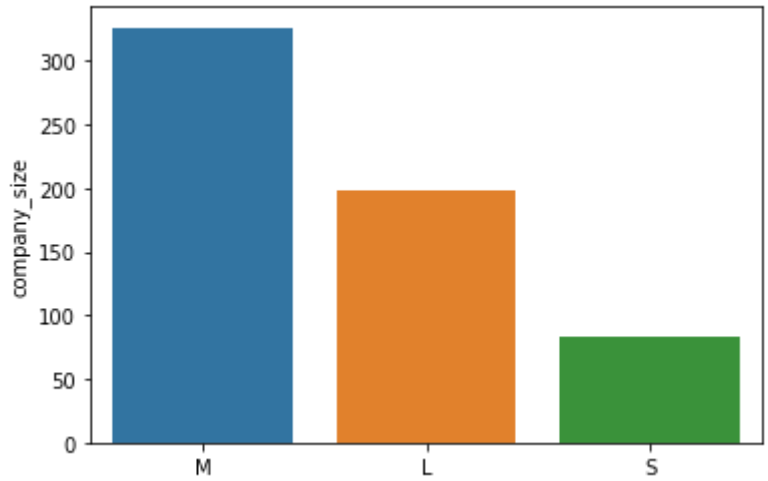
Processing

2. Fit a Decision Tree to classify the size of company in which each data scientist work. This is represented by the feature 'company_size', which is the target variable. (**3 points)**
   2.1. Set the maximum depth of the tree to 3. (1 point)
   2.2. Plot the decision tree. (2 point)
3. Interpret the results of the decision tree **(3 points)**
   3.1. Write a paragraph interpreting the structure of the decision tree.

Post-processing

4. Measure the performance of the decision tree to make accurate predictions.
   4.1. By hold-out cross-validation performance **(3 points)**
      4.1.1. Estimate the accuracy, interpret the results. (1.5 points)
      4.1.2. Estimate the Matthews correlation coefficient, interpret the results (1.5 points)

Figure 1. Distribution of classes for 'company_size'



- The target variable, or the dependent variable, that we want to predict is unbalanced among the classes. Work market for data scientists is composed by medium and large companies (13.67% of cases).
- Hold-out cross validation algorithm:
  - ✓ Select a sample size for learning and testing the Decision Tree. Example: 80% of sample to learn and 20% to test.
  - ✓ Randomly split the data into train and test subsets.
  - ✓ Fit a Decision Tree to the training data and estimate the score on the test data. Example: estimate accuracy of the Decision Tree on the test subset.
- The hold-out cross validation can be repeated several times with different random seeds to draw a distribution of scores and perform a robust score evaluation.
- Any other feature engineering method, or dimensionality reduction can be tested to obtain better results.