Machine Learning

Supervised learning 2

S. Nõmm

[1]Department of Software Science, Tallinn University of Technology

23.02.2021

Linear model building 1

- Choose or determine all the hyperparameters. Possible order limitations, backward elimination / forward selection/ batch processing, set the level of significance and threshold for correlation. These parameters also define stopping criteria.
- Stop when: model is significant, and goodness parameters as expected OR no more variables to add or delete OR maximal or minimal order is reached etc.
- Investigate if available explanatory variables (predictors) are linearly independent. Strong dependencies between variables chosen as "independent" lead problems with inverting matrix X. Compute *multicollinearity* matrix where element in ith row and jth column is Pearson correlation coefficients computed for variables i and j. Based on this table determine subset(s) of variables which are linearly independent.

Linear model building 2

- Repeat
- Apply mean squares (or other technique) to build the model from selected variables.
- Evaluate significance- and quality- of the model. For quality observe determination coefficient and error. For significance use F - test and t-test variable wise.
- If model fail goodness or significance check then return to the previous model and choose another set of variables to add/delete.
- Starting from second iteration prove, using F - test, that as a result of adding/deliting variables model quality has improved/did not decreased significantly.
- If adding/deliting variables was not successful return to the previous model and if possible chose another variable(s) to add /delete or report the model from previous step.
- If goodness criteria (quality and significance) is met stop and return the model.
- If goodness criteria was not met but adding deleting variables proved to be successful chose the set of variables to be added or deleted (t-test) on the next step.
- Until stopping criteria is reached.
- Report the results.

Linear model building 3

Reminder p - is the number of variables n is the sample size.

- F -test of overall significance in regression analysis.
- Test for model significance. $H_0 : b_1 = \ldots = b_p = 0$, $H_1 : \exists i : 1 \leq i \leq p \& b_i \neq 0$.
- Test statistic:

$$F = \frac{\frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{p-1}}{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n-p}}$$

- Rejection rule: Determine using F-table or corresponding software function with chosen significance level, n degrees of freedom in denominator and p degrees of freedom in nominator.

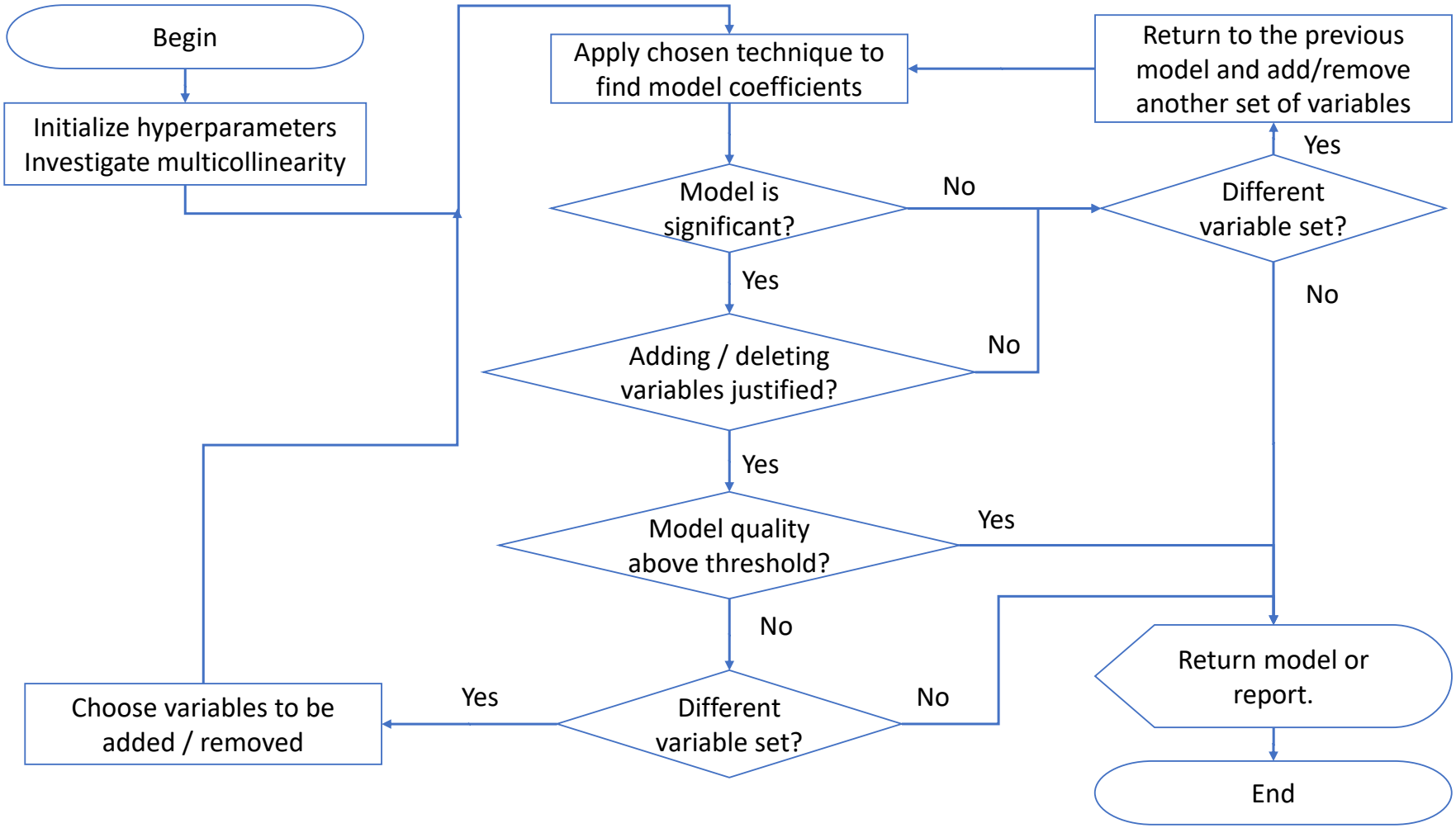
Linear model building 4

- F -test to determine significance of change in model quality caused by adding variables
  - $H_0 : RSS_S \leq RSS_C$, $H_1 : RSS_S > RSS_C$.
  - Test statistic:

$$F = \frac{\frac{RSS_S - RSS_C}{m}}{\frac{RSS_C}{n-p-1}}$$

  - Rejection rule: Determine using F-table or corresponding software function with chosen significance level, $n - p - 1$ degrees of freedom in denominator and m degrees of freedom in nominator.
- t - test on individual regression coefficients
  - $H_0 : b_i = 0$, $H_1 : b_i \neq 0$.
  - Test statistic: $t = \hat{b_i}/se(\hat{b_i})$
  - Use t - table or corresponding function to find rejection rule for chosen significance and $n - 2$ degrees of freedom.

Linear model building 5



Nonlinear regression

- By replacing independent variables X with a nonlinear mapping $\phi(X)$.
- This will lead
$$f_\theta(X) = \theta^T \phi(X)$$
- This process is referred as basis function expansion.
- Example: Polynomial regression has basis function $\phi(X) = [1, x, x^2, \ldots, x^d]$. The model remains linear in the parameters.

Polynomial regression 1

- Higher degree polynomial models tend to over fit. The coefficients become relatively large, which causes the regression curve to "wiggle".
- In order to achieve "encourage" smaller weight values introduce zero-mean Gaussian prior:

$$p(\theta) = \prod_j \mathcal{N}(\theta_j|0, \tau^2)$$

  where $1/\tau^2$ controls the strength of prior.
- This lead following log-likelihood estimate

$$\ell = \sum_{i=1}^{N} \log \mathcal{N}(y_i|\theta^T x_i, \sigma^2) + \sum_{j=1}^{p} \log \mathcal{N}(\theta_j|0, \tau^2)$$

- The solution is given by:

$$\hat{\theta}_r = (\lambda I + X^T X)^{-1} X^T y$$

Logistic regression

- Remind that linear regression may be written in the following form:

$$p(y|x,\theta) = \mathcal{N}(y|\mu(x),\sigma^2(x))$$

- This may be generalized to the binary setting as follows:

$$p(y|x,\theta) = \mathrm{Ber}(y|\mathrm{sigm}(\theta^T x))$$

where $\mathrm{sigm}(\eta) = (1+e^{-\eta})^{-1}$. Will be referred as *logistic regression*.

- Fitting is usually done by maximum likelihood

$$\ell(\theta) = \sum_{i=1}^{N} \log p_(g_i)(x_i|\theta) = \sum_{i=1}^{N} \left\{ y_i \beta^T x_i - \log(1+e^{\beta^T x_i}) \right\}$$

- Solving the last one is done by means of iterative algorithm.

$$\begin{aligned} b^{\text{new}} &= \arg\min_b (z - Xb)^T W (z - Xb) \\ z &= Xb + W^{-1}(y - p) \end{aligned}$$

where W is a $N \times N$ diagonal matrix with ith element $p(x_i, |b)(1 - p(x_i|b))$

Decision trees

- Non-parametric supervised learning technique.
- Tree-like graph is used to represent the model of decision making and possible consequences of such decisions.
- Internal nodes are conditions (questions). terminal nodes represent labels of classes.
- Questions or conditions play a role of features. Answers to the questions are referred as feature values.
- Training a tree model is referred as *tree growing*.

Growing a tree 1

Greedy heuristic is the most popular technique. Let F be the possible set of features and S is the subset of data. The idea is to find most useful feature (among remaining) at each node.

$$j(S) = \arg\min_{j \in F} \text{cost}(\{x_i, y_i : x_i \in S, x_{i,j} = c_k\}) + \text{cost}(\{x_i, y_i \; x_i \in S, x_{i,j} \neq c_k\})$$

Classification cost:

$$\hat{\pi}_c = \frac{1}{|S|} \sum_{x_i \in S} \mathbb{1}\{y_i = c\}$$

Misclassification rate:

$$\frac{1}{|S|} \sum_{x_j} \in S \mathbb{1}(y_i \neq \hat{y}) = 1 - \hat{\pi y}$$

Cost functions

- Entropy:

$$\mathbb{H}(\hat{\pi}) = -\sum_{c=1}^{C} \hat{\pi}_c \log_2 \hat{\pi}_c$$

  Minimizing entropy is equivalent to maximizing information gain which is $\mathbb{H}(Y) - \mathbb{H}(Y|X_j)$.

- Gini index:

$$G = \sum_{c=1}^{C} \hat{\pi}_c (1 - \hat{\pi}_c)$$

Growing a tree 3

- Repeat:
  - For each feature divide data into corresponding subsets. Evaluate accuracy of such split with respect to response variable.
  - "Most accurate" feature wins. It will become condition at a given node.
  - Exclude chosen feature from the feature set.
- Until no more features left.

Example: When to play tennis

Outlook	Temperature	Humidity	Wind	Play
sunny	warm	high	weak	no
sunny	warm	high	strong	no
rain	warm	high	weak	yes
rain	cool	normal	weak	yes
rain	cool	normal	strong	no
sunny	cool	normal	strong	yes
sunny	warm	high	weak	no
sunny	cool	normal	weak	yes
rain	warm	normal	weak	yes
sunny	warm	normal	strong	yes
rain	warm	high	strong	yes
sunny	warm	normal	weak	yes
rain	warm	high	strong	no

Information gain

Definition

Information gain G_I of an action is the decrease of the ambiguity achieved as the result of the action.

- In the context of decision tree growing the action is splitting the node.
- If entropy is chosen as the cost function then information gain is defined as follows:

$$G_I = E - (E_l \cdot p_l + E_r \cdot p_r)$$

where E is the entropy before splitting E_l is the entropy of left child and E_r is the entropy of the right child. Indexes r and l have the same meaning for the proportions p.

Growing the tree: case of continues features

Denote X the matrix where columns correspond to different features and rows correspond to the different observation points.

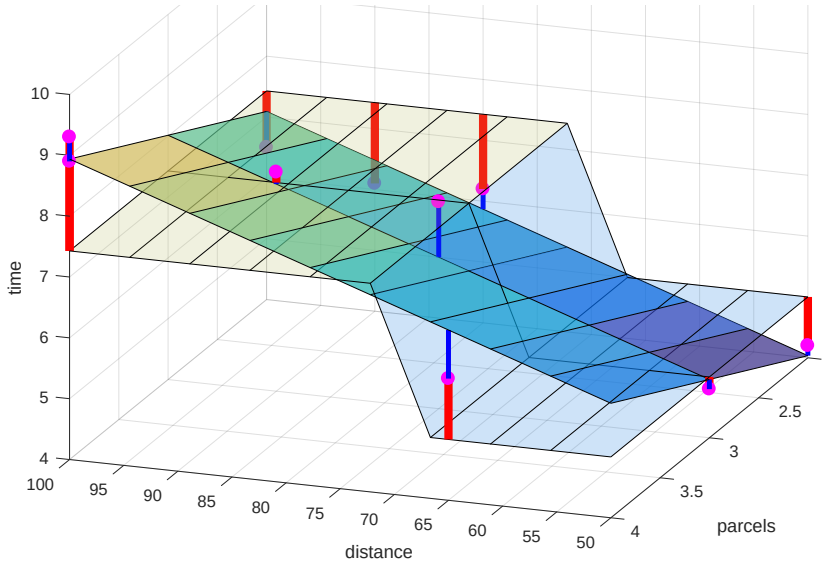
- If all the data points are of the same class return the leaf node that predicts this class.
- Among all splitting points for each column find the one giving largest information gain.
- Then chose the column with the maximum gain.
- Perform splitting.
- If stopping criteria is satisfied return the tree.
- If stopping criteria is not satisfied apply tree growing procedure to each child.

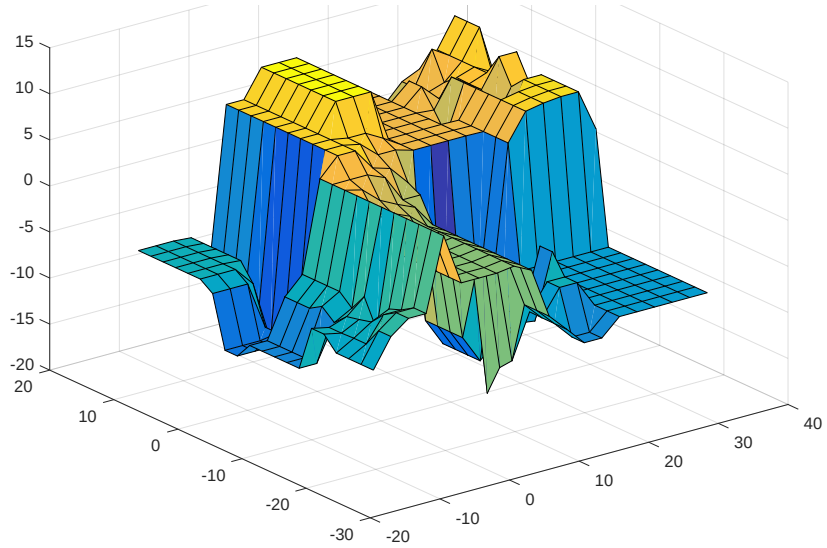
Pruning

- In order prevent overfitting stop growing the tree when the decrease is not sufficient to justify adding extra subtree.
- Grow a full tree and then prune the branches giving less decrease in error.

Regression trees

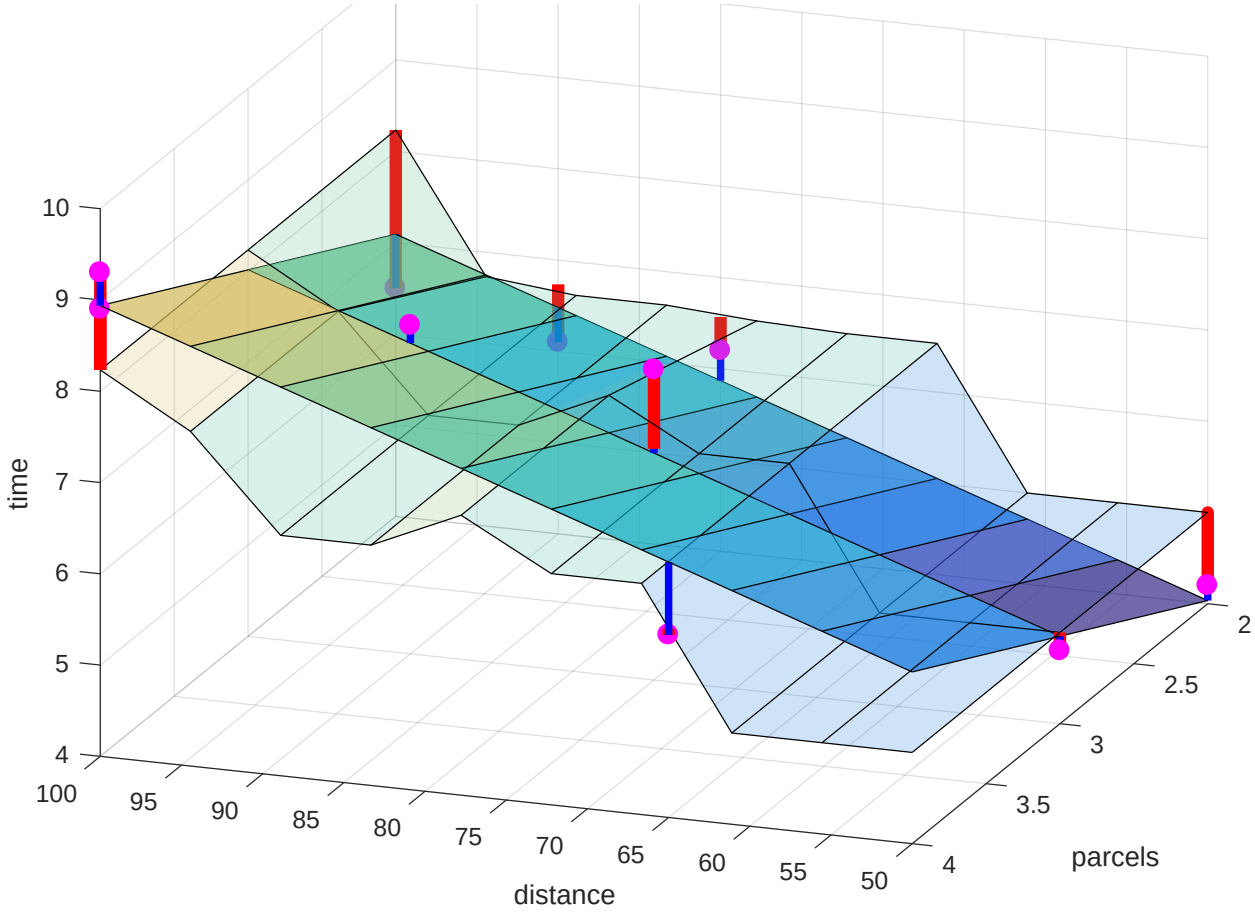
- Partition the feature space into the set of rectangles.
- Fit a simple model (for example constant) in each rectangle.
- Fitting the model is similar to the case of classification trees.





k-nn regression

The value of the response (dependent variable) defined as the average of its k nearest neighbours from the training set.



Self practice

- Program your own implementation for multivariate regression model building.
- Program your own implementation of decision tree growing.
- Program your own implementation of k - nearest neighbors regression.