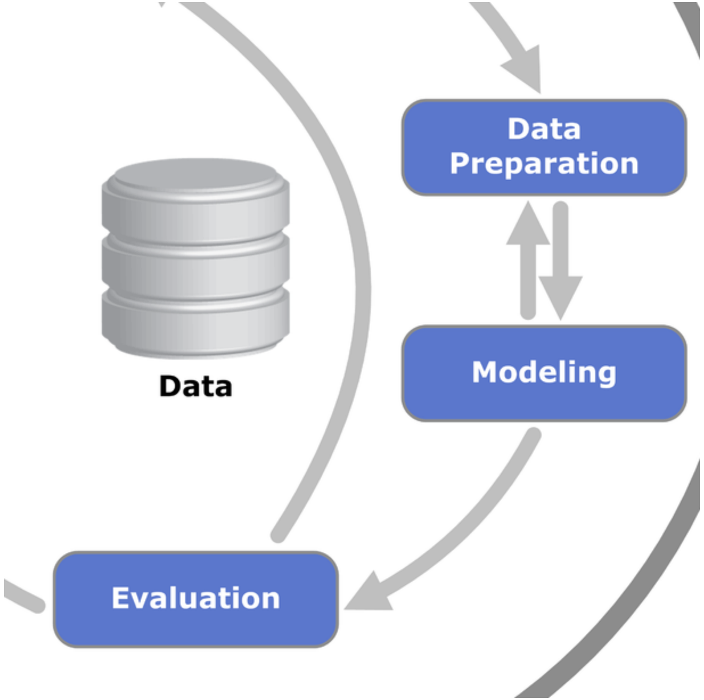
NDBI048 – Data Science
Modeling overview

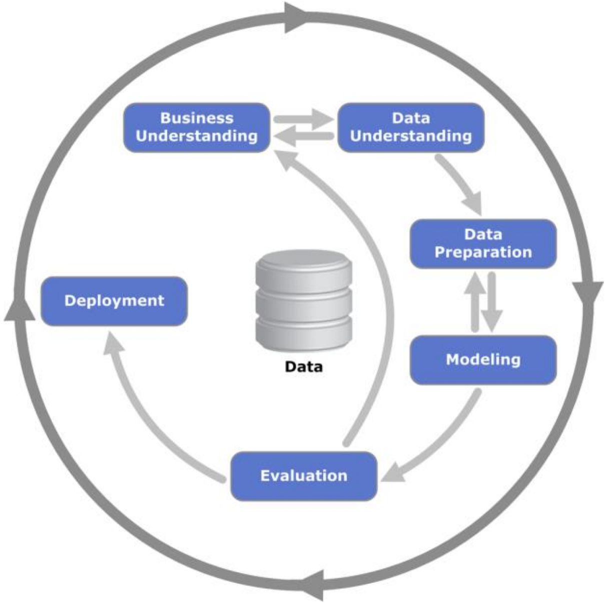
Jan Hučín

19. 11. 2025

Where we are now

PROFINIT





Outline

PROFINIT

1. analytical and modelling approach
2. basic terms, types of models
3. data for modelling: train, test, validation
4. model evaluation and limits of metrics
5. model selection
6. feature selection
7. some model methods

Analytical (inferential) approach

unit (human, animal, picture, action, proces, ...)

- I have **data** about it
- I have data about other features
  - day, time, salary, body height etc.

I want to: understand the world & make conclusions.

Dependent Variable — Population Y intercept — Population Slope Coefficient — Independent Variable — Random Error term

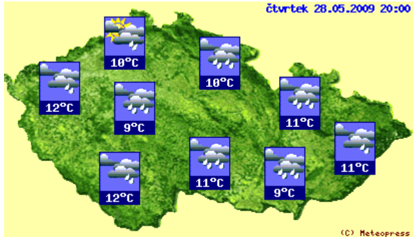
$$Y_i = \underbrace{\beta_0 + \beta_1 X_i}_{\text{Linear component}} + \underbrace{\varepsilon_i}_{\text{Random Error component}}$$

Modeling approach

unit (human, animal, picture, action, proces, ...)

- has (or will have) a **feature I don‘t know now**
  - Man, or woman? Fair, or deceit? Age? How many ºC? Which of kinds?
- but I know something else
  - living place; history; behaviour; body measures etc.

I want to: estimate / classify / predict the unit‘s unknown feature.



Linear regression and two approaches

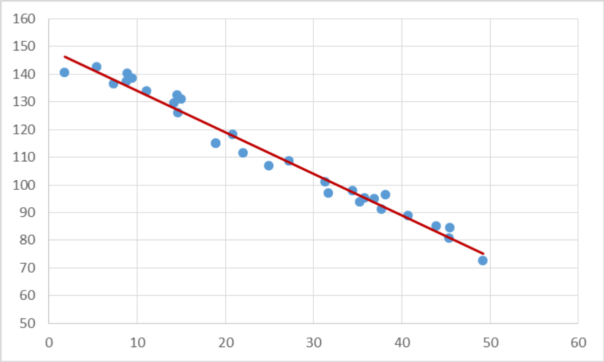
PROFINIT

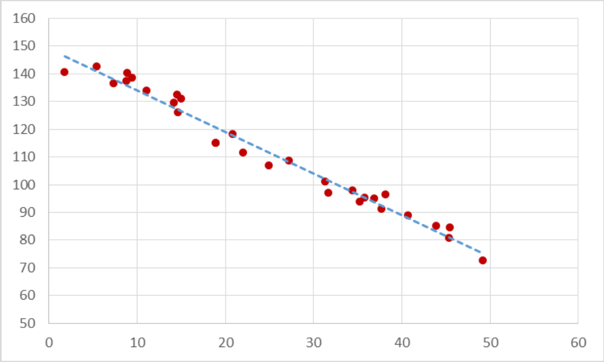
analytical approach:

- interested in trend (inferention)
- *„How does the world work?“*
- getting β is a goal

modeling approach:

- interested in points (estimate, prediction)
- *„If I have this value of x, how many will be y?“*
- getting β is a mean





Get the difference

PROFINIT

Analytical approach

- I have data.
- Trying to describe the world.
- **Fitting relations in data.**
- outcome = explanation (inference).

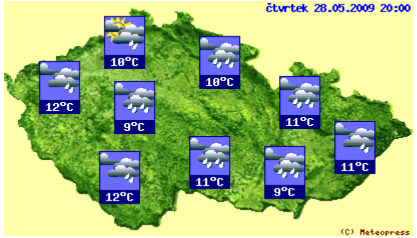
Modeling approach

- I have a problem.
- Trying to get any info.
- **Fitting relations in data.**
- outcome = prediction, classification etc.

Same methods, same mathematics, different aims.

Basic terms

- „unknown feature“ = **target, response**
- „what I know“ = **feature, predictor, explanatory var.**
- „result“ = **prediction, classification** etc. (see later)
- „how we get the result“ = **modeling method**
- „data for modelling“ = **dataset, model matrix**



Types of models

PROFINIT

- By info about target (data labeling)
  - yes, for enough & representative units → **supervised model**
  - yes, but for few or non-representative units → **semi-supervised model**
  - no → **unsupervised model**

- By target type
  - binary
  - categorical (ordinal, non-ordinal)
  - numerical

- Used method?
  - linear (regression etc.)
  - rule-based (decision trees etc.)
  - similarity (kNN etc.)
  - „blackbox“ (neural networks, gradient boosting etc.)

From now: **supervised** and mostly **binary** models.

Data and dataset

PROFINIT

- Data need to be **understood** and **prepared**.
- **Training dataset** = table (matrix):
  - columns = id, target(s), features
  - rows = units
- Dataset division:
  - **train** set – where we **fit** a model
  - **test** set – where we **evaluate** a model
- **Validation dataset**
  - where we **prove** the model is good
- see later

Image Id	No. of Exudates	Area of Largest Span	Largest Spot Major Axis	Largest Spot Minor Axis	yellowness
Image001	12	1061	45.97	29.82	0.55
Image002	11	413	25.92	20.60	0.44
Image003	19	880	44.03	25.96	0.62
Image004	10	530	28.57	24.51	0.15
Image005	4	536	28.23	24.96	0.44
Image006	13	338	25.80	17.29	0.50
Image007	18	14809	169.96	113.87	0.59
Image008	8	5997	152.68	53.95	0.59
Image009	9	1967	64.20	40.08	0.16
Image010	10	4161	99.38	54.83	0.62
Image011	28	4023	86.23	61.83	0.64
Image012	21	630	53.66	15.68	0.62
Image013	8	1748	57.13	39.38	0.67
Image014	15	1079	62.78	22.55	0.53
Image015	12	383	27.64	18.43	0.54
Image016	20	1175	53.81	28.20	0.57
Image017	10	694	36.29	25.36	0.61
Image018	24	1601	71.12	29.01	0.64
Image019	4	535	28.47	24.81	0.61
Image020	11	626	35.54	23.97	0.57

Model evaluation

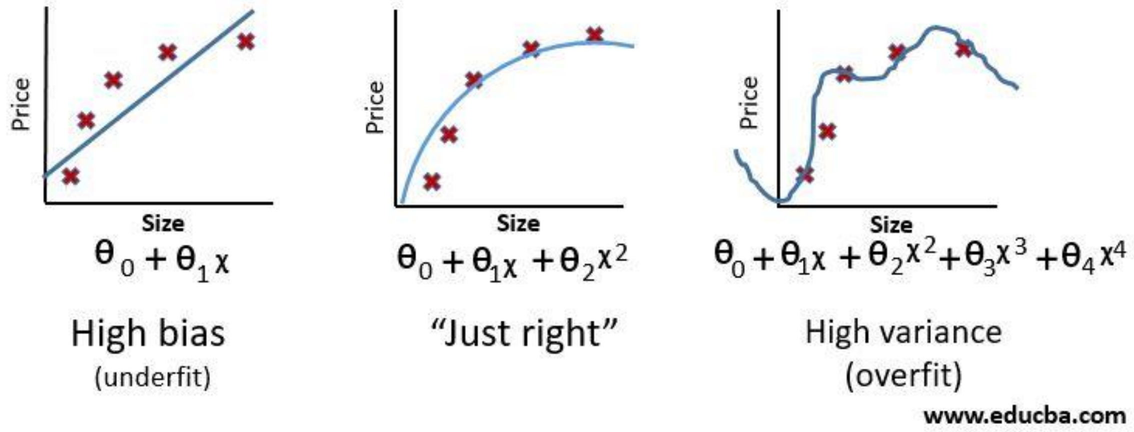
PROFINIT

~~How well does my model fit my past data?~~

How well would I predict (classify, estimate) in reality?

› Can‘t evaluate on the same data as for fit → **overfitting**

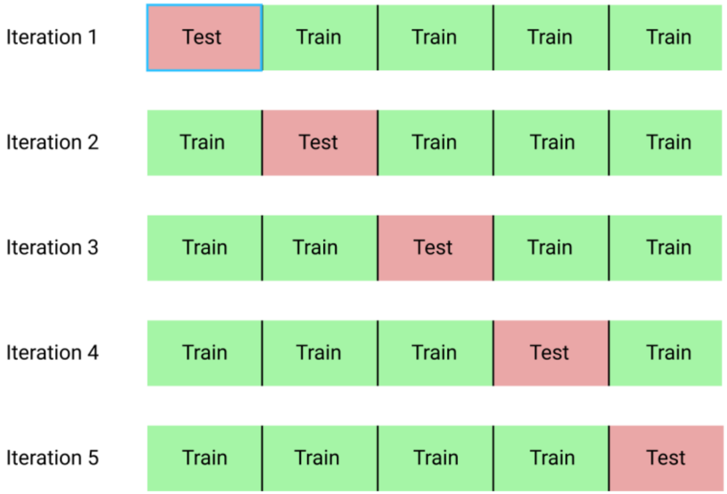
› I need to „simulate unknown reality“→ **cross-validation**



Cross-validation

PROFINIT

- Take 1/*k* of data as the **test** set, the rest as the **train** set.
- Fit on train set, predict on the test set.
- Repeat for each 1/*k* of data.
- Now we have predicted values for all units. Compare with actual target values.
- *Variation: compare for each 1/k separately and aggregate metrics.*



Model evaluation: metrics (binary target)

PROFINIT

- technical (logLik, R^2)
- analytics (ROC AUC, prediction SD)
- business (expected profit/loss)

id	predicted	actual target
1	0.34	1
2	0.76	0
3	0.04	0
4	0.29	0
5	0.88	1
...	...	...

Model evaluation: metrics (binary target)

PROFINIT

confusion matrix

- give a threshold for pos/neg prediction
- similar to hypothesis testing (error type I, II)
- **recall** (true positive rate) = $\frac{TP}{TP+FN}$
- **sensitivity** = recall
- **precision** = $\frac{TP}{TP+FP}$
- **specificity** (true negative rate) = $\frac{TN}{TN+FP}$
- **false positive rate** = $\frac{FP}{TN+FP}$, **false negative rate** = $\frac{FN}{TP+FN}$
- **accuracy** = $\frac{TP+TN}{TP+FN+FP+TN}$

	predicted true	predicted false
actual true	TP	FN
actual false	FP	TN

Model evaluation: metrics (binary target)

PROFINIT

confusion matrix in one number

- Phi coefficient
  (also Matthews corr. coef., *MCC*)
- $$\frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FN)(FP+TN)(TP+FP)(FN+TN)}}$$

	predicted true	predicted false
actual true	TP	FN
actual false	FP	TN

Model evaluation: confusion matrix, example

PROFINIT

- **recall** (true positive rate), **sensitivity** = $= \frac{TP}{TP+FN} = 0.8$
- **precision** $= \frac{TP}{TP+FP} = 0.5$
- **specificity** (true negative rate) =
- $= \frac{TN}{TN+FP} = \frac{11}{15} \sim 0.73$
- **false positive rate** $= \frac{FP}{TN+FP} = \frac{4}{15} \sim 0.27$
- **false negative rate** $= \frac{FN}{TP+FN} = 0.2$
- **accuracy** $= \frac{TP+TN}{TP+FN+FP+TN} = 0.75$
- **Phi** = 0.48

	pred true	pred false	total
actual true	40	10	**50**
actual false	40	110	**150**
total	**80**	**120**	**200**

Model evaluation: metrics (binary target)

PROFINIT

id	predicted probabililty	predicted (thresh 0.5)	predicted (thresh 0.3)	actual target
1	0.34	0	1	1
2	0.76	1	1	1
3	0.04	0	0	0
4	0.29	0	0	0
5	0.48	0	1	0
...	...			...

different threshold → different recall, FPR etc.

Model evaluation: metrics (binary target)

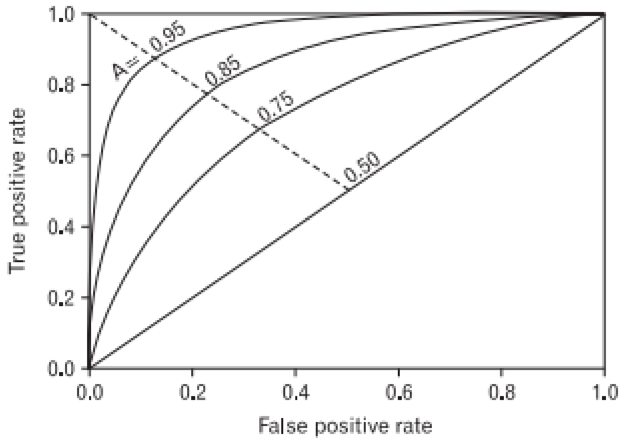
PROFINIT

Confusion matrix depends on the threshold value:

- small threshold → high recall, but high FPR too
- and vice versa

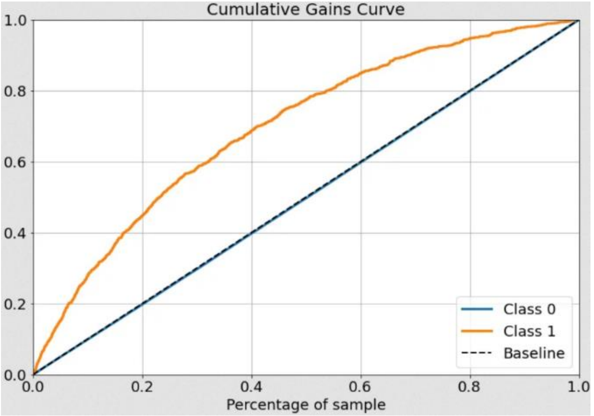
→ receiver operation curve (**ROC**)

- threshold runs 0→1
- for various thresholds, we count TPR & FPR
- we make curve of points [FPR; TPR]
- random guessing – diagonal
- perfect model – through top left
- performance: area under curve – **AUC**



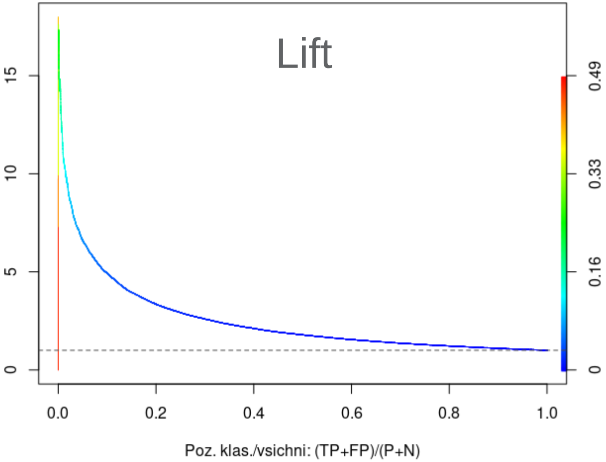
Model evaluation: metrics (binary target)

	predicted true	predicted false	
actual true	TP	FN	P
actual false	FP	TN	N
	$\hat{P}$	$\hat{N}$	S



Gain

- recall in sample by model vs. recall in random sample
- (TP / P) vs. $(\hat{P} / S)$



Lift

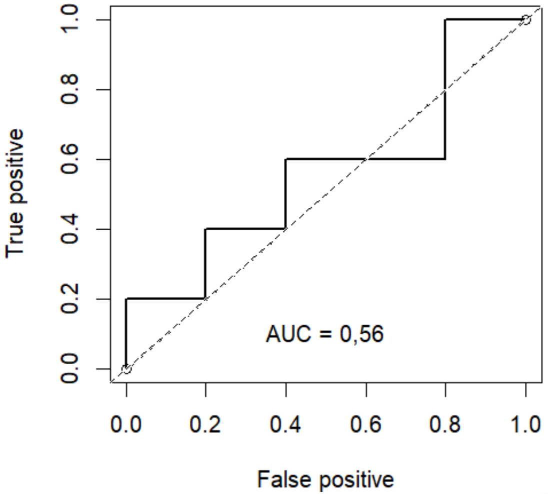
- precision in sample by model over precision in random sample
- $(\mathrm{TP} / \hat{P}) : (P / S) = (\mathrm{TP} / \hat{P}) : (P / S)$

Metric limits

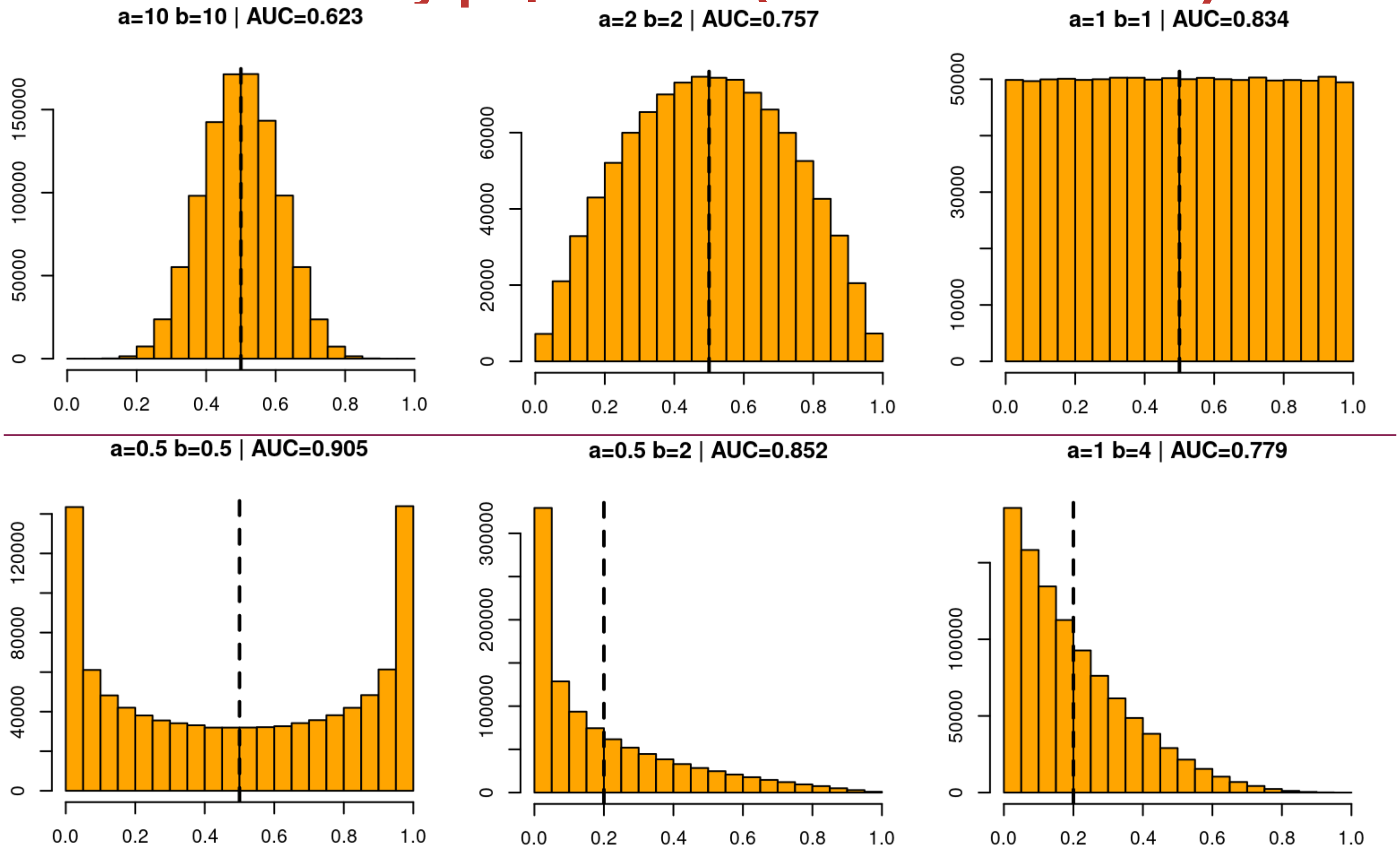
PROFINIT

0,41	0,43	0,45	0,47	0,49	0,51	0,53	0,55	0,57	0,59	P (exact)
0	0	0	0	0	1	1	1	1	1	prediction
0	1	1	0	0	1	0	1	0	1	result

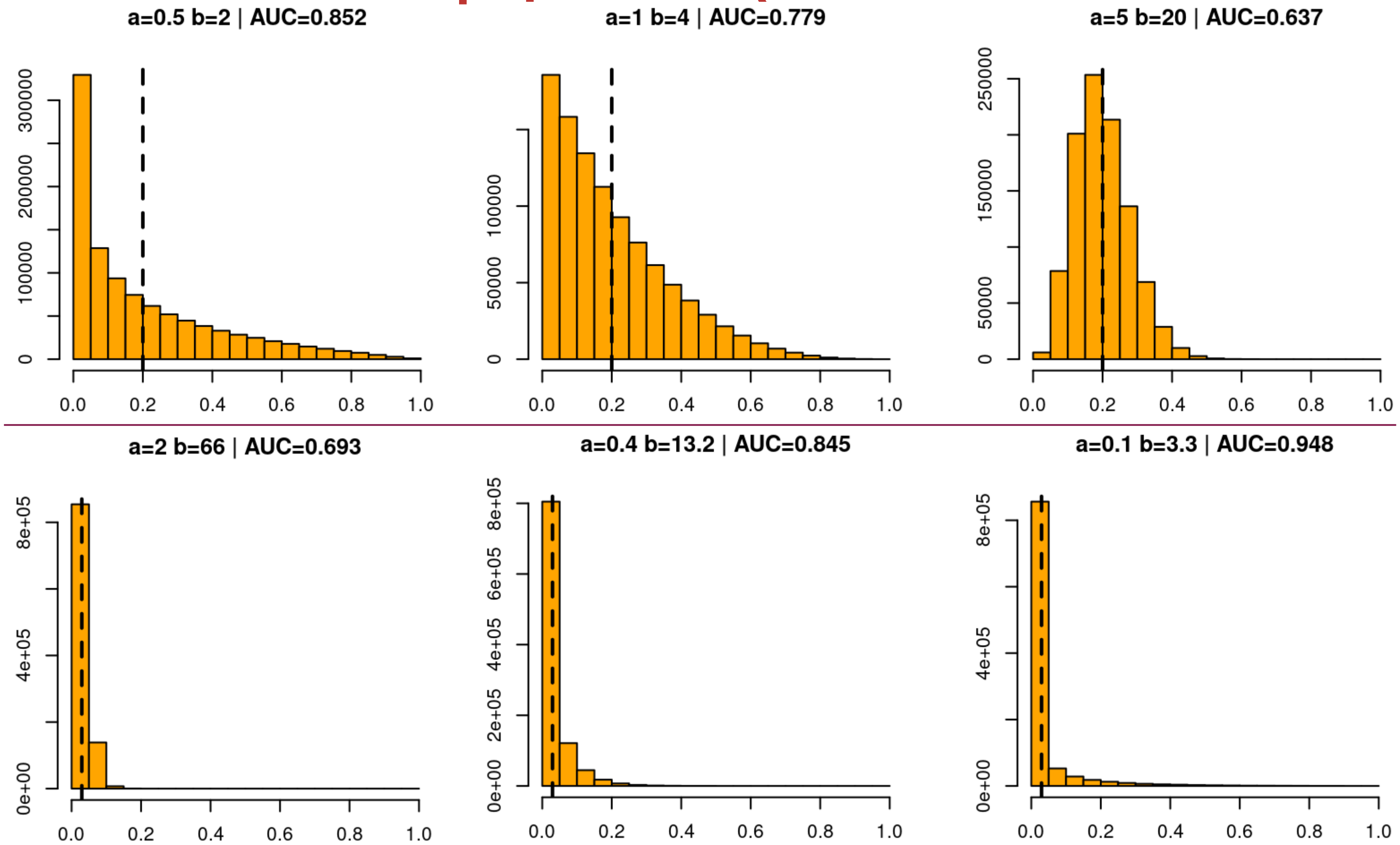
- exact probabilities but low performance
- why?
- **classification**: exact classification possible
- **prediction**: exact prediction impossible – due to randomness



Metric limits by population (beta distribution)



Metric limits by population (beta distribution)



Model requirements

PROFINIT

- meeting customer requirements
- high performance
- fast
- cheap (performance : price ratio)
- interpretable
- easy to implement and maintain **(bus factor)**

Implementation requirements

PROFINIT

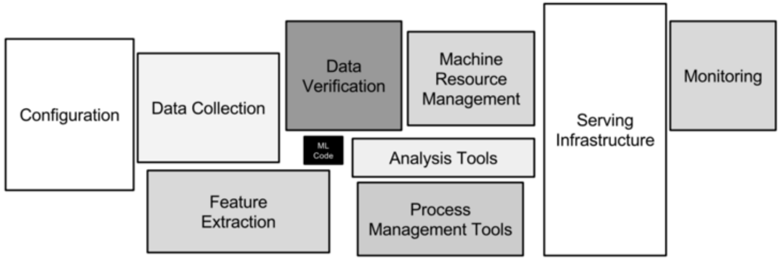
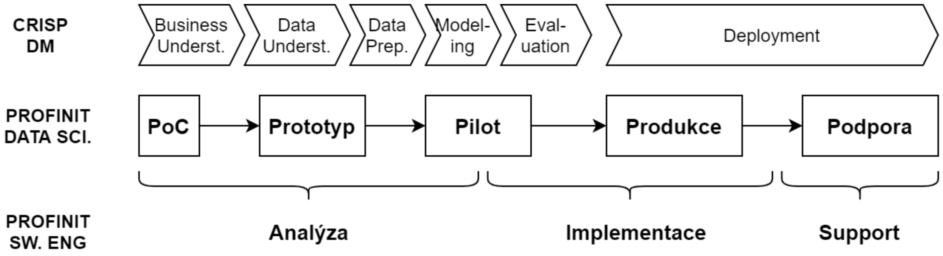


Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.



- technologies
- knowledge
- connection to the world
- maintenance
- → price

Model selection – technical base

PROFINIT

- requested mode (real-time, near real-time, batch) → **SLA**
- how much data to process for a result?
- can I / need I have something precomputed?
- is partial or approximate result allowed?
- technologies (SQL, Big Data, R/Python/C/Java)

Model selection – technical base

- quantitative change: beware of complexity ($O(N^2)$, $O(N^3)$, …)
- qualitative change: usually risky
  - technology / version change
  - workflow change
  - data format change
  - new result requirements
  - → **should be robust**
- stable (champion) vs. candidate (challenger) model
- automatic monitoring

Don‘t change a winning team.
English proverb

Model building

simple model

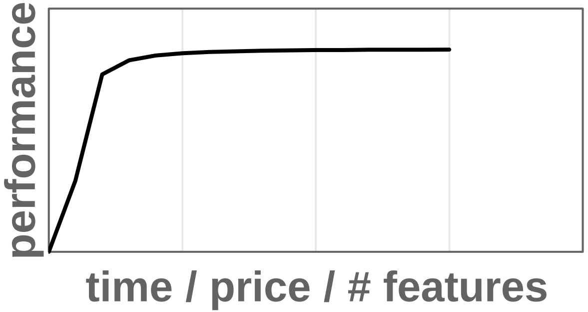
- domain knowledge
- DIY

baseline / referential model

- strong and easily available features
- simple method (regression, small tree)
- sometimes sufficient

final model

- long journey, good to automate (MLops)



Model selection – features

PROFINIT

Forward

- start from null model (intercept only)
- try a predictor & evaluate performance
- choose the one with the highest added performance, add it
- repeat until there is no performance increase

Backward

- start from full model (all predictors)
- omit a predictor & test (p-value, ML metrics)
- choose the one with highest p-value or added performance, drop it
- repeat until the performance gets worse

Model selection – forward or backward?

PROFINIT

forward

- in early steps, for referential model building
- good for simple and interpretable methods

backward

- exploration of a new feature family
- estimation of performance limit
- requires huge sources, regularization, automatized process
- good for a complex methods

Model regularization

PROFINIT

- When some predictors highly correlated – computation numerically unstable.
- Solution: prefer lower values of coefficients – penalization
- Methods: Lasso, L2 (ridge regression)

Modeling methods – linear model

PROFINIT

X = predictor matrix, Y = target, β – coefficients (parameters, effects)

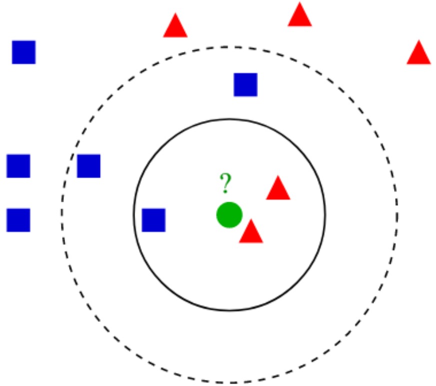
- $\mathrm{E}\boldsymbol{Y} = \boldsymbol{X\beta}$ **linear regression**
- $\mathrm{P}(\boldsymbol{Y} = 1) = \frac{e^{X\beta}}{1+e^{X\beta}}$ **logistic regression**
- „scoring model“: $\widehat{Y}_i = f\left(\sum_{j=1}^{k} \beta_j X_{ij}\right)$ – additive effects

Modeling methods – nearest neighbors

PROFINIT

Similar units will have **similar target**.

1. Train set: units with known target (labeled).
2. New unit (unknown target) arrives.
3. By some distance metric, we found *k* nearest units from the train set (nearest neighbors).
4. Estimated target = aggregation of neighbors' targets.



Distance metric: e. g. euclidean, cosine, Levenshtein…

Aggregation: voting, weighted mean, median

Modeling methods – Bayes classifier

PROFINIT

Bayes classifier

$$P(Y=C_i|\boldsymbol{X}=\boldsymbol{x}) = \frac{P(\boldsymbol{X}=\boldsymbol{x}|Y=C_i)\cdot P(Y=C_i)}{P(\boldsymbol{X}=\boldsymbol{x})}$$

- Y = target, C_i = category, $\boldsymbol{X}$ = predictors, $\boldsymbol{x}$ = observed values
- find i where $P(\boldsymbol{X}=\boldsymbol{x}|Y=C_i)\cdot P(Y=C_i)$ biggest $\rightarrow$ classification
- for binary target:

$$\frac{P(Y=1|E)}{P(Y=0|E)} = \frac{\frac{P(E|Y=1)\cdot P(Y=1)}{P(E)}}{\frac{P(E|Y=0)\cdot P(Y=0)}{P(E)}} = \frac{P(Y=1)}{P(Y=0)}\cdot\frac{P(E|Y=1)}{P(E|Y=0)}$$

Questions?