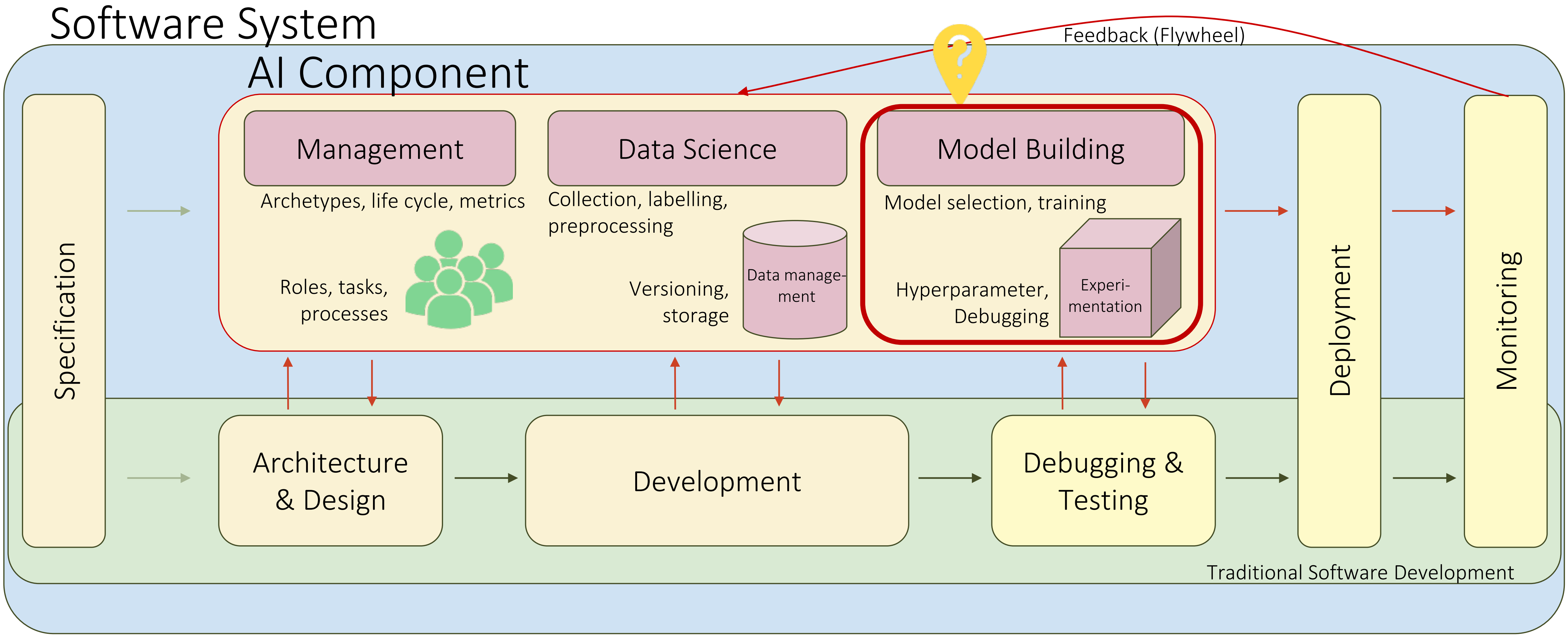
Software Engineering for AI-Enabled Systems

Prof. Dr.-Ing. Norbert Siegmund
Software Systems



How to develop an AI system, including the data science process, coding, and experimentation?

- How to define experiments?
- How to ensure validity of the results?
- How to derive meaningful metrics and know when a technique really improves over existing solutions?
- How to make experiments reproducible?

Topic: Validity of AI-Experiments

TL;DR:

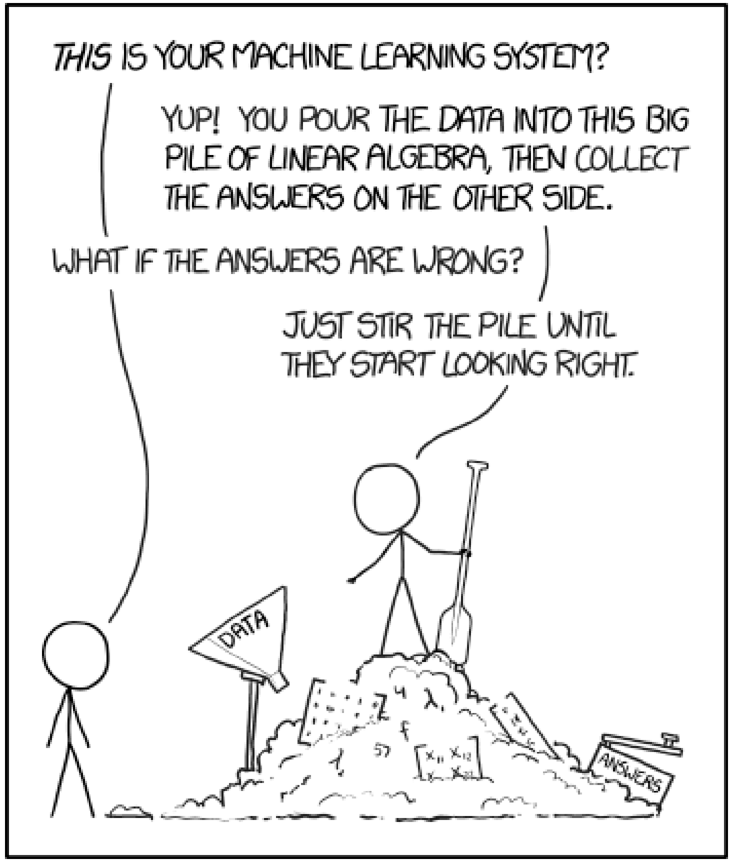
- Experimentation can often go wrong; know possibly errors
- Align the experiment to the actual application goal by choosing suitable metrics
- Avoid leaking information from test to training and fool yourself
- Make the setup reproducible and explicit to increase transparency and easy debugging
- Know important evaluation metrics and what they mean

Do you know „Clever Hans“?

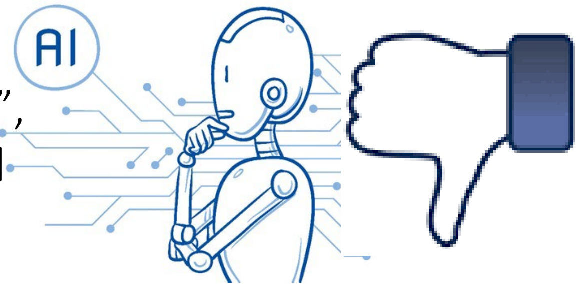
It also muddies the origin of certain data sets. This can mean that researchers miss important features that skew the training of their models. Many unwittingly used a data set that contained chest scans of children who did not have covid as their examples of what non-covid cases looked like. But as a result, the AIs learned to identify kids, not covid.

Driggs's group trained its own model using a data set that contained a mix of scans taken when patients were lying down and standing up. Because patients scanned while lying down were more likely to be seriously ill, the AI learned wrongly to predict serious covid risk from a person's position.

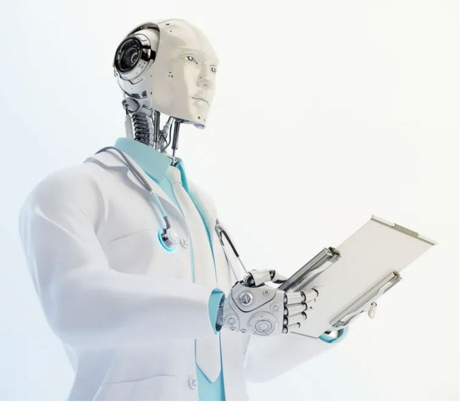
In yet other cases, some AIs were found to be picking up on the text font that certain hospitals used to label the scans. As a result, fonts from hospitals with more serious caseloads became predictors of covid risk.


THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA
ANSWERS

Beware of the AI-Hype



Facebook is "hiring over 10,000 more people this year to work on safety and security", but warns that it is hard to that sort of moderation "at a global scale … since it is **hard for machines to understand the cultural nuances** of political intimidation."



IBM Watson

IBM Watson set out to "eradicate cancer" …
4 years later the collaboration has been canceled
No trust in decisions, **no ways of explaining** treatment proposals

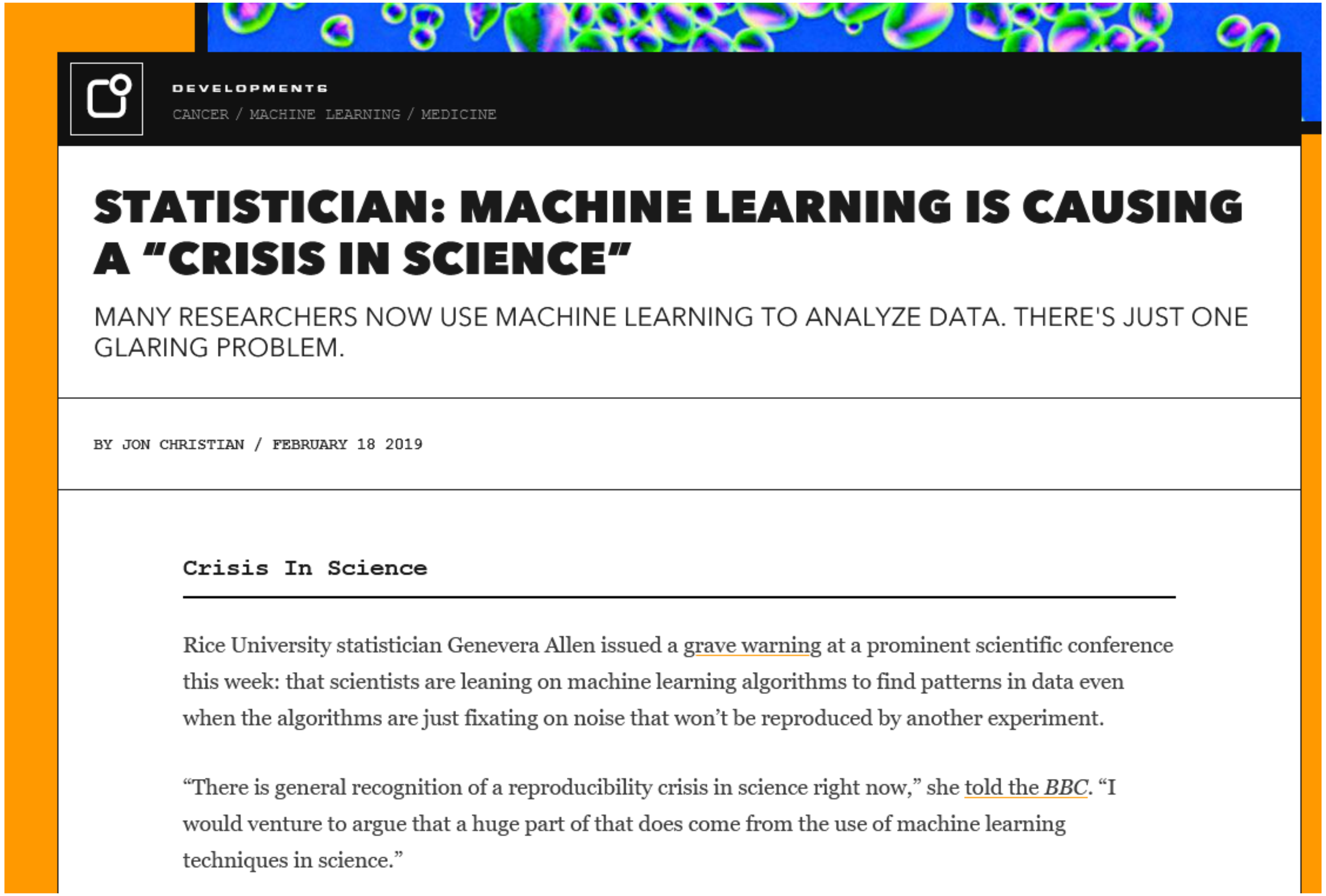
"We build autonomous systems that affect the world in a direct, physical manner, we risk bad actors accessing it. We risk glitches and errors causing physical harm."

https://www.cnbc.com/2019/10/23/alphabet-exec-admits-google-overhyped-self-driving-cars.html

Waymo's "chief external officer" Tekedra N. Mawakana says hype around its self-driving cars became "unmanageable."

https://finfeed.com/opinion/ctrl-alt-del/we-should-abandon-autonomous-vehicles/

Mind the **AI Solutionism**



DEVELOPMENTS

CANCER / MACHINE LEARNING / MEDICINE

STATISTICIAN: MACHINE LEARNING IS CAUSING A "CRISIS IN SCIENCE"

MANY RESEARCHERS NOW USE MACHINE LEARNING TO ANALYZE DATA. THERE'S JUST ONE GLARING PROBLEM.

BY JON CHRISTIAN / FEBRUARY 18 2019

Crisis In Science

Rice University statistician Genevera Allen issued a grave warning at a prominent scientific conference this week: that scientists are leaning on machine learning algorithms to find patterns in data even when the algorithms are just fixating on noise that won't be reproduced by another experiment.

"There is general recognition of a reproducibility crisis in science right now," she told the *BBC*. "I would venture to argue that a huge part of that does come from the use of machine learning techniques in science."

https://www.bbc.com/news/science-environment-47267081

Misinterpretation & -analysis

- Blindly using machine learning on problems that are stochastic in nature
- Finding patterns that solely exist in data, but not in the real world
- Reproducibility crisis

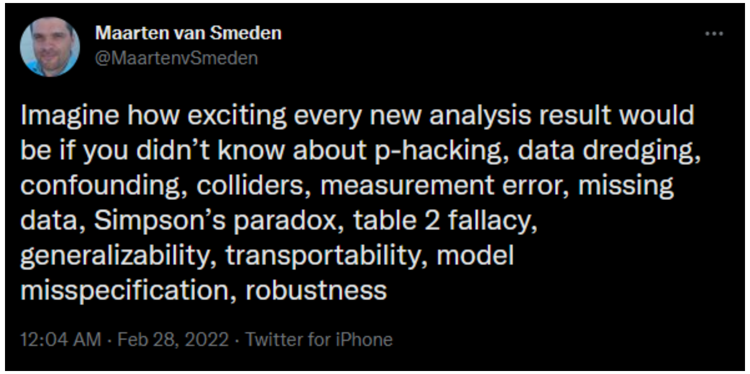
P-hacking

- With plenty of (Big) data, it is easy to find a statistically significant result, leading to spurious correlations
- In a mountain of data, we find something to report…

Topic I:
Experimental Setup

Issues threatening the validity of your experiments

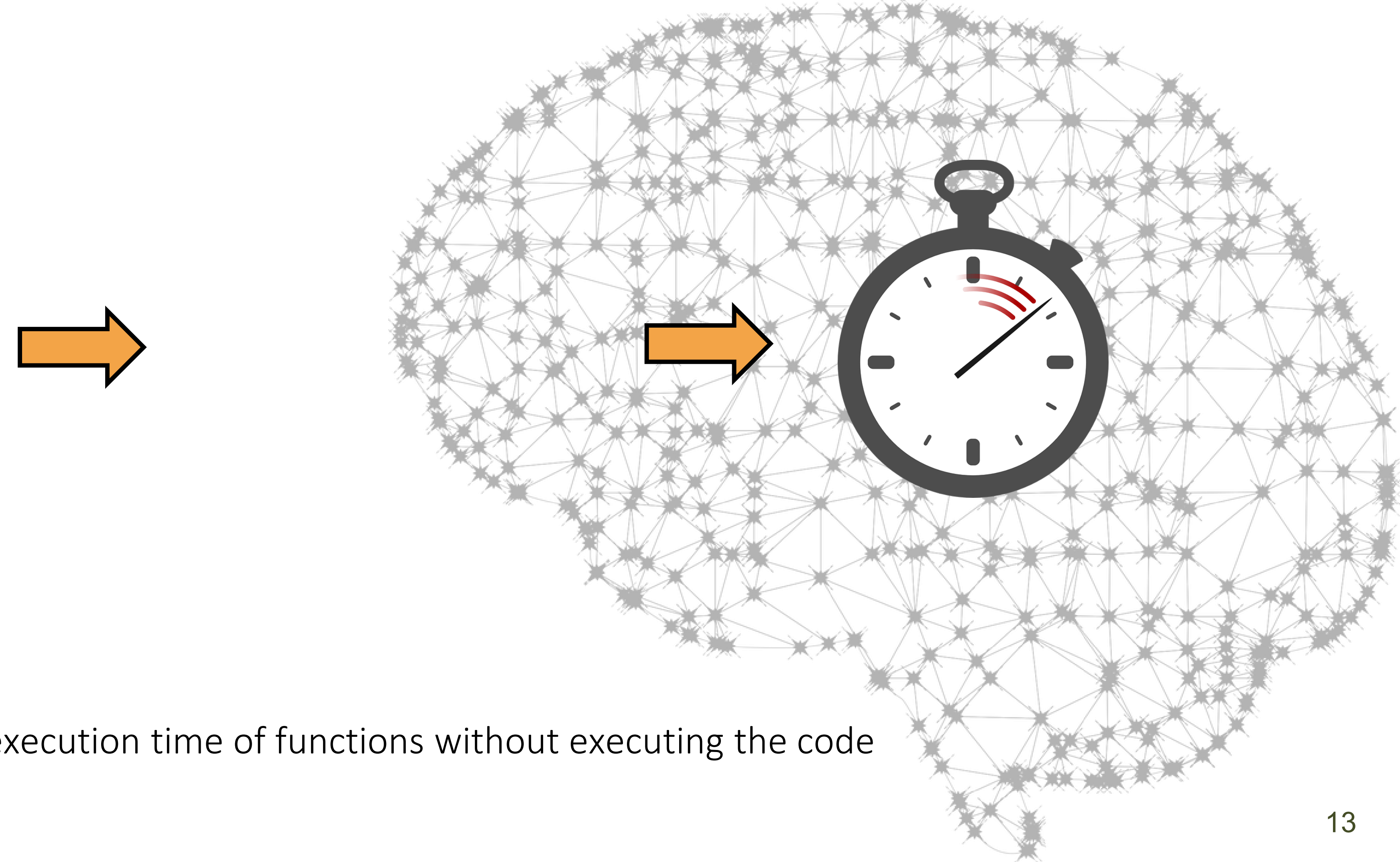
- p-hacking
- Data dredging
- Confounding factors
- Colliders
- Measurement error
- Missing data
- Simpson's paradox
- Table 2 fallacy
- Generalizability
- Transportability
- Model misspecification
- Robustness
- Type I and Type II error
- Overfitting
- Sparse sample bias
- Winner's curse
- Non-collapsibility
- Ecological fallacy
- Competing risk
- Informative censoring
- Publication bias
- Spin
- Immortal time bias
- Conditional probabilities
- Selection bias
- …



https://twitter.com/MaartenvSmeden/status/1498071542583881730

Scenario of our Running Example

```
FUNCTION PBMAIN
    DIM ma4(1 TO 4,1 TO 4) AS DOUBLE, Det AS DOUBLE, T$, I&, J&
    ' matrix
    ma4(1,1) =  1 : ma4(1,2) =  3 : ma4(1,3) = -3 : ma4(1,4) =  5
    ma4(2,1) =  4 : ma4(2,2) =  2 : ma4(2,3) =  1 : ma4(2,4) =  2
    ma4(3,1) =  3 : ma4(3,2) =  2 : ma4(3,3) = -2 : ma4(3,4) =  2
    ma4(4,1) =  0 : ma4(4,2) =  1 : ma4(4,3) =  2 : ma4(4,4) = -1
    Det = 1
    CALL MakeResultsString(ma4(),4,Det,T$,"Original")
    CALL MatrixInversion(ma4(), 4 , Det)
    CALL MakeResultsString(ma4(),4,Det,T$,"Inverted")
    CALL MatrixInversion(ma4(), 4 , Det)
    CALL MakeResultsString(ma4(),4,Det,T$,"Inversion of inverted matrix = Original")
    MSGBOX T$,,"Results:"
END FUNCTION
'
SUB MatrixInversion(A() AS DOUBLE, M AS LONG, Determinant AS DOUBLE)
    ' Gauss reduction inversion method.
    ' M is the order of the square matrix A()
    ' A() inverse is returned in A().
    ' Determinant is returned.
    LOCAL I, J, K, L AS LONG, T AS DOUBLE, Pivot AS DOUBLE
    Determinant = 1
    FOR J = 1 TO M
        Pivot = A(J,J) : A(J,J) = 1
        Determinant = Determinant * Pivot
        IF Determinant = 0 THEN MSGBOX "Matrix singular " _
          +  "- cannot invert",,"Problem": EXIT SUB
        '
        ' Divide pivot row with pivot element.
        FOR K = 1 TO M : A(J,K) = A(J,K) / Pivot : NEXT
        FOR K = 1 TO M
            ' Reduce the non pivot rows.
            IF K <> J THEN
                T = A(K,J) : A(K,J) = 0
                FOR L = 1 TO M : A(K,L) = A(K,L) - A(J,L) * T : NEXT
            END IF
        NEXT
    NEXT
END SUB
'
```



Our research goal: Estimate execution time of functions without executing the code

Step 1: Feature Selection

```
FUNCTION PBMAIN
    DIM ma4(1 TO 4,1 TO 4) AS DOUBLE, Det AS DOUBLE, T$, I&, J&
    ' matrix
    ma4(1,1) =  1 : ma4(1,2) =  3 : ma4(1,3) = -3 : ma4(1,4) =  5
    ma4(2,1) =  4 : ma4(2,2) =  2 : ma4(2,3) =  1 : ma4(2,4) =  2
    ma4(3,1) =  3 : ma4(3,2) =  2 : ma4(3,3) = -2 : ma4(3,4) =  2
    ma4(4,1) =  0 : ma4(4,2) =  1 : ma4(4,3) =  2 : ma4(4,4) = -1
    Det = 1
    CALL MakeResultsString(ma4(),4,Det,T$,"Original")
    CALL MatrixInversion(ma4(), 4 , Det)
    CALL MakeResultsString(ma4(),4,Det,T$,"Inverted")
    CALL MatrixInversion(ma4(), 4 , Det)
    CALL MakeResultsString(ma4(),4,Det,T$,"Inversion of inverted matrix = Original")
    MSGBOX T$,,"Results:"
END FUNCTION
'
SUB MatrixInversion(A() AS DOUBLE, M AS LONG, Determinant AS DOUBLE)
    ' Gauss reduction inversion method.
    ' M is the order of the square matrix A()
    ' A() inverse is returned in A().
    ' Determinant is returned.
    LOCAL I, J, K, L AS LONG, T AS DOUBLE, Pivot AS DOUBLE
    Determinant = 1
    FOR J = 1 TO M
        Pivot = A(J,J) : A(J,J) = 1
        Determinant = Determinant * Pivot
        IF Determinant = 0 THEN MSGBOX "Matrix singular " _
          +  "- cannot invert",,"Problem": EXIT SUB
        '
        ' Divide pivot row with pivot element.
        FOR K = 1 TO M : A(J,K) = A(J,K) / Pivot : NEXT
        FOR K = 1 TO M
            ' Reduce the non pivot rows.
            IF K <> J THEN
                T = A(K,J) : A(K,J) = 0
                FOR L = 1 TO M : A(K,L) = A(K,L) - A(J,L) * T : NEXT
            END IF
        NEXT
    NEXT
END SUB
'
```

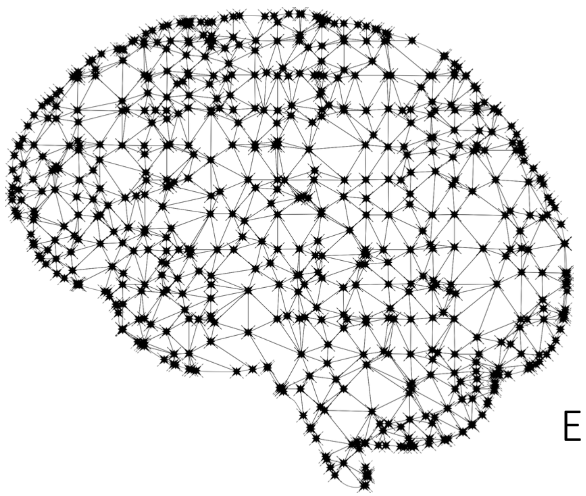
M1: #LOC
M2: #Loops
M3: #LOC in Loops
M4: #Variables
M5: #Operations
M6: #Operations in Loops
M7: Cyclomatic complexity

Why these features and not others? Why do they make sense?
Do you have a hypothesis that the individual features are useful for performance estimation?

Confirmatory research: Communicate the rationale of your feature selection, state hypothesis upfront to make it easier to explain why somethings (not) works
Exploratory research: Requires sensitivity /qualitative analysis later

Note that we will learn more aspects on how to chose a proper model later in the course.

Step 2: Algorithm Selection



Deep learning

Last resort: Learned features unknown
Often, no insights provided

Linear regression

Classification and regression trees

Ensemble learners (e.g., random forests)

Support vector regression

Selection needs to be driven by the goal:
explainability vs. accuracy vs. speed
If not clear -> independent variables

Easy over Hard: A Case Study on Deep Learning

Wei Fu, Tim Menzies
Computer Science Department, North Carolina State University
890 Oval Drive
Raleigh, North Carolina 27606
wfu@ncsu.edu,tim.menzies@gmail.com

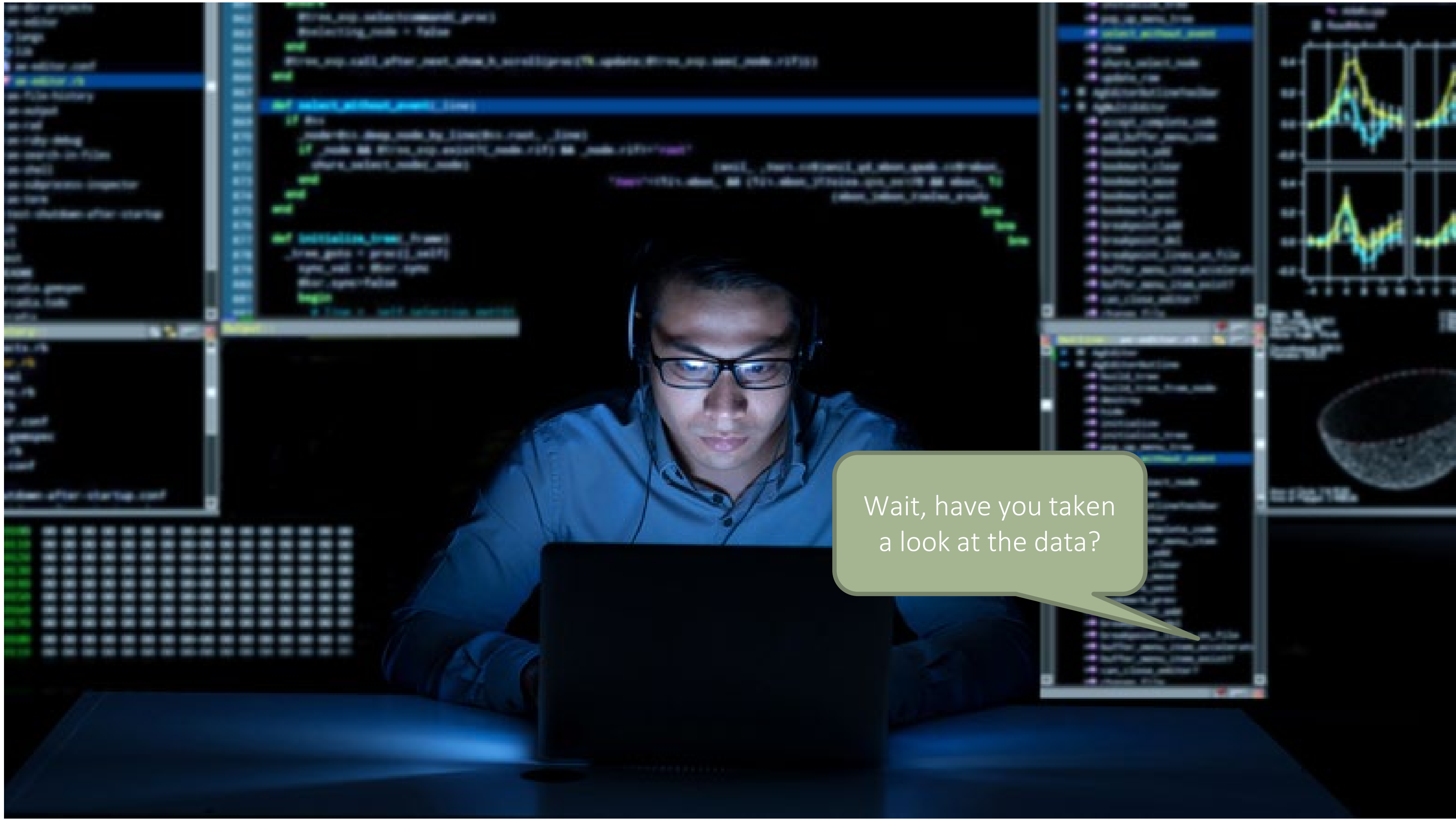
ABSTRACT
While deep learning is an exciting new technique, the benefits of this method need to be assessed w.r.t. its computational cost. This is particularly important for deep learning since these learners need hours (to weeks) to train the model. Such long CPU times limit the ability of (a) a researcher to test the stability of their conclusion via repeated runs with different random seeds; and (b) other researchers to repeat, improve, or even refute that original work.

For example, recently, deep learning was used to find which questions in the Stack Overflow programmer discussion forum can

a question along with its entire set of answers posted on Stack Overflow as a *knowledge unit* (KU). If two knowledge units are semantically related, they're considered as *linkable* knowledge units.

In their paper, they used a convolution neural network (a kind of deep leaner [42]) to predict whether two knowledge units are linkable. Such CNNs are highly computationally expensive, often requiring network composed of 10 to 20 layers, hundreds of millions of weights and billions of connections between units [42]. Even with advanced hardware and algorithm parallelization, training deep learning models still requires hours to weeks. For example:

Start experimenting…



Step 3: Formulate Expectations

```
FUNCTION PBMAIN
    DIM ma4(1 TO 4,1 TO 4) AS DOUBLE, Det AS DOUBLE, T$, I&, J&
    ' matrix
    ma4(1,1) =  1 : ma4(1,2) =  3 : ma4(1,3) = -3 : ma4(1,4) =  5
    ma4(2,1) =  4 : ma4(2,2) =  2 : ma4(2,3) =  1 : ma4(2,4) =  2
    ma4(3,1) =  3 : ma4(3,2) =  2 : ma4(3,3) = -2 : ma4(3,4) =  2
    ma4(4,1) =  0 : ma4(4,2) =  1 : ma4(4,3) =  2 : ma4(4,4) = -1
    Det = 1
    CALL MakeResultsString(ma4(),4,Det,T$,"Original")
    CALL MatrixInversion(ma4(), 4 , Det)
    CALL MakeResultsString(ma4(),4,Det,T$,"Inverted")
    CALL MatrixInversion(ma4(), 4 , Det)
    CALL MakeResultsString(ma4(),4,Det,T$,"Inversion of inverted matrix = Original")
    MSGBOX T$,,"Results:"
END FUNCTION
'
SUB MatrixInversion(A() AS DOUBLE, M AS LONG, Determinant AS DOUBLE)
    ' Gauss reduction inversion method.
    ' M is the order of the square matrix A()
    ' A() inverse is returned in A().
    ' Determinant is returned.
    LOCAL I, J, K, L AS LONG, T AS DOUBLE, Pivot AS DOUBLE
    Determinant = 1
    FOR J = 1 TO M
        Pivot = A(J,J) : A(J,J) = 1
        Determinant = Determinant * Pivot
        IF Determinant = 0 THEN MSGBOX "Matrix singular " _
          +  "- cannot invert",,"Problem": EXIT SUB
        '
        ' Divide pivot row with pivot element.
        FOR K = 1 TO M : A(J,K) = A(J,K) / Pivot : NEXT
        FOR K = 1 TO M
            ' Reduce the non pivot rows.
            IF K <> J THEN
                T = A(K,J) : A(K,J) = 0
                FOR L = 1 TO M : A(K,L) = A(K,L) - A(J,L) * T : NEXT
            END IF
        NEXT
    NEXT
END SUB
'
```

M1: 20
M2: 0
M3: 0
M4: 10
M5: 42
M6: 0
M7: 3

25ms

M1: 40
M2: 3
M3: 17
M4: 3
M5: 22
M6: 8
M7: 9

437ms

Is the problem actually learnable? How much data is needed? Are there theoretical bounds and guarantees about accuracy? How is the data distributed?

Try to make the learning problem as easy as possible by engineering suitable features.
Try to get an intuition about the complexity of the learning problem -> linearity, discontinuity, interaction degree, size of search space, determinism, uncertainty, etc.

Data Alone is not Enough

When learning a Boolean function with 100 variables, are a million samples enough?

$2^{100} - 10^{6}$ = 1,267,650,600,228,229,401,496,702,205,376 unknown classes

Flipping a coin is maybe the best way

No free lunch theorem by Wolpert'96:
No learner can beat random guessing over all possible functions to be learned.
-> Embody some knowledge or assumptions to the learning algorithm beyond the data
Assumptions: smoothness, similar examples have similar classes, limited dependences, limited complexity, etc. often hold

A Few Useful Things to Know about Machine Learning

Pedro Domingos
Department of Computer Science and Engineering
University of Washington
Seattle, WA 98195-2350, U.S.A.
pedrod@cs.washington.edu

ABSTRACT
Machine learning algorithms can figure out how to perform important tasks by generalizing from examples. This is often feasible and cost-effective where manual programming is not. As more data becomes available, more ambitious problems can be tackled. As a result, machine learning is widely used in computer science and other fields. However, developing successful machine learning applications requires a substantial amount of "black art" that is hard to find in textbooks. This article summarizes twelve key lessons that machine learning researchers and practitioners have learned. These include pitfalls to avoid, important issues to focus on, and answers to common questions.

correct output y_t for future examples $\mathbf{x_t}$ (e.g., whether the spam filter correctly classifies previously unseen emails as spam or not spam).

2. LEARNING = REPRESENTATION + EVALUATION + OPTIMIZATION

Suppose you have an application that you think machine learning might be good for. The first problem facing you is the bewildering variety of learning algorithms available. Which one to use? There are literally thousands available, and hundreds more are published each year. The key to not getting lost in this huge space is to realize that it consists of combinations of just three components. The components are:

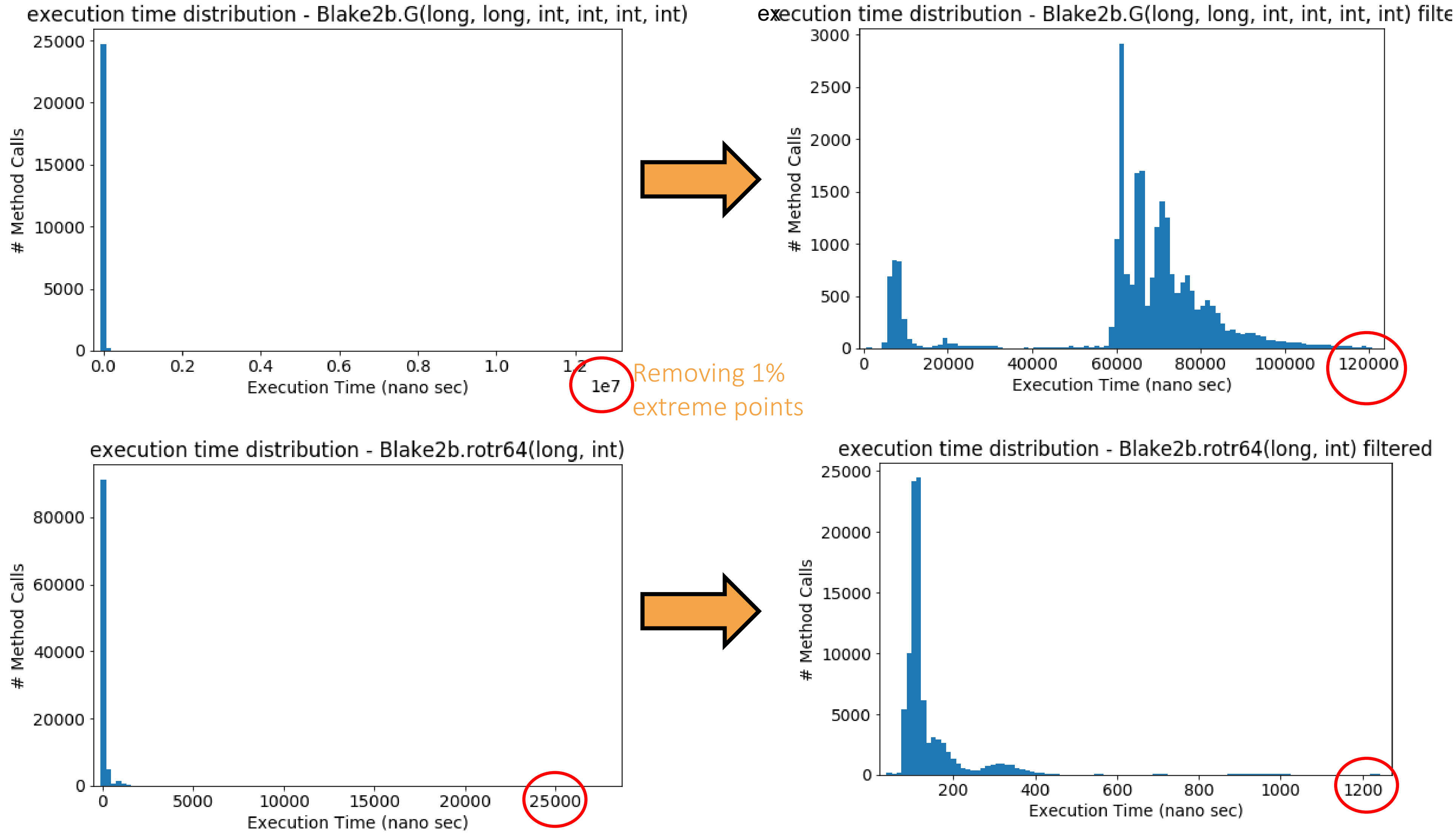
ARTICLE **Communicated by Steven Nowlan**

The Lack of A Priori Distinctions Between Learning Algorithms

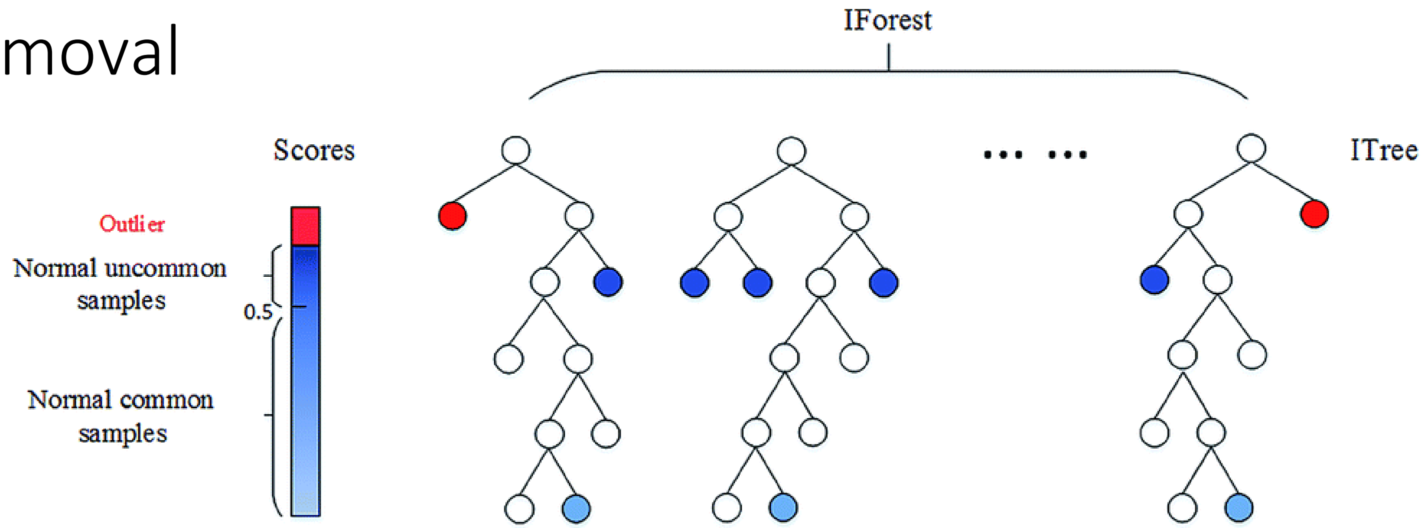
David H. Wolpert
The Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM, 87501, USA

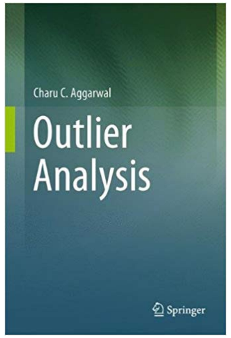
Encode your domain knowledge into the AI
Prior assumptions guide selection of learning algorithm (representation)

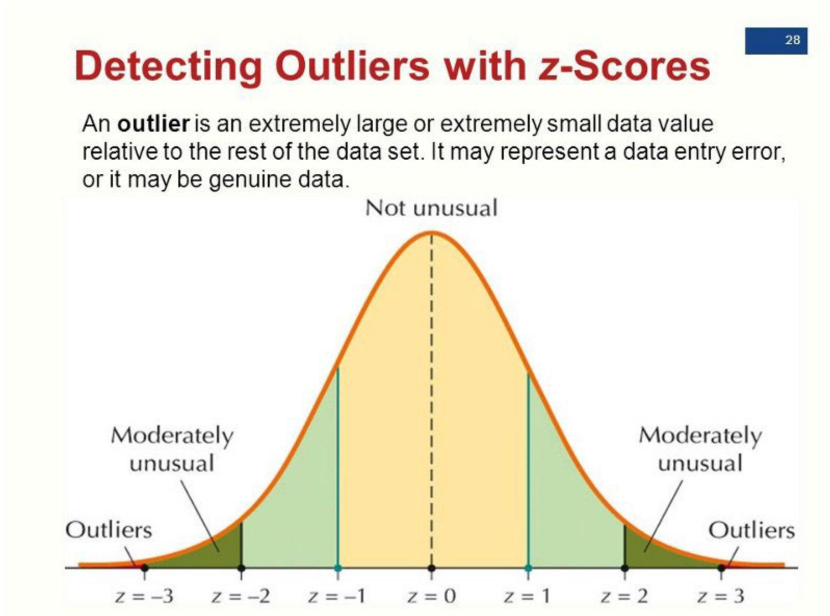
Outlier Removal



Outlier Removal







Detect and handle outliers (try to explain their root cause)

Z-score z_i indicates how much a given value x_i differs from the standard deviation. ($\bar{x}$:mean, s: std)

$$z_i = \frac{x_i - \bar{x}}{s}$$

https://heartbeat.fritz.ai/how-to-make-your-machine-learning-models-robust-to-outliers-44d404067d07

Before Start Learning … Let us Talk About Value Ranges

What are the value ranges of our metrics?

How can we cope with that?

- Normalization:

$$X' = \frac{X - X_{\mathbf{min}}}{X_{\mathbf{max}} - X_{\mathbf{min}}}$$

Scales features within the interval [0,1]

When to do normalization? Before or after the training-validation split?

Best practice consider normalization prior to the split to best inform the model and have a larger data range covered for better generalization (store normalization values, such as min, max, mean, std).

Step 4: Experimental Setup

- (Hypothesis formulation)
- Independent variables
- Dependent variables (which metrics to measure success?)
- Controlling validity threats, for example:
  - Internal validity:
    - Data splitting
    - Measurement bias
  - External validity:
    - Generalization error, over- and underfitting
    - Generality of data set (also, more realistic)
    - Reproducibility
- Comparing against competitors / state of the art
  - Meta/hyperparameter tuning

Hypothesis Formulation in Data Science Projects

Goal: Cleary articulate what the experiment is trying to test and identify which variables to be measured

Enables to collect relevant data and helps verifying whether the experiment contributes to the project goal and accompanioned research questions.

Hypothesis ease reproduction since we clearly formulate what to do.

Avoids data dredging: testing multiple hypotheses until a statistically significant result has been found
Reduced the danger of overfitting

Communicates expectations on the results

Experimentation: Independent and Dependent Variables

Independent variables: variables manipulated by the research; independent because its values are no dependent on any other variable in the experiment; assumed to have a causal effect on the dependent variable

Dependent variables: variables that are being measured or observed by the researcher; changes in the independent variables are expected to change (or not) the dependent variable; expectations are formulated in the hypotheses

Examples:

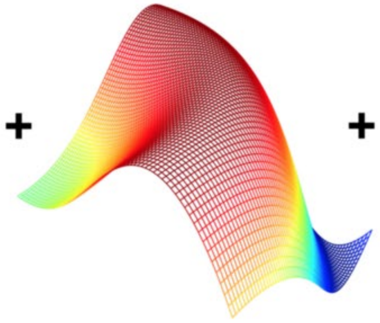
Independent variables: used features (e.g., code metrics), used training algorithms, used preprocessing, etc.

Dependent variables: accuracy of predictions, different metrics computed on prediction results

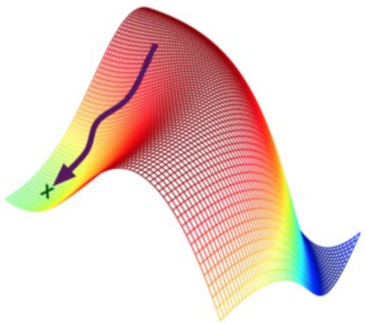
Independent Variables

Learning = Representation + Evaluation + Optimization

+



+



Instances (e.g., K-nearest neighbor)
Hyperplanes (e.g., logistic regression)
Decision trees
Rule systems
Neural networks
…
Depends on goal
and assumptions

Accuracy / Error rate
Precision and recall
Squared error
Likelihood
K-L divergence
…
Depends on goal
and representation

Combinatorial optimization (e.g. ant system, CSP)
Continuous optimization (e.g., gradient decent)
Constraint continuous (e.g., linear programming)
…

Depends on representation

This defines your evaluation setup and substantially influences validity of your experiments.

Forms of Validity

Internal validity: Internal validity refers to the extent to which a study or experiment is free from bias and accurately measures the effect of the independent variable on the dependent variable. In data science experiments, internal validity refers to the degree to which the results obtained from a model accurately represent the relationship between the independent and dependent variables in the underlying population.

Threats: Selection bias, data leakage
Mitigation strategies: train-test split, k-fold cross-validation, leave-one-out cross-validation

External validity: External validity refers to the generalizability of the findings from a study or experiment to other populations, settings, and conditions. In data science experiments, external validity refers to the degree to which the results obtained from a model can be generalized to new, unseen data.

Threats: Sampling bias, oversimplistic models (see bias-variance tradeoff late)
Mitigation strategies: Diverse data set, check coverage of population characteristics, settings, and conditions

Internal Validity: Selection Bias

Experimental setup:

Software system A

A

$a_1()$
$a_2()$
$a_3()$
$a_4()$
…
$a_n()$

x

M

M1
M2
M3
M4
M5
M6
M7

Software system B

B

$b_1()$
$b_2()$
$b_3()$
$b_4()$
…
$b_m()$

	A	B	C	D
1	Method	Metric	Value	Time
2	a1	M1	20	5
3	a1	M2	10	5
4	a1	M3	5	5
5	a1	M4	37	5
6	..	..	..	..
7	a1	M7	42	5
8	a2	M1	50	12
9	..	..	..	..
10	a2	M7	12	12
11	..	..	..	..
12	an	M1	2	145
13	..	..	..	..
14	an	M7	0	145
15	b1	M1	45	69
16	..	..	..	..
17	bm	M7	4	2

vs

	A	B	C	D	E	F	G	H	I
1	Method	M1	M2	M3	M4	M5	M6	M7	Time
2	a1	20	10	5	37	12	4	42	5
3	a2	12	5	0	0	0	4	3	12
4	..	..	..	..	..	..	..	..	..
5	an	2	1	2	2	0	2	2	145
6	b1	45	2	4	56	0	54	4	69
7	b2	166	57	3	4	43	53	3	44
8	..	..	..	..					..
9	bm	13	12	5	8	0	0	4	2

Learning Testing

How to split?

Selection bias: training set is not representation of the population

Internal Validity: Data Leakage

(1) Random split with first encoding

Random: (method, metric)

$L = \{(f, m)\} \mid f \in A \cup B \wedge m \in M \wedge f, m \text{ randomly chosen}$

$T = ((A \cup B) \times M) \setminus L$

Response times in test set are already present in learning set! -> internal threat

Learning L	Testing T
a_1():M2,M5,M7	a_1():M1,M3,M4,M6
a_2():M2,M3,M5,M6	a_2():M1,M4,M7
…	…
a_n():M1,M3,M4	a_n():M2,M3,M4
b_1():M1,M2	b_1():M3,M4, …
b_2():M5	b_2():M2,M3,…
…	…
b_m():M2,M4,M5	b_m():M1,M3,M6,M7

(2) Random split with second encoding

Random: (method)

$L = \{(f, M)\} \mid f \in A \cup B \wedge f \text{ randomly chosen}$

$T = ((A \cup B) \times M) \setminus L$

a_1(), a_5(), … a_{n-1}() x M1..M7	a_2(), … a_n() x M1..M7
b_2(), b_3(), … b_m() x M1..M7	b1(), … b_{m-1}() x M1..M7

Response times of either system are already in learning set! -> external threat

What to do?

Make it explicit in the experiment design, for example:

- Formulate research questions whether the approach can learn estimating response time (a) within a system or (b) across systems
- Draw conclusions from this (e.g., domain dependence, API dependence, programmer style, etc.)

Make application scenario of the approach clear and evaluate it accordingly

Further issues:
How to sample methods if too many?
How to obtain the ground truth?
Does the process of collecting the label/ground truth affects the outcome?

Topic II:
Experiment Analysis

Metrics & Baselines

What is a metric?

An indicator to measure a certain quantitative property of interest. Different metrics measure different properties of a subject, for instance, accuracy, edit distance, #outliers, false positives, F1-score, etc. Usually, a metric does not match exactly to an objective (e.g., usefulness, practicality, business success), but acts as a proxy: accuracy ~ practicality. Multiple, metrics may be needed to get a better picture.

What is a baseline?

Achieved value/score of a metric of a known process. A baseline acts as a minimal reachable target value/score and is used to have a point of reference for a new process / model. Gives a sense of about the irreducible error.

Metrics & Baselines

Metrics can rate the quality of the model, but also link to non-functional requirements of the software systems (e.g., inference time). Improving all metrics ***simultaneously*** is often ***not feasible*** and also not necessary. Instead, concentrate on 1-2 metrics to improve and set certain thresholds to all remaining metrics.

Use baselines to identify **sensible thresholds** and know when and by how much a system improves.

Speech Recognition: What to optimize?

Source	Accuracy	% of data
Clear speech	95%	60%
Background noise	89%	10%
Conversation	90%	5%
Low audio quality	72%	25%

What to optimize?

Source: Deeplearning.AI

Speech Recognition: What to optimize?

Source	Accuracy	% of data
Clear speech	95%	60%
Background noise	89%	10%
Conversation	90%	5%
Low audio quality	72%	25%

What to optimize?

Source: Deeplearning.AI

Establishing Baselines

Ask human subjects: Human level performance (HLP)

Literature search and open-source systems

Quick-and-dirty implementation

Performance of prior system

Simple statistical models

Random

Majority vote

How strong are these baselines?

Metric: Confusion / Error Matrix

Multi-class problem

	True A	True B	True C
Predicted A	**18**	5	3
Predicted B	3	**12**	4
Predicted C	6	4	**16**

Accuracy (# of all correct predictions / # all predictions)

Accuracy = $\frac{18+12+16}{18+5+3+3+12+4+6+4+16} = 0.64$

Two-class problem

	True A	True !A
Predicted A	**True positive (TP)**	False positive (FP)
Predicted !A	False negative (FN)	**True negative (TN)**

False alarm: Type I error

Missing prediction: Type II error

Measures for Classification Tasks

Recall: Measures the fraction of the actual class we correctly classified; called sensitivity or hit rate; high as possible

$$Recall = \frac{TP}{TP + FN}$$

	True A
Predicted A	True positive (TP)
Predicted !A	False negative (FN)

Miss-rate: Measures how many do we miss; low as possible

$$False\ negative\ rate = 1 - Recall = \frac{FN}{TP + FN}$$

Precision: Measures how often we are correct (accurate); high as possible

$$Precision = \frac{TP}{TP + FP}$$

	True A	True !A
Predicted A	True positive (TP)	False positive (FP)

False positive rate: Measures how often we misclassify; low as possible

$$False\ positive\ rate = \frac{FP}{FP + TN}$$

	True !A
Predicted A	False positive (FP)
Predicted !A	True negative (TN)

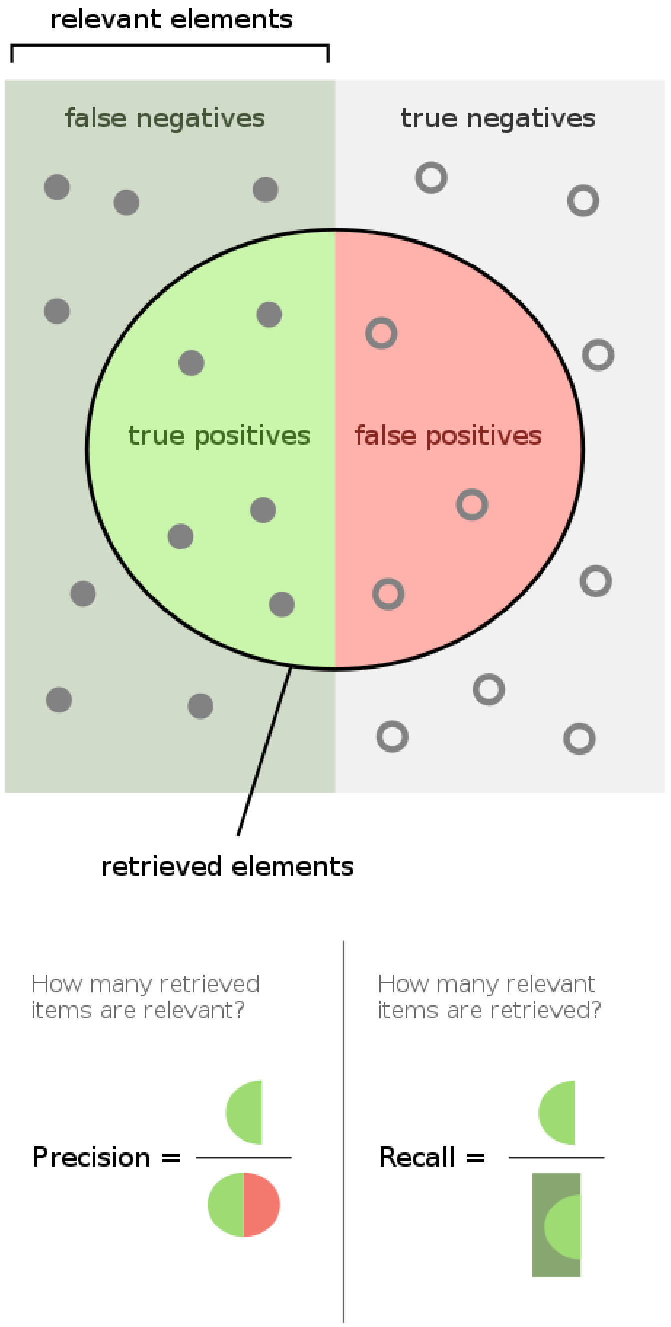
Harmonic Mean or F1-score

Goal: We need a measure to combine precision and recall

- Balances both metrics (harmonic mean)
- Punishes low score of a single metric
- Works also for multi-class problems

$$F1 = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

$$F1 = 2\frac{recall * precision}{recall + precision}$$



So, are false positives equally bad as false negatives?

Have you considered the base probability of the classes?
How many true positives exist at all? How does it affect precision & recall?

ROC Curve

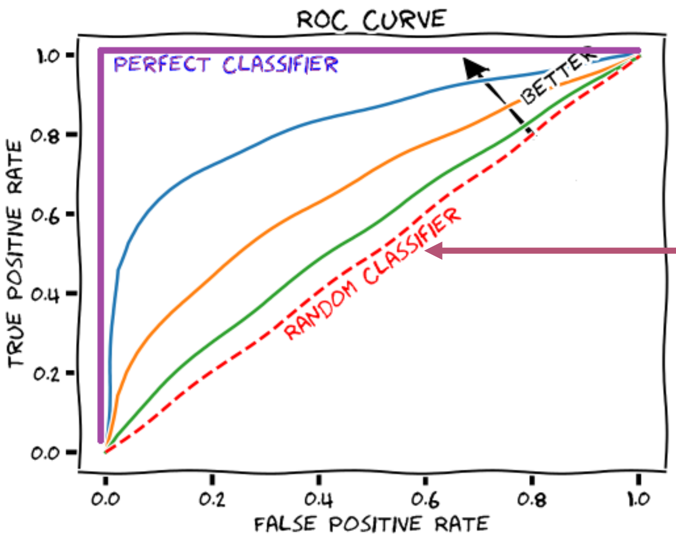
ROC (Receiver Operating Characteristic) curves plot the true positive rate $(TP\ /\ P)$ (i.e. **recall**) against the false positive rate $(FP\ /\ N)$ (i.e., **1 – precision**)

Useful to find optimal thresholds for classification tasks

```
from sklearn.metrics import roc_curve

fpr, tpr, thresholds = roc_curve(true_y, predicted_proba_y)
```

Fraction of actually correct samples are predicted correctly

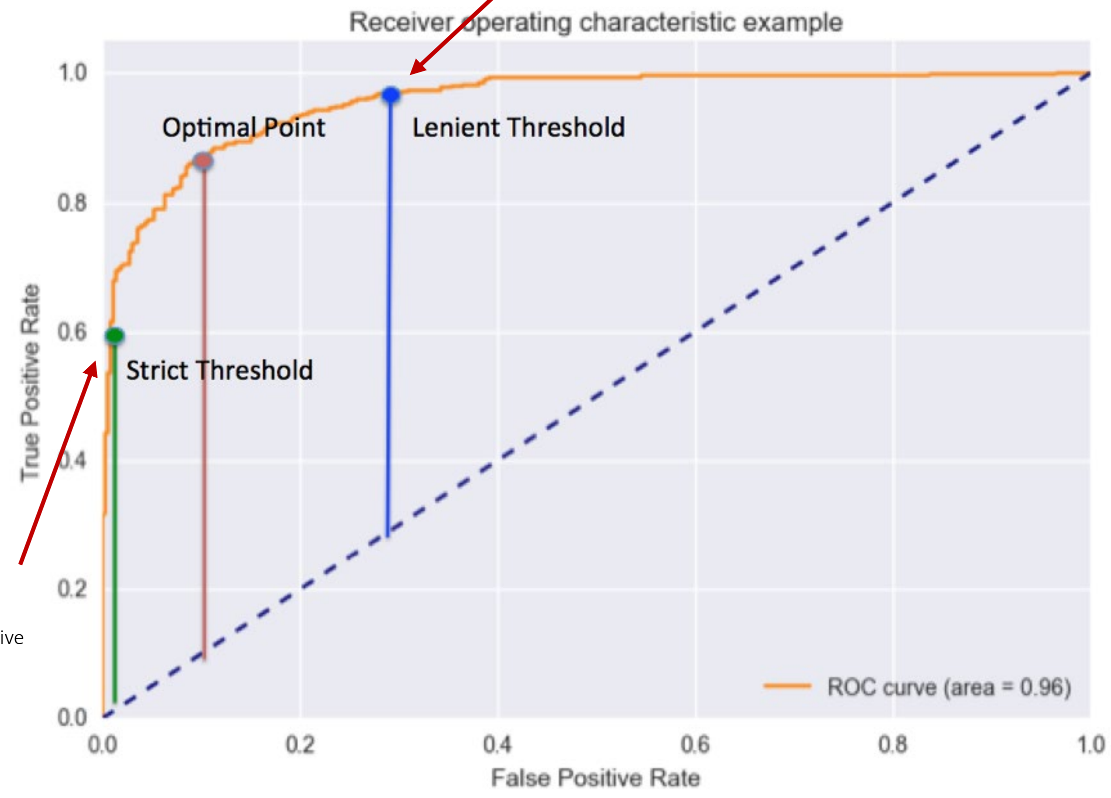


Area under the curve (AUROC) as a performance metric (the higher the better); 0.5 = random; >0.7 desired

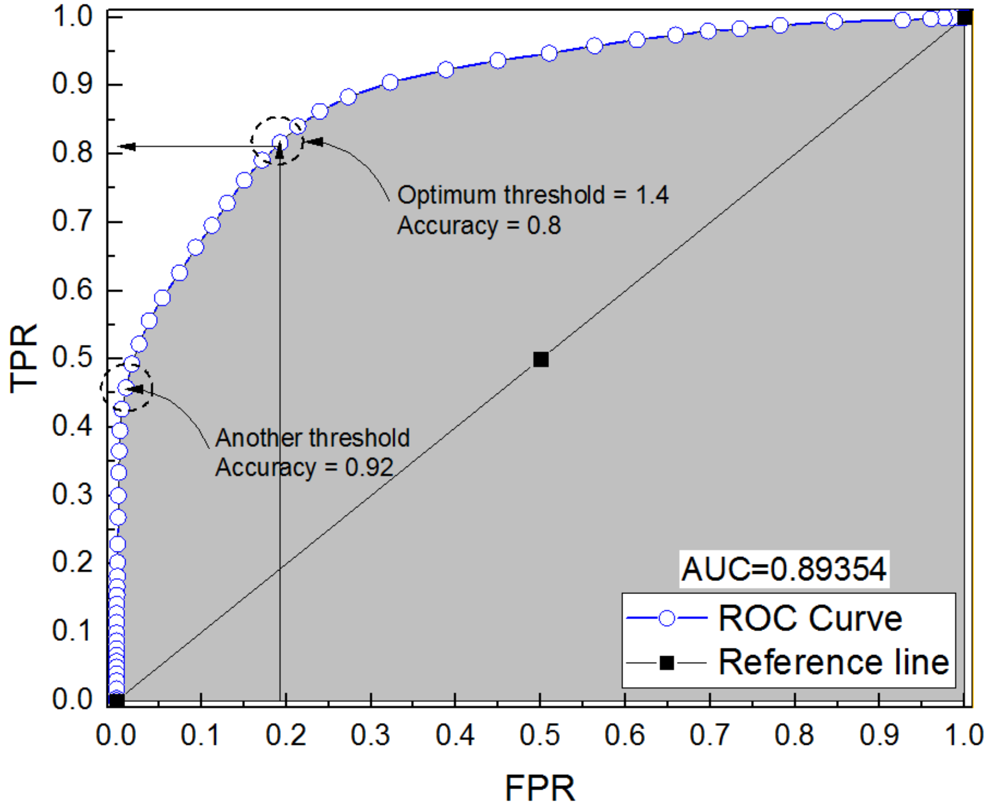
Fraction of actually false samples are predicted as correct

ROC Curve: Interpretation and Optimization

More likely to predict a positive outcome



Less likely to predict a positive outcome

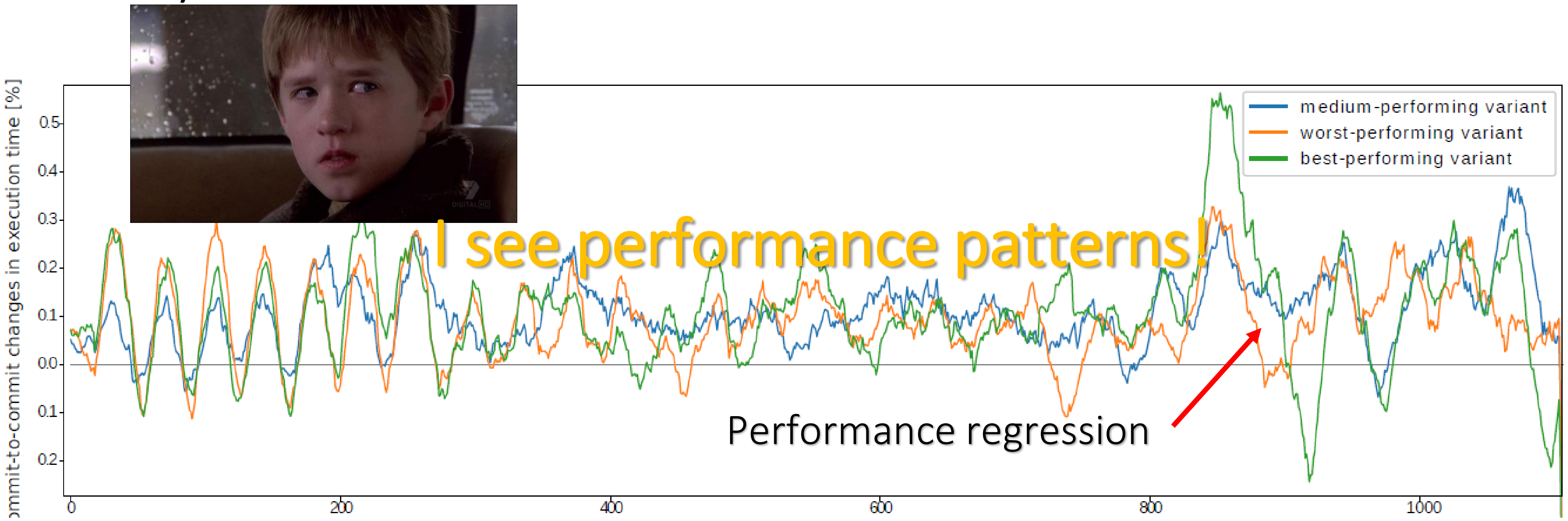


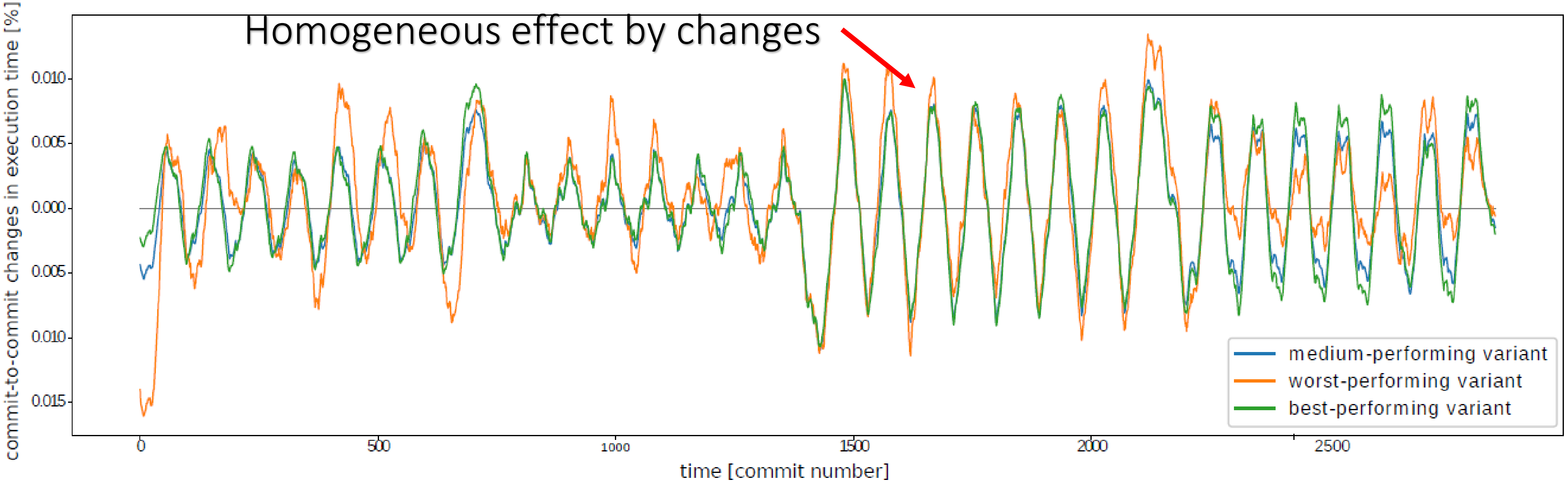
Can We Finally Start Learning?

	A	B	C	D	E	F	G	H	I
1	Method	M1	M2	M3	M4	M5	M6	M7	Time
2	a1	20	10	5	37	12	4	42	5
3	a2	12	5	0	0	0	4	3	12
4	..	..	..	..	..	..	..	..	..
5	an	2	1	2	2	0	2	2	145
6	b1	45	2	4	56	0	54	4	69
7	b2	166	57	3	4	43	53	3	44
8	..	..	..	..					..
9	bm	13	12	5	8	0	0	4	2

Do we have only a single measurement for each method?

Internal Validity: Measurement Bias





Producing Wrong Data Without Doing Anything Obviously Wrong!

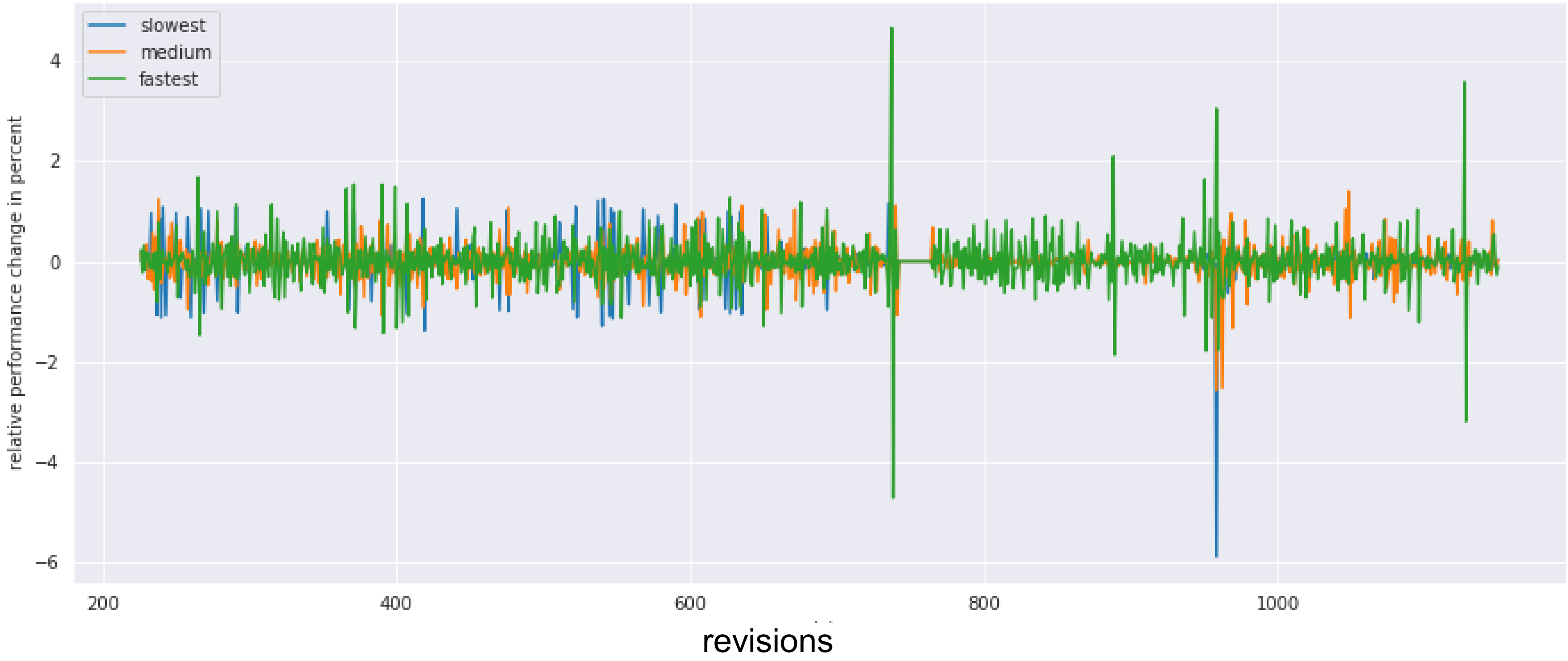
Todd Mytkowicz Amer Diwan
Department of Computer Science
University of Colorado
Boulder, CO, USA
{mytkowit,diwan}@colorado.edu

Matthias Hauswirth
Faculty of Informatics
University of Lugano
Lugano, CH
Matthias.Hauswirth@unisi.ch

Peter F. Sweeney
IBM Research
Hawthorne, NY, USA
pfs@us.ibm.com

ASPLOS'09

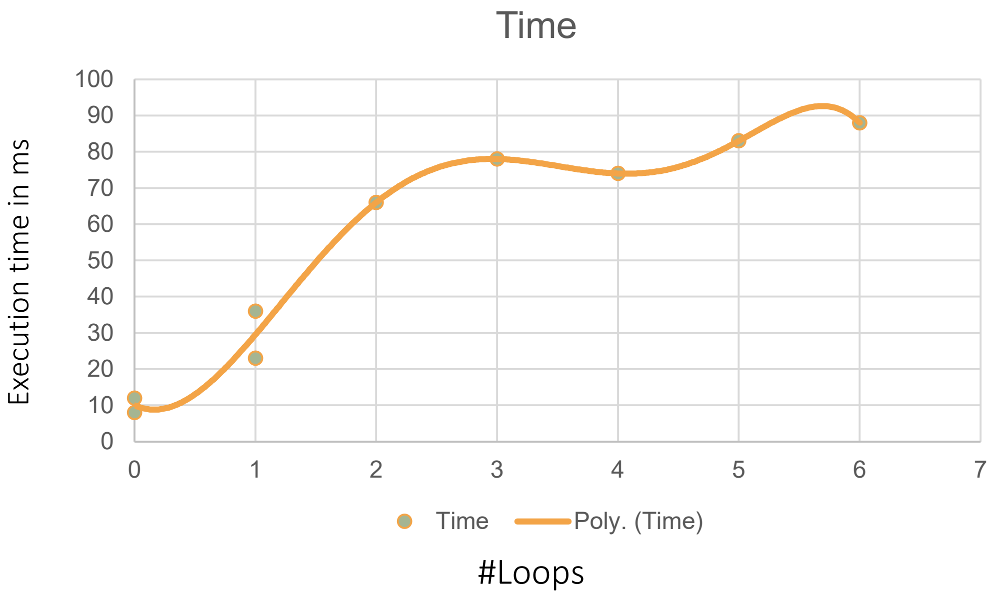
Repetitions are a Must!



Repeat initial measurements and assess the variations
Compute number of repetitions until the deviations are below a reasonable threshold
Determine random, non-deterministic measurements and exclude, but report them
(Are the excluded samples substantial wrt. the whole system?)

Start experimenting…

We Found Something! It Works!



Looks like we can predict method execution time accurately with just looking at number of loops!

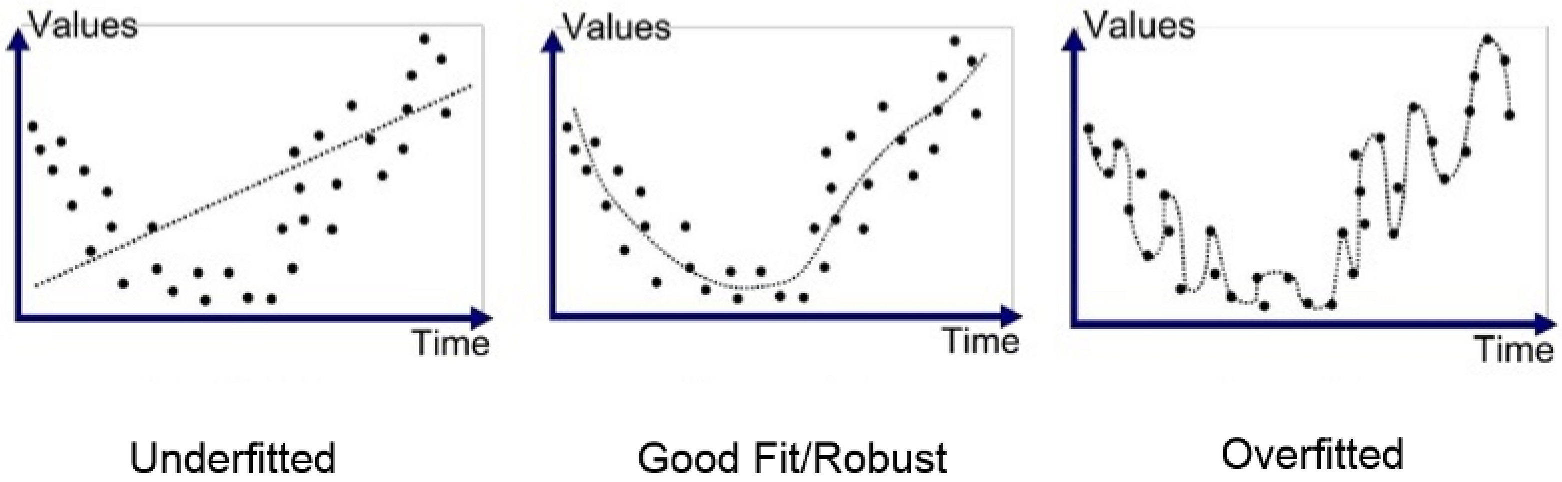
What is the main goal of our evaluation?

Goal: Evaluate to what degree our model generalizes to unseen problems!

External Validity / Generalization: Overfitting & Underfitting

Degrees of freedom

Too general trends



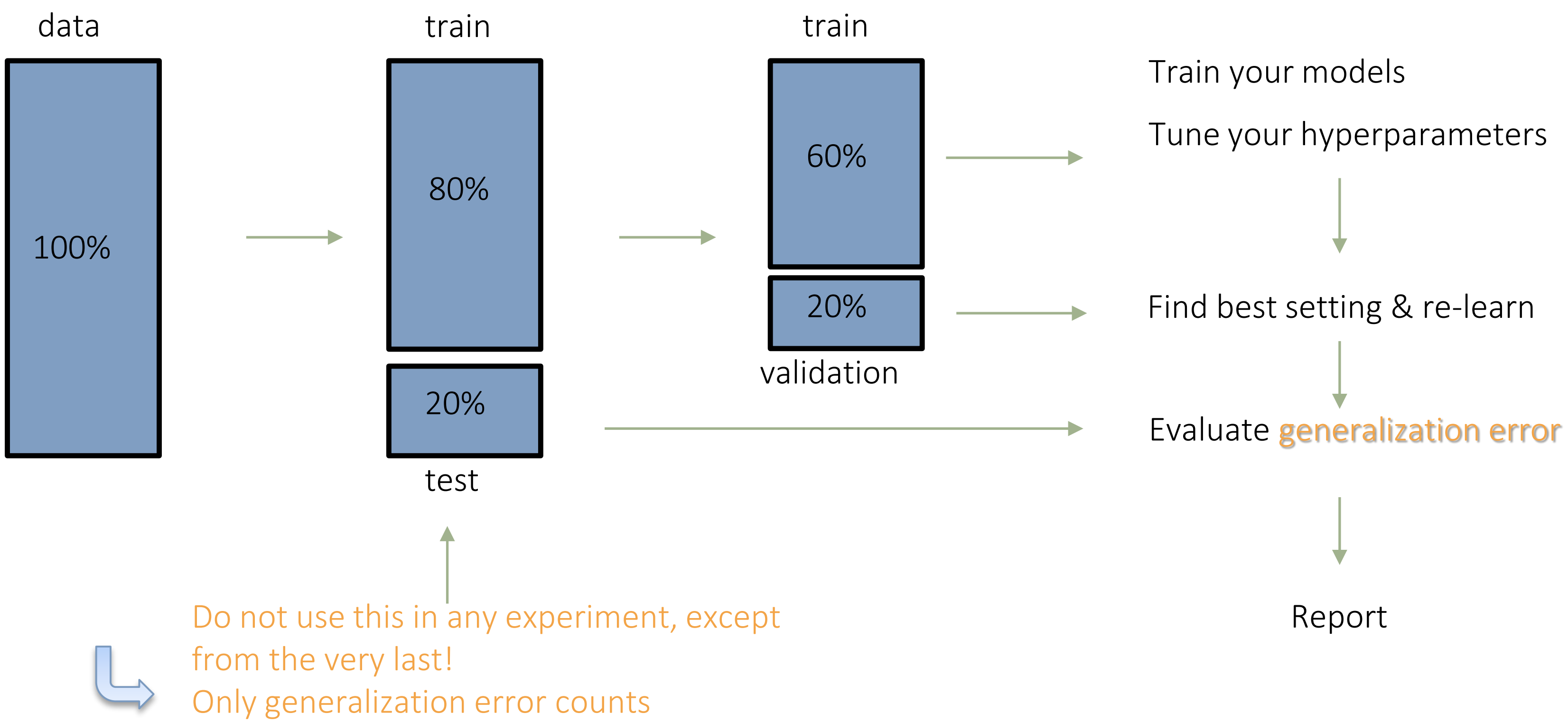
Memorized data set, including noise

Bad accuracy, but explainable

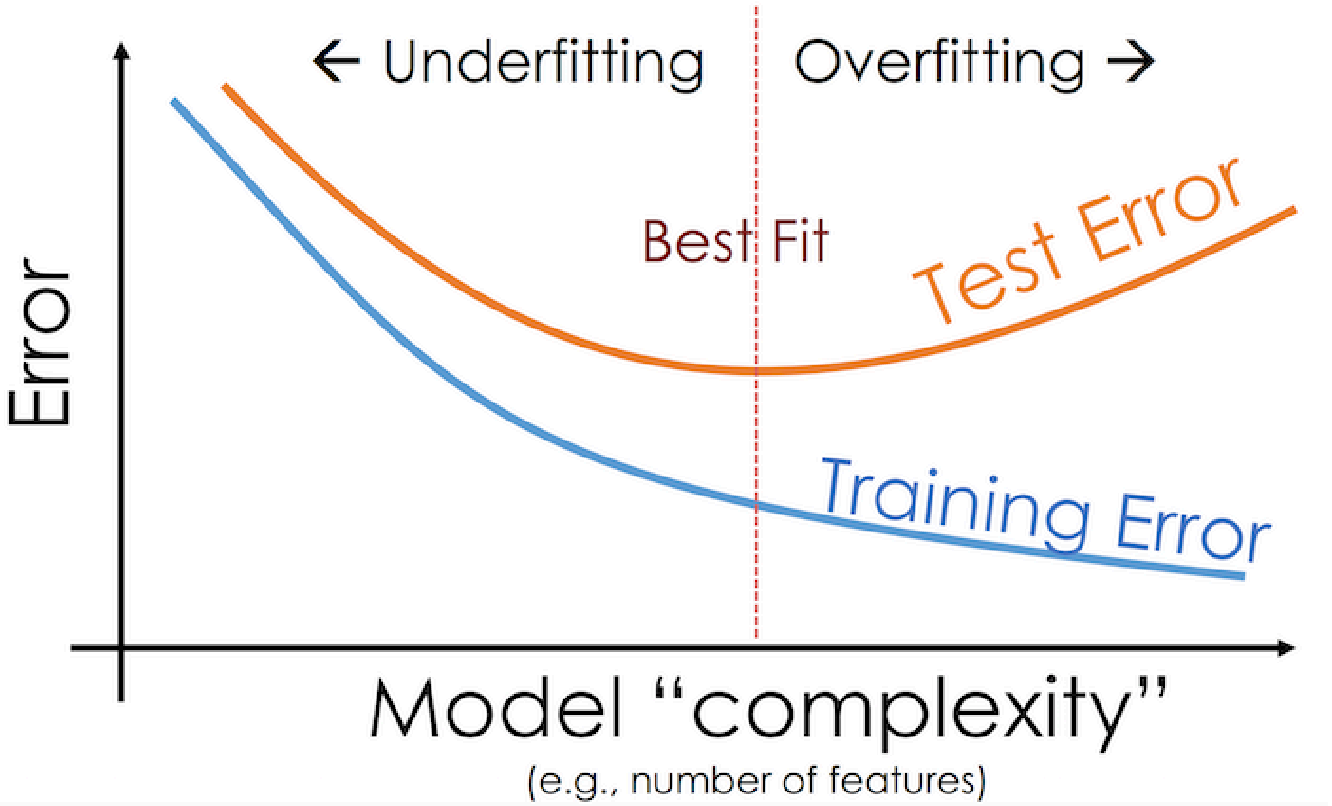
Ideal solution

Great accuracy in experiment, but not generalizable

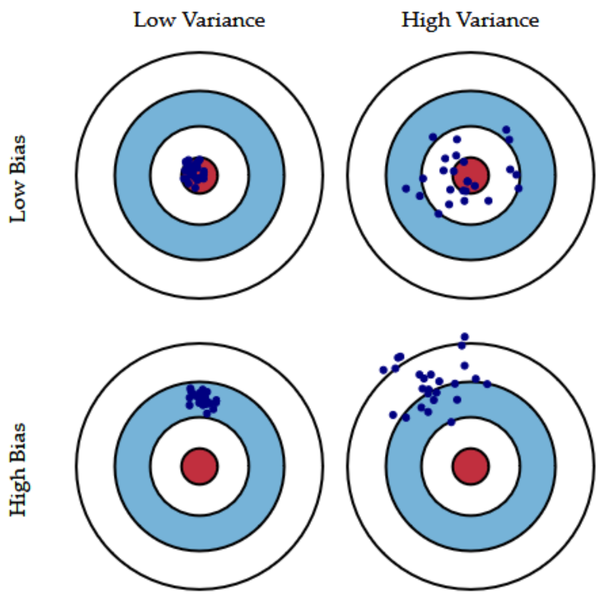
Assess Generalization Error with Test Set



Compare Generalization Error with Training Error



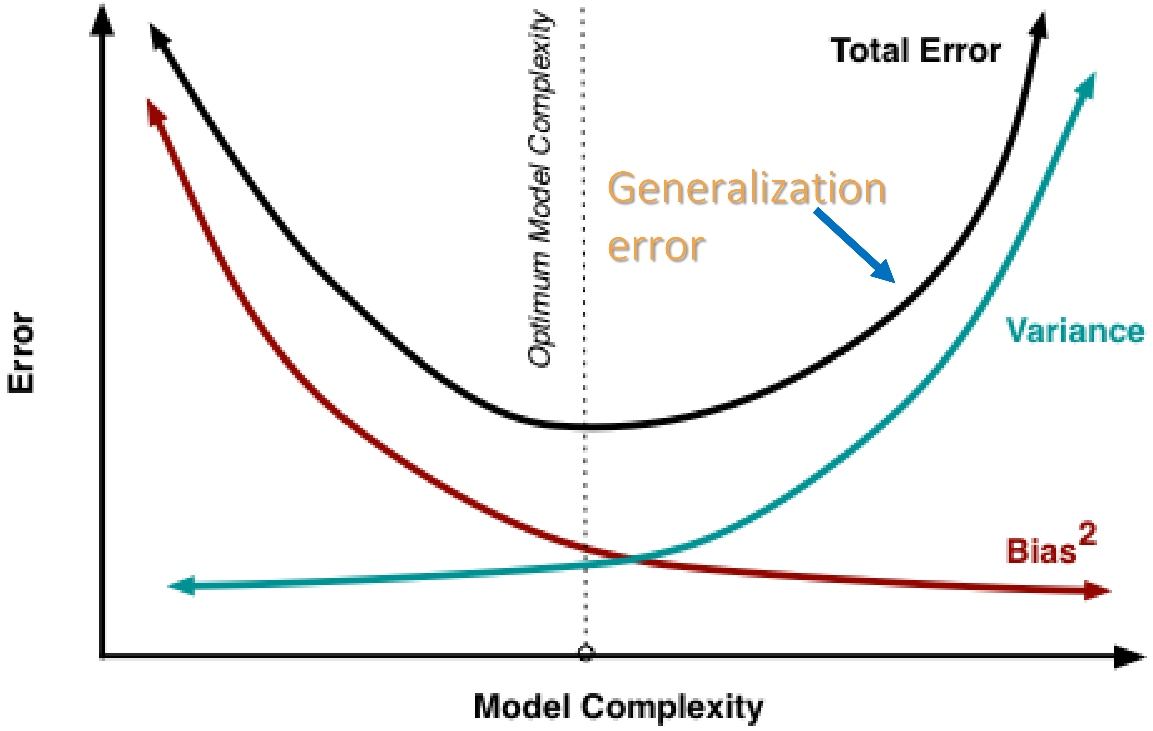
Bias – Variance Tradeoff



Bias: Degrees of freedom (or complexity) of an algorithm; assumptions made

Variance: How much does the estimate change for changes in the training data

↳ Do not favor minimize bias at the cost of variance

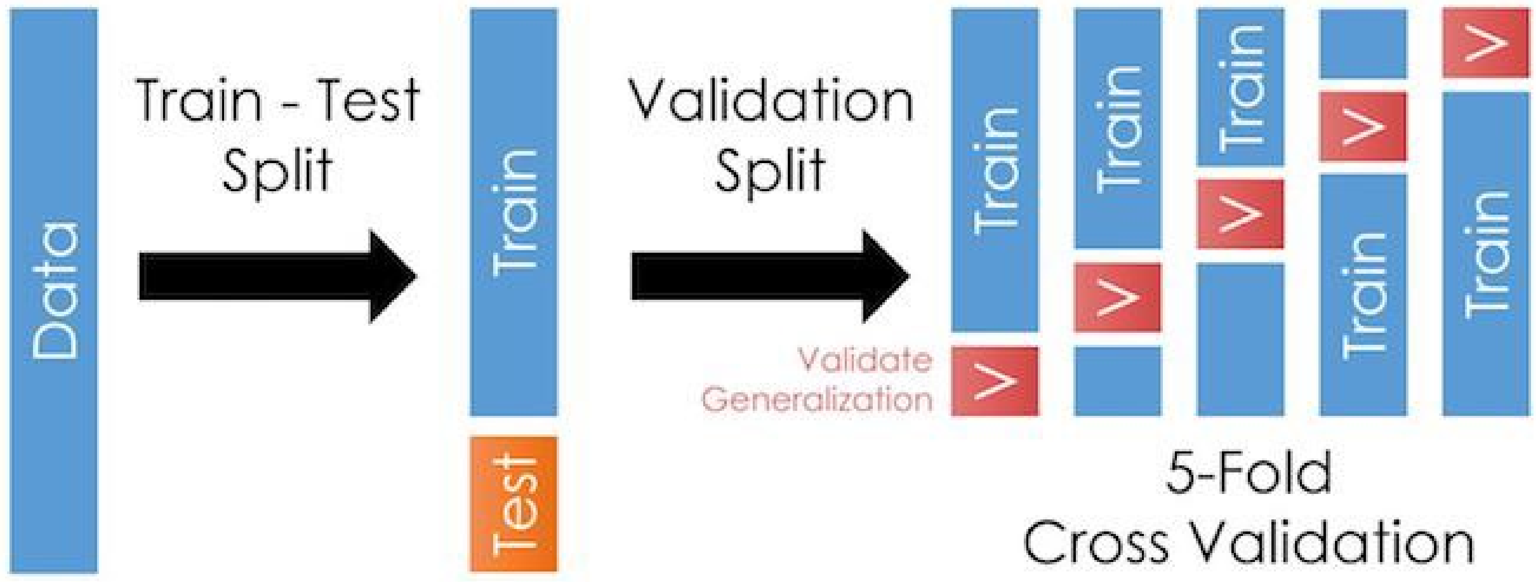


↳ Look into bagging and resampling techniques.

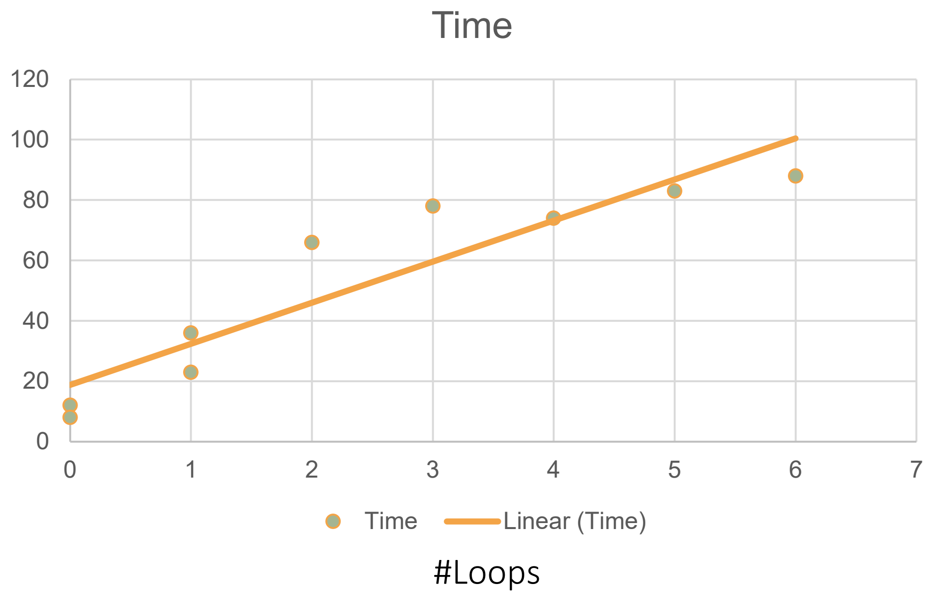
K-Fold Crossvalidation

Goal: Use validation split not one, but multiple times and get a better estimate of the generalization error already with the validation set

Benefits: Every data points is used in the validation only once, but k-1 times for training; averages errors of multiple training runs together (more robust)



We Found Something! Now, truly!

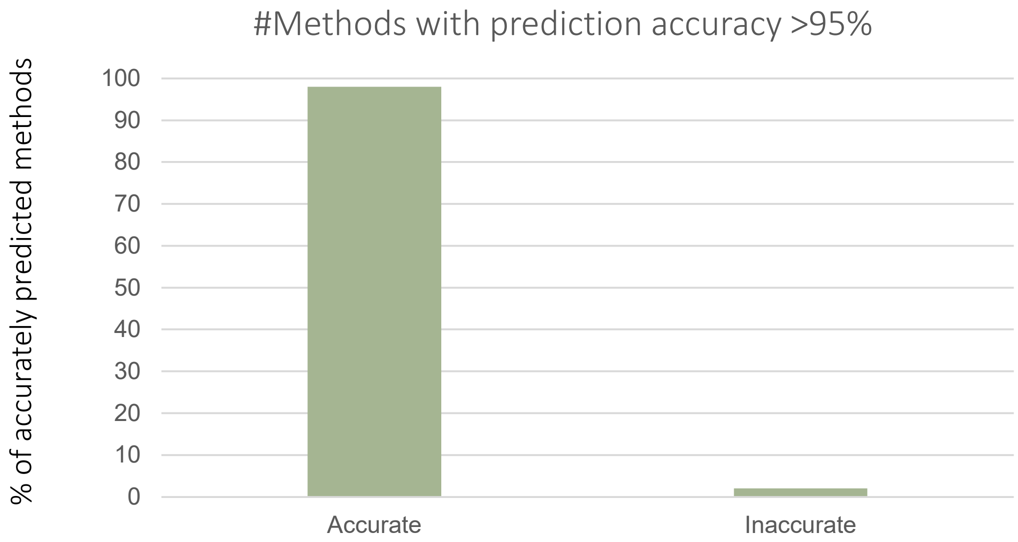


The more loops we have, the longer the method runs. So, reducing loops will speed up method execution?!

Correlation != Causation: A model can learn only correlations.

Here, this is not true… in order to keep the same functionality while reducing the number of loops, we may need to introduce additional code or recursive methods. The result would an increased execution time!

How is our Accuracy Doing?



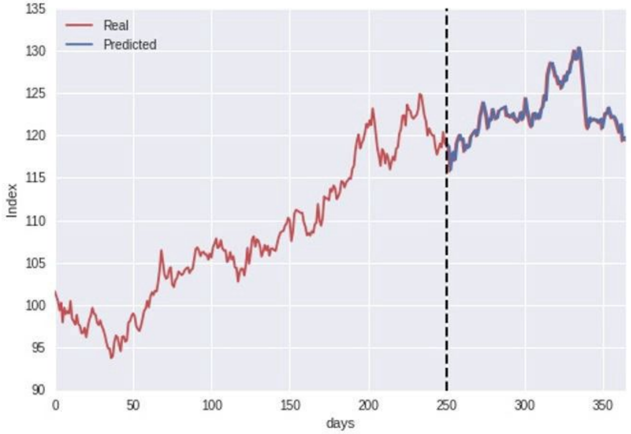
80% of methods are getter/setter with <1ms

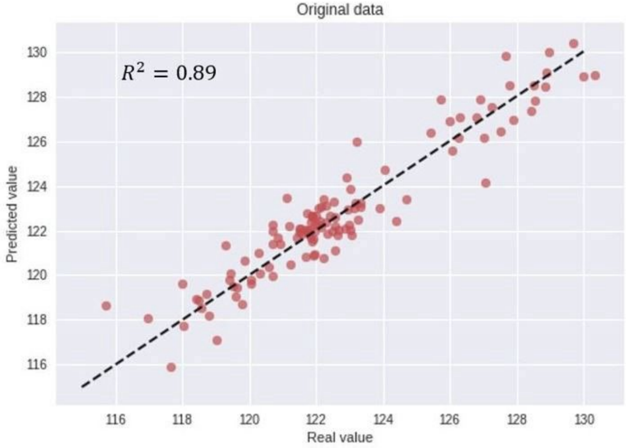
?

Another result:

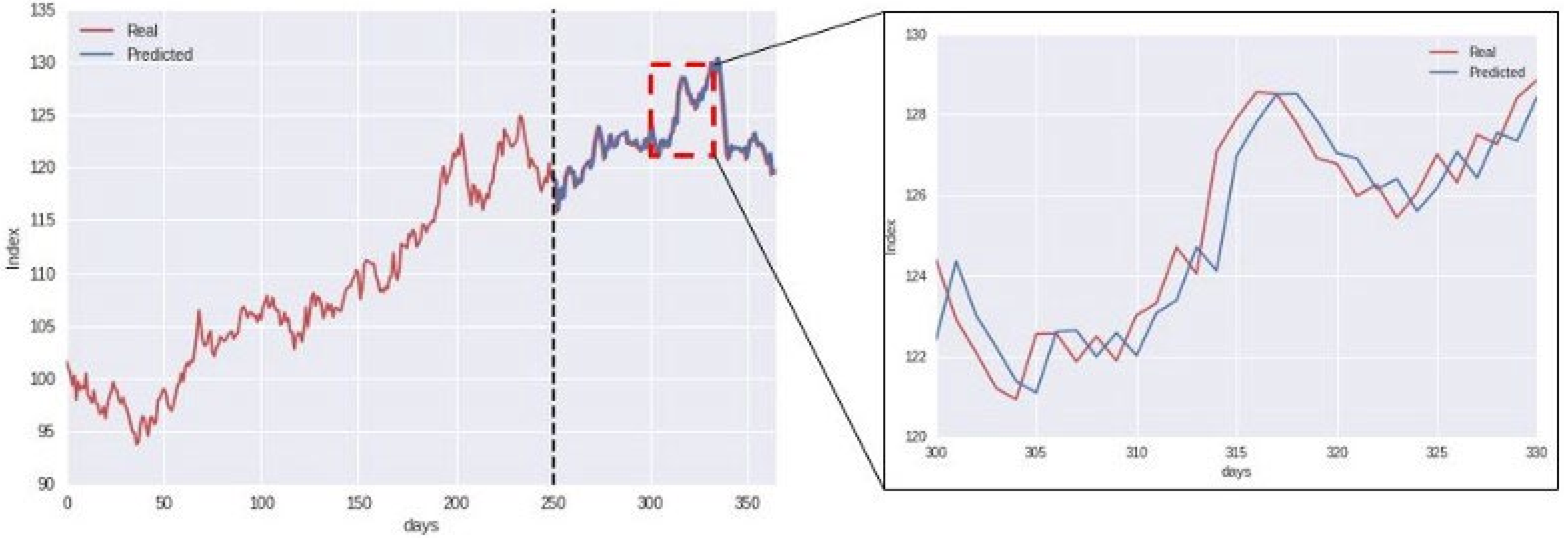
Look at your results from different angles
Answer your research questions honestly

Test: What Have you Learned?





Autocorrelation is the problem: value at t+1 is close to value at t



Solution: Change Setup

Predict the difference in values between time steps rather than the value itself