

# Machine Learning

## Supervised learning 1

S. Nõmm

<sup>1</sup>Department of Software Science, Tallinn University of Technology

15.02.2022

# Supervised learning

Is a task of inferring function (training a model) on the basis of labeled training data. The goal is to construct a function (train a model) which would mimic (in a certain sense) behaviour of the underlying process.

- Regression: Dependent variable (continuous) plays a role of labels.
  - ▶ Linear
  - ▶ Nonlinear
  - ▶ Application of trees and SVM for regression.
  - ▶ Advanced methods like Neural Networks, etc.
- Classification labels are discrete (categorical values).
  - ▶  $k$ -nearest neighbours.
  - ▶ Decision trees.
  - ▶ Support Vector Machines.
  - ▶ Neural networks.
  - ▶ Ensemble (committee).
  - ▶ Boosted techniques.
- Markov models.

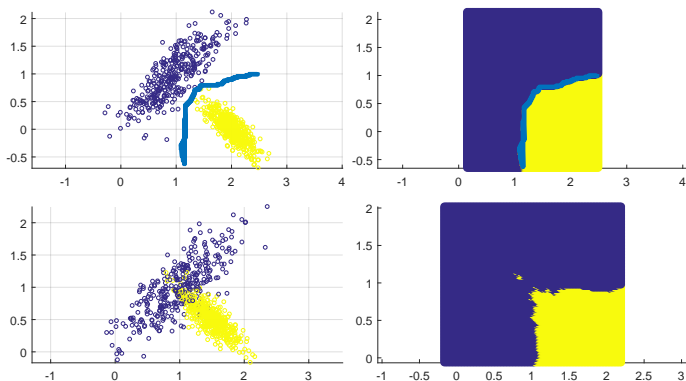
# Classification

- Learning existing grouping on the basis of the labeled (training) set.
- The goal is to generate (choose the structure and train) a model which would mimic existing grouping.
- Based on the features of the element model should estimate which class element belong to or estimate value of dependent variable.
- Unlike the case of unsupervised learning miss classification may be precisely measured.
- What is the cost of miss classification or error in the case of regression?

## $k$ - nearest neighbours ( $k$ -NN)

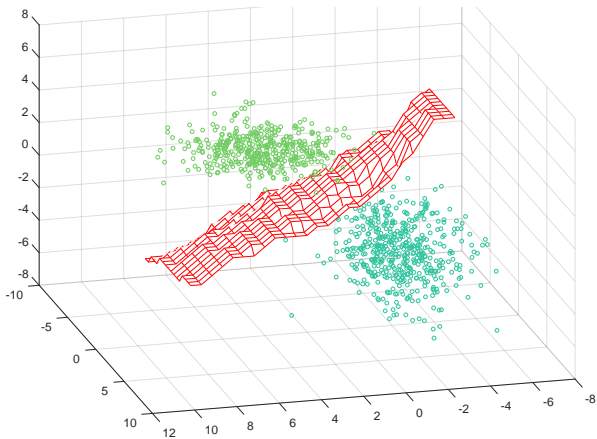
- Let  $D$  denote training (labeled) data set.
- For each unlabeled point (point to be classified)
  - ▶ Find  $k$  - nearest neighbours.
  - ▶ Assign mode (majority) label of  $k$  - nearest neighbours.

## k - nearest neighbors, geometric interpretation, 2D



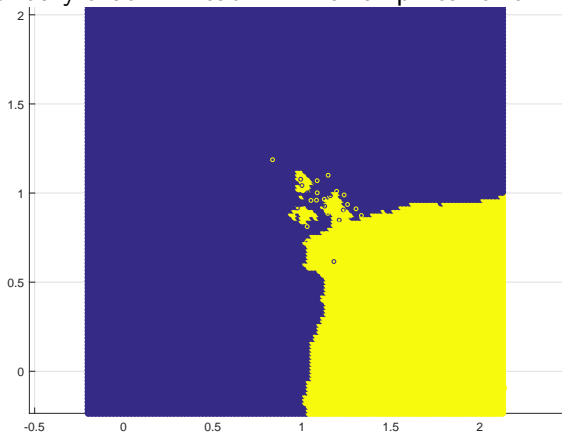
- Decision boundary (decision surface) (statistical classification with two classes) is a hypersurface that partitions the data set into two subsets, one for each class.
- Classifier tries to learn (construct) decision boundary that will lead minimal empirical error.

## k - nearest neighbors, 3D



# Accuracy

During the training (learning) process classifier tries to learn (construct) decision boundary that will lead minimal empirical error.



How good is trained classifier?

# Validation

- Overall accuracy and Confusion matrix (table), computed for the validation subset, are the goodness parameters of trained classifier.

	Predicted Class 1	Predicted class 2
Actual class 1	58	2
Actual class 2	6	134

- How reliable these parameters are ?



# Cross validation

- Non-exhaustive do not use all possible ways of splitting into training and validation sets
  - ▶  $k$  - fold.
  - ▶ Holdout.
  - ▶ Repeated random sub-sampling.
- Exhaustive: use all possible ways to divide the data set into training and validation sets
  - ▶ Leave  $p$ -out cross validation.
  - ▶ Leave one out cross validation.

## Cross validation: $k$ - fold validation

- Divide the training data (after removing test data) randomly into  $k$  - folds.
- Perform following  $k$  experiments:
  - ▶ Compose the training data by concatenating  $k-1$  folds leaving one fold out.
  - ▶ Train the model on those  $k-1$  folds
  - ▶ Test it on the left-out fold
  - ▶ Record the result
- Report the average of the  $k$  experiments.

# Learning: Underfitting and overfitting

- *Underfitting* the learned function is too simple In the context of human learning: underfitting similar to the case when one learns too little.
- *Overfitting* the learned function is too complex In the context of human learning: overfitting is more similar to memorizing than learning.

## Feature selection for classification

- Case of categorical data: Gini Index or Entropy. Value specific:

$$G(v_i) = 1 - \sum_{j=1}^k p_j^2; \quad E(v_i) = - \sum_{j=1}^k p_j \log_2(p_j)$$

where  $p_j$  is the fraction of data points containing attribute value  $v_i$ . Lower values of Gini index or Entropy imply greater discriminative power.

- Feature specific: Let  $n_i$  is the number of data points taking value  $v_i$ . Feature specific Gini index is defined as the weighted average value of value specific Gini indexes.

$$G = \sum_{i=1}^r \frac{n_i G(v_i)}{n}$$

where  $r$  is the number of different values  $v_i$  and  $n = \sum n_i$ .

- Feature specific values of Entropy are computed in the similar way.

## Feature selection for classification II

- Case of numeric data: Fisher's score

$$F = \frac{\sum_{j=1}^k p_j (\mu_j - \mu)^2}{\sum_{j=1}^k p_j \sigma_j^2}$$

Greater values imply greater discriminative power of the variable.

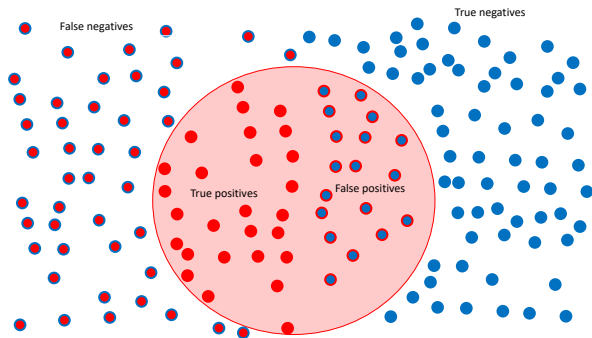
- Wrapper methods.

# Classification model goodness!

- How good is the model?
- What is the goal of modeling?

# Classification outcome

- Consider binary classifier.
- In the data set there are two classes: Positive (P) and negative (N)
- Outcomes of the classification: True positive, true negative, false positive (type I error), false negative (type II error).



# Context of information retrieval

NB! Observe notions!

- Relevant elements of the data set. One is interested to find (retrieve elements of the certain class).
- Precision is defined as:

$$\text{precision} = \frac{|\text{relevant} \cap \text{retrieved}|}{|\text{retrieved}|}$$

- Recall (sensitivity, hit rate, True Positive Rate) is defined as:

$$\text{recall} = \frac{|\text{relevant} \cap \text{retrieved}|}{|\text{relevant}|}$$



# Context of classification I

Denote:  $tp$  - true positive,  $tn$  - true negative,  $fp$  - false positive and  $fn$  - false negative.

- Precision (positive predictive value):

$$\text{Precision} = \frac{tp}{tp + fp}$$

- Recall (sensitivity, hit rate, TPR):

$$\text{Recall} = \frac{tp}{tp + fn}$$

- True negative rate (Specificity, selectivity):

$$\text{TNR} = \frac{tn}{tn + fp}$$

- Accuracy:

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

- Predicted positive condition rate

$$\text{Predicted positive condition rate} = \frac{tp + fp}{tp + tn + fp + fn}$$

## F-measure *not to be confused with similarly named values!!!*

Frequently referred as  $F_1$ -score ... is harmonic average of precision and recall.



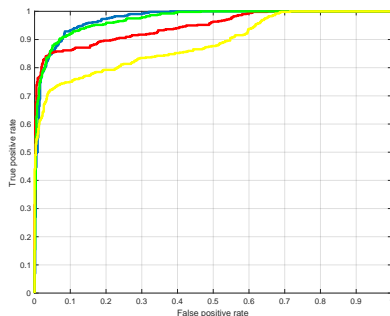
$$F = 2 * \frac{\text{precisionrecall}}{\text{precision} + \text{recall}}$$

- More general definition:

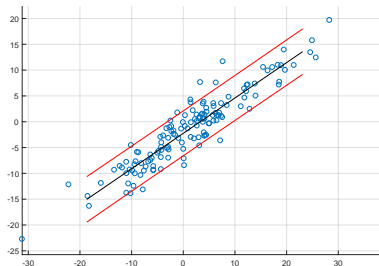
$$F_{\beta} = (1 + \beta^2) \frac{\text{precisionrecall}}{\beta^2 \text{precision} + \text{recall}}$$

# Receiver Operating Characteristic or ROC curve

- Let  $\mathcal{D} = \{x_i, y_i\}$  is the labeled data set.
- Assume also that  $\delta(x) = \mathbb{I}(f(x) > \tau)$  - decision rule.  $f(x)$  is the confidence function and  $\tau$  threshold parameter
- Each particular value of  $\tau$  corresponds to a certain decision rule.
- For each decision rule one may compute recall and false positive rate.
- Associate recall values with the axis Y and false positive rate values with axis X.



# Linear regression: probably the oldest machine learning technique



- Find linear correlation coefficient.
- Compute coefficients of the linear equation

$$\hat{y} = ax + b$$

- Evaluate the model

- In multivariate case it is required to identify coefficients of the model

$$\hat{y} = a_1x_1 + a_2x_2 + \dots + a_nx_n + b.$$

This leads the necessity to choose variables (perform model building).

# Linear regression

- Correlation coefficient.

$$\rho = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}}$$

where,  $n$  - is the sample size,  $x$  and  $y$  are the variable of interest.

- $-1 \leq \rho \leq 1$
- Assumption there are exist  $\alpha$  and  $\beta$  such that for any  $i = 1, \dots, n$   $y_i = \alpha x_i + \beta + \varepsilon_i$  holds. Assumption:  $\varepsilon$  is sufficiently small normally distributed.
- The goal of regression is to find estimates of the coefficients  $\alpha$  and  $\beta$ , such that for  $a$  and  $b$

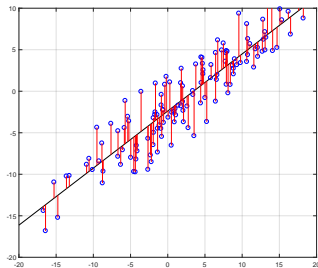
$$y_i = ax_i + b + \hat{\varepsilon}_i$$

sum of squares of  $\hat{\varepsilon}_i$  would be minimal. NB! notation  $\hat{\alpha}$  and  $\hat{\beta}$  is also widely use.

# Least squares method

Least squares method:

$$a = \frac{\sum_{i=1}^n x_i y_i}{\sum_{i=1}^n x_i^2}; \quad b = \bar{y} - a\bar{x}$$



For an arbitrary number of variables:

$$y = b_1 x_1 + \dots + b_n x_n + b_0$$

then

$$\hat{b} = (X^T X)^{-1} X^T y.$$

where each row of matrix  $X$  is input vector with 1 in the first position.

## Model validation

- Coefficient of determination  $R^2$  and adjusted  $R^2$ .
- Significance of the model and model coefficients.
- Verify assumption that residuals are normally distributed.
- Residual sum squares.  $RSS = \sum_{i=1}^N (y_i - x_i^T \beta)^2$ .
- Sum squares of the regression  $SSR = \sum_{i=1}^N (\hat{y}_i - \bar{y})^2$ .
- Total sum squares or sum of squares about the mean  $SST = \sum_{i=1}^N (y_i - \bar{y})^2$ .
- $R^2$  computed as the ratio of Sum squares of the regression to total sum squares or one minus ratio of Residual sum squares to total sum squares whereas adjusted  $R^2$  is one minus ratio of residual sum squares computed for  $n - 1$  to Total sum squares for  $n - p$  observation points.

# MLE for regression least squares I

- Linear regression is the model of the form

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

where  $\beta$  are the coefficients of the linear model,  $\sigma$  is the standard deviation of  $x$  and  $\theta = (\beta, \sigma^2)$

- Parameter estimation of a statistical model is usually performed by computing MLE  $\hat{\theta} = \arg \max_{\theta} \log p(\mathcal{D}|\theta)$ . remind that  $\mathcal{D}$  denotes the data set



## MLE for regression least squares II

- Assumption: elements of the training set are independent and identically distributed.
- Then log likelihood is given by
$$\ell(\theta) = \log p(\mathcal{D}|\theta) = \sum_{i=1}^N \log p(y_i|x_i, \theta).$$
- As usually instead of maximizing the log-likelihood one may minimize negative log likelihood.
- 

$$\begin{aligned}\ell(\theta) &= \sum_{i=1}^N \log \left[ \left( \frac{1}{2\pi\sigma^2} \right) \exp\left( -\frac{1}{2\sigma^2} (y_i - \beta^T x_i)^2 \right) \right] \\ &= \frac{-1}{2\sigma^2} \text{RSS}(\beta) - \frac{N}{2} \log(2\pi\sigma^2).\end{aligned}$$

# MLE for regression least squares II

- In order to minimize RSS differentiate its equation which lead

$$\nabla\theta = X^T X\beta - X^T y.$$

- Equate it to zero and solve for  $\beta$

$$\beta = (X^T X)^{-1} X^T Y$$

last equation is referred as *normal equation*.

# Regularization

- Overfitting may be caused by the fact that chosen model structure and data are not conform on another.
- Regularization is the technique used to overcome overfitting.
- Regularization imposes cost or penalty on the cost function and prevent larger values of the coefficients.
- Loosely speaking, regularization shrinks the coefficients towards zero and towards one another.

# Ridge regression

- Ridge regression shrinks the coefficients by penalizing their size.

$$\hat{\beta}^{\text{ridge}} = \operatorname{argmin}_{\beta} \left\{ \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2 + \lambda \sum_{j=1}^p \beta_j^2 \right\}$$

$\lambda$  is the nonnegative shrinkage parameter, its large values correspond to the greater amount of shrinkage applied.

- Alternatively the following notation is widely used:

$$\hat{\beta}^{\text{ridge}} = \operatorname{argmin}_{\beta} \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2$$

subject to  $\sum_{j=1}^p \beta_j^2 \leq t$

# The Lasso

- Ridge regression shrinks the coefficients by penalizing their size.

$$\hat{\beta}^{\text{lasso}} = \operatorname{argmin}_{\beta} \frac{1}{2} \left\{ \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2 + \lambda \sum_{j=1}^p |\beta_j| \right\}$$

$\lambda$  is the nonnegative shrinkage parameter, its large values correspond to the greater amount of shrinkage applied.

- Alternatively the following notation is widely used:

$$\hat{\beta}^{\text{ridge}} = \operatorname{argmin}_{\beta} \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2$$

subject to  $\sum_{j=1}^N |\beta_j| \leq t$

- Computing the lasso solution is a quadratic programming problem.

# Statistical hypothesis testing (brief reminder I)

- Assumption about a parameter of population is a statistical hypothesis.
- Usually a pair of hypothesis is stated  $(H_0, H_1)$ , notation  $(H_0, H_a)$ .
  - ▶  $H_0$  the null hypothesis usually states that there is no statistically significant relationship between two phenomena.
  - ▶  $H_1$  the alternative hypothesis usually states the opposite to the  $H_0$ .
- Choose and compute test statistic and rejection rule.
- Interpret the results.
- What can possibly go wrong?

## Statistical hypothesis testing (brief reminder II)

	Accept $H_0$	Reject $H_0$
$H_0$ is true	Correct	Type 1 Error
$H_0$ is false	Type II Error	Correct

## Model building (feature selection)

Let us suppose that observed process has  $p$  independent variables  $x_1, \dots, x_p$  and one dependent variable  $y$ . Should one build the regression equation using all  $p$  variables or not?

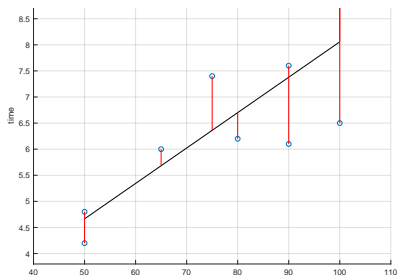
- Are all the variables  $x_1, \dots, x_p$  uncorrelated?
- Which subset of variables result in a "better" model?
- How to prove that as a result of adding or deleting a variable model quality has improved?



## ”Butler tracking company” example

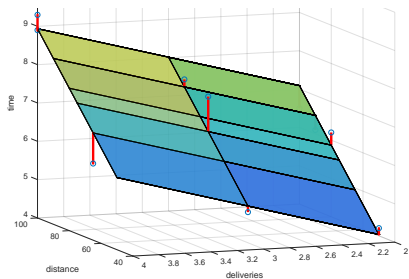
- Independent variables: Distance to drive and number of parcels to deliver. Dependent variable: time.
- Distances to drive for each assignment: 100, 50, 100, 100, 50, 80, 75, 65, 90, 90.
- Number of parcels to deliver: 4, 3, 4, 2, 2, 2, 3, 4, 3, 2
- Time in hours: 9.3, 4.8, 8.9, 6.5, 4.2, 6.2, 7.4, 6, 7.6, 6.1.
- Pearson correlation coefficient between distance and time is 0.81.

# "Butler tracking company" example continued



## Model 1

Is significant  $p = 0.004$ ,  
 $F = 15.1846$  whereas  
 $R^2 = 0.6641$ .



## Model 2

Is significant  $p = 0.000276$ ,  
 $F = 32.9$  whereas adjusted  
 $R^2 = 0.87$ .

Is it enough to say that model 2 is more precise?

## Quality comparison

- To compare different models *residual sum of squares* (RSS) is used.
- Hypothesis statements:  $H_0 : \text{RSS}_s \leq \text{RSS}_c$   $H_1 : \text{RSS}_s > \text{RSS}_c$ .
- Test statistic (empirical parameter) for ANOVA:

$$F_{stat} = \left( \frac{\text{RSS}_s - \text{RSS}_c}{m} \right) \left( \frac{\text{RSS}_c}{n - p - 1} \right)^{-1}$$

where  $\text{RSS}_c$  is the residuals sum squares of model with more variables,  $\text{RSS}_s$  - is the residuals sum squares of model with less variables,  $m$  number of variables added or removed,  $n$  is the number of observation points,  $p$  - is the number of variables in more complicated model.

- Rejection rule for  $\alpha$  (significance level), degrees of freedom: first is the number of variables added or removed, second is  $n - p - 1$ .
- Decision:
  - ▶ (if adding variables) rejected null hypothesis proves that adding variables caused model quality to increase significantly.
  - ▶ (if deleting variables) rejected alternative hypothesis proves that deleting variables did not cause model quality to significant decrease.

## "Butler tracking company" example continued

- $RSS_1 = 15.8713$ ,  $RSS_2 = 2.2994$  NB! Observe that corresponding MATLAB notation is SSE!!!
- choose  $\alpha = 0.05$  degrees of freedom: first will be 1 (one variable (number of parcels)) were added, second 7 ( $n = 10, p = 2$ ).
- Rejection rule: reject  $H_0$  if  $F_{stat} > 5.5914$
- Compute  $F_{stat} = 17.4411$ . (use table, or MATLAB or EXCEL)
- Reject  $H_0$ . Adding the variable has increased the model quality.