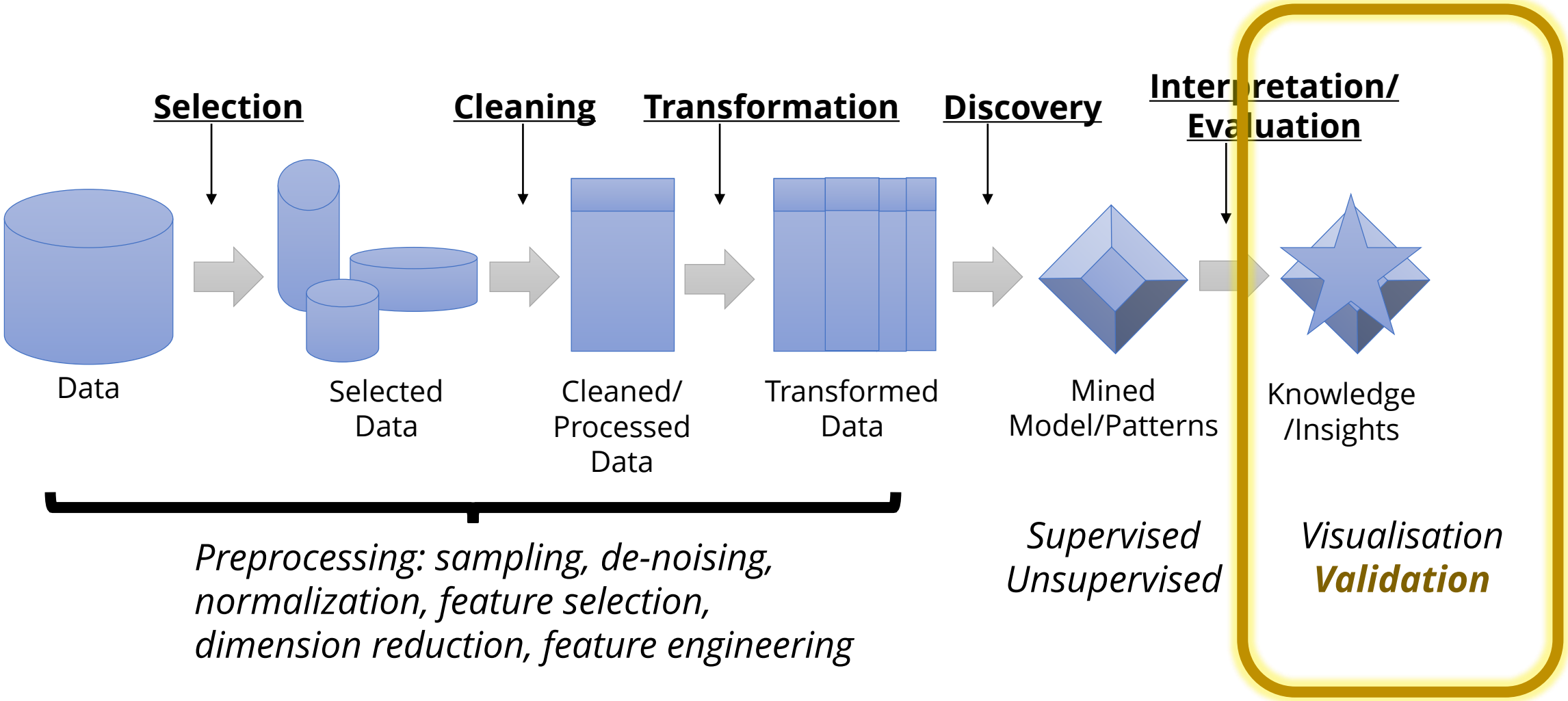
Knowledge Management and Business Intelligence

Model Evaluation

Ch 10, 11

This course


Selection
Cleaning
Transformation
Discovery
Interpretation/
Evaluation
Data
Selected
Data
Cleaned/
Processed
Data
Transformed
Data
Mined
Model/Patterns
Knowledge
/Insights
Preprocessing: sampling, de-noising,
normalization, feature selection,
dimension reduction, feature engineering
Supervised
Unsupervised
Visualisation
Validation

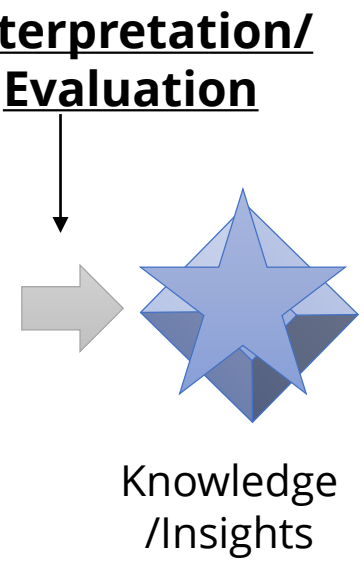
Model validation and evaluation

What to validate?

- Model specification (e.g. selection of variables, definition of business question)
- Model quantification (e.g. estimation of coefficients, lay-out of decision tree)
- Model performance: the predictive ability of the model

Types of validity

- Apparent (own sample)
- Internal (own population)
- External (other population)
- Model performance on train, test, validation set



Model validation

- The **business relevance** of the analytical model should be guaranteed
- **Statistical performance and validity** need to be balanced against statistical performance, i.e. justifiability and interpretability
- **Operational efficiency and economic cost** need to be taken into account
- **Regulatory compliance** is becoming increasingly important

https://medium.com/dataminingapps-articles/critical-success-factors-for-analytical-models-be35e2cbdef2#.lpxh6v89u

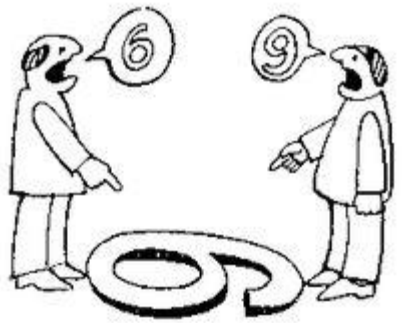
http://www.dataminingapps.com/dataminingapps-newsletter/

Business relevance

- The analytical model should solve the business problem that it was developed for
- This requires a thorough business knowledge and understanding of the problem to be addressed before any analysis can start
- Some example kick-off questions are: how do we define (what is it?), measure (how to see it?) and manage (what to do with it?) fraud, churn, a sale…?

Often many ways to define a problem:

- E.g. predict which customers will reach gold status next year
- Classification on the current group at time *t*?
- Classification on the difference *t* -> *t+1*?
- Regression on the value determining the gold status threshold?
- Time series forecasting on the value determining the gold status threshold?
- How much is enough? How much do you want to know?



Model performance and validity

True Label	Prediction	Predicted Label	Correct?
no	0.11	no	Correct
no	0.20	no	Correct
yes	0.85	yes	Correct
yes	0.84	yes	Correct
yes	0.80	yes	Correct
no	0.65	yes	Incorrect
yes	0.44	no	Incorrect
no	0.10	no	Correct
yes	0.32	no	Incorrect
yes	0.87	yes	Correct
yes	0.61	yes	Correct
yes	0.60	yes	Correct
yes	0.78	yes	Correct
no	0.61	yes	Incorrect

Threshold 0.50

- Confusion matrix

		Prediction	
Reference		*no*	*yes*
	no	**3**	**2**
	yes	**2**	**7**

Model performance and validity

		Predicted condition		
	Total population	Predicted Condition positive	Predicted Condition negative	Prevalence $= \frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$
True condition	condition positive	**True positive**	**False Negative** (Type II error)	True positive rate (TPR), Sensitivity, Recall $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$
	condition negative	**False Positive** (Type I error)	**True negative**	False positive rate (FPR), Fall-out $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$
	Accuracy (ACC) $= \frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	Positive predictive value (PPV), Precision $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Test outcome positive}}$	False omission rate (FOR) $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Test outcome negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$
		False discovery rate (FDR) $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Test outcome positive}}$	Negative predictive value (NPV) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Test outcome negative}}$	Negative likelihood ratio (LR−) $= \frac{\text{FNR}}{\text{TNR}}$

False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$		
True negative rate (TNR), Specificity (SPC) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$		
	Diagnostic odds ratio (DOR) $= \frac{\text{LR+}}{\text{LR−}}$	

Model performance and validity

True Label	Prediction	Predicted Label	Correct?
no	0.11	no	Correct
no	0.20	no	Correct
yes	0.85	yes	Correct
yes	0.84	yes	Correct
yes	0.80	yes	Correct
no	0.65	yes	Incorrect
yes	0.44	no	Incorrect
no	0.10	no	Correct
yes	0.32	no	Incorrect
yes	0.87	yes	Correct
yes	0.61	yes	Correct
yes	0.60	yes	Correct
yes	0.78	yes	Correct
no	0.61	yes	Incorrect

Threshold 0.50

- Confusion matrix

		Prediction	
Reference		*no*	*yes*
no		**3 tn**	**2 fp**
yes		**2 fn**	**7 tp**

- **Accuracy** = (tp + tn) / total = (3 + 7) / 14 = 0.71
- **Balanced accuracy** = (recall + specificity) / 2 = (0.5 * tp) / (tp + fn) + (0.5 * tn) / (tn + fp) = 0.5 * 0.78 + 0.5 * 0.60 = 0.69
- **Recall (sensitivity)** = tp / (tp + fn) = 7 / 9 = 0.78 "How of the positives did we catch?"
- **Precision** = tp / (tp + fp) = 7 / 9 = 0.78 "How much are we getting it wrong?"

Model performance and validity

True Label	Prediction	Predicted Label	Correct?
no	0.11	no	Correct
no	0.20	no	Correct
yes	0.85	yes	Correct
yes	0.84	yes	Correct
yes	0.80	yes	Correct
no	0.65	yes	Incorrect
yes	0.44	no	Incorrect
no	0.10	no	Correct
yes	0.32	no	Incorrect
yes	0.87	yes	Correct
yes	0.61	yes	Correct
yes	0.60	yes	Correct
yes	0.78	yes	Correct
no	0.61	yes	Incorrect

Threshold 0.50

- Confusion matrix

		Prediction
Reference	*no*	*yes*
no	**3 tn**	**2 fp**
yes	**2 fn**	**7 tp**

- **Recall and Precision often counteract each other:** to catch more of the positives you need to be prepared to make additional mistakes

Model performance and validity

True Label	Prediction
no	0.11
no	0.20
yes	0.85
yes	0.84
yes	0.80
no	0.65
yes	0.44
no	0.10
yes	0.32
yes	0.87
yes	0.61
yes	0.60
yes	0.78
no	0.61

Best threshold

- For each possible threshold $t \in T$ with T the set of all predicted probabilities, we can obtain a confusion matrix
- So which threshold is the best one?

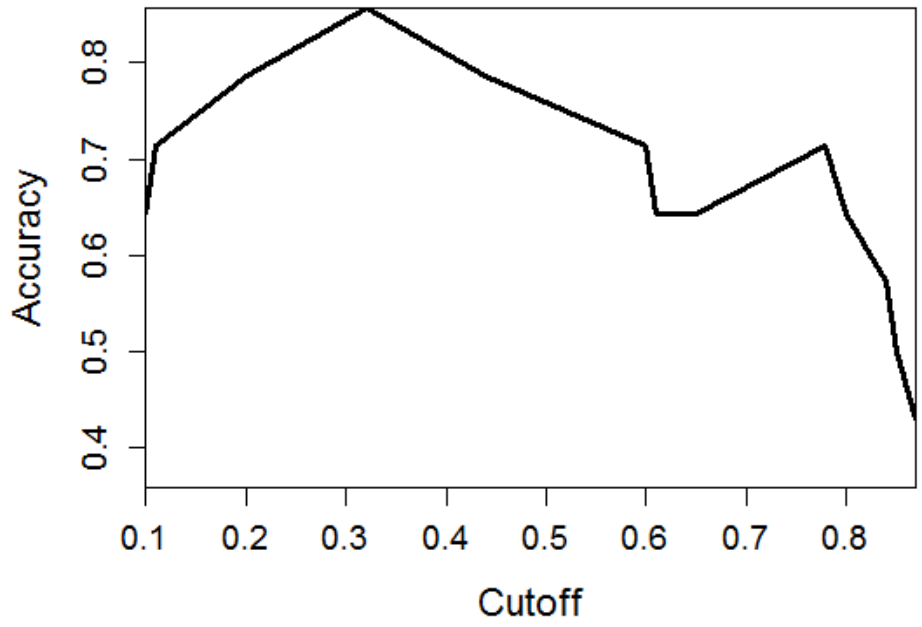
Threshold	tp	fp	tn	fn
0.1	9	5	0	0
0.11	9	4	1	0
0.2	9	3	2	0
0.32	9	2	3	0
0.44	8	2	3	1
0.6	7	2	3	2
0.61	6	2	3	3
0.65	5	1	4	4
0.78	5	0	5	4
0.8	4	0	5	5
0.84	3	0	5	6
0.85	2	0	5	7
0.87	1	0	5	8

Model performance and validity

True Label	Prediction
no	0.11
no	0.20
yes	0.85
yes	0.84
yes	0.80
no	0.65
yes	0.44
no	0.10
yes	0.32
yes	0.87
yes	0.61
yes	0.60
yes	0.78
no	0.61

Best threshold

- For each possible threshold $t \in T$ with T the set of all predicted probabilities, we can obtain a confusion matrix
- So which threshold is the best one? **ACC-threshold curve**

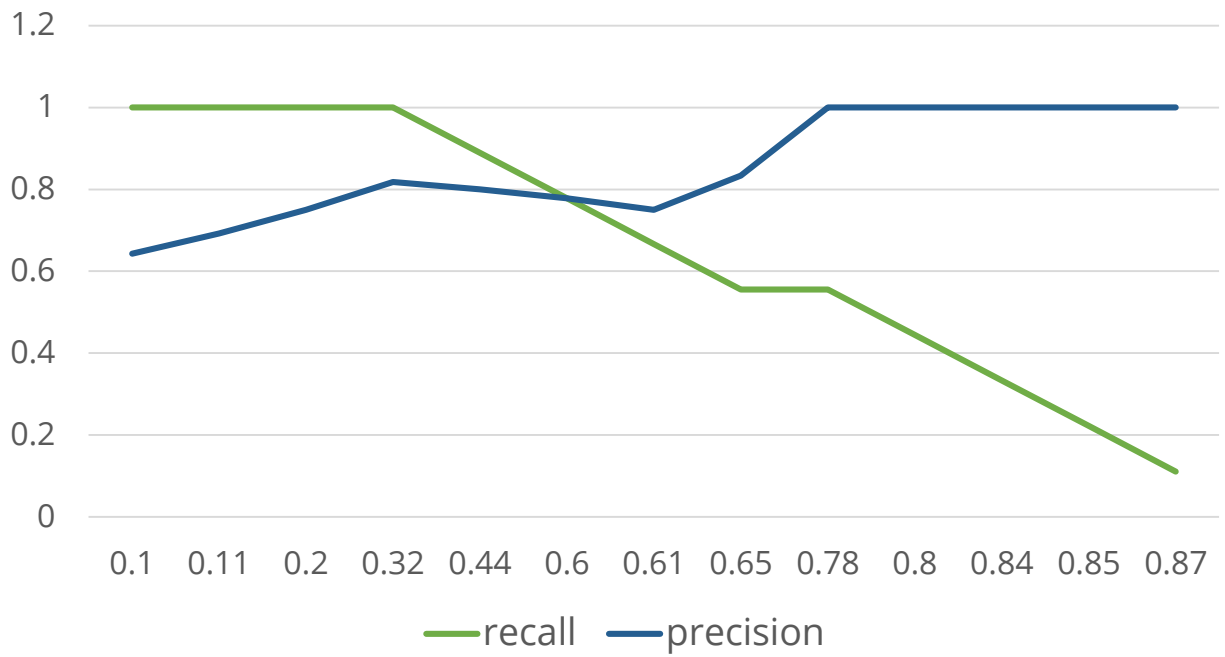


Model performance and validity

True Label	Prediction
no	0.11
no	0.20
yes	0.85
yes	0.84
yes	0.80
no	0.65
yes	0.44
no	0.10
yes	0.32
yes	0.87
yes	0.61
yes	0.60
yes	0.78
no	0.61

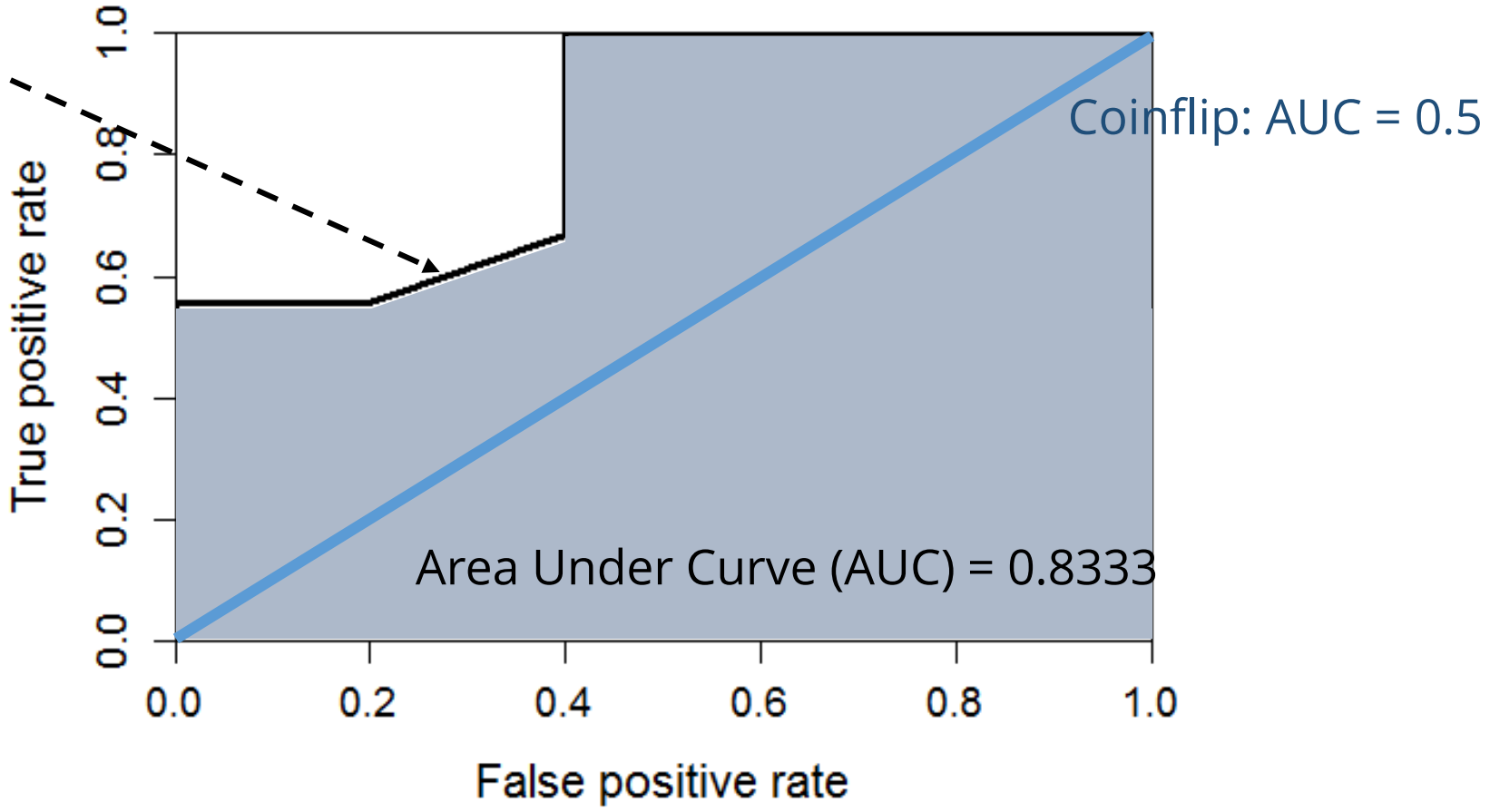
Best threshold

- For each possible threshold $t \in T$ with T the set of all predicted probabilities, we can obtain a confusion matrix
- So which threshold is the best one? **Recall-precision curve**



Model performance and validity

Receiver Operator Characteristic (ROC) curve

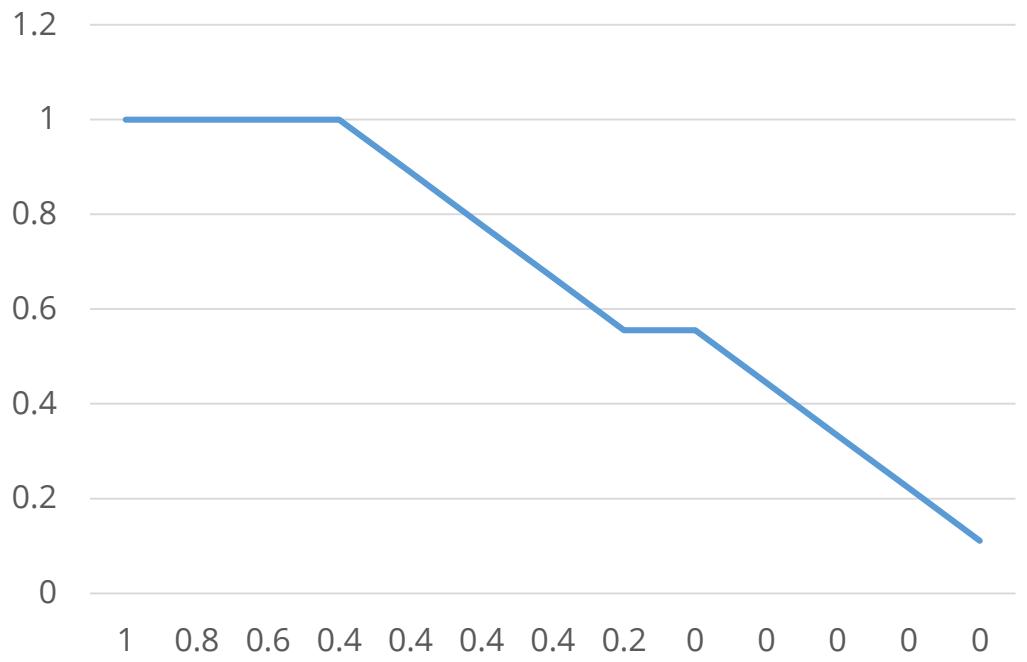


Model performance and validity

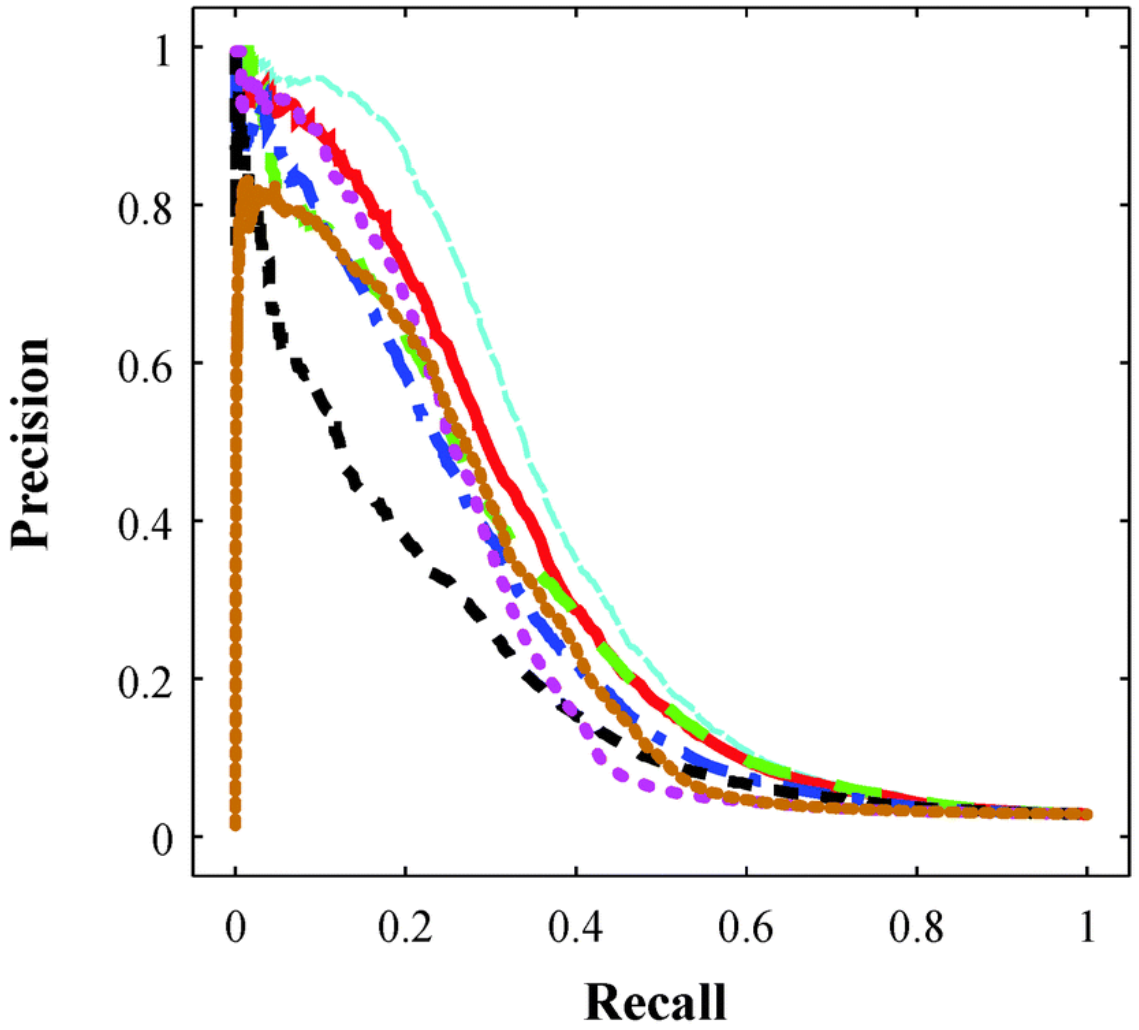
True Label	Prediction
no	0.11
no	0.20
yes	0.85
yes	0.84
yes	0.80
no	0.65
yes	0.44
no	0.10
yes	0.32
yes	0.87
yes	0.61
yes	0.60
yes	0.78
no	0.61

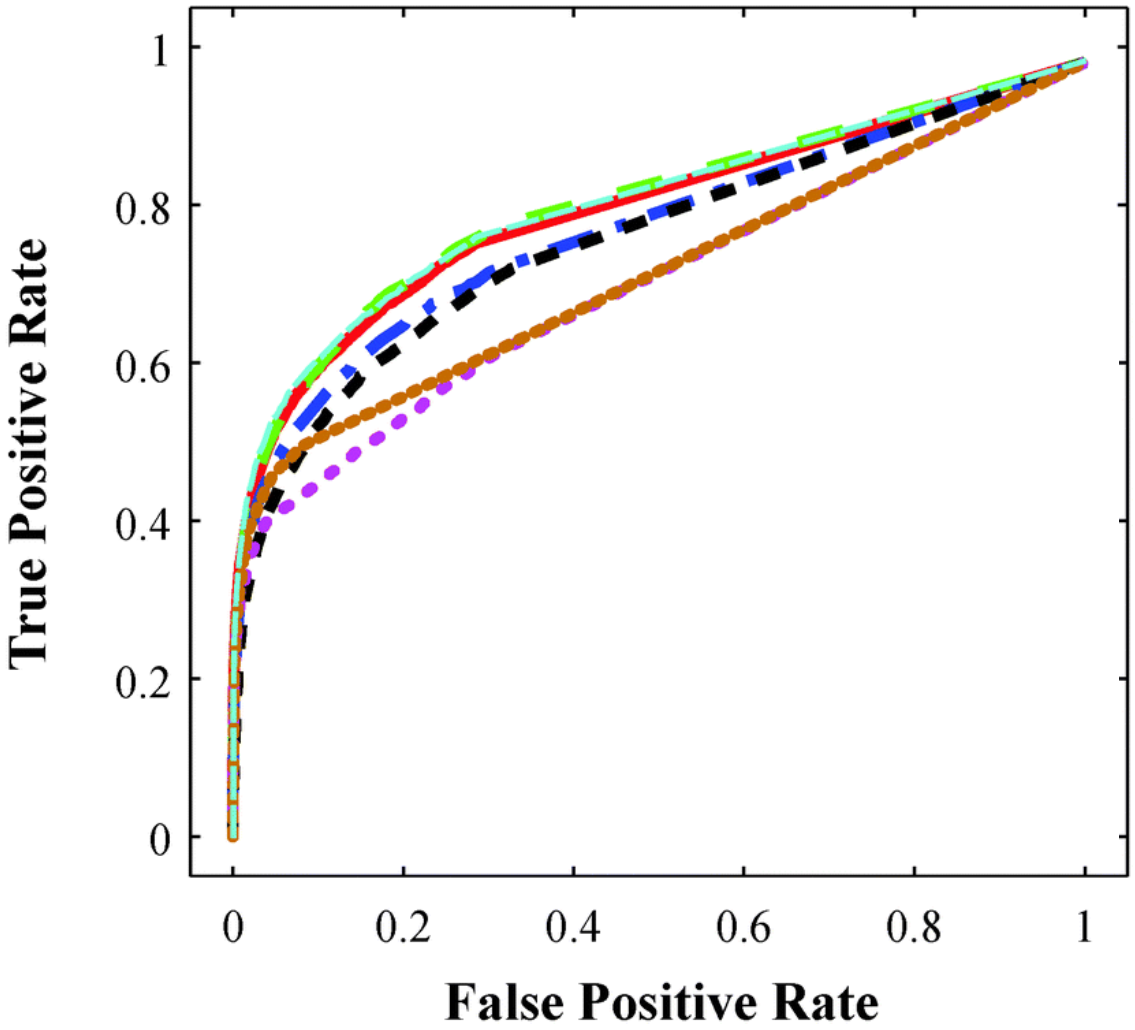
Best threshold

- For each possible threshold $t \in T$ with T the set of all predicted probabilities, we can obtain a confusion matrix
- So which threshold is the best one? **ROC curve and AUC**



Model performance and validity





Model performance and validity

True Label	Prediction
no	0.11
no	0.20
yes	0.85
yes	0.84
yes	0.80
no	0.65
yes	0.44
no	0.10
yes	0.32
yes	0.87
yes	0.61
yes	0.60
yes	0.78
no	0.61

Best threshold

- For each possible threshold $t \in T$ with T the set of all predicted probabilities, we can obtain a confusion matrix
- So which threshold is the best one? **Depends on the target**

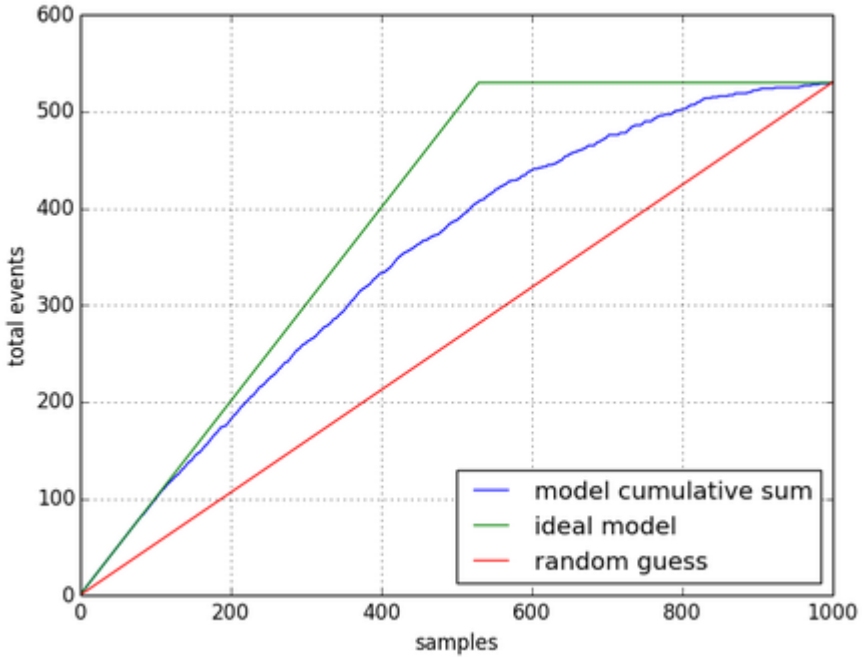
Threshold	tp	fp	tn	fn	recall	precision	tp-rate	fp-rate
0.1	9	5	0	0	1	0.642857	1	1
0.11	9	4	1	0	1	0.692308	1	0.8
0.2	9	3	2	0	1	0.75	1	0.6
0.32	9	2	3	0	1	0.818182	1	0.4
0.44	8	2	3	1	0.888889	0.8	0.888889	0.4
0.6	7	2	3	2	0.777778	0.777778	0.777778	0.4
0.61	6	2	3	3	0.666667	0.75	0.666667	0.4
0.65	5	1	4	4	0.555556	0.833333	0.555556	0.2
0.78	5	0	5	4	0.555556	1	0.555556	0
0.8	4	0	5	5	0.444444	1	0.444444	0
0.84	3	0	5	6	0.333333	1	0.333333	0
0.85	2	0	5	7	0.222222	1	0.222222	0
0.87	1	0	5	8	0.111111	1	0.111111	0

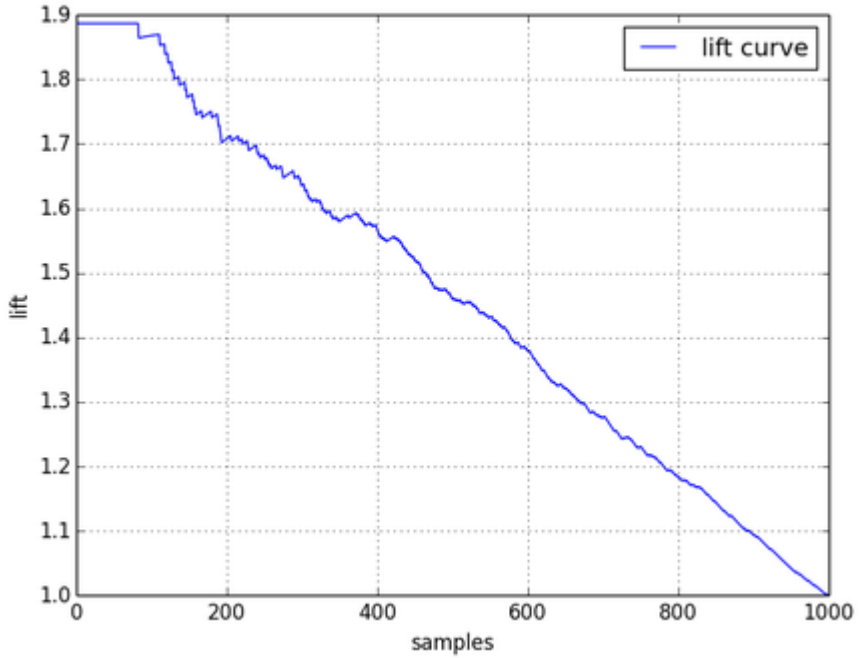
Model performance and validity

True Label	Prediction
no	0.11
no	0.20
yes	0.85
yes	0.84
yes	0.80
no	0.65
yes	0.44
no	0.10
yes	0.32
yes	0.87
yes	0.61
yes	0.60
yes	0.78
no	0.61

Best threshold

- For each possible threshold $t \in T$ with T the set of all predicted probabilities, we can obtain a confusion matrix
- So which threshold is the best one? **Lift (ratio of a model to a random guess)**



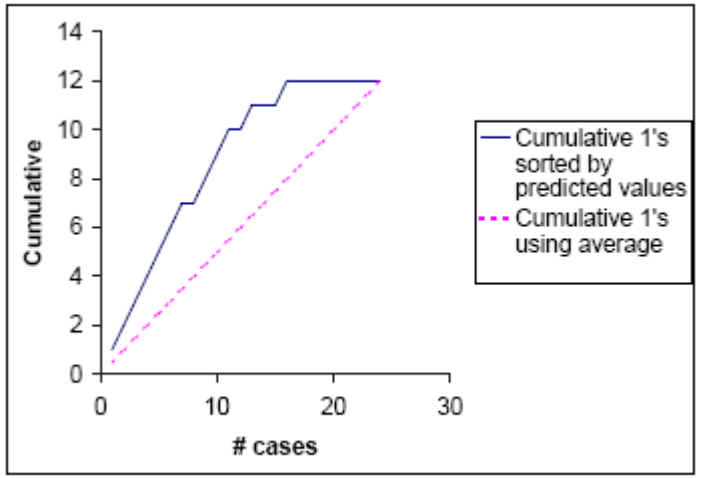


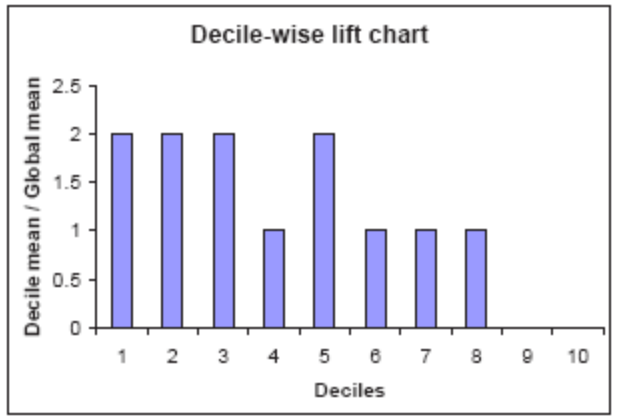
Model performance and validity

True Label	Prediction
no	0.11
no	0.20
yes	0.85
yes	0.84
yes	0.80
no	0.65
yes	0.44
no	0.10
yes	0.32
yes	0.87
yes	0.61
yes	0.60
yes	0.78
no	0.61

Best threshold

- For each possible threshold $t \in T$ with T the set of all predicted probabilities, we can obtain a confusion matrix
- So which threshold is the best one? **Decile lift (ratio of a model to a random guess)**

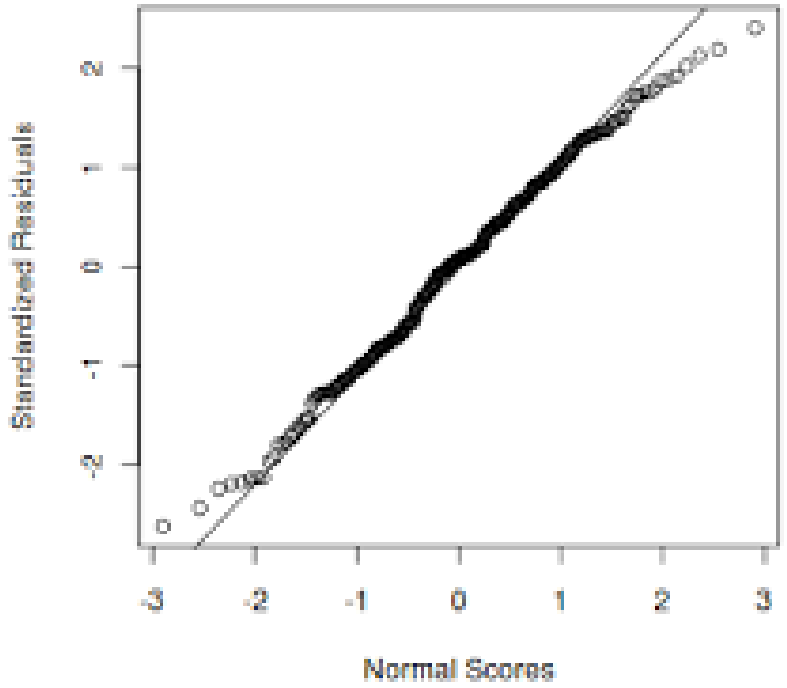




Model performance and validity

And more

- Residuals plot
- Correlation between prediction and true label
- Etc.

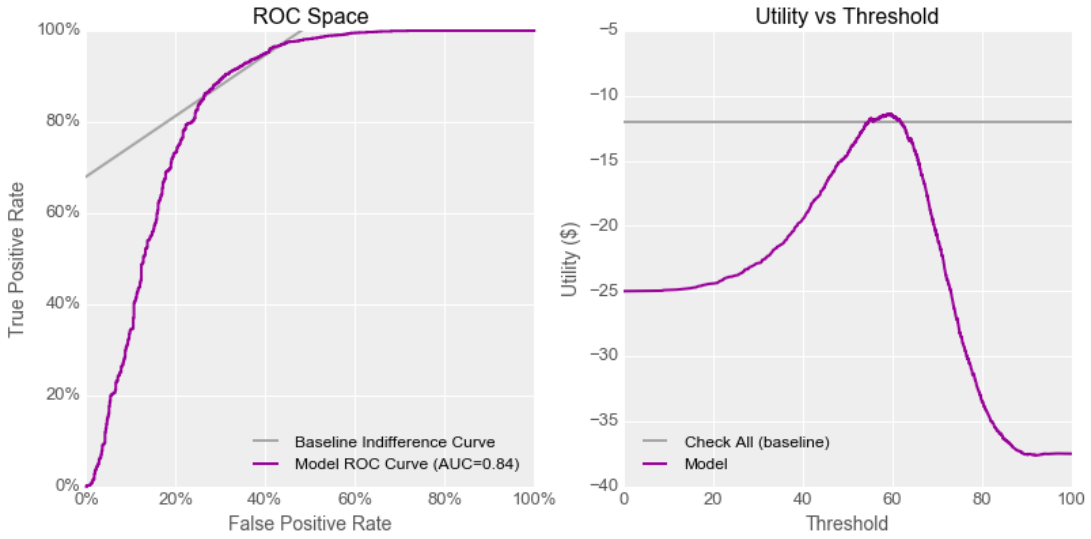


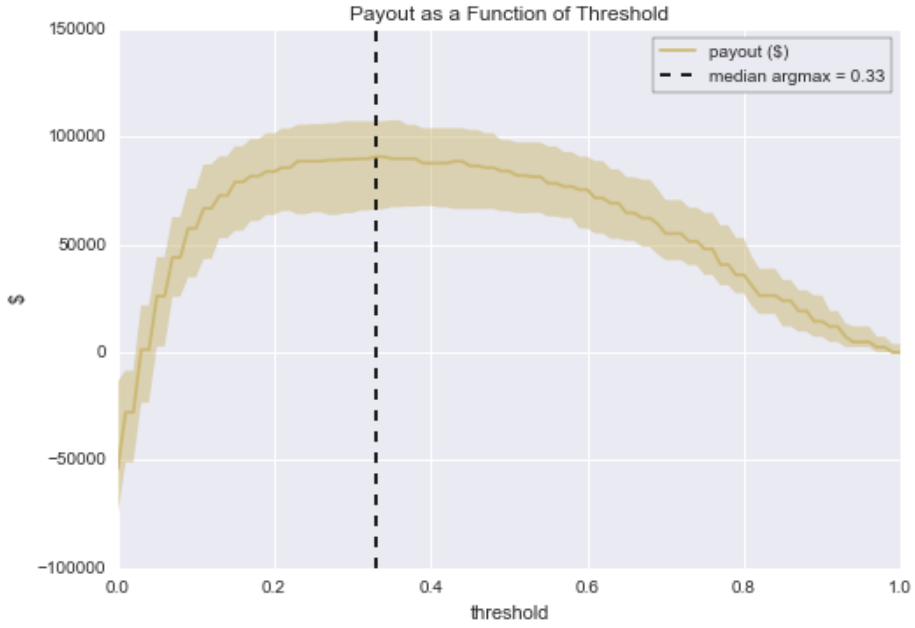
Model performance and validity

And more

- Setting misclassification costs
- Calculate utility cost / benefit

Reference	Prediction *no*	*yes*
no	**3 +**	**2 ---**
yes	**2 -**	**7 ++**

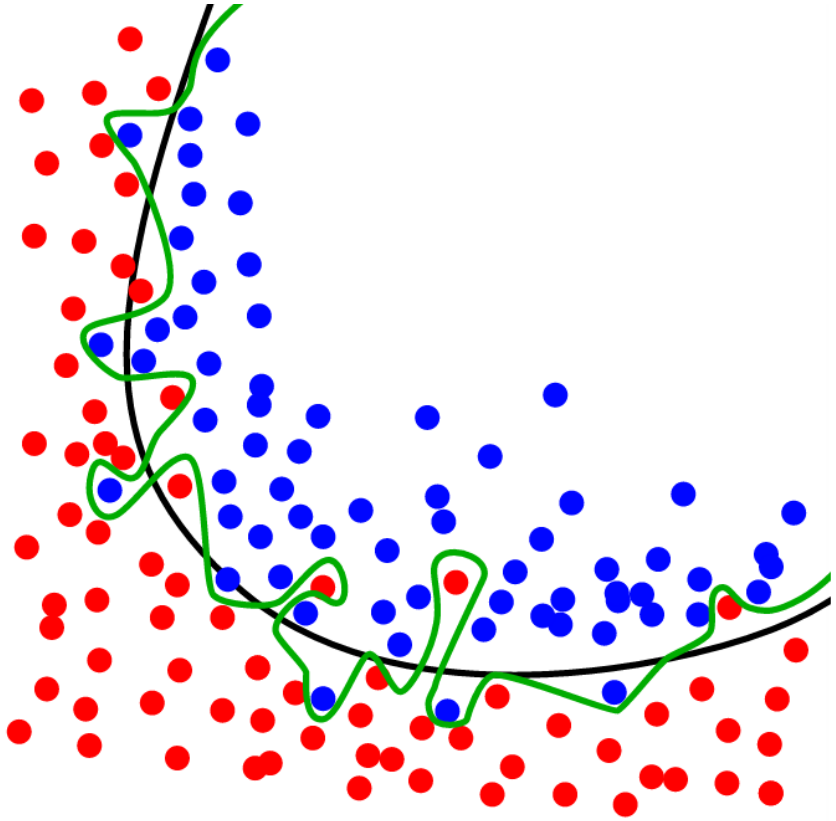




Model performance and validity

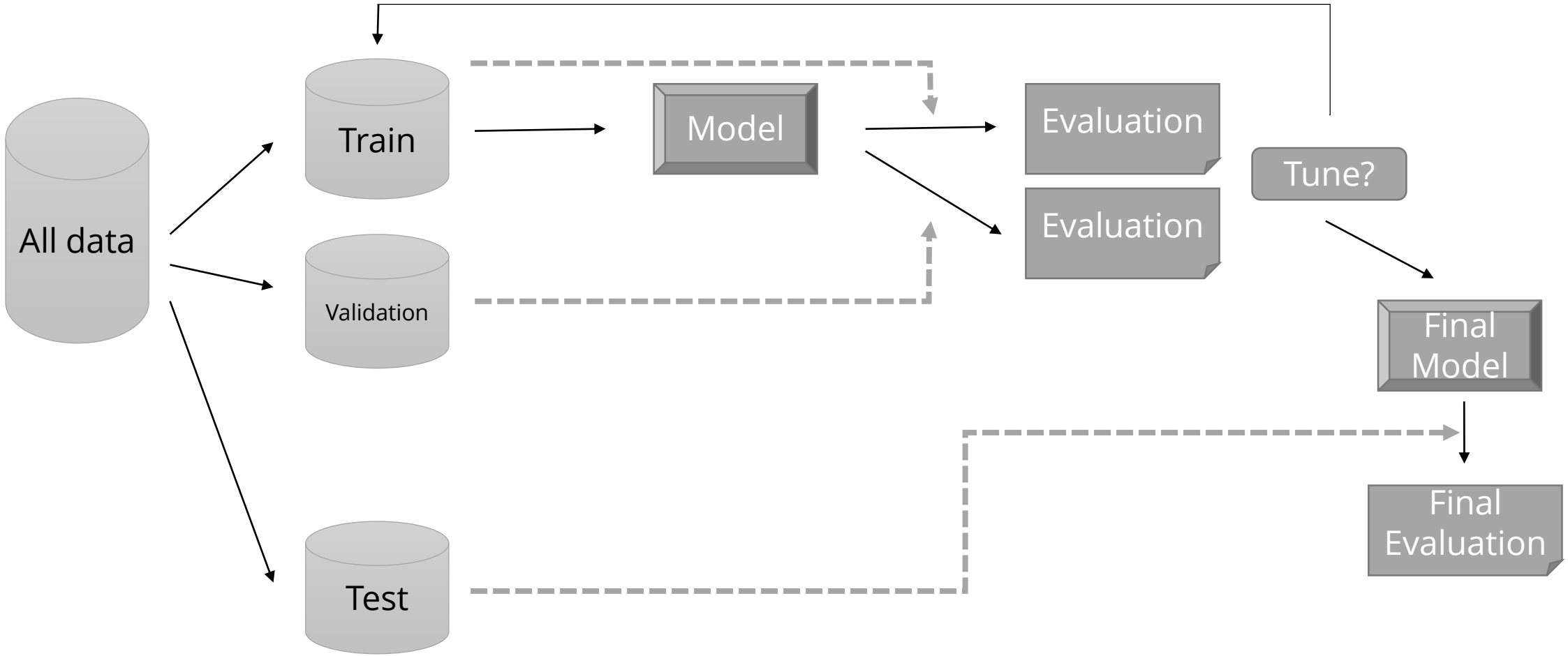
Preventing overfitting, being able to generalize

- Whilst still being able to optimally tune model
- Train, validation and test set



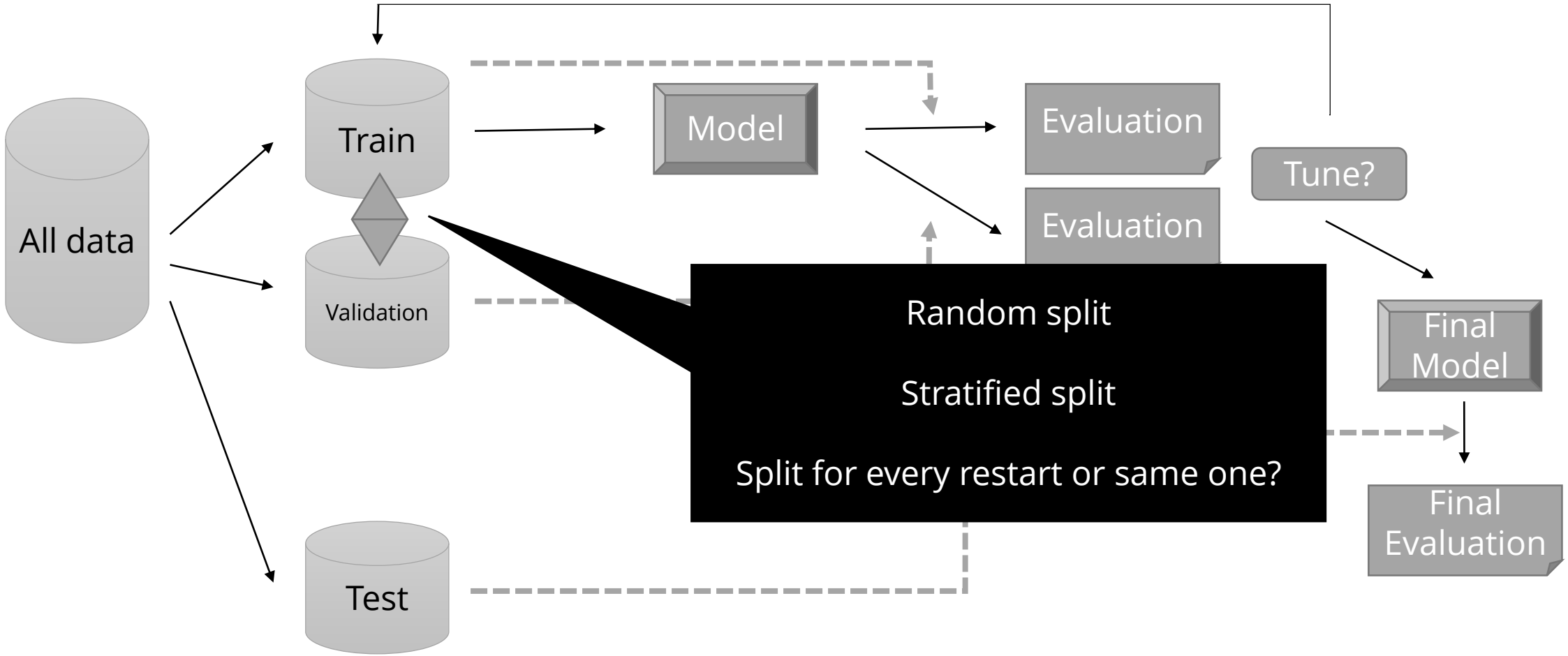
Model performance and validity

Train, validation and test set



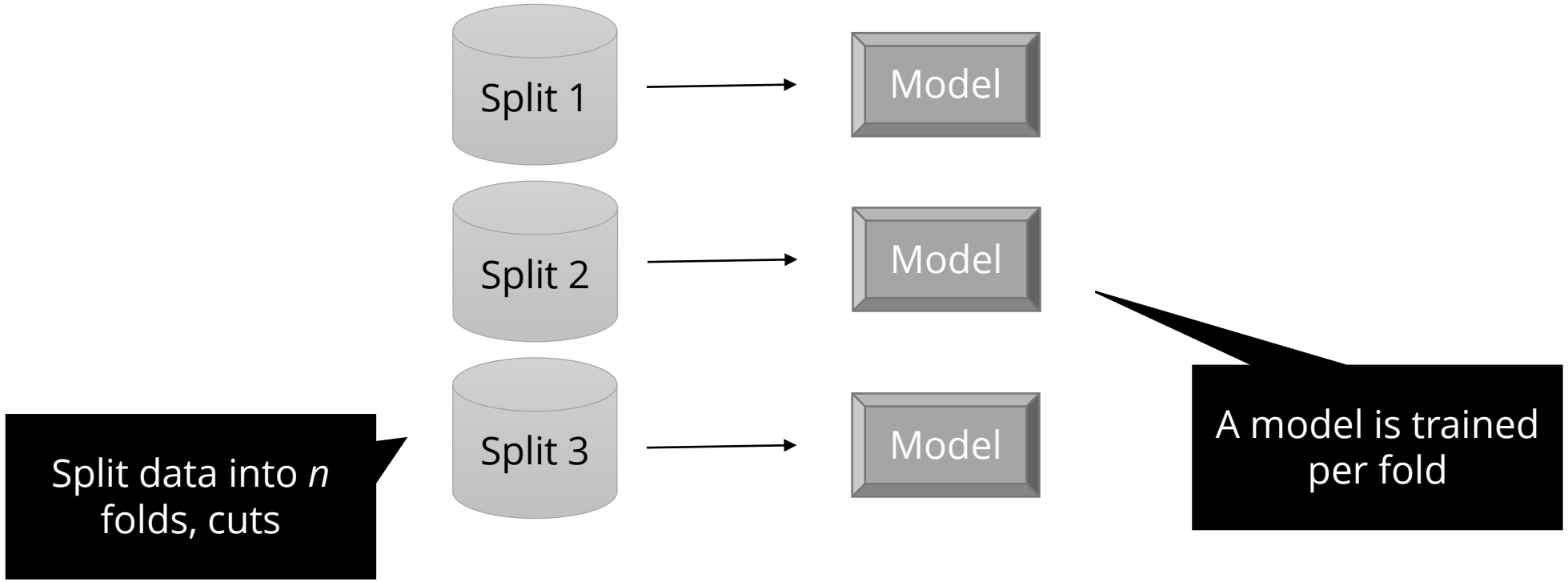
Model performance and validity

Train, validation and test set



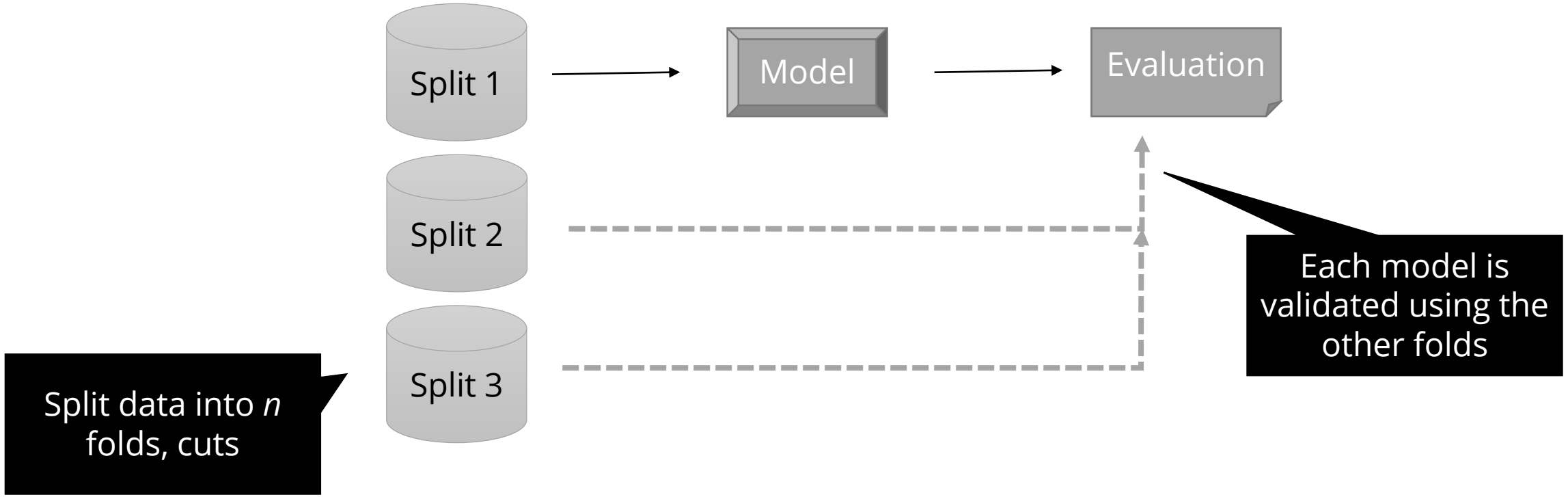
Model performance and validity

Cross-validation



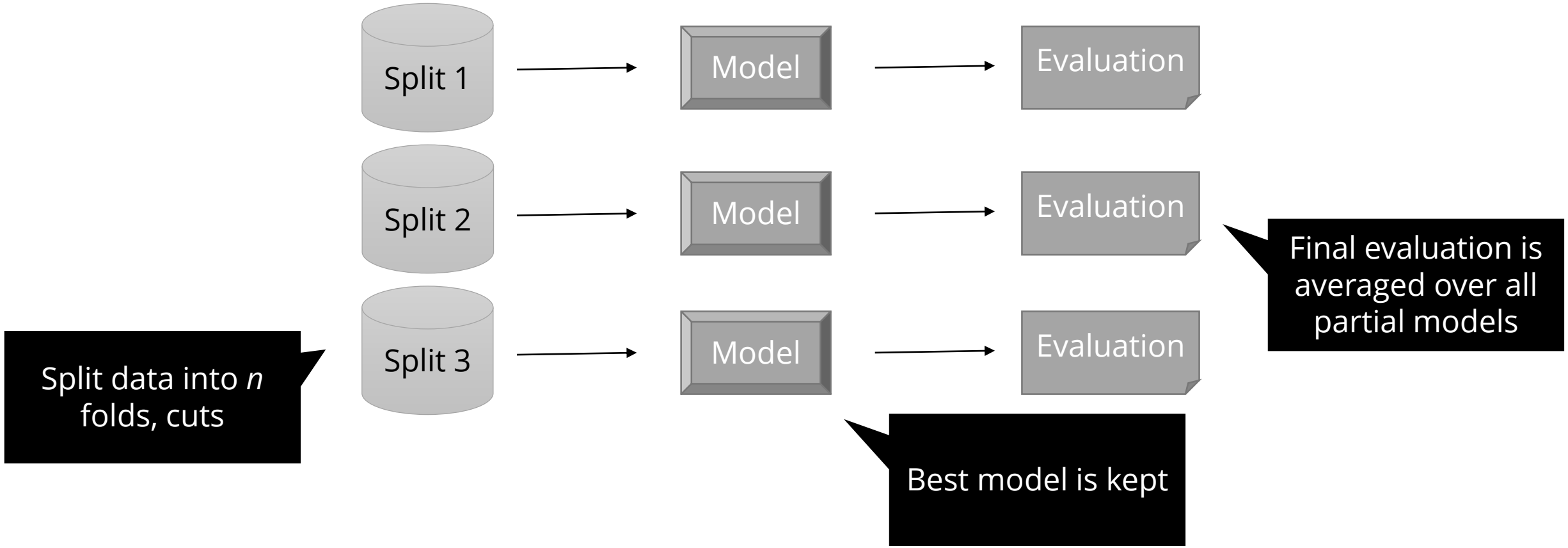
Model performance and validity

Cross-validation



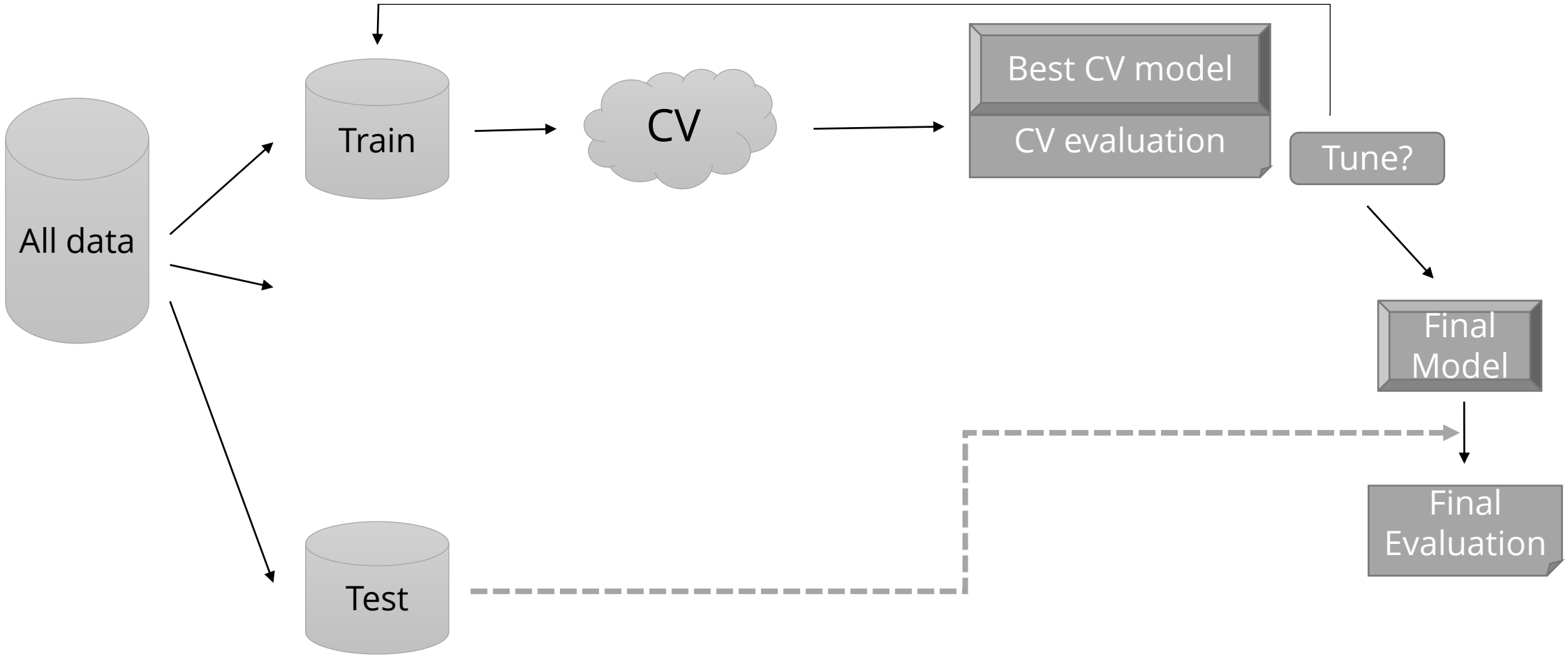
Model performance and validity

Cross-validation



Model performance and validity

Train, validation and test set



Model performance and validity

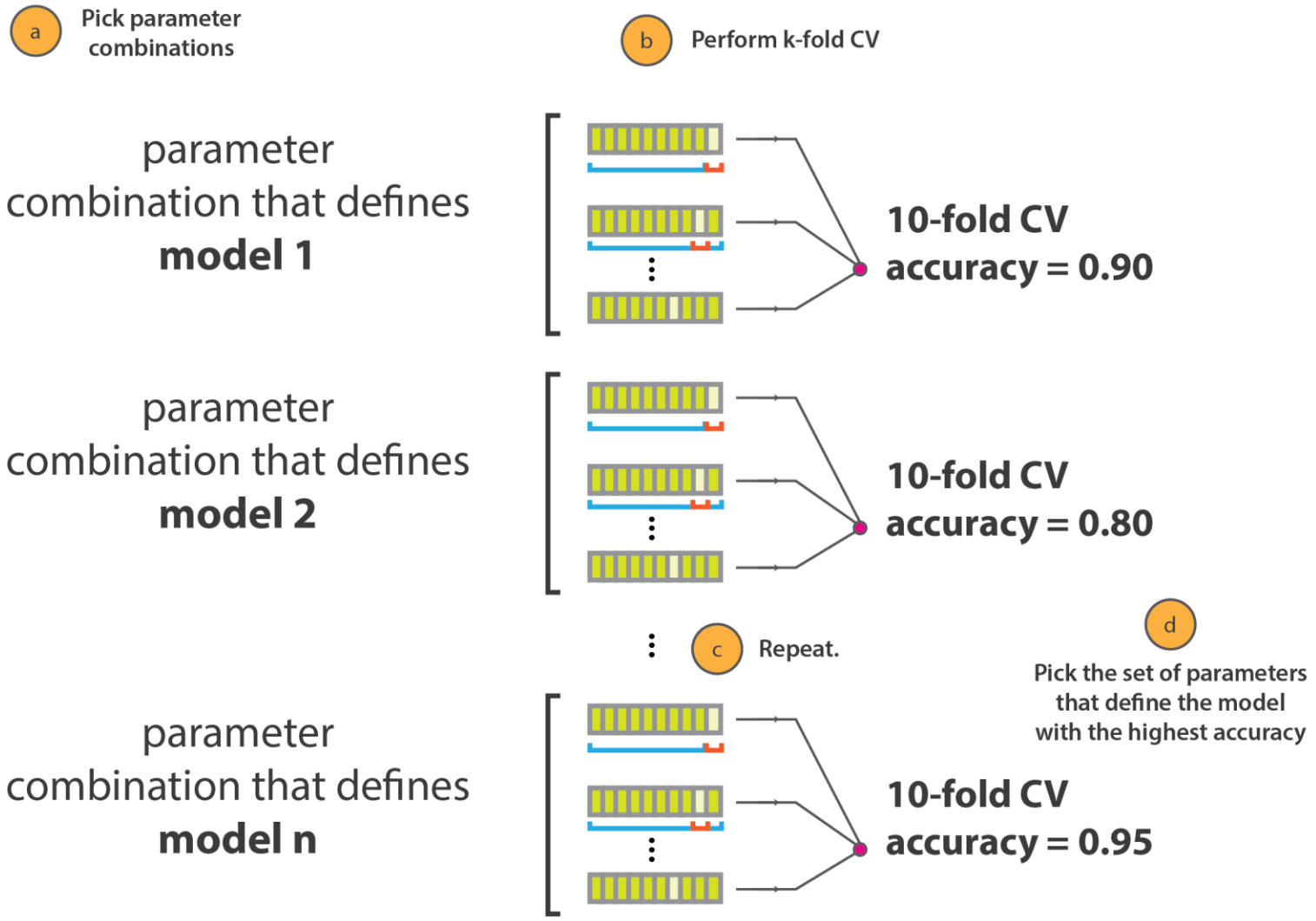
Cross-validation

- Two-fold, holdout: this is the normal train/validation split
- K-fold
- Leave-one out, jackknifing: always remove one instance and train model on the remaining instances

Model performance and validity

Final concerns:

- What if final test set evaluation gives bad results? *(Throw away the whole project?)*
- Should feature engineering and transformation be done on the whole data set? *(It's so hard not to)*
- Too much re-use of same train/validation split leads to hidden overtraining *(I'll just make a small parameter tuning)*
- So does too much parameter combination runs (over-usage of the same data) *(It's okay, I'm cross-validating on each run)*
- Some models try to avoid overfitting by themselves (recall: bootstrapping)
- Also, if scores are too good to be true, they probably are (target variable "leakage")



Operational efficiency and economic cost

- Operational efficiency relates to the effort that is needed to evaluate, monitor, backtest or rebuild the model
- From this perspective, it is quite obvious that a neural network or random forest is less efficient that e.g. a plain vanilla regression model or decision tree
- In some settings like credit card fraud detection, operational efficiency is very important because a decision should be made within a few seconds after the credit card transaction was initiated
- Economical cost refers to the cost that is needed to gather the model inputs, run the model and process its outcome(s)
- Also the cost of external data and/or models should be taken into account here
- Calculating the economic return on the analytical model is not a straightforward exercise
- Technical debt and maintenance… will models have to work 1 year from now? 10 years?

Other concerns

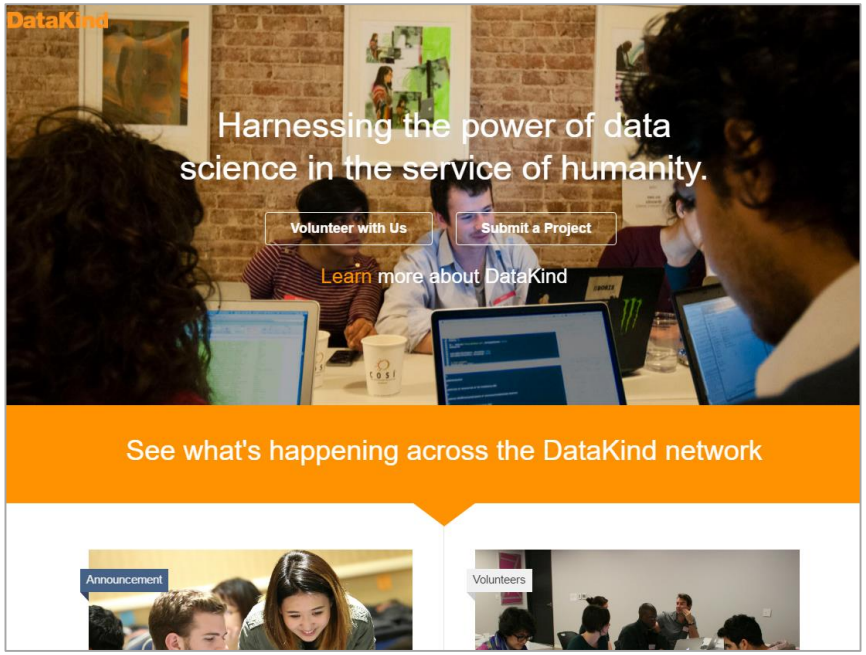
- Regulatory compliance: e.g. no black-boxes, no use allowed of certain features
- Questions on ethics…
- Can an algorithm be racist? Sexist?
- “Will Predictive Models Outliers Be The New Socially Excluded?”
- Companies like DataKind, or Bayes Impact
- Concept of “open models”

Data Mining: Where Legality and Ethics Rarely Meet

By Kelly Shermach
Aug 25, 2006 4:00 AM PT

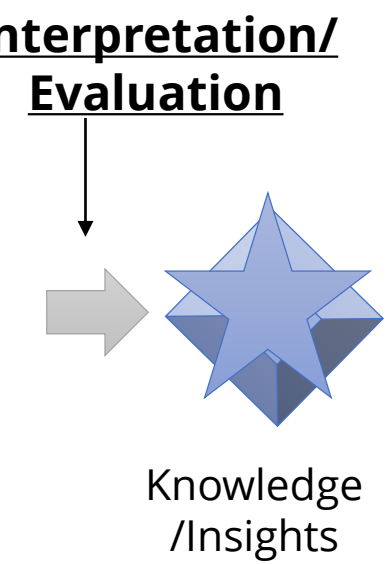
Print
Email

More than ever, knowingly or unknowingly, consumers disseminate personal data in daily activities. Credit and debit card transactions, ATM visits, Web site browsing and purchases -- even mobile phone use -- all generate data downloaded for analysis and customer profiling. Collectors may use this data to enhance customers' experience, but may also share information with marketers more focused on customer acquisition.



Wrap-up

- Business relevance
- Statistical performance and validity
- Operational efficiency and economic cost
- Regulatory compliance



Wrap-up

- During the first KDD Cup 1997, the goal was to select a subset of lapsed donors to contact. An entry from one well-known company selected the worst possible candidates for mailing - their results were significantly worse than random!

  Apparently their data miners switched the sign somewhere. Fortunately for them, the names of the contestants were kept anonymous. (Gregory Piatetsky-Shapiro)

- In the 1980s, the Pentagon wanted to harness computer technology to make their tanks harder to attack.

  The preliminary plan was to fit each tank with a digital camera hooked up to a computer. The computer would continually scan the environment outside for possible threats (such as an enemy tank hiding behind a tree), and alert the tank crew to anything suspicious. The only possible way to solve the problem was to employ a neural network. The research team went out and took 100 photographs of tanks hiding behind trees, and then took 100 photographs of trees - with no tanks. They took half the photos from each group and put them in a vault for safe-keeping, then scanned the other half into their mainframe computer. The huge neural network was fed each photo one at a time and asked if there was a tank hiding behind the trees. Over time it got better and better until eventually it was getting each photo correct. It could correctly determine if there was a tank hiding behind the trees in any one of the photos.

  But the scientists were worried: had it actually found a way to recognize if there was a tank in the photo, or had it merely memorized which photos had tanks and which did not? So the scientists took out the photos they had been keeping in the vault and fed them through the computer. To their immense relief the neural net correctly identified each photo as either having a tank or not having one.

  The Pentagon was very pleased with this, but a little bit suspicious. They commissioned another set of photos (half with tanks and half without) and scanned them into the computer and through the neural network. The results were completely random. For a long time nobody could figure out why. After all nobody understood how the neural had trained itself.

  Eventually someone noticed that in the original set of 200 photos, all the images with tanks had been taken on a cloudy day while all the images without tanks had been taken on a sunny day. The neural network had been asked to separate the two groups of photos and it had chosen the most obvious way to do it - not by looking for a camouflaged tank hiding behind a tree, but merely by looking at the colour of the sky. The military was now the proud owner of a multi-million dollar mainframe computer that could tell you if it was sunny or not.

Wrap-up

- Occasionally researchers using pruning algorithms on their decision trees get carried away. Instead of pruning unnecessary branches in the interests of reducing overfitting. The experimenter just burns down the tree until it is a decision stump (Data Mining Disasters: a report)

- In pre-Big Data days, for example, a hotel chain used some pretty sophisticated mathematics, data mining, and time series analysis to coordinate its yield management pricing and promotion efforts. The forecasting models—which were marvels—mapped out revenues and margins by property and room type. The projections worked fine for about a third of the hotels but were wildly, destructively off for another third.

  The forensics took weeks; the data were fine. Were competing hotels running unusual promotions that screwed up the model? Nope. For the most part, local managers followed the yield management rules.

  Almost five months later, after the year's financials were totally blown and HQ's credibility shot, the most likely explanation materialized: The modeling group—the data scientists of the day—had priced against the hotel group's peer competitors. They hadn't weighted discount hotels into either pricing or room availability. For roughly a quarter of the properties, the result was both lower average occupancy and lower prices per room (Learn from Your Analytics Failures, HBR)