Introduction to machine learning

Victor Kitov

v.v.kitov@yandex.ru
victorkitov.github.io

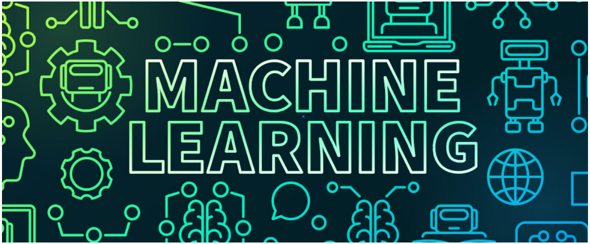


Table of Contents

Relevance

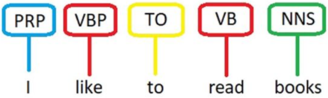
"The fourth technological revolution is built on the ubiquitous and mobile internet, by smaller and more powerful sensors that have become cheaper, and by artificial intelligence and machine learning." Klaus Martin Schwab, President of the World Economic Forum (2016).

Formal definitions of machine learning

- Machine learning is a field of study that gives computers the ability to learn without being explicitly programmed.
- A computer program is said to learn from **experience E** with respect to some **class of tasks T** and **performance measure P**, if its performance P at tasks in T improves with experience E.

Examples

- Spam filtering
  - if sender belongs to black-list -> spam
  - if contains phrase 'buy now' and sender is unknown -> spam
  - ...
- Part-of-speech tagger.
  - if ends with 'ed' -> verb
  - if previous word is 'the' -> noun
  - ...



- ML finds decision rules automatically with labelled data!

Manual approach and machine learning

Manual approach:

- difficult to find specialists
- expensive
- slow
- imprecise (simple rules only)

These problems are solved using machine learning.

- but data is needed for a formalized description of experience.

Where ML gives advantage

- hard to formulate explicit rules
    - complex inter-relationships
        - e.g. image recognition
    - too many attributes
        - e.g. text categorization
- fine-tuning performance on huge datasets
    - e.g. threshold for credibility in credit scoring
- fast adaptation to changing conditions
    - e.g. stock prices/volatility prediction
- many models are needed
    - due to further adaptation to usage conditions
    - e.g. voice detection

Formal problem statement

- Set of objects Z
- Each object is described by a vector of known characteristics $\mathbf{x} \in \mathcal{X}$ and predicted characteristics $y \in \mathcal{Y}$.

$$z \in Z \longrightarrow (\mathbf{x}, y)$$

- Task: find a mapping f, which could accurately approximate $\mathcal{X} \to \mathcal{Y}$.
    - using a finite **known set** of objects.
    - apply model for objects from the **test set**.
- test set way be known or not.

Types of features[1]

- Full object description $\mathbf{x} \in \mathcal{X}$ consists of individual features $x^i \in \mathcal{X}_i$
- Types of feature (e.g. for credit scoring):
  - $\mathcal{X}_i = \{0, 1\}$ - binary feature
    - e.g. marital status
  - $|\mathcal{X}_i| < \infty$ - categorical (nominal) feature
    - e.g. occupation
  - $|\mathcal{X}_i| < \infty$ and $\mathcal{X}_i$ is ordered - ordinal feature
    - e.g.education level
  - $\mathcal{X}_i = \mathbb{R}$ - real feature
    - e.g. age

[1] Actually any type is possible. Listed are most common types.

Specification of known/test sets

Known set:

- **supervised learning**: $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ...(\mathbf{x}_N, y_N)$
  - e.g. regression, classification.
- **unsupervised learning**: $\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_N$ -
  - e.g. dimensionality reduction, clustering, outlier analysis
- **semi-supervised learning**: $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ...(\mathbf{x}_N, y_N)$, $\mathbf{x}_{N+1}$, $\mathbf{x}_{N+2}, ...\mathbf{x}_{N+M}$

If test set objects $\mathbf{x}_1'$, $\mathbf{x}_2'$, $...\mathbf{x}_K'$ are known in advance, then this is **transductive learning**.

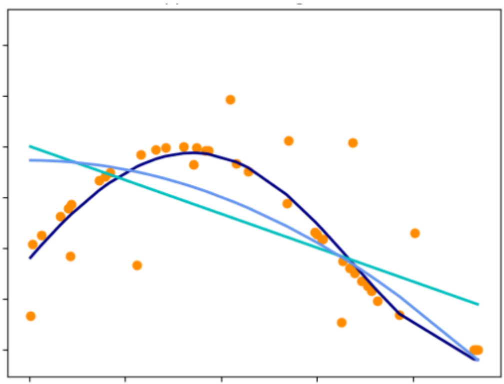
Reinforcement learning

- **Reinforcement learning** setup:
  - a set of environment and agent states S;
  - a set of actions A, of the agent
  - $P(s_{t+1} = s'|s_t = s, a_t = a)$ is the probability of transition from state s to state s' under action a.
  - $R_a(s, s')$ is the (expected) immediate reward after transition from s to s' with action a.
  - rules that describe what the agent observes
    - full / partial observability
- Well-suited to problems which include a long-term versus short-term reward trade-off
- Applications: robot control, elevator scheduling, games (chess, go), etc.

Table of Contents

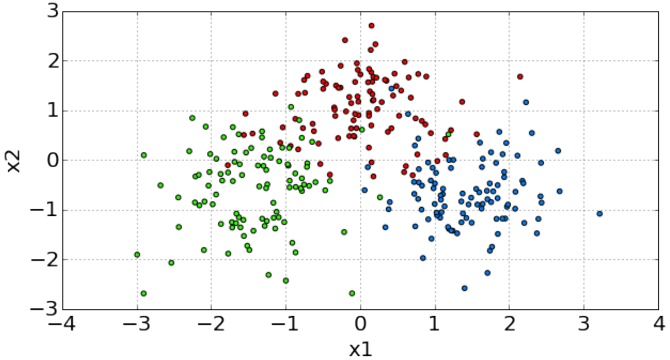
2 Visual examples
- Supervised learning
- Unsupervised learning

Regression



Predict $y \in \mathbb{R}$ for any x.

Classification



Predict class y shown with color for any point.

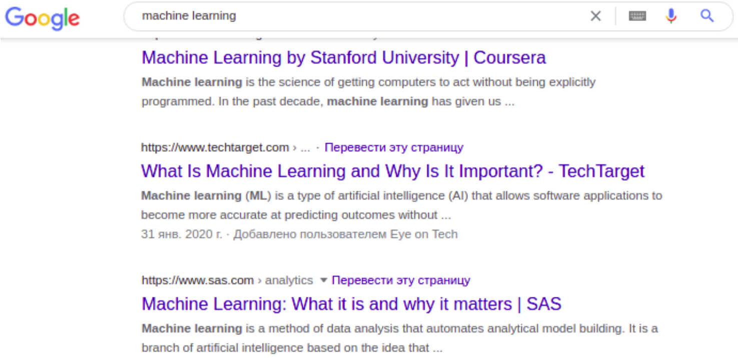
Example: medical applications

- **Objects**: patients
- **Features**:
    - real: age, pulse, blood pressure, hemoglobin content in the blood, drug dose.
    - binary: gender, presence of headache, weakness, nausea
    - categorical: past illnesses
    - ordinal: severity of the condition
- **Possible targets**:
    - **classification**: determine the type of disease, method of treatment.
    - **regression**: duration of treatment and recovery.

Example: Predicting Customer Behavior

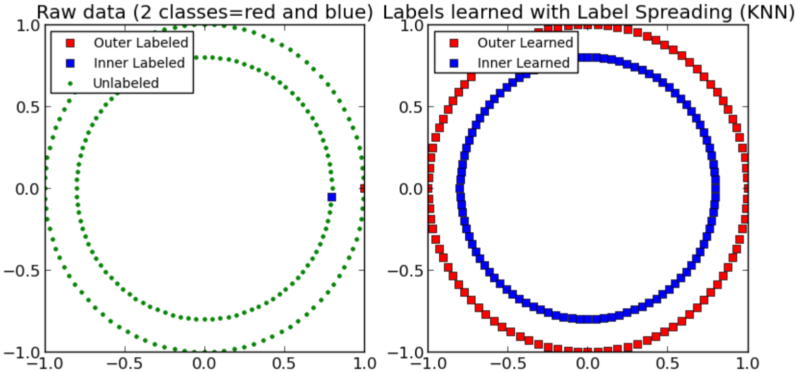
- **Objects**: the client at the current time.
- **Features**:
  - real: age, historical frequency of use of services and spending on services.
  - binary: gender, whether there were arrears in payments.
  - categorical: what services does he use.
  - ordinal: assessment of the company according to the client.
- **Possible targets**:
  - **classification**: will the client go to competitors? will the service be enabled?
  - **regression**: how many times will the service be used? how much money will be deposited into the account?

Supervised Learning - Ranking



How could the ranking problem be solved through regression or classification?

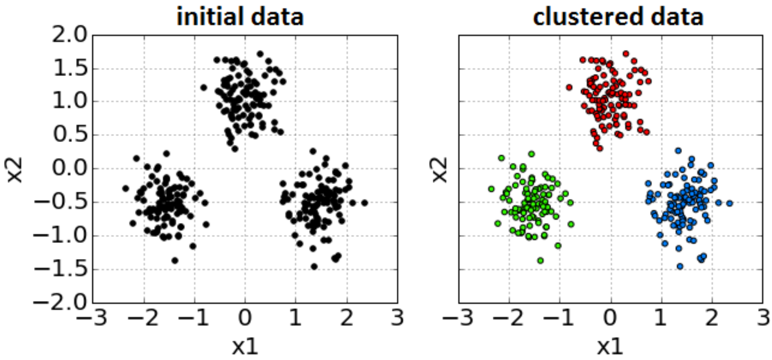
Semi-supervised classification



Semi-supervised learning.

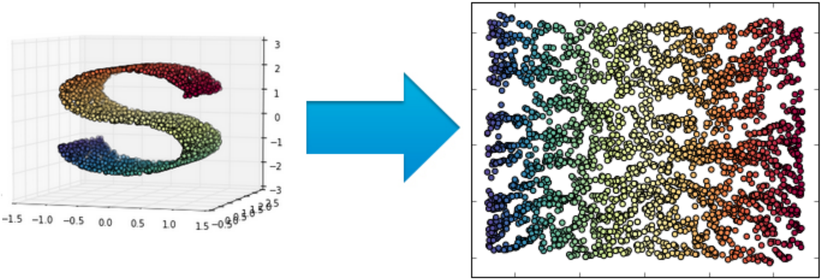
2. Visual examples
   - Supervised learning
   - Unsupervised learning

Clustering



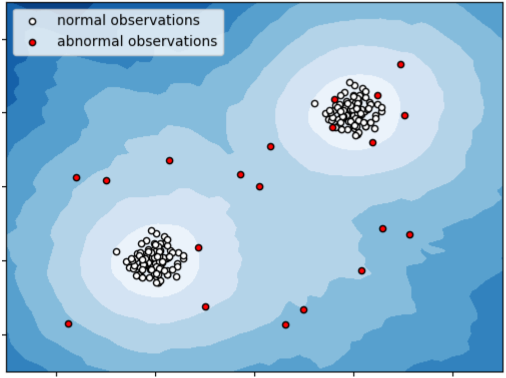
Cluster points into distinct similarity groups.

Dimensionality reduction



Reduce dimension from 3D to 2D with minimal distortion.

Outlier detection



Detect untypical observations.

Search for association rules (rule induction)

rhs	lhs	support	confidence	lift
bottled beer	liquor, red/blush wine	0.0016268429	0.9411765	19.53
	liquor, red/blush wine, soda	0.0006100661	1.0000000	20.75
	liquor, soda	0.0010167768	0.7692308	15.96
bottled water	bottled beer, misc. beverages	0.0005083884	0.5000000	8.29
citrus fruit	meat, turkey	0.0004067107	0.5714286	6.97
coffee	condensed milk, sugar	0.0004067107	0.8000000	25.71
frankfurter	liver loaf, sausage	0.0005083884	0.5000000	8.48
liquor	bottled beer, red/blush wine	0.0016268429	0.6666667	81.96
	bottled beer, red/blush wine, s...	0.0006100661	1.0000000	122.94
	red/blush wine, soda	0.0006100661	0.5454545	67.06

Market basket analysis.

Using sets $\{a, b, c\}, \{a, d, e\}, \{a, b\}, \{a, b, g, h\}$ generate rules: $a \to b,\ \ b \to a, \ldots$

General problem statement

- We want to find $f(x): X \to Y$.
- How it may be used:
  - prediction of Y
  - qualitative analysis, understanding of $X \to Y$ dependency
  - untypical objects detection (where model fails)
- Questions solved in ML:
  - what target y we are predicting?
  - how to select object descriptors (features) x?
  - what is the kind of mapping f?
  - in what sense a mapping f should approximate true relationship?
  - how to tune f?

Types of target variable (supervised learning)[3]

- $\mathcal{Y} = \mathbb{R}$ - regression
    - e.g. flat price
- $\mathcal{Y} = \mathbb{R}^M$ - vector regression
    - e.g. stock price dynamics
- $\mathcal{Y} = \{\omega_1, \omega_2, ...\omega_C\}$ - classification.
    - C=2: binary classification.
        - e.g. spam / not spam
    - C>2: multi-class classification
        - e.g. identity recognition, activity recognition
- $\mathcal{Y}$ - set of all sets of $\{\omega_1, \omega_2, ...\omega_C\}$ - labeling.[2]
    - e.g. news categorization

[2] How to solve labeling using classification?
[3] Actually any type is possible. Listed are most common types.

Table of Contents

Linear function class

- **Regression**: $\widehat{y} = g(x), \quad g(x)$ parametrized by θ.

[4] Are discriminant functions uniquely defined for fixed mapping $X \to Y$?

Linear function class

- **Regression**: $\widehat{y} = g(x), \quad g(x)$ parametrized by θ.
- **Milciclass classifier** $(y \in \{1, 2, \ldots C\})$[4]:

$$\widehat{y}(x) = \arg\max_{c} g_c(x), \quad g(x) \text{ parametrized by } \theta.$$

$$\{x : g_i(x) = g_j(x)\}, \quad \text{border between classes } i, j.$$

$$M(x, y) = g_y(x) - \max_{c \neq y} g_c(x), \quad \text{margin (quality of classification)}$$

[4] Are discriminant functions uniquely defined for fixed mapping $X \to Y$?

Linear function class

- **Regression**: $\widehat{y} = g(x), \quad g(x)$ parametrized by θ.
- **Milciclass classifier** $(y \in \{1, 2, ... C\})$[4]:

$$\widehat{y}(x) = \arg\max_{c} g_c(x), \quad g(x) \text{ parametrized by } \theta.$$

$$\{x : g_i(x) = g_j(x)\}, \quad \text{border between classes } i, j.$$

$$M(x, y) = g_y(x) - \max_{c \neq y} g_c(x), \quad \text{margin (quality of classification)}$$

- **Binary classifier** $(y \in \{+1, -1\})$:

$$\widehat{y}(x) = \arg\max_{c \in \{+1, -1\}} g_c(x) = \text{sign}\left(g_{+1}(x) - g_{-1}(x)\right) = \text{sign}\left(g(x)\right)$$

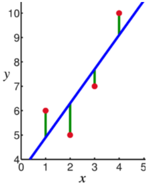
$$M(x, y) = g_y(x) - g_{-y}(x) = y\left(g_{+1}(x) - g_{-1}(x)\right) = yg(x)$$

[4] Are discriminant functions uniquely defined for fixed mapping $X \to Y$?

Examples

linear regression $y \in \mathbb{R}$:

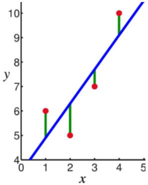
$$f(x|\theta) = \theta_0 + \theta_1 x$$



Examples

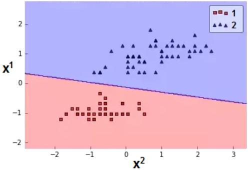
linear regression $y \in \mathbb{R}$:

$$f(x|\theta) = \theta_0 + \theta_1 x$$



linear classification $y \in \{1, 2\}$:

$$g_c(\mathbf{x}|\theta) = \theta_c^0 + \theta_c^1 x^1 + \theta_c^2 x^2,\ c = 1, 2.$$
$$f(\mathbf{x}|\theta) = \arg\max_c g_c(x|\theta)$$



Score versus loss

- In machine learning predictions, functions, objects can be assigned:
    - **score, rating** - this should be maximized
    - **loss, cost** - this should be minimized[5]

[5] how can one convert score$\leftrightarrow$loss?

Loss function $\mathcal{L}(\widehat{y}, y)$[6]

- Examples:
  - **classification:**
    - misclassification rate

$$\mathcal{L}(\widehat{y}, y) = \mathbb{I}[\widehat{y} \neq y]$$

  - **regression:**
    - MAE (mean absolute error):

$$\mathcal{L}(\widehat{y}, y) = |\widehat{y} - y|$$

    - MSE (mean squared error):

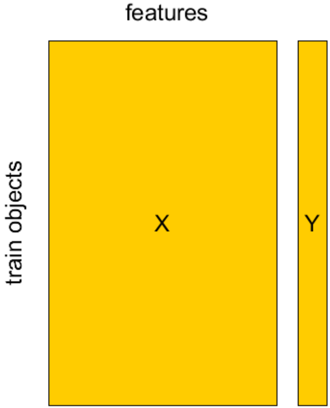
$$\mathcal{L}(\widehat{y}, y) = (\widehat{y} - y)^2$$

[6] Selecting realistic loss is not trivial. Consider e.g. demand forecasting.

Table of Contents

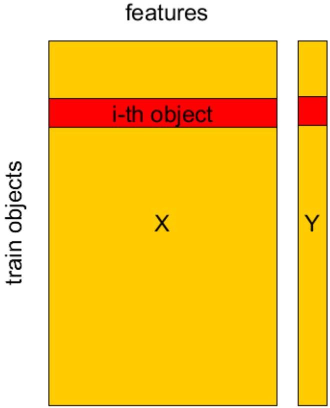
Known set

Known set: $(\mathbf{x}_1, y_1), ...(\mathbf{x}_M, y_M)$,
design matrix $X = [\mathbf{x}_1, ...\mathbf{x}_M]^T$, $Y = [y_1, ...y_M]^T$.



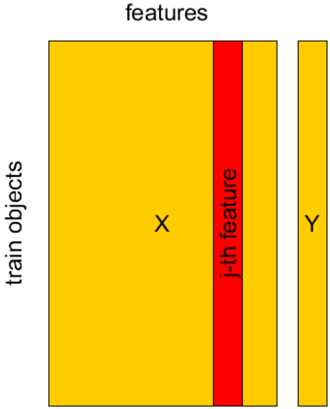
Known set

Known set: $(\mathbf{x}_1, y_1), ...(\mathbf{x}_M, y_M)$,
design matrix $X = [\mathbf{x}_1, ...\mathbf{x}_M]^T$, $Y = [y_1, ...y_M]^T$.

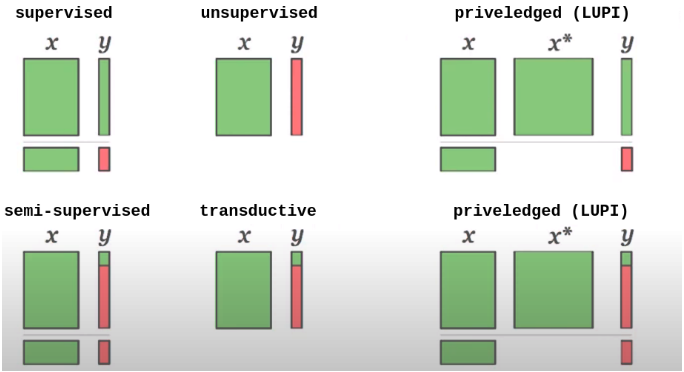


Known set

Known set: $(\mathbf{x}_1, y_1), ...(\mathbf{x}_M, y_M)$,
design matrix $X = [\mathbf{x}_1, ...\mathbf{x}_M]^T$, $Y = [y_1, ...y_M]^T$.



Types of learning



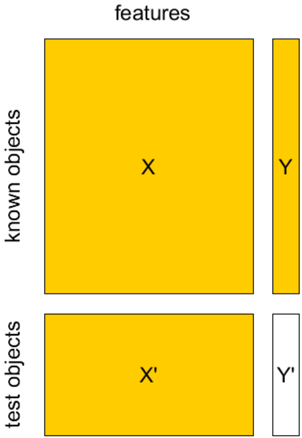
LUPI - Learning Using Priveledged Information[7].

---

[7] Vapnik V., Vashist A. A new learning paradigm: Learning Using Priveledged Information // Neural Networks. 2009.

Known set, test set

- Known sample X, Y: $(\mathbf{x}_1, y_1), ...(\mathbf{x}_M, y_M)$
- Test sample X', Y': $(\mathbf{x}'_1, y'_1), ...(\mathbf{x}'_K, y'_K)$



Empirical risk

- Want to minimize *expected risk*:

$$\int\int \mathcal{L}(f_\theta(\mathbf{x}), y)p(\mathbf{x}, y)d\mathbf{x}dy \to \min_\theta$$

[8]We assume that objects are i.i.d.

Empirical risk

- Want to minimize *expected risk*:

$$\int\int \mathcal{L}(f_\theta(\mathbf{x}), y)p(\mathbf{x}, y)d\mathbf{x}dy \to \min_\theta$$

- Can minimize only *empirical risk*[8]:

$$L(\theta|X, Y) = \frac{1}{N}\sum_{n=1}^{N}\mathcal{L}(f_\theta(\mathbf{x}_n), y_n)$$

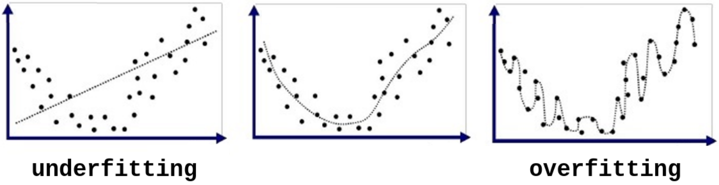
- Method of empirical risk minimization:

$$\widehat{\theta} = \arg\min_\theta L(\theta|X, Y)$$

[8] We assume that objects are i.i.d.

Underfitting and overfitting

- Underfitting: too simple model for true dependency.
    - doesn't account for subtle dependencies
- Overfitting: too complex model for true dependency.
    - adjusts for noise in measurements



Estimation of empirical risk

- What is the relationship between $L(\widehat{\theta}|X,Y)$ and $L(\widehat{\theta}|X',Y')$?

Estimation of empirical risk

- What is the relationship between $L(\widehat{\theta}|X,Y)$ and $L(\widehat{\theta}|X',Y')$?
- Typically
$$L(\widehat{\theta}|X,Y) < L(\widehat{\theta}|X',Y')$$
- How to get realistic estimate of $L(\widehat{\theta}|X',Y')$?

Estimation of empirical risk

- What is the relationship between $L(\widehat{\theta}|X, Y)$ and $L(\widehat{\theta}|X', Y')$?
- Typically

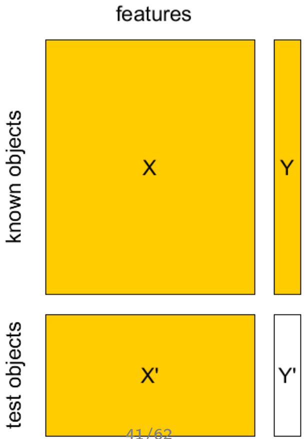
$$L(\widehat{\theta}|X, Y) < L(\widehat{\theta}|X', Y')$$

- How to get realistic estimate of $L(\widehat{\theta}|X', Y')$?
  - separate **validation set**
  - **cross-validation**
  - **leave-one-out** method

4. Function estimation
    - Separate validation set
    - Cross-validation
    - A/B testing

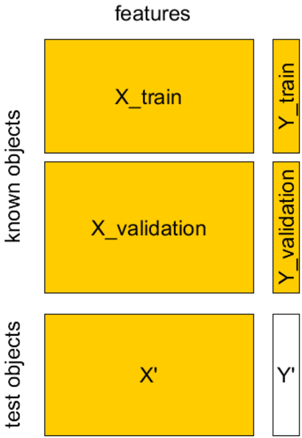
Separate validation set

- Known sample X, Y: $(\mathbf{x}_1, y_1), ...(\mathbf{x}_M, y_M)$
- Test sample X', Y': $(\mathbf{x}'_1, y'_1), ...(\mathbf{x}'_K, y'_K)$



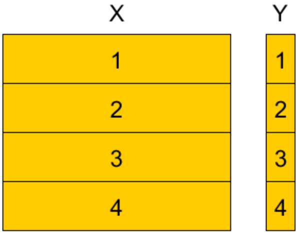
Separate validation set

Divide known set randomly or randomly with stratification:



4. Function estimation
   - Separate validation set
   - Cross-validation
   - A/B testing

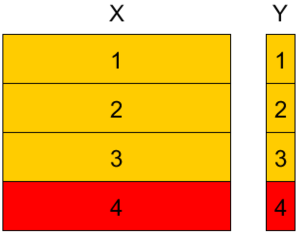
4-fold cross-validation example



Divide training set into K parts, referred as «folds» (here $K = 4$).
Variants:

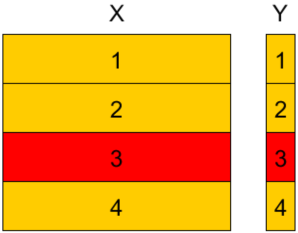
- randomly
- randomly with stratification (w.r.t target value or feature value).

4-fold cross validation example



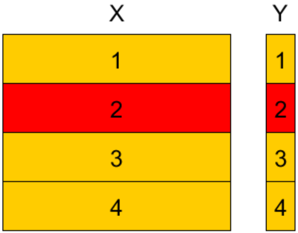
Use folds 1,2,3 for model estimation and fold 4 for model evaluation.

4-fold cross validation example



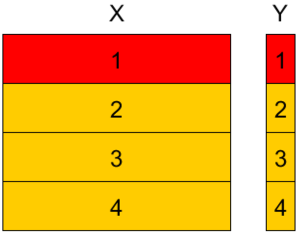
Use folds 1,2,4 for model estimation and fold 3 for model evaluation.

4-fold cross validation example



Use folds 1,3,4 for model estimation and fold 2 for model evaluation.

4-fold cross validation example



Use folds 2,3,4 for model estimation and fold 1 for model evaluation.

4-fold cross validation example

- Denote
  - $k(n)$ - fold to which observation $(\mathbf{x}_n, y_n)$ belongs to: $n \in I_k$.
  - $\widehat{\theta}^{-k}$ - parameter estimation using observations from all folds except fold k.

[9] will samples be correlated?

4-fold cross validation example

- Denote
  - $k(n)$ - fold to which observation $(\mathbf{x}_n, y_n)$ belongs to: $n \in I_k$.
  - $\widehat{\theta}^{-k}$ - parameter estimation using observations from all folds except fold k.

Cross-validation empirical risk estimation

$$\widehat{L}_{total} = \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(f_{\widehat{\theta}^{-k(n)}}(x_n), y_n)$$

[9] will samples be correlated?

4-fold cross validation example

- Denote
  - $k(n)$ - fold to which observation $(\mathbf{x}_n, y_n)$ belongs to: $n \in I_k$.
  - $\widehat{\theta}^{-k}$ - parameter estimation using observations from all folds except fold k.

Cross-validation empirical risk estimation

$$\widehat{L}_{total} = \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(f_{\widehat{\theta}^{-k(n)}}(x_n),\ y_n)$$

- For K-fold CV we have:
  - K parameters $\widehat{\theta}^{-1}, ...\widehat{\theta}^{-K}$
  - K models $f_{\widehat{\theta}^{-1}}(\mathbf{x}), ...f_{\widehat{\theta}^{-K}}(\mathbf{x})$.
    - can use ensembles
  - K estimations of empirical risk: $\widehat{L}_k = \frac{1}{|I_k|} \sum_{n \in I_k} \mathcal{L}(f_{\widehat{\theta}^{-k}}(\mathbf{x}_n),\ y_n),\ k = 1, 2, ...K.$
    - can estimate variance & use statistics![9]

[9] will samples be correlated?

Comments on cross-validation

- When number of folds K is equal to number of objects N, this is called **leave-one-out method**.
- Cross-validation uses the i.i.d.[10] property of observations
- Stratification by target y helps for imbalanced/rare classes.

[10] i.i.d.=independent and identically distributed

4 Function estimation

- Separate validation set
- Cross-validation
- A/B testing

A/B testing

- Observe test set **after the models were built**.
- A/B testing procedure:
    1. divide test objects randomly into two groups - A and B.
    2. apply base model to A
    3. apply modified model to B
    4. compare final results

A/B testing

- Observe test set **after the models were built**.
- A/B testing procedure:
    1. divide test objects randomly into two groups - A and B.
    2. apply base model to A
    3. apply modified model to B
    4. compare final results

Cross-validation vs. A/B testing

Comparison of cross-validation and A/B test:

	cross-validation	**A/B test**
realism	use retrospective analysis, rely on i.i.d. assumption	full realism
overfitting	possible (when use it multiple times)	almost impossible (possible if A/B split is inadequate)
costs	uses available data, only computational costs	requires time and resources for collecting & evaluating feedback from objects of groups A and B

- When forecast affects true outcome (e.g. in recommender system) A/B test is more adequate.

General modelling pipeline[11]

- Understand business problem
- Problem formalization
- **Data collection**
- **Data preprocessing**
- **Modelling**
- Model evaluation
- Deployment
- Maintenance

[11] Steps covered in kaggle competitions.

Examples of ML applications by domain

- WEB
  - Web-page ranking
  - Spam filtering
    - e-mails, web pages in search results
- Computer networks
  - Authentication systems
    - by voice, face, fingerprint
    - by behavior
  - Intrusion detection
- Business
  - Fraud detection
  - Churn prediction
- Banking
  - Credit scoring
  - Stock prices forecasting
  - Risks estimation

Examples of ML applications by data type

- Texts
  - Document classification
  - POS tagging, semantic parsing,
  - named entities detection
  - sentimental analysis
  - automatic summarization
- Images
  - Handwriting recognition
  - Face detection, pose detection
  - Person identification
  - Image classification
  - Image segmentation
  - Adding artistic style
- Other
  - Target detection / classification
  - Particle classification

Recommended materials

- **The Elements of Statistical Learning: Data Mining, Inference, and Prediction.** Trevor Hastie, Robert Tibshirani, Jerome Friedman, 2nd Edition, Springer, 2009.
- **Data Mining: The Textbook.** Charu C. Aggarwal, Springer, 2015.
- **Statistical Pattern Recognition.** 3rd Edition, Andrew R. Webb, Keith D. Copsey, John Wiley & Sons Ltd., 2011.
- Any additional public sources:
  - wikipedia, articles, tutorials, video-lectures.
- Practical questions:
  - StackOverflow, scikit-learn documentation, kaggle forum.

Table of Contents

Notation used in the course[12]

- **Objects and outputs:**
  - x - vector of known input characteristics of an object
  - y - predicted target characteristics of an object specified by x
  - x_i - i-th object of a set, y_i - corresponding target characteristic
  - x^k - k-th feature of object specified by x
  - x_i^k - k-th feature of object specified by x_i
- **General definitions:**
  - D - dimensionality of the feature space: $x \in \mathbb{R}^D$
  - N - the number of objects in the training set
  - C - total number of classes in classification.
  - Possible classes: $\{1, 2, ...C\}$ or $\{\omega_1, \omega_2, ...\omega_C\}$

[12] If this corresponds the context and there are no redefinitions

Notation used in the course

- **Training set:**
  - X - design matrix, $X \in \mathbb{R}^{NxD}$
  - $Y \in \mathbb{R}^N$ - target characteristics of a training set
- **Optimization:**
  - $\mathcal{L}(\widehat{y}, y)$ - loss function for 1 object
    - y is the true value and $\widehat{y}$ is the predicted value.
  - $L(\theta) = \sum_{n=1}^{N} \mathcal{L}(f_\theta(x_n), y_n)$ loss function for the whole the training set.

Notation used in the course

- **Special functions:**
  - $[x]_+ = \max\{x, 0\}$
  - $\mathbb{I}[\text{condition}] = \begin{cases} 1, & \text{if condition is satisfied} \\ 0, & \text{if condition is not satisfied} \end{cases}$
  - $\text{sign}(x) = \begin{cases} 1, & x \geq 0 \\ -1, & x < 0 \end{cases}$
- **Other definitions:**
  - $\widehat{z}$ defines an estimate of z, based on the training set: for example, $\widehat{\theta}$ is the estimate of θ, $\widehat{y}$ is the estimate of y, etc.
  - r.v.=random variable, w.r.t.=with respect to, e.g.=for example.
  - $A \succcurlyeq 0$ means that A is a square positive semi-definite matrix.
- All vectors are vectors-columns, e.g. if $x \in \mathbb{R}^D$ its dimensions are $Dx1$.

Summary

- Machine learning algorithms reconstruct relationship between features x and outputs y.
- Relationship is reconstructed by optimal function $\widehat{y} = f_{\widehat{\theta}}(x)$ from function class $\{f_\theta(x),\, \theta \in \Theta\}$.
- θ is particular controls model complexity, models may be too simple and too complex.
- $\widehat{\theta}$ selected to minimize empirical risk $\frac{1}{N}\sum_{n=1}^{N}\mathcal{L}(f_\theta(x_n), y_n)$ for some loss function $\mathcal{L}(\widehat{y}, y)$.
- Overfitting - non-realistic estimate of expected loss on the training set.
- To avoid overfitting - use validation sets, cross-validation, A/B test.