Day 2 – Text (Basic)

UMN CSS Workshop 2025

Instructor: Alvin Zhou

Learning Goals

- Understand the logic behind basic text analysis
- Learn core techniques: preprocessing, dictionary methods, classification
- Recognize common pitfalls
- Interpret empirical applications

Why Text? Why Now?

- Explosion of text data (social media, news, transcripts)
- Still the foundational data source
- Easy to collect, easy to analyze, and with robust communities
- Text as behavior, signal, belief, and strategy
- Social scientists want to measure, predict, and explain

Text as Data: Challenges

- High-dimensional, unstructured
  - Once you convert raw text into a format that a machine can process—usually as a vector of features—it often has a huge number of variables (dimensions) relative to the number of observations (documents).
  - Unstructured means the data is not "clean" with a pre-defined format
- Context sensitivity (e.g., irony, sarcasm)
- Requires transformation into features: **preprocessing** so that we can process text as data

Foundational Workflow - Tokenization

- Tokenization: the process of breaking down the text
- Raw text → Tokens → Features → Models → Interpretation
- Instead of working with text, text-as-data works with the matrix
- text = "Cats are cute." → ["Cats", "are", "cute"]

	it	is	puppy	cat	pen	a	this
it is a puppy	1	1	1	0	0	1	0
it is a kitten	1	1	0	0	0	1	0
it is a cat	1	1	0	1	0	1	0
that is a dog and this is a pen	0	2	0	0	1	2	1
it is a matrix	1	1	0	0	0	1	0

Text Preprocessing 101

- Lowercasing, punctuation removal, etc.
- Tokenization: words vs. subwords
  - Words: "I love backpacking!" → ["I", "love", "backpacking"]
    - Simplicity, speed, but can't handle rare/new words
  - Subwords: "backpacking" → ["back", "##pack", "##ing"]
    - Flexibility, efficiency, but harder to interpret
- Stopword removal
  - "the" "a" "so" "then" etc
  - Context-dependent, you can designate new stopwords that are of no relevance to your research questions
- Lemmatization vs. stemming
  - Stemming: chop off word endings with no mercy, fast but crude
    - running → run studies → studi universities → univers
  - Lemmatization: dictionary + grammar to find the root
    - running → run studies → study better → good

n-Grams

- Unigrams, bigrams, trigrams
- Why use n-grams:
  - Capture adjacent context (e.g., “not good”)
- Tradeoff: complexity vs. sparsity
- Most methods we discuss use unigram.

From Text to Matrix

- Document-Term Matrix (DTM)
  - Sparse, high-dimensional

	words1	words2	words3	words4	words5
doc1	0	0	1	0	0
doc2	2	0	1	1	0
doc3	0	0	1	1	0
doc4	0	0	1	1	1

- TF (Term Frequency, frequency of a word in a document): In the sentence "Data is the new oil", TF for "data" is 1/5 = 0.2
- IDF (Inverse Document Frequency, how unique is a word across documents): Words that appear in many documents (like “the”, “is”) get low IDF, while unique words get higher IDF
- TF-IDF: Putting it Together

TF-IDF

- Term Frequency × Inverse Document Frequency
- High **TF-IDF** = word is frequent in a document **but rare in the corpus**
- Filters out common “filler” words
- Retains contextually meaningful, distinctive terms

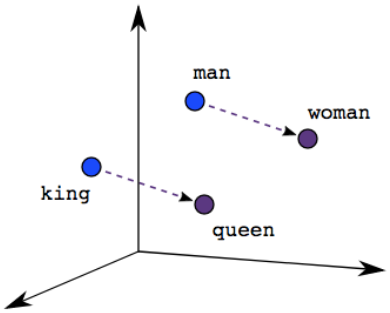
Term	**Doc A TF**	**IDF**	**TF-IDF**
data	0.2	1.5	0.3
the	0.2	0.2	0.04
oil	0.2	2.0	0.4

TF-IDF vs. Word Embeddings

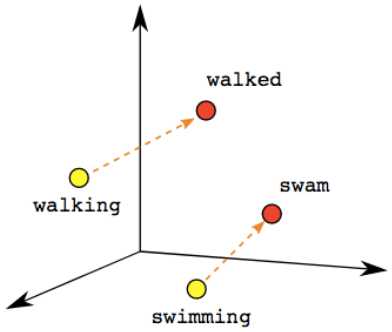
- TF-IDF and word embeddings are two different ways to represent text as numbers.
- TF-IDF is only relevant for classical machine learning models , such as logistic regression and SVM, where we used to construct our own text classification models.
- More recently, the field has been shifting towards AI-assisted models for text classification, where embedding is more prominent, a topic we will discuss in the future.
- However, understanding how text-as-data works is crucial and foundational to gaining a comprehensive understanding of social data science.

Word Embeddings

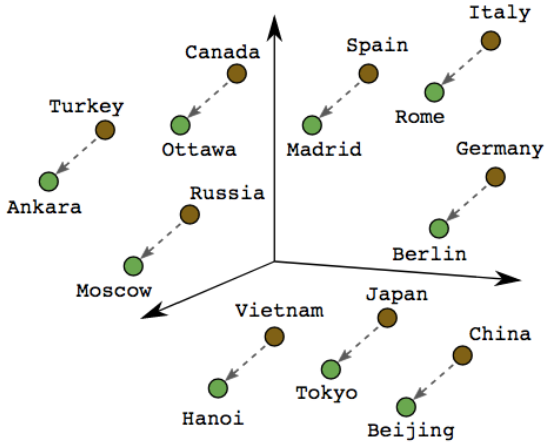
- Each word = low-dimensional dense vector (e.g., 300 dimensions)
  - TF-IDF is high-dimensional (many columns, one column per word)
- Learned from context in large corpora (e.g., Word2Vec, GloVe, BERT)
- Captures semantic relationships
  - e.g., vec(“king”) – vec(“man”) + vec(“woman”) ≈ vec(“queen”)



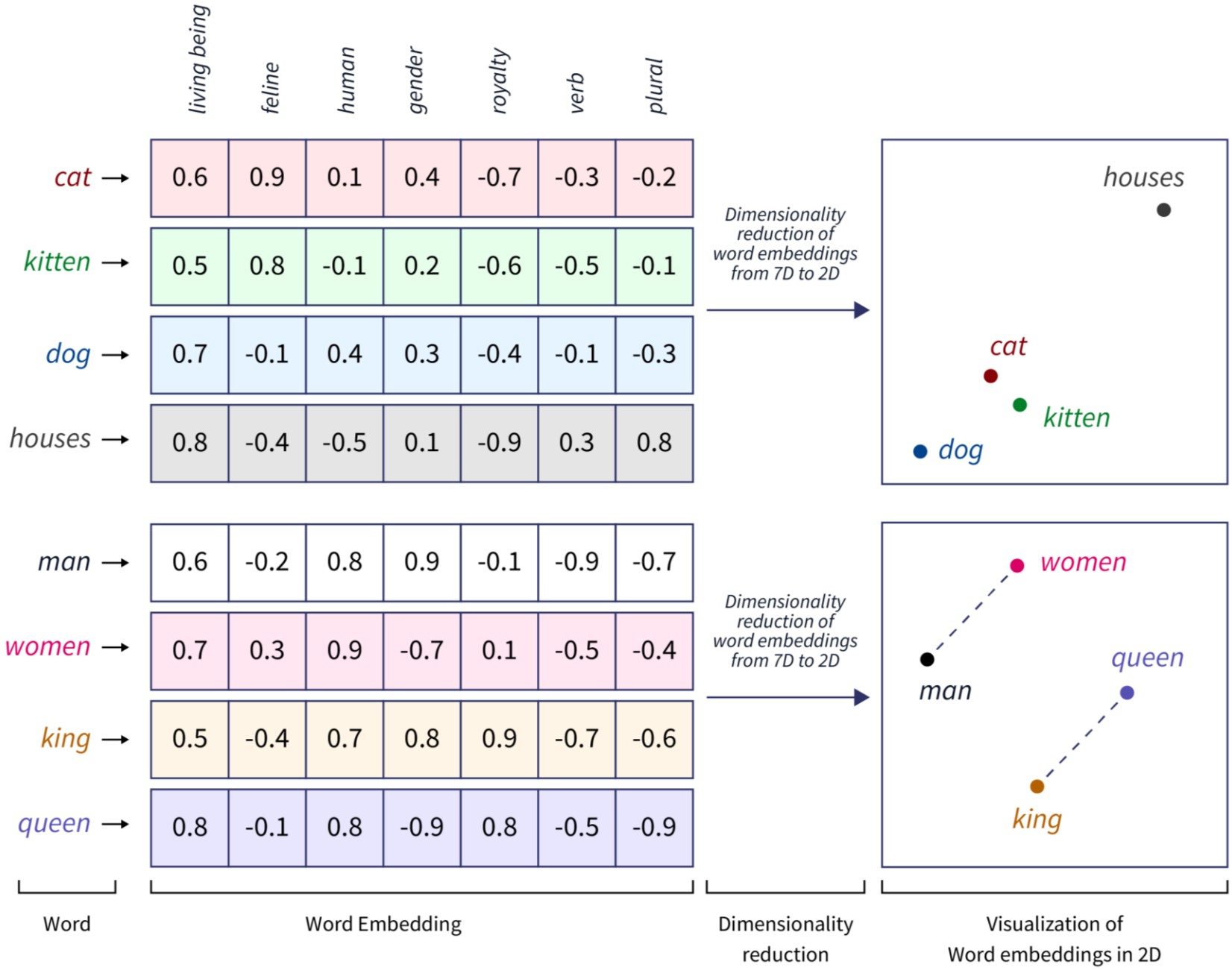
Male-Female



Verb Tense



Country-Capital


living being
feline
human
gender
royalty
verb
plural
cat → 0.6 0.9 0.1 0.4 -0.7 -0.3 -0.2
kitten → 0.5 0.8 -0.1 0.2 -0.6 -0.5 -0.1
dog → 0.7 -0.1 0.4 0.3 -0.4 -0.1 -0.3
houses → 0.8 -0.4 -0.5 0.1 -0.9 0.3 0.8
man → 0.6 -0.2 0.8 0.9 -0.1 -0.9 -0.7
women → 0.7 0.3 0.9 -0.7 0.1 -0.5 -0.4
king → 0.5 -0.4 0.7 0.8 0.9 -0.7 -0.6
queen → 0.8 -0.1 0.8 -0.9 0.8 -0.5 -0.9
Dimensionality reduction of word embeddings from 7D to 2D
Dimensionality reduction of word embeddings from 7D to 2D
houses
cat
kitten
dog
women
queen
man
king
Word
Word Embedding
Dimensionality reduction
Visualization of Word embeddings in 2D

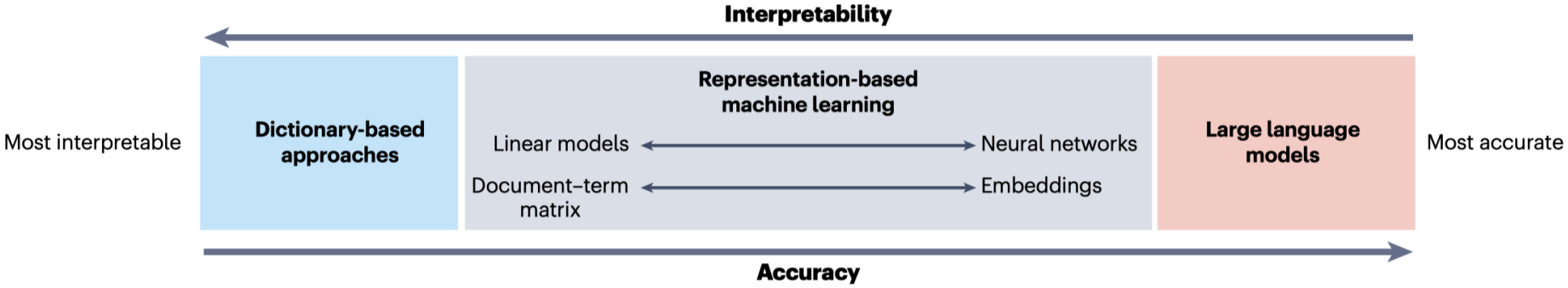
Word Embeddings

Feature	TF-IDF	Embedding
Based on	Term frequency + rarity	Context in language
Dimension	Large (10k+)	Small (e.g., 300)
Output	Sparse vector	Dense vector
Learns meaning?	No	Yes
Used in deep learning?	Rarely	Yes

- If you're building a model, you'll choose:
  - **TF-IDF** if you're doing classic ML (e.g., logistic regression, SVM)
    - logistic regression, SVM, LIWC, sentiment, moral foundations
  - **Embeddings** if you're doing modern NLP (e.g., transformers, neural nets)
    - Many off-the-shelf tools to choose from
  - **TF-IDF dominated social science text-as-data work before ~2020 and embedding-based approaches have become more common since then**, particularly as tools like Word2Vec, GloVe, BERT, and GPT became more accessible and pre-trained models more widely available.

Dictionary-Based Methods

- Most interpretable
  - The most accurate model might not be what social scientists value
- Dictionary: predefined word lists
- Examples: LIWC, MFD, Lexicoder
- Applications: sentiment, moral values, word use



Case: LIWC

- Tausczik & Pennebaker (2010)
  - Tausczik, Y. R., & Pennebaker, J. W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology, 29*(1), 24–54. https://doi.org/10.1177/0261927X09351676
- Count words that correspond to a feature (e.g., anxiety, anger)

Case: LIWC

Affective or emotional processes	**Abbrev (affect)**	**Examples (happy, ugly, bitter)**	**# Words 615**
Positive emotions	Posemo	happy, pretty, good	261
Positive feelings	Posfeel	happy, joy, love	43
Optimism and energy	Optim	certainty, pride, win	69
Negative emotions	Negmo	hate, worthless, enemy	345
Anxiety or fear	Anx	nervous, afraid, tense	62
Anger	Anger	hate, kill, pissed	121
Sadness or depression	Sad	grief, cry, sad	72
Time	**Abbrev (time)**	**Examples (hour, day, clock)**	**# Words 113**
Past tense verb	Past	walked, were, had	144
Present tense verb	Present	walk, is, be	256
Future tense verb	Future	will, might, shall	14
Leisure activity	**Abbrev (leisure)**	**Examples (house, TV, music)**	**# Words 113**
Home	Home	house, kitchen, lawn	26
Sports	Sports	football, game, play	28
Television and movies	TV	TV, sitcom, cinema	19
Music	Music	tunes, song, CD	31

Case: LIWC

- For example:
  - Processing tweets published by people living in
    - Regions with rainfall
    - Regions with sunshine
  - and measuring tweets' positive emotions (the "Posemo" variable)
  - We might see that tweets geo-matched with sunshine regions have a higher Posemo value

  - Let's say you ran LIWC on a document with 1,000 words, and you got Posemo = 76
  - That means:
    - 76 words (7.6% of the total) matched words in the "positive emotion" category, such as "happy," "love," "great," "joy," etc.
  - It doesn't account for context, negation, or sentence structure.

Case: Moral Foundations Theory

- Wang & Inbar (2021)
  - Wang, S.-Y. N., & Inbar, Y. (2021). Moral-language use by U.S. political elites. *Psychological Science, 32*(1), 14–26. https://doi.org/10.1177/0956797620960397
- Five moral foundations across culture

Foundation	**Basic Intuition**	**Virtues**	**Violations**
Care/Harm	Protect others, prevent suffering	Compassion, kindness	Cruelty, aggression
Fairness/Cheating	Justice, equality, reciprocity	Fairness, justice	Cheating, exploitation
Loyalty/Betrayal	Group identity and cohesion	Patriotism, self-sacrifice	Disloyalty, treason
Authority/Subversion	Respect for tradition, hierarchy	Obedience, deference	Disrespect, rebellion
Sanctity/Degradation	Purity, sacredness	Cleanliness, piety	Impurity, perversion

Case: Different Dictionaries

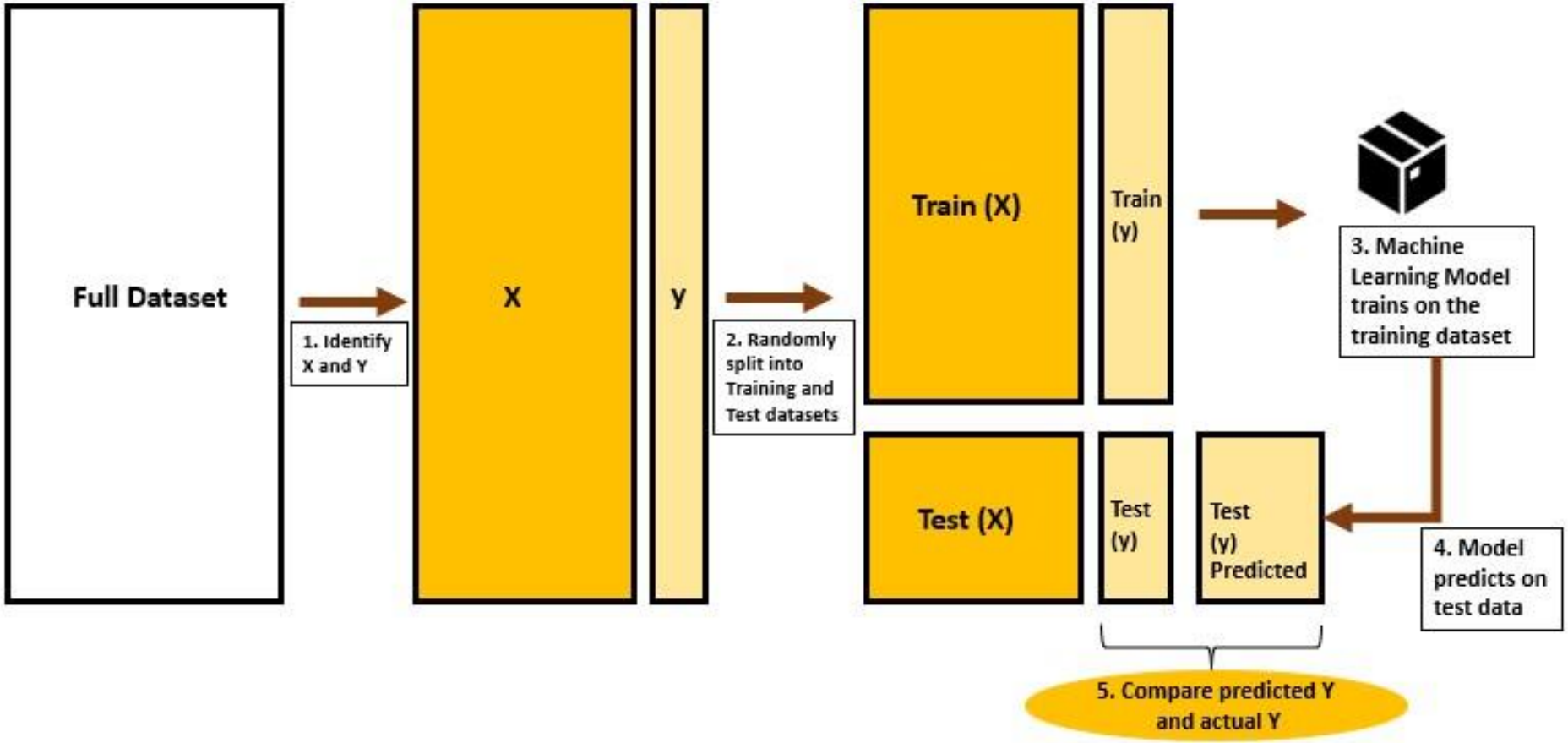
- Graham, J., Haidt, J., & Nosek, B. A. (2009). Liberals and conservatives rely on different sets of moral foundations. *Journal of Personality and Social Psychology*, *96*(5), 1029–1046. https://doi.org/10.1037/a0015141
- Frimer, J. (2019). *Moral foundations dictionary 2.0*. https://osf.io/ezn37/
- Hopp, F. R., Fisher, J. T., Cornell, D., Huskey, R., & Weber, R. (2021). The extended Moral Foundations Dictionary (eMFD): Development and applications of a crowd-sourced approach to extracting moral intuitions from text. *Behavior Research Methods*, *53*(1), 232–246. https://doi.org/10.3758/s13428-020-01433-0
- **[Embedding]** Duan, Z., Shao, A., Hu, Y., Lee, H., Liao, X., Suh, Y. J., Kim, J., Yang, K.-C., Chen, K., & Yang, S. (2025). Constructing vectionaries to extract message features from texts: A case study of moral content. *Political Analysis*, 1–21. https://doi.org/10.1017/pan.2025.6

Limitations of Dictionaries

- Lack of context sensitivity
- Vocabulary drift/misspecification
- Not designed for predictive accuracy
  - Dictionaries (e.g., LIWC, MFD, Lexicoder) are used to measure constructs like anxiety, anger, authority, fairness, etc.
  - Valuable for theory-driven analysis — especially when interpretability matters.
  - Not built for predictive accuracy, e.g.,
    - Accuracy
    - F1-score
    - AUC
  - Dictionary scores are rarely used to predict:
    - Whether someone votes
    - Who wins an election
    - Whether someone is depressed

Supervised Machine Learning

- Goal: predict a label (e.g., sentiment, topic, ideology)
- Requires labeled data
- Features: n-grams, TF-IDF, dictionaries
- Classifier models:
  - Logistic regression
    - L1 (Lasso) Regularization
    - L2 (Ridge) Regularization
    - Elastic Net – hybrid of L1 and L2
  - Support Vector Machine (SVM)
  - Tree-Based Models
  - Neural Networks (e.g., Recurrent Neural Network (RNN))
  - K-Nearest Neighbors (KNN)
  - Etc.

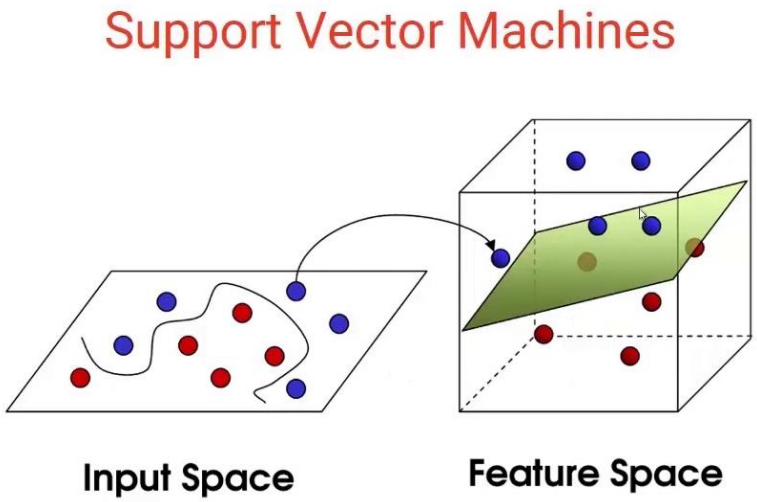

Full Dataset
1. Identify X and Y
X
y
2. Randomly split into Training and Test datasets
Train (X)
Train (y)
Test (X)
Test (y)
Test (y) Predicted
3. Machine Learning Model trains on the training dataset
4. Model predicts on test data
5. Compare predicted Y and actual Y

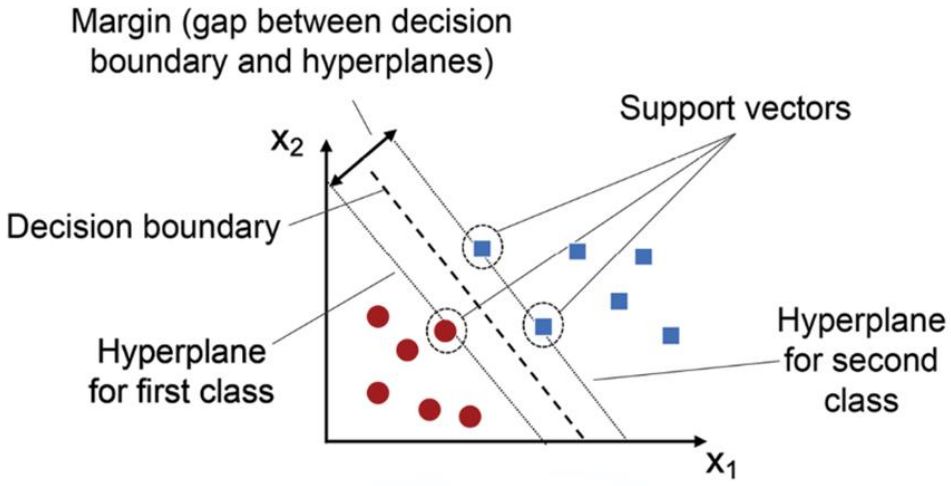
Logistic Regression

- Binary outcome: logit(p) = β0 + β1X
  - Whether_uncivil = β0 + β1* “asshole” + β2* “beautiful” + …
- Interpretable coefficients
- Use in social science: classification + inference
- Remember, in TF-IDF, there are many dimensions, other than “asshole” “beautiful” there are thousands of n-grams that could have coefficients, therefore, we need regularization, which shrinks coefficients to improve generalization to new data.
  - **L1 (Lasso) Regularization**
    - Encourages sparsity: sets some coefficients to exactly zero
    - Good for feature selection when many irrelevant features
  - **L2 (Ridge) Regularization**
    - Shrinks coefficients toward zero but keeps all features
    - Good for handling multicollinearity and overfitting
  - **Elastic Net – hybrid of L1 and L2**

SVM

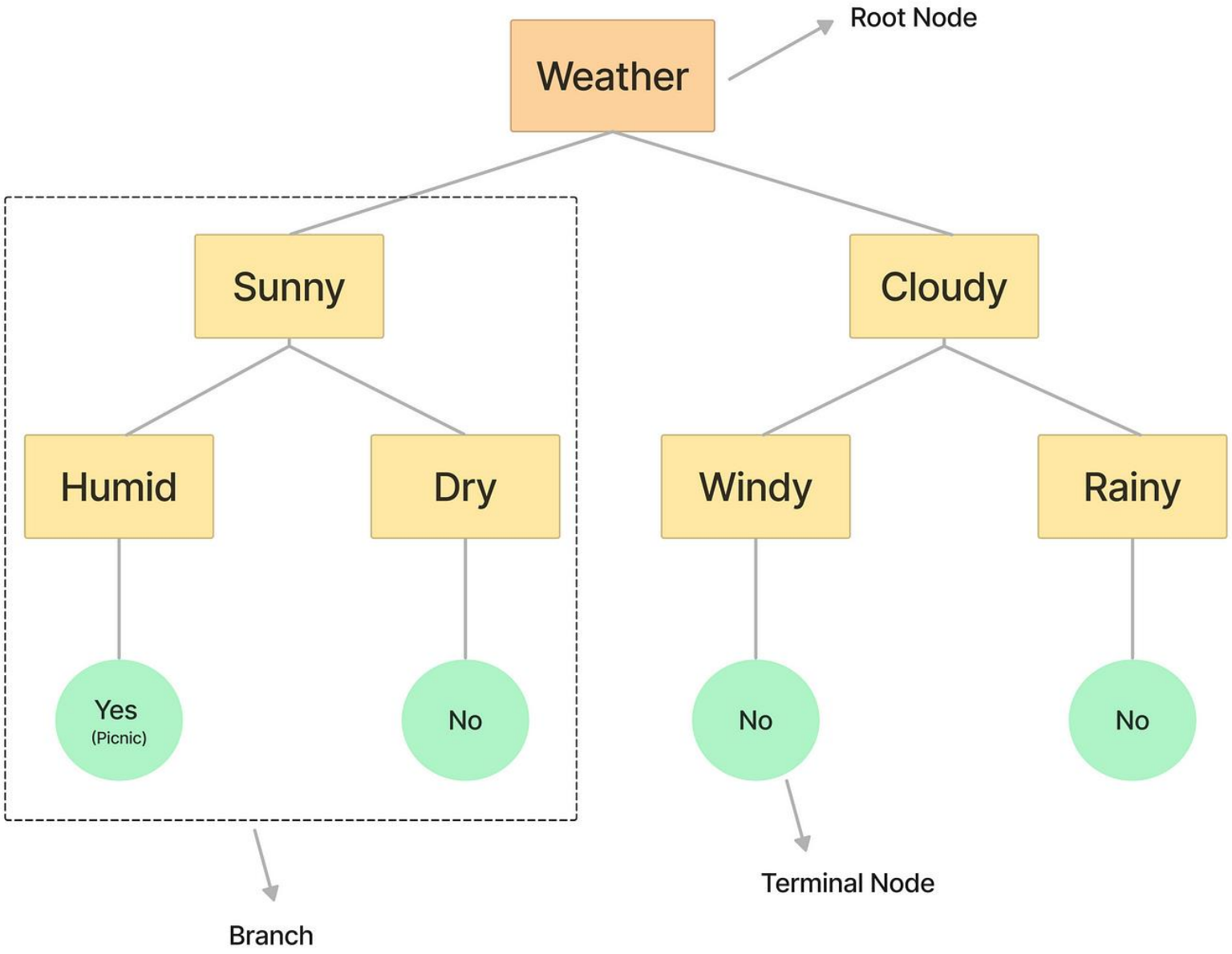
- Margin maximization
- Good when many features, few observations
- Less interpretable than logistic regression





Tree-Based Models

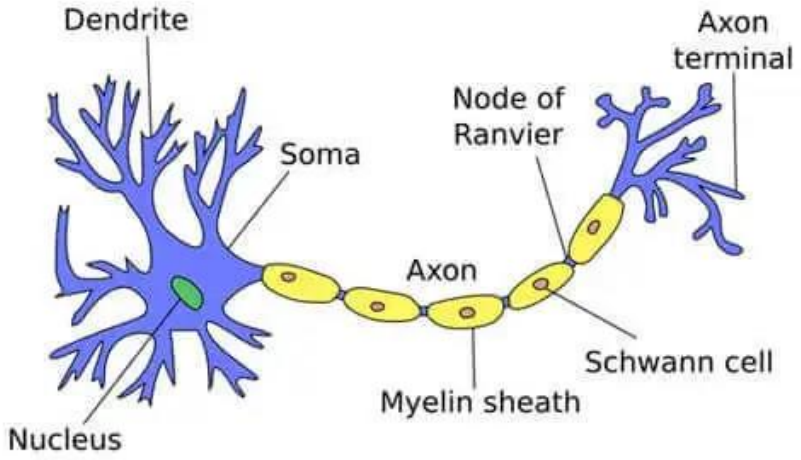
- Decision Tree (a.k.a. Classification Tree)
- Random Forest – ensemble of trees, more stable
- Gradient Boosted Trees (e.g., XGBoost, LightGBM, CatBoost)
- Etc.

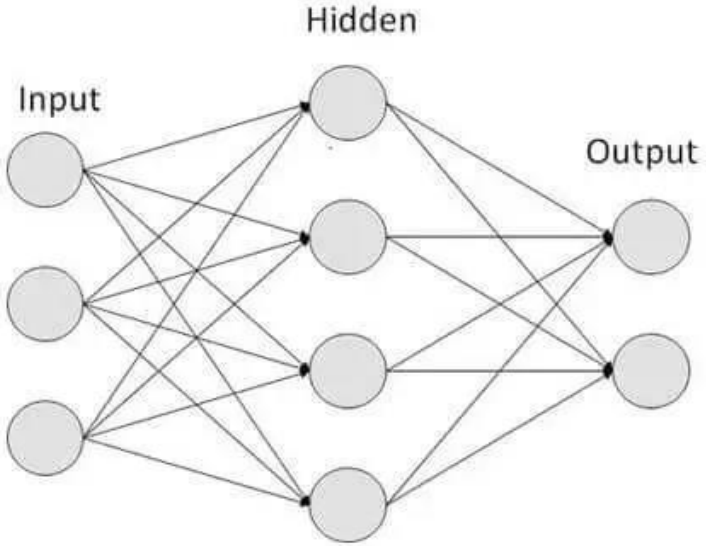

Root Node
Weather
Sunny
Cloudy
Humid
Dry
Windy
Rainy
Yes
(Picnic)
No
No
No
Terminal Node
Branch

Neural Networks

- Loosely inspired by the brain: layers of nodes ("neurons")
- Used for nonlinear classification and complex pattern recognition
- input → hidden layers → output
- Text input needs to be embedded first (e.g., with word vectors)
  - Handles high-dimensional input well
  - Can model interactions and nonlinearity
  - Requires tuning, more data
  - Often a black box (hard to interpret)

Neural Networks



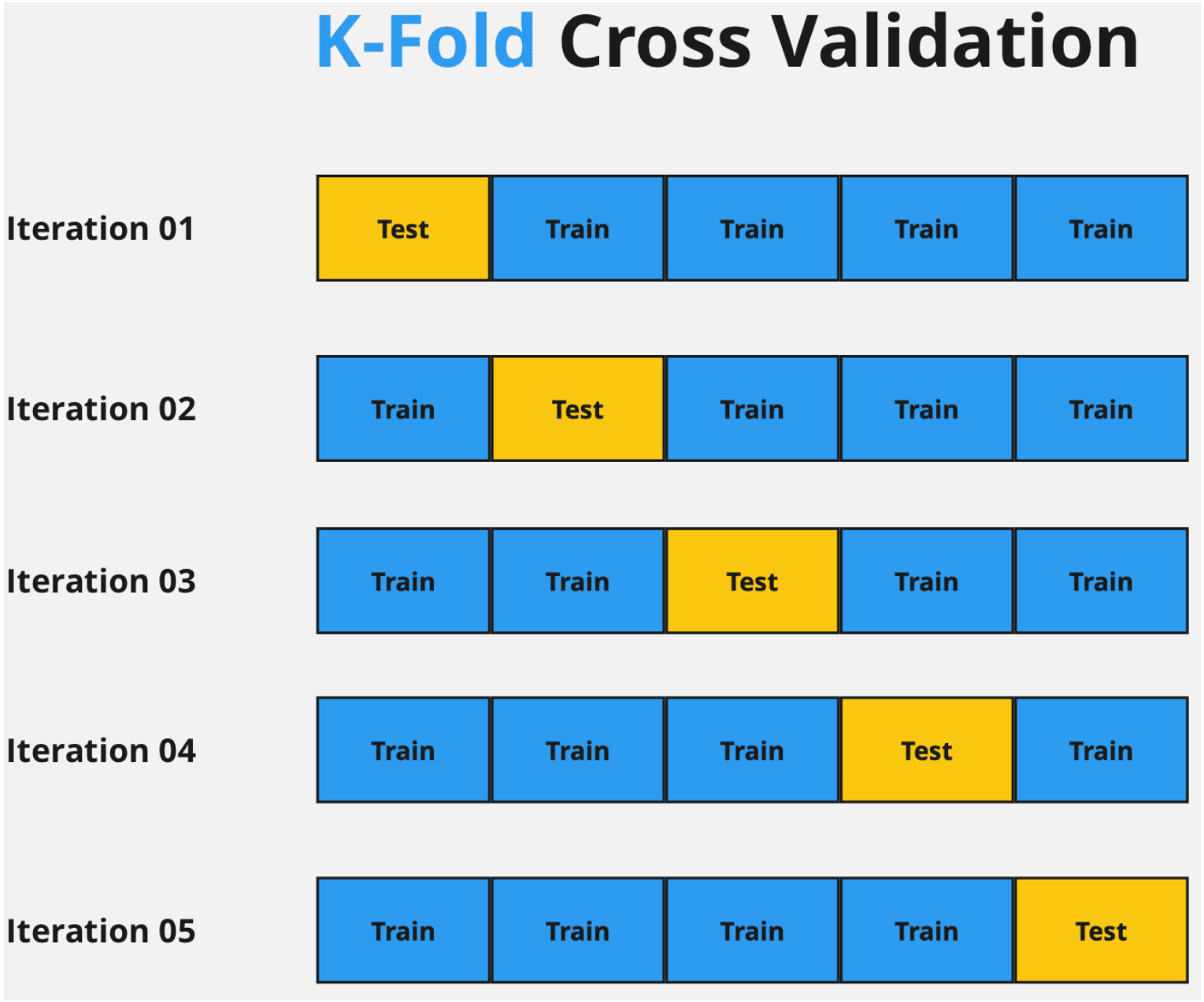


How we evaluate model

- Cross-validation (k-fold):
  - Partition the dataset into ‘k’ subsets
  - Iteratively using each subset as the test set while the remaining subsets form the training set
  - This process helps in mitigating overfitting (prevents overfitting to one split of the data) and provides a more generalized evaluation that could work better for unseen/new data.

- Some people also do “Holdout Validation (Train/Test Split)”
  - Split once into training and testing sets (e.g., 80/20)
  - Fast and simple, but results may vary depending on the random split
  - Use with large datasets or when training time is high

K-Fold Cross Validation



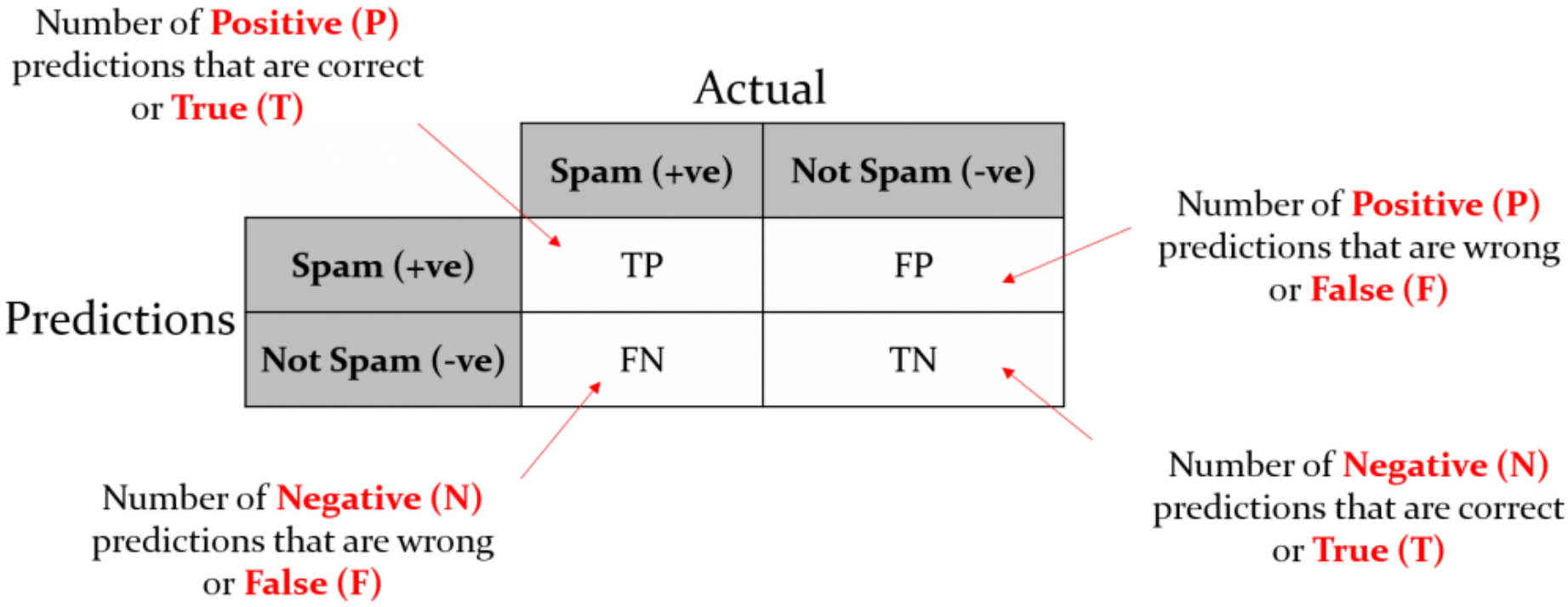
Evaluation Metrics

- Accuracy, Precision, Recall, F1-score
  - **Accuracy:** (TP + TN) / (TP + FP + TN + FN)
  - **Precision**: TP / (TP + FP)
  - **Recall (Sensitivity)**: TP / (TP + FN)
  - **F1-Score:** 2 * (Precision * Recall) / (Precision + Recall)

Metric	Rule of Thumb	Notes
Accuracy	> 80% is solid	But misleading if classes are imbalanced
Precision	> 70% is usable; > 90% is excellent	High precision = few false positives
Recall	> 70% is usable; > 90% is excellent	High recall = few false negatives
F1-score	> 75% is good; > 85% is strong	Balanced tradeoff between precision and recall

Evaluation Metrics

- Accuracy, Precision, Recall, F1-score

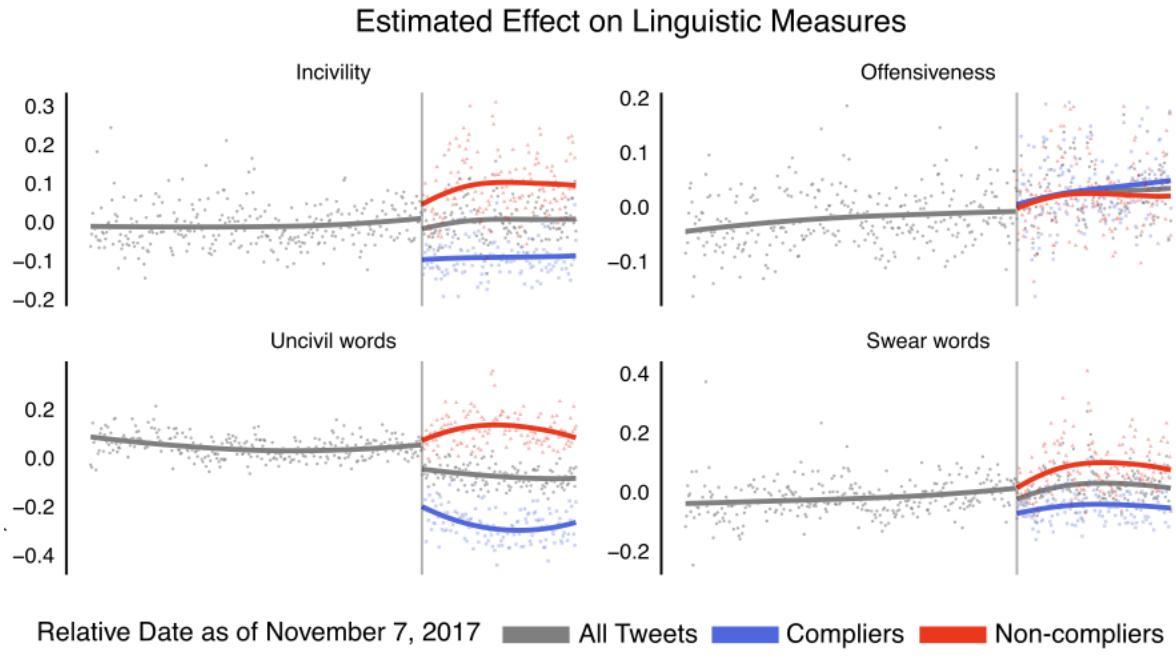


Evaluation Metrics

- You run 5-fold cross-validation on a sentiment classifier. You get F1-scores of:
  - Fold 1: 0.74
  - Fold 2: 0.77
  - Fold 3: 0.76
  - Fold 4: 0.75
  - Fold 5: 0.73
- The final reported F1-score is the mean: 0.75

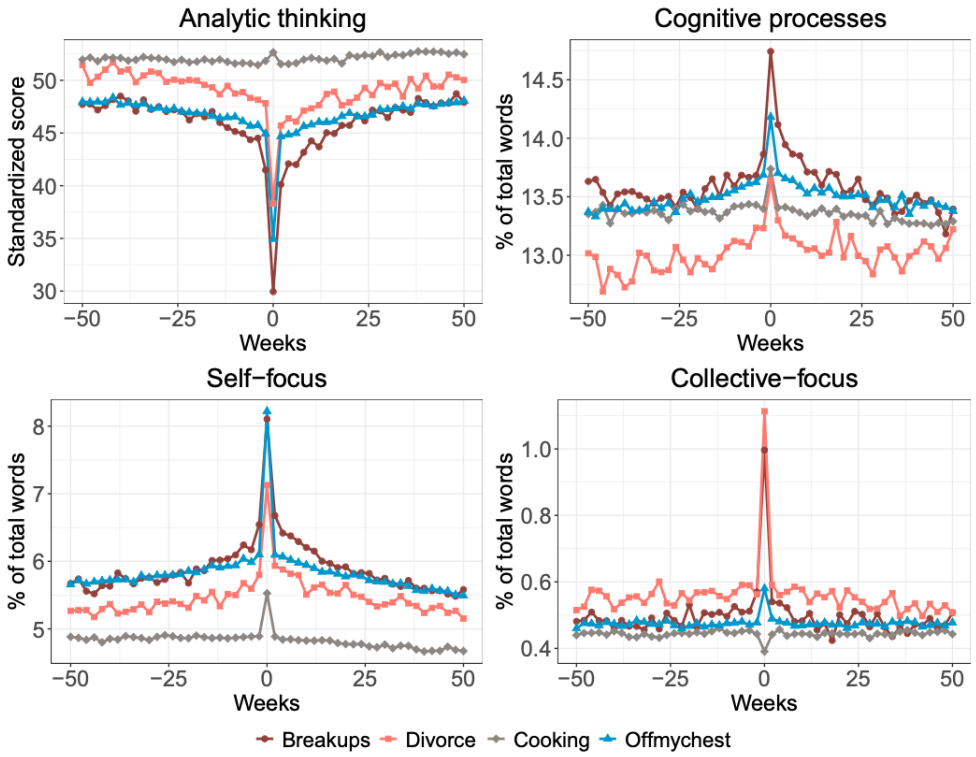
Case: Jaidka et al. (2019)

- Classify tweets by communication variables (e.g., incivility)
- Variable from the tweets: length, hashtags, sentiment, etc.
- Logistic regression with feature selection (L1)



Case: Seraj et al. (2021)

- Predict breakup distress from text
- Time-series of emotion words pre/post breakup
- High face validity + useful visualization



Optional: Zero-Shot + Pretrained Models

- If you don’t have labeled data… GPT, BERT, etc.
- Not for today, we will talk about GPT and LLM later
- Caveats: hallucination, black box

Lab Preview

- LIWC
- Sentiment
- Moral Foundations
- **Google Perspective**
  - Go to https://www.perspectiveapi.com/
  - Apply for the API
  - It takes a few hours to get this API access
  - We will use it in the next lab