# LABELLING

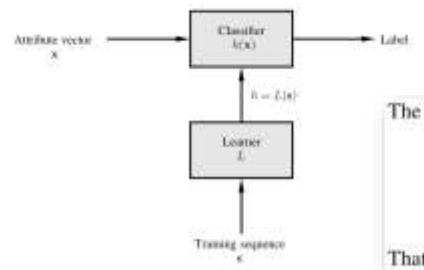
Interaction with Machine Learning

Cambridge MPhil ACS + CST Part II / III 2021-2022

# What's the big deal?

Supervised learning: a quick reminder

We don't want to design $h$ explicitly.




So we use a *learner* $L$ to infer it on the basis of a sequence $\mathbf{s}$ of *training examples*.

The *training sequence* $\mathbf{s}$ is a sequence of $m$ *labelled examples*.

$$\mathbf{s} = \begin{pmatrix} (\mathbf{x}_1, y_1) \\ (\mathbf{x}_2, y_2) \\ \vdots \\ (\mathbf{x}_m, y_m) \end{pmatrix}$$

That is, examples of attribute vectors $\mathbf{x}$ with their correct label attached.

## The human-centric approach to labelling

- Explicitly acknowledges human work involved in building and deploying ML systems
- A central role is for humans to specify behaviour through training labels
- Are labels an objective mathematical truth?
- *End-user activity of labelling is particularly interesting*

The *human-centric* approach to machine learning explicitly acknowledges the human work involved in building and deploying machine learning systems. A central role for humans is to specify the desired behaviour of the system through the provision of training data with labels. When viewed through the lens of traditional statistical philosophy, these labels are intended to capture an objective mathematical property of the data. However, when faced with the irregular, noisy, and subjective application domains of human-centric systems, this assumption unfortunately produces numerous challenges which can result in both a poor user experience as well as poorer resultant models.

These challenges can be effectively addressed by addressing the interaction design of the end-user activity of *labelling.* This is because not only is labelling the primary mechanism for non-expert interaction with machine learning, but also because it is where the end-user most clearly encounters the tension between the statistical ideals of supervised learning and human-centricity.

Interactive machine learning (IML) systems enable users to train, customise, and apply machine learning models in a variety of domains. The end-users of these systems are typically non-experts with no knowledge of machine learning or programming. In contrast, the professional practice of machine learning, engineering or ‘data science’ typically requires expertise in both those areas. The key design strategy for reducing the expertise requirements of applied IML systems is to abstract away using automation nearly all technical aspects of training and applying models, *except* the provision of training data.

# Crayons

Fails, J. A., & Olsen, D. R. (2003). Interactive machine learning. *Proceedings of the 8th International Conference on Intelligent User Interfaces - IUI'03*, 39. https://doi.org/10.1145/604050.604056

**Figure 5** – Crayons interaction process

In the*Crayons*application (Fails&Olsen, 2003), userscan train a model to segment images into different parts. Crayons enables end-usersto build image segmentation classifiers, that is, pixel-level binary classifiers whichsegment portions of an image as falling into one of two classes. For example, a 'hand detector' classifier would take a 2D image of size$w \times h$as input, and as output, produce$w \cdot h$binary labels, one for each pixel, corresponding to whether or not the pixel is partof a hand in the image. To build such a classifier in Crayons, users paint labels onan image as they would using a brush tool in a graphics application such as MicrosoftPaint or Adobe Photoshop, being able to toggle between two 'brushes' for the twoclasses. As the user paints, a model is trained, and the output of the model is renderedonto the same image, through a translucent overlay. This allows the user to focus further annotation on misclassified areas.


Figure 1. The EluciDebug prototype. (A) List of folders. (B) List of messages in the selected folder. (C) The selected message. (D) Explanation of the selected message's predicted folder. (E) Overview of which messages contain the selected word. (F) Complete list of words the learning system uses to make predictions.

# Elucidebug

Kulesza, T., Burnett, M., Wong, W.,&Stumpf, S. (2015). Principles of Explanatory Debugging to Personalize Interactive Machine Learning. In*Proceedings of the 20th International Conference on Intelligent User Interfaces - IUI'15*(pp. 126–137). https://doi.org/10.1145/2678025.2701399

Another example of an end-user controlled IML system is *EluciDebug* (Kulesza,Burnett, Wong,&Stumpf, 2015). EluciDebug allows end-users to build multi-class classifiers for organising short to medium-length pieces of text, such as email. The user performs manual annotation by moving emails to folders, where each folder represents a class. As the user organises their email, a model is trained, and the output of the model is presented as suggestions for classification within the email client itself, whichthe user may accept or overrule. The key thing to note is that both systems involve a training loop, where the user provides annotations either in the form of trainingexamples or potentially by manually adjusting model parameters (as can be done inEluciDebug). Next, a model is trained and the model output is somehow presented backto the user for further action in such a way as to directly suggest which furtherannotation or adjustment actions would be useful.

- Users interact with IML systems by providing labelled training instances that exemplify how the system ought to behave
- In labelling data in this way, users are forced to abide by statistical assumptions of supervised machine learning models that have been implicitly embedded in IML systems.

## Labelling *could* be viewed as programming or model construction…

- Model construction:
  - *Fitting models to data*
  - *Uncovering ‘natural law’ (Breiman, L. (2001). Statistical Modeling: The Two Cultures.Statistical Science, 16(3), 199–215.)*
  - *A ‘techno-pragmatist’ view*

These examples of interacting with a system in order to control its future behaviour can be considered either as programing, or as model construction. The programming perspective suggests that the user wants the system to behave in a certain way, and is training it to do so. The model construction perspective suggests that the system is trying to discover what the user wants, and is building a model of the user’s intentions based on observations of the user’s behaviour. These two perspectives carry very different philosophical assumptions.

Let’s start with the model construction view:

The practice of fitting models to data has its roots in the statistical philosophy that there exists some natural law underlying observed data (Breiman, 2001). Due to imperfections in the data collection process, the observed data is subject to noise. The objective of data modelling, then, is to uncover the parameters of the underlying law. This philosophy has influenced the design of supervised learning algorithms, and in turn, the assumptions of supervised learning have, by default, driven the design of IML systems. This design influence may be termed ‘techno-pragmatism’, where the interaction is designed around satisfying the technical needs of statistical models. The purpose of the user, within the overall system design, is to satisfy the requirement for an ‘objective’ function, encoding the underlying ‘law’, in which the labels provided by the user define the ‘ground truth’ of that law. The techno-pragmatist statistical view of IML is therefore fundamentally concerned with notions of truth, law and objectivity.

# The model construction approach is limiting

- IML is often inherently subjective
- Consider machine translation, music reharmonsiation, artistic style transfer

In contrast to the techno-pragmatist view, in which the user is regarded as a source of objective ground truth for a statistical inference algorithm, we argue that the function of an intelligent machine learning system is to be subjective, or more precisely, to replay versions of subjective behaviour that has previously been captured from humans. This type of “intelligence” can be distinguished from mere objective automation, of the kind exhibited by a heating thermostat or adaptive suspension, where behaviour is determined by direct measurement and physical laws. Those objective systems do not require labelling (or at least, the labels are implicit in the design of the sensing channels). Examples of subjective judgements include giving names to things, composing texts, making valuations, or expressing desires – all related to human needs and interpretations. None would be meaningful in the absence of any human to interpret the result, meaning that they are inherently subjective.

In many cases, a machine learning system is therefore expected to emulate subjective human judgments, and it does this by replicating judgments that humans have been seen to make. Here are some extreme examples: machine translation systems are trained using texts that have been written by humans; music harmonisation systems are trained using music that has been written by humans; and artistic style generators are trained using pictures painted by humans. In a sense, these “intelligent” algorithms offer a kind of mechanised plagiarism, in which the statistical algorithm simply mashes up and disguises the original works until it is impossible to sort out who the rightful authors were.

These kinds of creative “intelligence” offer an extreme case of machine behaviour that is derived from subjective human decisions, but almost all supervised learning systems demonstrate similar dependencies. Data is acquired by observing humans (whether researchers, volunteers, anonymous Mechanical Turkers or Google searchers) making decisions and expressing themselves. The actions of those humans are then replayed by the system as appropriate, based on statistical likelihood that a human would dothe same thing in that situation.

## Labelling is an act of programming

- A label is an instruction to the system
- Label providers are engaging in intentional creative acts, which are statistically encoded

This human-centred perspective on machine learning systems focuses on the ways in which system behaviour depends on human actions rather than following physical laws. When a machine appears to behaviour autonomously, we ask whether this behaviour has been derived by observing humans. The observation may either be covert, in which case the intelligence of the system has been achieved by appropriating the subjectively authored intentions of others, or else it is done with their awareness and permission. In the latter (overt) case those users become programmers, determining future system behaviour by authoring examples of what that behaviour should look like.

Labelling is thus a kind of programming, albeit one that is often highly collaborative. A label is an instruction to the system, instructing it by example to behave in a certain way in a certain kind of situation. The system users who provide category labels for supervised learning systems are engaging in (minor) intentional creative acts. Of course, these intentional acts are statistically encoded and aggregated in ways that make it difficult or impossible to acknowledge who the original author was – but the original authors are undeniably humans.

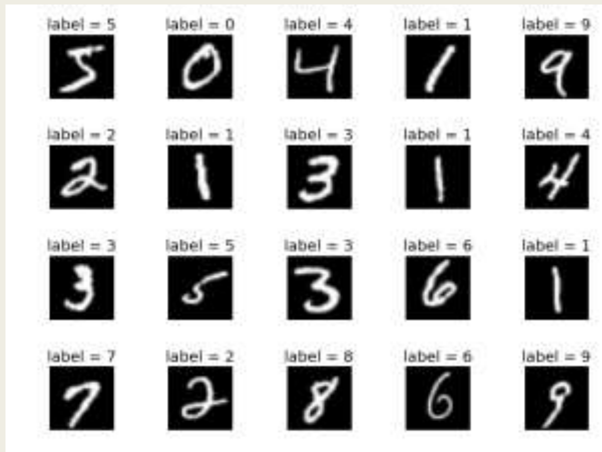
# Human judgement types (non-exhaustive)

- Perceptual judgements
- Judgements that reflect domain expertise
- Judgements of patterns in human experience
- Judgement of patterns in individual intent

So, the purpose of the statistical model in an IML system is not to capture a natural law. Rather, an IML system aims to reproduce human judgment ability. In order to analyse the implications for design, we categorise human judgments into four (non-exhaustive) types.

perceptual judgements,
judgements that reflect domain expertise,
judgement of patterns in human experience, and
judgement of patterns in individual intent.

# Perceptual judgements




*Perceptual* judgments are those that rely principally on the human perceptual system for assignment of a stimulus to a perceptual category. An example is labelling digits in the MNIST database (LeCun Yann, Cortes Corinna,&Burges Christopher, 1998).These are often presented as ‘objective’ judgments, although the assumption of objectivity is only possible because the training examples themselves have been selected to reflect a consensus judgment that the labeller is assumed to share. The MNIST database does not include invalid ‘digits’, non-digits, ambiguous shapes, or artistic subversions of the concept of a digit. Think about the following question: are labels representative of objective ‘facts’ about the neuroscience of human vision, or the subjective assumptions shared by the labellers and data set designers?

# Domain expertise

Sarkar, A., Morrison, C., Dorn, J. F., Bedi, R., Steinheimer, S., Boisvert, J.,...Lindley, S. (2016). Setwise Comparison: Consistent, Scalable, Continuum Labels for Computer Vision. In*Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI'16*(pp. 261–271). New York, New York, USA: ACM Press. https://doi.org/10.1145/2858036.2858199

Chen, N. (2016). Challenges of Applying Machine Learning to Qualitative Coding.*ACM SIGCHI Workshop on Human-Centered Machine Learning*. Retrieved from http://hcml2016.goldsmithsdigital.com/program/

- Concepts may have unclear definitions
- Inter-rater variability (previous experience, training, methods and heuristics used for labelling)
- Access to adequate experts poses logistical challenges, e.g., quorum for averaging

*Domain expertise* judgments rely on labellers' recognised expertise in a particular area. Two example are multiple sclerosis assessment through the analysis of patient videos (Sarkar et al., 2016), and assigning qualitative codes to social science research data (Chen, 2016). Despite these judgments being provided by experts, the concepts being labelled may have unclear definitions, impairing label quality. Moreover, many sources may contribute to inter-rater variability, such as variations in previous experience, training, methods and heuristics used for labelling. Finally, for domain expertise judgments, access to experts is clearly a prerequisite, which may pose logistical challenges if such expertise is rare.

## Human experience judgements


- Universalism
- Variations across age, gender, culture, not encoded, but a primary challenge for affective computing (Picard, 2003)

*Human experience* judgments are those that aim to capture some universal aspect of the human experience. This might be regarded as a special case of the domain expertise judgment where the domain is being human, as opposed to say, a dog or a monkey. An example is capturing labels for affect recognition (Picard, 1997). Here, there is a tenuous assumption that any given person is acting as a representative judge on behalf of all humanity, in relation to universal human experience. In practice, people differ.Typical approaches to mitigate this variation include crowdsourcing and averagingacross labellers. Nonetheless, affect labelling is subject to variations across age, gender,culture, and other factors which are yet to be modelled. While such variation isrecognised as a primary challenge for affective computing (Picard, 2003), it is notexplicitly modelled or acknowledged in the labelling interface (for example, by askingthe labeller to assess the extent of their own individuality).

# Individual intent



- Poor user motivation to provide information, and poor ability to self-report (Afzal & Robinson, 2014)
- ‘Implicit’ signals can work well, but not perfect

*Individual intent* judgments reflect personal feelings, desires, and attributes. Unlike the previous three categories, which appeal to different standards of objectivity (perceptual reality, objective expertise, and universality) these judgements are acknowledged to be inherently subjective because they model an individual. For example, applications built with the EmotionSense platform (Lathia et al., 2013) aim to use emotional inference from mobile phone sensors to induce behavioural change, as a sort of personal therapist. However, the system relies at least partially on self-reporting affective states, which suffers from two issues: users may not be motivated to provide this information repeatedly and consistently, and more importantly, theymay not be capable of consistently self-reporting their emotional state (Afzal&Robinson, 2014). Recommender systems such as Amazon’s product recommendationscircumvent this issue by measuring judgments from concrete actions supposedlyreflecting revealed intent rather than expressed intent: products which were viewedor not viewed, bought or not bought. Such actions are unambiguous signals of intent(because the user interface paradigm enforces this), but are still not immune tomisdirection, for example when a user clicks on multiple irrelevant links in order todisguise their search history.

# Problems of labelling

- Ethical challenges of data collection, e.g. consent
- Label quality depends a lot on the labeller: expertise, judgement ability, attentiveness
- 'Data-hungriness' of models. Solutions: One-shot learning, TrueSkill, etc.?
- Distinction between unclear labels and unclear label boundaries
- Outliers and 'unrateables'
- Incorrect framing of regression as classification

Even before it has been labelled, training data reflects human judgements and priorities. Modern supervised learning techniques require large training sets to build stable models, but the scale of data acquisition can raise ethical challenges, including consent to use data for new purposes, protected categories of data such as clinical patient data, and privacy and anonymity concerns which make it difficult to aggregate data.

While labeling data is a seemingly simple task, it is actuallyfraught with problems (e.g., [9, 19, 26]). Labels reflect a labeler's mapping between the data and their underlying *concept*(i.e., their abstract notion of the target class). Thus, label quality is affected by factors such as the labeler's expertise or familiarity with the concept or data, theirj udgment ability and attentiveness during labeling, and the ambiguity and changing distribution of the data itself.

Moreover, some applications require fast convergence. For instance, the TrueSkill system (Herbrich, Minka,&Graepel, 2006) was developed for matching players inonline games. A gross mismatch in skill results in a less enjoyable experience for allplayers: the weaker player outclassed, and the stronger player unchallenged. A fastestimate of the player's skill, requiring only a few games, is also desirable, as repeatedmismatches may cause players to stop playing the game. Another example of atechnical approach dealing with fast convergence is one-shot learning (Fei-Fei, Fergus,&Perona, 2006).

Data itself carries epistemological assumptions that have been embedded in the way it was collected. From the machine learning perspective, there may not be a formal distinction between *examples* which cannot be placed exactly in the space of labels, and label *boundaries* which are not precise. However, they are very different from the perspective of a human labeller. Imprecise label boundaries may undermine labeller confidence throughout the entire labelling activity. Training examples may also pose problems because they are outliers, or simply unrateable. As noted by Chen (Chen,2016), outliers are typically discarded in quantitative analyses, but become the focus of attention in qualitative analyses. Examples that are unratable (perhaps because of data corruption or because they contain no meaningful information) may impair the labelling process if the labelling tool has no provision to mark examples as unrateable, or the labeller is not equipped to identify such a situation should it arise.

In some cases, a regression problem is incorrectly framed as a classification problem for the purpose of labelling – it is easier to ask labellers to provide one of a discrete set of labels than a real number on a continuous scale. However, this can result in the unnecessary conceptualisation of examples as belonging to a set of discrete categories, which causes issues for examples on the boundaries of different categories. This is the problem faced by the Assess MS problem, detailed in the next section. Unclear concepts cause problems generally in precision, but less so for accuracy.

# Accommodating flexibility


Figure 1. Revolt creates labels for unanimously labeled "certain" items (e.g., cats and not cats), and surfaces categories of "uncertain" items enriched with crowd feedback (e.g., cats and dogs and cartoon cats in the dotted middle region are annotated with crowd explanations). Rich structures allow label requesters to better understand concepts in the data and make post-hoc decisions on label boundaries (e.g., assigning cats and dogs to the cats label and cartoon cats to the not cats label) rather than providing crowd-workers with a priori label guidelines.


Figure 4. Human Intelligence Task (HIT) interface for the Explain Stage. Crowdworkers enter a short description for each item that was labeled differently in the Vote Stage. They were informed that disagreement occurred, but not the distribution of different labels used.


Figure 3. Human Intelligence Task (HIT) interface for the Vote Stage. In addition to the predefined labels, crowdworkers can also select Maybe/NotSure when they were uncertain about the item.

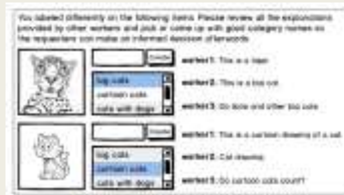


Figure 5. Human Intelligence Task (HIT) interface for the Categorize Stage. Crowdworkers select or create categories for items that were labeled differently in the Vote Stage, based on explanations from all three crowdworkers in the same group.

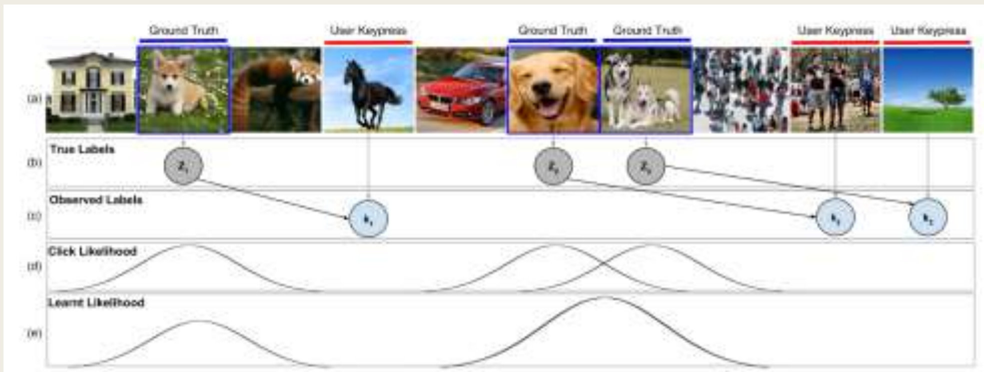
Revolt (Chee Chang et al., CHI 2017)

# Human fallibility, consistency and stamina


Humans are fallible. If there are large amounts of data to be labelled, the quality of judgements can be impaired as the labeller becomes tired. In the Assess MS projectdescribed in the next section, neurologists would spend an entire workday, sometimes two, continuously labelling short video clips (Sarkar et al., 2016). Appropriate tools,such as the setwise comparison tool developed for Assess MS, can mitigate this problem. Explicit strategies to maintain interest and prevent boredom have been applied inexperiments such as the Galaxy Zoo (Lintott et al., 2008) which show compellingevidence for the benefit of ludic and engaging labelling tools.

Even in optimum conditions, people still make mistakes, misinterpret instructions ordisagree with each other. This is well understood in scientific studies where data mustbe categorised by an observer, such as coding of free-text questionnaire responses.Where one researcher might interpret an observed response in one way, another seesit differently. This difference might come from not stating or communicating criteriathat have been applied by one rater, or from terminological imprecision, for example,stemming from a different understanding of the criteria that two raters might have,or simply their wishful thinking in relation to a hypothesis.

# Embracing error to improve speed




Krishna et al., 2016 (Embracing Error to Enable Rapid Crowdsourcing. CHI 2016)

## Measuring label reliability

- Inter and intra-rater reliability measurements
  - *E.g., Cohen's Kappa, Krippendorff's Alpha*
- Error with respect to 'ground truth'

In response to this problem, qualitative social science researchers monitor thereliability of classification judgments. They want to know whether a judge consistentlymakes the same judgment in equivalent cases, and also whether two judges make thesame decision as each other. The second is more often discussed, because it happensso consistently. It is described as inter-rater reliability (IRR), and is often summarisedby a statistical measure such as Cohen's kappa (for the case of two raters), whichcompares the level of agreement to what might be expected from chance. IRR testingis intuitively appealing to computer scientists such as HCI researchers, because thefirst rating can be considered as a design decision, and the second rating as a test ofthat decision. Inter-rater reliability is never 100%, but pragmatic allowance for thelimits of human performance means that certain thresholds are considered acceptablewithin the range of observation error.

The question of whether a single person agrees with themselves (when repeating thesame judgment) is less often asked in computer science, but of more concern inmedicine, where it is quite likely that a clinician might assess the same patient morethan once, with a considerable interval between the assessments. Clinical researchsuggests that this test-retest reliability is also imperfect, with clinicians applyingdifferent criteria at different times, perhaps because of explicit training and correction,or perhaps because of changing tacit or contextual factors that the clinician may notbe consciously aware of. We discuss this issue further next.

# STRUCTURED LABELLING FOR CONCEPT EVOLUTION

Case study I

# Problem: Label concepts evolve over time

- Concept evolution: user process of defining/refining concepts
- Concept drift: labels change over time (related but different)

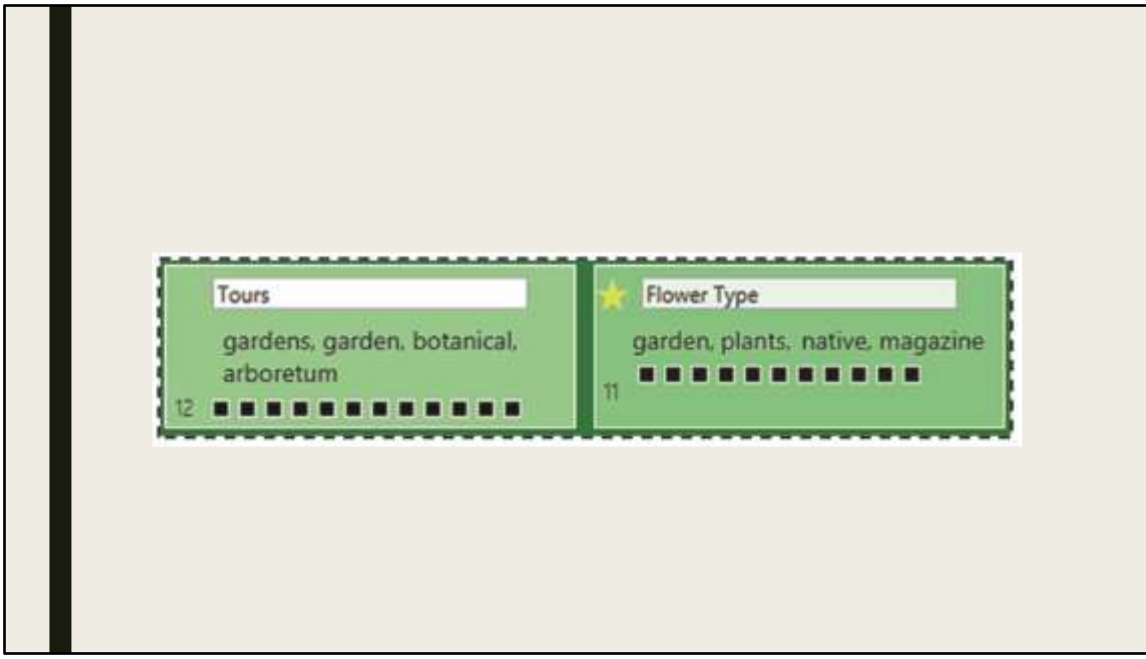
(Mostly from the paper)

This paper addresses a distinct problem in labeling data that we refer to as *concept evolution*. Concept evolution refers to the labeler's process of defining and refining a concept in their minds, and can result in different labels being applied to similar items due to changes in the labeler's notion of theunderlying concept. The paper presents a formative study where the authors found that people labeling a set of web pages twice with a four-week gap between labeling sessions were,on average, only 81% consistent with their initial labels. This inconsistency in labeling similar items can be harmful to machine learning, which is fundamentally based on the ideathat similar inputs should have similar outputs

A separate problem in data labeling is *concept drift*, where the underlying data is fundamentally changingover time [29]. An example of concept drift is a news recommender that attempts to recommend the most interesting recent news. Here, the concept of *interesting* may remain the same over time, but the data (in this case the news) is constantly drifting as a result of changing current events. Most solutions to concept drift model concepts temporally, such as by discarding or weighting information according to a moving window over the data (e.g., [27, 33)or by automatically identifying new types of data (e.g., [5,15]). Critically, none of these solutions are intended to help a *user* refine their own idea of a concept, a problem which may be exacerbated in the presence of concept drift.


we introduce *structured labelling* (Figure 1), a novel interaction technique for helping people define and refine their concepts as they label data. Structured labeling allows people to organize their concept definition by grouping and tagging data (as much or as little as they choose) within a *traditional labelling* scheme (e.g., labeling into mutually exclusive categories such as‘yes’, ‘no’, and ‘could be’). This organization capability helps to increase label consistency by helping people explicitly surface and recall labeling decisions. Further, because the structure is malleable (users can create, delete,split, and merge groups), it is well-suited for situations where users are likely to frequently refine their concept definition as they observe new data.

**Kulesza’s structured labeling approach allows people to group data in whatever way makes sense to them. By seeing theresulting structure, people can gain a deeper understanding of the concept they are modeling. Here, the user sees an uncategorized page (top left) and can drag it to an existing group (right), or create a new group for it. The thumbnails (bottom left) show similar pages in the dataset to help the user gauge whether creating a new group is warranted.**




**Our assisted structuring tool provides users with automatic summaries of each group's contents (below the user-supplied tag area) and recommends a group for the current item via an animation and yellow star indicator. The black squares indicate how many items are in each group.**

# SORTABLE

Case study II

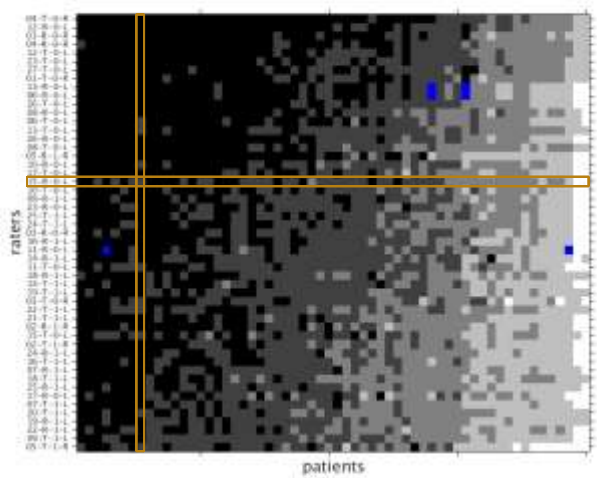
# Assess MS

- Aim: a more consistent way of quantifying progression of motor illness in multiple sclerosis
- Input: Kinect RGB + depth videos of standard clinical movements
- Output: a standardised clinical disability score

# Problem: consistent labels

- Numeric scoring has poor labeller agreement
  - concept boundaries unclear even after iteration
- Crowdsource?
  → can't, need highly expert labellers
- Average across labellers?
  → can't, patient confidentiality
- Model individual labeller noise/bias?
  → can't, learning effects

## Inter-rater consistency is limited






NOVARTIS

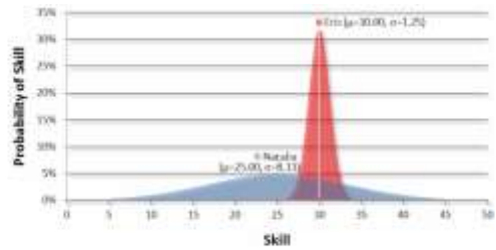
# Partial solution

- Preference judgements
  - ‘this is **better / worse / equal** to that’ as opposed to ‘this is a **3**, that is a **4**’.
  - Not scalable :(

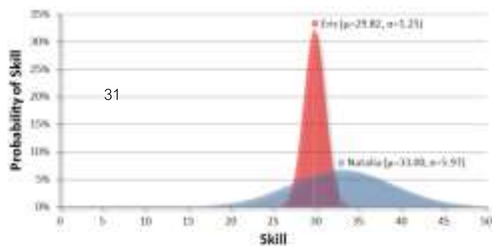
# A better solution

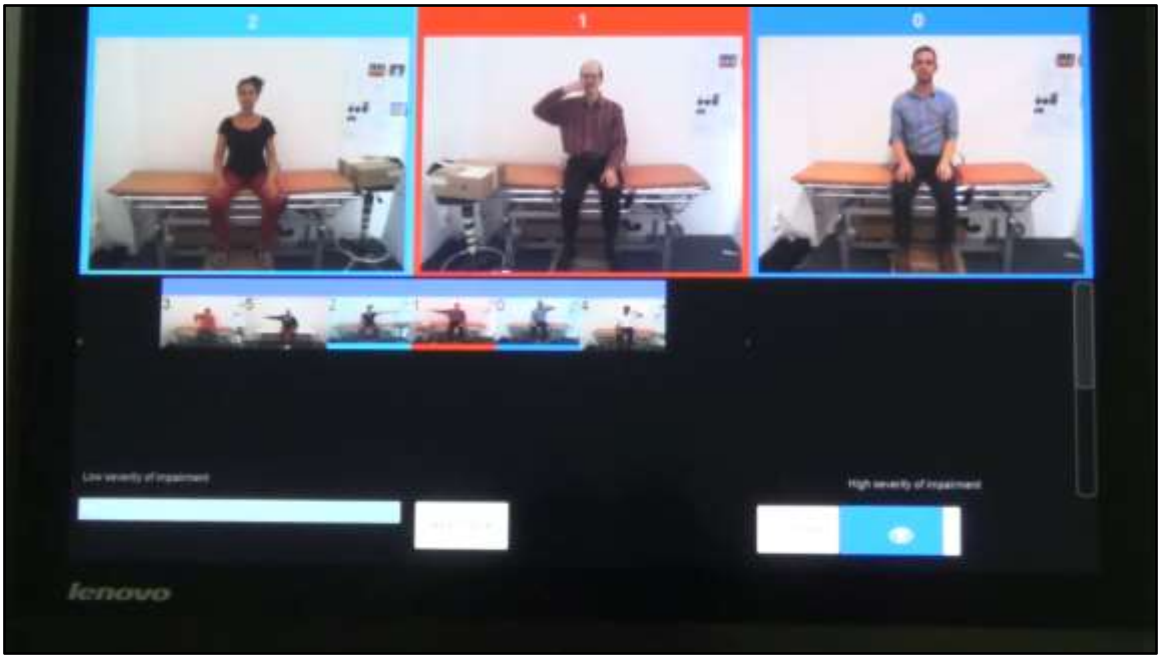
- Setwise comparison + TrueSkill inference
  - Order **sets** of videos with overlap
  - but don’t need all pairwise comparisons
  - **Infer** remaining relationships

Prior



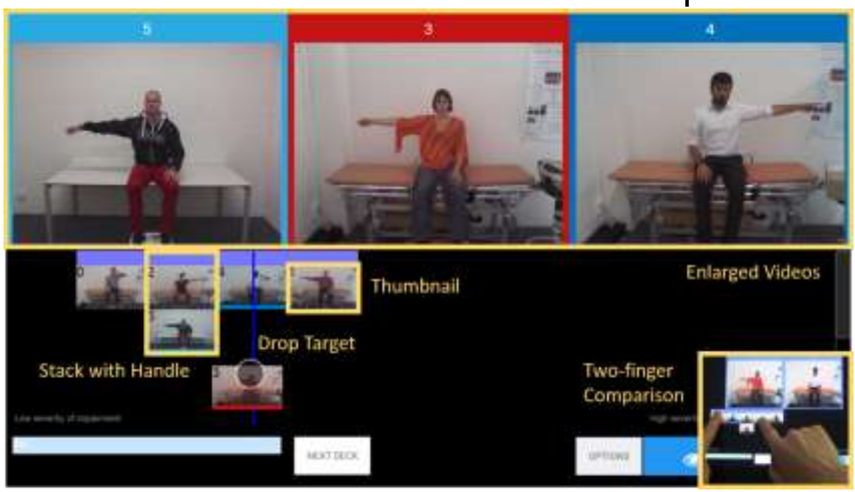

After Natalia wins




2
1
0
Low severity of impairment
High severity of impairment
lenovo

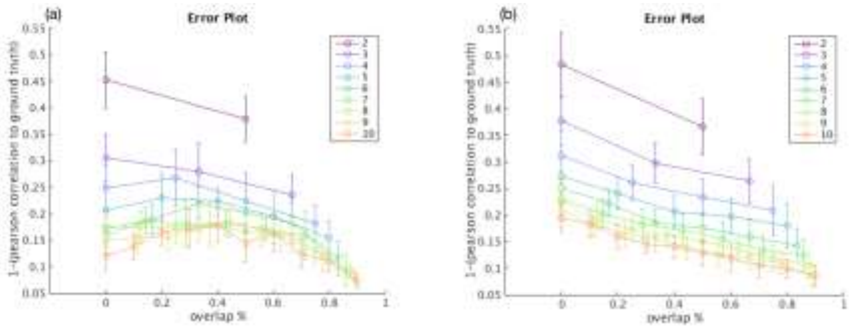
# SorTable

## an interface for setwise comparison

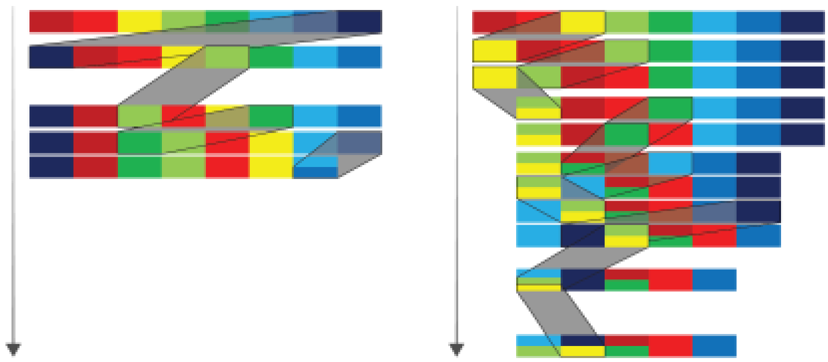



33

# Choosing deck size and overlap



# Sorting strategies

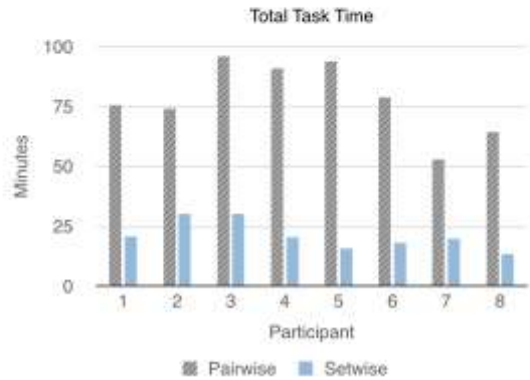


# So, does it work?

- Already known: pairwise comparison achieves higher consistency than assigning numerical scores, but very slow
- **Question**: Does setwise comparison achieve a better efficiency-consistency tradeoff?
- Compared pairwise and setwise using 8 neurologists rating a set of 40 videos

# Result 1:
# Setwise comparison is more efficient

- Setwise task time was
  54 minutes less
  on average
  ($p = 4 \cdot 10^{-5}$)




37

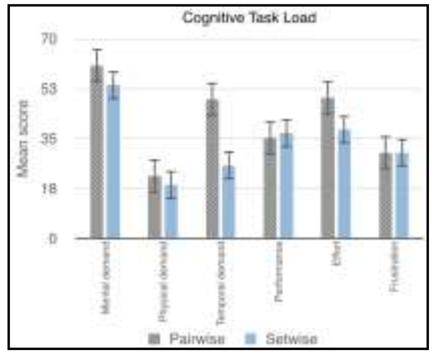
# Result 2:
# Setwise comparison is more consistent!

Agreement ***between*** labellers

| | **Global ICC** | **Average ICC** |
|---|---|---|
| | | mean±sd [min−max] |
| *Pairwise* | 0.70 | $0.77 \pm 0.1 [0.64 - 0.94]$ |
| *Setwise* | 0.83 | $0.85 \pm 0.07 [0.72 - 0.95]$ |
| *t-test* | | $p = 5 \cdot 10^{-4}$ |

# Why is it more consistent???

- *Inferring* missing comparisons was better than *measuring* all comparisons.
- Cognitive load assessment was inconclusive.
- Potential explanations:
  - Fatigue
  - TrueSkill's implicit noise modelling
  - Increased reference points



## Sortable: conclusions

- Labels need not be solicited directly, but can be inferred
- Interaction design eased the burden of labelling
- The most informative labels are not necessarily the best

We reframed the problem so that users were not providing labels directly, but providing information from which labels could be reconstructed. In this way, we could build upon strong human capability in relative judgement and still provide the classification labels required by the Assess MS system. This overcame noisy labels,improving the accuracy of the algorithm by 10%.

A key insight was to by enabling setwise rather than pairwise comparison, achieving three benefits for the users. First, the presentation of videos in sets builds upon human short-term memory to make multiple comparisons at once. Second, the ability to create stacks to indicate that videos are the same can substantially reduce the number of comparisons the labeller needs to make when sorting. Third, SorTable facilitates mixed-strategy sorting, including the automatic display of the left and right neighbours of the currently selected video, and the ability to compare any two videos with a two-finger gesture. All interactions are touch based.

We found that choosing videos to label to maximise TrueSkill's information gain and ultimately decrease the number of required labels was not a good strategy for human labellers. It is less cognitively taxing for people to differentiate between very different videos rather than similar ones. Put differently, labels that satisfy a classifier's information needs perfectly may also be the hardest for humans to give

(Lang&Baum,1992), and increase stress and fatigue.

# INFERRING UNITS IN SPREADSHEETS

Top search results for *spreadsheet*.

10 contain numbers that have some form of *unit*.

456/867 unit annotated workbooks from EUSES referred to some unit.

By units we mean physical units like grams, seconds, or currencies.

Units are core to many spreadsheet domains.

Unit information is valuable for:

- Catching errors.
- Presenting information.
- Localisation.
- Comprehension.

*But most spreadsheet systems do not directly support units and even if they did, users may not provide new unit information.*

Our challenge is ***unit inference:*** given a numeric cell, tell me its unit.

# The Task

| | A | B |
|---|---|---|
| 1 | 10 | weight in kgs |
| 2 | 20 | |
| 3 | 30 | ($/kg) |
| 4 | =(A1+A2)*A3 | |

Given a spreadsheet, only a subset of the cells must have a unit annotation in order to fully infer the units in the sheet.

These are the **critical cells**. They could be: { A1, A3 }, { A2, A3 }, or { A3, A4 }.

**Our task:** synthesise a unit annotation for critical cells using text in the sheet.

Orchard et al. Evolving Fortran types with inferred units-of-measure. ICCS 15.

# Our Solution (Part One)

We know that inference is worthwhile, and we have a mechanism to evaluate it. We just need to implement it!

1. Run a logical inference algorithm. Output critical cells.
2. Annotate critical cells using nearby text cells that match unit templates such as:
   "Area (*acres*)" or "*dollars* per *month*".

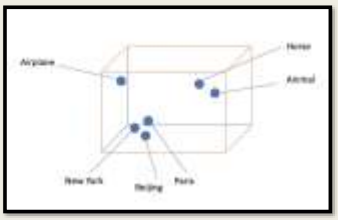
Problem.

Many text cells are like "Credit card charges" rather than "Area (acres)".

Our templates are precise, but have low recall.

# Our Solution (Part Two)

Use a machine learning model to extract dimensions from text cells if we fail to match a template.




We start with a word embedding that maps words into a vector space. ‘Similar’ words are ‘close’ in the vector space.

For a given text cell, we assign a score to each dimension (rather than unit).

In words:

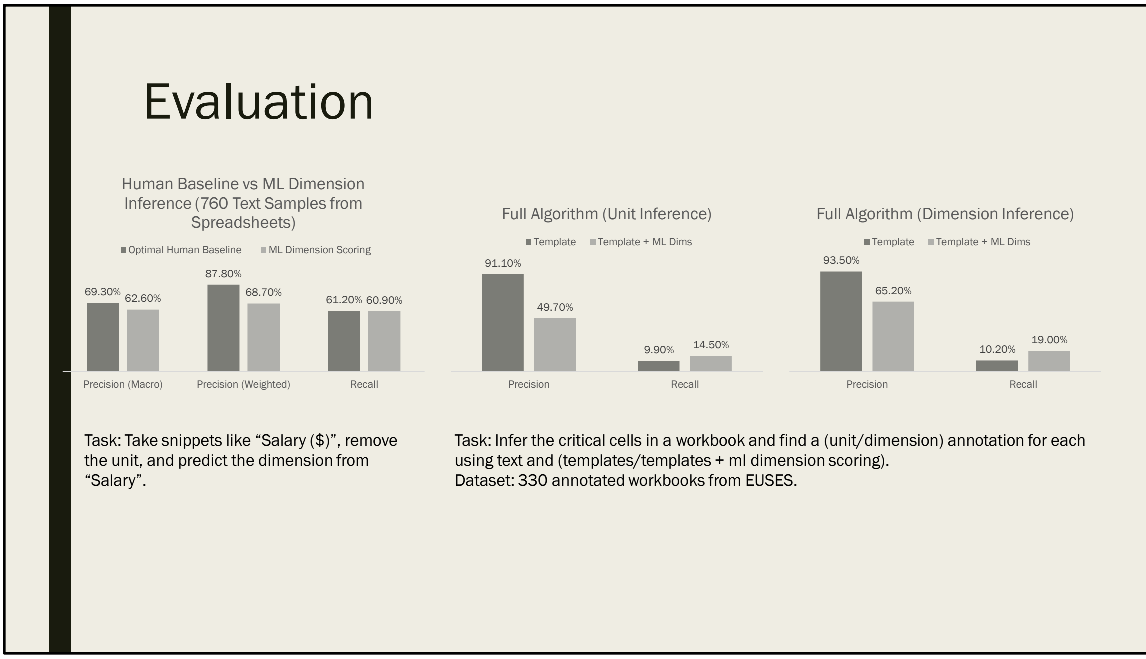
- The score for a dimension ***d*** with respect to a text cell ***t*** is the max score of a unit ***u*** in ***d*** with respect to ***t***.
- The score for a unit ***u*** with respect to ***t*** is the average ‘distance’ between the embedding for ***u*** and the embedding of each word in ***t***.

Subject to a weak transitivity constraint.

Cosine similarity.

at the end of this… so we’re done, right?

We take the inference approach of Chambers and Erwig,although we aim to infer concrete physical units (insteadof dimensions). Through a fully-automated process based onformulas, formatting and nearby textual labels (described inSection V), we infer the units of each critical variable withoutany upfront user attention requirements. By reducing the(apparent) cost to the user to zero, we can greatly reduce thebarrier to adoption.Of course, there is no free lunch.



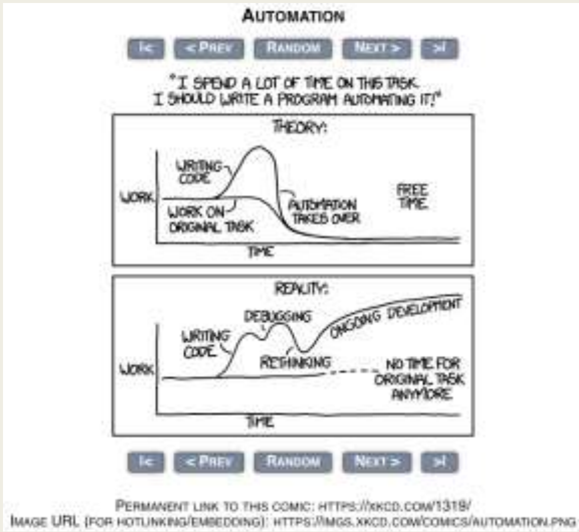

The catch is that inferenceis not perfect, and when inferred units are incorrect, the userwill need to invest attention to rectify the inference (a tradeoffthat has not been previously acknowledged in such work).The question is under what circumstances does this result in asituation beneficial to the user, i.e., under what conditions doesthe unit inference system result in a lower overall attentioninvestment cost?

# Attention Investment (Blackwell)

- The decision to start programming is based on an implicit cost-benefit analysis:
  - *cost of getting the work done manually*
  - *investment cost of automation*
  - *pay-off: the overall cost reduction as a result of automation*
  - *risk: probability no payoff will result, or additional costs incurred*

- Blackwell, Alan F. "First steps in programming: A rationale for attention investment models." *Proceedings IEEE 2002 Symposia on Human Centric Computing Languages and Environments*. IEEE, 2002.

This question is precisely the one answered by the decisioncalculus of Horvitz's principles for mixed-initiative systems[17], but applied to the user's attention. Our key observation,which allows us to combine the theories of attention invest-ment and mixed-initiative systems, is that the utility functionsin Horvitz's calculus can be expressed in terms of Blackwell'sattention units.




- 'Automating' comes from the roots 'auto-' meaning 'self-', and 'mating', meaning 'screwing'.

# Simplified model of error production

- Over the course of interacting with a spreadsheet (authoring, editing, reading, etc.), a unit error occurs with some probability $P_u$.
- If a unit error occurs, the user incurs an attentional cost $R_u$ of recovering from the unit error.
- However, if we have a working inference system, the cost of recovering from a unit error is zero.
- If there is an inference error (which occurs with probability $P_i$), the user must recover from it (with cost $R_i$).

Without inference, the expected cost is

$$P_u R_u + (1 - P_u) \cdot 0 = P_u R_u$$

The cost with inference is:

$$P_u(P_i R_i + (1 - P_i) \cdot 0) + (1 - P_u)(P_i R_i + (1 - P_i) \cdot 0)$$
$$= P_i R_i$$

So the system lowers the overall attentional costs of using spreadsheets if:

$$P_i R_i < P_u R_u$$

Finally if we design the system such that: $R_i \leqslant R_u$ then we obtain the bound: $P_i < P_u$

Similarly, we derive an expression for the expected cost with inference, with terms corresponding to the four cases where unit errors do and do not occur, and inference errors do and do not occur. Recall our assumption that when inference works, the cost of fixing a unit error is zero. Therefore, in the case where there is both a unit error and an inference error, we assume that resolving a unit inference error must also resolve any unit errors and therefore costs at mos tRi, not Ri + Ru.

If we now furtherassume our system is designed such thatRi¤Ru, that is, thecost of recovering from a unit inference error is not higherthan the cost of recovering from a unit error (a reasonabledesign objective), we obtain the boundPiPu.

Thus, we arrive at a simple, calculable criterion by whichwe can contextualise the performance of an imperfect error-prevention system: in order for an inference system to lowerthe expected attentional cost to the user, the rate of inferenceerror must be less than the natural rate of the error that thesystem is designed to prevent. Previous work estimates thatdimension errors occur in 42.5% of spreadsheets [2], thus theerror rate of our system must also not exceed 42.5%.

# Simplifying assumptions

- Risk-neutrality
- No external costs
- Single error
- Guaranteed error discovery and recovery
- Zero-sum inference
- Inference has cheaper recovery
- Fixed error probabilities and costs
- Short-term/long-term conflation

Risk-neutrality: we assume the user is risk-neutral; that is,it is sufficient for the expected attentional cost of a system withinference to be merely lower than the expected attentional costwithout inference. However, behavioural economics shows thatpeople can be risk-averse or risk-loving, with most peoplebeing slightly risk-averse [18]. For example: given the choiceof a 50% chance of winning $100, or a guaranteed win of$50, which would you choose? A risk-neutral person viewsboth options as equivalent due to their equal expected payoff. risk-averse person prefers the uncertain win only if theexpected payoff is higher than that of the certain win; the difference between those two quantities is known as theperson'srisk premium. It is almost certainly the case that usersof inference systems are slightly risk-averse, and thereforeour inference system must not merely match the attentionrequirements of the status quo, but improve upon it by a riskpremium (that might be possible to empirically determine, buthas not yet been done).

No external costs: we only modelattentionalcosts andutility. The full cost of an error in a spreadsheet variesaccording to its context; a unit error might result in incorrectreal-world decisions, financial and reputational loss, and manyother negative externalities. It is unclear how to model oraccount for these in a principled way.

Single error: we do not model multiple errors and episodesof error recovery.

Guaranteed error discovery and recovery: we do notmodel the likelihood of the usernotdetecting unit andinference errors, and ofnotfixing them. We assume that ifa unit or inference error exists, the user always discovers it,chooses to fix it, and does so successfully. In the case whereboth a unit and an inference error occurs, the user discoversand fixes the inference error (which automatically fixes theunit error, see next point).

Zero-sum inference: we assume that if unit inferenceworks, then the cost of recovering from a unit error is zero.This would be trivially the case if unit inference preventedunit errors from occurring in the first place. In this casePucan be interpreted as the probability that a unit errorwouldhaveoccurred without the interface. This assumption and theprevious one subsume another assumption we make (whichHorvitz's model is particularly concerned about), namelyperfect inference of user goals. That is, we assume that theway in which our inference system ultimately fixes or preventsunit errors is always perfectly aligned with the user's goals.

Inference has cheaper recovery: the cost of recoveringfrom a unit inference error is less than or equal to the cost ofrecovering from a unit error (note a corollary design principle:incorrect inference should not be error-genic; if the inferencesystem introduces the very error it is designed to prevent, thecost of recovering from an inference error cannot be less thanthe cost of recovering from a unit error).

Fixed error probabilities and costs: we model the proba-bility of unit and inference errors to be fixed for all users andspreadsheets (e.g., interpreted as an empirical probability).

Short-term/long-term conflation: we do not distinguishbetween Blackwell's long-term focus (on the inference systemas a whole) and Horvitz's short-term focus (on each individualopportunity for inference and user interruption). In the futurewe might treat these differently, using long-term empiricalprobabilities for the former analysis, and sheet-specific prob-abilities generated by our inference model for the latter

# Attention investment & mixed-initiative systems, two sides of the same coin?

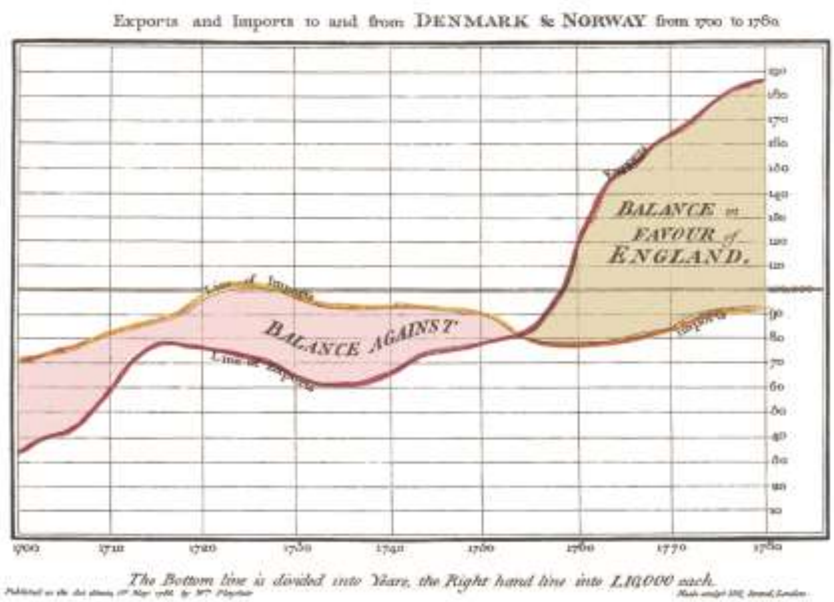
| Aspect | Attention investment | Mixed-initiative systems |
| --- | --- | --- |
| Purpose of model | To explain user behaviour | To determine system behaviour |
| Decision problem | Is the expected payoff of automation greater than that of non-automation? If so, the user takes action. | Is the expected utility of the (automated) action greater than that of inaction? If so, the system takes action. |
| Instance of concern | This model applies at each investment opportunity, that is, each time the user has an opportunity to automate something. | This model applies at each inference/automation/interruption opportunity, that is, each time the system can take an individual action. |
| Implementation of model | This is a long-term calculus in the user's mind. In our context, we assume a rational, learning user, who will eventually approximate $P_u$ to be the long term rate of unit error, $P_i$ to be the overall inference error rate. | This is a short-term calculus which the system can calculate for any given prediction. In our context, $P_u$ would be interpreted as the sheet or cell error likelihood, and $P_i$ would be the inference confidence in a specific prediction. |

Since our system sits at the intersection of concerns treatedby both Blackwell's account of attention investment and Horvitz's account of mixed-initiative systems, we have con-ducted an analysis that draws on concepts from both. In doingso, we have been able to identify a number of similaritiesand differences between them. In Table II, we present ourcomparison of the two theories.These theories approach two different problems from twovery different perspectives, but ultimately produce a mathe-matically identical solution (namely, to compute the expectedpayoff to the user of implementing a technical intervention,versus not implementing it). Therefore, when applying thesetheories in new contexts, it is important to consider theirdifference in perspective, because though the equations arethe same, our interpretation of the quantities encoded varies.

# VISUALISATION

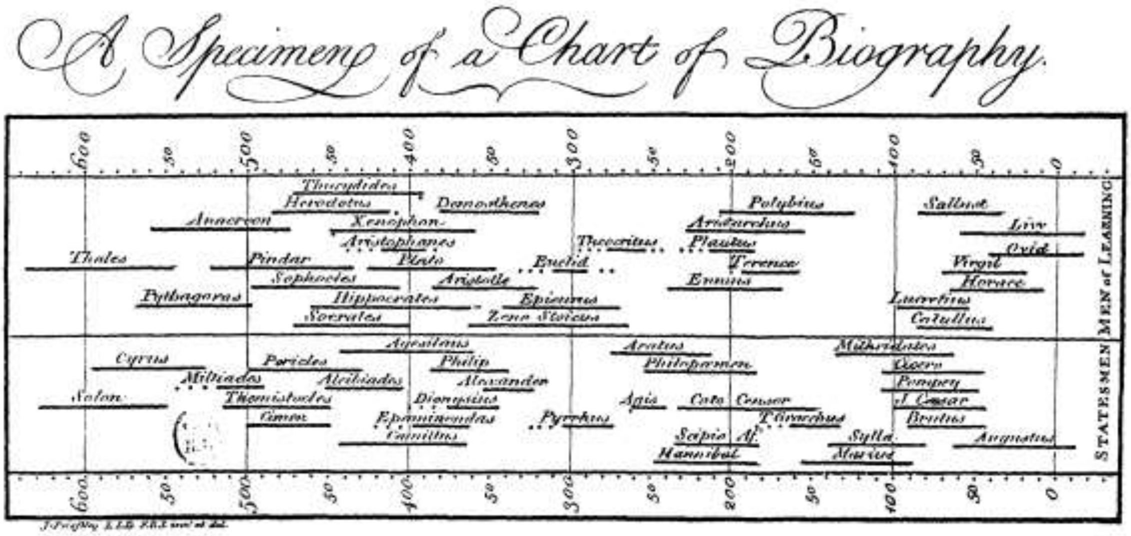
Interaction with Machine Learning

Cambridge MPhil ACS + CST Part II / III 2021-2022



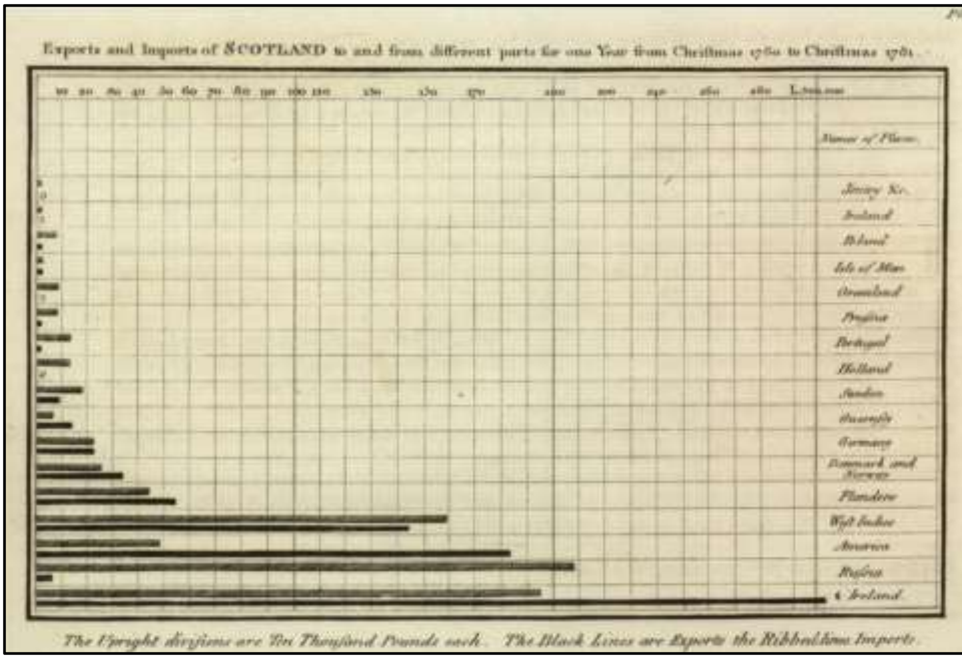

Time Series area chart
William Playfair, c1786

**William Playfair** (22 September 1759 – 11 February 1823) was a Scottish engineer and political economist, the founder of graphical methods of statistics.[1] He invented several types of diagrams: in 1786 the line, area and bar chart of economic data, and in 1801 the pie chart and circle graph, used to show part-whole relations.




Lifespan chart
Joseph Priestly, c1765

Two decades before Playfair's first achievements, in 1765 Joseph Priestley had created the innovation of the first timeline charts, in which individual bars were used to visualise the life span of a person, and the whole can be used to compare the life spans of multiple persons. According to James R. Beniger and Robyn (1978) "Priestley's timelines proved a commercial success and a popular sensation, and went through dozens of editions".



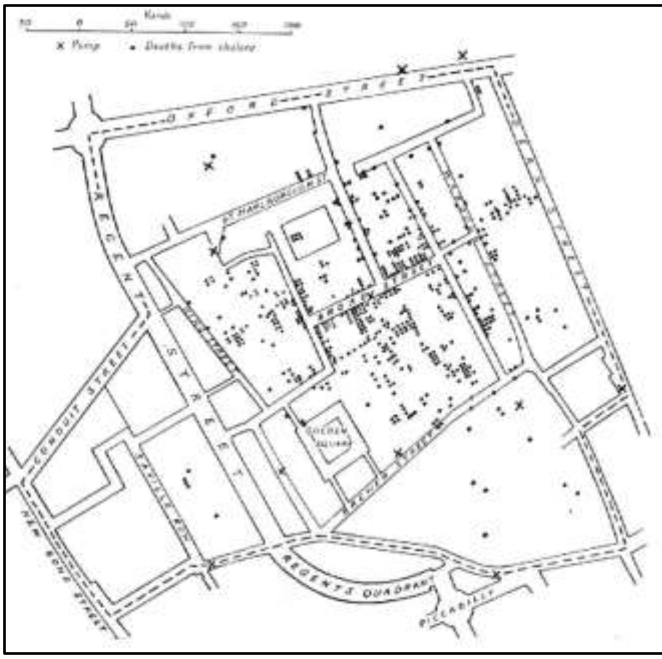

Bar chart
William Playfair, c1786

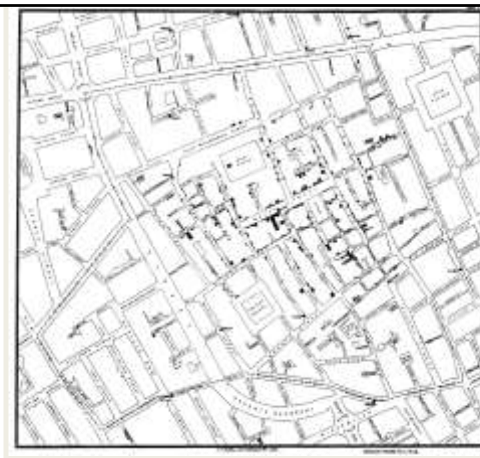
These timelines directly inspired Wiliam Playfair's invention of the bar chart, which first appeared in his *Commercial and Political Atlas*, published in 1786.

Playfair was driven to this invention by a lack of data. In his Atlas he had collected a series of 34 plates about the import and export from different countries over the years, which he presented as line graphs or surface charts: line graphs shaded or tinted to show the difference [skip back to slide].

Because Playfair lacked the necessary series data for Scotland, he graphed its trade data for a single year as a series of 34 bars, one for each of 17 trading partners, In this bar chart Scotland's imports and exports from and to 17 countries in 1781 are represented.

"This bar chart was the first quantitative graphical form that did not locate data either in space, as had coordinates and tables, or time, as had Priestley's timelines. It constitutes a pure solution to the problem of discrete quantitative comparison".



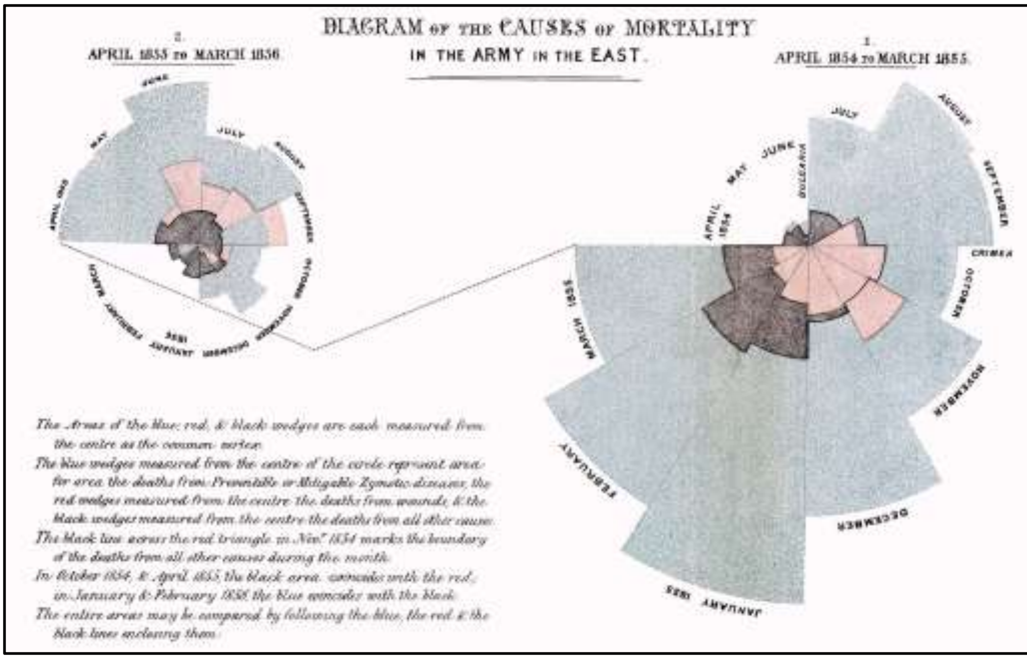
Cholera map
John Snow, 1854

John Snow (15 March 1813 – 16 June 1858) was an English physician and a leader in the adoption of anaesthesia and medical hygiene. He is considered one of the fathers of modern epidemiology, in part because of his work in tracing the source of a cholera outbreak in Soho, London, in 1854

Snow was a skeptic of the then-dominant miasma theory that stated that diseases such as cholera and bubonic plague were caused by pollution or a noxious form of "bad air". The germ theory of disease had not yet been developed, so Snow did not understand the mechanism by which the disease was transmitted. His observation of the evidence led him to discount the theory of foul air. He first publicised his theory in an 1849 essay, *On the Mode of Communication of Cholera*,[14] followed by a more detailed treatise in 1855 incorporating the results of his investigation of the role of the water supply in the Soho epidemic of 1854.[15][16]

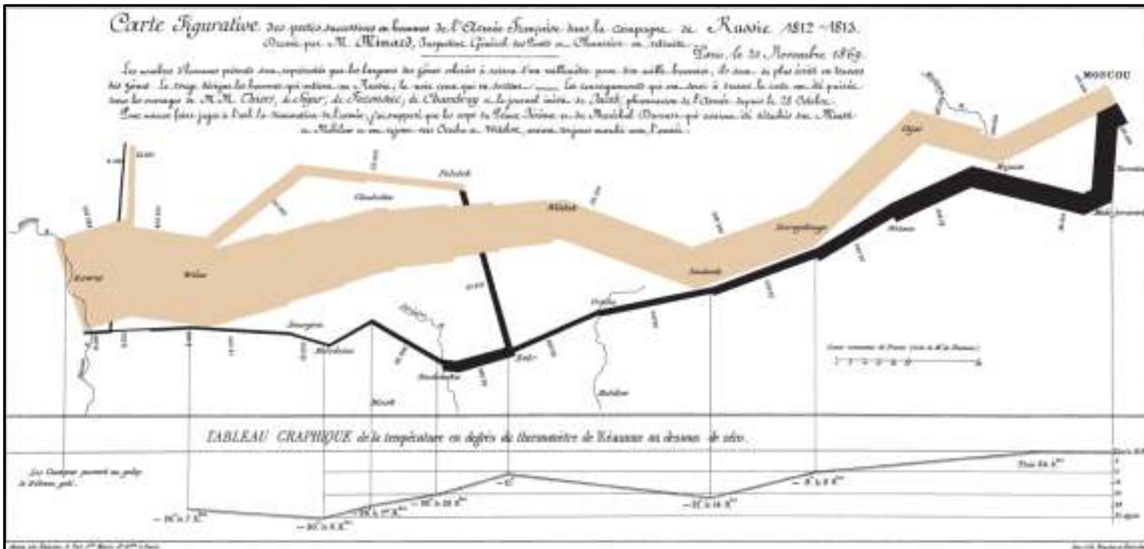
By talking to local residents (with the help of Reverend Henry Whitehead), he identified the source of the outbreak as the public water pump on Broad Street (now Broadwick Street). Although Snow's chemical and microscope examination of a water sample from the Broad Street pump did not conclusively prove its danger, his studies of the pattern of the disease were convincing enough to persuade the local council to disable the well pump by removing its handle.

Snow used a dot map to illustrate the cluster of cholera cases around the pump. He also used statistics to illustrate the connection between the quality of the water source and cholera cases. He showed that the Southwark and Vauxhall Waterworks Company was taking water from sewage-polluted sections of the Thames and delivering the water to homes, leading to an increased incidence of cholera. Snow's study was a major event in the history of public health and geography. It is regarded as the founding event of the science of epidemiology. Snow's map, demonstrating the spatial clustering of cholera deaths around the Broad Street well, provided strong evidence in support of his theory that cholera was a water-borne disease. Snow used some proto-GIS methods to buttress his argument: first he drew Thiessen polygons around the wells, defining straight-line least-distance service areas for each. A large majority of the cholera deaths fell within the Thiessen polygon surrounding the Broad Street pump, amd a large portion of the remaining deaths were on the Broad Street side of the polygon surrounding the bad-tasting Carnaby Street well. Next, using a pencil and string, Snow redrew the service area polygons to reflect shortest routes along streets to wells. An even larger proportion of the cholera deaths fell within the shortest-travel-distance area around the Broad Street pump.




Rose diagrams
Florence Nightingale, c1858

In 1858 nurse, statistician, and reformer Florence Nightingale published *Notes on Matters Affecting the Health, Efficiency, and Hospital Administration of the British Army. Founded Chiefly on the Experience of the Late War. Presented by Request to the Secretary of State for War*. This privately printed work contained a color statistical graphic entitled "Diagram of the Causes of Mortality in the Army of the East" which showed that epidemic disease, which was responsible for more British deaths in the course of the Crimean War than battlefield wounds, could be controlled by a variety of factors including nutrition, ventilation, and shelter. The graphic, which Nightingale used as a way to explain complex statistics simply, clearly, and persuasively, has become known as Nightingale's "Rose Diagram."




*Figurative Map of the successive losses in men of the French Army in the Russian campaign 1812-1813.*

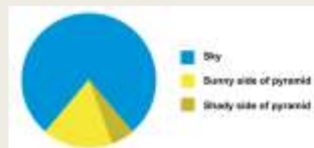
*Drawn by Charles Minard, Inspector General of Bridges and Roads (retired). Paris, November 20, 1869.*

Map of Napoleon's army by Charles Joseph Minard. Minard was a pioneer of the use of graphics in engineering and statistics. He is most well known for his cartographic depiction of numerical data on a map of Napoleon's disastrous losses suffered during the Russian campaign of 1812. The illustration depicts Napoleon's army departing the Polish-Russian border. A thick band illustrates the size of his army at specific geographic points during their advance and retreat.

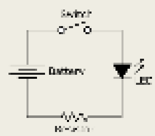
This graphic is notable for displaying six types of data in two dimensions: the number of Napoleon's troops; the distance traveled; temperature; latitude and longitude; direction of travel; and location relative to specific dates.[2] This type of band graph for illustration of flows was later called a Sankey diagram, although Matthew Sankey used this visualisation 30 years later and only for thematic energy flow).

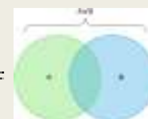
# What is visualisation?

- Charts & statistical visualisations
- Typography & typesetting
- Diagrams
- Illustrations and drawings
- Infographics
- Symbols
- Marks



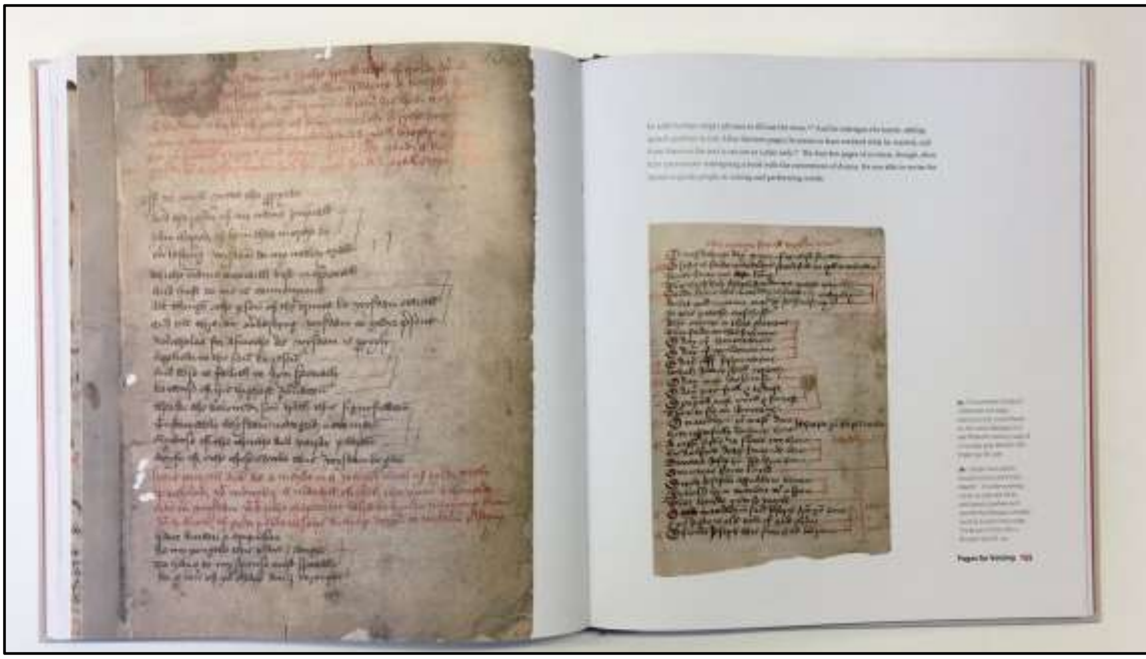




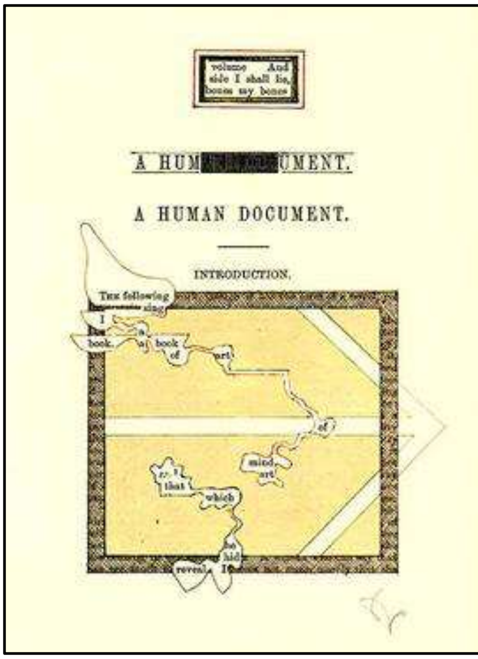




When you hear the word visualisation, you might think of a bar chart or a pie chart.


Daniel Wakelin



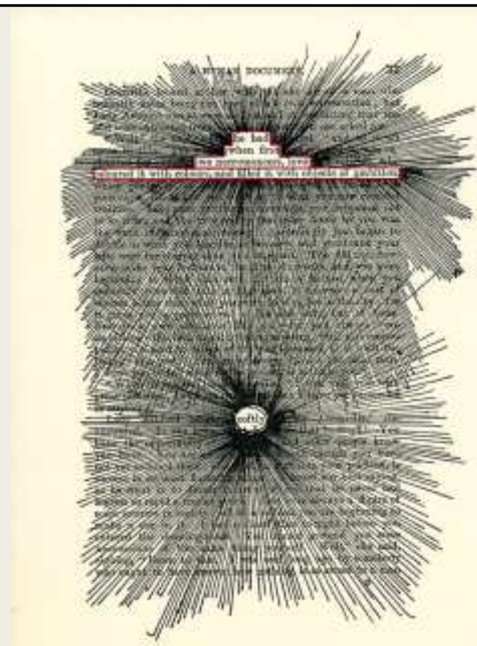
Late 15th century morality play.
A poem converted into a play.

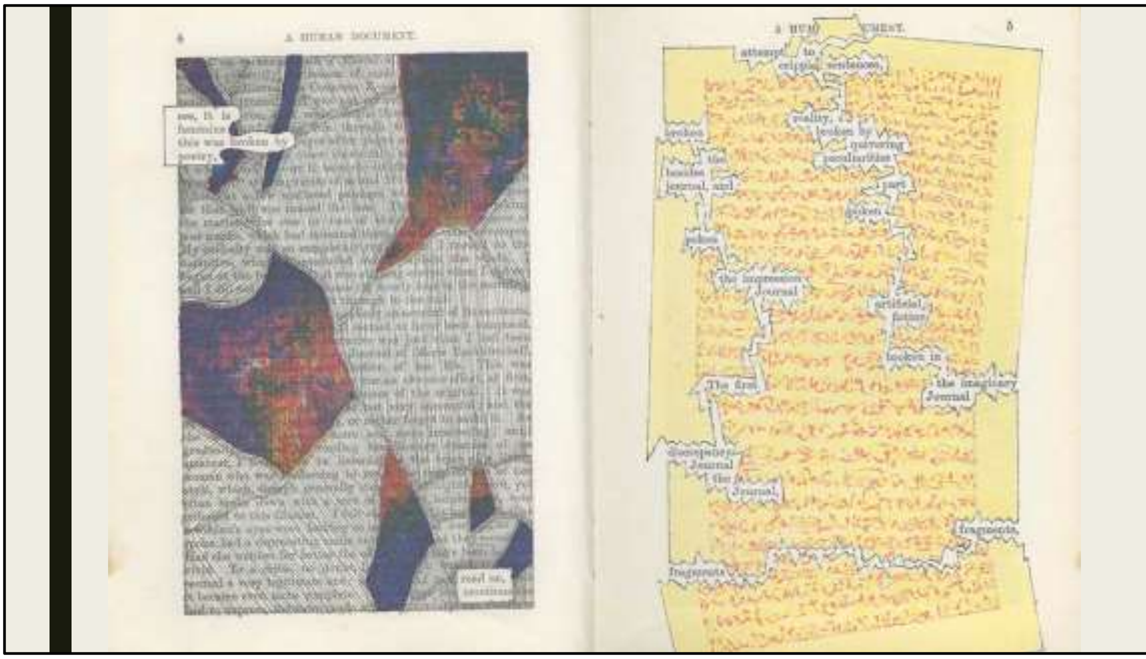



A Humument
Tom Phillips, 1960s

"It is a forgotten Victorian novel found by chance … plundered, mined, and undermined its text to make it yield the ghosts of other possible stories, scenes, poems and replaced the text [he'd] stripped away with visual images of all kinds."

Tom Phillips, 1960s

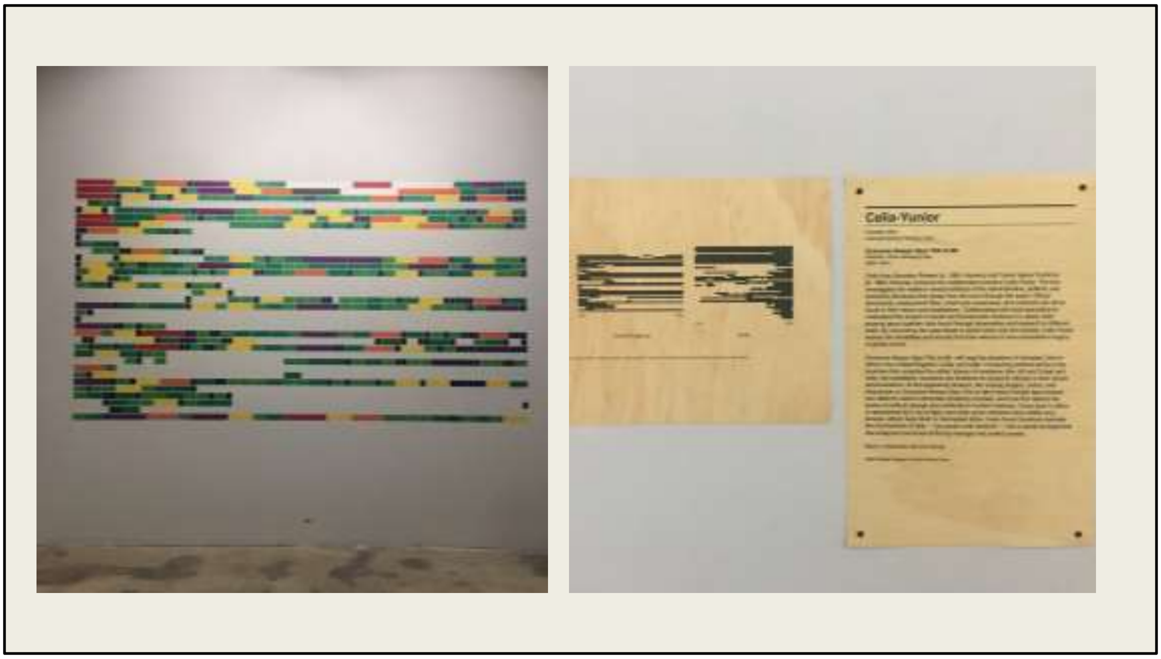



A HUMAN DOCUMENT.
attempt to cripple sentences,
reality, broken by quivering peculiarities
broken
fragments
fragments.


CURATORIAL NOTE
POSSIBILITIES
അന്യതയിൽ
FOR A NON-
നിന്നും അന്യോന്യത
ALIENATED LIFE
യിലേക്ക്


India infographic 1950s - Chittaprosad Bhattacharya - cartoonist

Celia Yunior – growth of IT industry in Kerala, income disparity




Celia Yunior – positions of power and control in administrations

History – a construct

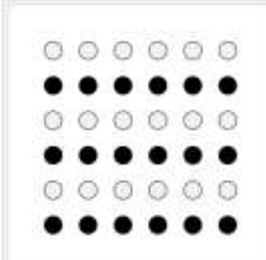
# THEORIES OF VISUALISATION

Gestalt principles of visual perception

Law of Proximity

Law of good continuation

Law of common fate—a flock of birds.



Law of Similarity

Law of Closure

The **principles of grouping** (or **Gestalt laws of grouping**) are a set of principles in psychology, first proposed by Gestalt psychologists in the early 20th century to account for the observation that humans naturally perceive objects as organized patterns and objects, a principle known as Prägnanz. Gestalt psychologists argued that these principles exist because the mind has an innate disposition to perceive patterns in the stimulus based on certain rules.

For example, the law of common fate. Birds may be distinguished from their background as a single flock because they are moving in the same direction and at the same velocity, even when each bird is seen—from a distance—as little more than a dot. The moving 'dots' appear to be part of a unified whole. The law of common fate is used extensively in user-interface design, for example where the movement of a scrollbar is synchronised with the movement (i.e. cropping) of a window's content viewport; The movement of a physical mouse is synchronised with the movement of an on-screen arrow cursor, and so on.

The principle of similarity states that, all else being equal, perception lends itself to seeing stimuli that physically resemble each other as part of the same object, and stimuli that are different as part of a different object.

The Gestalt law of proximity states that "objects or shapes that are close to one another appear to form groups".

The principles of similarity and proximity often work together to form a Visual Hierarchy. Either principle can dominate the other, depending on the application and combination of the two. For example, in the grid to the left, the similarity principle dominates the proximity principle and you probably see rows before you see columns.

The principle of closure refers to the mind's tendency to see complete figures or forms even if a picture is incomplete

The law of good continuation. When there is an intersection between two or more objects, people tend to perceive each object as a single uninterrupted object.

| | Graphic Resources | Correspondence | Design Uses |
|---|---|---|---|
| Marks | Shape<br>Orientation<br>Size<br>Texture<br>Saturation<br>Colour<br>Line | Literal (visual imitation of physical features)<br>Mapping (quantity, relative scale)<br>Conventional (arbitrary) | Mark position, identify category (shape, texture colour)<br>Indicate direction (orientation, line)<br>Express magnitude (saturation, size, length)<br>Simple symbols and colour codes |
| Symbols | Geometric elements<br>Letter forms<br>Logos and icons<br>Picture elements<br>Connective elements | Topological (linking)<br>Depictive (pictorial conventions)<br>Figurative (metonym, visual puns)<br>Connotative (professional and cultural association)<br>Acquired (specialist literacies) | Texts and symbolic calculi<br>Diagram elements<br>Branding<br>Visual rhetoric<br>Definition of regions |
| Regions | Alignment grids<br>Borders and frames<br>Area fills<br>White space<br>Gestalt integration | Containment<br>Separation<br>Framing (composition, photography)<br>Layering | Identifying shared membership<br>Segregating or nesting multiple surface conventions in panels<br>Accommodating labels, captions or legends |
| Surfaces | The plane<br>Material object on which marks are imposed (paper, stone)<br>Mounting, orientation and display context<br>Display medium | Literal (map)<br>Euclidean (scale and angle)<br>Metrical (quantitative axes)<br>Juxtaposed or ordered (regions, catalogues)<br>Image-schematic<br>Embodied/situated | Typographic layouts<br>Graphs and charts<br>Relational diagrams<br>Visual interfaces<br>Secondary notations<br>Signs and displays |


Bertin, J. (1967). Semiologie graphique. Paris: Editions Gauthier-Villars. English translation by WJ. Berg (1983)as Semiology of graphics, Madison, WI: University of Wisconsin Press

Blackwell, A.F. and Engelhardt, Y. (2002). A meta-taxonomy for diagram research. In M. Anderson&B. Meyer&P. Olivier (Eds.), Diagrammatic Representation and Reasoning, London: Springer-Verlag, pp. 47-64.

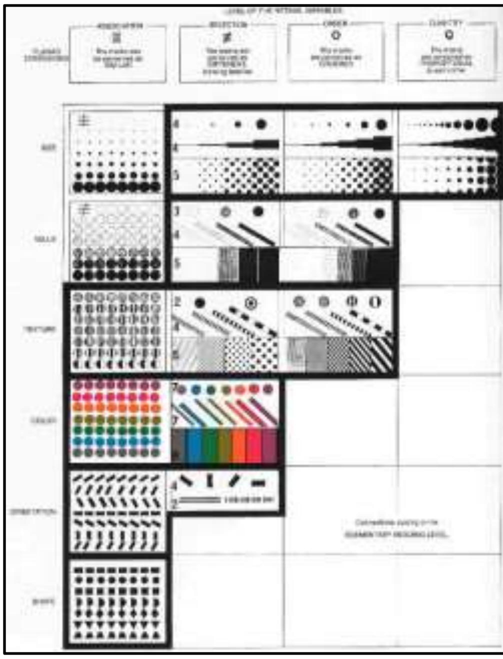
Engelhardt, Y. (2002). The Language of Graphics. A framework for the analysis of syntax and meaning in maps,charts and diagrams. PhD Thesis, University of Amsterdam.

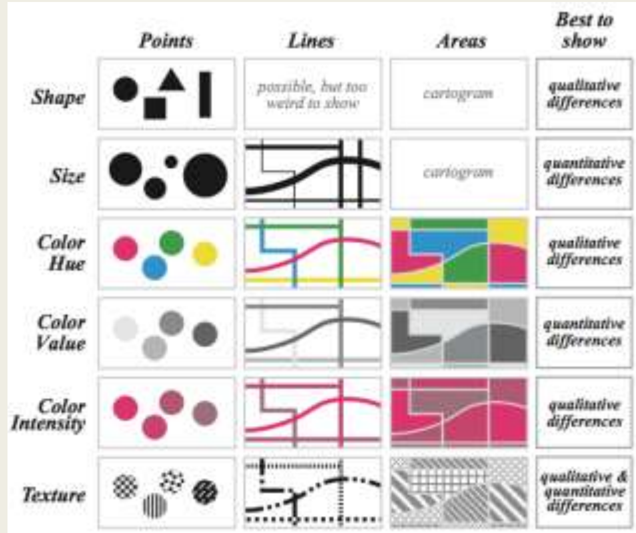
MacEachren, A.M. (1995). How maps work: Representation, visualization, and design. Guilford.


Bertin, Richards, MacEachren, Blackwell&Engelhardt and Engelhardt.

One approach is to take a holistic perspective on visual language, information design, notations, or diagrams. Specialist research communities in these fields address many relevant factors from low-level visual perception to critique of visual culture. Across all of them, it can be necessary to ignore (or not be distracted by) technical and marketing claims, and to remember that all visual representations simply comprise marks on a surface that are intended to correspond to things understood by the reader. The two dimensions of the surface can be made to correspond to physical space (in a map), to dimensions of an object, to a pictorial perspective, or to continuous abstract scales (time or quantity). The surface can also be partitioned into regions that should be interpreted differently. Within any region, elements can be aligned, grouped, connected or contained in order to express their relationships. In each case, the correspondence between that arrangement, and the intended interpretation, must be understood by convention or explained. Finally, any individual element might be assigned meaning according to many different semiotic principles of correspondence.

Jacques Bertin, Semiology of Graphics, 1967

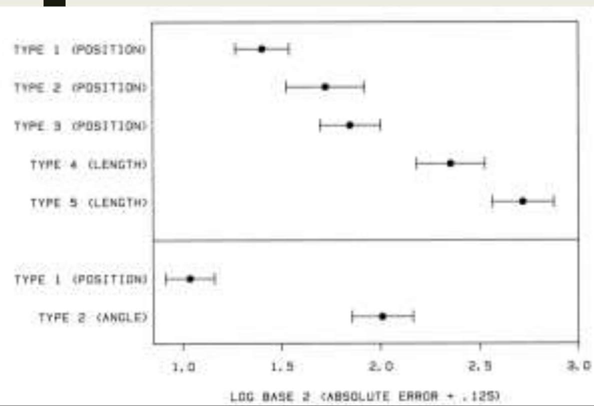





Graphic resources
"Planar dimensions"
Retinal variables

Cleveland, W. S.,&McGill, R. (1984). Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods.*Journal of the American Statistical Association*,*79*(387), 531–554. https://doi.org/10.2307/2288400

Heer, J.,&Bostock, M. (2010). Crowdsourcing graphical perception: using {Mechanical Turk} to assess visualisation design.*ACM Human Factors in Computing Systems (CHI)*, 203–212.



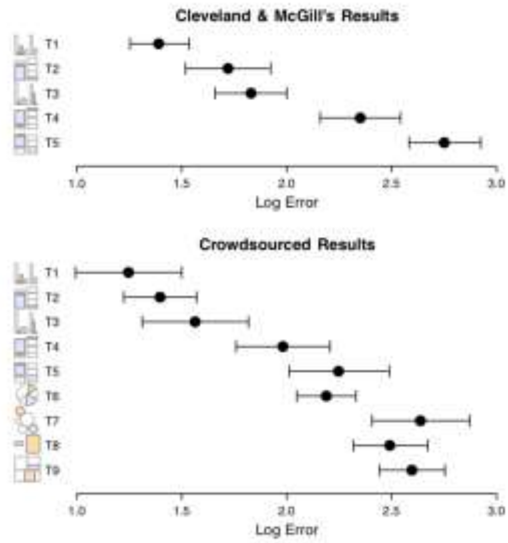




Figure 4: Proportional judgment results (Exp. 1A & B). Top: Cleveland & McGill's [7] lab study. Bottom: MTurk studies. Error bars indicate 95% confidence intervals.

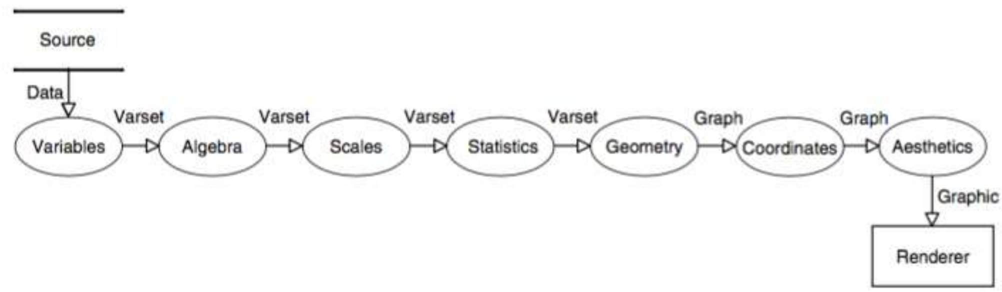
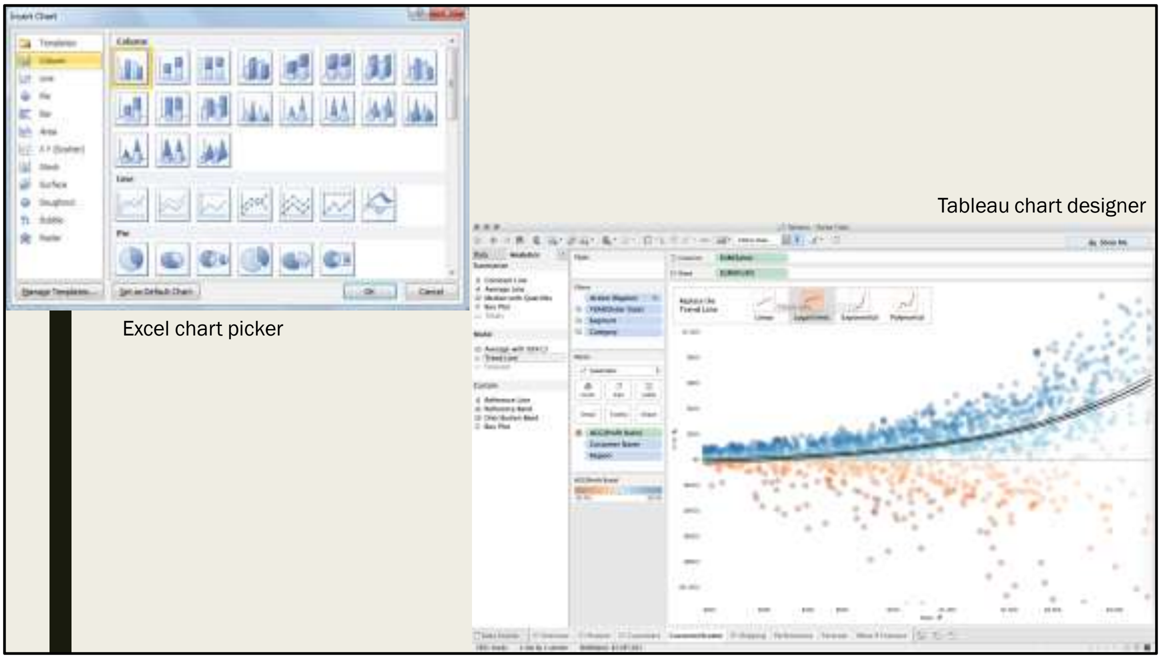



**FIGURE 1** | The grammar of graphics data flow.

Leland Wilkinson, The Grammar of Graphics, 1999
Later extended by Hadley Wickham

Take a framework like this and formally encode it.

The grammar of graphics was the foundation for the [R](#) package [ggplot2](#)




Grammar of graphics is great for people who think about visualisation in such rareified planes of abstraction, but it is not really suited to the mental models and expertise of most end-users. So we have simplified alternatives such as the Excel chart picker, which reframe the pipeline in terms of concrete examples. This is perhaps limiting in terms of the types of visualisations you can achieve, but is vastly more usable. Another point in the spectrum is Tableau’s chart designer. This came out of Christopher Stolte’s PhD work at Stanford in the late 90s, early 2000s.

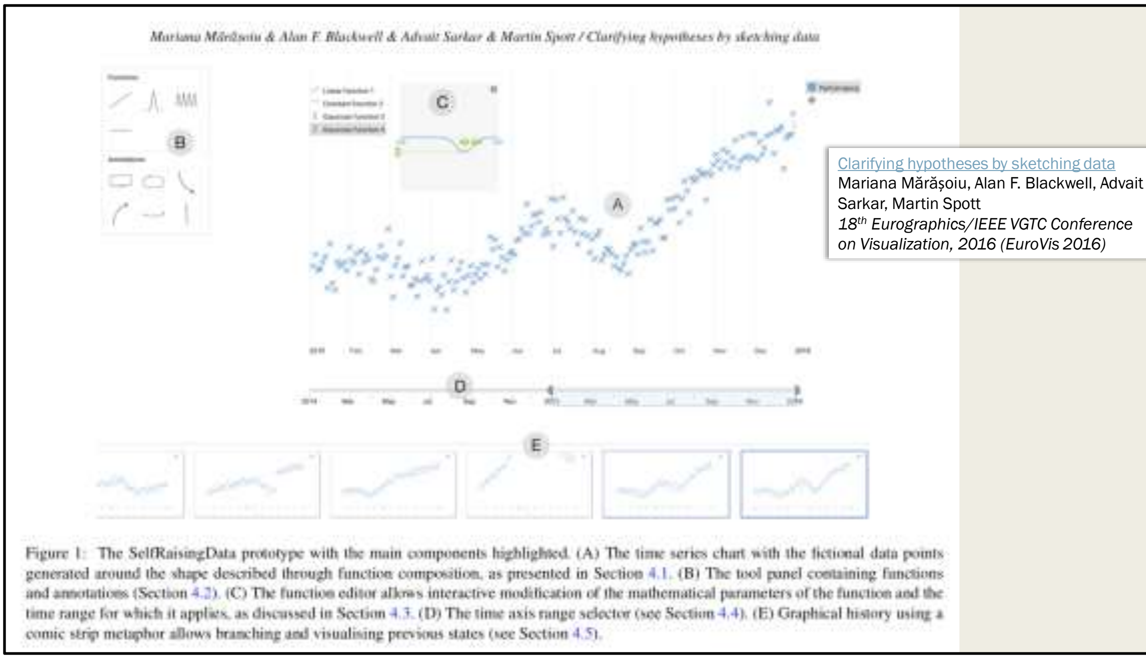

Mariana Mărăşoiu & Alan F. Blackwell & Advait Sarkar & Martin Spott / Clarifying hypotheses by sketching data

Figure 1: The SelfRaisingData prototype with the main components highlighted. (A) The time series chart with the fictional data points generated around the shape described through function composition, as presented in Section 4.1. (B) The tool panel containing functions and annotations (Section 4.2). (C) The function editor allows interactive modification of the mathematical parameters of the function and the time range for which it applies, as discussed in Section 4.3. (D) The time axis range selector (see Section 4.4). (E) Graphical history using a comic strip metaphor allows branching and visualising previous states (see Section 4.5).

Clarifying hypotheses by sketching data
Mariana Mărăşoiu, Alan F. Blackwell, Advait Sarkar, Martin Spott
*18th Eurographics/IEEE VGTC Conference on Visualization, 2016 (EuroVis 2016)*

The directionality of data -> visualisation in the grammar of graphics can also be limiting. What about visualisation -> data?

# Principles of visualisation

- Structural: e.g., Bertin, Wilkinson/Wickham
- Perceptual/cognitive: e.g., Bertin, Cleveland & McGill
- Aesthetic/designerly: e.g., Edward Tufte (Visual Display of Quantitative Information)

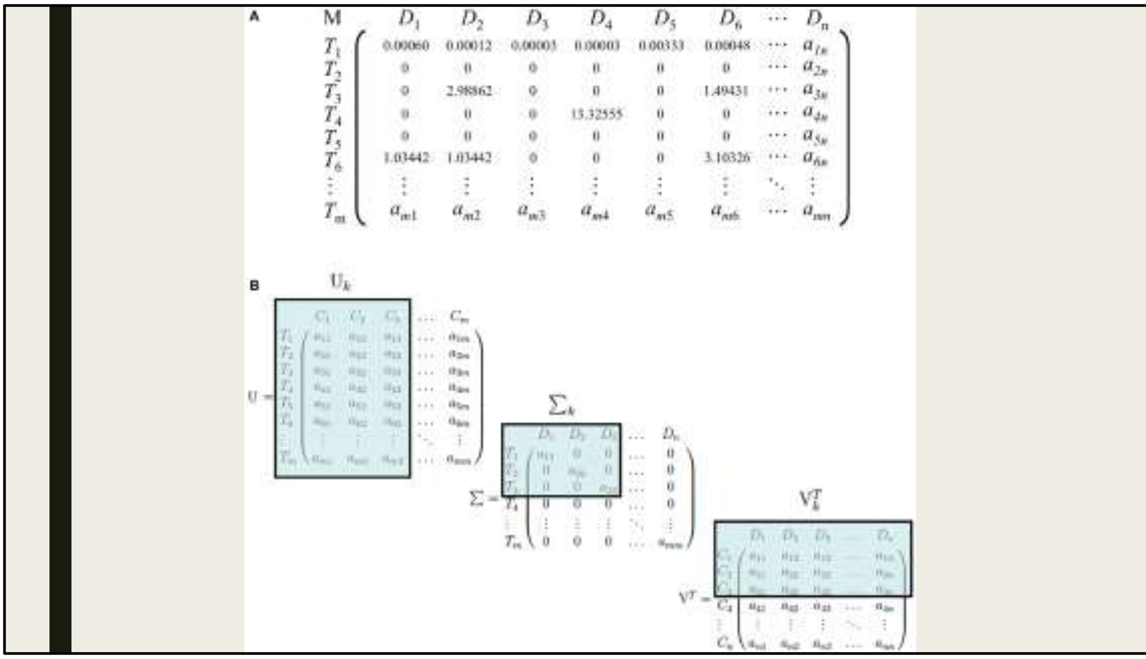
# Interaction and visualisation

- Shneiderman’s mantra: Overview, zoom, filter, detail-on-demand
- Yi et al (2007): Yi, J. S., Kang, Y.-A., Stasko, J.,&Jacko, J. (2007). Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*,*13*(6), 1224–31. https://doi.org/10.1109/TVCG.2007.70515
- Lam, H (2008): Lam, H. (2008). A framework of interaction costs in information visualization.*IEEE Transactions on Visualization and Computer Graphics*,*14*(6), 1149–56. https://doi.org/10.1109/TVCG.2008.109


Yi, J. S., Kang, Y.-A., Stasko, J.,&Jacko, J. (2007). Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*,*13*(6), 1224–31. https://doi.org/10.1109/TVCG.2007.70515

Lam, H. (2008). A framework of interaction costs in information visualization.*IEEE Transactions on Visualization and Computer Graphics*,*14*(6), 1149–56. https://doi.org/10.1109/TVCG.2008.109

# LATENT SEMANTIC ANALYSIS



An example of a term-document matrix with a weighting function (tf-idf). M, D, and T refer to the term-document matrix, the set of all documents in the corpus, and the set of all terms in the corpus, respectively. $T_1$ is an example of a common word that occurs frequently in documents, whereas $T_3$, $T_4$, and $T_6$ are comparatively rarer words and receive a higher weight. **(B)** An illustration of the dimensionality-reduction step of LSI. U, Σ, and $V^T$ are truncated and become $\Sigma_k$, $U_k$, and $V^T{}_k$, respectively. C, D, and T refer to the set of LSI topics, documents, and terms, respectively. Here, we illustrate a reduction to three dimensions.

These matrices can then be used as a distance metric for both terms and documents. Any two documents can be compared by computing the cosine distance between their corresponding column vectors in $V^T$. Likewise, any two terms can be compared by computing the cosine distance between their corresponding row rectors in U.

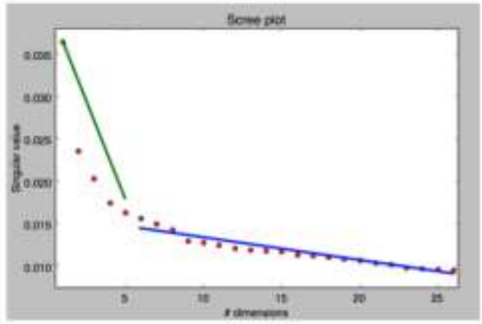
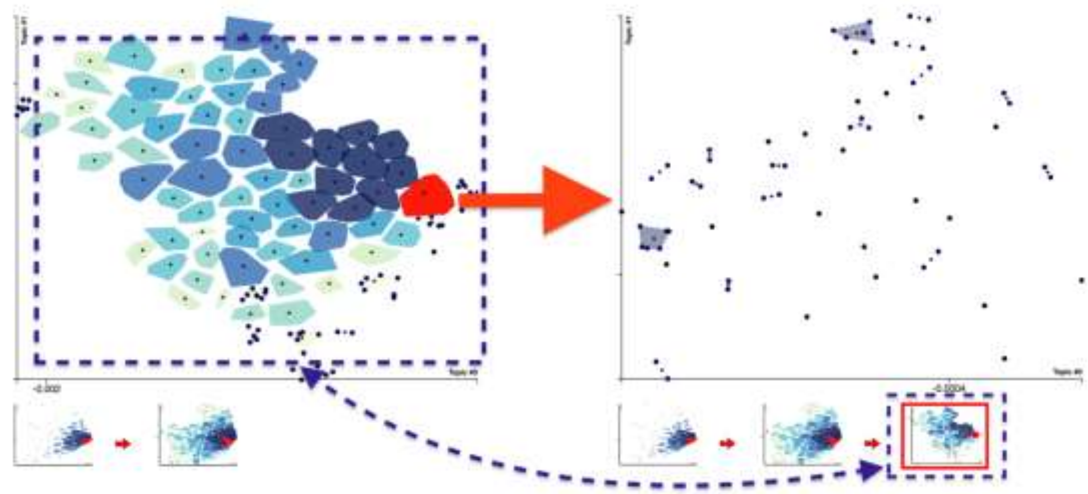



Figure 3.1: Singular value scree plot with a knee found by L-method at the $5^{th}$ singular value



Overview + detail, Semantic zooming, Graphical interaction histories

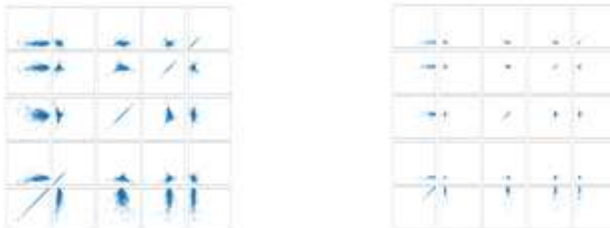


Figure 3.9: Random sampling performed in each scatt
very dense areas and a number of potentially interestin
and the shape is distorted. Sampling more values in w
performance.

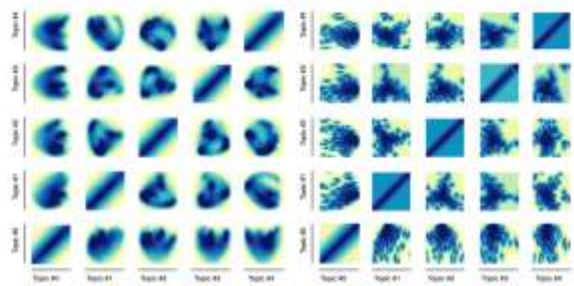


Figure 3.10: Two examples of heat map matrices. The colour scale ranging from light yellow to dark blue indicates the estimated probability density of the data distribution. Blue areas indicate higher probabilities of data points at that position.

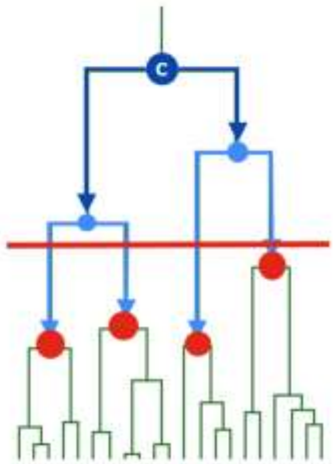



Figure 3.15: Obtaining the expansion of a cluster. To determine which clusters would become $C$'s children in the expansion tree, a cut (in red) is made at the height corresponding to the minimum displayable distance between clusters. $C$'s children are then expanded until the clusters immediately below the cut are reached; these are then chosen as $C$'s expansion.

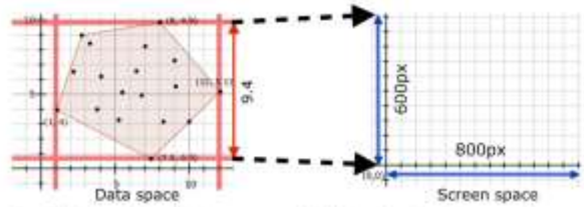



Figure 3.16: Mapping from data to screen space. The cluster shown is a cluster we want to expand and will be fragmented into its descendant clusters. By knowing the extent on one dimension in data space and the size of the y-axis in screen space, we can obtain a linear mapping between the two spaces. We can do the same for the other data dimension and x-axis.

# BRAINCEL

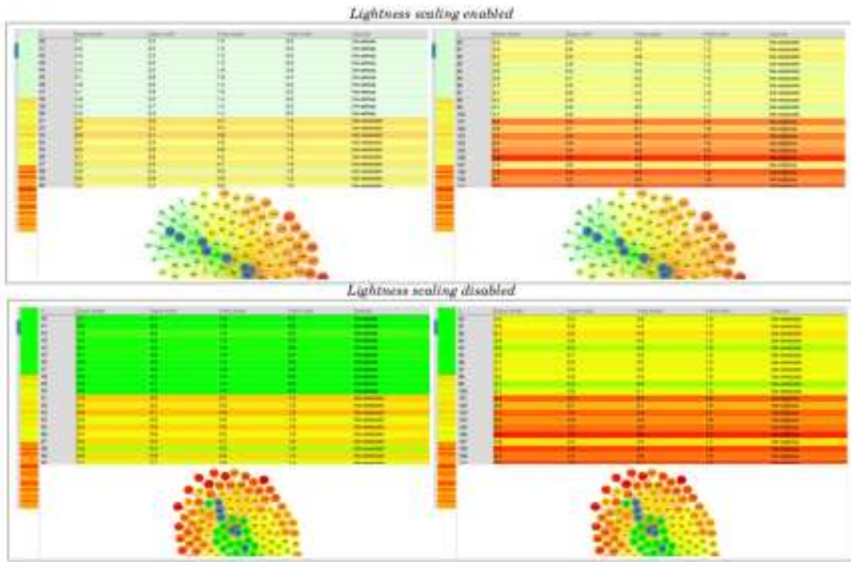
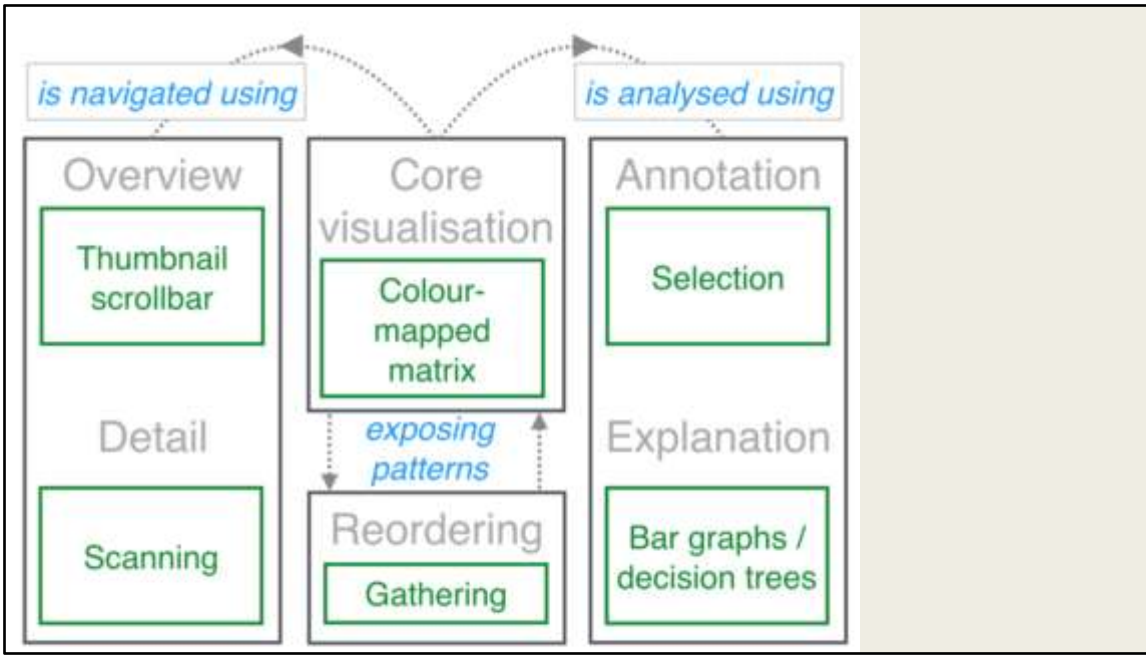
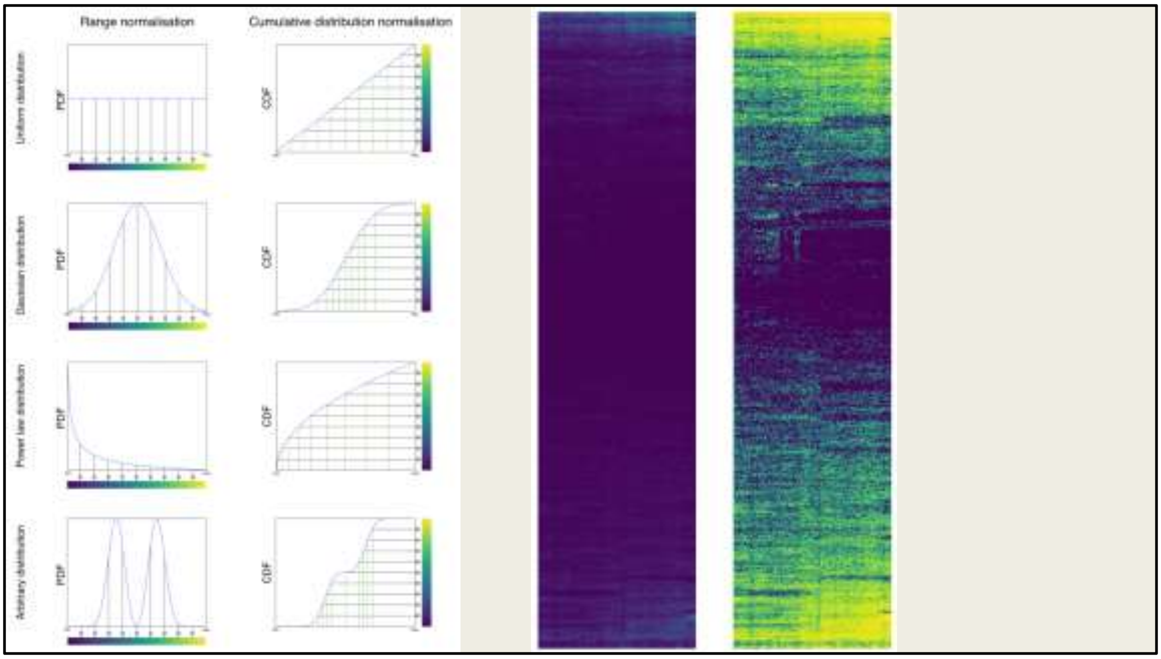



**Figure 5.5:** The effect of lightness scaling. Without lightness scaling, high-confidence (green) rows command disproportionately greater visual attention (the effect is most apparent onscreen).

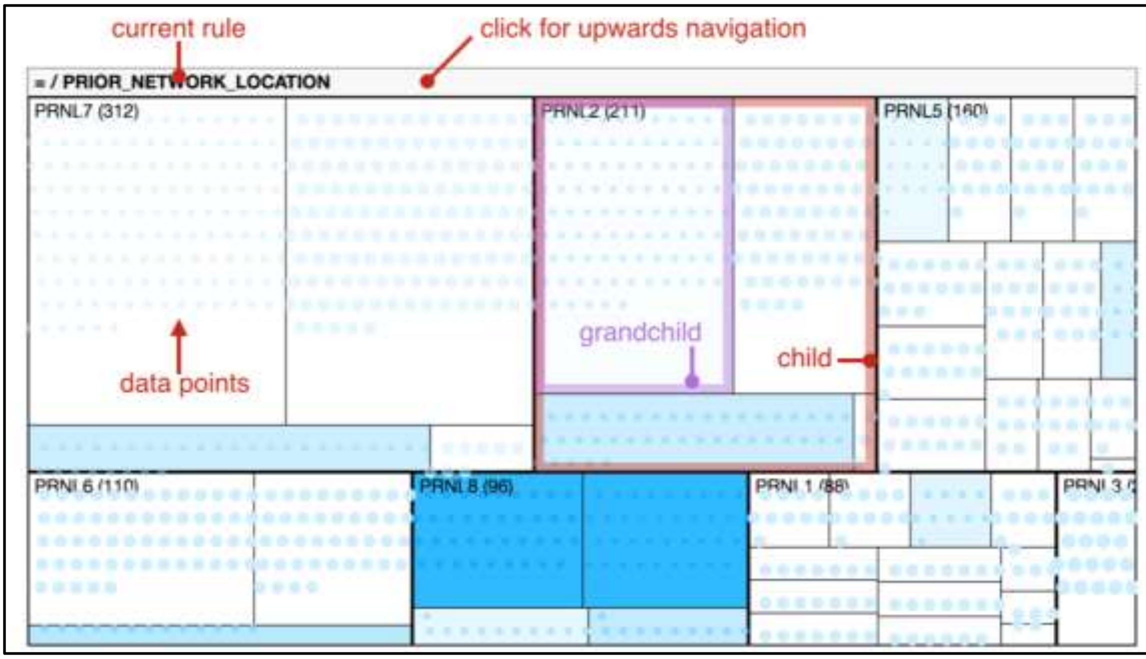
# GATHERMINER



is navigated using
is analysed using
Overview
Thumbnail scrollbar
Detail
Scanning
Core visualisation
Colour-mapped matrix
exposing patterns
Reordering
Gathering
Annotation
Selection
Explanation
Bar graphs / decision trees



Range normalisation
Cumulative distribution normalisation
Uniform distribution
Gaussian distribution
Power law distribution
Arbitrary distribution
PDF
CDF


current rule
click for upwards navigation
≡ / PRIOR_NETWORK_LOCATION
PRNL7 (312)
PRNL2 (211)
PRNL5 (160)
grandchild
child
data points
PRNL6 (110)
PRNL8 (96)
PRNL1 (88)

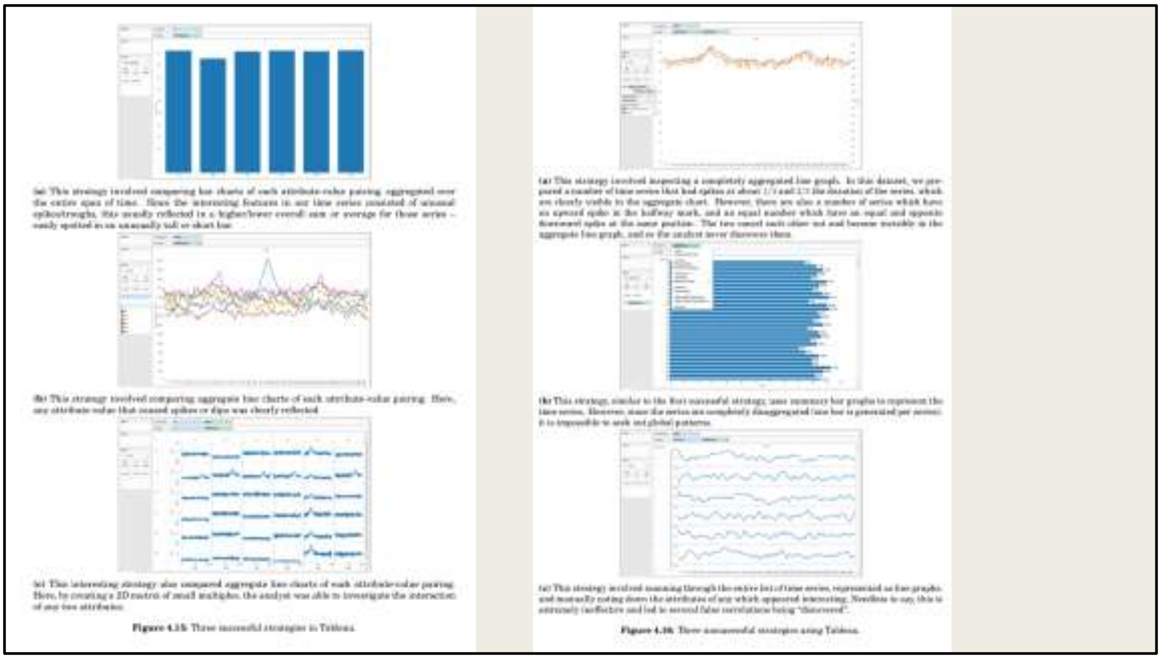


Figure 4.15: Three successful strategies in Tableau.
Figure 4.16: Three unsuccessful strategies using Tableau.