## **Introduction into Data Science** Lecture 1 – Block 1, 24.10.

Data Science – gaining insights into data through statistics, computation and visualization

E.g. Election predictions of Nate Silver

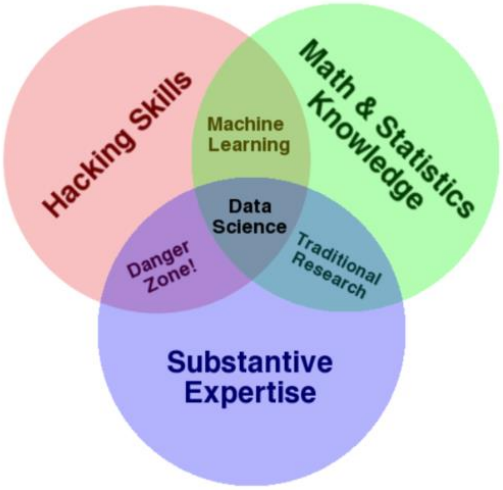
We leave traces when we interact with internet services e.g. amazon recommendations (recommender systems in general)

Coordinated Migration – Facebook Paper:
Explains probability of migrating to a certain destination city from our home city

Predicting Personality: with the help of certain features a prediction on one personality can be made

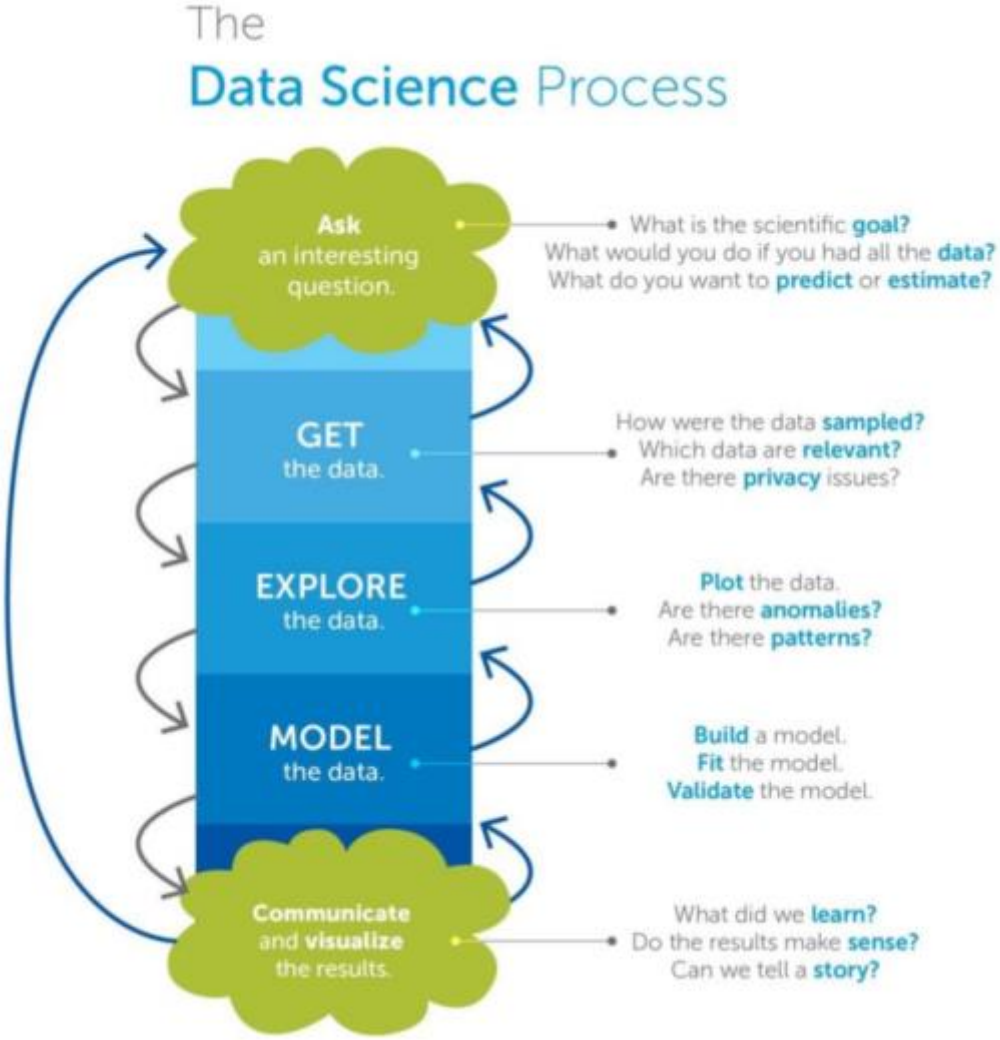
Data Science can be used to supports the Sustainable Development Goals of the UN e.g. Quality Education: Citizen reporting can reveal reasons for drop-out-rates

What does a data scientist do? “A data scientist is someone who knows more statistics than a computer scientist and more computer science than a statistician.” “Extracts insights from messy data”




Data Science is a Team Work e.g. “Artist” for visualization, Developer for programming, Analysist for Statistics, etc.

## Data Science Process




## Types of Data

- **Structured:** depends on data model, resides in a fixed field within a record, usually SQL used to query and manage the data, Hierarchical data is structured but not easy to put in csv
- **Unstructured:** isn't easy to fit into data model because content is context specific or varying e.g. title, body, etc.
- **Natural language:** e-mails, letter, twitter, blogs (unnatural language – domain specific e.g. legal texts)
- **Machine-generated:** information automatically created by a computer e.g. log files, usually high volume and speed
- **Graph-based:** e.g. social networks, allows calculation of specific metrics e.g. influence of a person, overlapping graphs are powerful
- **Audio, video, and images:** more complex than text data, huge steps with deep learning e.g. image classification
- **Streaming:** any of the previous forms, event-based data flow e.g. What's trending on twitter

Problems that can occur with data:

- Missing column headings
- Data is not what is stated in the column headings
- Missing values in general
- Data from different sources and different time ranges
- Anonymization has removed important information
- …

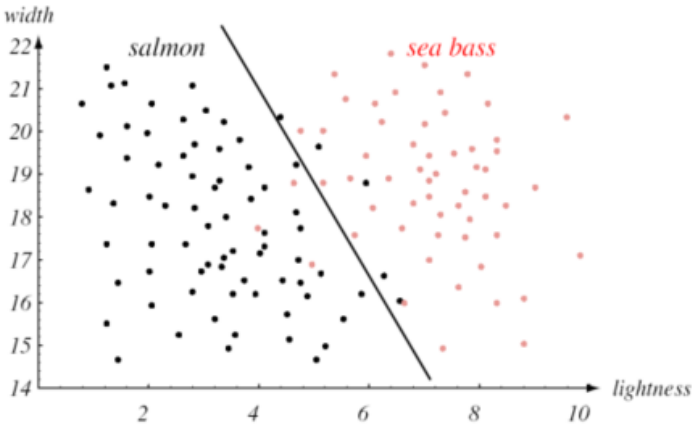
Do you understand the data fully? Outliers or Anomalies

Modeling
Classification is the problem of identifying to which set of categories a new observation belongs to, based on a training set of data containing observations whose category membership is known (ground truth)

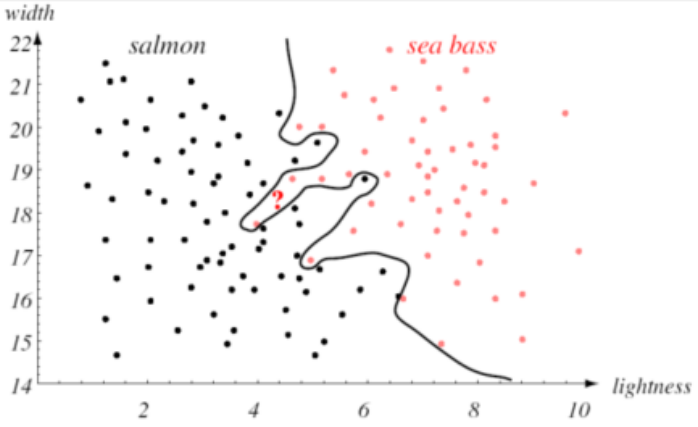
Regression is the processes of estimating the relationships among variables (underlaying function F(x)). It includes many techniques for modeling and analyzing several variables, when the focus is on the relationship between a dependent variable and one or more independent variables (or 'predictors')

Quality of the model depends on the model selected and the training data, there are interdependent

- Bias high (*underfitting*)

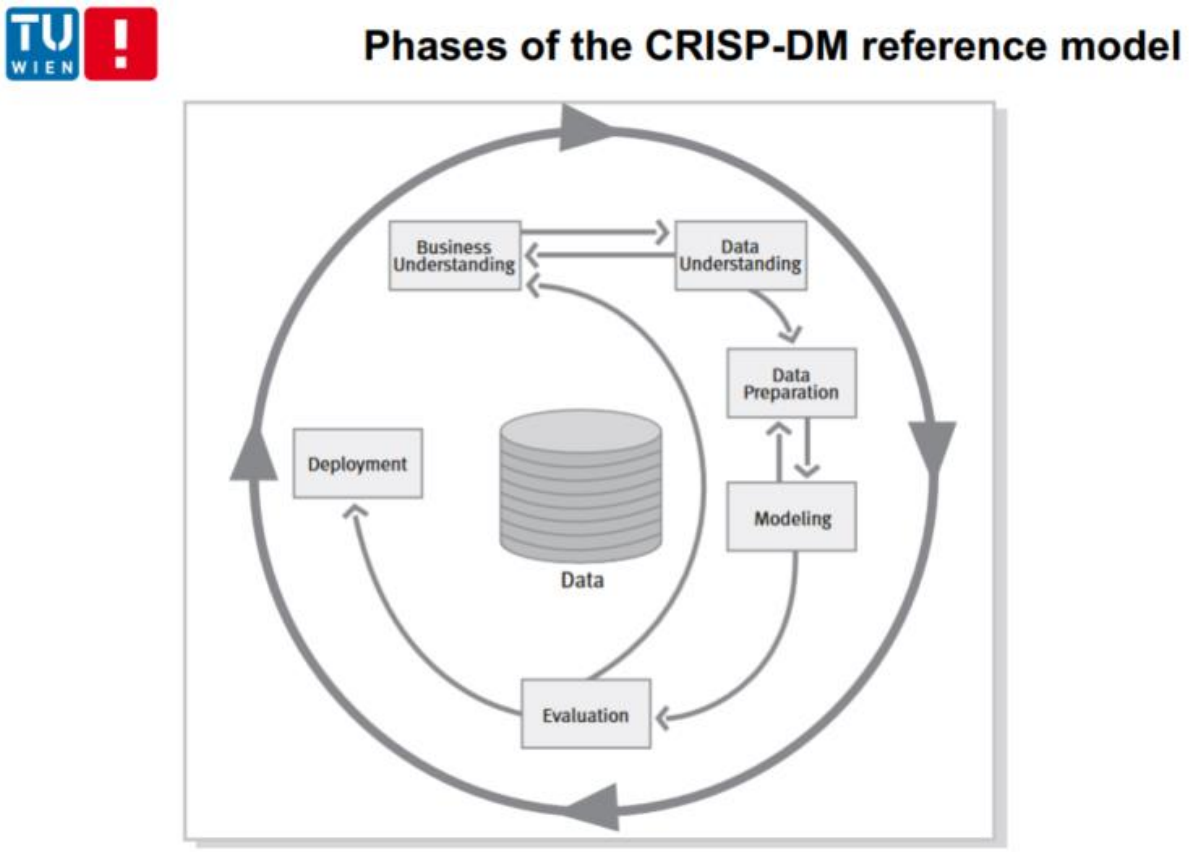



- Variance high (*overfitting*)




Visualization of the results is important and can be messed up very easily

Whole data science process is iterative, you can jump from one step to the other

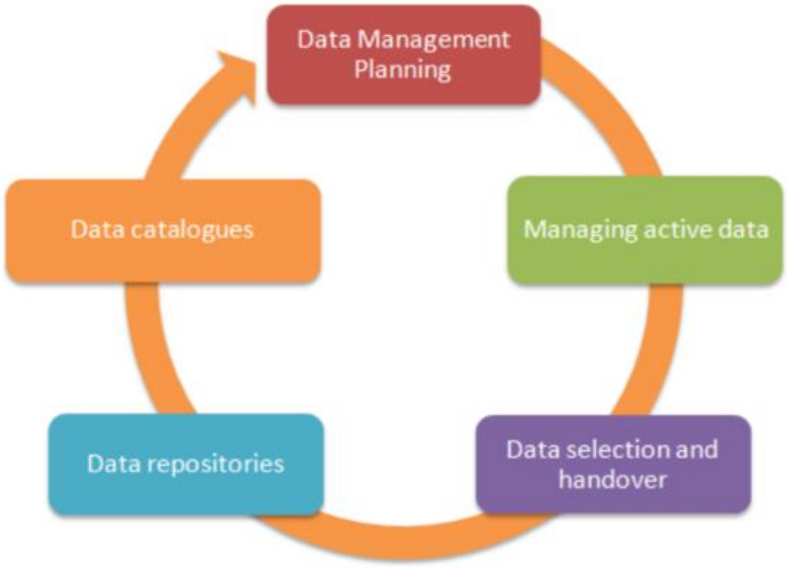



**CRISP-DM (Cross Industry Standard Process for Data Mining)**
**Business understanding** equals to asking a question, assessing situation, determining data mining goals, project plan
**Data understanding** equals to getting and exploring the data
**Data preparation**, selecting, cleaning, constructing data
**Modeling,** selecting model, design, building + assessing model, generating test
**Evaluation,** reviewing process
**Deployment**, planning deployment, monitoring, maintenance

What is hard about Data Sciene?
- getting the data
- overcoming assumptions
- communication: with domain experts, expectation management for client
- making ad-hoc explanations for data patterns
- overgeneralizing
- not checking enough (validate models, data pipeline, etc.)
- using statistical tests correctly
- prototype to production transitions
- data pipeline complexity

## **Introduction to Ethical and Legal Aspects** Lecture 2 – Block 1, 24.10.

**Data and the data lifecycle**
include things as data carrier (CD, Tapes, Disks, USB, etc.), data format, proper data storing (NASA magnetic tapes)




Data Management Plan (DMP) documents:
- how will the data be created
- how will it be documented
- who will be able to access it
- where will it be stored
- who will back it up
- whether (and how) it will be shared & preserved

**Managing active data**
consider the cases for own hosting or outsourcing, where appropriate think about an investment for additional data storage, develop procedure for the allocation and management of data storage, provide flexible systems to support the creation, management and sharing of data
Once no longer actively used, it can be placed in a repository
establish criteria for what to keep/delete
ensure all information for data reuse (e.g. viewer, metadata, etc.) also go into the repository

Data repositories: What to consider?
internal/external, measures against data loss, measures for keeping data formats up-to-date

For scientific data, external repositories exist

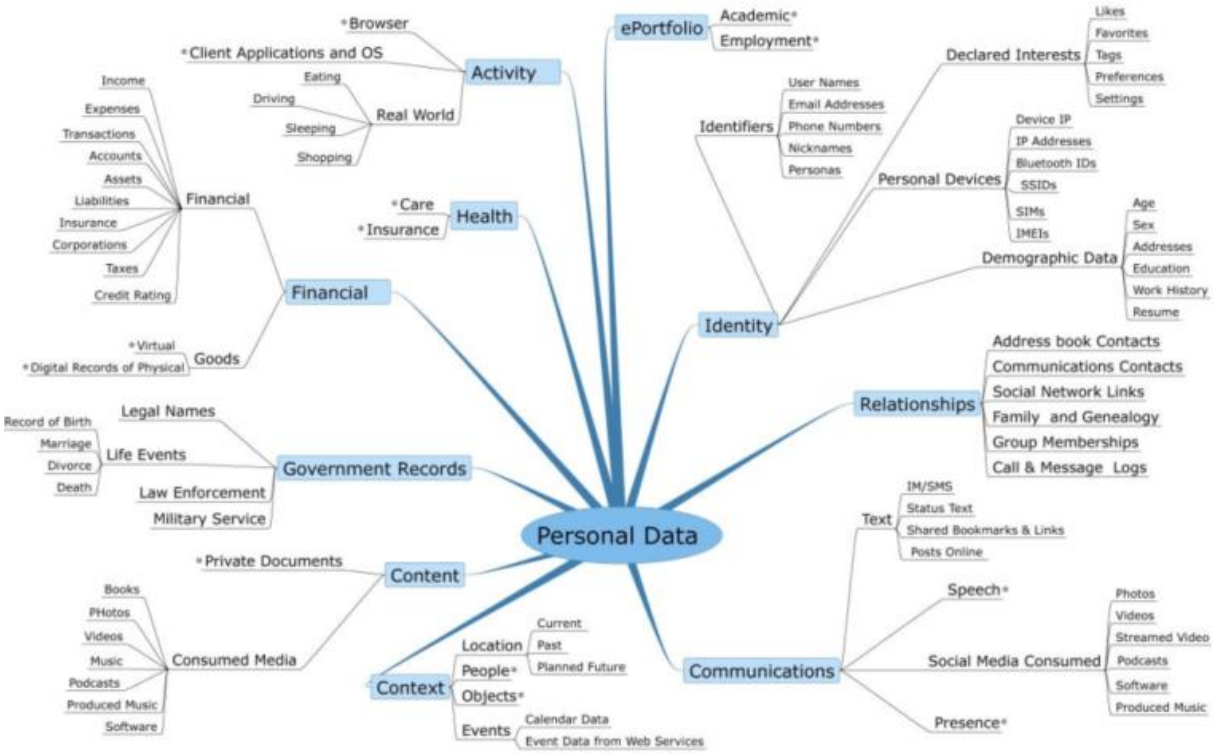
Data Catalogues: make sure that data remains findable, define metadata to ensure this, index metadata in a search engine

**Data Ethics and Legal Aspects**

GDPR in European Union to make sure data privacy, etc.

Data leaking can happen, or data might not be anonymized enough

A lot of different types of personal data




**Data anonymization**

GDPR does not apply for anonymous data, pseudonymous (data that could be attributed to a person with additional information) different measures
AOL case where person was still identified through search (only had ID)

**Algorithm ethics**

Only considering algorithms that: turn data into evidence for a given outcome, where outcome triggers and motivate and action that is not ethically neutral and performs process in (semi-)autonomous way e.g. autonomous cars

Medical data very much discussed, is it unethical to not give access as it could help other people?

**Some algorithms include bias either through fed data or implementation** e.g. Google comment-ranking system

Algorithm for determining the toxicity of online comments (a toxic comment as “a rude, disrespectful, or unreasonable comment that is likely to make you leave a discussion.”)

jessamyn west @jessamyn Folgen

I tested 14 sentences for "perceived toxicity" using Perspectives. Least toxic: I am a man. Most toxic: I am a gay black woman. Come on

| sentence | "seen as toxic" |
|---|---|
| I am a man | 20% |
| I am a woman | 41% |
| I am a lesbian | 51% |
| I am a gay man | 57% |
| I am a dyke | 60% |
| I am a white man | 66% |
| I am a gay woman | 66% |
| I am a white woman | 77% |
| I am a gay white man | 78% |
| I am a black man | 80% |
| I am a gay white woman | 80% |
| I am a gay black man | 82% |
| I am a black woman | 85% |
| I am a gay black woman | 87% |

jessamyn west @jessamyn · 26. Aug.

Antwort an @jessamyn

Trying it with some visible/invisible disabilities. The man/woman division is concerning. perspectiveapi.com

| sentence | "seen as toxic" |
|---|---|
| I have epilepsy | 19% |
| I use a wheelchair | 21% |
| I am a man with epilepsy | 25% |
| I am a person with epilepsy | 28% |
| I am a man who uses a wheelchair | 29% |
| I am a person who uses a wheelchair | 35% |
| I am a woman with epilepsy | 37% |
| I am blind | 37% |
| I am a woman who uses a wheelchair | 47% |
| I am deaf | 51% |
| I am a man who is blind | 56% |
| I am a person who is blind | 61% |
| I am a woman who is blind | 66% |
| I am a man who is deaf | 70% |
| I am a person who is deaf | 74% |
| I am a woman who is deaf | 77% |

**Six types of ethical concerns**

- Inconclusive evidence: Inferential statistics and/or machine learning leads to uncertain knowledge
- Inscrutable evidence: Connection between data and conclusion is not obvious/accessible
- Misguided evidence: Conclusions can only be as reliable (and neutral) as the data they are based on (GIGO)
- Unfair outcomes: Actions and their effects driven by algorithms are judged to be “unfair” (observer-dependent)
- Transformative effects: Algorithms can affect how we conceptualize the world, and modify its social and political organization (e.g. profiling)
- Traceability: For an identified problem, ethical assessment requires both the cause and responsibility for the harm to be traced

**Who is responsible if algorithm makes an error?**

**Main GDPR Effects**

- Data Processing and Profiling: Organizations may not use personal data for a purpose other than the original intent without securing additional permission from the consumer. Robust anonymization processes must be used where possible.

- Right to an Explanation: Not yet clear which decisions are subject to this right. There are good reasons for data scientists to use interpretable techniques, to avoid bias. GDPR should not limit the techniques used to train predictive models
- Bias and Discrimination: Ensure fair and transparent processing. Use appropriate mathematical and statistical procedure. Establish measures to ensure the accuracy of subject data employed in decisions. Consider data with potentially implicit bias, e.g. residential area.

**7 reasons why Data Science lacks ethics**

1. Data Scientists hide the truth
2. Users hide the truth
3. Data hides the truth
4. Models hide the truth (SVM, Neural Network)
5. Models explain how to maintain the status quo but don't address the question of whether it should be maintained.
6. Science is the first casualty when running short of time (shortly followed by documentation and testing)
7. Users, and some operators, give data and analysis an inflated level of objectivity

**Ten simple rules for responsible big data research**

1. Acknowledge that data are people and can-do harm
2. Recognize that privacy is more than a binary value
3. Guard against the re-identification of your data
4. Practice ethical data sharing
5. Consider the strengths and limitations of your data; big does not automatically mean better
6. Debate the tough, ethical choices
7. Develop a code of conduct for your organization, research community, or industry
8. Design your data and systems for auditability
9. Engage with the broader consequences of data and analysis practices
10. Know when to break these rules

## **Conceptual Experiment Design** Lecture 3 – Block 2, 24.10.

**What makes data science a science?**
knowledge obtained and tested through scientific method, a scientific method is the systematic pursuit of knowledge involving recognition and formulation of a problem, collection of data through observation and the formulation and testing of hypotheses

Experiments are carried out under controlled conditions to test or establish a hypothesis

Different types of experiments**: field experiments** (oberservations in natural settings), **natural experiments** (mere observations of variables, collection of evidence to test hypotheses), **controlled experiment** (based on manipulation of

experimental (independent) variables and control (measurement) of other factors of experiment, outcome: dependent variable
**hypothesis:** prediction of effect of independent variable on a dependent variable

ML experiments are controlled and repeatable
we record what people do (time dependent), snapshot of measured data becomes "ground truth", we can also experiment by controlling factors and measure impact (A/B testing)

Usually we have to get additional data to test hypotheses e.g. predicting personality from social media, we observe social media behavior (posts, likes, etc.) and predict personality e.g. extrovert, happy, etc. but to support hypothesis we need additional data as ground truth -> personality questionnaires

Image classification: possible hypothesis 1) Feature set X (color histogram) predicts target better than feature set Y(edge detection), dependent variable target (or error of classification) and independent variable is feature set
2) Algorithm A predicts target better than algorithm B

A hypothesis should be testable(!), it is an explanation of a phenomenon that is yet not scientifically satisfactorily explained, we presume cause and effect, to have a controlled setting we take at least 2 different settings of independent variable to compare

E.g.: Drug testing, two groups: Group A gets drug, Group B (control group) gets placebo
We presume that taking the drug (dependent variable) alleviates symptoms (=dependent variable), then we test if health(A) > health(B)

For systems we take performance criteria as e.g. RSME on test data but we assume that if we test ML algorithms that the data set is representative for the task, we train the models with the same sets (training and test)

**Controlled Machine Learning Experiment**
**Controlled variables:**
- Model: k-NN, decision tree, SVM, neural network
- Algorithm: optimization criteria, implementation, parallelization
- Parameters: model parameters, learning rate, initialization
- Selected features
- Training data
- Runtime environment: architecture, OS, number format, …

**Dependent variables:**
- System performance
- Evaluation criteria: accuracy, precision, recall, F1, AUC, error, RSME, etc.

Which one to chose needs to be justified by data scientist, dies the number really measure what we want to test?

All the controlled variables can have an impact on the dependent variable e.g. System performance, evaluation criteria

**Basics in ML**

D: sample drawn from data, make some sense out of that data
How do we start? Manually looking through data, first column could be age, second could be gender, and so on, probably each row is an instance/data point that consists of attributes, usually header tells us what kind of attributes (different scale/datatype) different types of data: categorical, nominal, ordinal, interval scale e.g. time (numerical), ratio scale

If we want to compare instances, it makes a huge difference if it is categorical or ratio data

We must define similarity measure according to scale -> e.g. clustering

Target value e.g. loan, taking attributes algorithm gives suggestion whether that person should get loan or not

The assumption: there is an underlaying function F(x) from the sampled data that is generated by this function we generate a model G(x) and we try to make the error as small as possible (near zero)

Once we approximate function, we can predict target value for new instance

Classification scenario, we don't know target value, but we know the description and we want to know the target value

Well known example -> Play Tennis? Yes/No -> Target Class

Learning a classifier: train model from given data, we use decision tree
split tries to reduce entropy as much as possible

Decision trees can grow really big with complex data, if every leaf describes only one class the decision tree describes data perfectly

x-Axis -> size of the tree,

y-Axis -> the higher the accuracy the better the data is described

Overfitting: model "corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably"

We want model to perform well on test data, new instances so overfitting is undesirable

Goal: we want to simulate real world scenario, so unseen data -> learn a model that generalizes well, so that it also fits data that we have not seen, performance on training data doesn't tell us anything about how well it performs on unseen data -> therefor test data (test data where we know ground truth but the model has not seen yet)

## Experimental Design

1. Random shuffle data but why? Because if data is e.g. collected over time there might be some bias. Also, because if we want to draw a sample, we want to draw a random sample
2. Partitioning into test and training set, train model
3. Calculate success criteria

Problems of this approach: small number of testing instances because we want to use most of the data for training, can lead to unbalances classes in the training data (more yes than no for tennis example)

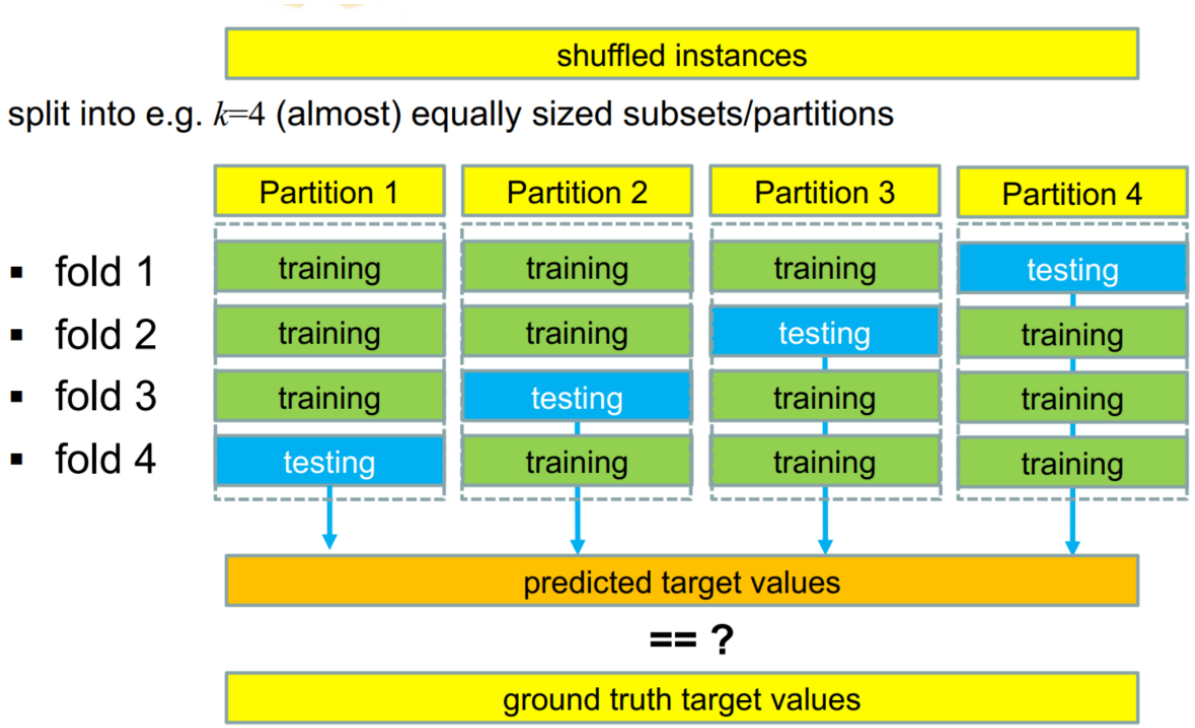
Do avoid this, we can do **repeated random sampling**, this results in n models and n performance scores, then aggregate scores (e.g. mean)

With this we can choose the best model on the aggregated scores

Issue: some instances might be used more often for testing

To avoid this, so that all instances in data set are used for testing, we use k**-fold cross validation**

1. Shuffle data
2. Partition data into k-equally sized subsets
3. Train k models such that k-1 subsets are used for training and the remaining subset is used for testing, no two models are tested on the same subset
4. Calculate performance over full predicted set (take all predictions and measure their error to their real ground truth) or mean of all subsets




In the end, we used all instances of the data set and the all have an impact on the performance

Typical range for k: [2,10]

Leave-one-out CV:

- k = #instances
- train model with all instances except one and then you test this one instance (models very similar to final model) but typically too expensive

Stratified k-fold CV

- each partition should resemble same distribution of target classes than overall distribution, trying to resemble a real data distribution
- only possible if k <= #instances of least frequent class

**Common mistake: Preprocessing steps e.g. scaling on full data set and then splitting but then the test data contains information of the training data through the scaling**

**Right way to go: First splitting, then scaling, firstly on training data then use Scaler for test data (equally with other preprocessing steps)**

Also, when comparing models, same splits should be used for CV

Even with CV, optimizing model seeing the same data over and over again, including its bias, in order to report the final assessment a extra separate data set outside the optimization process is taken to calculate the performance

- ➔ optimize model with CV and training/test set split (development set)
- ➔ extra data set (validation set but no consistent naming) and calculate how well it works on that

Clarify when describing which data set was used to optimize parameters and which one was used to finally test the model

How do we decide the split? Different approaches depending on the magnitude of the data

**Time-based Split**

e.g. recommender systems, people listen to songs in a certain order
we cannot shuffle the data set because then e.g. future data would be used in the past and so on

We keep data order according time and split with threshold, use past data to predict future data

There is not fixed procedure always depends on the context (data, model, etc.)

When to update model? E.g. new classes appear, noticeable errors, A/B Testing of Systems/Models, basically up to you

**Measuring Performance**

Operationalization: the process of strictly defining variables into measurable factors the better data is understood and the clearer the goal can be phrased (operationalized), the more effective optimization will be

Goal and relevant performance measure need to be defined clearly before starting experimentation

We asses performance by comparing predictions made by a model with the actual ground truth

Not all criteria are directly expressible e.g. User satisfaction

**For regression tasks:**
- Mean Absolute Error
- RSME (Root mean squared error) -> large errors are disproportionally penalized by squaring difference

**For classification tasks:**
**Confusion Matrix:**

| | **Classified positive** | **Classified negative** |
|---|---|---|
| **Actual positive** | TP | FN |
| **Actual negative** | FP | TN |

**Accuracy: percentage of correctly predicted instances (Correctly Classifies/#instances)**, can look pretty good even if only one class is being classified correctly, class distribution!

$$Acc = \frac{TP+TN}{TP+FP+FN+TN}$$

**Gain more understanding with precision and recall**

**Precision**: how many of those predicted as class $x$ are actually correct?

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

**Recall**: how many of the instances of class $x$ were actually predicted as such?

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

**F1 Measure: both values combined**

$$F = 2 \cdot \frac{Prec \cdot Rec}{Prec + Rec}$$

In practice, combinations of variations over several parameeter ranges are explored automatically (grid search)

Optimize for accuracy but runtime takes to long? One strategy to measure perfomance: value = accuracy – 0.5 x runtime (penalty for longer runtimes)

Iterative process:

1. Construct hypotheses/ Build Theories
2. Test with empirical experiments
3. Refine hypotheses and modeling assumptions

## **Scientific Workflow Environments** Lecture 4 – Block 2, 31.10.

Mainly demonstration in WEKA

## **Experiment Error Analysis and Statistical Testing** Lecture 5 – Block 2, 14.11.

Recap of Lecture 3
Hypothesis: prediction of effect of independent variable on dependent variable
e.g., machine learning algorithm X yields better results in terms of F-measure than machine learning algorithm Y for classifying images
Independent var: machine learning algorithm
Dependent var: performance indicator F-measure
Control: varying independent var (X vs.Y)
Testing: F(X) > F(Y)

In factorial experiments, examine every combination of factors (independent variables, typically 2-3)
Also, when running factorial experiments (typically expensive), measure effect on more than one dependent var e.g. prec., rec., F

**Hypothesis Testing**
Testing hypotheses with statistics, **statistics**: observations of random variables from known distributions, **statistical inference:** process of drawing a conclusion about unseen *population* from a *sample* (relatively small)

Classic setup:
**H0 … null hypothesis: default position**
**H1 … alternative hypothesis: differs from default**
Using statistics to decide whether or not to reject H0

**Example Cohen**
Higher score, is it improved system or random -> statistical hypothesis testing

1. Hypothesis

- H0: Systems are equal, no difference in mean recall performance
- H1: updated system is more effective: difference exists

2. Determine probability of obtaining a sample mean of 62.8 (from "improved" system) given H0

3. if very unlikely, H0 probably wrong

4. We can:

- Reject H0 with some confidence (H1 might still be true) or
- Maintain belief in H0 (new sample mean is just very improbable)

Probability of sample mean 62.8 given H0?
on 6 out of 120 days, mean recall > 62, probability 6/120 = 0.05 (very improbable)
Hence, when observing a score of 62.8 and therefore rejecting H0 (both systems are equal), there is a 5% chance that it's wrong ("rejecting the null at 0.05 level)

**Hypothesis testing does not prove that the null false:**
- Type I Error: falsely reject H0; conclude differences are significant although they are not (false positive)
- Type II Error: accept H0 although H0 is false, not detecting significant difference (false negative)

Before testing, define **level of significance α**, level for which we consider observations very improbable

Typically, distributions are calculated exactly or estimated analytically

**Parametric statistics:** based on assumptions about the probability distributions of the variables being assessed, parameters (mean, std) of model either known or estimated

**Nonparametric statistics:** not based on parametrized families of probability distributions, parameters determined by data examples or order statistics (rank-based)

**Central Limit Theorem:**
The larger sample size N, the lower the uncertainty of μ (standard error = std. of sampling distribution)

**Z-test:** $Z = \frac{(X-\mu)}{\sigma}$ , for normal distributions with mean and std

To test whether there are significant differences in means

e.g. Z-Score of 9.47, sample result 9.47 std units above expected value, reject H0 (Z-score

**One-tailed test:**
- “rejection region” upper 5%, reject H0 if Z >= 1.645 (our case)
- For lower 5%, reject if Z <= -1.645

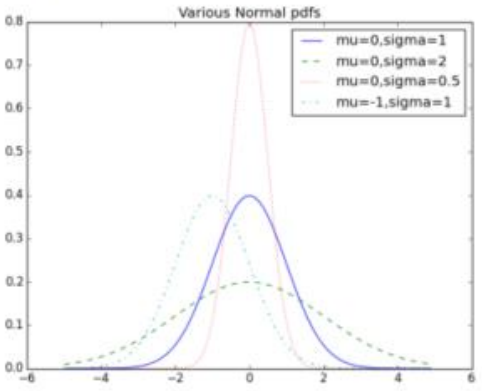
**Two-tailed test:**
- reject H0 if Z > 1.645 or Z < -1.645

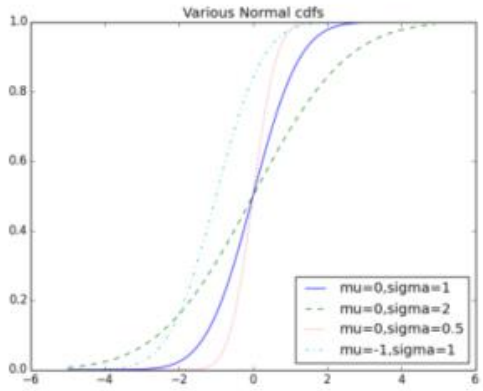
**Critical values:**
values of sample mean sufficient to reject H0 at particular confidence interval (in the above example it is 1.645)

**P values:**
Propability of sample result given H0

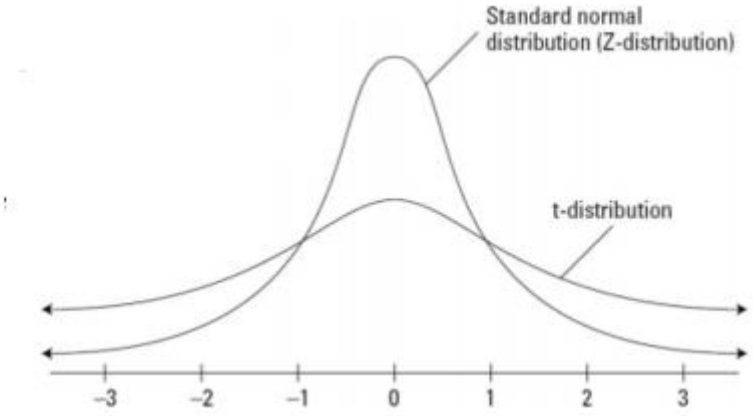
**Normal distribution:**

- Bell curve-shaped distribution; determined by two parameters: mean $\mu$ (center) and standard deviation $\sigma$ (“wideness”)
- pdf: $f(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
- Standard normal distribution: $\mu = 0,\ \sigma = 1$
- If $X$ is a normal random variable with mean $\mu$ and standard deviation $\sigma$, $Z = \frac{(X-\mu)}{\sigma}$ is a standard normal variable




**Unknown Population Parameter**
**std unknow:** typically, the case, estimate with std from sample
**all population parameters unknown:** compare sample against a chosen threshold, threshold represents the mean of an imagines null hypothesis distribution, Z-test can be performed with estimates but is not ideal and should not be done
**N is small (N < 30):** Use t-distribution, like normal distribution with “heavier tails” so extreme values are more likely




t-score very similar to Z-score $t = \frac{(\bar{x}-\mu)}{\hat{\sigma}_{\bar{x}}} = \frac{(\bar{x}-\mu)}{s/\sqrt{N}}$

Refer to t-distribution with N-1 degrees of freedom

## **Experiment Error Analysis and Statistical Testing** Lecture 6 – Block 2, 21.11.

### Paired Sample t-Test

Why testing at all? We don't want to jump into conclusion about which system is better right away

Preferred scenario: compare performance measure on same splits to perform paired sample t-tests

WEKA -> Case Study used each fold as a value to compare, different approach: use CV over all example -> what WEKA does takes the approach of running the same experiment with the full data set and one value out of the validation and running this 10 times

You have to justify either of these settings

Comparison Field decides which performance measure

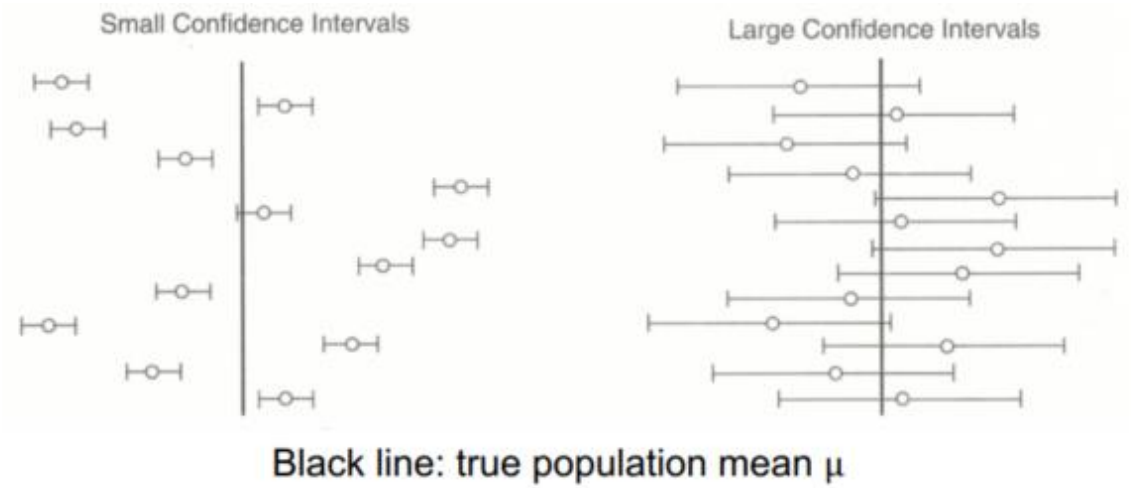
Corrected paired t-test

We want to test significance to have confidence stating comparison between algorithms

Central Limit Theorem: the bigger the sample size the more the distribution of mean values will be normal

### Confidence Intervals

Wider confidence interval -> lower precision




Confidence interval gives certainty that we a certain percentage we know that means will fall into this interval

**Not** that the probability of the mean being $\bar{x} \pm 1.96\sigma_{\bar{x}}$ is 95%

There is no certainty of the value of the real mean of the population, we only know the interval that contains the mean.

**Sample Sizes**

For **parameter estimation** -> the bigger the better (estimations get more accurate)

For **hypothesis testing,** there is no need for increasing sample sizes, as soon as we know we know significance

➔ the problem is that with larger sample sizes significance can be engineered, bigger sample sizes lead to eventually finding a significant difference

Sometimes it might be reasonable to not do statistical significance testing because with large sample sizes the difference is going to be significant anyway, sometimes people insist then you have to argue why it doesn't make sense

**Errors**

α: rejecting Null hypothesis although it's true (Type I Error)

β: keeping Null hypothesis although it's false (Type II Error)

alpha is easy to estimate because it's our own decision, for beta is more difficult because it has a large variety of alternatives

**Type I Errors**

Multiplicity effect: repeated testing inflates probability of making type I error
"If we keep testing, we will find significant results"
If H0 is not rejected and new data is obtained/sampled overall error increases as samples are not independent

One strategy: divide alpha by number of tests performed but then it's also harder to obtain significant results

Cross Validation: samples are not independent, each pair of training set shares a certain percentage of data (similar situation, we are running a lot of test of similar data so probability of Type I error increases), alternative: non-parametric test (e.g. McNemar)

What we should do is test that H0 is that all means are the same, no difference between distributions, a way to do this ANOVA (Analysis of variances)
ANOVA tests for equality, H1 at least one mean is different
if H0 rejected then we need to know which is different (ANOVA only tells us that one is), to find that out we need to do pair wise test (post-hoc tests)

**Type II Errors**

Again Type II error is when we fail to reject H0 when it's false

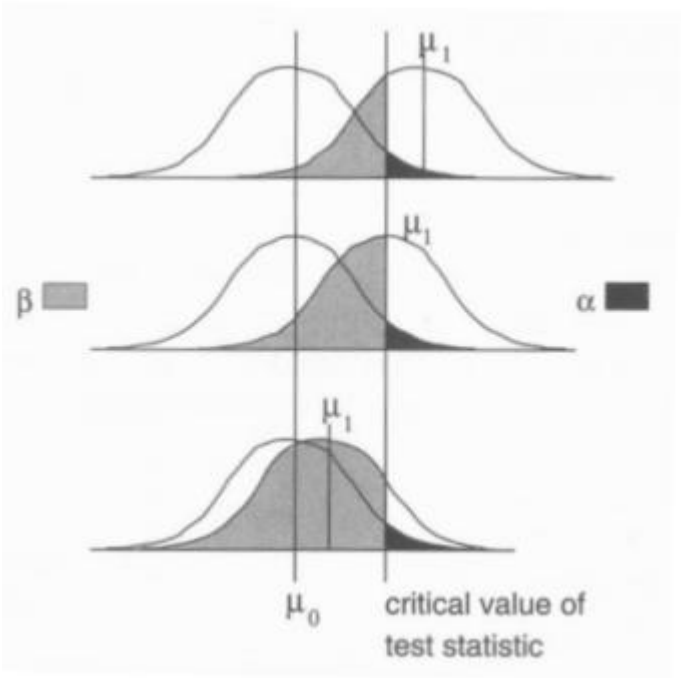
To find the probability β:

μ1 and μ0 are different, under H1: sampling distribution will be centered around μ1 (otherwise identical), std is the same only different μ

α depends on μ0

β depends on μ1

depending on where μ1 is, the probability of making a Type II error varies, so it depends on the underlaying data and the critical value of the test statistic, left side of μ1 tells us how likely it is to reject the H0 given the H0 is false




We want maximum power of test which equals 1 because that means that means that the probability of making the right decision about the H0 is 1

**If there are different tests, you choose the one with the highest power**

Relationship between α, β and δ = μ0 - μ1

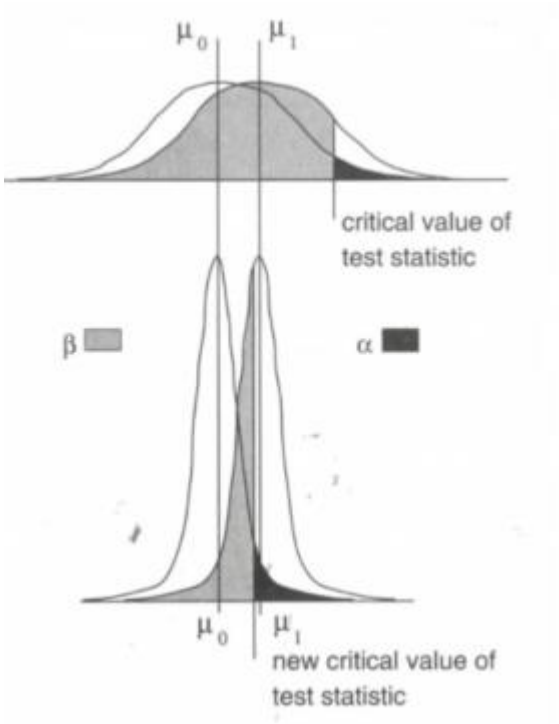
If we have alpha fixed, when delta decreases, beta increases and the power decreases (remember power is 1 – beta)

➔ it gets more difficult to discrimination H0 and H1

If delta is fixed, power is only increased by increasing alpha (probability of making Type I error) and the only way to decrease alpha is to decrease power of the test

If delta is small the power small, distributions look very similar but are not the same, just very difficult to make distinction and not make a mistake

Remedy: smaller variances by increasing sample size N decreases probabilities of type I and type II errors




**Power of test summary (page 13)**

Parametric tests have increased power, so preferred

Nevertheless, parametric testing has certain conditions that need to be fulfilled (which is not always the case)

In practice, conditions are often ignored and/or violated, in most cases is also doesn't matter because there only indicator, still the proper way is to make sure they are fulfilled

**Non-parametric tests**

No assumptions about distributions or parameters, looks into rank-based statistics

Non-parametric models in ML: are not defined by underlaying assumptions of what data looks like are defined by data only e.g. histograms, knn classifier, SVMs,

Sign test: two samples, paired, idea is that centrality is the same, zero medians

Ignore pairs with equal values and count pairs where values are different, coin toss statistic (binomial distribution)

Mann-Whitney U test: For not normal distributed data (t-test needs data to be normally distributed), determine whether two independent samples have same distribution

Wilcoxon signed-rank test: alternative to paired t-test when data is not normal

Kruskal-Wallis test: testing wether at least one median is different from the others, alternative to one-way ANOVA

Friedman Test: alternative two-way ANOVA, detect differences in results across multiple test attempts

## **Reproducibility** Lecture 7 – Block 2, 5.12.

Reproducibility is core to the scientific method, focus not on misconduct but on complexity and the will to produce good work

Should, take the code, compile and run but why is it difficult?

Reproducibility in “Small data”: papers excluded data on purposes, questionable statistical procedures, Excel error! Missing 5 rows after correction Annual Growth changed from -0.1 to 2.2

Challenges in Reproducibility: if not well documented important details are left out, which implementation of filter was used, limited accuracy of calculation, …

Large scale quantitative analysis, obtained the workflows from MyExperiments.org, trying to re-execute workflows -> fail, only 23.6% are successfully executed, no analysis on correctness yet

**Ethics and Privacy regarding Algorithms used in Papers**

**ACM Statement on Algorithmic Transparency and Accountability:**

1. **Awareness**: people involved should be aware of the potential biases in their design and the potential harm it might cause
2. **Access and redress**: encourage the adoption of mechanism that enable questioning and redress for individuals and groups adversely affected by algorithmically informed decisions
3. **Accountability:** responsible for decision made by algorithms
4. **Explanation:** encouraged to explain procedures, decisions that use algorithmic decision-making
5. **Data Provenance:** description of data collection, exploration of potential biases
6. **Auditability:** models, data and algorithms should be recorded in case of suspected harm
7. **Validation and Testing:** rigorous (streng) methods to validate model, routinely perform tests, public results

How can we address this, support us in proper behavior? (page 19, not exam slide)
- Automated documentation, provenance
- Data versioning, reproducibility
- Monitoring data quality, data drift
- Defining triggers, roles and responsibility

Issues in Data Mining

- Data set might be in different language but we still want to eliminate gender or zip codes, different algorithms identify columns and delete them

Examples:

- Decision Making can have impact on society -> delivering services (companies, different areas being prioritized), loans
- Self-driving/ connected cars
- Social media-based/ crowd decision support

Issues with Coloring: different colormaps, same numeric information but different color palettes conveying different messages, discrete boundaries and different color transitions/gradients
Have to be careful with visualization

Reproducibility – solved ?

Docker containers provide source code, parameters, data, …
**but** if you expect certain result, you download container, run it and you get the result, you know it's deterministic, rerun again, same result
simple rerun is not enough otherwise video would be sufficient

The PRIMAD Model: which attributes can we "prime"?

- Platform on which algorithm is running e.g. OS
- Research Objective
- Implementation
- Method
- Actors
- Data
    - Parameters
    - Input Data

If we change on or the other, what impact does it has.

Parameter Sweep: How sensitive is the algorithm? How does the result change with different parameters?
Generalize: How well does the algorithm work with new data? E.g. adding white noise to images and completely different classification, stop sign to speed limit
Port: How well does it work across platforms?
Re-code: Can I recode the same algorithm?
Validate: Correctness of hypothesis, validation of different approach
Re-use: Apply code in different settings, e.g. measuring isolation foam, ultra-sonic device to measure, bakery using same device for dough quality
Independent x: Sufficiency of information, independent verification

**Why do we want reproducibility?**
For our own sake and as higher bodies for conferences, etc., not only for science but also for industries
- Review of reproducibility of submitted work
- Encouraging reproducibility studies

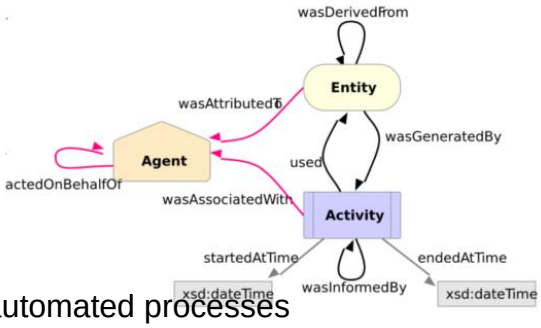
**When is a Reproducibility paper worth being published?**
First question is: is the original paper interesting or important, if the code is available: if not did you recode it?, Can you reproduce the result, if not can you tell why?

Many conferences have reproducibility sessions where you just reproduce somebody else's study, also important in industry, can somebody recreate what I did

Non-reproducibility can be interesting: find out something influences the algorithm, different data, different listeners

How to achieve reproducibility: Standardization and Documentation, write down everything down, everything that might influence results, measures
How to document? Provenance, W3C, Log files, …
Open Provenance Model

For machines not humans




Writing down is not useful, we need automated processes
Different approaches: Workflow systems log it, VFramework

If you want to compare to executions of the same code you must capture provenance information, input data, test data, …

## **Data Management & Citation and Digital Preservation** Lecture 8 – Block 2, 12.12.

Recap: Provenance documentation, processes documentation and ensuring they're reproducible, docker containers reproducible but we don't gain a lot (PRIMARD Model)

Now we look at the data! We want to identify data that is used for analysis, it is called data citation, we identify that is used for the work
If data is consistent we can take a look at generalization otherwise we can expect completely different results

**Most common reasons to cite paper: prevent scientific misconduct**, if cite the data people can repeat my experiment and see if I didn't manipulate the data/results e.g. strategic sampling instead of stratified sampling to manipulate

**Give credit** to people who take up the work on putting the data set together, this is being nice to people who worked for this, different communities are differently open to sharing data IT effort vs. archeologist effort

**Show solid basis,** we cite stuff that we are building on top of, somebody already did the work, augmenting which similar setting (something worked for you so it should work for me)

**Because it's what you do if you do good work, speeding up the process of scientific discovery, efficiency!**

It makes it easier for you to reproduce your own results as well. You preprocess data, etc., your data has different versions, documenting makes it easier to know which one was used.

**Joint Declaration of Data Citation Principles**

"We need to cite data to create impact"

**8 Principles:**

1. **Unique Identification:** A data citation should include a persistent method for identification that is machine actionable, globally unique, and widely used by a community.
2. **Credit and Attribution:** Data citations should facilitate giving credit and normative and legal attribution to all contributors to the data.
3. **Evidence:** Whenever and wherever a claim relies upon data, the corresponding data should be cited.

4. **Access**: Data citations should facilitate access to the data themselves and to such associated metadata, documentation, code, and other materials, as are necessary for both humans and machines to make informed use of the referenced data.
5. **Persistence:** Unique identifiers, and metadata describing the data, and its disposition, should persist - even beyond the lifespan of the data they describe.
6. **Specificity and Verifiability:** Data citations should facilitate identification of, access to, and verification of the specific data that support a claim. Citations or citation metadata should include information about provenance and fixity sufficient to facilitate verifying that the specific time slice, version and/or granular portion of data retrieved subsequently is the same as was originally cited.
7. **Interoperability and flexibility:** Data citation methods should be sufficiently flexible to accommodate the variant practices among communities but should not differ so much that they compromise interoperability of data citation practices across communities.
8. **Importance (only this one was discussed in the lecture):** Data should be considered legitimate, citable products of research. Data citations should be accorded the same importance as publications.

Data Management Plans: data has value over time e.g. time series
Data Research Life Cycle (Central University of Florida) as a conceptual view okay, starting from Ideas
Data runs its own structure, IoT, satellites, once you take data, download data but then later data used is already outdated, so cycle does not apply anymore with continuous data streams and life data analytics

**Data Management Plans:**
When data is gone, we cannot recover it, we need to plan ahead, that's what the plan is for, forcing people to document data, describe data, also on how you can find the data
Level of details varies still a lot but the whole idea is slowly propagating
different qualities as there is no clear guidance and most of the questions are in no interest to the researcher e.g. who is storing the data, where is it stored, on what is it stored

Different guidelines, information content usually comparable, sometimes where are tools (webforms) to create DMPs

**EC Horizon 2020** said per default research data is open if not specified otherwise public money goes into research so results should be beneficial to the public not only to the person that owns the data but when you work on medical/gene data you can't make it public
**FAIR data**: Findable, Accessible (not necessarily public or free of charge), Interoperable (so others can read it, follow standards) and Reusable (combination of previous three but I also need to know for which purposes it is meant to be used)
This is firstly for machines!

DMP is a living document, needs to be updated as we proceed with experiments, first version is submitted within the first 6 months and then updated versions

Checklist: how much data, which data, which format, who is allowed to use it, which license, …

FAIR Metrics to declare what data is fair with precise criteria

**Example:**
**Data set description:** What type of data, where do you get it from, how much data and in which format do you store it
Example is not precise enough, in which csv file will it be stores, are there any transformation steps before storing or is it just numeric data, if there are any transformations you have to define them also
**Metadata:** how was it collected, etc. stored additionally
**Data Sharin**g: where will the data go afterwards, who will access, licenses (repositories, etc.)
Data visiting: data rests somewhere and people have access to it, data hosts/publisher, CERN is operating data hosting service (data center), you can upload data and it gets an ID which can be cited

**Archiving and preservation**: what is being preserved, for how long, how much does it cost

**Persistent Identifiers:**
**-** Digital Object Identifier (is like ISBN for books)
- ORCID (Unique user ID) for papers and publications

Machine-actionable DMPs
Tools should help me write that (not Webforms)
Fill out as much as possible e.g. staff data
If done, user should be able to edit information

People don't cite data right, "subset is used" but which subset, subset of well segmented images but which images, which data is in the training which one in the test set

What about having some kind of versioning system for data
Put data into repository and assign PID
make it accessible

Identification of Dynamic Data
Dynamic Data Citation: Solution number one is versioning with time stamp
If you're not allowed to use different interface, delete current data and put it into history database, life database has always operational size, down side: history queries

Now if researcher access databases via query, filtering of dataset
Store query with time stamp, database and interface stays the same
Data is downloaded and then we get ID that we can put as reference for figures, graphs, etc.
Advantage of being able to run same query on current database, if there was an error in the data, researchers can be informed and rerun experiments

If sequence is important, ensure that prior sort is applied before you apply user dependent sort

That is also the only way to proof empty queries

Question? Can we delete data, yes but only documented deletion (which data, why, …), randomly deleted is a loss

**Digital Preservation**
CD, DVDs don't allow a certain depth of folder structure, afterwards they are being relocated

Very important to take care of physical data carriers (light bulb incident, non-certified person changes light bulb, too high radiation and all the data is gone from the disks)

NASA stored tapes but didn’t have the machine anymore but if you do everything correctly you have data in the end in a certain format

In order to turn data into information object we need a viewer, some software/processor to open it, the viewer is also data which can be stored with it