# Feature Selection **(scored tasks: 4 points)**

**Tasks (Lab 13):**

1. Data generation. Generate datasets as follows. Consider 3 parameters: $n$ (size of the training data), $p$ (number of all features) and $k$ (number of significant features).

   (a) Dataset 1:

   - Generate $X_1, X_2, \ldots, X_p \sim N(0,1)$.
   - Denote by $\chi^2_k(0.5)$ median of chi squared distribution with $k$ degrees of freedom.
   - Set $Y = 1$ if $\sum_{j=1}^{k} X_j^2 > \chi^2_k(0.5)$ and $Y = 0$ otherwise.
   - Generate training data of size $n$.

   (b) Dataset 2:

   - Generate $X_1, X_2, \ldots, X_p \sim N(0,1)$.
   - Set $Y = 1$ if $\sum_{j=1}^{k} |X_j| > k$ and $Y = 0$ otherwise.
   - Generate training data of size $n$.

2. Comparison of feature selection/feature ranking methods. Consider the following methods:

   - Variable importance measures based on Random Forest (mean decrease in impurity and permutation based measure).
   - Boruta algorithm (you can use R package `Boruta` or Python package `boruta_py`).

   Tasks:

   (a) Check if the considered algorithms assign the highest variable importance scores to the significant variables.

   (b) Try different values of $n$, $p$ and $k$. First, you can use values: $n = 500$, $p = 50$ and $k = 10$.

   (c) Repeat data generation process $L = 50$ times and estimate the probabilities of correct ordering, i.e. compute the fraction of simulations in which the significant variables are selected as relevant or are placed on the top of the ranking list. Take smaller $L$ if simulations are too expensive computationally.

   (d) Fix $n = 200$, $p = 500$, $k = 20$. Let $t$ denote the number of top-ranked features according to the considered feature importance measure. Train the classifier (e.g. random forest) using $t$ top-ranked features and analyze how the classification accuracy depends on $t = 5, 10, 15, 20, 50, 100, 200, 300, 400, 500$. Generate a plot showing the dependence.