

Machine Learning for Research

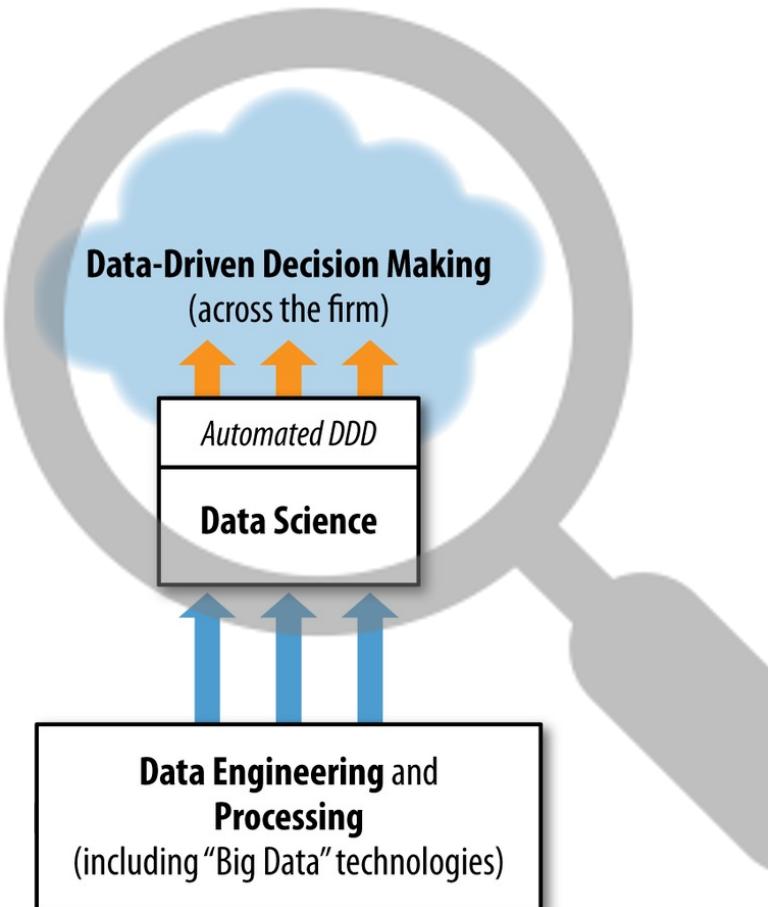
Prof. Dr. Oliver Müller

oliver.mueller@upb.de; @mueller_oli

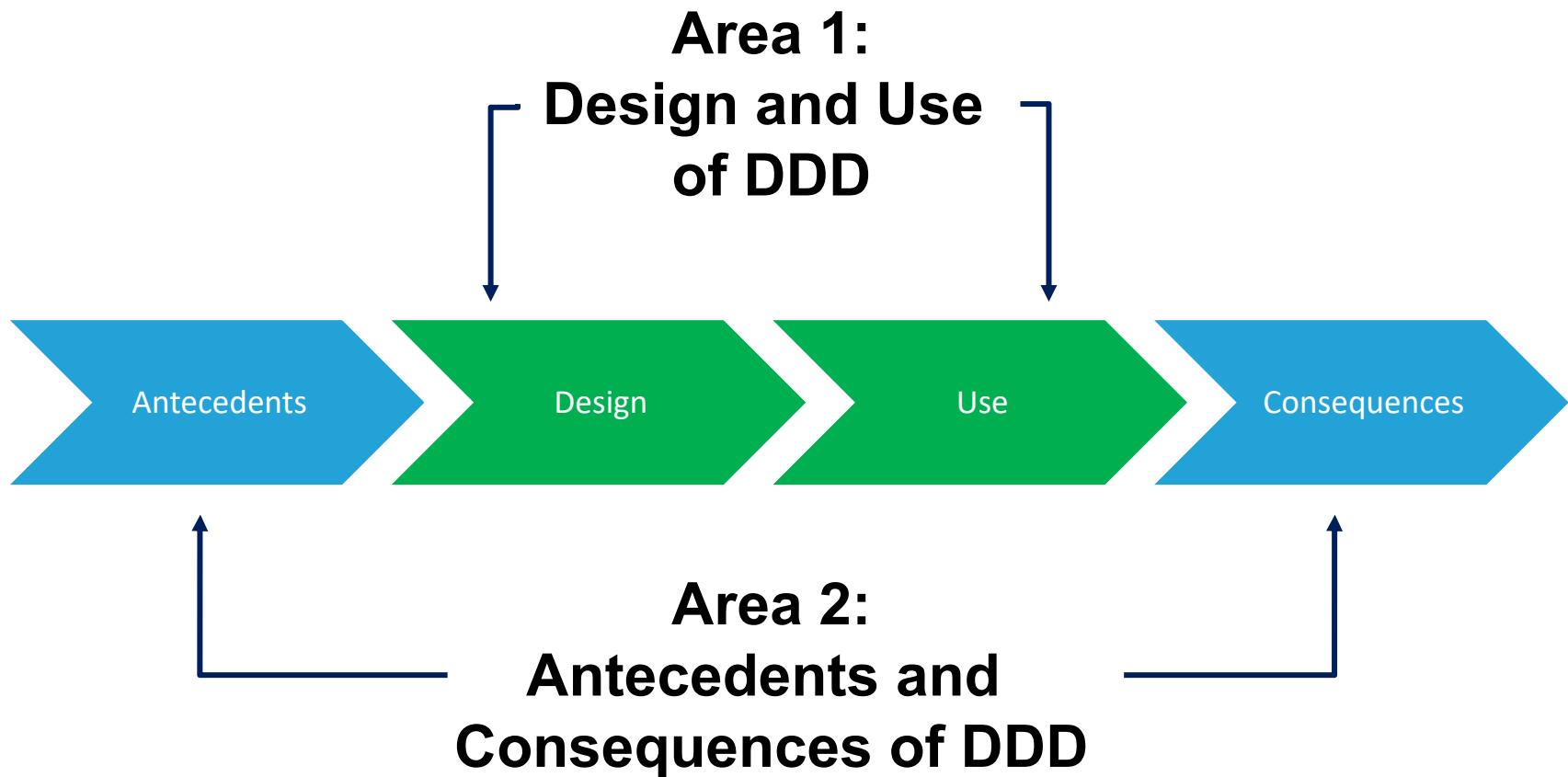
About Me

Chair of Information Systems, esp. Data Analytics

- **Data-driven decision making (DDD)** refers to organizational decision-making practices that emphasize the use of data and statistical analysis instead of relying on human judgment only. (Brynjolfsson et al. 2011; Foster & Provost 2013)
- Theoretical foundations
 - Statistical learning theory
 - Decision theory
 - Behavioural economics, Judgment and decision making
 - Human-Computer Interaction
 - Psychology
 - ...

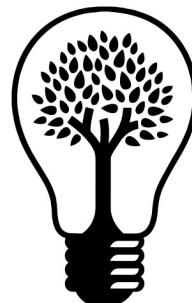


Research Topics



About Me

Selected Research Projects



Supply and demand of/for sustainable energy (industry and residential), Maintenance of electricity grids

Retirement saving, Corporate credit risk, Real estate appraisal, Stock return volatility

Player market values, Scouting, On-pitch performance, Set-piece tactics



**Who are you?
What are your expectations?**

The Course

Syllabus

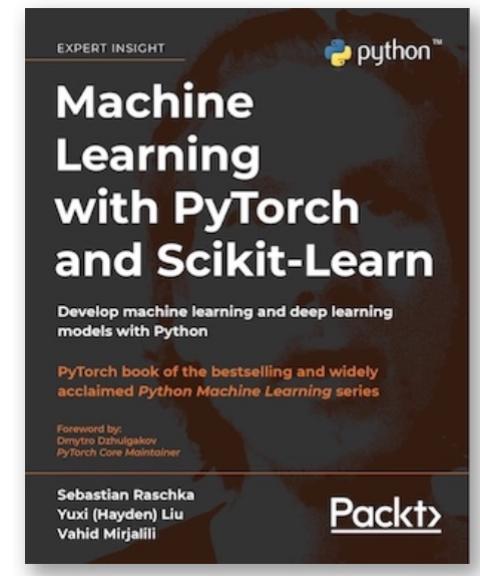
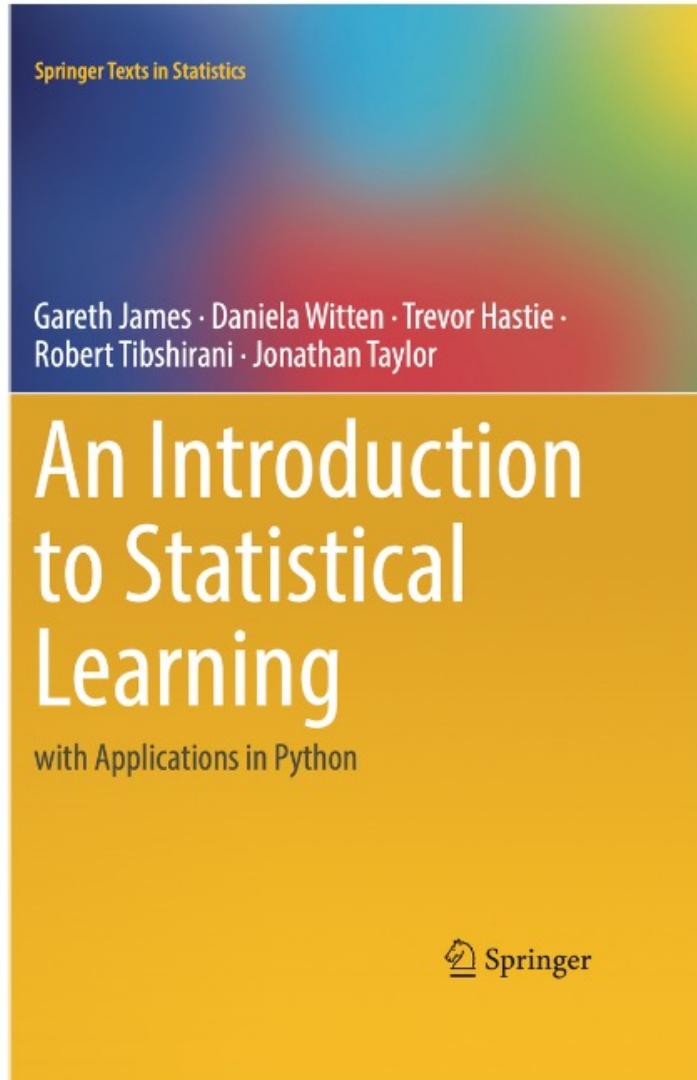
- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Philosophy

- Machine learning should not be viewed as a series of black boxes.
- However, while it is important to know what job is performed by each cog, it is not necessary to have the skills to construct the machine inside the box!
- One has to understand the simpler methods first, in order to grasp the more sophisticated ones.

The Course

Textbooks



Papers

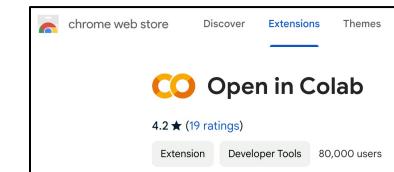
- Michel, J. B., Shen, Y. K., Aiden, A. P., Veres, A., Gray, M. K., Pickett, J. P., ... & Pinker, S. (2011). Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014), 176-182.
- Ginsberg, J., Mohebbi, M. H., Patel, R. S., Brammer, L., Smolinski, M. S., & Brilliant, L. (2009). Detecting influenza epidemics using search engine query data. *Nature*, 457(7232), 1012-1014.
- Lazer, D., Kennedy, R., King, G., & Vespignani, A. (2014). The parable of Google Flu: traps in big data analysis. *Science*, 343(6176), 1203-1205.
- Debortoli, S., Junglas, I., Müller, O., & vom Brocke, J. (2016). Text Mining for Information Systems Researchers: An Annotated Topic Modeling Tutorial. *Communications of the Association for Information Systems*, 39(1), 7.
- Müller, O., Junglas, I., vom Brocke, J., & Debortoli, S. (2016). Utilizing big data analytics for information systems research: challenges, promises and guidelines. *European Journal of Information Systems*, 25(4), 289-302.
- Schmiedel, T., Müller, O., & vom Brocke, J. (2019). Topic modeling as a strategy of inquiry in organizational research: A tutorial with an application example on organizational culture. *Organizational Research Methods*, 22(4), 941-968.
- ...

The Course

Tools



colab



The Course

Syllabus

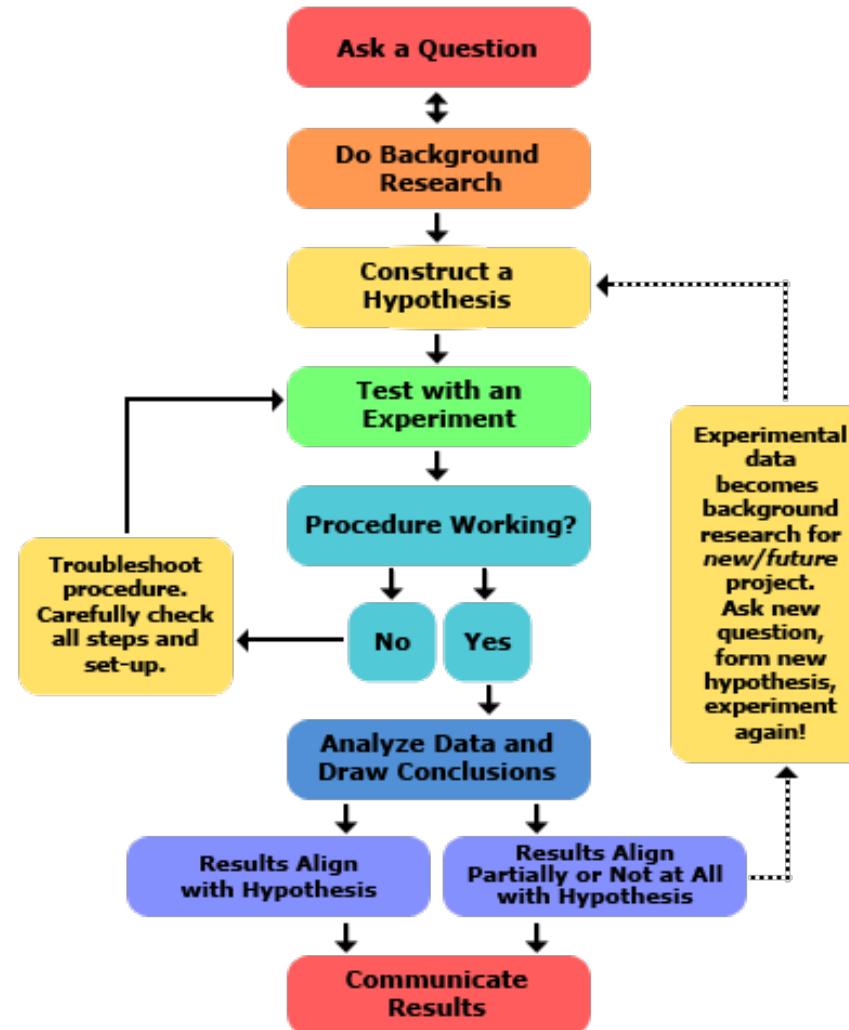
- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Introduction to Data Science:
_ Data Science vs. The Scientific Method

Data Science vs. The Scientific Method

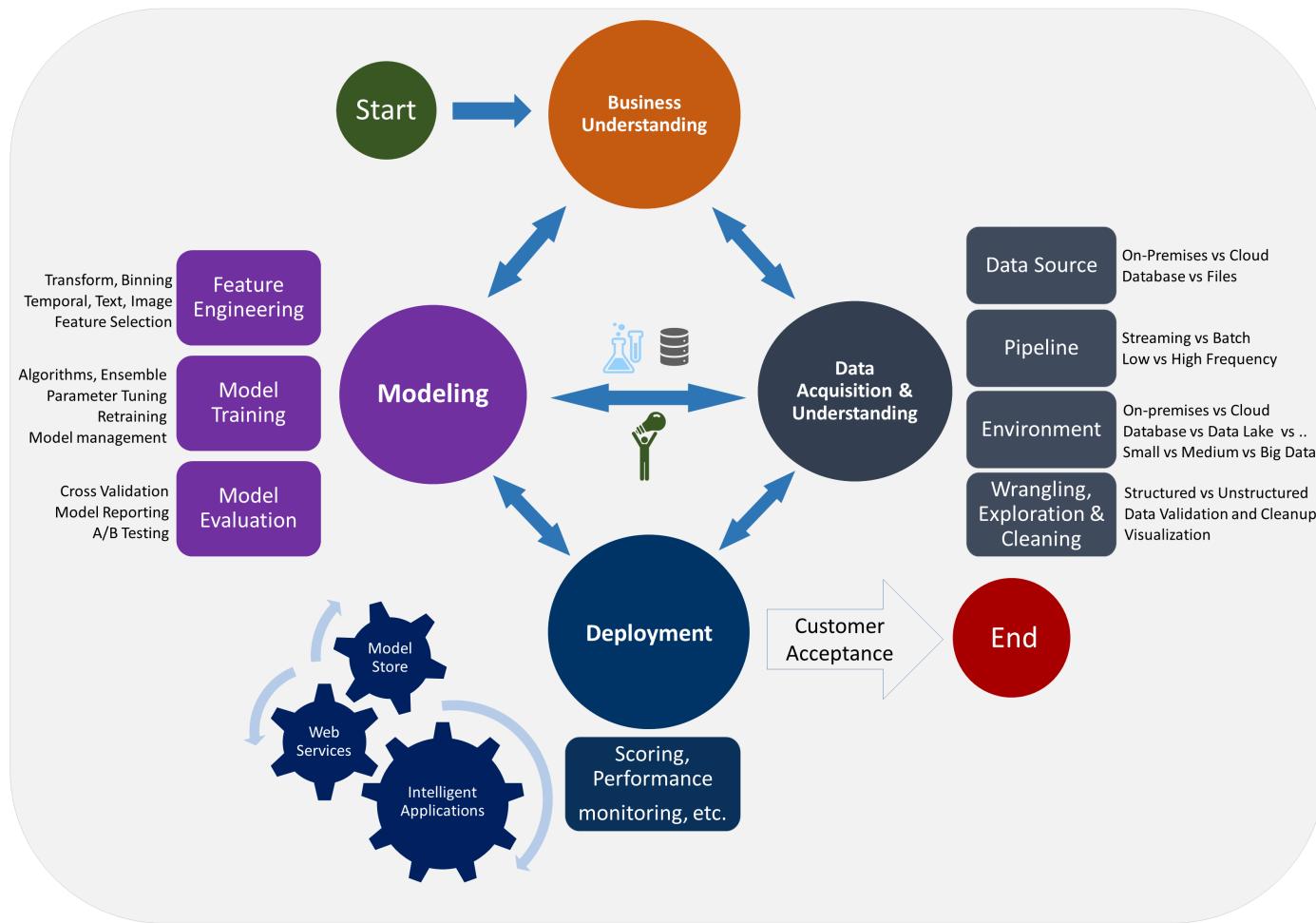
The Scientific Method: Hypothetico-deductive Model



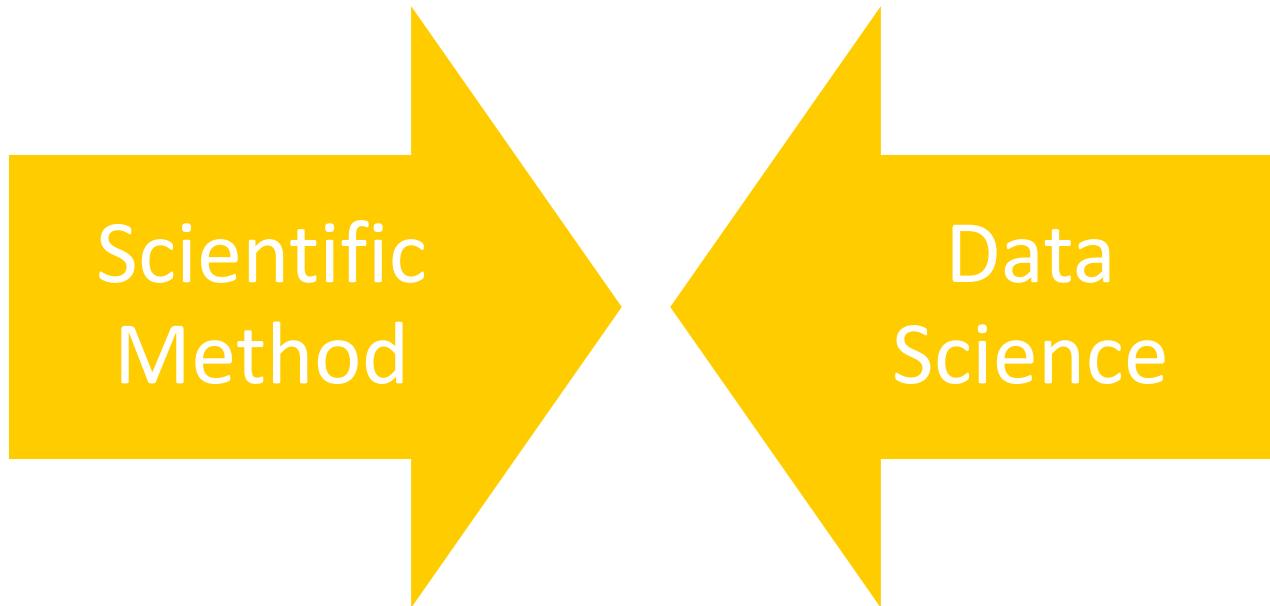
Data Science vs. The Scientific Method

Team Data Science Process (TDSP)

Data Science Lifecycle



Data Science vs. The Scientific Method



Data Science vs. The Scientific Method



Watch Read Attend Participate About

Search...



Log in Sign up

Jean-Baptiste Michel, Erez Lieberman Aiden:

What we learned from 5 million books



TEDxBoston 2011 · 14:08 · Posted Sep 2011

Subtitles available in 37 languages

View transcript

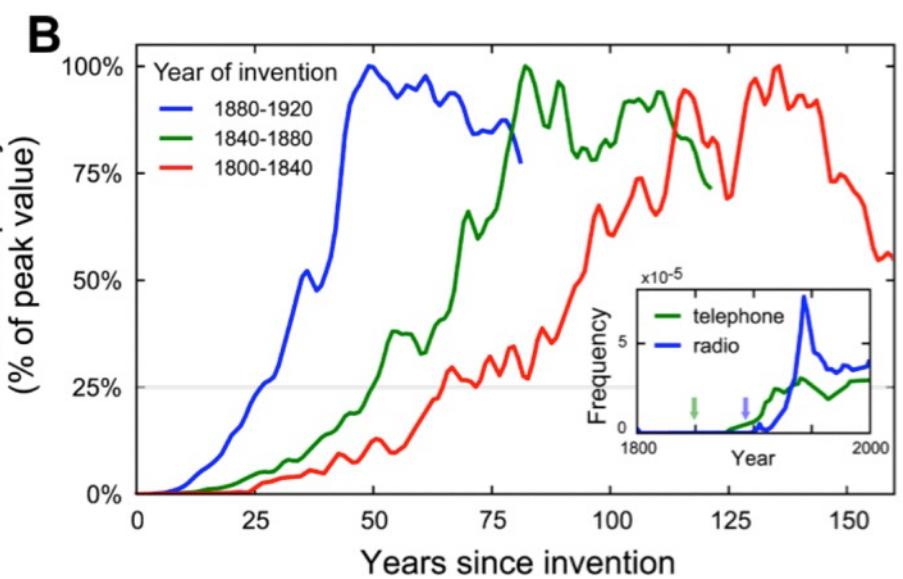
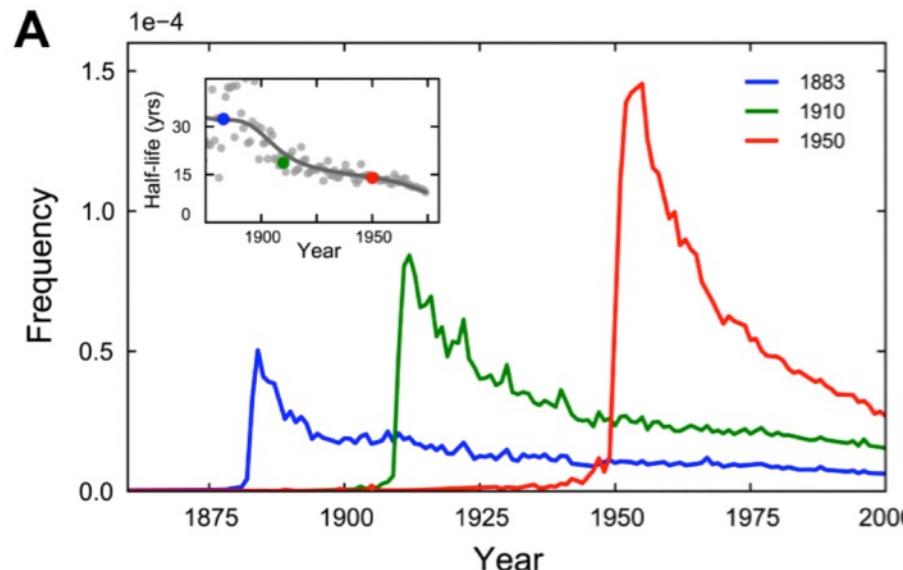
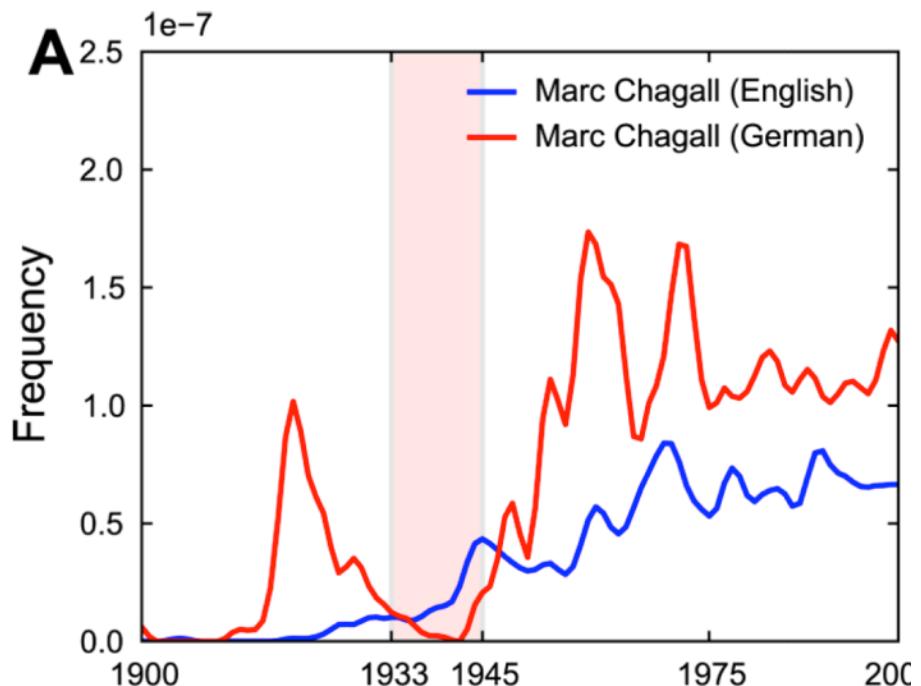


Rate

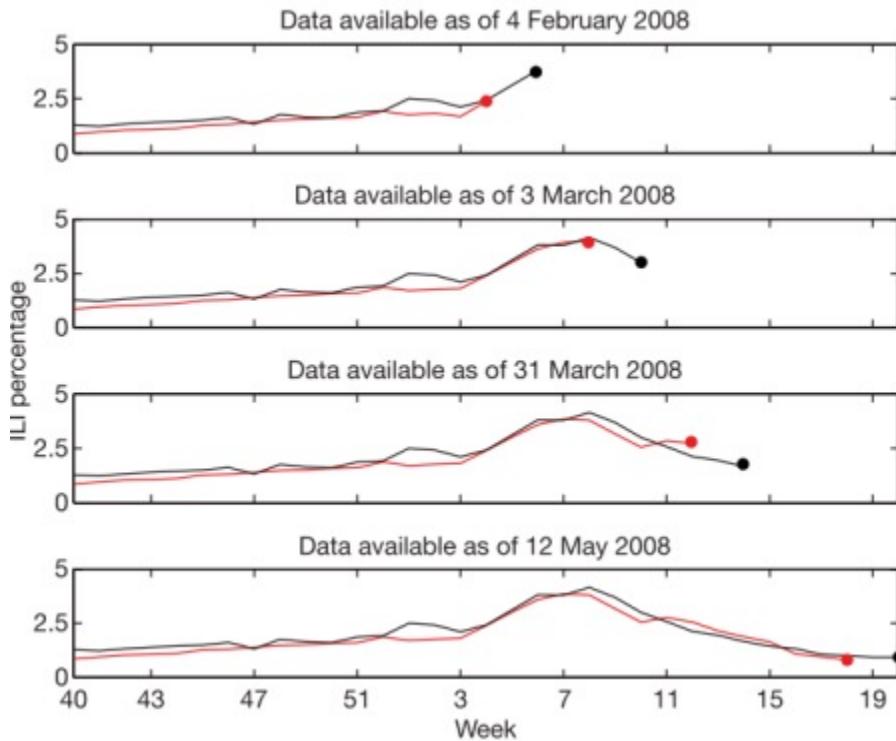


1,278,448 TOTAL VIEWS

Data Science vs. The Scientific Method



Data Science vs. The Scientific Method



nature
International weekly journal of science

"The final model was validated on 42 points per region of previously untested data from 2007 to 2008, which were excluded from all previous steps. Estimates generated for these 42 points obtained a **mean correlation of 0.97** (min: 0.92, max: 0.99, n: 9 regions) with the CDC-observed ILI percentages.

Harnessing the collective intelligence of millions of users, **Google web search logs can provide one of the most timely, broad-reaching influenza monitoring systems available today.**

Data Science vs. The Scientific Method



Thank you for stopping by.

Google Flu Trends and Google Dengue Trends are [no longer publishing](#) current estimates of Flu and Dengue fever based on search patterns. The historic estimates produced by Google Flu Trends and Google Dengue Trends are available below. It is still early days for nowcasting and similar tools for understanding the spread of diseases like flu and dengue – we're excited to see what comes next. Academic research groups interested in working with us should fill out this [form](#).

Sincerely,

The Google Flu and Dengue Trends Team.

Data Science vs. The Scientific Method

FINAL FINAL || **POLICYFORUM** ||

BIG DATA

The Parable of Google Flu: Traps in Big Data Analysis

David Lazer,^{1,2*} Ryan Kennedy,^{1,3,4} Gary King,² Alessandro Vespignani^{1,5,6}

In February 2013, Google Flu Trends (GFT) made headlines but not for a reason that Google executives or the creators of the flu tracking system would have hoped. *Nature* reported that GFT was predicting more than double the proportion of doctor visits for influenza-like illness (ILI) than the Centers for Disease Control and Prevention (CDC), which bases its estimates on surveillance reports from laboratories across the United States (*1, 2*). This happened despite the fact that GFT was built to predict CDC reports. Given that GFT is often held up as an exemplary use of big data (*3, 4*), what lessons can we draw from this error?

The problems we identify are not limited to GFT. Research on whether search or social media can predict *x* has become commonplace (*5–7*) and is often put in sharp contrast with traditional methods and hypotheses. Although these studies have shown the value of these data, we are far from a place where they can supplant more traditional methods or theories (*8*). We explore two issues that contributed to GFT's mistakes—big data hubris and algorithm dynamics—and offer lessons for moving forward in the big data age.

Big Data Hubris

"Big data hubris" is the often implicit assumption that big data are a substitute for, rather than a supplement to, traditional data collection and analysis. Elsewhere, we have asserted that there are enormous scientific possibilities in big data (*9–11*). However, quantity of data does not mean that one can ignore foundational issues of measurement and construct validity and reliability and dependencies among data (*12*). The core challenge is that most big data that have received popular attention are not the output of instruments designed to produce valid and reliable data amenable for scientific analysis.

The initial version of GFT was a particularly problematic marriage of big and small data. Essentially, the methodology was to find the best matches among 50 million search terms to fit 1152 data points (*13*). The odds of finding search terms that match the propensity of the flu but are structurally unrelated, and so do not predict the future, were quite high. GFT developers, in fact, report weeding out seasonal search terms unrelated to the flu but strongly correlated to the CDC data, such as those regarding high school basketball (*13*). This should have been a warning that the big data were overfitting the small number of cases—a standard concern in data analysis. The ad hoc method of throwing out peculiar search terms failed when GFT completely missed the nonseasonal 2009 influenza A–H1N1 pandemic (*2, 14*). In short, the initial version of GFT was part flu detector, part winter detector. GFT engineers updated the algorithm in 2009, and this model has run ever since, with a few changes announced in October 2013 (*10, 15*).

Although not widely reported until 2013, the new GFT has been persistently overestimating flu prevalence for a much longer time. GFT also missed by a very large margin in the 2011–2012 flu season and has missed high for 100 out of 108 weeks starting with August 2011 (see the graph). These errors are not randomly distributed. For example, last week's errors predict this week's errors (temporal autocorrelation), and the direction and magnitude of error varies with the time of year (seasonality). These patterns mean that GFT overlooks considerable information that could be extracted by traditional statistical methods.

Even after GFT was updated in 2009, the comparative value of the algorithm as a stand-alone flu monitor is questionable. A study in 2010 demonstrated that GFT accuracy was not much better than a fairly simple projection forward using already available (typically on a 2-week lag) CDC data (*4*). The comparison has become even worse since that time, with lagged models significantly outperforming GFT (see the graph). Even 3-week-old CDC data do a better job of projecting current flu prevalence than GFT [see supplementary materials (SM)].

Considering the large number of approaches that provide inference on influenza activity (*16–19*), does this mean that the current version of GFT is not useful? No, greater value can be obtained by combining GFT with other near-real-time health data (*2, 20*). For example, by combining GFT and lagged CDC data, as well as dynamically recalibrating GFT, we can substantially improve on the performance of GFT or the CDC alone (see the chart). This is no substitute for ongoing evaluation and improvement, but, by incorporating this information, GFT could have largely healed itself and would have likely remained out of the headlines.

*Corresponding author. E-mail: d.lazer@neu.edu.

CREDIT: ADAPTED FROM AXEL KORNBERGER & DIRK HUHN/STOCKPHOTO.COM

¹Lazer Laboratory, Northeastern University, Boston, MA 02115, USA; ²Institute for Quantitative Social Science, Harvard University, Cambridge, MA 02138, USA; ³University of Houston, Houston, TX 77204, USA; ⁴Laboratory for the Modeling of Biological and Sociotechnical Systems, Northeastern University, Boston, MA 02115, USA; ⁵Institute for Scientific Interchange Foundation, Turin, Italy.

www.sciencemag.org SCIENCE VOL 343 14 MARCH 2014

1203

Data Science vs. The Scientific Method

Is this really science, or just engineering?



Data Science vs. The Scientific Method

Perspective

Integrating explanation and prediction in computational social science

<https://doi.org/10.1038/s41586-021-03659-0>

Received: 23 February 2021

Accepted: 20 May 2021

Published online: 30 June 2021



Jake M. Hofman^{1,17}, Duncan J. Watts^{2,3,4,17}, Susan Athey⁵, Filiz Garip⁶, Thomas L. Griffiths^{7,8}, Jon Kleinberg^{9,10}, Helen Margetts^{11,12}, Sendhil Mullainathan¹³, Matthew J. Salganik⁶, Simine Vazire¹⁴, Alessandro Vespignani¹⁵ & Tal Yarkoni¹⁶

Computational social science is more than just large repositories of digital data and the computational methods needed to construct and analyse them. It also represents a convergence of different fields with different ways of thinking about and doing science. The goal of this Perspective is to provide some clarity around how these approaches differ from one another and to propose how they might be productively integrated. Towards this end we make two contributions. The first is a schema for thinking about research activities along two dimensions—the extent to which work is explanatory, focusing on identifying and estimating causal effects, and the degree of consideration given to testing predictions of outcomes—and how these two priorities can complement, rather than compete with, one another. Our second contribution is to advocate that computational social scientists devote more attention to combining prediction and explanation, which we call integrative modelling, and to outline some practical suggestions for realizing this goal.

Data Science vs. The Scientific Method

Tension in Epistemic Values

- “[S]ocial scientists have traditionally prioritized the formulation of interpretatively satisfying **explanations** of individual and collective human behaviour, often invoking **causal mechanisms derived from substantive theory**.”
- “[C]omputer scientists have traditionally been more concerned with developing **accurate predictive models**, **whether or not they correspond to causal mechanisms or are even interpretable**.”

Data Science vs. The Scientific Method

Table 1 | A schematic for organizing empirical modelling along two dimensions, representing the different levels of emphasis placed on prediction and explanation

	No intervention or distributional changes	Under interventions or distributional changes
Focus on specific features or effects	Quadrant 1: Descriptive modelling Describe situations in the past or present (but neither causal nor predictive)	Quadrant 2: Explanatory modelling Estimate effects of changing a situation (but many effects are small)
Focus on predicting outcomes	Quadrant 3: Predictive modelling Forecast outcomes for similar situations in the future (but can break under changes)	Quadrant 4: Integrative modelling Predict outcomes and estimate effects in as yet unseen situations

The rows highlight where we focus our attention (on either specific features that might affect an outcome of interest, or directly on the outcome itself), whereas the columns specify what types of situations we are modelling (a ‘fixed’ world in which no changes or interventions take place, or one in which features or inputs are actively manipulated or change owing to other uncontrolled forces).

Data Science vs. The Scientific Method

Table 2 | A label scheme for clarifying the nature and granularity of research contributions according to the four quadrants discussed above

Granularity	Quadrant 1		Quadrant 2	Quadrant 3	Quadrant 4
	Describes something	Tests a causal claim		Tests a (passive) predictive claim	Tests a claim both for causality and predictive accuracy
Low	Reports stylized facts	Tests for a non-zero effect		Predicts directional or aggregate outcomes	Predicts directional or aggregate outcomes under changes or interventions
Medium	Reports population averages	Tests for a directional effect		Predicts magnitude and direction of aggregate outcomes	Predicts magnitude and direction of aggregate outcomes under changes or interventions
High	Reports individual outcomes	Estimates the magnitude and direction of an effect		Predicts magnitude and direction of individual outcomes	Predicts magnitude and direction of individual outcomes under changes or interventions

The rows distinguish between different levels of granularity in each quadrant. By 'directional', we mean results that report only whether a given association or effect is positive or negative in sign, whereas by 'magnitude and direction' we mean not only the sign of a relationship but also the numerical size of the correlation or effect.

Data Science vs. The Scientific Method

Table 2. A Taxonomy of Theory Types in Information Systems Research

Theory Type	Distinguishing Attributes
I. Analysis	Says what is. The theory does not extend beyond analysis and description. No causal relationships among phenomena are specified and no predictions are made.
II. Explanation	Says what is, how, why, when, and where. <i>The theory provides explanations but does not aim to predict with any precision.</i> There are no testable propositions.
III. Prediction	Says what is and what will be. <i>The theory provides predictions and has testable propositions but does not have well-developed justificatory causal explanations.</i>
IV. Explanation and prediction (EP)	Says what is, how, why, when, where, and what will be. Provides predictions and has both testable propositions and causal explanations.
V. Design and action	Says how to do something. The theory gives explicit prescriptions (e.g., methods, techniques, principles of form and function) for constructing an artifact.

“The dominance of causal–explanatory statistical modeling and rarity of predictive analytics for theory building and testing exists not only in IS but in the social sciences in general, as well as in other disciplines such as economics and finance.”

(Shmueli & Koppius, 2011, p. 554)



Syllabus

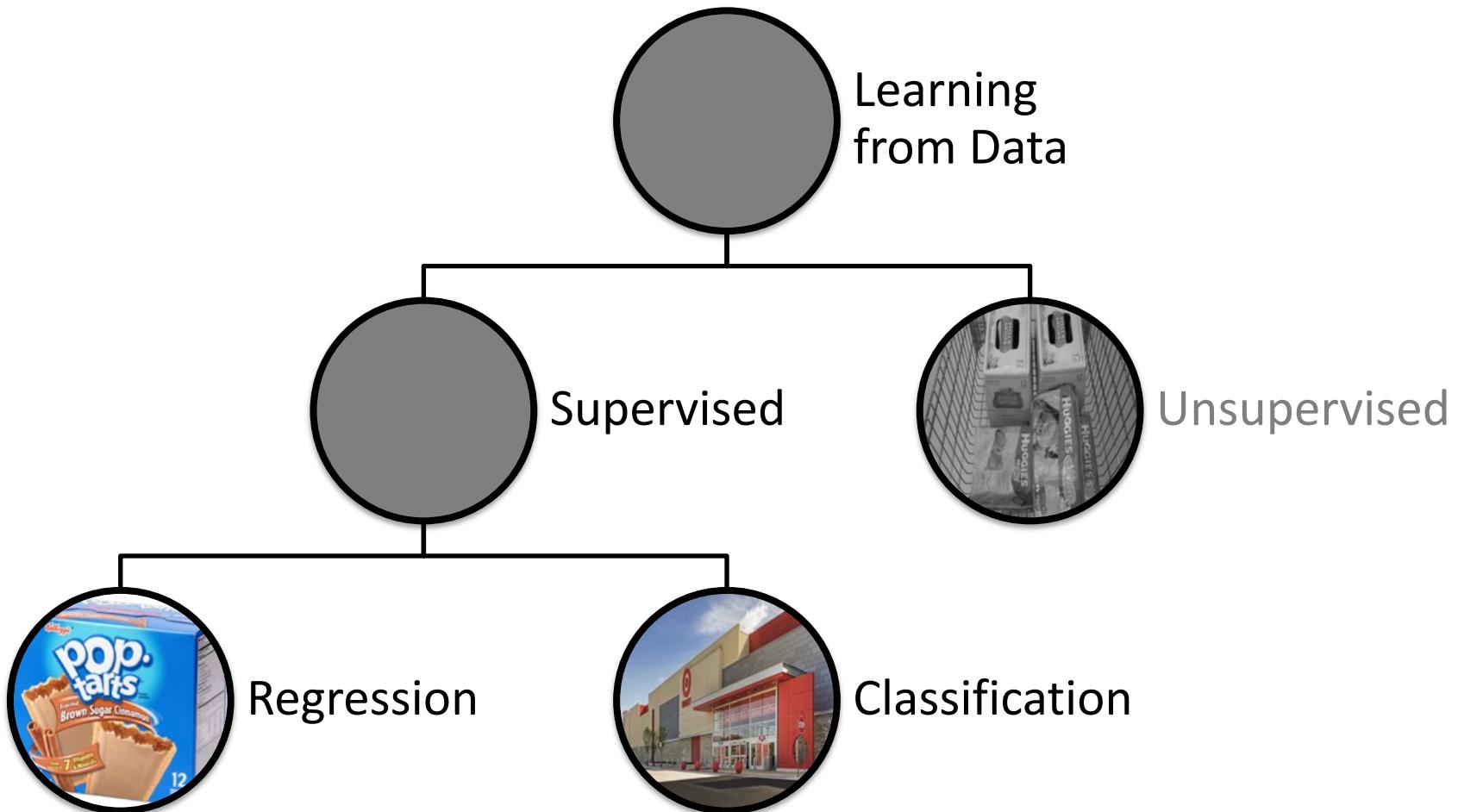
Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Agenda

Supervised Learning

Supervised Learning



Supervised Learning

Variables

- **Input variables**
 - Also known as predictors, features, or independent variables
 - Typically denoted by $X_{1..p}$
- **Output variable**
 - Also known as response, outcome, target, or dependent variable
 - Typically denoted by Y

Modeling the Relationship between X and Y

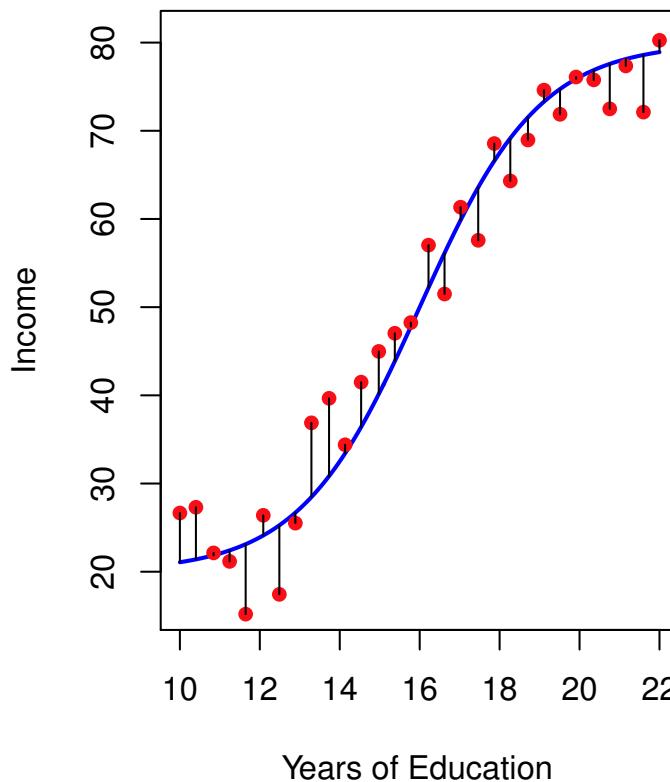
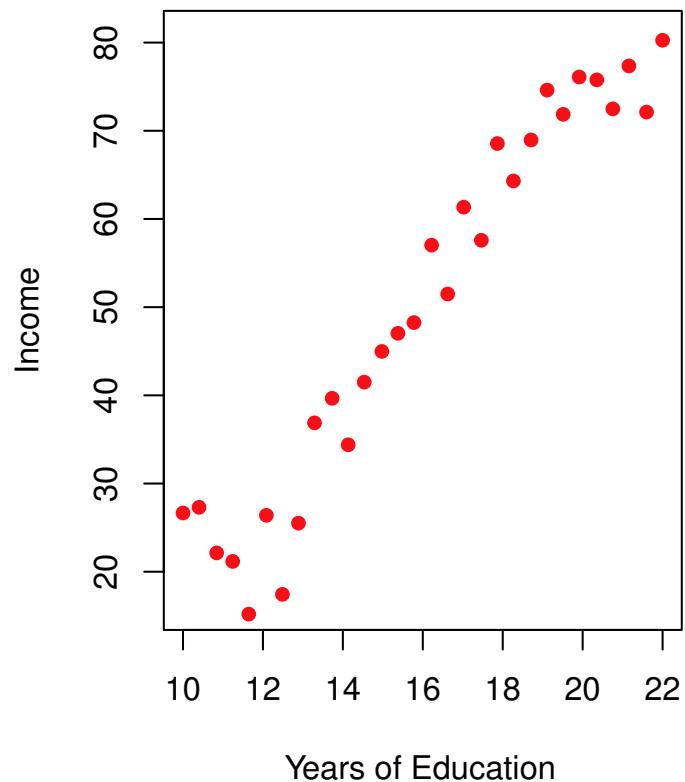
- Suppose that we observe a quantitative response Y and p different predictors, X_1, X_2, \dots, X_p .
- We assume that there is some relationship between Y and $X = (X_1, X_2, \dots, X_p)$, which can be written in the very general form:

$$Y = f(X) + \varepsilon$$

- f is some fixed but unknown function of X_1, X_2, \dots, X_p
- ε is a random error term, which is independent of X and has mean zero.

Supervised Learning

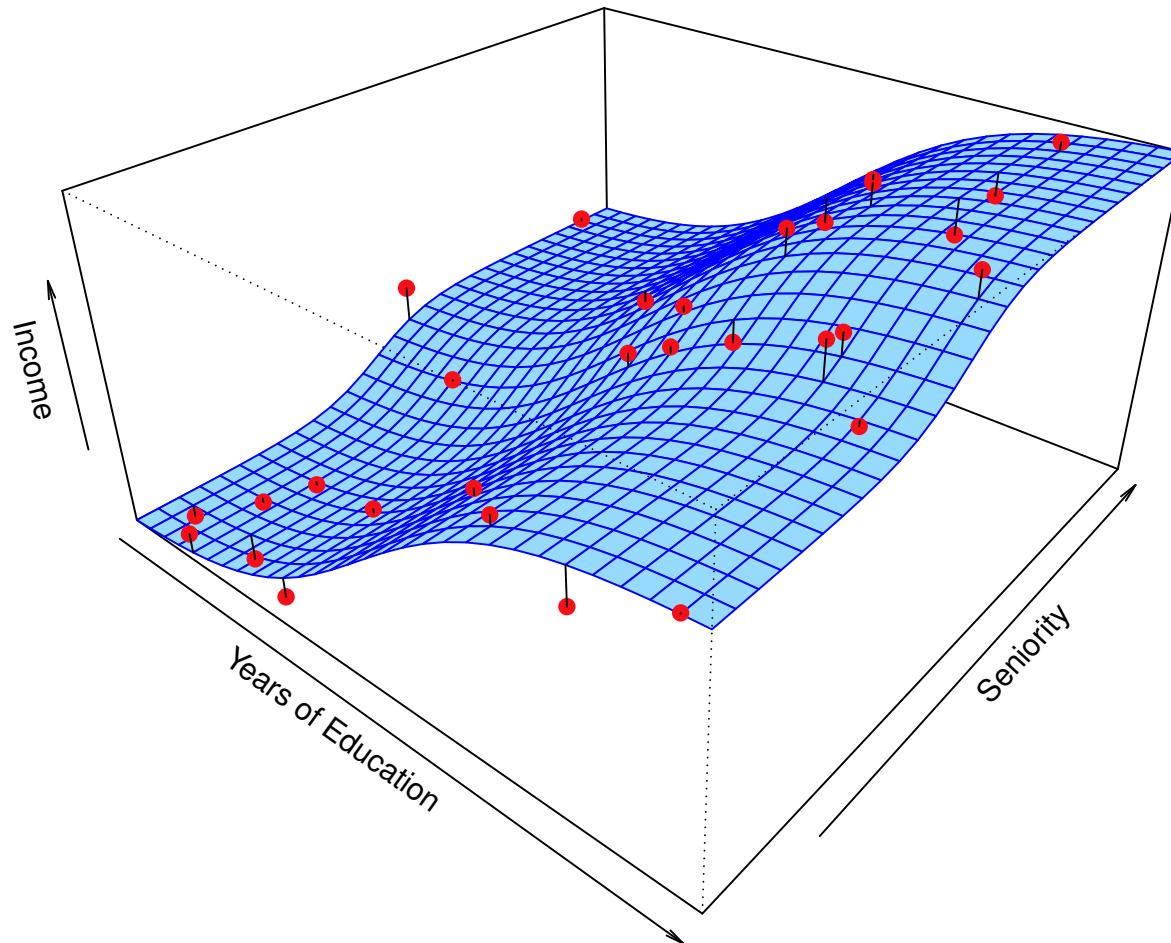
Example



Supervised Learning



Example



Supervised Learning

Why Estimate $f(X)$?

- **Prediction**
 - In many situations, we have the set of inputs X , but are lacking the output Y
 - In this case, we can predict Y using $f(X)$
 - In this setting, f can be treated as a *black box*, provided that it yields accurate predictions for Y

Why Estimate $f(X)$?

- **Explanation**
 - We are often interested in *understanding* the way Y is affected by a change in X_1, \dots, X_p
 - In this setting, f cannot be treated as black box, because we want to know its exact form
- The exact form of f can answer the following questions:
 - Which inputs are associated with the output?
 - How does the output change, if one of the inputs changes by one unit?
 - ...

How to Estimate $f(X)$?

- Parametric methods
 1. Make assumptions about the functional form of f . For example, we may assume that f is linear and additive (i.e., linear model):

$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$$

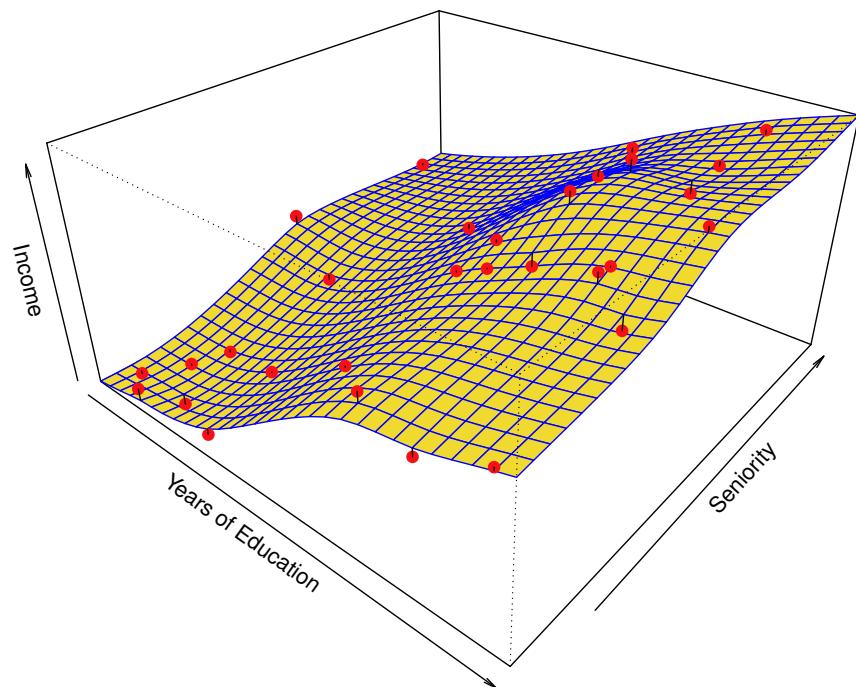
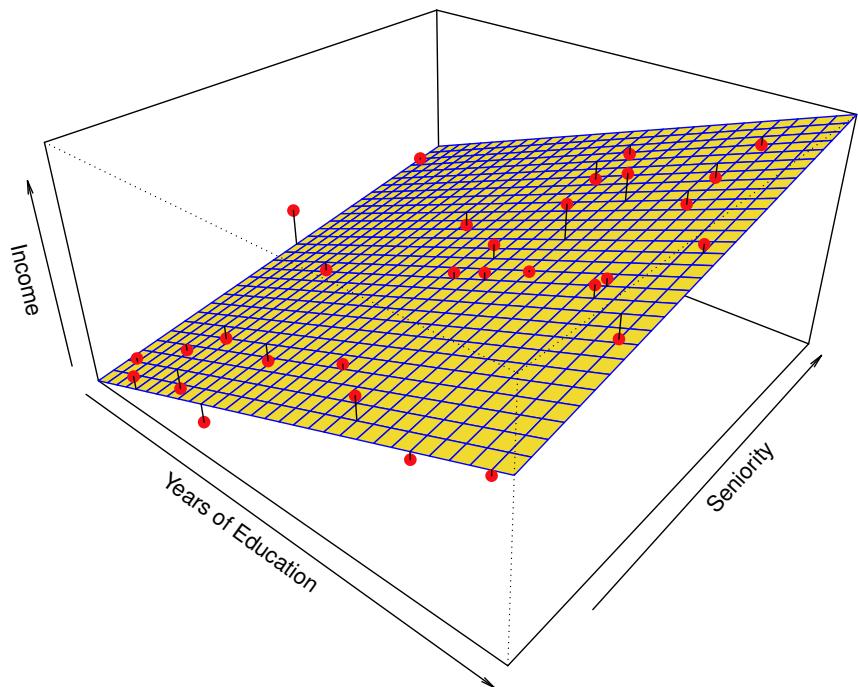
Supervised Learning

How to Estimate $f(X)$?

- Non-parametric methods
 - Also have parameters!
 - Do not make explicit assumptions about the functional form of f
 - Instead, they seek an estimate of f that gets as close as possible to the data points (i.e., they are more flexible)
 - Advantage:
 - They have the potential to accurately fit a wider range of possible shapes of f (i.e., they are more accurate)
 - Disadvantages:
 - Require more observations
 - Can easily overfit the data
 - Are more difficult to interpret

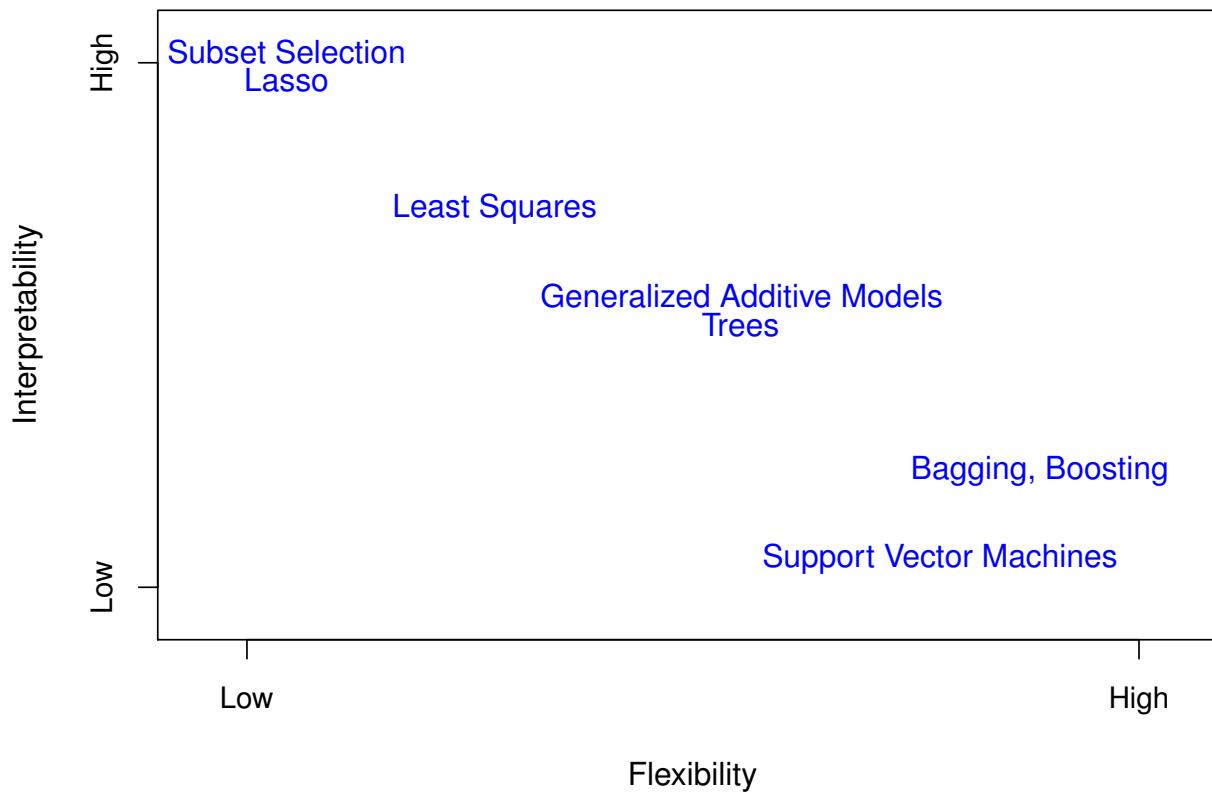
Supervised Learning

Example



Supervised Learning

Trade-Off between Accuracy and Interpretability



Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

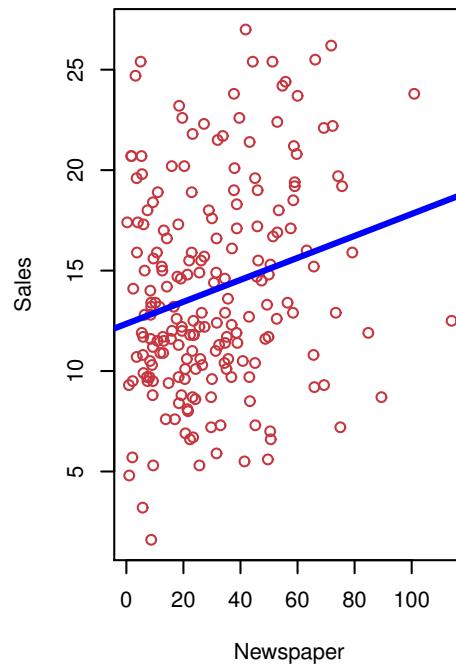
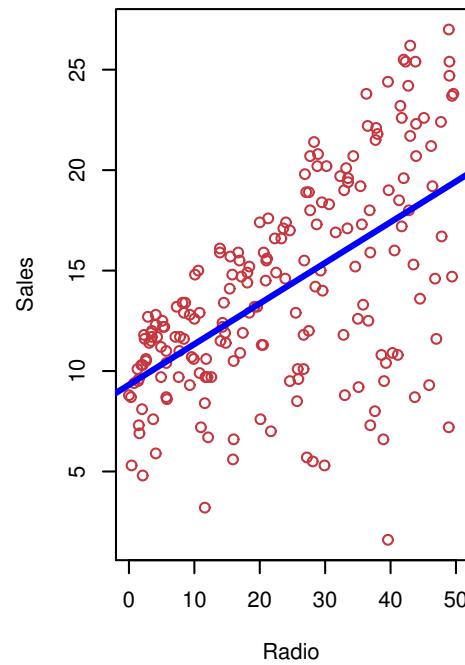
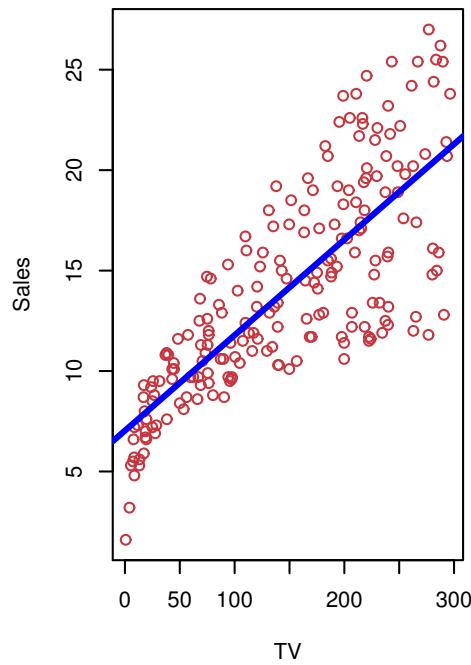
Syllabus

Supervised Learning:
_ Fundamentals of Linear Regression

Fundamentals of Linear Regression



Graphical Interpretation



Fundamentals of Linear Regression



As a Formula

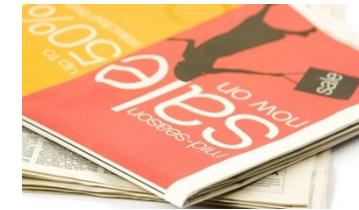
- Simple linear regression is a method to estimate a quantitative response Y on the basis of a singular predictor variable X
- It assumes that there is approximately a linear relationship between X and Y

$$Y \approx \beta_0 + \beta_1 X$$

$$\text{sales} \approx \beta_0 + \beta_1 \times \text{TV}$$

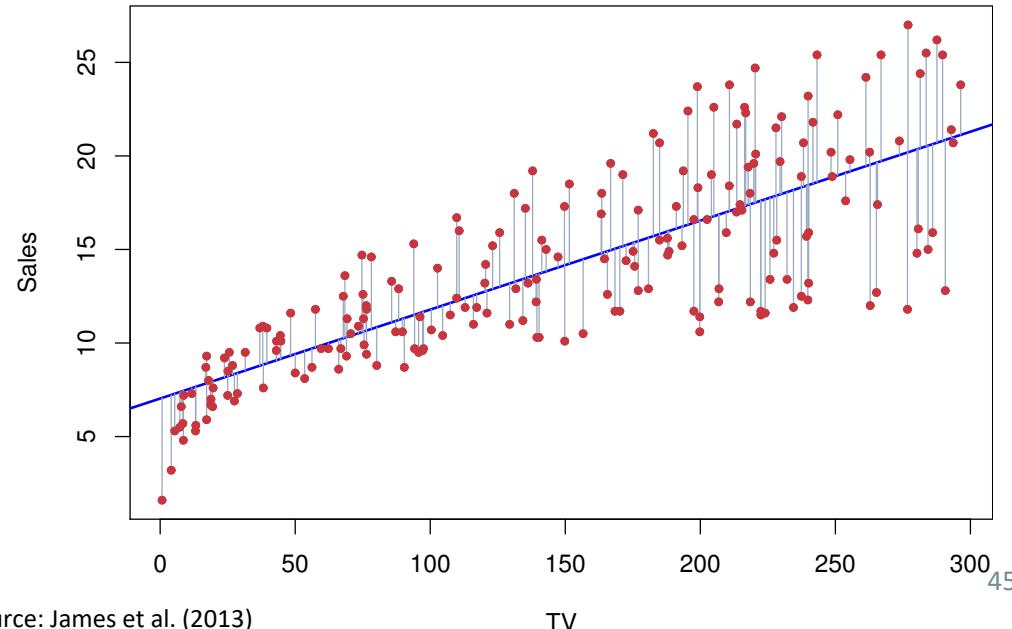
↑ ↑
Intercept Slope

Fundamentals of Linear Regression



Estimating the Coefficients

- In practice, the coefficients (intercept and slope) are unknown
- Ordinary Least Squares (OLS) approach is used to estimate the coefficients
 - $e_i = y_i - \hat{y}_i$
 - $RSS = e_1^2 + e_2^2 + e_3^2 + \dots + e_n^2$
- OLS chooses β_0 and β_1 to minimize RSS



Fundamentals of Linear Regression

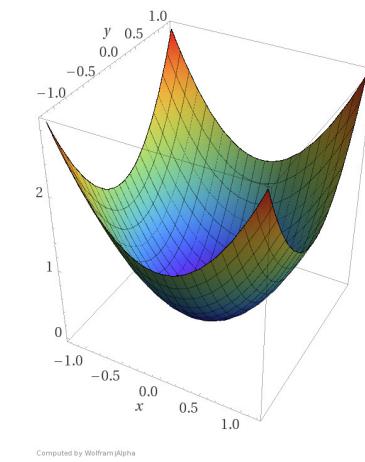
Finding the Best Fit

- How do we find intercept (β_0) and slope (β_1) values that minimize RSS?
- Available techniques
 - Brute Force (computationally expensive)
 - Closed Form Equation (only applicable for simple linear regression)
 - Matrix Inversion or Matrix Decomposition (requires a good command of linear algebra)
 - **Gradient Descent**
 - ...

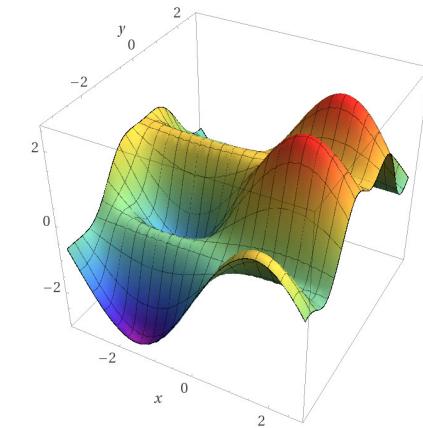
Fundamentals of Linear Regression

Intuition of Gradient Descent

- Optimization algorithm for finding a local minimum of a differentiable function.
- Thought experiment:
 - You are a hiker (parameter value) in a mountain range (loss function) at night.
 - Your goal is to reach the valley (global minimum) as fast as possible.
 - Using your flashlight (differentiation), you can see the slope of the mountain (partial derivative, also known as gradient) around your current location.
 - You take a step downward in the direction where the slope is steepest (parameter value update).
 - You take long steps when the slope is steep (learning rate * gradient) and short steps when the slope is relatively flat (learning rate * gradient).
 - After many steps, you will reach the valley (global minimum) or get stuck in a hole (local minimum).



Computed by WolframAlpha



Computed by WolframAlpha

Fundamentals of Linear Regression

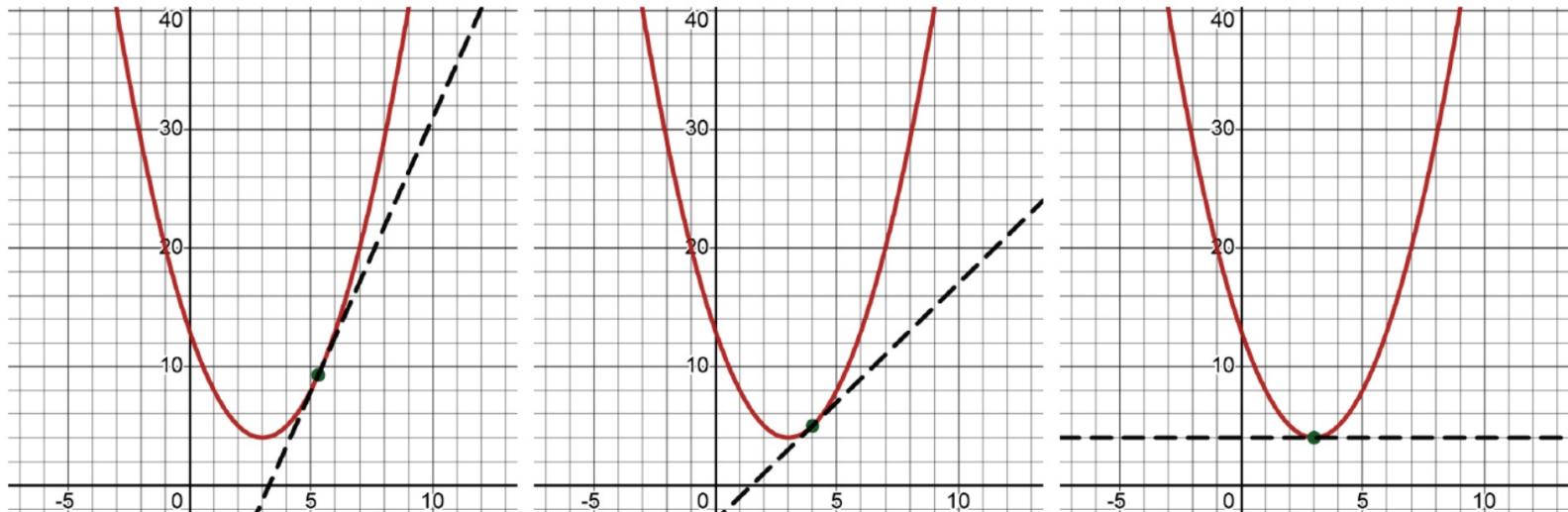
Numerical Example of Gradient Descent

$$f(x) = (x - 3)^2 + 4$$

$$\frac{\partial f(x)}{\partial x} = 2(x - 3)$$

$$x_{new} = x - learning_rate * \frac{\partial f(x)}{\partial x}$$

$$learning_rate = 0.1$$



Fundamentals of Linear Regression

Gradient Descent for Simple Linear Regression

- Function to minimize:

$$RSS = \sum_{i=1}^n (\beta_0 + \beta_1 x_i - y_i)^2$$

- Partial derivatives:

$$\frac{\partial RSS}{\partial \beta_0} = \sum_{i=1}^n 2(\beta_0 + \beta_1 x_i - y_i)$$

$$\frac{\partial RSS}{\partial \beta_1} = \sum_{i=1}^n 2x_i(\beta_0 + \beta_1 x_i - y_i)$$

- Parameter updates:

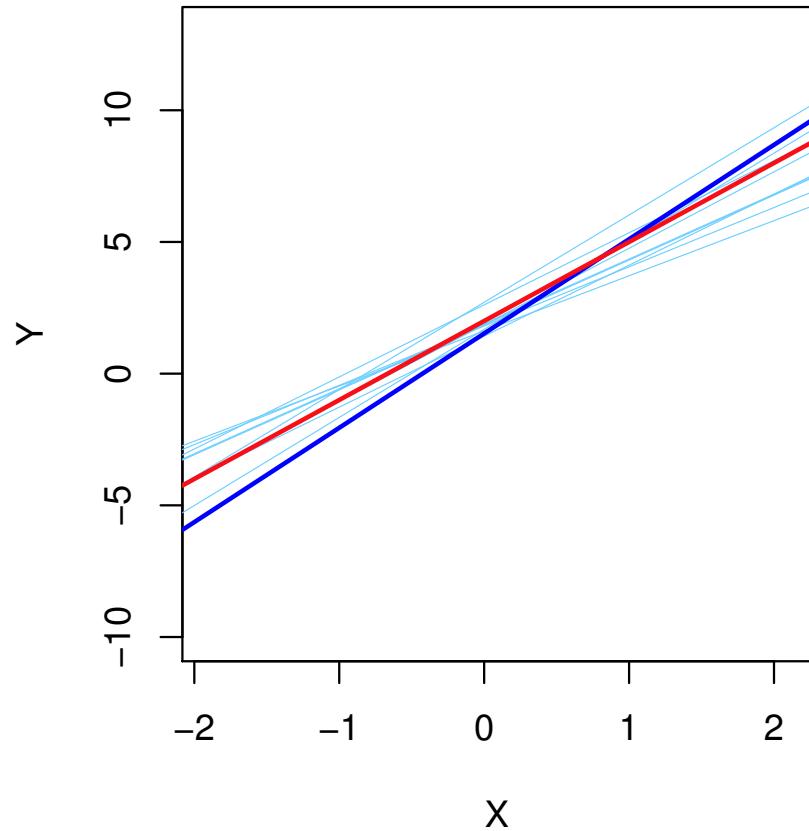
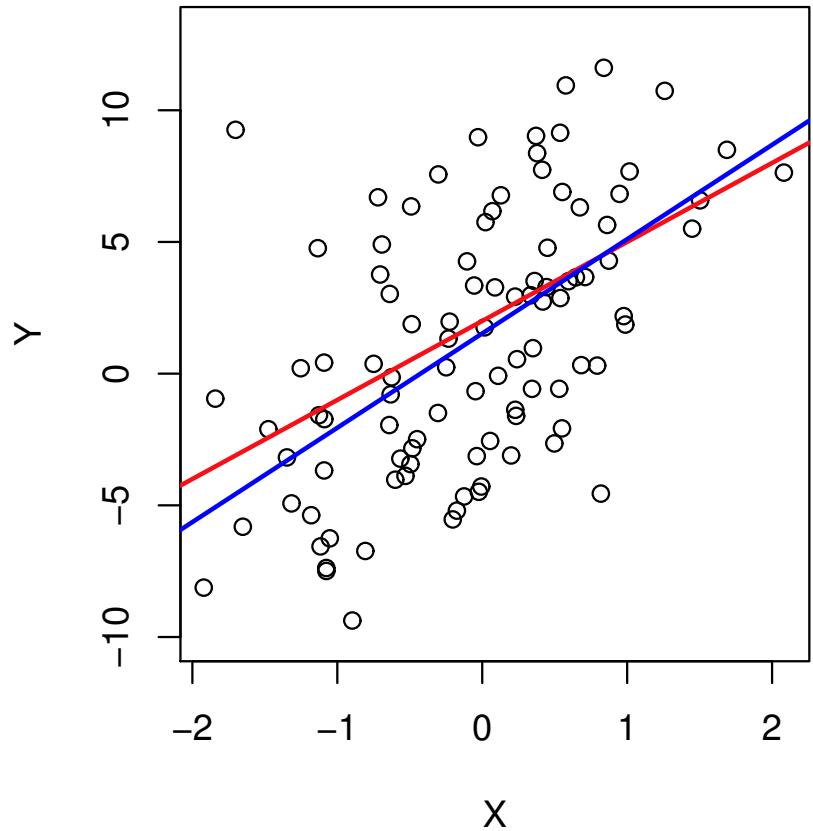
$$\beta_{0new} = \beta_0 - learning_rate * \frac{\partial RSS}{\partial \beta_0}$$

$$\beta_{1new} = \beta_1 - learning_rate * \frac{\partial RSS}{\partial \beta_1}$$

KEEP
CALM
IT'S
DEMO
TIME

Fundamentals of Linear Regression

Accuracy of Coefficient Estimates



Fundamentals of Linear Regression



Accuracy of Coefficient Estimates

Estimates of the coefficients (i.e., intercept and slopes)

	Coefficient	Std. error	t-statistic	p-value
Intercept	7.0325	0.4578	15.36	< 0.0001
TV	0.0475	0.0027	17.67	< 0.0001

Measures of the accuracy of the estimates of the coefficients

Fundamentals of Linear Regression

Accuracy of Coefficient Estimates

- How accurate are the estimates of the model's coefficients?
- **Standard Error**
 - Roughly speaking, the standard error tells us the average amount that the estimate of β differs from the actual value of β .

$$\text{SE}(\hat{\beta}_0)^2 = \sigma^2 \left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \right], \quad \text{SE}(\hat{\beta}_1)^2 = \frac{\sigma^2}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

- **t-statistic**
 - The t-statistic measures the number of standard deviations that β is away from 0.

$$t = \frac{\hat{\beta}_1 - 0}{\text{SE}(\hat{\beta}_1)}$$

Fundamentals of Linear Regression

Accuracy of Coefficient Estimates

- The **p-value** of a statistical test tells us how likely it is – given the collected data – to get results at least as extreme as the ones you observed, given that the null hypothesis (i.e., "there is no relationship") is correct.
- In the social sciences, the threshold for the p-value is typically set to 0.05
- Some misunderstandings and critiques
 - A p-value does not measure the size of an effect or the importance of a result.
 - Scientific conclusions or policy decisions should not be based only on whether a p-value passes a specific threshold ("What's the difference between 0.049 and 0.051?").
 - Any threshold (e.g. 0.1, 0.05, 0.01) is an arbitrary threshold.
 - P-values naturally shrink with increasing sample size ("too big to fail").
 - ...

Fundamentals of Linear Regression

Accuracy of Model

- To what extent does the whole model fit the data?
- R^2
 - Relative measure of model fit (proportion of variance explained), always between 0 and 1

$$R^2 = \frac{\text{TSS} - \text{RSS}}{\text{TSS}} = 1 - \frac{\text{RSS}}{\text{TSS}}$$

$$\text{RSS} = e_1^2 + e_2^2 + \cdots + e_n^2$$

$$\text{TSS} = \sum(y_i - \bar{y})^2$$

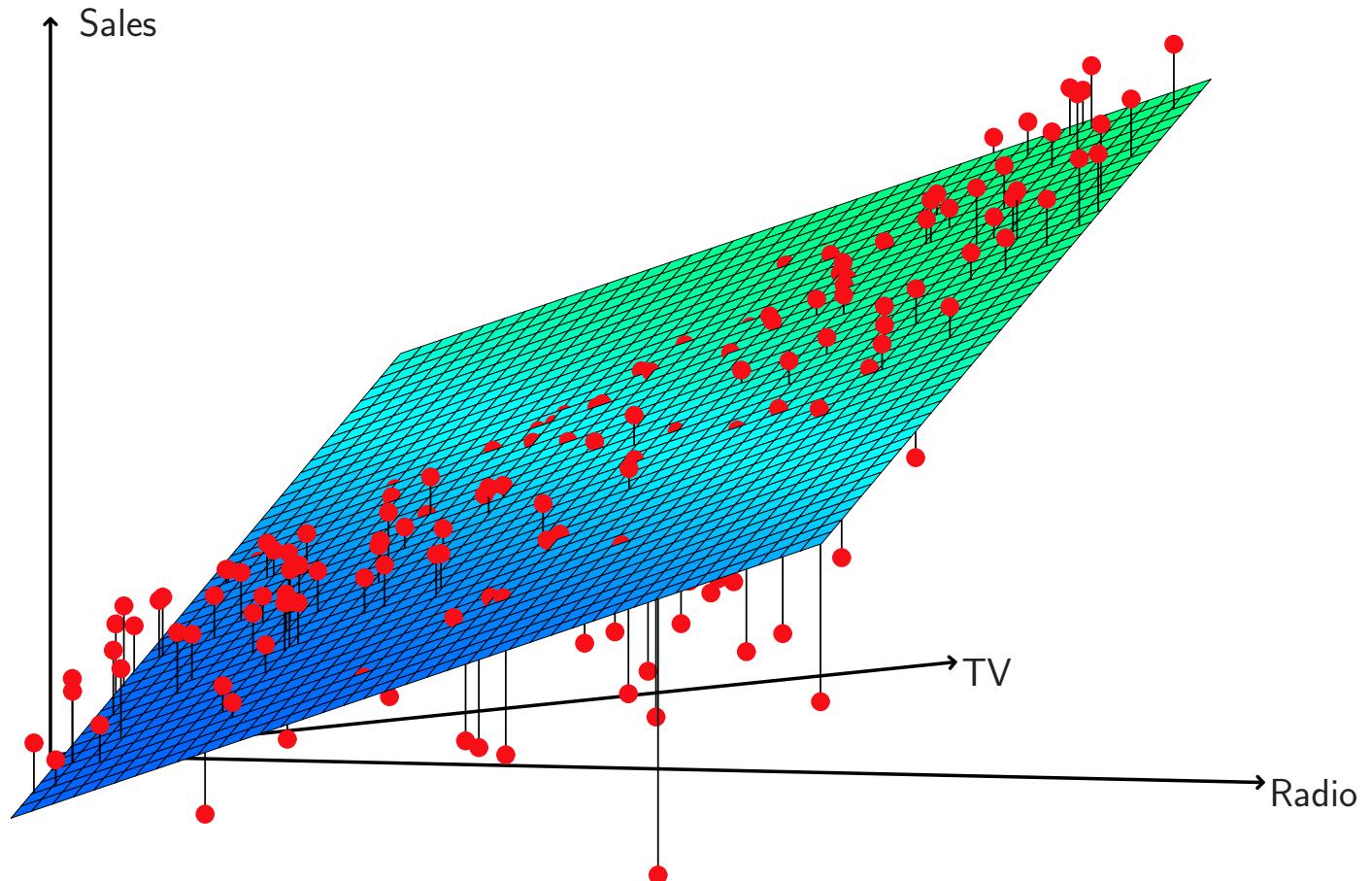
Fundamentals of Linear Regression

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

Fundamentals of Linear Regression



Two Predictors



Fundamentals of Linear Regression



Three Predictors

	Coefficient	Std. error	t-statistic	p-value
Intercept	2.939	0.3119	9.42	< 0.0001
TV	0.046	0.0014	32.81	< 0.0001
radio	0.189	0.0086	21.89	< 0.0001
newspaper	-0.001	0.0059	-0.18	0.8599

Fundamentals of Linear Regression

The Big 5 Assumptions

1. Linear and additive relationship between IVs and DV
2. Homoscedasticity of errors
3. Independence of errors
4. Normality of errors
5. No multicollinearity

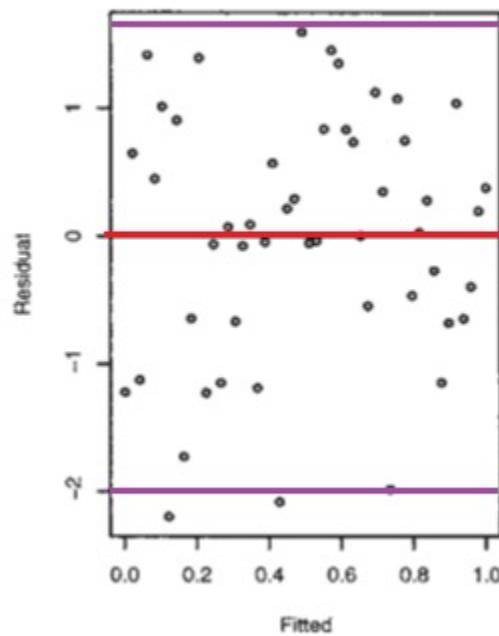


Fundamentals of Linear Regression

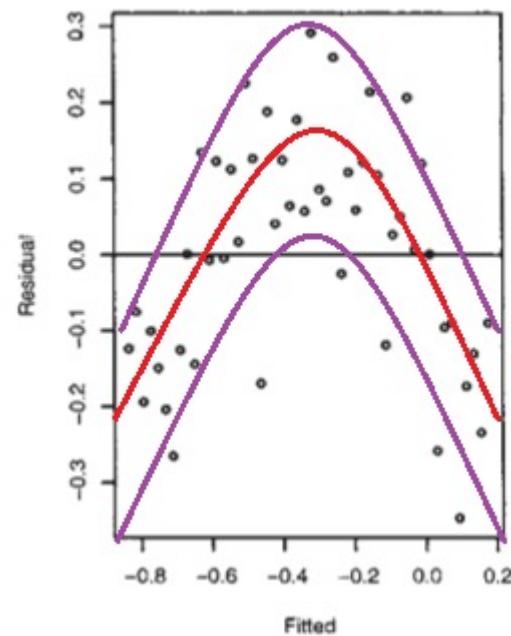
1) Linear and additive relationship between IVs and DV



No problem



Nonlinear

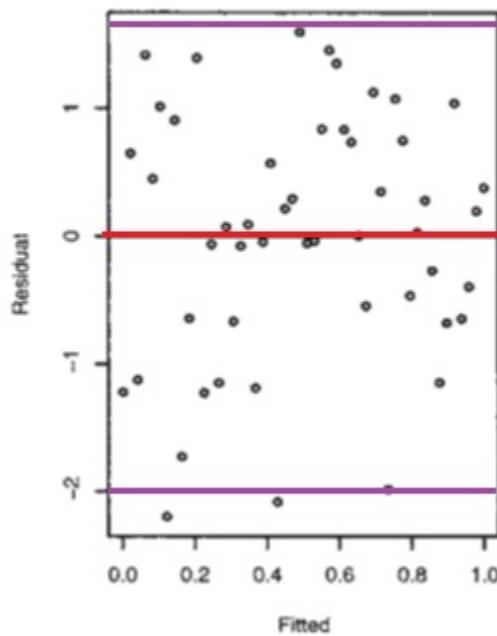


Fundamentals of Linear Regression

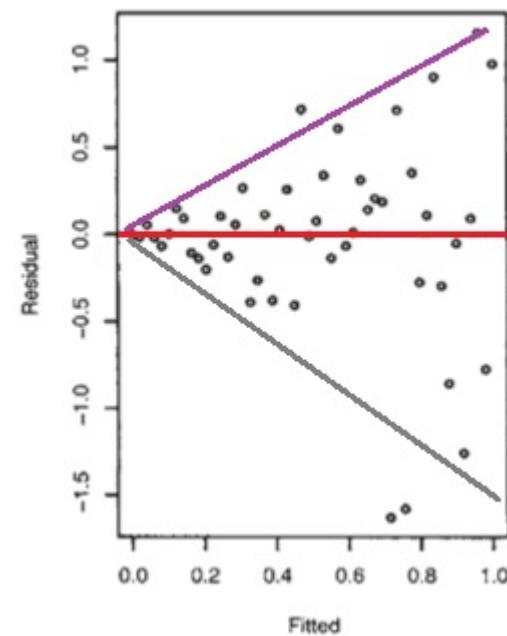
2) Homoscedasticity of errors



No problem

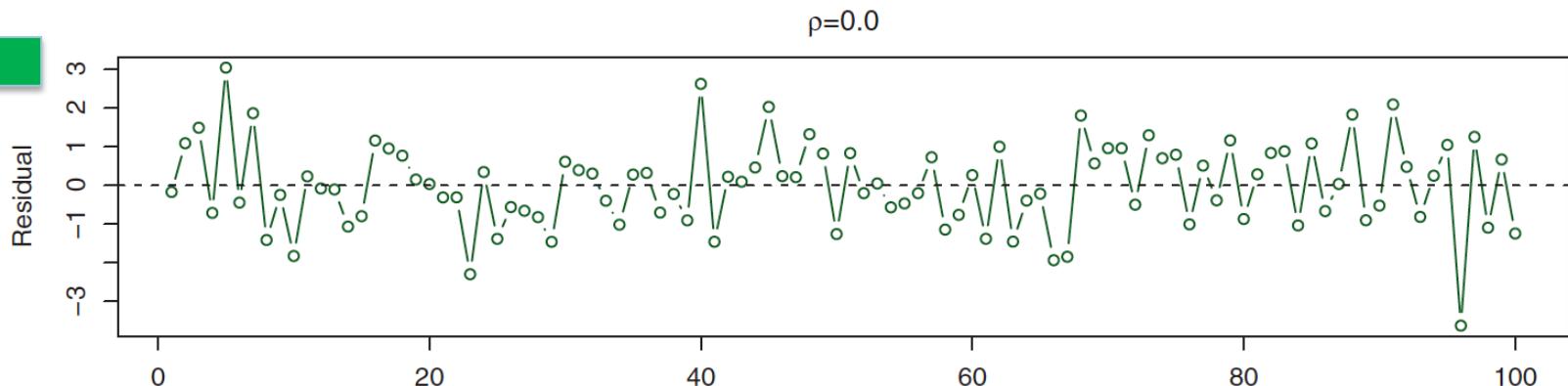


Heteroscedasticity

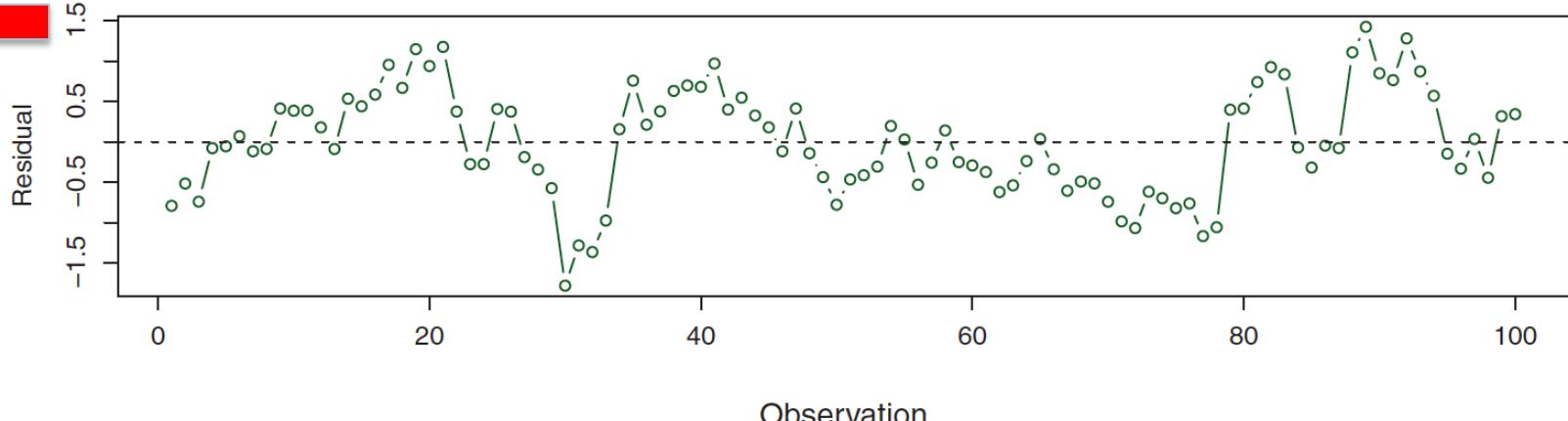


Fundamentals of Linear Regression

3) Independence of errors



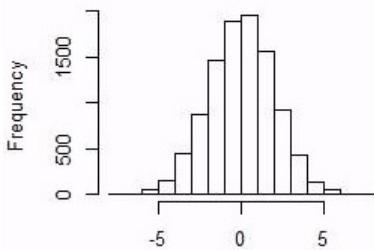
$\rho=0.9$



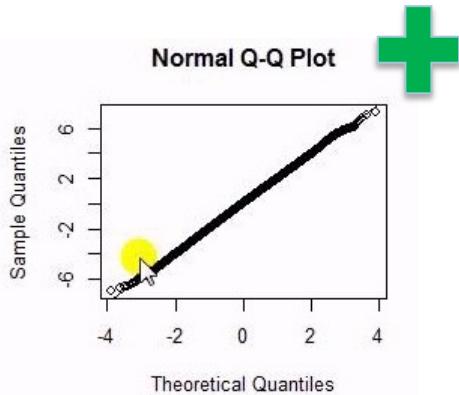
Fundamentals of Linear Regression

4) Normality of errors

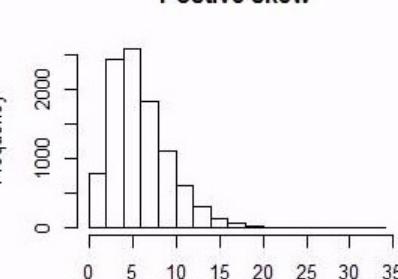
Symmetric distribution



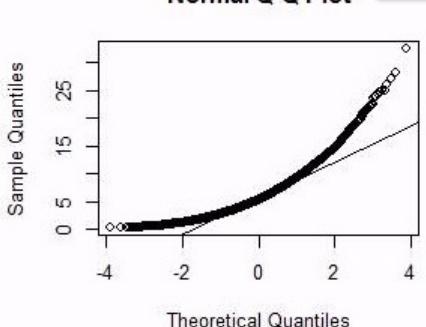
Normal Q-Q Plot



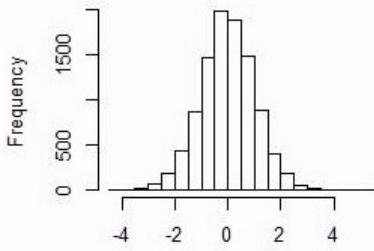
Positive skew



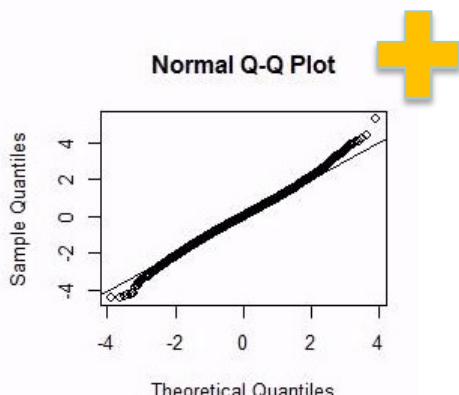
Normal Q-Q Plot



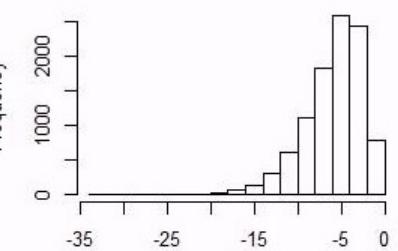
Symmetric with fat tails



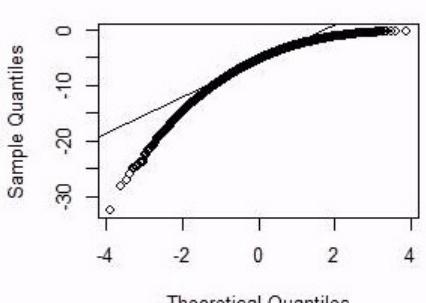
Normal Q-Q Plot



Negative skew



Normal Q-Q Plot



Fundamentals of Linear Regression

5) No multicollinearity



```
cor(data[,c("TV", "Radio", "Newspaper")])
```

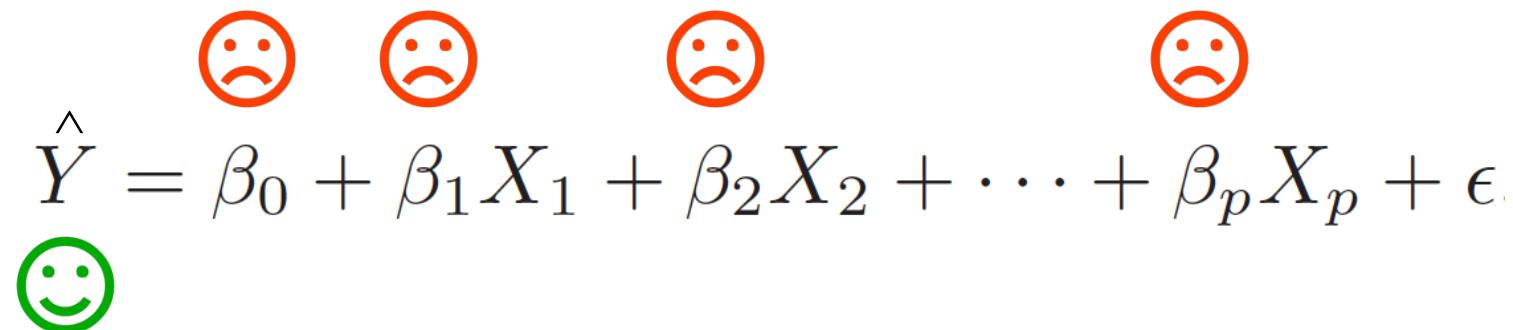
```
##                      TV      Radio  Newspaper
## TV            1.00000000 0.05480866 0.05664787
## Radio          0.05480866 1.00000000 0.35410375
## Newspaper      0.05664787 0.35410375 1.00000000
```

```
vif(fit_all)
```

```
##          TV      Radio  Newspaper
## 1.004611 1.144952 1.145187
```

Fundamentals of Linear Regression

What if the Big 5 are violated?

$$\hat{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$


The equation $\hat{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$ is displayed below four red circular icons containing sad faces. A single green circular icon containing a smiley face is positioned below the equation.



Syllabus

Syllabus

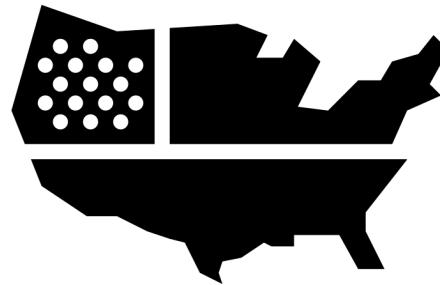
- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Supervised Learning:
_ Feature Engineering

Feature Engineering

Encoding Categorical Predictors



Feature Engineering

Categorical Predictors with Two Levels

- Create *one* dummy variable that takes on *two* possible values
- Example: House owner $x_i = \begin{cases} 1 & \text{if } i\text{th person owns a house} \\ 0 & \text{if } i\text{th person does not own a house} \end{cases}$

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i = \begin{cases} \beta_0 + \beta_1 + \epsilon_i & \text{if } i\text{th person owns a house} \\ \beta_0 + \epsilon_i & \text{if } i\text{th person does not.} \end{cases}$$

Feature Engineering

Categorical Predictors with Two Levels



Predicting the credit card balance of customers

	Coefficient	Std. error	t-statistic	p-value
Intercept	509.80	33.13	15.389	< 0.0001
own [Yes]	19.73	46.05	0.429	0.6690

Reference level of "own": No

Feature Engineering

Categorical Predictors with More than Two Levels

- Create *multiple* dummy variables. For n levels, we need $n-1$ dummy variables.
- Example: Region with 3 levels (i.e., East, West, South)



$$x_{i1} = \begin{cases} 1 & \text{if } i\text{th person is from the South} \\ 0 & \text{if } i\text{th person is not from the South} \end{cases} \quad x_{i2} = \begin{cases} 1 & \text{if } i\text{th person is from the West} \\ 0 & \text{if } i\text{th person is not from the West} \end{cases}$$
$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \epsilon_i = \begin{cases} \beta_0 + \beta_1 + \epsilon_i & \text{if } i\text{th person is from the South} \\ \beta_0 + \beta_2 + \epsilon_i & \text{if } i\text{th person is from the West} \\ \beta_0 + \epsilon_i & \text{if } i\text{th person is from the East.} \end{cases}$$

Feature Engineering

Categorical Predictors with More than Two Levels



Predicting the credit card balance of customers

	Coefficient	Std. error	t-statistic	p-value
Intercept	531.00	46.32	11.464	< 0.0001
region[South]	-18.69	65.02	-0.287	0.7740
region[West]	-12.50	56.68	-0.221	0.8260

Reference level of “region”: East

Feature Engineering

Multiple Categorical Predictors



Dependent variable:	
Balance	
Constant	520.880*** (51.901)
OwnYes	20.038 (46.178)
RegionSouth	-12.653 (56.740)
RegionWest	-19.371 (65.107)
Observations	400
R2	0.001
Adjusted R2	-0.007
Residual Std. Error	461.337 (df = 396)
F Statistic	0.092 (df = 3; 396)

Note: *p<0.1; **p<0.05; ***p<0.01

Reference level: Own=No & Region=East

Feature Engineering

Encoding Predictors with Many Categories

- What happens when the number of categories becomes very large? For example, more than 40K ZIP codes exist in the US.
- **Problems** that can arise in such situations:
 - Dummy coding will produce an overabundance of dummy variables relative to the number of data points.
 - Some categories might be much more frequent than others (“long tail” distribution), which can lead to **(a)** dummy variables that contain all zeros when sampling your data or **(b)** novel category values appearing in the future.
- **Possible solutions:**
 - Only dummy code the most frequent categories
 - Subsume infrequent or novel categories under “others” category

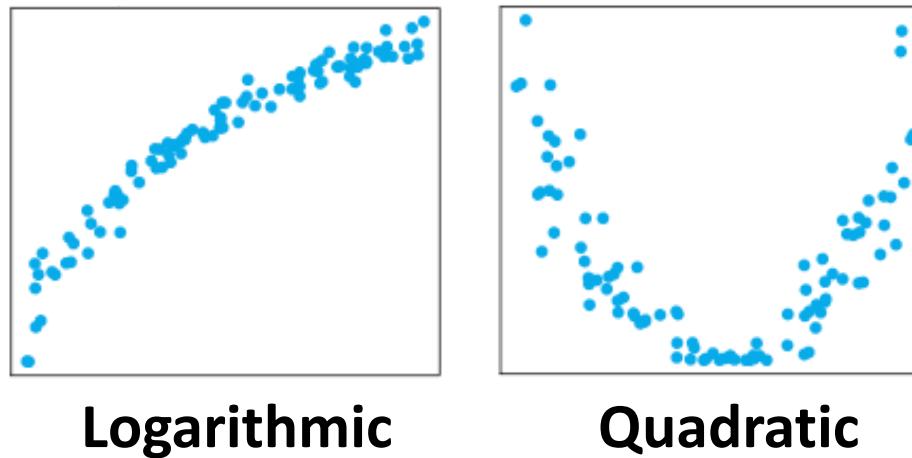
Feature Engineering

Effect Encoding

- A supervised way of transforming categorical predictors to numeric predictors
- Idea of effect encoding
 - For each level of a categorical predictor, measure its effect on the response variable (e.g., by simply calculating the average response variable value per level).
 - Take this numerical effect estimate as the encoding of that category level.
 - Example: Calculate the median credit card balance for each ZIP from the training data and use this statistic to represent ZIP in the model.
- Potential issues with effect encoding
 - This can easily lead to overfitting, especially if the encoding is estimated on the same dataset as the main modeling
 - The risk of overfitting increases for rare categories with high variation

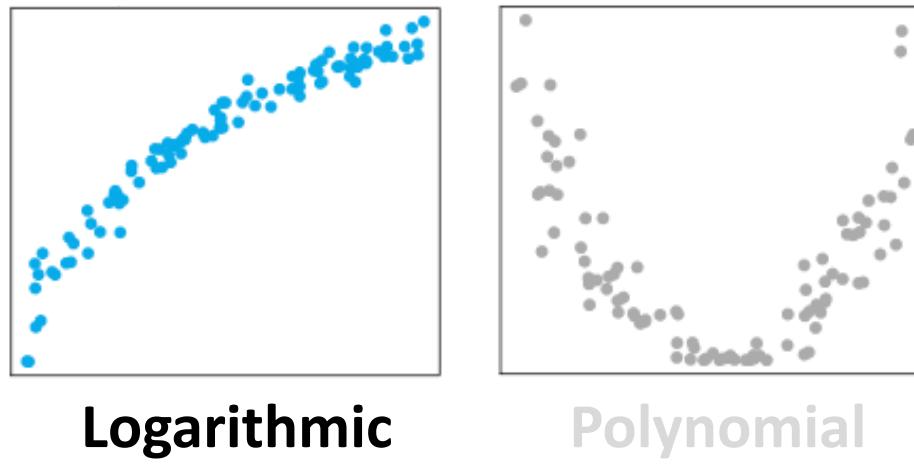
Feature Engineering

Engineering Numerical Predictors



Feature Engineering

Engineering Numerical Predictors

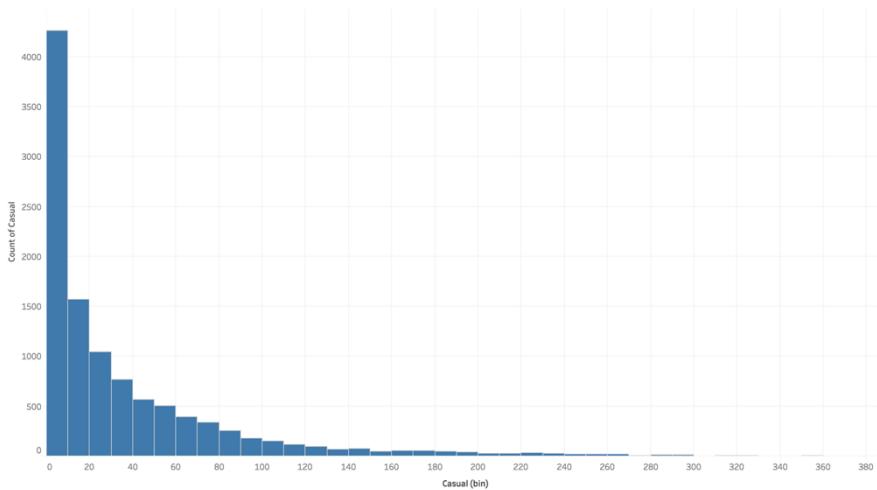


Feature Engineering

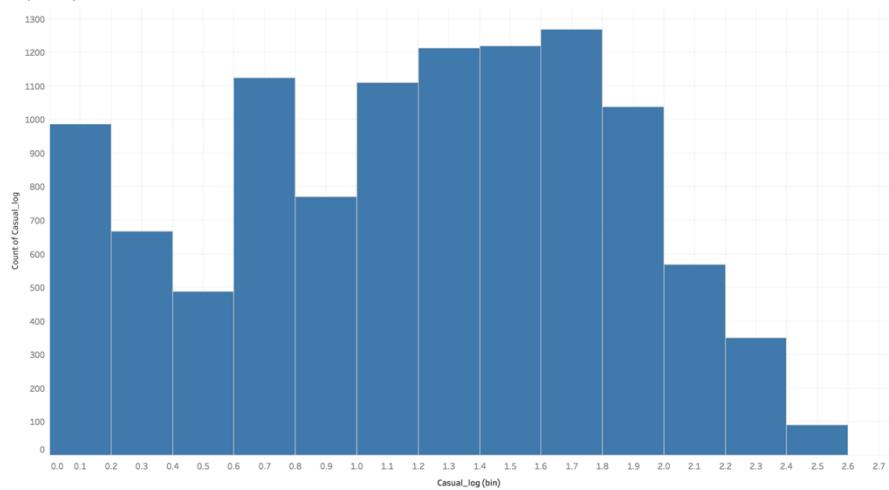
Log-Transforming the Outcome



Casual riders

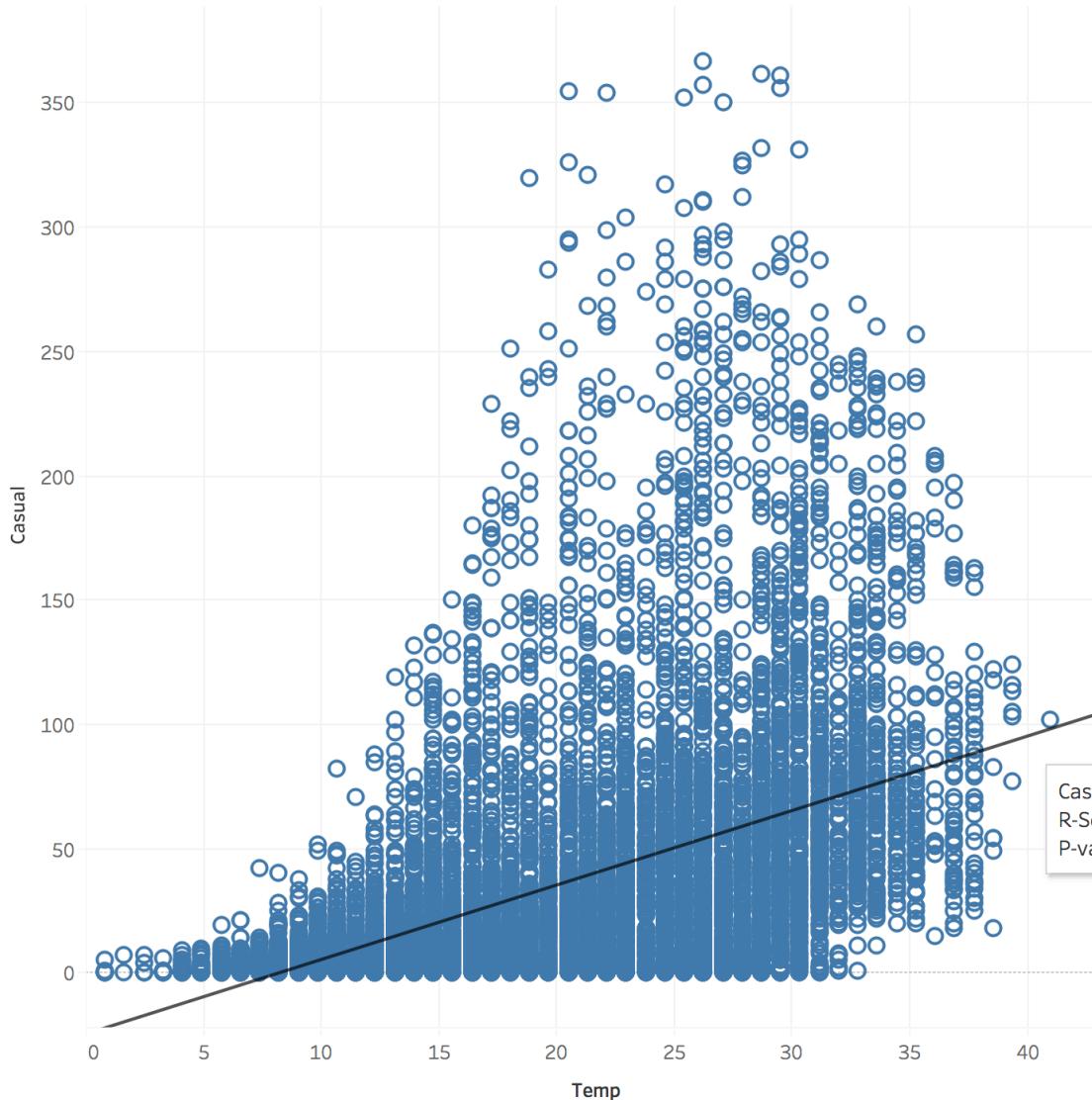


Logarithm of casual riders



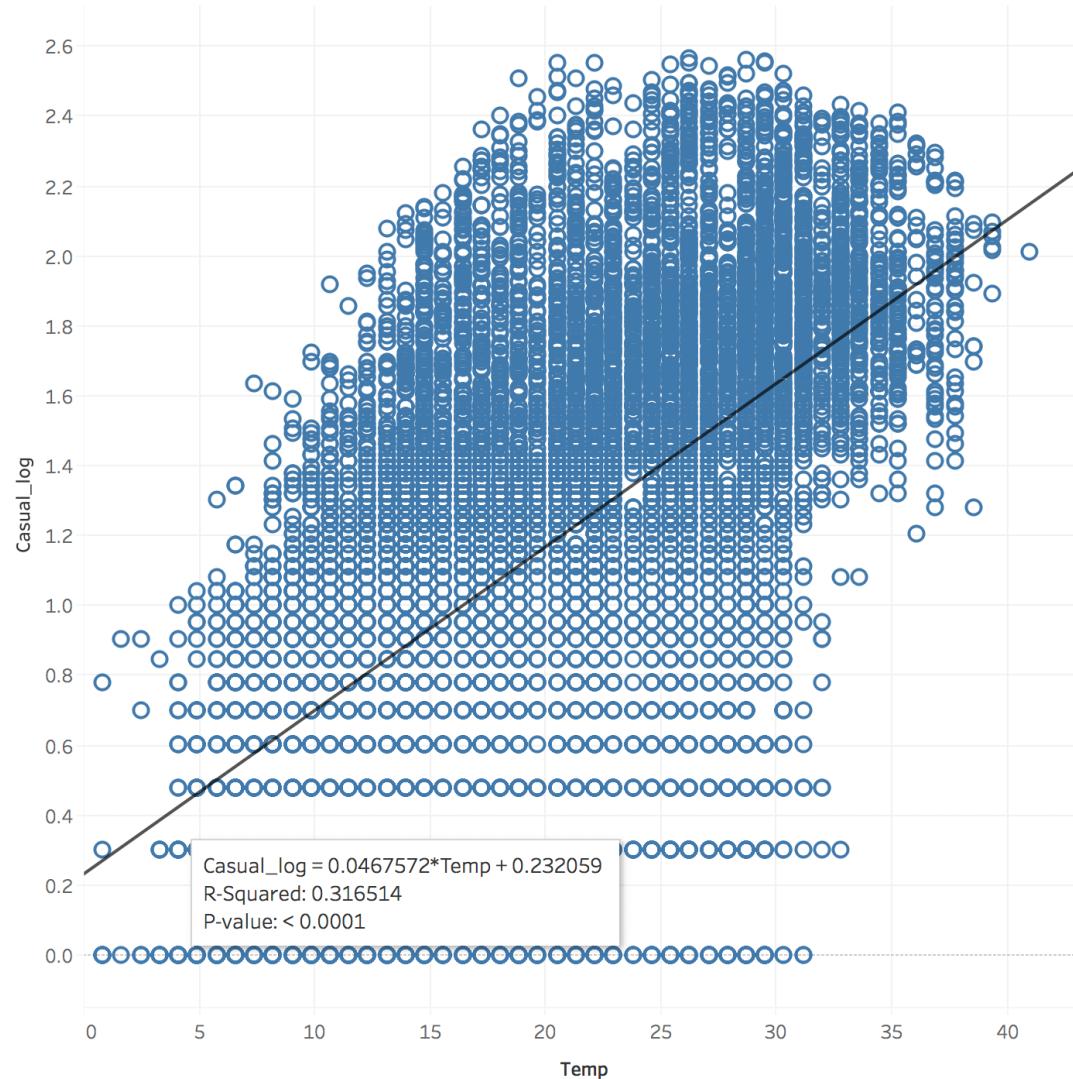
Feature Engineering

Log-Transforming the Outcome



Feature Engineering

Log-Transforming the Outcome

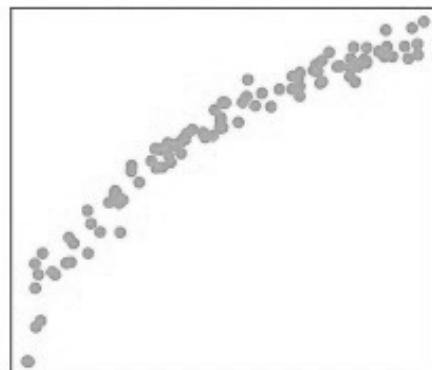


Log-Transforming the Outcome

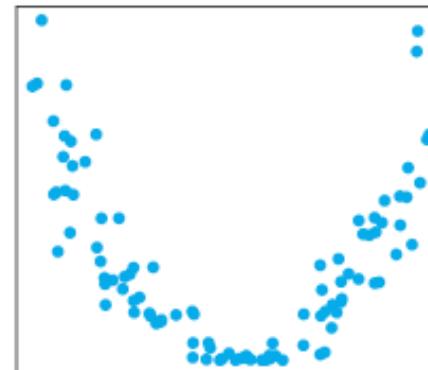
- How do I interpret a regression model when the response variable is log transformed?
 - Note that β_1 is now the expected change in $\log(Y)$ with respect to a one-unit increase in X_1
- We can then interpret the exponentiated regression coefficient $\exp(\beta_1)$, since exponentiation is the inverse of the logarithm function.
- The exponentiated coefficient corresponds to the expected percentage-change in Y for a one-unit increase in X_1
 - Example: for a one-unit increase in temp, we expect to see a 11% increase in casual riders, since $\exp(0.108) = 1.11$

Feature Engineering

Engineering Numerical Predictors



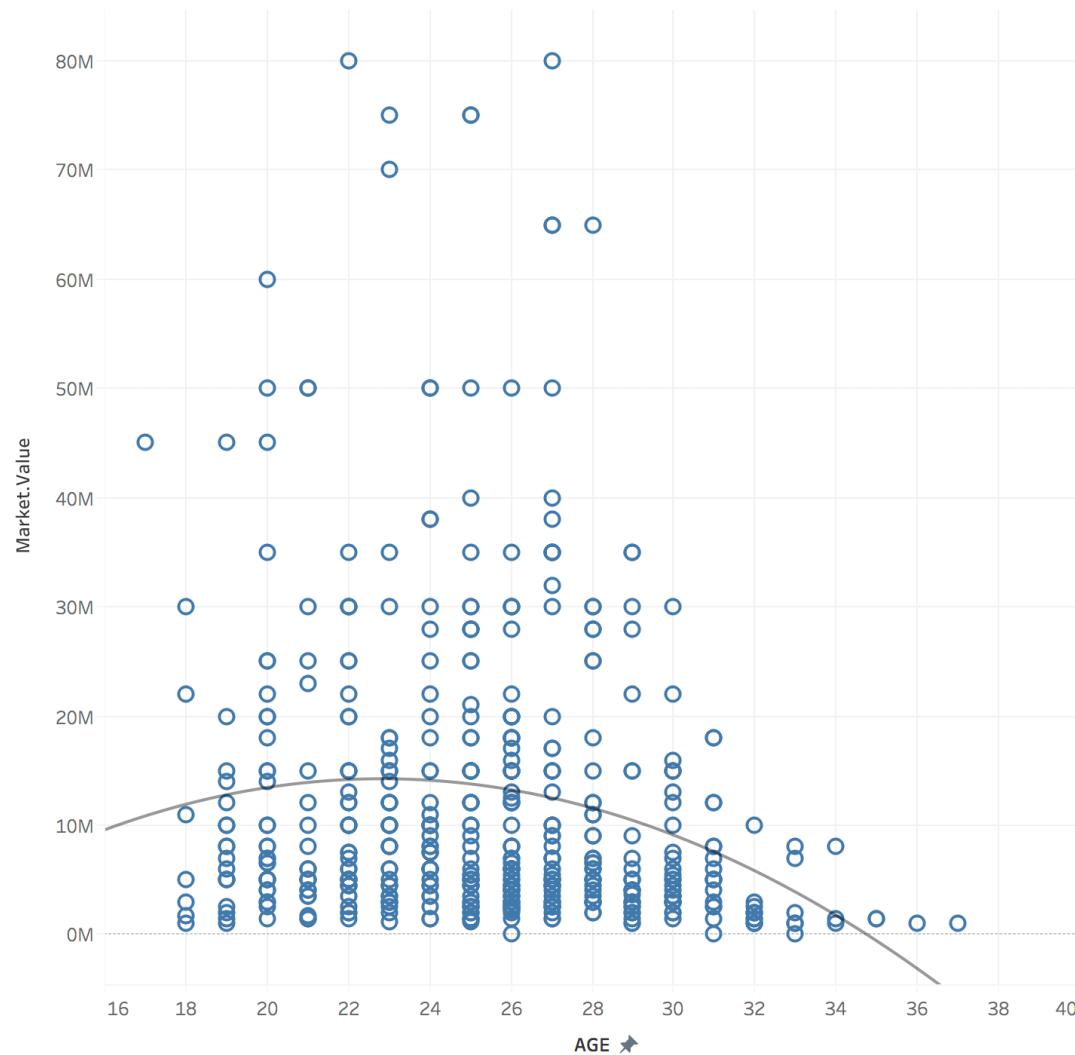
Logarithmic



Polynomial

Feature Engineering

Quadratic Transformation of a Predictor



Feature Engineering

Quadratic Transformation of a Predictor



Dependent variable:	

MARKET.VALUE	

Constant	26,576,226.000*** (4,874,635.000)
AGE	-562,176.400*** (190,553.800)

Observations	394
R2	0.022
Adjusted R2	0.019
Residual Std. Error	13,465,351.000 (df = 392)
F Statistic	8.704*** (df = 1; 392)

Note:	*p<0.1; **p<0.05; ***p<0.01

Feature Engineering

Quadratic Transformation of a Predictor



Dependent variable:	

MARKET.VALUE	

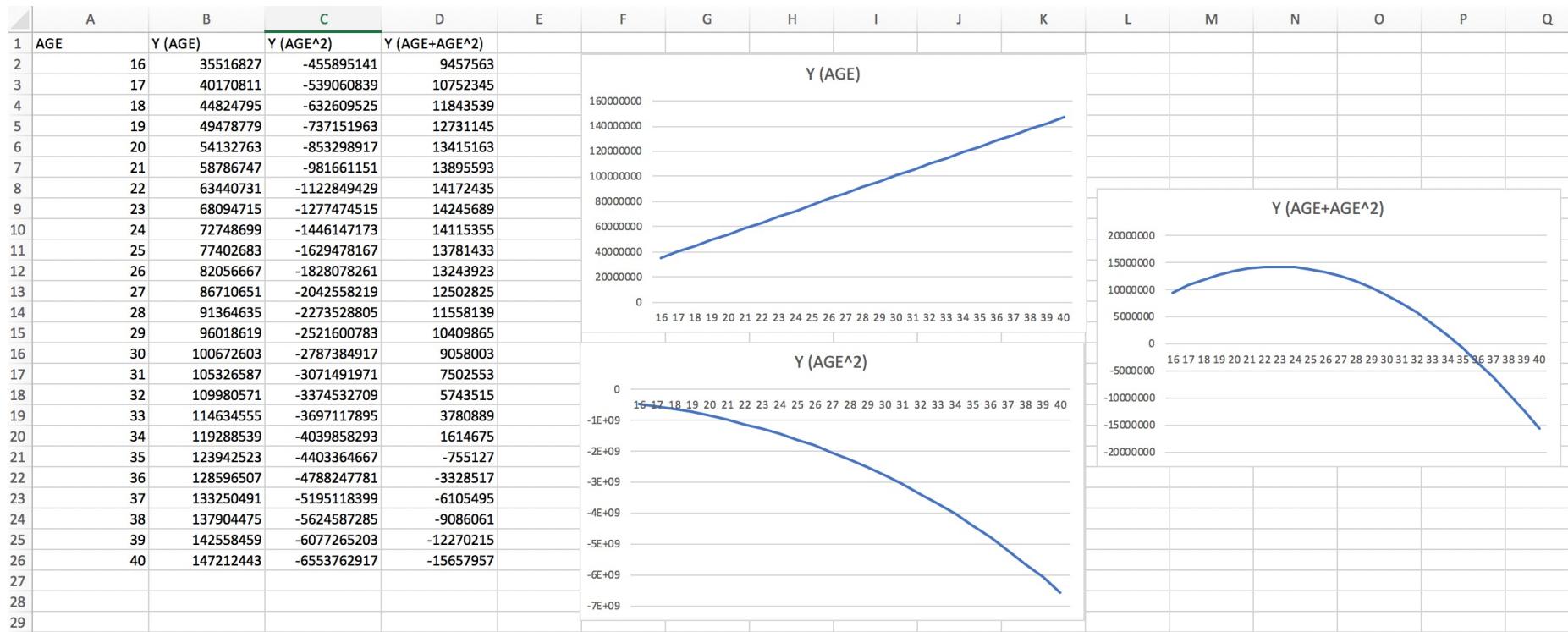
Constant	-38,946,917.000 (26,592,569.000)
AGE	4,653,984.000** (2,090,177.000)
I(AGE2)	-101,794.500** (40,622.640)

Observations	394
R2	0.037
Adjusted R2	0.032
Residual Std. Error	13,375,583.000 (df = 391)
F Statistic	7.550*** (df = 2; 391)

Note:	*p<0.1; **p<0.05; ***p<0.01

Feature Engineering

Visualizing Quadratic Relationships in Excel



Feature Engineering

Detecting Interaction Effects

SYNERGY
 $1+1>2$



Feature Engineering

Interaction Effects

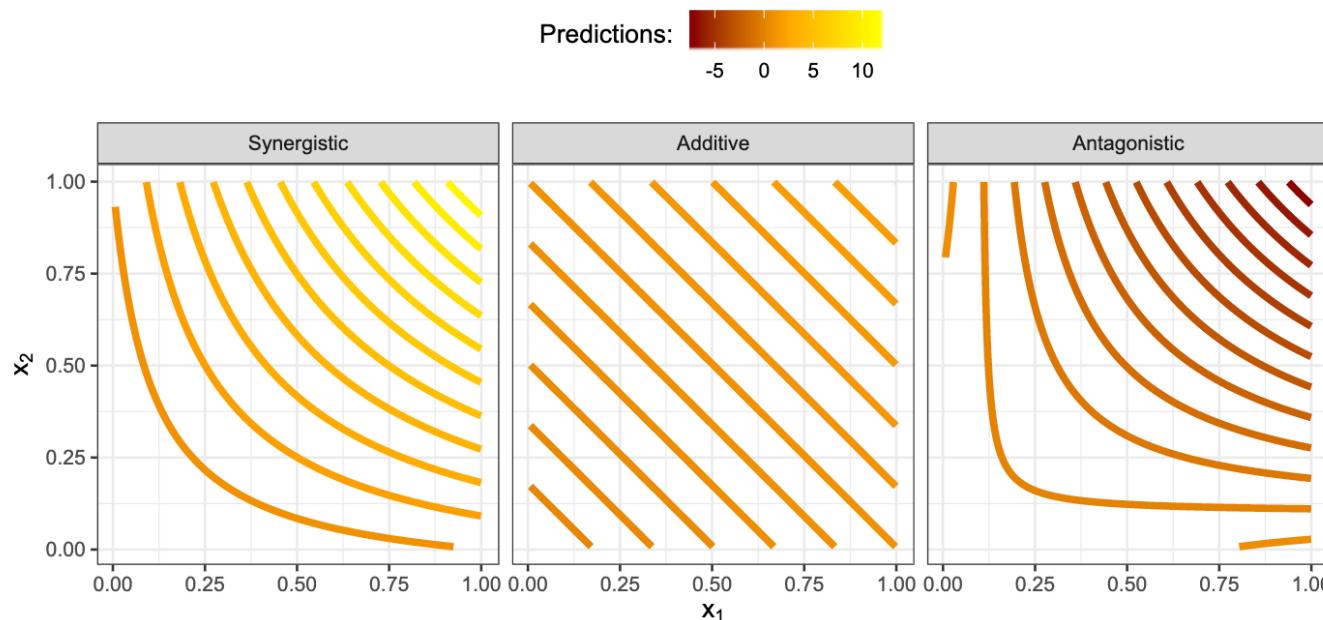
- The standard linear model assumes that the effect of one predictor on the outcome is independent of the effect of the other predictors on the outcome
- This is not always true! Example: Spending money on radio advertising may increase the effectiveness of TV advertising (synergy)

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \underline{\beta_3 X_1 X_2} + \epsilon$$

Feature Engineering

Interaction Effects

- Two or more predictors are said to interact, if their combined effect is different (less or greater) than what we would expect if we were to simply add their individual impacts.



Feature Engineering

Interaction of Numerical Predictors



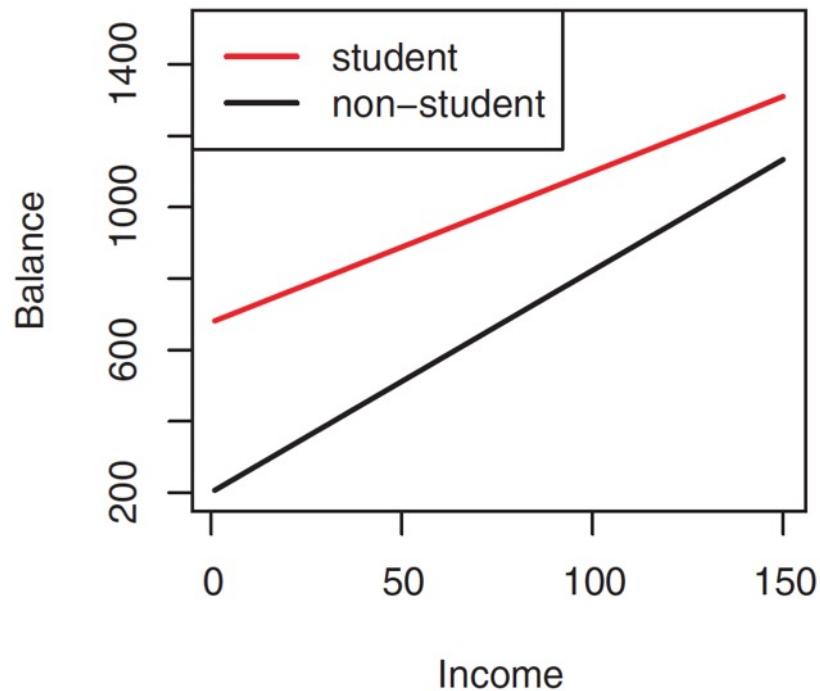
	Coefficient	Std. error	t-statistic	p-value
Intercept	6.7502	0.248	27.23	< 0.0001
TV	0.0191	0.002	12.70	< 0.0001
radio	0.0289	0.009	3.24	0.0014
TV×radio	0.0011	0.000	20.73	< 0.0001

Feature Engineering

Interaction of Numerical and Categorical Predictors



$$\text{balance}_i \approx \begin{cases} (\beta_0 + \beta_2) + (\beta_1 + \beta_3) \times \text{income}_i & \text{if student} \\ \beta_0 + \beta_1 \times \text{income}_i & \text{if not student.} \end{cases}$$



Identifying Predictive Interactions

- Due to the large search space and a risk of overfitting, expert-driven identification of interactions should be preferred over data-driven identification
- Guiding principles for identifying interactions
 - **Hierarchy principle:** First pairwise interactions, then three-way interactions, then four-way interactions, ...
 - **Effect sparsity:** Three-way or higher order interactions contain only rarely a substantial predictive signal
 - **Heredity principle:** Only test higher order interactions, if all of the involved variables carry a robust signal



Demo & Hands-on Exercise



Can you predict the sale price of a house?



<https://www.kaggle.com/t/1d6895b250f0b775eb9216ed364bd855>

<https://www.kaggle.com/competitions/ml-for-research-2024-ames-housing>

kaggle

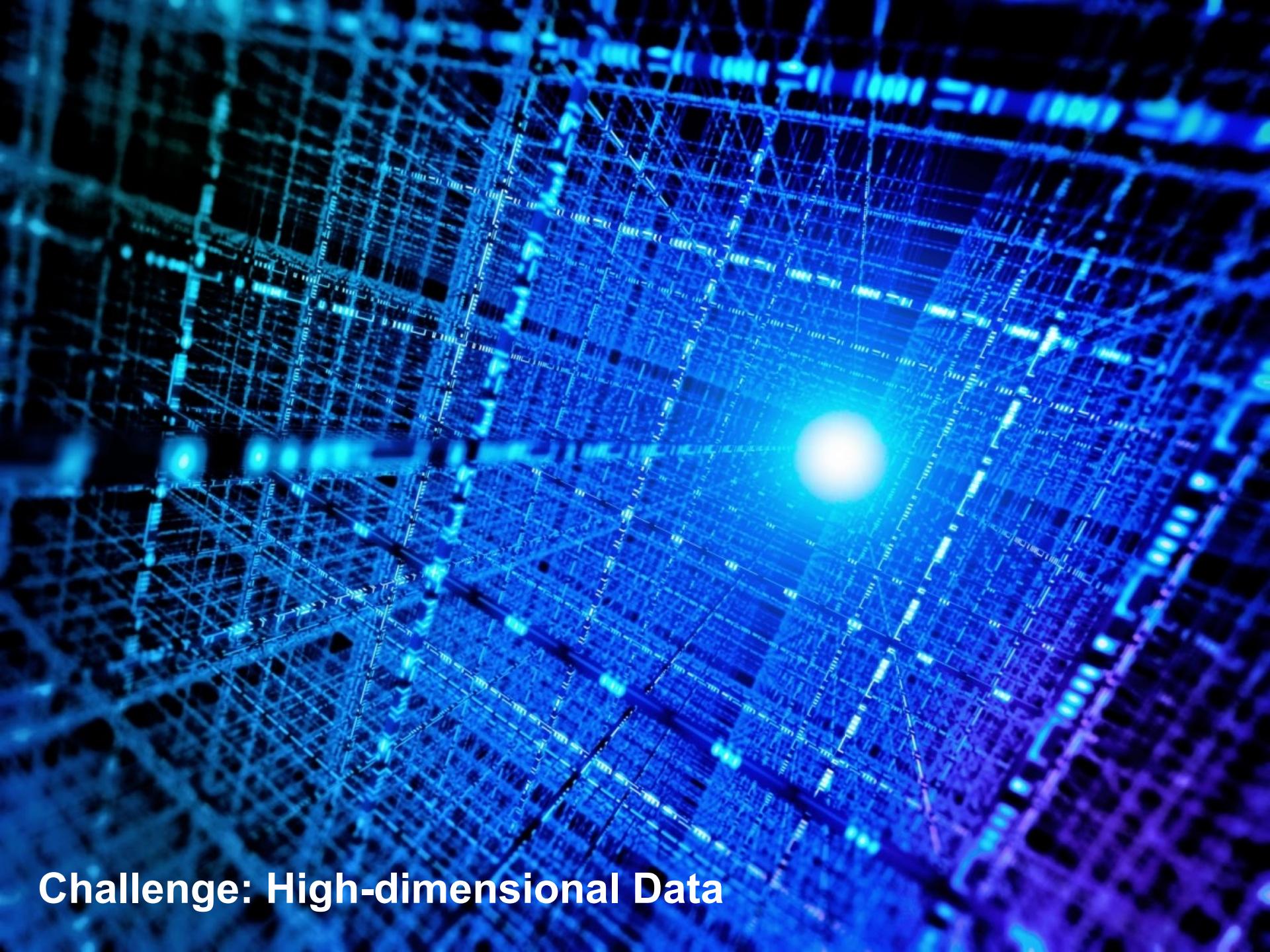
Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Supervised Learning:
_ Feature Selection, Regularization, and Splines

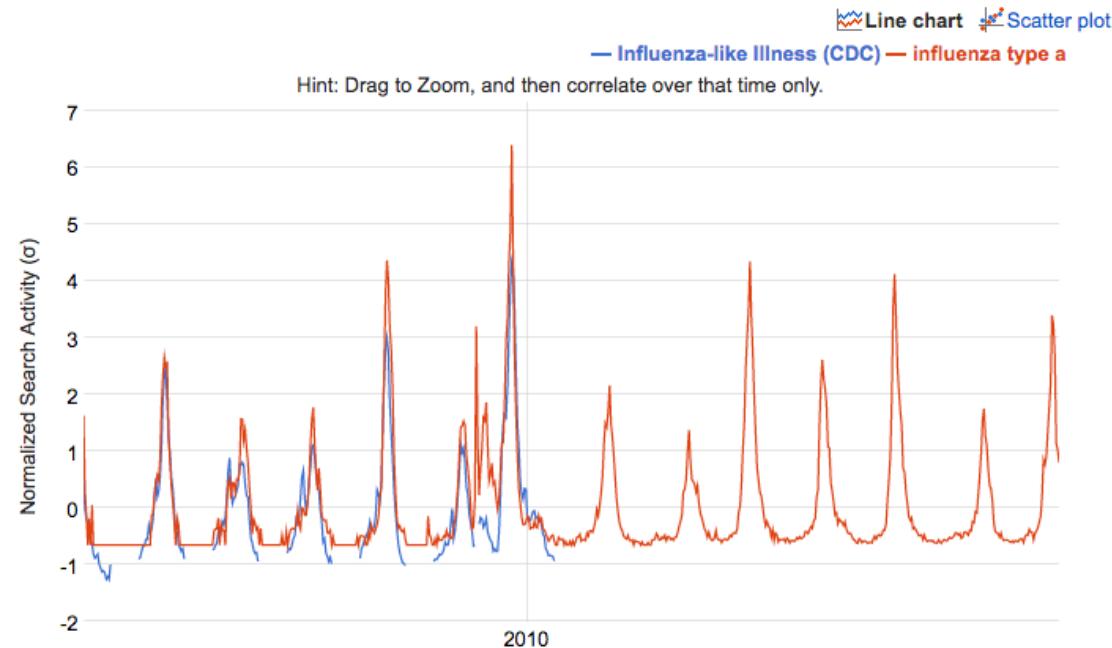


Challenge: High-dimensional Data

Feature Selection, Regularization, and Splines

Example: Relationship between Flu and Search Terms

User uploaded activity for Influenza-like Illness (CDC) and United States Web Search activity for influenza type a
 $r=0.9069$

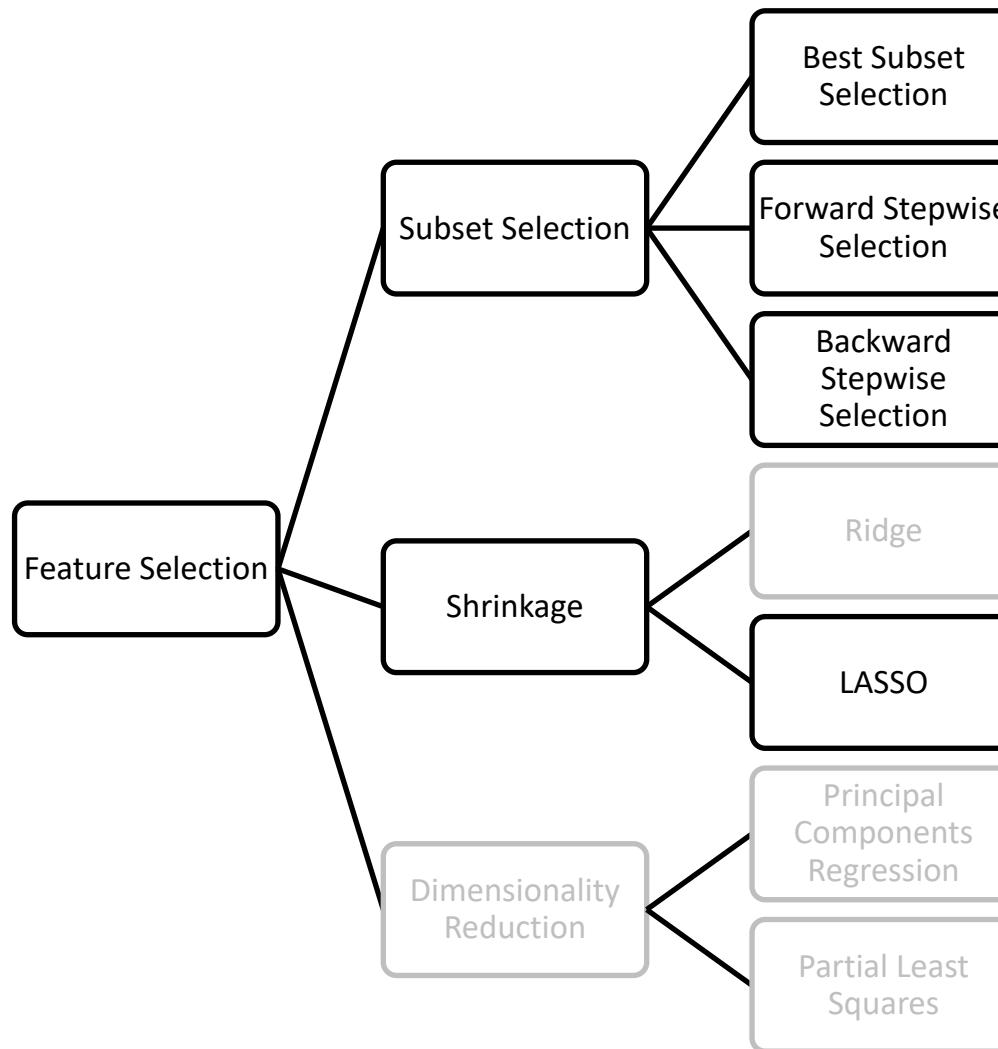


Correlated with Influenza-like Illness (CDC)

- 0.9069 influenza type a
- 0.9038 symptoms of flu
- 0.9033 flu duration
- 0.8919 flu contagious
- 0.8851 flu fever
- 0.8831 treat the flu
- 0.8830 how to treat the flu
- 0.8815 signs of the flu
- 0.8798 over the counter flu
- 0.8775 how long is the flu
- 0.8741 symptoms of the flu
- 0.8717 flu recovery
- 0.8711 flu medicine
- 0.8688 flu or cold
- 0.8602 is flu contagious
- 0.8582 how long does the flu
- 0.8576 treat flu
- 0.8484 is the flu contagious
- 0.8472 flu treatment
- 0.8454 flu vs cold
- 0.8439 how long is the flu contagious
- 0.8405 get over the flu
- 0.8387 treating flu
- 0.8387 flu vs. cold
- 0.8372 having the flu
- 0.8364 treatment for flu
- 0.8357 human temperature
- 0.8324 dangerous fever
- 0.8303 the flu
- 0.8294 remedies for flu

Feature Selection, Regularization, and Splines

Feature Selection and Regularization Approaches



Feature Selection, Regularization, and Splines

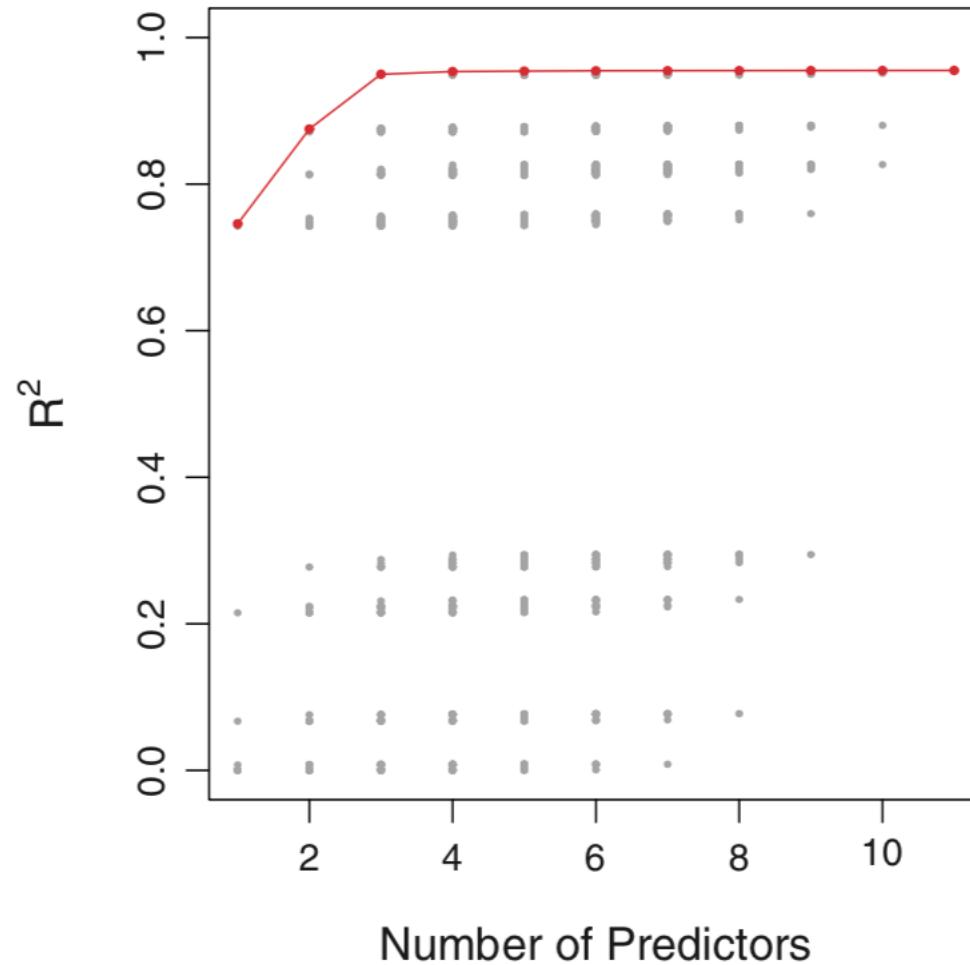
Best Subset Selection

Algorithm 6.1 Best subset selection

1. Let \mathcal{M}_0 denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.
 2. For $k = 1, 2, \dots, p$:
 - (a) Fit all $\binom{p}{k}$ models that contain exactly k predictors.
 - (b) Pick the best among these $\binom{p}{k}$ models, and call it \mathcal{M}_k . Here *best* is defined as having the smallest RSS, or equivalently largest R^2 .
 3. Select a single best model from among $\mathcal{M}_0, \dots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2 .
-

Feature Selection, Regularization, and Splines

Best Subset Selection



Feature Selection, Regularization, and Splines

Forward Stepwise Selection

Algorithm 6.2 *Forward stepwise selection*

1. Let \mathcal{M}_0 denote the *null* model, which contains no predictors.
 2. For $k = 0, \dots, p - 1$:
 - (a) Consider all $p - k$ models that augment the predictors in \mathcal{M}_k with one additional predictor.
 - (b) Choose the *best* among these $p - k$ models, and call it \mathcal{M}_{k+1} . Here *best* is defined as having smallest RSS or highest R^2 .
 3. Select a single best model from among $\mathcal{M}_0, \dots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2 .
-

Feature Selection, Regularization, and Splines

Forward Stepwise Selection



# Variables	Best subset	Forward stepwise
One	rating	rating
Two	rating, income	rating, income
Three	rating, income, student	rating, income, student
Four	cards, income, student, limit	rating, income, student, limit

Feature Selection, Regularization, and Splines

Backward Stepwise Selection

Algorithm 6.3 Backward stepwise selection

1. Let \mathcal{M}_p denote the *full* model, which contains all p predictors.
 2. For $k = p, p - 1, \dots, 1$:
 - (a) Consider all k models that contain all but one of the predictors in \mathcal{M}_k , for a total of $k - 1$ predictors.
 - (b) Choose the *best* among these k models, and call it \mathcal{M}_{k-1} . Here *best* is defined as having smallest RSS or highest R^2 .
 3. Select a single best model from among $\mathcal{M}_0, \dots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2 .
-

Can you predict the sales price of a house?



Feature Selection, Regularization, and Splines

Least Absolute Shrinkage and Selection Operator (LASSO)

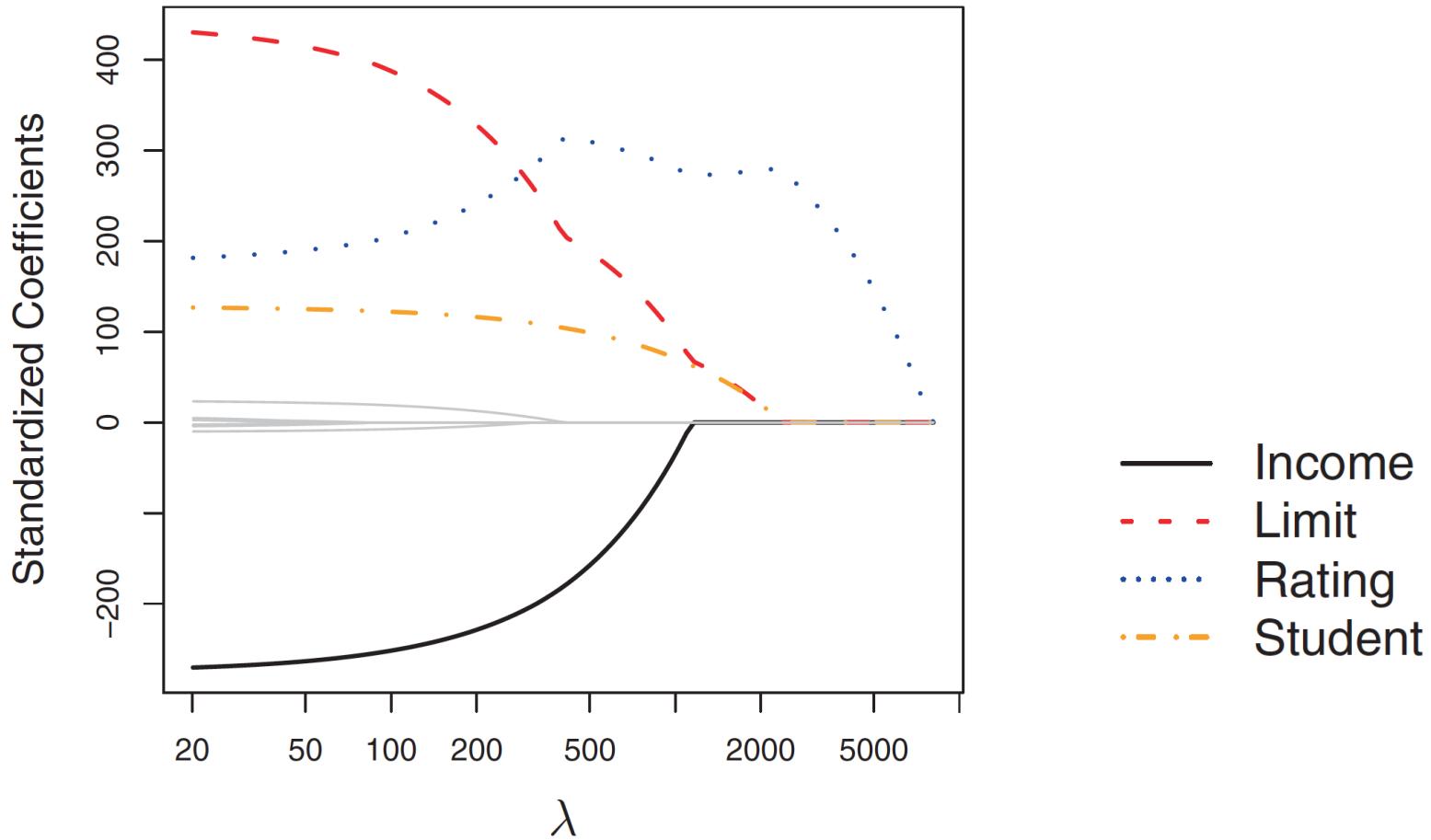
- Linear regression with variable selection
- To fit a model to a dataset, LASSO minimizes:

$$\text{RSS} + \lambda \sum_{j=1}^p |\beta_j|$$

- The penalty term has the effect of forcing some of the coefficient estimates to be exactly equal to 0 when the tuning parameter λ is sufficiently large.
- Hence, LASSO produces sparse models that are easier to interpret than linear models with many features.

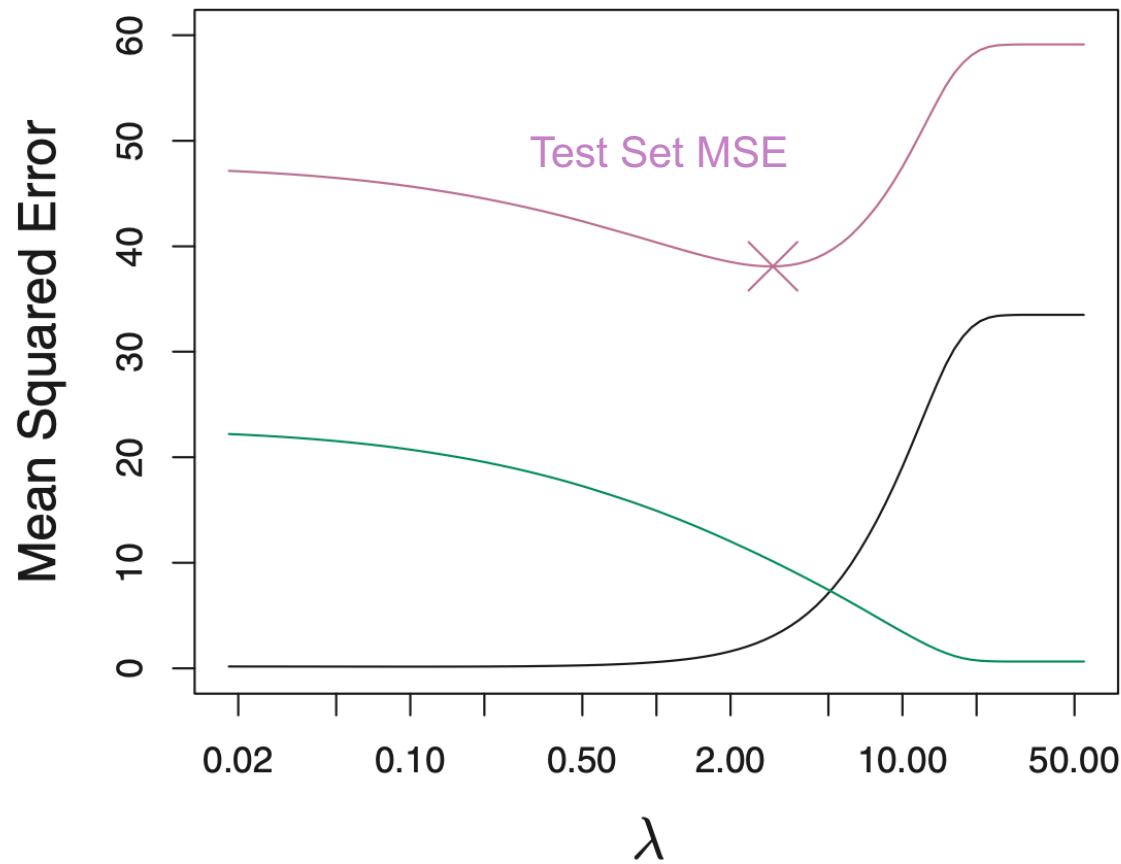
Feature Selection, Regularization, and Splines

Example: Lambda vs. Size of Standardized Coefficients



Feature Selection, Regularization, and Splines

Example: Lambda vs. Mean Squared Error



Can you predict the sales price of a house?



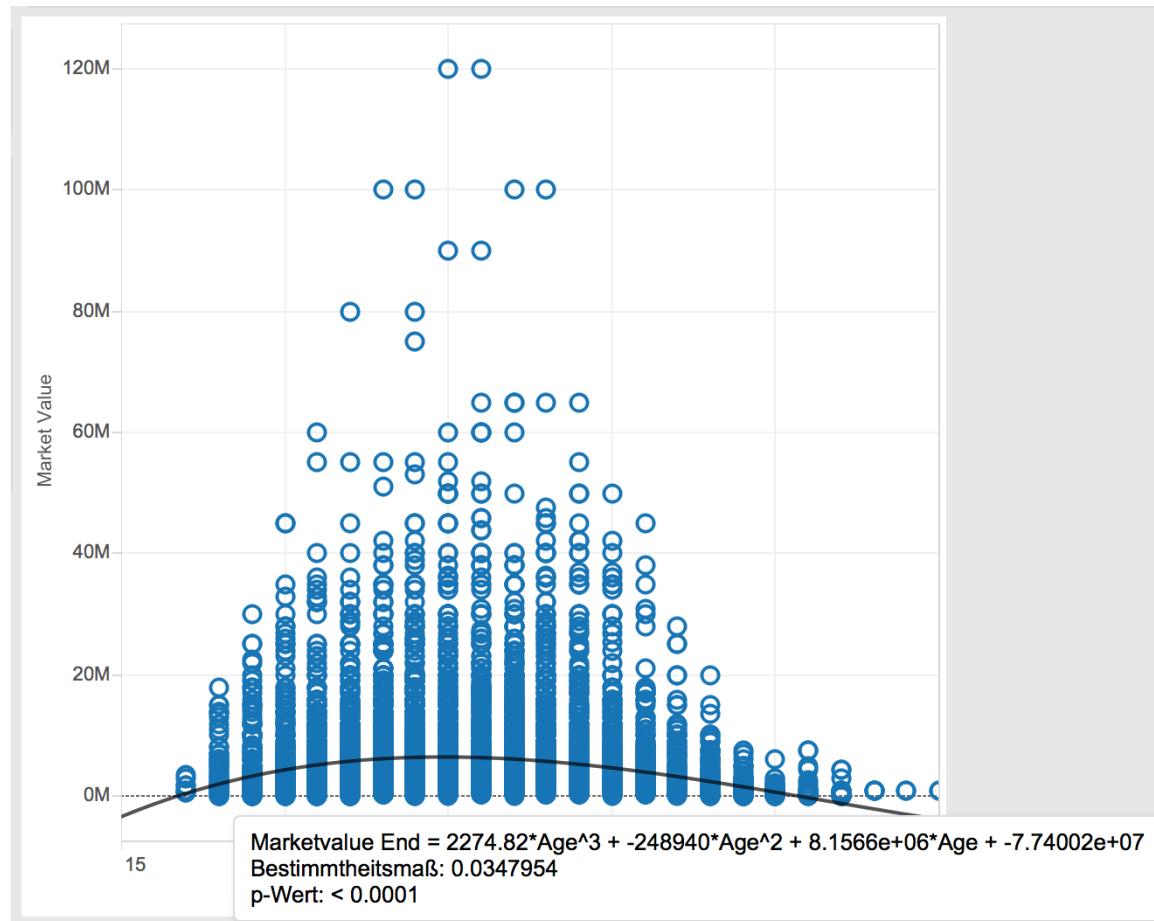


The background of the slide features a complex, abstract fractal pattern composed of numerous thin, translucent red and purple lines. These lines form intricate loops and swirls, creating a sense of depth and motion. In the center of the image, there is a distinct, larger circular pattern that resembles an eye or a flower, with a bright red center and several layers of radiating lines in shades of pink, purple, and red.

Challenge: Non-linear Relationships

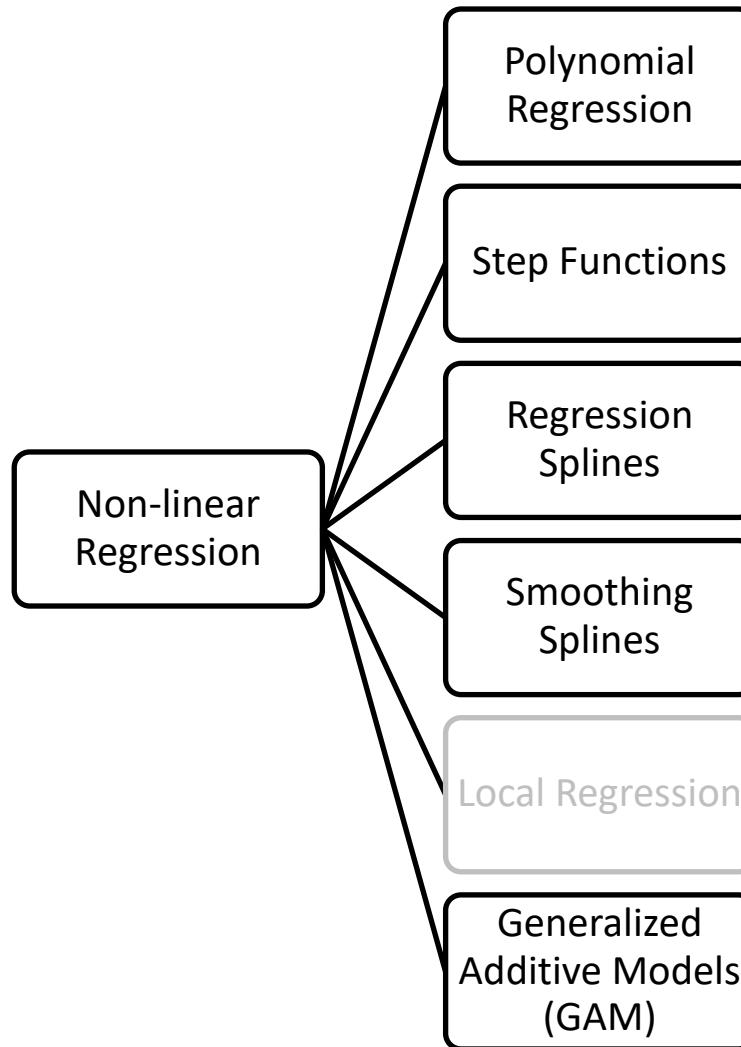
Feature Selection, Regularization, and Splines

Example: Relationship between Player Market Value and Age



Feature Selection, Regularization, and Splines

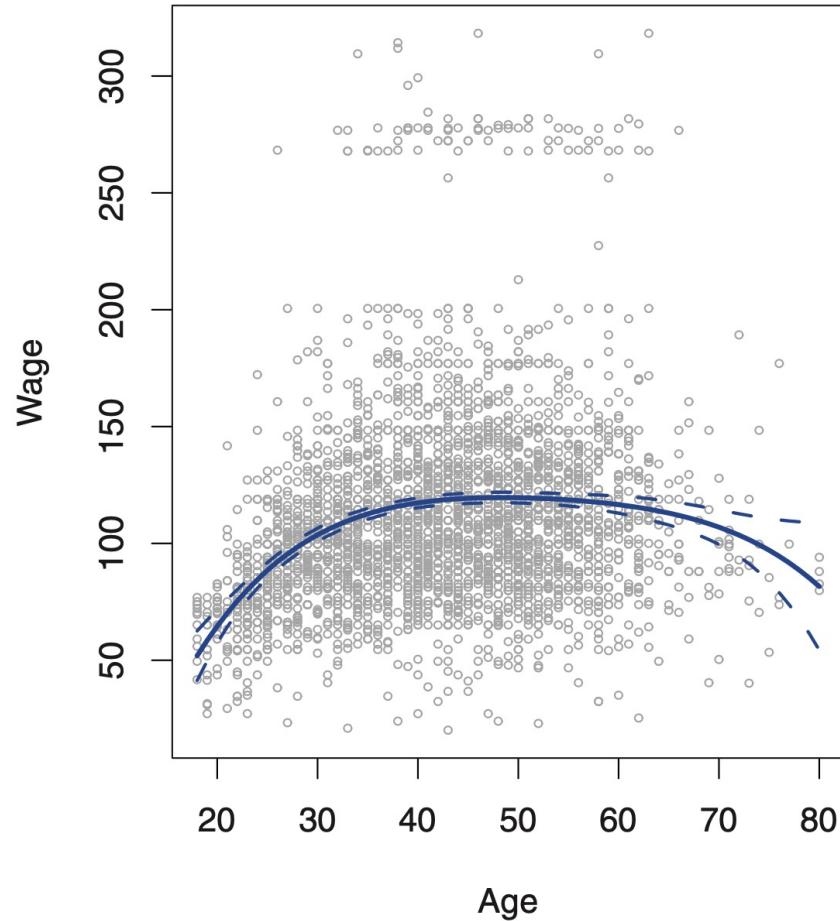
Approaches for Non-linear Regression



Feature Selection, Regularization, and Splines



Polynomial Regression

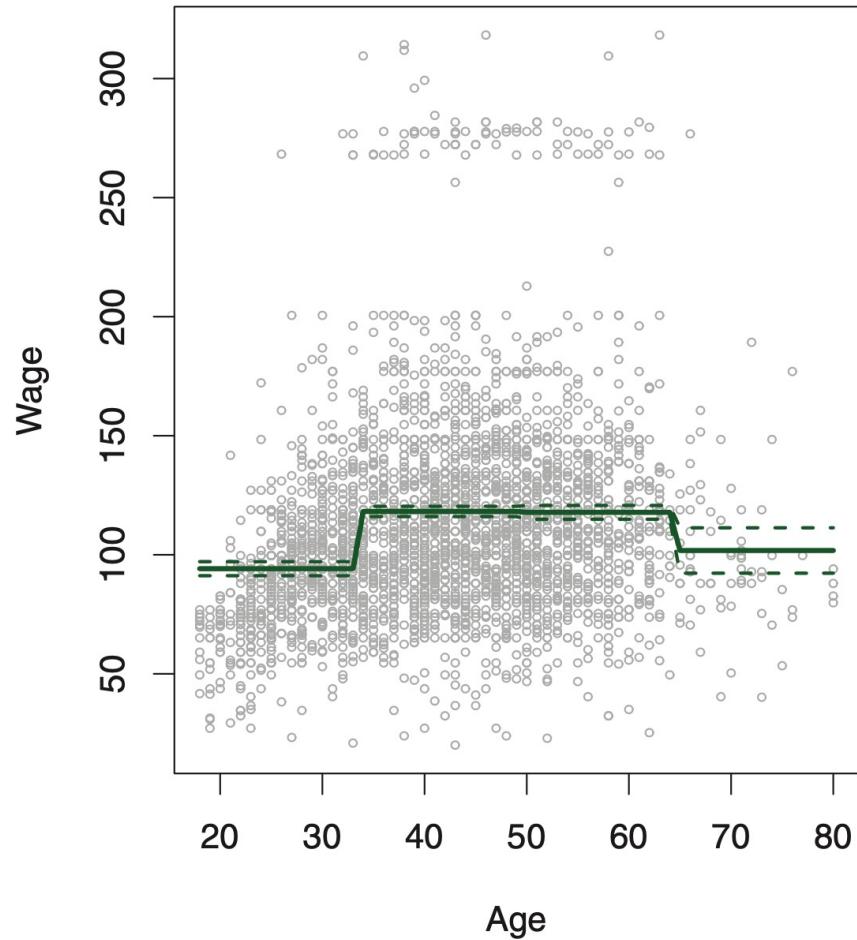


$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \dots + \beta_d x_i^d + \epsilon_i$$

Feature Selection, Regularization, and Splines



Step Functions



Feature Selection, Regularization, and Splines

Step Functions

- We break the range of X into bins and fit a different constant in each bin
- We create cut points (also called knots) c_1, c_2, \dots, c_K in the range of X , and then construct $K + 1$ new variables:

$$\begin{aligned} C_0(X) &= I(X < c_1), \\ C_1(X) &= I(c_1 \leq X < c_2), \\ C_2(X) &= I(c_2 \leq X < c_3), \\ &\vdots \\ C_{K-1}(X) &= I(c_{K-1} \leq X < c_K), \\ C_K(X) &= I(c_K \leq X), \end{aligned}$$

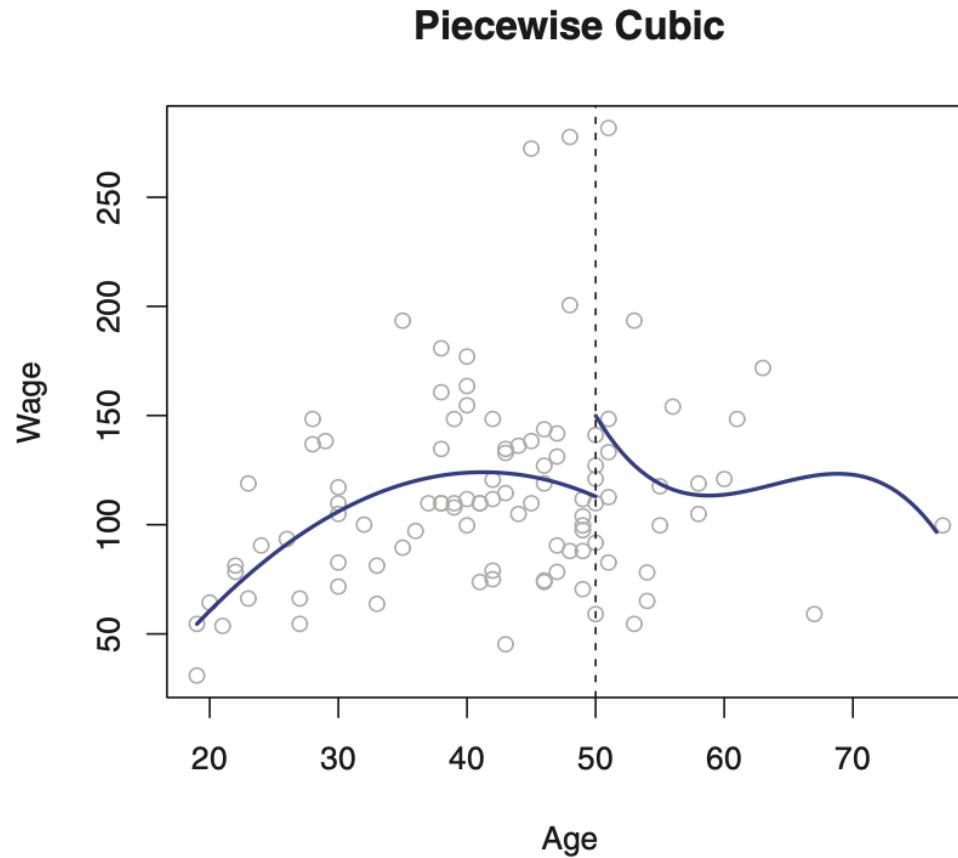
- $I(\cdot)$ is an indicator function that returns a 1 if the condition is true, and returns a 0 otherwise.
- We then use least squares to fit a linear model using $C_1(X), C_2(X), \dots, C_K(X)$ as predictors:

$$y_i = \beta_0 + \beta_1 C_1(x_i) + \beta_2 C_2(x_i) + \dots + \beta_K C_K(x_i) + \epsilon_i$$

Feature Selection, Regularization, and Splines



Piecewise Polynomials



Feature Selection, Regularization, and Splines

Piecewise Polynomials

- Piecewise polynomial regression fits separate low-degree polynomials over different regions of X.
- For example, a piecewise cubic polynomial works by fitting a cubic regression model of the form

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \epsilon_i,$$

- where the coefficients β_0 , β_1 , β_2 , and β_3 differ in different parts of the range of X.
- For example, piecewise cubic polynomial with a single knot at a point c takes the form:

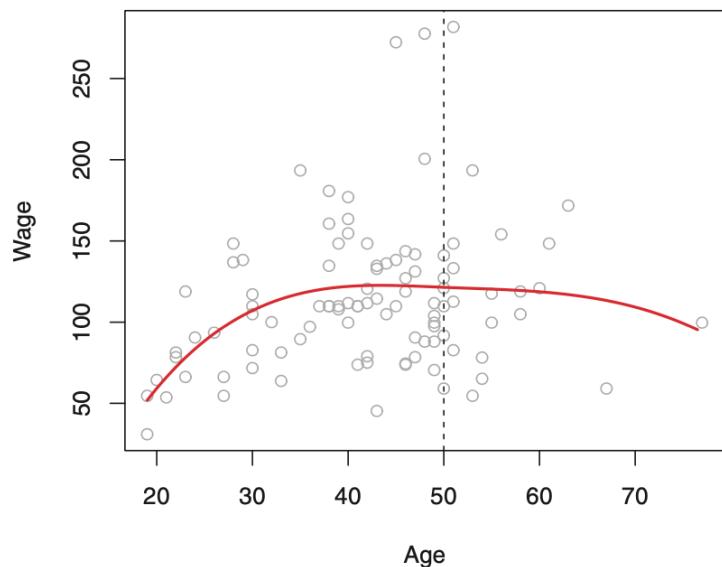
$$y_i = \begin{cases} \beta_{01} + \beta_{11}x_i + \beta_{21}x_i^2 + \beta_{31}x_i^3 + \epsilon_i & \text{if } x_i < c; \\ \beta_{02} + \beta_{12}x_i + \beta_{22}x_i^2 + \beta_{32}x_i^3 + \epsilon_i & \text{if } x_i \geq c. \end{cases}$$

Feature Selection, Regularization, and Splines

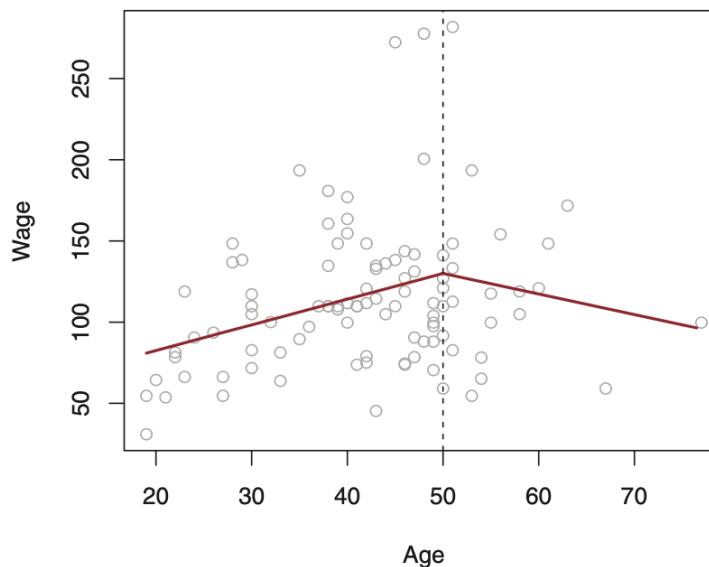
Splines

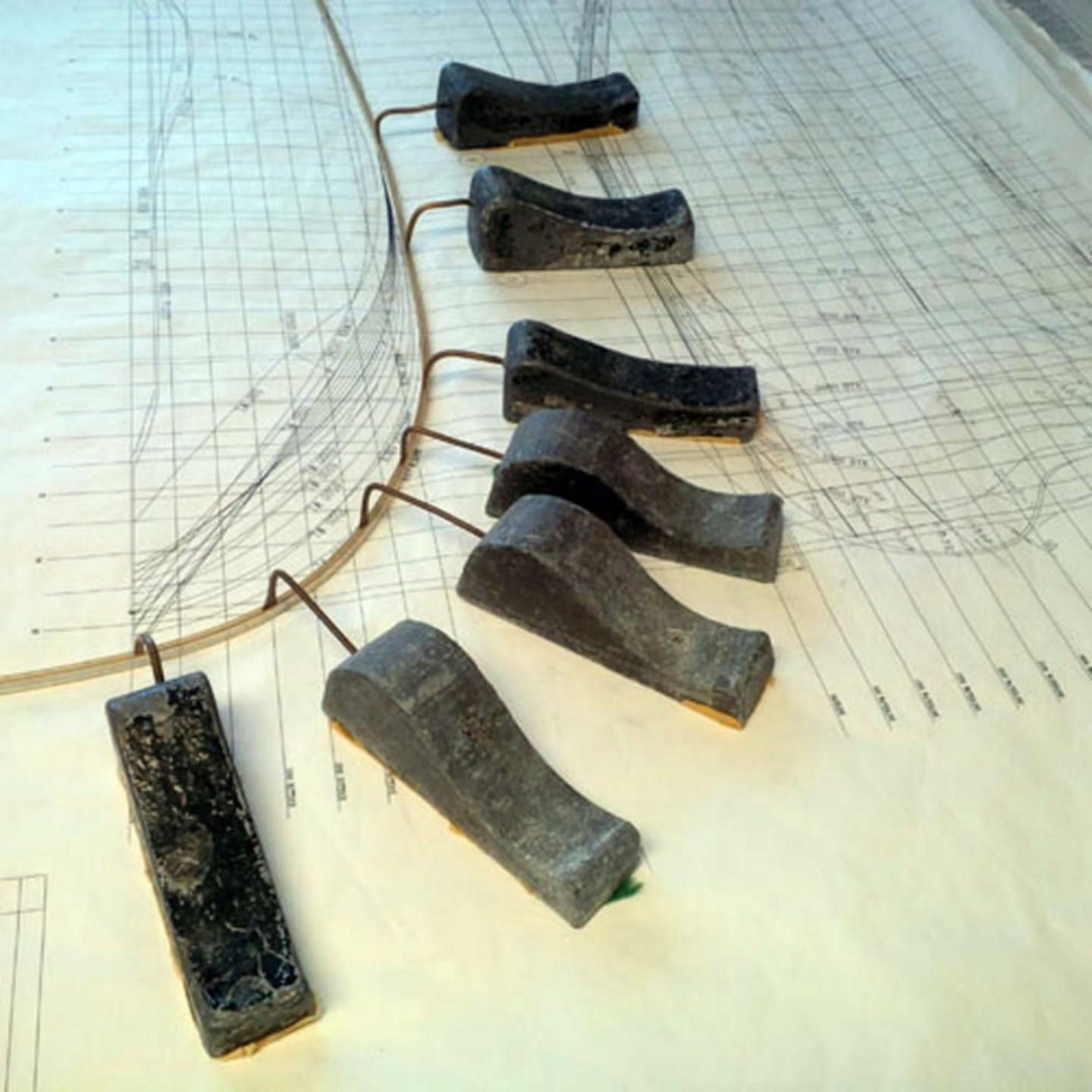


Cubic Spline



Linear Spline





Feature Selection, Regularization, and Splines

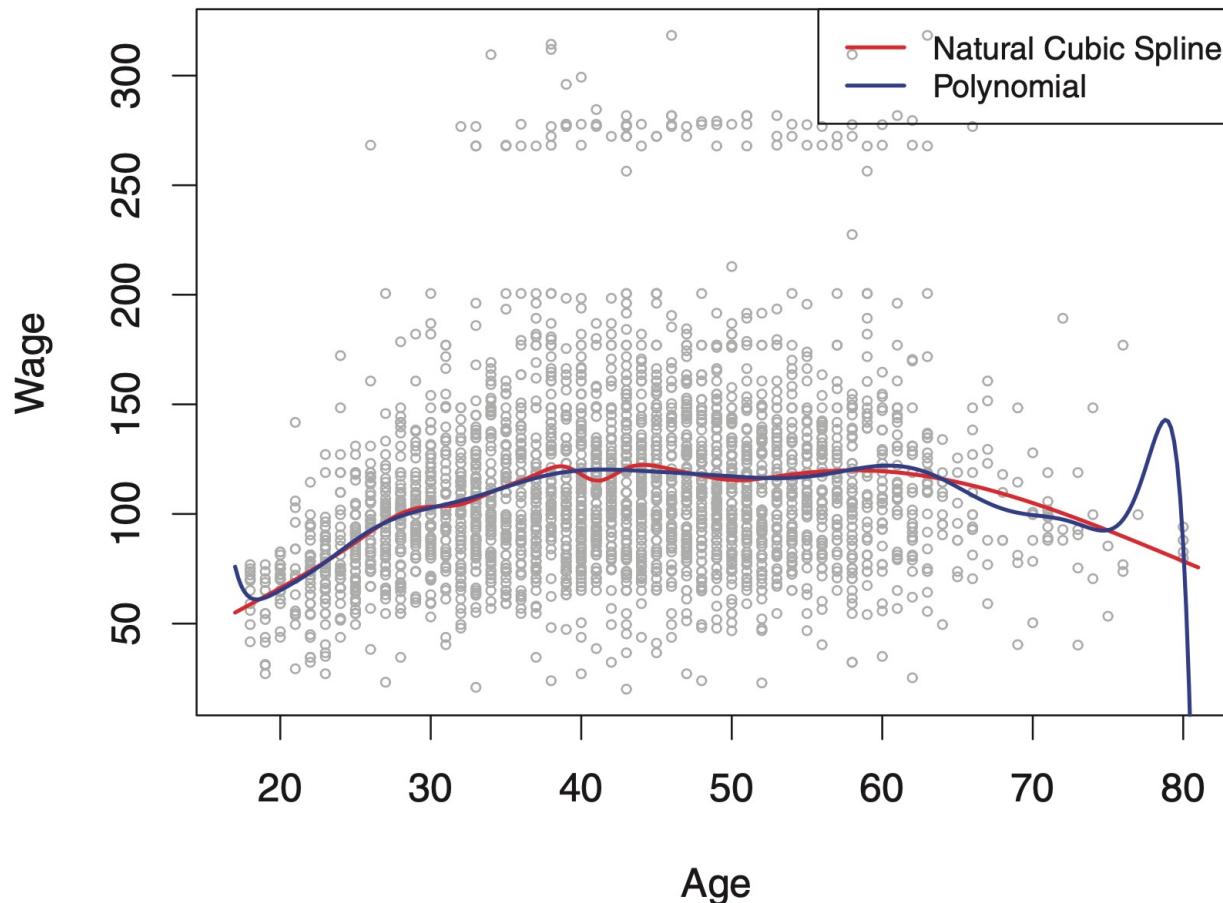
Splines

- The piecewise cubic regression in the example contains a “jump” at the cut point (knot), which is undesirable in most use cases.
- Splines are a way to remedy this problem by fitting a piecewise polynomial under the **constraint** that the fitted curve **must be continuous** (there cannot be “jumps” in the function).
- The cubic spline in the example contains two constraints to make the function “smooth”:
 1. The first derivative of the piecewise polynomials (i.e., the slope of the function) is continuous at the cut point.
 2. The second derivative of the piecewise polynomials (i.e., the change in the slope of the function) is continuous at the cut point.
- A **natural spline** has one more constraint: Its function is required to be **linear at the boundaries** (i.e., where X is smaller than the smallest knot or larger than the largest knot).

Feature Selection, Regularization, and Splines



Example: Natural cubic spline vs. 15-degree polynomial regression



Feature Selection, Regularization, and Splines

Choosing the Number and Locations of the Knots

- Where should we place the knots?
 - One option is to place more knots in regions where we think that the function is very variable, and less knots in regions where the function seems to be stable.
 - Another option is to predefined a number of knots and distribute them uniformly over the range of possible feature values.
- How many knots should we use?
 - Try different numbers and see which produces the “best looking” curve.
 - Try different numbers and see which number leads to the best predictive performance.

Feature Selection, Regularization, and Splines

Smoothing Splines

- Smoothing splines represent a different approach to producing splines, which does not require to explicitly choose knots.
- The idea is to find a **non-linear function $g(x)$** which **fits the data well**, but is at the same time **not too flexible** (i.e., it is smooth).
- This can be achieved by minimizing the **loss function**:

$$\sum_{i=1}^n (y_i - g(x_i))^2 + \lambda \int g''(t)^2 dt$$

- where λ is a non-negative tuning parameter.

Feature Selection, Regularization, and Splines

Smoothing Splines

Loss

Measures the
goodness of fit
function g

$$\sum_{i=1}^n (y_i - g(x_i))^2 + \lambda \int g''(t)^2 dt$$

Penalty

Measures the total
change in the slope of
function g over its entire
range (“wiggliness”)

Feature Selection, Regularization, and Splines

Smoothing Splines

- The function $g(x)$ that minimizes the above loss+penalty function is a
 - piecewise cubic polynomial with knots at every unique value of x_1, \dots, x_n , and
 - continuous first and second derivatives at each knot.
 - Furthermore, it is linear in the region outside of the extreme knots.
- In other words, it is a **natural cubic spline with knots at x_1, \dots, x_n !**

Feature Selection, Regularization, and Splines

Generalized Additive Model (GAM)

- The presented non-linear approaches allow predicting a response Y on the basis of a *single* predictor X.
- GAMs provide a general framework for extending the standard linear model by **allowing non-linear functions of multiple variables**, while **maintaining additivity**.
- More formally, a GAM is specified as

$$y_i = \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + \cdots + f_p(x_{ip}) + \epsilon_i$$

- where $f_j(x_{ij})$ is a non-linear function (e.g., step function, piecewise polynomial, natural spline).

Feature Selection, Regularization, and Splines



Example of a GAM

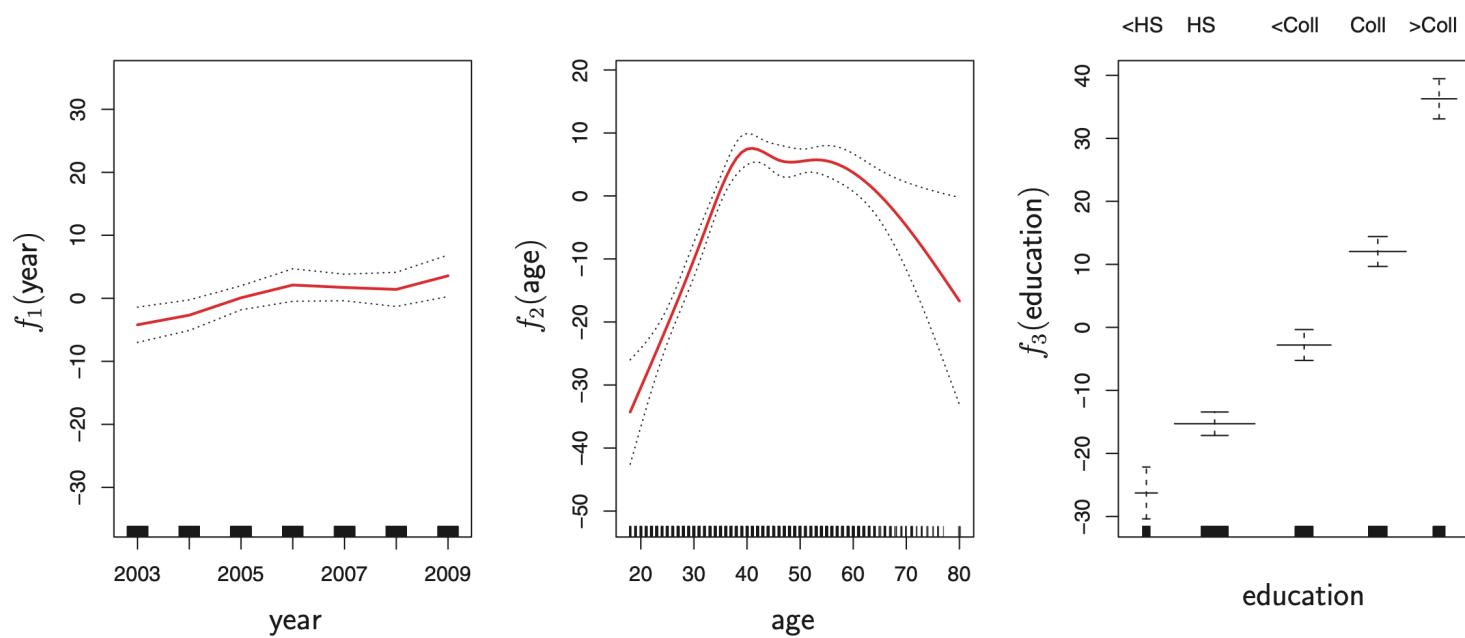


FIGURE 7.11. For the `Wage` data, plots of the relationship between each feature and the response, `wage`, in the fitted model (7.16). Each plot displays the fitted function and pointwise standard errors. The first two functions are natural splines in `year` and `age`, with four and five degrees of freedom, respectively. The third function is a step function, fit to the qualitative variable `education`.

Hands-on Exercise

Can you predict the sales price of a house?





Syllabus

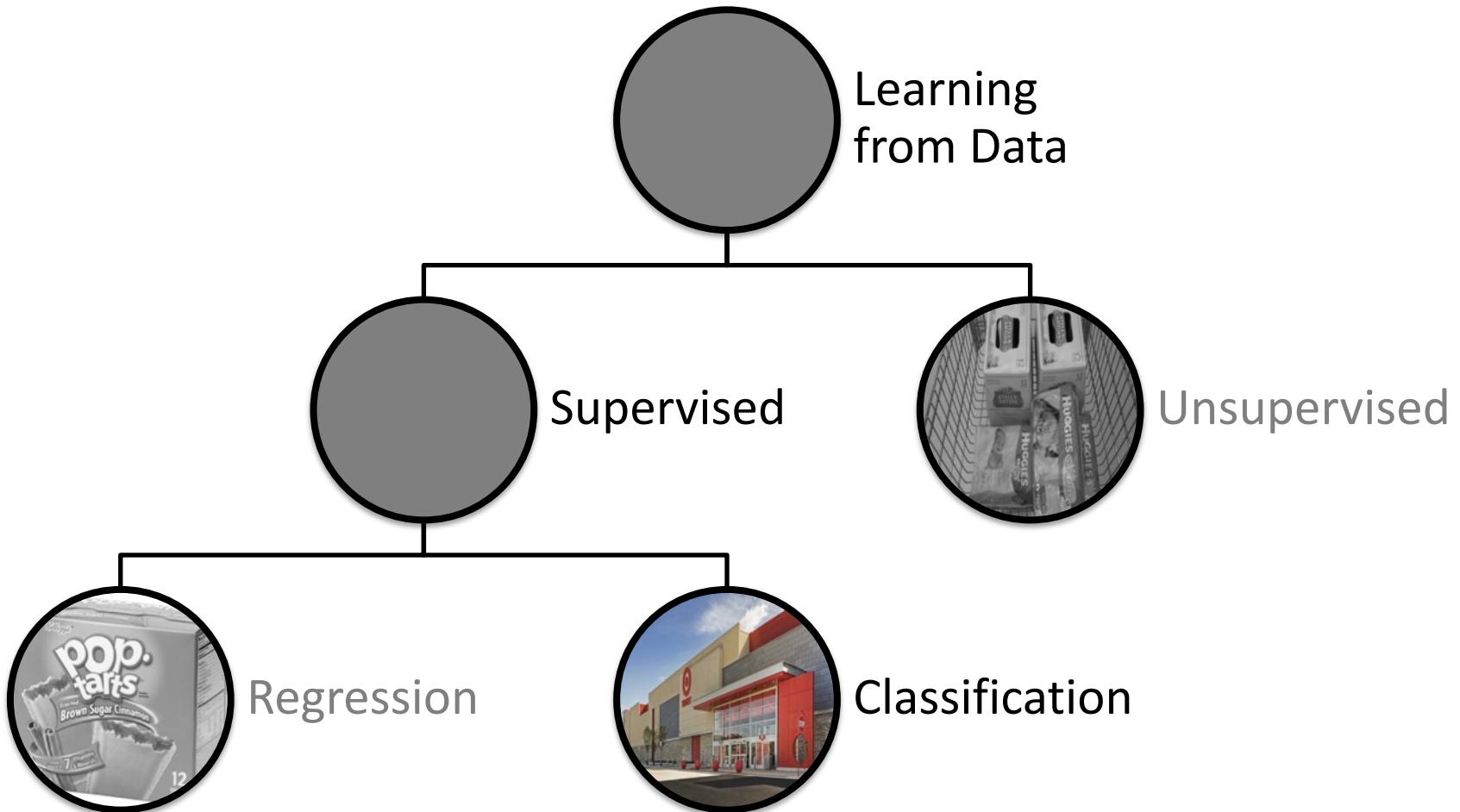
Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

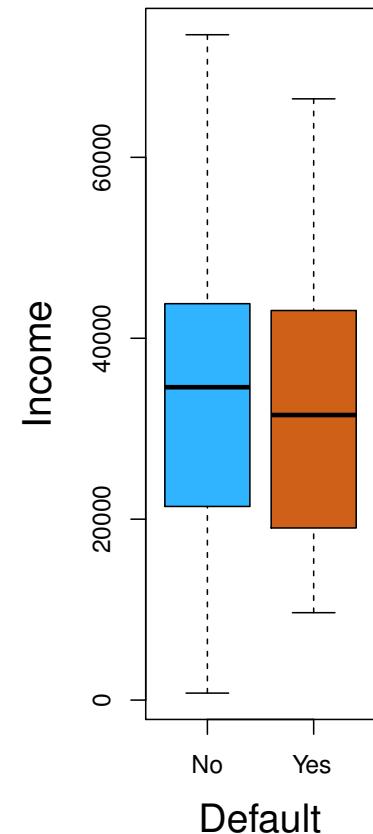
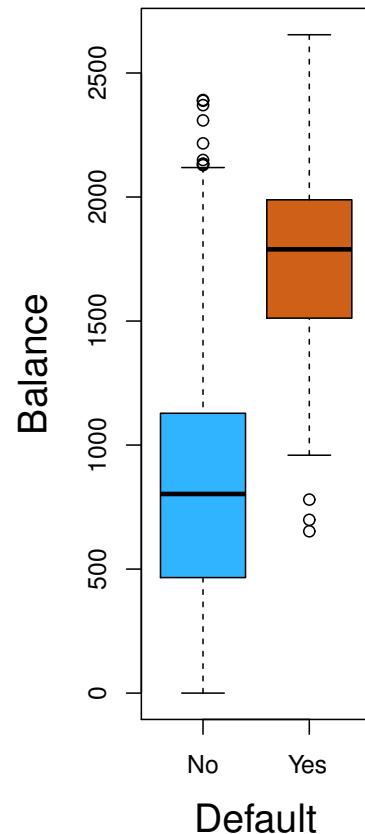
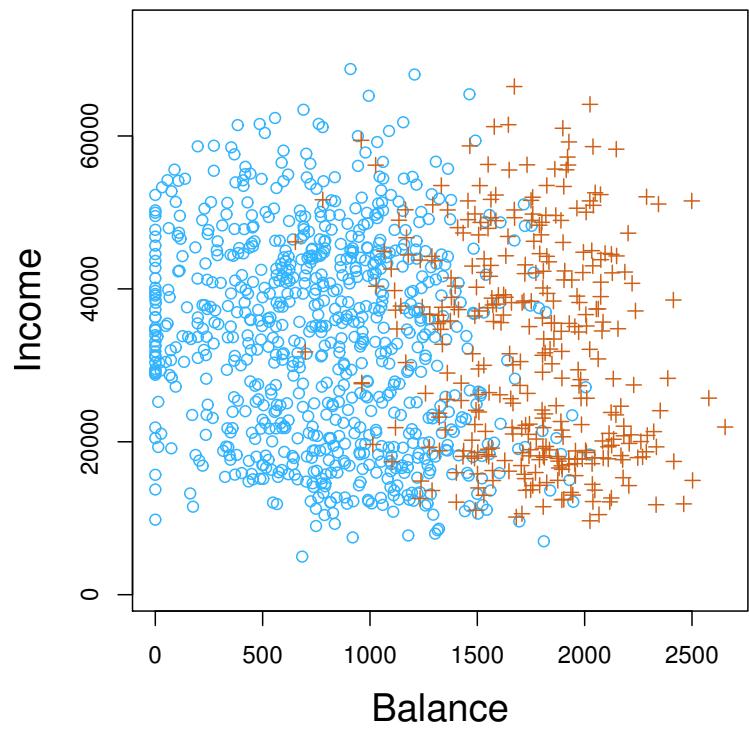
Supervised Learning:
_ Logistic Regression

Logistic Regression



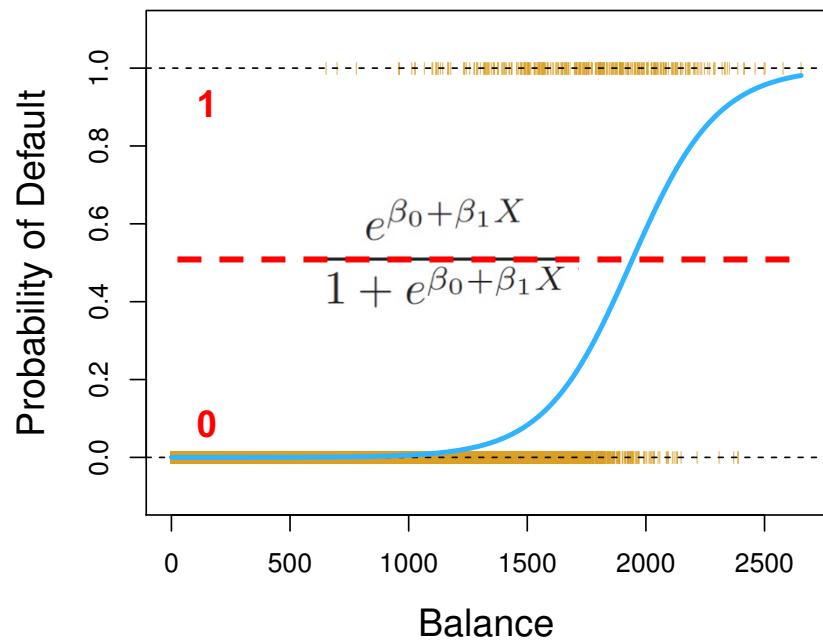
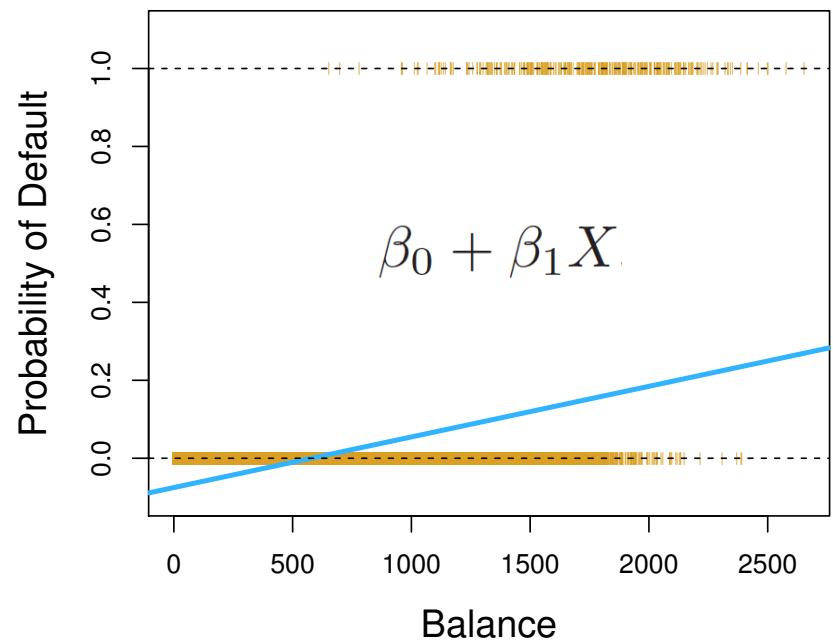
Logistic Regression

Example: Credit Card Default



Logistic Regression

From Linear Regression to Logistic Regression



Logistic Regression

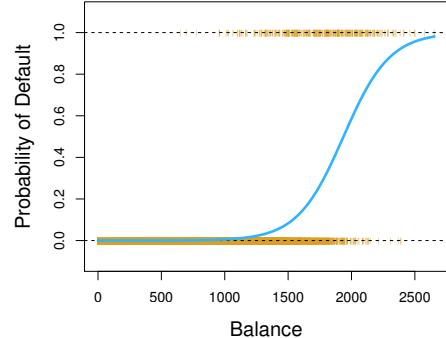
The Logistic Model

- The logistic function

$$p(X) = \Pr(Y = 1|X) = \frac{e^{\beta_0 + \beta_1 X}}{1 + e^{\beta_0 + \beta_1 X}}$$

- Log-Odds (or logit)

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X$$



Logistic Regression

Log-Odds

$$\log \left(\frac{p(X)}{1 - p(X)} \right) = \beta_0 + \beta_1 X$$

p	odds	logodds
.001	.001001	-6.906755
.01	.010101	-4.59512
.15	.1764706	-1.734601
.2	.25	-1.386294
.25	.3333333	-1.098612
.3	.4285714	-.8472978
.35	.5384616	-.6190392
.4	.6666667	-.4054651
.45	.8181818	-.2006707
.5	1	0
.55	1.222222	.2006707
.6	1.5	.4054651
.65	1.857143	.6190392
.7	2.333333	.8472978
.75	3	1.098612
.8	4	1.386294
.85	5.666667	1.734601
.9	9	2.197225
.999	999	6.906755
.9999	9999	9.21024

Logistic Regression



Example: Credit Card Default

	Coefficient	Std. error	Z-statistic	P-value
Intercept	−10.6513	0.3612	−29.5	<0.0001
balance	0.0055	0.0002	24.9	<0.0001

Measured in log-odds

Logistic Regression



Example: Credit Card Default

	Coefficient	Std. error	Z-statistic	P-value
Intercept	-3.5041	0.0707	-49.55	<0.0001
student [Yes]	0.4049	0.1150	3.52	0.0004

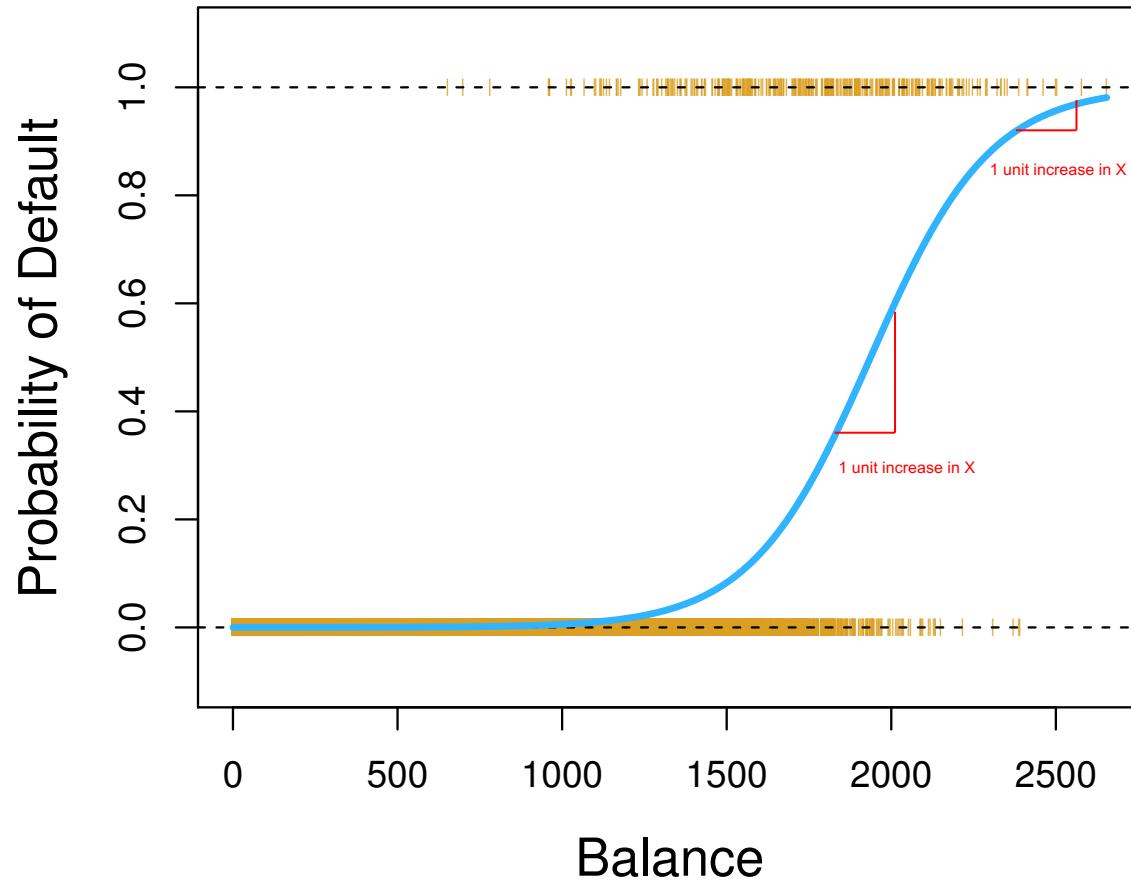
Measured in log-odds



Logistic Regression



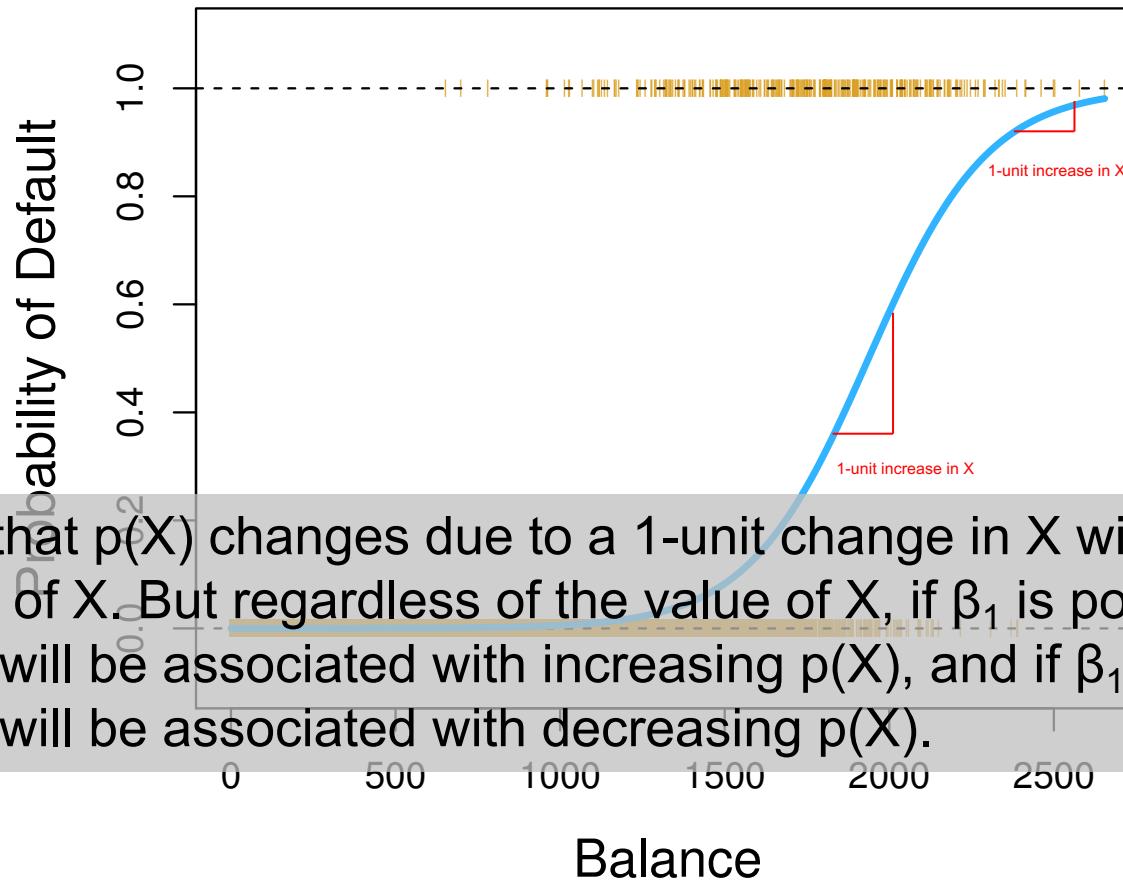
How Do I Interpret the Coefficients?



Logistic Regression



How Do I Interpret the Coefficients?



Logistic Regression



Making Predictions

	Coefficient	Std. error	Z-statistic	P-value
Intercept	-10.6513	0.3612	-29.5	<0.0001
balance	0.0055	0.0002	24.9	<0.0001

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}} = \frac{e^{-10.6513 + 0.0055 \times 1,000}}{1 + e^{-10.6513 + 0.0055 \times 1,000}} = 0.00576$$

Measured in percent

Multiple Logistic Regression

$$\log \left(\frac{p(X)}{1 - p(X)} \right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Logistic Regression



Example: Credit Card Default

	Coefficient	Std. error	Z-statistic	P-value
Intercept	-10.8690	0.4923	-22.08	<0.0001
balance	0.0057	0.0002	24.74	<0.0001
income	0.0030	0.0082	0.37	0.7115
student [Yes]	-0.6468	0.2362	-2.74	0.0062

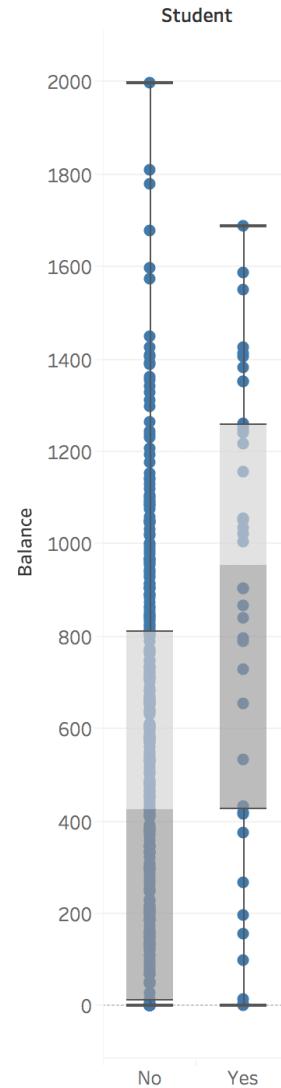
Why did the coefficient of student[Yes] change from positive to negative?

Logistic Regression



Association between Student and Balance

Even though an individual student with a given credit card balance will tend to have a lower probability of default than a non-student with the same credit card balance, the fact that students on the whole tend to have higher credit card balances means that overall, students tend to default at a higher rate than non-students.



Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Supervised Learning:
_ Evaluating Model Accuracy

Evaluating Model Accuracy

Naïve Approach

Dataset

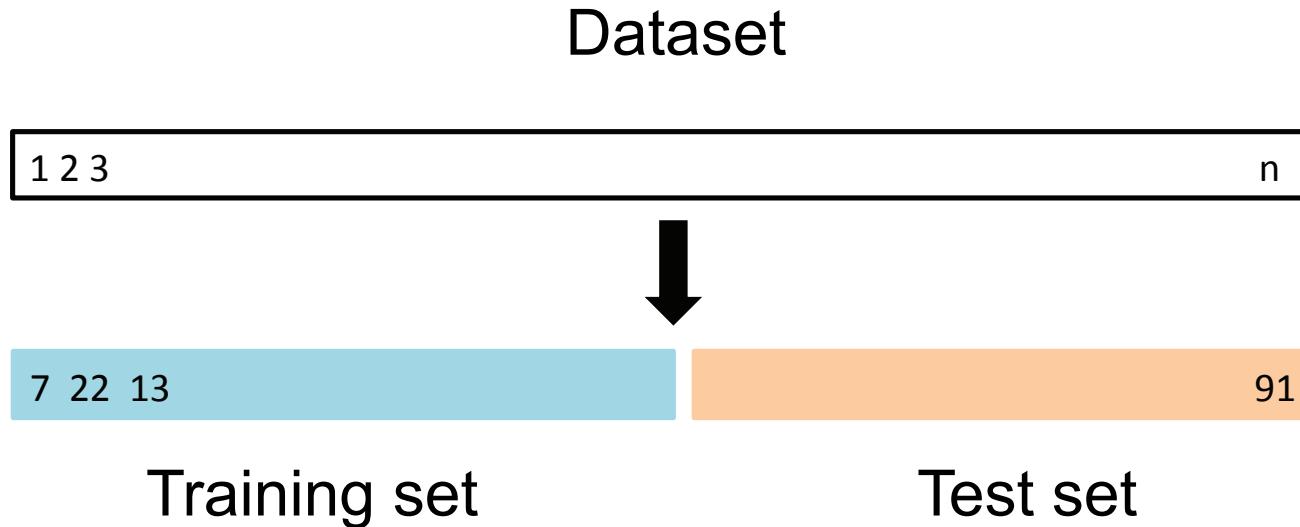
1 2 3

n

Train and test on the same dataset

Evaluating Model Accuracy

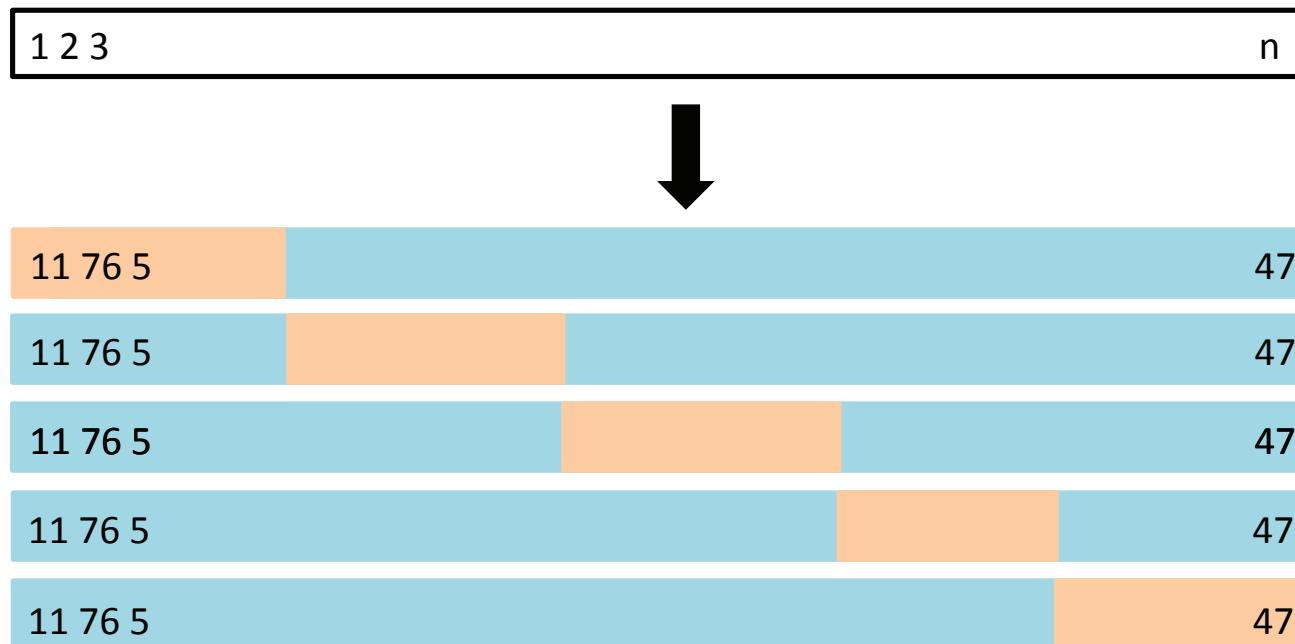
Train/Test Set Approach



Evaluating Model Accuracy

K-fold Cross Validation

Example: 5-fold CV



$$CV_{(n)} = \frac{1}{n} \sum_{i=1}^n Err_i$$

Evaluating Model Accuracy

Metrics for Evaluating the Accuracy of Regression Models

- Without separate test set (i.e., using the same dataset)
 - R^2
- With test set
 - Mean Absolute Error (MAE)
 - Mean Squared Error (MSE)
 - Root Mean Squared Error (RMSE)
 - Mean Absolute Percentage Error (MAPE)

Evaluating Model Accuracy

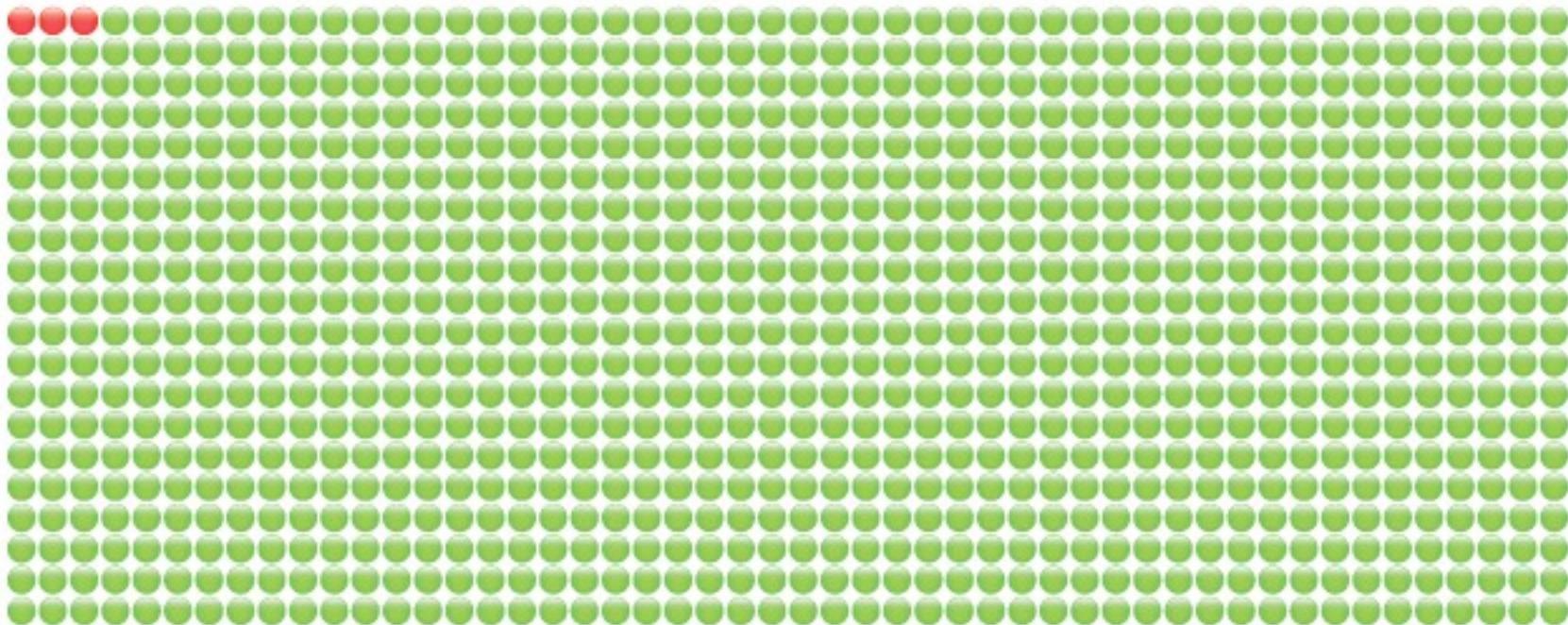
Metrics for Evaluating the Accuracy of Classification Models

$$\text{accuracy} = \frac{\text{Number of correct decisions made}}{\text{Total number of decisions made}}$$

Evaluating Model Accuracy

Metrics for Evaluating the Accuracy of Classification Models

Be careful when using classification error as a metric when working with unbalanced datasets!



Evaluating Model Accuracy

Metrics for Evaluating the Accuracy of Classification Models

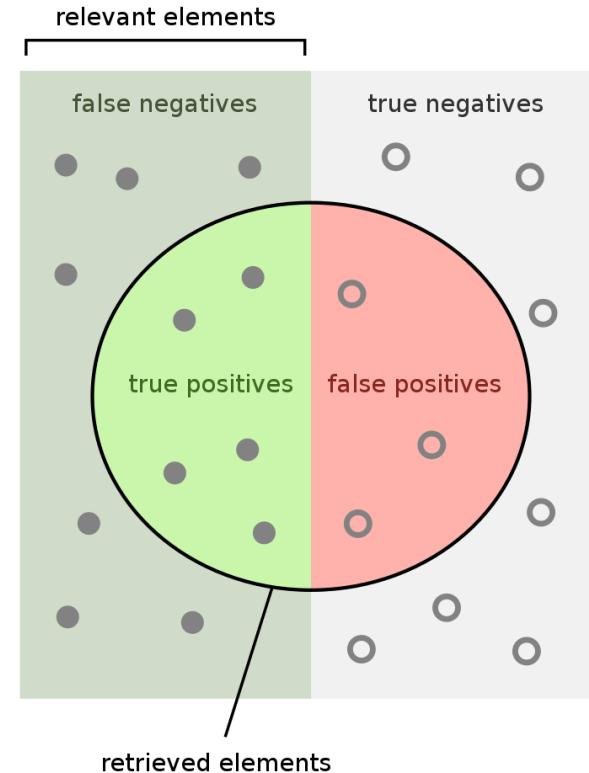
		<i>Predicted class</i>		
		– or Null	+ or Non-null	Total
<i>True class</i>	– or Null	True Neg. (TN)	False Pos. (FP)	N
	+ or Non-null	False Neg. (FN)	True Pos. (TP)	P
Total		N*	P*	

- **Sensitivity:** Percentage of positives that are identified (TP/P)
- **Specificity:** Percentage of negatives that are identified (TN/N)

Evaluating Model Accuracy

Metrics for Evaluating the Accuracy of Classification Models

- **Precision** is the fraction of relevant instances among the retrieved instances.
- **Recall** is the fraction of relevant instances that were retrieved.
- **F1 score** is the harmonic mean of recall and precision
(i.e., $2 * \frac{precision * recall}{precision + recall}$).



How many retrieved items are relevant?

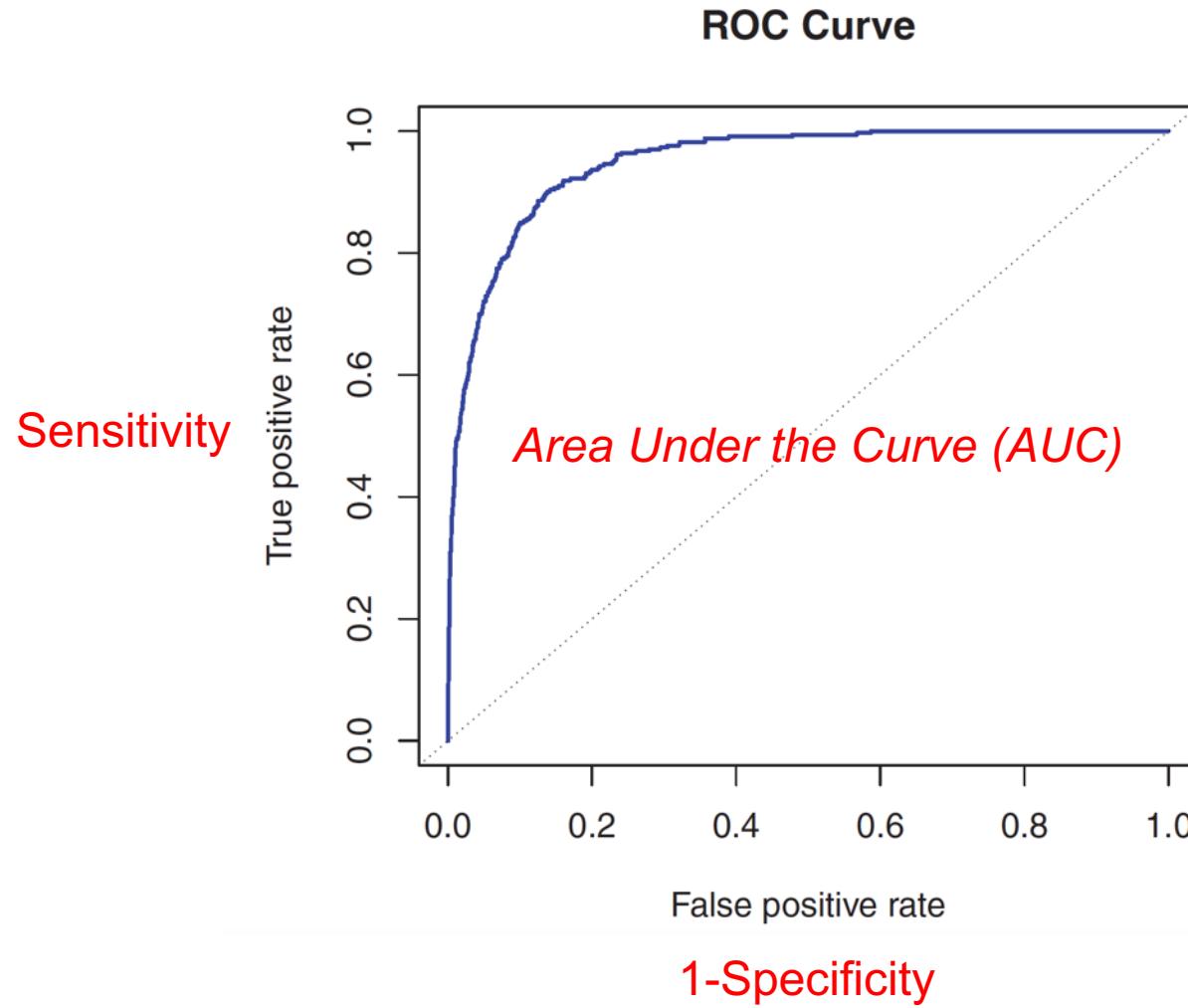
$$\text{Precision} = \frac{\text{green}}{\text{green} + \text{red}}$$

How many relevant items are retrieved?

$$\text{Recall} = \frac{\text{green}}{\text{green} + \text{grey}}$$

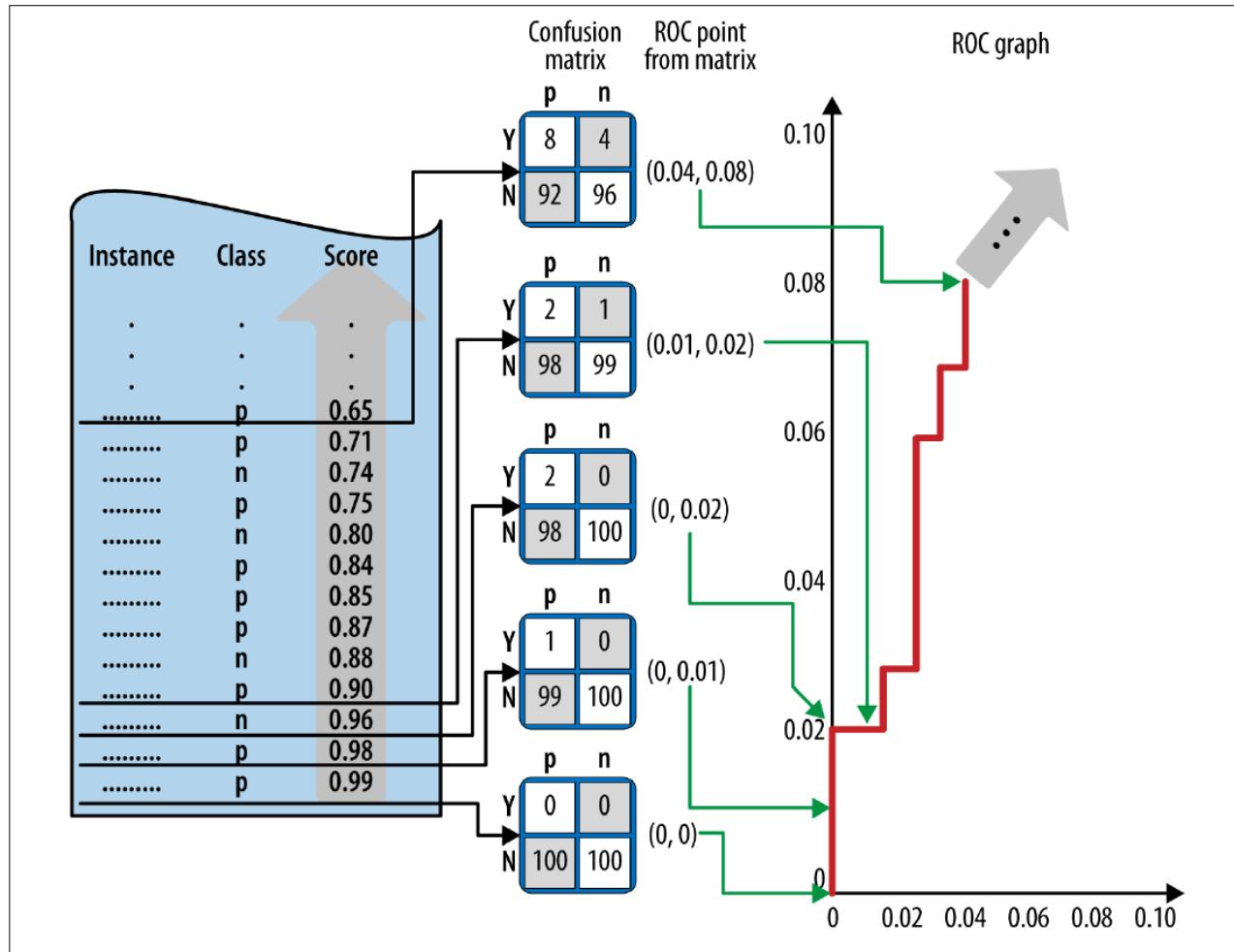
Evaluating Model Accuracy

Metrics for Evaluating the Accuracy of Classification Models



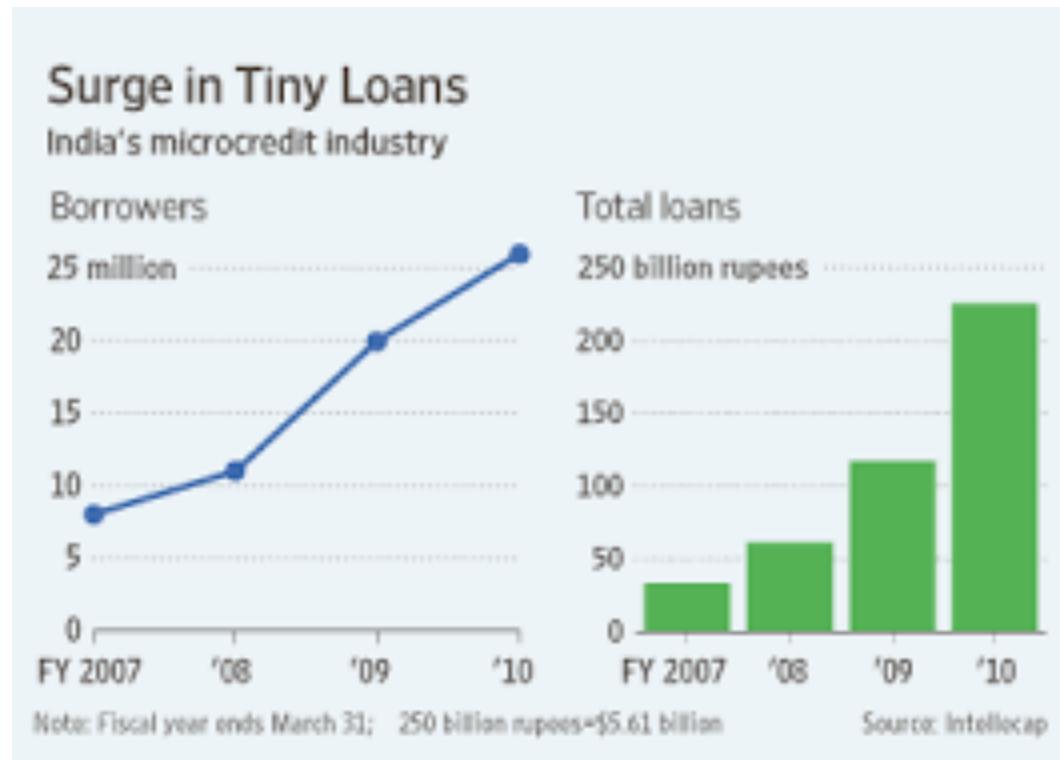
Evaluating Model Accuracy

Intuition behind the ROC curve



Hands-on Exercise

Micro Loans in India



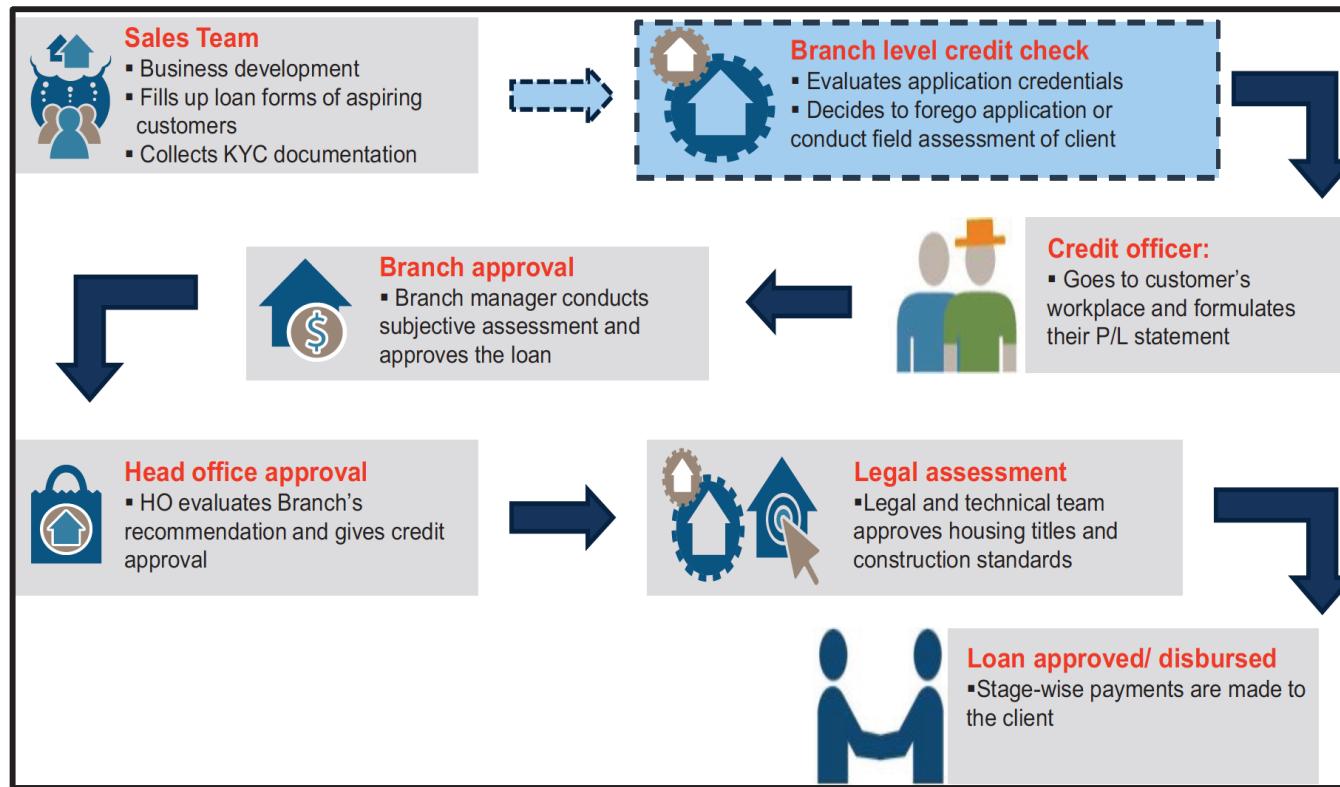
Setting

- In India, there are about 20 million home loan (mortgage) aspirants working in the informal sector
 - Monthly income between INR 20,000-25,000 (\$ 325-400)
 - Typically no formal accounts and documents (e.g., tax returns, income proofs, bank statements)
 - Often use services of money lenders with interest rates between 30 and 60% per annum
- Providing micro mortgages to this group of customers requires to quickly and efficiently assess their creditworthiness

Hands-on Exercise

As-Is Process

Shubham's Loan Approval Process



Strength and Weaknesses

- **Strength** of the current process
 - Interview-based field assessment
 - Relaxation of document requirements
- **Weaknesses** of the current process
 - Costly (total transaction costs as high as 30% of loan volume)
 - Subjective judgments; depends on individual skills and motivations
 - Low reliability across branches and credit officers
 - Risk of corruption and fraud

Hands-on Exercise

Data

Variable	Definition
ID	Unique Identifier for each application
Decision	Credit decision taken for the applicant 1 = Sanction, 0 = Reject
Build_Selfcon	Variable to indicate whether applicant seeks a home loan for self-construction or a builder-promoted project
Tier	City tier where the loan was sought. Tier-1 = Major City, Tier-2 = Minor City, Tier-3 = Town/Village
Accommodation_Class	Variable to indicate whether applicant resides currently in rented or non-rented premises
Loan_Type	Variable to indicate if loan was sought for Home loan or Home Improvement loan
Gender	Applicant's Gender
Employment_Type	Variable to indicate whether the applicant was salaried or self-employed

Hands-on Exercise

Data

Variable	Definition
doc_proof_inc	Indicates whether the applicant submitted documentary proof of income
Marital_Status	Indicates if applicant is married or single currently
Employer_Type	Applicant's Employer's category (Business, Corporate, Government, Ind/Small Business)
Education_Class	Education of the applicant
Mode_of_origin_class	The source from which the application originated
eom_25	Variable to indicate whether the application was received after the 25 th of the month

Hands-on Exercise

Data

Variable	Definition
oldemi_d	Variable to indicate if applicant had old loans
bs_d	Variable to indicate if applicant has bank savings
Age	Age of applicant
Yrsadd	Years in current residential address
Yrsjob	Years in current job
Expen	Monthly expenses of applicant
Totinc	Monthly income of applicant
Dispinc	Total monthly income - Total monthly expenses
Marval	Market value of the property for which loan is sought
Oldemi	EMI for earlier loans that the applicant pays every month
Loanreq	Loan amount requested by applicant
Term	Term for the loan
Dwnpay	Down payment by applicant
Banksave	Bank saving of applicants
Calcemi	EMI calculated for the applicant's requested loan amount

Hands-on Exercise

Data

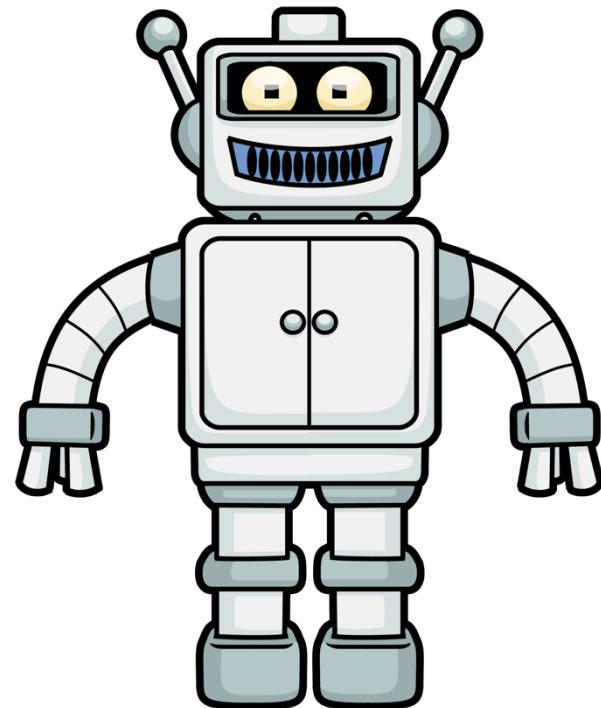
Standard Mortgage Ratios

IIR	$\frac{\text{Equated Monthly Installments (EMI)}}{\text{Total Household Income}}$
IAR	$\frac{\text{Equated Monthly Installments (EMI)}}{\text{Disposable Income}}$ Disposable Income = Total Household Income – Total Expenses
LTV	$\frac{\text{Total Loan Requested}}{\text{Market Value of Property}}$
LVR	$\frac{\text{Total Loan Requested}}{\text{Property Value}}$ Property value is registered value of property at the municipality
FOIR	$\frac{\text{EMI} + \text{Ongoing Loan EMI}}{\text{Total Household Income}}$

Hands-on Exercise

Your Task

- Develop and evaluate a **logistic regression** model to classify micro-mortgage applications for Shubham!
- Use a training and test set approach
- Calculate
 - Accuracy
 - Confusion matrix
 - Recall, Precision, F1-Score
 - ROC/AUC





Syllabus

Syllabus

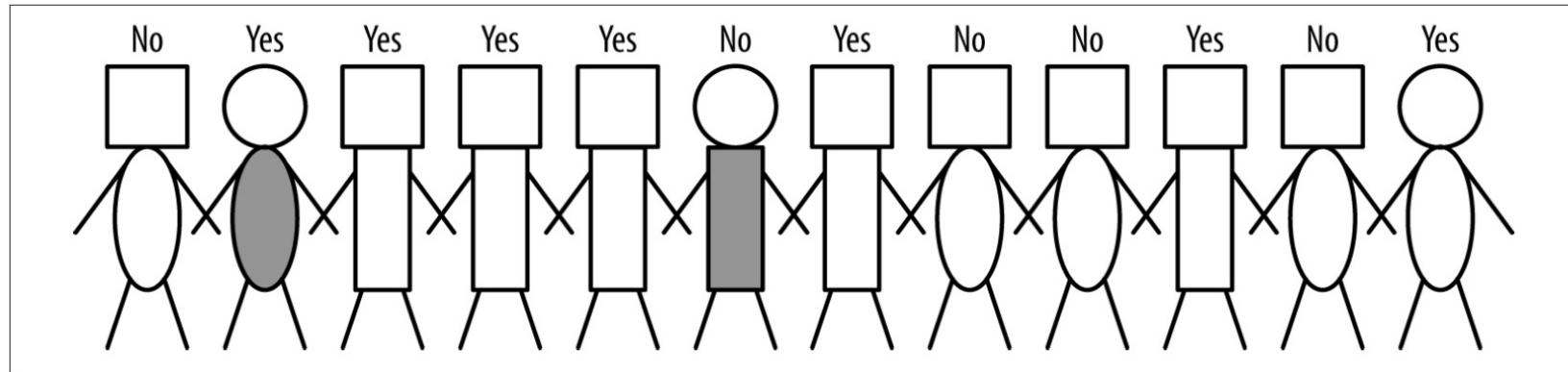
- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Supervised Learning:
_ Tree-based Models

Tree-based Models

Classification by Hand...



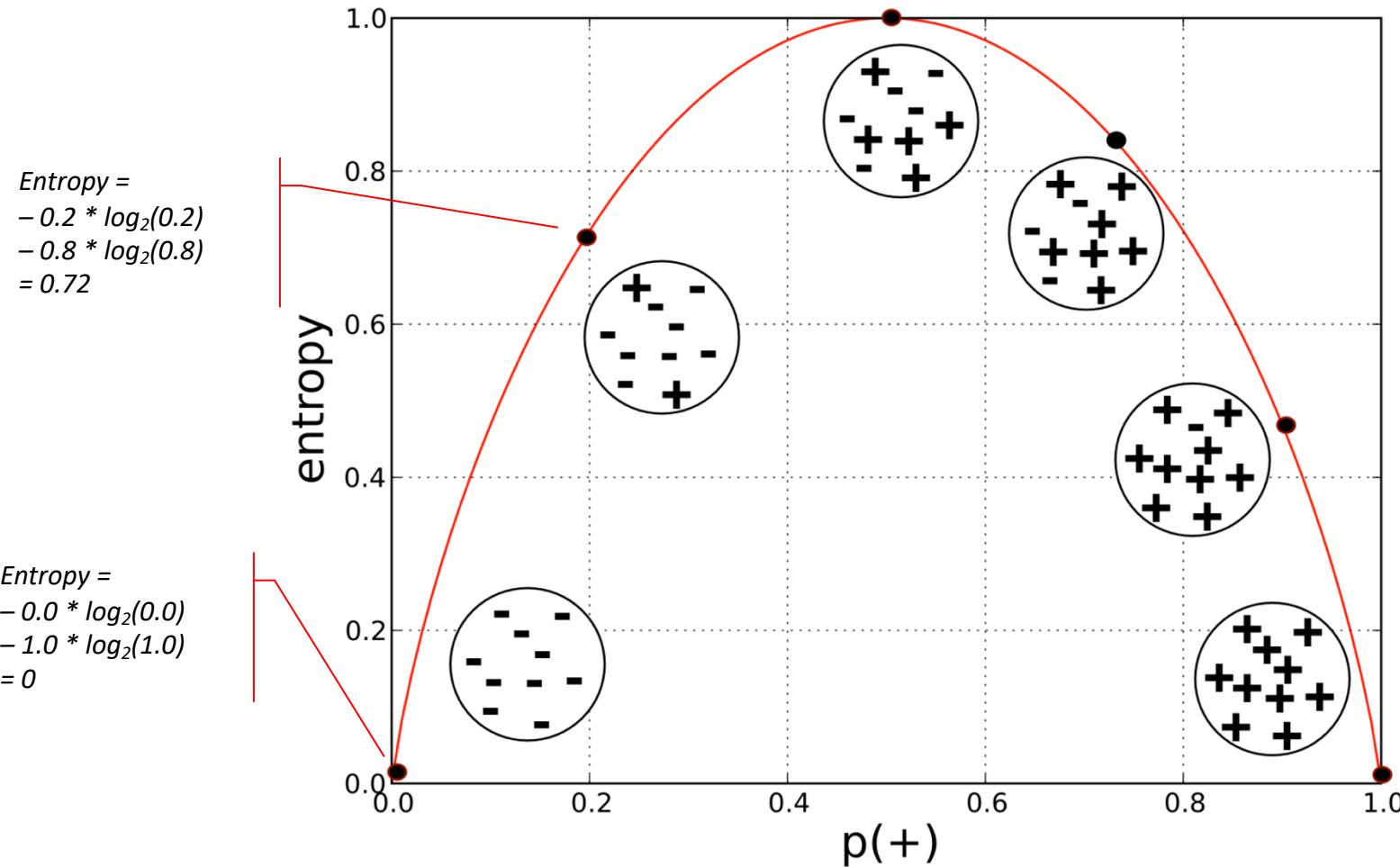
...idea: partition the population into groups that differ from each other with respect to the target variable. The groups should be as pure as possible.

Entropy

- Goal: Quantify how well an attribute splits a set of observations into groups, with respect to a target variable (how pure are the groups?)
- Entropy is a *measure of disorder* (or impurity) in a group
- Formula:
 - $\text{Entropy} = -p_1 * \log_2(p_1) - p_2 * \log_2(p_2) - \dots$
 - with p_i representing the probability (relative percentage) of property i within the group

Tree-based Models

Illustration of Entropy

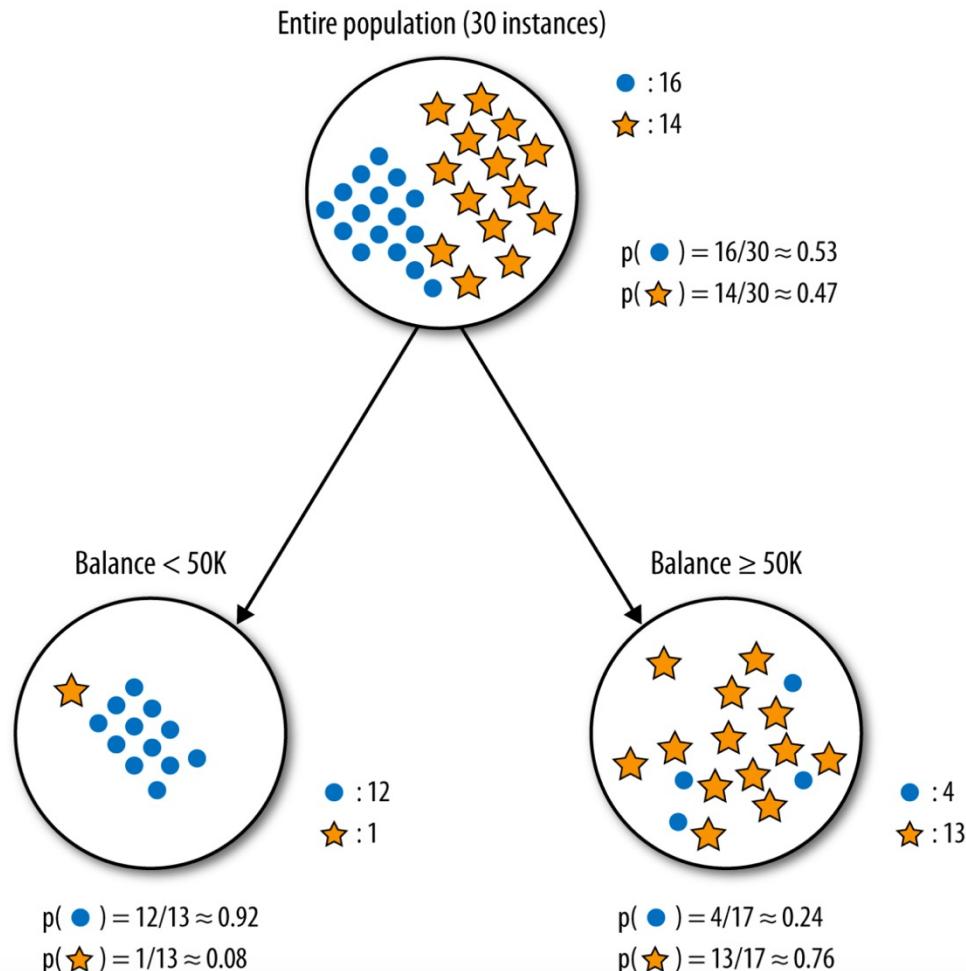


Information Gain

- Information gain measures the *change in entropy* due to any amount of new information being added
- Formula
 - $IG(\text{parent}, \text{children}) = \text{entropy}(\text{parent}) - [p(c_1) * \text{entropy}(c_1) + p(c_2) * \text{entropy}(c_2) + \dots]$
 - with $p(c_i)$ representing the proportion of observations belonging to child i

Tree-based Models

Example: Split on Balance (Split point: 50K)



Tree-based Models

Example: Split on Balance (Split point: 50K)

$$\begin{aligned} \text{entropy}(\text{parent}) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\star) \times \log_2 p(\star)] \\ &\approx -[0.53 \times (-0.9) + 0.47 \times (-1.1)] \\ &\approx 0.99 \quad (\text{very impure}) \end{aligned}$$

$$\begin{aligned} \text{entropy}(\text{Balance} < 50K) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\star) \times \log_2 p(\star)] \\ &\approx -[0.92 \times (-0.12) + 0.08 \times (-3.7)] \\ &\approx 0.39 \end{aligned}$$

$$\begin{aligned} \text{entropy}(\text{Balance} \geq 50K) &= -[p(\bullet) \times \log_2 p(\bullet) + p(\star) \times \log_2 p(\star)] \\ &\approx -[0.24 \times (-2.1) + 0.76 \times (-0.39)] \\ &\approx 0.79 \end{aligned}$$

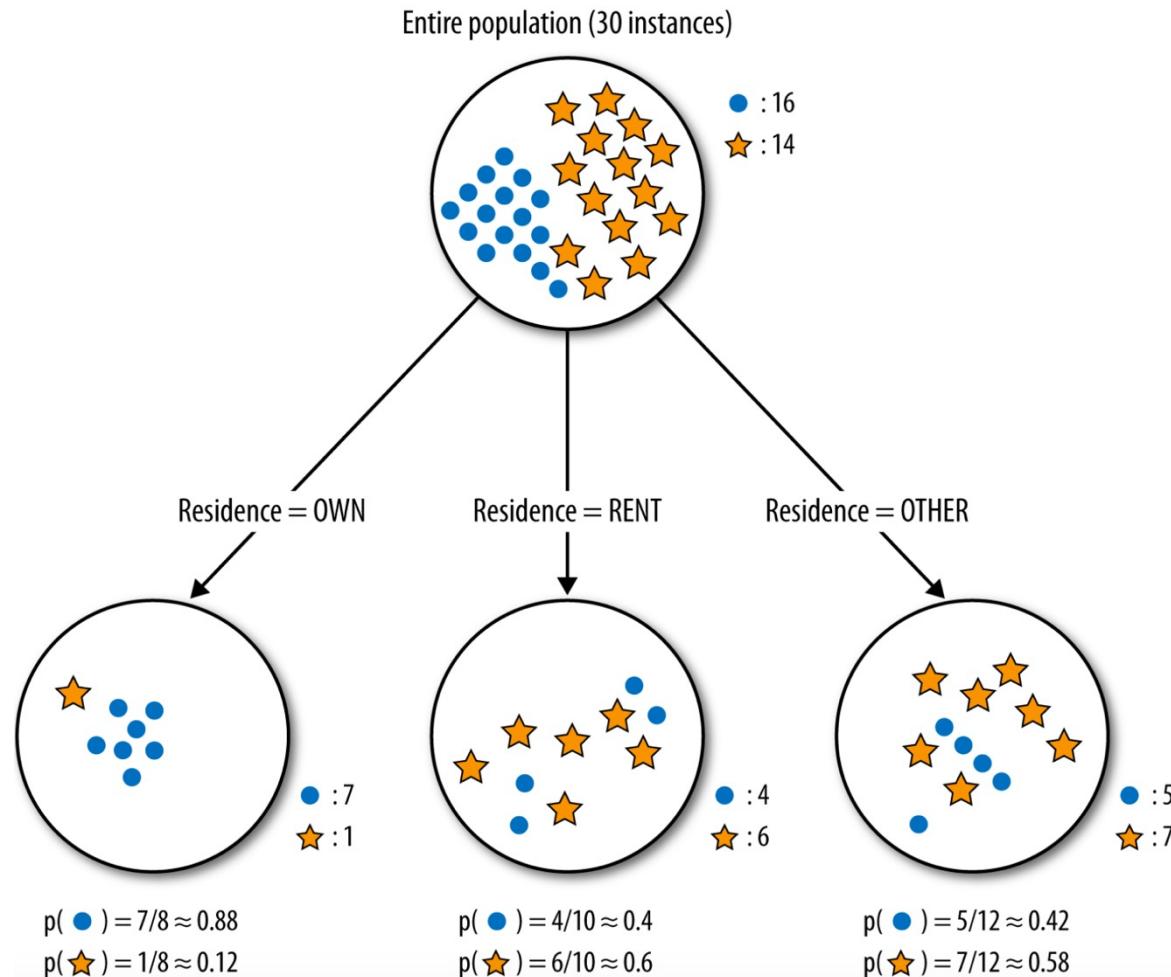
Tree-based Models

Example: Split on Balance (Split point: 50K)

$$\begin{aligned} IG &= \text{entropy}(\text{parent}) - [p(\text{Balance} < 50\text{K}) \times \text{entropy}(\text{Balance} < 50\text{K}) \\ &\quad + p(\text{Balance} \geq 50\text{K}) \times \text{entropy}(\text{Balance} \geq 50\text{K})] \\ &\approx 0.99 - [0.43 \times 0.39 + 0.57 \times 0.79] \\ &\approx 0.37 \end{aligned}$$

Tree-based Models

Example: Split on Residence Categories



Example: Split on Residence Categories

$$\text{entropy}(\text{parent}) \approx 0.99$$

$$\text{entropy}(\text{Residence=OWN}) \approx 0.54$$

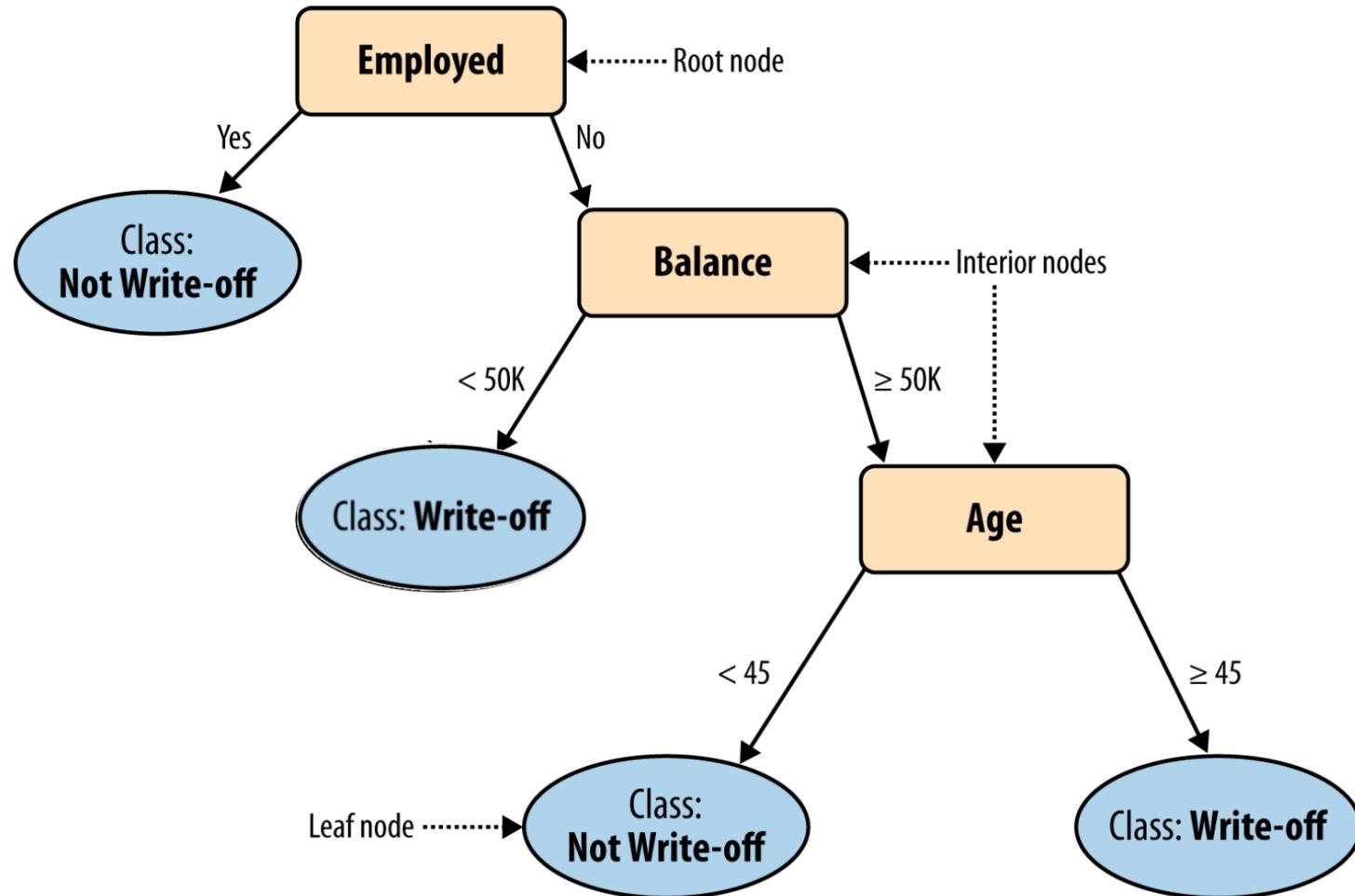
$$\text{entropy}(\text{Residence=RENT}) \approx 0.97$$

$$\text{entropy}(\text{Residence=OTHER}) \approx 0.98$$

$$IG \approx 0.13$$

Tree-based Models

Classification Tree



Growing a Tree with the CART Algorithm

- It is computationally infeasible to consider all possible combinations and sequences of splits
- Instead, grow tree via **recursive binary partitioning**
 - **Top-down:** Start with zero splits and successively partition the feature space into two parts
 - **Greedy:** At each step, make the best possible split at that particular step
 - Stop when some condition (e.g., minimal number of observations in one leaf) is met
- That is, we consider all predictors X_1, \dots, X_p , and all possible split points s for each of the predictors, and then choose the predictor and split point with the **highest information gain** at each step.

Tree-based Models

Moneyball – Regression with Trees

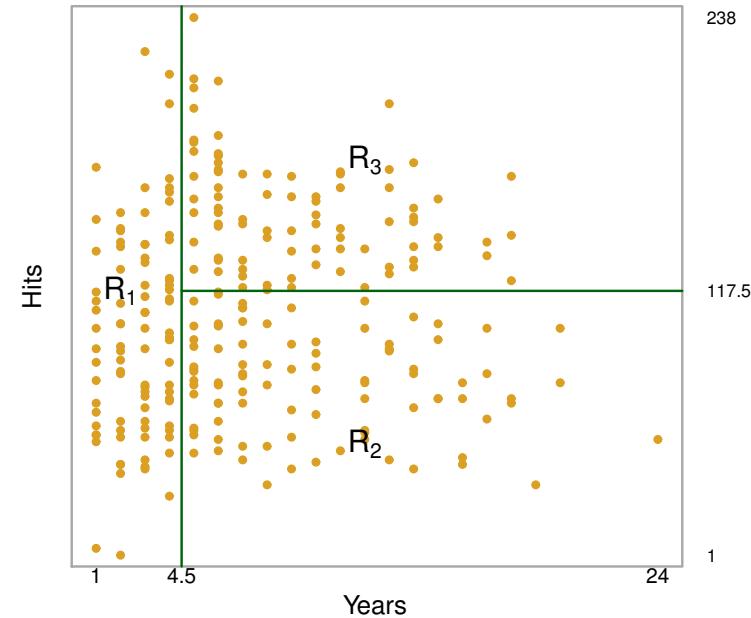
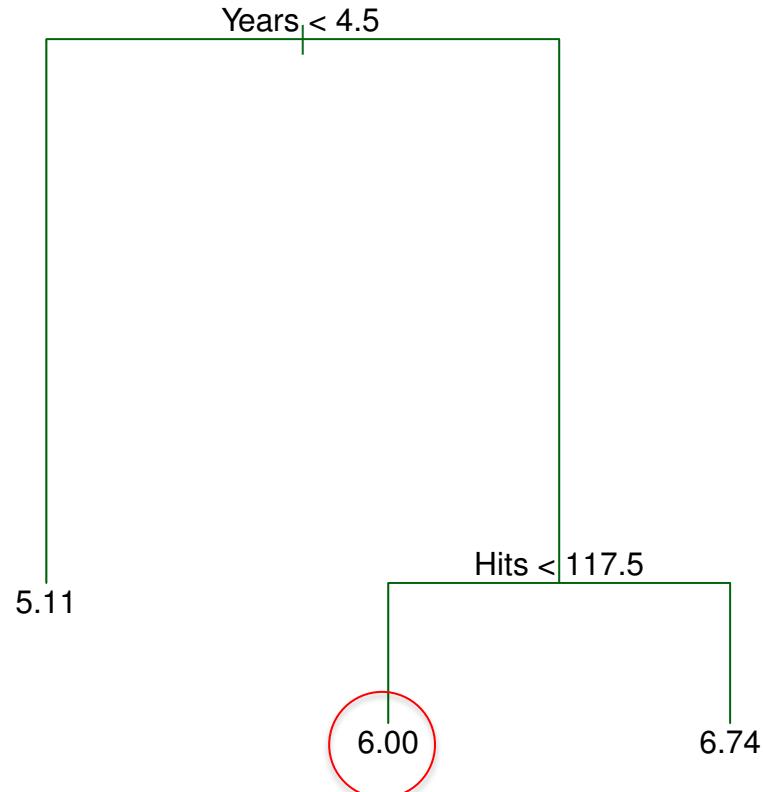


Tree-based Models



Predicting Baseball Players' Salaries (Regression)

(on logarithmic scale in thousands of dollars)



$$= \text{EXP}(6.0) * 1000 = 403,429 \text{ USD}$$

Regression with Trees

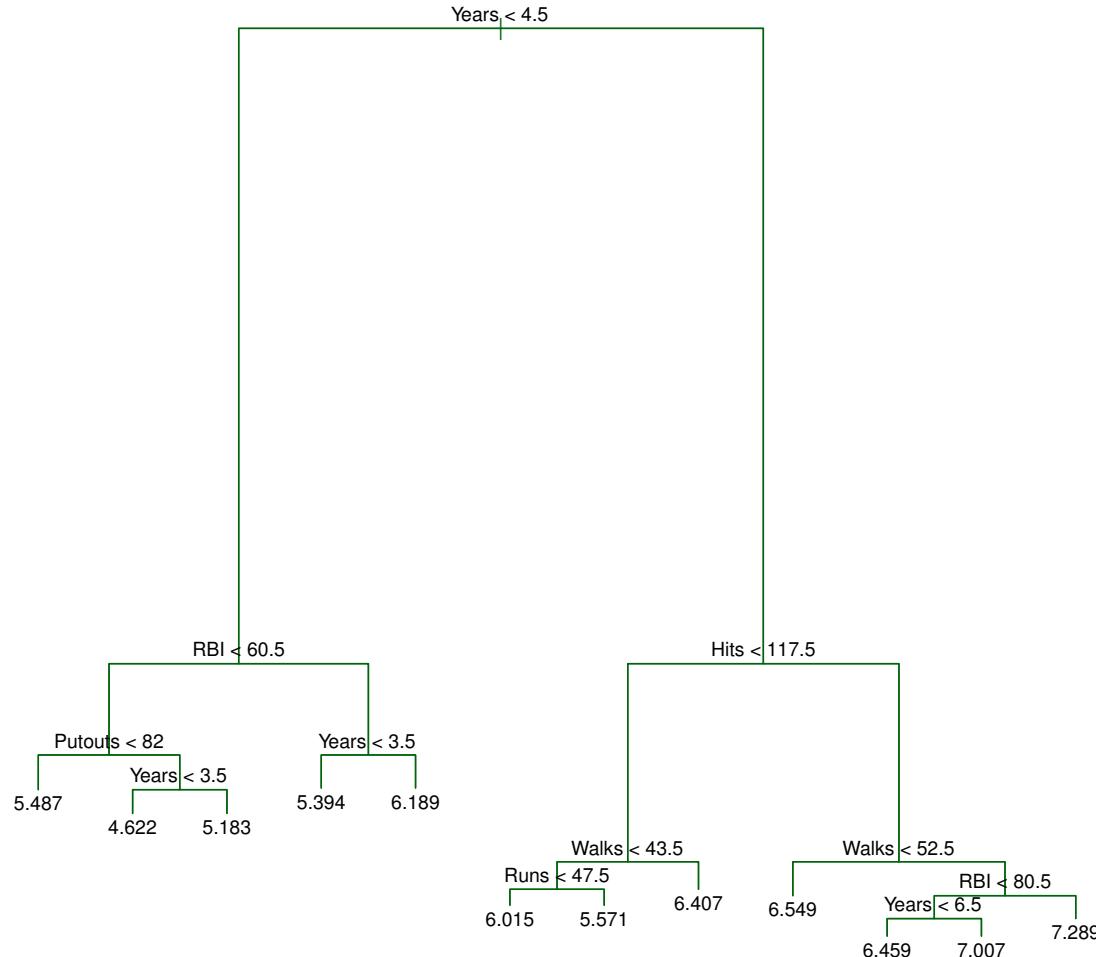
- Instead of using entropy, the goal is to find boxes that minimize the RSS:

$$\sum_{j=1}^J \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2.$$

Tree-based Models

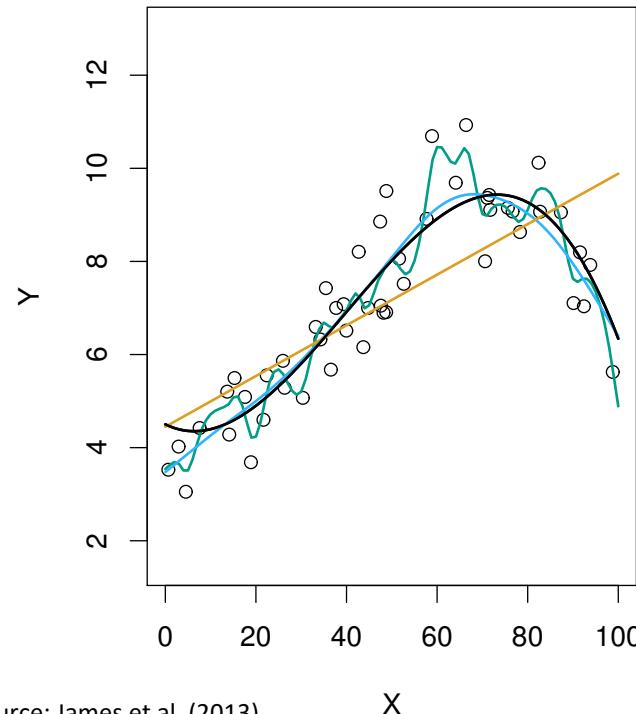


Result of Tree Growing for Baseball Dataset



Problem: Overfitting

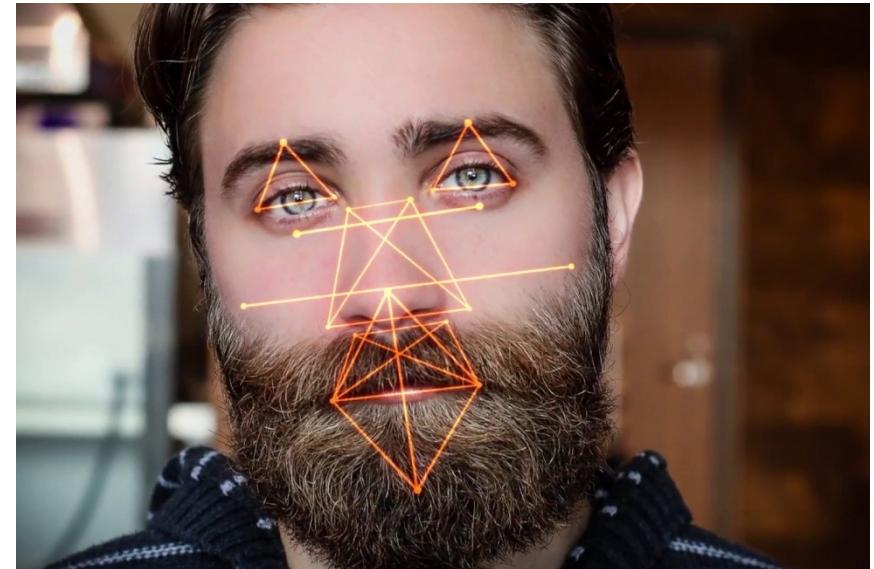
- Definition: Situation in which a model starts learning patterns that are caused by random chance (noise)
- Symptoms: Good performance on the training set, but bad performance on the test set
- Cause: Overfitting occurs when a model is too flexible or too complex



Tree-based Models

Problem: Overfitting

- Overfitting is the tendency of machine learning algorithms to **tailor models to the training data, at the expense of generalization** to previously unseen data points.

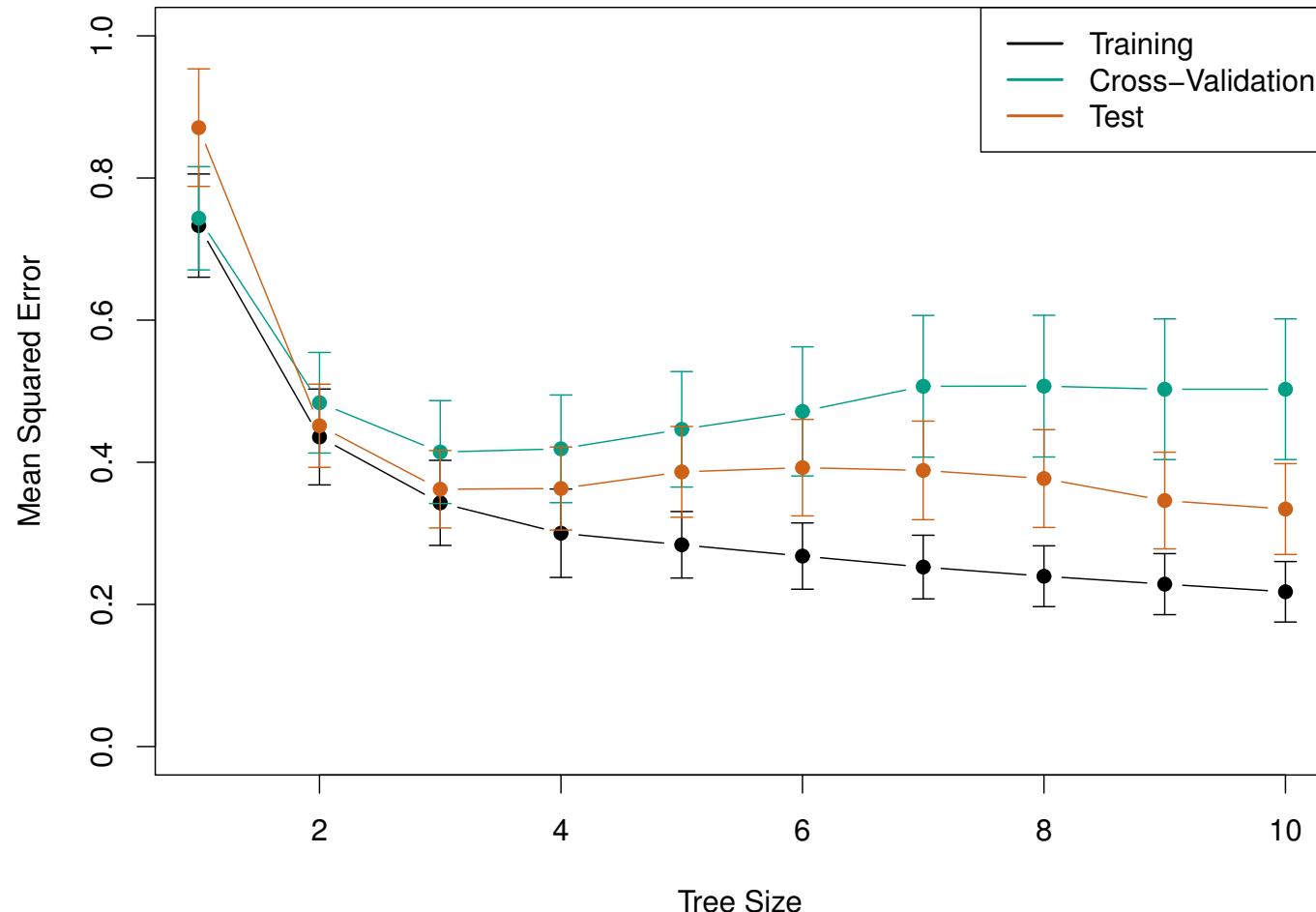


Pruning a Tree

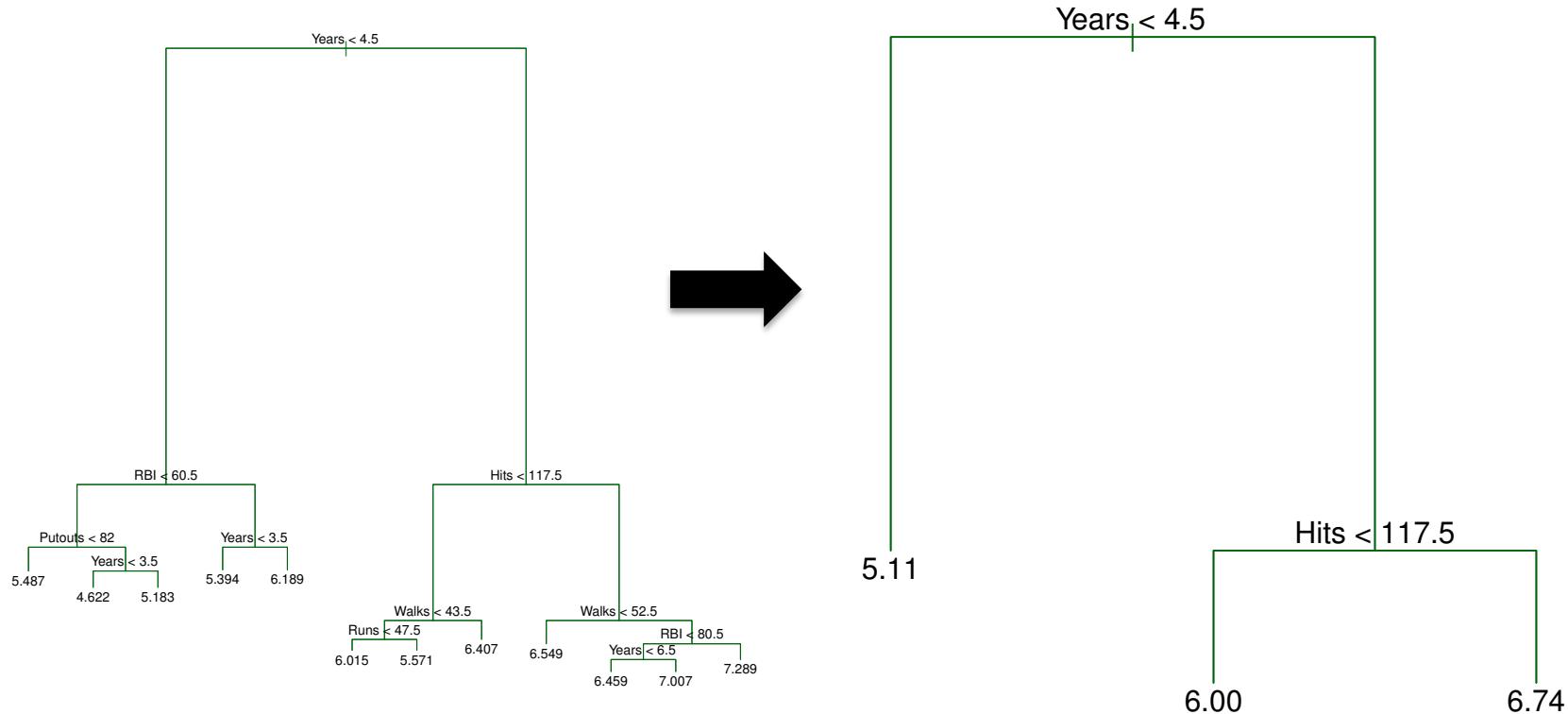
- Growing a **large tree** typically leads to accurate predictions for the training set but is likely to **overfit** the data (poor performance on the test set).
- Often, a **smaller sub-tree** exists that **outperforms the large tree** on the test set.
- Idea
 - Pruning aims to **select the subtree that leads to the lowest test error rate**.
 - Given a subtree, we can **estimate its test error using cross-validation** or the validation set approach.
 - So we can try out all possible sub-trees and select the one with the lowest test error. (Note: There are smarter ways to find the best sub tree – see ISL book page 307 ff.)



Tree Pruning for Baseball Dataset



Tree-based Models



Pros and Cons of Trees

Pros

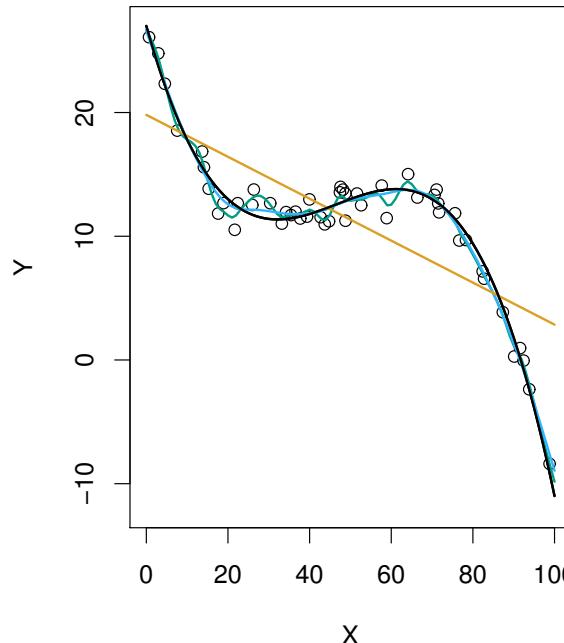
- Easy to understand and explain
- Similar to human judgment and decision making
- Can be visualized nicely
- Can easily handle qualitative predictors

Cons

- Risk of overfitting
- Not as accurate as other methods

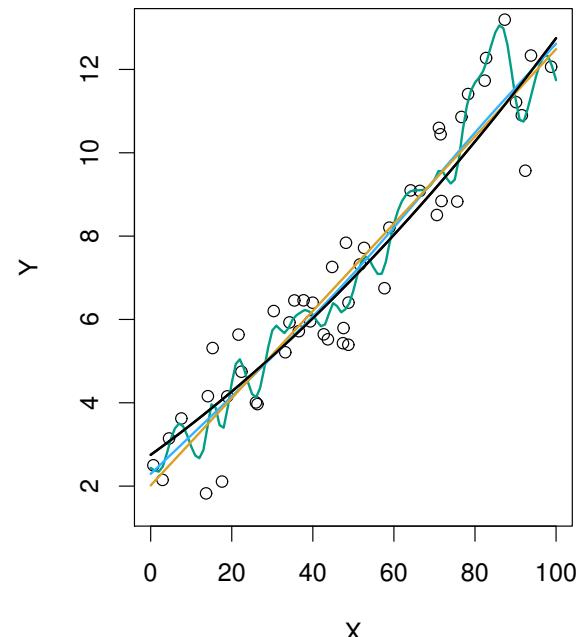
The Bias-Variance Tradeoff

- Roughly speaking, the test set error of a model can be broken down into three components, **bias**, **variance**, and **irreducible error**
- **Bias** refers to the error that is introduced by approximating a real-life problem, which may be extremely complex, by a simpler model



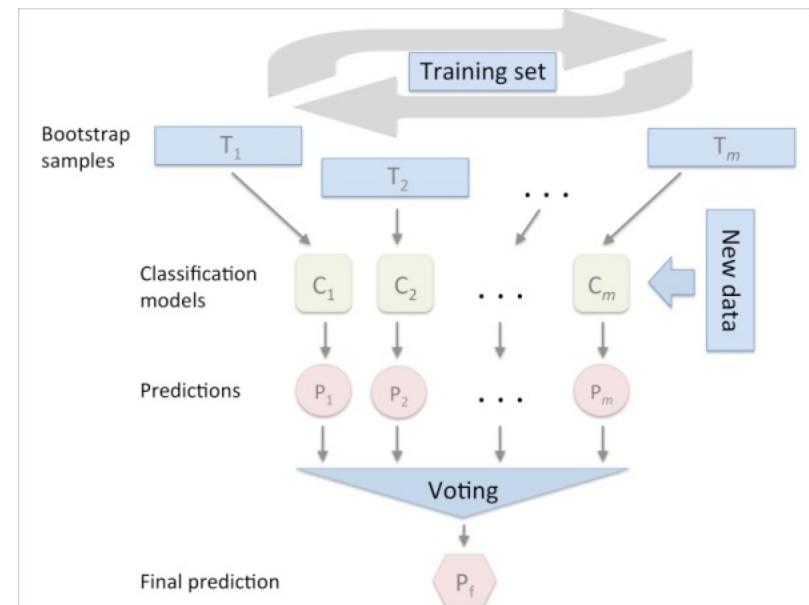
The Bias-Variance Tradeoff

- Roughly speaking, the test set error of a model can be broken down into three components, **bias**, **variance**, and **irreducible error**
- **Variance** refers to the amount by which f would change, if we estimated it using a different training set



Bagging

- A way to reduce the variance of a machine learning algorithm (and thereby prevent overfitting) is to sample many training sets from the population, build a separate model on each sample, and average the resulting predictions.
- This procedure is called **bootstrap aggregation**, or **bagging**.



Tree-based Models

Random Forest

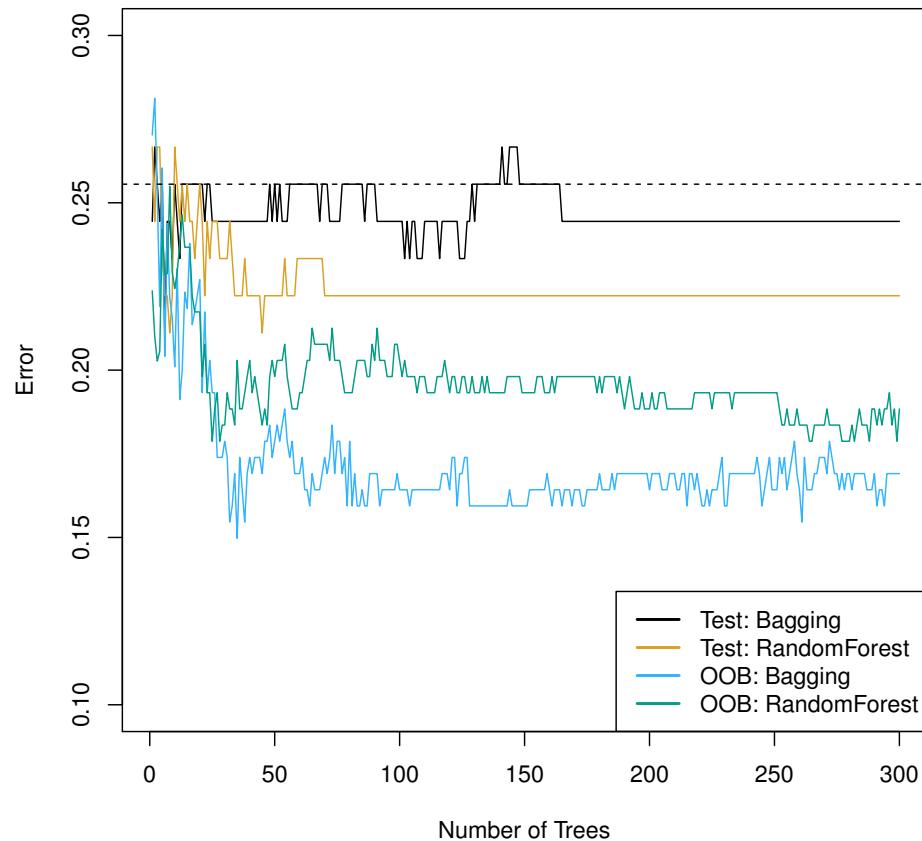


Random Forest

- **Following the idea of bagging**, we draw multiple random samples (bootstrap samples) from the training data and grow a CART tree on each sample
 - Typically, 2/3 of the rows in the training set
- However, in Random Forests we **allow only a random subset (m) of all the predictors (p) to be used at each split of the decision tree**
 - Typically, $m = \text{SQRT}(p)$
- **Why** does this work?
 - In bagging, if there is one strong predictor, all the trees will use this predictor in the top split
 - ➔ all of the trees will look quite similar to each other
 - ➔ their predictions will be highly correlated
 - ➔ only a little bit of variance will be removed

Tree-based Models

Example: Heart Dataset



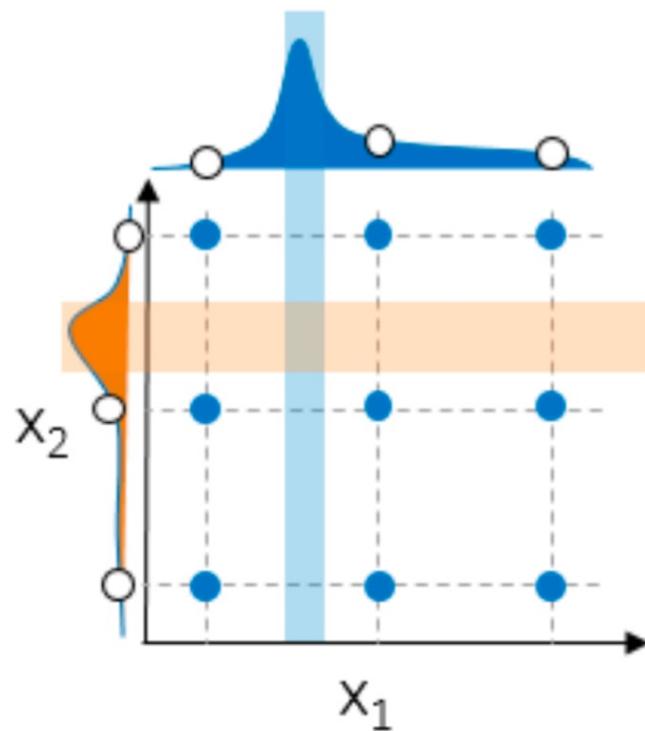
OOB: Out of bag observations, that is, the observations which were not included in the bootstrap samples (usually 1/3 of the observations)

Hyperparameters of a Random Forest

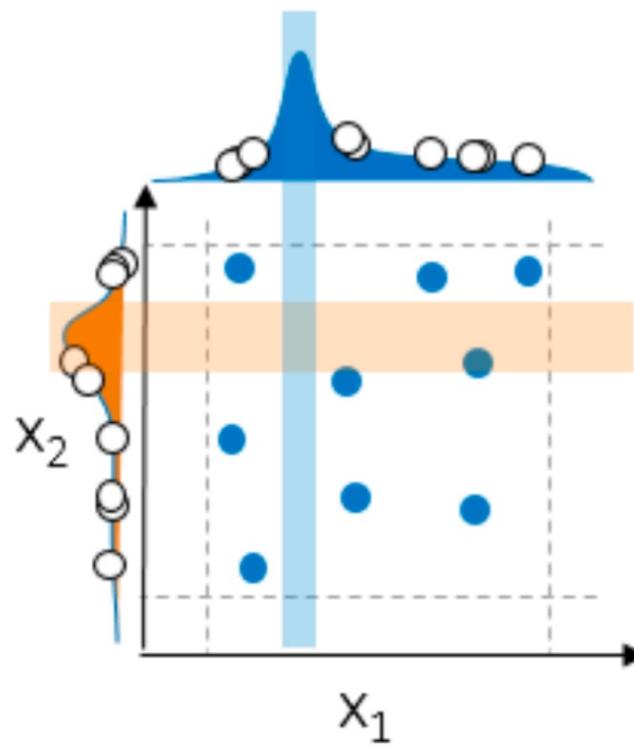
- **ntree**
 - Number of trees to grow. This should not be set to too small a number, to ensure that every input row gets predicted at least a few times.
- **mtry**
 - Number of variables randomly sampled as candidates at each split. Note that the default values are different for classification (\sqrt{p}) where p is number of variables in x) and regression ($p/3$)
- **sampsize**
 - Size of samples to draw. Default is 63% of observations in the training set.
- **nodesize**
 - Minimum size of terminal nodes. Setting this number larger causes smaller trees to be grown (and thus take less time). Note that the default values are different for classification (1) and regression (5).
- **maxnodes**
 - Maximum number of terminal nodes trees in the forest can have. If not given, trees are grown to the maximum possible (subject to limits by nodesize).

Tree-based Models

Tuning Hyperparameters



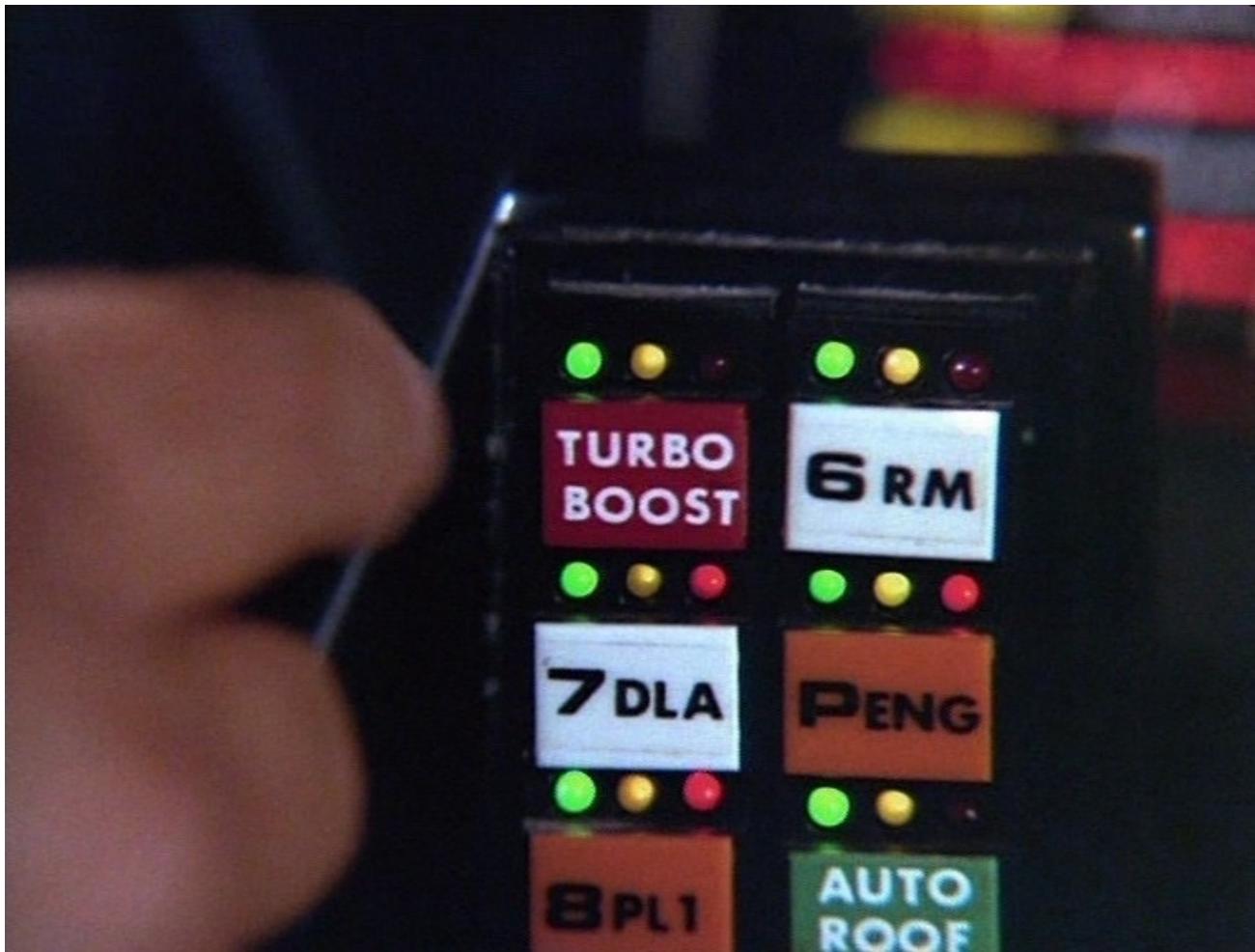
(a) Standard Grid Search



(b) Random Search

Tree-based Models

Boosting



Boosting

- Bagging involves drawing multiple *independent* samples from the original training set and fitting a separate decision tree to each sample.
- In boosting, **each tree is grown sequentially using information from the previously grown trees**
 - Each tree is grown in such a way that it corrects the errors of the previous trees

Boosting for Regression Trees

Algorithm 8.2 *Boosting for Regression Trees*

1. Set $\hat{f}(x) = 0$ and $r_i = y_i$ for all i in the training set.
2. For $b = 1, 2, \dots, B$, repeat:
 - (a) Fit a tree \hat{f}^b with d splits ($d + 1$ terminal nodes) to the training data (X, r) .
 - (b) Update \hat{f} by adding in a shrunken version of the new tree:

$$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x).$$

- (c) Update the residuals,

$$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i).$$

3. Output the boosted model,

$$\hat{f}(x) = \sum_{b=1}^B \lambda \hat{f}^b(x).$$

Tree-based Models

Or in other “words”...

$$Y = f(X) + \varepsilon_1$$

$$\varepsilon_1 = f^1(X) + \varepsilon_2$$

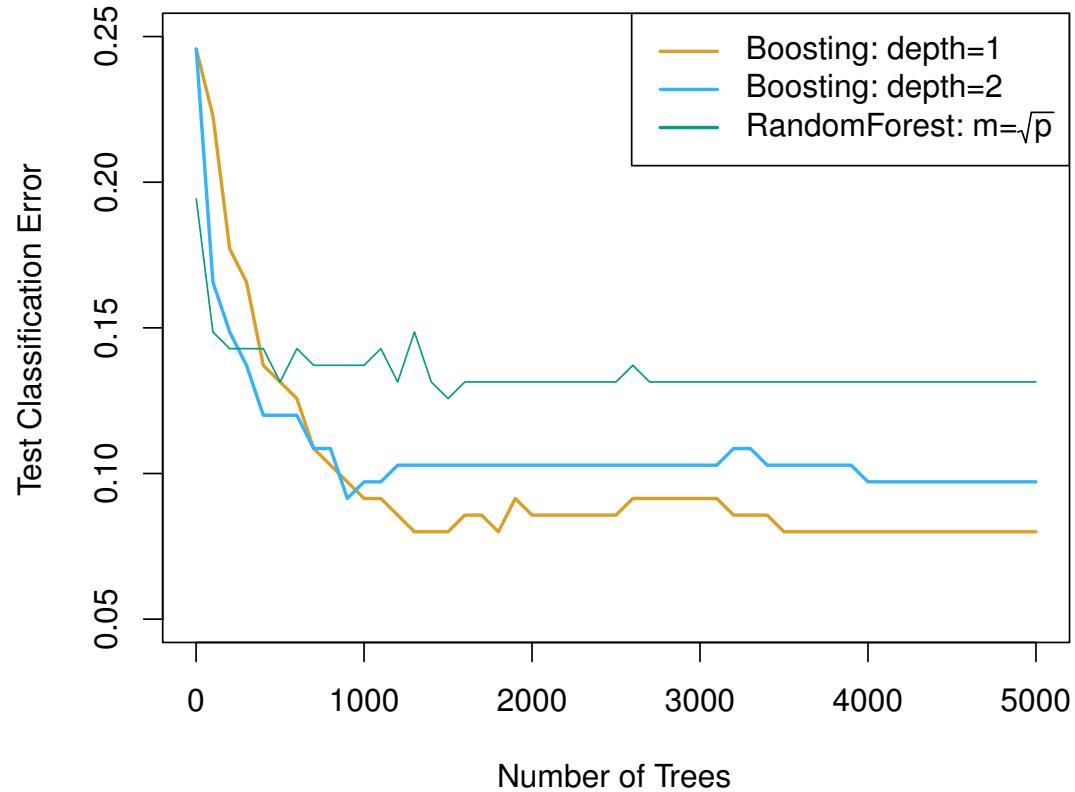
$$\varepsilon_2 = f^2(X) + \varepsilon_3$$

$$\varepsilon_3 = \dots$$

$$Y = \lambda * f(X) + \lambda * f^1(X) + \lambda * f^2(X) + \dots$$

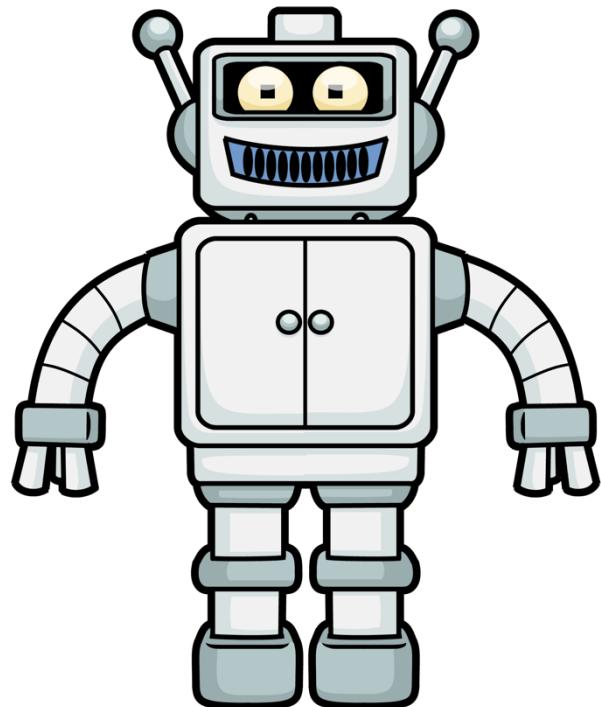
Tree-based Models

Example: Gene Expression Dataset



Your Task

- Improve the model for classifying micro-mortgage applications for Shubham using **tree-based methods!**
- Use a training and test set approach.
- Calculate
 - Accuracy
 - Confusion matrix
 - Recall, Precision, F1-Score
 - ROC/AUC





Syllabus

Syllabus

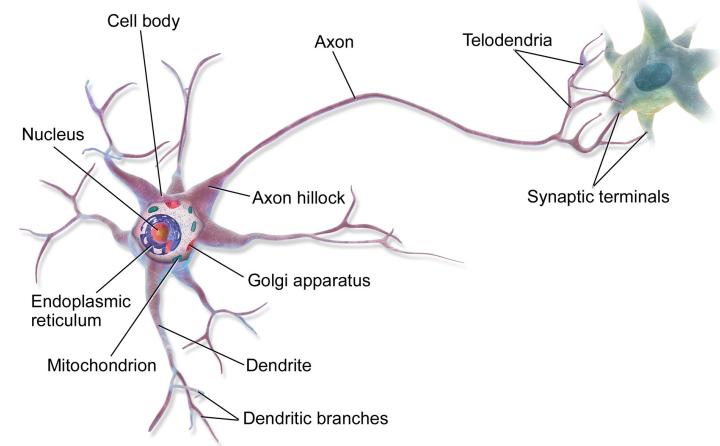
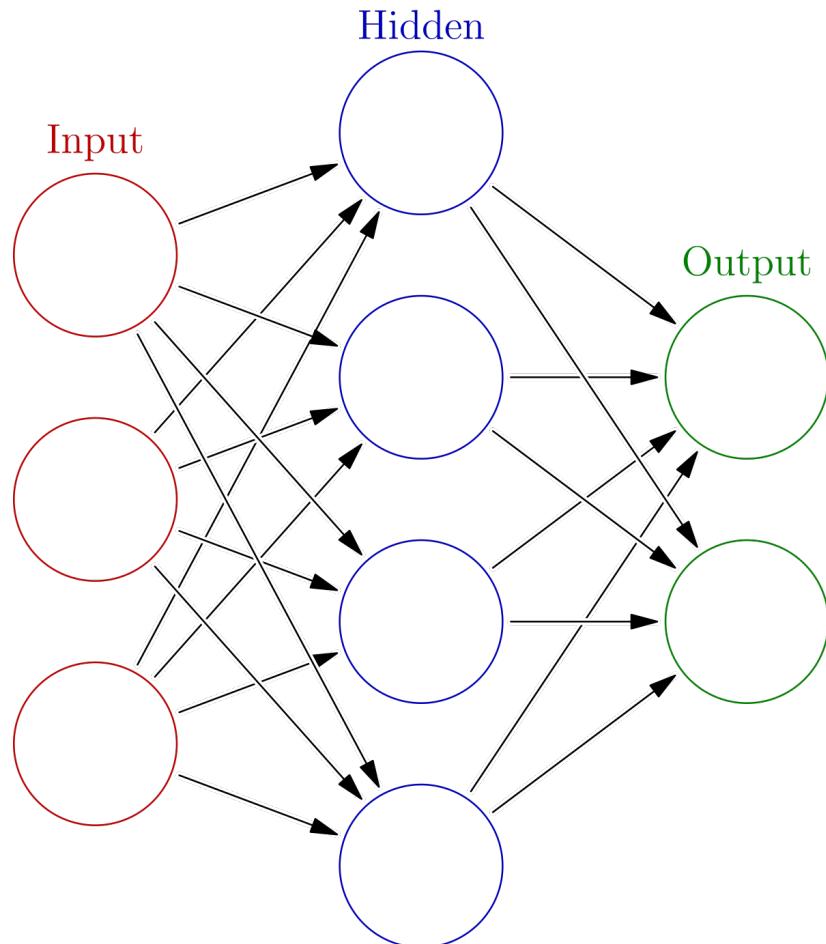
- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Neural Networks

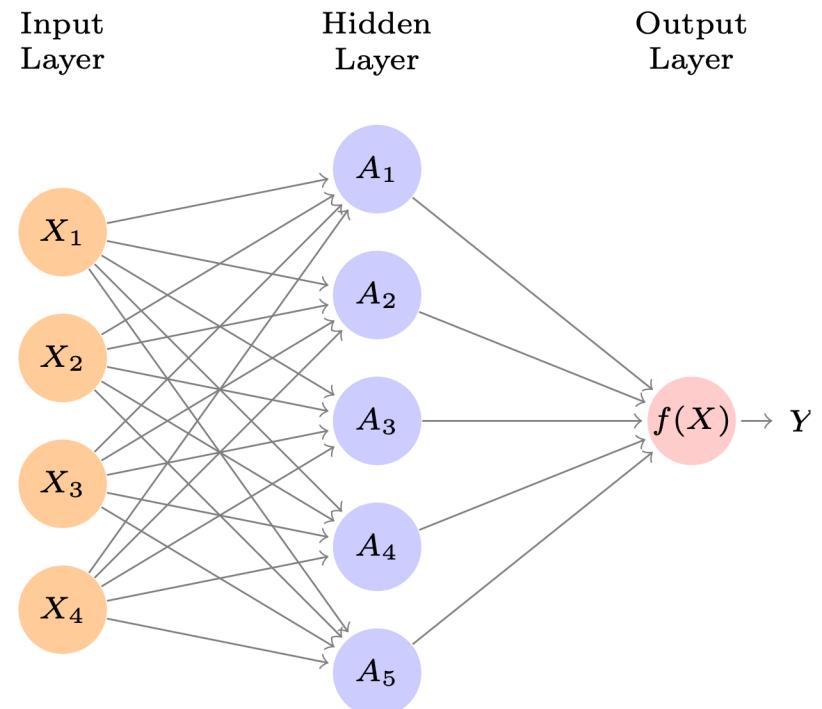
Neural Networks

Artificial Neural Networks



Network Structure

- The units X in the **input layer** are made up of the p features of the dataset.
- The arrows indicate that each input feeds into all k units A in the **hidden layer**.
- In the **output layer**, the results from the hidden layer are again combined to form the final predictions Y .



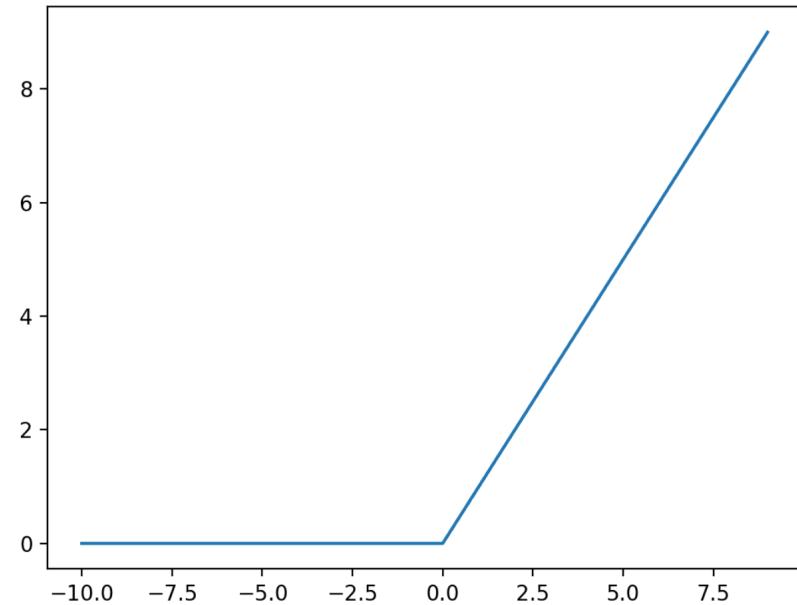
$$A_k = h_k(X) = g(w_{k0} + \sum_{j=1}^p w_{kj} X_j)$$

$$f(X) = \beta_0 + \sum_{k=1}^K \beta_k A_k$$

Activation Functions

- Note the function g on the previous slide.
- The activation functions are non-linear transformations of the outputs of a unit.
- The typical choice for an activation function in modern neural networks is the **ReLU** (rectified linear unit) function:

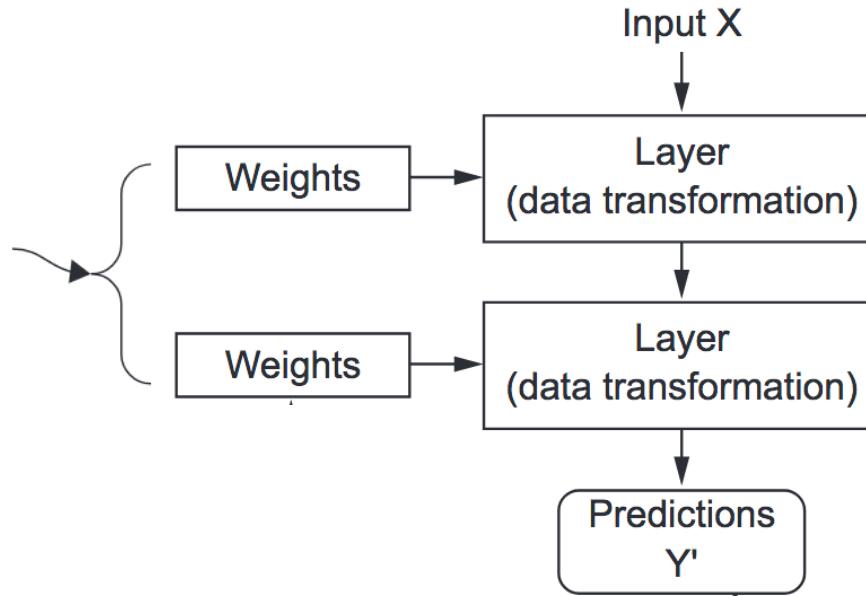
$$g(z) = (z)_+ = \begin{cases} 0 & \text{if } z < 0 \\ z & \text{otherwise.} \end{cases}$$



Neural Networks

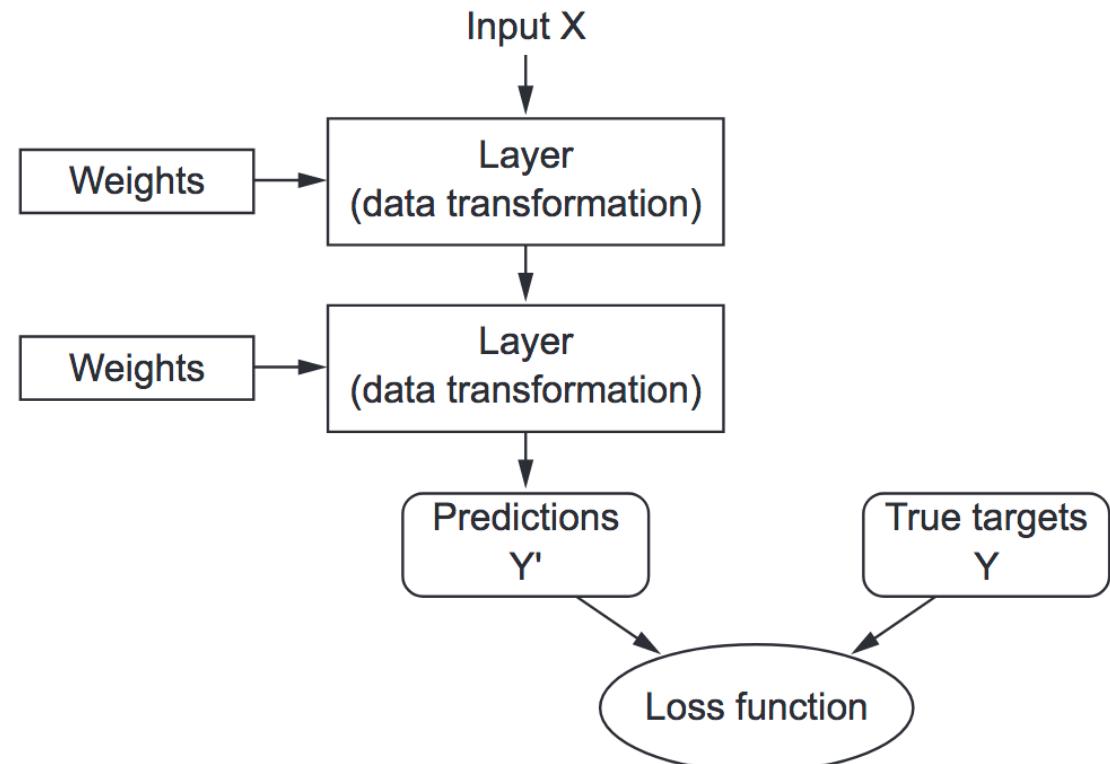
Fitting a Neural Network

Goal: finding the right values for these weights



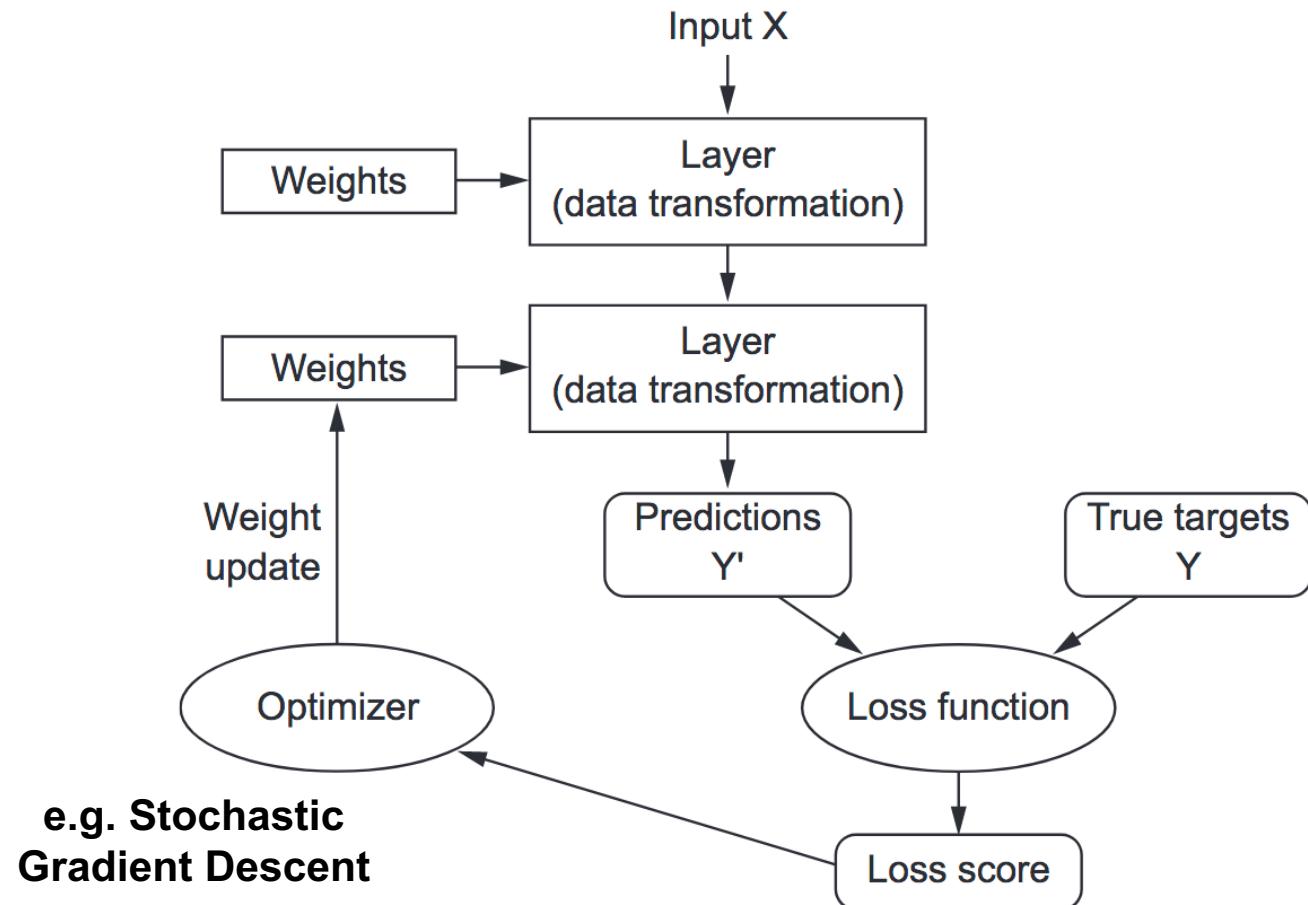
Neural Networks

Fitting a Neural Network



Neural Networks

Fitting a Neural Network



e.g. Stochastic
Gradient Descent

Can you predict the sales price of a house?



Syllabus

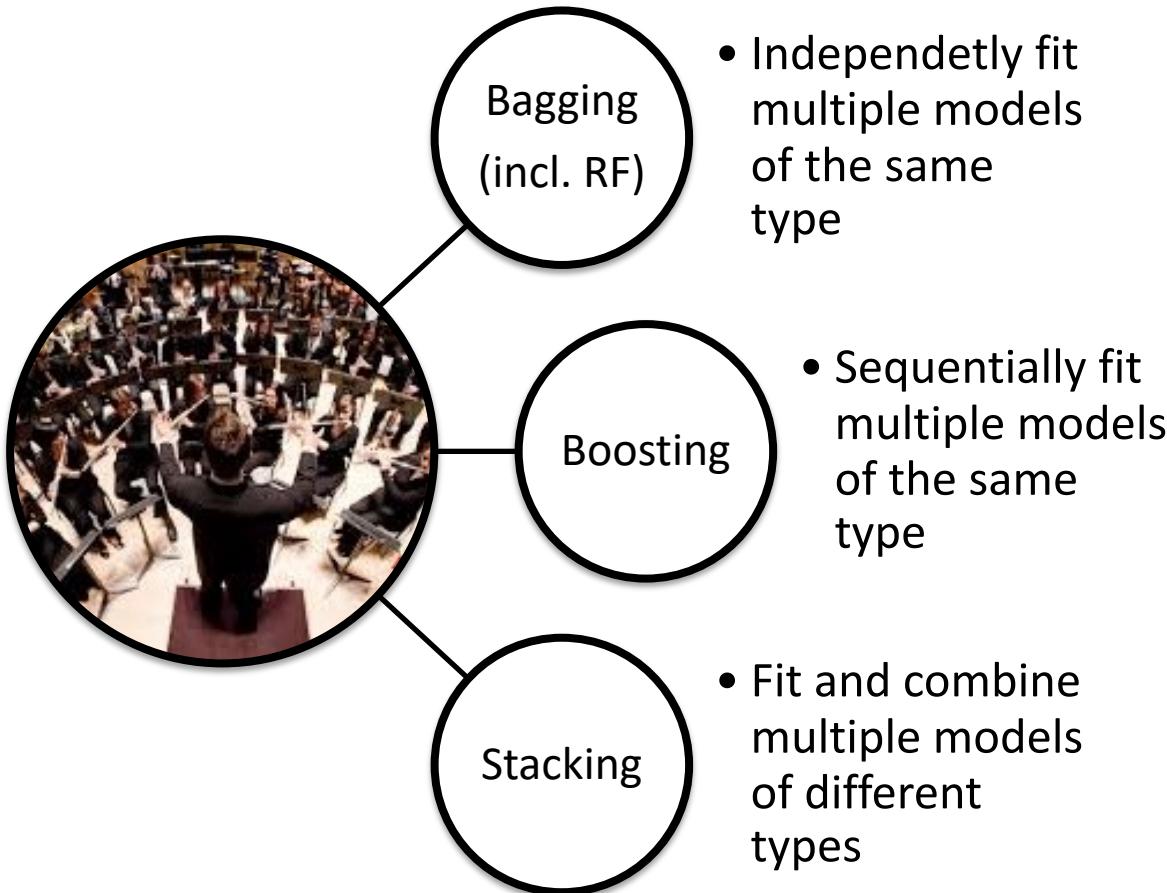
Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Model Interpretability
 - Reproducibility of ML-based Research
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

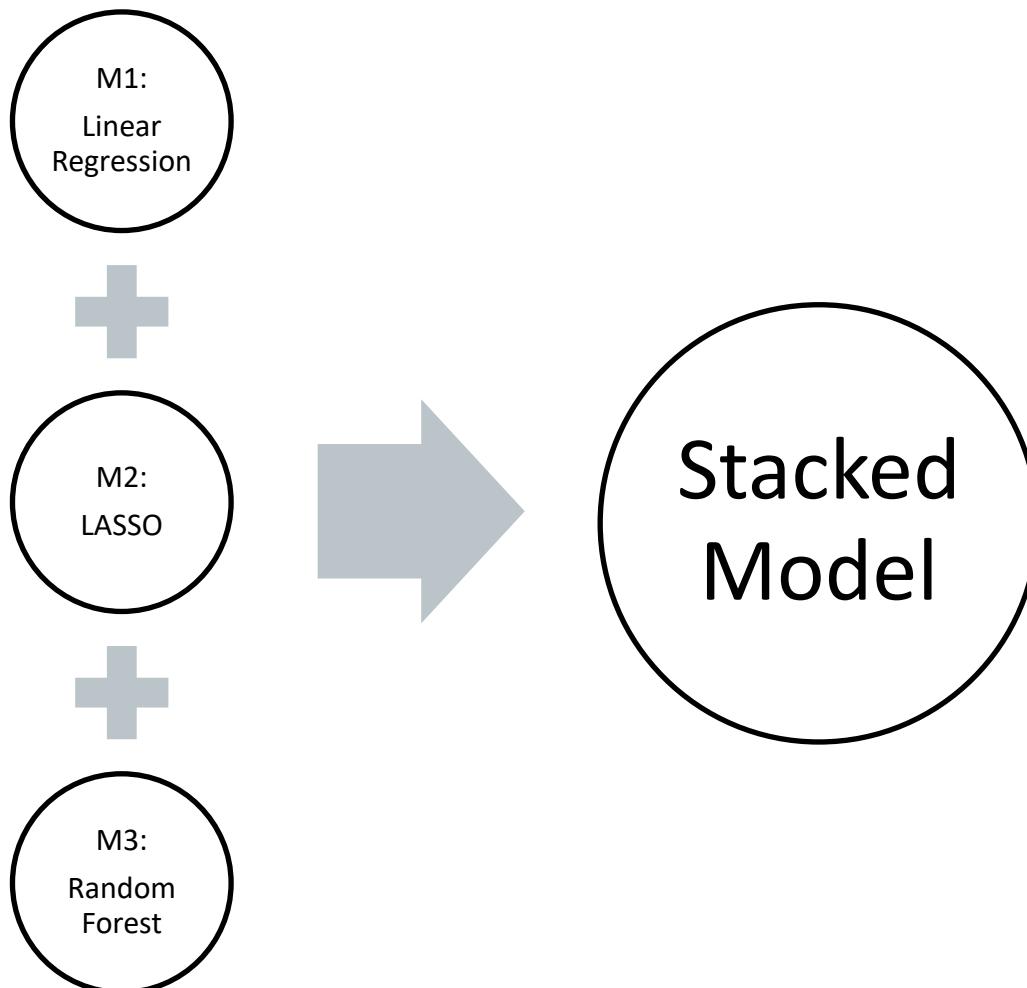
Syllabus

Ensembles and AutoML

Types of Ensembles

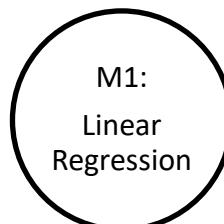


Stacking

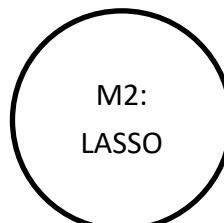


Stacking

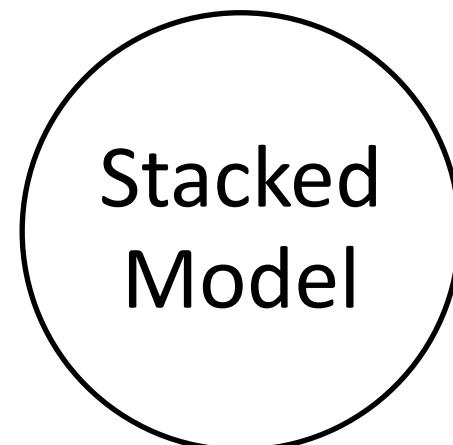
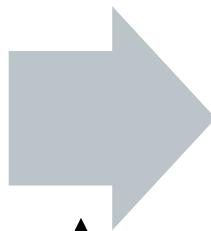
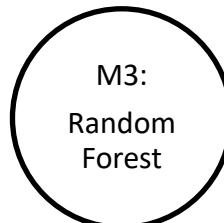
$$\widehat{Y}_{M1} = f_1(X)$$



$$\widehat{Y}_{M2} = f_2(X)$$



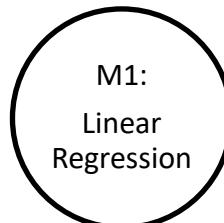
$$\widehat{Y}_{M3} = f_3(X)$$



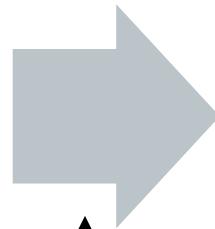
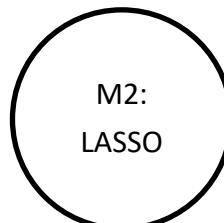
$$\hat{Y} = \frac{1}{3} * \widehat{Y}_{M1} + \frac{1}{3} * \widehat{Y}_{M2} + \frac{1}{3} * \widehat{Y}_{M3}$$

Stacking

$$\widehat{Y}_{M1} = f_1(X)$$

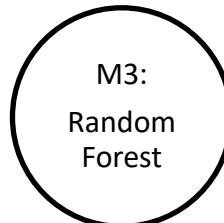


$$\widehat{Y}_{M2} = f_2(X)$$



Stacked
Model

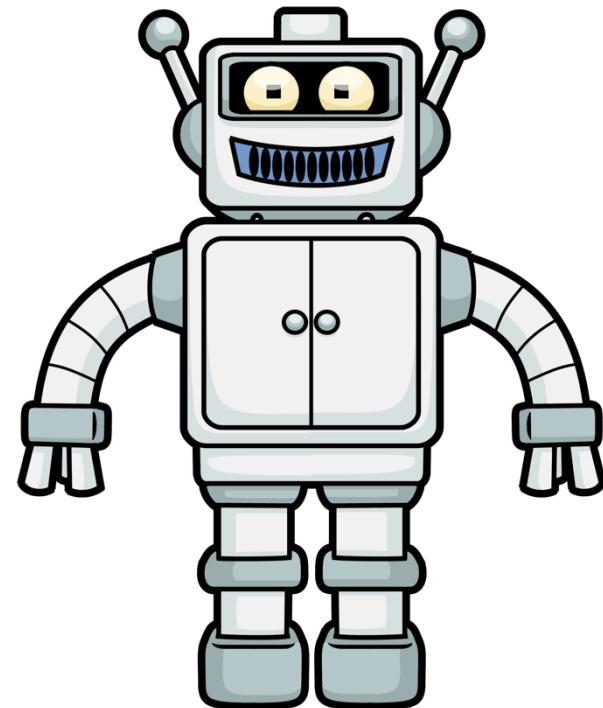
$$\widehat{Y}_{M3} = f_3(X)$$



$$\widehat{Y} = \beta_0 + \beta_1 * \widehat{Y}_{M1} + \beta_2 * \widehat{Y}_{M2} + \beta_3 * \widehat{Y}_{M3}$$

Your Task

- Improve the model for classifying micro-mortgage applications for Shubham using **stacking!**
- Use a training and test set approach.
- Calculate
 - Accuracy
 - Confusion matrix
 - Recall, Precision, F1-Score
 - ROC/AUC



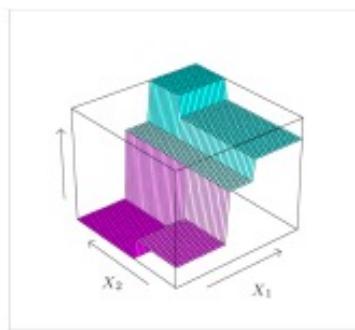
Ensembles and AutoML

AutoML with H2O

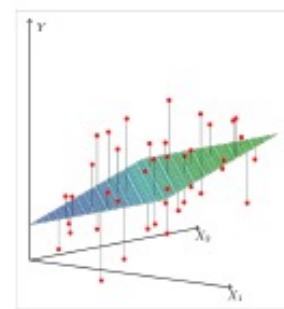
**DRF
XRT**



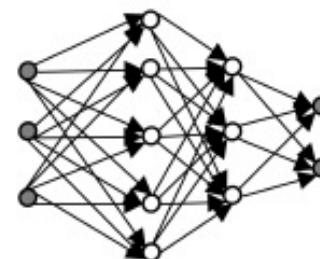
**GBM
XGBoost**



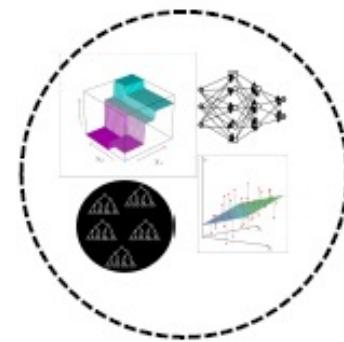
GLM



DNN



Stacked Ensemble



*with
hyperparameter
tuning*

*with
hyperparameter
tuning*

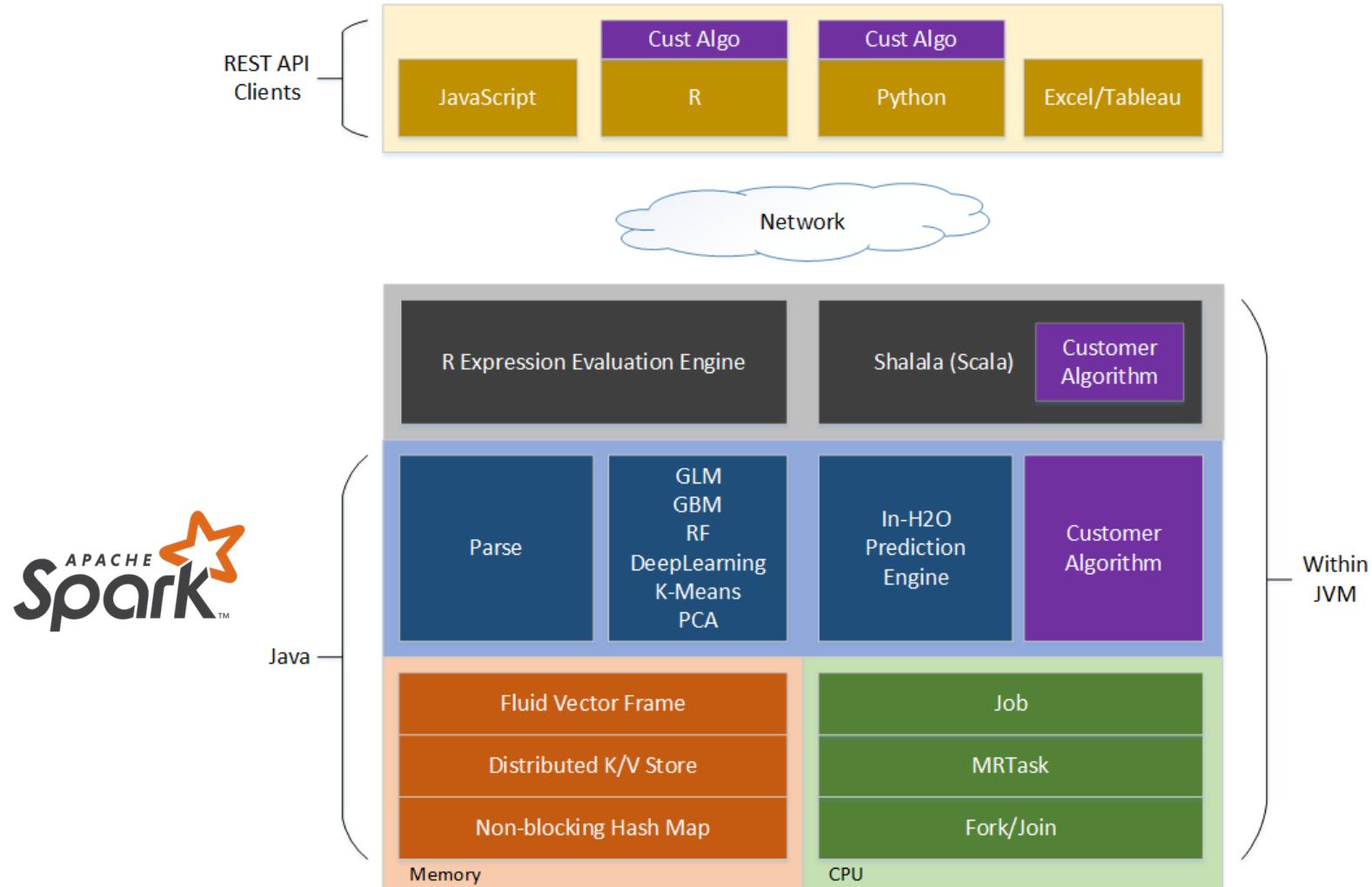
*with
hyperparameter
tuning*

*with
hyperparameter
tuning*

*with
optimal weighting*

Ensembles and AutoML

H2O Software Stack



Can you predict the sales price of a house?





Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Reproducibility of ML-based Research

Reproducibility of ML-based Research

Reproducibility Crisis in ML-based Research

- Many quantitative science fields are adopting the paradigm of predictive modeling using machine learning.
- Yet, there are many reasons for caution:
 - Performance evaluation is notoriously tricky in machine learning.
 - ML code tends to be complex and as yet lacks standardization.
 - Subtle pitfalls arise from the differences between explanatory and predictive modeling.
 - The hype and overoptimism about commercial AI may spill over into ML-based scientific research.
 - Pressures and publication biases that have led to past reproducibility crises are also present in ML-based science.

Reproducibility of ML-based Research

Reproducibility Crisis in ML-based Research

Field	Paper	Number of papers reviewed									
		Number of papers with pitfalls		[I.1.1] No test set		[I.1.2] Pre-proc. on train-test		[I.1.3] Feature sel. on train-test		[I.1.4] Duplicates	
Medicine	Bouwmeester et al. (2012)	71	27	○							○
Neuroimaging	Whelan & Garavan (2014)	—	14	○	○						
Bioinformatics	Blagus & Lusa (2015)	—	6	○							
Autism Diagnostics	Bone et al. (2015)	—	3		○			○	○	○	○
Nutrition Research	Ivanescu et al. (2016)	—	4	○					○	○	
Software Eng.	Tu et al. (2018)	58	11			○		○	○	○	○
Toxicology	Alves et al. (2019)	—	1		○			○	○		
Clinical Epidem.	Christodoulou et al. (2019)	71	48		○				○		
Satellite Imaging	Nalepa et al. (2019)	17	17			○		○	○		
Tractography	Poulin et al. (2019)	4	2	○				○	○	○	○
Brain-computer Int.	Nakanishi et al. (2020)	—	1	○							○
Histopathology	Oner et al. (2020)	—	1			○					
Neuropsychiatry	Poldrack et al. (2020)	100	53	○	○				○	○	
Neuroimaging	Ahmed et al. (2021)	—	1			○					
Neuroimaging	Li et al. (2021)	122	18			○					
IT Operations	Lyu et al. (2021)	9	3			○				○	
Medicine	Filho et al. (2021)	—	1			○					
Radiology	Roberts et al. (2021)	62	16	○		○		○	○		○
Neuropsychiatry	Shim et al. (2021)	—	1		○				○		
Medicine	Vandewiele et al. (2021)	24	21		○		○	○	○	○	○
Computer Security	Arp et al. (2022)	30	22	○	○	○	○	○	○	○	○
Genomics	Barnett et al. (2022)	41	23	○							○

Figure 1. Survey of 22 papers that identify pitfalls in the adoption of ML methods across 17 fields, collectively affecting 294 papers

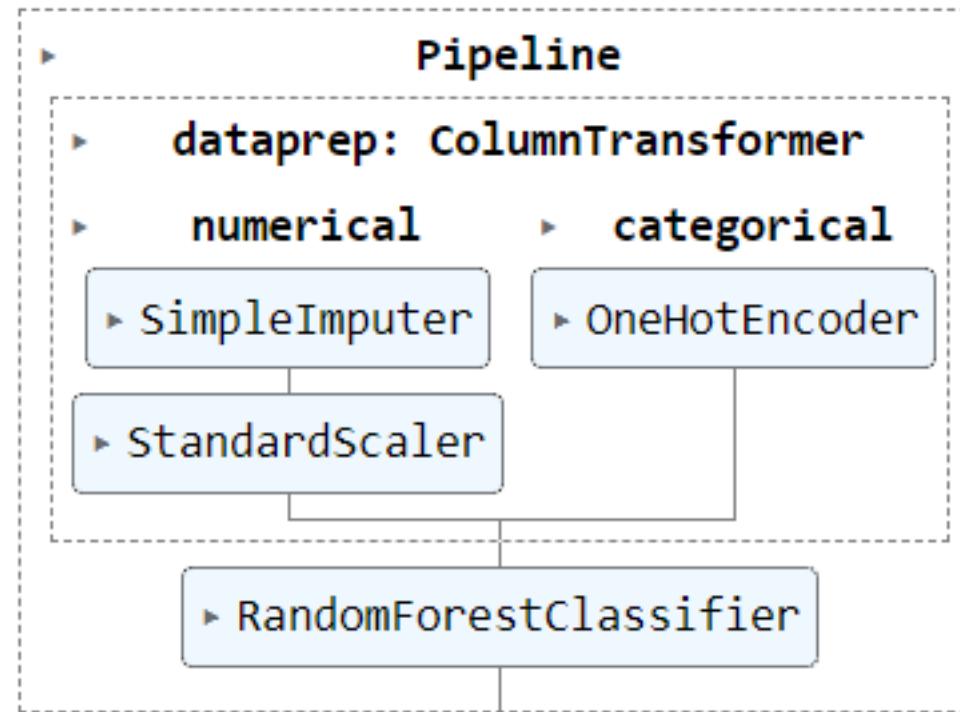
Reproducibility of ML-based Research

Taxonomy of Pitfalls

1. **Lack of clean separation of training and test set:** The training dataset has to be separated from the test dataset during all pre-processing, modeling, and evaluation steps:
 1. No test set
 2. Preprocessing based on training and test set
 3. Feature selection based on training and test set
 4. Duplicates across training and test set
2. **The model uses features that are not legitimate:** The model has access to features that should not be legitimately available for use in the modeling exercise, for instance, if they are a proxy for the outcome variable.
3. **The test set is not drawn from the distribution of interest:** The distribution of data on which the performance of an ML model is evaluated differs from the distribution of data about which the scientific claims are made.
 1. Temporal leakage
 2. Dependencies between training and test set
 3. Sampling bias in the test set

Reproducibility of ML-based Research

Scikit-learn Pipelines



Syllabus

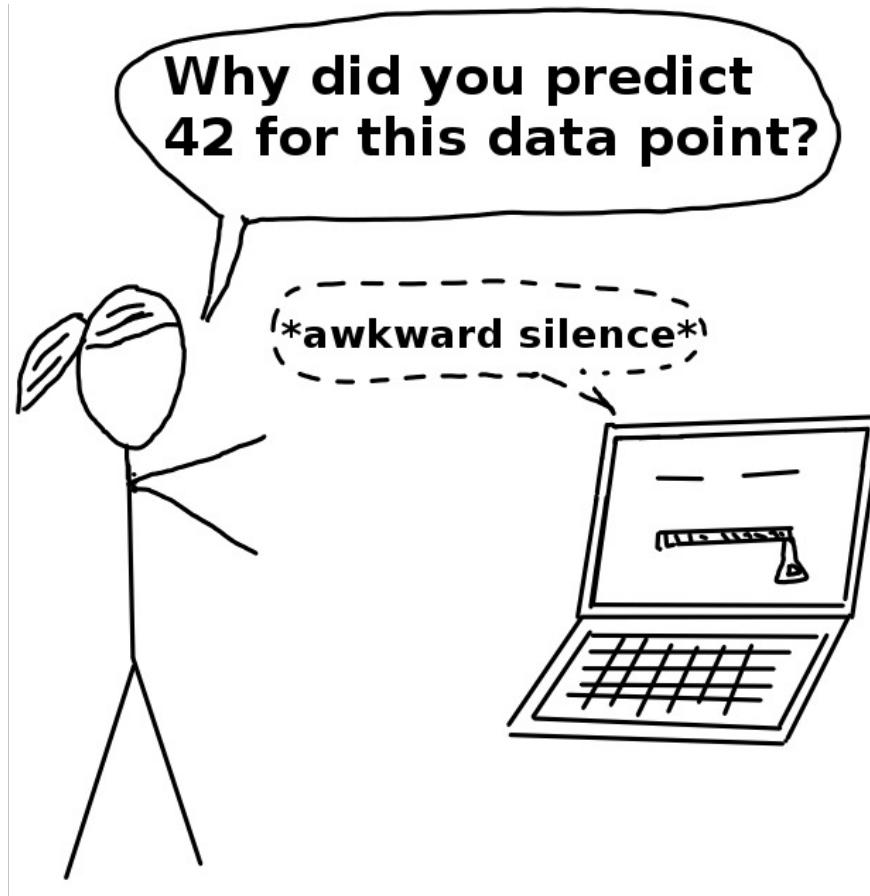
Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Model Interpretability

Model Interpretability



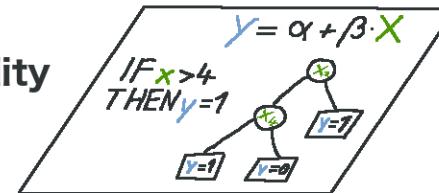
Model Interpretability

Humans



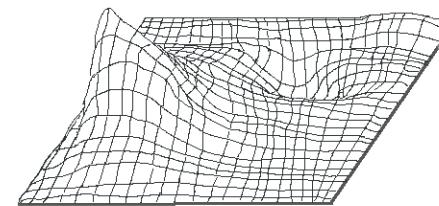
↑ inform

Interpretability
Methods



↑ extract

Black Box
Model



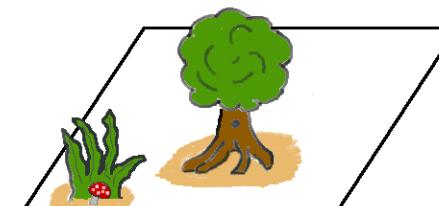
↑ learn

Data

	X_1	X_2	X_3	\dots	\dots	\dots	X_n
X_1	10	2	0				0
5	5	NA	0				0
1	-1	0					0

↑ capture

World



Model Interpretability

Dimensions of Interpretability

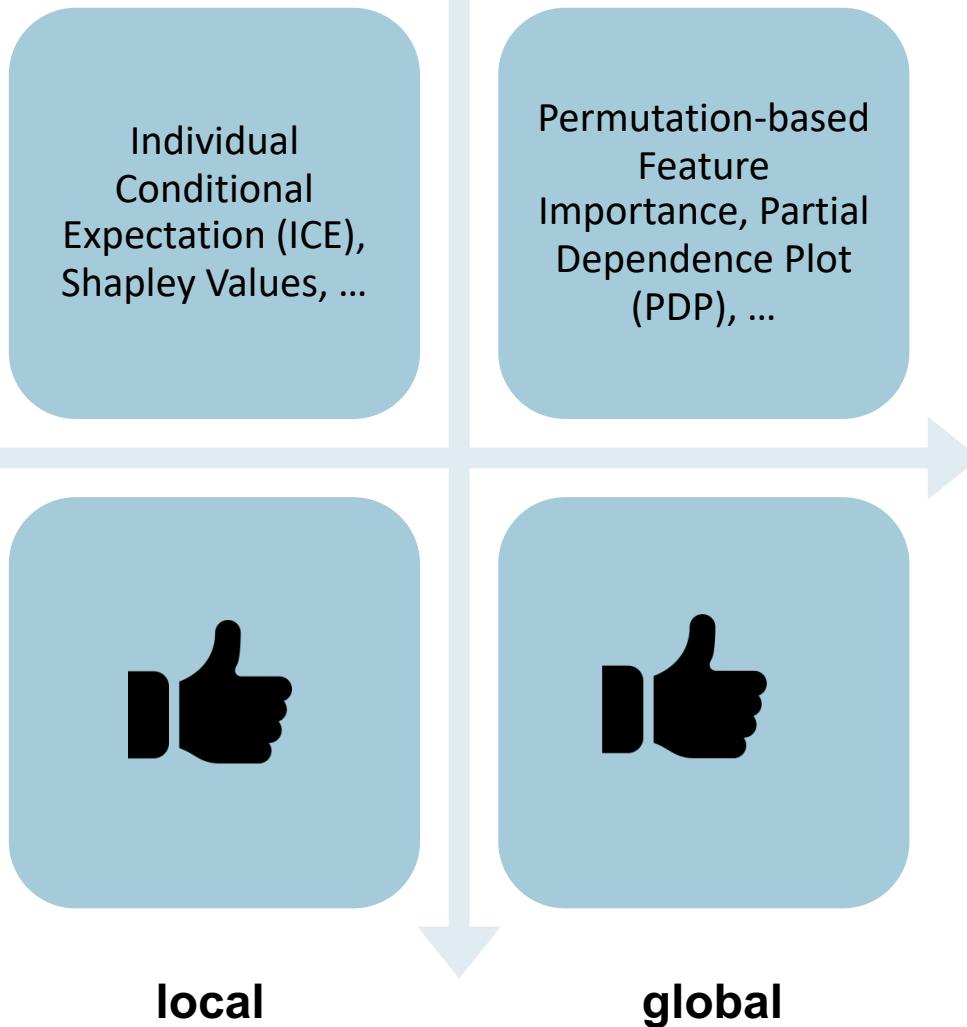
- **Type of interpretation**
 - **Inherently transparent model:** Models with human-interpretable parameters (e.g., linear/logistic regression, decision trees)
 - **Post-hoc explanations:** Human-interpretable explanations that try to emulate the logic of black-box models (e.g., random forest, boosting, neural networks)
- **Scope of interpretation**
 - **Global:** Interpretation of the general relationships between X and Y for all possible values of X
 - **Local:** Interpretation of single predictions of Y for specific values of X

Model Interpretability

Dimensions of Interpretability

post-hoc explanations

inherently transparent models



Permutation-based Feature Importance

- Originally introduced by Breiman (2001) for random forests
- The importance of a feature is measured by calculating the increase in the model's prediction error after permuting the feature.
 - A feature is “important” if shuffling its values increases the model error, because in this case the model relied on the feature for the prediction.
 - A feature is “unimportant” if shuffling its values leaves the model error unchanged, because in this case the model ignored the feature for the prediction.

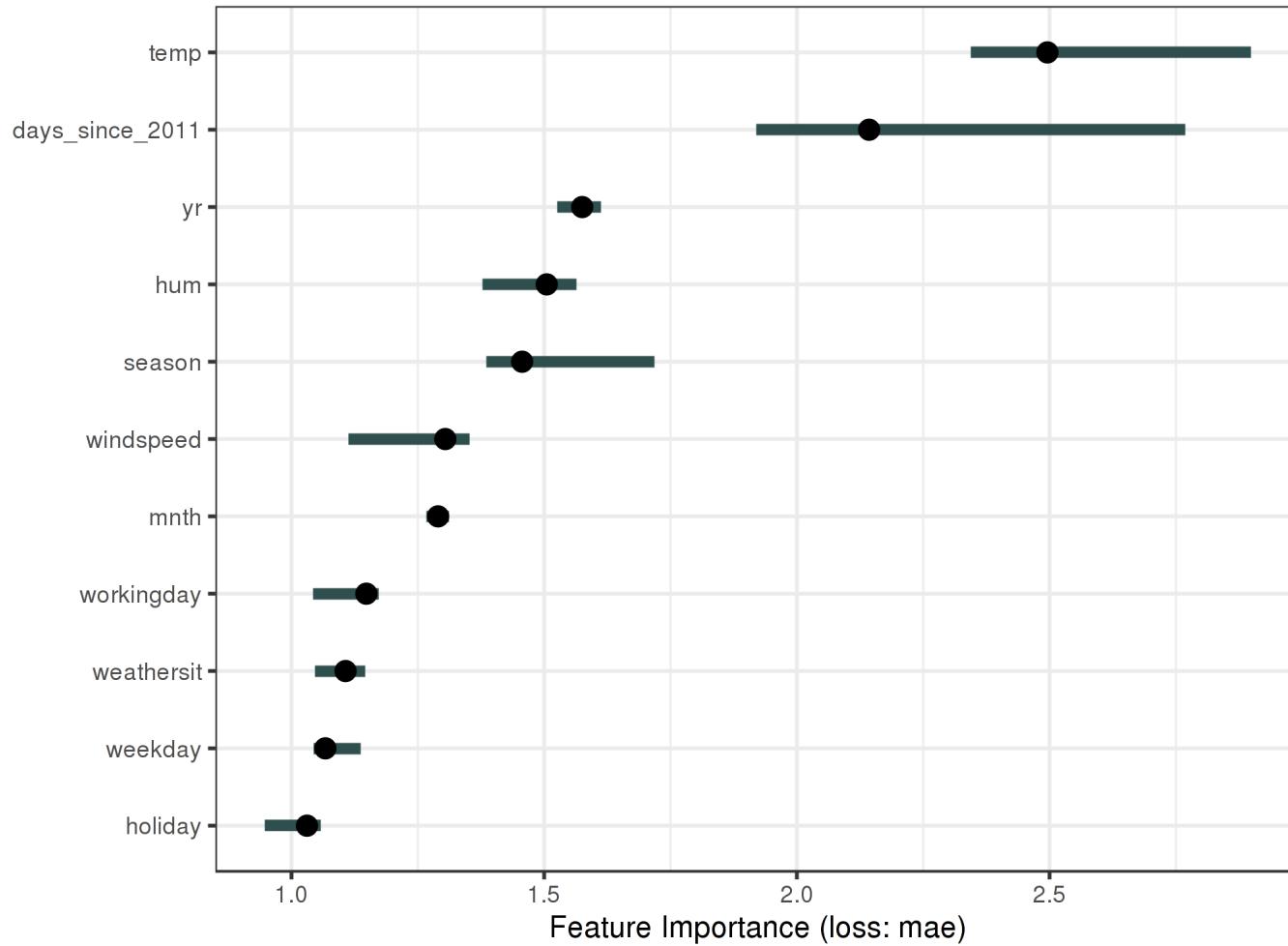
Permutation-based Feature Importance

1. Compute loss for the original model
2. For feature i in $\{1, \dots, p\}$ do
 - | Permute values of feature i
 - | Compute predictions for all observations in the training set
 - | Compute loss for permuted training set
 - | Calculate feature importance, i.e., difference or ratio between permuted loss and original loss
- End
3. Sort variables by descending feature importance

Model Interpretability



Permutation-based Feature Importance



Partial Dependence Plot (PDP)

- A PDP shows the marginal effect of one or two features on the predicted outcome of a model.
- To construct a PDP, we generate all plausible values for the one or two features of interest and hold the values of all other features constant at their average value.
- An assumption of the PDP is that the features of interest are not correlated with the other features of the model.
 - If this assumption is violated, the averages calculated for the partial dependence plot will include data points that are very unlikely or even impossible.

Model Interpretability

Partial Dependence Plot (PDP)

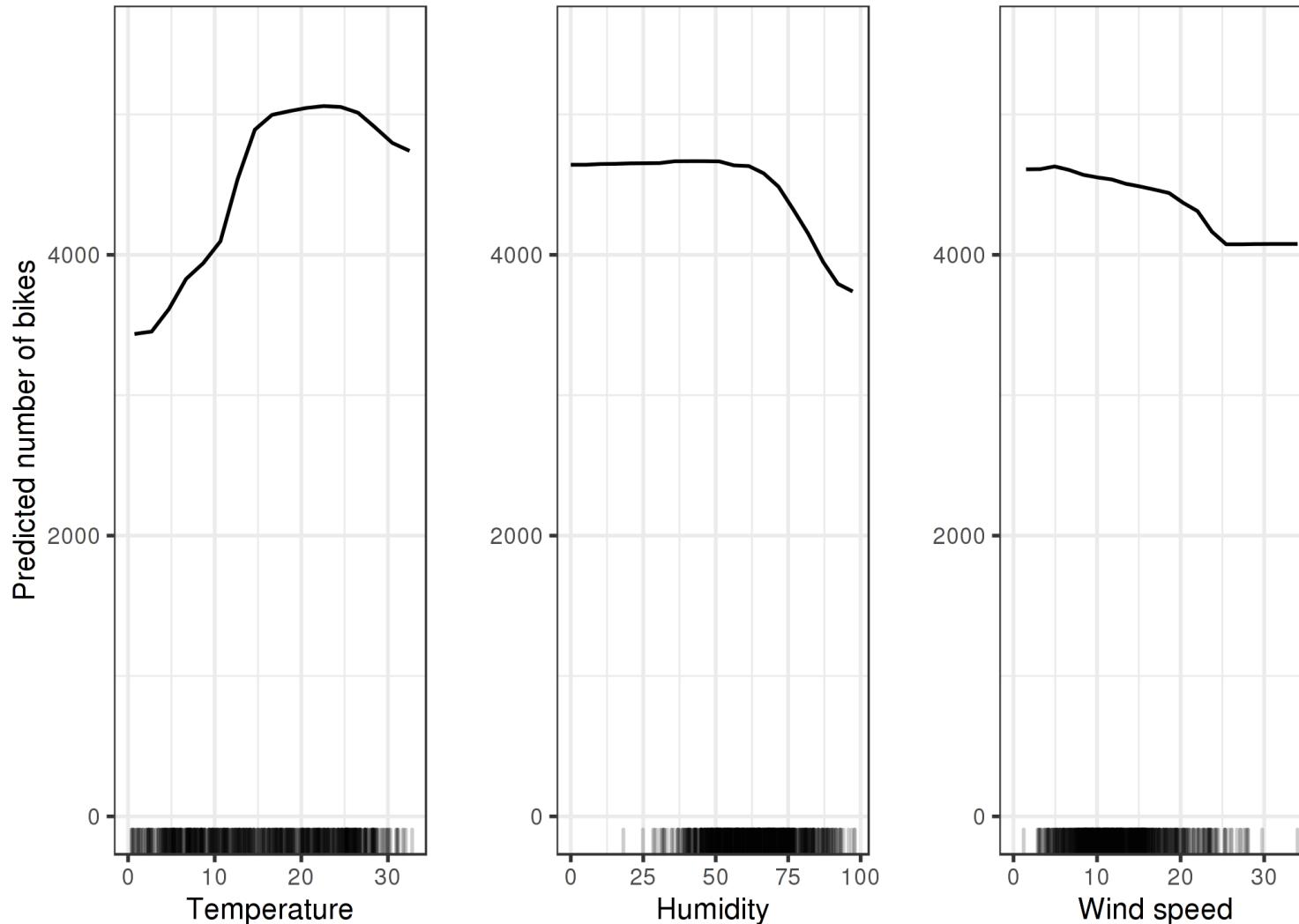
For a selected predictor (x)

1. Construct a grid of j evenly spaced values across the distribution of x : $\{x_1, x_2, \dots, x_j\}$
2. For i in $\{1, \dots, j\}$ do
 - | Copy the training data and replace the original values of x with the constant x_i
 - | Compute predictions for all observations
 - | Average predictions
- End
3. Plot the averaged predictions against x_1, x_2, \dots, x_j

Model Interpretability



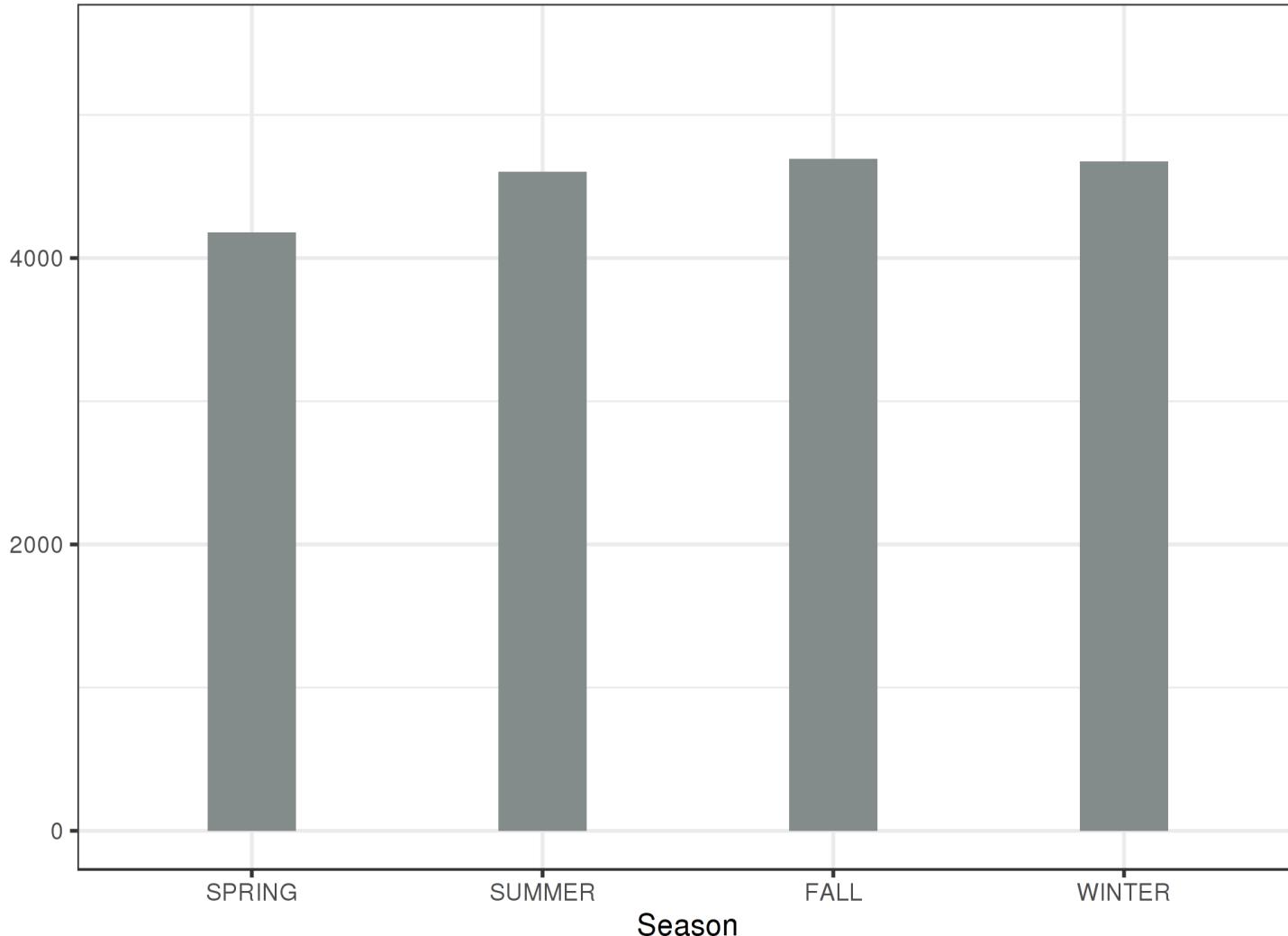
Partial Dependence Plot (PDP)



Model Interpretability

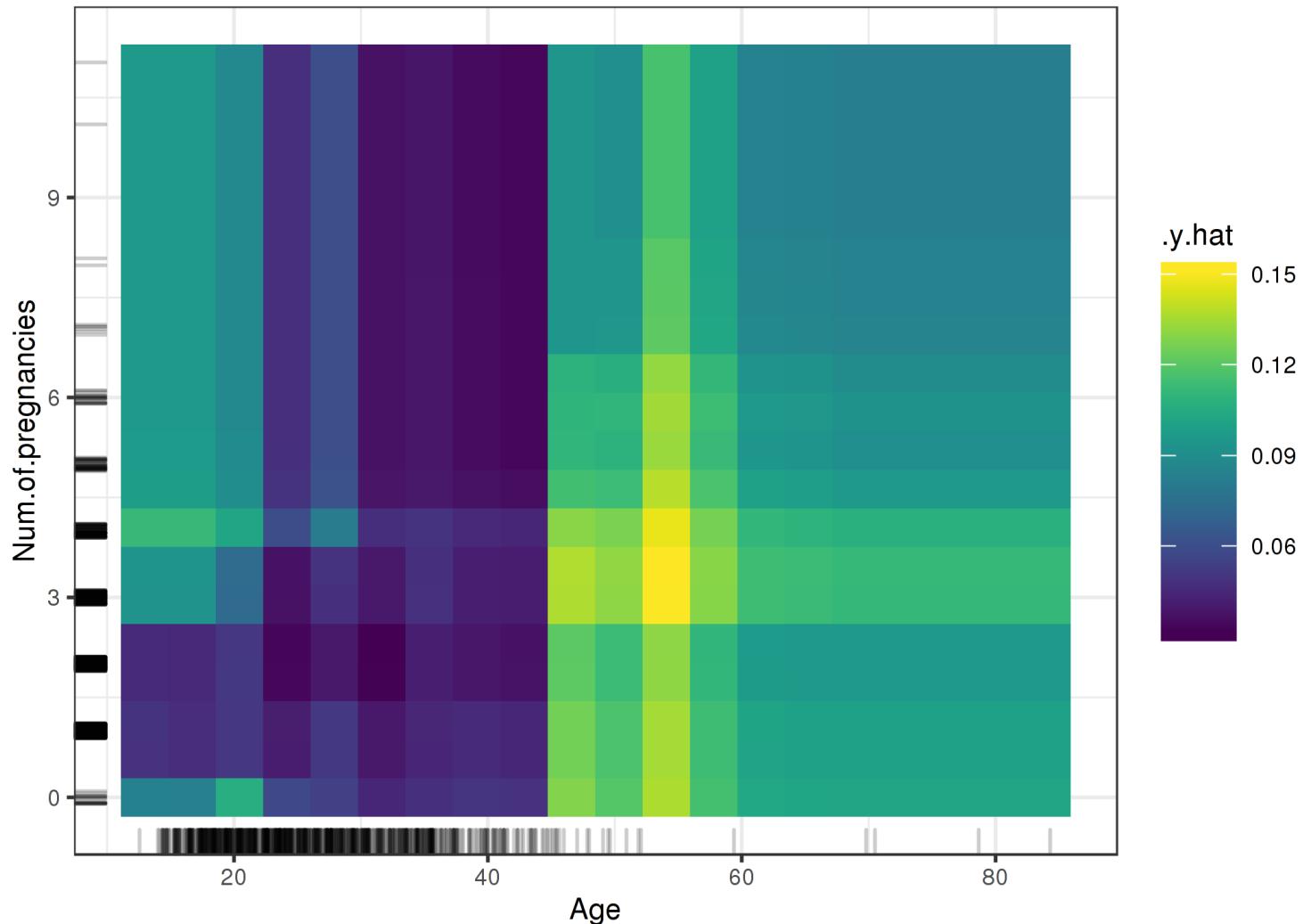


Partial Dependence Plot (PDP)



Model Interpretability

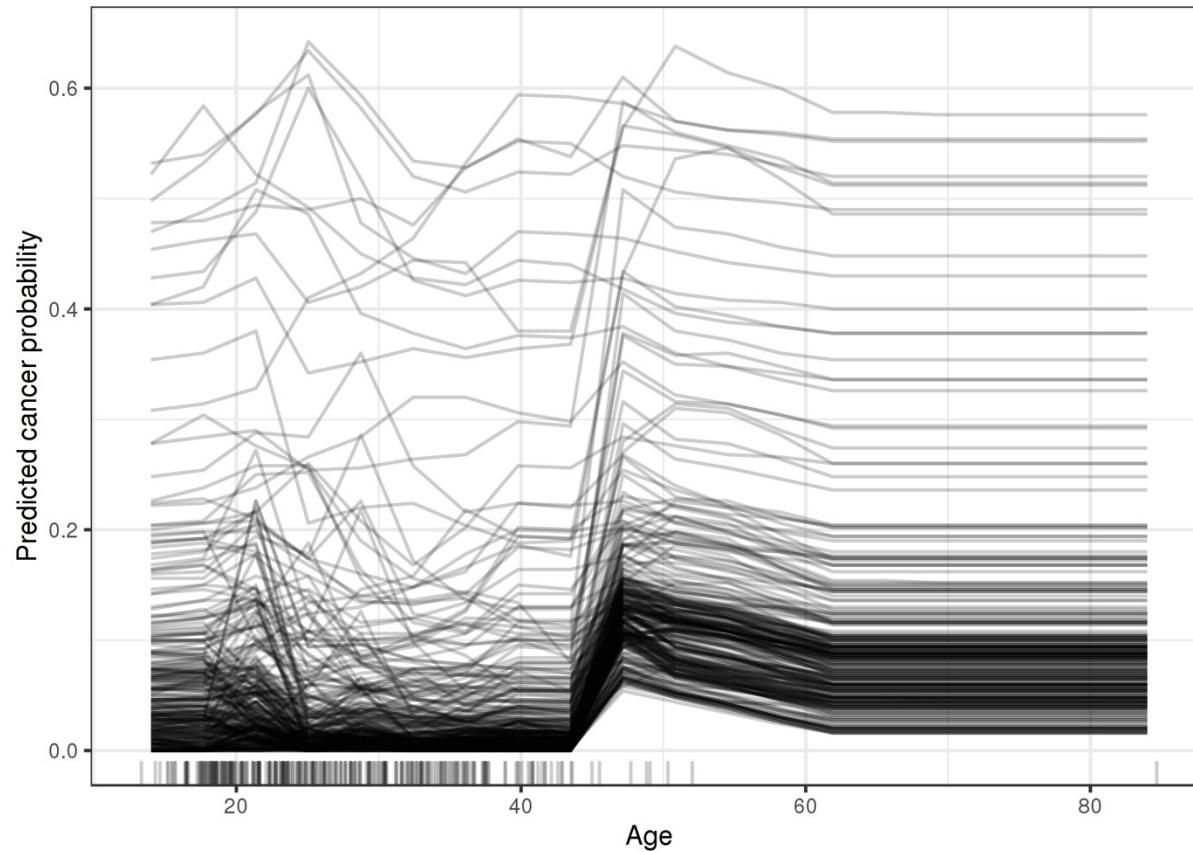
Partial Dependence Plot (PDP)



Model Interpretability

Individual Conditional Expectation (ICE)

- The individual conditional expectation (ICE) plot is the equivalent of a PDP for individual prediction instances.



Model Interpretability

Individual Conditional Expectation (ICE)

```
For a selected predictor (x)
1. Construct a grid of j evenly spaced values across the distribution
   of x: {x1, x2, ..., xj}
2. For i in {1,...,j} do
   | Copy the training data and replace the original values of x
     with the constant xi
   | Compute predictions for all observations
End
3. Plot the predictions against x1, x2, ..., xj with lines connecting
   observations that correspond to the same row number in the original
   training data
```

Model Interpretability

Shapley Values

- Idea adapted from game theory
 - The “game” is the prediction task for a single instance of the dataset.
 - Each feature value of the instance is a “player” in the game.
 - The “payout” is the actual prediction for this instance minus the average prediction over all instances.

Model Interpretability

Shapley Values

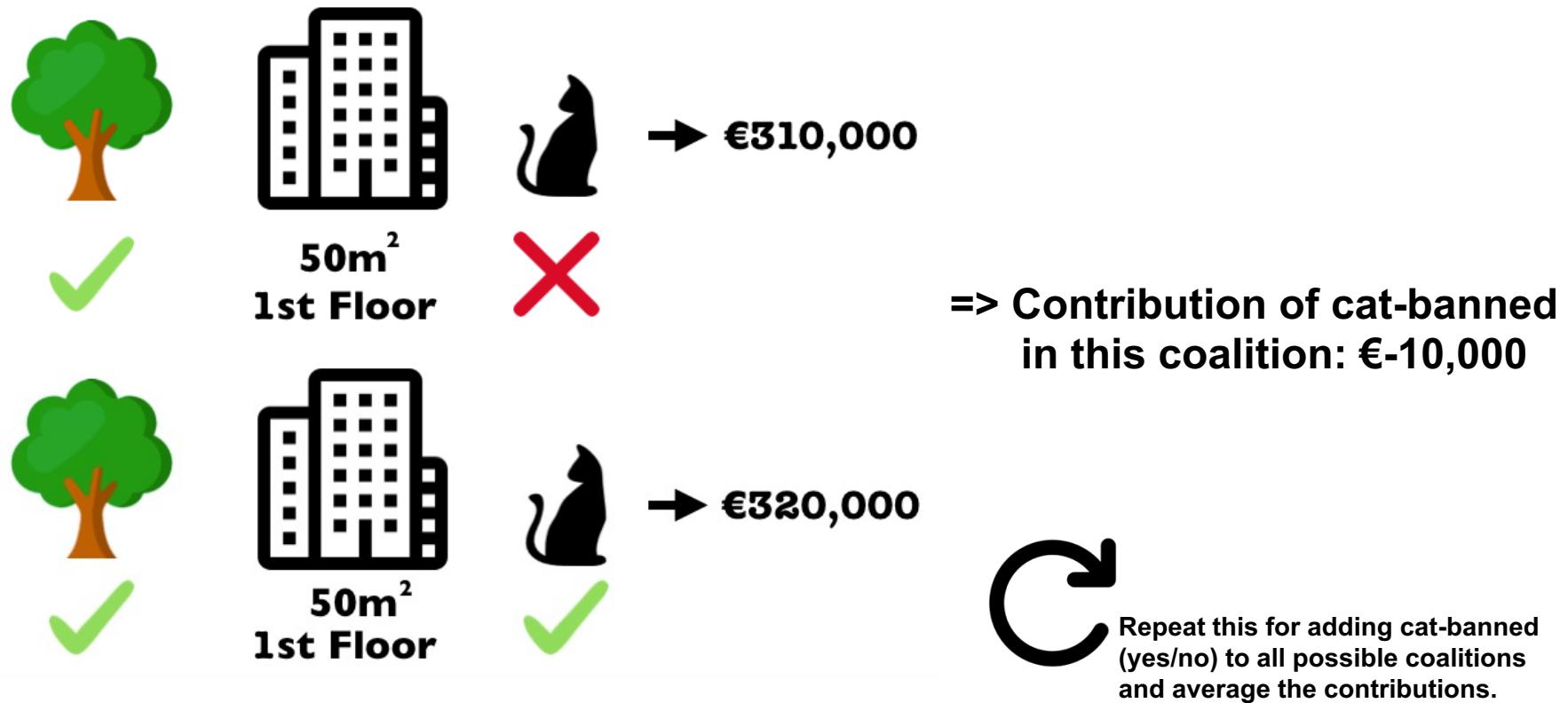
- The predicted price for a 50 m² 2nd floor apartment with a nearby park and cat ban is €300,000.
- The average prediction over all apartments is €310,000.
- How much has each feature value (player) contributed to the €10,000 price decrease (payout)?



Model Interpretability

Shapley Values

- The Shapley value is the average marginal contribution of a feature value across all possible coalitions.
- Example: Cat-banned

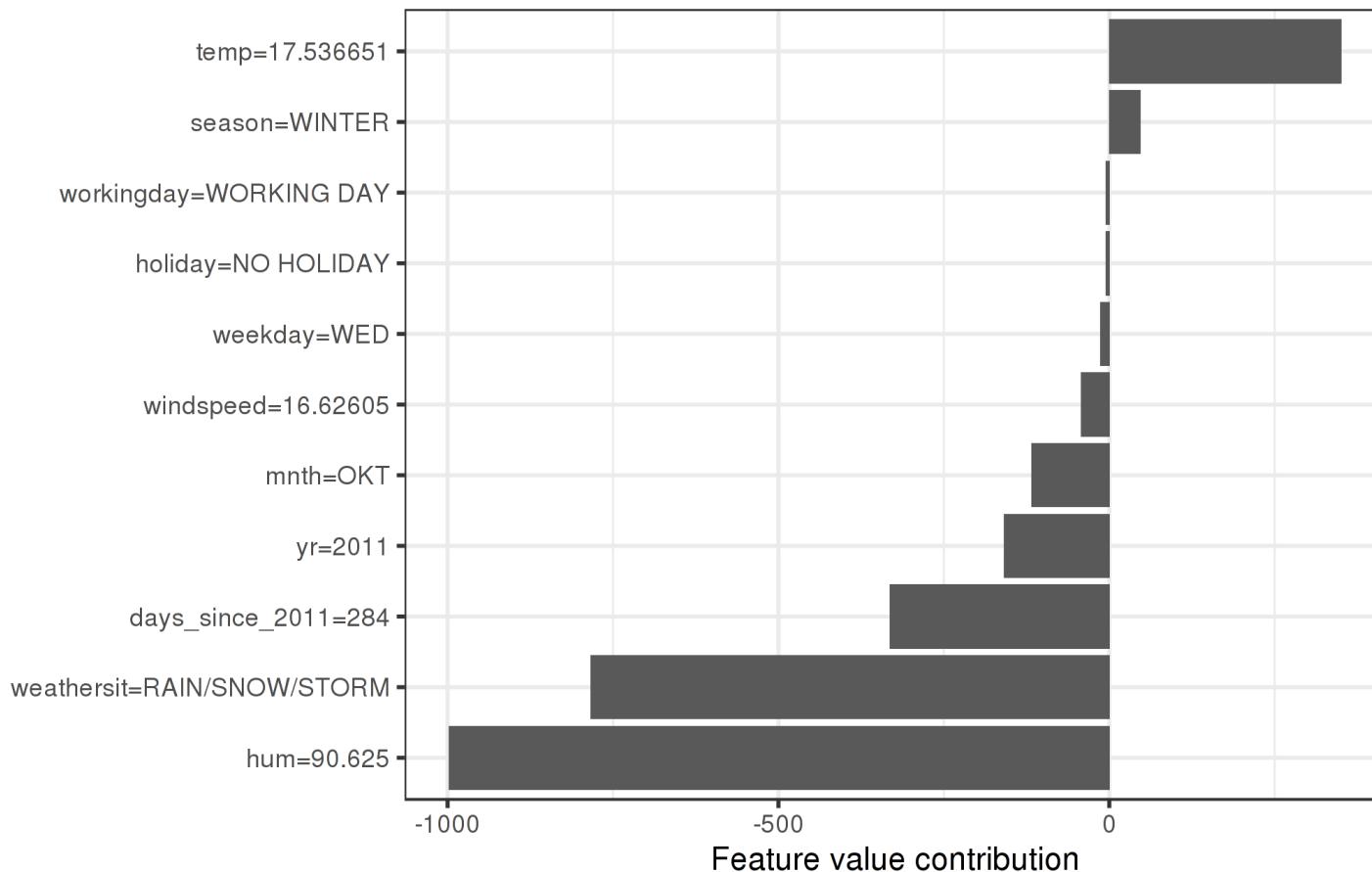


Model Interpretability



Shapley Values

Actual prediction: 2475
Average prediction: 4516
Difference: -2041



Can you predict the sales price of a house?





Hands-on Exercise

Expected Goals (xG)





Syllabus

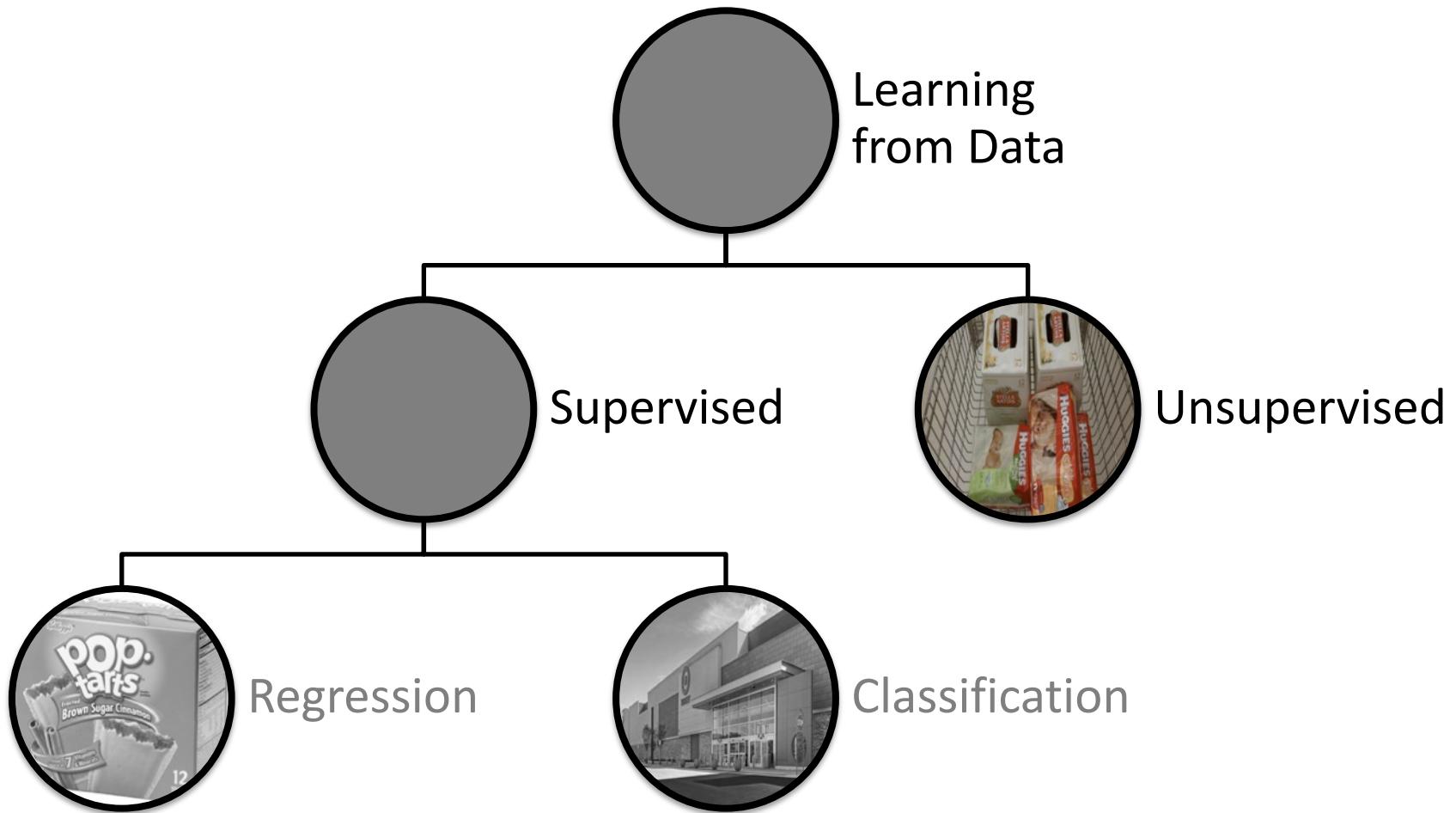
Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Unsupervised Learning

Unsupervised Learning



Remember

- Situation
 - We have **only a set of features** X_1, X_2, \dots, X_p measured on n observations
 - We are **not interested in predicting a target variable** Y , because we do not have such a variable
 - **Instead**, we want to **learn interesting things** about the given dataset
- Challenge
 - Due to the lack of a target, it can be very difficult to evaluate the results of unsupervised model

Unsupervised Learning

Applications

- Exemplary application areas
 - Biology and medicine
 - Marketing
- Unsupervised learning is often done as part of
 - Exploratory data analysis
 - Feature engineering for supervised learning

Syllabus

Unsupervised Learning:
_ Clustering

Introduction

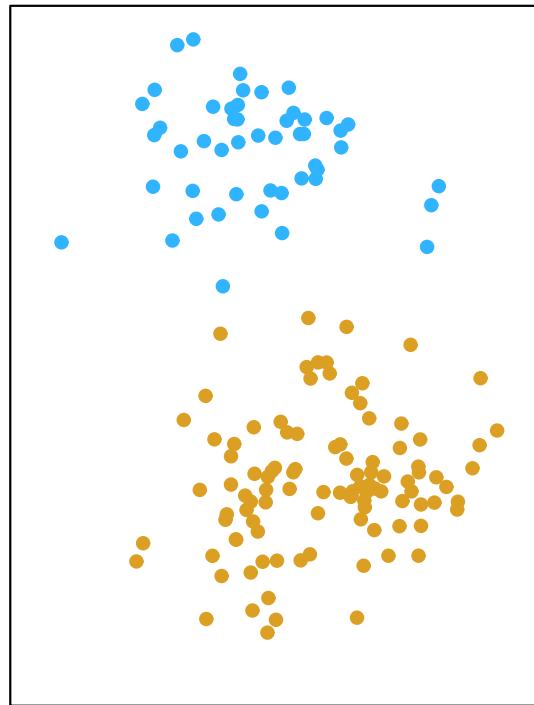
- Clustering refers to a broad class of techniques for **finding subgroups**, or clusters, in a dataset
- We seek to partition a data set into distinct groups so that the **observations within each group are similar** to each other, while **observations in different groups are different** from each other



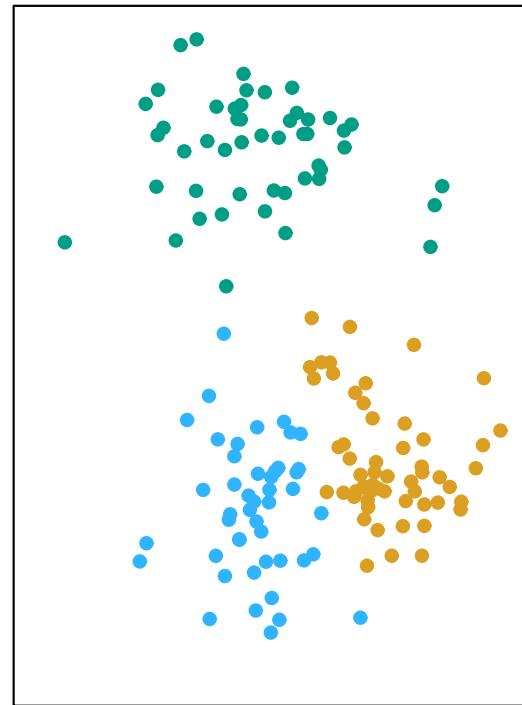
Clustering

K-Means Clustering

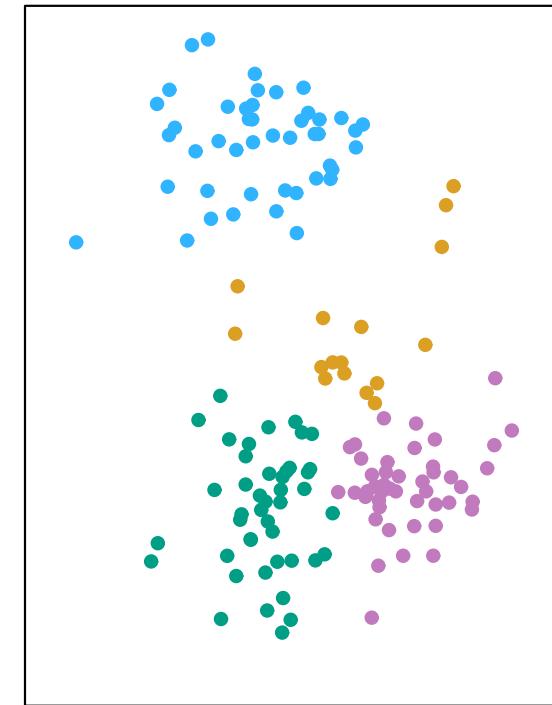
K=2



K=3



K=4



K-Means Clustering

- Goal: Partitioning a data set into k distinct, non-overlapping clusters
- For a given number k , the algorithm assigns each observation to exactly one cluster
- A good clustering is one for which the **within-cluster variation $W(C_k)$ is as small as possible**, i.e. the amount by which observations within a cluster differ from each other
- Hence, we need to solve the problem:

$$\underset{C_1, \dots, C_K}{\text{minimize}} \left\{ \sum_{k=1}^K W(C_k) \right\}$$

Within-cluster Variation

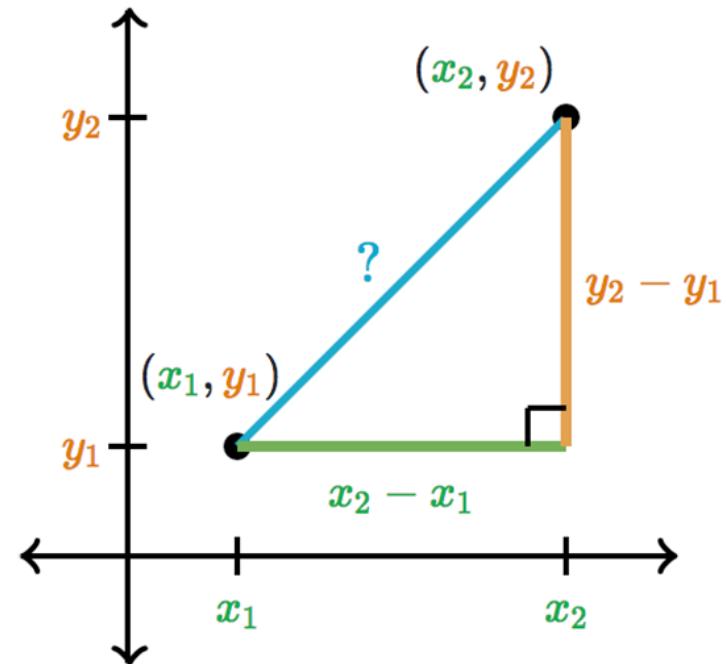
- Typically, the **squared Euclidean distance** is used a concrete measure of within-cluster variation:

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^p (x_{ij} - x_{i'j})^2,$$

- Where $|C_k|$ denotes the number of observations in cluster k and p denotes the number of attributes which describe each observation.

Euclidean Distance in 2D

- The Euclidean distance is the straight-line distance between two points in an Euclidean space.



$$?^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$$

Challenge

There are almost K^n ways
to partition n observations into K clusters.

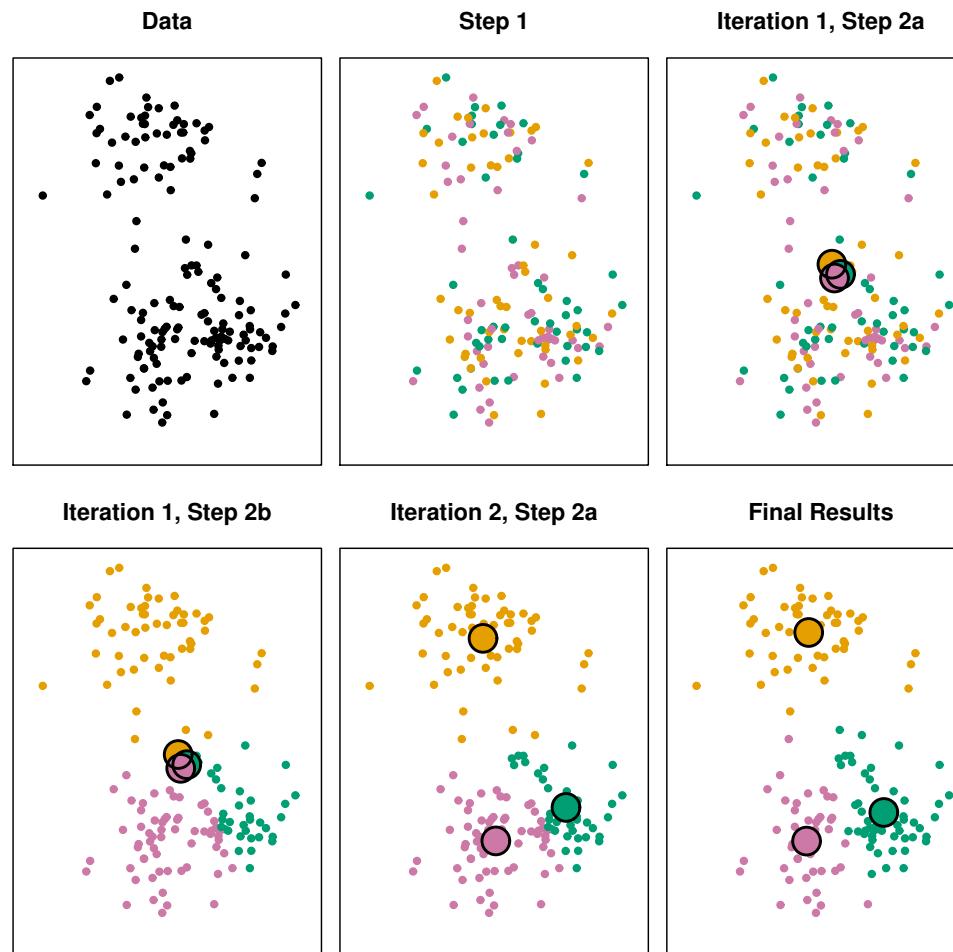
K-Means Algorithm

Algorithm 10.1 *K*-Means Clustering

1. Randomly assign a number, from 1 to K , to each of the observations. These serve as initial cluster assignments for the observations.
 2. Iterate until the cluster assignments stop changing:
 - (a) For each of the K clusters, compute the cluster *centroid*. The k th cluster centroid is the vector of the p feature means for the observations in the k th cluster.
 - (b) Assign each observation to the cluster whose centroid is closest (where *closest* is defined using Euclidean distance).
-

Clustering

K-Means Algorithm (k=3)



Clustering

Attention: Local, rather than global optimum!

- The K-Means algorithm finds only a locally optimal solution, not the global optimum.
- Hence, it is important to **run the algorithm multiple times** and select the solution with the **smallest within-cluster variation**.

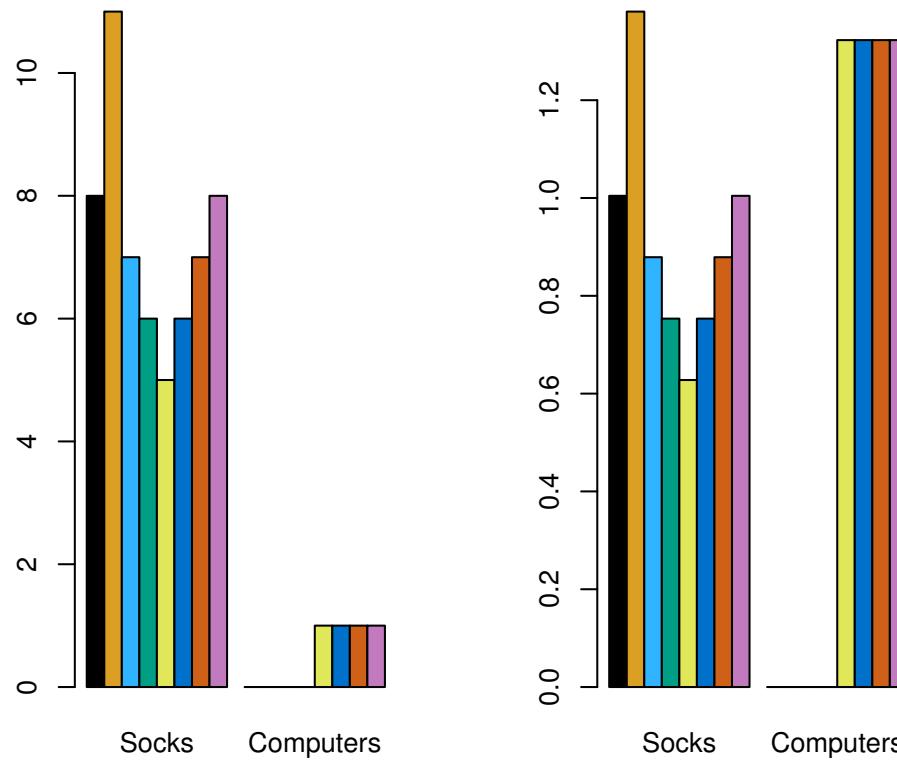


Transforming Features

- Often, variables are measured in very different units of measurement or have very different values
 - Minutes played versus goals scored for a football player
 - Sales of plastic bags versus sales of champagne in a supermarket
- These differences can have a huge influence on the results of clustering (and other algorithms)
- Typical transformations:
 - **Centering:** Transforming a variable so that it has a mean of 0
 - **Scaling:** Transforming a variable so that it has a standard deviation of 1
 - **Standardizing:** Centering & Scaling

Clustering

Example



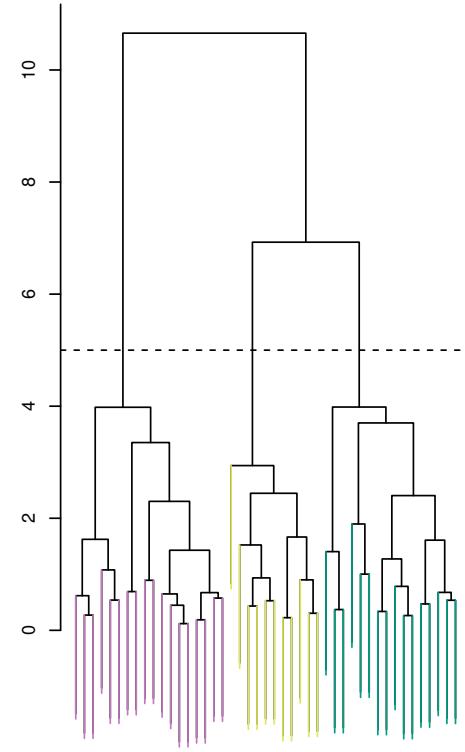
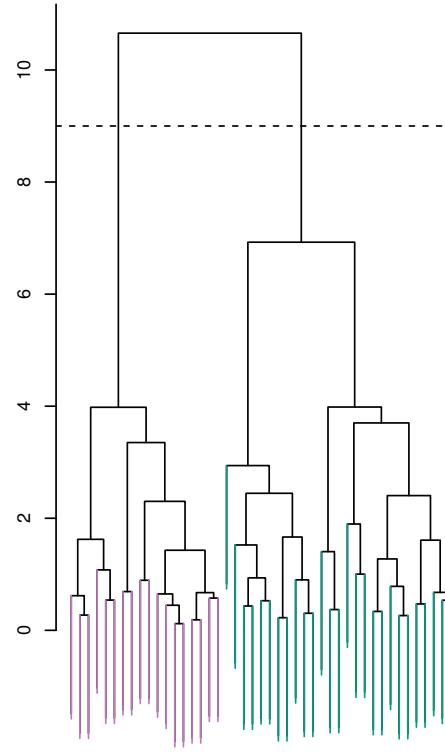
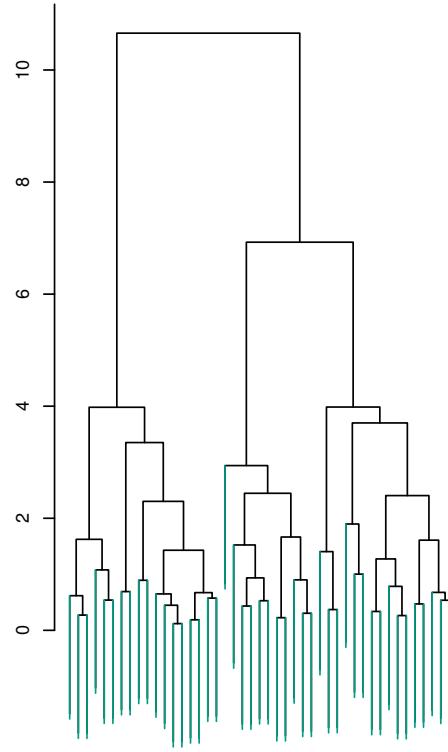
Learning about Protein Consumption



Hierarchical Clustering

- A disadvantage of K-means clustering is that it requires us to predefine the number of clusters K .
- Hierarchical clustering is an alternative clustering method that does not require to commit to a fixed choice of K .
- In addition, hierarchical clustering produces an attractive tree-based representation of the clustering (dendrogram)
- In the following, we will look at **bottom-up or agglomerative** hierarchical clustering

Hierarchical Clustering: Dendrogram

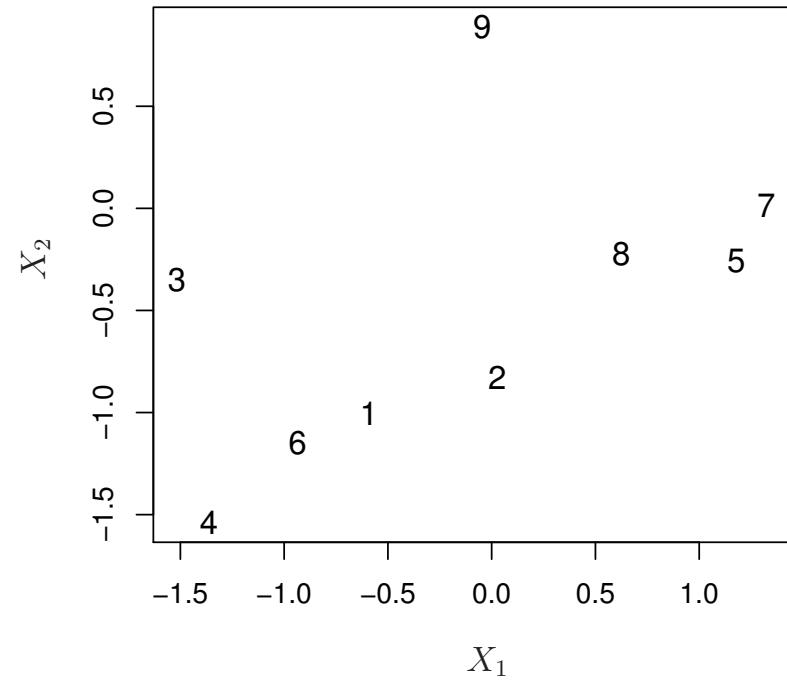
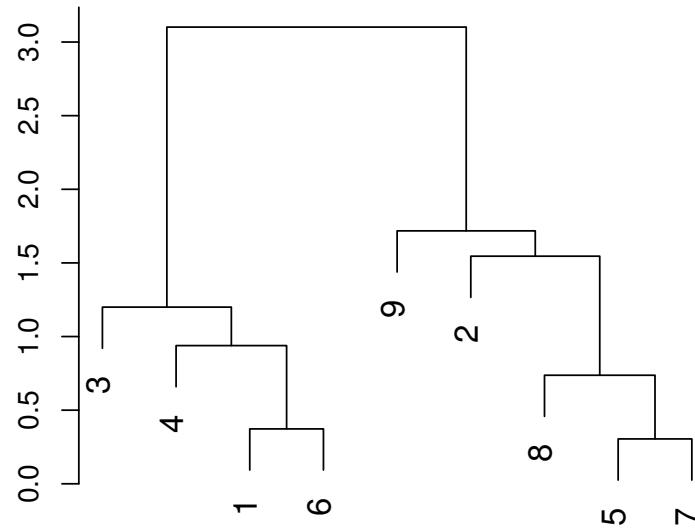


Hierarchical Clustering: Dendrogram

- Vertical axis
 - The earlier (from bottom to top) fusions occur, the more similar the groups of observations are
- Horizontal axis
 - No interpretation!!

Clustering

Hierarchical Clustering: Dendrogram

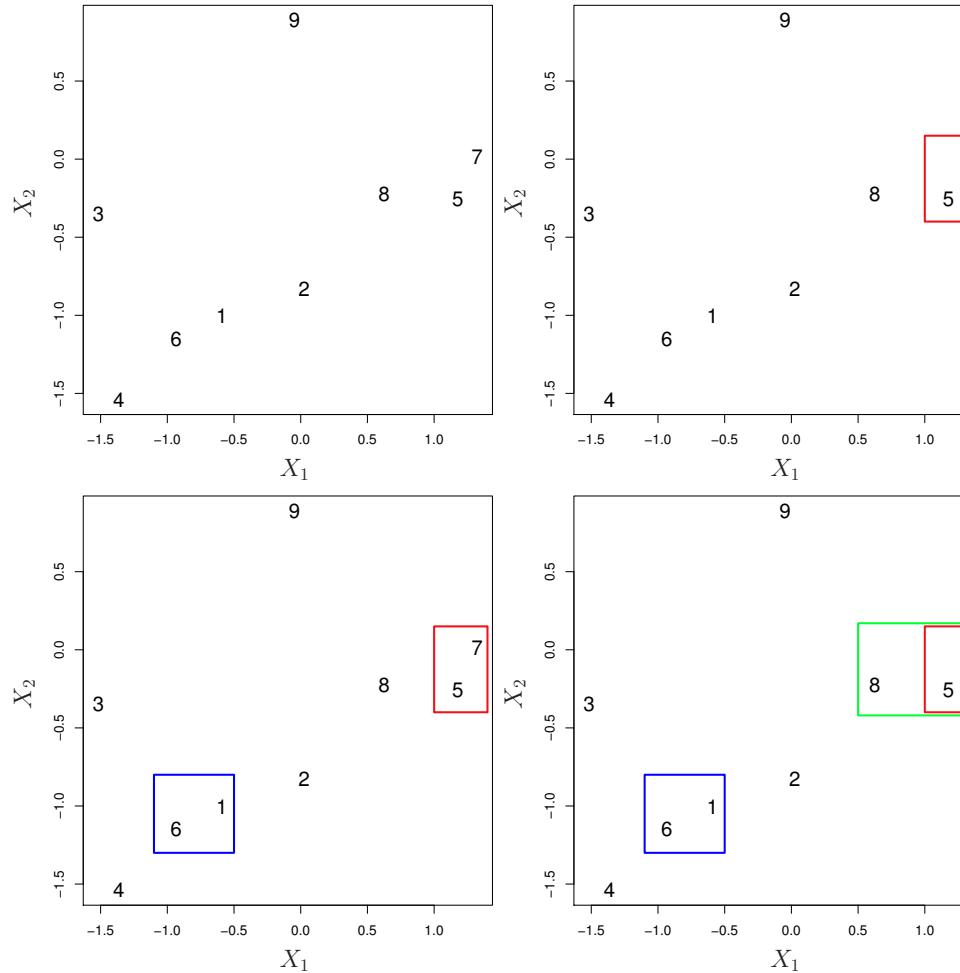


Hierarchical Clustering Algorithm

Algorithm 10.2 *Hierarchical Clustering*

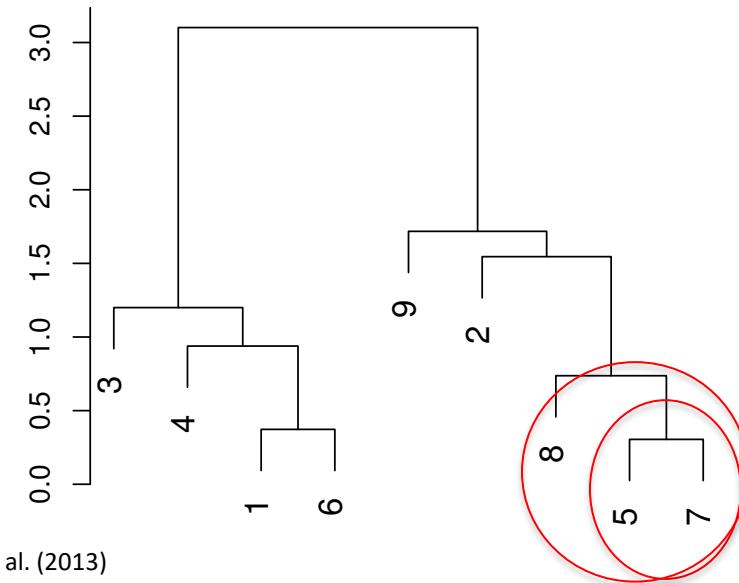
1. Begin with n observations and a measure (such as Euclidean distance) of all the $\binom{n}{2} = n(n - 1)/2$ pairwise dissimilarities. Treat each observation as its own cluster.
 2. For $i = n, n - 1, \dots, 2$:
 - (a) Examine all pairwise inter-cluster dissimilarities among the i clusters and identify the pair of clusters that are least dissimilar (that is, most similar). Fuse these two clusters. The dissimilarity between these two clusters indicates the height in the dendrogram at which the fusion should be placed.
 - (b) Compute the new pairwise inter-cluster dissimilarities among the $i - 1$ remaining clusters.
-

Hierarchical Clustering Algorithm



Similarity between Clusters

- We have a simple method for determining similarity between clusters consisting of individual observations (e.g., Euclidean distance)
- But how can we determine the similarity between two clusters, if one of them (or both) contain multiple observations?

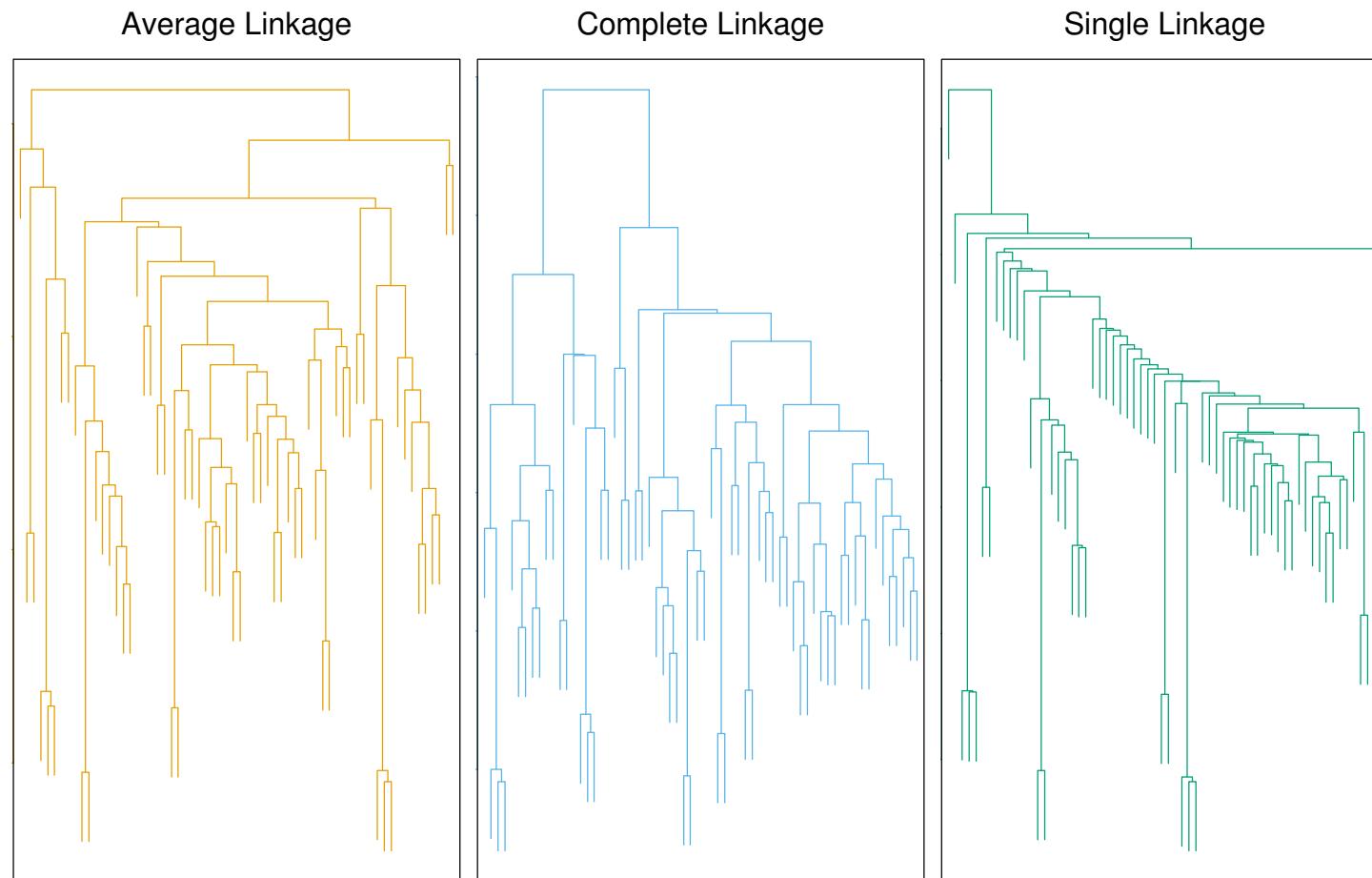


Linkage

<i>Linkage</i>	<i>Description</i>
Complete	Maximal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the <i>largest</i> of these dissimilarities.
Single	Minimal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the <i>smallest</i> of these dissimilarities. Single linkage can result in extended, trailing clusters in which single observations are fused one-at-a-time.
Average	Mean intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the <i>average</i> of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable <i>inversions</i> .

TABLE 10.2. A summary of the four most commonly-used types of linkage in hierarchical clustering.

Linkage

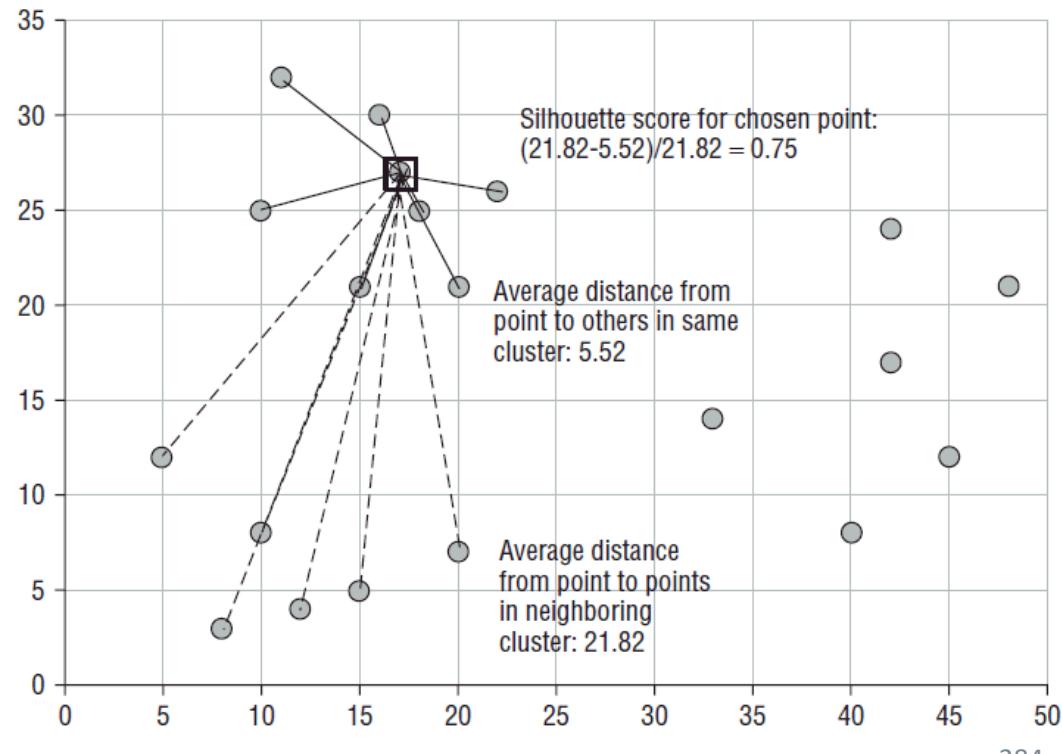


Evaluating Clusterings with the Silhouette Score

- Let $a(i)$ be the average distance between i and all other data points within the same cluster.
- Let $b(i)$ be the average distance of i to all data points in the neighboring cluster.

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

$$-1 \leq s(i) \leq 1$$



Learning about Protein Consumption



Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

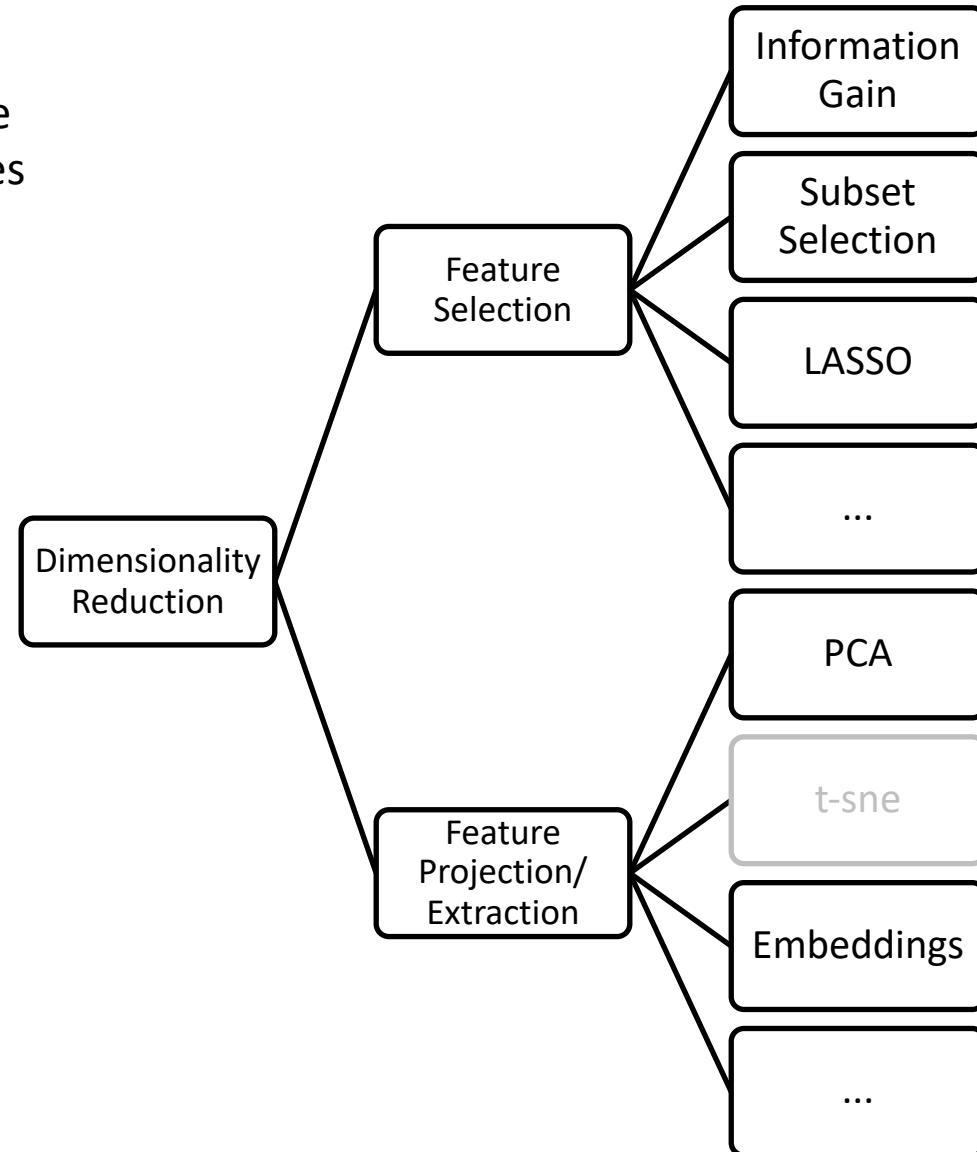
Agenda

Unsupervised Learning:
_ Dimensionality Reduction

Dimensionality Reduction

Overview

- The process of reducing the number of random variables under consideration by obtaining a set of principal variables.



Dimensionality Reduction

Feature Projection/Extraction



Dimensionality Reduction

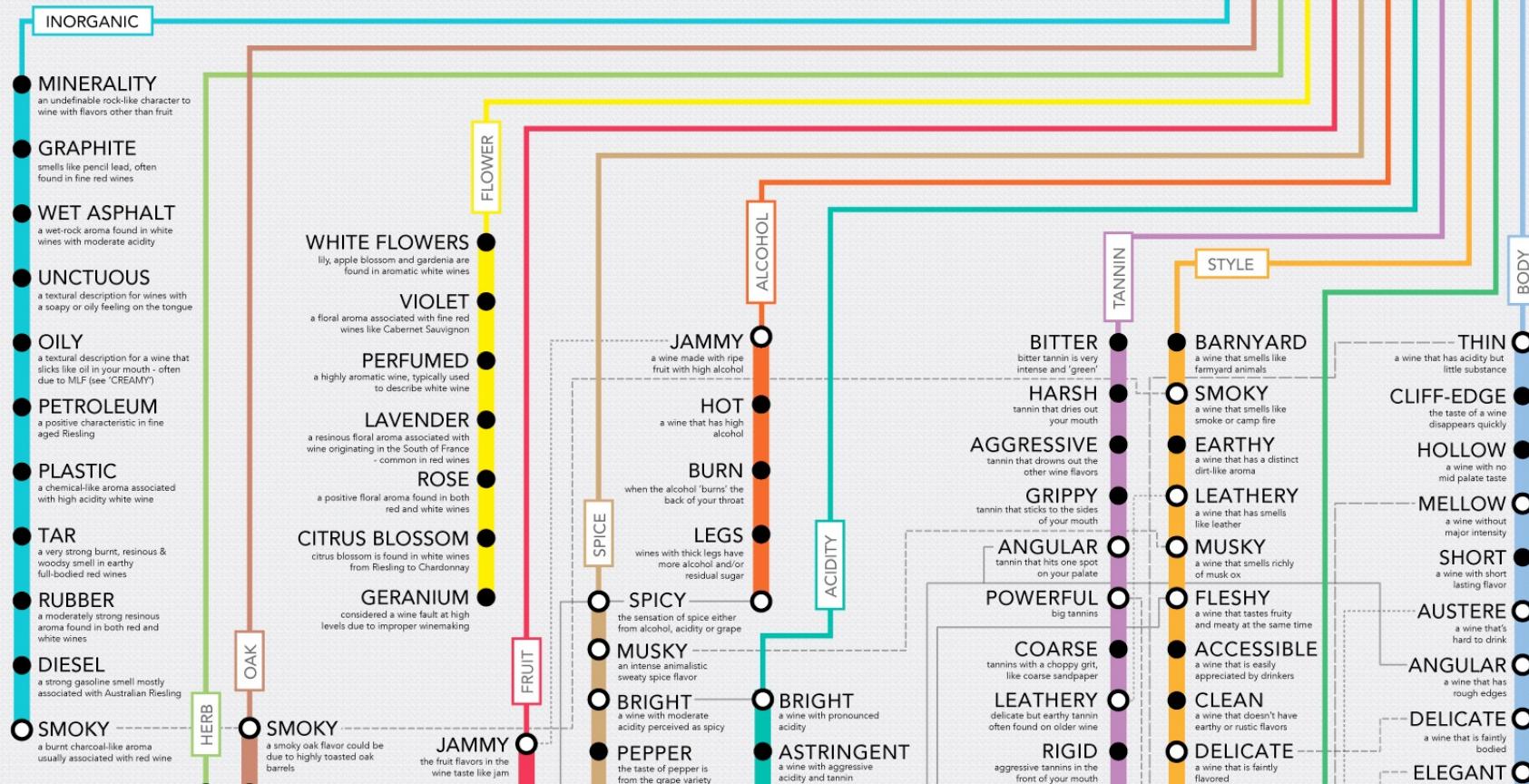
Principal Components Analysis (PCA)

- When faced with a **large set of correlated variables**, PCA allows us to **summarize this set with a smaller number of representative variables** that collectively explain most of the variability in the original set
- In other words: PCA finds a **low-dimensional representation** of a data set without loosing too much information
- Applications
 - Visualization
 - Feature engineering for supervised learning

WINE DESCRIPTIONS

& WHAT THEY MEAN

wine descriptions can be divided into
TWELVE CATEGORIES



Dimensionality Reduction

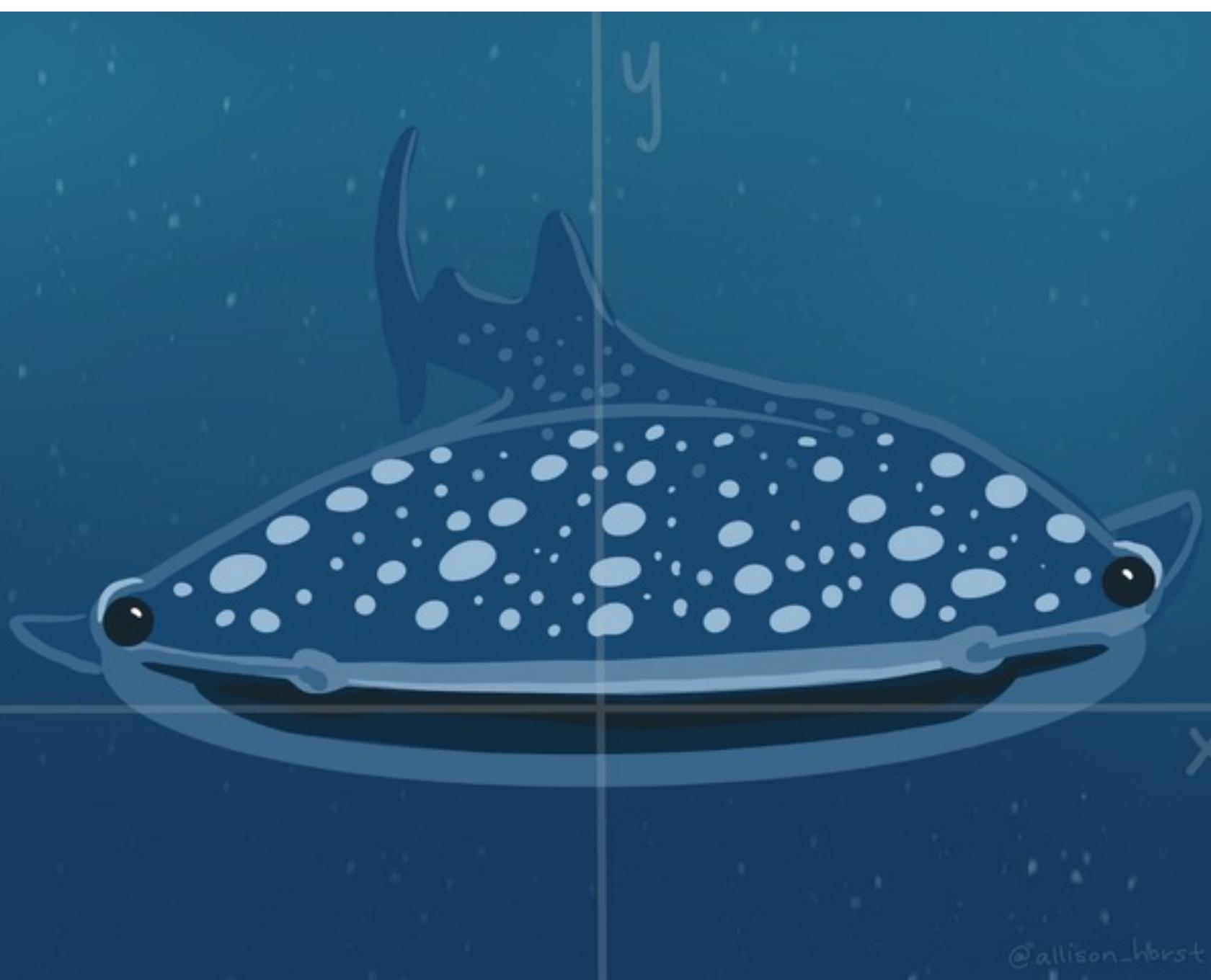
Finding Principal Components

- Principal components are **linear combinations** of the original features
- For example, the first principal component looks like this:

$$z_{i1} = \phi_{11}x_{i1} + \phi_{21}x_{i2} + \dots + \phi_{p1}x_{ip}$$

- PCA finds linear combinations of features that show **as much variation across observations as possible**
- Therefore, we have to solve the following optimization problem:

$$\underset{\phi_{11}, \dots, \phi_{p1}}{\text{maximize}} \left\{ \frac{1}{n} \sum_{i=1}^n \left(\sum_{j=1}^p \phi_{j1} x_{ij} \right)^2 \right\} \text{ subject to } \sum_{j=1}^p \phi_{j1}^2 = 1$$



y

x



y

x

@allison_horst

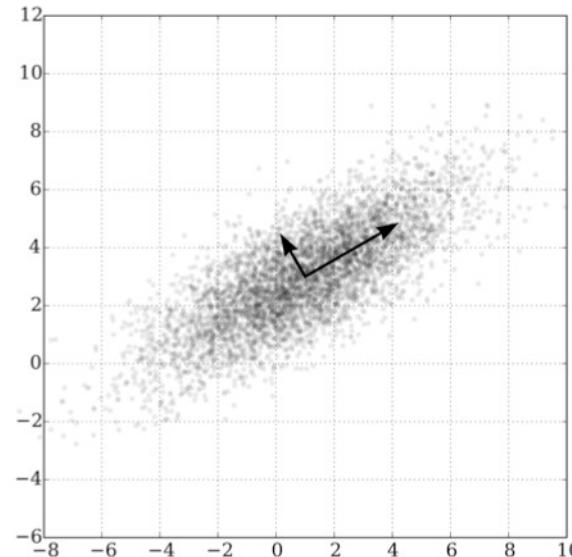
Dimensionality Reduction

Finding Principal Components

- After the first principal component of a data set has been found, we can find the second principal component:

$$z_{i2} = \phi_{12}x_{i1} + \phi_{22}x_{i2} + \dots + \phi_{p2}x_{ip}$$

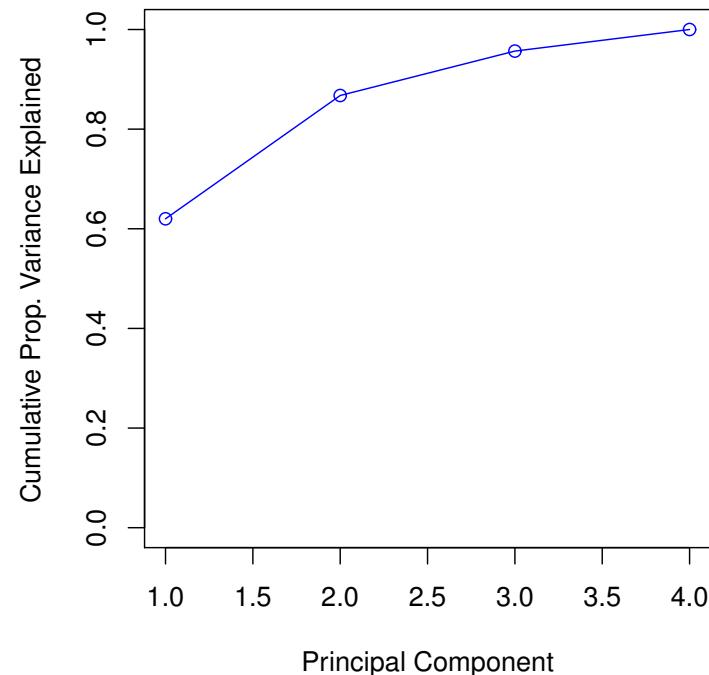
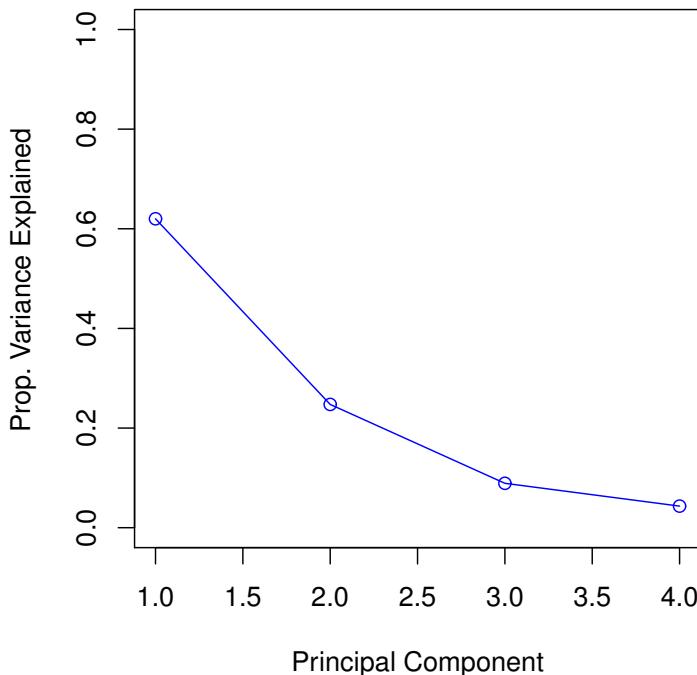
- The second principal component Z_2 is the linear combination that has maximal variance across observations out of all linear combinations that are uncorrelated (orthogonal) with Z_1 .



Dimensionality Reduction

Proportion of Variance Explained by Principal Components

- How much of the information in a given data set is lost by reducing the number of dimensions?



Learning about Protein Consumption





Your Task



BUNDESLIGA



Syllabus

Syllabus

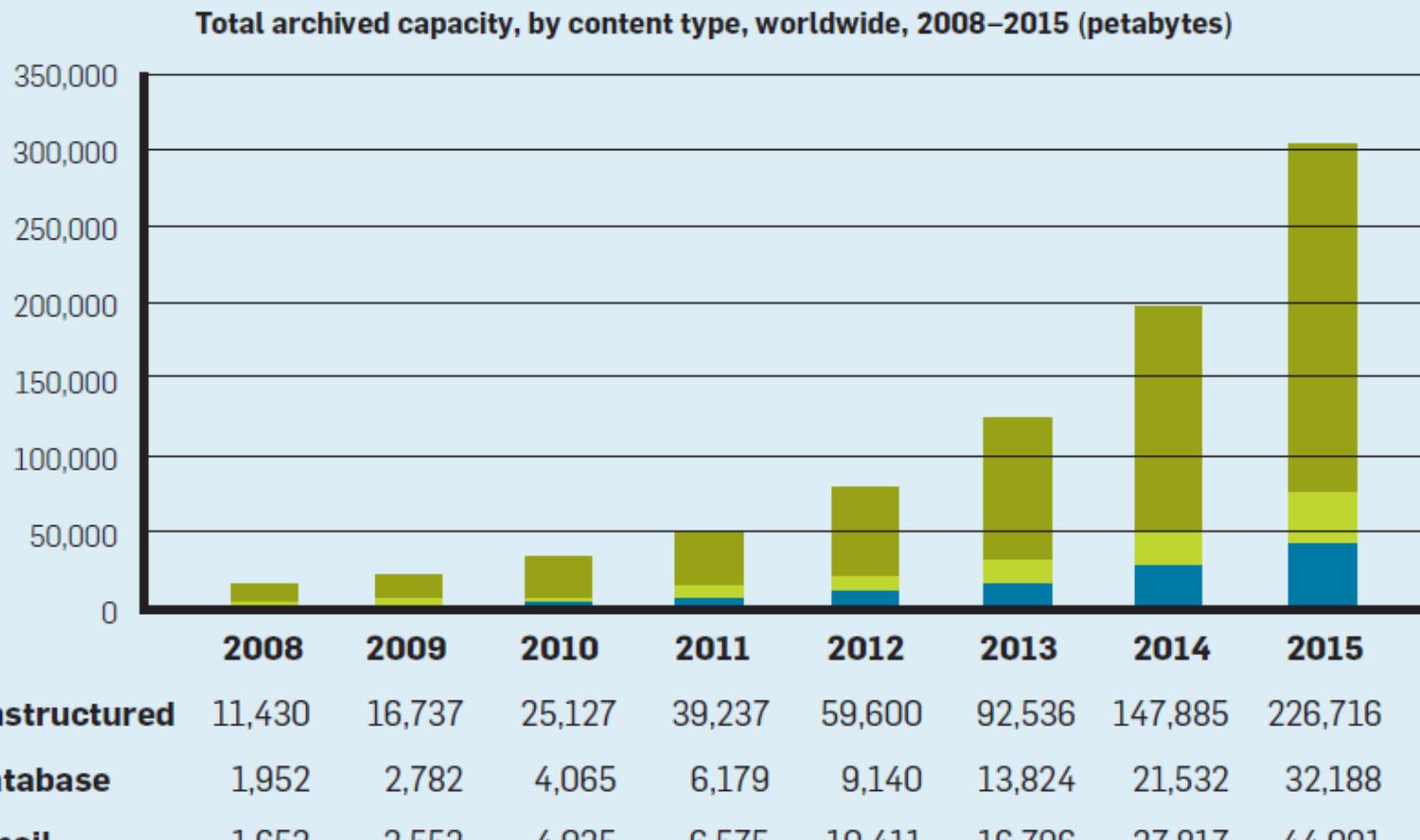
- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Natural Language Processing (NLP)

Natural Language Processing (NLP)

Why is NLP Important?



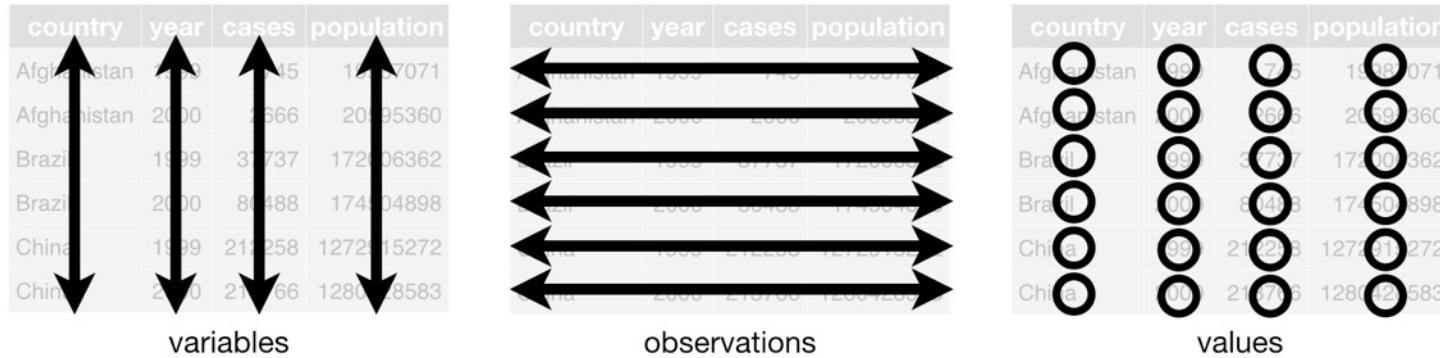
Natural Language Processing (NLP)

Why is NLP Difficult?

- **Text is Messy**
 - Cannot easily be represented in rows and columns of tables
 - Has complex linguistics structures that differ across languages
- **Text is Uncontrolled**
 - Lots of words that are in no dictionary (e.g., spelling mistakes, slang, abbreviations, technical terms)
- **Text is Ambiguous**
 - Meaning of words depends on context

Natural Language Processing (NLP)

Text is Messy



Lorem Ipsum

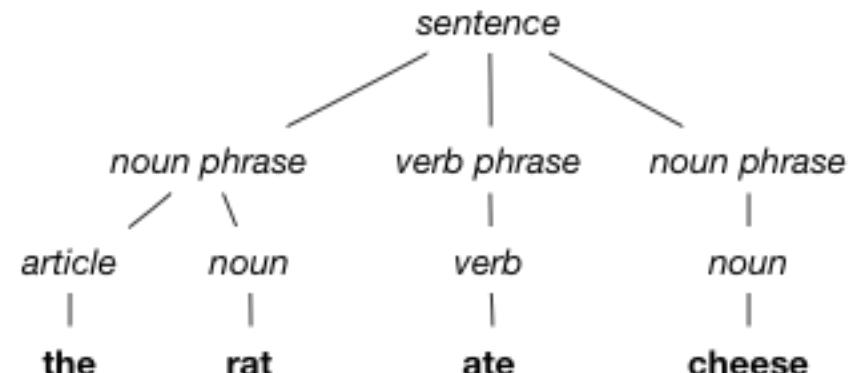
"Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur adipisci velit..."
"There is no one who loves pain itself, who seeks after it and wants to have it, simply because it is pain..."

What is Lorem Ipsum?

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).

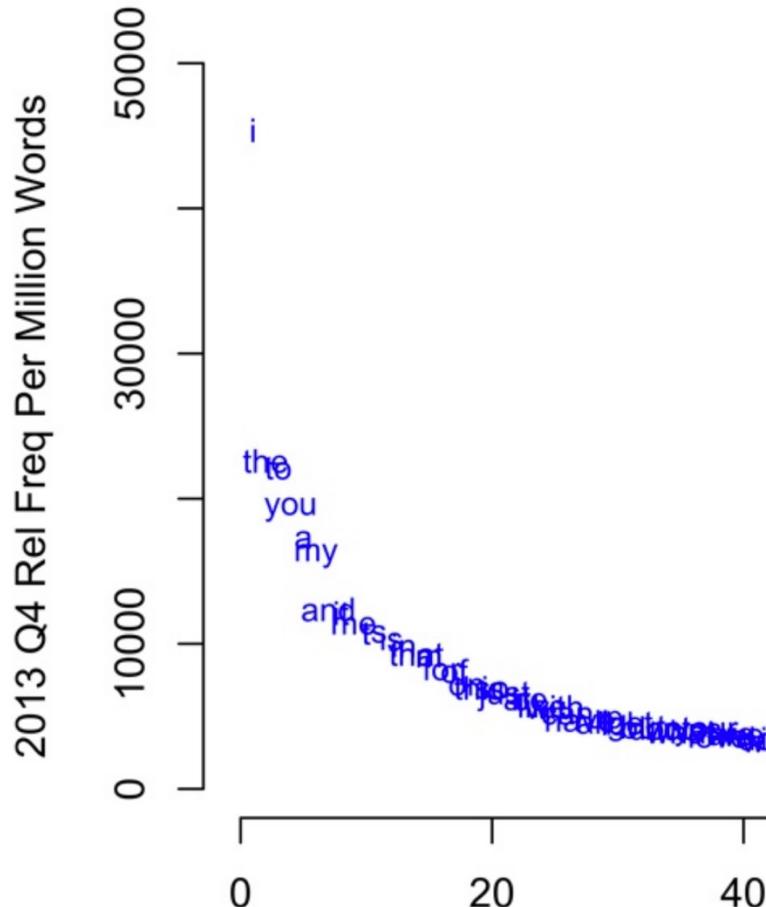
Why do we use it?

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).



Natural Language Processing (NLP)

Text is Uncontrolled

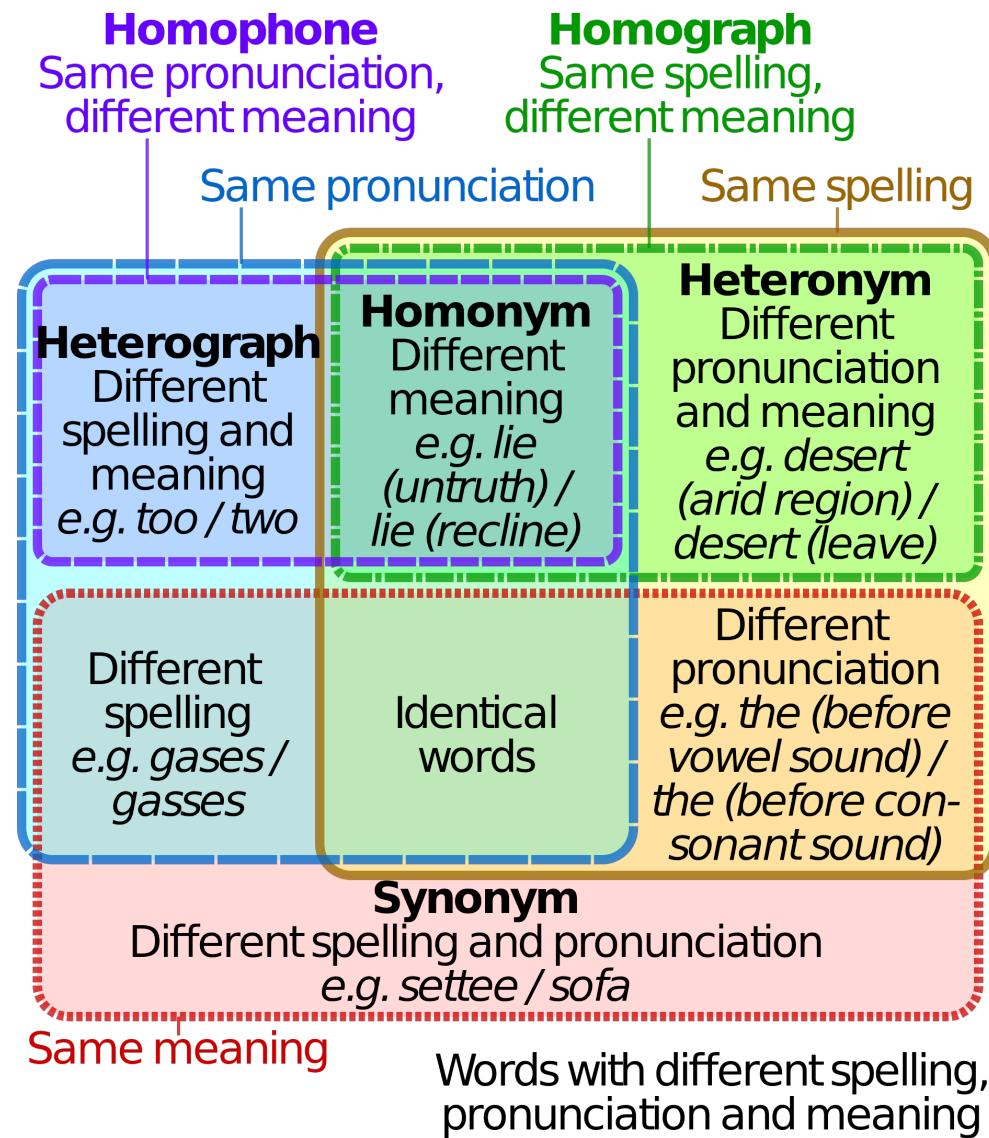


The Cool Parent's Guide to Internet Slang and Abbreviations

AFAIK	As Far as I Know	MMB	Message Me Back
AFK	Away from Keyboard	msg	Message
ASL	Age/Sex/Location?	MYOB	Mind Your Own Business
ATM	At The Moment	N/A	Not Available
b/c	Because	NC	No Comment
b/w	Between	ne1	Anyone
b4	Before	NM	Not much
BBIAB	Be Back in a bit	noob	Newbie
BBL	Be back later	NP	No Problem
BFF	Best Friends Forever	NTN	No Thanks Needed
BRB	Be Right Back	OMG	Oh My Gosh
BTW	By The Way	OMW	On My Way
CTN	Can't Talk Now	OT	Off Topic
CYE	Check Your E-mail	PC	Personal Computer
d1	Download	pls	Please
ETA	Estimated Time of Arrival	POS	Parent Over Shoulder
FWIW	For What It's Worth	ppl	People
FYI	For Your Information	qt	Cutie
GG	Good Game	re	Regarding
GJ	Good Job	SMH	Shaking my head
GL	Good Luck	Sry	Sorry
gr8	Great	TBA	To Be Announced
GTG	Cot To Go	TBC	To Be Continued
GMV	Got My Vote	TC	Take Care
HTH	Hope this helps	thx	Thanks
hw	Homework	TIA	Thanks In Advance
IAC	In Any Case	TLC	Tender Loving Care
IC	I See	TMI	Too Much Information
IDK	I Don't Know	TTFN	Ta-ta For Now
IIRC	If I Remember Correctly	TTYL	Talk To You Later
IKR	I Know, Right?	txt	Text
IM	Instant Message	TY	Thank You
IMO	In My Opinion	w/e	Whatever
IMHO	In My Humble Opinion	w/o	Without
IRL	In Real Life	W8	Wait
J/K	Just kidding	XOXO	Hugs and kisses
K	OK	Y	Why
L8	Late	YNT	Why Not
I8r	Later	YOLO	You Only Live Once
LMK	Let Me Know	YW	You're Welcome
LOL	Laughing Out Loud	ZZZ	Sleeping

Natural Language Processing (NLP)

Text is Ambiguous



Natural Language Processing (NLP)

Categorization



Natural Language Processing (NLP)

Assumptions and Costs of Text Categorization Methods

	Manual coding (bottom up)	Manual coding (top down)	Dictionaries	Supervised machine learning	Unsupervised machine learning
Assumptions					
Categories are predefined	No	Yes	Yes	Yes	No
Relevant text features are known	Yes	Yes	Yes	Yes	Yes
Mapping between text features and categories is known	No	No	Yes	No	No
Costs					
<i>Pre-analysis costs</i>					
Person-hours spent conceptualizing	Low	High	High	High	Low
Level of substantive knowledge	Low	High	High	High	Low
<i>Analysis costs</i>					
Person-hours spent per text	High	High	Low	Low	Low
Level of substantive knowledge	Moderate	Moderate	Low	Low	Low
<i>Post-analysis costs</i>					
Person hours spent interpreting	Moderate	Low	Low	Low	Moderate
Level of substantive knowledge	High	High	High	High	High

Natural Language Processing (NLP)

Assumptions and Costs of Text Categorization Methods

	Manual coding (bottom up)	Manual coding (top down)	Dictionaries	Supervised machine learning	Unsupervised machine learning
Assumptions					
Categories are predefined	No	Yes	Yes	Yes	No
Relevant text features are known	Yes	Yes	Yes	Yes	Yes
Mapping between text features and categories is known	No	No	Yes	No	No
Costs					
<i>Pre-analysis costs</i>					
Person-hours spent conceptualizing	Low	High	High	High	Low
Level of substantive knowledge	Low	High	High	High	Low
<i>Analysis costs</i>					
Person-hours spent per text	High	High	Low	Low	Low
Level of substantive knowledge	Moderate	Moderate	Low	Low	Low
<i>Post-analysis costs</i>					
Person hours spent interpreting	Moderate	Low	Low	Low	Moderate
Level of substantive knowledge	High	High	High	High	High

Natural Language Processing (NLP)

Assumptions and Costs of Text Categorization Methods

	Manual coding (bottom up)	Manual coding (top down)	Dictionaries	Supervised machine learning	Unsupervised machine learning
Assumptions					
Categories are predefined	No	Yes	Yes	Yes	No
Relevant text features are known	Yes	Yes	Yes	Yes	Yes
Mapping between text features and categories is known	No	No	Yes	No	No
Costs					
<i>Pre-analysis costs</i>					
Person-hours spent conceptualizing	Low	High	High	High	Low
Level of substantive knowledge	Low	High	High	High	Low
<i>Analysis costs</i>					
Person-hours spent per text	High	High	Low	Low	Low
Level of substantive knowledge	Moderate	Moderate	Low	Low	Low
<i>Post-analysis costs</i>					
Person hours spent interpreting	Moderate	Low	Low	Low	Moderate
Level of substantive knowledge	High	High	High	High	High

Natural Language Processing (NLP)

Assumptions and Costs of Text Categorization Methods

	Manual coding (bottom up)	Manual coding (top down)	Dictionaries	Supervised machine learning	Unsupervised machine learning
Assumptions					
Categories are predefined	No	Yes	Yes	Yes	No
Relevant text features are known	Yes	Yes	Yes	Yes	Yes
Mapping between text features and categories is known	No	No	Yes	No	No
Costs					
<i>Pre-analysis costs</i>					
Person-hours spent conceptualizing	Low	High	High	High	Low
Level of substantive knowledge	Low	High	High	High	Low
<i>Analysis costs</i>					
Person-hours spent per text	High	High	Low	Low	Low
Level of substantive knowledge	Moderate	Moderate	Low	Low	Low
<i>Post-analysis costs</i>					
Person hours spent interpreting	Moderate	Low	Low	Low	Moderate
Level of substantive knowledge	High	High	High	High	High

Natural Language Processing (NLP)

Assumptions and Costs of Text Categorization Methods

	Manual coding (bottom up)	Manual coding (top down)	Dictionaries	Supervised machine learning	Unsupervised machine learning
Assumptions					
Categories are predefined	No	Yes	Yes	Yes	No
Relevant text features are known	Yes	Yes	Yes	Yes	Yes
Mapping between text features and categories is known	No	No	Yes	No	No
Costs					
<i>Pre-analysis costs</i>					
Person-hours spent conceptualizing	Low	High	High	High	Low
Level of substantive knowledge	Low	High	High	High	Low
<i>Analysis costs</i>					
Person-hours spent per text	High	High	Low	Low	Low
Level of substantive knowledge	Moderate	Moderate	Low	Low	Low
<i>Post-analysis costs</i>					
Person hours spent interpreting	Moderate	Low	Low	Low	Moderate
Level of substantive knowledge	High	High	High	High	High

Natural Language Processing (NLP)

Assumptions and Costs of Text Categorization Methods

	Manual coding (bottom up)	Manual coding (top down)	Dictionaries	Supervised machine learning	Unsupervised machine learning
Assumptions					
Categories are predefined	No	Yes	Yes	Yes	No
Relevant text features are known	Yes	Yes	Yes	Yes	Yes
Mapping between text features and categories is known	No	No	Yes	No	No
Costs					
<i>Pre-analysis costs</i>					
Person-hours spent conceptualizing	Low	High	High	High	Low
Level of substantive knowledge	Low	High	High	High	Low
<i>Analysis costs</i>					
Person-hours spent per text	High	High	Low	Low	Low
Level of substantive knowledge	Moderate	Moderate	Low	Low	Low
<i>Post-analysis costs</i>					
Person hours spent interpreting	Moderate	Low	Low	Low	Moderate
Level of substantive knowledge	High	High	High	High	High

Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

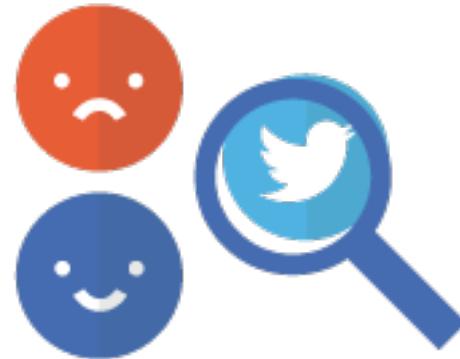
Syllabus

Natural Language Processing (NLP):
_ Dictionary-based Methods

Dictionary-Based Methods

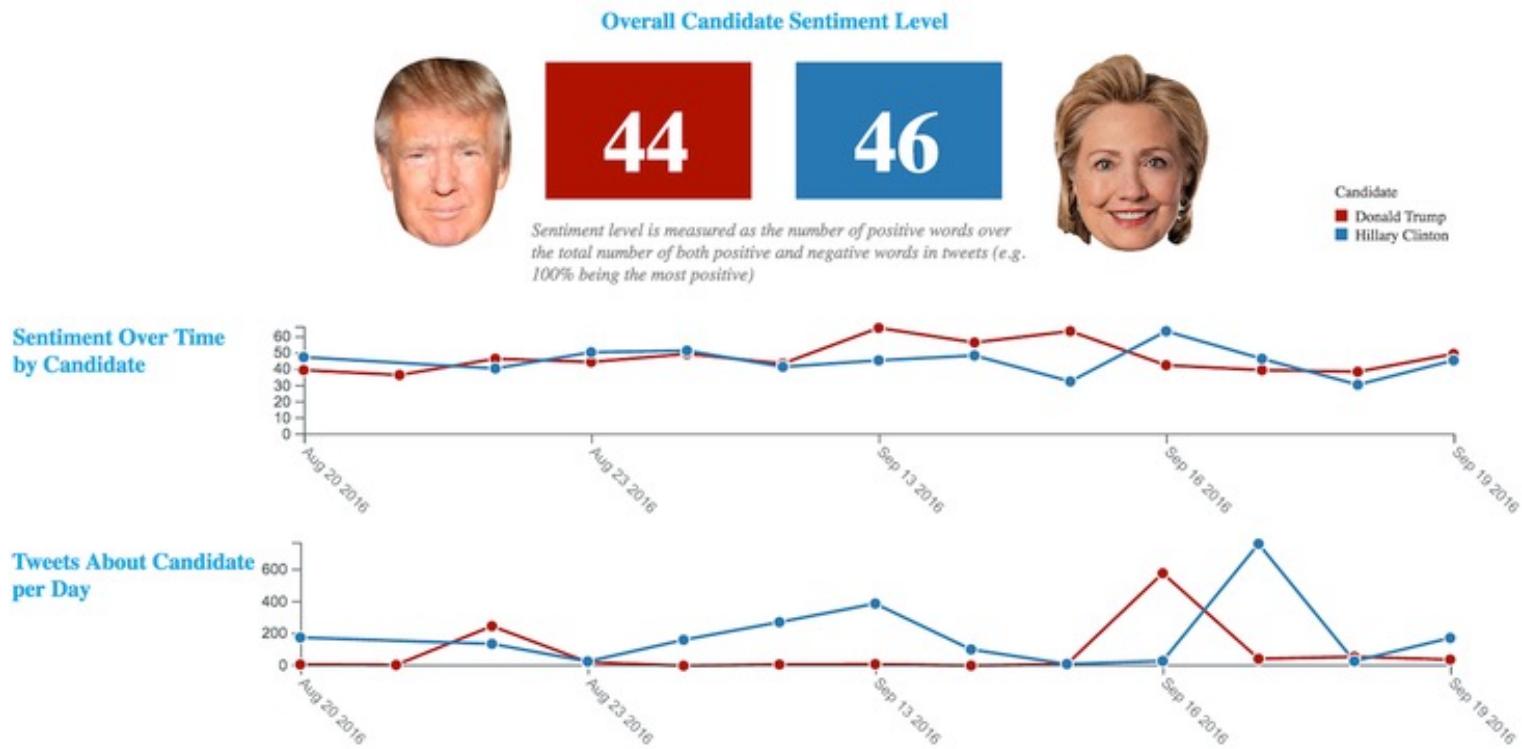
What is Sentiment Analysis?

The computational treatment of opinion, sentiment, and subjectivity in text.



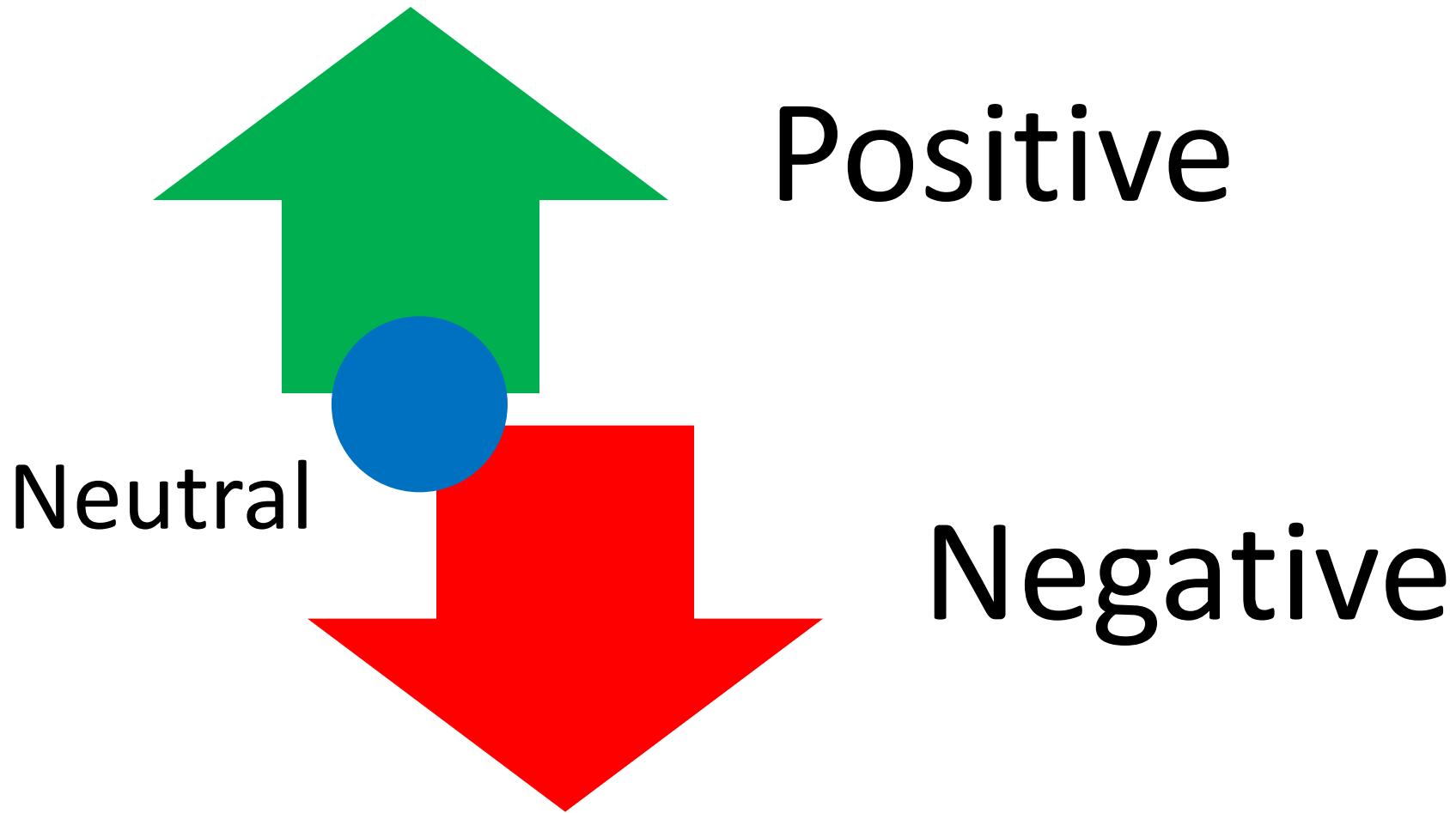
Dictionary-Based Methods

Example



Dictionary-Based Methods

Valence



Dictionary-Based Methods

Ekman's Six Basic Emotions



Fearful



Angry



Sad



Happy



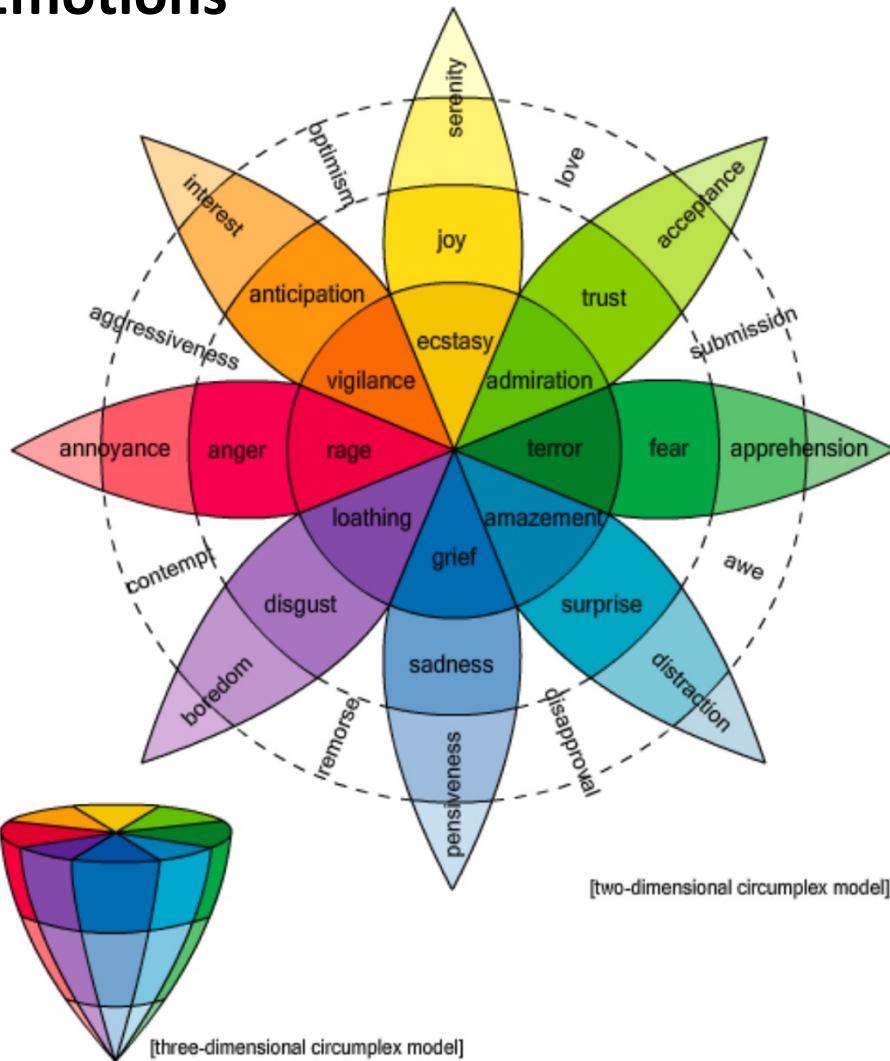
Disgusted



Surprised

Dictionary-Based Methods

Plutchik's Wheel of Emotions



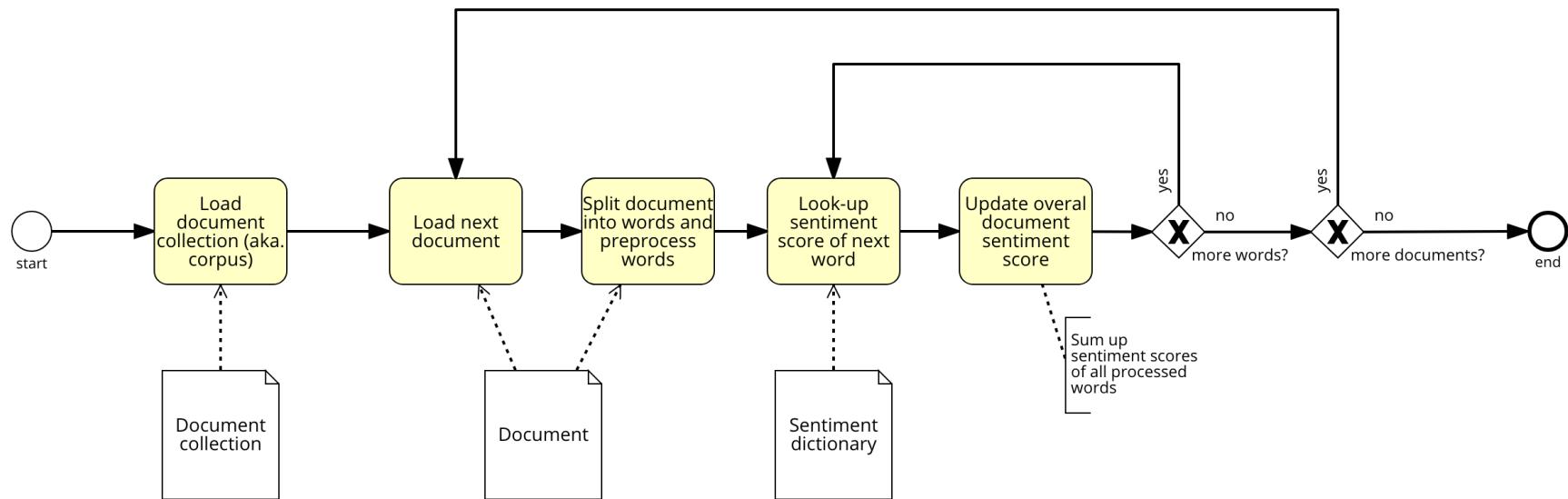
Dictionary-Based Methods

Facebook's Reactions



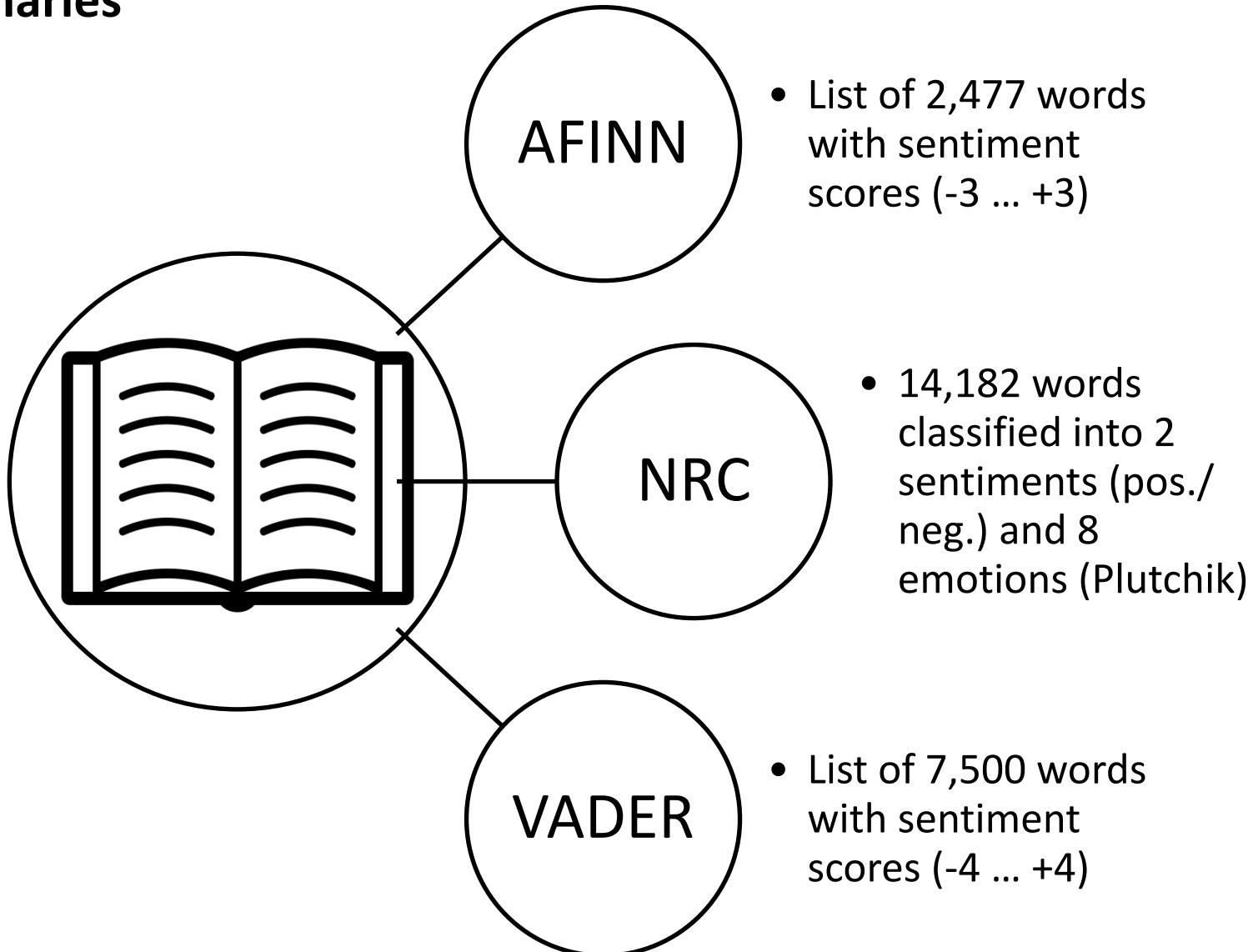
Dictionary-Based Methods

Algorithm



Dictionary-Based Methods

Dictionaries



Dictionary-Based Methods

AFINN

- Finn Årup Nielsen
- <https://github.com/fnielsen/afinn>

1	abandon	-2
2	abandoned	-2
3	abandons	-2
4	abducted	-2
5	abduction	-2
6	abductions	-2
7	abhor -3	
8	abhorred	-3
9	abhorrent	-3
10	abhors -3	
11	abilities	2
12	ability	2
13	aboard 1	
14	absentee	-1
15	absentees	-1
16	absolve	2
17	absolved	2
18	absolves	2
19	absolving	2
20	absorbed	1
21	abuse -3	
22	abused -3	
23	abuses -3	
24	abusive	-3
25	accept 1	
26	accepted	1
27	accepting	1
28	accepts	1
29	accident	-2
30	accidental	-2
31	accidentally	-2
32	accidents	-2
33	accomplish	2
34	accomplished	2
35	accomplishes	2
36	accusation	-2
37	accusations	-2
38	accuse -2	
39	accused	-2
40	accuses	-2
41	accusing	-2

Dictionary-Based Methods

NRC

- Saif M. Mohammad
- <http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm>

Affect Categories: A treemap showing the number of words associated with each category.



Affect Categories Legend

■ anger

Note: 'anticip' is short for anticipation.

Word-Sentiment Associations

disobey	negative
disparage	negative
disparaging	negative
disparity	negative
displaced	
displeased	negative
dispossessed	negative
dispute	negative
disqualified	negative
disreputable	negative
disrespect	negative
disrespectful	negative
disruption	negative

y

joy

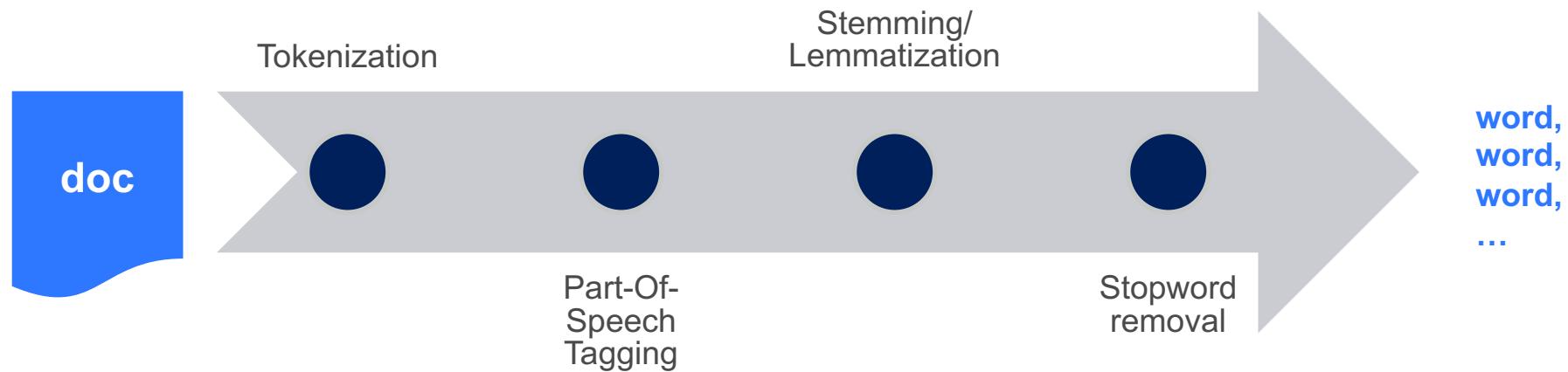
Dictionary-Based Methods

VADER

- C.J. Hutto & Eric Gilbert
- <https://github.com/cjhutto/vaderSentiment>
- Allows for handling of
 - typical **negations** (e.g., "not good")
 - use of **contractions** as negations (e.g., "wasn't very good")
 - conventional use of **punctuation** to signal increased sentiment intensity (e.g., "Good!!!")
 - conventional use of **word-shape** to signal emphasis (e.g., using ALL CAPS for words/phrases)
 - using **degree modifiers** to alter sentiment intensity (e.g., intensity boosters such as "very" and intensity dampeners such as "kind of")
 - understanding many sentiment-laden **slang** words (e.g., 'sux')
 - understanding many sentiment-laden **emoticons** such as :) and :D
 - translating utf-8 encoded **emojis** such as ❤️ and 💋 and 😊
 - understanding sentiment-laden initialisms and **acronyms** (for example: 'lol')

Dictionary-Based Methods

A Typical Preprocessing Pipeline

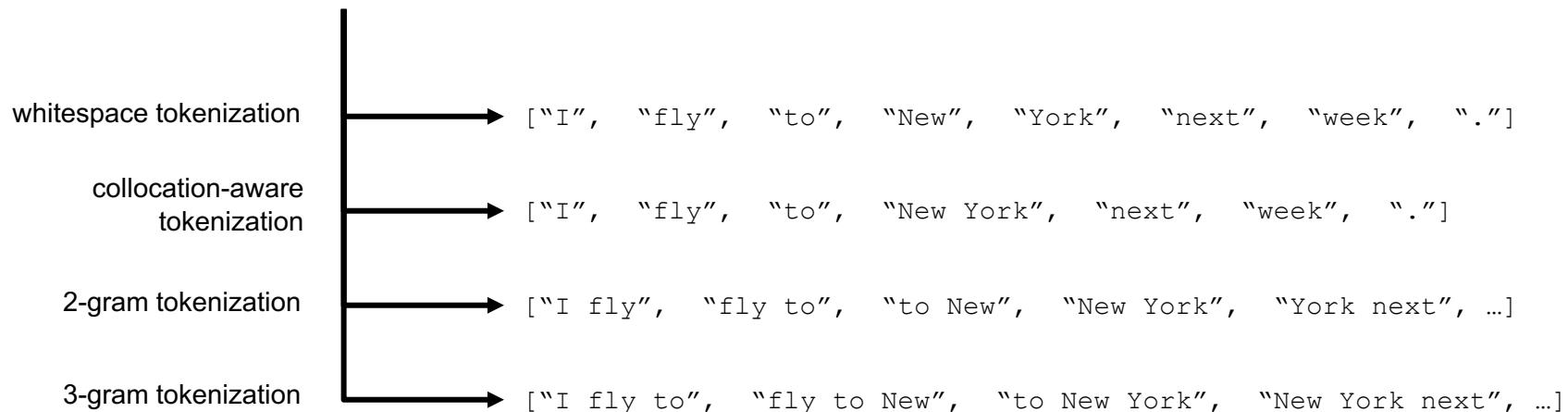


Dictionary-Based Methods

Tokenization

- Tokenization is the process of segmenting texts into sequences of words.

"I fly to New York next week."



Dictionary-Based Methods

Part-Of-Speech Tagging

- Part-Of-Speech (POS) tagging is the process of marking up a word in a text as corresponding to a particular part of speech

Universal Part-of-Speech Tagset

Tag	Meaning	English Examples
ADJ	adjective	<i>new, good, high, special, big, local</i>
ADP	adposition	<i>on, of, at, with, by, into, under</i>
ADV	adverb	<i>really, already, still, early, now</i>
CONJ	conjunction	<i>and, or, but, if, while, although</i>
DET	determiner, article	<i>the, a, some, most, every, no, which</i>
NOUN	noun	<i>year, home, costs, time, Africa</i>
NUM	numeral	<i>twenty-four, fourth, 1991, 14:24</i>
PRT	particle	<i>at, on, out, over per, that, up, with</i>
PRON	pronoun	<i>he, their, her, its, my, I, us</i>
VERB	verb	<i>is, say, told, given, playing, would</i>
.	punctuation marks	<i>., ; !</i>
X	other	<i>ersatz, esprit, dunno, gr8, univeristy</i>

Dictionary-Based Methods

- **Stemming** is the process of reducing inflected or derived words to their word stem.
- **Lemmatization** is the process of reducing inflected or derived words to their dictionary form.

analyze
analyzed
analyzer

analyze
analyzed
analyzer

analyze

Three red arrows point from the words "analyze", "analyzed", and "analyzer" to the stem "analyz".

analyze
analyzed
analyzer

analyze
analyzed
analyzer

analyze
analyzer

Two red arrows point from "analyze" to "analyze". One red arrow points from "analyzed" to "analyze". One red arrow points from "analyzer" to "analyzer".

Dictionary-Based Methods

Stopword Removal

i me my myself we our ours ourselves you your yours yourself yourselves he him his himself she her hers herself it its itself they them their theirs themselves what which who whom this that these those am is are was were be been being have has had having do does did doing a an the and but if or because as until while of at by for with about against between into through during before after above below to from up down in out on off over under again further then once here there when where why how all any both each few more most other some such no nor not only own same so than too very s t can will just don should now

Twitter Airline Sentiment





Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Natural Language Processing (NLP):
_ Bag-of-Words Models

Bag-of-Word Models

Technical Terms

- Corpus
 - A collection of documents
- Document
 - A collection of sentences and words
- Token
 - The occurrence of a word in a document
- Word
 - Unique tokens
- Vocabulary
 - All words appearing in a corpus

Bag-of-Word Models

How to Represent Text?

Lorem Ipsum

"Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit..."
"There is no one who loves pain itself, who seeks after it and wants to have it, simply because it is pain..."

What is Lorem Ipsum?

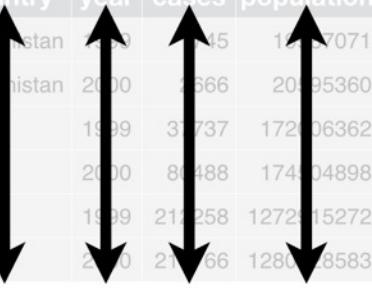
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Why do we use it?

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).

country	year	cases	population
Afghanistan	1993	45	1987071
Afghanistan	2000	2666	2050360
Brazil	1999	37737	17206362
Brazil	2000	80488	17404898
China	1999	212258	127215272
China	2000	21366	128042583

variables



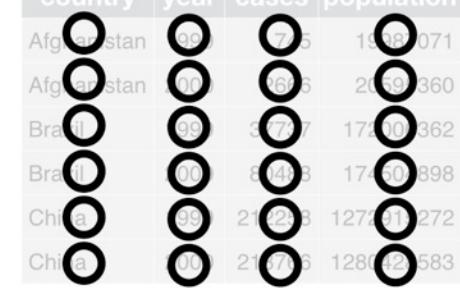
country	year	cases	population
Afghanistan	1993	45	1987071
Afghanistan	2000	2666	2050360
Brazil	1999	37737	17206362
Brazil	2000	80488	17404898
China	1999	212258	127215272
China	2000	21366	128042583

observations



country	year	cases	population
Afghanistan	1993	45	1987071
Afghanistan	2000	2666	2050360
Brazil	1999	37737	17206362
Brazil	2000	80488	17404898
China	1999	212258	127215272
China	2000	21366	128042583

values



Bag-of-Word Models

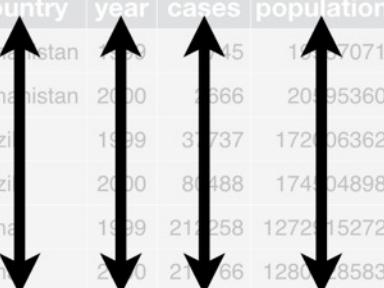


Bag of Words (BOW)

- Treat every document as a unordered set of words
- Ignore word order, sentence structure, and punctuation
- Tidy data frame:
 - Every document is an observation (row)
 - Every word is a variable (column)
 - The presence of a word in a document (aka. token) is represented by the cell values

country	year	cases	population
Afghanistan	1993	145	18087071
Afghanistan	2000	2666	20501360
Brazil	1999	37737	172406362
Brazil	2000	80488	174504898
China	1999	212258	127215272
China	2000	21266	128042583

variables



country	year	cases	population
Afghanistan	1993	145	18087071
Afghanistan	2000	2666	20501360
Brazil	1999	37737	172406362
Brazil	2000	80488	174504898
China	1999	212258	127215272
China	2000	21266	128042583

observations



country	year	cases	population
Afghanistan	1993	145	18087071
Afghanistan	2000	2666	20501360
Brazil	1999	37737	172406362
Brazil	2000	80488	174504898
China	1999	212258	127215272
China	2000	21266	128042583

values



Bag-of-Word Models

Example

- **D1** = "If it walks like a duck and quacks like a duck, it must be a duck."
- **D2** = "Beijing Duck is mostly prized for the thin, crispy duck skin with authentic versions of the dish serving mostly the skin."
- **D3** = "Bugs' ascension to stardom also prompted the Warner animators to recast Daffy Duck as the rabbit's rival, intensely jealous and determined to steal back the spotlight while Bugs remained indifferent to the duck's jealousy, or used it to his advantage. This turned out to be the recipe for the success of the duo."
- **D4** = "6:25 PM 1/7/2007 blog entry: I found this great recipe for Rabbit Braised in Wine on cookingforengineers.com."
- **D5** = "Last week Li has shown you how to make the Sechuan duck. Today we'll be making Chinese dumplings (Jiaozi), a popular dish that I had a chance to try last summer in Beijing. There are many recipies for Jiaozi."

Bag-of-Word Models

Example

Term	Term present?				
	D1	D2	D3	D4	D5
Beijing	0	1	0	0	1
Dish	0	1	0	0	1
Duck	1	1	1	0	1
Rabbit	0	0	1	1	0
Recipe	0	0	1	1	1
...					

↑

IF document contains term: 1
ELSE: 0

Bag-of-Word Models

Distributional Hypothesis of Computational Linguistics

- If **documents** have similar column vectors in a term-document matrix, then they tend to have similar meanings.
- If **terms** have similar row vectors in a term-document matrix, then they tend to have similar meanings.

Term	Term present?				
	D1	D2	D3	D4	D5
Beijing	0	1	0	0	1
Dish	0	1	0	0	1
Duck	1	1	1	0	1
Rabbit	0	0	1	1	0
Recipe	0	0	1	1	1

Bag-of-Word Models

Term Frequency (TF)

Term	Term Frequency				
	D1	D2	D3	D4	D5
Beijing	0	1	0	0	1
Dish	0	1	0	0	1
Duck	3	2	2	0	1
Rabbit	0	0	1	1	0
Recipe	0	0	1	1	1
...					

Inverse Document Frequency (IDF)

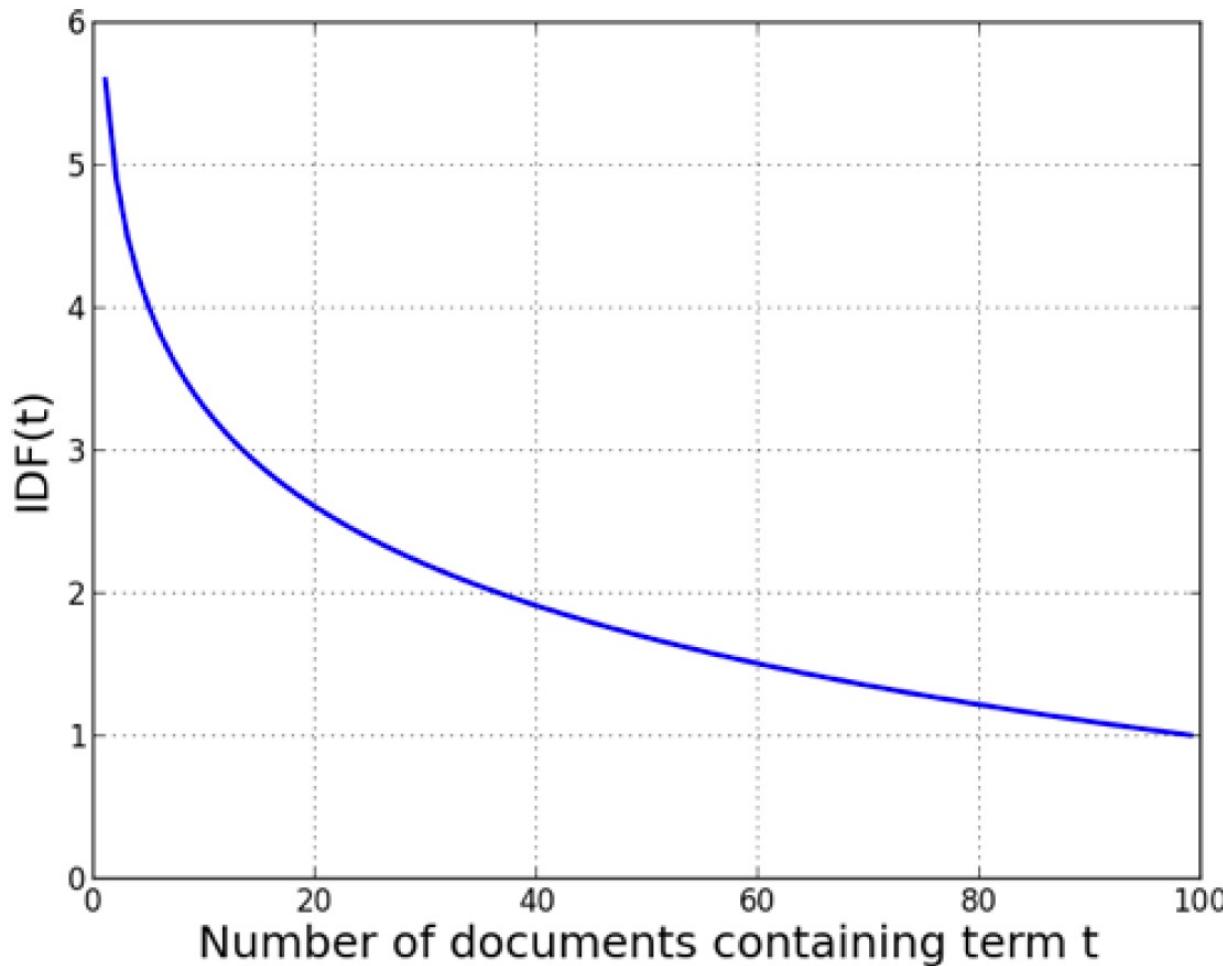
- Term frequency (TF) measures how prevalent a term is in a single document
- For deciding which terms are good features, it is also important to know **how prevalent a term is in the entire corpus**
- **Rare terms**
 - Terms that occur only one or two times in a large corpus (e.g., xylophone) do not carry much information
 - Such terms are not useful, for example, for clustering or classification
 - So, there is typically a lower limit for term frequency
- **Common terms**
 - Terms that virtually occur in every document (e.g., the, and, you) also do not carry much information
 - Such terms are not useful, for example, for document classification
 - So, there is typically an upper limit for term frequency

Inverse Document Frequency (IDF)

$$\text{IDF}(t) = 1 + \log \left(\frac{\text{Total number of documents}}{\text{Number of documents containing } t} \right)$$

Bag-of-Word Models

Example: IDF of a term t within a corpus of 100 documents



Bag-of-Word Models

TF & IDF

Term	Term Frequency				
	D1	D2	D3	D4	D5
Beijing	0	1	0	0	1
Dish	0	1	0	0	1
Duck	3	2	2	0	1
Rabbit	0	0	1	1	0
Recipe	0	0	1	1	1

Term	Inverse Document Frequency
Beijing	1.398
Dish	1.398
Duck	1.097
Rabbit	1.398
Recipe	1.222

Bag-of-Word Models

TF*IDF

Term	Term Frequency * Inverse Document Frequency				
	D1	D2	D3	D4	D5
Beijing	0	1.398	0	0	1.398
Dish	0	1.398	0	0	1.398
Duck	3.291	2.194	2.194	0	1.097
Rabbit	0	0	1.398	1.398	0
Recipe	0	0	1.222	1.222	1.222

Bag-of-Word Models

Classification/Regression with Text

	Bejing	Dish	Duck	Rabbit	Recipe	...	Outcome
D1	0	0	3.291	0	0	...	NO
D2	1.398	1.398	2.194	0	0	...	YES
D3	0	0	2.194	1.398	1.222	...	NO
D4	0	0	0	1.398	1.222	...	YES
D5	1.398	1.398	1.097	0	1.222	...	YES
...

For example, for a logistic regression: $\Pr(Y = 1|X) = \frac{e^{\beta_0 + \beta_1 X}}{1 + e^{\beta_0 + \beta_1 X}}$

Twitter Airline Sentiment, revisited





Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

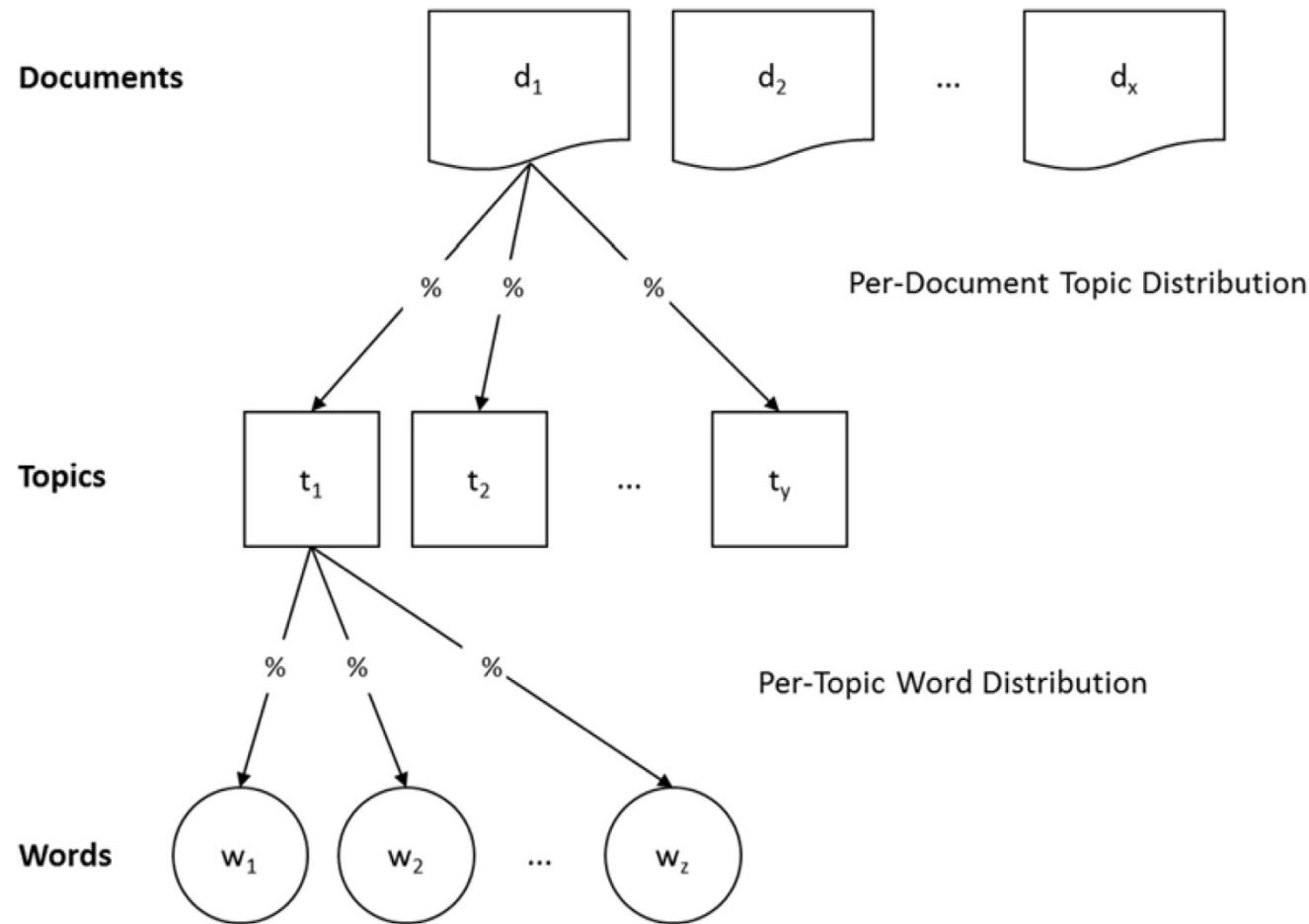
Natural Language Processing (NLP):
_ Topic Models

What are Topic Models?

- Unsupervised machine learning methods for text mining (e.g., Latent Semantic Analysis, Latent Dirichlet Allocation)
- Theoretical grounding: **Distributional hypothesis** of linguistics
 - Words that co-occur together in similar contexts (e.g., ball, goal, offside) tend to have similar meanings
 - Co-occurrence patterns can be interpreted as topics (e.g., football) and used to cluster documents

Topic Models

Schematic Overview of Probabilistic Topic Modeling with LDA



Topic Models

Illustrative Example of Probabilistic Topic Modeling with LDA



Exemplary Customer Review about a Fitbit Flex

I bought this for my 14 year old daughter as a gift. She received it in July. It works great - she lost 6 pounds in 2 weeks. The Fitbit makes staying in shape easy. The iPhone app works fine.

Topics

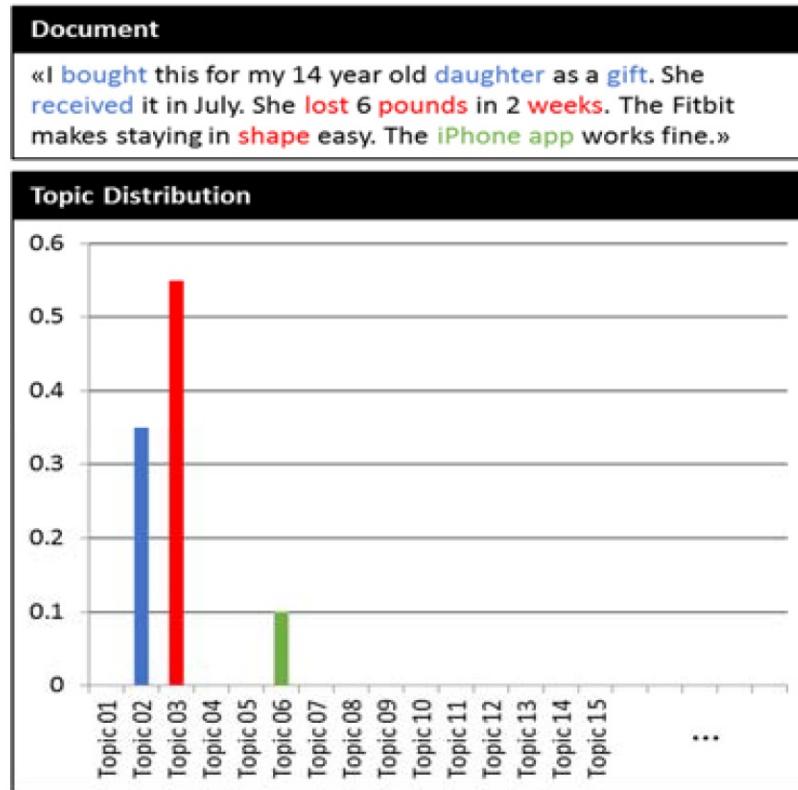
Birthday present

Loosing weight

Mobile app

Topic Models

Illustrative Example of Probabilistic Topic Modeling with LDA



Topic 01	Topic 02	Topic 03	
love	0,13	gift	0,10
recommend	0,08	love	0,07
thing	0,07	christmas	0,07
color	0,07	bought	0,06
purchased	0,06	husband	0,05
band	0,06	daughter	0,04
buy	0,03	received	0,03
mine	0,03	birthday	0,03
amazing	0,03	present	0,02
friend	0,02	son	0,02
		weight	0,08
		loss	0,05
		pounds	0,04
		lose	0,04
		week	0,02
		lb	0,02
		month	0,02
		helped	0,01
		eat	0,01
		goal	0,01

Topic 04	Topic 05	Topic 06	
battery	0,15	heart	0,11
day	0,09	rate	0,10
charge	0,07	monitor	0,07
life	0,05	blood	0,02
week	0,03	pressure	0,02
time	0,02	pedometer	0,02
hour	0,02	measure	0,02
low	0,01	tracking	0,01
recharge	0,01	device	0,01
dead	0,01	glucose	0,01
		app	0,12
		iphone	0,08
		sync	0,03
		ipad	0,02
		work	0,01
		io	0,01
		apple	0,01
		android	0,01
		computer	0,01
		update	0,01

Topic Models



Topic Modeling Walkthrough



tidyverse, quanteda, stm

Topic Models



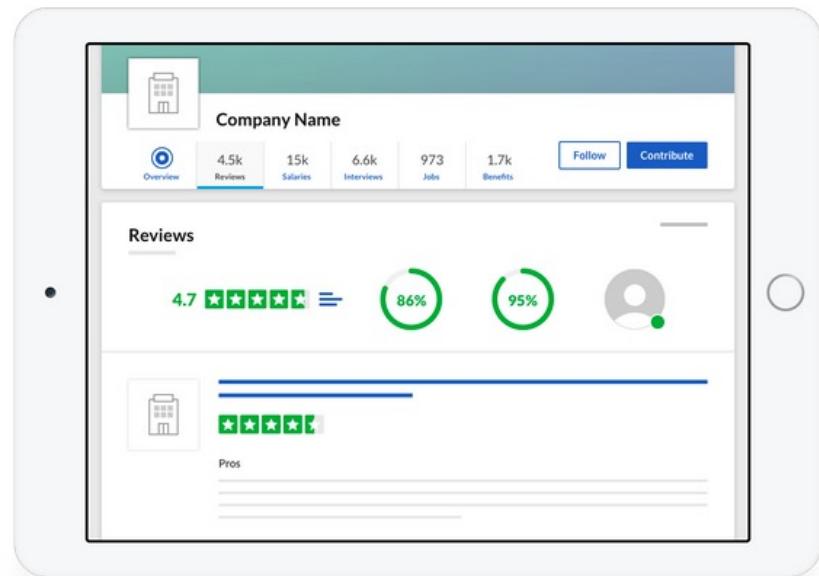
Research Question

“What factors drive employees’ company ratings?”

Company reviews and ratings. Get the whole story.

Search ratings and reviews of over 600,000 companies worldwide. Get the inside scoop and find out what it's really like from people who've actually worked there.

[Write a Review](#)



Topic Models

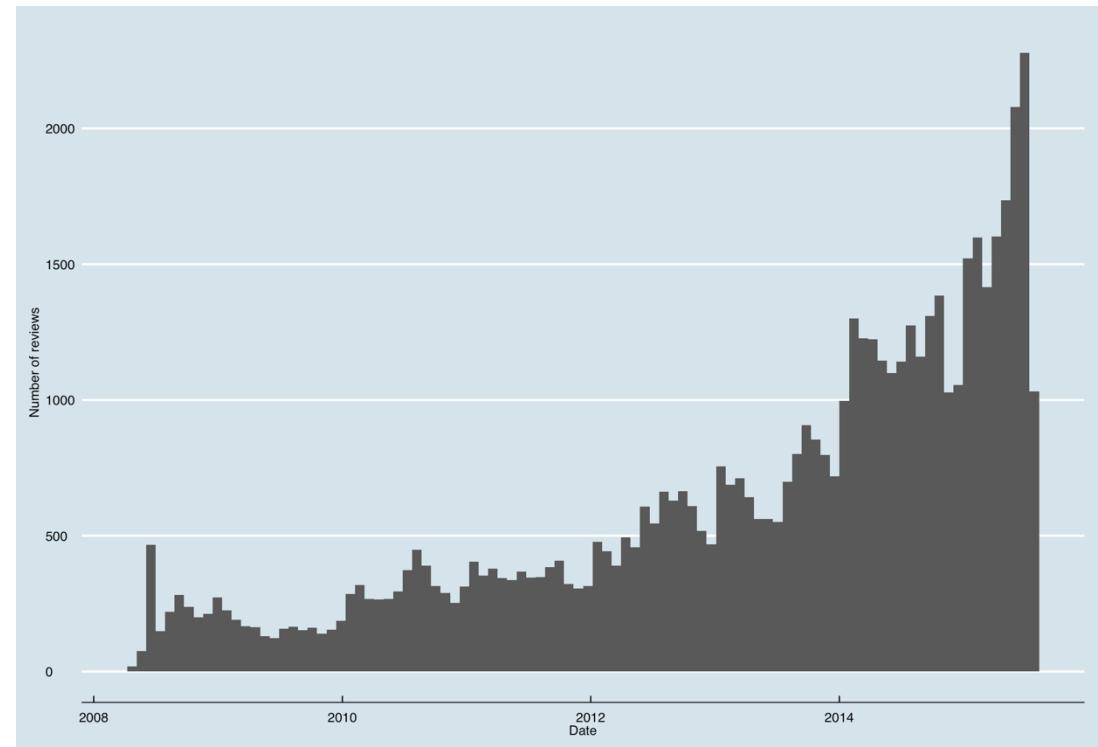
Data Understanding

```
{  
  "shortName": "Bank of America",  
  "date": "2010-10-03",  
  "stars": 3,  
  "text": "Great benefits for associates, Paid maternity/paternity leave, most associates receive 3  
  weeks of vacation leave per year (SSS, PB, AM and four weeks for BCM). Micro-management,  
  poor leadership, lack of recognition, extremely under staffed. Do not forget the human aspect.  
  Micro-management is not the answer to every situation. Put more people in the branches."  
}
```

Topic Models

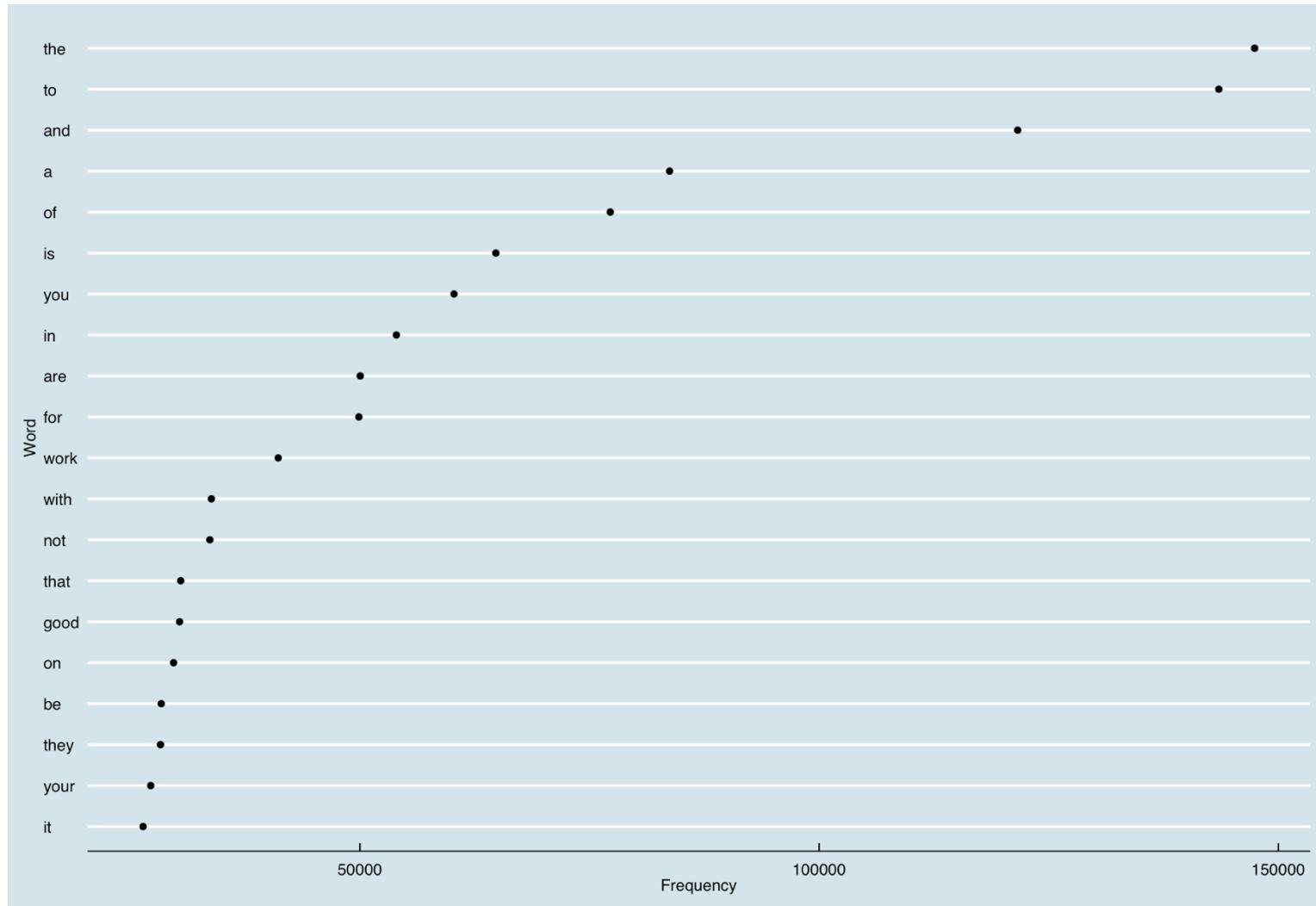
Data Understanding

- Sub-sample
 - Finance industry only
- Number of documents
 - 57,765
- Number of words (tokens)
 - 1,608,259
- Number of unique words
 - 1,740



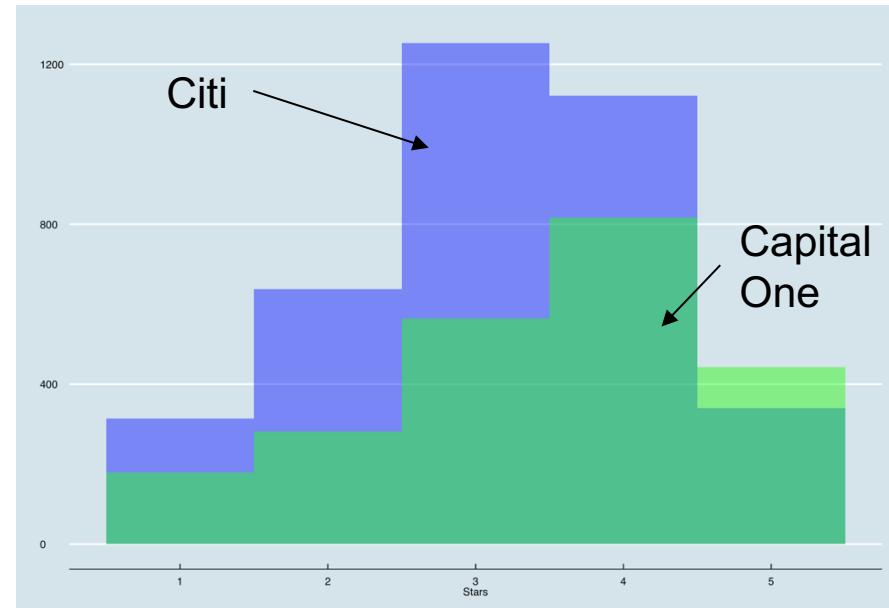
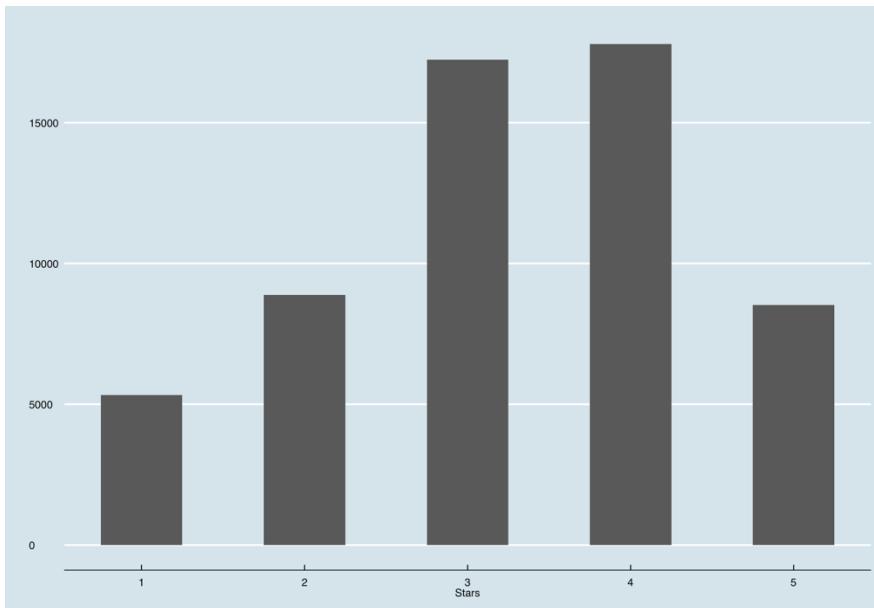
Topic Models

Data Understanding



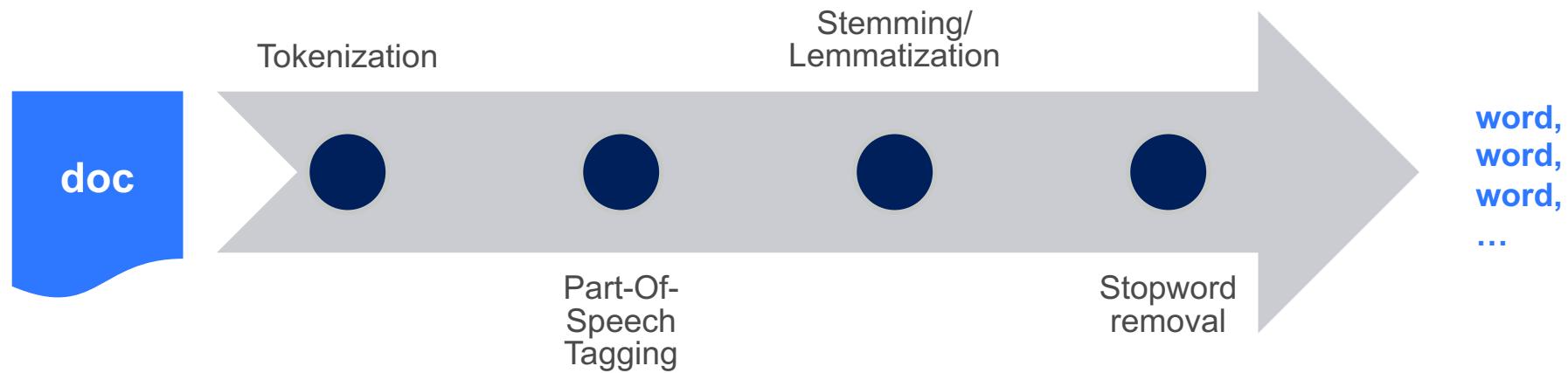
Topic Models

Data Understanding



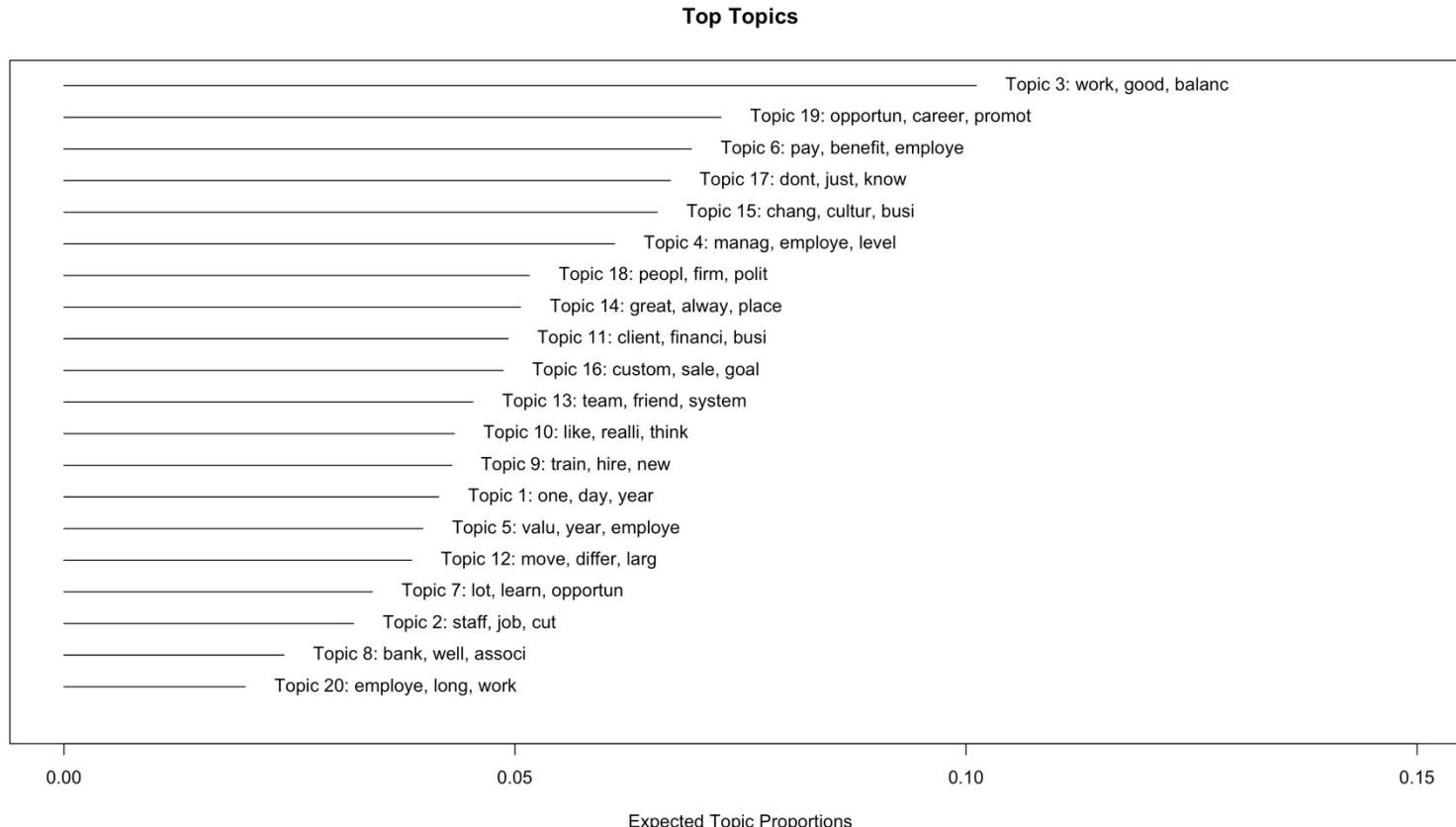
Topic Models

Data Preparation



Modeling and Evaluation – Iteration #1

- Estimate 30 most prevalent topics
 - Takes approx. 10 minutes on a MacBook Pro
 - Wait for the “aha” effect



Modeling and Evaluation – Iteration #1

- Estimate 30 most prevalent topics
 - Takes approx. 10 minutes on a MacBook Pro
 - Wait for the “aha” effect

Top-3 Documents for Topic 6

Health insurance, Vacation, sick pay, paid maternity leave 12 weeks. Every year perks decrease and are eliminated. Uneducated people with their nose in the air.

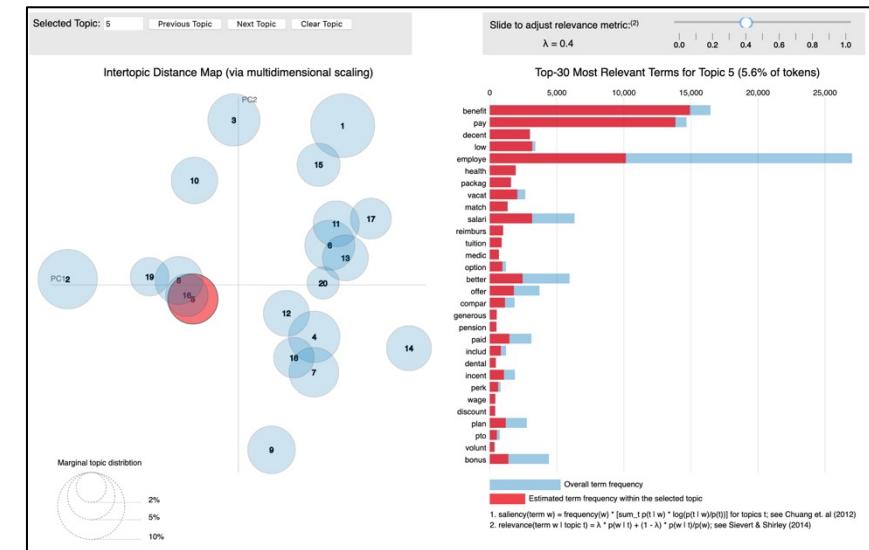
Decent benefits, decent bonuses, decent vacation/off time. Inadequate pay and inadequate coverage.

Huge Annual Bonus amount, Paid Overtime amount can be earned, 1 Time free meal & free transport facility. Less On paper CTC offered. Should include the approximate Annual Bonus & Gratuity in the offered CTC on-paper.

Modeling and Evaluation – Iteration #2

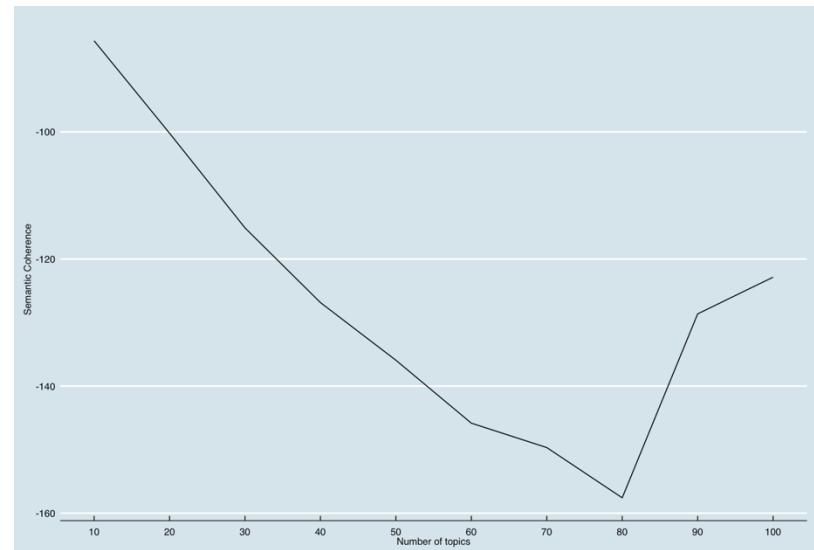
- Find the right number of topics
 - Manual investigation
 - Are topics coherent? No duplicate topics? No fused topics?
 - Increase or reduce number of topics

LDAvis: A Python/R package for interactive topic model visualization.



Modeling and Evaluation – Iteration #2

- Find the right number of topics
 - **Automated** search
 - e.g.: From 10 to 100 topics, in steps of 10
 - Takes several hours on a MacBook Pro
 - Evaluate models with regards to **Semantic Coherence**



Modeling and Evaluation – Iteration #3

- Final **experimental evaluation** through human coders
 - **Word** intrusion task:
 - Which of the following words does not belong to the list?

life, time, balance, bird, work

car, desk, fish, coffee, tree

Modeling and Evaluation – Iteration #3

- Final **experimental evaluation** through human coders



Modeling and Evaluation – Iteration #3

- Final **experimental evaluation** through human coders
 - **Best topic** task:
 - Which of the following lists of keywords describes the text best?

Huge Annual Bonus amount, Paid Overtime amount can be earned, 1 Time free meal & free transport facility. Less On paper CTC offered. Should include the approximate Annual Bonus & Gratuity in the offered CTC on-paper.

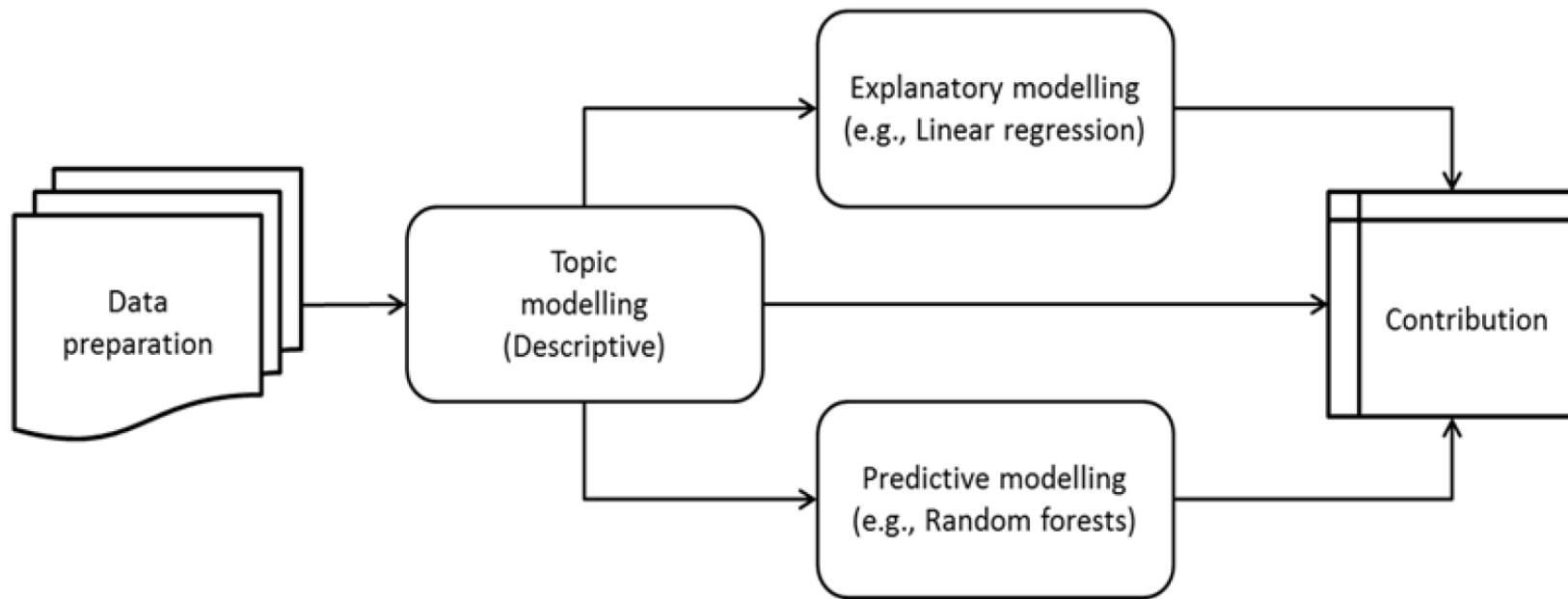
life, time, balance, work, spare

money, bonus, pay, benefit, perks

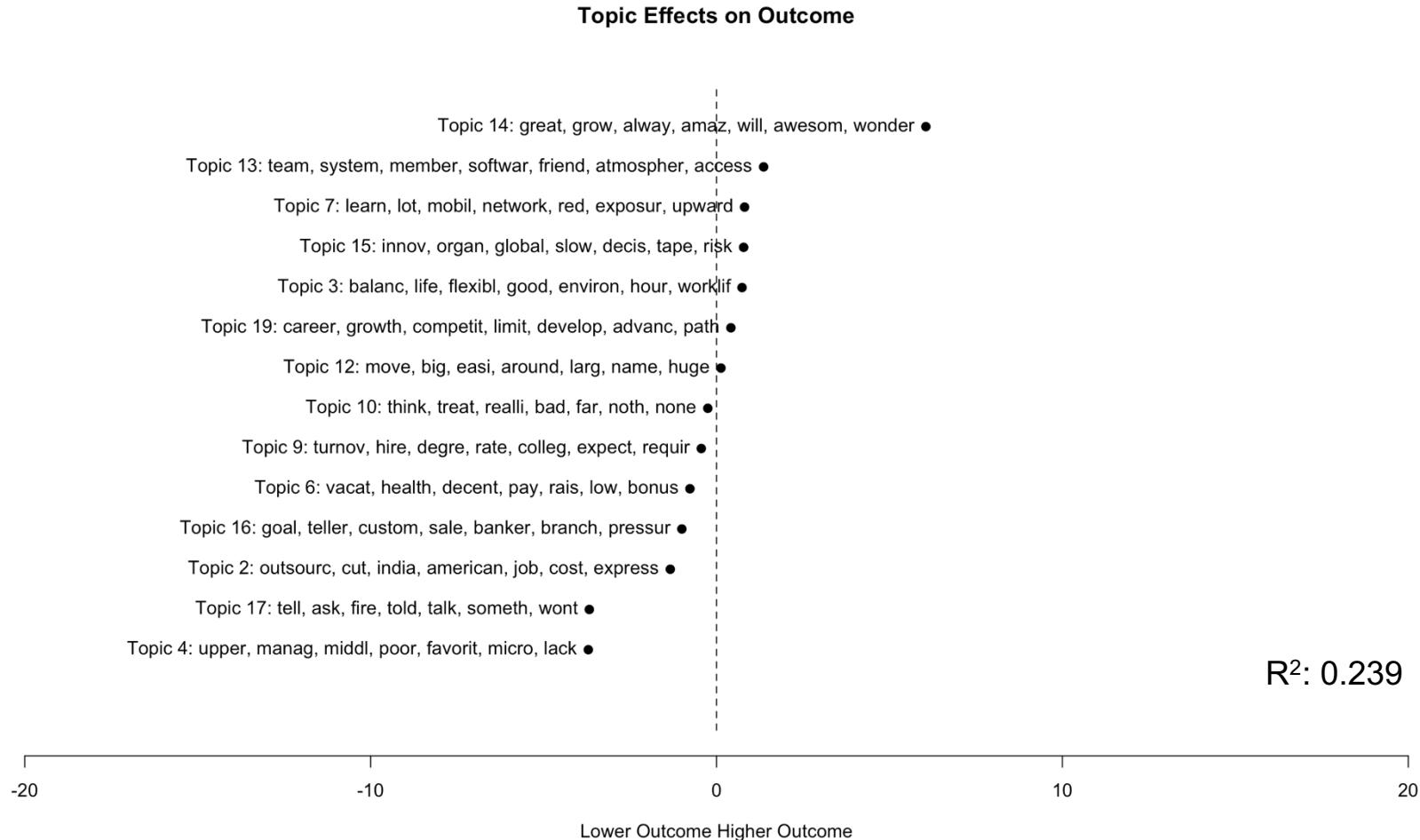
training, development, teach, learn

Modeling and Evaluation – Iteration #4

- Modeling the relationship between topics and stars



Modeling and Evaluation – Iteration #4



Topic Models

Communication

 Communications of the
Association for Information Systems

Tutorial ISSN: 1529-3181

Text Mining For Information Systems Researchers: An Annotated Topic Modeling Tutorial

Stefan Debortoli
University of Liechtenstein
Institute of Information Systems
Vaduz, Liechtenstein
stefan.debortoli@uni.li

Oliver Müller
IT University of Copenhagen
Information Management Section
Copenhagen, Denmark

Iris Junglas
Florida State University
College of Business
Tallahassee, FL, USA

Jan vom Brocke
University of Liechtenstein
Institute of Information Systems
Vaduz, Liechtenstein

Abstract:

Analysts have estimated that more than 80 percent of today's data is stored in unstructured form (e.g., text, audio, image, video)—much of it expressed in rich and ambiguous natural language. Traditionally, to analyze natural language, one has used qualitative data-analysis approaches, such as manual coding. Yet, the size of text data sets obtained from the Internet makes manual analysis virtually impossible. In this tutorial, we discuss the challenges encountered when applying automated text-mining techniques in information systems research. In particular, we showcase how to use probabilistic topic modeling via Latent Dirichlet allocation, an unsupervised text-mining technique, with a LASSO multinomial logistic regression to explain user satisfaction with an IT artifact by automatically analyzing more than 12,000 online customer reviews. For fellow information systems researchers, this tutorial provides guidance for conducting text-mining studies on their own and for evaluating the quality of others.

Keywords: Text Mining, Topic Modeling, Latent Dirichlet Allocation, Online Customer Reviews, User Satisfaction.

This manuscript underwent editorial review. It was received 05/26/2015 and was with the authors for 8 months for 3 revisions. The Associate Editor chose to remain anonymous.

Volume 39 Paper 7 pp. 110 – 135 July 2016



Article

Organizational Research Methods
1-28
© The Author(s) 2018
Reprints and permission:
sagepub.com/journalsPermissions.nav
DOI: 10.1177/109426811873858
jorm.sagepub.com/home/term


Topic Modeling as a Strategy of Inquiry in Organizational Research: A Tutorial With an Application Example on Organizational Culture

Theresa Schmiedel¹, Oliver Müller², and Jan vom Brocke¹

Abstract

Research has emphasized the limitations of qualitative and quantitative approaches to studying organizational phenomena. For example, in-depth interviews are resource-intensive, while questionnaires with closed-ended questions can only measure predefined constructs. With the recent availability of large textual data sets and increased computational power, text mining has become an attractive method that has the potential to mitigate some of these limitations. Thus, we suggest applying topic modeling, a specific text mining technique, as a new and complementary strategy of inquiry to study organizational phenomena. In particular, we outline the potentials of structural topic modeling for organizational research and provide a step-by-step tutorial on how to apply it. Our application example builds on 428,492 reviews of Fortune 500 companies from the online platform Glassdoor, on which employees can evaluate organizations. We demonstrate how structural topic models allow to inductively identify topics that matter to employees and quantify their relationship with employees' perception of organizational culture. We discuss the advantages and limitations of topic modeling as a research method and outline how future research can apply the technique to study organizational phenomena.

Keywords
topic modeling, structural topic model, tutorial, organizational culture, online reviews

Introduction

Organizational research follows both quantitative and qualitative research paradigms and applies various empirical methods for data collection and analysis (Currall, Hammer, Baggett, & Doniger,

¹University of Liechtenstein, Liechtenstein
²IT University of Copenhagen, Copenhagen, Denmark

Corresponding Author:
Theresa Schmiedel, University of Liechtenstein, Fuerst-Franz-Josef-Strasse, 9490 Vaduz, Principality of Liechtenstein.
Email: theresa.schmiedel@uni.li

Twitter Airline Sentiment, revisited, again





Syllabus

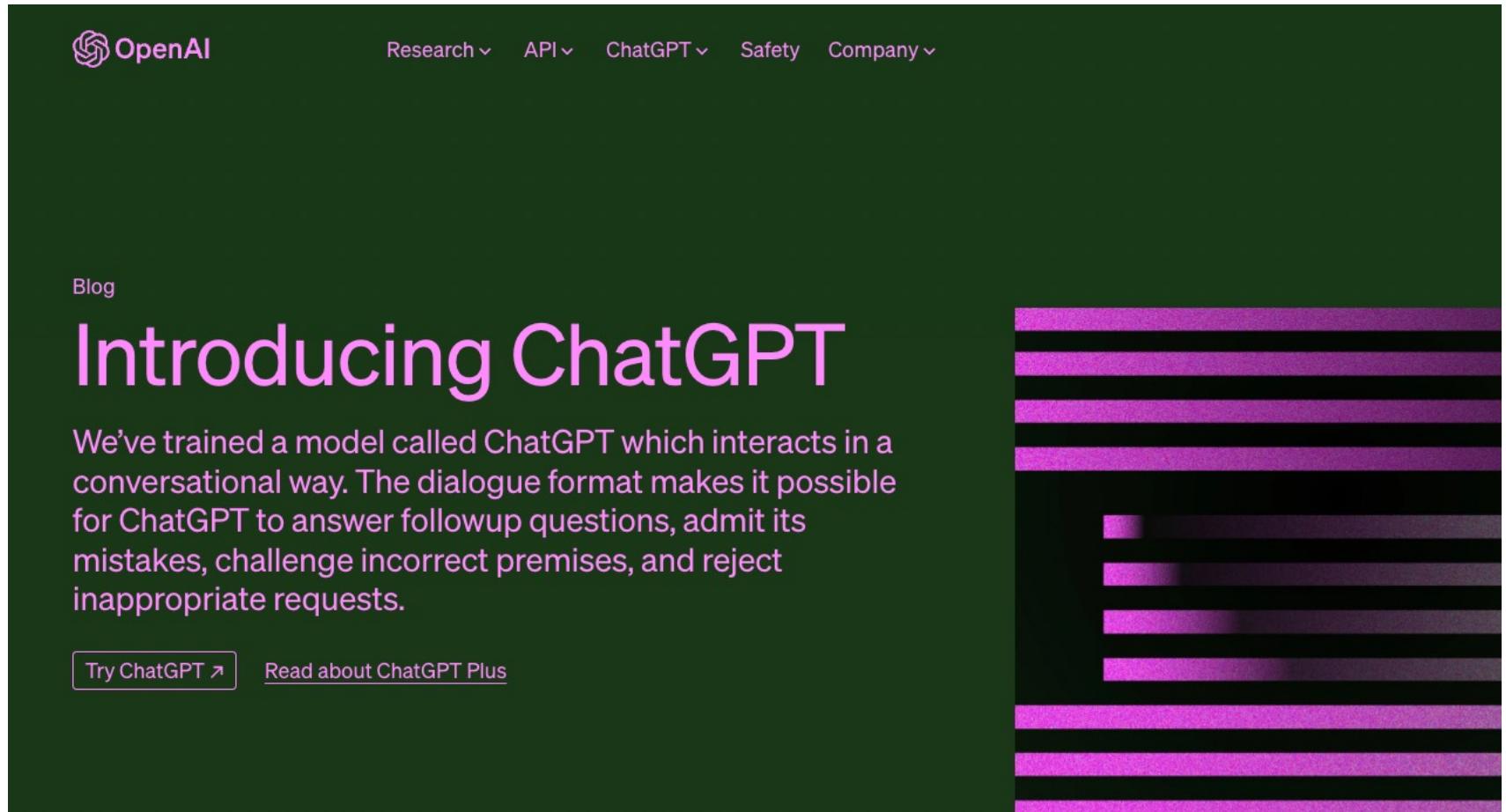
Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Natural Language Processing (NLP):
_ Transformers

Generative Pre-trained Transformer (GPT)



The screenshot shows the official OpenAI website. At the top, there's a dark header with the OpenAI logo and navigation links for Research, API, ChatGPT, Safety, and Company. Below the header, a large banner features the text "Introducing ChatGPT" in a large, white, sans-serif font. A paragraph below it explains that ChatGPT is a conversational AI model capable of interacting in a dialogue format, answering followup questions, admitting mistakes, challenging premises, and rejecting inappropriate requests. Two buttons are visible at the bottom left: "Try ChatGPT" and "Read about ChatGPT Plus". To the right of the main content area, there's a decorative graphic consisting of several horizontal bars of varying shades of purple and black.

OpenAI

Research ▾ API ▾ ChatGPT ▾ Safety Company ▾

Blog

Introducing ChatGPT

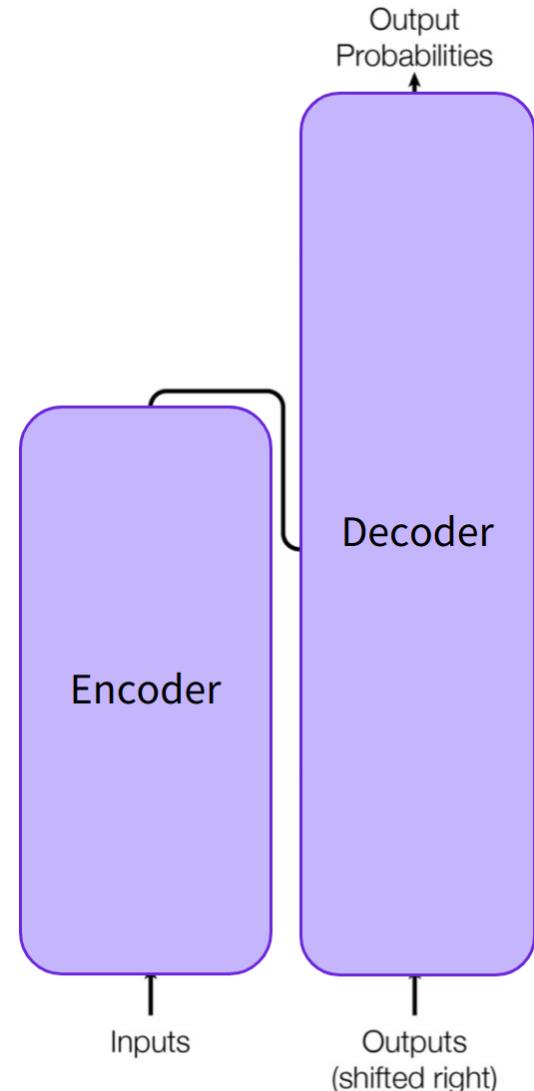
We've trained a model called ChatGPT which interacts in a conversational way. The dialogue format makes it possible for ChatGPT to answer followup questions, admit its mistakes, challenge incorrect premises, and reject inappropriate requests.

[Try ChatGPT ↗](#) [Read about ChatGPT Plus](#)

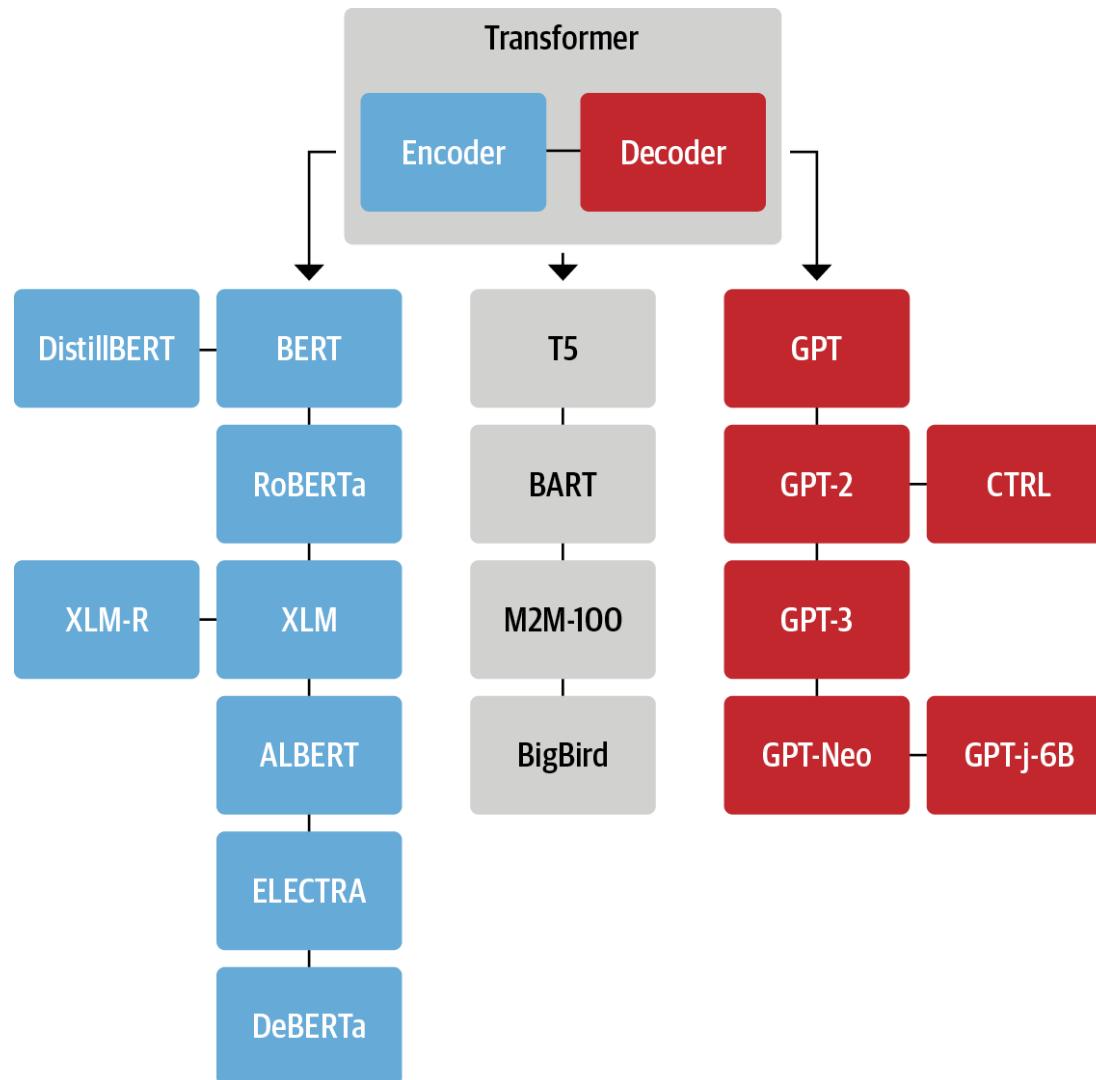
Transformers

What are Transformers?

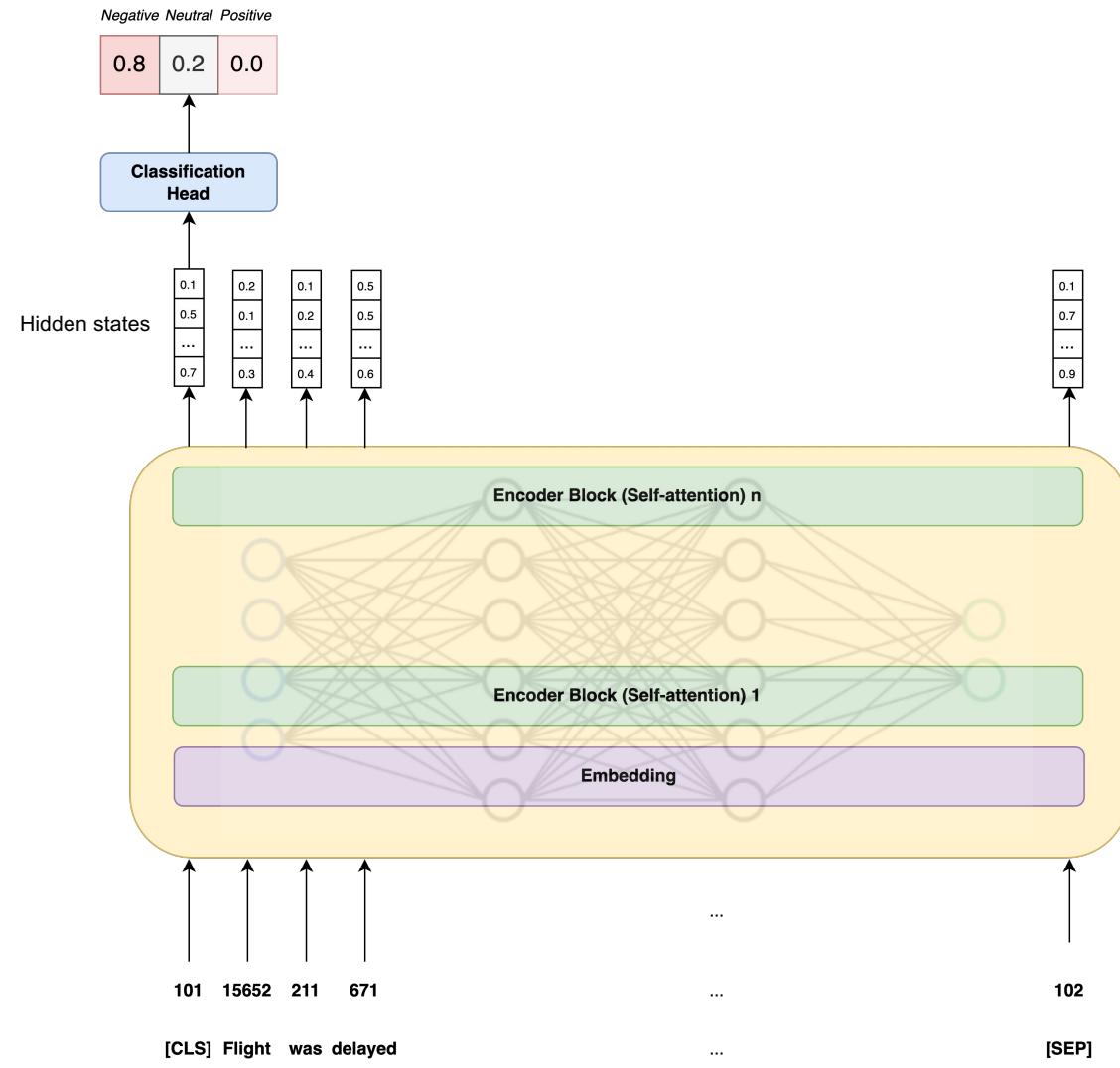
- The general **Transformer architecture** consists of two components:
 - **Encoder:** The encoder processes an input sequence and builds a numerical representation (feature vector) of it. This component focuses on understanding the input.
 - **Decoder:** The decoder processes the feature vector produced by the encoder, plus other inputs, to generate an output sequence. This component focuses on generating outputs.



Transformer Models



Bidirectional Encoder Representations from Transformers (BERT)



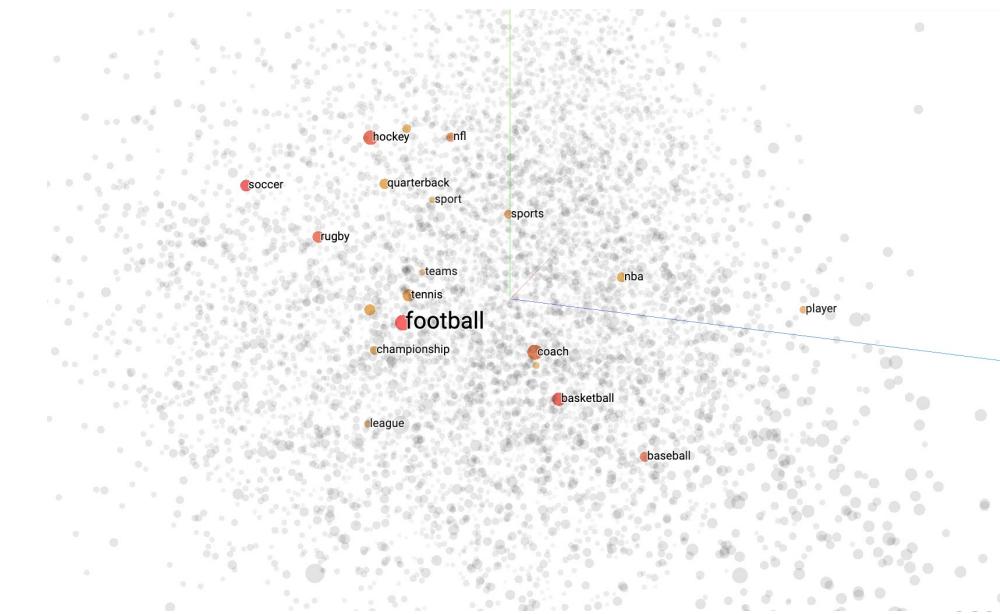
Transformers

Word Embeddings

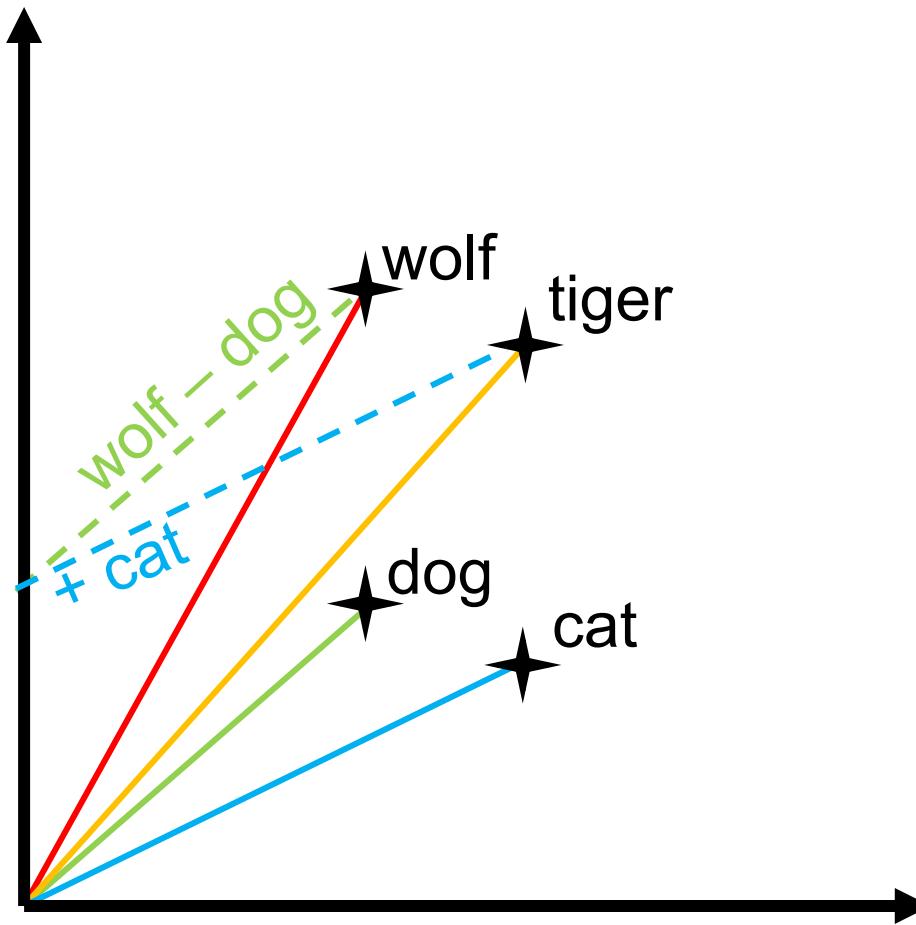
- Words are represented by n-dimensional vectors
- Vectors are constructed so that semantically similar words get similar vectors
- E.g.: “football” is similar to “tennis” and “rugby”

$$f_{\text{word2vec}} : V \rightarrow \mathbb{R}^d$$

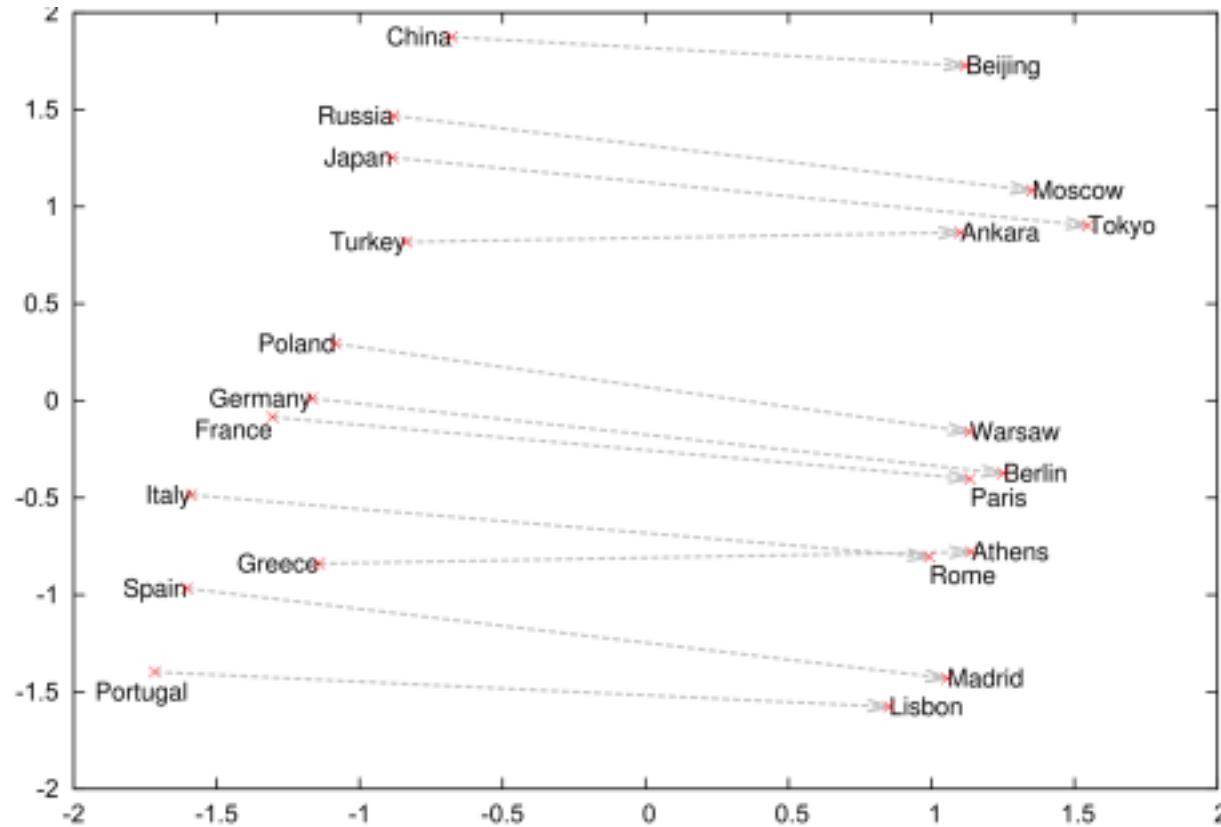
$$v_{\text{play}} = \begin{pmatrix} -0.224 \\ 0.130 \\ \dots \\ 0.276 \end{pmatrix}$$



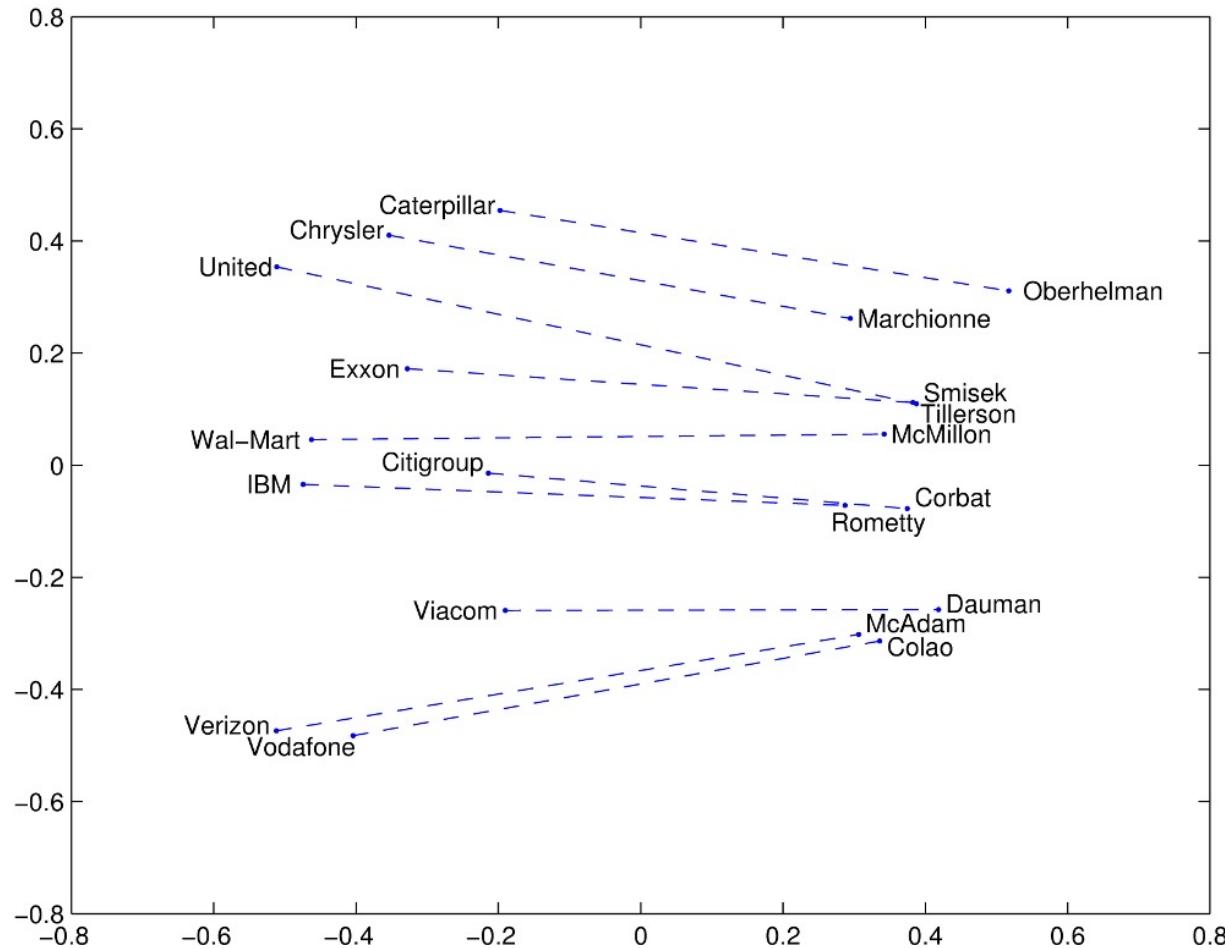
Word Embeddings



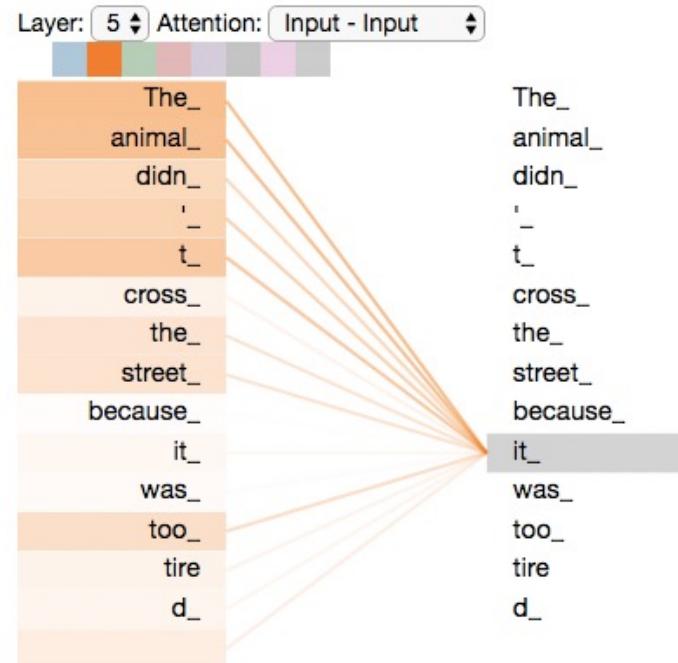
Word Embeddings



Word Embeddings

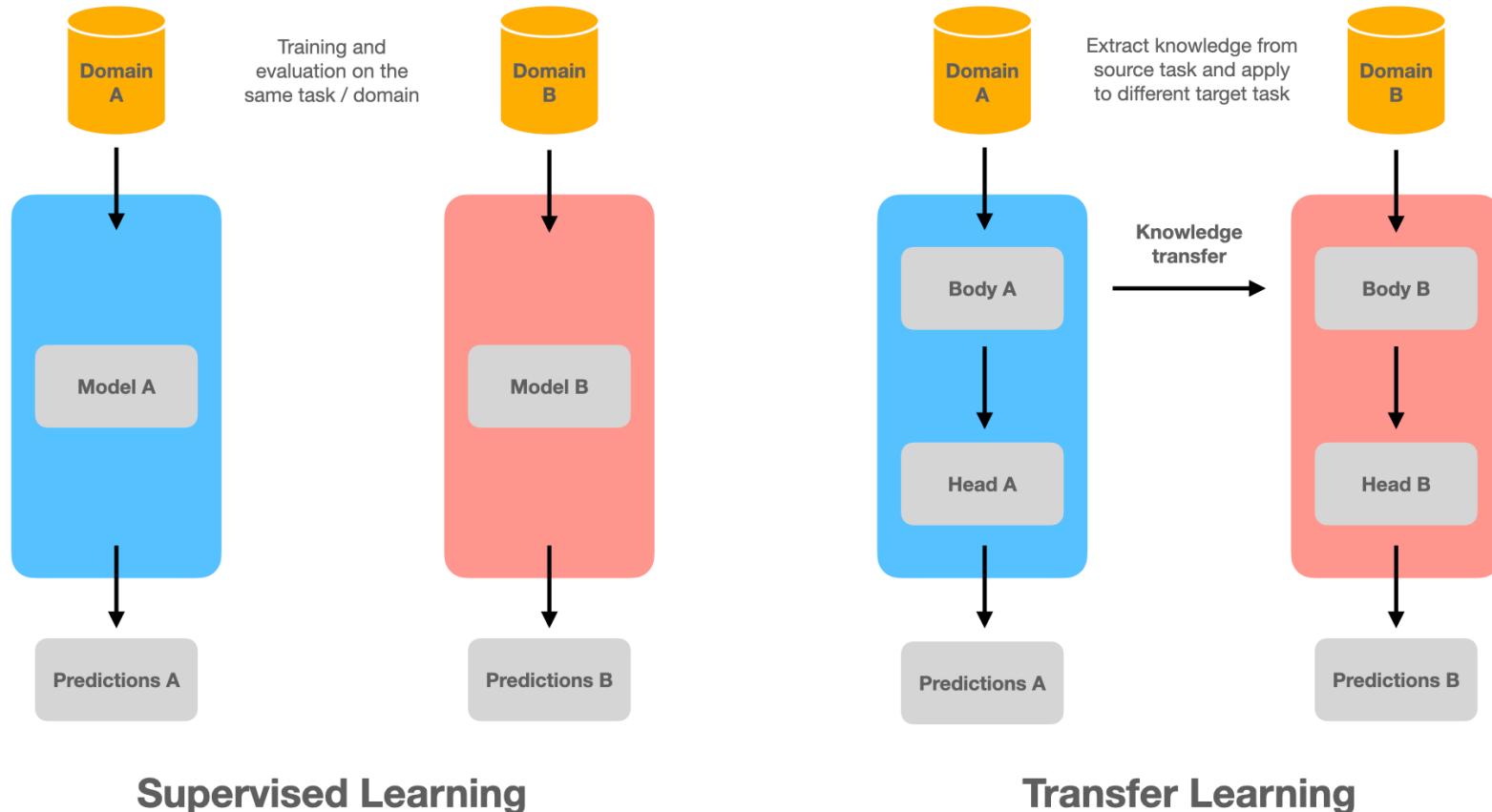


Self-Attention

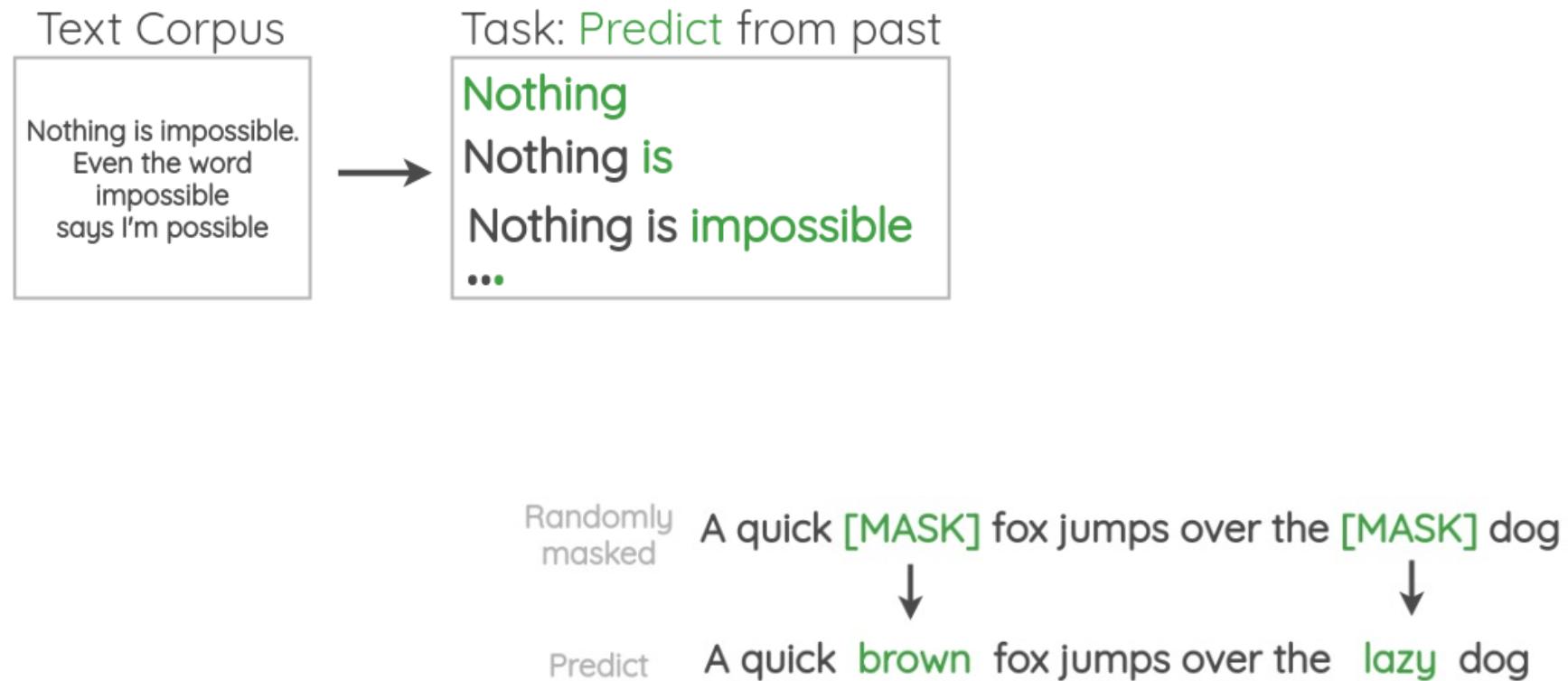


As we are encoding the word "it" in encoder #5 (the top encoder in the stack), part of the attention mechanism was focusing on "The Animal", and baked a part of its representation into the encoding of "it".

Transfer Learning



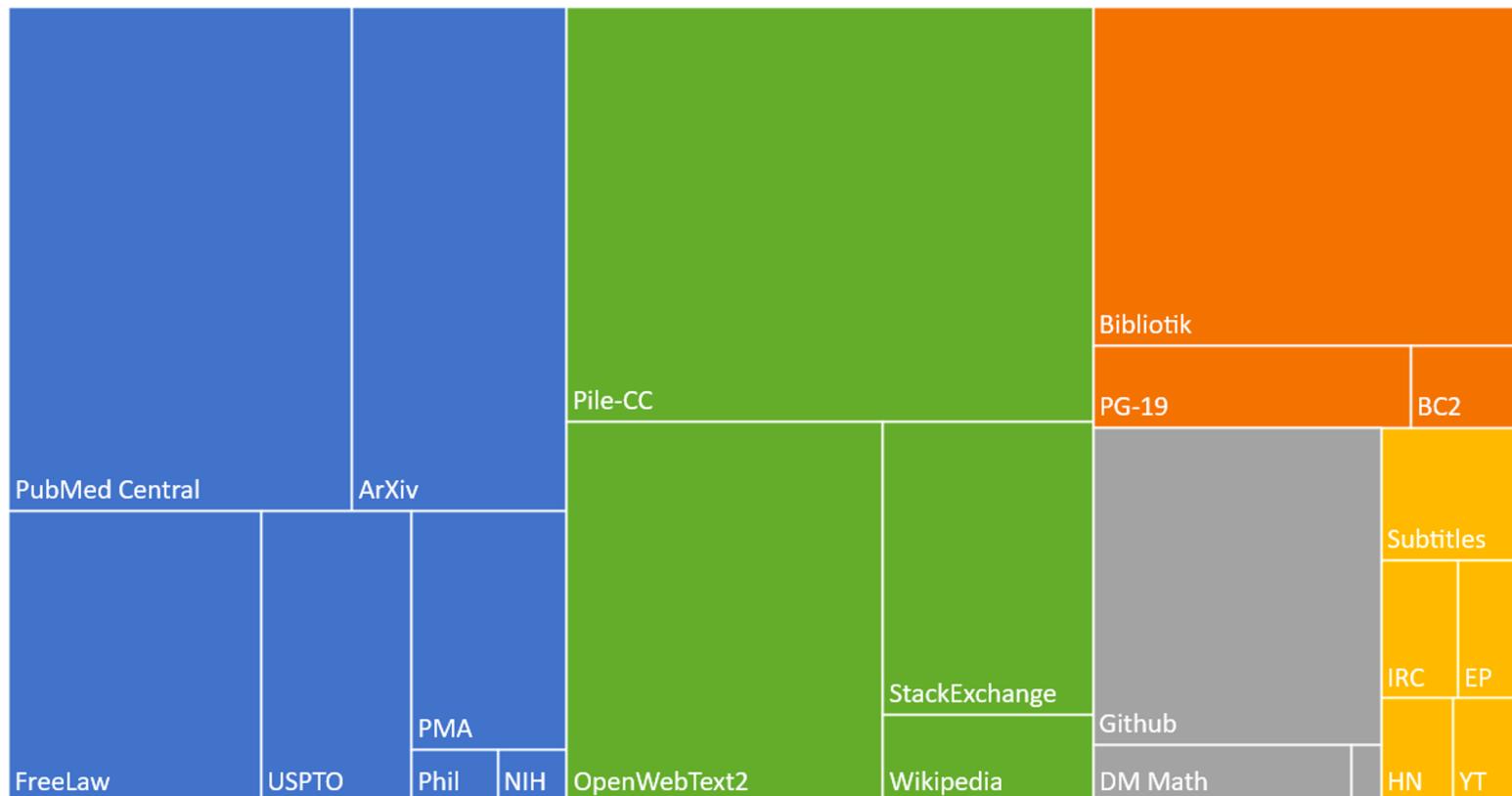
Pre-Training Transformers



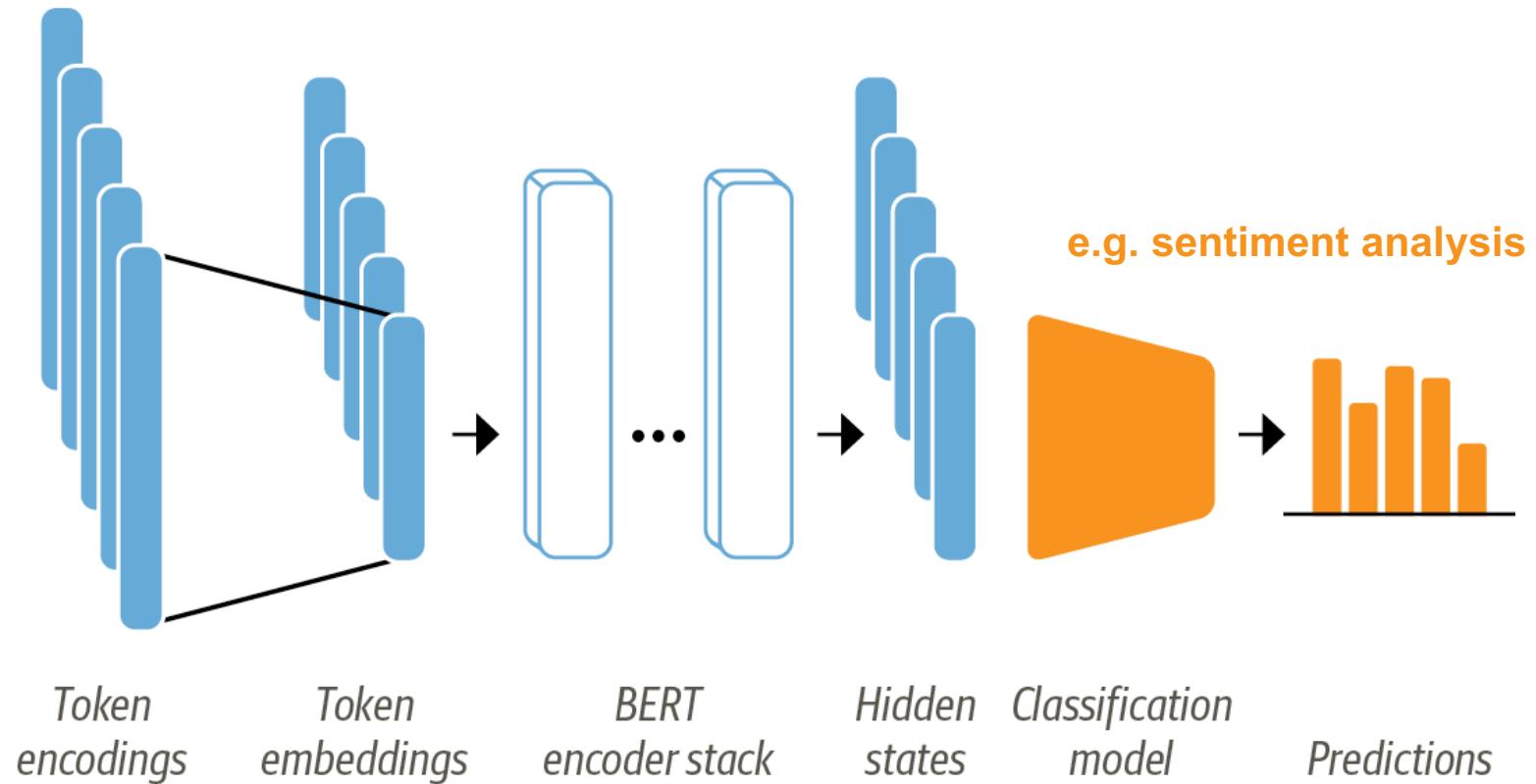
Pre-Training Transformers

Composition of the Pile by Category

■ Academic ■ Internet ■ Prose ■ Dialogue ■ Misc



Fine-tuning Transformers



Huggingface Model Hub

Hugging Face

Models 3,107 new Full-text search ↑ Sort: Trending

Task 1 Libraries Datasets Languages Licenses Other

Multimodal

- Feature Extraction Text-to-Image
- Image-to-Text Image-to-Video Text-to-Video
- Visual Question Answering
- Document Question Answering
- Graph Machine Learning Text-to-3D
- Image-to-3D

Computer Vision

- Depth Estimation Image Classification
- Object Detection Image Segmentation
- Image-to-Image Unconditional Image Generation
- Video Classification Zero-Shot Image Classification
- Mask Generation Zero-Shot Object Detection

Models 3,107 new Full-text search ↑ Sort: Trending

sentiment

 [cardiffnlp/twitter-roberta-base-sentiment-latest](#)
Text Classification • Updated May 28, 2023 • 11.6M • 312

 [cardiffnlp/twitter-xlm-roberta-base-sentiment](#)
Text Classification • Updated Jul 19, 2023 • 674k • 163

 [citizenlab/twitter-xlm-roberta-base-sentiment-finetuned](#)
Text Classification • Updated Dec 2, 2022 • 19.1k • 24

 [mrm8488/distilroberta-finetuned-financial-news-sentiment](#)
Text Classification • Updated 24 days ago • 12.1M • 175

 [finiteautomata/bertweet-base-sentiment-analysis](#)
Text Classification • Updated Feb 17, 2023 • 626k • 101

 [blanchefort/rubert-base-cased-sentiment-rusentiment](#)
Text Classification • Updated Apr 6, 2023 • 6.6k • 7

 Models Datasets Spaces Posts Docs Solutions Pricing Log In Sign Up

 [distilbert/distilbert-base-uncased-finetuned-sst-2-english](#) like 404

Text Classification Transformers PyTorch TensorFlow Rust ONNX Safetensors sst2 glue English doi:10.5796/hf0181 distilbert Eval Results Inference Endpoints arxiv:1910.01108 License: apache-2.0

Model card Files and versions Community 28 Edit model card

Downloads last month 16,520,805

 [mdraw/german-nlp](#)
Text Classification • Updated 24 days ago • 1.1M • 1.1M

 [oliverguhr/german-nlp](#)
Text Classification • Updated 24 days ago • 1.1M • 1.1M

Table of Contents

- Model Details
- How to Get Started With the Model
- Uses
- Risks, Limitations and Biases
- Training

Model Details

Model Description: This model is a fine-tune checkpoint of [DistilBERT-base-uncased](#), fine-tuned on SST-2. This model reaches an accuracy of 91.3 on the dev set (for comparison, Bert bert-base-uncased version reaches an accuracy of 92.7).

Inference API

Text Classification Examples

I like you. I love you

Compute

Computation time on cpu: cached

POSITIVE	1.000
NEGATIVE	0.000
JSON Output	
Maximize	



Syllabus

Syllabus

- **Introduction to Data Science**
 - Data Science vs. The Scientific Method
- **Supervised Learning**
 - Fundamentals of Linear Regression
 - Feature Engineering
 - Feature Selection, Regularization, and Splines
 - Logistic Regression
 - Evaluating Model Accuracy
 - Tree-based Models
 - Neural Networks
 - Ensembles and AutoML
 - Reproducibility of ML-based Research
 - Model Interpretability
- **Unsupervised Learning**
 - Clustering
 - Dimensionality Reduction
- **Natural Language Processing**
 - Dictionary-based Methods
 - Bag-of-word Models
 - Topic Models
 - Transformers
- **Data Science Mini-Project (“Hackathon”)**

Syllabus

Mini-Project (“Hackathon”)

Mini-Project (“Hackathon”)



Dataset

Dataset

Challenge

Documentation

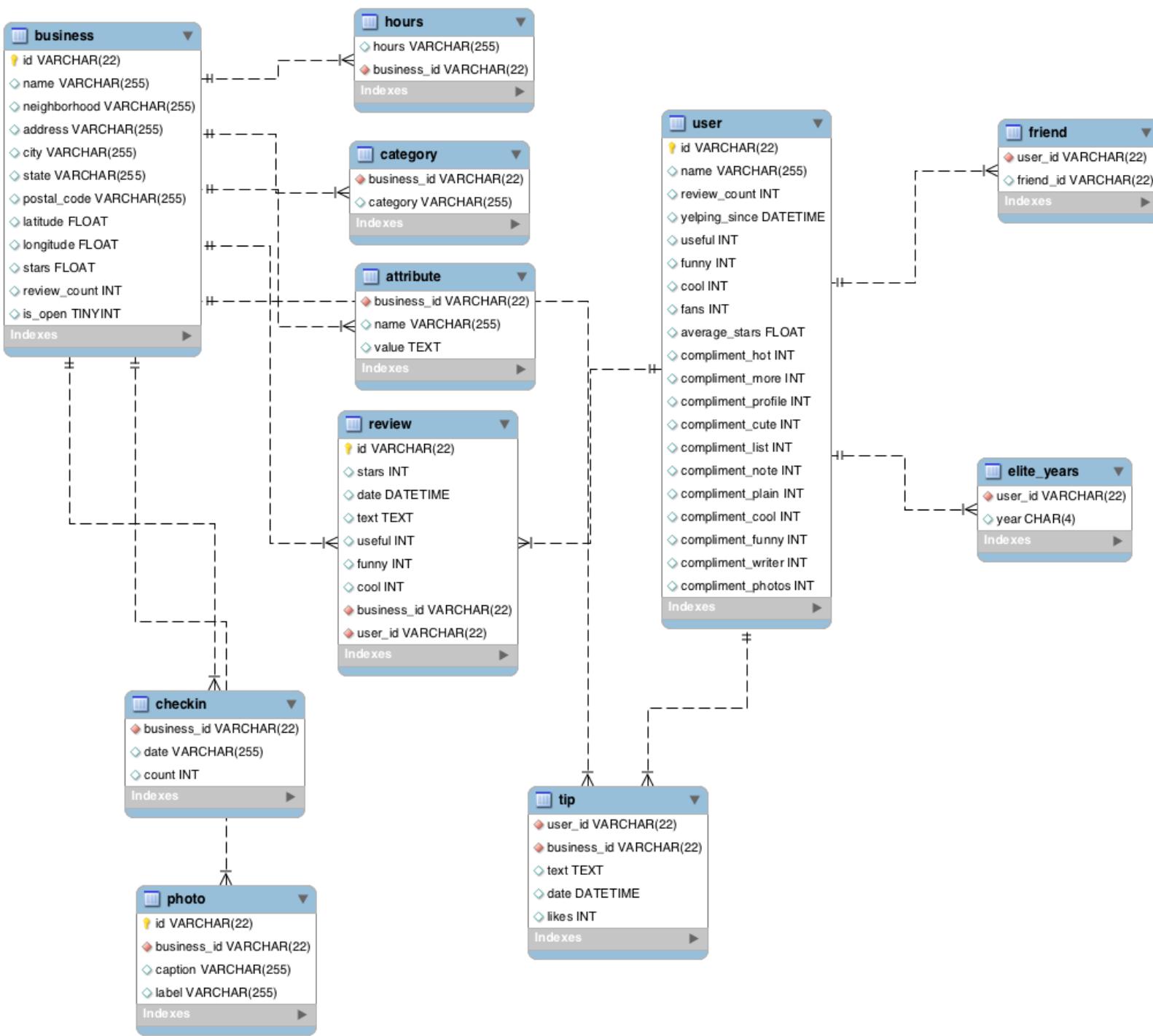
Yelp Dataset Challenge

Discover what insights lie hidden in our data.



What is the dataset challenge?

The challenge is a chance for students to conduct research or analysis on our data and share their discoveries with us. Whether you're trying to figure out how food trends start or identify the impact of different connections from the local graph, you'll have a chance to win cash prizes for your work! See some of the [past winners](#) and [hundreds of academic papers written](#) using the dataset.



The End